A PROJECT REPORT

ON

EFFICIENT TRACEABLE AUTHORIZED SEARCH SYSTEM FOR SECURE CLOUD STORAGE

A Project report submitted in partial fulfilment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE & ENGINEERING

Submitted By

P. Naga Lakshmi	192P1A0595
M. Mahaboob Basha	192P1A0576
P. Bindu Shruthika	192P1A0596
M. Karthik	192P1A0569
M. Mahendra Kumar Reddy	192P1A0575

Under The Esteemed Guidance of

Mr. N. SRINIVASAN M.Tech P. Naga Lakshmi

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AN ISO 9001:2015 CERTIFIED INSTITUTION

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY

(Sponsored by Bharathi Educational Society)

(Affiliated to J.N.T.U.A., Anantapuramu, Approved by AICTE, New Delhi)

Recognized by UGC Under the Section 2(f) & 12(B) of UGC Act, 1956

Accredited by NAAC & NBA

Autonomous

Vidyanagar, Proddatur-516360, Y.SR.(Dist.), A.P.

2019 - 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AN ISO 9001:2015 CERTIFIED INSTITUTION

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY

(Sponsored by Bharathi Educational Society)

(Affiliated to J.N.T.U.A., Anantapuramu, Approved by AICTE, New Delhi)

Recognized by UGC Under the Section 2(f) & 12(B) of UGC Act, 1956

Accredited by NAAC & NBA

Autonomous

Vidyanagar, Proddatur-516360, Y.SR.(Dist.), A.P.

CERTIFICATE

This is to certify that the project work entitled "**EFFICIENT TRACEABLE AUTHORIZED SEARCH SYSTEM FOR SECURE CLOUD STORAGE**" is a bonafide work of P. Naga Lakshmi (192P1A0595), M. Mahaboob Basha (192P1A0576), P. Bindu Shruthika (192P1A0596), M. Karthik (192P1A0569), M. Mahendra Kumar Reddy (192P1A0575) submitted to Chaitanya Bharathi Institute of Technology, Proddatur in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in COMPUTER SCIENCE AND ENGINEERING. The work reported here in does not form part of any other thesis on which a degree has been awarded earlier. This is to further certify that they have worked for a period of one semester for preparing their work under our supervision and guidance.

INTERNAL GUIDE
Mr. N. SRINIVASAN M.Tech
Assistant Professor

HEAD OF THE DEPARTMENT
Mrs. D. Salma Faroze M.Tech
Assistant Professor

PROJECT COORDINATOR
Mr. N. SRINIVASAN M.Tech
Assistant Professor

INTERNAL EXAMINER

EXTERNAL EXAMINER

Certificate

This is to certify that following team members **B.Tech (CSE)** (2019-23) studying final year from **CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY**, Proddatur, YSR Kadapa (dist.), A.P., (Affiliated to J.N.T.U – UNIVERSITY Anantapur, A.P., India), have been successfully completed their **ACADEMIC MAJOR PROJECT** titled "**EFFICIENT TRACEABLE AUTHORIZED SEARCH SYSTEM FOR SECURE CLOUD STORAGE**" by using **JAVA TECHNOLOGY** and other related tools under the guidance of this organization. The following is the list of students who were involved with the design, develop and deployment of the above-mentioned project.

S. No	Roll No	Name	Class
1	192P1A0595	P. Naga Lakshmi	CSE IV Year
2	192P1A0576	M. Mahaboob Basha	CSE IV Year
3	192P1A0596	P. Bindu Shruthika	CSE IV Year
4	192P1A0569	M. Karthik	CSE IV Year
5	192P1A0575	M. Mahendra Kumar Reddy	CSE IV Year

We offered them the complete project guidance & assistance. We place our appreciation on records for their commitment and hard work done during the design & development of this project and the project was completed to our best satisfaction.

Thanks & Regards,

Suman

Trainer

DECLARATION BY THE CANDIDATES

We are M. Karthik, M. Mahaboob Basha, P. Bindhu Shruthika, P. Naga Lakshmi, M. Mahendra Kumar Reddy with respective Roll No : (192P1A0569), (192P1A0576), (192P1A0596), (192P1A0595), (192P1A0575) here by declare that the Project Report entitled "EFFICIENT TRACEABLE AUTHORIZED SEARCH SYSTEM FOR SECURE CLOUD" under the guidance of Mr. N. SRINIVASAN M.Tech, Assistant Professor, Department of CSE is submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Computer Science & Engineering.

This is a record of bonafide word carried out by us and the results embodied in the Project Report have not been reproduced or copied from any source. The results embodied in this Project Report have not been submitted to any other University or Institute for the Award of any other Degree or Diploma.

P. Naga Lakshmi	192P1A0595
M. Mahaboob Basha	192P1A0576
P. Bindu Shruthika	192P1A0596
M. Karthik	192P1A0569
M. Mahendra Kumar Reddy	192P1A0575

Dept. of Computer Science & Engineering
Chaitanya Bharathi Institute of Technology
Vidyanagar, Proddatur, Y.S.R.(Dist.)

ACKNOWLEDGEMENT

An endeavour over a long period can be successful only with the advice and support of many well-wishers. We take this opportunity to express our gratitude and appreciation of all of them.

We are extremely thankful to our beloved Chairman **Dr. V. Jayachandra Reddy** who took keen interest and encouraged us in every effort throughout this course.

We owe our gratitude to our Principal **Dr. G. Sreenivasula Reddy** **M.Tech., (PhD).,** for permitting us to use facilities available to accomplish the project successfully.

We express our heartful thanks to **Mrs. D. Salma Faroze** **M.Tech** Head of Dept. CSE for his kind attention and valuable guidance to us throughout this course.

We also express our deep sense of gratitude towards **Mr. N. Srinivasan** **M.Tech**, Dept. CSE for his support and guidance in completing our project.

We express our profound respect and gratitude to our project coordinator **Mr. N. Srinivasan** **M.Tech**, Dept. CSE for his valuable support and guidance in completing the project successfully.

We are highly thankful to **MR. B. SUMAN, MR. G. SANDEEP,** Trylogic Software, Hyderabad, who have been kind enough to guide us in the preparation and execution of this project.

We also thank all the teaching & non-teaching staff of Dept. of CSE for their support throughout our B.Tech course.

We express our heartful thanks to our parents for their valuable support and encouragement in completion of our course. Also, we express our heartful regards to our friends for being supportive in completion of the project.

CONTENT

ABSTRACT

Secure search over encrypted remote data is crucial in cloud computing to guarantee the data privacy and usability. To prevent unauthorized data usage, fine-grained access control is necessary in multi-user system. However, authorized user may intentionally leak the secret key for financial benefit.

Thus, tracing and revoking the malicious user who abuses secret key needs to be solved imminently. In this project we propose an escrow free traceable attribute based multiple keywords subset search system with verifiable outsourced decryption (EF-TAMKS-VOD).

The key escrow free mechanism could effectively prevent the key generation centre (KGC) from unscrupulously searching and decrypting all encrypted files of users. Also, the decryption process only requires ultra lightweight computation, which is a desirable feature for energy-limited devices.

In addition, efficient user revocation is enabled after the malicious user is figured out. Moreover, the proposed system is able to support flexible number of attributes rather than polynomial bounded.

Flexible multiple keyword subset search pattern is realized, and the change of the query keywords order does not affect the search result. Security analysis indicates that EF-TAMKS-VOD is provably secure. Efficiency analysis and experimental results show that EF-TAMKS-VOD improves the efficiency and greatly reduces the computation overhead of users' terminals.

List Of Figures

List Of Screens

EFFICIENT TRACEABLE AUTHORIZED SEARCH SYSTEM FOR SECURE CLOUD STORAGE

CHAPTER 1
INTRODUCTION

With the development of new computing paradigm, cloud computing becomes the most notable one, which provides convenient, on-demand services from a shared pool of configurable computing resources. Therefore, an increasing number of companies and individuals prefer to outsource their data storage to cloud server. Despite the tremendous economic and technical advantages, unpredictable security and privacy concerns become the most prominent problem that hinders the widespread adoption of data storage in public cloud infrastructure.

Encryption is a fundamental method to protect data privacy in remote storage. However, how to effectively execute keyword search for plaintext becomes difficult for encrypted data due to the unreadability of ciphertext. Searchable encryption provides mechanism to enable keyword search over encrypted data for the file sharing system, such as multi-owner multiuser scenario, fine-grained search authorization is a desirable function for the data owners to share their private data with another authorized user.

However, most of the available systems require the user to perform a large amount of complex bilinear pairing operations. These overwhelmed computations become a heavy burden for user's terminal, which is especially serious for energy constrained devices. The outsourced decryption method allows user to recover the message with ultra lightweight decryption. However, the cloud server might return wrong half-decrypted information as a result of malicious attack or system malfunction.

Thus, it is an important issue to guarantee the correctness of outsourced decryption in public key encryption with keyword search (PEKS) system. The authorized entities may illegally leak their secret key to a third party for profits. Suppose that a patient someday suddenly finds out that a secret key corresponding his electronic medical data is sold on e-Bay. Such despicable behaviour seriously threatens the patient's data privacy.

Even worse, if the private electronic health data that contain serious health disease is abused by the insurance company or the patient's employment corporation, the patient would be declined to renew the medical insurance or labour contracts. The intentional secret key leakage seriously undermines the foundation of authorized access control and data privacy protection. Thus, it is extremely urgent to identify the malicious user or even prove it in a court of justice. In attribute-based access control system, the secret key of user is associated with a set of attributes rather than individual's identity.

As the search and decryption authority can be 2168-7161 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution for more information. This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2018.2820714, IEEE TRANSACTIONS ON CLOUD COMPUTING 2 shared by a set of users who own the same set of attributes, it is hard to trace the original key owner. Providing traceability to a fine-grained search authorization system is critical and not considered in previous searchable encryption systems. More importantly, in the original definition of PEKS scheme, key generation centre (KGC) generates all the secret keys in the system, which inevitably leads to the key escrow problem.

That is, the KGC knows all the secret keys of the users and thus can unscrupulously search and decrypt on all encrypted files, which is a significant threat to data security and privacy. Besides, the key escrow problem brings another problem when traceability ability is realized in PEKS. If a secret key is found to be sold and the identity of secret key's owner (i.e., the traitor) is identified, the traitor may claim that the secret key is leaked by KGC. There is no technical method to distinguish who is the true traitor if the key escrow problem is not solved.

1.1 Literature Review:

1.1.1

TITLE: Secure Ranked Keyword Search over Encrypted Cloud Data
AUTHORS: C. Wang, N. Cao, J. Li, K. Ren, W. Lou
CONTENT:

As Cloud Computing becomes prevalent, sensitive information are being increasingly centralized into the cloud. For the protection of data privacy, sensitive data has to be encrypted before outsourcing, which makes effective data utilization a very challenging task. Although traditional searchable encryption schemes allow users to securely search over encrypted data through keywords, these techniques support only Boolean search, without capturing any relevance of data files.

1.1.2

TITLE: Dual-Server Public-Key Encryption with Keyword Search for Secure Cloud Storage
AUTHORS: R. Chen, Y. Mu, G. Yang, F. Guo and X. Wang
CONTENT:

Searchable encryption is of increasing interest for protecting the data privacy in secure searchable cloud storage. In this paper, we investigate the security of a well-known cryptographic primitive, namely, public key encryption with keyword search (PEKS) which is very useful in many applications of cloud storage. Unfortunately, it has been shown that the traditional PEKS framework suffers from an inherent insecurity called inside keyword guessing attack (KGA) launched by the malicious server.

1.1.3

TITLE: Privacy-Preserving Double-Projection Deep Computation Model with Crowd sourcing on Cloud for Big Data Feature Learning

AUTHORS: Q. Zhang, L. T. Yang, Z. Chen, P. Li, M. J. Deen.

CONTENT:

Recent years have witnessed a considerable advance of Internet of Things with the tremendous progress of communication theories and sensing technologies. A large number of data, usually referring to big data, have been generated from Internet of Things. we present a double-projection deep computation model (DPDCM) for big data feature learning, which projects the raw input into two separate subspaces in the hidden layers to learn interacted features of big data by replacing the hidden layers of the conventional deep computation model (DCM) with double-projection layers.

1.1.4

TITLE: An Efficient Privacy-Preserving Outsourced Calculation Toolkit With Multiple Keys

AUTHORS: X. Liu, R.H. Deng, K.K.R. Choo, J. Weng.

CONTENT:

we propose a toolkit for efficient and privacy-preserving outsourced calculation under multiple encrypted keys (EPOM). Using EPOM, a large scale of users can securely outsource their data to a cloud server for storage. Moreover, encrypted data belonging to multiple users can be processed without compromising on the security of the individual user's (original) data and the final computed results.

1.1.5

TITLE: Building an encrypted and searchable audit log

AUTHORS: B. R. Waters, D. Balfanz, G. Durfee, and D. K. Smetters

CONTENT:

Audit logs are an important part of any secure system, and they need to be carefully designed in order to give a faithful representation of past system activity. This is especially true in the presence of adversaries who might want to tamper with the audit logs. While it is important that auditors can inspect audit logs to assess past system activity, the content of an audit log may contain sensitive information, and should therefore be protected from unauthorized parties. Protecting the contents of audit logs from unauthorized parties (i.e., encrypting it), while making it efficiently searchable by authorized auditors poses a problem. We describe an approach for constructing searchable encrypted audit logs which can be combined with any number of existing approaches for creating tamper-resistant logs.

CHAPTER 2
SYSTEM ANALYSIS

2.1 EXISTING SYSTEMS:

Most of the available systems require the user to perform a large amount of complex bilinear pairing operations. These overwhelmed computations become a heavy burden for user's terminal, which is especially serious for energy constrained devices.

The outsourced decryption method allows user to recover the message with Most ultra lightweight decryption. However, the cloud server might return wrong half-decrypted information as a result of malicious attack or system malfunction. The authorized entities may illegally leak their secret key to a third party for profits.

As a result, the data privacy will be at risk. so, it is extremely urgent to identify the malicious user or even prove it in a court of justice. However which is not possible to identify illegal leakage of secret keys.

DISADVANTAGES OF EXISTING SYSTEM:

- **Inflexible system extension**:

  Many existing authorization schemes are inflexible for the system extension. The attribute set needs to be predefined during the system establishment phase, and a maximum number of the attribute set should be determined.

  If a new attribute is to be added to the system, the entire system has to be re-constructed and all encrypted files have to be re-encrypted. It would be a disaster to the cloud storage system.

- **Inefficient decryption**:

  A main drawback of many ABE based fine-grained access control schemes is that the computation cost required for decryption grows linearly with the complexity of access structure.

- **Inefficient user revocation**:

  User revocation function is important for a multi-user cloud storage system. Most of the available searchable encryption schemes do not support this function.

2.2 PROPOSED SYSTEMS:

- **Flexible System Extension**:

  We propose an efficient system which supports flexible system extension, which accommodates flexible number of attributes.

  The attributes are not fixed in the system initialization phase and the size of attribute set is not restricted to polynomial bound, so that new attribute can be added to the system at any time.

  Moreover, the size of public parameter does not grow with the number of attributes. No matter how many attributes are supported in the system, no additional communication nor storage costs is brought to the system. This feature is desirable for the cloud system for its ever-increasing user volume.

- **Efficient Verifiable Decryption**.

  We propose a system which adopts the outsourced decryption mechanism to realize efficient decryption. Most of the decryption computation are outsourced to the cloud server, and the data user is able to complete the final decryption

- **Efficient User Revocation**.

  Once a user is identified as traitor through tracing algorithm, system revokes this malicious user from the authorized group. Compared with the existing scheme this revocation mechanism has much better efficiency.

ADVANTAGES OF PROPOSED SYSTEM:

- Flexible System Extension:
- Efficient Verifiable Decryption.
- Efficient User Revocation.
- Traceability of Abused Secret Key.

CHAPTER 3
REQUIREMENTS SPECIFICATION

3.1 HARDWARE REQUIREMENTS:

- Processor : Pentium Dual Core or later version
- Hard Disk : Minimum 500 GB
- Ram : Minimum 2GB.

3.2 SOFTWARE REQUIREMENTS:

- Operating system : Windows 7 or Later Version
- Coding Language : JAVA/J2EE
- Tool/IDE : Net beans 7.2.1 or later version
- Database : MYSQL

CHAPTER 4
SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE:

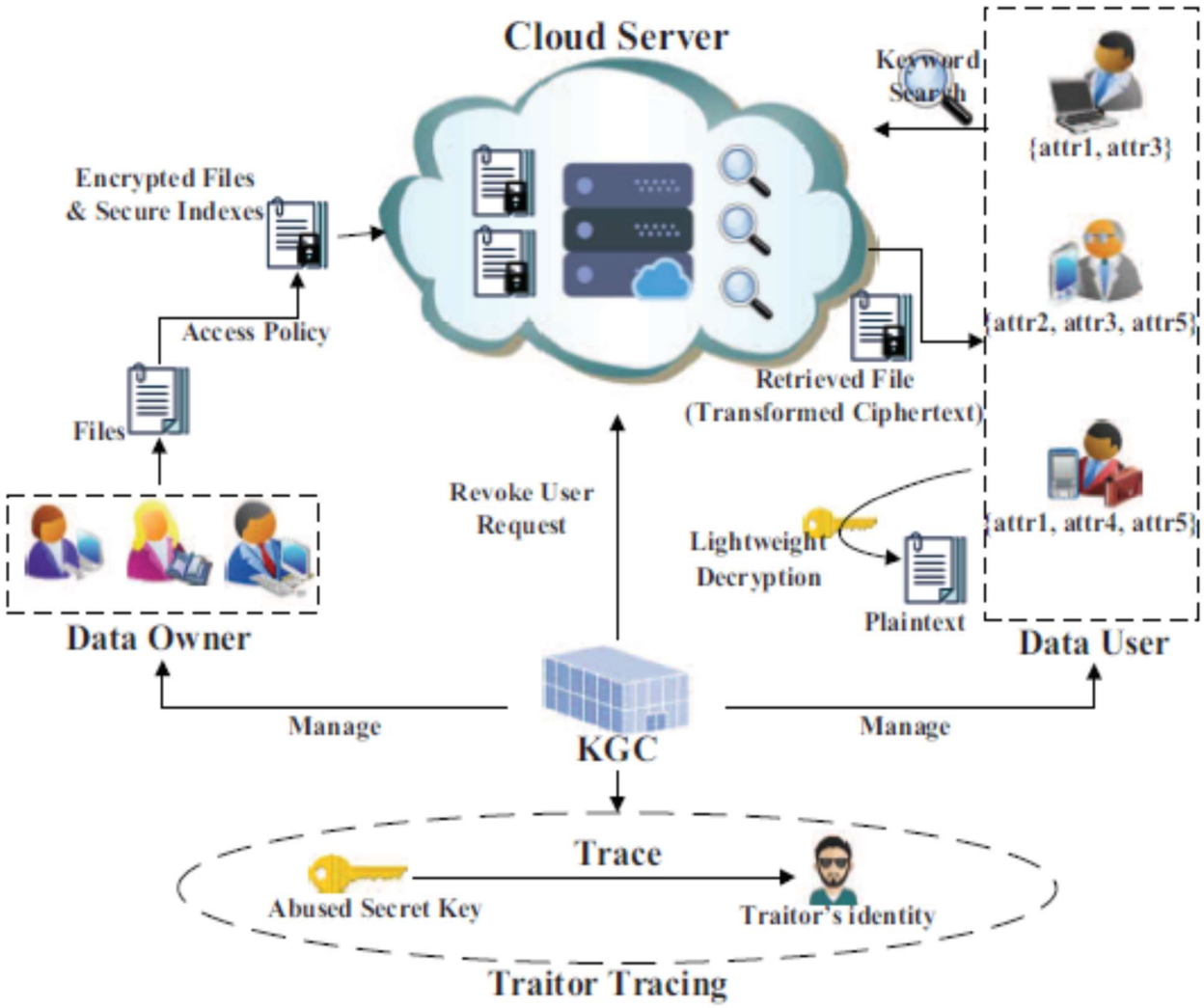


Fig-4.1: System Architecture

xxxxxxx

DATA FLOW DIAGRAM:

1. components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
2. The DFD is also called as bubble chart.
3. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
4. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These
5. DFD shows how the information moves through the system and how it is modified by a series of transformations.
6. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
7. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction.
8. DFD may be partitioned into levels that represent increasing information flow and functional detail.

