DATA LEAKAGE DETECTION IN MAIL SERVER

A project report submitted to

DEPARTMENT OF INFORMATION TECHNOLOGY

In partial fulfillment of the requirements for the award of the Degree of

BACHELOR OF COMPUTER APPLICATION

Submitted by

LEO DANIEL A

D. No: 19UBC616

Under the guidance of

Dr. S. HENDRY LEO KANICKAM, MSc., B.Ed., MPhil., Ph.D.,

ST. JOSEPH'S COLLEGE (AUTONOMOUS)

Special Heritage Status Awarded by UGC
Accredited at "A^{++}" Grade (4th cycle) by NAAC
College with Potential for Excellence by UGC
DBT-STAR & DST – FIST Sponsored College

TIRUCHIRAPPALLI-620 002

JUNE – 2022

DEPARTMENT OF INFORMATION TECHNOLOGY

ST. JOSEPH'S COLLEGE (AUTONOMOUS)

TIRUCHIRAPPALLI-620 002

CERTIFICATE

This is to certify that the Project work titled as "**DATA LEAKAGE DETECTION IN MAIL SERVER**" is a bona-fide report of **LEO DANIEL A, D.NO-19UBC616** submitted to the Department of Information Technology in partial fulfillment of the requirements for the award of the Degree of **Bachelor of Computer Application** during the period, **FEBRUARY 2022 - JUNE 2022**

HEAD OF THE DEPARTMENT **GUIDE**

Submitted for the viva voce on_____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

Date: **Date:**

ACKNOWLEDGEMENT

First and foremost, I am graceful to the **GOD** Almighty, who has showered his blessing on me throughout this project program.

I profoundly thank Rev. **Dr .M. Arockiasamy Xavier SJ**, Principal, St.Joseph's college (autonomous), Tiruchirappalli-2, for providing me the opportunity to be a part of the institution.

My sincere gratitude to **Dr. P. Joseph Charles, M.Sc., M.Phil., Ph.D. Head of the Department of Information Technology St. Joseph's College(Autonomous)**, Tiruchirappalli-02, for motivating me to be a Positive personality.

I gratefully acknowledge my personal thank to **Dr. S. Hendry Leo Kanickam MSc., B.Ed., MPhil., Ph.D.,** who guided me with his scholarly valuable interventions.

I express my special thanks to my class-in-charge **Dr. R. Mangai Begum, MSc(IT). ,MTech., MPhil. ,PhD.** and other **staff members of the Department Information Technology, St. Joseph's College (Autonomous),** Tiruchirappalli-2.

I own a deep debt of gratitude to **my parents and my friends** for their love and kindness which sustained my effort throughout my ideas.

LEO DANIEL A

INTRODUCTION

1.INTRODUCTION

1.1 About the project

This project, Data leak detection content inspection, sampling, alignment, dynamic Programming, parallelism. Reports show that the number of leaked sensitive data records has grown 10 times in the last 4 years, and it reached a record high of 1.1 billion in 2014. A significant portion of the data leak incidents are due to human errors, for example, a lost or stolen laptop containing unencrypted sensitive files, or transmitting sensitive data without using end to-end encryption such as PGP. A recent Kaspersky Lab survey shows that accidental leak by staff is the leading cause for internal data leaks in corporate. The data-leak risks posed by accidents exceed the risks posed by vulnerable software. In order to minimize the exposure of sensitive data and documents, an organization needs to prevent clear text sensitive data from appearing in the storage or communication. A screening tool can be deployed to scan computer file systems, server storage, and inspect outbound network traffic. The tool searches for the occurrences of plaintext sensitive data in the content of files or network traffic. It alerts users and administrators of the identified data exposure vulnerabilities. For example, an organization’s mail server can inspect the content of outbound email messages searching for sensitive data appearing in unencrypted messages.

SYSTEM STUDY

2.SYSTEM STUDY

2.1 Existing System

In an existing system straight forward realizations of data-leak detection require the Plaintext Sensitive data. However, this requirement is undesirable, as it may threaten the confidentiality of the Sensitive information.

Existing commercial data leak detection/prevention solutions include Symantec DLP, Identity Finder, Global Velocity, and Go Cloud DLP. Global Velocity uses FPGA to accelerate the system. All solutions are likely based on n-gram set intersection. A detection system is compromised, then it may exposed the plaintext Sensitive data. In addition, the data owner may need to outsource the data leak detection to providers, but may be unwilling to reveal the plaintext.

There was no privacy preserving in existing system, so providers can access the data without data-owners permission.

2.2 Disadvantages of Existing System

- Data loss
- Is not effective

2.3 Proposed System

In my proposed system I propose a data-leak detection solution which can be outsourced from organization, I design and implement Lucerne search engine framework Levenshtein-distance technique to avoid data leak and also provide privacy preserving to Sensitive data. Data Owner owns the Sensitive data and authorizes the DLD provider to inspect the network traffic from the organizational networks for anomalies, namely inadvertent data leak. Mail Server - DLD provider inspects the network traffic for potential data leaks. Focus on detecting inadvertent data leaks.

2.4 Advantages of Proposed System

- Is more efficient
- Data leakage is reduced and more securable.

2.5 Problem Definition and Description

Information privacy, or data privacy is the relationship between collection and dissemination of data, technology, the public expectation of privacy, and the legal and political issues surrounding them. Privacy concerns exist wherever personally identifiable information is collected and stored - in digital form or otherwise. Improper or non-existent disclosure control can be the root cause for privacy issues.

This project contains the following modules are:

Admin Login:

Admin is the super user where they will maintain the complete details of the files

User Login:

The user can send files to the other users with private key.

Request Key:

The user can send request to the owner of the files to send a private key.

User Request:

The admin can accept or decline the request from the user which they send for their user registration.

SYSTEM ANALYSIS

3.SYSTEM ANALYSIS

3.1 Packages selected

- Front End : JAVA
- Back end : MYSQL

3.2 Resources Required

3.2.1 Software Resources

- Operating System : Window 10
- IDE : Eclipse 4.5
- Web Server : Tomcat Apache 9
- Browser : Internet Explorer 10
- Documentation : MS word

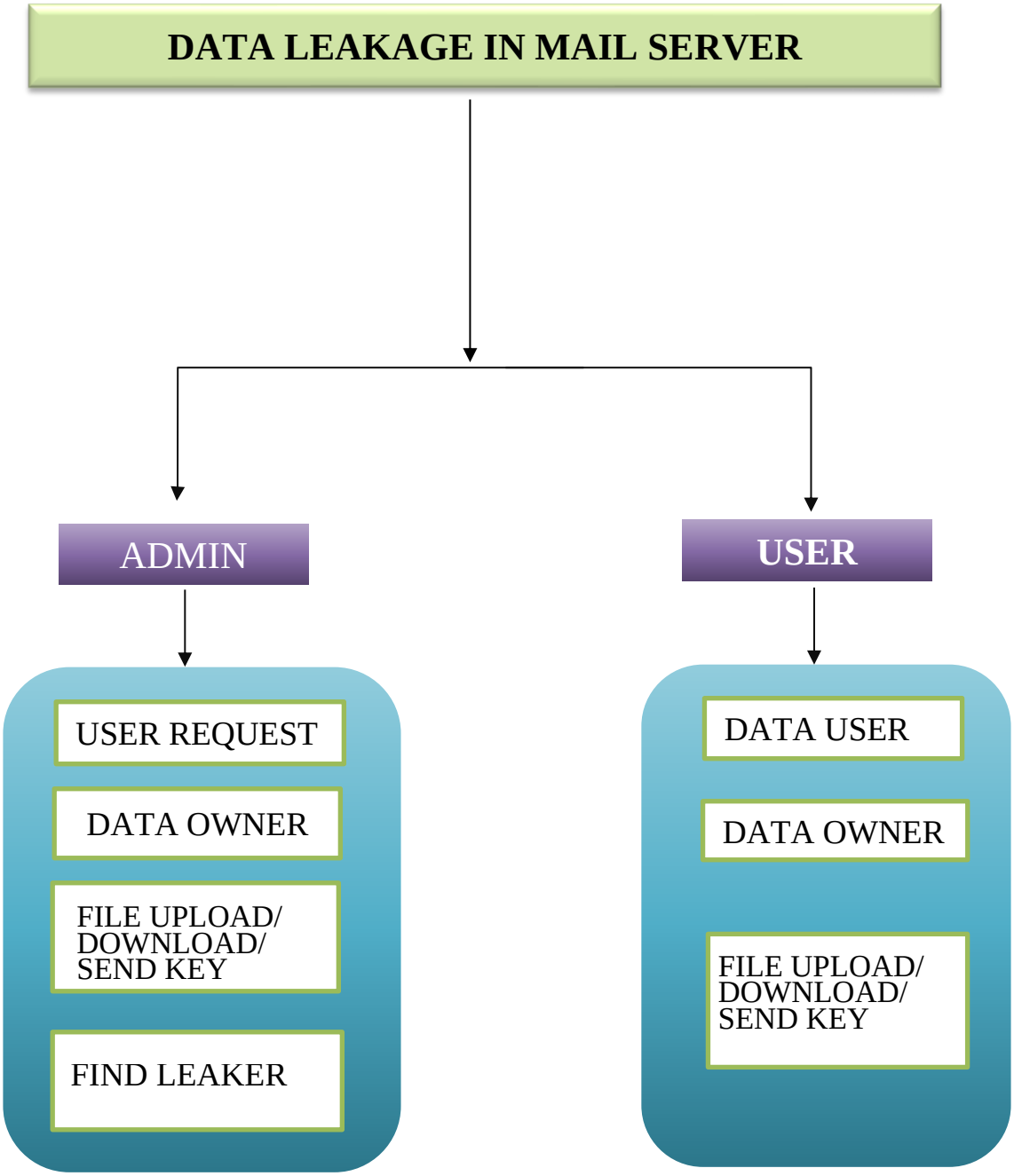
3.2.2 Hardware Resources

- Processor : Core i5
- RAM : 4GB
- Hard disk : 500 GB
- Compact Disk : 650 Mb
- Keyboard : Standard keyboard
- Monitor : 15 inch color monitor

