A NEW APPROACH FOR PASSPORT AUTHENTICATION

A PROJECT REPORT

Submitted by

R.V.ACHYUDHAN	71205104301
K.VIVEK KUMAR	71205104307

in partition fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

ANNA UNIVERSITY: CHENNAI-600 025

MAY 2009

BONAFIDE CERTIFICATE

Certified that this project report entitled "**A NEW APPROACH FOR PASSPORT AUTHENTICATION**" is the bonafide work of R.V.Achyudhan and K.Vivek kumar who carried out the research under my supervision. Certified also, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr. S. Thangasamy ,

HEAD OF THE DEPARTMENT ,

Department of Computer Science
& Engineering,
Kumaraguru College of Technology,
Coimbatore-641006.

P. Malini

SIGNATURE

Ms. P. Malini,

SUPERVISOR,

Lecturer,

Department of Computer Science
& Engineering,
Kumaraguru College of Technology,
Coimbatore-641006.

The candidate with University Register Nos. 71205104301, 71205104307 were examined by us in the project viva-voce examination held on 28-4-2009

INTERNAL EXAMINER

EXTERNAL EXAMINER

DECLARATION:

We hereby declare that the project entitled ” **A NEW APPROACH FOR PASSPORT AUTHENTICATION**” is a record of original work done by us and to the best of our knowledge, a similar work has not been submitted to Anna University or any Institutions, for fulfillment of the requirement of the course study.

The report is submitted in partial fulfillment of the requirement for the award of the Degree of Bachelor of Computer Science and Engineering of Anna University, Chennai.

Place: Coimbatore

Date : 28-4-2009

R.V. Achyudhan

R.V.Achyudhan

K.Vivek kumar

ACKNOWLEDGEMENT

We extend our sincere thanks to our principal, **Dr. Joseph V.Thanikal, Phd.**, Principal, Kumaraguru College Of Technology, Coimbatore, for being a constant source of inspiration and providing us with the necessary facility to work on this project.

We would like to make a special acknowledgement and thanks to **Dr. S. Thangasamy, Ph.D.**, Dean, Professor and Head of Department of Computer Science &Engineering, for his support and encouragement throughout the project.

We express deep gratitude and gratefulness to our guide **Mrs.P.Malini (M.E)** Department of Computer Science &Engineering, for her supervision, enduring patience, active involvement and guidance.

We would also like to thank our project coordinator, **Mrs. Devaki** for her support during the course of our project.

We would like to convey our honest thanks to all members of staff of the Department for their enthusiasm and wealth of experience from which we have greatly benefited.

DEDICATED

TO MY

BELOVED PARENTS

RESPECTFUL STAFF

&

LOVABLE FRIENDS

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.

LIST OF FIGURES AND TABLES

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENTS

1. INTRODUCTION

1.1 INTRODUCTION

Steganography provides for the embedding of information in a block of host data in conditions where perceptible modification of the host data is intolerable. Steganographic techniques are highly dependent on the character of the host data.

A technique for embedding information in image makes subtle changes in hue, while a method for embedding information in audio data could exploit the limitations of the human ear by encoding the encapsulated information in audible frequency ranges. Current implementations of textual Steganography exploit tolerances in type setting by making minute changes in line placement and kerning in order to encapsulate hidden information, making them vulnerable to simple retype setting attacks. Manipulating digital media in an effort to disable or remove the embedded messages is a simpler task than detecting the messages.

Any image can be manipulated with the intent of destroying some hidden information whether an embedded message exists or not. Detecting the existence of a hidden message will save time in the activity to disable or remove messages by guiding the analyst to process only the media that contain hidden information .Because the most successful hiding method is the uncommon one, but it is unthinkable one and needs a knowledge and experience to be discovered.

Every few years, computer security has to re-invert itself. New technologies and new applications bring new threats, and force us to invent new protection mechanisms.

1.2 Existing Problem:

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

- 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.

- This concept is based on the properties of the human visual system or the human audio system. The embedded information is imperceptible if an average human subject is unable to distinguish between carries that contain the hidden information and those that do not.

- The concept of Undetectability is inherently tied to the statistical model of the image source. If an attacker has a more detailed model of the source, he may be able to detect the presence of a hidden image, but this does not imply the ability to read the hidden message.

- 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 (except the secret key). Nowadays, computer involved in all life details. One of these issues is producing the passport document by using the computer application. To fill full this objective there are several requirements such as using a computerize photo for the passport holder with special colors (grey and white). Also most of the passport Offices are connected through a network to exchange their information about the correctness of passport information and the authentication of the passport holder. It is possible to transfer the passport image between different offices to get information confirmation. Here, the main problem is how to confirm and authenticate the passport's photo with the information about the holder. There is no other way to tell if the photo been replaced with a new one (for the current holder) because there is no physical evidence.

1.3 Proposed Method

The aim of the proposed method is to develop a firm connection between the passport's photo and the passport's details. In this case it is possible to use this method for confirmation of passport's information. The summery of this method is by converting the holder's name (1st, 2nd, 3rd and family name) in addition to the passport number into one form called an invisible

Watermark. This watermark will be disguised and distributed inside the passport's photo. This process will be done during the issue of the passport for the first time. Note that all the watermark requirements will be considered. The proposed method consists of several algorithms.

Each algorithm is responsible for one type of process. All the required validations processes will be taken in consideration by the proposed method.

1.3.1.3 Hiding Technique

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.

1.3.1Algorithms

1.3.1.1 Parameters Acquisition Technique

1. Read first, second, third and family name.
2. Read Passport Number.
3. Validate entries.
4. Assign each letter a number according to a table.
5. Keep each name's numbers.

1.3.1.2 Converting Technique

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".

1.4 About the Implementation

The project has been implemented using java .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.

1.4.1 Language Description

- **Java Platform Architecture**

The java platform is unique in that it's a software-only platform that runs on top of other hardware-based platforms. Hence, the java platform lets to develop and deploy java applications on desktops and servers, as well as today's demanding embedded and real-time environments. Java also includes classes that support the development of web services and provides the foundation for enterprise development.

- **The Java Programming Language**

Java programming language is syntactically similar to C++, but differs in execution. While C++ uses unsafe pointers and programmers are responsible for allocating and freeing memory, the java programming language uses type-safe object references, and unused memory is reclaimed automatically.

Further more, the java programming language eschews multiple inheritances-a likely source of confusion and ambiguity in C++ in favor of a cleaner construct, interfaces. Java doc is a tool for generating API documentation in HTML format from documentation comments in source code.

- **The Java Virtual Machine(JVM)**

The java virtual machine forms the foundation of the java platform. This architecture offers several attractive features: the virtual machine can be implemented to run a top a variety of operating systems and hardware, with binary-compatible java applications operating consistently across many implementations.

- **Java Runtime Environment**

The JRE provides java API's, java virtual machine, and other components necessary to run applets and applications written in the java programming language. It is also the foundation for the technologies in the Java Platform, Enterprise Edition (Java EE).The JRE does not contain tools and utilities such as compilers or debuggers for developing applets and applications.

2. METHODOLOGY

2.1 STEGANOGRAPHY

Steganography is the art and science of writing hidden messages in such a way that no-one, apart from the sender and intended recipient, suspects the existence of the message, a form of security through obscurity. The word steganography is of Greek origin and means "concealed writing".

The advantage of steganography, over cryptography alone, is that messages do not attract attention to themselves. Plainly visible encrypted messages—no matter how unbreakable—will arouse suspicion, and may in themselves be incriminating in countries where encryption is illegal. Therefore, whereas cryptography protects the contents of a message, steganography can be said to protect both messages and communicating parties.

Steganography includes the concealment of information within computer files. In digital steganography, electronic communications may include steganographic coding inside of a transport layer, such as a document file, image file, program or protocol. Media files are ideal for steganographic transmission because of their large size. As a simple example, a sender might start with an innocuous image file and adjust the color of every 100th pixel to correspond to a letter in the alphabet, a change so subtle that someone not specifically looking for it is unlikely to notice it.

2.2 Digital Watermarking

Digital watermarking is the process of possibly irreversibly embedding information into a digital signal. The signal may be audio, pictures or video, for example. If the signal is copied, then the information is also carried in the copy.

In visible watermarking, the information is visible in the picture or video. Typically, the information is text or a logo which identifies the owner of the media. The image on the right has a visible watermark. When a television broadcaster adds its logo to the corner of transmitted video, this is also a visible watermark.

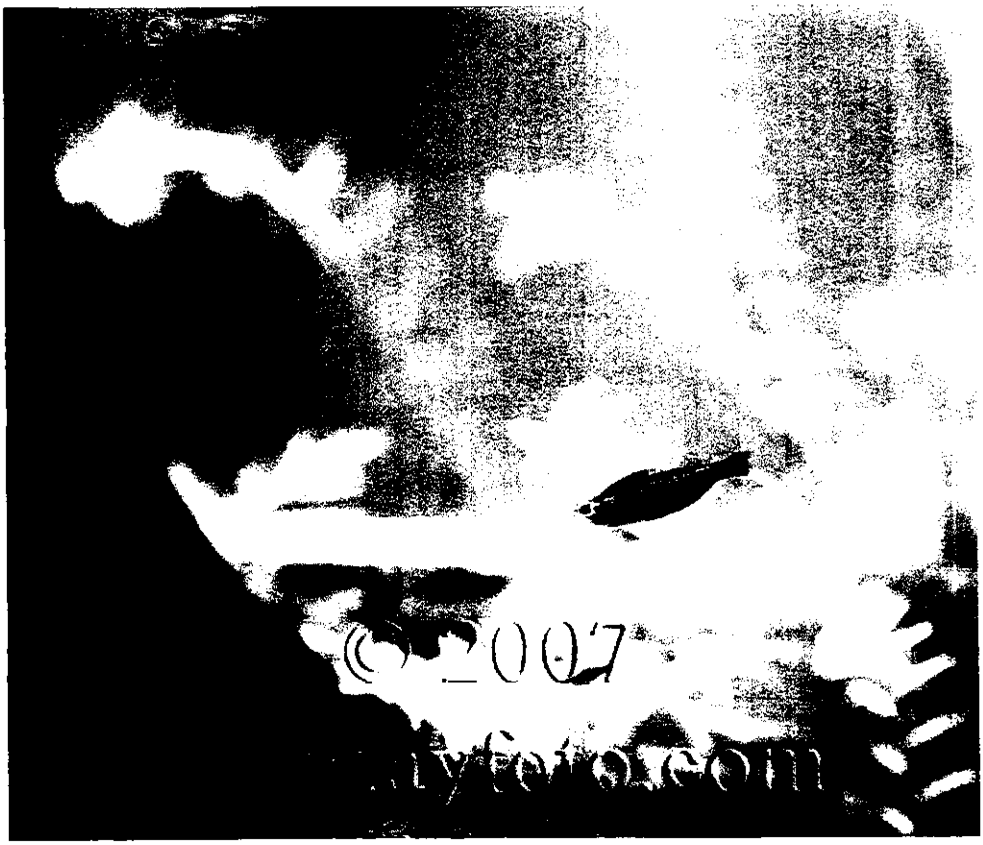


Figure 1 Visible watermarking

In invisible watermarking, information is added as digital data to audio, picture or video, but it cannot be perceived as such (although it is possible to detect the hidden information). An important application of invisible watermarking is to copyright protection systems, which are intended to prevent or deter unauthorized copying of digital media. Steganography is an application of digital watermarking, where two parties communicate a secret message embedded in the digital signal. Annotation of digital photographs with descriptive information is another application of invisible watermarking. While some file formats for digital media can contain additional information called metadata, digital watermarking is distinct in that the data is carried in the signal itself.

