

ANALYSIS ON VOTING DATA USING DEDUPLICATION TECHNIQUES IN CLOUD

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

by

Ch. Sandeep Kumar
(Reg. No. 38110111)



**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 GANDHI SALAI,
CHENNAI -600 119**

MARCH - 2022



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



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project report is the bonafide work of **Ch. SANDEEP KUMAR (Reg.No. 38110111)** project who carried out the project entitled **"ANALYSIS ON VOTING DATA USING DEDUPLICATION TECHNIQUES IN CLOUD"** under my supervision from November 2021 to March 2022.

Internal Guide

Dr. T. ANANDHI, M.E., Ph.D.,

Head of the Department

Dr. S. Vigneswari, M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **SANDEEP KUMAR** hereby declare that the Project Report entitled “**ANALYSIS ON VOTING DATA USING DEDUPLICATION TECHNIQUES IN CLOUD**” is done by me under the guidance of **Dr T. ANANDHI, M.E.,Ph.D.**, Department of Computer Science and Engineering at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

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

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

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

Abstract

Data deduplication is one of important data compression techniques for eliminating duplicate copies of repeating data, and has been widely used in cloud storage to reduce the amount of storage space and save bandwidth. To protect the confidentiality of sensitive data while supporting deduplication, the convergent encryption technique has been proposed to encrypt the data before outsourcing. To better protect data security, this project makes the first attempt to formally address the problem of authorized data deduplication. Different from traditional deduplication systems, the differential privileges of users are further considered in duplicate check besides the data itself. We also present several new deduplication constructions supporting authorized duplicate check in hybrid cloud architecture. Security analysis demonstrates that our scheme is secure in terms of the definitions specified in the proposed security model. As a proof of concept, the proposed work implements a prototype of our proposed authorized duplicate check scheme and conduct test bed experiments using our prototype. The proposed work shows that our proposed authorized duplicate check scheme incurs minimal overhead compared to normal operations.

TABLE OF CONTENTS

CHAPTER No	TITLE	PAGE No
	ABSTRACT	Vi
	LIST OF FIGURES	Viii
1.	INTRODUCTION	1
	1.1 OVERVIEW	1
2.	LITERATURE SURVEY	3
3.	SYSTEM DESIGN	6
	3.1 SCOPE OF THE PROJECT	6
	3.2 OBJECTIVE	6
	3.3 PROBLEM DEFINITION	6
	3.4 EXISTING SYSTEM	6
	3.5 PROPOSED SYSTEM	7
4.	METHODS & ALGORITHMS USED	9
	4.1 HARDWARE REQUIREMENTS	9
	4.2 SOFTWARE REQUIREMENTS	9
	4.3 SYSTEM ARCHITECTURE	19
	4.4 MODULE OVERVIEW	20
	4.5 PROPESED ALGORITHM	22
5.	SYSTEM IMPLEMENTATION	25

LIST OF FIGURES

FIG No	FIGURE NAME	PAGE No
4.1	SYSTEM ARCHITECTURE	20
4.2	USER MODULE	20
4.3	SERVER START UP& UPLOAD FILE	21
4.4	SECURING DEDUPLICATION SYSTEM	21
4.5	DOWNLOADED FILE	22
5.1	DATA FLOW DIAGRAM	25
6.1	DATA SET	38
6.2	INTERFACE	39
6.3	REGISTRATION	40
6.4	LOGIN	41
6.5	FILE UPLOAD	42
6.6	VIEW UPLOAD FILES	43
6.7	SELECT THE FILE	44
6.8	GET THE APPLY KEY	45
6.9	USER LOGIN	46
6.10	COPY THE APPLY KEY	47
6.11	FINAL DOWNLOADED FILE	48

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Cloud computing provides seemingly unlimited “virtualized” resources to users as services across the whole Internet, while hiding platform and implementation details. Today’s cloud service providers offer both highly available storage and massively parallel computing resources at relatively low costs. As cloud computing becomes prevalent, an increasing amount of data is being stored in the cloud and shared by users with specified *privileges*, which define the access rights of the stored data. One critical challenge of cloud storage services is the management of the ever-increasing volume of data. To make data management scalable in cloud computing, de duplication has been a well-known technique and has attracted more and more attention recently.

Data de duplication is a specialized data compression technique for eliminating duplicate copies of repeating data in storage. The technique is used to improve storage utilization and can also be applied to network data transfers to reduce the number of bytes that must be sent. Instead of keeping multiple data copies with the same content, deduplication eliminates redundant data by keeping only one physical copy and referring other redundant data to that copy. De duplication can take place at either the file level or the block level. For filelevel de duplication, it eliminates duplicate copies of the same file. De duplication can also take place at the block level, which eliminates duplicate blocks of data that occur in non-identical files. Although data deduplication brings a lot of benefits, security and privacy concerns arise as users’ sensitive data are susceptible to both insider and outsider attacks. Traditional encryption, while providing data confidentiality, is incompatible with data deduplication.

Specifically, traditional encryption requires different users to encrypt their data with their own keys. Thus, identical data copies of different users will lead to different cipher texts, making de duplication impossible. Convergent encryption has been proposed to enforce data confidentiality while making de duplication feasible.

It encrypts/decrypts a data copy with a convergent key, which is obtained by computing the cryptographic hash value of the content of the data copy. After key generation and data encryption, users retain the keys and send the cipher text to the cloud. Since the encryption operation is Deterministic and is derived from the data content, identical data copies will generate the same convergent key and hence the same cipher text. To prevent unauthorized access, a secure proof of ownership protocol is also needed to provide the proof that the user indeed owns the same file when a duplicate is found. After the proof, subsequent users with the same file will be provided a pointer from the server without needing to upload the same file. A user can download the encrypted file with the pointer from the server, which can only be decrypted by the corresponding data owners with their convergent keys.

Thus, convergent encryption allows the cloud to perform de duplication on the cipher texts and the proof of ownership prevents the unauthorized user to access the file. However, previous de duplication systems cannot support differential authorization duplicate check, which is important in many applications. In such an authorized de duplication system, each user is issued a set of privileges during system initialization. Each file uploaded to the cloud is also bounded by a set of privileges to specify which kind of users is allowed to perform the duplicate check and access the files. Before submitting his duplicate check request for some file, the user needs to take this file and his own privileges as inputs. The user is able to find a duplicate for this file if and only if there is a copy of this file and a matched privilege stored in cloud. For example, in a company, many different privileges will be assigned to employees.

In order to save cost and efficiently management, the data will be moved to the storage server provider (SSP) in the public cloud with specified privileges and the de duplication technique will be applied to store only one copy of the same file. Because of privacy consideration, some files will be encrypted and allowed the duplicate check by employees with specified privileges to realize the access control. Traditional de duplication systems based on convergent encryption, although providing confidentiality to some extent; do not support the duplicate check with differential privileges. In other words, no differential privileges have been considered in the de duplication based on convergent encryption technique.

CHAPTER 2

LITERATURE SURVEY

2. LITERATURE SURVEY:

J.Johin, R.K. Rohith, R.S.S. Yukesh Kumar, G. Paavi [1] in the paper on **“Deduplication in Cloud for Improvising Efficiency Towards Potential Practical Usage”** proposed that if the duplicate of files is available on the cloud server, the application needs to map the original files to the current file that is uploaded so that user will not get affected by the mapping but on backend no duplicate files will be created but the files get mapped by the application, In this way the duplication files on the cloud server will be avoided thereby reducing the storage space requirement on the cloud server.

N. Amutha¹, Dr. S. Sujatha [2] in the paper on **“Secure Data De-Duplication on 2PVC Algorithm in Web Services”**. In this they propose several increasingly stringent levels of policy consistency constraints, and present different enforcement approaches to guarantee the trustworthiness of transactions executing on web services. Intelligibility can emerge when utilizing a situation dependent on the option to utilize a cloud-based capacity framework and to secure weak resources.

Nandhana Kuttappan [3] in her paper on **“Data Deduplication in Cloud Storage with and Ownership Management”** in this article De-Duplication method is very effective when multiple users storing the same data in outsource. This time relating security and ownership issues. The proof-of-ownership schemes allow any owner of the same data to prove to the cloud storage server that he owns the data in a robust way, Then many users are likely to encrypt their data before outsourcing them to the cloud storage to preserve privacy, but this hampers de duplication because of the randomization property of encryption.

Ashutosh Avadhani , Amruta Chaudhari, Ankit Powar, Ishwar Borse [4] in the paper on **“Database Management and Storage Optimization Using Duplication”** This system is designed to be low cost and expandable. The aim of this is to build an android application which will store only unique contents and de-

uplicated files from the server which can be accessed from anywhere.

Dr. MK Jayanthi, P. Sri Vaibhavi, P.V. Naga Sathya [5] in the paper on **“Achieved Efficient Data Deduplication and Key Aggregation”** in this the project implemented a scenario where the cloud service can deduplicate the upload data. The data was prevented from unwanted exposure and unauthorized access by placing a proper access control mechanism. Authorized data deduplication aims at data security to keep the data aims at data security to keep the data secured and avoid unauthorized access. Deduplication at encryption level saves a lot of memory and memory can be utilized efficiently.

F. Bonomi, R. Milito, J. Zhu, and S. Addepalli [6] in the paper **“Fog Computing and its Role in the Internet of Things”** As a result of this the Fog must support several types of storage, from ephemeral at the lowest tier to semi-permanent at the highest tier. We also note that the higher the tier, the wider the geographical coverage is provided by the cloud, which is used as repository for data that has a permeance of months and years and which is the bases for business intelligence analytics. This is the typical HMI environment of reports and dashboards the display key performance.

R. Atan, A.M.Talib and M.A.A. Murad [7] in his paper on **“Formulating a Security Layer of Cloud Data Storage Framework Based on Multi Agent System”** had proposed that we investigated the problem of data security in cloud data storage, to ensure the correctness of users data in cloud data storage; we proposed a security framework and MAS architecture to facilitate security of cloud data storage. This security framework consists of two main layers as agent layer and cloud data storage later. The propose MAS architecture includes five types in order to facilitate the huge amount of security, our MAS architecture offered eleven security attributes generated from four main security policies of correctness integrity, confidentiality and availability of user’s data in the cloud.

Y. Li, T Wang, G. Wang, J. Liang and H. Chen [8] in the paper on **“Efficient data Collection in Sensor Cloud System Using Multiple Mobile Sinks”** in this paper the results shown that our algorithm can uploaded the data from WSNs to

cloud with the limited latency and minimized the energy consumption as well. The outcome shows that our calculation can communicate information from WSN to the cloud in a brief time frame and lessen power utilization. Distributed computing further develops information handling and wireless sensor networks(WSNs).

CHAPTER 3

SYSTEM DESIGN

3.1 SCOPE OF THE PROJECT

Data de duplication techniques are widely employed to backup data and minimize network and storage overhead by detecting and eliminating and redundancy among data.

3.2 OBJECTIVE

The main goal is to enable de duplication and distributed storage of the data across multiple storage servers.

3.3 PROBLEM DEFINITION

The existing system only performs the de duplication either on block level or file level. It does not provide very high security needed for the message to be transmitted. Due to this the third party or hacker may find the data that is being transmitted between the users.

3.4 EXISTING SYSTEM

Data de duplication systems, the private cloud are involved as a proxy to allow data owner/users to securely perform duplicate check with differential privileges. Such architecture is practical and has attracted much attention from researchers. The data owners only outsource their data storage by utilizing public cloud while the data operation is managed in private cloud.

Data de duplication is a specialized data compression technique for eliminating duplicate copies of repeating data in storage. The technique is used to improve storage utilization and can also be applied to network data transfers to reduce the number of bytes that must be sent. Instead of keeping multiple data copies with the same content, de duplication eliminates redundant data by keeping only one physical copy and referring other redundant data to that copy.

De duplication can take place at either the file level or the block level. For file level de duplication, it eliminates duplicate copies of the same file. De duplication can also take place at the block level, which eliminates duplicate blocks of data that occur in non-identical files. Identical data copies of different users will lead to different cipher texts, making de duplication impossible.

3.4.1 DISADVANTAGES

- Traditional encryption, while providing data confidentiality, is incompatible with data de duplication.
- Identical data copies of different users will lead to different cipher texts, making de duplication impossible.

3.5 PROPOSED SYSTEM

In this proposed work, the system enhanced with security. Specifically, it present an advanced scheme to support stronger security by encrypting the file with differential privilege keys. In this way, the users without corresponding privileges cannot perform the duplicate check. Furthermore, such unauthorized users cannot decrypt the cipher text even collude with the S-CSP. Security analysis demonstrates that our system is secure in terms of the definitions specified in proposed security.