SYSTEM DESIGN

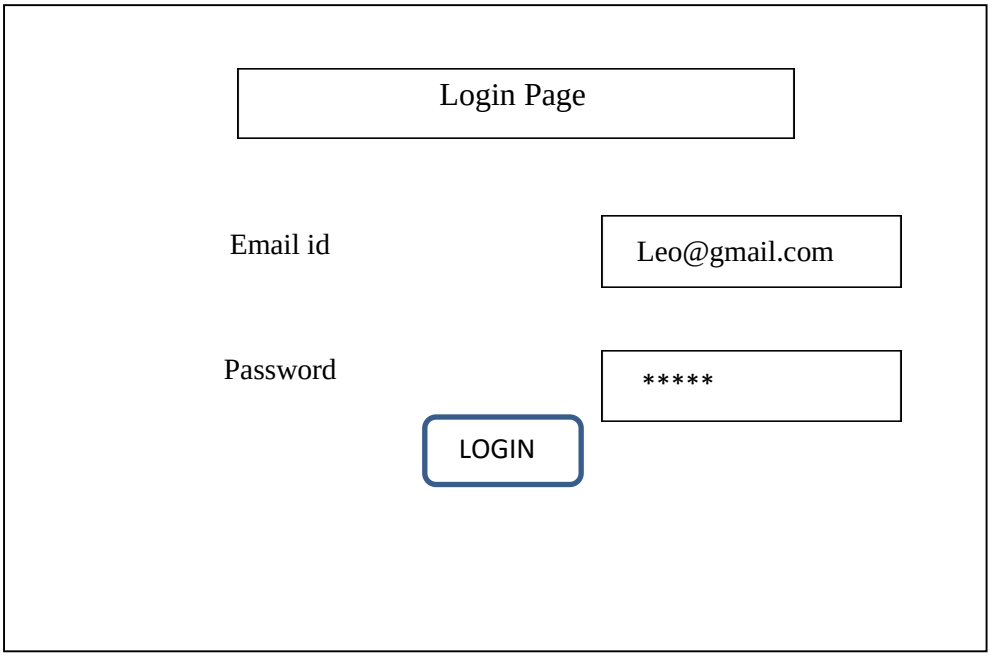
4.SYSTEM DESIGN

4.1 Architectural Design

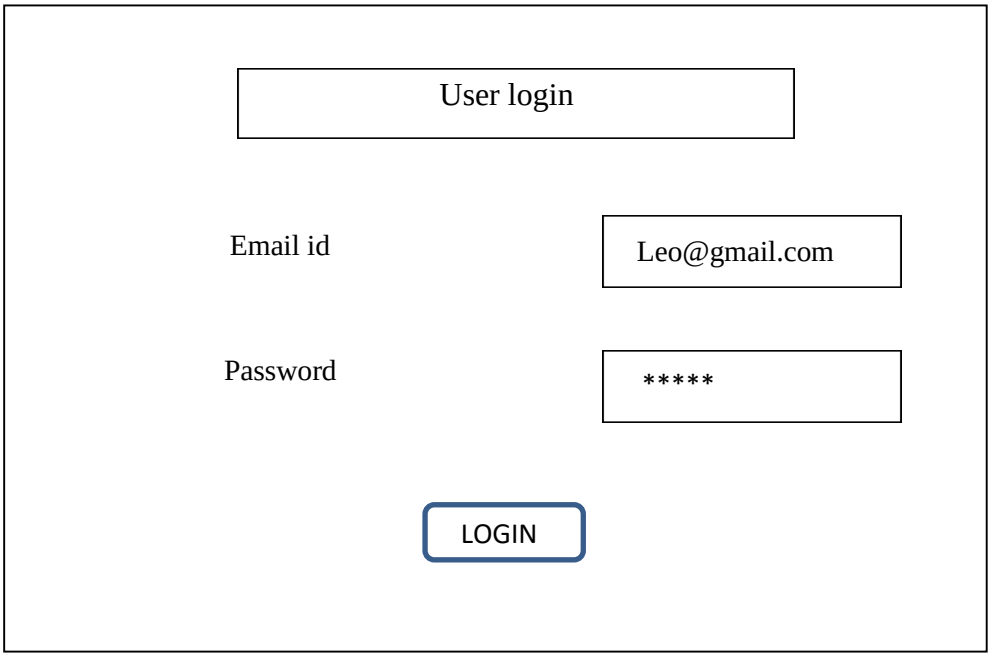


4.2 I/O Form design

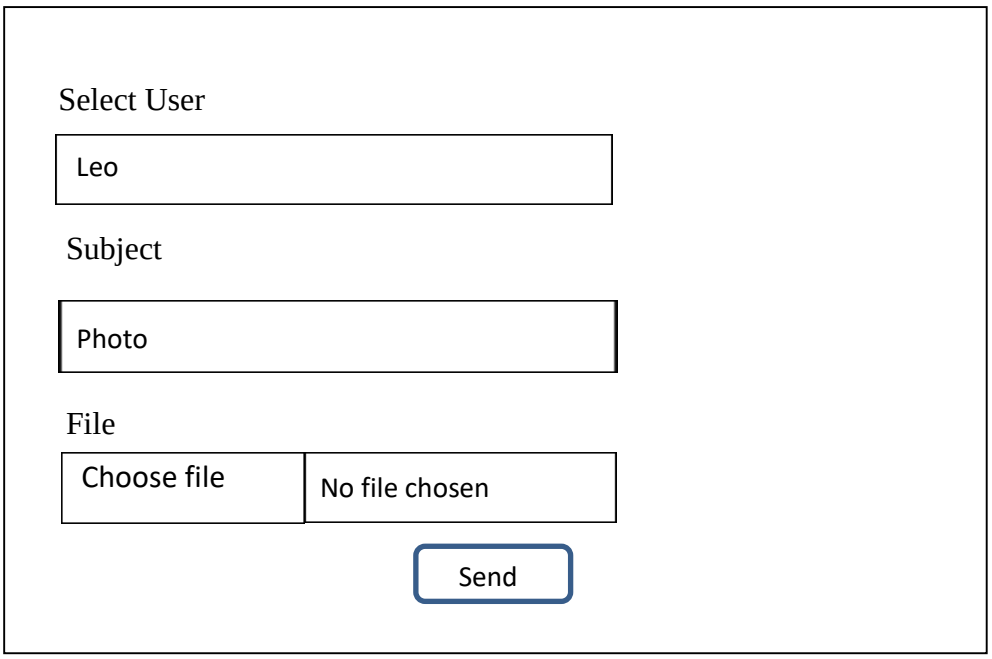
Admin login



User login



Sent files



User registration request

Sr No	User Name	Email Id	Activate account
1	Daniel **Gender**: male **Moblie:**	Daniel@gmail.com	Deactivate Block Remove
2	Leo **Gender:** male **Moblie:**	leo@gmail.com	Deactivate Block Remove

Files leakers

Sr no	File Details	Leaker
1	**Subject:** Files **File Name:** Pdf **File Size:**2.41Mb	**Username:** Daniel **Email:**daniel@gmail.com Block Remove
2	**Subject:** Photo **File Name:** Photo **File Size:**5Mb	**Username:** leo **Email:**leo@gmail.com Block Remove

4.3 Tables

4.3.1 Table name: Admin

Field	Type	Default
Email	varchar(255)	NULL
Password	varchar(30)	NULL

4.3.2 Table name: Request key

Field	Type	Default
Request	Varchar(200)	NULL
File	Varchar(300)	NULL
Status	Varchar(20)	NULL

4.3.3 Table name: Users

Field	Type	Default
Username	Varchar(10)	NULL
Email	Varchar(20)	NULL
Password	Varchar(10)	NULL
Gender	Varchar(10)	NULL
Moblie	Varchar(10)	NULL

4.4 Normalization

Database Normalization is a technique of organizing the data in the database. Normalization is a systematic approach of decomposing tables to eliminate data redundancy (repetition) and undesirable characteristics like Insertion, Update and Deletion Anamolies. It is a multi-step process that puts data into tabular form, removing duplicated data from the relation tables.

Normalization is used for mainly two purposes,

- Eliminating redundant (useless) data.
- Ensuring data dependencies make sense i.e. data is logically stored.

4.4.1 Unnormalized Table

Id	First name	Last name	Username	Mail id	Password	Phone number
1	Leo	A	Leo	leo@gmail.com	*****	8610663372
1	Leo	A	Leo	leo@gmail.com	*****	8610663372
2	Daniel	A	Daniel	daniel@gmail.com	*******	9486651819
2	Daniel	A	Daniel	daniel@gmail.com	*******	9486651819

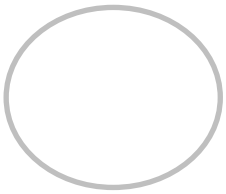
4.4.2 Normalized Table

Id	First name	Last name	Username	Mail id	Password	Phone number
1	Leo	A	Leo	leo@gmail.com	******	8610663372
2	Daniel	A	Daniel	daniel@gmail.com	******	9486651819
3	Domy	A	Domy	domy@gmail.com	******	8457656978

4.5 Data flow diagram:

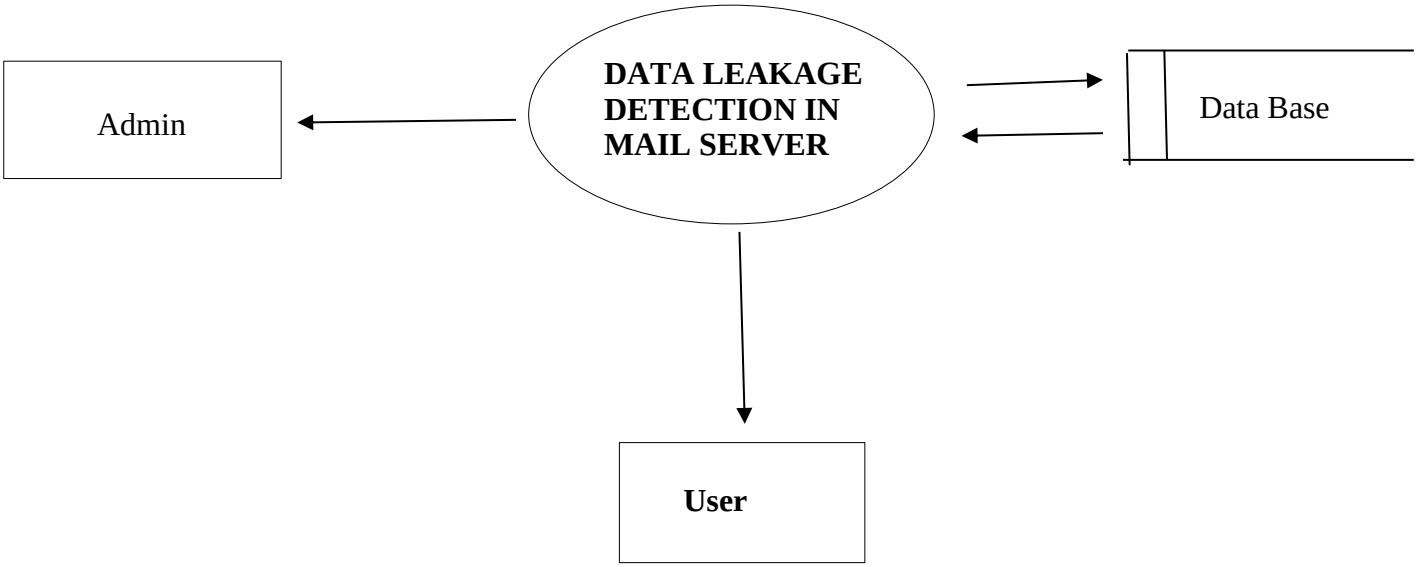
A two-dimensional diagram explains how data is processed and transferred in a system. The graphical depiction identifies each source of data and how it interacts with other data sources to reach a common output. Individuals seeking to draft a data flow diagram must identify external inputs and outputs, determine how the inputs and outputs relate to each other, and explain with graphics how these connections relate and what they result in. This type of diagram helps business development and design teams visualize how data is processed and identify or improve certain aspects.