3. SYSTEM DESIGN

The following diagram explains how the calculations are done using the data to normalize.

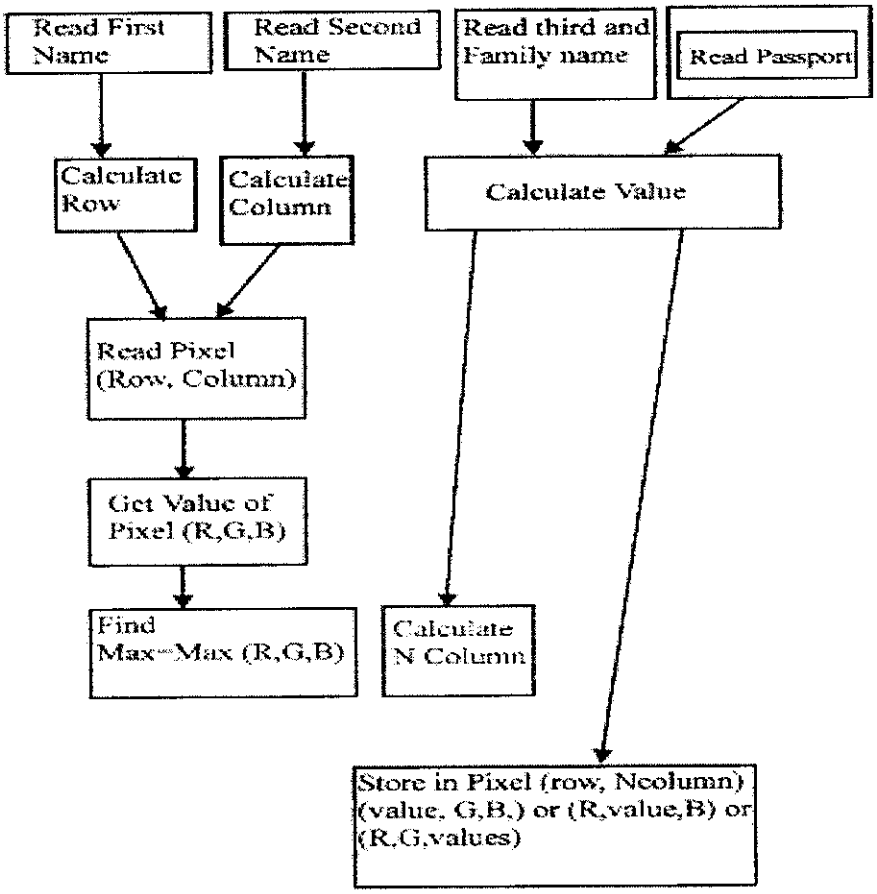


Figure 2 Method for Confirmation of Passport's Information

3.1 DATAFLOW DIAGRAM

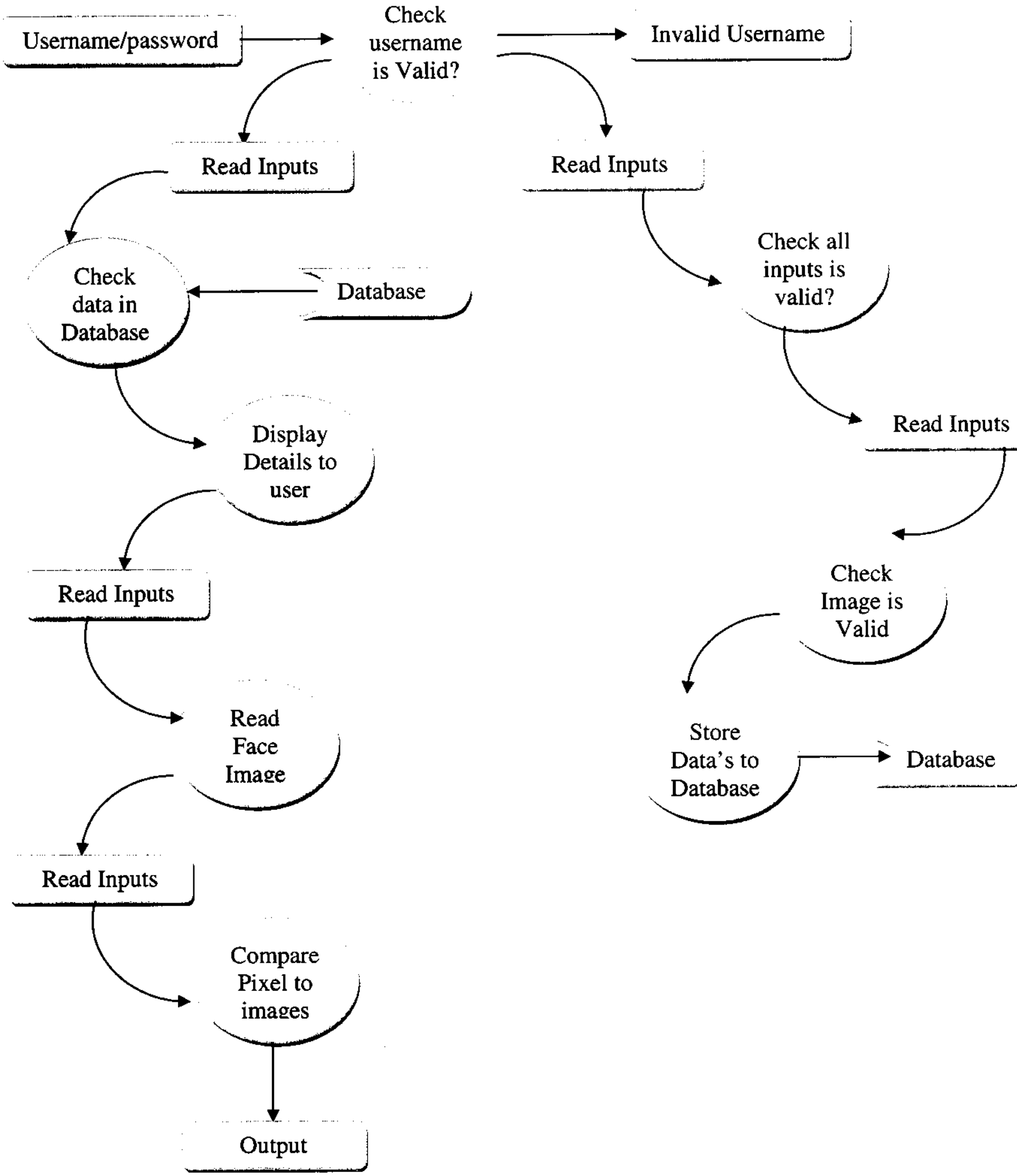


Figure 3: DFD

3.2 DATABASE DESIGN:

Registration

Field Name	Data types
First Name	Varchar(15)
Second Name	Varchar(15)
Third Name	Varchar(15)
Family Name	Varchar(15)
Passport No	Varchar(10)
Face Image Path	Varchar(100)

Table 3.1

Face Normalization

Field Name	Data types
Passport No	Varchar(10)
Row	Integer
Column	Integer
Value	Integer
Path Image	Varchar(100)

Table 3.2

4. SYSTEM ENVIRONMENT:

4.1 HARDWARE CONFIGURATION:

Processor	:	Pentium IV
Processor Speed	:	2.80GHz
Main Storage	:	512MB RAM

4.2 SOFTWARE CONFIGURATION

Operating System	:	Windows XP
Front end	:	JAVA(Jdk 1.5)
Back end	:	MYSQL

5. TESTING

Testing is a series of different tests that whose primary purpose is to fully exercise the computer based system. Although each test has a different purpose, all work should verify that all system element have been properly integrated and performed allocated function. Testing is the process of checking whether the developed system works according to the actual requirement and objectives of the system.

The philosophy behind testing is to find the errors. A good test is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers the undiscovered error. Test cases are devised with this purpose in mind. A test case is a set of data that the system will process as an input. However the data are created with the intent of determining whether the system will process them correctly without any errors to produce the required output.

Types of Testing:

- Unit testing
- Integration testing
- Validation testing
- Output testing
- User acceptance testing

5.1 Unit Testing

All modules were tested and individually as soon as they were completed and were checked for their correct functionality.

5.2 Integration Testing

The entire project was split into small program; each of this single programs gives a frame as an output. These programs were tested individually; at last all these programs where combined together by creating another program where all these constructors were used. It give a lot of problem by not functioning is an integrated manner.

The user interface testing is important since the user has to declare that the arrangements made in frames are convenient and it is satisfied. when the frames where given for the test, the end user gave suggestion. Based on their suggestions the frames where modified and put into practice.

5.3 Validation Testing

At the culmination of the black box testing software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of test i.e., Validation succeeds when the software function in a manner that can be reasonably accepted by the customer.

5.4 Output Testing

After performing the validation testing the next step is output testing of the proposed system. Since the system cannot be useful if it does not produce the required output. Asking the user about the format in which the system is required tests the output displayed or generated by the system under consideration. Here the output format is considered in two ways. one is on screen and another one is printed format. The output format on the screen is found to be corrected as the format was designed in the system phase according to the user needs. And for the hardcopy the output comes according to the specifications requested by the user.

5.5 User Acceptance System

An acceptance test as the objective of selling the user on validity and reliability of the system. It verifies that the procedures operate to system specification and mat the integrity of vital is maintained.

5.6 Performance Testing

This project is a application based project, and the modules are interdependent with the other modules, so the testing cannot be done module by module. So the unit testing is not possible in the case of this driver. So this system is checked only with their performance to check their quality.

. CONCLUSION

This project has suggested a new method for authentication. An invisible watermark has been suggested to authenticate the passport's holder. All the requirements for the watermark technique have been taken into account. 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. The security requirement is achieved by the random distribution of the watermark over the entire image, which makes the watermark detection is time consuming. To preserve the undetectability requirement, the invisible watermark has a small capacity comparing to the image size which is unnoticeable even if we use the statistical comparison for the images (before and after hiding). One additional factor has been used to make the watermark undetectable and that by changing the pixel color to be compatible with RGB color for the original image. 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.

