A NEW APPROACH FOR AUTHENTICATION TECHNIQUE

A PROJECT REPORT

Submitted by

DEEPA.R	71204205005
SIVAMALAR.M.M	71204205050

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

ANNA UNIVERSITY: CHENNAI 600 025

APRIL 2008

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **"A NEW APPROACH FOR AUTHENTICATION TECHNIQUE"** is the bonafide work of **"DEEPA.R** and **SIVAMALAR.M.M"** who carried out the project work under my supervision.

SIGNATURE
Dr. S.Thangasamy.B.E(Hons).,Ph.D.,

SIGNATURE
Mr.K.R.BASKARAN.B.E.,M.S
Asst.Professor/IT

Head **~~DEAN~~ OF THE DEPARTMENT**
Dept of Information Technology,
Kumaraguru College of Technology,
Coimbatore – 641 006.

SUPERVISOR'S
Dept of Information Technology,
Kumaraguru College of Technology,
Coimbatore – 641 006.

The candidates with University Register No 71204205005 and 71204205050 were examined by us in the project viva-voice examination held on..24.04.2008

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

We express our sincere thanks to our chairman **Padmabhushan Arutselvar Dr. N. Mahalingam B.Sc, F.I.E** and correspondent **Prof. Balasubramanian.,** for all their support and ray of strengthening hope extended. We are immensely grateful to our principal **Dr. Joseph V. Thanikal M.E., Ph.D., PDF., CEPIT.,** for his invaluable support to the outcome of this project.

We also thank **Dr.S.Thangasamy.B.E(Hons).,** Dean of the department of Computer Science and engineering for his valuable advice and useful suggestions throughout this project.

We also extend our heartfelt thanks to our project guide **Mr.K.R.BASKARAN.B.E.,M.S.,Asst.Professor,** Department of Information Technology who rendered his valuable guidance and support to perform our project work extremely well.

We express our heartiest thanks to our project coordinators **Mr.K.R.BASKARAN.B.E.,M.S.,Asst.Professor,** Department of Information Technology who have helped us to perform our project work extremely well.

We thank the teaching and non-teaching staffs of our Department for providing us the technical support in the duration of our project.

We express our humble gratitude and thanks to our beloved parents and family members who have supported and helped us to complete the project and our friends, for lending us valuable tips, support and co-operation throughout the project work.

DECLARATION

DECLARATION

We,

DEEPA.R	**71204205005**
SIVAMALAR.M.M	**71204205050**

hereby declare that the project entitled **"A NEW APPROACH FOR AUTHENTICATION TECHNIQUE"**, submitted in partial fulfillment to Anna University as the project work of Bachelor of Technology (Information Technology) Degree, is a record of original work done by us under the supervision and guidance of Department of Information Technology, Kumaraguru college of Technology, Coimbatore.

Place: Coimbatore

Date: 22.04.08

R. Deepa

[Deepa.R]

M.M.Sivamalar

[Sivamalar.M.M]

Project Guided by

[Mr.K.R.Baskaran]

ABSTARCT

ABSTRACT

One of the serious problems is how to authenticate the passport document for its holder. The major factor of this authenticity is the corresponding of the Passport's photo with its holder. Most of the Passport document contains a holder's signature in addition, of course, to the full name. We propose a firm authentication method by extracting some features for the original name of the holder with the passport number and digest them in a form, by applying some techniques that can be hidden in the passport's photo. The modern method of issuing a passport now is by using a computer in fixing the passport's photo (imaging).

In using this method we could hide the invisible watermark which contains the digest name and passport number inside the passport's photo. During the hidden process there are many techniques could be applied to disguise any color difference appears during the hidden Process. After using this technique, it is very simple to use the computer in scanning and verifying, at check point, that the passport's photo has been not replaced and that by comparing the invisible Watermark with the digest name of the holder and passport number.

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES

LIST OF ABBREVATIONS

LIST OF ABBREVATIONS

JVM	: Java Virtual Machine
GUI	: Graphical User Interface
API	: Application Programming Interface
ASCII	: American Standard Code for Information Interchange
AWT	: Abstract Window Toolkit
JFC	: Java Foundation Classes

INTRODUCTION

1. INTRODUCTION

1.1 GENERAL:

"A New Approach for Authentication Technique" is to authenticate the passport's photo with the information about the holder. Most of the Passport document contains a holder's signature in addition, to the full name. We propose a firm authentication method by extracting some features for the original name of the holder with the passport number and digest them in a form, by applying some techniques that can be hidden in the passport's photo. The modern method of issuing a passport now is by using a computer in fixing the passport's photo (imaging).

In using this method we could hide the invisible watermark which contains the digest name and passport number inside the passport's photo. During the hidden process there are many techniques could be applied to disguise any color difference appears during the hidden Process. After using this technique, it is very simple to use the computer in scanning and verifying, at check point, that the passport's photo has been not replaced and that by comparing the invisible Watermark with the digest name of the holder and passport number.

1.2 PROBLEM DEFINITION:

The main objective of our project is to design a system that prevents duplication of photo in passport. The modern method of issuing a passport now is by using a computer in fixing the passport's photo (imaging). By using watermarking technique we can hide the name and passport number inside the passport's photo.

Watermarking technique has been suggested to play an important role in securing the information, as it allows placing an imperceptible mark in the multimedia data to identify the legitimate owner and to prosecute the pirate. There are number of techniques for watermarking technique:

Robustness: The embedded information is said to be robust if its presence can be reliably detected after the image has been modified, but not destroyed beyond recognition.

Invisibility: The embedded information is imperceptible if an average human is unable to distinguish between carries that contain the hidden information and those that do not.

Undetectability: The concept of undectectability is that if an attacker has a more detailed model of the source, he may be able to detect the presence of the hidden image, but this does not imply the ability to read the hidden message.

Security: The embedding algorithm is said to be secure, if the embedded information can not remove beyond reliable detection by targeted attacks based on a full knowledge of the embedding algorithm and the detector.

LITERATURE REVIEW

2. LITERATURE REVIEW

2.1 FEASIBILITY STUDY:

2.1.1 EXISTING SYSTEM:

In present system most of the passport document contains the holder's signature in addition to the full name. The issuing of passport is done by using a compüter in fixing the passport's photo. There is no connection between the passport's photo and the passport's details.

Limitations:

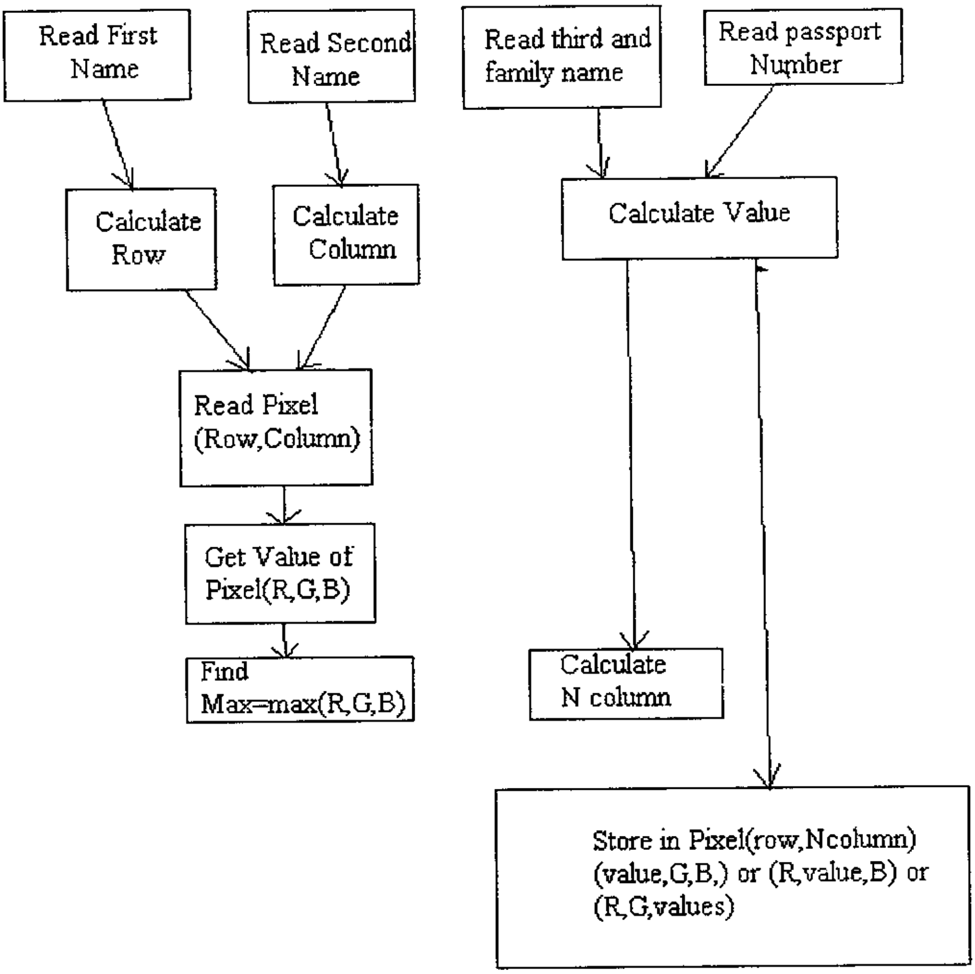
The main problem is how to confirm and authenticate the passport's photo with the information about the holder. In the existing system there is possibility of the photos being replaced.