4.5.1 Data flow symbols:

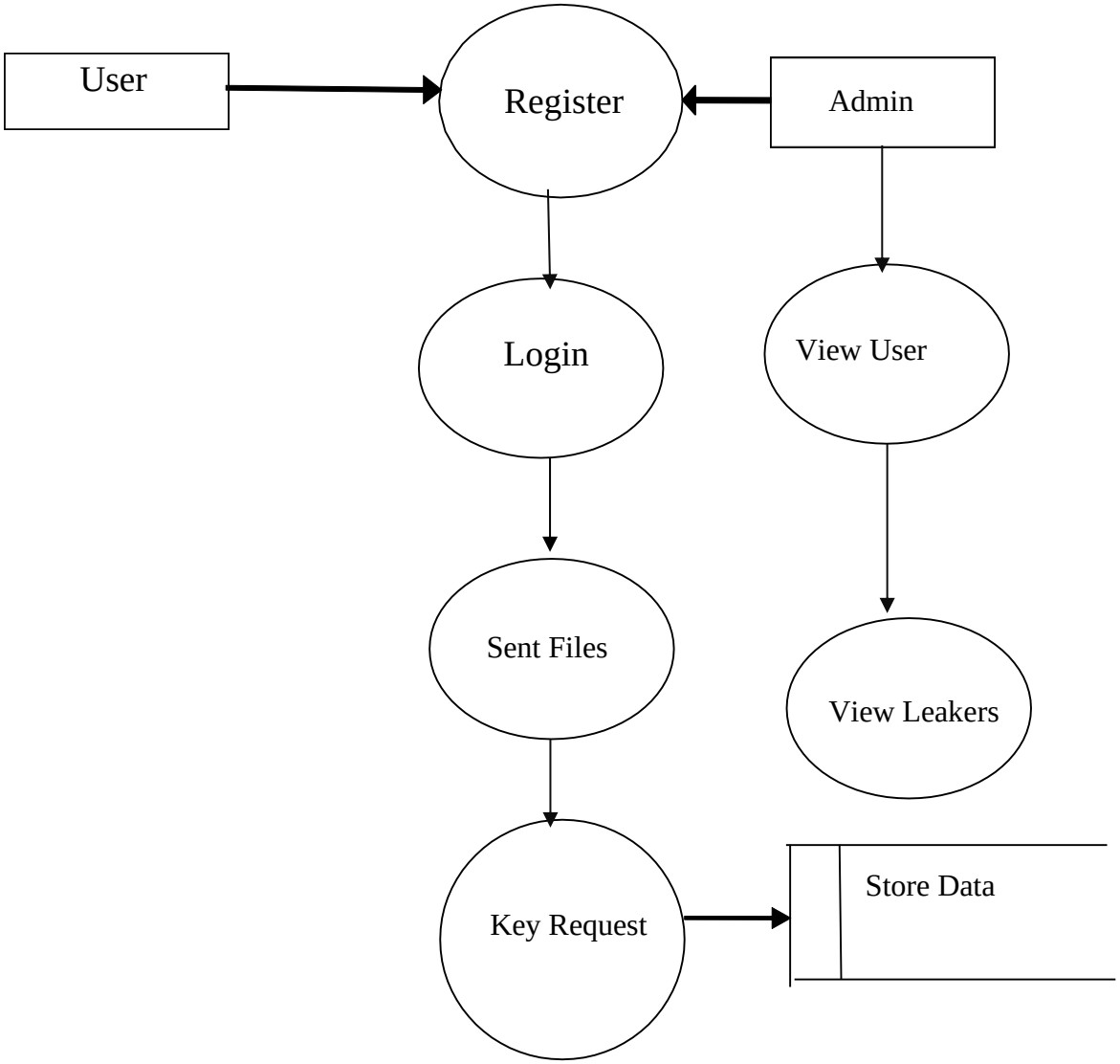
Symbol	Description
	An **entity**. A source of data or a destination for data.
	A **process** or task that is performed by the system.
	A **data store**, a place where data is held between processes.
	A **data flow**.

4.5.2 Level 0

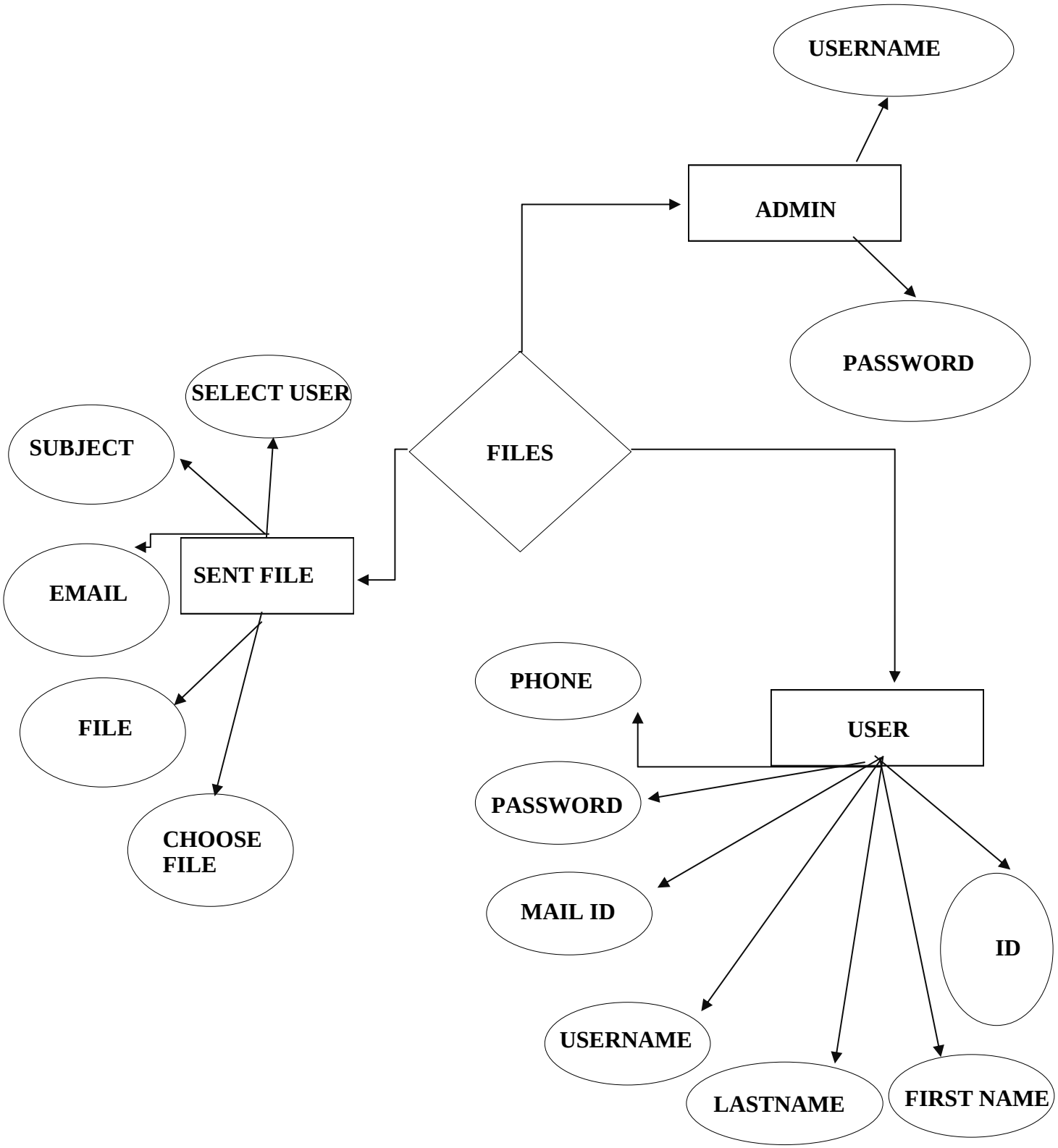
`



4.5.3 Level-1


User
Register
Admin
Login
View User
Sent Files
View Leakers
Key Request
Store Data

4.6 Entity Relationship Diagram



4.7 Data Dictionary

4.7.1 Table name: Admin

Field	Type	Description	Example
Username	varchar(255)	Name for the admin	admin
Password	varchar(30)	Password for the admin	admin

4.7.2 Table name: User registration

Field	Type	Description	Example
First name	text	First part of the name of user	Leo
Last name	text	Second part or initial of the name of user	Daniel
username	varchar(100)	Unique name for the user to login	leo@123
Mail id	varchar(100)	Mail id of the user	leo@gmail.com
password	varchar(100)	Password for the user to login	9721774483
Phone	text	Phone number of the user	8610663372

4.7.3 Table name_ Sent files

Field	Type	Description	Example
Select User	int(255)	Select user to send file	leo
Subject	Varchar(100)	Write subject about the file	PDF
File	varchar(100)	Choose a file	19ubc616.pdf

SYSTEM DEVELOPMENT

5. SYSTEM DEVELOPMENT

5.1 Functional documentation

The project “Data Leakage Detection In Mail Server” has following modules

- **Sign up**
- **Login**
- **Admin**
    - User Registration Request
    - View Leaker
    - Send Files
    - Key Request
    - View Files
- **User**
    - Send Files
    - Key Request
    - View Files

5.1.1 Module description

- **Sign up**

In this module the registration for the user name, email id and password has been done, with the help of a form.

- **Login**

In this module, we will login to the home page of the website using the registered email id and password. In this we will have two login User login and Admin login.

- **Admin**

- User registration request

This module displays the user registration request that are available in this website. According to this admin can manage these courses.

- View leakers

  In this module, the admin can view the details of the leakers who are leaked the files.

- Send files

  In this module, the admin can send a files to the users.

- Key request

  In this module, the admin can request key for the files.

- View files

  In this modules, the user can view the files which is sent by others

➢ **USER**

- Send files

  In this module, the admin can send a files to the users.

- Key request

  . In this module, the admin can request key for the files.

- View files

  In this modules, the user can view the files which is sent by others

5.2 Special Features of the Language

5.2.1 Front End

Java, the language, is a high-level object-oriented programming language, influenced in various ways by C, C++, and Smalltalk, with ideas borrowed from other languages as well. Its syntax was designed to be familiar to those familiar with C-descended "curly brace" languages, but with arguably stronger OO principles than those found in C++, static typing of objects, and a fairly rigid system of exceptions that require every method in the call stack to either handle exceptions or declare their ability to throw them. Garbage collection is assumed, sparing the developer from having to free memory used by obsolete objects.

The Java platform is the ideal platform for network computing. Running across all platforms -- from servers to cell phones to smart cards -- Java technology unifies business infrastructure to create a seamless, secure, networked platform for your business.

The Java platform benefits from a massive community of developers and supporters that actively work on delivering Java technology-based products and services as well as evolving the platform through an open, community-based, standards organization known as the Java Community Process program.

You can find Java technology in cell phones, on laptop computers, on the Web, and even trackside at Formula One Grand Prix races. The fact is today, you can find Java technology just about everywhere!

5.2.2 Back End

MySQL is the world's most used open source relational database management system (RDBMS) as of 2008 that run as a server providing multi-user access to a number of databases. 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 XAMP open source web application software stack—XAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-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, Joomla, Word Press, php BB, MyBB, Drupal and other software built on the LAMP software stack. MySQL is also used in many high-profile, large-scale World Wide Web products, including Wikipedia, Google(though not for searches), Image book Twitter, Flickr, Nokia.com, and YouTube.

Graphical

The official MySQL Workbench is a free integrated environment developed by MySQL AB, that enables users to graphically administer MySQL databases and visually design database structures. MySQL Workbench replaces the previous package of software, MySQL GUI Tools. Similar to other third-party packages, but still considered the authoritative MySQL frontend, MySQL Workbench lets users manage database design & modeling, SQL development (replacing MySQL Query Browser) and Database administration (replacing

MySQL Administrator).

`MySQL Workbench is available in two editions, the regular free and open source Community Edition which may be downloaded from the MySQL website, and the proprietary Standard Edition which extends and improves the feature set of the Community Edition.

Command line

MySQL ships with some command line tools. Third-parties have also developed tools to manage a MySQL server, some listed below.

- Maatkit - a cross-platform toolkit for MySQL, PostgreSQL and Memcached, developed in Perl Maatkit can be used to prove replication is working correctly, fix corrupted data, automate repetitive tasks, and speed up servers. Maatkit is included with several GNU/Linux distributions such as CentOS and Debian and packages are available for Programming

MySQL works on many different system platforms, including AIX, BSDi, FreeBSD, HP-UX, eComStation, i5/OS, IRIX, Linux, Mac OS X, Microsoft Windows, NetBSD, Novell NetWare, OpenBSD, OpenSolaris, OS/2 Warp, QNX, Solaris, Symbian, SunOS, SCO OpenServer, SCO UnixWare, Sanos and Tru64. A port of MySQL to OpenVMS also exists.[32]
MySQL is written in C and C++. Its SQL parser is writtenin yacc, and a home-brewed lexical analyzer. Many programming languages with language-specific APIs include libraries for accessing MySQL databases. These include MySQL Connector/Net for integration with Microsoft's Visual Studio (languages such as C# and VB are most commonly used) and the JDBC driver for Java. In addition, an ODBCinterimage called MyODBC allows additional programming languages that support the ODBC interimage to communicate with a MySQL database, such as ASP or ColdFusion. The HTSQL - URL-based query method also ships with a MySQL adapter, allowing direct interaction between a MySQL database and any web client via structured URLs.