7. APPENDIX

7.1 SAMPLE CODE

```
//Code for Login Form
import java.awt.Dimension;
import java.awt.Graphics;
import java.awt.Image;
import java.awt.Toolkit;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.JButton;
import javax.swing.JDialog;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;
import javax.swing.JTextField;
public class LoginForm extends JDialog implements ActionListener {
        static final long serialVersionUID = 102;
        JFrame parent;
        JLabel userLabel,passLabel;
        JButton btnSubmit,btnCancel;
        JTextField txtUName;JPasswordField passValue;
        int count = 0;
        JPanel panel;
```

```
Image img;
public LoginForm(JFrame par) {
        super(par,true);
        parent = par;
        toInitialize();
        setTitle("Login Form");
        setResizable(false);
        setUndecorated(true);
        setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE);
        Dimension size = Toolkit.getDefaultToolkit().getScreenSize();
        setBounds(((size.width - 285)/2),((size.height - 300)/2),285,140);
        setVisible(true);
 }
 public void toInitialize() {
        img = Toolkit.getDefaultToolkit().getImage("Images/page.jpg");
        panel = new JPanel() {
        private static final long serialVersionUID = 4321714013019129694L;
        protected void paintComponent(Graphics g)
        {
                        super.paintComponent(g);
                        g.drawImage(img,0,0,super.getWidth(),
                        super.getHeight(), this);
        }
        };
        userLabel = new JLabel("UserName");
        passLabel = new JLabel("Password");
        btnSubmit = new JButton("Submit");
```

```
    btnCancel = new JButton("Cancel");
    txtUName = new JTextField();
    passValue = new JPasswordField();
    panel.setLayout(null);
    panel.add(userLabel);
    panel.add(passLabel);
    panel.add(txtUName);
    panel.add(passValue);
    panel.add(btnSubmit);
    panel.add(btnCancel);
    getRootPane().setDefaultButton(btnSubmit);
    btnSubmit.addActionListener(this);
    btnCancel.addActionListener(this);
    btnCancel.addActionListener(this);
    userLabel.setBounds(30,20,70,20);
    passLabel.setBounds(30,55,70,20);
    txtUName.setBounds(110,20,150,20);
    passValue.setBounds(110,55,150,20);
    btnSubmit.setBounds(30,100,90,22);
    btnCancel.setBounds(160,100,90,22);

    //getContentPane().setLayout(null);
    getContentPane().add(panel);
    //panel.setBounds(8,10,280,150);

}

public void actionPerformed(ActionEvent ae) {
```

```
String str,str1;
if(ae.getActionCommand() == "Submit") {
        str = txtUName.getText();
        char ch[] =passValue.getPassword();
        str1 = new String(ch);

        if(str.equalsIgnoreCase("passport")&&
        str1.equalsIgnoreCase("passport")) {
                setVisible(false);


        }
        else {
                count++;
                if(count == 3) {
                JOptionPane.showMessageDialog(this,"Only
                        Three Attempt Press Ok to Exit");
                        System.exit(0);
                }

                JOptionPane.showMessageDialog(this,"Invalid
        UserName or Password");
                txtUName.setText("");
                passValue.setText("");
                txtUName.requestFocus();
        }
 }
 if(ae.getActionCommand() == "Cancel") {
```

```
                System.exit(0);
          }
     }

}

//Code for Normaization

import java.awt.Color;
import java.awt.Dimension;
import java.awt.Toolkit;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.FileInputStream;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JInternalFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
```

```
mport javax.swing.JTextField;
mport javax.swing.border.CompoundBorder;
mport javax.swing.border.EtchedBorder;
mport javax.swing.event.DocumentEvent;
mport javax.swing.event.DocumentListener;
ublic class FaceNormailzation extends JInternalFrame implements
ctionListener{
    private static final long serialVersionUID = -4524269887617605305L;
    JLabel labelPassportNo,labelFaceImage,labelImage;
    static JTextField
xtPassportNo,txtFName,txtLName,txtMName,txtFMName;
    JButton btnSubmit,btnCompute,btnDelete,btnClose,btnSense;
    static JTextArea txtOutput;
    JScrollPane scrollView;
    JLabel labelFName,labelLName,labelMName,labelFMName;
    int fval,sval,tval,lval,pval,row,column,value;
    String fname,sname,tname,lname,pnum,path;
    Connection con;
    Statement st;
    ResultSet rs;
    public FaceNormailzation() {
        setDBConnection();
        toInitialize();
        Dimension dim = Toolkit.getDefaultToolkit().getScreenSize();
        setTitle("Normalization Form");
        setBounds((dim.width-800)/2,(dim.height-600)/2,800,490);
        setVisible(true);
    }
    public void actionPerformed(ActionEvent ae) {
        if(ae.getActionCommand() == "Close") {
            setVisible(false);
        }
        if(ae.getActionCommand() == "Submit") {
            try {
                st = con.createStatement();
                rs = st.executeQuery("select * from registration where
passportNo = '"+txtPassportNo.getText()+"'");
                if(rs.next()) {
                    fname = rs.getString("FirstName");
                    sname = rs.getString("SecondName");
                    tname = rs.getString("ThridName");
                    lname = rs.getString("FamilyName");
                    pnum  = rs.getString("FirstName");
                    path  = rs.getString("FaceImagePath");
                    txtFName.setText(fname);
                    txtMName.setText(sname);
                    txtLName.setText(tname);
                    txtFMName.setText(lname);
                    labelImage.setIcon(new ImageIcon(path));
                    btnCompute.setEnabled(true);
                    btnDelete.setEnabled(true);
                }
                else {
```

```
    JOptionPane.showMessageDialog(PassportMain.deskView,   "Record   Not
ound");
                        }
                }catch(Exception e) {

    JOptionPane.showMessageDialog(PassportMain.deskView,"Error          in
Connecting"+e);
                }
            }

            if(ae.getActionCommand() == "Delete") {
                try {
                        st = con.createStatement();
                        String selectQuery = "select * from faceNormalization
where Passportno = '"+txtPassportNo.getText()+"'";
                        rs = st.executeQuery(selectQuery);
                        if(rs.next()) {
                                String    deleteQuery    =    "delete    from
faceNormalization where Passportno = '"+txtPassportNo.getText()+"'";
                                int                option                =
JOptionPane.showConfirmDialog(PassportMain.deskView,   "Sure   U   Want
Delete?");
                                if(option == JOptionPane.YES_OPTION) {
                                        st.executeUpdate(deleteQuery);
    JOptionPane.showMessageDialog(PassportMain.deskView,"Successfully
Deleted");
                        }
                        else;
                    }
                    else {

    JOptionPane.showMessageDialog(PassportMain.deskView,    "Record    Not
Found");
                    }
                }catch (Exception e) {
                    String str[] = e.toString().split(":");
                    String data = "";
                    for(int i = 1;i < str.length;i++) {
                        data += str[i]+"\n";
                    }

    JOptionPane.showMessageDialog(PassportMain.deskView,data);
                }
            }

            if(ae.getActionCommand() == "...") {
                //new SenseNormal();
            }
            if(ae.getActionCommand() == "Compute") {
```

```
f((fname.equals(""))||(sname.equals(""))||(tname.equals(""))||(lname.equals(""))||(p
um.equals(""))){
                JOptionPane.showMessageDialog(null, "Values should not be
ull", "Message", JOptionPane.ERROR_MESSAGE);
            }
            else {
                ConvertAlgorithm conAlg = new ConvertAlgorithm();
                fval=conAlg.calculateValue(fname);
                sval=conAlg.calculateValue(sname);
                tval=conAlg.calculateValue(tname);
                lval=conAlg.calculateValue(lname);
                pval=conAlg.calculateValue(pnum);
                FaceNormailzation.txtOutput.append("\nFirst Name:"+fval);
                FaceNormailzation.txtOutput.append("\nSecond Name :"+sval);
                FaceNormailzation.txtOutput.append("\nthird Name:"+tval);
                FaceNormailzation.txtOutput.append("\nFamily Name:"+lval);
                FaceNormailzation.txtOutput.append("\nPassport Num:"+pval);
                      row=fval;
                      column=sval;
                      value=tval+lval+pval;
                      FaceNormailzation.txtOutput.append("\nRow:"+row);

                 FaceNormailzation.txtOutput.append("\nColumn:"+column);
                 FaceNormailzation.txtOutput.append("\nValue is:"+value);
                      try {
                           st = con.createStatement();
```

```
                    String      insertQuery .    =     "insert     into
                    faceNormalization values('"+
                    txtPassportNo.getText()+"'," +
                    row+"," +
                    column+"," +
                    value+",'" +
                    path.replaceAll("\\\\","/")+"')";
                    st.executeUpdate(insertQuery);
    JOptionPane.showMessageDialog(PassportMain.deskView,"Successfully
Stored");
                }catch (Exception e) {
        JOptionPane.showMessageDialog(PassportMain.deskView,"Error   in
Connecting"+e);
                }


            }


        }


    }

    private void setDBConnection() {
        try {
            FileInputStream          stream          =          new
FileInputStream("Database.config");
            byte b[] = new byte[stream.available()];
            stream.read(b);
```

```
                String str = new String(b);
                String server[] = str.split(":");
                Class.forName("com.mysql.jdbc.Driver").newInstance();
          con                                                                          =
DriverManager.getConnection("jdbc:mysql://"+server[0]+":3306/"+server[1],serve
[2],server[3]);

          }catch (Exception e) {

     JOptionPane.showMessageDialog(PassportMain.deskView,"Error              in
Connecting"+e);
          }
     }
     private void toInitialize() {
          JPanel panel = new JPanel();
          JPanel panel1 = new JPanel();
          labelPassportNo = new JLabel("Passport No");
          labelFName = new JLabel("FirstName");
          labelLName = new JLabel("LastName");
          labelMName = new JLabel("MiddleName");
          labelFMName = new JLabel("FamilyName");
          txtPassportNo = new JTextField();
          //txtPassportNo.setFocusable(false);
          txtFName = new JTextField();
          txtFName.setFocusable(false);
          txtLName = new JTextField();
          txtLName.setFocusable(false);
```