Convergent encryption has been proposed to enforce data confidentiality while making de duplication feasible. It encrypts/decrypts a data copy with a convergent key, which is obtained by computing the cryptographic hash value of the content of the data copy. After key generation and data encryption, users retain the keys and send the cipher text to the cloud. Since the encryption operation is deterministic and is derived from the data content, identical data copies will generate the same convergent key and hence the same cipher text. To prevent unauthorized access, a secure proof of ownership protocol is also needed to provide the proof that the user indeed owns the same file when a duplicate is found.

3.5.1 ADVANTAGES

- The user is only allowed to perform the duplicate check for files marked with the corresponding privileges.
- We present an advanced scheme to support stronger security by encrypting the file with differential privilege keys.
- Reduce the storage size of the tags for integrity check. To enhance the security of de duplication and protect the data confidentiality.

CHAPTER 4

METHODS AND ALGORITHMS USED

4.1 HARDWARE REQUIREMENT

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 it also shows how it should be implemented. It specifies the speed of the system that should be used in this project. The hardware requirement for this project is mentioned below.

System	: Pentium IV 2.4 GHz
Hard Disk	: 40 GB
Floppy Drive	: 44 Mb
Monitor	: 15 VGA Colour
Ram	: 8 GB

4.2 SOFTWARE REQUIREMENT

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the team and tracking the team's progress throughout the development activity. The software requirement specifies the application software that is being used in the project. The software needed to develop this project is mentioned below.

Operating system	: Windows 10
IDE	: Visual Studio
Coding Language	: Java

INTRODUCTION

Java is one of the world's most important and widely used computer languages, and it has held this distinction for many years. Unlike some other computer languages whose influence has wearied with passage of time, while Java's has grown.

APPLICATION OF JAVA

Java is widely used in every corner of world and of human life. Java is not only used in software's but is also widely used in designing hardware controlling software components. There are more than 930 million JRE downloads each year and 3 billion mobile phones run java.

Following are some other usage of Java:

- Developing Desktop Applications
- Web Applications like Linkedin.com, Snapdeal.com etc
- Mobile Operating System like Android
- Embedded Systems
- Robotics and games etc.

FEATURES OF JAVA

The prime reason behind creation of Java was to bring portability and security feature into a computer language. Beside these two major features, there were many other features that played an important role in moulding out the final form of this outstanding language. Those features are;

- **Simple**

Java is easy to learn and its syntax is quite simple, clean and easy to understand. The confusing and ambiguous concepts of C++ are either left out in Java or they have been re-implemented in a cleaner way.

Eg: Pointers and Operator Overloading are not there in java but were an important part of C++.

- **Object-Oriented**

In java everything is Object which has some data and behaviour. Java can be easily extended as it is based on Object Model.

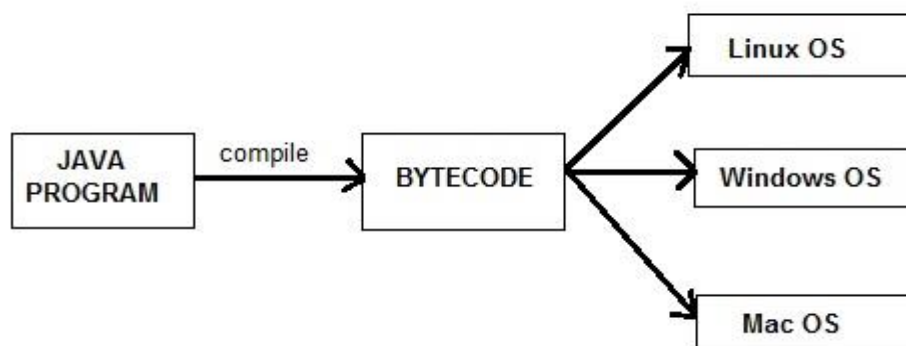
3) Robust

Java makes an effort to eliminate error prone codes by emphasizing mainly on compile time error checking and runtime checking. But the main areas which Java improved were Memory Management and mishandled Exceptions by introducing automatic Garbage Collector and Exception Handling.

4) Platform Independent

Unlike other programming languages such as C, C++ etc. which are compiled into platform specific machines. Java is guaranteed to be write-once, run-anywhere language.

On compilation Java program is compiled into byte code. This byte code is platform independent and can be run on any machine, plus this byte code format also provide security. Any machine with Java Runtime Environment can run Java Programs.



5) SECURE

When it comes to security, Java is always the first choice. With java secure features it enable us to develop virus free, temper free system. Java program always runs in Java runtime environment with almost null interaction with system OS, hence it is more secure.

6) MULTI-THREADING

Java multithreading feature makes it possible to write program that can do many tasks simultaneously. Benefit of multithreading is that it utilizes same memory and other resources to execute multiple threads at the same time, like While typing, grammatical errors are checked along.

7) ARCHITECTURAL NEUTRAL

Compiler generates byte codes, which have nothing to do with a particular computer architecture, hence a Java program is easy to interpret on any machine.

- **PORTABLE**

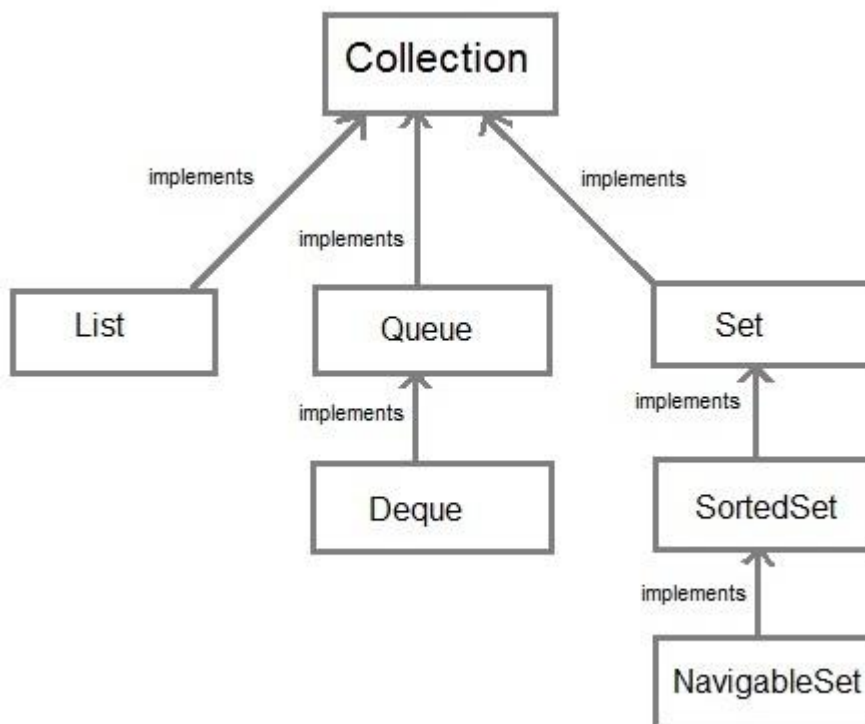
Java Byte code can be carried to any platform. No implementation dependent features. Everything related to storage is predefined, example: size of primitive data types.

- **HIGH PERFORMANCE**

Java is an interpreted language, so it will never be as fast as a compiled language like C or C++. But, Java enables high performance with the use of just-in-time compiler.

- **COLLECTION FRAMEWORK**

The collection framework was not part of original Java release. Collections was added to J2SE 1.2. Prior to Java 2, Java provided adhoc classes such as Dictionary, Vector, Stack and Properties to store and manipulate groups of objects. Collection framework provides many important classes and interfaces to collect and organize group of alike objects.



MYSQL

MySQL, officially, but also called "My Sequel" is the world's most widely used open-source relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases, though SQLite probably has more total embedded deployments. The SQL phrase stands for Structured Query Language.

The MySQL development project has made its source code available under

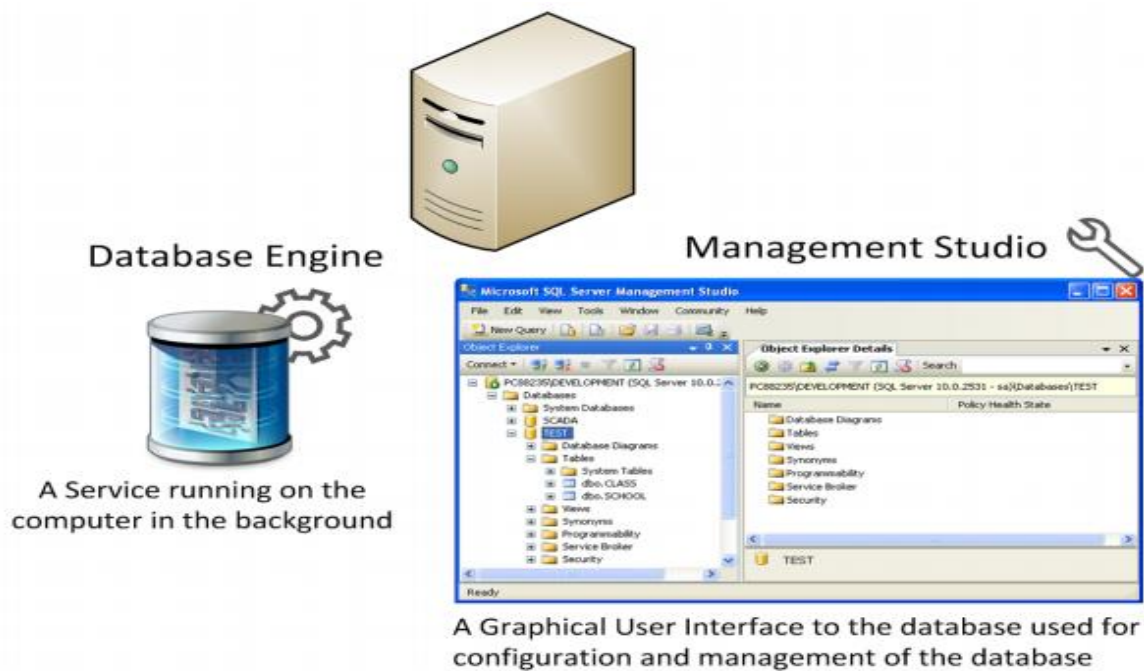
the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks).

LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Freesoftware-open source projects that require a full-featured database management system often use MySQL.

For commercial use, several paid editions are available, and offer additional functionality. Applications which use MySQL databases include: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal and other software. MySQL is also used in many high-profile, large-scale websites, including Wikipedia, Google (though not for searches), Facebook, Twitter, Flickr and YouTube. Microsoft is the vendor of SQL Server. We have different editions of SQL Server, where SQL Server Express is free to download and use. SQL Server uses T-SQL (Transact-SQL). T-SQL is Microsoft's proprietary extension to SQL. TSQL is very similar to standard SQL, but in addition it supports some extra functionality, built in functions, etc.

T-SQL expands on the SQL standard to include procedural programming, local variables, and various support functions for string processing, date processing, mathematics, etc. SQL Server consists of a Database Engine and a Management Studio (and lots of other stuff which we will not mention here). The Database engine has no graphical interface - it is just a service running in the background of your computer (preferable on the server). The Management Studio is graphical tool for configuring and viewing the information in the database.

SQL Server Management Studio can also be used to create a new database, alter any existing database schema by adding or modifying tables and indexes, or analyze performance. It includes the query windows which provide a GUI based interface to write and execute queries.