Features

As of April 2009, MySQL offered MySQL 5.1 in two different variants: the open source MySQL Community Server and the commercial Enterprise Server. MySQL 5.5 is offered under the same licenses. They have a common code base and include the following features:

- A broad subset of ANSI SQL 99, as well as extensions
- Cross-platform support
- Stored procedures

- Triggers
- Updatable Views
- Information schema
- Strict mode (ensures MySQL does not truncate or otherwise modify data to conform to an underlying data type, when an incompatible value is inserted into that type)
- X/Open XAdistributed transaction processing (DTP) support; two phase commit as part of this, using Oracle's InnoDB engine
- Independent storage engines (MyISAM for read speed, InnoDB for transactions and referential integrity, MySQL Archive for storing historical data in little space)
- Transactions with the InnoDB, and Cluster storage engines; savepoints with InnoDB
- SSL support
- Query caching
- Sub-SELECTs (i.e. nested SELECTs)
- Full-text indexing and searching using MyISAM engine
- Embedded database library
- Unicode support (however prior to 5.5.3 UTF-8 and UCS-2 encoded strings are limited to the BMP, in 5.5.3 and later use utf8mb4 for full Unicode support)
- ACID compliance when using transaction capable storage engines (InnoDB and Cluster)
- Partitioned tables with pruning of partitions in optimizer
- Shared-nothing clustering through MySQL Cluster
- Hot backup (via mysqlhotcopy) under certain conditions

Multiple storage engines, allowing one to choose the one that is most effective for each table in the application (in MySQL 5.0, storage engines must be compiled in; in MySQL 5.1, storage engines can be dynamically loaded at run time): Native storage engines (MyISAM, Falcon, Merge, Memory (heap), Federated, Archive, CSV, Blackhole, Cluster, EXAMPLE, Maria, and InnoDB, which was made the default as of 5.5). Partner-developed storage engines (solidDB, NitroEDB, ScaleDB, TokuDB, Infobright (formerly Brighthouse), Kickfire, XtraDB, IBM DB2). InnoDB used to be a partner-developed storage engine, but with recent acquisitions, Oracle now owns both MySQL core and InnoDB.

5.3 Pseudo Code

5.3.1 Admin

Login Form

Step 1: Start

Step 2: Enter the username and password.

Step 3: If the username and password is correct then it will redirect to dashboard page.

Step 4: Stop.

Activate Users

Step 1: Start

Step 2: All registered users are shown here.

Step 3: Stop

View Files

Step 1: Start

Step 2: All files sent by others are shown in this page.

Step 3: Stop.

Logout Form

Step 1: Start

Step 2: Invalidate session and redirect to login page.

Step 3: Stop.

5.3.2 Users

Login Form

Step 1: Start

Step 2: Enter the username and password.

Step 3: If the username and password is correct then it will redirect to dashboard page.

Step 4: Stop.

View Files

Step 1: Start

Step 2: All files sent by others are shown in this page.

Step 3: Stop.

Request Key

Step 1: Start

Step 2: Enter the request key in the fields.

Step 3: The user request a private key to download a files.

Step 4: Stop.

Logout Form

Step 1: Start

Step 2: Invalidate session and redirect to login page.

Step 3: Stop.

TESTING

6. TESTING

6.1 Types of Testing Done

Testing is a series of different tests that whose primary purpose is to fully exercise the computer based system. Although each test has a different purpose, all work should verify that all system element have been properly integrated and performed allocated function. Testing is the process of checking whether the developed system works according to the actual requirement and objectives of the system. The philosophy behind testing is to find the errors. A good test is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers the undiscovered error. Test cases are devised with this purpose in mind. A test case is a set of data that the system will process as an input.

- Unit Testing
- Integration Testing
- Validation Testing

6.1.1 Unit Testing

The first test in the development process is the unit test. The source code is normally divided into modules, which in turn are divided into smaller units called units. These units have specific behavior. The test done on these units of code is called unit test. Unit test depends upon the language on which the project is developed. Unit tests ensure that each unique path of the project performs accurately to the documented specifications and contains clearly defined inputs and expected results. Functional and reliability testing in an Engineering environment. Producing tests for the behavior of components (nodes and vertices) of a product to ensure their correct behavior prior to system integration

6.1.2 Integration Testing

Testing in which modules are combined and tested as a group. Modules are typically code modules, individual applications, source and destination applications on a network, etc. Integration Testing follows unit testing and precedes system testing. Testing after the product is code complete. Betas are often widely distributed or even distributed to the public at large in hopes that they will buy the final product when it is release.

6.1.3 Validation Testing

Valid and invalid data should be created and the program should be made to process this data to catch errors. When the user of each module wants to enter into the page by the login page using the use rid and password .If the user gives the wrong password or use rid then the information is provided to the user like “you must enter user id and password”. Here the inputs given by the user are validated. That is password validation, format of date are correct, textbox validation. Changes that need to be done after result of this testing.

6.2 Test Done and output

6.2.1 Unit Testing

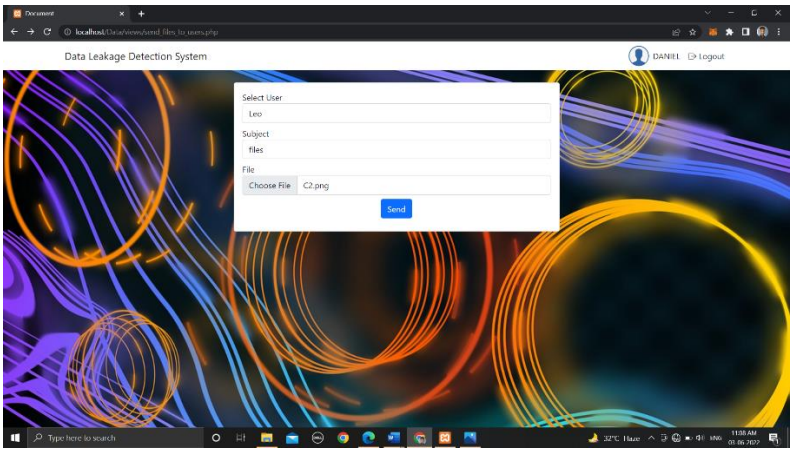


Fig 6.2.1 Make new request samples for unit testing

6.2.2 Validation Testing

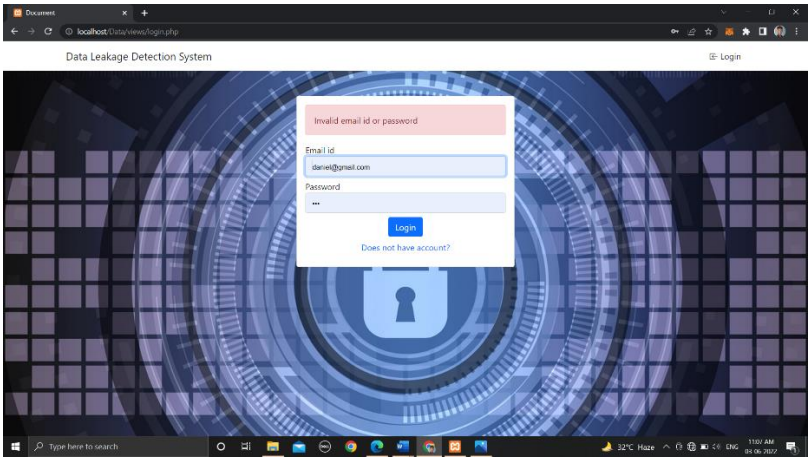


Fig 6.2.2 Login session-Enter invalid email for validation testing

USER MANUAL

7.USER MANUAL

7.1 Hardware Requirements

- Processor : Dual core processor 2.6.0 GHZ
- RAM : 1GB
- Hard disk : 160 GB
- Compact Disk : 650 Mb
- Keyboard : Standard keyboard
- Monitor : 15 inch color monitor

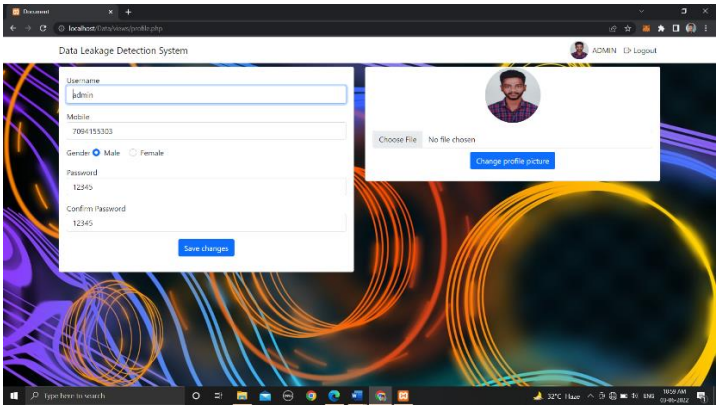
7.2 Software Requirements

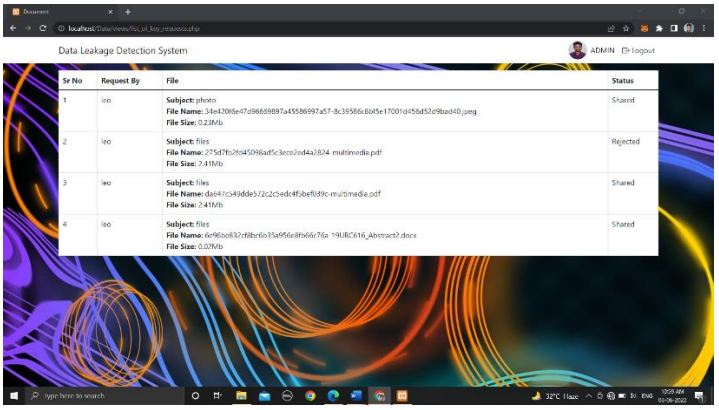
- Operating system : Windows 10
- Web Server : Tomcat Apache 9
- Browser : Internet Explorer
- Documentation : Ms word

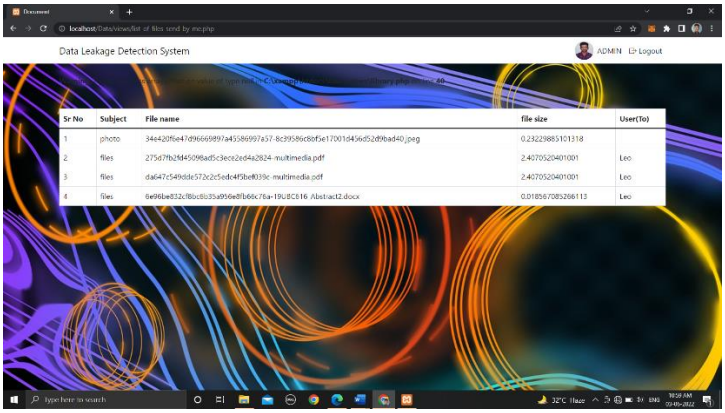
7.3 Installation procedure

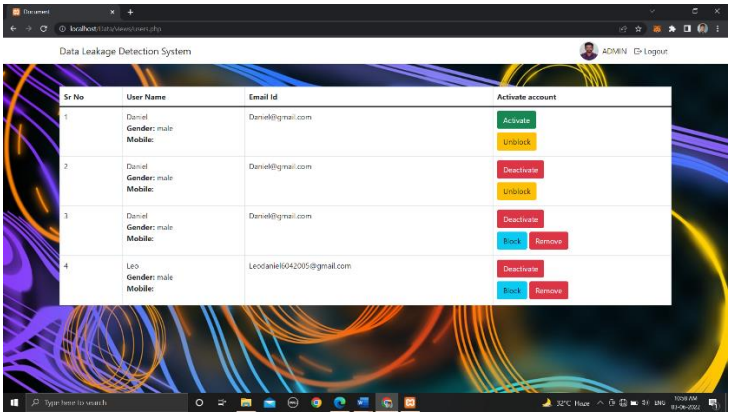
- Need to install Netbeans
- Need to install JDK on both side client and server side
- Need to install MySql on server side
- Need to install XAMPP
- For Client Side: make exe of project and run on the client side.