2.1.2 PROPOSED SYSTEM:

In this proposed system we adopt a firm authentication method by extracting some features for the original name of the holder with the passport number and digest them in a form, by applying some techniques that can be hidden in the passport's photo.

Advantages of the proposed system:

After using this technique, it is very simple to use the computer in scanning and verifying, at check point, that the passport's photo has been not replaced and that by comparing the invisible Watermark with the digest name of the holder and passport number. The information's hidden are robust, undecetable, invisible and secure.



(Fig.1) Method for confirmation of passport's information

2.2 HARDWARE SPECIFICATIONS:

Processor	:	Intel Pentium IV 2.2 GHZ
Hard Disk Drive	:	20 GB.
RAM	:	256 MB RAM minimum.

2.3 SOFTWARE SPECIFICATIONS:

Operating System:		Windows 2000/XP
Software	:	JCreator 3.50 LE, Netbeans

2.4. SOFTWARE OVERVIEW:

2.4.1 JAVA:

Java is a new computer programming language developed by Sun Microsystems.-Java has a good chance to be the first really successful new computer language in several decades. Advanced programmers like it because it has a clean, well-designed definition. Business likes it because it dominates an important new application, Web programming.

Java has several important features:

- A Java program runs exactly the same way on all computers. Most other languages allow small differences in interpretation of the standards.
- It is not just the source that is portable. A Java program is a stream of bytes that can be run on any machine. An interpreter program is built into Web browsers, though it can run separately. Java programs can be distributed through the Web to any client computer.

- Java applets are safe. The interpreter program does not allow Java code loaded from the network to access local disk files, other machines on the local network, or local databases. The code can display information on the screen and communicate back to the server from which it was loaded.

A group at Sun reluctantly invented Java when they decided that existing computer languages could not solve the problem of distributing applications over the network. C++ inherited many unsafe practices from the old C language. Basic was too static and constrained to support the development of large applications and libraries.

Today, every major vendor supports Java. Netscape incorporates Java support in every version of its Browser and Server products. Oracle will support Java on the Client, the Web Server, and the Database Server. IBM looks to Java to solve the problems caused by its heterogeneous product line.

The Java programming language and environment is designed to solve a number of problems in modern programming practice. It has many interesting features that make it an ideal language for software development. It is a high-level language that can be characterized by all of the following buzzwords:

Features

Sun describes Java as

- Simple
- Object-oriented

- Distributed
- Robust
- Secure
- Architecture Neutral
- Portable
- Interpreted
- High performance
- Multithreaded
- Dynamic.

Java is an object–oriented programming language developed by sun Microsystems, a company best known for its high-end UNIX/LINUX workstations.Modeled after C++, the java language was designed to be small, simple and portable across platforms and operating systems, both at source and the binary level, which means that java programs can run on any machine that has java virtual machine installed. They are two types of java programs. They are java applets and java applications.

To develop application programs java provides so many classes in "awt" package and in java 2.0 there is another package called "swing" the extended version of awt package is used to design the application programs as very much like popular application packages like Visual Basic, Developer 2000.

Java is a platform independent at both the source level and the binary level; platform independence means that a program can run on any computer system. Java programs can run or any system for which a Java virtual machine has been installed. Unlike other programming languages when Java programs are compiled byte codes are generated which is a special set of machine instructions that are not specific to any one-processor or computer system.

Unlike most object-oriented languages, Java includes a set of input and output capabilities and other utility functions. Then basic libraries are part of standard environment, which also includes simple libraries from networking, common Internet protocols and user interface toolkit functions. Because the libraries are written in Java, they are portable across platforms as all Java applications are. In Java 2.0 so many features are included such as Swing, RMI and so on.

2.4.2 OVERVIEW OF SWING:

The Swing package is part of Java Foundation Classes (JFC) in the Java platform. The JFC encompasses a group of features to help people build GUIs; Swing provides all the components from buttons to split panes and tables.

The Swing package was first available as an add-on to JDK 1.1. Prior to the introduction of the Swing package, the Abstract Window Toolkit (AWT) components provided all the UI components in the JDK1.0 and 1.1 platforms. Although the Java2 Platform still supports the AWT components, we strongly encourage using Swing components instead. You can identify Swing components because their names start with J. The AWT button class, for example, is named Button, whereas the Swing button class is named JButton. In addition, the AWT components are in the java.awt package, whereas the swing components are in the javax.swing package.

As a rule, programs should not use "heavyweight" AWT components alongside Swing components. Heavyweight components include all the ready-to-use AWT components, such as Menu and ScrollPane, and all components that inherit from the AWT canvas and Panel classes. When Swing components (and all other "lightweight" components) overlap with heavyweight components, the heavyweight component is always painted on top.

Swing provides many standard GUI components such as buttons, lists, menus and textareas, which we combine to create our program's GUI. It also includes containers such as windows and tool bars.

More Swing Features and Concepts:

These are some of the major concepts we need to know to build Swing GUIs—the containment hierarchy, layout management, event handling, painting, and threads. These are some of the other important Swing features. They are

- Features that JComponent provides
- Icons
- Actions
- Pluggable Look and Feel support
- Support for assistive technologies
- Separate data and state models

Except for the top-level containers, all components that begin with J inherit from the JComponent class. They get many features from JComponent, such as the ability to have borders, tool tips, and a configurable look and feel. They also inherit many convenient methods.

Pluggable Look and Feel:

This Feature enables the user to switch the look- and-feel of Swing components without restarting the application. The Swing library supports cross-platform look-and-feel---also Java look-and-feel--- that remains the same across all platforms wherever the program runs. The native look-and-feel is native to whatever particular system on which the program happens to be running, including Windows and Motif. The Swing library provides an API that gives great flexibility in determining the look-and-feel of applications. It also enables you to create our own look-and-feel.

Some Swing Components of Java:

JScrollPane:

Provides a scrollable view of a lightweight component. It is used to display a child component with a built-in scrolling facility. The scrolling of a child component, when its size is larger than the available view port, is performed in horizontal or vertical directions by using the scrollbars associated with the scroll pane. Scroll panes are very easy to implement because the adjustment events fired by the scrollbars are already taken care of by the scrollpane object. A Swing scroll pane is an object of type JScrollPane that extends from the class JComponent.

JButton:

Swing buttons are represented by the objects of class JButton, and each button is basically an implementation of a push-type button. Unlike AWT buttons, Swing buttons can be displayed with text labels as well as icons. We can also set different icon for different states of the buttons by using supporting methods.

JFrame:

An extended version of java.awt.Frame that adds support for the JFC/Swing components architecture. The JFrame class is slightly incompatible with Frame. Like all other JFC/Swing top-level containers, a JFrame contains a JRootPane as its only child. The content pane provided by the root pane should, as a rule, contain all the non-menu components displayed by the JFrame.

JTextField:

The swing text field can be used to display or edit a single line of plain text. The component appears similar to the AWT text field; however, the Swing text field is a lightweight component. A text-field object is created by using the class JTextField, which is a direct subclass of JTextComponent. Thus, the functionality of JTextField spreads into JTextComponent and JComponent. JTextField objects can fire the action and mouse events that can be captured by a registered listener.

2.4.3 JCREATOR:

JCreator is a powerful IDE for Java technologies that provides more power at your fingertips than all the ordinary IDEs combined. Unlike most IDEs, JCreator wraps around your existing projects and allows you to use different JDK profiles. Get down to writing code quickly with our project templates. You don't have to spend valuable time on Classpath configuration—JCreator does it all for you. Customize our user interface the way that you like it.Set up your own run-time environments to run your application as an applet, in a JUnit environment or in a DOS window.

JCreator has lower system requirements, yet faster speed, than all those other IDEs. JCreator is faster and more efficient than Java-based IDEs. This powerful tool is written entirely in a native Windows language. JCreator is professionally designed to meet Microsoft Windows interface guidelines: you can work quickly and efficiently with the intuitive Windows interface.

A powerful search and replace engine for single and multiple documents. The result of the search will be displayed in the output view. External file modification detection .JCreator will automatically ask to reload a file when it has been altered outside the editor.

