ABSTRACT

INTRODUCTION

SYSTEM DESIGN

TESTING

IMPLEMENTATION

CONCLUSION

BIBLIOGRAPHY

CODING

PLAGAIRISM REPORT

SOFTWARE REQUIREMENT ANALYSIS AND SPECIFICATION

3. SYSTEM DESIGN

System Design is the process or art of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. One could see it as the application of systems theory to product development. There is some overlap and synergy between the disciplines of systems analysis, systems architecture, and systems engineering.

3.1 Data Design

Database design is part of the development process. In the linear development cycle, it is used during the system requirements phase to construct the data components of the analysis model. This model represents the major data objects and the relationships between them. It should not be confused with data analysis, which takes place in the system design phase. As in a DFD, a model of data consists of a number of symbols joined up according to certain conventions. System designers describe this conceptual modeling using symbols from a modeling method known as entity relationship analysis.

Entity Relationship Diagram

Entity relationship analysis uses three major abstractions to describe data. These are

- Entities, which are distinct things in the enterprise.
- Relationships, which are meaningful interactions between objects,
- Attributes, which are the properties of the entities and relationships.
- The relative simplicity and pictorial clarity of this diagramming technique may well account in large part for the widespread use of the ER model. Such a diagram consists of the following major components:

3.3 UML Design

To understand the UML, you need to form a conceptual model of the language, and this requires learning three major elements: the UML's basic building blocks, the rules that dictate how these building blocks may be put together, and some common mechanisms that apply throughout the UML. Once you have grasped these ideas, you will be able to read UML models and create some basic ones. As you gain more experience in applying UML, you can build on this conceptual model by using more advanced features of the language.

Building Blocks of the UML

The vocabulary of the UML encompasses three kinds of building blocks.

1. Things
2. Relationships
3. Diagrams

Things are abstractions that are first-class citizens in a model; relationships tie these things together, and diagrams group interesting collections of things.

Things in the UML

There are four kinds of things in the UML.

1. Structural things
2. Behavioral things
3. Grouping things
4. Annotational things

Structural things

Structural things are the nouns of UML models. These are the mostly static parts of a model, representing elements that are either conceptual or physical. In all, there are seven kinds of structural things.

4. TESTING

In general, software engineers distinguish software faults from software failures. In the event of a failure, the software does not do what the user expects. A fault is a programming error that may or may not actually manifest as a failure. A fault can also be described as an error in the correctness of the semantics of a computer program. A fault will become a failure if the exact computation conditions are met, one of which is that the faulty portion of computer software executes on the CPU. A fault can also turn into a failure when the software is ported to a different hardware platform, compiled with a different compiler, or extended.

Software testing may be viewed as a sub-field of Software Quality Assurance but typically exists independently (and there may be no SQA areas in some companies). In SQA, software process specialists and auditors take a broader view on software and its development. They examine and change the software engineering process itself to reduce the number of faults that end up in the code or deliver it faster.

Regardless of the methods used or level of formality involved the desired result of testing is a level of confidence in the software so that the organization is confident that the software has an acceptable defect rate. What constitutes an acceptable defect rate depends on the nature of the software. An arcade video game designed to simulate flying an airplane would presumably have a much higher tolerance for defects than the software used to control an actual airliner.

A problem with software testing is that the number of defects in a software product can be very large, and the number of configurations of the product can be even larger. Bugs that occur infrequently are difficult to find in testing. A rule of thumb is that a system that is expected to function without faults for a certain length of time must have already been tested for at least that length of time. This has severe consequences for projects to write long-

CONTENTS

BIBLIOGRAPHY

Appendix – A

- URL Listings
- References

Appendix – B

- Glossary

Appendix – C

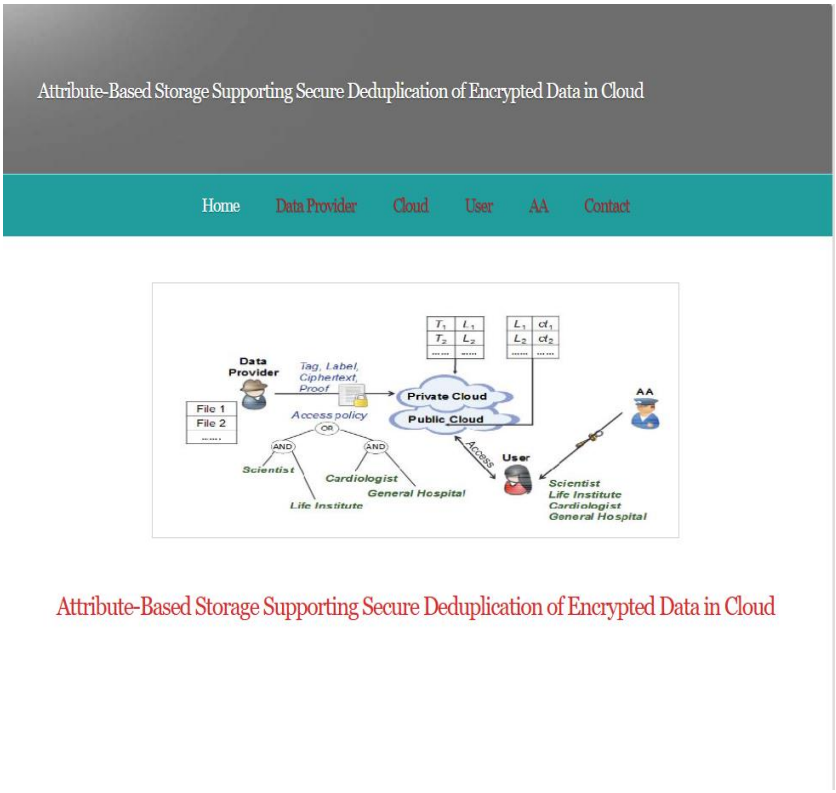
- List of Tables
- List of Figures
- List of Screens and Reports

Appendix – D

- Coding
- Plagiarism report

5.1 Sample Screens

Home page



Screen 5.1.1 Home page

Description: This Screen shows Home page

Appendix - A

URL Listing

Websites	Reference Books
www.elsevier.com	Some of the advanced information about Java such as its driver types and the connectivity information about JDK.
www.training-classes.com	The Designing part information hasbeen gathered.
http://www.wikipedia.org	Searching for any information that canbe used in documentation.
http://www.google.co.in	Details about the code have been collected have been collected.
http://www.google.co.in	Any information searching and downloading.

References

[1] D. Quick, B. Martini, and K. R. Choo, Cloud Storage Forensics. Syngress Publishing / Elsevier, 2014. [Online].Available: http://www.elsevier.com/books/cloud-storageforensics/quick/978-0-12-419970-5

[2] K. R. Choo, J. Domingo-Ferrier, and L. Zhang, “Cloud cryptography: Theory, practice and future research directions,” Future Generation Comp. Syst., vol. 62, pp. 51–53, 2016.

[3] K. R. Choo, M. Herman, M. Organ, and B. Martini, “Cloud forensics: State-of-the-art and future directions,” Digital Investigation, vol. 18, pp. 77–78, 2016.

[4] Y. Yang, H. Zhu, H. Lu, J.Weng, Y. Zhang, and K. R. Choo, “Cloud based data sharing with fine-grained proxy re-encryption,” Pervasive and Mobile Computing, vol. 28, pp. 122–134, 2016.

[5] D. Quick and K. R. Choo, “Google drive: Forensic analysis of data remnants,” J. Network and Computer Applications, vol. 40, pp. 179–193, 2014.

[6] A. Sahai and B. Waters, “Fuzzy identity-based encryption,” in Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings, ser. Lecture Notes in Computer Science, vol. 3494. Springer, 2005, pp. 457–473.

[7] B. Zhu, K. Li, and R. H. Patterson, “Avoiding the disk bottlenecking the data domain deduplication file system,” in 6th USENIXConference on File and Storage Technologies, FAST 2008, February 26-29, 2008, San Jose, CA, USA. USENIX, 2008, pp. 269–282.

[8] M. Bellaire, S. Keelveedhi, and T. Ristenpart, “Message-locked encryption and secure deduplication,” in Advances in Cryptology-EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings, ser. Lecture Notes in Computer Science, vol. 7881. Springer, 2013, pp. 296–312.

[9] M. Abaci, D. Bone, I. Maroon, A. Raghunathan, and G. Segev,“Message-locked encryption for lock-dependent messages,” in Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I, ser. Lecture Notes in Computer Science, vol. 8042. Springer, 2013, pp. 374–391.

[10] S. Keelveedhi, M. Bellaire, and T. Ristenpart, "Dupless: Serveraidedencryption for reduplicated storage," in Proceedings of the22th USENIX Security Symposium, Washington, DC, USA, August14-16, 2013. USENIX Association, 2013, pp. 179–194.

[11] M. Bellaire and S. Keelveedhi, "Interactive message-locked encryption and secure deduplication," in Public-Key Cryptography – PKC2015 - 18th IACR International Conference on Practice and Theory in Public-Key Cryptography, Gaithersburg, MD, USA, March 30 – April1, 2015, Proceedings, ser. Lecture Notes in Computer Science, vol.9020. Springer, 2015, pp. 516–538.