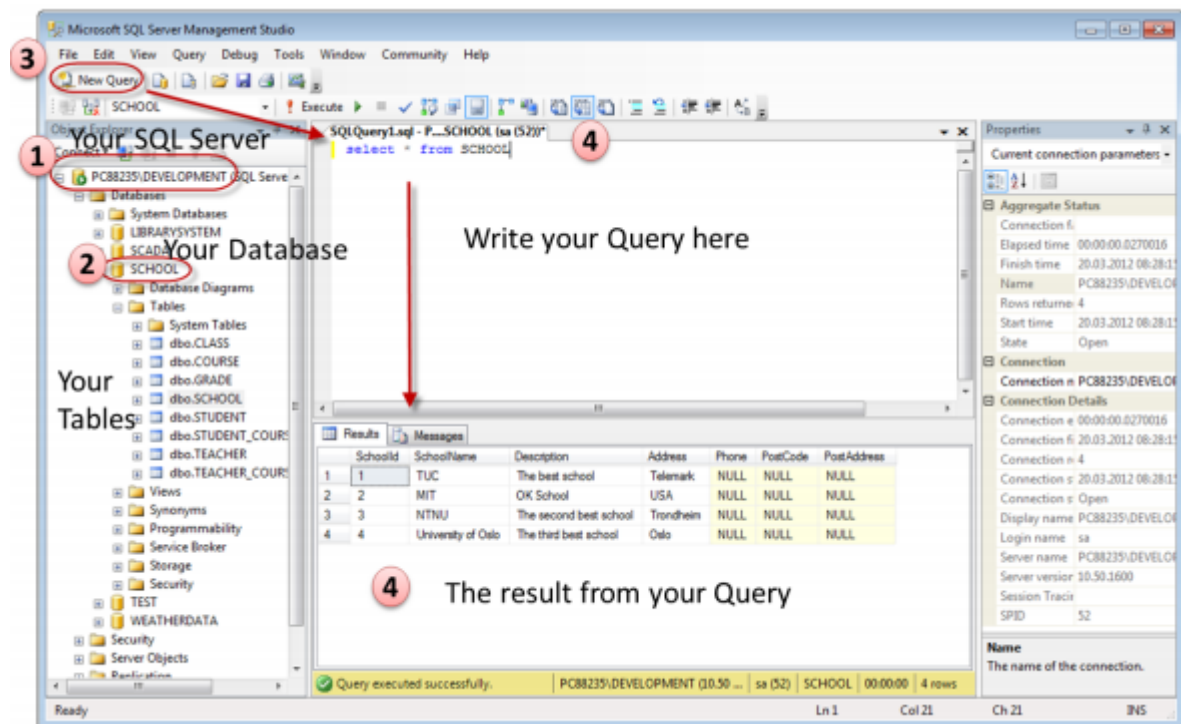
4.3.1 SQL SERVER MANAGEMENT STUDIO

SQL Server Management Studio is a GUI tool included with SQL Server for configuring, managing, and administering all components within Microsoft SQL Server. The tool includes both script editors and graphical tools that work with objects and features of the server. As mentioned earlier, version of SQL Server Management Studio is also available for SQL Server Express Edition, for which it is known as SQL Server Management Studio Express.

A central feature of SQL Server Management Studio is the Object Explorer, which allows the user to browse, select, and act upon any of the objects within the server. It can be used to visually observe and analyze query plans and optimize the database performance, among others. SQL Server Management Studio can also be used to create a new database, alter any existing database schema by adding or modifying tables and indexes, or analyze performance. It includes the query windows which provide a GUI based interface to write and execute queries.

When creating SQL commands and queries, the “Query Editor” (select “New Query” from the Toolbar) is used (shown in the figure above). With SQL and the “Query Editor” we can do almost everything with code, but sometimes it is also a good idea to use the different Designer tools in SQL to help us do the work without

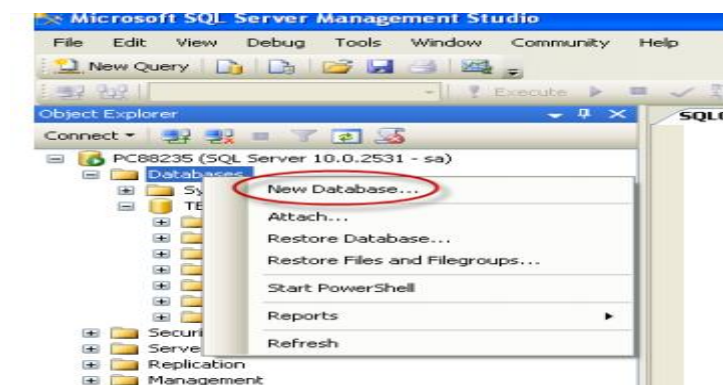
coding (so much).



SQL SERVER MANAGEMENT STUDIO

4.3.2 CREATE A NEW DATABASE

It is quite simple to create a new database in Microsoft SQL Server. Just right-click on the “Databases” node and select “New Database...”



CREATE A NEW DATABASE

The OLAP Services feature available in SQL Server version 7.0 is now called SQL Server 2000 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component. The Repository component available in SQL Server version 7.0 is now called Microsoft SQL Server 2000 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Services

SQL-SERVER database consist of six type of objects,
They are,

- TABLE
- QUERY
- FORM
- REPORT
- MACRO

- **Table**

A database is a collection of data about a specific topic.

- **Views of Table**

We can work with a table in two types,

1. **Design View**

2. **Datasheet View**

1. **Design View**

To build or modify the structure of a table we work in the table design view.
We can specify what kind of data will be hold.

2. **Datasheet View**

To add, edit or analyses the data itself we work in tables datasheet view mode.

Query

A query is a question that has to be asked the data. Access gathers data that answers the question from one or more table. The data that make up the answer is either dynaset (if you edit it) or a snapshot (it cannot be edited). Each time we run query, we get latest information in the dynaset. Access either displays the dynaset or snapshot for us to view or perform an action on it, such as deleting or updating.

FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

The feasibility study investigates the problem and the information needs of the stakeholders. It seeks to determine the resources required to provide an information systems solution, the cost and benefits of such a solution, and the feasibility of such a solution.

The goal of the feasibility study is to consider alternative information systems solutions, evaluate their feasibility, and propose the alternative most suitable to the organization. The feasibility of a proposed solution is evaluated in terms of its components.

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

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. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL 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 SYSTEM ARCHITECTURE

The system architecture establishes the basic structure of the system, defining the essential core design features and elements that provide the framework for the system. The systems architecture provides the architects view of the users' vision for what the system needs to be and do, and the paths along which it must be able to evolve and strives to maintain the integrity of that vision as it evolves during detailed design and implementation.

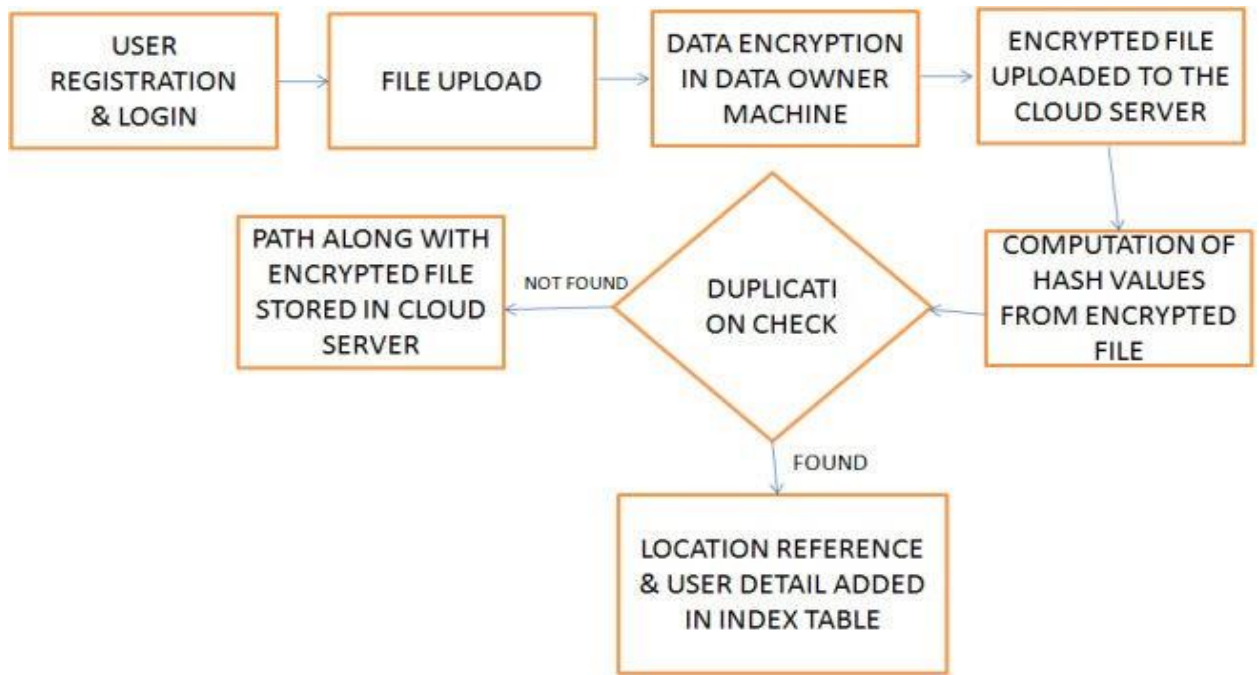


FIG. 4.1 SYSTEM ARCHITECTURE

4.4 MODULE OVERVIEW

- User Module
- Server start up and Upload file
- Secure DE duplicate System
- Download file

4.4.1 USER MODULE

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

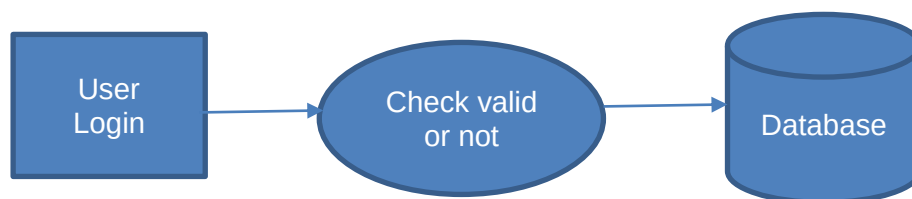


FIG. 4.2 USER MODULE

4.4.2 SERVER START UP AND UPLOAD FILE

The user can start up the server after cloud environment is opened. Then the user can upload the file to the cloud.

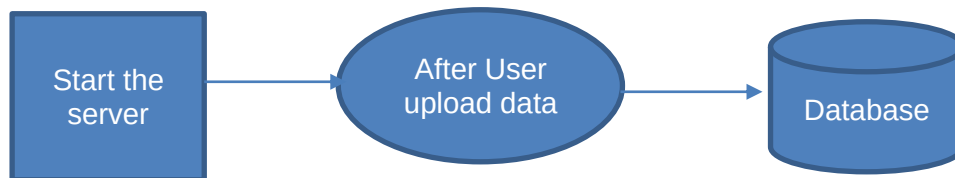


FIG. 4.3 SERVER START UP AND UPLOAD FILE

4.4.3 SECURE DE DUPLICATION SYSTEM

To support authorized de duplication the tag of a file F will be determined by the file F and the privilege. To show the difference with traditional notation of tag, we call it file token instead. To support authorized access a secret key K_P will be bounded with a privilege p to generate a file Token. De duplication exploits identical content, while encryption attempts to make all content appear random; the same content encrypted with two different keys results in very different cipher text. Thus, combining the space efficiency of de duplication with the secrecy aspects of encryption is problematic.



FIG 4.4 SECURE DE DUPLICATION SYSTEM

4.4.4 DOWNLOAD FILE

After the cloud storage, the user can download the file based on key or token. Once the key request was received, the sender can send the key or he can decline it. With this key and request id which was generated at the time of sending key request the receiver can decrypt the message.

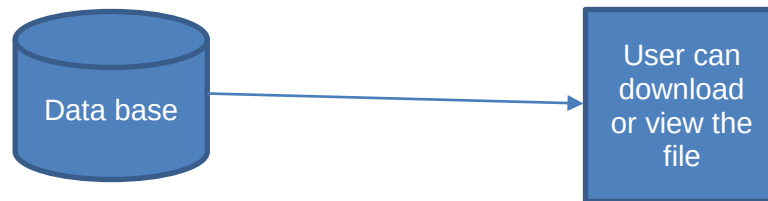


FIG 4.5 DOWNLOAD FILE

4.5 PROPOSED ALGORITHM ENCRYPTION

M3 ENCRYPTION:

- The algorithm itself is very complex and secure, but using it is as simple.
- The basic principle of this algorithm is character-remapping based on key self-mutation.
- The lifespan of a key is equal to the length of the key. This means that any state of the key will only be responsible for encrypting a part of the clear-text that is equal in length to the length of that version of the key before the key is self-mutated into a new version. This new version will then encrypt the next part of the clear-text, etc.
- In the overall process, you have a clear-key entered by the user which is diverted into 4 separate "threads" of different and constantly self-mutating keys. These 4 different keys are responsible for simultaneously converting the clear-text letters one letter at a time into cipher text by 2 different methods, these methods being: array remapping, and a sort of dynamic "substitution cipher". This whole process is repeated over and over again, re-encrypting everything a number of times before the cipher-text is finalized.

- An attempt of reversing the process by a potential attacker would require figuring out the end state of 4 different keys simultaneously going backwards one mutation-version at a time. As 2 of these keys are used for array remapping, it is necessary to get the whole of these keys per letter decoded in the clear-text.

DECRYPTION ALGORITHM

DATA ENCRYPTION STANDARD (DES)

This stands for Data Encryption Standard and it was developed in 1977. It was the first encryption standard to be recommended by NIST (National Institute of Standards and Technology). DES is 64 bits key size with 64 bits block size. Since that time, many attacks and methods have witnessed weaknesses of DES.