DETAILS OF METHODOLGY EMPLOYED

3. DETAILS OF METHODOLOGY EMPLOYED

In this section, we elaborate on our proposed passport authentication system which consists of three algorithms. Each algorithm is responsible for one type of process.

3.1 PARAMETER ACQUISITION:

1. Read first, second, third and family name.
2. Read passport number.
3. Validate entries.
4. Assign each letter a number according to Table: 1.
5. Keep each name’s number.

Table: 1

Letters with Their Equivalent Numbers

A 1	B 2	C 3	D 4	E 5	F 6	G 7	H 8	I 9	J 10	K 11	L 12	M 13
N 14	O 15	P 16	Q 17	R 18	S 19	T 20	U 21	V 22	W 23	X 24	Y 25	Z 26

3.2 CONVERT:

1. Consider the key value. E.g. key = "1, 2, 3, 4"
2. Get the summation of the first name by adding the code of each character multiplied by the key's character on a sequence manner. E.g. code [1]* key [1] + code [2[*key [2] + code [3]*key [3] +……
3. Consider the result as “row ”.

4. Repeat step (2) for the second name.

5. Consider the result as “column ".

6. Repeat step (2) for the third name and family name.

7. Add the third name, family name and the passport number and the result will be "value".

3.3 HIDE:

1. Read the value of the pixel on location (row, column) from the original Image.

2. Find the largest value of RGB color for that pixel and assign it to "large ".

3. Divide "value” on “large “to get number of pixels.

4. Calculate the modulo of “value” over “large” and assign it to " color ".

5. Calculate Ncolumn so that equals to

    " column " + " No. of pixel " + 1.

6. Get the pixel value in location (row, Ncolumn).

7. Replace the largest value of RGB for that pixel with "color".

8. Restore the pixel at the same location.

CONCLUSION

4. CONCLUSION

This system a new approach for authentication technique has great potential for practical applications. The suggested invisible watermark has satisfied the Invisibility, undetectibility, and Security requirements. Because the proposed technique has used a small area for hiding (one pixel), so it satisfied the robustness against image compression. This system possesses many advantages, such as avoiding the replacement of photo, low cost, easy use and high efficiency.

FUTURE ENHANCEMENT

5. FUTURE ENHANCEMENT

The proposed method is secure and effective, but it works for one National State and could be used between more than two States and that by transmitting copy of the passport between them for authentication. It is possible to make the suggested method works globally and that by using a public key for each Country to hide the watermark and a private key to Open it by different countries. In future, this technique can be used in real-time application such the images can be transferred from web camera to system.

APPENDICES

6. APPENDICES

6.1 SOURCE CODE

MAIN:

```
import javax.swing.*;
 import java.text.ParseException;
import javax.swing.text.MaskFormatter;
import javax.swing.text.DefaultFormatterFactory;
public class main extends javax.swing.JFrame
 {
   JFileChooser filechooser;
   int fval,sval,tval,lval,pval,row,column,value;
   String f1=new String();
   /** Creates new form main */
   public main()
   {
      initComponents();
      filechooser=new JFileChooser();
      filechooser.setFileSelectionMode(JFileChooser.FILES_ONLY);
   }

   /** This method is called from within the constructor to
    * initialize the form.
    * WARNING: Do NOT modify this code. The content of this method is
    * always regenerated by the Form Editor.
    */
   private void initComponents()
   {
       //GEN-BEGIN:initComponents
     java.awt.GridBagConstraints gridBagConstraints;
     MaskFormatter withFocus=null,withoutFocus=null;
   try
    {
    withFocus = new MaskFormatter("HHHHH");
    withoutFocus = new MaskFormatter("?#######");
    }
    catch (ParseException pe)
    {
}

  DefaultFormatterFactory factory = new
  DefaultFormatterFactory(withFocus);
  jPanel1 = new javax.swing.JPanel();
  jLabel1 = new javax.swing.JLabel();
  jPanel2 = new javax.swing.JPanel();
  jButton1 = new javax.swing.JButton();
  jButton2 = new javax.swing.JButton();
  jButton3 = new javax.swing.JButton();
  jPanel3 = new javax.swing.JPanel();
  jPanel4 = new javax.swing.JPanel();
  jLabel2 = new javax.swing.JLabel();
  jTextField1 = new javax.swing.JTextField();
  jTextField1.setInputVerifier(new algorithm1());
  jLabel3 = new javax.swing.JLabel();
  jTextField2 = new javax.swing.JTextField();
  jTextField2.setInputVerifier(new algorithm1());
  jLabel4 = new javax.swing.JLabel();
  jTextField3 = new javax.swing.JTextField();
  jTextField3.setInputVerifier(new algorithm1());
  jLabel5 = new javax.swing.JLabel();
  jTextField4 = new javax.swing.JTextField();
  jTextField4.setInputVerifier(new algorithm1());
  jLabel6 = new javax.swing.JLabel();
  jFormattedTextField5 = new
  javax.swing.JFormattedTextField(withoutFocus);
  jPanel5 = new javax.swing.JPanel();
  jScrollPane1 = new javax.swing.JScrollPane();
  jLabel7 = new javax.swing.JLabel();

  addWindowListener(new java.awt.event.WindowAdapter() {
    public void windowClosing(java.awt.event.WindowEvent evt) {
      exitForm(evt);
    }
  });

  jPanel1.setBackground(new java.awt.Color(204, 204, 255));
  jLabel1.setFont(new java.awt.Font("Microsoft Sans Serif", 3, 12));
  jLabel1.setText("PASSPORT AUTHENDICATION");
```

```
jPanel1.add(jLabel1);

getContentPane().add(jPanel1, java.awt.BorderLayout.NORTH);

jPanel2.setBackground(new java.awt.Color(204, 204, 255));
jButton1.setFont(new java.awt.Font("Microsoft Sans Serif", 3, 11));
jButton1.setText("Select");
jButton1.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      jButton1ActionPerformed(evt);
   }
});

jPanel2.add(jButton1);

jButton2.setFont(new java.awt.Font("Microsoft Sans Serif", 3, 11));
jButton2.setText("Compute");
jButton2.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      jButton2ActionPerformed(evt);
   }
});

jPanel2.add(jButton2);

jButton3.setFont(new java.awt.Font("Microsoft Sans Serif", 3, 11));
jButton3.setText("Compare");
jButton3.addActionListener(new java.awt.event.ActionListener() {
   public void actionPerformed(java.awt.event.ActionEvent evt) {
      jButton3ActionPerformed(evt);
   }
});

jPanel2.add(jButton3);

getContentPane().add(jPanel2, java.awt.BorderLayout.SOUTH);
jPanel3.setLayout(new java.awt.GridLayout(1, 2));

jPanel4.setLayout(new java.awt.GridBagLayout());
jPanel4.setBackground(new java.awt.Color(204, 204, 255));
```

```
jLabel2.setFont(new java.awt.Font("Microsoft Sans Serif", 3, 11));
jLabel2.setText("First Name");
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.ipadx = 4;
gridBagConstraints.insets = new java.awt.Insets(10, 14, 10, 14);
gridBagConstraints.anchor = java.awt.GridBagConstraints.WEST;
jPanel4.add(jLabel2, gridBagConstraints);

jTextField1.setPreferredSize(new java.awt.Dimension(100, 21));
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.gridwidth =
java.awt.GridBagConstraints.REMAINDER;
jPanel4.add(jTextField1, gridBagConstraints);

jLabel3.setFont(new java.awt.Font("Microsoft Sans Serif", 3, 11));
jLabel3.setText("Second Name");
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.insets = new java.awt.Insets(10, 14, 10, 14);
gridBagConstraints.anchor = java.awt.GridBagConstraints.WEST;
jPanel4.add(jLabel3, gridBagConstraints);

jTextField2.setPreferredSize(new java.awt.Dimension(100, 21));

gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.gridwidth =
java.awt.GridBagConstraints.REMAINDER;
jPanel4.add(jTextField2, gridBagConstraints);

jLabel4.setFont(new java.awt.Font("Microsoft Sans Serif", 3, 11));
jLabel4.setText("Third Name");
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.insets = new java.awt.Insets(10, 14, 10, 14);
gridBagConstraints.anchor = java.awt.GridBagConstraints.WEST;
jPanel4.add(jLabel4, gridBagConstraints);

jTextField3.setPreferredSize(new java.awt.Dimension(100, 21));


gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.gridwidth =
java.awt.GridBagConstraints.REMAINDER;
jPanel4.add(jTextField3, gridBagConstraints);

jLabel5.setFont(new java.awt.Font("Microsoft Sans Serif", 3, 11));
jLabel5.setText("Family Name");
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.insets = new java.awt.Insets(8, 14, 8, 14);
gridBagConstraints.anchor = java.awt.GridBagConstraints.WEST;
jPanel4.add(jLabel5, gridBagConstraints);

jTextField4.setPreferredSize(new java.awt.Dimension(100, 21));
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.gridwidth =
java.awt.GridBagConstraints.REMAINDER;
jPanel4.add(jTextField4, gridBagConstraints);

jLabel6.setFont(new java.awt.Font("Microsoft Sans Serif", 3, 11));
jLabel6.setText("Passport No");
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.insets = new java.awt.Insets(8, 14, 8, 14);
gridBagConstraints.anchor = java.awt.GridBagConstraints.WEST;
jPanel4.add(jLabel6, gridBagConstraints);

jFormattedTextField5.setPreferredSize(new java.awt.Dimension(100,
21));
jPanel4.add(jFormattedTextField5, new java.awt.GridBagConstraints());
jPanel3.add(jPanel4);

jPanel5.setLayout(new java.awt.GridBagLayout());
jPanel5.setBackground(new java.awt.Color(204, 204, 255));
jLabel7.setBackground(new java.awt.Color(204, 204, 255));
jLabel7.setPreferredSize(new java.awt.Dimension(150,130));
jScrollPane1.setViewportView(jLabel7);

jPanel5.add(jScrollPane1, new java.awt.GridBagConstraints());
jPanel3.add(jPanel5);
getContentPane().add(jPanel3, java.awt.BorderLayout.CENTER);
pack();
}//GEN-END:initComponents
```