[12] S. Bugiel, S. N¨ urnberger, A. Sadeghi, and T. Schneider, "Twin clouds: Secure cloud computing with low latency - (full version),"in Communications and Multimedia Security, 12th IFIP TC 6 / TC11 International Conference, CMS 2011, Ghent, Belgium, October 19-21,2011. Proceedings, ser. Lecture Notes in Computer Science, vol.7025. Springer, 2011, pp. 32–44.

[13] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof-systems (extended abstract)," in Proceedings of the 17th Annual ACM Symposium on Theory of Computing, May 6-8, 1985, Providence, Rhode Island, USA. ACM, 1985, pp. 291–304.

[14] M. Fechlin and R. Fechlin, "Efficient non-malleable commitment schemes," in Advances in Cryptology - CRYPTO 2000, 20th Annual International Cryptology Conference, Santa Barbara, California, USA,August 20-24, 2000, Proceedings, ser. Lecture Notes in Computer Science, vol. 1880. Springer, 2000, pp. 413–431.

[15] S. Goldwasser and S. Micali, "Probabilistic encryption," J. Comput.Syst. Sci., vol. 28, no. 2, pp. 270–299, 1984.

[16] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proceedings of the 13th ACM Conference on Computer and Communications November 3, 2006, ser. Lecture Notes in Computer Science, vol.5126. Springer, 2006, pp. 89–98.

[17] R. Ostrovsky, A. Sahai, and B.Waters, "Attribute-based encryption with no-monotonic access structures," in Proceedings of the 2007ACM Conference on Computer and Communications Security, CCS2007, Alexandria, Virginia, USA, October 28-31, 2007. ACM, 2007, pp. 195–203.

[18] A. B. Lewko and B. Waters, "Unbounded HIBE and attributebasedencryption," in Advances in Cryptology - EUROCRYPT 2011 -30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings, ser. Lecture Notes in Computer Science, vol. 6632. Springer, 2011, pp. 547–567.

[19] J. Bettencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in 2007 IEEE Symposium on Security And Privacy (S&P 2007), 20-23 May 2007, Oakland, California, USA.IEEE Computer Society, 2007, pp. 321–334.

[20] L. Cheung and C. C. Newport, "Provably secure ciphertext policy ABE," in Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA,October 28-31, 2007. ACM, 2007, pp. 456–465.

[21] V. Goyal, A. Jain, O. Pandey, and A. Sahai, "Bounded ciphertext policy attribute based encryption," in Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik

Appendix - B

Glossary

TA	:	Trusted Authority
ESP	:	Encryption Service Provider
DSP	:	Decryption Service Provider
DBMS	:	Database Management System
GUI	:	Graphic User Interface
HTML	:	Hyper Text Markup Language
HTTP	:	Hyper Text Transfer Protocol
JSP	:	Java Server Pages
J2SE	:	Java2 Standard Edition
SQL	:	Structured Query Language
TCP	:	Transmission Control Protocol
UML	:	Unified Modeling Language
URL	:	Uniform Resource Locator
WWW	:	World Wide Web
JS	:	Java Script
SRS	:	Software Requirement specification
JDK	:	Java Development Kit

ERD	:	Entity Relationship Diagram
JRE	:	Java Runtime Environment
JDBC	:	Java Databases Connectivity
ODBC	:	Open Databases Connectivity

Appendix-C

List Of Screens and Reports

66	5.1.24	File Upload to Cloud in DataProvider Page	80	Implementation
67	5.1.25	Upload files in data provider Page	81	Implementation

APPENDIX: D

Coding

#Dbconnection

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * And open the template in the editor.
 */
package attributebased;

import java.sql.Connection;
import java.sql.DriverManager;

/**
 *
 * @author java2
 */
public class Dbconnection {

    public static Connection getConnection() {
        Connection con = null;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection("jdbc:mysql://localhost:3306/deduplication", "root", "root");
        } catch (Exception ex) {
            ex.printStackTrace();
        }
        return con;
    }
}
```

#Decryption

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package attributebased;

/**
 *
 * @author java2
 */
import com.sun.org.apache.xerces.internal.impl.dv.util.Base64;
import java.io.ByteArrayOutputStream;
import java.io.FileInputStream;
import java.io.FileWriter;
import java.util.Scanner;

import javax.crypto.Cipher;
import javax.crypto.KeyGenerator;
import javax.crypto.SecretKey;
import javax.crypto.spec.SecretKeySpec;
import javax.swing.JOptionPane;
import sun.misc.BASE64Decoder;
import sun.misc.BASE64Encoder;

public class decryption {

   public String decrypt(String txt, String skey) {
      String decryptedtext = null;
      try {

         //converting string to secretkey
         byte[] bs = Base64.decode(skey);
```

```
            SecretKey sec = new SecretKeySpec(bs, "AES");
            System.out.println("converted string to seretkey:" + sec);

            System.out.println("secret key:" + sec);

             Cipher aesCipher = Cipher.getInstance("AES");//getting AES instance
                  aesCipher.init(Cipher.ENCRYPT_MODE,  sec);//initiating ciper encryption using secretkey

                                        byte[] byteCipherText = new BASE64Decoder().decodeBuffer(txt); //encrypting data

            //  System.out.println("ciper text:"+byteCipherText);
                              aesCipher.init(Cipher.DECRYPT_MODE,  sec, aesCipher.getParameters());//initiating ciper decryption

                                        byte[] byteDecryptedText = aesCipher.doFinal(byteCipherText);
            decryptedtext = new String(byteDecryptedText);

            System.out.println("Decrypted Text:" + decryptedtext);
        } catch (Exception e) {
            System.out.println(e);
        }
        return decryptedtext;
    }

}
```

#Download

```
/*
 * To change this license header, choose License Headers in Project
```

```
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package attributebased;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author java2
 */
public class download extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and
<code>POST</code>
     * methods.
     *
```

```
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException, SQLException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            /* TODO output your page here. You may use following
sample code. */
            String[] filedetails = request.getQueryString().split(",");
            String filename = null, skey = null;

            InputStream is = null;

            Connection con = Dbconnection.getConnection();
            Statement st = con.createStatement();
            ResultSet rt = st.executeQuery("select * from uploadcloud
where filename='" + filedetails[0] + "' AND  owner='" + filedetails[1]
+ "'");
            if (rt.next()) {
                filename = rt.getString("filename");
                skey = rt.getString("skey");
                is = (InputStream) rt.getAsciiStream("data");
            } else {
                out.println("error while retreiving data");
            }
            BufferedReader br = new BufferedReader(new
InputStreamReader(is));
            String temp = null;
            StringBuffer sb = new StringBuffer();
```

```
            while ((temp = br.readLine()) != null) {
                sb.append(temp + "\n");
            }
            String content = new decryption().decrypt(sb.toString(), skey);
            response.setHeader("Content-Disposition",
"attachment;filename=\"" + filename + "\"");
            out.write(content);
        }
    }

    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods.
Click on the + sign on the left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doGet(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
        try {
            processRequest(request, response);
        } catch (SQLException ex) {

Logger.getLogger(download.class.getName()).log(Level.SEVERE,
null, ex);
        }
    }
```

```
    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
        try {
            processRequest(request, response);
        } catch (SQLException ex) {

Logger.getLogger(download.class.getName()).log(Level.SEVERE,
null, ex);
        }
    }

    /**
     * Returns a short description of the servlet.
     *
     * @return a String containing servlet description
     */
    @Override
    public String getServletInfo() {
        return "Short description";
    }// </editor-fold>

}
```

#AA

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<!--
Design by TEMPLATED
http://templated.co
Released for free under the Creative Commons Attribution License

Name      : Big Business
Description: A two-column, fixed-width design with a bright color scheme.
Version   : 1.0
Released  : 20120210
-->
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="" />
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and
Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>

  <%
      if (request.getParameter("message") != null) {%>
    <script>alert('User Registered Successfully');</script>
    <%}
      if (request.getParameter("aalogin1") != null) {%>
    <script>alert('AA_Login_Failed ');</script>

    <%}
    %>
```

```
<div id="wrapper">
  <div id="header">
    <div id="logo">
      <h4><a href="#">Attribute-Based Encryption Approach for Storage,
Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
    </div>
    <div id="slogan">

    </div>
  </div>

    <center>
  <div id="menu">

    <ul>
      <li><a href="index.html">Home</a></li>
      <li><a href="dataproviderlogin.jsp">Data Provider</a></li>
      <li><a href="cloudlogin.jsp">Cloud</a></li>
                    <li><a href="userlogin.jsp">User</a></li>
      <li  class="selected"><a href="aa.jsp">AA</a></li>
      <li class="last"><a href="contact.html">Contact</a></li>
    </ul>
    <br class="clearfix" />
  </div>
  <div id="splash">
    <img class="pic" src="images/arcte.JPG" width="600" height="230"
alt="" />
  </div>
    <br><br>
          <h1><font color="black">AA Login</h1>
          <center> <form name="f" action="aaact.jsp" method="post"
onsubmit="return check()">
```

```
<table>

    <tr>
        <td>
          <center>      <strong><font size="4"
color="black">Username:</font></strong>
           <input type="text" name="username" id="userName1"
placeholder= Username style="height:30px; width:170px"></input>
        </td>
</tr></center>
<br>
    <tr>
        <td>
          <strong><font size="4" color="black">Password:
</font></strong>
            <input type="password" name="password"
id="password1" placeholder= Password style="height:30px;
width:170px"></input>
        </td>
    </tr>