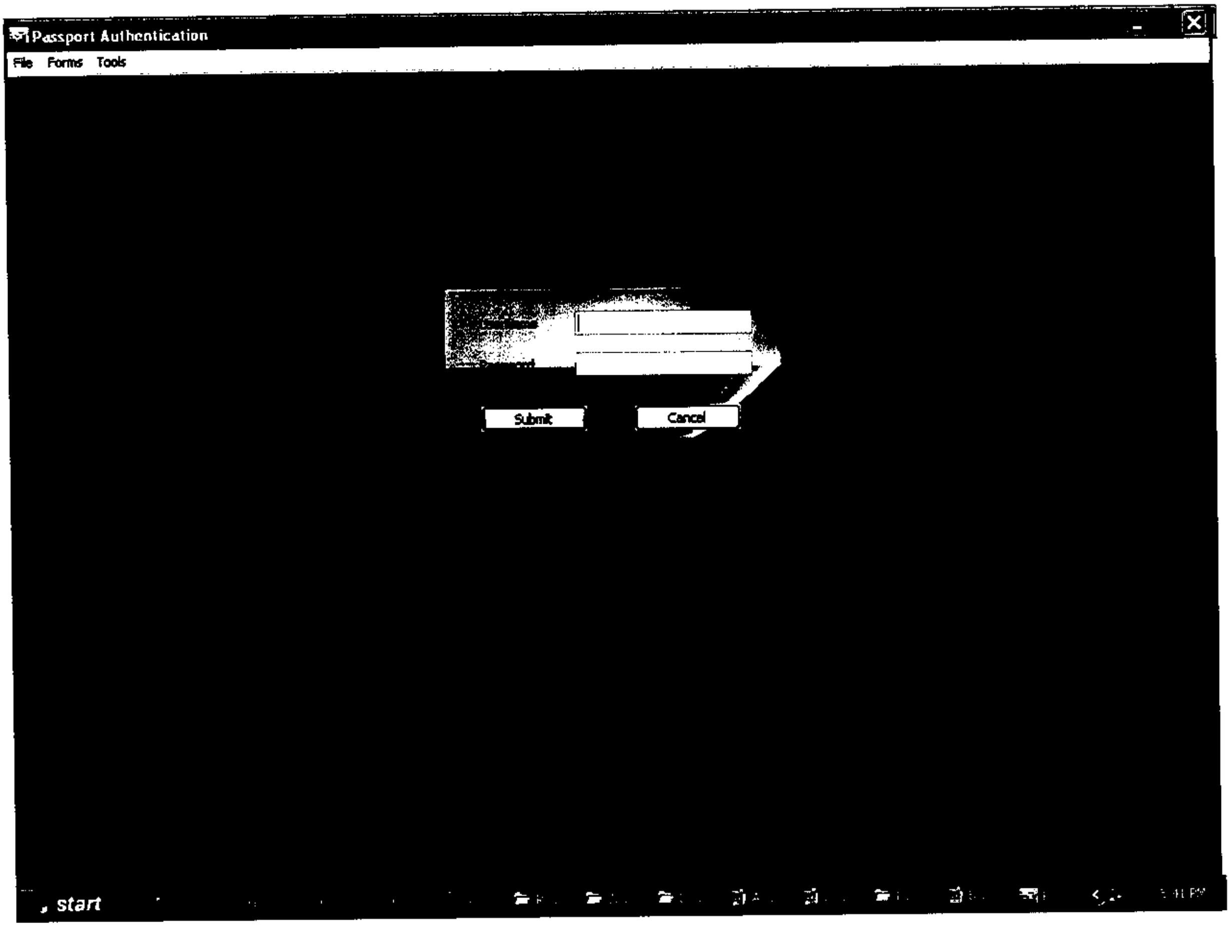
```
        txtMName = new JTextField();
        txtMName.setFocusable(false);
        txtFMName = new JTextField();
        txtFMName.setFocusable(false);
        btnSense = new JButton("...");
        btnSense.addActionListener(this);
        txtPassportNo.getDocument().addDocumentListener(new
DocumentListener(){

                public void changedUpdate(DocumentEvent e) {}
                public void insertUpdate(DocumentEvent e) {
                        if(txtPassportNo.getText()!= null) {
                                if(txtPassportNo.getText().length() > 0) {
                                        btnSubmit.setEnabled(true);
                                }
                                else
                                        btnSubmit.setEnabled(false);
                        }
                        else
                                btnSubmit.setEnabled(false);
                }
                public void removeUpdate(DocumentEvent e) {
                        if(txtPassportNo.getText()!= null) {
                                if(txtPassportNo.getText().length() > 0) {
                                        btnSubmit.setEnabled(true);
                                }
                                else
                                btnSubmit.setEnabled(false);
                    }
                    else
                          btnSubmit.setEnabled(false);
              }

          });
          labelFaceImage = new JLabel("Face Image");
          labelImage = new JLabel();
          labelImage.setBorder(new EtchedBorder());
          btnSubmit = new JButton("Submit");
          btnCompute = new JButton("Compute");
          btnDelete = new JButton("Delete");
          btnClose = new JButton("Close");
          btnSubmit.addActionListener(this);
          btnClose.addActionListener(this);
          btnDelete.addActionListener(this);
          btnCompute.addActionListener(this);
          txtOutput = new JTextArea();
          txtOutput.setEditable(false);
          scrollView = new JScrollPane(txtOutput);



          txtPassportNo.getDocument().addDocumentListener(new
DocumentListener() {
                  public void removeUpdate(DocumentEvent de){
```

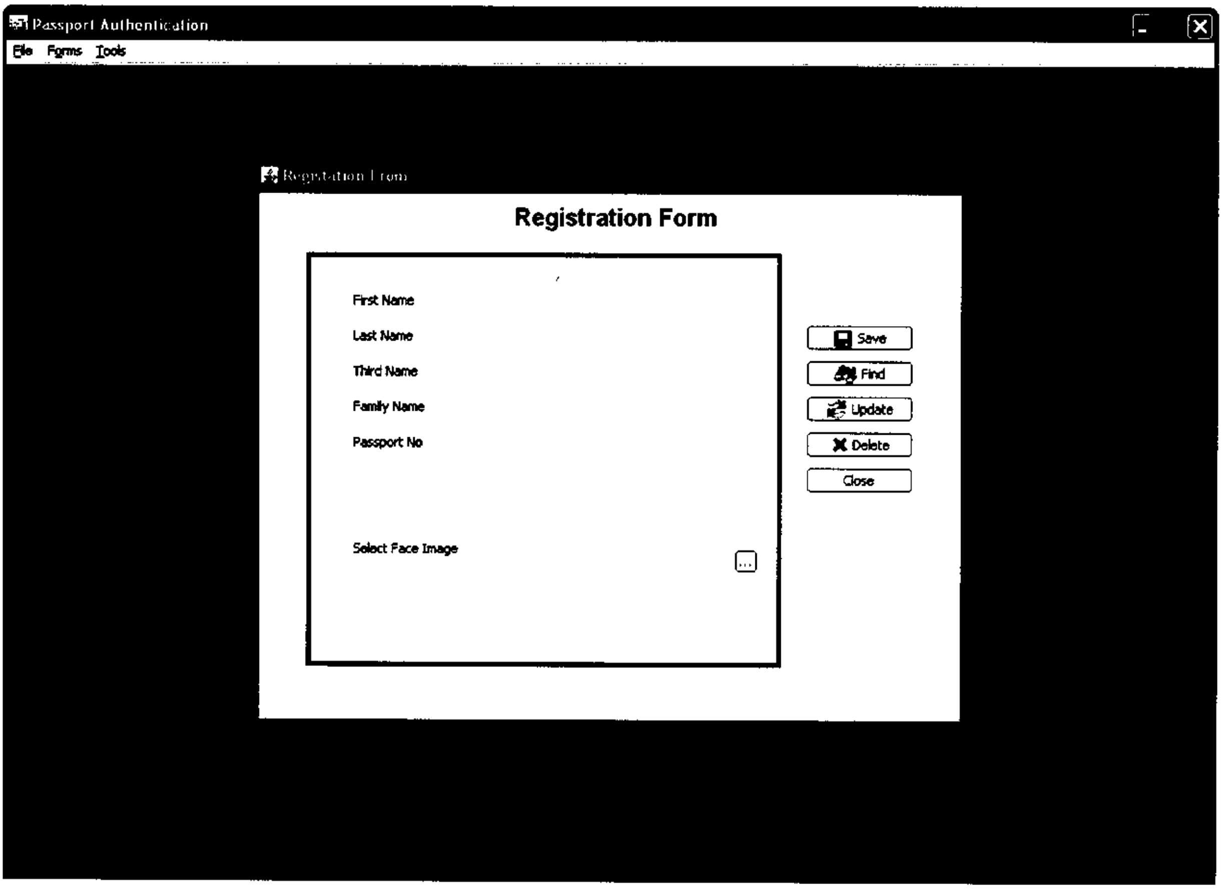
```
            if(txtPassportNo.getDocument().getLength() > 0) {
                btnSubmit.setEnabled(true);
            }
            else
                btnSubmit.setEnabled(false);
        }
        public void changedUpdate(DocumentEvent de){}
        public void insertUpdate(DocumentEvent de){
            if(txtPassportNo.getDocument().getLength() > 0) {
                btnSubmit.setEnabled(true);
            }
            else
                btnSubmit.setEnabled(false);
        }
    });

    panel.setLayout(null);
    panel1.setLayout(null);
    panel.add(labelPassportNo);
    panel.add(txtPassportNo);
    panel1.add(labelFName);
    panel1.add(txtFName);
    panel1.add(labelLName);
    panel1.add(txtLName);
    panel1.add(labelMName);
    panel1.add(txtMName);
    panel1.add(labelFMName);
```

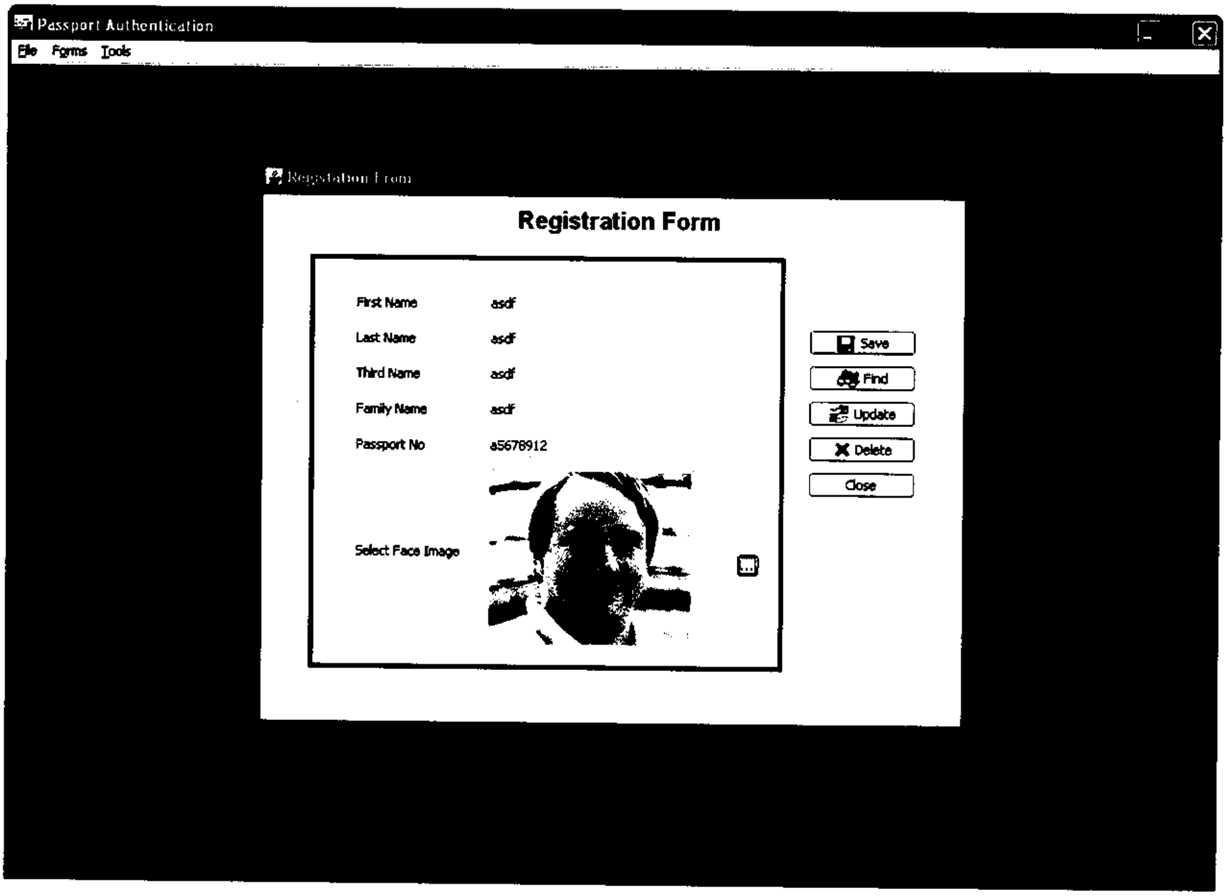
```
panel1.add(txtFMName);
panel.add(btnSense);
labelFName.setBounds(20,20,80,20);
txtFName.setBounds(120,20,180,20);
labelLName.setBounds(20,50,80,20);
txtLName.setBounds(120,50,180,20);
labelMName.setBounds(20,80,80,20);
txtMName.setBounds(120,80,180,20);
labelFMName.setBounds(20,110,80,20);
txtFMName.setBounds(120,110,180,20);
panel.add(labelFaceImage);
panel.add(labelImage);
panel1.add(btnSubmit);
panel1.add(btnCompute);
panel1.add(btnDelete);
panel1.add(btnClose);
panel.add(scrollView);
btnSubmit.setBounds(105,170,90,20);
btnCompute.setBounds(105,200,90,20);
btnDelete.setBounds(105,230,90,20);
btnClose.setBounds(105,260,90,20);
labelPassportNo.setBounds(20,20,80,20);
txtPassportNo.setBounds(120,20,250,20);
btnSense.setBounds(370,20,20,20);
labelFaceImage.setBounds(20,100,80,20);
labelImage.setBounds(120,60,250,150);
btnDelete.setEnabled(false);
```

```
        btnCompute.setEnabled(false);
        btnSubmit.setEnabled(false);
        scrollView.setBounds(25,250,350,150);
        getContentPane().setLayout(null);
        getContentPane().add(panel);
        EtchedBorder bor = new EtchedBorder(Color.blue,Color.green);
        panel.setBorder(new CompoundBorder(bor,bor));
        panel1.setBorder(new CompoundBorder(bor,bor));
        panel.setBounds(20,20,400,420);
        getContentPane().add(panel1);
        panel1.setBounds(450,20,320,420);
    }



}
```