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton3ActionPerformed

 hide1 h=new hide1(row,column,value,f1);
 this.hide();

}//GEN-LAST:event_jButton3ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{

     String fname=jTextField1.getText();
     String sname=jTextField2.getText();
     String tname=jTextField3.getText();
     String lname=jTextField4.getText();
     String pnum=jFormattedTextField5.getText();
     System.out.println("Passport Number is"+pnum);

     if((fname.equals(""))||(sname.equals(""))||(tname.equals(""))||(lname.
     equals(""))||(pnum.equals("")))
      {
      JOptionPane.showMessageDialog(null, "Values should not be null",
     "Message", JOptionPane.ERROR_MESSAGE);
      }
      else
      {

      algorithm2 a1=new algorithm2();
      fval=a1.cal(fname);
      sval=a1.cal(sname);
      tval=a1.cal(tname);
      lval=a1.cal(lname);
      pval=a1.cal(pnum);
      System.out.println("First Name:"+fval);
      System.out.println("Second Name :"+sval);
      System.out.println("third Name:"+tval);
      System.out.println("Family Name:"+lval);
      System.out.println("Passport Num:"+pval);
      row=fval;
      column=sval;
      value=tval+lval+pval;
      System.out.println("Row:"+row);
      System.out.println("Column:"+column);
      System.out.println("Value is:"+value);

            }


      }

   private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_jButton1ActionPerformed
      if(evt.getSource()==jButton1)
     {
       filechooser.showOpenDialog(this);
       f1=filechooser.getSelectedFile().getAbsolutePath();
       System.out.println(f1);
       ImageIcon im=new ImageIcon(f1);
       jLabel7.setIcon(im);
     }

   }//GEN-LAST:event_jButton1ActionPerformed

   /** Exit the Application */
   private void exitForm(java.awt.event.WindowEvent evt)
   {//GEN-FIRST:event_exitForm
      System.exit(0);
   }//GEN-LAST:event_exitForm

   /**
    * @param args the command line arguments
    */
   public static void main(String args[])
    {
      new main().show();
    }

   // Variables declaration - do not modify//GEN-BEGIN:variables
   private javax.swing.JButton jButton1;
```

```
 private javax.swing.JButton jButton2;
 private javax.swing.JButton jButton3;
 private javax.swing.JTextField jTextField1;
 private javax.swing.JTextField jTextField2;
 private javax.swing.JTextField jTextField3;
 private javax.swing.JTextField jTextField4;
 private javax.swing.JFormattedTextField jFormattedTextField5;
 private javax.swing.JLabel jLabel1;
 private javax.swing.JLabel jLabel2;
 private javax.swing.JLabel jLabel3;
 private javax.swing.JLabel jLabel4;
 private javax.swing.JLabel jLabel5;
 private javax.swing.JLabel jLabel6;
 private javax.swing.JLabel jLabel7;
 private javax.swing.JPanel jPanel1;
 private javax.swing.JPanel jPanel2;
 private javax.swing.JPanel jPanel3;
 private javax.swing.JPanel jPanel4;
 private javax.swing.JPanel jPanel5;
 private javax.swing.JScrollPane jScrollPane1;
 // End of variables declaration//GEN-END:variables

}
class algorithm1 extends InputVerifier {
      public boolean verify(JComponent input) {
                  JTextField tf = (JTextField) input;
                  String s=tf.getText();
                  byte c[]=new byte[s.length()];
                  c=s.getBytes();
                  int i;
                  boolean flag=false;
                   for( i=0;i<s.length();i++)
                      {

                        if((c[i]>=65&&c[i]<=122 || c[i]==32))
                  {

                         flag=true;
                   }
```

```
                Else
                  {
                   JOptionPane.showMessageDialog(null,"Please Enter
                   Valid Input");
                   flag=false;
                   break;
                 }

          }

      return(flag);
   }
   }
```

ALGORITHM 2:

```
class algorithm2
{
        int cal(String input)
        {
                String name=new String(input);
                int sum=0;
                int ii=1;
                int result;
                String uname=name.toUpperCase();
                int namelen=uname.length();
                System.out.println(namelen);
                char[] name1=new char[namelen];
                name1=uname.toCharArray();
                for(int i=0;i<namelen;i++)
                {
                        ii=ii+0;
```

```
//        System.out.println(name1[i]);
          String val=new String();
          val=val.valueOf(name1[i]);
          System.out.println(val);
          table t=new table();
          if(((byte)name1[i]>=48) && ((byte)name1[i]<=57))
          {
                  System.out.println("entered into if loop");
                  result=Integer.parseInt(val);
          }
          else
          {
                  result=(Integer)t.alphabets.get(val);
          }
          System.out.println(result);
          if(((ii%5)!=0))
          {
            if((byte)name1[i]!=32)
            {
              if(((byte)name1[i]>=48) && ((byte)name1[i]<=57))
                 {
                     result=Integer.parseInt(val);
                     sum=sum+result;
                 }
               else
                 {
                         sum=sum+ii*result;
                        }
                   }
              }
              else
              {
                 if((byte)name1[i]!=32)
              {
              if(((byte)name1[i]>=48) && ((byte)name1[i]<=57))
               {
                 result=Integer.parseInt(val);
                sum=sum+result;
              }
               else
                 {
                      ii=1;
                       sum=sum+ii*result;
               }
           }
          }
          ii++;
    }
          System.out.println(sum);
          return sum;
          }
          //public static void main(String ar[])
          //{
```

```
        //      algorithm2 m=new algorithm2();
        //}
}
```

TABLE:

```
import java.util.*;
public class table
{
        Hashtable alphabets=new Hashtable();
        Enumeration names;
        String str;
        table()
        {
                alphabets.put("A",new Integer(1));
                alphabets.put("B",new Integer(2));
                alphabets.put("C",new Integer(3));
                alphabets.put("D",new Integer(4));
                alphabets.put("E",new Integer(5));
                alphabets.put("F",new Integer(6));
                alphabets.put("G",new Integer(7));
                alphabets.put("H",new Integer(8));
                alphabets.put("I",new Integer(9));
                alphabets.put("J",new Integer(10));
                alphabets.put("K",new Integer(11));
                alphabets.put("L",new Integer(12));
                alphabets.put("M",new Integer(13));
```

```
            alphabets.put("N",new Integer(14));
            alphabets.put("O",new Integer(15));
            alphabets.put("P",new Integer(16));
            alphabets.put("Q",new Integer(17));
            alphabets.put("R",new Integer(18));
            alphabets.put("S",new Integer(19));
            alphabets.put("T",new Integer(20));
            alphabets.put("U",new Integer(21));
            alphabets.put("V",new Integer(22));
            alphabets.put("W",new Integer(23));
            alphabets.put("X",new Integer(24));
            alphabets.put("Y",new Integer(25));
            alphabets.put("Z",new Integer(26));
            alphabets.put(" ",new Integer(-32));
    }
}
```

HIDE1:

```
import javax.swing.*;
import java.io.*;
import java.awt.MediaTracker.*;
import java.awt.image.*;
import java.awt.*;
import javax.imageio.*;
import java.util.*;
import java.lang.Math.*;
```