4.2 BLOCK DIAGRAM:

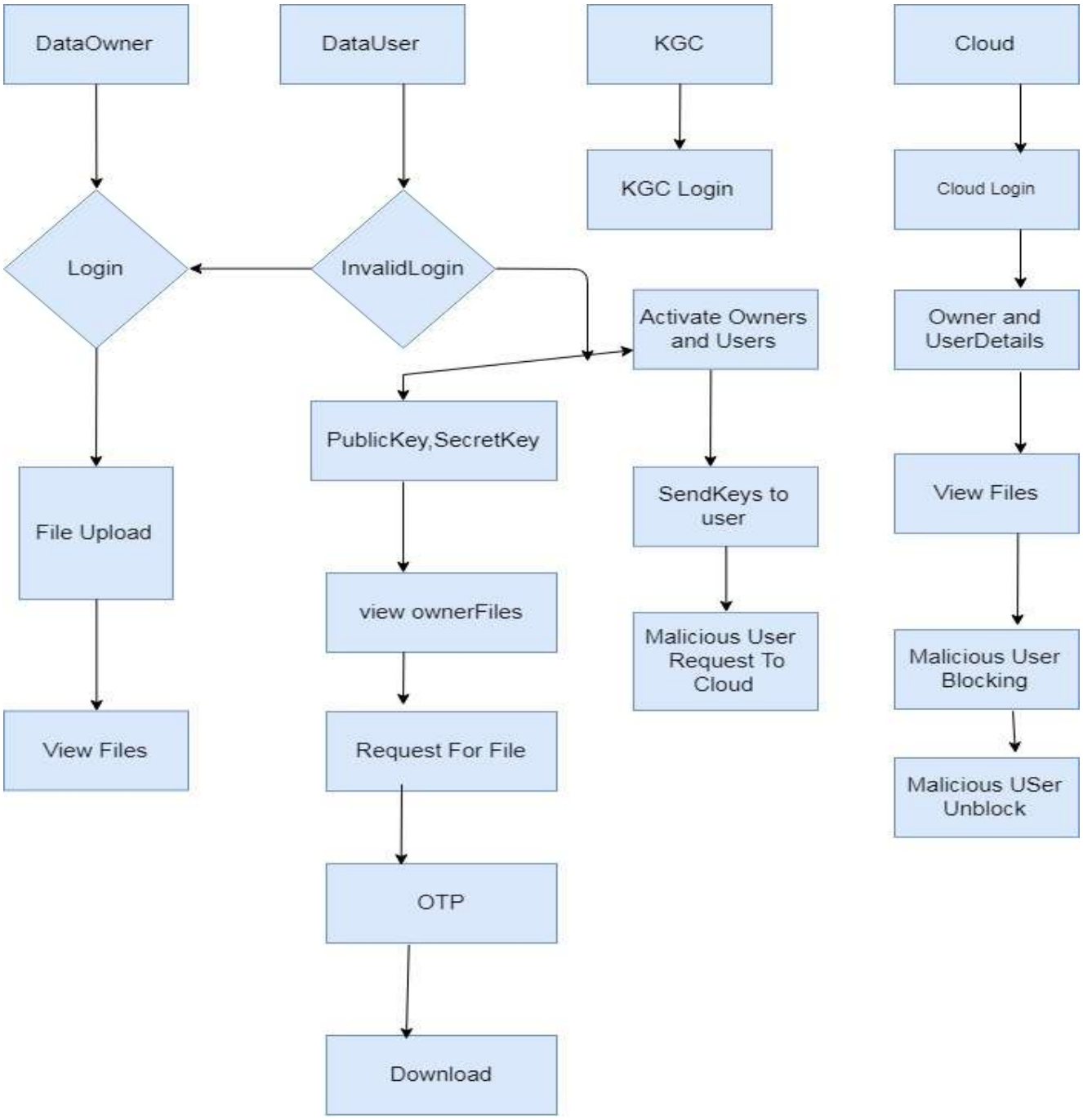


Fig-4.2: Block Diagram For Secure Cloud Storage

4.3 UML DIAGRAMS:

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

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

GOALS:

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

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.

4.3.1 USE CASE DIAGRAMS:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis.

Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

Data Owner:

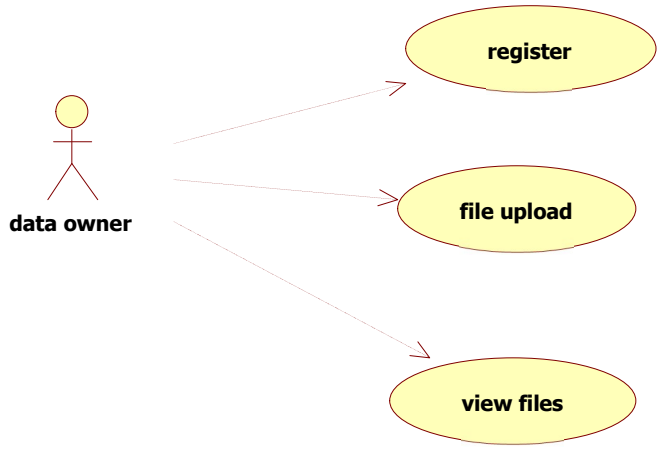


Fig-4.3.1.a: Use Case Diagram Of Data Owner For Secure Cloud Storage

Data User:

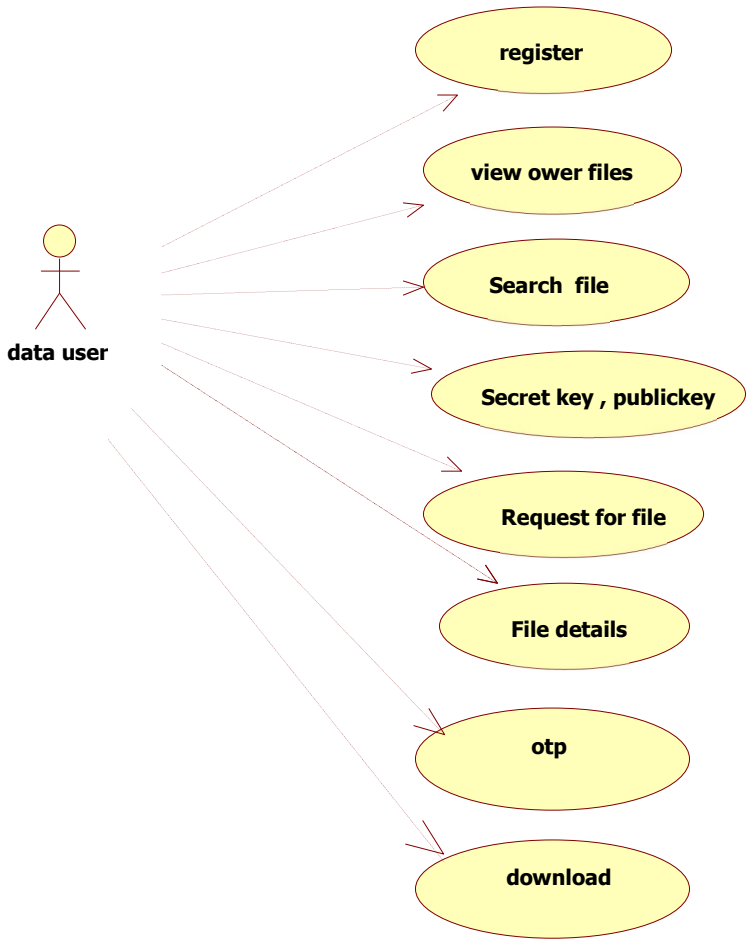


Fig-4.3.1.b: Use Case Diagram Of Data User For Secure Cloud Storage

KGC:

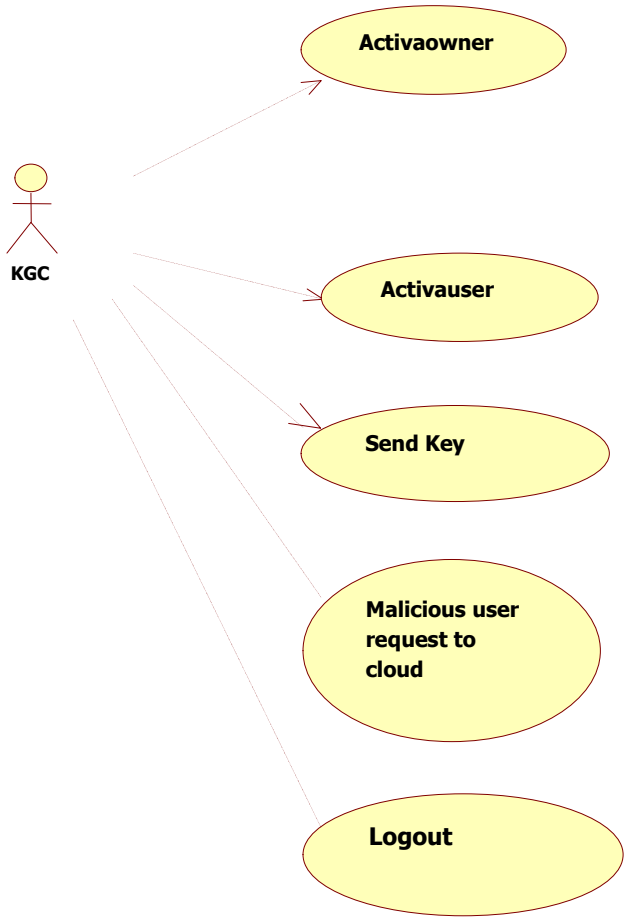


Fig-4.3.1.c: Use Case Diagram Of KGC For Secure Cloud Storage

Cloud:

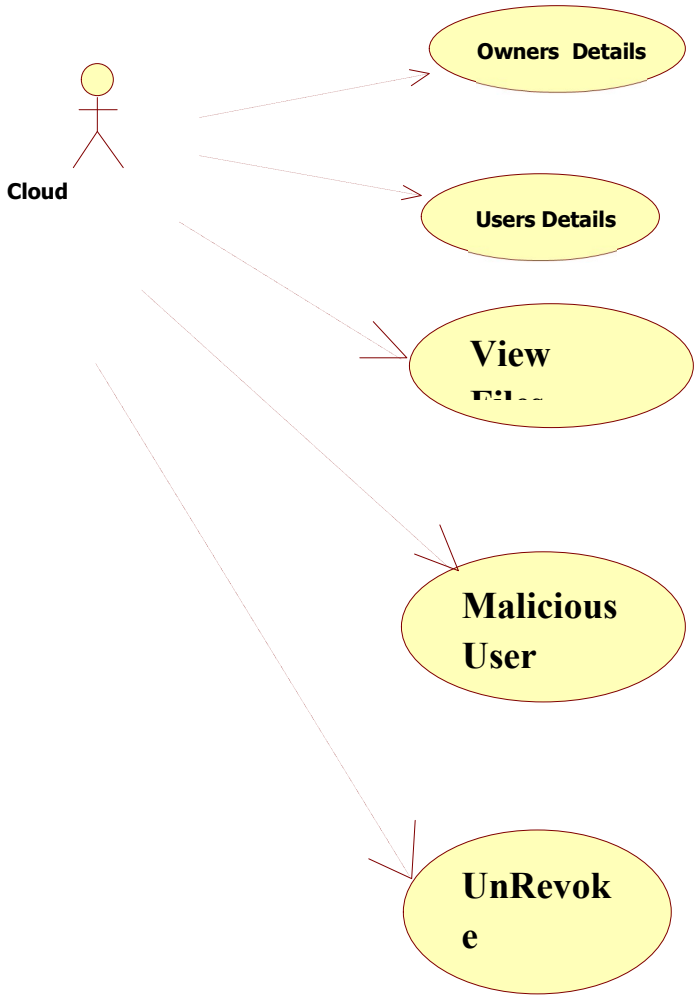


Fig-4.3.1.d: Use Case Diagram Of Cloud For Secure Cloud Storage

4.3.2 CLASS DIAGRAM:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

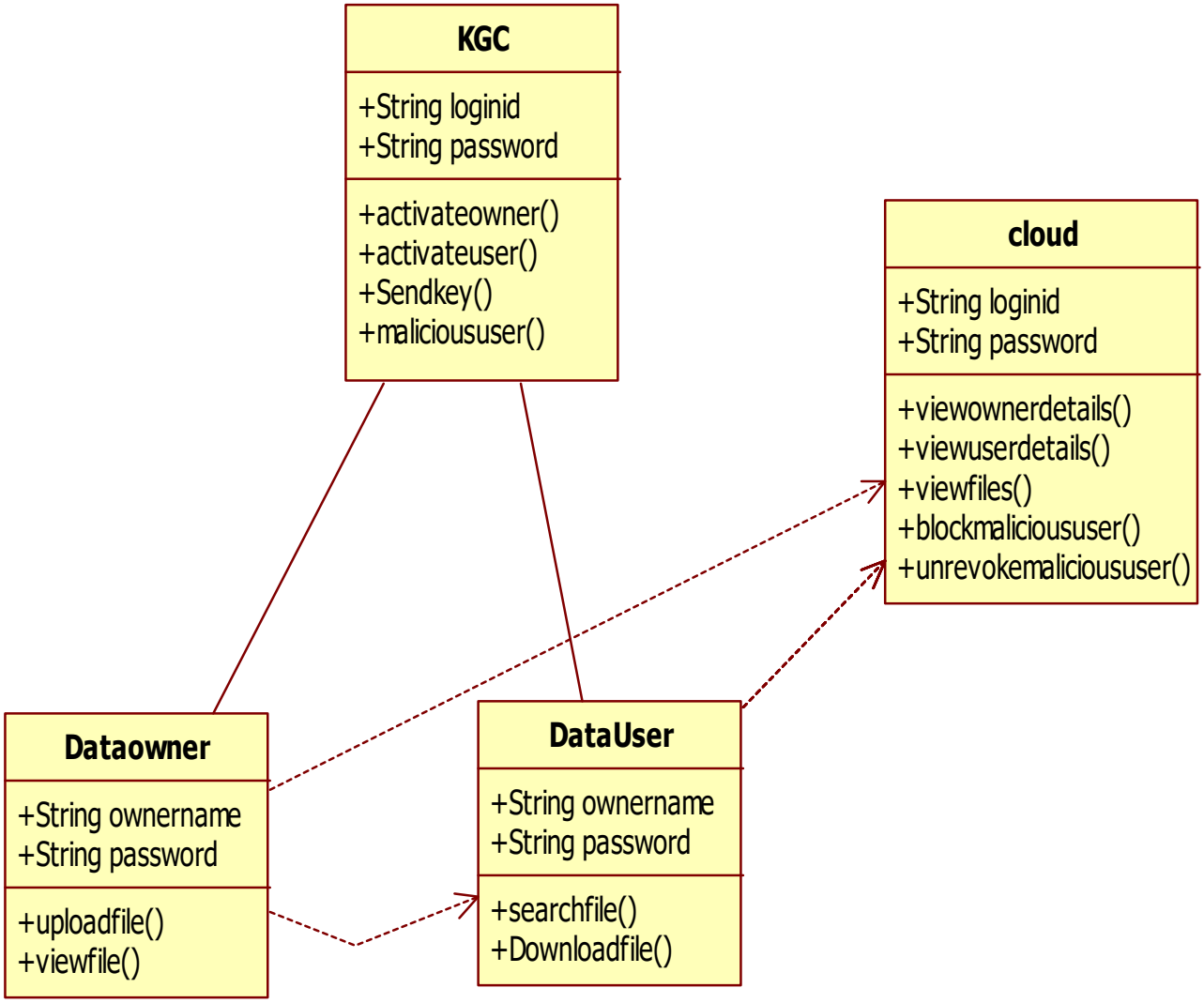


Fig-4.3.2: Class Diagram For Secure Cloud Storage

4.3.3 SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

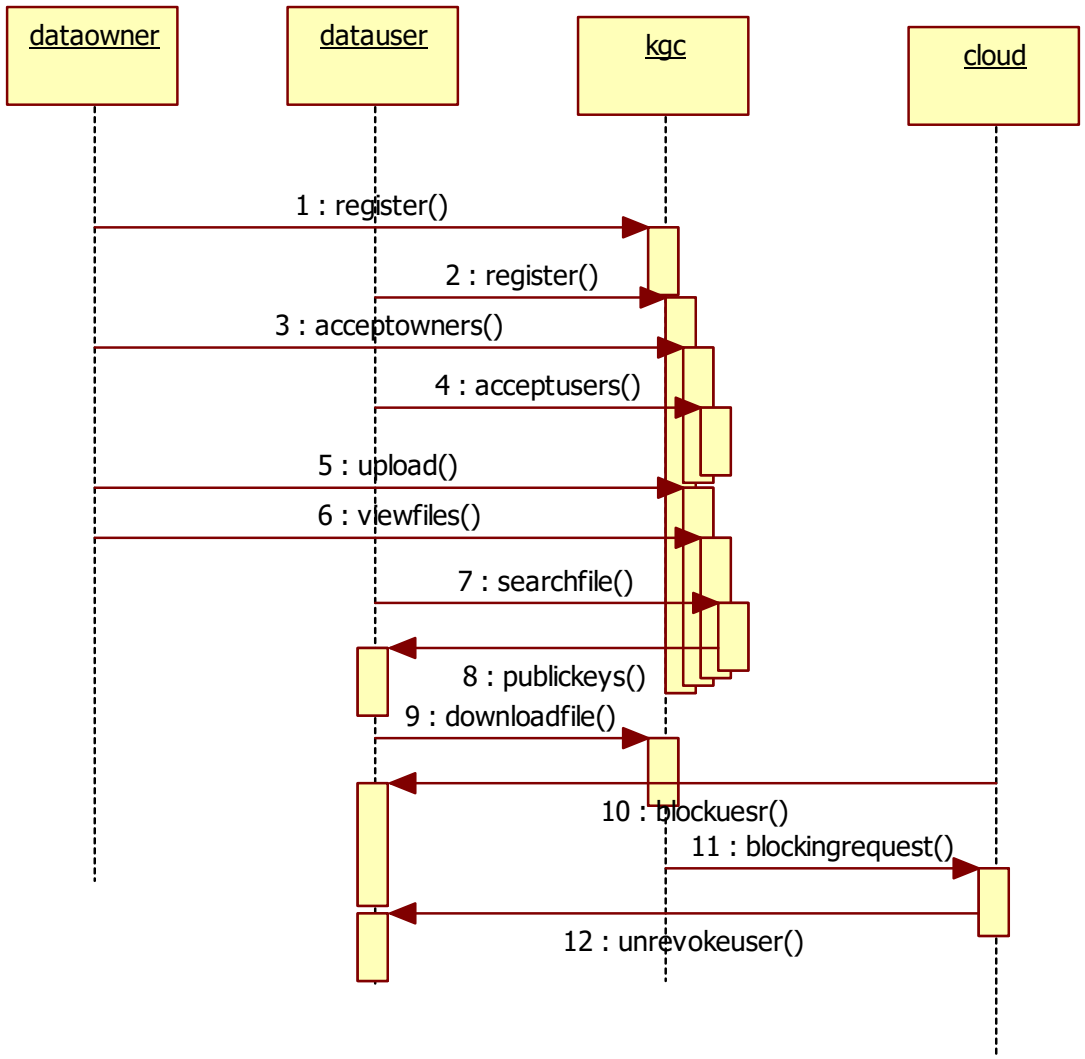


Fig-4.3.3: Sequence Diagram For Secure Cloud Storage

4.3.4 ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

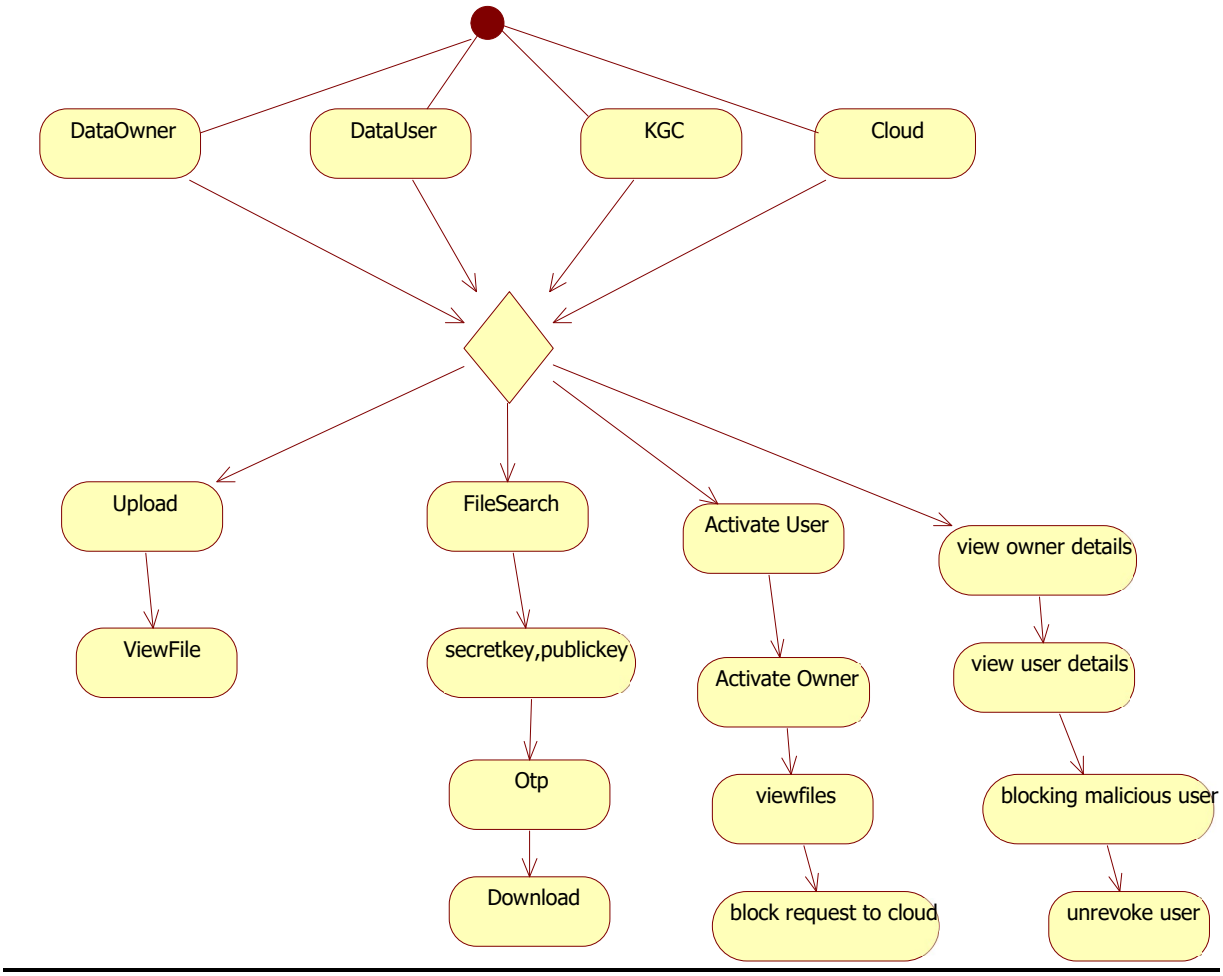
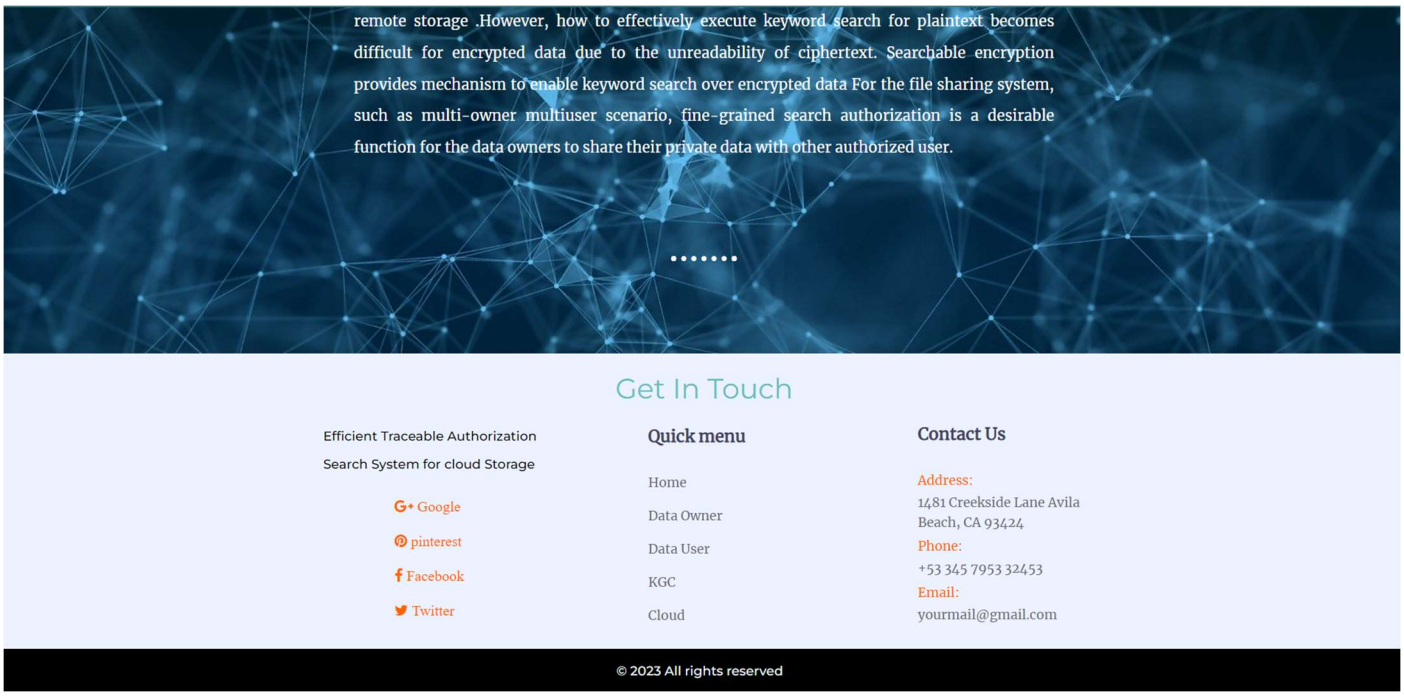


Fig-4.3.4: Activity Diagram For Secure Cloud Storage

CHAPTER 5

CODING AND IMPLEMENTING

Module I: Home Page



Home Page Implementation:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
<head>
<meta charset="utf-8">
<title>Home</title>
<link rel="stylesheet" href="css/styles.css">
<link rel="shortcut icon" href="favicon.ico">
<link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
```

```
<link
href="https://fonts.googleapis.com/css2?family=Merriweather&family=Montserrat&family=
Sacramento&display=swap" rel="stylesheet">

<link rel="stylesheet" type="text/css" href="styles/bootstrap4/bootstrap.min.css">

<link href="plugins/font-awesome-4.7.0/css/font-awesome.min.css" rel="stylesheet"
type="text/css">

<link href="plugins/video-js/video-js.css" rel="stylesheet" type="text/css">

<link rel="stylesheet" type="text/css" href="styles/about.css">

<link rel="stylesheet" type="text/css" href="styles/about_responsive.css">

</head>

<body style="background-image: url(images/image4.jpg);">

<header class="header" style="position: initial" >

<div class="header_container" style="background: none;">

<div class="container">

<div class="row">

<div class="col" style="padding: 0;">

<div class="header_content d-flex flex-row align-items-center justify-content-start">

<div class="logo_container">

<a href="index.html"></a>

</div>

<nav class="main_nav_contaner ml-auto">

<ul class="main_nav">

<li class=""><a href="index.html" style="border-radius: 50px;">home</a></li>

<li><a href="dataownerlogin.jsp" style="border-radius: 50px;">DATA OWNER</a></li>
```

USER

KGC

CLOUD

<!-- Hamburger -->

</nav>

<h2 style="font-family: 'Merriweather', 'Montserrat', sans-serif; color: white; font-weight: normal; padding-bottom: 10px; text-align: center; margin-top: 50px; margin-bottom: 40px; font-size: 3rem;"> Efficient Traceable Authorization Search System
 for cloud Storage </h2>

<hr style="border: dotted white 6px; border-bottom: none; width: 4%; margin: 10px auto;">

<h3 style="color: white; font-size: 2.5rem;">Abstract</h3>

<p style="text-align: justify; text-justify: inter-word; font-family: 'Merriweather', serif; color: white; font-size: 1rem;">Secure search over encrypted remote data is crucial in cloud computing to guarantee the data privacy and usability. To prevent unauthorized data usage, fine-grained access control is necessary in multi-user system. However, authorized user may intentionally leak the secret key for financial benefit. Thus, tracing and revoking the malicious user who abuses secret key needs to be solved imminently. In this paper, we propose an escrow free traceable attribute based multiple keywords subset search system with verifiable outsourced decryption (EF-TAMKS-VOD). The key escrow free mechanism could effectively prevent the key generation centre (KGC) from unscrupulously searching and decrypting all encrypted files of users. Also, the decryption process only requires ultra lightweight computation, which is a desirable feature for energy-limited devices. In addition, efficient user

revocation is enabled after the malicious user is figured out. Moreover, the proposed system is able to support flexible number of attributes rather than polynomial bounded.

<img style="width: 50%; float: right; margin-left: 30px;" class="chillie-img" src="images/image6.jpg" alt="">

<h3 style="color: white; font-size: 2.5rem;">Introduction</h3>

With the development of new computing paradigm, cloud computing becomes the most notable one, which provides convenient, on-demand services from a shared pool of configurable computing resources. Security and privacy concerns become the most prominent problem that hinders the widespread adoption of data storage in public cloud infrastructure. Encryption is a fundamental method to protect data privacy in remote storage. However, how to effectively execute keyword search for plaintext becomes difficult for encrypted data due to the unreadability of ciphertext. Searchable encryption provides mechanism to enable keyword search over encrypted data for the file sharing system, such as multi-owner multiuser scenario, fine-grained search authorization is a desirable function for the data owners to share their private data with another authorized user.

<hr style="border: dotted white 6px; border-bottom: none; width: 4%; margin: 100px auto;">

<footer class="footer" style="padding-top: 20px; padding-bottom: 0px; height: 320px; background-color: #ECF2FF; font-family: 'Merriweather';"><h2 style="font-family: 'Montserrat', sans-serif; color: #66BFBF; font-weight: normal; padding-bottom: 10px; text-align: center; font-size: 1.9rem;">Get In Touch</h2>

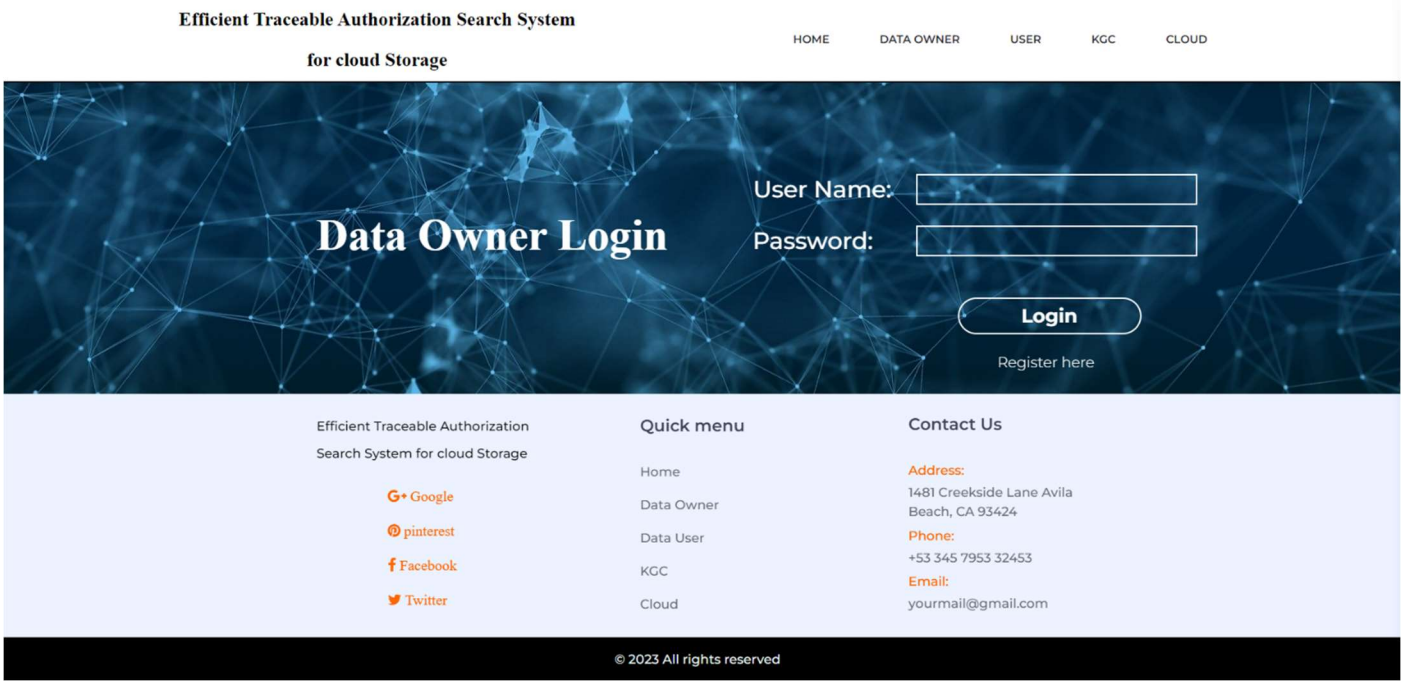
<div class="container" style=margin-left:10%;">

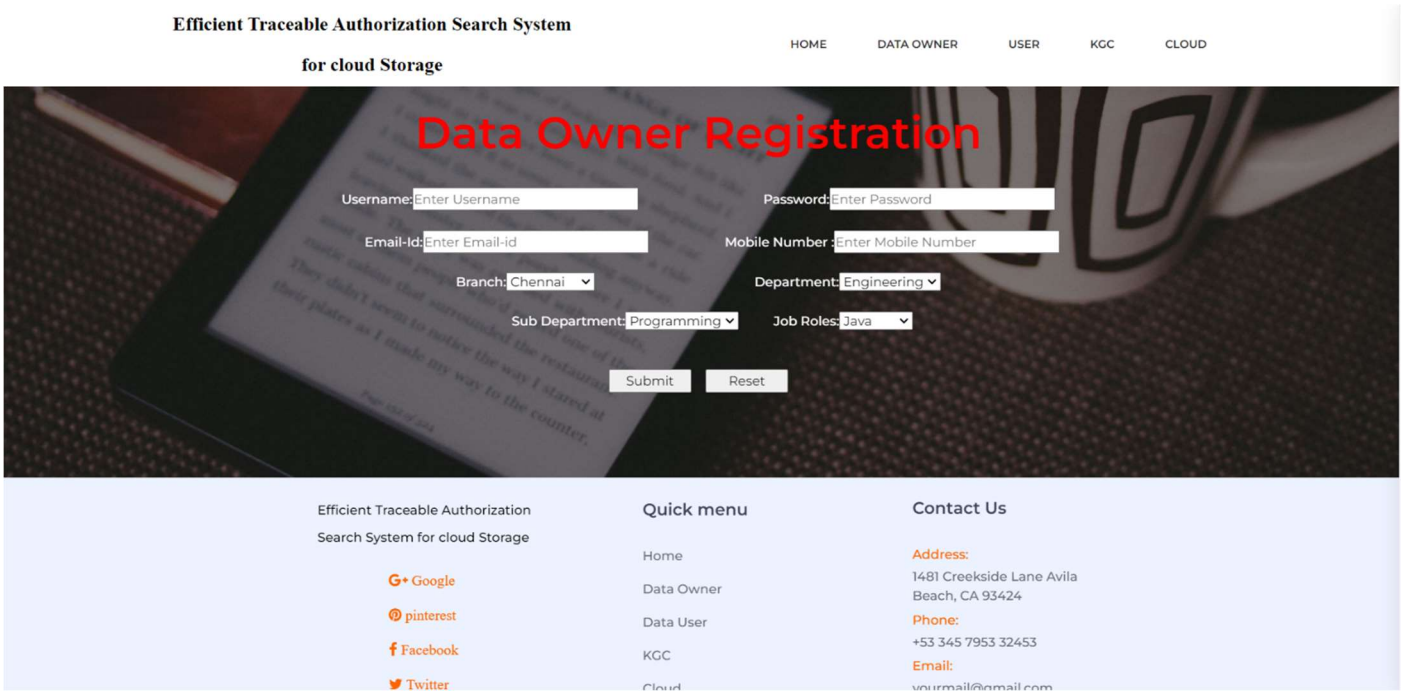
<!-- About -->

```
<div class="footer_about">
<div class="logo_container">
<a href="#">
<div class="footer_about_text" style="padding-top: 0px;">
<p style="padding-left: 20px; color: black;">Efficient Traceable Authorization Search System for cloud Storage</p>
<li><a href="x.html">Home</a></li>
<li><a href="dataownerlogin.jsp">Data Owner</a></li>
<li><a href="userlogin.jsp">Data User</a></li>
<li><a href="kgclogin.jsp">KGC</a></li>
<li><a href="CLoudlogin.jsp">Cloud</a></li>
<!--<li><a href="#">Facts</a></li>-->
</ul></div></div>
<div class="col-lg-3 footer_col">
<div class="footer_contact">
<div class="footer_title">Contact Us</div>
<div class="footer_contact_info">
<div class="footer_contact_item" style="margin-bottom: 5px;">
<div class="footer_contact_title">Address:</div>
<div class="footer_contact_line">1481 Creekside Lane Avila Beach, CA 93424</div>
</div>
</body>
</html>
```

EFFICIENT TRACEABLE AUTHORIZED SEARCH SYSTEM FOR SECURE CLOUD STORAGE

Module II: Data Owner Page





Data Owner Page Implementation:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Data Owner</title>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="description" content="Efficient Traceable project">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="styles/bootstrap4/bootstrap.min.css">
<link href="plugins/font-awesome-4.7.0/css/font-awesome.min.css" rel="stylesheet" type="text/css">
<link href="plugins/video-js/video-js.css" rel="stylesheet" type="text/css">
<link rel="stylesheet" type="text/css" href="styles/about.css">
<link rel="stylesheet" type="text/css" href="styles/about_responsive.css">
</head>
<body>
<div class="super_container">
<!-- Header -->
<header class="header" style="border-bottom: 2px solid black;">
<!-- Top Bar -->
<!-- Header Content -->
```

```
<div class="header_container">
<div class="container">
<div class="row">
<div class="col">
<div class="header_content d-flex flex-row align-items-center justify-content-start">
<div class="logo_container">
<a href="index.html">
<p style="font-family: 'Merriweather', serif; font-size: 1.3rem; color: black; text-align: center; font-weight: bold">
Efficient Traceable Authorization Search System <br> for cloud Storage
</p>
<!--<div class="logo_content d-flex flex-row align-items-end justify-content-start">
<div class="logo_img"><img src="images/logo.png" alt=""></div>
</div>--></a>div>
<nav class="main_nav_contaner ml-auto">
<ul class="main_nav">                                                    <ul class="main_nav">
<li class=""><a href="index.html" style="border-radius: 50px;">home</a></li>
li><a href="dataownerlogin.jsp" style="border-radius: 50px;">DATA OWNER</a></li>
li><a href="userlogin.jsp" style="border-radius: 50px;">USER</a></li>
<li><a href="KGClogin.jsp" style="border-radius: 50px;">KGC</a></li>
<li><a href="CLoudlogin.jsp" style="border-radius: 50px;">CLOUD</a></li>
</ul>
<!-- Hamburger -->
```