7.2 SCREEN SHOTS:

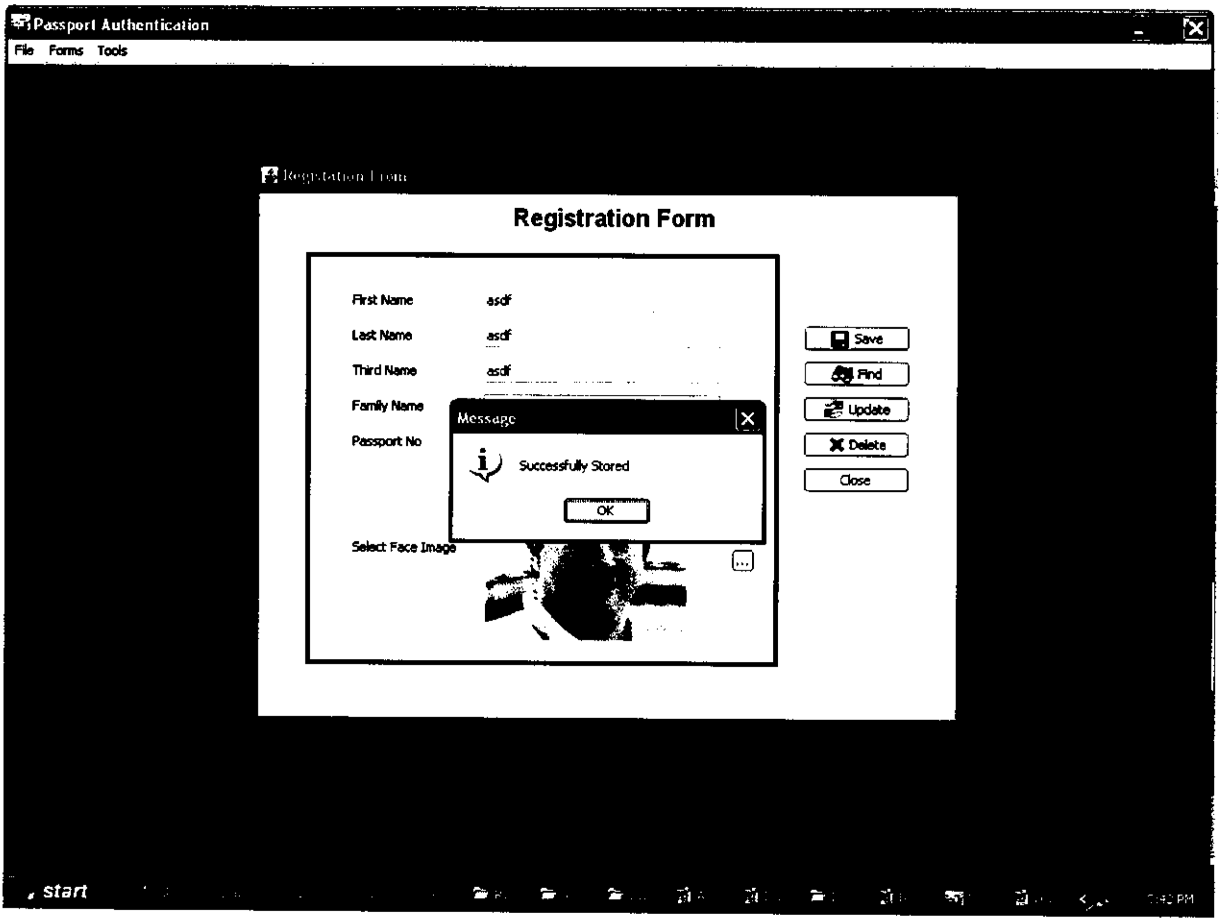



Passport Authentication
File
Forms
Tools
Registration From
Registration Form
First Name
Last Name
Third Name
Family Name
Passport No
Select Face Image
...
Save
Find
Update
Delete
Close



Registration Form

First Name	asdf
Last Name	asdf
Third Name	asdf
Family Name	asdf
Passport No	a5678912
Select Face Image	