     <tr></tr>
  <tr></tr>
  <tr></tr>
  <tr>
        <td>
           <input type="submit" value="Login"  style="height:30px;
width:65px" />
           <button type="button" class="cancelbtn" style="height:30px;
width:65px" >Cancel</button>
                
        </td>
              </tr>
```

```
        <tr></tr>
        <tr></tr><tr></tr><tr></tr><tr></tr>
   <tr>
   <td>
    </td>

<td>                   

   </td>
   </tr>
 </table>
    </form>
                <BR><BR><BR><BR>
</body>

</html>
```

#AAHome

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<!--
Design by TEMPLATED
http://templated.co
Released for free under the Creative Commons Attribution License

Name      : Big Business
Description: A two-column, fixed-width design with a bright color scheme.
Version   : 1.0
Released  : 20120210
-->
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
<meta name="description" content="" />
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>

  <%
      if (request.getParameter("message") != null) {%>
    <script>alert('User Registered Successfully');</script>
    <%}
      if (request.getParameter("aalogin1") != null) {%>
    <script>alert('AA_Login_Failed ');</script>

    <%}
    %>

<div id="wrapper">
 <div id="header">
  <div id="logo">
    <h4><a href="#">Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
  </div>
  <div id="slogan">

  </div>
 </div>

  <center>
 <div id="menu">
```

```
  <ul>
    <li><a href="index.html">Home</a></li>
    <li><a href="dataproviderlogin.jsp">Data Provider</a></li>
    <li><a href="cloudlogin.jsp">Cloud</a></li>
                <li><a href="userlogin.jsp">User</a></li>
    <li  class="selected"><a href="aa.jsp">AA</a></li>
    <li class="last"><a href="contact.html">Contact</a></li>
  </ul>
  <br class="clearfix" />
 </div>
 <div id="splash">
    <img class="pic" src="images/arcte.JPG" width="600" height="230"
alt="" />
 </div>
  <br><br>
       <h1><font color="black">AA Login</h1>
            <center> <form name="f" action="aaact.jsp" method="post"
onsubmit="return check()">
<table>

     <tr>
        <td>
                          <center>              <strong><font size="4"
color="black">Username:</font></strong>
                  <input type="text" name="username" id="userName1"
placeholder= Username style="height:30px; width:170px"></input>
        </td>
</tr></center>
<br>
     <tr>
        <td>
                    <strong><font size="4" color="black">Password:
```

```
</font></strong>
                     <input type="password" name="password" id="password1" placeholder= Password style="height:30px; width:170px"></input>
          </td>
      </tr>

       <tr></tr>
   <tr></tr>
   <tr></tr>
   <tr>
          <td>
                <input type="submit" value="Login" style="height:30px; width:65px" />
              <button type="button" class="cancelbtn" style="height:30px; width:65px" >Cancel</button>
                  
          </td>
                </tr>

          <tr></tr>
          <tr></tr><tr></tr><tr></tr><tr></tr>
   <tr>
   <td>
    </td>

<td>               

   </td>
   </tr>
 </table>
```

```
    </form>
                <BR><BR><BR><BR>
</body>

</html>
```

#Cloud Home

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="" />
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
    <%
        if (request.getParameter("login") != null) {%>
      <script>alert('Cloud Login_Successfully');</script>
      <%}
        if (request.getParameter("msgg") != null) {%>
      <script>alert('Login_Failed ');</script>

      <%}
      %>

<div id="wrapper">
   <div id="header">
      <div id="logo">
            <h4><a href="#">Attribute-Based Encryption Approach for
```

```
Storage, Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
      </div>
      <div id="slogan">

      </div>
   </div>
   <div id="menu">
      <ul>
         <li class="selected"><a href="cloudhome.jsp">Home</a></li>
         <li><a href="cloudviewfiles.jsp">Private Cloud</a></li>
         <li><a href="cloudviewfilesprivate.jsp">Public Cloud</a></li>
         <li><a href="cloudviewusers.jsp">View Users</a></li>
                        <li><a href="cloudviewdataproviders.jsp">View Data
Providers</a></li>
         <li><a href="index.html">Logout</a></li>

      </ul>
      <br class="clearfix" />
   </div>
   <div id="splash">
      <img class="pic" src="images/arcte.JPG" width="820" height="230"
alt="" />
   </div>

      <div id="splash">
            <center><h3>Welcome to Cloud</h3></center>

      </div>
      <br class="clearfix" />
   </div>

</body>
</html>
```

#DataProvider Login

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="" />
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>

  <%
      if (request.getParameter("message") != null) {%>
    <script>alert('Data Provider Registered Successfully');</script>
    <%}
      if (request.getParameter("email") != null) {%>
    <script>alert('Email Id you Entered already in Use ');</script>
     <%}
      if (request.getParameter("msgg") != null) {%>
    <script>alert('Data Provider Login Fail ');</script>
    <%}
    %>

<div id="wrapper">
 <div id="header">
  <div id="logo">
    <h4><a href="#">Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
  </div>
```

```
   <div id="slogan">

   </div>
  </div>

   <center>
  <div id="menu">

   <ul>
     <li><a href="index.html">Home</a></li>
              <li      class="selected"><a     href="dataproviderlogin.jsp">Data
Provider</a></li>
     <li><a href="cloudlogin.jsp">Cloud</a></li>
                  <li><a href="userlogin.jsp">User</a></li>
     <li><a href="aa.jsp">AA</a></li>
     <li class="last"><a href="contact.html">Contact</a></li>
   </ul>
   <br class="clearfix" />
  </div>



</html>
  <center>
  <div class="content">
   <div class="content_resize">
     <div class="mainbar">
       <div class="article">
          <br>
        <div class="clr"></div>
        <div id="splash">
     <img  class="pic"  src="images/arcte.JPG"  width="600"  height="230"
alt="" />
```

```
 </div><br><br>
      <!---Start Body --->
      <h1><font color="black">Data Provider Login</h1>
     <center><table>
  <form action="dataproviderloginact.jsp" method="get"> <br><br>
     <tr>
         <td>
                                                  <strong><font size="3"
color="black">Username:</font></strong>
                    <input type="text" name="username" id="userName1"
placeholder= Username style="height:30px; width:200px"></input>
         </td>
     </tr>

     <tr>
         <td>
                     <strong><font size="3" color="black">Password:
</font></strong>
                     <input type="password" name="password"
id="password1" placeholder= Password style="height:30px;
width:200px"></input>
         </td>
     </tr>

      <tr></tr>
   <tr></tr>
   <tr></tr>
   <tr>
         <td>
                                       
       <input type="submit"
value="Login"  style="height:30px; width:65px" />
               <button type="button" class="cancelbtn" style="height:30px;
```

```
width:65px" >Cancel</button>
                        <a  href="dataproviderreg.jsp"><font  color="#ff00cc"
size="4"><u>New User?SignUp</u></font></a>


          </td>
        </tr>

        <tr>
          <td>

           &
nbsp;         &n
bsp; 
          </td>
        </tr>


  </table>
            </center>
       <!---End Body --->
     <br><br><br><br></div>
    </div>

     <div class="clr"></div>
   </div>
  </div>


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

#User Login

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="" />
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>

  <%
      if (request.getParameter("userreg") != null) {%>
    <script>alert('User Registered Successfully');</script>
    <%}
      if (request.getParameter("userlogin1") != null) {%>
    <script>alert('User Login_Failed ');</script>
    <%}
      if (request.getParameter("userlogin2") != null) {%>
    <script>alert('User not Activated ');</script>

    <%}
    %>

<div id="wrapper">
 <div id="header">
  <div id="logo">
    <h4><a href="#">Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
  </div>
```

```
  <div id="slogan">

  </div>
 </div>

  <center>
 <div id="menu">

  <ul>
    <li><a href="index.html">Home</a></li>
    <li><a href="dataproviderlogin.jsp">Data Provider</a></li>
    <li><a href="cloudlogin.jsp">Cloud</a></li>
                <li class="selected"><a href="userlogin.jsp">User</a></li>
    <li><a href="aa.jsp">AA</a></li>
    <li class="last"><a href="contact.html">Contact</a></li>
  </ul>
  <br class="clearfix" />
 </div>
 </html>
 <center>
 <div class="content">
  <div class="content_resize">
   <div class="mainbar">
    <div class="article">
      <br>
     <div class="clr"></div>
     <!---Start Body --->
     <div id="splash">
   <img class="pic" src="images/arcte.JPG" width="600" height="230"
alt="" />
 </div><br><br>
          <h1><font color="black">User Login</h1><br>
    <center><table>
```

```
<form action="userloginact.jsp" method="get">
    <tr>
        <td>
                                                <strong><font size="3" color="black">Username:</font></strong>
                    <input type="text" name="username" id="userName1" placeholder= Username style="height:30px; width:200px"></input>
        </td>
    </tr>

    <tr>
        <td>
                    <strong><font size="3" color="black">Password: </font></strong>
                     <input type="password" name="password" id="password1" placeholder= Password style="height:30px; width:200px"></input>
        </td>
    </tr>

     <tr></tr>
  <tr></tr>
  <tr></tr>
  <tr>
        <td>

             <input type="submit" value="Login" style="height:30px; width:65px" />
              <button type="button" class="cancelbtn" style="height:30px; width:65px" >Cancel</button>
          <a href="userreg.jsp"><font color="#ff00cc" size="4"><u>New User?SignUp</u></font></a>
```

```
            </td>
          </tr>

          <tr>
             <td>
              </td>
          </tr>
         </table>
               </center>
         <!---End Body --->
      <br><br><br><br></div>
     </div>

      <div class="clr"></div>
    </div>
  </div></div>
</body>
</html>
```