7.4 sample I/O


Data Leakage Detection System
ADMIN
Logout
Username
admin
Mobile
7094155303
Gender
Male
Female
Password
12345
Confirm Password
12345
Save changes
Choose File
No file chosen
Change profile picture

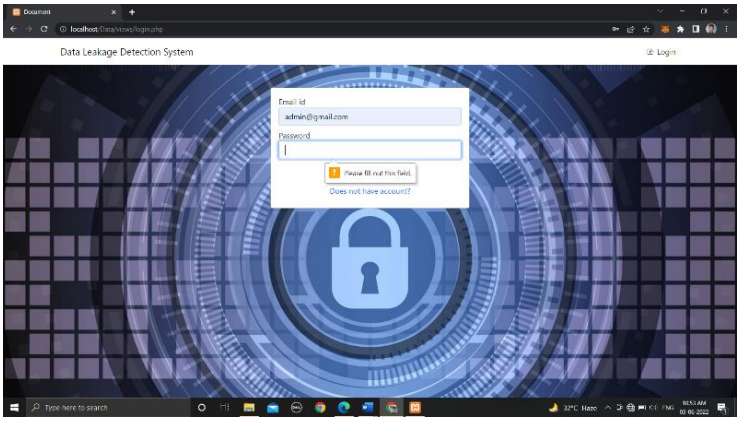

Data Leakage Detection System
ADMIN
Logout
Sr No
Request By
File
Status
1
leo
Subject: photo
File Name: 34e4206e47d96669897a45586997a57-8c39586c8bf5e17001d456d52d9bad40.jpeg
File Size: 0.23Mb
Shared
2
leo
Subject: files
File Name: 275d7fb2fd45096ad5c3ece2ed4a2824-multimedia.pdf
File Size: 2.41Mb
Rejected
3
leo
Subject: files
File Name: da647c549dde572c2c5edc4f5be8039c-multimedia.pdf
File Size: 2.41Mb
Shared
4
leo
Subject: files
File Name: 6e96be832cf8bc6b35a956e8fb66c76a-19UBC616_Abstract2.docx
File Size: 0.02Mb
Shared


Data Leakage Detection System
ADMIN
Logout
Sr No
Subject
File name
file size
User(To)
1
photo
34e4206e47d96669897a45586997a57-8c39586c8bf5e17001d456d52d9bad40.jpeg
0.232298851013 18
2
files
275d7fb2fd45096ad5c3ece2ed4a2824-multimedia.pdf
2.4070520401001
Leo
3
files
da647c549dde572c2c5edc4f5be8039c-multimedia.pdf
2.4070520401001
Leo
4
files
6e96be832cf8bc6b35a956e8fb66c76a-19UBC616_Abstract2.docx
0.018567085266113
Leo

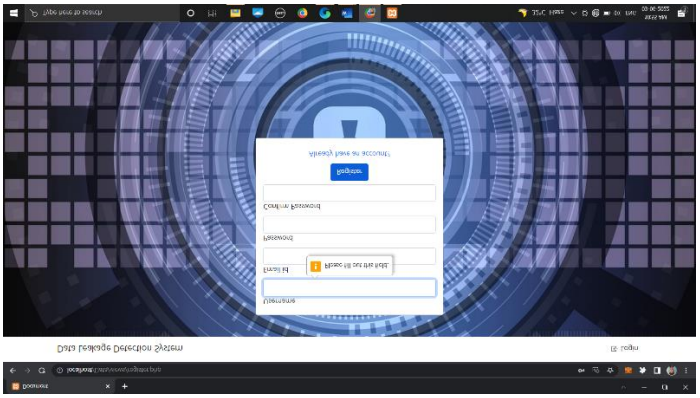

Data Leakage Detection System
ADMIN
Logout
Sr No
User Name
Email Id
Activate account
1
Daniel
Gender: male
Mobile:
Daniel@gmail.com
Activate
Unblock
2
Daniel
Gender: male
Mobile:
Daniel@gmail.com
Deactivate
Unblock
3
Daniel
Gender: male
Mobile:
Daniel@gmail.com
Deactivate
Block
Remove
4
Leo
Gender: male
Mobile:
Leodaniel6042005@gmail.com
Deactivate
Block
Remove

7.5 Error message

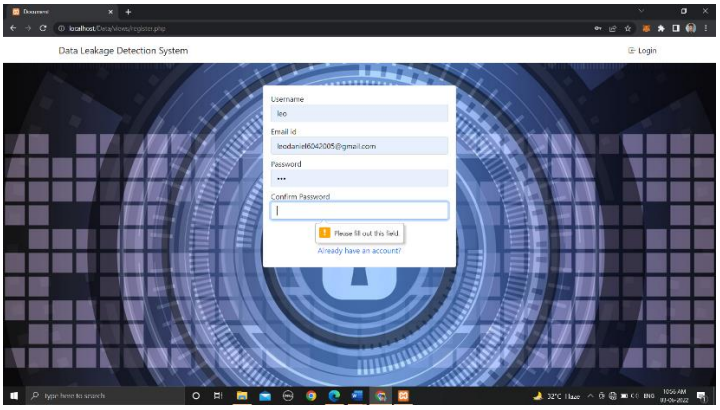
7.5.1 User Register: All fields should be filled



7.5.2 User Register: Email should be in correct format



7.5.3 User Register: Email already Registered



CONCLUSION

8.CONCLUSION

8.1 Summary of the project

Fast detection of data-leakage framework to avoid sensitive data exposure and also provide privacy-preserving to sensitive data. Lucerne search framework to detect the sensible data easily using indexing technique. Levenshtein distance algorithm to detect the shuffling of transferred mail content. To implement the own logics for detect sampling of n-grams in transferred mail content appropriately. I implement threshold rate based on assigning and checking domains based user filtering technique

8.2 Future Enhancements

Since this project was started with very little knowledge about the data leakage, It came to know about the enhancement capability during the process of building it. Some of the scope it can increase for the betterment and effectiveness. The application can never be carried out to the fullest extend in a stipulated time, the main reason why revisions of the application are always introduced in course of time. In future work of this project is to developing using more advantage data detection and highly secure project, Which will more useful for company.

BIBLIOGRAPHY

BIBLIOGRAPHY

Book References

- Gerken, Till, and Tobias Ratschiller. Web Application Development with JAVA. New Riders Publishing, 2000.
- Hawryczkiewycz I.T, "Introduction to System Analysis and Design", PHI, New Delhi, 1994.
- Rajesh Narang, "Database Management Systems", PHI Learning Private Limited, New Delhi, sixth printing, 2010.
- Greenspan, Jay, and Brad Bulger. MySQL/JAVA database applications. John Wiley & Sons, Inc., 2001.
- Scollo, Chris, and Sascha Shumann. Professional JAVA programming. Wrox Press Ltd., 1999.
- Ware, Brent. Open source development with LAMP: using Linux, Apache, MySQL and JAVA. Addison-Wesley Longman Publishing Co., Inc., 2002.
- Yank, Kevin. Build your own database driven website using JAVA& MySQL. Site Point Pty Ltd, 2004.

Website References

- www.freetechbook.com
- www.slideshare.com
- www.w3schools.com
- www.programmersheaven.com
- www.javareferencebook.com