```
class hide1
{
BufferedImage img,img1;
int h,w;
int pixel[];
int pixel1[];
int pixels[][];
int pixels1[][];
static ImageIcon outputimage;
int red1=0;
int blue1=0;
int green1=0;


public hide1(int row,int col,int value,String f1)
{
try
{
img=ImageIO.read(new File(f1));
h=img.getHeight();
w=img.getWidth();
pixel=new int[w*h];
pixel1=new int[w*h];
pixels =new int[w][h];
pixels1 =new int[w][h];
pixel1=img.getRGB(0,0,w,h,pixel,0,w);
```

```
System.out.println("The height of the image:"+h);
System.out.println("The width of the image:"+w);
System.out.println("The rgb value:"+pixel1);
}
catch(Exception e)
{
}
int red=0;
int blue=0;
int green=0;
int noofpixel;
int color,Ncoloum,t;
        pixels[row][col]=pixel[row*col];
        red=(pixels[row][col]>>16)&0xFF;
        green=(pixels[row][col]>>8)&0xFF;
        blue=(pixels[row][col])&0xFF;

         System.out.println("The value of red:"+red);
          System.out.println("The value of blue:"+blue);
          System.out.println("The value of green:"+green);
         System.out.println("The pixel value:"+pixels);

         if(red>blue)
         {
               if(red>green)
               System.out.println("The highest RGB value is red :"+red);
               noofpixel=value/red;
        System.out.println("The no of pixels:"+noofpixel);
        color=value%red;
        System.out.println("The color:"+        color);
        Ncoloum=noofpixel+col+1;
        System.out.println("The Nthcoloum:"+Ncoloum);


}
        else
        {
                if(blue>green)
                {
        System.out.println("The highest RGB value is blue:"+blue);
        noofpixel=value/blue;
        System.out.println("The no of pixels:"+noofpixel);
        color=value%blue;
        System.out.println("The color:"+        color);
        Ncoloum=noofpixel+col+1;
        System.out.println("The Nthcoloum:"+Ncoloum);

        }
        else{

        System.out.println("The highest RGB value is green:"+green);
        noofpixel=value/green;
        System.out.println("The no of pixels:"+noofpixel);
        color=value%green;
        System.out.println("The color:"+color);
```

```
                Ncoloum=noofpixel+col+1;
                System.out.println("The Nthcoloum:"+Ncoloum);


                }
        }


int p[];
int p1[];
int ps[][];
int ps1[][];



p=new int[w*h];
p1=new int[w*h];
ps =new int[w][h];
ps1 =new int[w][h];

//pixel2= new double[w][h];
p1=img.getRGB(0,0,w,h,p,0,w);
ps[row][Ncoloum]=p[row*Ncoloum];
red=(ps[row][Ncoloum]>>16)&0xFF;
green=(ps[row][Ncoloum]>>8)&0xFF;
blue=(ps[row][Ncoloum])&0xFF;

 System.out.println(ps);
 System.out.println("The value of red:"+red);
 System.out.println("The value of blue:"+blue);
System.out.println("The value of green:"+green);
int b[]= new int[3];
b[0]=red;
b[1]=green;
b[2]=blue;
        if(b[0]>b[2])
        {
          if(b[0]>b[1])
                System.out.println("The  second highest RGB value is red
                :"+b[0]);
                t=b[0];
                b[0]=color;
                color=t;
                System.out.println("Replaced value red:"+b[0]);
        //      System.out.println("Replaced value color:"+color);
           }
           else
                {
                        if(b[2]>b[1])
                        {
                  System.out.println("The second highest RGB value is
                  blue:"+b[2]);
                  t=b[2];
                  b[2]=color;
                  color=t;
                  System.out.println("Replaced value blue:"+b[2]);
                  //System.out.println("Replaced value color:"+color);
```

```
            }
            Else
            {
            System.out.println("The second highest RGB value is
             green:"+b[1]);
            t=b[1];
            b[1]=color;
            color=t;
            System.out.println("Replaced value green:"+b[1]);
            //System.out.println("Replaced value color:"+color);
            }
        }
System.out.println("The final red value:"+b[0]);
System.out.println("The final green value:"+b[1]);
System.out.println("The final blue value:"+b[2]);

int[] OneDimPixel;
int[]  pixel1;
int[][] TwoDimPixel;
int[][][] ThreeDimPixel;
int[] oneD;


OneDimPixel=new int[w*h];
ThreeDimPixel=new int[w][h][4];
pixel1=new int[w*h];
TwoDimPixel=new int[w][h];
```

```
oneD=new int[w*h];
pixel1=img.getRGB(0,0,w,h,OneDimPixel,0,w);


int counter=0;


        for(int i=0;i<w;i++)
        {
                for(int j=0;j<h;j++)
                {
                        TwoDimPixel[i][j]=OneDimPixel[counter];
                        counter++;


                 }
         }
         ThreeDimPixel=convertToThreeD( OneDimPixel,w,h);
   for(int i=0;i<25;i++)
   {


         ThreeDimPixel[row][Ncoloum][1]=b[0];
         ThreeDimPixel[row][Ncoloum][2]=b[1];
         ThreeDimPixel[row][Ncoloum][3]=b[2];

    }

          oneD=convertToOneDim(ThreeDimPixel,w,h);
          System.out.println("Three Dim Array is"+ThreeDimPixel);
          System.out.println("one Dim Array is"+oneD);
          try
    {

    BufferedImage b1=new
     BufferedImage(w,h,BufferedImage.TYPE_INT_RGB);
     b1.setRGB(0,0,w,h,oneD, 0, w);
     File outfile = new File("image1.jpg");
      ImageIO.write(b1,"jpg", outfile);

img1=ImageIO.read(new File("image1.jpg"));
int h1=img.getHeight();
int w1=img.getWidth();
int[] pixelp=new int[w*h];
int[] pixel1p=new int[w*h];
int[][] pixelsp =new int[w][h];
int[][] pixels1p =new int[w][h];
pixel1p=img1.getRGB(0,0,w1,h1,pixelp,0,w1);
System.out.println("The height of the image:"+h1);
System.out.println("The width of the image:"+w1);
System.out.println("The rgb value:"+pixel1p);


//int noofpixel1;
//int color1,Ncoloum1,t1;
pixelsp[row][Ncoloum]=pixelp[row*Ncoloum];
red1=(pixelsp[row][Ncoloum]>>16)&0xFF;
green1=(pixelsp[row][Ncoloum]>>8)&0xFF;
blue1=(pixelsp[row][Ncoloum])&0xFF;
```

```
System.out.println("The value of red1:"+red1);
System.out.println("The value of blue1:"+blue1);
System.out.println("The value of green1:"+green1);
    }
    catch(Exception e)
    {
    }
    int rr=b[0];
    int gg=b[1];
    int bb=b[2];
    System.out.println(row+":"+Ncoloum+":"+rr+":"+gg+":"+bb);
    compare c=new compare(w,h,oneD,row,Ncoloum,red1,green1,blue1);
    c.show();
    //this.hide();

 }
 int[][][] convertToThreeD(int[] oneDPix,int imgCols,int imgRows)
 {
   int[][][] data =new int[imgRows][imgCols][4];
   for(int row = 0;row < imgRows;row++)
     {
      int[] aRow = new int[imgCols];
      for(int col = 0; col < imgCols;col++)
      {
        int element = row * imgCols + col;
        aRow[col] = oneDPix[element];
}
 //end for loop on col
for(int col = 0;col < imgCols;col++)
 {
  //Alpha data
  data[row][col][0] = (aRow[col] >> 24)& 0xFF;
  //Red data
  data[row][col][1] = (aRow[col] >> 16)& 0xFF;
  //Green data
  data[row][col][2] = (aRow[col] >> 8)& 0xFF;
  //Blue data
  data[row][col][3] = (aRow[col])& 0xFF;
 }
 //end for loop on col
}
//end for loop on row
return data;
}
int[] convertToOneDim(int[][][] data,int imgCols,int imgRows)
{
 //Create the 1D array of type int to be
 // populated with pixel data, one int value
 // per pixel, with four color and alpha bytes
 // per int value.
 int[] oneDPix = new int[imgCols * imgRows * 4];

 //Move the data into the 1D array.  Note the
 // use of the bitwise OR operator and the
 // bitwise left-shift operators to put the
 // four 8-bit bytes into each int.
 for(int row = 0,cnt = 0;row < imgRows;row++)
 {
  for(int col = 0;col < imgCols;col++)
  {
   oneDPix[cnt] = ((data[row][col][0] << 24)& 0xFF000000)|
((data[row][col][1] <<
   16)& 0x00FF0000)| ((data[row][col][2] << 8) & 0x0000FF00)
|((data[row][col][3])&
   0x000000FF);
    cnt++;
   }//end for loop on col


  }//end for loop on row
  return oneDPix;
 }
}
```

COMPARE:

```
import javax.swing.*;
import java.io.*;
import java.awt.MediaTracker.*;
import java.awt.image.*;
import java.awt.*;
import javax.imageio.*;
import java.util.*;
import java.lang.Math.*;
public class compare extends javax.swing.JFrame {
static ImageIcon outputimage;
JFileChooser filechooser;
String f1=new String();
BufferedImage img;
BufferedImage img1;
int height,width;

int pixel[];
int pixel1[];
int pixels[][];
int row,Ncoloum;
int rr,gg,bb;

/** Creates new form compare */
   public compare(int w,int h,int[] oneD,int row1,int Ncoloum1,int rr1,int
gg1,int bb1)
    {
          System.out.println(w);
         row=row1;
         System.out.println(row);
         Ncoloum=Ncoloum1;
         System.out.println(Ncoloum);
         rr=rr1;
         System.out.println(rr);
    gg=gg1;
    System.out.println(gg);
    bb=bb1;
    System.out.println(bb);
    try
    {
     BufferedImage b1=new
   BufferedImage(w,h,BufferedImage.TYPE_INT_RGB);
  b1.setRGB(0,0,w,h,oneD, 0, w);
  outputimage=new ImageIcon(b1);

  filechooser=new JFileChooser();
  filechooser.setFileSelectionMode(JFileChooser.FILES_ONLY);
  initComponents();
  }
  catch(Exception e)
  {
  }
  }
/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
private void initComponents() {//GEN-BEGIN:initComponents
    java.awt.GridBagConstraints gridBagConstraints;
```

```
jPanel1 = new javax.swing.JPanel();
jLabel1 = new javax.swing.JLabel();
jPanel2 = new javax.swing.JPanel();
jButton1 = new javax.swing.JButton();
jButton2 = new javax.swing.JButton();
jButton3 = new javax.swing.JButton();
jPanel3 = new javax.swing.JPanel();
jLabel2 = new javax.swing.JLabel();
jLabel3 = new javax.swing.JLabel();

addWindowListener(new java.awt.event.WindowAdapter() {
   public void windowClosing(java.awt.event.WindowEvent evt) {
      exitForm(evt);
   }
});

jPanel1.setBackground(new java.awt.Color(204, 204, 255));
jLabel1.setFont(new java.awt.Font("MS Sans Serif", 3, 14));
jLabel1.setText("Authentication Checking");
jPanel1.add(jLabel1);

getContentPane().add(jPanel1, java.awt.BorderLayout.NORTH);

jPanel2.setLayout(new java.awt.GridBagLayout());

jPanel2.setBackground(new java.awt.Color(204, 204, 255));
jButton1.setFont(new java.awt.Font("MS Sans Serif", 3, 11));
```

```
jButton1.setText("Select a Photo");
jButton1.addActionListener(new java.awt.event.ActionListener() {
  public void actionPerformed(java.awt.event.ActionEvent evt) {
    jButton1ActionPerformed(evt);
  }
});

gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.insets = new java.awt.Insets(0, 23, 0, 23);
jPanel2.add(jButton1, gridBagConstraints);

jButton2.setFont(new java.awt.Font("MS Sans Serif", 3, 11));
jButton2.setText("Compare");
jButton2.addActionListener(new java.awt.event.ActionListener() {
  public void actionPerformed(java.awt.event.ActionEvent evt) {
    jButton2ActionPerformed(evt);
  }
});

gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.insets = new java.awt.Insets(0, 23, 0, 23);
jPanel2.add(jButton2, gridBagConstraints);

jButton3.setFont(new java.awt.Font("MS Sans Serif", 3, 11));
jButton3.setText("Back");
jButton3.addActionListener(new java.awt.event.ActionListener() {
  public void actionPerformed(java.awt.event.ActionEvent evt) {
          jButton3ActionPerformed(evt);
        }
      });

      gridBagConstraints = new java.awt.GridBagConstraints();
      gridBagConstraints.insets = new java.awt.Insets(0, 23, 0, 23);
      jPanel2.add(jButton3, gridBagConstraints);

      getContentPane().add(jPanel2, java.awt.BorderLayout.SOUTH);

      jPanel3.setLayout(new java.awt.GridLayout());

      jLabel2.setBackground(new java.awt.Color(204, 204, 255));
      jLabel2.setBorder(new

javax.swing.border.BevelBorder(javax.swing.border.BevelBorder.RAISED)
);
      jLabel2.setPreferredSize(new java.awt.Dimension(150, 150));
      jPanel3.add(jLabel2);
      jLabel2.setIcon(outputimage);

      jLabel3.setBackground(new java.awt.Color(204, 204, 255));
      jLabel3.setBorder(new

javax.swing.border.BevelBorder(javax.swing.border.BevelBorder.RAISED)
);
      jLabel3.setPreferredSize(new java.awt.Dimension(150, 150));
```

```
    jPanel3.add(jLabel3);

    getContentPane().add(jPanel3, java.awt.BorderLayout.CENTER);

    pack();
  }//GEN-END:initComponents

   private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
   {//GEN-FIRST:event_jButton3ActionPerformed
     // TODO add your handling code here:
    this.hide();
   }//GEN-LAST:event_jButton3ActionPerformed

  private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
  {//GEN-FIRST:event_jButton2ActionPerformed
     try
{
img=ImageIO.read(new File(f1));
height=img.getHeight();
width=img.getWidth();
pixel=new int[width*height];
pixel1=new int[width*height];
pixels =new int[width][height];
pixel1=img.getRGB(0,0,width,height,pixel,0,width);

 }
 catch(Exception e)
```

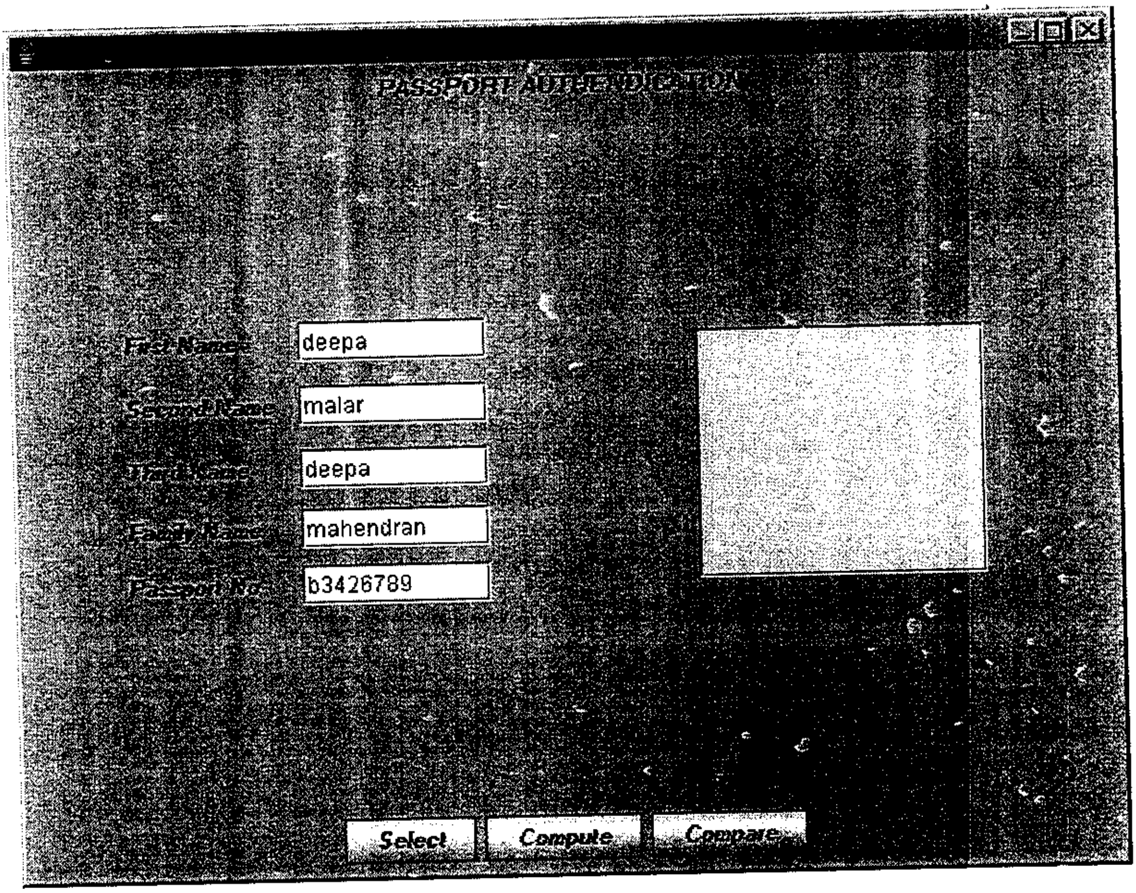
```
{
}
int red=0;
int blue=0;
int green=0;

        pixels[row][Ncoloum]=pixel[row*Ncoloum];
        red=(pixels[row][Ncoloum]>>16)&0xFF;
        green=(pixels[row][Ncoloum]>>8)&0xFF;
        blue=(pixels[row][Ncoloum])&0xFF;
        System.out.println("row is"+row);
        System.out.println("coloumn value is"+Ncoloum);
        System.out.println("red value is"+red);
        System.out.println("Green value is"+green);
        System.out.println("blue value is"+blue);
        System.out.println("red value is"+rr);
        System.out.println("Green value is"+gg);
        System.out.println("blue value is"+bb);
        if(red==rr)
      {
        if(green==gg)
        {