#Key Generation

```
 <%@page import="attributebased.encryption"%>
<%@page
import="com.sun.org.apache.xerces.internal.impl.dv.util.Base64"%>
<%@page import="javax.crypto.SecretKey"%>
<%@page import="javax.crypto.KeyGenerator"%>
<%@page import="java.util.Random"%>
<%@page import="attributebased.Mail"%>
<%@page import="attributebased.decryption"%>
<%@page import="java.io.InputStreamReader"%>
<%@page import="java.io.BufferedReader"%>
<%@page import="java.io.InputStream"%>
<%@page import="java.sql.ResultSet"%>
```

```
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.Connection"%>
<%@page import="attributebased.Dbconnection"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%

   KeyGenerator keyGen = KeyGenerator.getInstance("AES");
                        keyGen.init(128);
                       SecretKey secretKey = keyGen.generateKey();
                        System.out.println("secret key:" + secretKey);
                      // converting secretkey to String
                                byte[] be = secretKey.getEncoded();//encoding
secretkey
                       String skey = Base64.encode(be);
                           System.out.println("converted secretkey to string:" +
skey);
                                // String cipher = new encryption().encrypt(str,
secretKey)

   String filename=request.getParameter("filename");
    String owner=request.getParameter("owner");
     String data=request.getParameter("data");

     Random r= new Random();
     int i=r.nextInt(10000 - 5000) + 5000;
     String publickey = i+"";

     Random r1= new Random();
     int i1=r1.nextInt(10000 - 5000) + 5000;
     String privatekey = i1+"";

     try{
         Connection con = Dbconnection.getConnection();
```

```
        Statement st = con.createStatement();
            int j = st.executeUpdate("insert into encryptkey(filename,owner,data,dkey,privatekey)values('"+filename+"','"+owner+"','"+data+"','"+publickey+"','"+skey+"')");

     if (j !=0){

        response.sendRedirect("dataproviderfiles.jsp?key=success");
     }
     else{
        response.sendRedirect("dataproviderfiles.jsp?key1=Failed");
     }

} catch (Exception ex) {
   response.sendRedirect("dataproviderfiles.jsp?key2=Failed");
                              ex.printStackTrace();
                           }

                        %>
```

#Attribute Verify

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
  <%@page import="attributebased.decryption"%>
<%@page import="java.io.InputStreamReader"%>
<%@page import="java.io.BufferedReader"%>
<%@page import="java.io.InputStream"%>
<%@page import="java.sql.ResultSet"%>
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.Connection"%>
<%@page import="attributebased.Dbconnection"%>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="" />
```

```
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>

  <%
        if (request.getParameter("m1") != null) {%>
      <script>alert('Login Successfully');</script>
      <%}
        if (request.getParameter("dmsg") != null) {%>
      <script>alert('Vefication Failed   Attributes not Matched');</script>
       <%}
        if (request.getParameter("attributes") != null) {%>
      <script>alert('Vefication Failed   Attributes not Matched');</script>
       <%}
        if (request.getParameter("attributes1") != null) {%>
      <script>alert('Vefication Failed   Attributes not Matched');</script>

      <%}
      %>
<div id="wrapper">
  <div id="header">
     <div id="logo">Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
    </div>
    <div id="slogan">

    </div>
  </div>
```

```
<center>
<div id="menu">

<ul>
<li><a href="aahome.jsp">Home</a></li>
<li><a href="viewdataproviders.jsp">View Data Providers</a></li>
<li><a href="viewusers.jsp">View Users</a></li>
<li class="selected"><a href="aaverify.jsp">User Attribute Verify</a></li>
<li><a href="cloudviewfilesprivate1.jsp">Public Cloud</a></li>
<li><a href="index.html">Logout</a></li>

</ul>
<br class="clearfix" />
</div>
<div id="splash">
<img class="pic" src="images/arcte.JPG" width="600" height="230" alt="" />
</div>
<div id="body">

<div id="main">
<div id="right">
<h4></h4>

<!--==========body start==========-->
<br>
<center><h3>Attribute Verification</h3>

<%
```

```
        String filename = request.getQueryString();

   try{

 Connection con=Dbconnection.getConnection();
 Statement st=con.createStatement();
 ResultSet rs=st.executeQuery("select * from request where filename='"+filename+"' and status='Policy Vefied'");



%>
    <center> <table style="width:60%" border="2">
        <br>

��� <tr STYLE="background-color: yellowgreen;font-size: 15px;">

 ���  <td>File Name</td>
      <td>Owner</td>
     <td>Policy</td>
     <td>Time</td>
     <td>Experience</td>
     <td>Branch</td>
     <td>User</td>
     <td>Status</td>
     <td>View</td>

 ��� </tr>
<%

  if(rs.next()){

        %><tr>
```

```
<td><%=rs.getString(1)%></td>
<td><%=rs.getString(3)%></td>
<td><%=rs.getString(4)%></td>
<td><%=rs.getString(5)%></td>
<td><%=rs.getString(6)%></td>
<td><%=rs.getString(7)%></td>
  <td><%=rs.getString(8)%></td>
<td><%=rs.getString(9)%></td>
                                                                    <td><a
href="attributeverify1.jsp?filename=<%=rs.getString("filename")%>&own
er=<%=rs.getString("owner")%>&umail=<%=rs.getString("umail")%>">V
erify Attributes</a> </td>
      </tr>

       <%  }
%></table></center>

<%}
 catch(Exception e)
 {
  System.out.println(e);
 }
 %>
                  </table>
<br>
</div>