Algorithm:

function DES_Encrypt (M, K)

where M = (L, R)

M ← IP (M)

For round ← 1 to 16 do

K ← SK (K, round)

L ← L xor F(R, K_i)

swap(L, R)

end

swap (L, R)

M ← IP⁻¹ (M)

return M

End

4.5.2 CHUNKING TECHNIQUE FOR DEDUPLICATION

Chunking is a process to split a file into smaller files called chunks. In some applications, such as remote data compression, data synchronization, and data de duplication, chunking is important because it determines the duplicate detection performance of the system. Content-defined chunking (CDC) is a method to split files into variable length chunks, where the cut points are defined by some internal features of the files. Unlike fixed-length chunks, variable-length chunks are more

resistant to byte shifting. Thus, it increases the probability of finding duplicate chunks within a file and between files. However, CDC algorithms require additional computation to find the cut points which might be computationally expensive for some applications. In our previous work (Widodo et al., 2016), the hash-based CDC algorithm used in the system took more process time than other processes in the deduplication system. This proposed work shows high throughput hash-less chunking. Instead of using hashes, RAM uses bytes value to declare the cut points. The algorithm utilizes a fix-sized window and a variable-sized window to find a maximum-valued byte which is the cut point. The maximum-valued byte is included in the chunk and located at the boundary of the chunk. This configuration allows RAM to do fewer comparisons while retaining the CDC property. We compared RAM with existing hash-based and hash-less deduplication systems. The experimental results show that our proposed algorithm has higher throughput and bytes saved per second compared to other chunking algorithms.

Chapter - 5

SYSTEM IMPLEMENTATION

DATA FLOW DIAGRAM

The Data Flow diagram is a graphic tool used for expressing system requirements in a graphical form. The DFD also known as the “bubble chart” has the purpose of clarifying system requirements and identifying major transformations that to become program in system design. Thus DFD can be stated as the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. The DFD consist of series of bubbles joined by lines. The bubbles represent data transformations and the lines represent data flows in the system. A DFD describes what that data flow in rather than how they are processed. So it does not depend on hardware, software, data structure or file organization

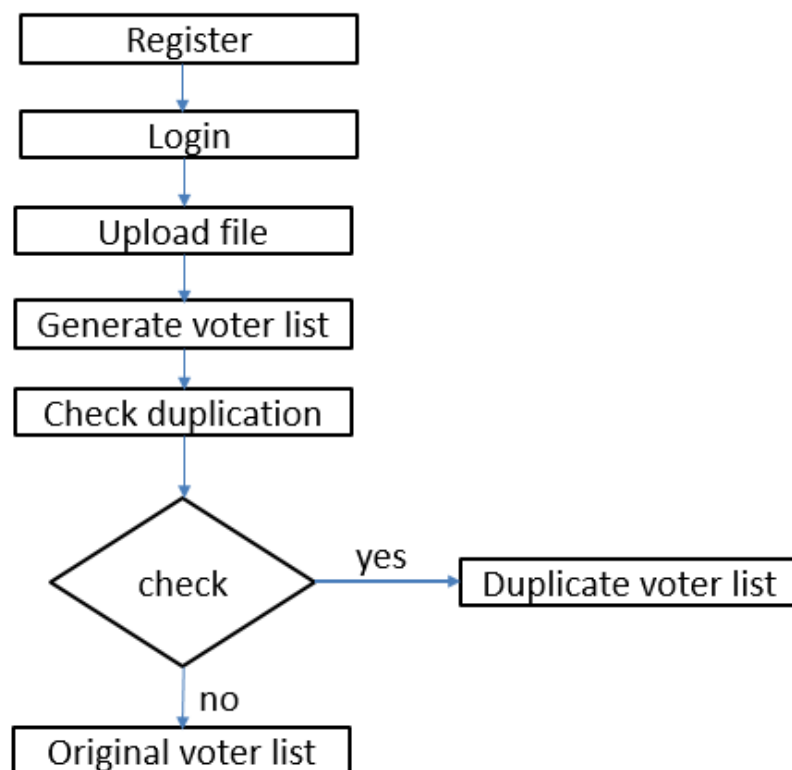


FIG 5.1 DATA FLOW DIAGRAM

CLASS DIAGRAM

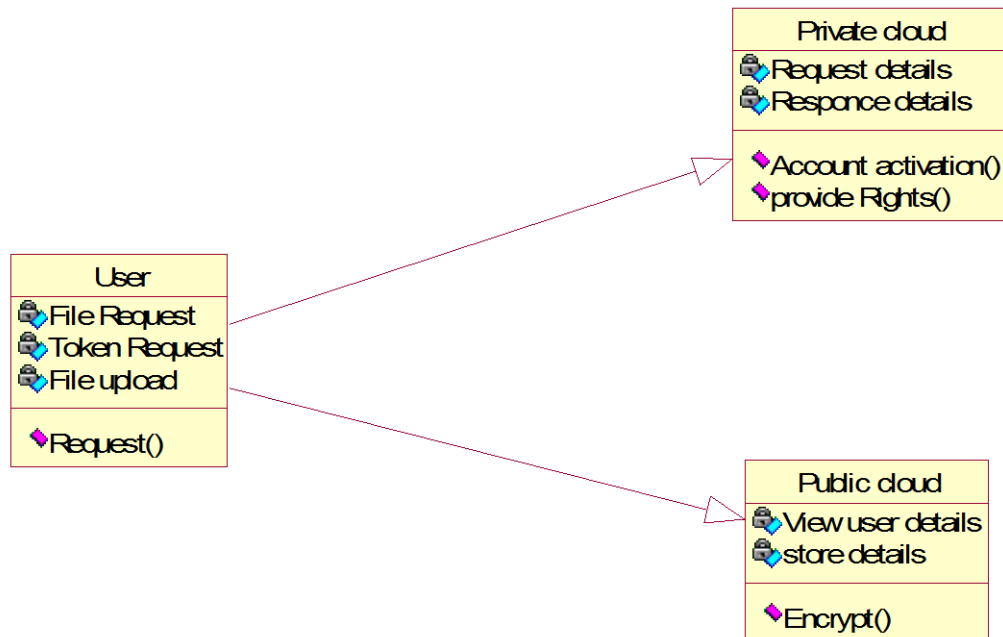


FIG 5.2 CLASS DIAGRAM

The class 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. The message sending procedure involves user registering with the system and logs into the system. The authenticated secret message involves the message extraction and authentication of the message.

USE CASE DIAGRAM

A use case is a set of scenarios that describing an interaction between a user and a system. A use case diagram displays the relationship among actors and use cases. The two main components a user or another system that will interact with the system modeled. A use case is an external view of the system that represents some action the user might perform in order to complete a task.

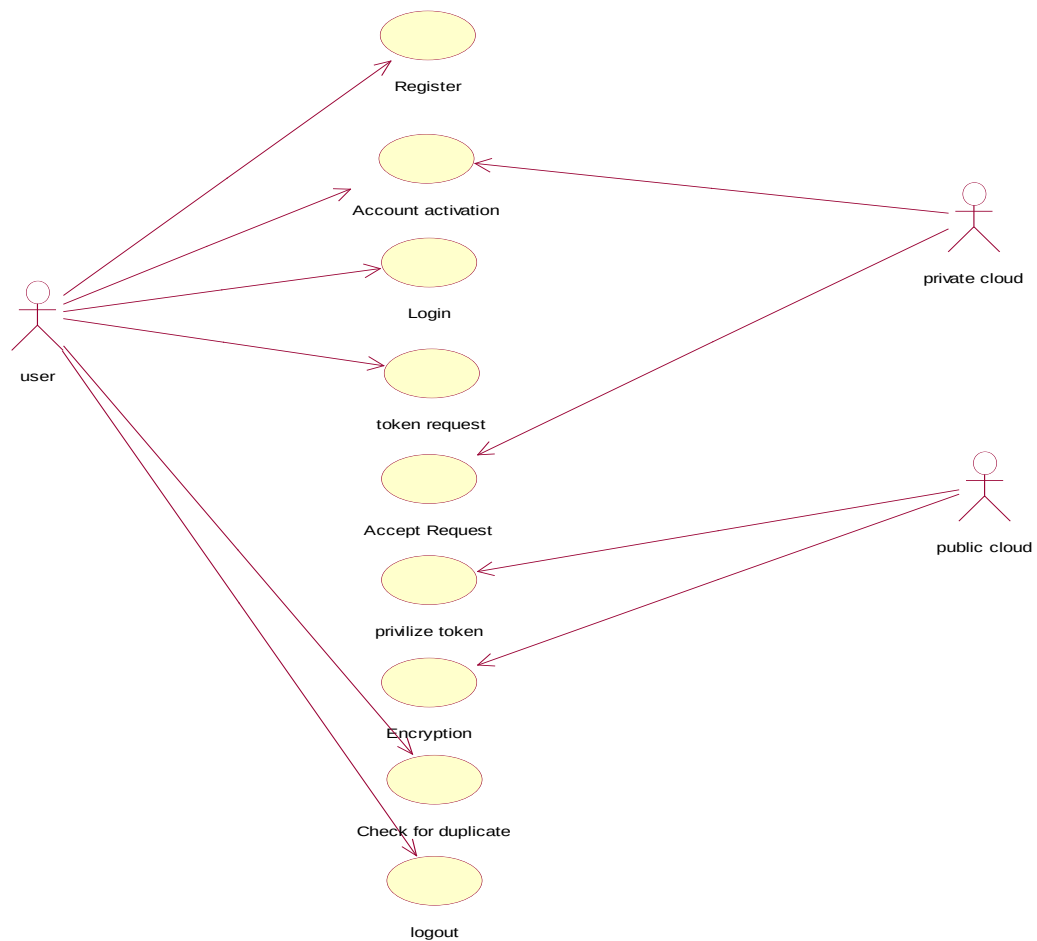


FIG 5.3 USE CASE DIAGRAM

ACTIVITY DIAGRAM

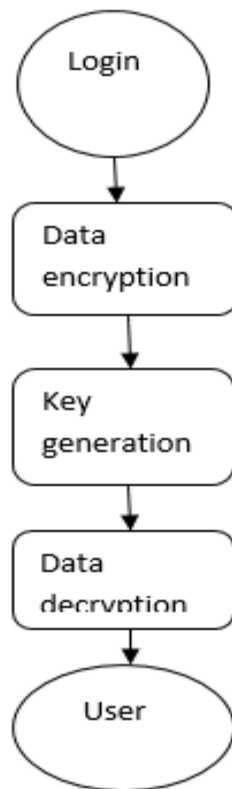


FIG 5.4 ACTIVITY DIAGRAM

SEQUENCE DIAGRAM

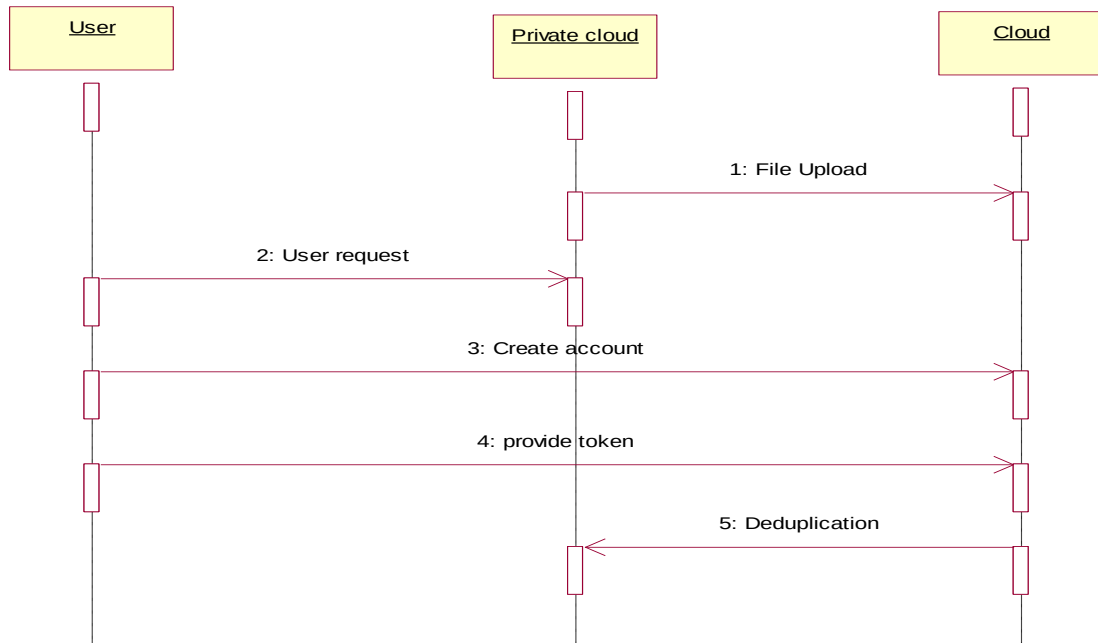


FIG 5.5 SEQUENCE DIAGRAM

The sequence diagram is used to show the flow of the system with the time frame of each activity. The sender logs into the system and generates the file. Then it gets split using the chunk algorithm and encrypts the secret message using blowfish algorithm. The key is generated. The sender gets the private key when he registers with the system and applies inverse blowfish algorithm and split the image into different blocks. Using the private key the message is decrypted and original message is received by the user.