APPENDIX

APPENDIX

```
package Client;
importjava.awt.Color;
 importjava.awt.event.ActionEvent;
 importjava.awt.event.ActionListener;
 importjava.io.DataInputStream;
importjava.io.DataOutputStream;
 importjava.io.InputStream;
 importjava.io.ObjectInputStream;
 importjava.io.OutputStream;
 importjava.net.ServerSocket;
importjava.net.Socket;
importutil.CodeGenerator;
 importutil.ErrorManager;
 importutil.ImagePanel;
 importutil.SignDecode;
 importutil.SignEncode;
 importjava.sql.Connection;
 importjava.sql.DriverManager;
 importjava.sql.Statement;
importjava.util.*;
importjavax.swing.*;
 //import com.intrution.util.SignDecode;
 //import com.intrution.util.SignEncode;
 //import com.intrution.util.SignDecode;
 //import com.intrution.util.SignEncode;
 public class Client extends JFrame
 {
 JFrame frame;
privateJPanelclientpanel;
JLabelmsglabel,publbl,diglbl,msg;
 // JComboBoxmsgcombo;
```

```
 JTextAreafiletextarea;
JTextFieldpubK,digsig;
JButtonreceivedbutton ;
private Thread receiveThrd;
 String uname="";
 public Client(String un)
 {
 uname=un;
 frame=new JFrame("Client");
 clientpanel=new JPanel();
 clientpanel.setLayout(null);
 clientpanel=new ImagePanel(new ImageIcon("./images/receive.jpg").getImage());
clientpanel.setBackground(Color.blue);
 filetextarea=new JTextArea();
 filetextarea.setBounds(5,140,300,150);
clientpanel.add(filetextarea);
publbl=new JLabel("PUBLIC KEY");
 publbl.setBounds(10,20,150,30);
 clientpanel.add(publbl);
 publbl.setForeground(Color.WHITE);
pubK =new JTextField("");
 pubK.setBounds(150,20,150,25);
 pubK.setEditable(false);
clientpanel.add(pubK);
 diglbl=new JLabel("SIGNATURE KEY");
 diglbl.setBounds(10,50,150,30);
 clientpanel.add(diglbl);
 diglbl.setForeground(Color.WHITE);
 digsig =new JTextField("");
 digsig.setBounds(150,50,150,25);
 digsig.setEditable(false);
 clientpanel.add(digsig);
 msg=new JLabel("MESSAGE");
```

```
 msg.setBounds(10,100,100,40);
 clientpanel.add(msg);
 msg.setForeground(Color.MAGENTA);
receivedbutton=new JButton(" FileReceived");
receivedbutton.setBounds(100,250,150,30);
clientpanel.add(receivedbutton);
 receivedbutton.addActionListener(new ActionListener()
{
 public void actionPerformed(ActionEvent e)
 {
msgReciever();
 }
 });
 msgReciever();
 receiveThrd.start();
 frame.add(clientpanel);
 frame.setSize(350,400);
frame.setLocation(200, 300);
 frame.setVisible(true);
 }
 /*
 public void msgReciever()
{
 receiveThrd = new Thread(new Runnable()
 {
 public void run()
 {
 try
{
 do
 {
newErrorManager("MSGRECEIVER....");
 ServerSocketss=new ServerSocket(2000);
```

```
 Socket s=ss.accept();
 ObjectInputStreamrcvObj = new ObjectInputStream(s.getInputStream());
 String str=(String)rcvObj.readObject();
StringTokenizerstrItr = new StringTokenizer(str, "|",true);
 String src1S = strItr.nextToken();
strItr.nextToken();
String rtr1S=strItr.nextToken();
 strItr.nextToken();
 String rtr2S = strItr.nextToken();
  strItr.nextToken();
//rtr3S = strItr.nextToken();
//forwardTxt.setText(rtr2S);
 //strItr.nextToken();
String destS = strItr.nextToken();
 //destTextfield.setText(destS);
 strItr.nextToken();
 String msgS = strItr.nextToken();
filetextarea.setText(msgS);
ss.close();
 s.close();
 // rcvObj.close();
 }
 while (true);
 }
 catch (Exception e)
 {
 System.out.println("fsdfsdfds"+ e);
}
 }
 });
 //receiveThrd.start();
 }
 */
```

```
 public void msgReciever()
{
 receiveThrd = new Thread(new Runnable()
 {
 public void run()
 {
 Try
 {
 do
{
  // new ErrorManager("Inside Receiver");
 String msgS, rtr1, rtr2, whoIs, key,src,dest,pri,pub;
ServerSocketrcvSkt = new ServerSocket(1500);
 // new ErrorManager("After Socket Inside Receiver");
 Socket skt = rcvSkt.accept();
whoIs = skt.getInetAddress().getHostName();
 ObjectInputStreamrcvObj = new ObjectInputStream(skt .getInputStream());
 msgS = (String) rcvObj.readObject();
 String url="jdbc:mysql://localhost:3306/intrusion";
 Class.forName("com.mysql.jdbc.Driver");
Connection con=DriverManager.getConnection(url,"root","root");
 Statement st=con.createStatement();
 StringTokenizerstrItr = new StringTokenizer(msgS, "|",true);
 // new ErrorManager("After Receiver String Token");
System.out.println(msgS);
 src=strItr.nextToken();
strItr.nextToken();
 rtr1 = strItr.nextToken();
 strItr.nextToken();
rtr2 = strItr.nextToken();
 strItr.nextToken();
dest = strItr.nextToken();
 strItr.nextToken();
```

```
 msgS = strItr.nextToken();
strItr.nextToken();
pub = strItr.nextToken();
 strItr.nextToken();
 pubK.setText(pub);
 pri = strItr.nextToken();
//strItr.nextToken();
String optionMsg = "Enter your public key send message";
String code = JOptionPane.showInputDialog(null, optionMsg,"Input Box", 1);
if((pubK.getText()).equals(code))
{
 CodeGenerator get = new CodeGenerator();
 String digsign=get.codeCreate();
 newSignEncode(digsign);
 SignDecode digital = new SignDecode();
 digsig.setText(digsign+".sig");
 String signFileMsg = "Enter your digital signature file name";
String digiFile = JOptionPane.showInputDialog(null, signFileMsg, "Input Box", 1);
 String qq=digital.signatureDecode(digiFile)+".sig";
 String qry="update register set signature='"+ digsig.getText() +"' where username='"+ uname+"'";
 st.executeUpdate(qry);
//
 filetextarea.append(qq+">>>>"+digsig.getText());
if((digsig.getText()).equals(qq))
{
 // msgTxt.append(whoIs+" >> "+msg + "\n");
filetextarea.append(src +">>" +msgS+"\n");
}
 Else
 {
 newErrorManager("Your Digital file mismatch");
newErrorManager("Intruder Found...Lost ur Message..");
 Socket s1=new Socket("localhost",2020);
```

```
 OutputStreamos=s1.getOutputStream();
 DataOutputStream dos=new DataOutputStream(os);
 dos.writeUTF("Digital Signature");
 OutputStream os1=s1.getOutputStream();
 DataOutputStream dos1=new DataOutputStream(os1);
 InputStream is1=s1.getInputStream();
 DataInputStream dis1=new DataInputStream(is1);
 dos1.writeUTF(dest);
newErrorManager("Message send");
 s1.close();
 os.close();
 dos.close();
 }
 }
 else
 {
newErrorManager("Invalid Key ...Lost ur Message");
 }
skt.close();
 rcvSkt.close();
 }
 while (true);
 }
catch (Exception e)
 {
 newErrorManager("Please Try again later");
 e.printStackTrace();
  }
 }
 });
 }
 //public static void main(String[] args) {
// new Client();
```

```
 //}
 }

package Client;
 importjava.awt.Color;
importjava.awt.event.ActionEvent;
 importjava.sql.*;
importjava.awt.event.ActionListener;
importjava.sql.Connection;
importjava.sql.DriverManager;
 importjava.sql.Statement;
 importjavax.swing.ImageIcon;
importjavax.swing.*;
 importRegister.register;
 importutil.ErrorManager;
importutil.ImagePanel;
 public class Login extends JFrame
{
 JFrame frame;
privateJPanelloginpanel;
 JLabelnamelabel,pwdlabel;
JTextFieldnametextfield;
JPasswordFieldpwdtextfield;
 JButtonresetbutton,loginbutton,regbtn;

public Login()
 {
 frame=new JFrame("LOGIN");
 loginpanel=new JPanel();
 frame.setLayout(null);
 //loginpanel.setLayout(null);
namelabel=new JLabel("USERNAME:");
 namelabel.setBounds(20, 20, 150,25);
```

```
pwdlabel=new JLabel("PASSWORD");
 pwdlabel.setBounds(20, 60, 150,25);
 nametextfield=new JTextField();
 nametextfield.setBounds(100, 20, 100, 25);
 pwdtextfield=new JPasswordField();
 pwdtextfield.setBounds(100, 60, 100, 25);
 loginbutton=new JButton("LOGIN");
loginbutton.setBounds(25,100,100,25);
 resetbutton=new JButton("RESET");
resetbutton.setBounds(125,100,100,25);
 regbtn=new JButton("Please register");
 regbtn.setBounds(50,150,160,30);
regbtn.setForeground(Color.red);
 loginpanel = new ImagePanel(new ImageIcon("./images/win.jpg").getImage());
//new JPanel();
 loginpanel.setBackground(Color.blue);
loginpanel.add(resetbutton);
loginpanel.add(loginbutton);
loginpanel.add(pwdtextfield);
loginpanel.add(nametextfield);
 loginpanel.add(pwdlabel);
 loginpanel.add(namelabel);
loginpanel.add(regbtn);
 frame.add(loginpanel);
frame.setSize(250,250);
frame.setLocation(200, 300);
 frame.setVisible(true);
 regbtn.addActionListener(new ActionListener()
 {
 public void actionPerformed(ActionEvent e)
{
 new register();
 frame.setVisible(false);
```

```
}
 });
 resetbutton.addActionListener(new ActionListener()
{
 public void actionPerformed(ActionEvent e)
{
 nametextfield.setText("");
pwdtextfield.setText("");
}
 });
 loginbutton.addActionListener(new ActionListener()
{
 public void actionPerformed(ActionEvent e)
{
 try
{
 String url="jdbc:mysql://localhost:3306/intrusion";
Class.forName("com.mysql.jdbc.Driver");
 Connection con=DriverManager.getConnection(url,"root","root");
Statement st=con.createStatement();
 String uname=nametextfield.getText();
String pass=pwdtextfield.getText();
String secure=MD5.getHash(pass);
 if(uname.equals("") &&pass.equals(""))
{
 newErrorManager("Please Fill in the Value");
}
 else {
 String qry="select * from register where username='"+ uname +"' and password='"+ secure +"'";
 ResultSetrs=st.executeQuery(qry);
 String un="",pwd="";
 if (rs.next())
{
```

```
 un=rs.getString("username");
 pwd=rs.getString("password");
}
 if (uname.equals(un) &&secure.equals(pwd))
 {
 new Client(un);
 frame.setVisible(false);
 }
 Else
 {
 newErrorManager("Login Failed");
}
 }
 }
 catch(Exception ex)
 {
newErrorManager("Exception"+ ex);
 }
 }
 });
 }
 public static void main(String[] args)
{
 new Login();
 }
 }

packageIntrusinDetection;
 importjava.awt.Color;
 importjava.awt.event.ActionEvent;
 importjava.awt.event.ActionListener;
 importjava.io.DataInputStream;
importjava.io.DataOutputStream;
```

```
importjava.io.InputStream;
importjava.io.OutputStream;
 importjava.net.Socket;
importjavax.swing.*;
 importutil.ImagePanel;
 public class Intrusion extends JFrame
 {
 JFrame frame;
privateJPanelDospanel;
 JLabeliplabel,portlabel;
JComboBoxipcombo;
privateJTextFieldprttxt,iptxt;
 JButtonattackbutton,exit ;
 public String flSer1="";
 publicint port=0;
Socket sndFl;
 Intrusion()
 {
 frame=new JFrame("DOS Attack");
 Dospanel=new JPanel();
 Dospanel.setLayout(null);
Dospanel=new ImagePanel(new ImageIcon("./images/3.jpg").getImage());
 Dospanel.setBackground(Color.CYAN);
iplabel=new JLabel("IP Address :");
 iplabel.setBounds(50,50,100,30);
 iplabel.setForeground(Color.DARK_GRAY);
Dospanel.add(iplabel);
iptxt=new JTextField();
iptxt.setBounds(150,50,150,30);
Dospanel.add(iptxt);
 portlabel=new JLabel("Port Number :");
portlabel.setBounds(50,100,100,30);
portlabel.setForeground(Color.DARK_GRAY);
```

```
Dospanel.add(portlabel);
 prttxt=new JTextField();
 prttxt.setBounds(150,100,150,30);
 Dospanel.add(prttxt);
 attackbutton=new JButton(" Attack");
 attackbutton.setForeground(Color.RED);
 attackbutton.setBounds(100,150,100,30);
 Dospanel.add(attackbutton);
exit=new JButton(" EXIT");
 exit.setBounds(100,200,100,30);
 exit.setForeground(Color.RED);
Dospanel.add(exit);
 exit.addActionListener(new ActionListener()
{
 public void actionPerformed(ActionEvent e)
{
 {
 System.out.println("\n exit_actionPerformed(ActionEvent e) called.");
// TODO: Add any handling code here
System.exit(0);
}
 }
});
 frame.add(Dospanel);
 frame.setSize(350,300);
 frame.setLocation(200, 300);
 frame.setVisible(true);
Dospanel.setBorder(BorderFactory.createTitledBorder("DOS"));
 Dospanel.setBorder(BorderFactory.createRaisedBevelBorder());
 //}
attackbutton.addActionListener(new ActionListener()
 {
 public void actionPerformed(ActionEvent e)
```

```
 {
//System.out.println("\njButton1_actionPerformed(ActionEvent e) called.");
 // TODO: Add any handling code here
try
 {
 flSer1=iptxt.getText();
 port=Integer.parseInt(prttxt.getText());
sndFl=new Socket(flSer1,port);
//DataOutputStream dos=new DataOutputStream(sndFl.getOutputStream());
 //dos.writeUTF("DOS");
 //DataOutputStream dos1=new DataOutputStream(sndFl.getOutputStream());

 //dos1.writeUTF(flSer1);
//new sound1();
 //Socket s=new Socket("localhost", 2010);
 OutputStreamos=sndFl.getOutputStream();
DataOutputStream dos=new DataOutputStream(os);
InputStream is=sndFl.getInputStream();
DataInputStream dis=new DataInputStream(is);
 //dos.writeUTF(encodedtxt.getText());
 dos.writeUTF("DOS");
 sndFl.close();
 dos.close();
}
 catch (Exception as)
 {
as.printStackTrace();
 }
 }
 });
}
 public static void main(String[] args)
{
```