<div
class="menu_close">

<form action="#" class="header_search_form menu_mm">

<input type="search" class="search_input menu_mm" placeholder="Search"
required="required"><button class="header_search_button d-flex flex-column align-items-
center justify-content-center menu_mm">

</button></form>

<nav class="menu_nav">

<ul class="menu_mm">

<li class="menu_mm">Home

<li class="menu_mm">Courses

<li class="menu_mm">Instructors

<li class="menu_mm">Events

<li class="menu_mm">Blog

<li class="menu_mm">Contact

</nav>

phone:(009) 35475 6688933
32

follow us

 Google

 pinterest

 Facebook

 Twitter

Quick menu

<ul class="footer_list">

Home

Data Owner

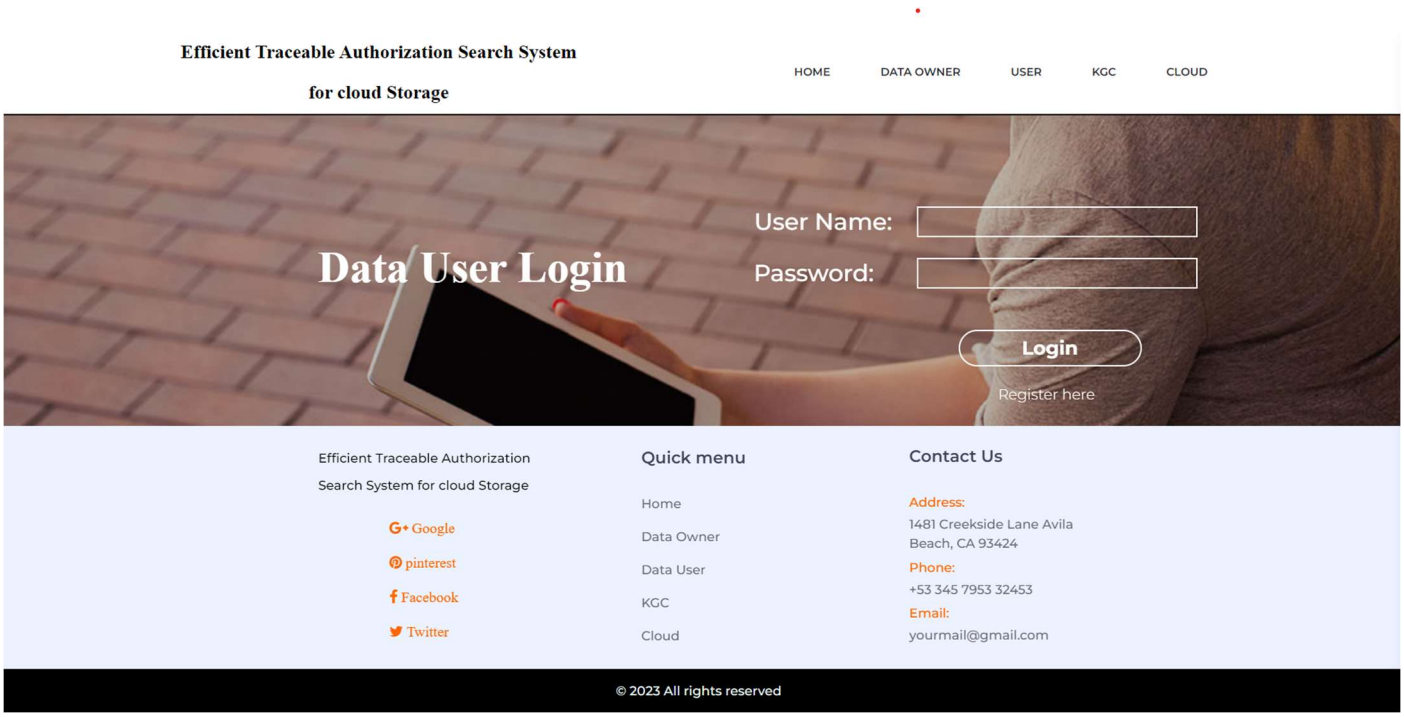
Address:

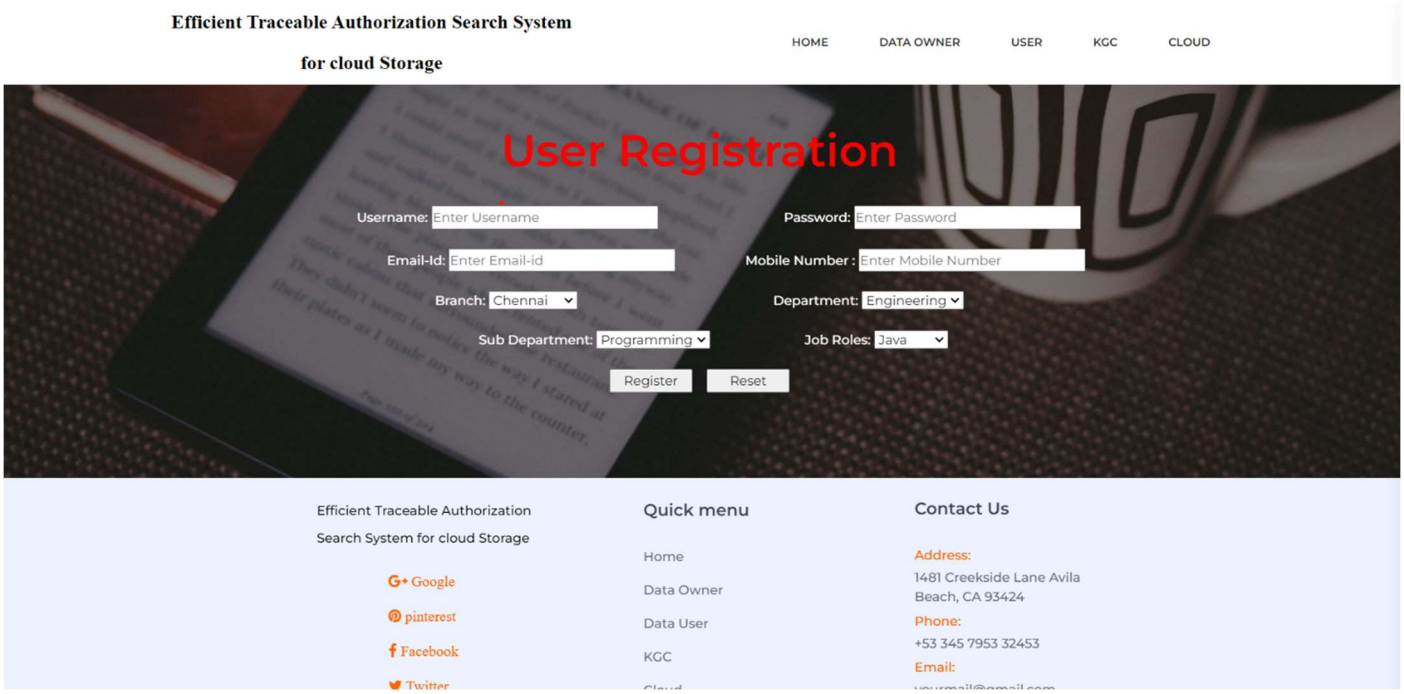
1481 Creekside Lane Avila Beach, CA 93424

Phone:

```
<div class="footer_contact_line">+53 345 7953 32453</div>
</div>
<div class="footer_contact_item" style="margin-bottom: 5px;">
<div class="footer_contact_title">Email:</div>
<div class="footer_contact_line">yourmail@gmail.com</div>
</div></div></div></div></div></div>
</footer>
<div style="background: #000000; height: 50px; width: 100%; ">
<h5 style="color: #EAF6F6; width: 190px; margin: auto; padding-top: 15px">
© <script>document.write(new Date().getFullYear());</script> All rights reserved
</h5></div>
<script src="plugins/scrollmagic/ScrollMagic.min.js"></script>
<script src="plugins/greensock/animation.gsap.min.js"></script>
<script src="plugins/greensock/ScrollToPlugin.min.js"></script>
<script src="plugins/easing/easing.js"></script>
<script src="plugins/parallax-js-master/parallax.min.js"></script>
<script src="js/about.js"></script>
</body>
</html>
```

Module III: Data User Page





Data User Page Implementation:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Data User</title>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="description" content="Efficient Traceable project">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="styles/bootstrap4/bootstrap.min.css">
<link href="plugins/font-awesome-4.7.0/css/font-awesome.min.css" rel="stylesheet" type="text/css">
<link href="plugins/video-js/video-js.css" rel="stylesheet" type="text/css">
<link rel="stylesheet" type="text/css" href="styles/about.css">
<link rel="stylesheet" type="text/css" href="styles/about_responsive.css">
</head>
<body>
<div class="super_container">
<!-- Header -->
<header class="header" style="border-bottom: 2px solid black;">
<!-- Header Content -->
<div class="header_container">
```

```
<div class="container">
<div class="row">
<div class="col">
<div class="header_content d-flex flex-row align-items-center justify-content-start">
<div class="logo_container">
<a href="index.html">
<p style="font-family: 'Merriweather', serif; font-size: 1.3rem; color: black; text-align: center; font-weight: bold">
Efficient Traceable Authorization Search System <br> for cloud Storage
</p></a></div>
<nav class="main_nav_contaner ml-auto">
<ul class="main_nav">
<ul class="main_nav">
<li class=""><a href="index.html" style="border-radius: 50px;">home</a></li>
<li><a href="dataownerlogin.jsp" style="border-radius: 50px;">DATA OWNER</a></li>
<li><a href="userlogin.jsp" style="border-radius: 50px;">USER</a></li>
<li><a href="KGClogin.jsp" style="border-radius: 50px;">KGC</a></li>
<li><a href="CLoudlogin.jsp" style="border-radius: 50px;">CLOUD</a></li>
</ul>
<!-- Hamburger -->
<div class="hamburger menu_mm">
<i class="fa fa-bars menu_mm" aria-hidden="true"></i>
</div></nav></div></div></div></div></div>
```

```
<li class="menu_mm"><a href="index.html">Home</a></li>
<li class="menu_mm"><a href="userlogin.jsp">Courses</a></li>
<li class="menu_mm"><a href="instructors.html">Instructors</a></li>
<li class="menu_mm"><a href="#">Events</a></li>
<li class="menu_mm"><a href="blog.html">Blog</a></li>
<li class="menu_mm"><a href="CLoudlogin.jsp">Contact</a></li>
</ul></nav>
<div class="menu_extra">
<div class="menu_phone"><span class="menu_title">phone:</span>(009) 35475 6688933 32</div>
<div class="menu_social">
<span class="menu_title">follow us</span>
<ul>
<li><a href="#"><i class="fa fa-pinterest" aria-hidden="true"></i></a></li>
<li><a href="#"><i class="fa fa-facebook" aria-hidden="true"></i></a></li>
<li><a href="#"><i class="fa fa-instagram" aria-hidden="true"></i></a></li>
<li><a href="#"><i class="fa fa-twitter" aria-hidden="true"></i></a></li>
</ul></div></div></div>
<!-- Home -->
<div class="home">
<div class="home_background parallax_background parallax-window" data-parallax="scroll" data-image-src="images/video.jpg" data-speed="0.8"></div>
<div class="home_content text-center">
```

Data User Login

<form action="user_login" method="post" style="height: 200px;">

<h5 style="color:white; font-size: 1.6rem; position: absolute; right: 30px; top: 40%; ">User Name:

<input style="margin-left: 20px; background: none; color: white; border: 2px solid white;" type="text" name="uname" autocomplete="off" />

</h5>

<h5 style="color:white; font-size: 1.6rem; position: absolute; right: 30px; bottom: 20%" >Password:

<input style="margin-left: 40px; background: none; color: white; border: 2px solid white;" type="password" name="upass"/>

</h5>

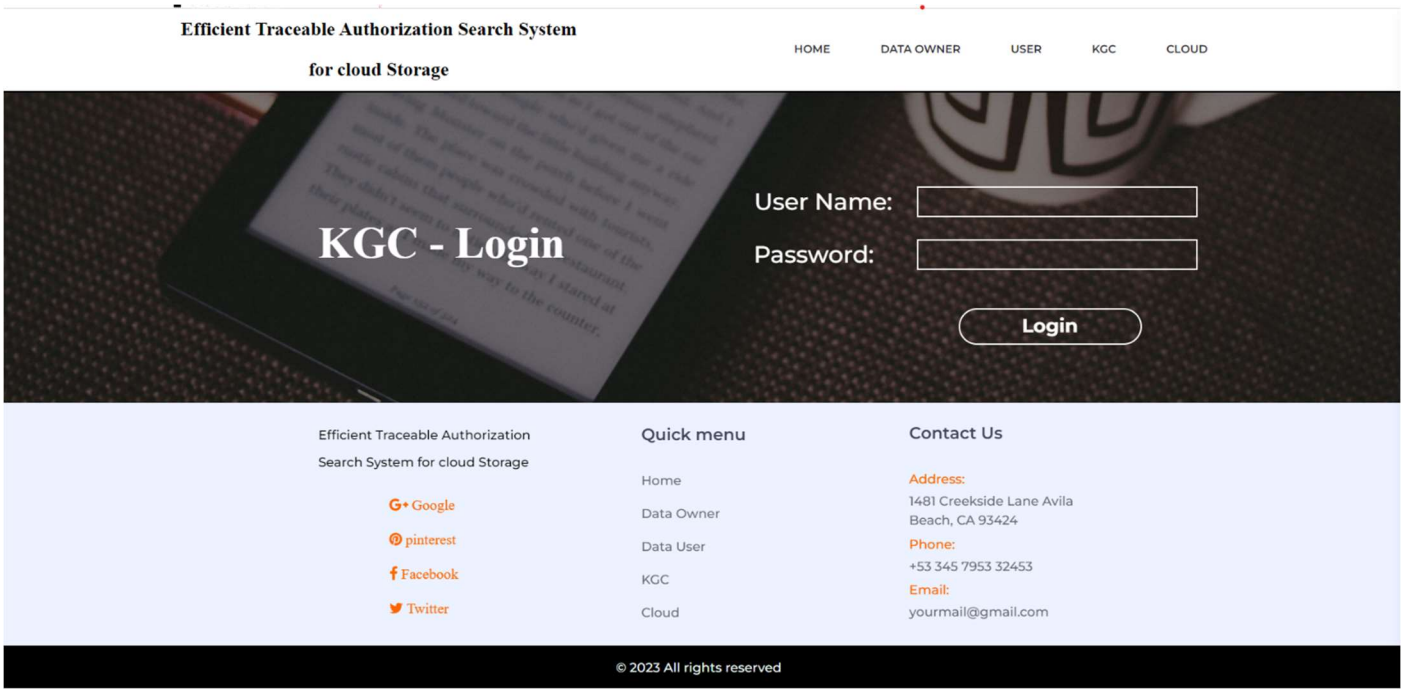
Efficient Traceable Authorization Search System for cloud Storage

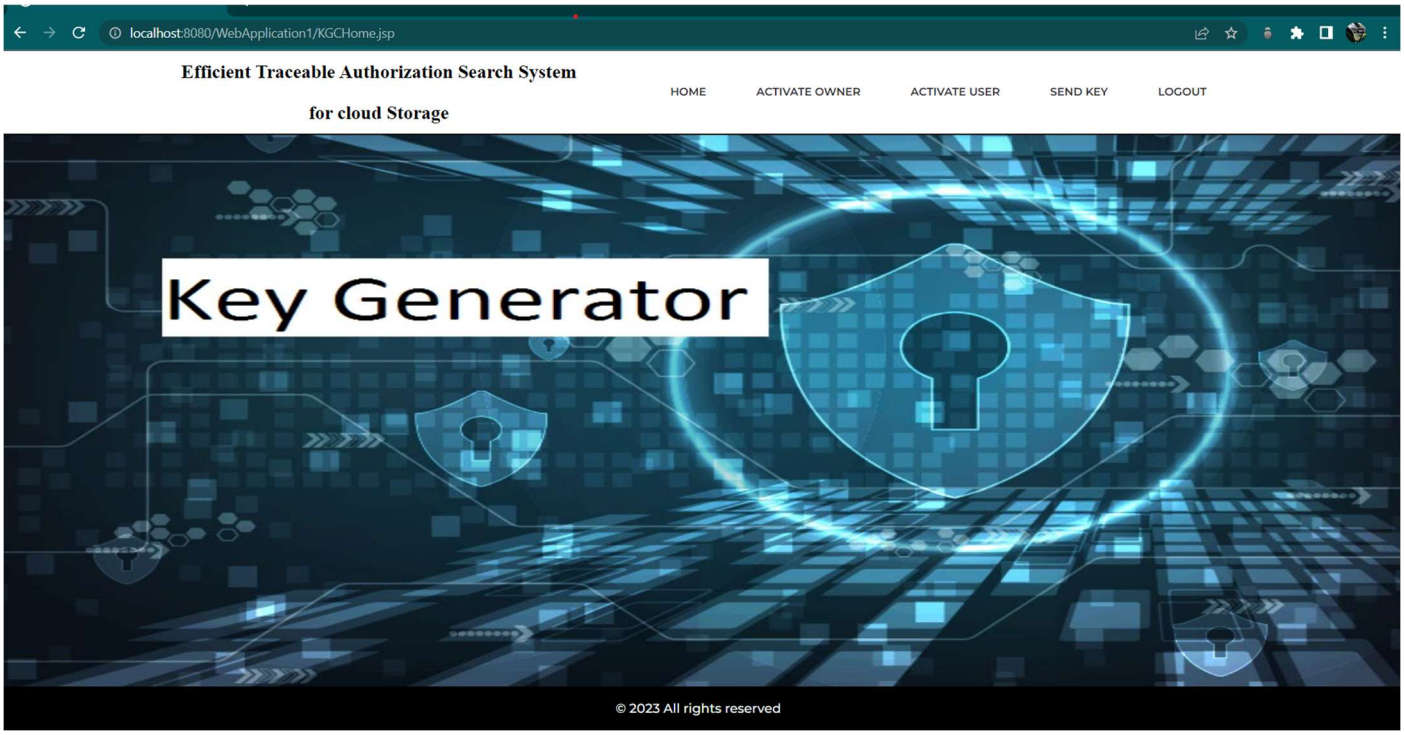
<ul style="display: flex; flex-direction: column; position: absolute; right: 25%;">

 Google

```
<li><a href="#"><i style="margin-bottom: 20px;" class="fa fa-pinterest" aria-
hidden="true"> pinterest</i></a></li>