SYSTEM DESIGN AND TESTING PLAN

INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.

- Confirm an action.

➤ **TEST PLAN**

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

➤ **VERIFICATION**

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

➤ **VALIDATION**

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

➤ **BASICS OF SOFTWARE TESTING**

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

➤ **BLACK-BOX TESTING**

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

➤ **WHITE-BOX TESTING**

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

often used for verification.

TYPES OF TESTING

There are many types of testing like

- **UNIT TESTING**
- **INTEGRATION TESTING**
- **FUNCTIONAL TESTING**
- **SYSTEM TESTING**
- **STRESS TESTING**
- **PERFORMANCE TESTING**
- **USABILITY TESTING**
- **ACCEPTANCE TESTING**
- **REGRESSION TESTING**
- **BETA TESTING**

- **UNIT TESTING**

Unit testing is the testing of an individual unit or group of related units. It falls under the class of white box testing. It is often done by the programmer to test that the unit he/she has implemented is producing expected output against given input.

- **INTEGRATION TESTING**

Integration testing is testing in which a group of components are combined to produce output. Also, the interaction between software and hardware is tested in integration testing if software and hardware components have any relation. It may fall under both white box testing and black box testing.

- **FUNCTIONAL TESTING**

Functional testing is the testing to ensure that the specified functionality required in the system requirements works. It falls under the class of black box testing.

- **SYSTEM TESTING**

System testing is the testing to ensure that by putting the software in different environments (e.g., Operating Systems) it still works. System testing is done with full system implementation and environment. It falls under the class of black box testing.

- **STRESS TESTING**

Stress testing is the testing to evaluate how system behaves under unfavorable conditions. Testing is conducted at beyond limits of the specifications. It falls under the class of black box testing.

- **PERFORMANCE TESTING**

Performance testing is the testing to assess the speed and effectiveness of the system and to make sure it is generating results within a specified time as in performance requirements. It falls under the class of black box testing.

- **USABILITY TESTING**

Usability testing is performed to the perspective of the client, to evaluate how the GUI is user-friendly? How easily can the client learn? After learning how to use, how proficiently can the client perform? How pleasing is it to use its design? This falls under the class of black box testing.

- **ACCEPTANCE TESTING**

Acceptance testing is often done by the customer to ensure that the delivered product meets the requirements and works as the customer expected. It falls under the class of black box testing.

- **REGRESSION TESTING**

Regression testing is the testing after modification of a system, component, or a group of related units to ensure that the modification is working correctly and is not damaging or imposing other modules to produce unexpected results. It falls under the class of black box testing.

REQUIREMENT ANALYSIS

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

FUNCTIONAL REQUIREMENTS

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

- **USABILITY**

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

- **ROBUSTNESS**

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

- **SECURITY**

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

there by providing high security.

- **RELIABILITY**

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

- **COMPATIBILITY**

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

- **FLEXIBILITY**

The flexibility of the project is provided in such a way that is has the ability to run on different environments being executed by different users.

- **SAFETY**

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

NON- FUNCTIONAL REQUIREMENTS

- **PORTABILITY**

It is the usability of the same software in different environments. The project can be run in any operating system.

- **PERFORMANCE**

These requirements determine the resources required, time interval, throughput and everything that deals with the performance of the system.

- **ACCURACY**

The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective.

- **MAINTAINABILITY**

Project is simple as further updates can be easily done without affecting its stability. Maintainability basically defines that how easy it is to maintain the

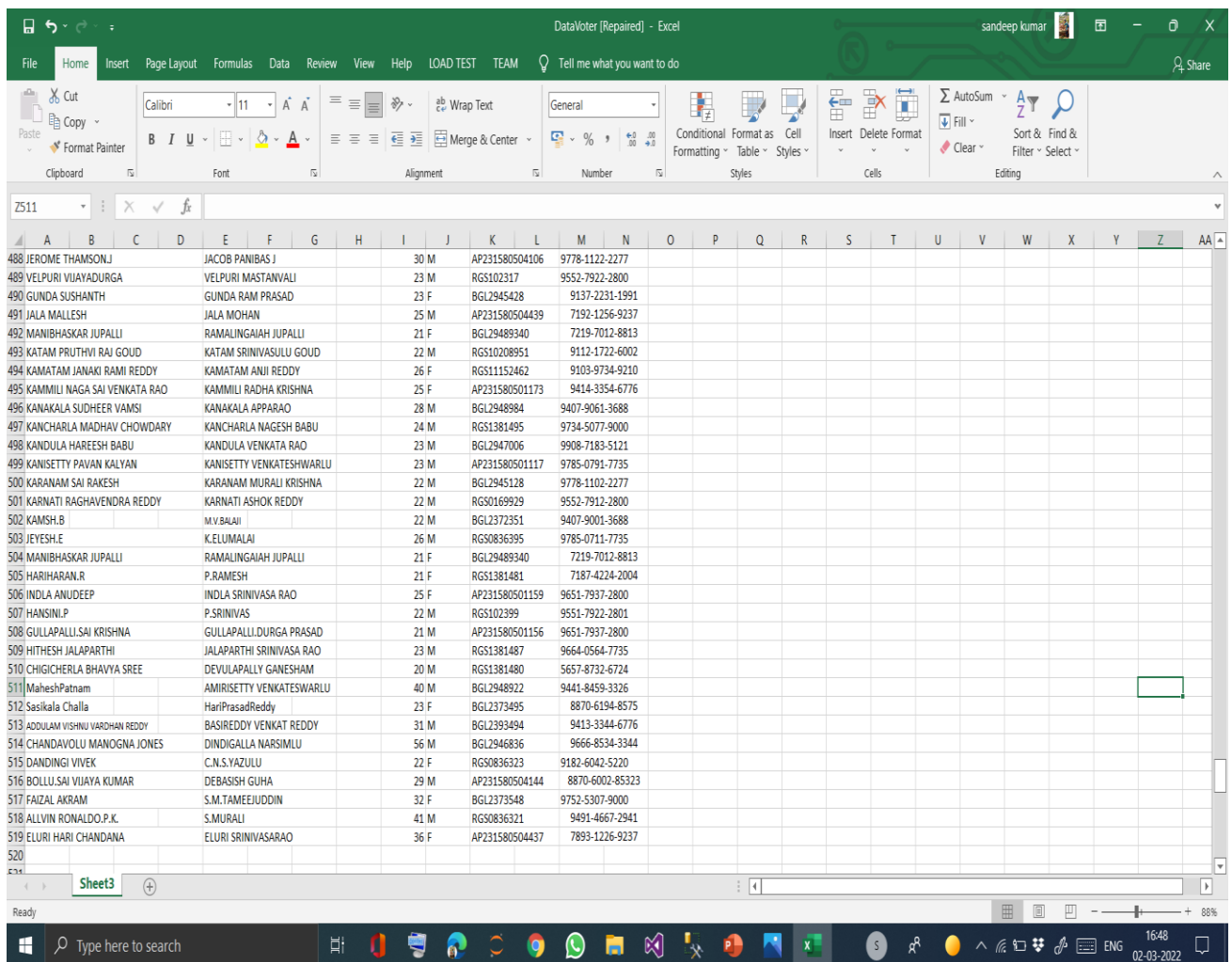
system. It means that how easy it is to maintain the system, analyse, change and test the application. Maintainability of this project is simple as further updates can be easily done without affecting its stability.

CHAPTER - 6

CONCLUSION

In this paper, the notion of authorized data de duplication was proposed to protect the data security by including differential privileges of users in the duplicate check. We also presented several new de duplication constructions supporting authorized duplicate check in hybrid cloud architecture, in which the duplicate-check tokens of files are generated by the private cloud server with private keys. Security analysis demonstrates that our schemes are secure in terms of insider and outsider attacks specified in the proposed security model. As a proof of concept, we implemented a prototype of our proposed authorized duplicate check scheme and conduct test bed experiments on our prototype. We showed that our authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer.

OUTPUT SCREENSHOTS



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
488	JEROME THAMSON J			JACOB PANIBAS J						30 M	AP231580504106		9778-1122-2277														
489	VELPURI VIYADURGA			VELPURI MASTANVALI						23 M	RG5102317		9552-7922-2800														
490	GUNDA SUSHANTH			GUNDA RAM PRASAD						23 F	BGL2945428		9137-2231-1991														
491	JALA MALLES			JALA MOHAN						25 M	AP231580504439		7192-1256-9237														
492	MANIBHASKAR JUPALLI			RAMALINGIAH JUPALLI						21 F	BGL29489340		7219-7012-8813														
493	KATAM PRUTHVI RAJ GOUD			KATAM SRINIVASULU GOUD						22 M	RG510208951		9112-1722-6002														
494	KAMATAM JANAKI RAMI REDDY			KAMATAM ANJI REDDY						26 F	RG511152462		9103-9734-9210														
495	KAMMILI NAGA SAI VENKATA RAO			KAMMILI RADHA KRISHNA						25 F	AP231580501173		9414-3354-6776														
496	KANAKALA SUDHEER VAMSI			KANAKALA APPARAO						28 M	BGL2948984		9407-9061-3688														
497	KANCHARLA MADHAV CHOWDARY			KANCHARLA NAGESH BABU						24 M	RG51381495		9734-5077-9000														
498	KANDULA HAREESH BABU			KANDULA VENKATA RAO						23 M	BGL2947006		9908-7183-5121														
499	KANISSETTY PAVAN KALYAN			KANISSETTY VENKATESHWARLU						23 M	AP231580501117		9785-0791-7735														
500	KARANAM SAI RAKESH			KARANAM MURALI KRISHNA						22 M	BGL2945128		9778-1102-2277														
501	KARNATI RAGHAVENDRA REDDY			KARNATI ASHOK REDDY						22 M	RG50169929		9552-7912-2800														
502	KAMSH.B			M.V.BALAI						22 M	BGL2372351		9407-9001-3688														
503	JEYESH.E			K.ELUMALAI						26 M	RG50836395		9785-0711-7735														
504	MANIBHASKAR JUPALLI			RAMALINGIAH JUPALLI						21 F	BGL29489340		7219-7012-8813														
505	HARIHARAN.R			P.RAMESH						21 F	RG51381481		7187-4224-2004														
506	INDLA ANUDEEP			INDLA SRINIVASA RAO						25 F	AP231580501159		9651-7937-2800														
507	HANSINLP			P.SRINIVAS						22 M	RG5102399		9551-7922-2801														
508	GULLAPALLI.SAI KRISHNA			GULLAPALLI.DURGA PRASAD						21 M	AP231580501156		9651-7937-2800														
509	HITHESH JALAPARTHI			JALAPARTHI SRINIVASA RAO						23 M	RG51381487		9664-0564-7735														
510	CHIGICHERLA BHAVYA SREE			DEVULAPALLY GANESHAM						20 M	RG51381480		5657-8732-6724														
511	MareshPatnam			AMIRISETTY VENKATESWARLU						40 M	BGL2948922		9441-8459-3326														
512	Sasikala Challa			HariPrasadReddy						23 F	BGL2373495		8870-6194-8575														
513	ADOLAM VISHNU VARDHAN REDDY			BASIREDDY VENKAT REDDY						31 M	BGL2393494		9413-3344-6776														
514	CHANDAVOLU MANOGBA JONES			DINDIGALLA NARSIMLU						56 M	BGL2946836		9666-8534-3344														
515	DANDINGI VIVEK			C.N.S.YAZULU						22 F	RG50836323		9182-6042-5220														
516	BOLLU.SAI VIJAYA KUMAR			DEBASISH GUHA						29 M	AP231580504144		8870-6002-85323														
517	FAIZAL AKRAM			S.M.TAMEEJUDDIN						32 F	BGL2373548		9752-5307-9000														
518	ALLVIN RONALDO.P.K.			S.MURALI						41 M	RG50836321		9491-4667-2941														
519	ELURI HARI CHANDANA			ELURI SRINIVASARAO						36 F	AP231580504437		7893-1226-9237														
520																											
521																											