Registration Form

First Name asdf

Last Name asdf

Third Name asdf

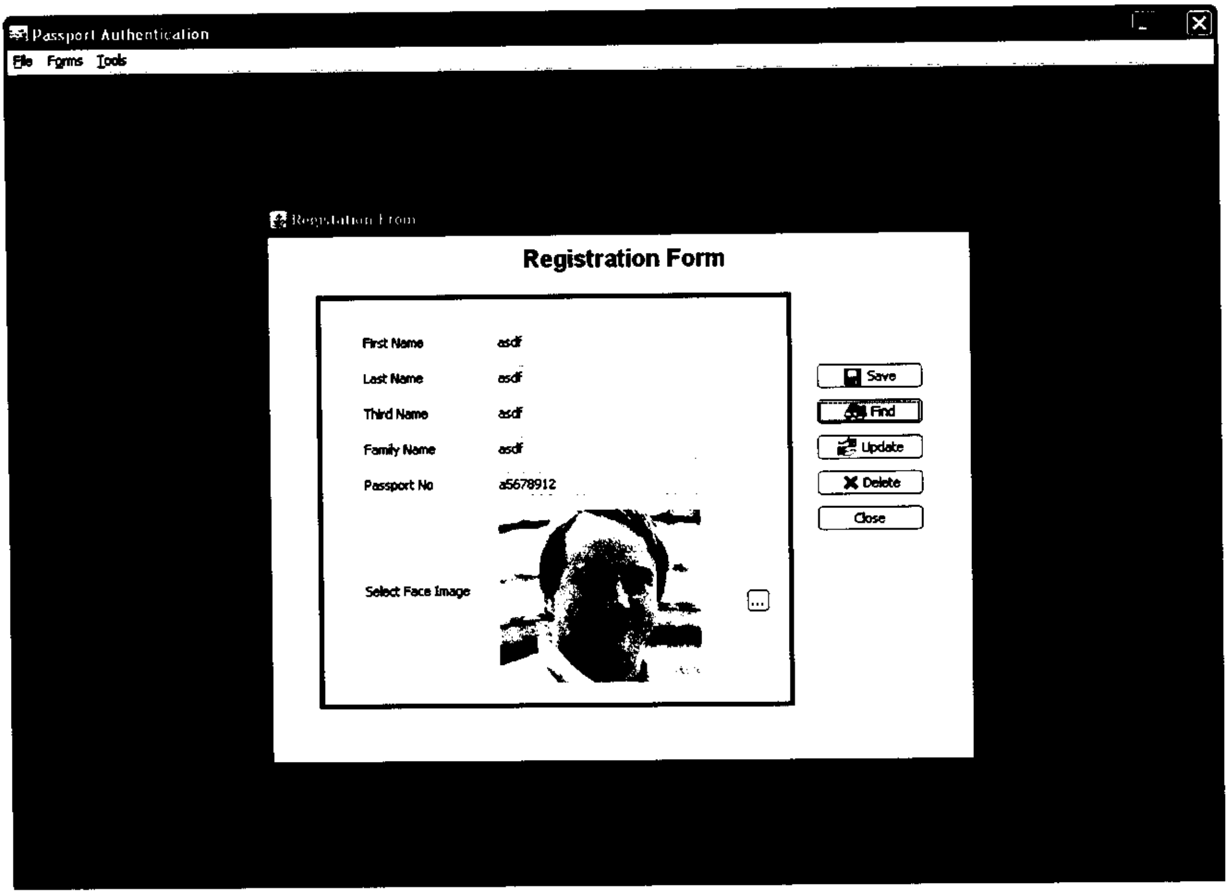
Family Name

Passport No

Select Face Image

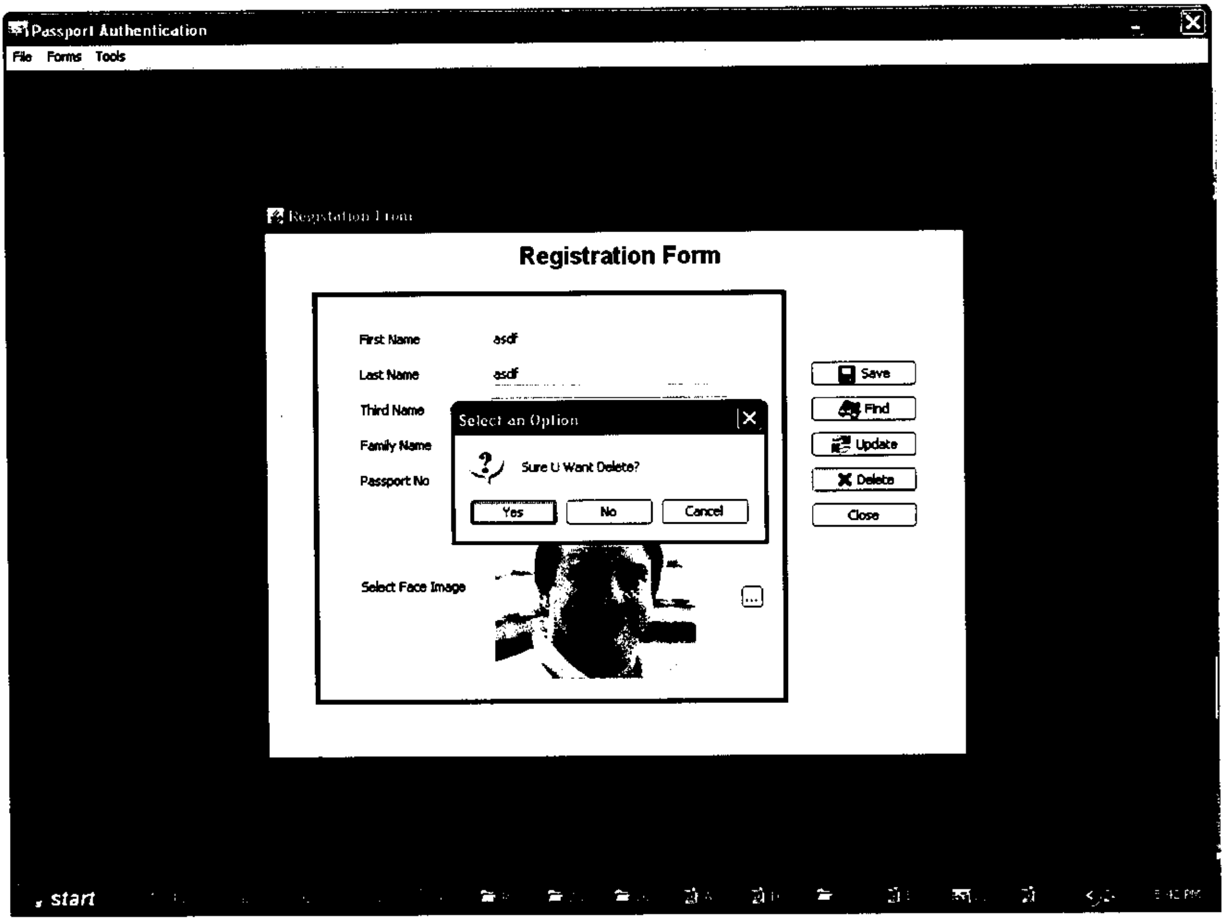
Message

Successfully Stored



Registration Form

First Name	asdf
Last Name	asdf
Third Name	asdf
Family Name	asdf
Passport No	a5678912
Select Face Image	



Registration Form

First Name asdf

Last Name asdf

Third Name

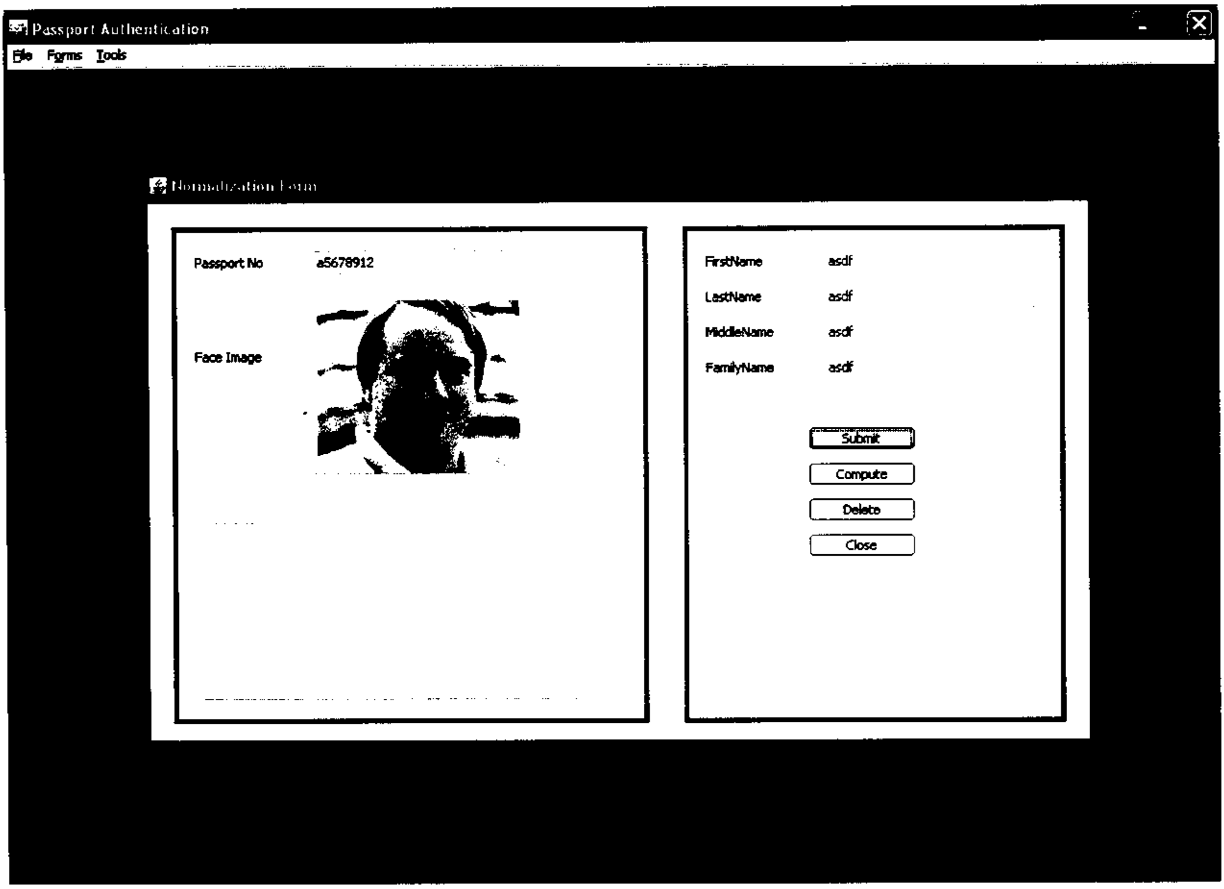
Family Name

Passport No

Select Face Image

Select an Option

Sure U Want Delete?