<li><a href="userlogin.jsp">Data User</a></li>

<li><a href="KGClogin.jsp">KGC</a></li>

<li><a href="CLoudlogin.jsp">Cloud</a></li>

</div></div></div></div></div></div></footer>

<div style="background: #000000; height: 50px; width: 100%; ">

<h5 style="color: #EAF6F6; width: 190px; margin: auto; padding-top: 15px">

© <script>document.write(new Date().getFullYear());</script> All rights reserved

</h5></div></div>

<script src="js/jquery-3.2.1.min.js"></script>

<script src="styles/bootstrap4/popper.js"></script>

<script src="styles/bootstrap4/bootstrap.min.js"></script>

<script src="plugins/greensock/TweenMax.min.js"></script>

<script src="plugins/greensock/TimelineMax.min.js"></script>
```

Module IV: Key Generation Centre Page





Key Generation Centre Page Implementation:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title> KGC </title>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="description" content="Efficient Traceable project">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="styles/bootstrap4/bootstrap.min.css">
<link href="plugins/font-awesome-4.7.0/css/font-awesome.min.css" rel="stylesheet" type="text/css">
<link href="plugins/video-js/video-js.css" rel="stylesheet" type="text/css">
<link rel="stylesheet" type="text/css" href="styles/about.css">
<link rel="stylesheet" type="text/css" href="styles/about_responsive.css">
</head><body>
<div class="super_container">
<!-- Header -->
<header class="header" style="border-bottom: 2px solid black;">
<!-- Header Content -->
<div class="header_container">
<div class="container">
```

Efficient Traceable Authorization Search System
 for cloud Storage

<nav class="main_nav_contaner ml-auto">

<ul class="main_nav"> ul class="main_nav">

<li class="">home

DATA OWNER

USER

KGC

CLOUD

<form action="#" class="header_search_form">

<input type="search" class="search_input" placeholder="Search" required="required">

<button class="header_search_button d-flex flex-column align-items-center justify-content-center">

</button></form>

</header>

<!-- Menu -->

<form action="#" class="header_search_form menu_mm">

<input type="search" class="search_input menu_mm" placeholder="Search" required="required">

<button class="header_search_button d-flex flex-column align-items-center justify-content-center menu_mm">

</button></form>

<li class="menu_mm">Blog

<li class="menu_mm">Contact

</nav>

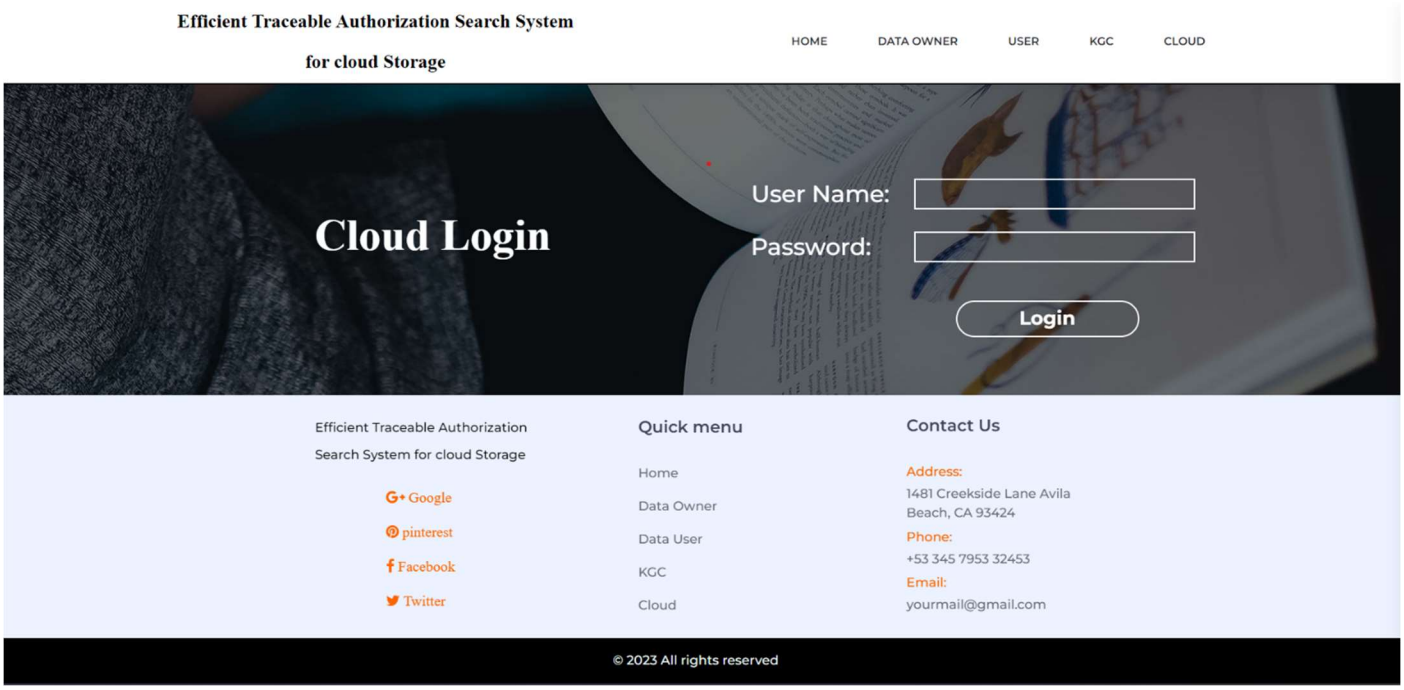
phone:(009) 35475 6688933 32

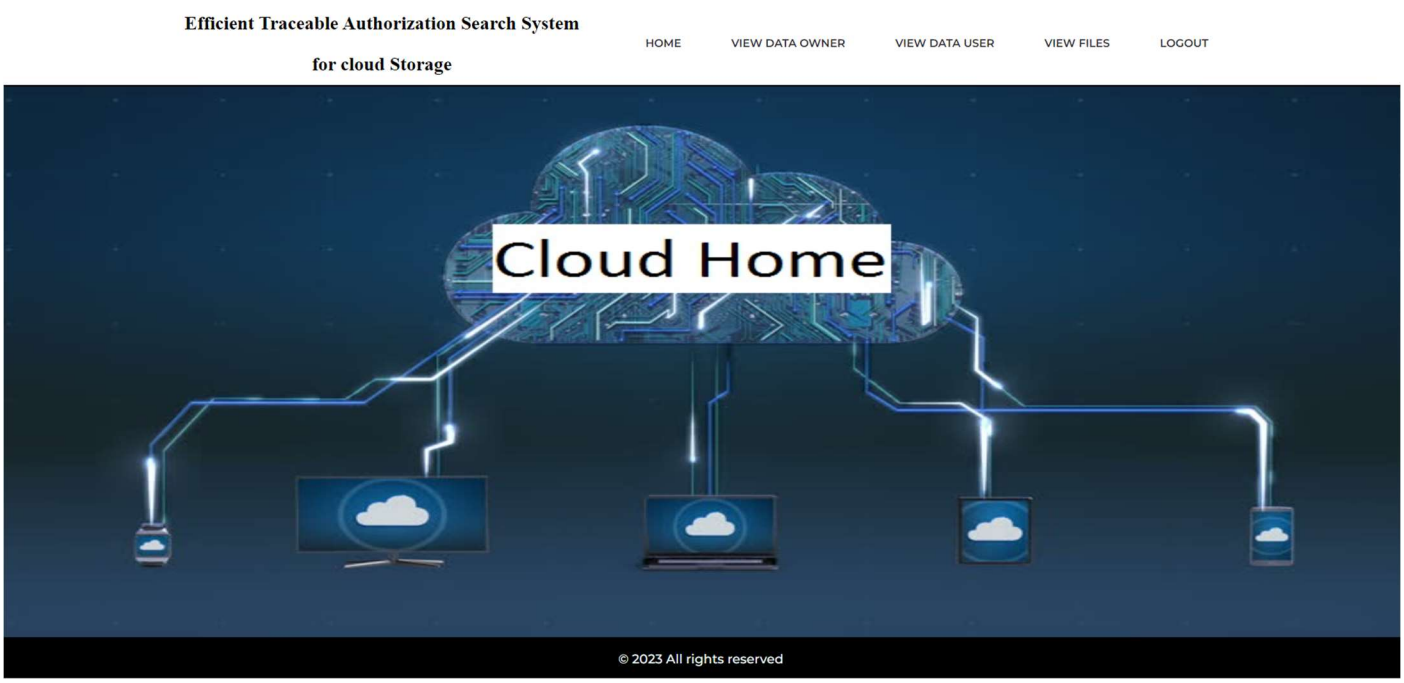
follow us


```
type="text" name="kname" autocomplete="off" />

</h5><br>

<h5 style="color:white; font-size: 1.6rem; position: absolute; right: 30px; bottom: 20%"
>Password:

<input style="margin-left: 40px; background: none; color: white; border: 2px solid white;"
type="password" name="kpass"/>

</h5>

<input style="position: absolute; top: 95%; right: 8%; width: 200px; border-radius: 20px;
height: 40px; background: none; color: white; font-size: 1.3rem; font-weight: bolder; border:
2px solid white;" type="submit" value="  Login  "/><br><br>

</form >

<%

String msgg=request.getParameter("msgg");

if(msgg !=null && msgg.equalsIgnoreCase("success")) {

left: 80px;">

<div class="footer_links">

<div class="footer_title">Quick menu</div>

<ul class="footer_list">

<li><a href="index.html">Home</a></li>

<li><a href="dataownerlogin.jsp">Data Owner</a></li>

<li><a href="userlogin.jsp">Data User</a></li>

<li><a href="KGClogin.jsp">KGC</a></li>

<li><a href="CLoudlogin.jsp">Cloud</a></li>

<!--<li><a href="#">Facts</a></li>-->
```

```
</ul></div></div>
Lane Avila Beach, CA 93424</div>
</div>
<div class="footer_contact_item" style="margin-bottom: 5px;">
<div class="footer_contact_title">Phone:</div>
<div class="footer_contact_line">+53 345 7953 32453</div>
</div>
<div class="footer_contact_item" style="margin-bottom: 5px;">
<div class="footer_contact_title">Email:</div>
<div class="footer_contact_line">yourmail@gmail.com</div>
</div></div></div></div></div></div></footer>
<div style="background: #000000; height: 50px; width: 100%; ">
<h5 style="color: #EAF6F6; width: 190px; margin: auto; padding-top: 15px">
© <script>document.write(new Date().getFullYear());</script> All rights reserved
</h5></div></div>
<script src="js/jquery-3.2.1.min.js"></script>
<script src="styles/bootstrap4/popper.js"></script>
<script src="styles/bootstrap4/bootstrap.min.js"></script>
<script src="plugins/greensock/TweenMax.min.js"></script>
</body>
</html>
```

Module V: Cloud Page





Cloud Page Implantation:

```
<!DOCTYPE html>
<html lang="en">
<head>
<title> Cloud </title>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="description" content="Efficient Traceable project">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" type="text/css" href="styles/bootstrap4/bootstrap.min.css">
<link href="plugins/font-awesome-4.7.0/css/font-awesome.min.css" rel="stylesheet" type="text/css">
<link href="plugins/video-js/video-js.css" rel="stylesheet" type="text/css">
<link rel="stylesheet" type="text/css" href="styles/about.css">
<link rel="stylesheet" type="text/css" href="styles/about_responsive.css">
</head><body>
<div class="header_content d-flex flex-row align-items-center justify-content-start">
<div class="logo_container">
<a href="index.html">
<p style="font-family: 'Merriweather', serif; font-size: 1.3rem; color: black; text-align: center; font-weight: bold">
Efficient Traceable Authorization Search System <br> for cloud Storage
```

</nav>

<button class="header_search_button d-flex flex-column align-items-center justify-content-center">

</button></form>

</header>

<!-- Menu -->

<form action="#" class="header_search_form menu_mm">

<input type="search" class="search_input menu_mm" placeholder="Search" required="required">

<button class="header_search_button d-flex flex-column align-items-center justify-content-center menu_mm">

</button></form> <nav class="menu_nav">

<ul class="menu_mm">

<li class="menu_mm">Home

<li class="menu_mm">Courses

<li class="menu_mm">Instructors

```
<li class="menu_mm"><a href="CLoudlogin.jsp">Contact</a></li>

</ul></nav><div class="menu_extra">

<div class="menu_phone"><span class="menu_title">phone:</span>(009) 35475 6688933 32</div>

<div class="menu_social">

<span class="menu_title">follow us</span>

<div class="home_content text-center">

<input style="margin-left: 20px; background: none; color: white; border: 2px solid white;" type="text" name="cname" autocomplete="off" />

</h5><br><h5 style="color:white; font-size: 1.6rem; position: absolute; right: 30px; bottom: 20%" >Password:

<input style="margin-left: 40px; background: none; color: white; border: 2px solid white;" type="password" name="cpass"/>

</h5>

<!--        
       <input type="reset" value="  Reset "/>          -->

<input style="position: absolute; top: 95%; right: 8%; width: 200px; border-radius: 20px; height: 40px; background: none; color: white; font-size: 1.3rem; font-weight: bolder; border: 2px solid white;" type="submit" value="  Login  "/><br><br>

</form >

<%

String msg=request.getParameter("msg");

if(msg !=null && msg.equalsIgnoreCase("failed")) {

out.println("<font color='Red'><b>Invalid Login Credentials</b></font>"); }
```

```
<div class="col-lg-3 footer_col">

<div class="footer_about">

<div class="logo_container">

<a href="#"></a>

</div><div class="footer_about_text" style="padding-top: 0px;">

<p style="padding-left: 20px; color: black;">Efficient Traceable Authorization Search System for cloud Storage</p>

</div><div class="footer_social">

<ul style="display: flex; flex-direction: column; position: absolute; right: 25%;">

<li><a href="#"><i style="margin-bottom: 20px;" class="fa fa-google-plus" aria-hidden="true"> Google</i></a></li>

<li><a href="#"><i style="margin-bottom: 20px;" class="fa fa-pinterest" aria-hidden="true"> pinterest</i></a></li>

<li><a href="#"><i style="margin-bottom: 20px;" class="fa fa-facebook" aria-hidden="true"> Facebook</i></a></li>


<li><a href="index.html">Home</a></li>

<li><a href="dataownerlogin.jsp">Data Owner</a></li>

<li><a href="userlogin.jsp">Data User</a></li>

<li><a href="KGClogin.jsp">KGC</a></li>

<li><a href="CLoudlogin.jsp">Cloud</a></li>

<!--<li><a href="#">Facts</a></li>-->

</ul></div></div>

<div class="col-lg-3 footer_col">
```

```
<div class="footer_contact">
<div class="footer_title">Contact Us</div>
<div class="footer_contact_info">
<div class="footer_contact_item" style="margin-bottom: 5px;">
<div class="footer_contact_title">Address:</div>
<div class="footer_contact_line">1481 Creekside Lane Avila Beach, CA 93424</div>
</div><div class="footer_contact_item" style="margin-bottom: 5px;">
<div class="footer_contact_title">Phone:</div>
<div class="footer_contact_line">+53 345 7953 32453</div>
</div><div class="footer_contact_item" style="margin-bottom: 5px;">
<div class="footer_contact_title">Email:</div>
<script src="styles/bootstrap4/bootstrap.min.js"></script>
<script src="plugins/greensock/TweenMax.min.js"></script>
<script src="plugins/greensock/TimelineMax.min.js"></script>
<script src="plugins/scrollmagic/ScrollMagic.min.js"></script>
<script src="plugins/greensock/animation.gsap.min.js"></script>
<script src="plugins/greensock/ScrollToPlugin.min.js"></script>
<script src="plugins/easing/easing.js"></script>
<script src="plugins/parallax-js-master/parallax.min.js"></script>
<script src="js/about.js"></script>
</body>
</html>
```

CHAPTER 6
TESTING

INTRODUCTION:

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

TYPES OF TESTS:

6.1 Unit Testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive.

Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2 Integration Testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.3 Functional Testing:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is cantered on the following items:

Valid Input	:	identified classes of valid input must be accepted.
Invalid Input	:	identified classes of invalid input must be rejected.
Functions	:	identified functions must be exercised.
Output	:	identified classes of application outputs must be exercised.
Systems/Procedures	:	interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

6.4 System Testing:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

6.5 White Box Testing:

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

6.6 Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document.

6.7 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER 7

RESULTS

OWNER REGISTRATION:

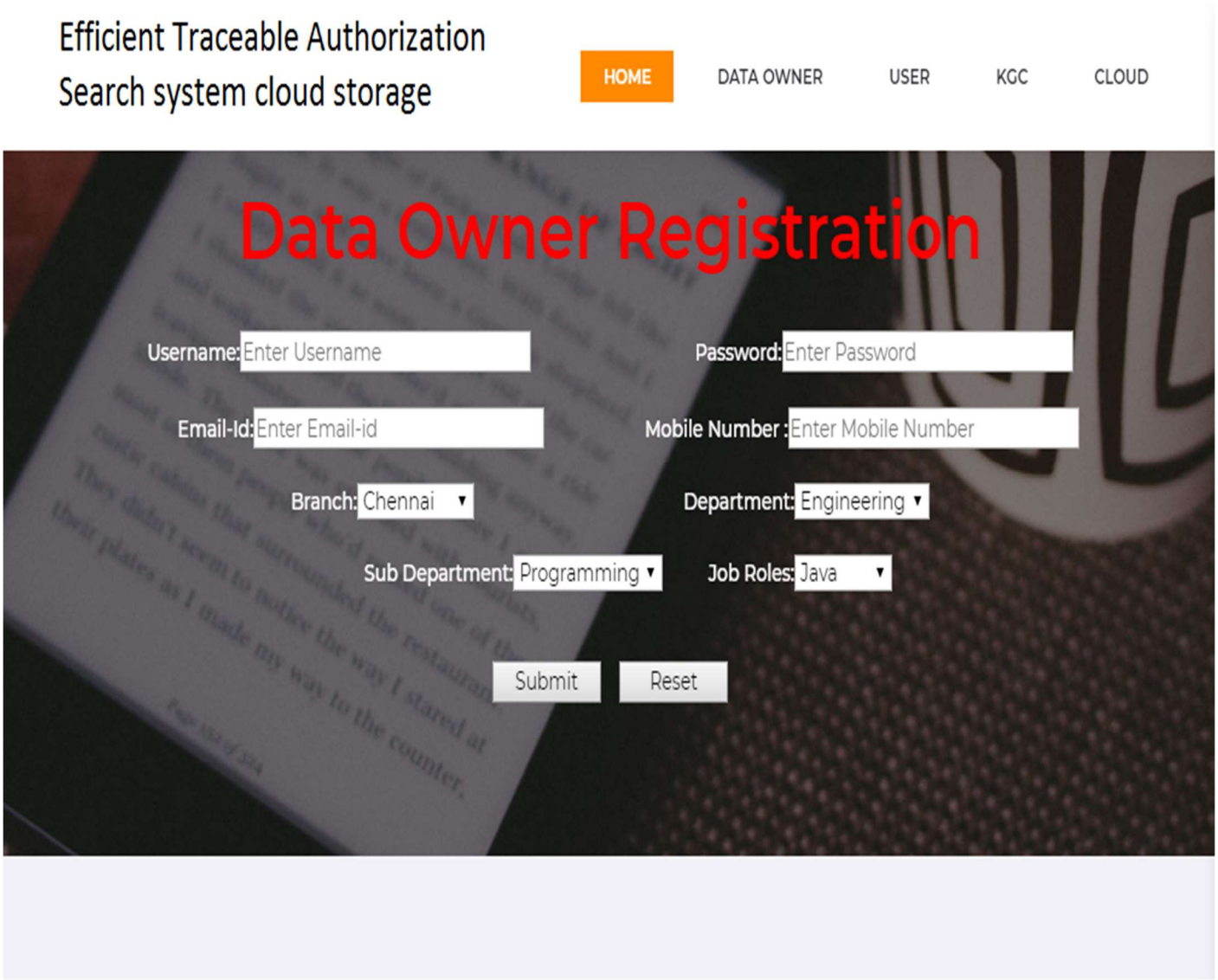


Fig-7.1: Owner Registration Page

USER REGISTRATION:

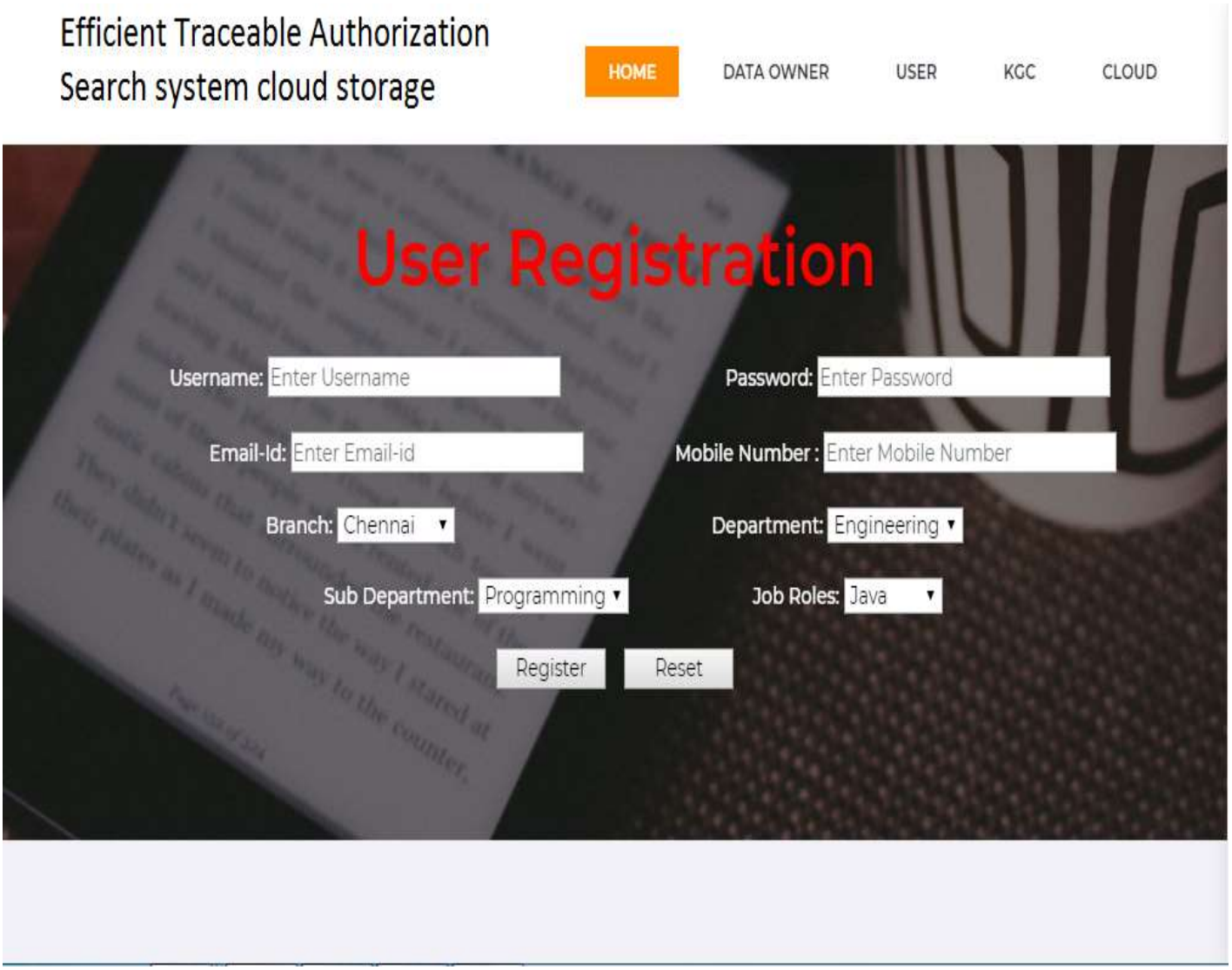


Fig-7.2: User Registration Page

OWNER LOGIN:

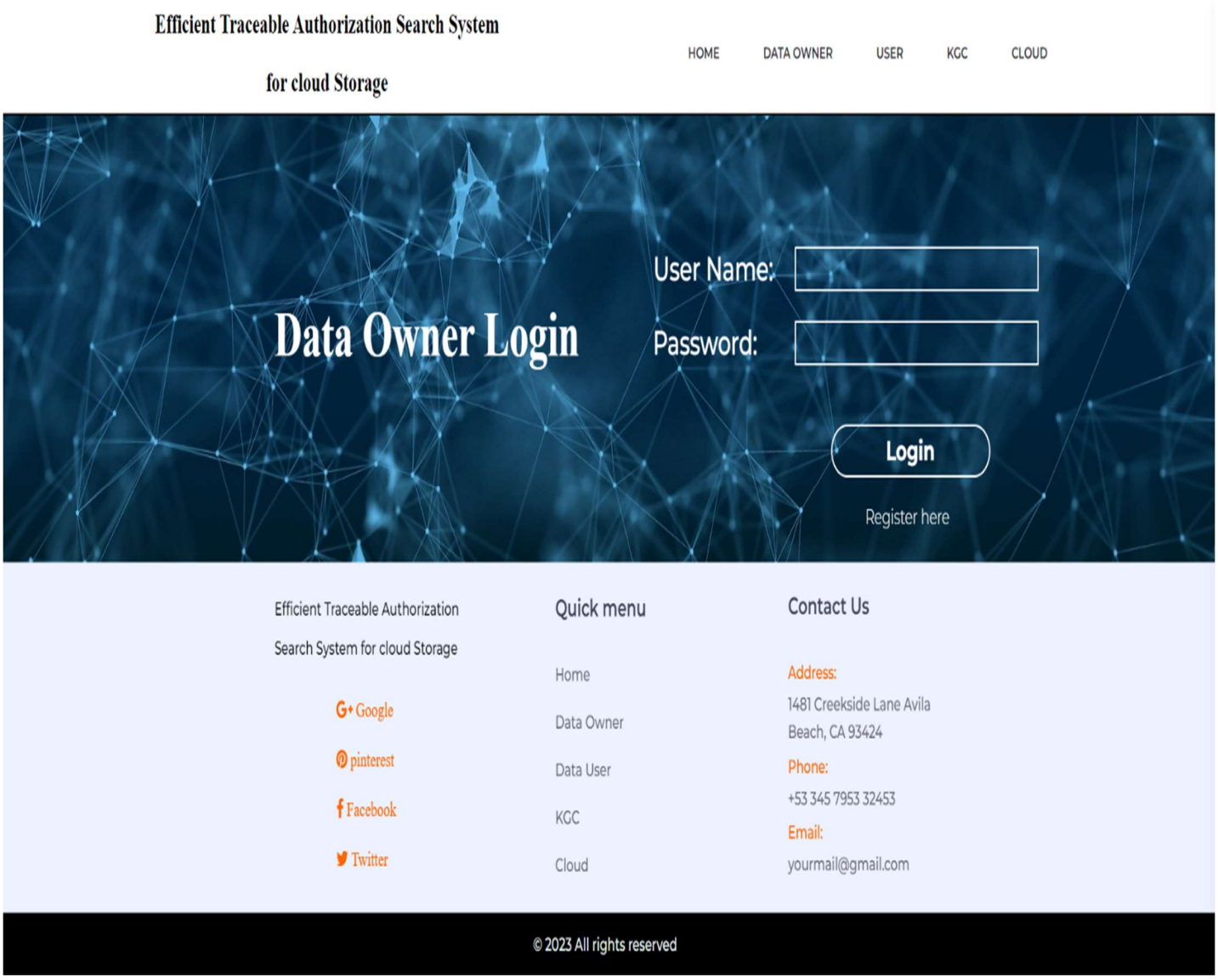


Fig-7.3: Owner Login Page

USER LOGIN:

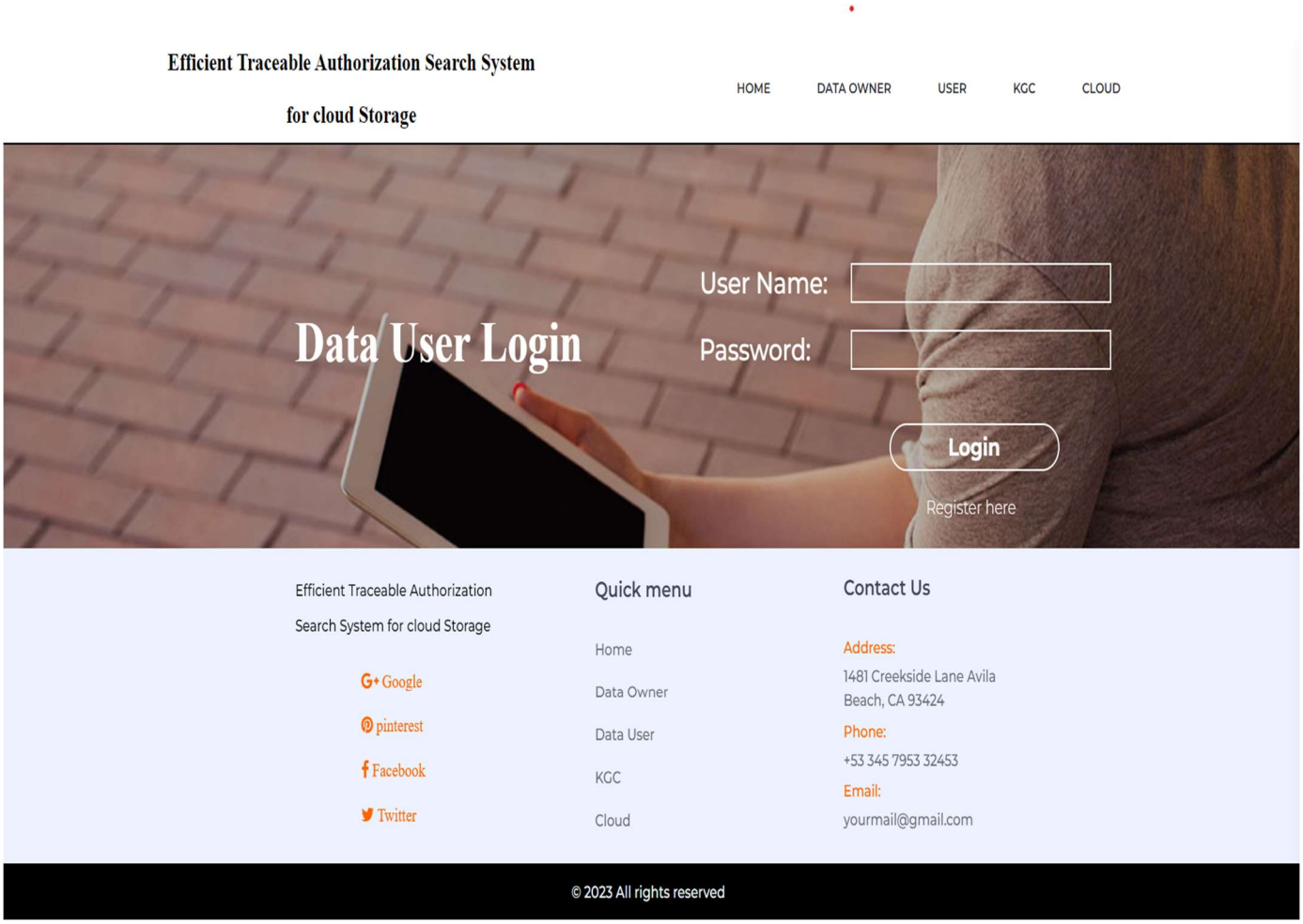


Fig-7.4: User Login Page

KGC LOGIN:

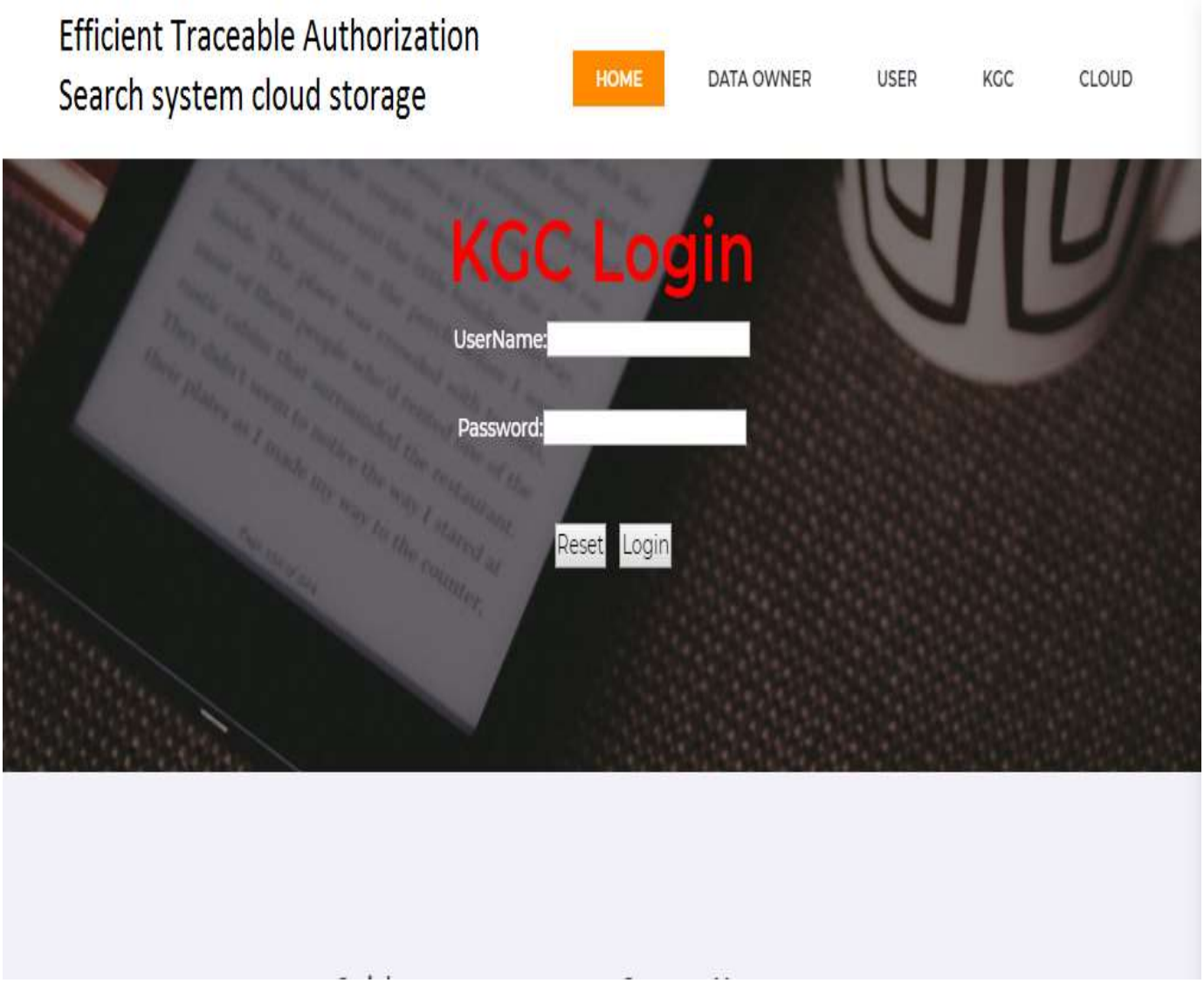


Fig-5: KGC Login Page

CLOUD LOGIN:

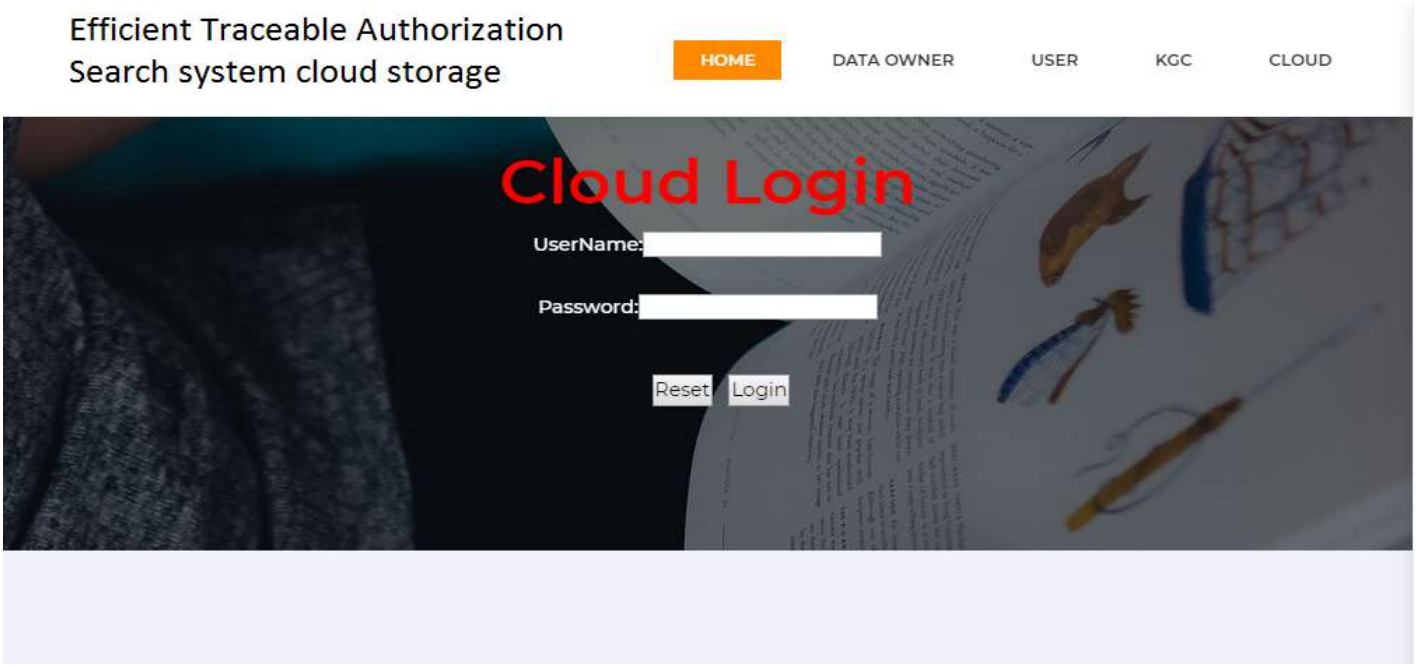


Fig-7.6: Cloud Login Page

FILE UPLOAD:

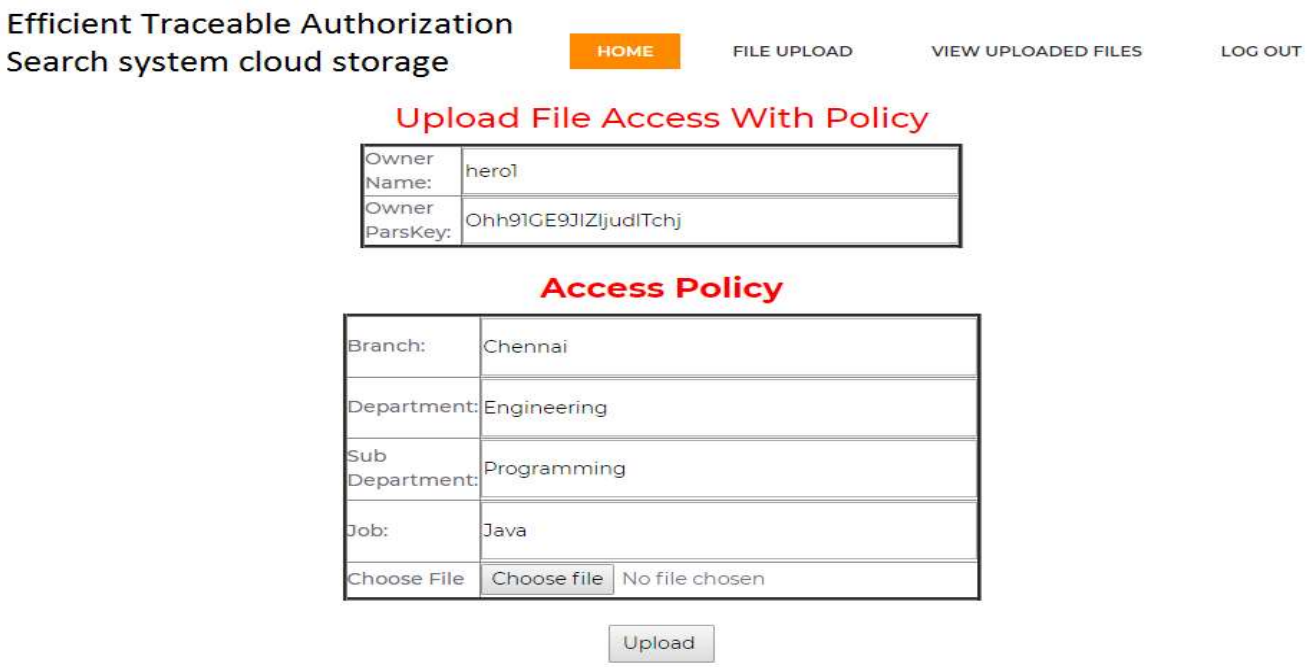


Fig-7.7: File Upload Page

VIEW FILES:

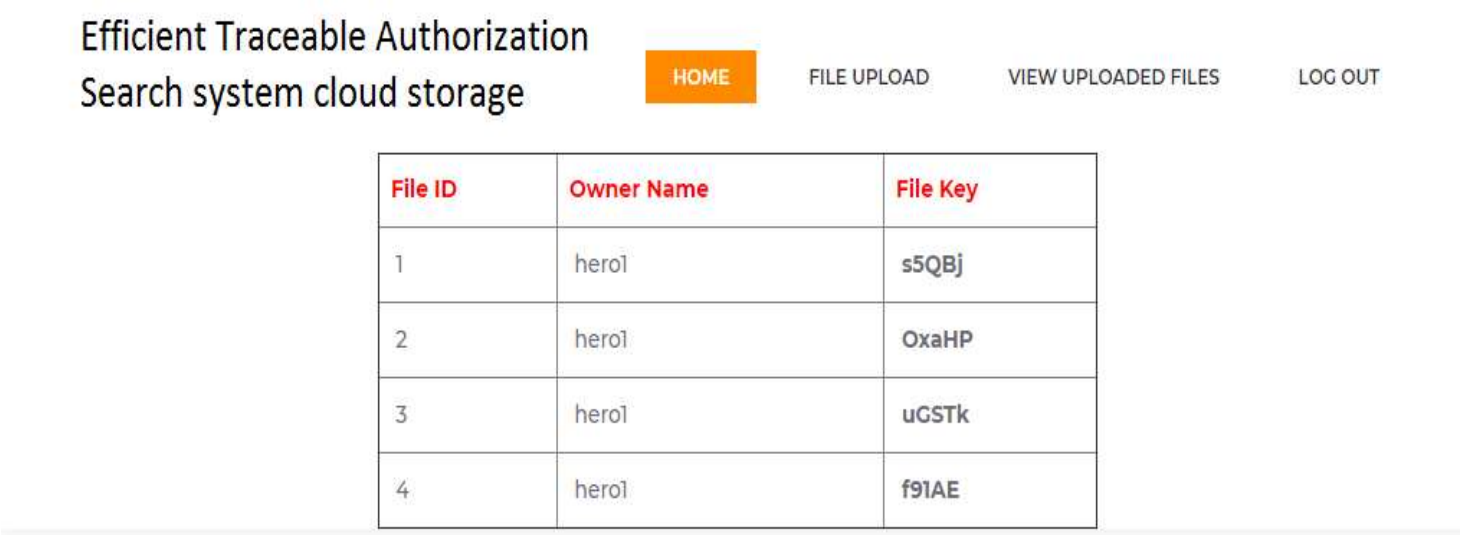


Fig-7.8: View Files Page

FILE SEARCH:

Fig-7.9: File Search Page

FILE DOWNLOAD:

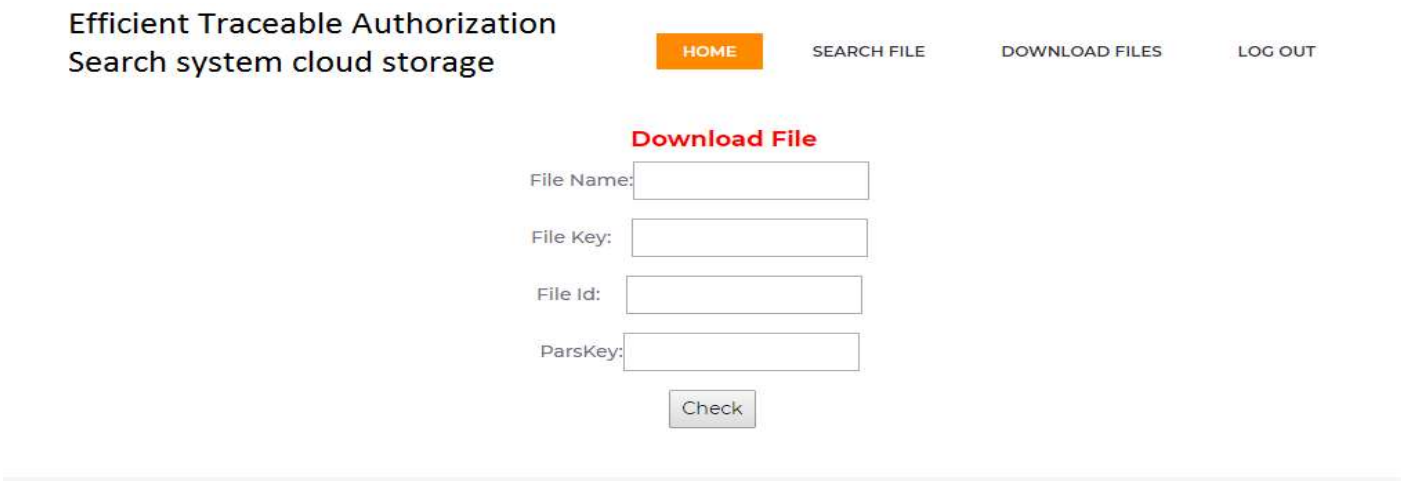
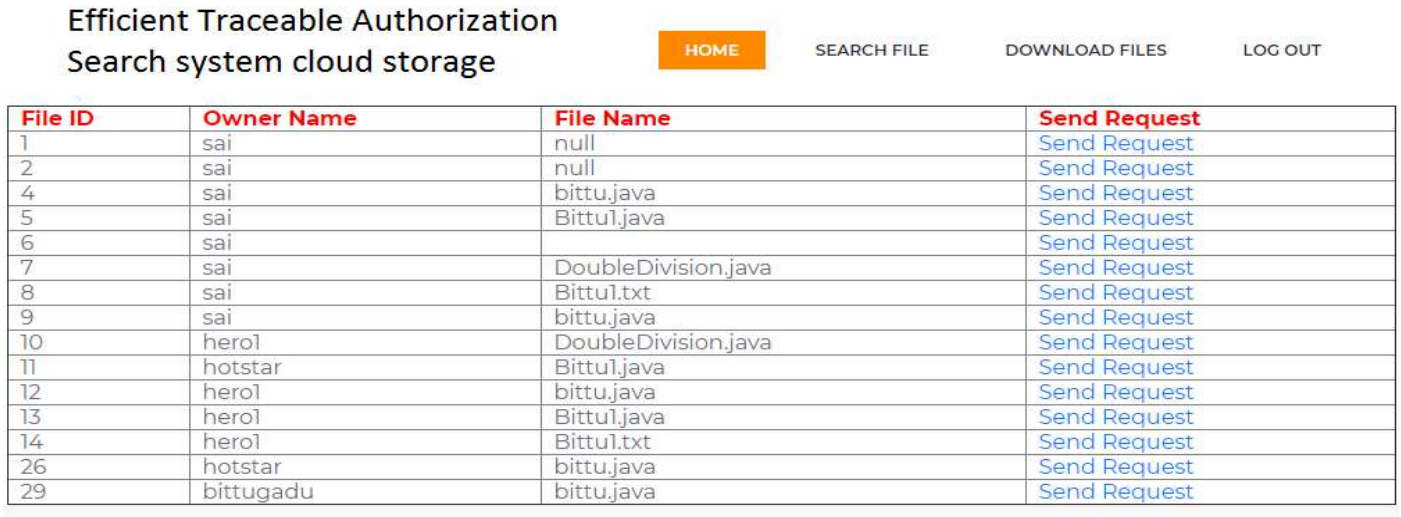


Fig-7.10: File Download Page

FILE REQUEST:



File ID	Owner Name	File Name	Send Request
1	sai	null	Send Request
2	sai	null	Send Request
4	sai	bittu.java	Send Request
5	sai	Bittu1.java	Send Request
6	sai		Send Request
7	sai	DoubleDivision.java	Send Request
8	sai	Bittu1.txt	Send Request
9	sai	bittu.java	Send Request
10	hero1	DoubleDivision.java	Send Request
11	hotstar	Bittu1.java	Send Request
12	hero1	bittu.java	Send Request
13	hero1	Bittu1.java	Send Request
14	hero1	Bittu1.txt	Send Request
26	hotstar	bittu.java	Send Request
29	bittugadu	bittu.java	Send Request

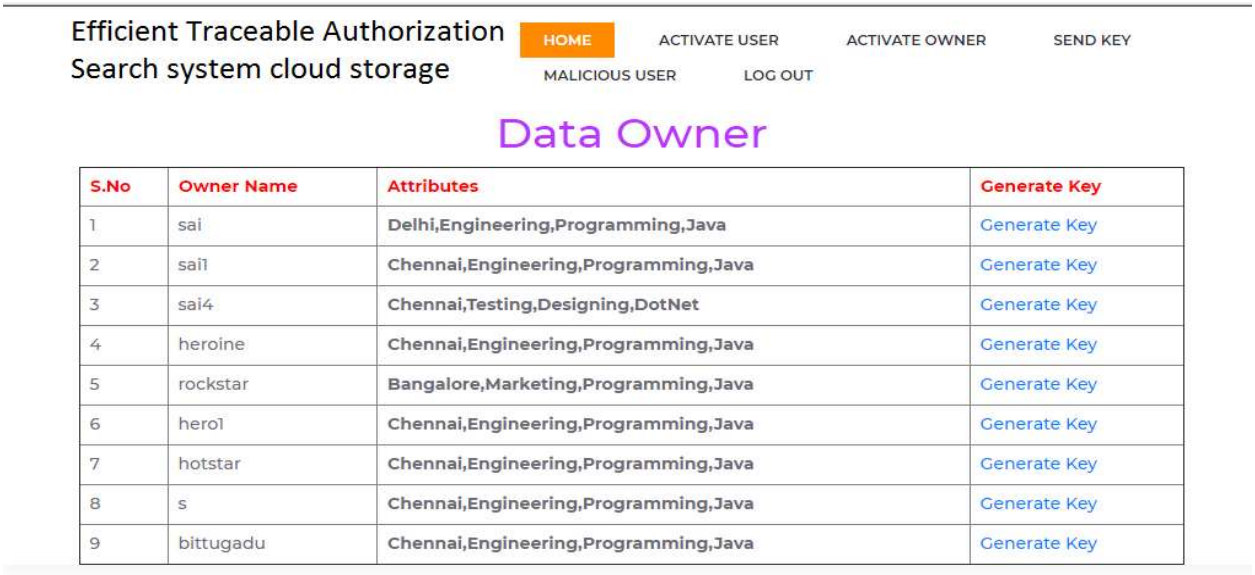
Fig-7.11 File Request Page

USER ACTIVATION:

S.No	User Name	Attributes	Generate Key
1	sai	Delhi,Engineering,Programming,Java	Generate Key
2	sai11	Mumbai,Accounting,Designing,Java	Generate Key
3	bittuboy	Bangalore,Testing,Designing,DotNet	Generate Key
4	hero	Delhi,Engineering,Programming,Java	Generate Key
5	siva	Bangalore,Testing,Designing,Java	Generate Key
6	hotstar	Chennai,Engineering,Programming,Java	Generate Key
7	sai	Chennai,Engineering,Programming,Java	Generate Key
8	bittugadu	Chennai,Engineering,Programming,Java	Generate Key

Fig-7.12: User Activation Page

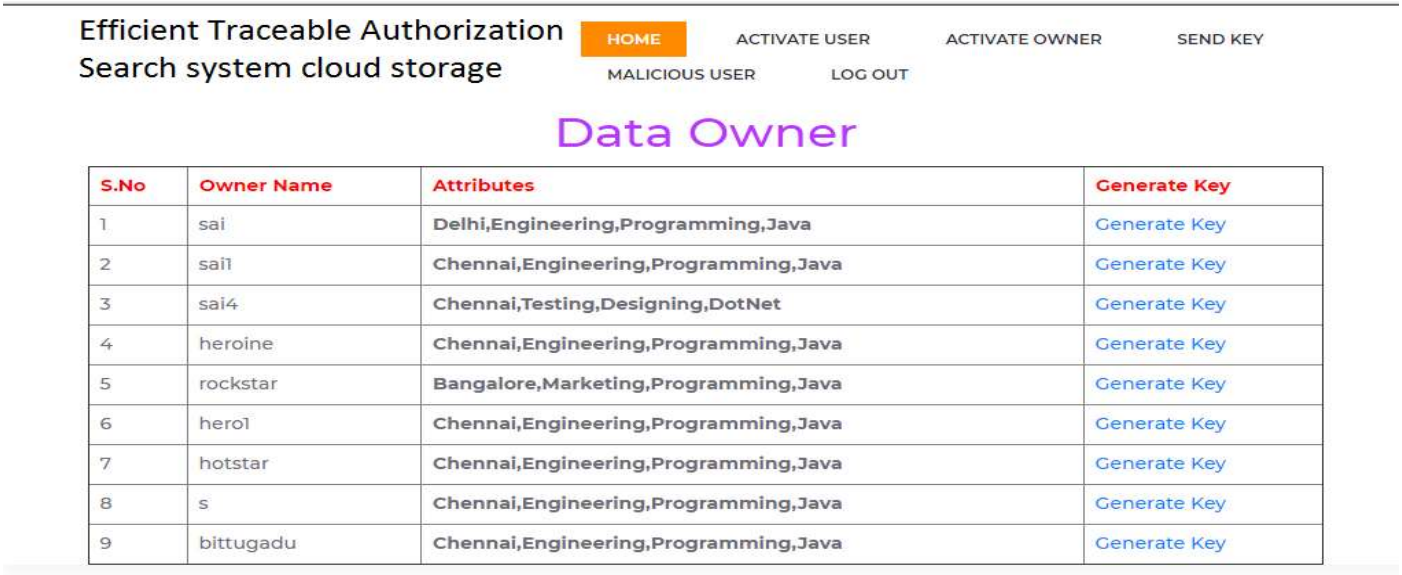
OWNER ACTIVATION:



S.No	Owner Name	Attributes	Generate Key
1	sai	Delhi,Engineering,Programming,Java	Generate Key
2	sai1	Chennai,Engineering,Programming,Java	Generate Key
3	sai4	Chennai,Testing,Designing,DotNet	Generate Key
4	heroine	Chennai,Engineering,Programming,Java	Generate Key
5	rockstar	Bangalore,Marketing,Programming,Java	Generate Key
6	hero1	Chennai,Engineering,Programming,Java	Generate Key
7	hotstar	Chennai,Engineering,Programming,Java	Generate Key
8	s	Chennai,Engineering,Programming,Java	Generate Key
9	bittugadu	Chennai,Engineering,Programming,Java	Generate Key

Fig-7.13: Owner Activation Page

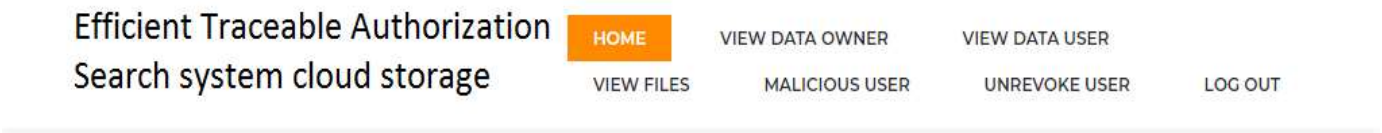
DATA OWNER:



S.No	Owner Name	Attributes	Generate Key
1	sai	Delhi,Engineering,Programming,Java	Generate Key
2	sai1	Chennai,Engineering,Programming,Java	Generate Key
3	sai4	Chennai,Testing,Designing,DotNet	Generate Key
4	heroine	Chennai,Engineering,Programming,Java	Generate Key
5	rockstar	Bangalore,Marketing,Programming,Java	Generate Key
6	hero1	Chennai,Engineering,Programming,Java	Generate Key
7	hotstar	Chennai,Engineering,Programming,Java	Generate Key
8	s	Chennai,Engineering,Programming,Java	Generate Key
9	bittugadu	Chennai,Engineering,Programming,Java	Generate Key

Fig-7.14: View Data Owner Page

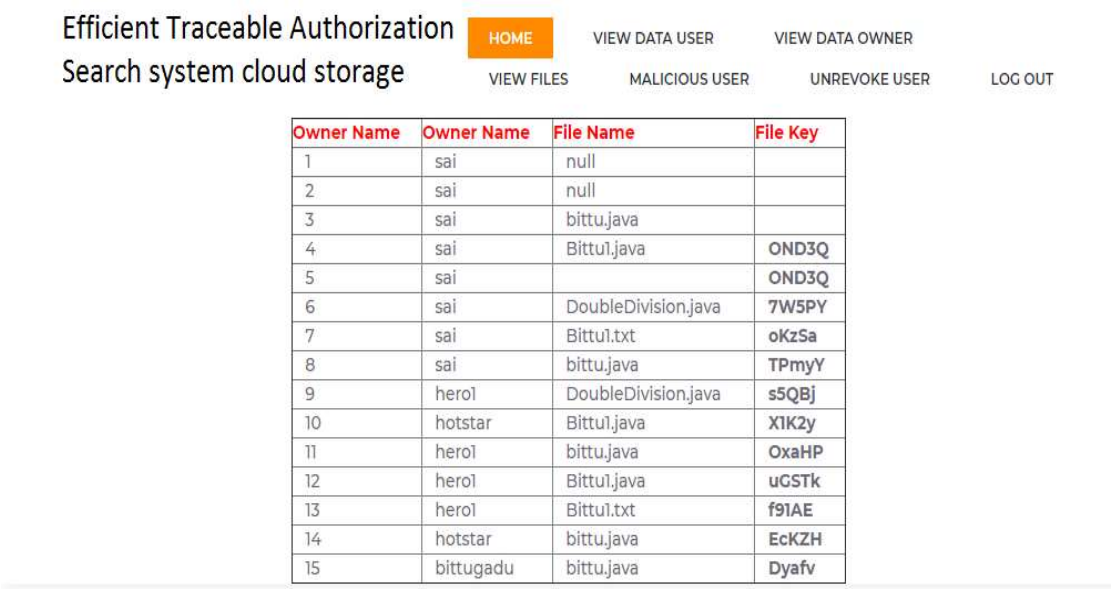
VIEW DATA USER:



OwnerID	OwnerName	MobileNumber	Mail	Attributes
1	sai	8885697874	prathap1.datapoint@gmail.com	Delhi,Engineering,Programming,Java
2	sai1	8885197874	prathap1.dat1apoint@gmail.com	Chennai,Engineering,Programming,Java
3	sai4	8885697874	prathap1.datapoi@gmail.com	Chennai,Testing,Designing,DotNet
4	heroine	8885697874	saiprathap587@gmail.com	Chennai,Engineering,Programming,Java
5	rockstar	8885697874	prathap1.datapoint@gmail.com	Bangalore,Marketing,Programming,Java
6	hero1	8885697874	prathap1.datapoint@gmail.com	Chennai,Engineering,Programming,Java
7	hotstar	8885697874	prathap1.datapoint@gmail.com	Chennai,Engineering,Programming,Java
8	s	s	s	Chennai,Engineering,Programming,Java
9	bittugadu	8885697874	prathap1.datapoint@gmail.com	Chennai,Engineering,Programming,Java

Fig-7.15: View Data User Page

FILE INFORMATION:



Owner Name	Owner Name	File Name	File Key
1	sai	null	
2	sai	null	
3	sai	bittu.java	
4	sai	Bittu1.java	OND3Q
5	sai		OND3Q
6	sai	DoubleDivision.java	7W5PY
7	sai	Bittu1.txt	oKzSa
8	sai	bittu.java	TPmyY
9	hero1	DoubleDivision.java	s5QBj
10	hotstar	Bittu1.java	X1K2y
11	hero1	bittu.java	OxaHP
12	hero1	Bittu1.java	uGSTk
13	hero1	Bittu1.txt	f91AE
14	hotstar	bittu.java	EcKZH
15	bittugadu	bittu.java	Dyafv

Fig-7.16: File Information Page

OTP:

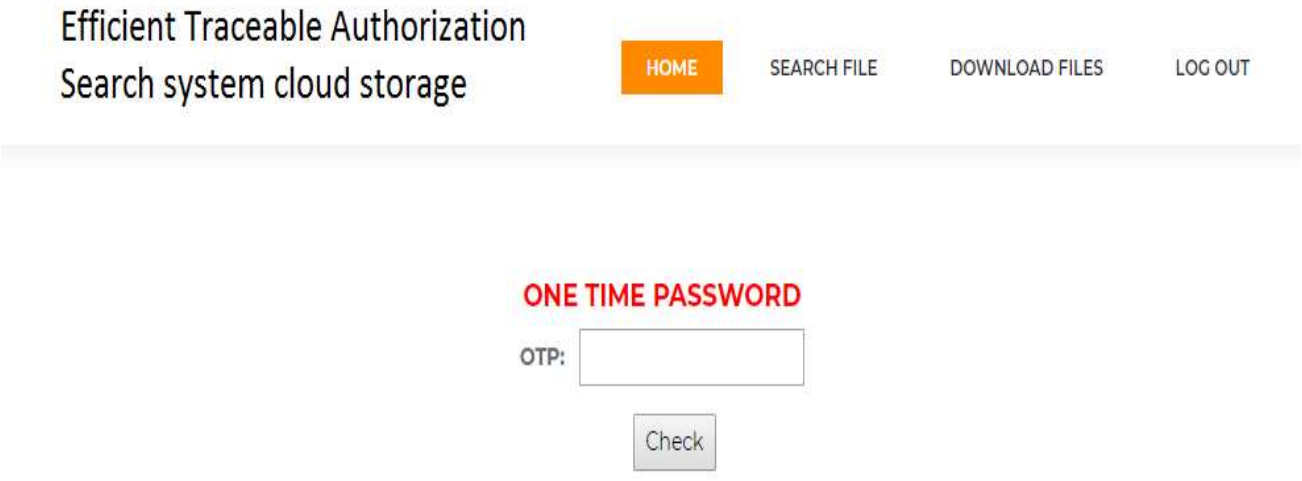


Fig-7.17: Generating OTP Page

FILE DOWNLOAD:

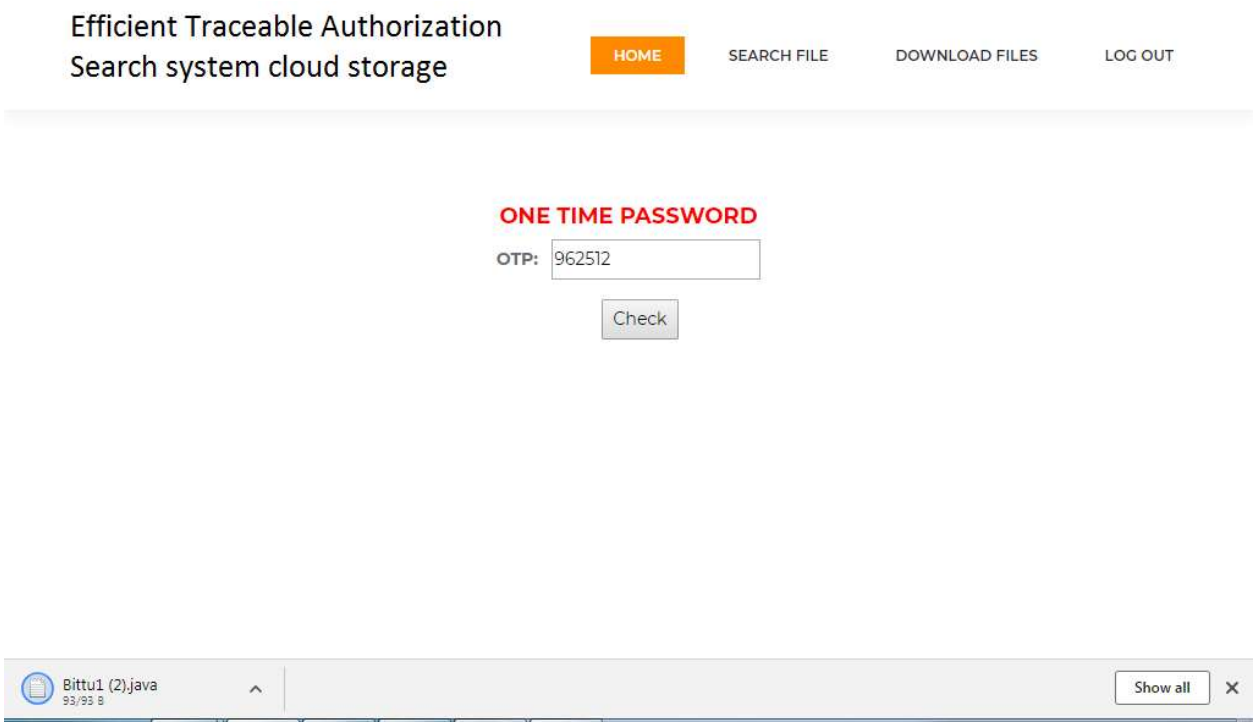


Fig-7.18: File Download Page

CONCLUSION

The enforcement of access control and the support of keyword search are important issues in secure cloud storage system. So, in our project, we defined a new paradigm of searchable encryption system, and proposed a concrete construction. The process involves the registration of both users and data owners, and the matching of their attributes or keywords to ensure that they are linked correctly. Once registered, both users and data owners need to be accepted by the KGC (Key Generation Center) and provided with public and private keys to access the cloud server.

Users can then search for files uploaded by the data owner and request details from KGC, who will check their authenticity and provide file details if verified. To access the file, KGC generates an OTP (One Time Password) that the user must enter correctly before being allowed to download the requested file from the cloud server. This process helps ensure the security and integrity of the data stored on the cloud server and restricts access to authorized users only.

FUTURE ENHANCEMENT

In Local Area Network, the proposed hybrid encryption mechanism may be customized for transferring the sensitive data from work station to host based applications. In web-based applications, the proposed mechanism enables the transfer of sensitive data from user to user, from user to server and from server to server which are located outside of the organization.

In a cloud environment, a greater number of people are accessing the web server locally or globally to share the sensitive data. The proposed hybrid encryption technique is very helpful to enhance the security for web-based transactions in future. And also, we can add additional feature to block and revoke Malicious user who sells secret key for benefit.

CHAPTER 10

APPENDIX

Java Technology:

Java technology is both a programming language and a platform.

The Java Programming Language:

Secure The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

- Simple
- Architecture neutral
- Object oriented
- Portable
- Distributed
- High performance
- Interpreted
- Multithreaded
- Robust
- Dynamic

With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted.

With the compiler, first you translate a program into an intermediate language called *Java byte codes* —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code

instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.

You can think of Java byte codes as the machine code instructions for the *Java Virtual Machine* (Java VM). Every Java interpreter, whether it's a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make "write once, run anywhere" possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.

The Java Platform:

A *platform* is the hardware or software environment in which a program runs. We've already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

The Java platform has two components:

- The *Java Virtual Machine* (Java VM)
- The *Java Application Programming Interface* (Java API)

You've already been introduced to the Java VM. It's the base for the Java platform and is ported onto various hardware-based platforms.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as *packages*. The next section, What Can Java Technology Do? Highlights what functionality some of the packages in the Java API provide.

Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability.

What Can Java Technology Do?

The most common types of programs written in the Java programming language are *applets* and *applications*. If you've surfed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java-enabled browser.

However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

- **Security**: Both low level and high level appropriate language.
- **Applets**: The set of conventions used by applets.
- **Networking**: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) sockets, and IP (Internet Protocol) addresses.

Internationalization: Help for How does the API support all these kinds of programs? It does so with packages of software components that provides a wide range of functionality. Every full implementation of the Java platform gives you the following features:

- **The essentials**: Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.
- writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the, including electronic signatures, public and private key management, access control, and certificates.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK.

JDBC:

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs.

This consistent interface is achieved through the use of "plug-in" database connectivity modules, or *drivers*. If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

Java has two things: a programming language and a platform. Java is a high-level programming language that is all of the following

Simple	Architecture-neutral
Object-oriented	Portable
Distributed	High-performance
Interpreted	multithreaded
Robust	Dynamic
Secure	

Java is also unusual in that each Java program is both compiled and interpreted. With a compile you translate a Java program into an intermediate language called Java byte codes the platform-independent code instruction is passed and run on the computer.

You can think of Java byte codes as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it's a Java development tool or a Web browser that can run Java applets, is an implementation of the Java VM. The Java VM can also be implemented in hardware.

Java byte codes help make "write once, run anywhere" possible. You can compile your Java program into byte codes on my platform that has a Java compiler. The byte codes can then be run any implementation of the Java VM. For example, the same Java program can run Windows NT, Solaris, and Macintosh.

What is a Java Web Application?

A Java web application generates interactive web pages containing various types of markup language (HTML, XML, and so on) and dynamic content. It is typically comprised of web components such as Java Server Pages (JSP), servlets and JavaBeans to modify and temporarily store data, interact with databases and web services, and render content in response to client requests.

Because many of the tasks involved in web application development can be repetitive or require a surplus of boilerplate code, web frameworks can be applied to alleviate the overhead associated with common activities. For example, many frameworks, such as Java Server Faces, provide libraries for templating pages and session management, and often promote code reuse.

What is Java EE?

Java EE (Enterprise Edition) is a widely used platform containing a set of coordinated technologies that significantly reduce the cost and complexity of developing, deploying, and managing multi-tier, server-centric applications. Java EE builds upon the Java SE platform and provides a set of APIs (application programming interfaces) for developing and running portable, robust, scalable, reliable and secure server-side applications.

Some of the fundamental components of Java EE include:

- Enterprise JavaBeans (EJB): a managed, server-side component architecture used to encapsulate the business logic of an application.
- EJB technology enables rapid and simplified development of distributed, transactional, secure and portable applications based on Java technology.
- Java Persistence API (JPA): a framework that allows developers to manage data using object-relational mapping (ORM) in applications built on the Java Platform.

JavaScript and Ajax Development:

JavaScript is an object-oriented scripting language primarily used in client-side interfaces for web applications. Ajax (Asynchronous JavaScript and XML) is a Web 2.0 technique that allows changes to occur in a web page without the need to perform a page refresh. JavaScript toolkits can be leveraged to implement Ajax-enabled components and functionality in web pages.

Web Server and Client:

Web Server is a software that can process the client request and send the response back to the client. For example, Apache is one of the most widely used web server. Web Server runs on some physical machine and listens to client request on specific port.

A web client is a software that helps in communicating with the server. Some of the most widely used web clients are Firefox, Google Chrome, Safari etc.

HTML and HTTP:

Web Server and Web Client are two separate software's, so there should be some common language for communication. HTML is the common language between server and client and stands for **H**yper **T**ext **M**arkup **L**anguage.

Web server and client needs a common communication protocol, HTTP (**H**yper **T**ext **T**ransfer **P**rotocol) is the communication protocol between server and client. HTTP runs on top of TCP/IP communication protocol.

MIME Type or Content Type:

If you see above sample HTTP response header, it contains tag "Content-Type". It's also called MIME type and server sends it to client to let them know the kind of data it's sending. It helps client in rendering the data for user. Some of the mostly used mime types are text/html, text/xml, application/xml etc.

Understanding URL:

URL is acronym of Universal Resource Locator and it's used to locate the server and resource. Every resource on the web has its own unique address. Let's see parts of URL with an example.

http://localhost:8080/FirstServletProject/jsps/hello.jsp

http:// – This is the first part of URL and provides the communication protocol to be used in server-client communication.

localhost – The unique address of the server, most of the times it's the hostname of the server that maps to unique IP address. Sometimes multiple hostnames point to same IP addresses and web server virtual host takes care of sending request to the particular server instance.

8080 – This is the port on which server is listening, it's optional and if we don't provide it in URL then request goes to the default port of the protocol. Port numbers 0 to 1023 are reserved

Web Application Directory Structure:

Java Web Applications are packaged as Web Archive (WAR) and it has a defined structure. You can export above dynamic web project as WAR file and unzip it to check the hierarchy. It will be something like below image.

Deployment Descriptor:

web.xml file is the deployment descriptor of the web application and contains mapping for servlets (prior to 3.0), welcome pages, security configurations, session timeout settings etc. Thats all for the java web application startup tutorial, we will explore Servlets and JSPs more in future posts.

MySQL:

MySQL, the most popular Open-Source SQL database management system, is developed, distributed, and supported by Oracle Corporation.

The MySQL Web site (http://www.mysql.com/) provides the latest information about MySQL software.

- **MySQL is a database management system:**

  A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network.

- **MySQL databases are relational.**

  A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and “pointers” between different tables.

- **MySQL software is Open Source.**

  Open-Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything.

CHAPTER 9

BIBLIOGRAPHY

RReferences:

[1] C. Wang, N. Cao, J. Li, K. Ren, W. Lou. "Secure ranked keyword search over encrypted cloud data"[C]//IEEE 30th International Conference on Distributed Computing Systems (ICDCS), IEEE, 2010: 253-262.

[2] Q. Zhang, L. T. Yang, Z. Chen, P. Li, M. J. Deen. "Privacy-preserving Double-Projection Deep Computation Model with Crowd sourcing on Cloud for Big Data Feature Learning," IEEE Internet of Things Journal, 2017, DOI: 10.1109/JIOT.2017.2732735.

[3] R. Chen, Y. Mu, G. Yang, F. Guo and X. Wang, "Dual-Server Public- Key Encryption with Keyword Search for Secure Cloud Storage," IEEE Transactions on Information Forensics and Security, 2016, vol.11, no. 4, 789-798.

[4] X. Liu, R.H. Deng, K.K.R. Choo, J. Weng. "An efficient privacy preserving outsourced calculation toolkit with multiple keys." IEEE Transactions on Information Forensics and Security 11.11 (2016): 2401-2414.

[5] B. R. Waters, D. Balfanz, G. Durfee, and D. K. Smetters, "Building an encrypted and searchable audit log," in NDSS, 2004.

[6] Y. Yang, X. Liu, R.H. Deng, "Multi-user Multi-Keyword Rank Search over Encrypted Data in Arbitrary Language". IEEE Transactions on Dependable and Secure Computing, 2018, publish online, DOI: 10.1109/TDSC.2017.2787588.

[7] W. Sun, S. Yu, W. Lou, Y. Hou and H. Li, "Protecting Your Right: Verifiable Attribute-based Keyword Search with Fine-grained Owner-enforced Search Authorization in the Cloud," IEEE Transactions on Parallel and Distributed Systems, 2016, vol. 27, no.4, pp. 1187-1198.

[8] K. Liang, W. Susilo, "Searchable Attribute-Based Mechanism with Efficient Data Sharing for Secure Cloud Storage," IEEE Transactions on Information Forensics and Security, 2015, vol. 10, no. 9, pp. 1981-1992.

[9] M. Green, S. Hohenberger, and B.Waters, "Outsourcing the decryption of ABE ciphertexts," in USENIX Security Symposium, ACM, 2011, pp. 34-34.

[10] J. Lai, R. H. Deng, C. Guan, and J. Weng, "Attribute-based encryption with verifiable outsourced decryption," IEEE Transactions on Information Forensics and Security, 2013, vol. 8, no. 8, pp. 1343-1354.

[11] B. Qin, R. H. Deng, S. Liu, and S. Ma, "Attribute-Based Encryption with Efficient Verifiable Outsourced Decryption," IEEE Transactions on Information Forensics and Security, 2015, vol. 10, no. 7, pp. 1384-1394.

[12] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in: EUROCRYPT, 2004, pp.506-522.

[13] Z. Liu, Z. Cao, D.S. Wong, "White-box traceable ciphertext-policy attribute-based encryption supporting any monotone access structures," IEEE Transactions on Information Forensics and Security,2013, vol. 8, no. 1, pp. 76-88.

[14] J. Ning, X. Dong, Z. Cao, L. Wei, X. Lin, "White-Box Traceable Ciphertext-Policy Attribute-Based Encryption Supporting Flexible Attributes," IEEE Transactions on Information Forensics and Security,2015, vol. 10, no. 6, pp. 1274-1288.

[15] Z. Liu, Z. Cao, D.S. Wong, "Traceable CP-ABE: how to trace decryption devices found in the wild," IEEE Transactions on Information Forensics and Security, 2015, vol. 10, no. 1, pp. 55-68.

[16] D. Boneh and B. Waters, "Conjunctive, subset, and range queries on encrypted data," in: 4th Theory Cryptography Conference, 2007, vol. 4392, pp. 535-554.

[17] P. Xu, H. Jin, Q. Wu and W. Wang, "Public-Key Encryption with Fuzzy Keyword Search: A Provably Secure Scheme under Keyword Gusssing Attack," IEEE Transactions on Computers, 2013, vol. 62, no. 11, 2266-2277.

[18] Q. Tang, "Nothing is for Free: Security in Searching Shared and Encrypted Data," IEEE Transactions on Information Forensics and Security, 2014, vol. 9, no. 11, 1943-1952.

Web Resources:

1. https://jpinfotech.org/efficient-traceable-authorization-search-system-for-secure-cloud-storage/
2. http://ieeexplore.ieee.org/document/8327889/
3. https://www.irjet.net/archives/V6/i12/IRJET-V6I1279.pdf
4. http://inpressco.com/secure-and-efficient-traceable-authorization-multi-keyword-search-system-for-cloud-storage-using-blockchain-technology/
5. https://journals.pen2print.org/index.php/ijr/article/view/19204/18887

CHAPTER – 11

BIO DATA

Name	Father Name, Address, Contact no & Email	Photo
P. Naga Lakshmi	P. Ananda Babu Proddatur 9014180870 posanagalakshmi93@gmail.com	
M. Mahaboob Basha	M. Mabu Peera Peddavangali 9392414396 mullamahaboob2002@gmail.com	
P. Bindu Shruthika	P. Sudhakar Proddatur 8688203890 bindhusruthika@gmail.com	
M. Karthik	M. Karthik Khadarabad 7901003343 karthik270402@gmail.com	
M. Mahendra Kumar Reddy	M. Venkata Rami Reddy Gudipadu 7601087278 mahendra070702@gmail.com