Fig 6.1 DATA SET

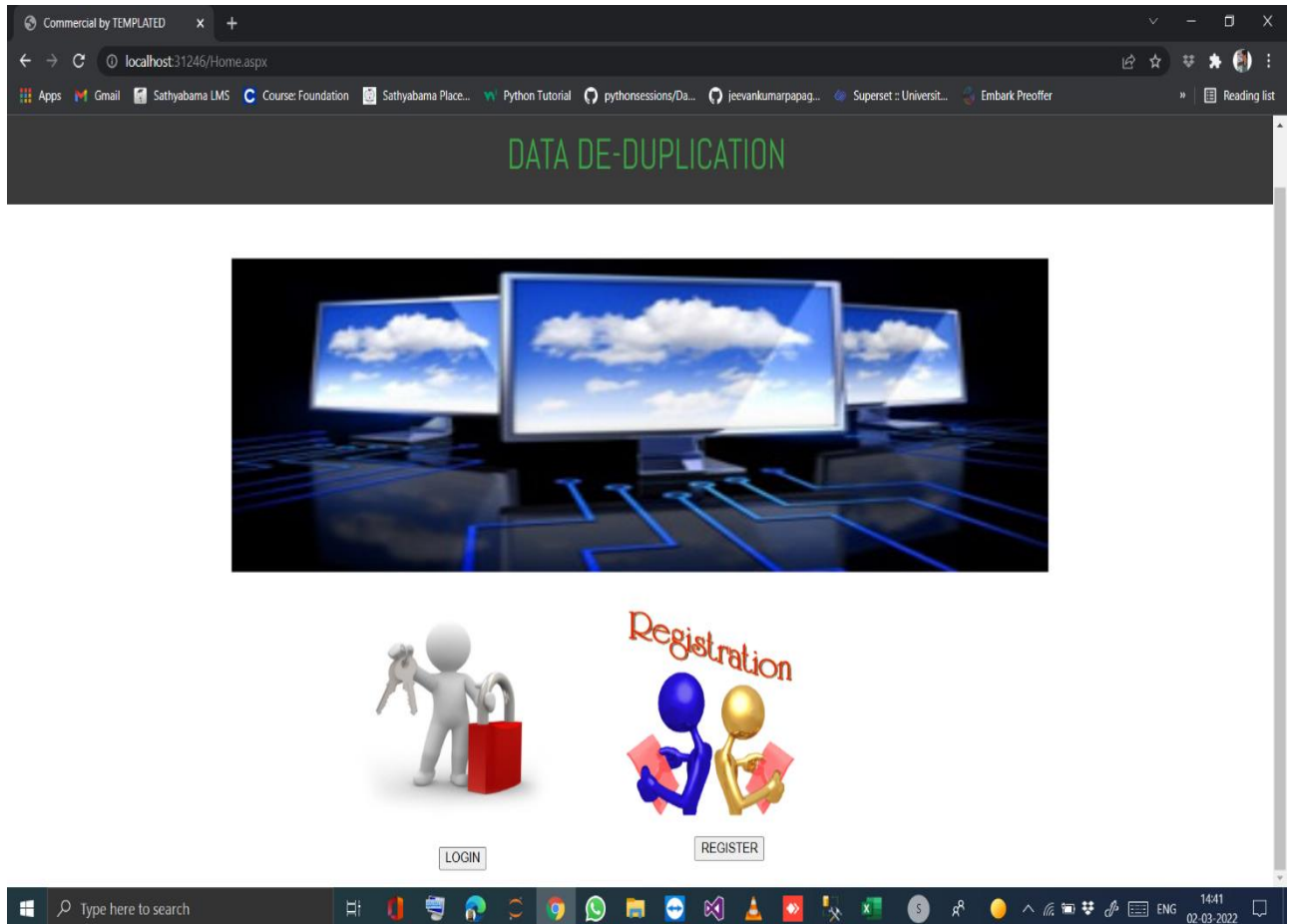


Fig 6.2 INTERFACE

The screenshot shows a web browser window with the title 'DATA DEDUPLICATION'. The address bar shows 'localhost:31246/REGISTER.aspx'. The browser's tab bar shows several open tabs, including 'Apps', 'Gmail', 'Sathyabama LMS', 'Course: Foundation', 'Sathyabama Place...', 'Python Tutorial', 'pythonsessions/Da...', 'jeevankumarpapag...', 'Superset - Universit...', and 'Embark Preoffer'. The main content area has a dark background with the text 'DATA DE-DUPLICATION' in green. Below this is a green navigation bar with links: 'HOME', 'DATAOWNER', 'CSP', 'USER', and 'LOGOUT'. The main heading is 'REGISTER NOW !!'. The registration form includes the following fields: 'Name', 'Email ID', 'Mobile NO', 'UserName', 'Password', and 'UserType' (a dropdown menu with 'select type' selected). At the bottom of the form are two buttons: 'REGISTER' and 'BACK'. The Windows taskbar at the bottom shows the search bar, task view button, and several application icons, including Edge, WhatsApp, File Explorer, and others. The system clock shows '14:41' on '02-03-2022'.

DATA DEDUPLICATION

HOME DATAOWNER CSP USER LOGOUT

REGISTER NOW !!