</div>
<!--content ends -->
<!--footer begins -->
<!-- footer ends-->
</body>
</html>
```

A

PROJECT REPORT ON

ATTRIBUTE BASED ENCRYPTION APPROACH FOR STORING, SHARING AND RETRIEVAL OF ENCRYPTED DATA IN THE CLOUD

Submitted in partial fulfillment of the
requirements for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

By

Ms V. SOWJANYA,
(Regd.No:215N1F00A1).

Under the Guidance of

Mr A. UMA MAHESWAR REDDY,
Assistant Professor, APGCCS.

&

Mr G. VENKAT RAO,
Project Leader,
Manosys Technologies Pvt Ltd,Bangalore.

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

ANNAMACHARYA P.G COLLEGE OF COMPUTER STUDIES

NEW BOYANAPALLI-516126, RAJAMPET (A.P)

(Approved by A.I.C.T.E., New Delhi & Affiliated to J.N.T.U.A,
Anantapuramu, UGC(2f) Recognized Institution)

(2021-2023)

***

ANNAMACHARYA P.G COLLEGE OF COMPUTER STUDIES

NEW BOYANAPALLI, RAJAMPET, A.P, 516126.

Affiliated to

JAWAHARLAL NEHRU TECHNOLOGICALUNIVERSITY ANANTAPUR, ANANTAPURAMU

DEPARTMENT OF

MASTER OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that the project work entitled **"Attribute Based Encryption Approach For Storing, Sharing And Retrieval Of Encrypted Data In The Cloud"** is the Bonafide work carried out by **Ms V. Sowjanya,** Regd. No:**215N1F00A1,**is submitted in the partial fulfillment of the requirements for the award of degree of **Master of Computer Applications** during the year 2021-2023.

Project Guide **Principal**

External Examiner

DECLARATION

I,**V. Sowjanya,** hereby declare that the project report entitled as **"Attribute Based Encryption Approach For Storing, Sharing And Retrieval Of Encrypted Data In The Cloud"** is done in **Manosys Technologies Pvt Ltd, Bangalore**, is original and independent record of work, submitted by me to JNTUA, Anantapuramu, under the guidance of **Mr A. UMA MAHESWAR REDDY,** Assistant Professor of **Annamacharya P.G College of Computer Studies,** Rajampet, for the award of the degree of **Master of Computer Applications** and has not been submitted either in partial or full for the award of any Degree or Diploma.

Place: Rajampet **V. SOWJANYA,**

Date: **(Regd.No:215N1F00A1).**

ACKNOWLEDGEMENT

An endeavor over a long period can be successful only with the advice of many well-wishers.I take this opportunity to express my deep gratitude and appreciation of all those who encourage me to successfully complete the project.

I wish to express my sincere gratitude to **Dr D.J. SAMATHANAIDU**, Principal of **Annamacharya P.G College of Computer Studies**, New Boyanapalli, Rajampet, for her consistent help and providing such facilities to complete.

I express my sincere thanks to my guide **Mr A. UMA MAHESWAR REDDY,** Assistant Professor in MCA Deportment for his valuable guidance and suggestions in analyzing and testing throughout the period of my project work.

I express my sincere thanks to my guide **Mr G. VENKAT RAO**, Project Leader, Manosys Technologies Pvt Ltd, Bangalore for his valuable guidance and suggestions in analyzing and testing throughout the period of my project work.

Last but not least, I would like to thank my friends, teaching and nonteaching, one and all those who helped me to complete this project successfully.

V. SOWJANYA,
(Regd.No:215N1F00A1).

CONTENTS

BIBLIOGRAPHY

Appendix-A

- URL Listing
- References

Appendix – B

- Glossary

Appendix – C

- List of Figures
- List of Tables
- List of Screens and Reports

Appendix – D

- Coding
- Plagiarism report

3. SYSTEM DESIGN

System Design is the process or art of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. One could see it as the application of systems theory to product development. There is some overlap and synergy between the disciplines of systems analysis, systems architecture, and systems engineering.

3.1 Data Design

Database design is part of the development process. In the linear development cycle, it is used during the system requirements phase to construct the data components of the analysis model. This model represents the major data objects and the relationships between them. It should not be confused with data analysis, which takes place in the system design phase. As in a DFD, a model of data consists of a number of symbols joined up according to certain conventions. System designers describe this conceptual modeling using symbols from a modeling method known as entity relationship analysis.

Entity Relationship Diagram

Entity relationship analysis uses three major abstractions to describe data. These are

- Entities, which are distinct things in the enterprise.
- Relationships, which are meaningful interactions between objects,
- Attributes, which are the properties of the entities and relationships.
- The relative simplicity and pictorial clarity of this diagramming technique may well account in large part for the widespread use of the ER model. Such a diagram consists of the following major components:

3.3 UML Design

To understand the UML, you need to form a conceptual model of the language, and this requires learning three major elements: the UML's basic building blocks, the rules that dictate how these building blocks may be put together, and some common mechanisms that apply throughout the UML. Once you have grasped these ideas, you will be able to read UML models and create some basic ones. As you gain more experience in applying UML, you can build on this conceptual model by using more advanced features of the language.

Building Blocks of the UML

The vocabulary of the UML encompasses three kinds of building blocks.

1. Things
2. Relationships
3. Diagrams

Things are abstractions that are first-class citizens in a model; relationships tie these things together, and diagrams group interesting collections of things.

Things in the UML

There are four kinds of things in the UML.

1. Structural things
2. Behavioral things
3. Grouping things
4. Annotational things

Structural things

Structural things are the nouns of UML models. These are the mostly static parts of a model, representing elements that are either conceptual or physical. In all, there are seven kinds of structural things.

4. TESTING

In general, software engineers distinguish software faults from software failures. In the event of a failure, the software does not do what the user expects. A fault is a programming error that may or may not actually manifest as a failure. A fault can also be described as an error in the correctness of the semantics of a computer program. A fault will become a failure if the exact computation conditions are met, one of which is that the faulty portion of computer software executes on the CPU. A fault can also turn into a failure when the software is ported to a different hardware platform, compiled with a different compiler, or extended.

Software testing may be viewed as a sub-field of Software Quality Assurance but typically exists independently (and there may be no SQA areas in some companies). In SQA, software process specialists and auditors take a broader view on software and its development. They examine and change the software engineering process itself to reduce the number of faults that end up in the code or deliver it faster.

Regardless of the methods used or level of formality involved the desired result of testing is a level of confidence in the software so that the organization is confident that the software has an acceptable defect rate. What constitutes an acceptable defect rate depends on the nature of the software. An arcade video game designed to simulate flying an airplane would presumably have a much higher tolerance for defects than the software used to control an actual airliner.

A problem with software testing is that the number of defects in a software product can be very large, and the number of configurations of the product can be even larger. Bugs that occur infrequently are difficult to find in testing. A rule of thumb is that a system that is expected to function without faults for a certain length of time must have already been tested for at least that length of time. This has severe consequences for projects to write long-

CONTENTS

BIBLIOGRAPHY

Appendix – A

- URL Listings
- References

Appendix – B

- Glossary

Appendix – C

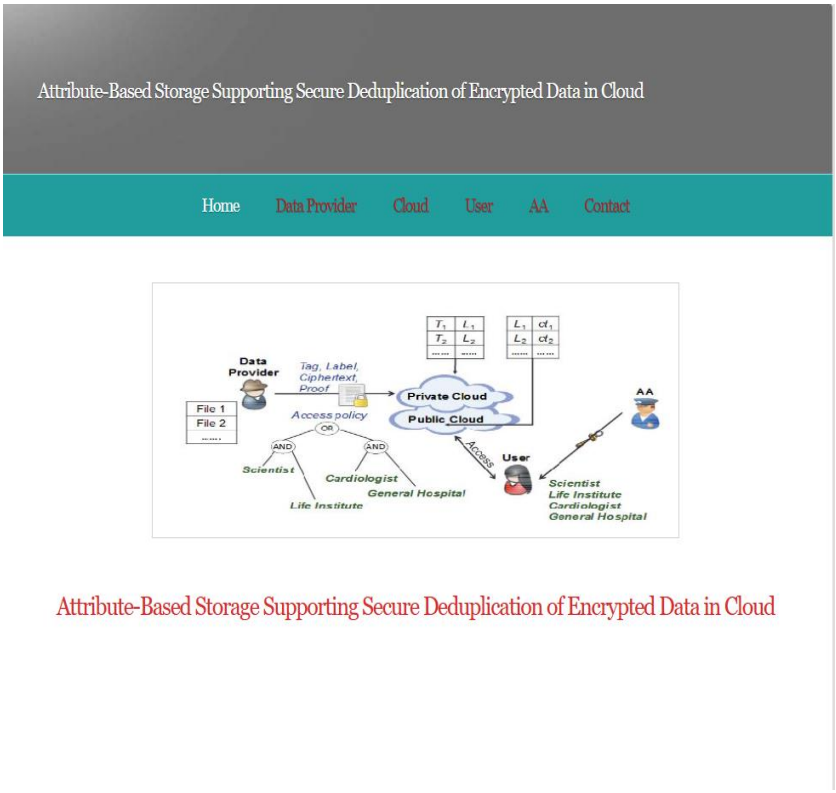
- List of Tables
- List of Figures
- List of Screens and Reports

Appendix – D

- Coding
- Plagiarism report

5.1 Sample Screens

Home page



Screen 5.1.1 Home page

Description: This Screen shows Home page

Appendix - A

URL Listing

Websites	Reference Books
www.elsevier.com	Some of the advanced information about Java such as its driver types and the connectivity information about JDK.
www.training-classes.com	The Designing part information hasbeen gathered.
http://www.wikipedia.org	Searching for any information that canbe used in documentation.
http://www.google.co.in	Details about the code have been collected have been collected.
http://www.google.co.in	Any information searching and downloading.

References

[1] D. Quick, B. Martini, and K. R. Choo, Cloud Storage Forensics. Syngress Publishing / Elsevier, 2014. [Online].Available: http://www.elsevier.com/books/cloud-storageforensics/quick/978-0-12-419970-5

[2] K. R. Choo, J. Domingo-Ferrier, and L. Zhang, “Cloud cryptography: Theory, practice and future research directions,” Future Generation Comp. Syst., vol. 62, pp. 51–53, 2016.

[3] K. R. Choo, M. Herman, M. Organ, and B. Martini, “Cloud forensics: State-of-the-art and future directions,” Digital Investigation, vol. 18, pp. 77–78, 2016.

[4] Y. Yang, H. Zhu, H. Lu, J.Weng, Y. Zhang, and K. R. Choo, “Cloud based data sharing with fine-grained proxy re-encryption,” Pervasive and Mobile Computing, vol. 28, pp. 122–134, 2016.

[5] D. Quick and K. R. Choo, “Google drive: Forensic analysis of data remnants,” J. Network and Computer Applications, vol. 40, pp. 179–193, 2014.

[6] A. Sahai and B. Waters, “Fuzzy identity-based encryption,” in Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings, ser. Lecture Notes in Computer Science, vol. 3494. Springer, 2005, pp. 457–473.

[7] B. Zhu, K. Li, and R. H. Patterson, “Avoiding the disk bottlenecking the data domain deduplication file system,” in 6th USENIXConference on File and Storage Technologies, FAST 2008, February 26-29, 2008, San Jose, CA, USA. USENIX, 2008, pp. 269–282.

[8] M. Bellaire, S. Keelveedhi, and T. Ristenpart, “Message-locked encryption and secure deduplication,” in Advances in Cryptology-EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings, ser. Lecture Notes in Computer Science, vol. 7881. Springer, 2013, pp. 296–312.

[9] M. Abaci, D. Bone, I. Maroon, A. Raghunathan, and G. Segev,“Message-locked encryption for lock-dependent messages,” in Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I, ser. Lecture Notes in Computer Science, vol. 8042. Springer, 2013, pp. 374–391.

[10] S. Keelveedhi, M. Bellaire, and T. Ristenpart, "Dupless: Serveraidedencryption for reduplicated storage," in Proceedings of the22th USENIX Security Symposium, Washington, DC, USA, August14-16, 2013. USENIX Association, 2013, pp. 179–194.

[11] M. Bellaire and S. Keelveedhi, "Interactive message-locked encryption and secure deduplication," in Public-Key Cryptography – PKC2015 - 18th IACR International Conference on Practice and Theory in Public-Key Cryptography, Gaithersburg, MD, USA, March 30 – April1, 2015, Proceedings, ser. Lecture Notes in Computer Science, vol.9020. Springer, 2015, pp. 516–538.

[12] S. Bugiel, S. N¨ urnberger, A. Sadeghi, and T. Schneider, "Twin clouds: Secure cloud computing with low latency - (full version),"in Communications and Multimedia Security, 12th IFIP TC 6 / TC11 International Conference, CMS 2011, Ghent, Belgium, October 19-21,2011. Proceedings, ser. Lecture Notes in Computer Science, vol.7025. Springer, 2011, pp. 32–44.

[13] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof-systems (extended abstract)," in Proceedings of the 17th Annual ACM Symposium on Theory of Computing, May 6-8, 1985, Providence, Rhode Island, USA. ACM, 1985, pp. 291–304.

[14] M. Fechlin and R. Fechlin, "Efficient non-malleable commitment schemes," in Advances in Cryptology - CRYPTO 2000, 20th Annual International Cryptology Conference, Santa Barbara, California, USA,August 20-24, 2000, Proceedings, ser. Lecture Notes in Computer Science, vol. 1880. Springer, 2000, pp. 413–431.

[15] S. Goldwasser and S. Micali, "Probabilistic encryption," J. Comput.Syst. Sci., vol. 28, no. 2, pp. 270–299, 1984.

[16] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proceedings of the 13th ACM Conference on Computer and Communications November 3, 2006, ser. Lecture Notes in Computer Science, vol.5126. Springer, 2006, pp. 89–98.

[17] R. Ostrovsky, A. Sahai, and B.Waters, "Attribute-based encryption with no-monotonic access structures," in Proceedings of the 2007ACM Conference on Computer and Communications Security, CCS2007, Alexandria, Virginia, USA, October 28-31, 2007. ACM, 2007, pp. 195–203.

[18] A. B. Lewko and B. Waters, "Unbounded HIBE and attributebasedencryption," in Advances in Cryptology - EUROCRYPT 2011 -30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings, ser. Lecture Notes in Computer Science, vol. 6632. Springer, 2011, pp. 547–567.

[19] J. Bettencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in 2007 IEEE Symposium on Security And Privacy (S&P 2007), 20-23 May 2007, Oakland, California, USA.IEEE Computer Society, 2007, pp. 321–334.

[20] L. Cheung and C. C. Newport, "Provably secure ciphertext policy ABE," in Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA,October 28-31, 2007. ACM, 2007, pp. 456–465.

[21] V. Goyal, A. Jain, O. Pandey, and A. Sahai, "Bounded ciphertext policy attribute based encryption," in Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik

Appendix - B

Glossary

TA	:	Trusted Authority
ESP	:	Encryption Service Provider
DSP	:	Decryption Service Provider
DBMS	:	Database Management System
GUI	:	Graphic User Interface
HTML	:	Hyper Text Markup Language
HTTP	:	Hyper Text Transfer Protocol
JSP	:	Java Server Pages
J2SE	:	Java2 Standard Edition
SQL	:	Structured Query Language
TCP	:	Transmission Control Protocol
UML	:	Unified Modeling Language
URL	:	Uniform Resource Locator
WWW	:	World Wide Web
JS	:	Java Script
SRS	:	Software Requirement specification
JDK	:	Java Development Kit

ERD	:	Entity Relationship Diagram
JRE	:	Java Runtime Environment
JDBC	:	Java Databases Connectivity
ODBC	:	Open Databases Connectivity

Appendix-C

List Of Screens and Reports

66	5.1.24	File Upload to Cloud in DataProvider Page	80	Implementation
67	5.1.25	Upload files in data provider Page	81	Implementation

APPENDIX: D

Coding

#Dbconnection

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * And open the template in the editor.
 */
package attributebased;

import java.sql.Connection;
import java.sql.DriverManager;

/**
 *
 * @author java2
 */
public class Dbconnection {

    public static Connection getConnection() {
        Connection con = null;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection("jdbc:mysql://localhost:3306/deduplication", "root", "root");
        } catch (Exception ex) {
            ex.printStackTrace();
        }
        return con;
    }
}
```