        }if(blue==bb)

        JOptionPane.showMessageDialog(this,"IMAGE
        MATCHED","Error",JOptionPane.ERROR_MESSAGE);
```

```
    }
    else
    {

     JOptionPane.showMessageDialog(this,"IMAGE NOT
      MATCHED","Error",JOptionPane.ERROR_MESSAGE);
  }
}

        // TODO add your handling code here:
 //GEN-LAST:event_jButton2ActionPerformed

 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
 {//GEN-FIRST:event_jButton1ActionPerformed
 if(evt.getSource()==jButton1)
   {
   filechooser.showOpenDialog(this);
   f1=filechooser.getSelectedFile().getAbsolutePath();
   System.out.println(f1);
   ImageIcon im=new ImageIcon(f1);
    jLabel3.setIcon(im);
    }

    // TODO add your handling code here:
  }//GEN-LAST:event_jButton1ActionPerformed

  /** Exit the Application */
```

```
private void exitForm(java.awt.event.WindowEvent evt)
{//GEN-FIRST:event_exitForm
   System.exit(0);
}//GEN-LAST:event_exitForm

/**
 * @param args the command line arguments
 */
//public static void main(String args[]) {
   // new compare().show();
//}

// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
// End of variables declaration//GEN-END:variables

}
```

6.2 SCREENSHOT:

1. GUI DESIGN:



PASSPORT AUTHENDICATION

First Name: deepa

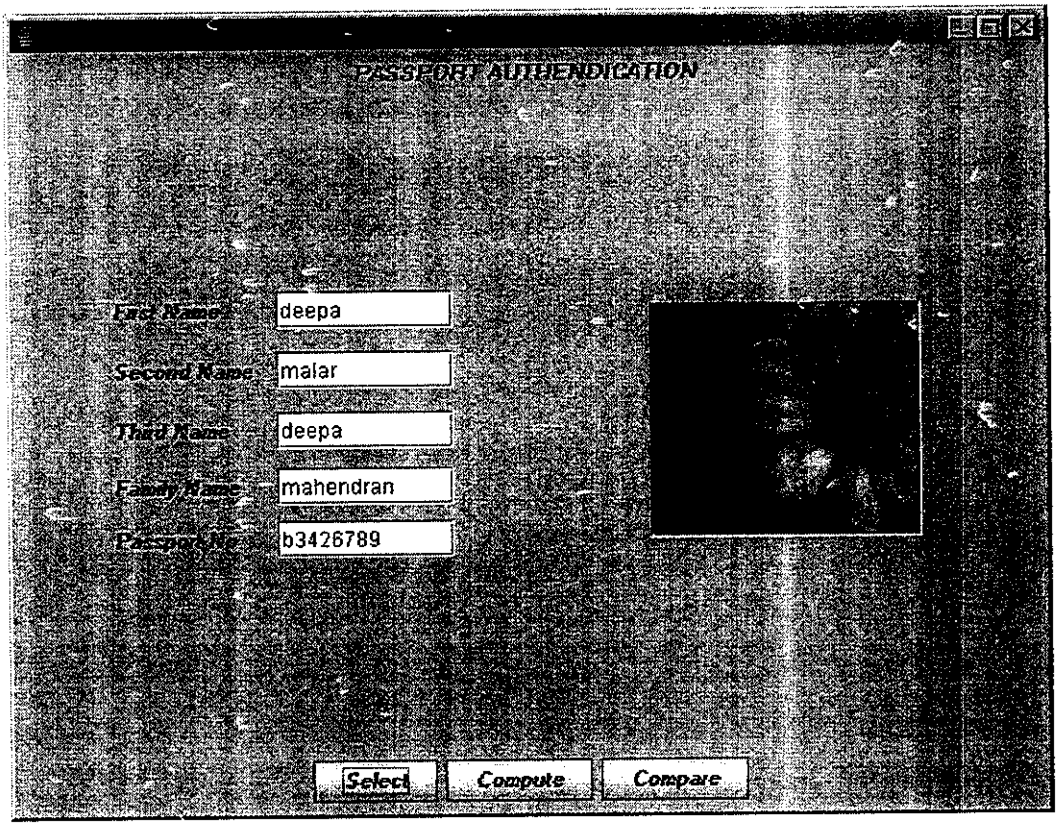
Second Name: malar

Third Name: deepa

Family Name: mahendran

Passport No: b3426789

2. SELECTING AN IMAGE:



PASSPORT AUTHENDICATION

First Name: deepa

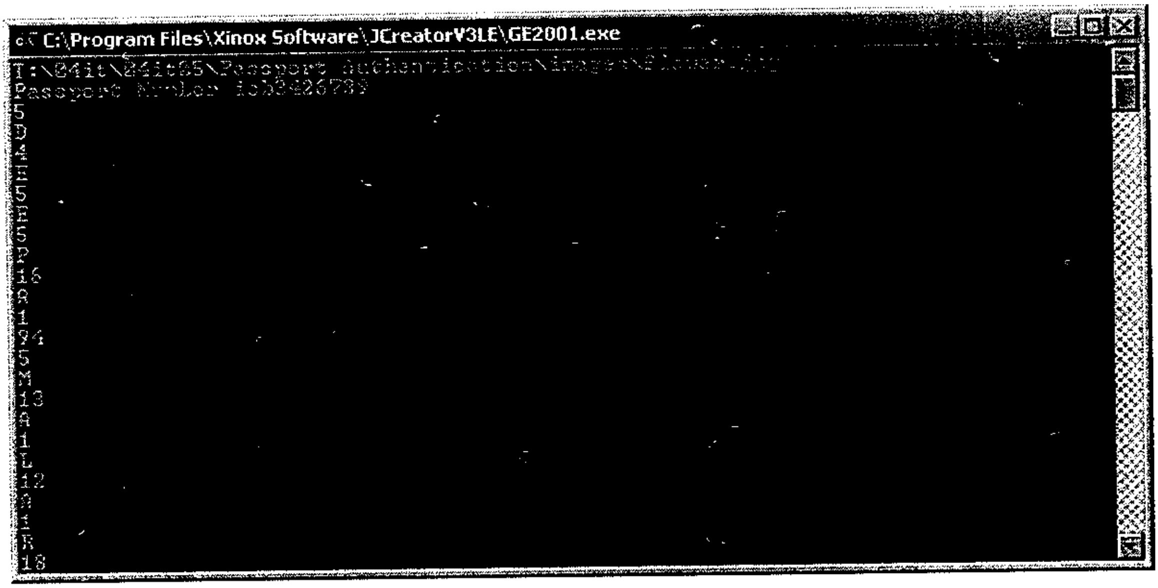
Second Name: malar

Third Name: deepa

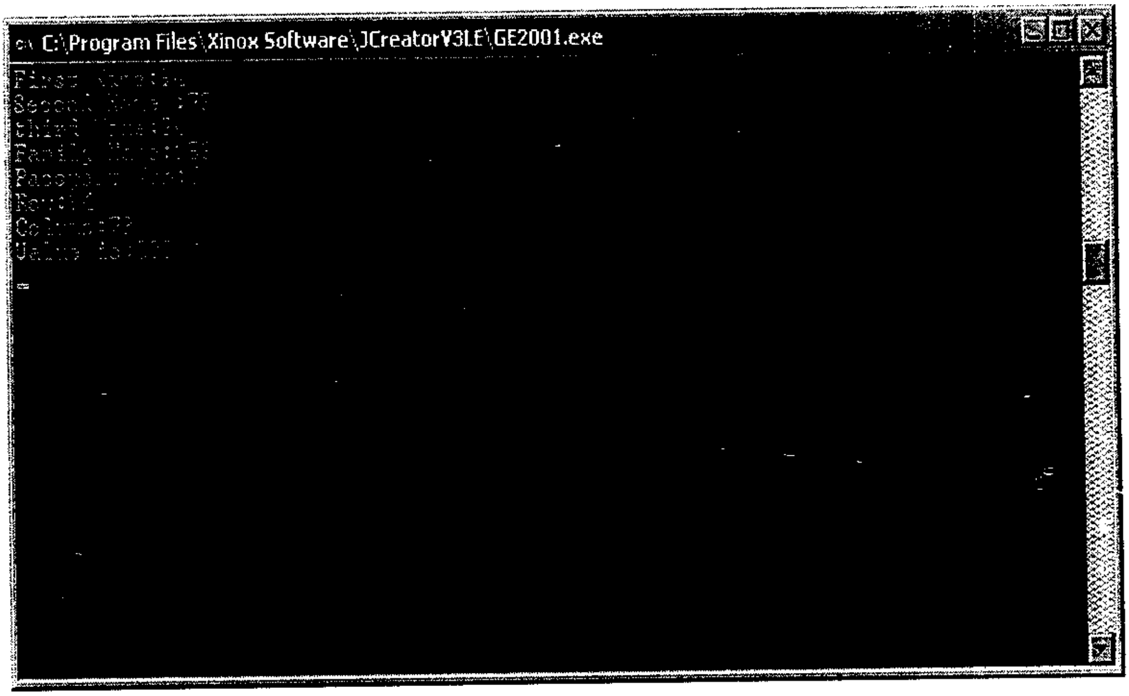
Family Name: mahendran

Passport No: b3426789

3. COMPUTATION:

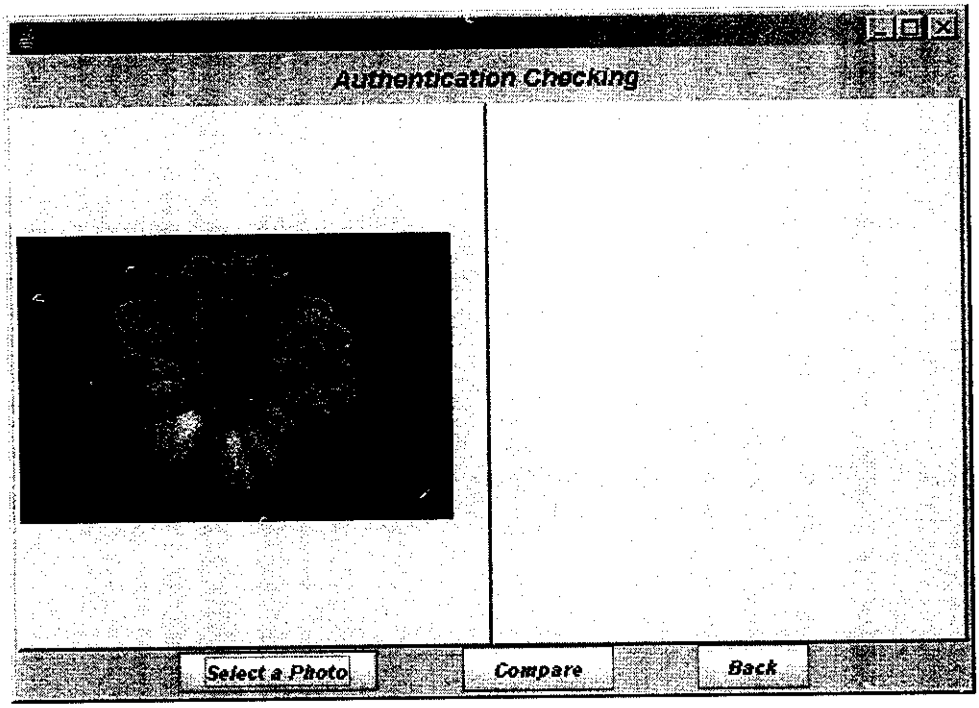