Name

Email ID

Mobile NO

UserName

Password

UserType

REGISTER BACK

Fig 6.3 REGISTRATION

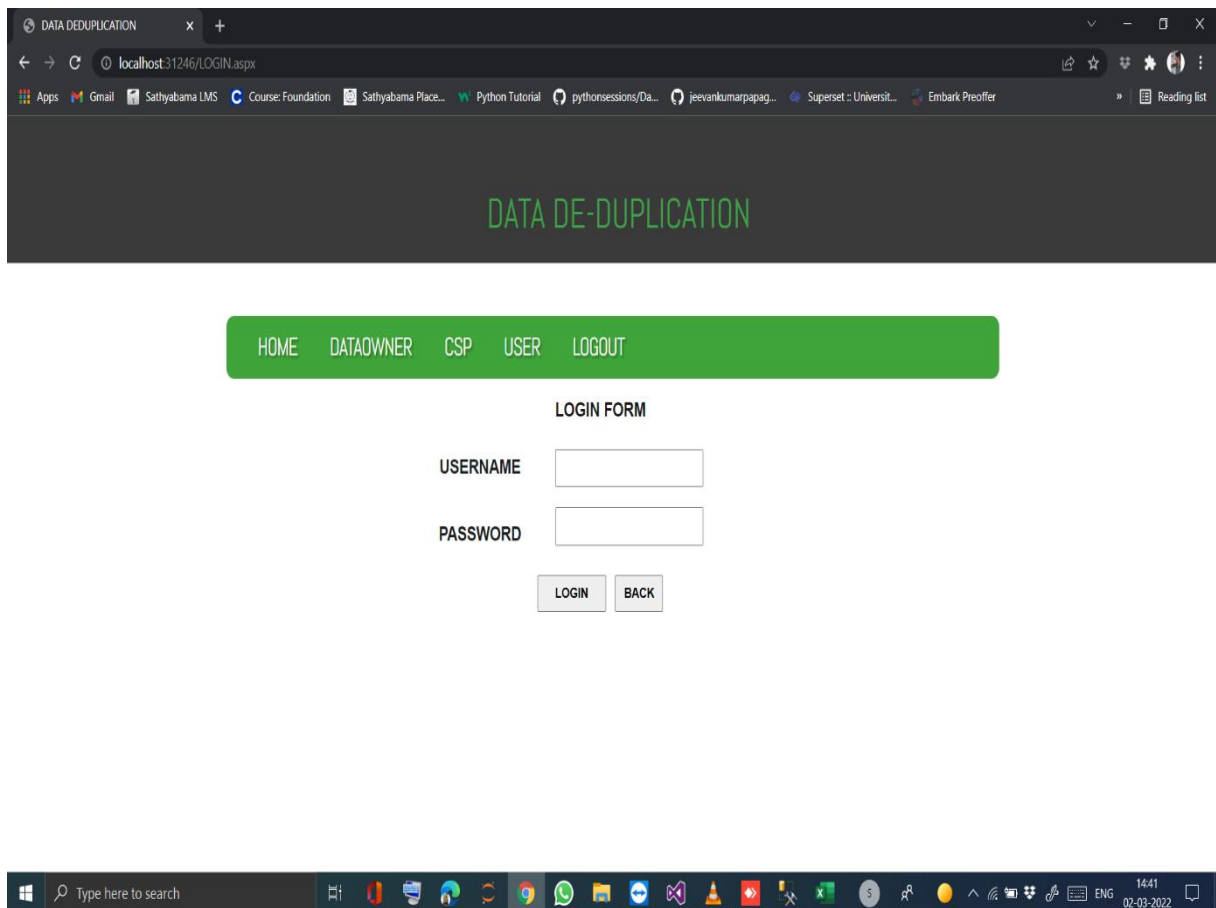


Fig 6.4 LOGIN

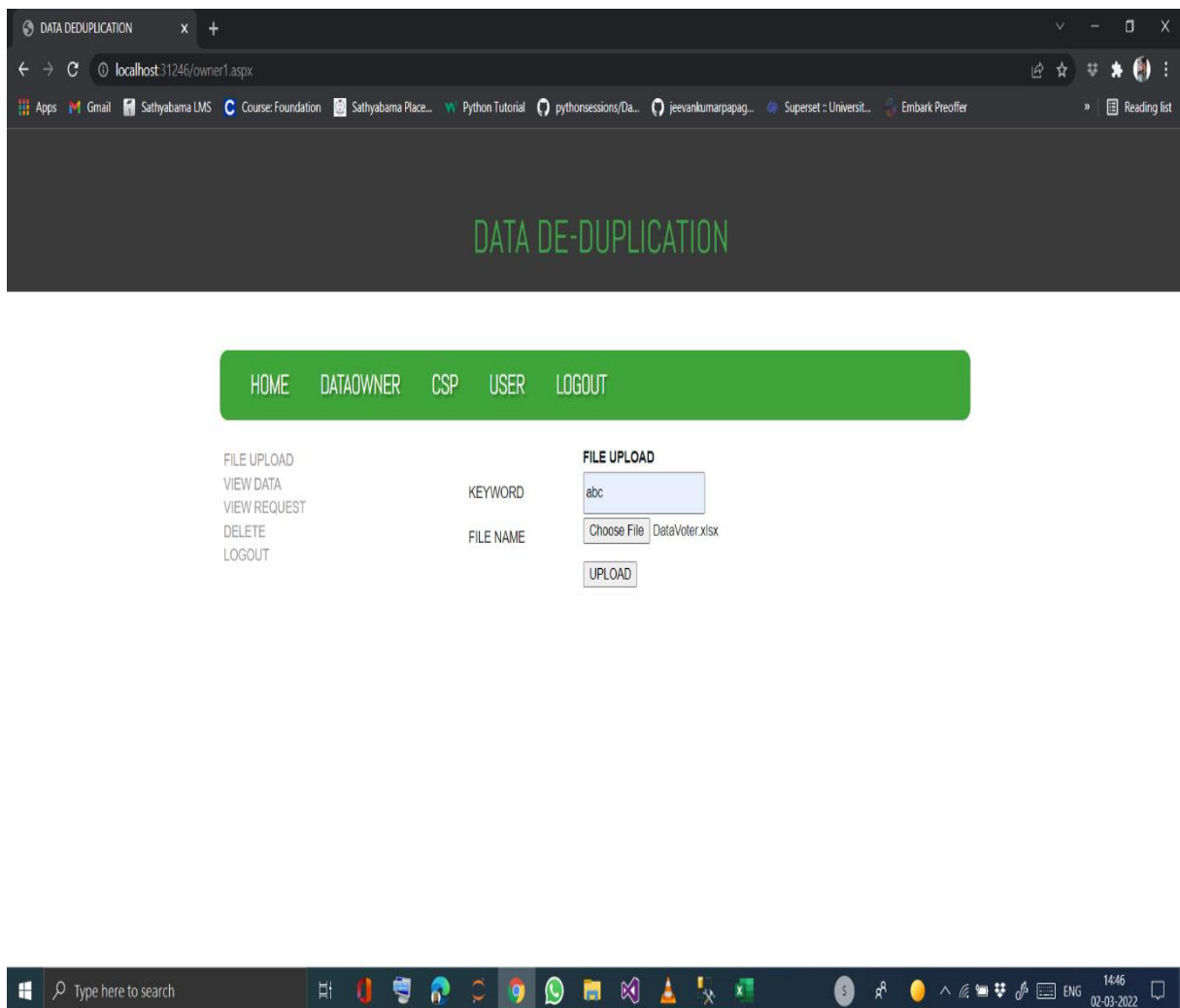


Fig 6.5 FILE UPLOAD

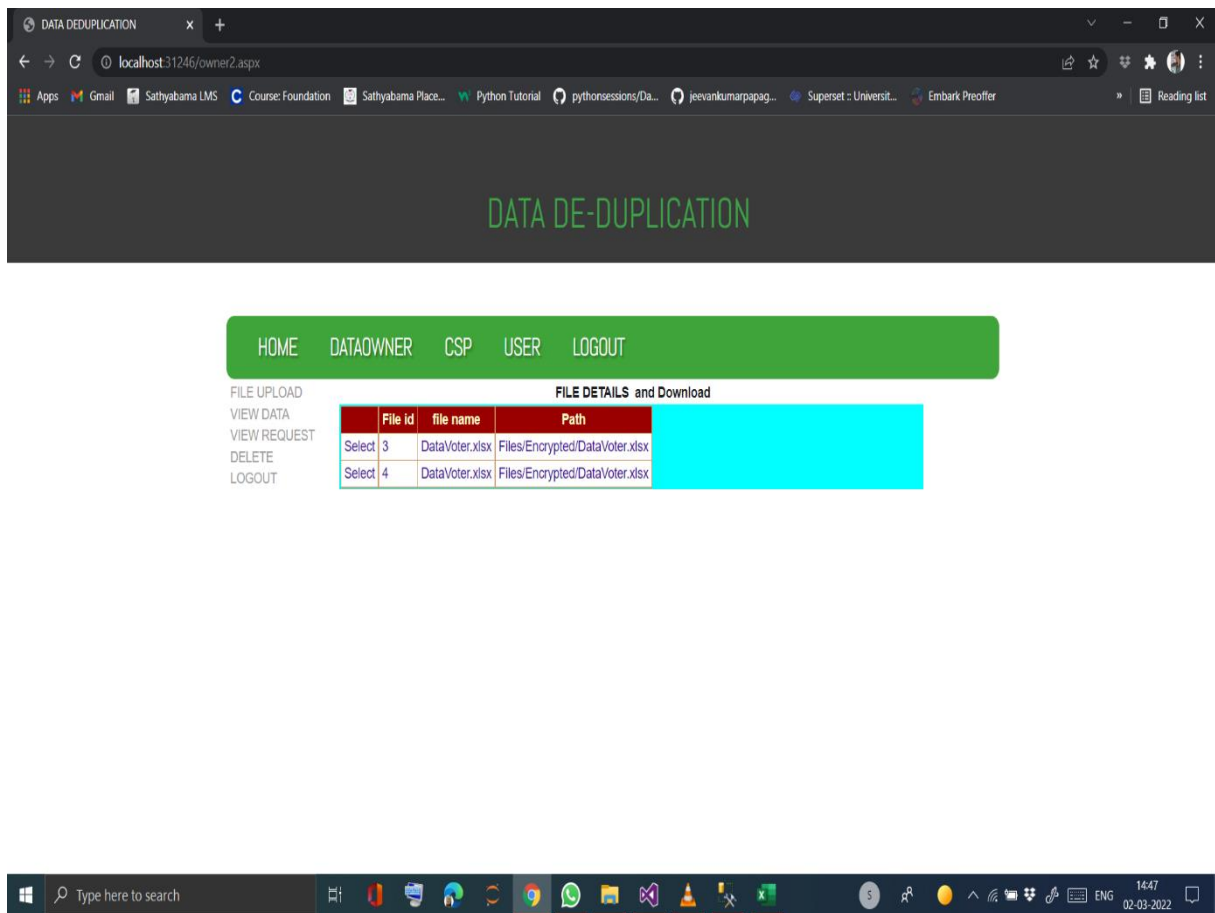


Fig 6.6 VIEW UPLOAD FILES

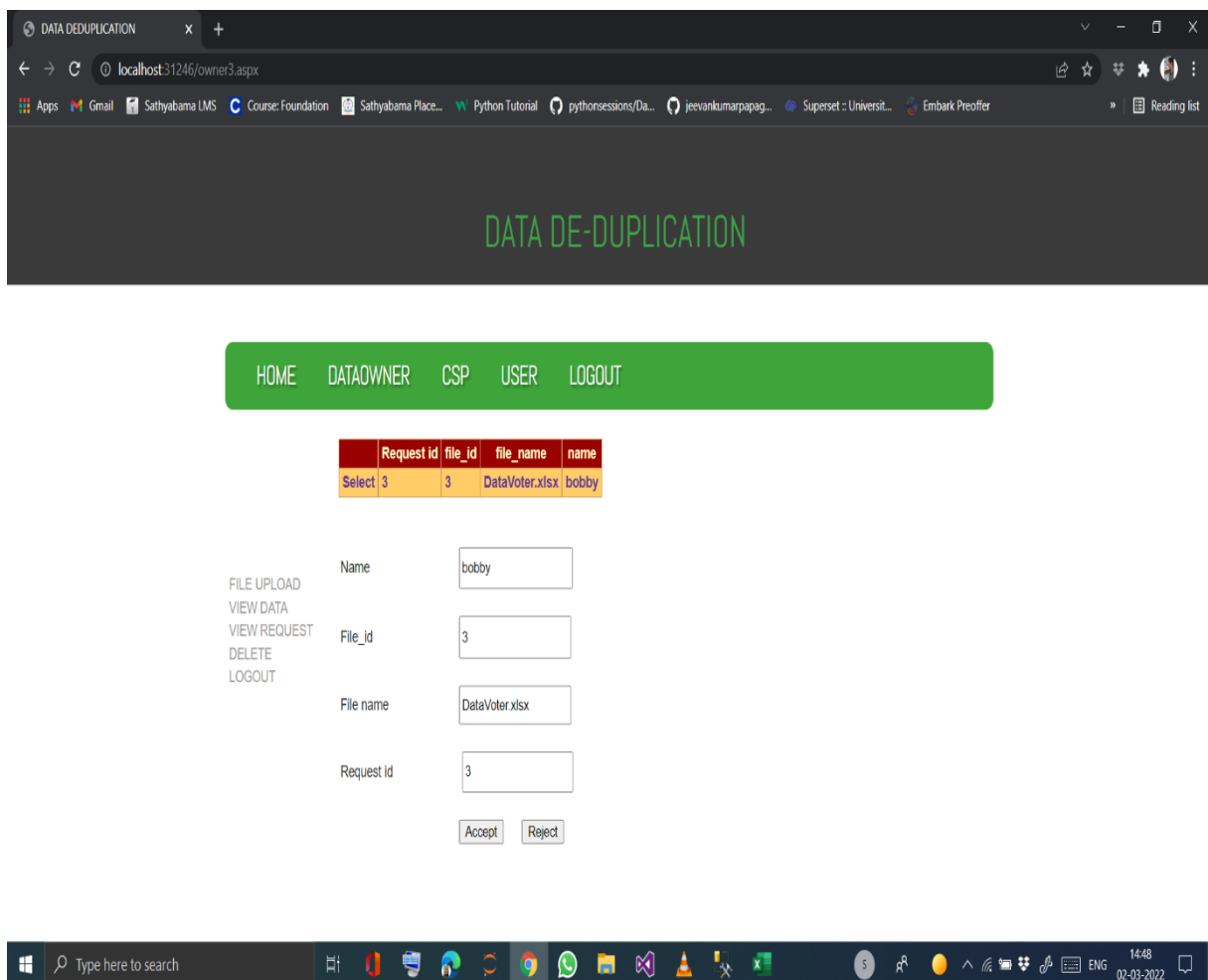


Fig 6.7 SELECT THE FILE

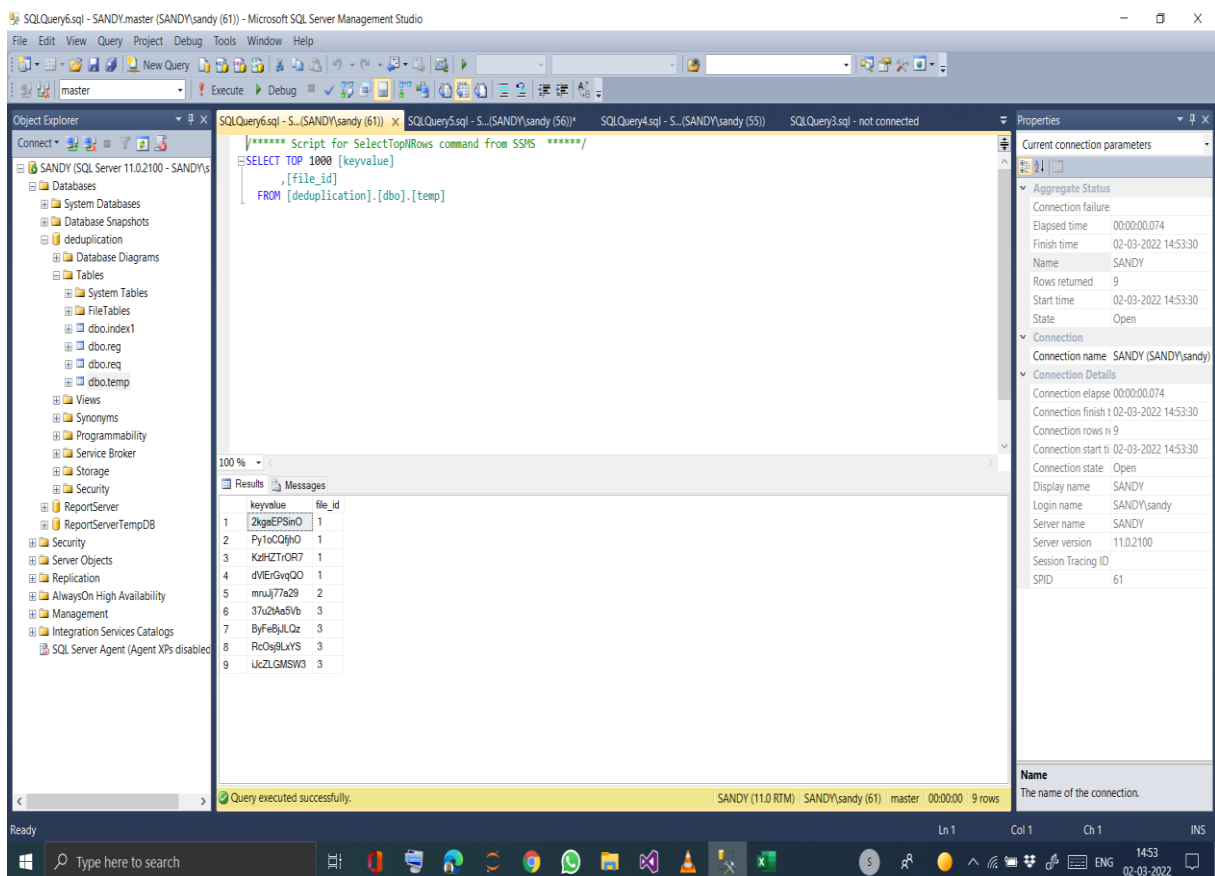


Fig 6.8 GET APPLY KEY

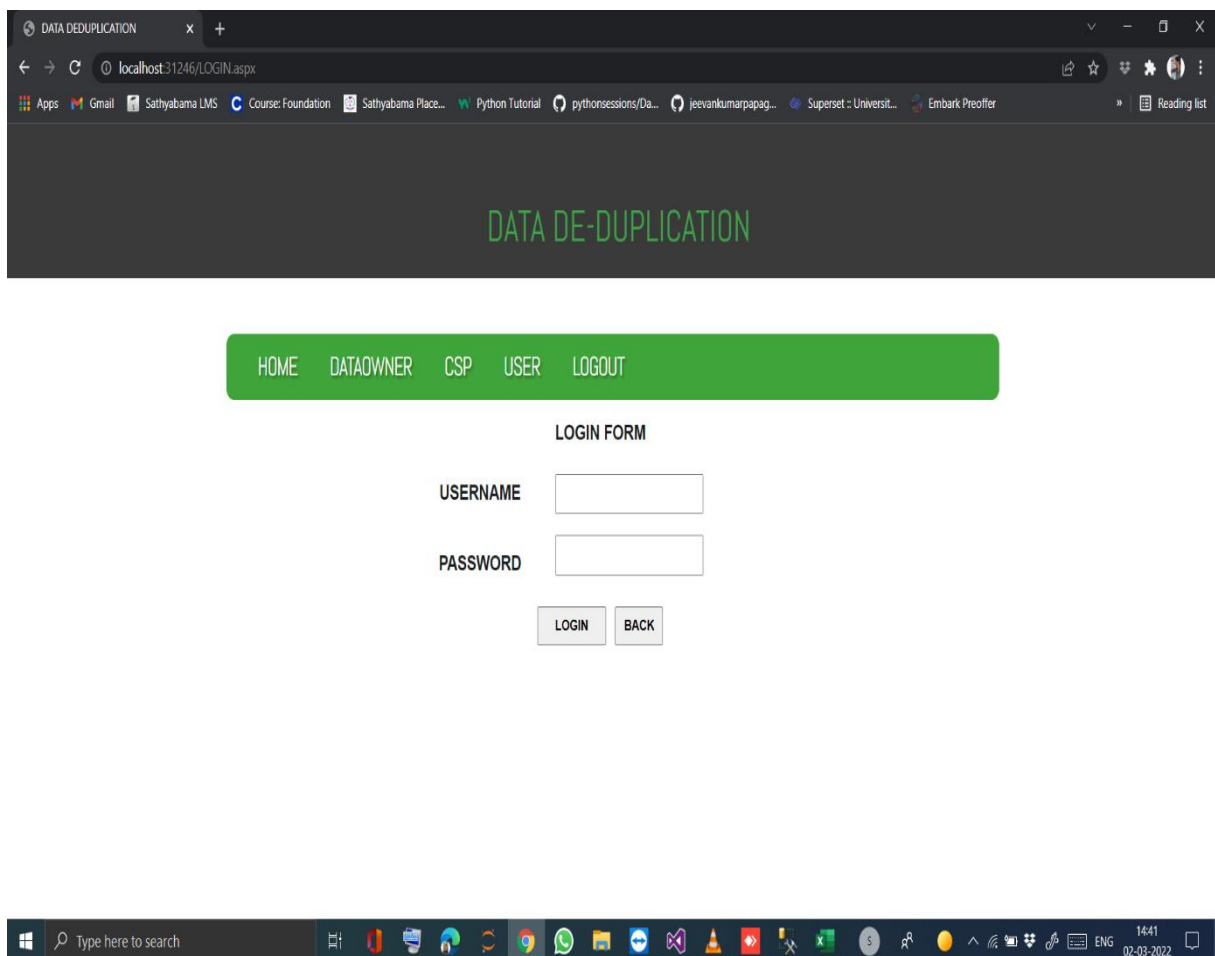


Fig 6.9 USER LOGIN

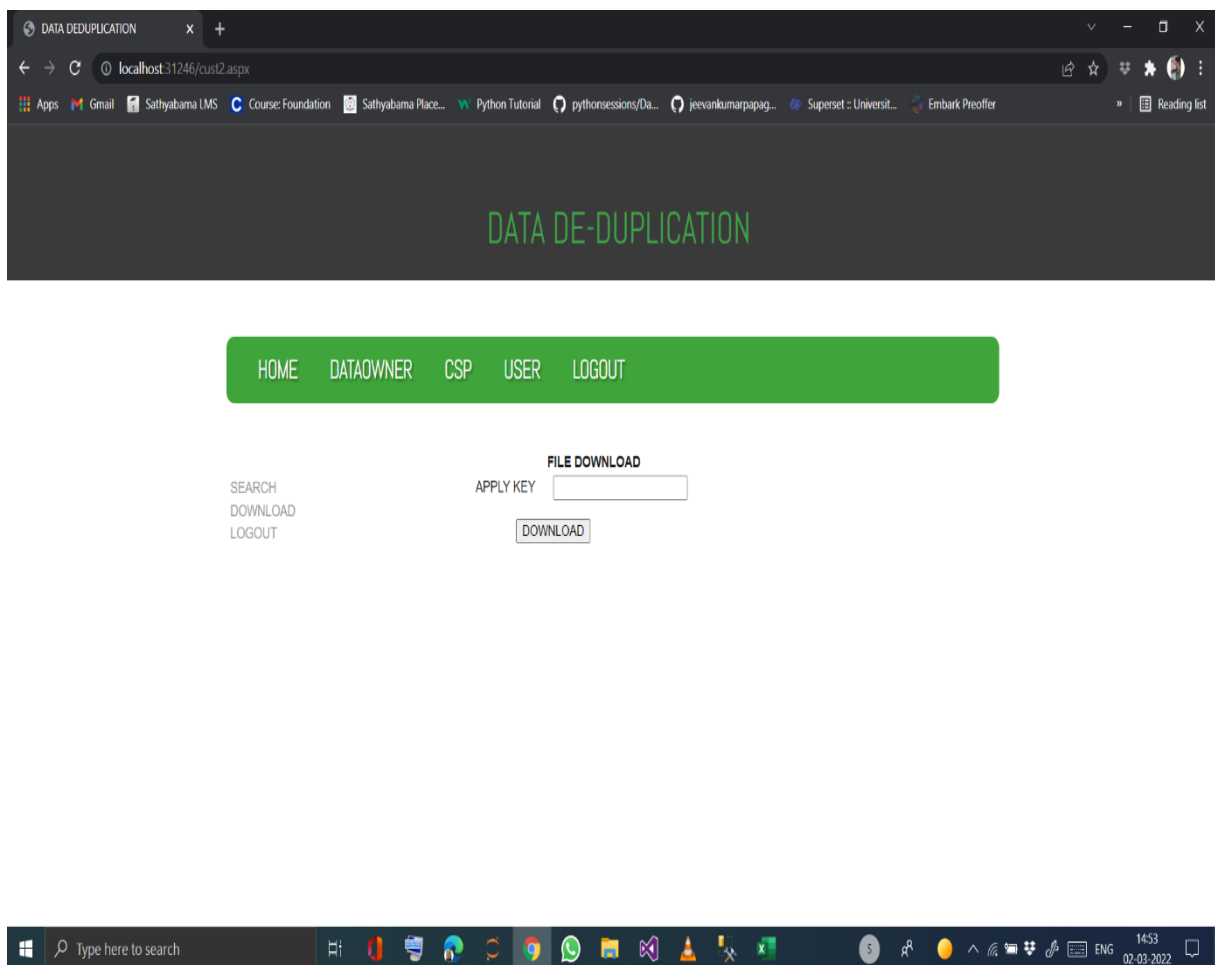


Fig 6.10 COPY THE APPLY KEY

Excel window: DataVoter - Excel, sandeep kumar

File Home Insert Page Layout Formulas Data Review View Help LOAD TEST TEAM Tell me what you want to do

Clipboard: Cut, Copy, Paste, Format Painter

Font: Calibri, 11, Bold, Italic, Underline, Text Color, Background Color

Alignment: Wrap Text, Merge & Center

Number: General, Percentage, Decimal, Fraction, Scientific, Text, Error

Conditional Formatting: Cell Styles, Conditional Formatting

Insert: Insert, Delete, Insert Rows, Insert Columns, Insert Hyperlink

Delete Format: Delete Format

AutoSum: AutoSum, Fill, Clear

Sort & Filter: Sort & Filter, Find & Select

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
448	HARSHITAA.A			ANAND.K					25	F	RG57378514		9001-9881-3688										
449	CH.HEMASAISREE			CH BHASKAR RAO					22	M	AP231580501155		9002-5357-9000										
450	HEMANTH REDDY MADIREDDY			PARI PURNA REDDY MADIREDDY					32	F	BGL2948916		9889-7823-5121										
451	HITHESH JALAPARTHI			JALAPARTHI SRINIVASA RAO					23	M	RG51381487		9664-0564-7735										
452	INAMPUDI JYOTSNA			INAMPUDI SRINIVASA RAO					24	F	BGL2947088		9996-1115-2277										
453	INDLA ANUDEEP			INDLA SRINIVASA RAO					25	F	AP231580501159		9651-7937-2800										
454	INTURI BHANU SAI PRASANTH			INTURI CHANDRA MOHAN					22	M	BGL2945130		9995-8006-2832										
455	SRI HARISH.J			T S JAYARAM					22	M	RG50169921		9143-5802-3830										
456	JALLURI GUNASEKHAR			J.VEERA VENKATA KRISHNA KISHORE					22	M	AP231580501152		8850-6052-85323										
457	RAMYA.J			J.SRINIVASARAO					23	M	BGL2408493		8337-4204-2004										
458	JANGITI SANDEEP			NARASIMHA					26	M	BGL2413494		9009-7012-1996										
459	JAWAHAR.K			K.VASU					27	M	RG50625		9613-3014-1996										
460	JOGI HARIKA			JOGI SRINIVASA RAO					30	M	BGL2946836		7519-9098-2332										
461	JONNALAGADDA SRI SAMHITHA			JONNALAGADDA SRI KRISHNA BHOGI					23	F	BGL2372397		9017-2001-1991										
462	JUJJAVARAPU DILEEP KUMAR			JUJJAVARAPU KRISHNA					23	M	BGL2946768		7813-1226-9237										
463	JUJJAVARPU SRIHARI			JUJJAVARPU VENKATRAJU					25	M	BGL2373549		7210-7002-8813										
464	MANI BHUSHAN.K			K KODANDA RAMA NAIDU					21	M	RG50836320		9121-1772-6002										
465	PRIYANKA.K			K KUMAR ACHAR					22	M	AP231580504141		9131-9774-9210										
466	PRUDHVI.K.S.S.V.S.M.P			KATTAMURI KRISHNA					26	F	RG5102392		9491-3344-6776										
467	KADAMUTHURU GAYATRI			KADAMUTHURU PERUMALLAIAH					25	M	BGL2945430		9431-9001-3688										
468	KADUMURI RAJESH VARMA			K.YUGANDHAR RAJU					28	M	AP231580504434		9762-5007-9000										
469	KAIGALA SATYA SRINIVAS			KAIGALA ANJANEYULU					24	F	BGL2948938		9985-7113-5121										
470	GURRALA BHAVESH SAI PRETHAM			GURRALA HRUSHIKESH					23	M	RG58322008		9714-0711-7735										
471	GURRAM MANMOHAN			G.M.RAYUDU					23	F	RG59265485		9916-1122-2277										
472	GURUCHARUN.K			D.KUBERAN					22	M	AP231580501157		9490-7922-2800										
473																							
474																							
475																							
476																							

Sheet3

Ready

Type here to search

1701 02-03-2022