#Decryption

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package attributebased;

/**
 *
 * @author java2
 */
import com.sun.org.apache.xerces.internal.impl.dv.util.Base64;
import java.io.ByteArrayOutputStream;
import java.io.FileInputStream;
import java.io.FileWriter;
import java.util.Scanner;

import javax.crypto.Cipher;
import javax.crypto.KeyGenerator;
import javax.crypto.SecretKey;
import javax.crypto.spec.SecretKeySpec;
import javax.swing.JOptionPane;
import sun.misc.BASE64Decoder;
import sun.misc.BASE64Encoder;

public class decryption {

   public String decrypt(String txt, String skey) {
      String decryptedtext = null;
      try {

         //converting string to secretkey
         byte[] bs = Base64.decode(skey);
```

```
            SecretKey sec = new SecretKeySpec(bs, "AES");
            System.out.println("converted string to seretkey:" + sec);

            System.out.println("secret key:" + sec);

             Cipher aesCipher = Cipher.getInstance("AES");//getting AES instance
                  aesCipher.init(Cipher.ENCRYPT_MODE,  sec);//initiating ciper encryption using secretkey

                                        byte[] byteCipherText = new BASE64Decoder().decodeBuffer(txt); //encrypting data

            //  System.out.println("ciper text:"+byteCipherText);
                              aesCipher.init(Cipher.DECRYPT_MODE,  sec, aesCipher.getParameters());//initiating ciper decryption

                                        byte[] byteDecryptedText = aesCipher.doFinal(byteCipherText);
            decryptedtext = new String(byteDecryptedText);

            System.out.println("Decrypted Text:" + decryptedtext);
        } catch (Exception e) {
            System.out.println(e);
        }
        return decryptedtext;
    }

}
```

#Download

```
/*
 * To change this license header, choose License Headers in Project
```

```
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package attributebased;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author java2
 */
public class download extends HttpServlet {

    /**
     * Processes requests for both HTTP <code>GET</code> and
<code>POST</code>
     * methods.
     *
```

```
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException, SQLException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            /* TODO output your page here. You may use following
sample code. */
            String[] filedetails = request.getQueryString().split(",");
            String filename = null, skey = null;

            InputStream is = null;

            Connection con = Dbconnection.getConnection();
            Statement st = con.createStatement();
            ResultSet rt = st.executeQuery("select * from uploadcloud
where filename='" + filedetails[0] + "' AND  owner='" + filedetails[1]
+ "'");
            if (rt.next()) {
                filename = rt.getString("filename");
                skey = rt.getString("skey");
                is = (InputStream) rt.getAsciiStream("data");
            } else {
                out.println("error while retreiving data");
            }
            BufferedReader br = new BufferedReader(new
InputStreamReader(is));
            String temp = null;
            StringBuffer sb = new StringBuffer();
```

```
            while ((temp = br.readLine()) != null) {
                sb.append(temp + "\n");
            }
            String content = new decryption().decrypt(sb.toString(), skey);
            response.setHeader("Content-Disposition",
"attachment;filename=\"" + filename + "\"");
            out.write(content);
        }
    }

    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods.
Click on the + sign on the left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doGet(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
        try {
            processRequest(request, response);
        } catch (SQLException ex) {

Logger.getLogger(download.class.getName()).log(Level.SEVERE,
null, ex);
        }
    }
```

```
    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
        try {
            processRequest(request, response);
        } catch (SQLException ex) {

Logger.getLogger(download.class.getName()).log(Level.SEVERE,
null, ex);
        }
    }

    /**
     * Returns a short description of the servlet.
     *
     * @return a String containing servlet description
     */
    @Override
    public String getServletInfo() {
        return "Short description";
    }// </editor-fold>

}
```

#AA

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<!--
Design by TEMPLATED
http://templated.co
Released for free under the Creative Commons Attribution License

Name      : Big Business
Description: A two-column, fixed-width design with a bright color scheme.
Version   : 1.0
Released  : 20120210
-->
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="" />
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and
Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>

  <%
      if (request.getParameter("message") != null) {%>
    <script>alert('User Registered Successfully');</script>
    <%}
      if (request.getParameter("aalogin1") != null) {%>
    <script>alert('AA_Login_Failed ');</script>

    <%}
    %>
```

```
<div id="wrapper">
  <div id="header">
    <div id="logo">
      <h4><a href="#">Attribute-Based Encryption Approach for Storage,
Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
    </div>
    <div id="slogan">

    </div>
  </div>

    <center>
  <div id="menu">

    <ul>
      <li><a href="index.html">Home</a></li>
      <li><a href="dataproviderlogin.jsp">Data Provider</a></li>
      <li><a href="cloudlogin.jsp">Cloud</a></li>
                    <li><a href="userlogin.jsp">User</a></li>
      <li  class="selected"><a href="aa.jsp">AA</a></li>
      <li class="last"><a href="contact.html">Contact</a></li>
    </ul>
    <br class="clearfix" />
  </div>
  <div id="splash">
    <img class="pic" src="images/arcte.JPG" width="600" height="230"
alt="" />
  </div>
    <br><br>
          <h1><font color="black">AA Login</h1>
          <center> <form name="f" action="aaact.jsp" method="post"
onsubmit="return check()">
```

```
<table>

    <tr>
        <td>
          <center>      <strong><font size="4"
color="black">Username:</font></strong>
           <input type="text" name="username" id="userName1"
placeholder= Username style="height:30px; width:170px"></input>
        </td>
</tr></center>
<br>
    <tr>
        <td>
          <strong><font size="4" color="black">Password:
</font></strong>
            <input type="password" name="password"
id="password1" placeholder= Password style="height:30px;
width:170px"></input>
        </td>
    </tr>

     <tr></tr>
  <tr></tr>
  <tr></tr>
  <tr>
        <td>
           <input type="submit" value="Login"  style="height:30px;
width:65px" />
           <button type="button" class="cancelbtn" style="height:30px;
width:65px" >Cancel</button>
                
        </td>
              </tr>
```

```
        <tr></tr>
        <tr></tr><tr></tr><tr></tr><tr></tr>
   <tr>
   <td>
    </td>

<td>                   

   </td>
   </tr>
 </table>
    </form>
                <BR><BR><BR><BR>
</body>

</html>
```