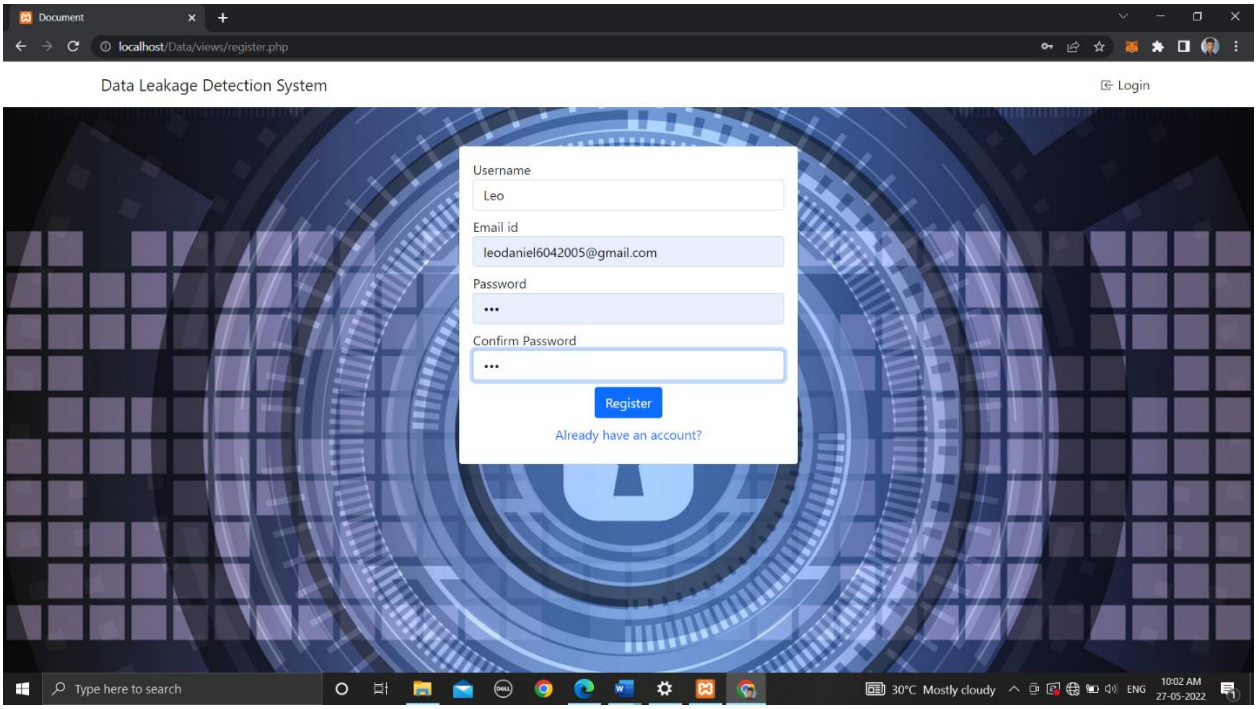
```
 new Intrusion();
 }
 }


packageKeygenerator;
 importjava.awt.Color;
importjava.io.ObjectInputStream;
 importjava.io.ObjectOutputStream;
importjava.net.ServerSocket;
importjava.net.Socket;
 importjava.util.ArrayList;
 importjavax.swing.*;
 import util.*;
public class Key extends JFrame
{
JFrame frame;
 privateJPanelkeypanel;
 JLabelpubliclabel,privatelabel;
 privateJProgressBarprg;
privateJTextFieldpublictextfield,privatetextfield;
 privateArrayListavailNodes;
 private String netView[];
 privatebooleanitrFlg=true;
 Thread serChk,ser;
 public Key()
 {
frame=new JFrame("KeyGenerator");
 keypanel=new JPanel();
 keypanel.setLayout(null);
 keypanel = new ImagePanel(new ImageIcon("./images/key1.jpg").getImage());
//new JPanel();
 keypanel.setBackground(Color.blue);
```

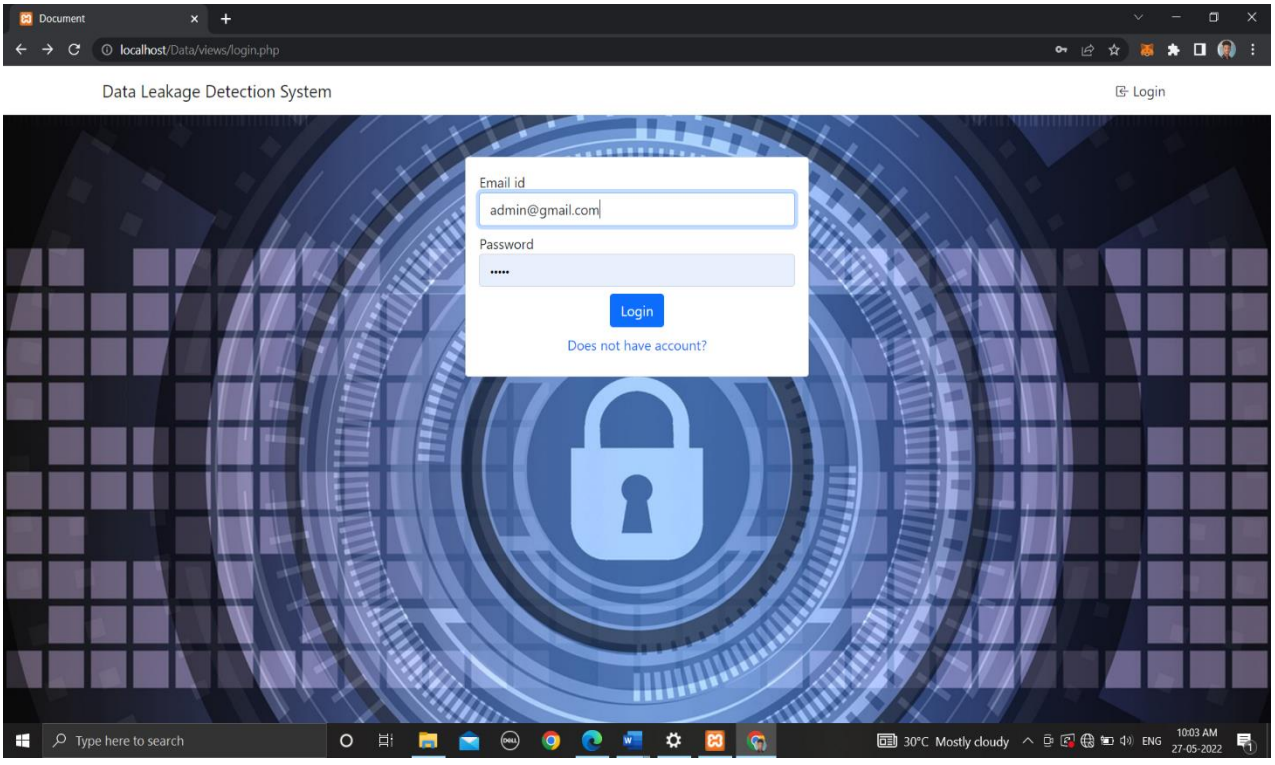
```
publiclabel=new JLabel("PUBLIC KEY");
 publiclabel.setBounds(2,70,150,30);
 publiclabel.setForeground(Color.DARK_GRAY);
 keypanel.add(publiclabel);
 privatelabel=new JLabel("PRIVATE KEY");
 privatelabel.setBounds(2,140,150,30);
 privatelabel.setForeground(Color.DARK_GRAY);
keypanel.add(privatelabel);
 publictextfield=new JTextField();
 publictextfield.setBounds(90,70,150,30);
 publictextfield.setEditable(false);
 keypanel.add(publictextfield);
 privatetextfield=new JTextField();
 privatetextfield.setBounds(90,140,150,30);
 privatetextfield.setEditable(false);
 keypanel.add(privatetextfield);
 frame.add(keypanel);
 //frame.setBackground(Color.BLUE);
frame.setSize(270,250);
 frame.setLocation(200, 300);
 frame.setVisible(true);
 checkServer();
serChk.start();
 responseKey();
ser.start();
 }
 public void responseKey()
{
 ser = new Thread(new Runnable()
 {
 public void run()
{
 try
```

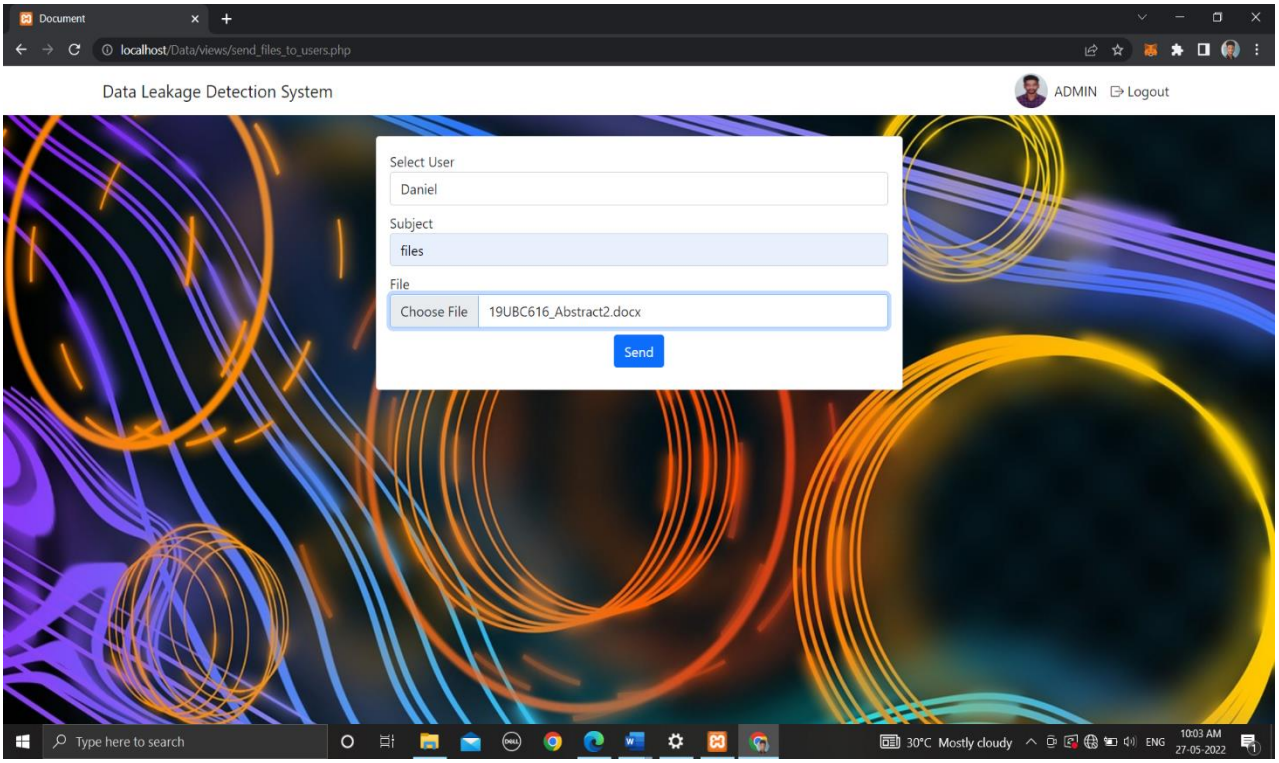
```
 {
 do
{
 AvailableNode nodes = new AvailableNode();
 availNodes = nodes.addAvailNode();
 netView = (String[]) availNodes.toArray(new String[availNodes.size()]);
 String whoIs;
 ServerSocket rcvSkt1 = new ServerSocket(6655);
 Socket skt1 = rcvSkt1.accept();
 ObjectInputStream rcvObj1 = new ObjectInputStream(skt1 .getInputStream());
 whoIs = (String) rcvObj1.readObject();
//senderT.setText(whoIs);
 for (intchkWho = 0; chkWho<netView.length; chkWho++)
{
If
 (whoIs.equalsIgnoreCase(netView[chkWho]))
{
 itrFlg = false;
break;
}
 }
 if(itrFlg==false)
 {
 CodeGenerator get = new CodeGenerator();
String pubKey = get.codeCreate();
 publictextfield.setText(pubKey);
String priKey = get.codeCreate();
 privatetextfield.setText(priKey);
 sendKey(whoIs, pubKey, priKey);
 }
 else
{
newErrorManager("Found Intruder : You dont have rights to communicate");
```