Passport No a5678912

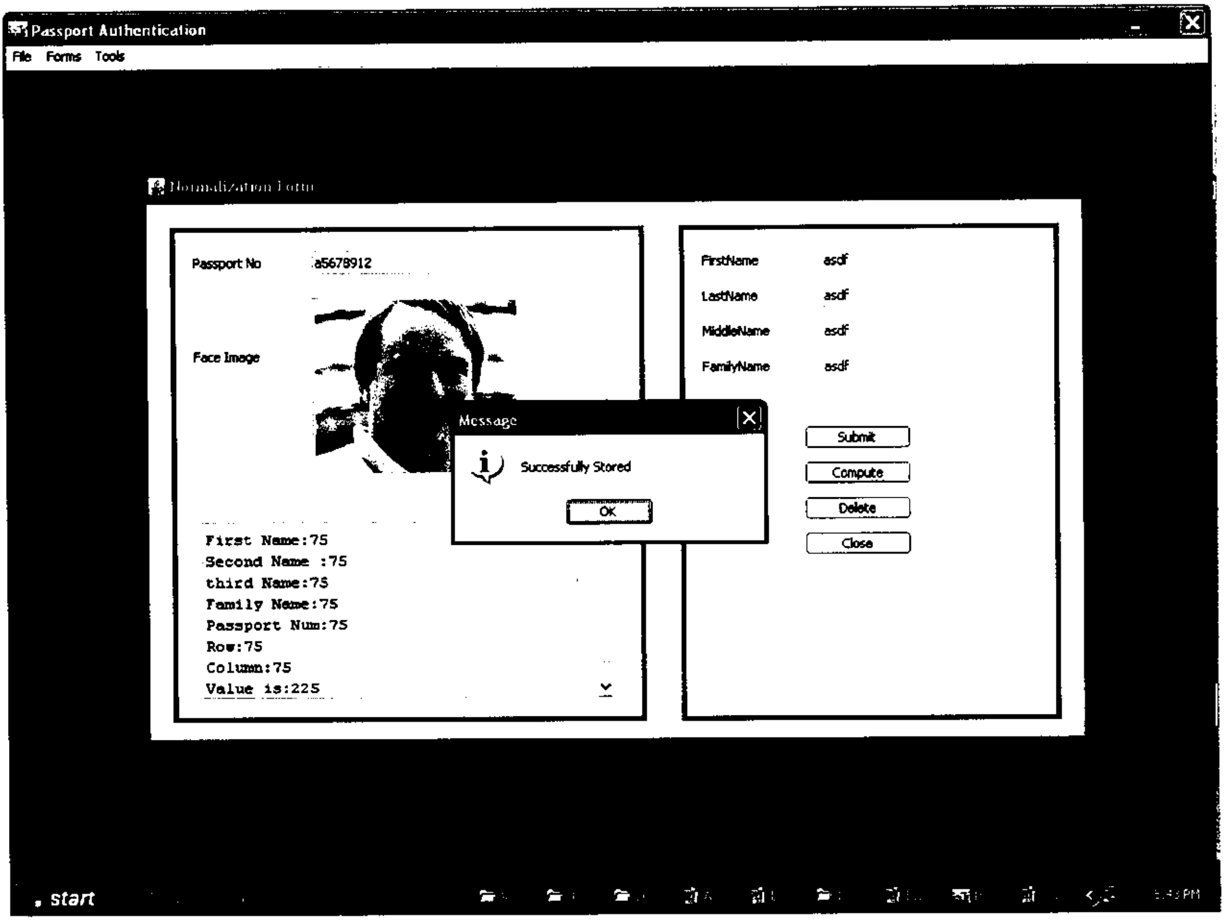
Face Image

FirstName asdf

LastName asdf

MiddleName asdf

FamilyName asdf



Passport No a5678912

Face Image

FirstName asdf

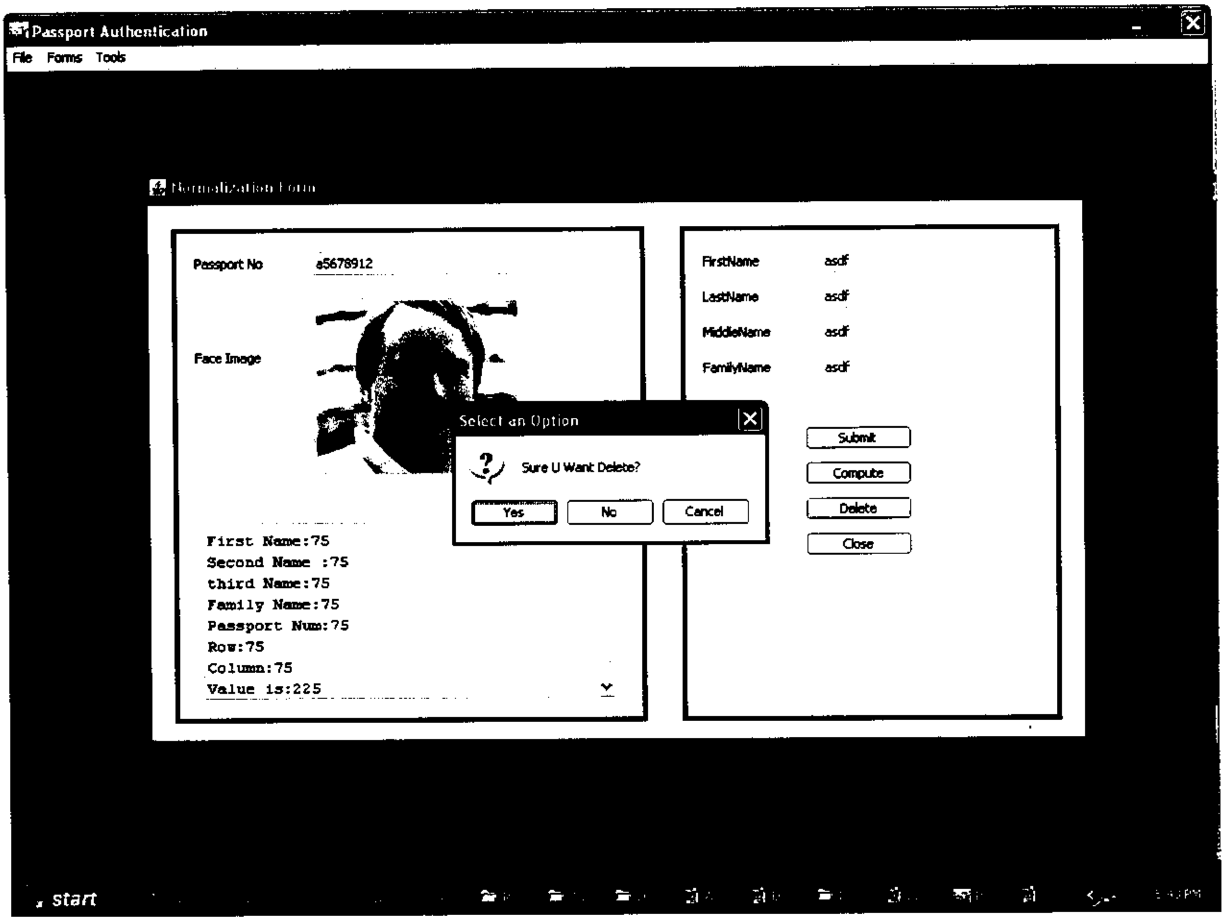
LastName asdf

MiddleName asdf

FamilyName asdf

Successfully Stored

```
First Name:75
Second Name :75
third Name:75
Family Name:75
Passport Num:75
Row:75
Column:75
Value is:225
```



Passport No a5678912

Face Image

```
First Name:75
Second Name :75
third Name:75
Family Name:75
Passport Num:75
Row:75
Column:75
Value is:225
```