```
T:\04it\04it05\Passport Authentication\images\[illegible]
Passport Number is b3426739
5
D
4
H
5
H
5
P
16
R
1
24
5
M
13
A
1
L
12
R
1
R
18
```

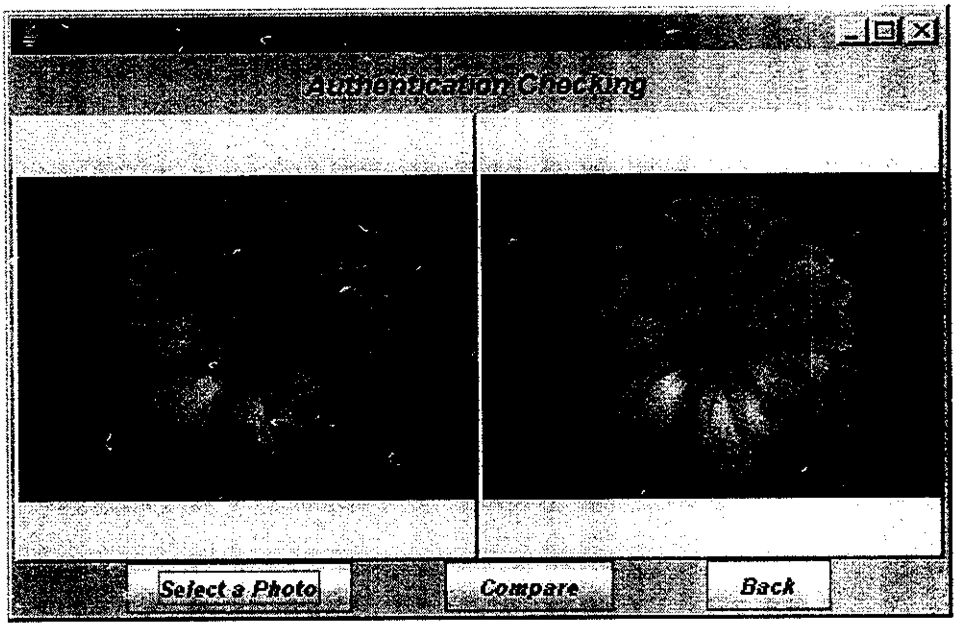


```
First [illegible]
Second [illegible]77
third [illegible]3
Family [illegible]5
Passp[illegible]
Row:2
Column:72
Value is[illegible]
```

4. AUTHENTICATION CHECKING:



5. COMPARE:



6. RESULT:

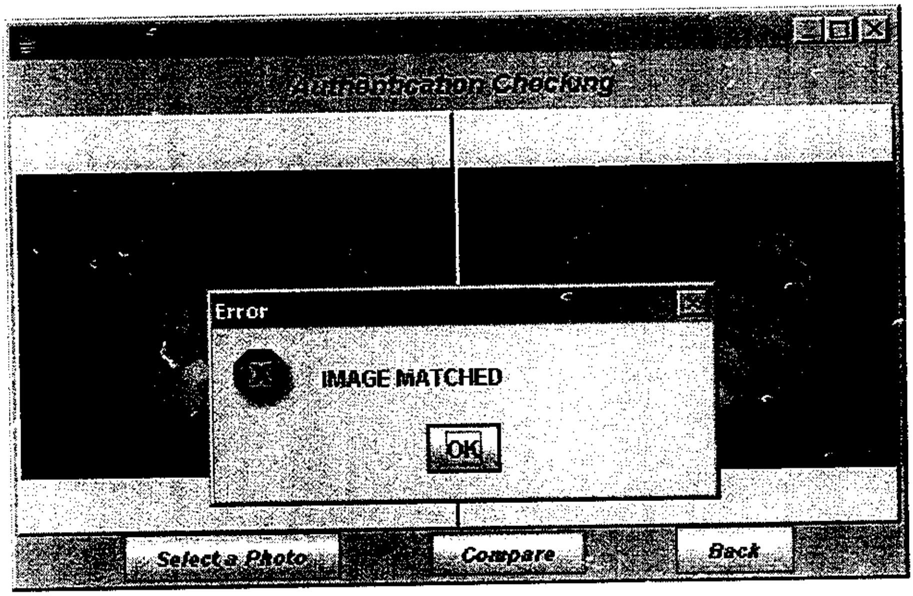


IMAGE MATCHED

REFERNCES

7.REFERENCES

[1] Neil, F. Johson, Zoran Duric and Sushil Jajodia ,"Information Hiding: Steganography and watermarking attacks and countermeasures",Kluwer Academic publishers,2001.

[2] Al-Hamami, Alaa and Soukaina Hassan ,"A proposed Firewall Security against different types of attacks", Al-Rafidain Magazine for
Science, No. 14, Baghdad, Iraq,2003.

[3]. Richard, Clark, "An introduction to JPEG 2000 and Watermarking",© Elysium Ltd.

[4] Stefan, K. and F.A.P. Petitcolas ,"Information Hiding techniques for steganography and digital watermarking", ISBN 1-58053-035-4 © Artech House, Inc,Edition 2000.

WEBSITES:

[1] http://www.jcreator.com/,on 15th,Dec 2007

[2] http://www.webopedia.com/

[3] http://java.sun.com/