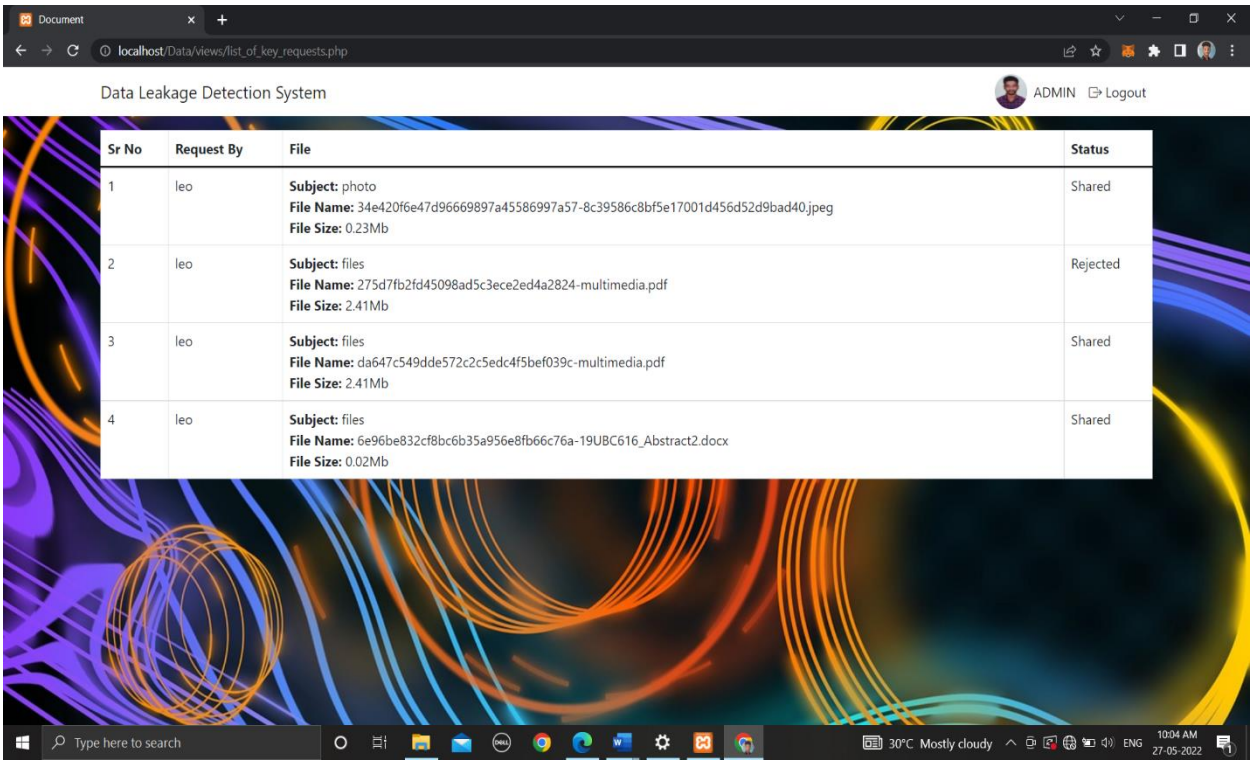
```
}
 itrFlg = true;
skt1.close();
 rcvSkt1.close();
}
 while (true);
 }
catch (Exception e)
{
 newErrorManager("Server : "+e.toString());
 e.printStackTrace();
}
 }
 });
 }
 public void sendKey(String destination, String puKey, String prKey)
{
 try
 {
 Socket sendSkt1 = new Socket(destination, 5566);
 ObjectOutputStream sendObj1 = new ObjectOutputStream(sendSkt1 .getOutputStream());
 sendObj1.flush();
 String packMsg = puKey + "|" + prKey ;
sendObj1.writeObject(packMsg);
 sendObj1.close();
 sendSkt1.close();
 }
 catch (Exception e)
 {
newErrorManager("Server : "+e.toString());
 e.printStackTrace();
}
 }
```

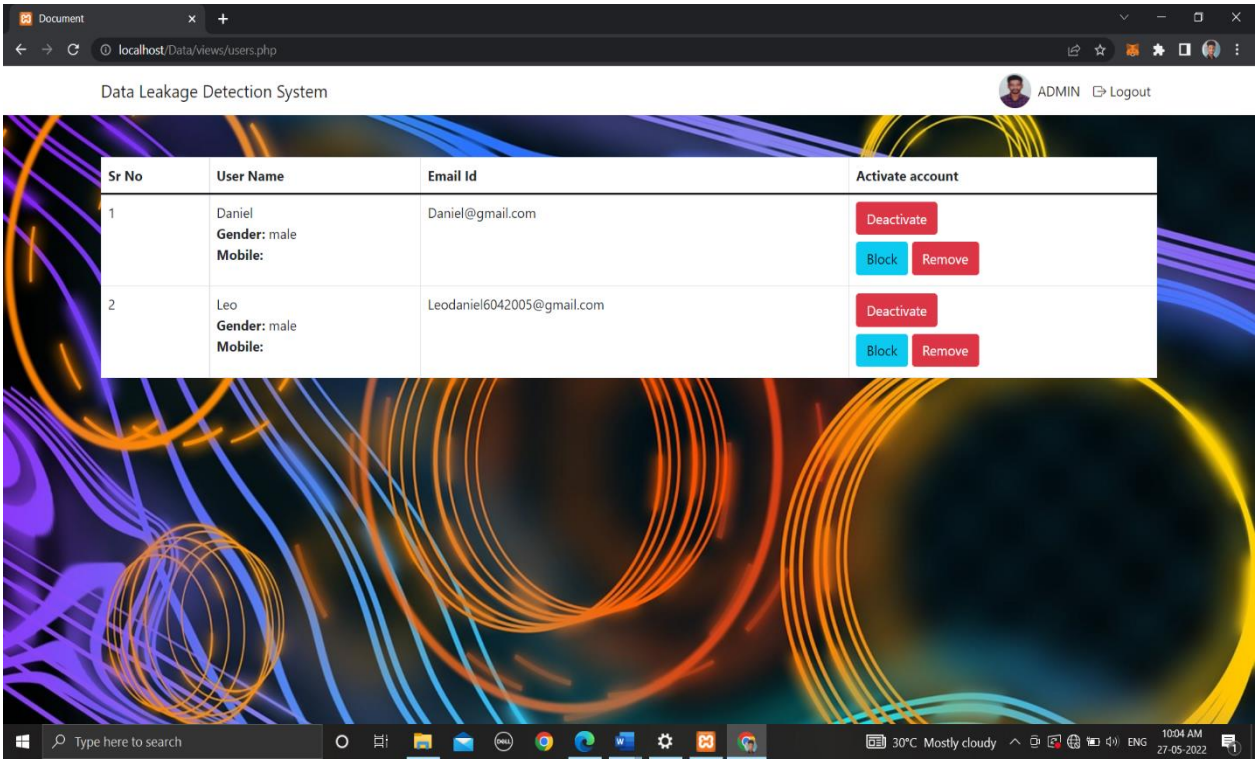
```
public void checkServer()
{
 serChk = new Thread(new Runnable()
 {
public void run()
 {
 try
{
 do
{
 ServerSocket rcvSkt1 = new ServerSocket(4444);
 Socket skt1 = rcvSkt1.accept();
 skt1.close();
 rcvSkt1.close();
}
 while (true);
 }
 catch (Exception e)
{
 newErrorManager("Server : "+e.toString());
e.printStackTrace();
 }
 }
 });
 }
}
```

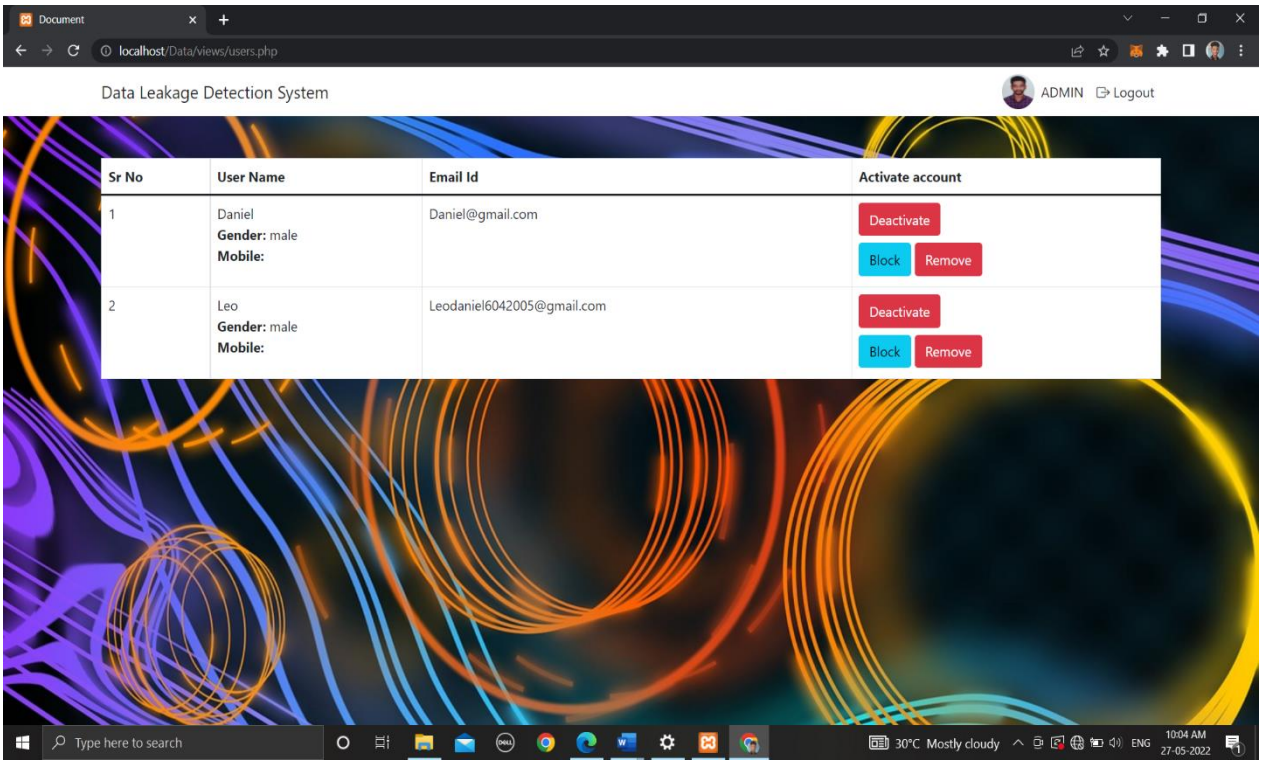

Document
localhost/Data/views/register.php
Data Leakage Detection System
Login
Username
Leo
Email id
leodaniel6042005@gmail.com
Password
Confirm Password
Register
Already have an account?
Type here to search
30°C Mostly cloudy
ENG
10:02 AM
27-05-2022


Document
localhost/Data/views/login.php
Data Leakage Detection System
Login
Email id
admin@gmail.com
Password
Login
Does not have account?
Type here to search
30°C Mostly cloudy
ENG
10:03 AM
27-05-2022


Document
localhost/Data/views/send_files_to_users.php
Data Leakage Detection System
ADMIN
Logout
Select User
Daniel
Subject
files
File
Choose File
19UBC616_Abstract2.docx
Send
Type here to search
30°C Mostly cloudy
ENG
10:03 AM
27-05-2022


Document
localhost/Data/views/list_of_key_requests.php
Data Leakage Detection System
ADMIN
Logout
Sr No
Request By
File
Status
1
leo
Subject: photo
File Name: 34e420f6e47d96669897a45586997a57-8c39586c8bf5e17001d456d52d9bad40.jpeg
File Size: 0.23Mb
Shared
2
leo
Subject: files
File Name: 275d7fb2fd45098ad5c3ece2ed4a2824-multimedia.pdf
File Size: 2.41Mb
Rejected
3
leo
Subject: files
File Name: da647c549dde572c2c5edc4f5bef039c-multimedia.pdf
File Size: 2.41Mb
Shared
4
leo
Subject: files
File Name: 6e96be832cf8bc6b35a956e8fb66c76a-19UBC616_Abstract2.docx
File Size: 0.02Mb
Shared
Type here to search
30°C Mostly cloudy
ENG
10:04 AM
27-05-2022


Document
localhost/Data/views/users.php
Data Leakage Detection System
ADMIN
Logout
Sr No
User Name
Email Id
Activate account
1
Daniel
Gender: male
Mobile:
Daniel@gmail.com
Deactivate
Block
Remove
2
Leo
Gender: male
Mobile:
Leodaniel6042005@gmail.com
Deactivate
Block
Remove
Type here to search
30°C Mostly cloudy
ENG
10:04 AM
27-05-2022


Document
localhost/Data/views/users.php
Data Leakage Detection System
ADMIN
Logout
Sr No
User Name
Email Id
Activate account
1
Daniel
Gender: male
Mobile:
Daniel@gmail.com
Deactivate
Block
Remove
2
Leo
Gender: male
Mobile:
Leodaniel6042005@gmail.com
Deactivate
Block
Remove
Type here to search
30°C Mostly cloudy
ENG
10:04 AM
27-05-2022