FirstName asdf

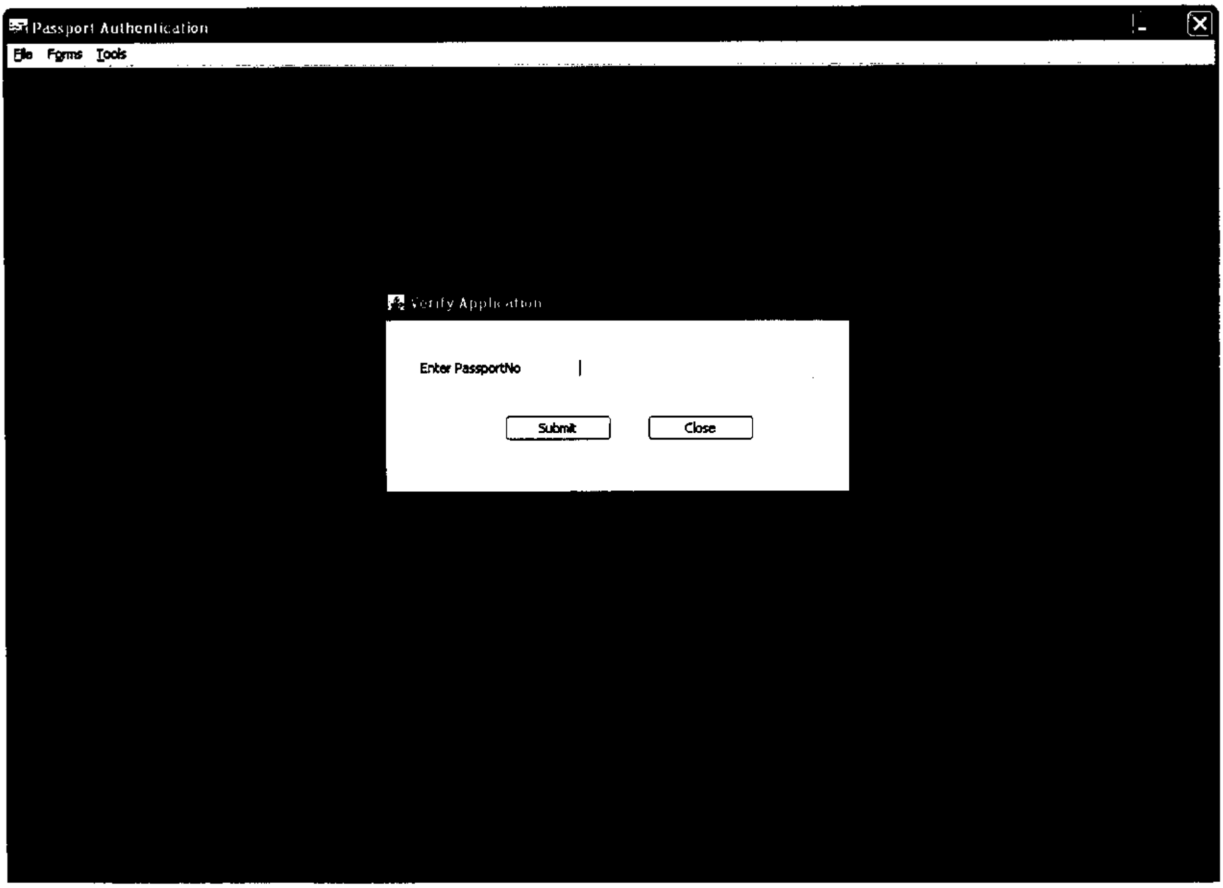
LastName asdf

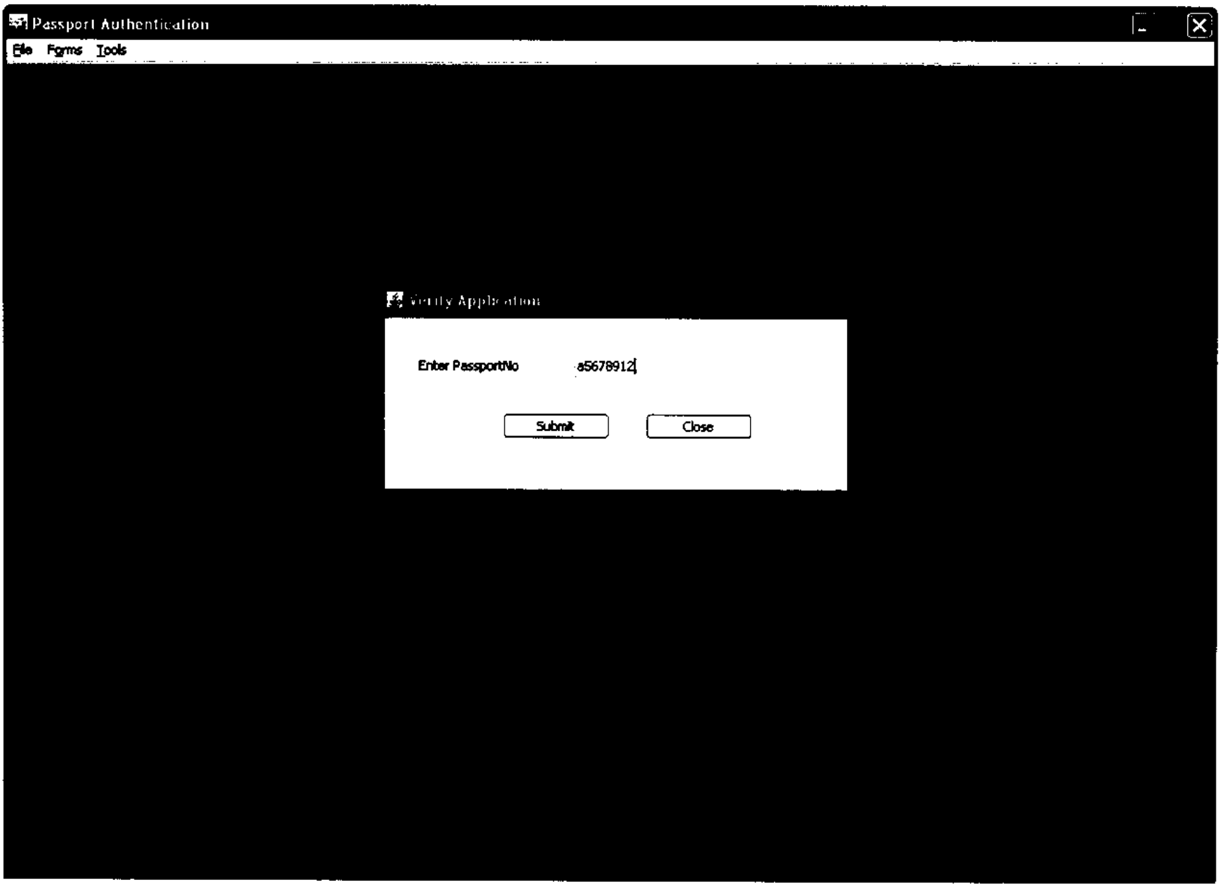
MiddleName asdf

FamilyName asdf

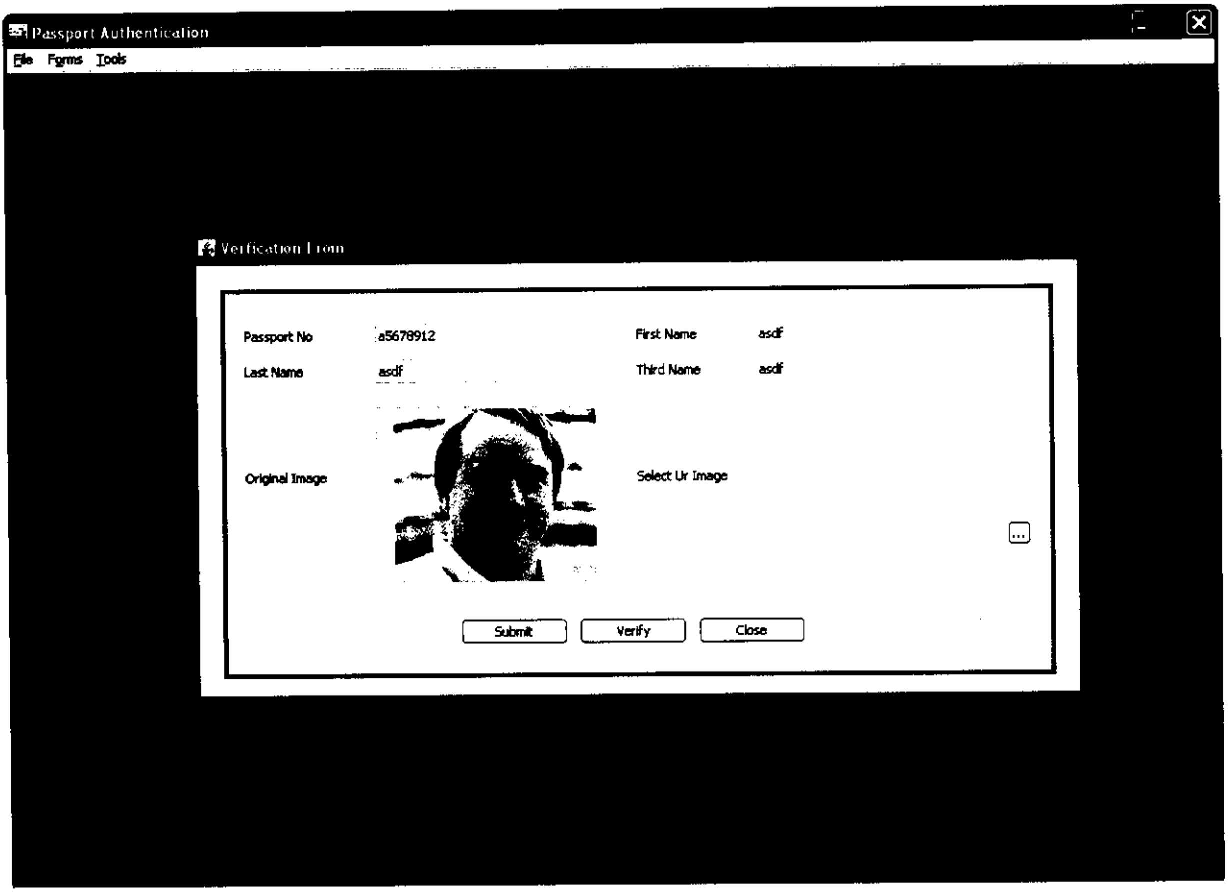
Select an Option

Sure U Want Delete?

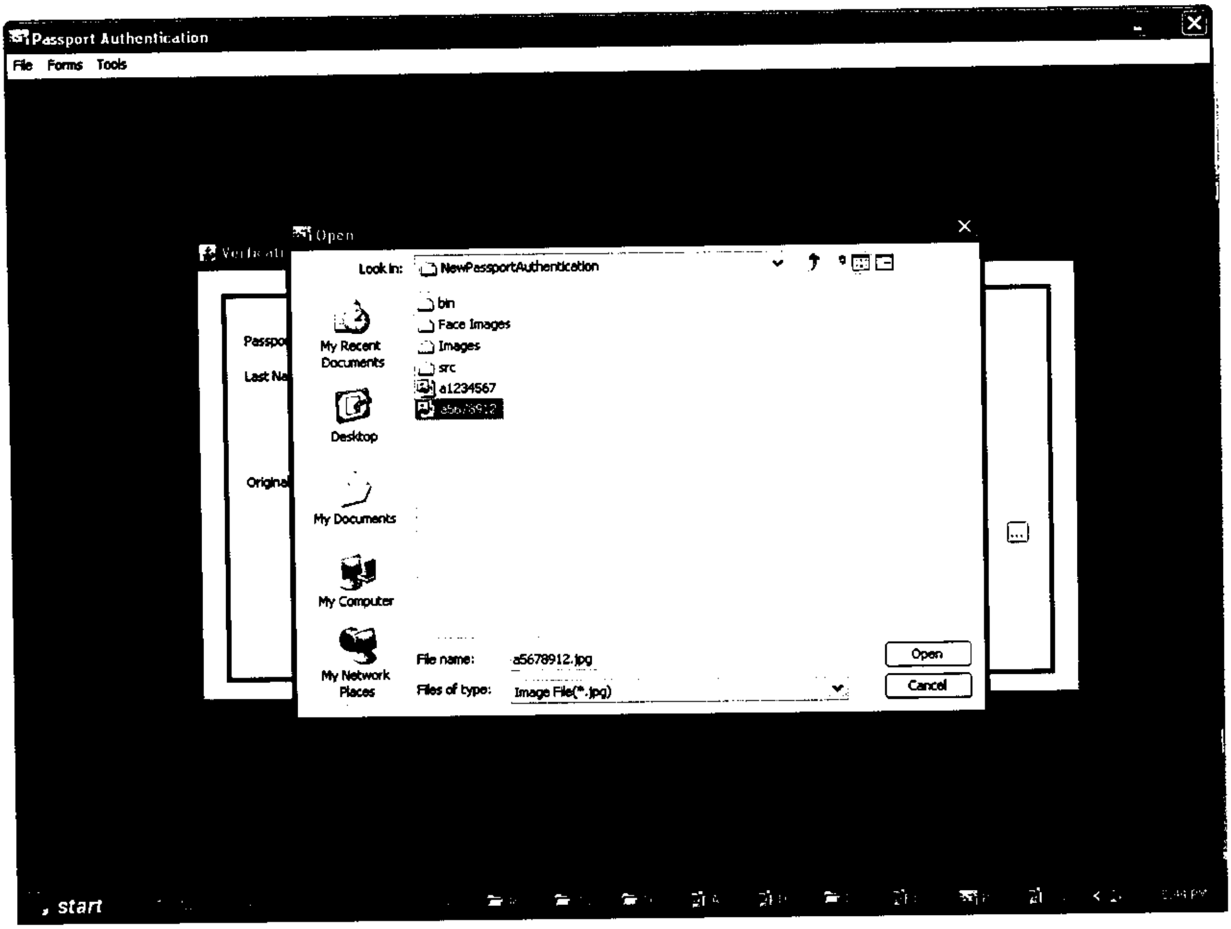

Passport Authentication
File
Forms
Tools
Verify Application
Enter PassportNo
Submit
Close



Enter PassportNo a5678912



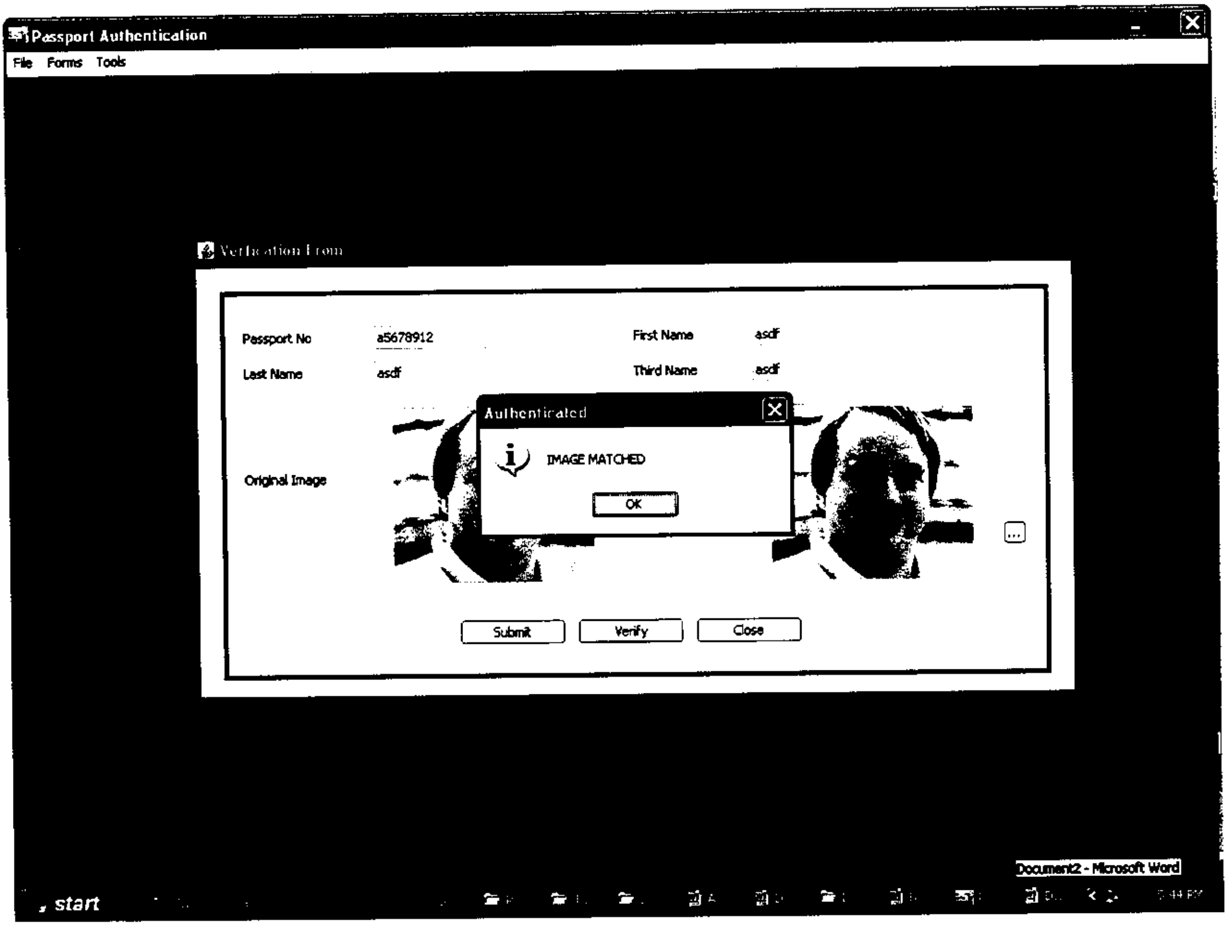
Passport No	a5678912	First Name	asdf
Last Name	asdf	Third Name	asdf
Original Image		Select Ur Image	



Look in: NewPassportAuthentication

File name: a5678912.jpg

Files of type: Image File(*.jpg)



Passport No a5678912

First Name asdf

Last Name asdf

Third Name asdf

IMAGE MATCHED

8. References

1. Scott Oaks, Henry Wong, Mike Loukides (Editor), "Java Threads Java Series", O'Reilly & Associates.
2. Patrick Naughton, Herbert Schildt, "Java™ 2: The Complete Reference", Third Edition, Tata McGraw-Hill Publishing Company Limited.
3. David Flanagan, "Java in a Nutshell", 2nd Edition, May 1997.
4. Mark Grand and Jonathan Knudsen, "Java Fundamental Classes Reference", Tata McGraw-Hill Publishing Company Limited, 1st Edition, May 1997.
5. Dietel & Deitel, Java How To Program, BPB Publishers, New Delhi, 2000.
6. O'reilly, Java Swings, Tata McGraw Hill, Fifth Edition, 2002.
7. Broukshickle, Thinking in Java, Calgotia Publishers, New Delhi, 2001