#AAHome

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<!--
Design by TEMPLATED
http://templated.co
Released for free under the Creative Commons Attribution License

Name      : Big Business
Description: A two-column, fixed-width design with a bright color scheme.
Version   : 1.0
Released  : 20120210
-->
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head>
<meta name="description" content="" />
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>

  <%
      if (request.getParameter("message") != null) {%>
    <script>alert('User Registered Successfully');</script>
    <%}
      if (request.getParameter("aalogin1") != null) {%>
    <script>alert('AA_Login_Failed ');</script>

    <%}
    %>

<div id="wrapper">
 <div id="header">
  <div id="logo">
    <h4><a href="#">Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
  </div>
  <div id="slogan">

  </div>
 </div>

  <center>
 <div id="menu">
```

```
  <ul>
    <li><a href="index.html">Home</a></li>
    <li><a href="dataproviderlogin.jsp">Data Provider</a></li>
    <li><a href="cloudlogin.jsp">Cloud</a></li>
                <li><a href="userlogin.jsp">User</a></li>
    <li  class="selected"><a href="aa.jsp">AA</a></li>
    <li class="last"><a href="contact.html">Contact</a></li>
  </ul>
  <br class="clearfix" />
 </div>
 <div id="splash">
    <img class="pic" src="images/arcte.JPG" width="600" height="230"
alt="" />
 </div>
  <br><br>
       <h1><font color="black">AA Login</h1>
            <center> <form name="f" action="aaact.jsp" method="post"
onsubmit="return check()">
<table>

     <tr>
        <td>
                          <center>              <strong><font size="4"
color="black">Username:</font></strong>
                  <input type="text" name="username" id="userName1"
placeholder= Username style="height:30px; width:170px"></input>
        </td>
</tr></center>
<br>
     <tr>
        <td>
                    <strong><font size="4" color="black">Password:
```

```
</font></strong>
                     <input type="password" name="password" id="password1" placeholder= Password style="height:30px; width:170px"></input>
          </td>
      </tr>

       <tr></tr>
   <tr></tr>
   <tr></tr>
   <tr>
          <td>
                <input type="submit" value="Login" style="height:30px; width:65px" />
              <button type="button" class="cancelbtn" style="height:30px; width:65px" >Cancel</button>
                  
          </td>
                </tr>

          <tr></tr>
          <tr></tr><tr></tr><tr></tr><tr></tr>
   <tr>
   <td>
    </td>

<td>               

   </td>
   </tr>
 </table>
```

```
    </form>
                <BR><BR><BR><BR>
</body>

</html>
```

#Cloud Home

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="" />
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
    <%
        if (request.getParameter("login") != null) {%>
      <script>alert('Cloud Login_Successfully');</script>
      <%}
        if (request.getParameter("msgg") != null) {%>
      <script>alert('Login_Failed ');</script>

      <%}
      %>

<div id="wrapper">
   <div id="header">
      <div id="logo">
            <h4><a href="#">Attribute-Based Encryption Approach for
```

```
Storage, Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
      </div>
      <div id="slogan">

      </div>
   </div>
   <div id="menu">
      <ul>
         <li class="selected"><a href="cloudhome.jsp">Home</a></li>
         <li><a href="cloudviewfiles.jsp">Private Cloud</a></li>
         <li><a href="cloudviewfilesprivate.jsp">Public Cloud</a></li>
         <li><a href="cloudviewusers.jsp">View Users</a></li>
                        <li><a href="cloudviewdataproviders.jsp">View Data
Providers</a></li>
         <li><a href="index.html">Logout</a></li>

      </ul>
      <br class="clearfix" />
   </div>
   <div id="splash">
      <img class="pic" src="images/arcte.JPG" width="820" height="230"
alt="" />
   </div>

      <div id="splash">
            <center><h3>Welcome to Cloud</h3></center>

      </div>
      <br class="clearfix" />
   </div>

</body>
</html>
```

#DataProvider Login

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="" />
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>

  <%
      if (request.getParameter("message") != null) {%>
    <script>alert('Data Provider Registered Successfully');</script>
    <%}
      if (request.getParameter("email") != null) {%>
    <script>alert('Email Id you Entered already in Use ');</script>
     <%}
      if (request.getParameter("msgg") != null) {%>
    <script>alert('Data Provider Login Fail ');</script>
    <%}
    %>

<div id="wrapper">
 <div id="header">
  <div id="logo">
    <h4><a href="#">Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
  </div>
```

```
   <div id="slogan">

   </div>
  </div>

   <center>
  <div id="menu">

   <ul>
     <li><a href="index.html">Home</a></li>
              <li      class="selected"><a     href="dataproviderlogin.jsp">Data
Provider</a></li>
     <li><a href="cloudlogin.jsp">Cloud</a></li>
                  <li><a href="userlogin.jsp">User</a></li>
     <li><a href="aa.jsp">AA</a></li>
     <li class="last"><a href="contact.html">Contact</a></li>
   </ul>
   <br class="clearfix" />
  </div>



</html>
  <center>
  <div class="content">
   <div class="content_resize">
     <div class="mainbar">
       <div class="article">
          <br>
        <div class="clr"></div>
        <div id="splash">
     <img  class="pic"  src="images/arcte.JPG"  width="600"  height="230"
alt="" />
```

```
 </div><br><br>
      <!---Start Body --->
      <h1><font color="black">Data Provider Login</h1>
     <center><table>
  <form action="dataproviderloginact.jsp" method="get"> <br><br>
     <tr>
         <td>
                                                  <strong><font size="3"
color="black">Username:</font></strong>
                    <input type="text" name="username" id="userName1"
placeholder= Username style="height:30px; width:200px"></input>
         </td>
     </tr>

     <tr>
         <td>
                     <strong><font size="3" color="black">Password:
</font></strong>
                     <input type="password" name="password"
id="password1" placeholder= Password style="height:30px;
width:200px"></input>
         </td>
     </tr>

      <tr></tr>
   <tr></tr>
   <tr></tr>
   <tr>
         <td>
                                       
       <input type="submit"
value="Login"  style="height:30px; width:65px" />
               <button type="button" class="cancelbtn" style="height:30px;
```

```
width:65px" >Cancel</button>
                        <a  href="dataproviderreg.jsp"><font  color="#ff00cc"
size="4"><u>New User?SignUp</u></font></a>


          </td>
        </tr>

        <tr>
          <td>

           &
nbsp;         &n
bsp; 
          </td>
        </tr>


  </table>
            </center>
       <!---End Body --->
     <br><br><br><br></div>
    </div>

     <div class="clr"></div>
   </div>
  </div>


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

#User Login

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="" />
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>

  <%
      if (request.getParameter("userreg") != null) {%>
    <script>alert('User Registered Successfully');</script>
    <%}
      if (request.getParameter("userlogin1") != null) {%>
    <script>alert('User Login_Failed ');</script>
    <%}
      if (request.getParameter("userlogin2") != null) {%>
    <script>alert('User not Activated ');</script>

    <%}
    %>

<div id="wrapper">
 <div id="header">
  <div id="logo">
    <h4><a href="#">Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
  </div>
```

```
  <div id="slogan">

  </div>
 </div>

  <center>
 <div id="menu">

  <ul>
    <li><a href="index.html">Home</a></li>
    <li><a href="dataproviderlogin.jsp">Data Provider</a></li>
    <li><a href="cloudlogin.jsp">Cloud</a></li>
                <li class="selected"><a href="userlogin.jsp">User</a></li>
    <li><a href="aa.jsp">AA</a></li>
    <li class="last"><a href="contact.html">Contact</a></li>
  </ul>
  <br class="clearfix" />
 </div>
 </html>
 <center>
 <div class="content">
  <div class="content_resize">
   <div class="mainbar">
    <div class="article">
      <br>
     <div class="clr"></div>
     <!---Start Body --->
     <div id="splash">
   <img class="pic" src="images/arcte.JPG" width="600" height="230"
alt="" />
 </div><br><br>
          <h1><font color="black">User Login</h1><br>
    <center><table>
```

```
<form action="userloginact.jsp" method="get">
    <tr>
        <td>
                                                <strong><font size="3" color="black">Username:</font></strong>
                    <input type="text" name="username" id="userName1" placeholder= Username style="height:30px; width:200px"></input>
        </td>
    </tr>

    <tr>
        <td>
                    <strong><font size="3" color="black">Password: </font></strong>
                     <input type="password" name="password" id="password1" placeholder= Password style="height:30px; width:200px"></input>
        </td>
    </tr>

     <tr></tr>
  <tr></tr>
  <tr></tr>
  <tr>
        <td>

             <input type="submit" value="Login" style="height:30px; width:65px" />
              <button type="button" class="cancelbtn" style="height:30px; width:65px" >Cancel</button>
          <a href="userreg.jsp"><font color="#ff00cc" size="4"><u>New User?SignUp</u></font></a>
```