Fig 6.11 FINAL DOWNLOADED FILE

REFERENCES

1. M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless: Serveraided encryption for deduplicated storage. In USENIX Security Symposium, 2013.
2. M. Bellare, S. Keelveedhi, and T. Ristenpart. Message-locked encryption and secure deduplication. In EUROCRYPT, pages 296–312, 2013.
3. M. Bellare, C. Namprempre, and G. Neven. Security proofs for identity-based identification and signature schemes. *J. Cryptology*, 22(1):1–61, 2009.
4. M. Bellare and A. Palacio. Gq and schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks. In CRYPTO, pages 162–177, 2002.
5. S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider. Twin clouds: An architecture for secure cloud computing. In Workshop on Cryptography and Security in Clouds (WCSC 2011), 2011.
6. P. Anderson and L. Zhang. Fast and secure laptop backups with encrypted de-duplication. In Proc. of USENIX LISA, 2010.
7. M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless: Serveraided encryption for deduplicated storage. In USENIX Security Symposium, 2013.
8. M. Bellare, S. Keelveedhi, and T. Ristenpart. Message-locked encryption and secure deduplication. In EUROCRYPT, pages 296– 312, 2013.
9. M. Bellare, C. Namprempre, and G. Neven. Security proofs for identity-based identification and signature schemes. *J. Cryptology*, 22(1):1–61, 2009.
10. M. Bellare and A. Palacio. Gq and schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks. In CRYPTO, pages 162–177, 2002.
11. S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider. Twin clouds: An architecture for secure cloud computing. In Workshop on Cryptography and Security in Clouds (WCSC 2011), 2011.
12. J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer. Reclaiming space from duplicate files in a serverless distributed file system. In ICDCS, pages 617– 624, 2002.
13. D. Ferraiolo and R. Kuhn. Role-based access controls. In 15th NIST-NCSC National Computer Security Conf., 1992.

14. S. Halevi, D. Harnik, B. Pinkas, and A. ShulmanPeleg. Proofs of ownership in remote storage systems. In Y. Chen, G. Danezis, and V. Shmatikov, editors, ACM Conference on Computer and Communications Security, pages 491–500. ACM, 2011.
15. J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou. Secure deduplication with efficient and reliable convergent key management. In IEEE Transactions on Parallel and Distributed Systems, 2013.
16. C. Ng and P. Lee. Revdedup: A reverse deduplication storage system optimized for reads to latest backups. In Proc. of APSYS, Apr 2013.
17. W. K. Ng, Y. Wen, and H. Zhu. Private data deduplication protocols in cloud storage. In S. Ossowski and P. Lecca, editors, Proceedings of the 27th Annual ACM Symposium on Applied Computing, pages 441–446. ACM, 2012.
18. R. D. Pietro and A. Sorniotti. Boosting efficiency and security in proof of ownership for deduplication. In H. Y. Youm and Y. Won, editors, ACM Symposium on Information, Computer and Communications Security, pages 81–82. ACM, 2012.
19. S. Quinlan and S. Dorward. Venti: a new approach to archival storage. In Proc. USENIX FAST, Jan 2002. [18] A. Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee, and J. C. S. Lui. A secure cloud backup system with assured deletion and version control. In 3rd International Workshop on Security in Cloud Computing, 2011.
20. R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. Role-based access control models. IEEE Computer, 29:38–47, Feb 1996.
21. J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl. A secure data deduplication scheme for cloud storage. In Technical Report, 2013.