```
            </td>
          </tr>

          <tr>
             <td>
              </td>
          </tr>
         </table>
               </center>
         <!---End Body --->
      <br><br><br><br></div>
     </div>

      <div class="clr"></div>
    </div>
  </div></div>
</body>
</html>
```

#Key Generation

```
 <%@page import="attributebased.encryption"%>
<%@page
import="com.sun.org.apache.xerces.internal.impl.dv.util.Base64"%>
<%@page import="javax.crypto.SecretKey"%>
<%@page import="javax.crypto.KeyGenerator"%>
<%@page import="java.util.Random"%>
<%@page import="attributebased.Mail"%>
<%@page import="attributebased.decryption"%>
<%@page import="java.io.InputStreamReader"%>
<%@page import="java.io.BufferedReader"%>
<%@page import="java.io.InputStream"%>
<%@page import="java.sql.ResultSet"%>
```

```
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.Connection"%>
<%@page import="attributebased.Dbconnection"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%

   KeyGenerator keyGen = KeyGenerator.getInstance("AES");
                        keyGen.init(128);
                       SecretKey secretKey = keyGen.generateKey();
                        System.out.println("secret key:" + secretKey);
                      // converting secretkey to String
                                byte[] be = secretKey.getEncoded();//encoding
secretkey
                       String skey = Base64.encode(be);
                           System.out.println("converted secretkey to string:" +
skey);
                                // String cipher = new encryption().encrypt(str,
secretKey)

   String filename=request.getParameter("filename");
    String owner=request.getParameter("owner");
     String data=request.getParameter("data");

     Random r= new Random();
     int i=r.nextInt(10000 - 5000) + 5000;
     String publickey = i+"";

     Random r1= new Random();
     int i1=r1.nextInt(10000 - 5000) + 5000;
     String privatekey = i1+"";

     try{
         Connection con = Dbconnection.getConnection();
```

```
        Statement st = con.createStatement();
            int j = st.executeUpdate("insert into encryptkey(filename,owner,data,dkey,privatekey)values('"+filename+"','"+owner+"','"+data+"','"+publickey+"','"+skey+"')");

     if (j !=0){

        response.sendRedirect("dataproviderfiles.jsp?key=success");
     }
     else{
        response.sendRedirect("dataproviderfiles.jsp?key1=Failed");
     }

} catch (Exception ex) {
   response.sendRedirect("dataproviderfiles.jsp?key2=Failed");
                              ex.printStackTrace();
                           }

                        %>
```

#Attribute Verify

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
  <%@page import="attributebased.decryption"%>
<%@page import="java.io.InputStreamReader"%>
<%@page import="java.io.BufferedReader"%>
<%@page import="java.io.InputStream"%>
<%@page import="java.sql.ResultSet"%>
<%@page import="java.sql.Statement"%>
<%@page import="java.sql.Connection"%>
<%@page import="attributebased.Dbconnection"%>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="description" content="" />
```

```
<meta name="keywords" content="" />
<title>Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</title>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>

  <%
        if (request.getParameter("m1") != null) {%>
      <script>alert('Login Successfully');</script>
      <%}
        if (request.getParameter("dmsg") != null) {%>
      <script>alert('Vefication Failed   Attributes not Matched');</script>
       <%}
        if (request.getParameter("attributes") != null) {%>
      <script>alert('Vefication Failed   Attributes not Matched');</script>
       <%}
        if (request.getParameter("attributes1") != null) {%>
      <script>alert('Vefication Failed   Attributes not Matched');</script>

      <%}
      %>
<div id="wrapper">
  <div id="header">
     <div id="logo">Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud</a></h4>
    </div>
    <div id="slogan">

    </div>
  </div>
```

```
<center>
<div id="menu">

<ul>
<li><a href="aahome.jsp">Home</a></li>
<li><a href="viewdataproviders.jsp">View Data Providers</a></li>
<li><a href="viewusers.jsp">View Users</a></li>
<li class="selected"><a href="aaverify.jsp">User Attribute Verify</a></li>
<li><a href="cloudviewfilesprivate1.jsp">Public Cloud</a></li>
<li><a href="index.html">Logout</a></li>

</ul>
<br class="clearfix" />
</div>
<div id="splash">
<img class="pic" src="images/arcte.JPG" width="600" height="230" alt="" />
</div>
<div id="body">

<div id="main">
<div id="right">
<h4></h4>

<!--==========body start==========-->
<br>
<center><h3>Attribute Verification</h3>

<%
```

```
        String filename = request.getQueryString();

   try{

 Connection con=Dbconnection.getConnection();
 Statement st=con.createStatement();
 ResultSet rs=st.executeQuery("select * from request where filename='"+filename+"' and status='Policy Vefied'");



%>
    <center> <table style="width:60%" border="2">
        <br>

��� <tr STYLE="background-color: yellowgreen;font-size: 15px;">

 ���  <td>File Name</td>
      <td>Owner</td>
     <td>Policy</td>
     <td>Time</td>
     <td>Experience</td>
     <td>Branch</td>
     <td>User</td>
     <td>Status</td>
     <td>View</td>

 ��� </tr>
<%

  if(rs.next()){

        %><tr>
```

```
<td><%=rs.getString(1)%></td>
<td><%=rs.getString(3)%></td>
<td><%=rs.getString(4)%></td>
<td><%=rs.getString(5)%></td>
<td><%=rs.getString(6)%></td>
<td><%=rs.getString(7)%></td>
  <td><%=rs.getString(8)%></td>
<td><%=rs.getString(9)%></td>
                                                                    <td><a
href="attributeverify1.jsp?filename=<%=rs.getString("filename")%>&own
er=<%=rs.getString("owner")%>&umail=<%=rs.getString("umail")%>">V
erify Attributes</a> </td>
      </tr>

       <%  }
%></table></center>

<%}
 catch(Exception e)
 {
  System.out.println(e);
 }
 %>
                  </table>
<br>
</div>

</div>
<!--content ends -->
<!--footer begins -->
<!-- footer ends-->
</body>
</html>
```

A

PROJECT REPORT ON

ATTRIBUTE BASED ENCRYPTION APPROACH FOR STORING, SHARING AND RETRIEVAL OF ENCRYPTED DATA IN THE CLOUD

Submitted in partial fulfillment of the
requirements for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

By

Ms V. SOWJANYA,
(Regd.No:215N1F00A1).

Under the Guidance of

Mr A. UMA MAHESWAR REDDY,
Assistant Professor, APGCCS.

&

Mr G. VENKAT RAO,
Project Leader,
Manosys Technologies Pvt Ltd,Bangalore.

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

ANNAMACHARYA P.G COLLEGE OF COMPUTER STUDIES

NEW BOYANAPALLI-516126, RAJAMPET (A.P)

(Approved by A.I.C.T.E., New Delhi & Affiliated to J.N.T.U.A,
Anantapuramu, UGC(2f) Recognized Institution)

(2021-2023)

***

ANNAMACHARYA P.G COLLEGE OF COMPUTER STUDIES

NEW BOYANAPALLI, RAJAMPET, A.P, 516126.

Affiliated to

JAWAHARLAL NEHRU TECHNOLOGICALUNIVERSITY ANANTAPUR, ANANTAPURAMU

DEPARTMENT OF

MASTER OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that the project work entitled **"Attribute Based Encryption Approach For Storing, Sharing And Retrieval Of Encrypted Data In The Cloud"** is the Bonafide work carried out by **Ms V. Sowjanya,** Regd. No:**215N1F00A1,**is submitted in the partial fulfillment of the requirements for the award of degree of **Master of Computer Applications** during the year 2021-2023.

Project Guide **Principal**

External Examiner

DECLARATION

I,**V. Sowjanya,** hereby declare that the project report entitled as **"Attribute Based Encryption Approach For Storing, Sharing And Retrieval Of Encrypted Data In The Cloud"** is done in **Manosys Technologies Pvt Ltd, Bangalore**, is original and independent record of work, submitted by me to JNTUA, Anantapuramu, under the guidance of **Mr A. UMA MAHESWAR REDDY,** Assistant Professor of **Annamacharya P.G College of Computer Studies,** Rajampet, for the award of the degree of **Master of Computer Applications** and has not been submitted either in partial or full for the award of any Degree or Diploma.

Place: Rajampet **V. SOWJANYA,**

Date: **(Regd.No:215N1F00A1).**

ACKNOWLEDGEMENT

An endeavor over a long period can be successful only with the advice of many well-wishers.I take this opportunity to express my deep gratitude and appreciation of all those who encourage me to successfully complete the project.

I wish to express my sincere gratitude to **Dr D.J. SAMATHANAIDU**, Principal of **Annamacharya P.G College of Computer Studies**, New Boyanapalli, Rajampet, for her consistent help and providing such facilities to complete.

I express my sincere thanks to my guide **Mr A. UMA MAHESWAR REDDY,** Assistant Professor in MCA Deportment for his valuable guidance and suggestions in analyzing and testing throughout the period of my project work.

I express my sincere thanks to my guide **Mr G. VENKAT RAO**, Project Leader, Manosys Technologies Pvt Ltd, Bangalore for his valuable guidance and suggestions in analyzing and testing throughout the period of my project work.

Last but not least, I would like to thank my friends, teaching and nonteaching, one and all those who helped me to complete this project successfully.

V. SOWJANYA,
(Regd.No:215N1F00A1).