A PROJECT REPORT

ON

AADHAAR CARD ENHANCEMENT

BY

Pranav Kumar Gupta (RA1511003040064)

Shreyas Keelary (RA15110030400084)

Harshit Soni (RA1511003040174)

15SE203-Object Oriented Analysis and Design

SEMESTER 4

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

April-2017

SRM UNIVERSITY

(UNDER SECTION 3 OF UGC ACT, 1956)

VADAPALANI CAMPUS

CHENNAI-600026

BONAFIDE CERTIFICATE

Certified to be the Bonafide Record of the work done by _____of II Year B.TECH Computer Science and Engineering degree course in the 15SE203-Object Oriented Analysis and Design in **SRM UNIVERSITY**, -Vadapalani, Chennai -26 during the academic year 2016-2017

Subject In charge **Head of the Department**

Date:

ACKNOWLEDGEMENT

We place our deep sense of gratitude towards our beloved **chancellor Mr. T. R. Pachaimuthu , SRM UNIVERSITY** for providing us with the required infrastructure throughout the course.

We take the opportunity to extend our heartfelt thanks to our respected Dean, Dr.K.Duraivelu for his support and impeccable guidance.

We are extremely grateful to the Head of the Department, Mrs. B.Padhmavathi for having encouraged and helped us throughout the course of the project. Without her, it would have been really very difficult for us to finish the project in a timely manner. Thus we feel deeply obliged for her support.

We are also grateful to our guide Mr.Rajiv Gandhi for having assisted us in finishing our project.

TABLE OF CONTENTS

LIST OF FIGURES

Abstract

Aadhar or a Unique Identification is a service available in India that allows a card to be a nationally accepted Identification medium for all services provided by the government and most of the private sector agencies. It enables the citizens of India to use that card as a proof of their identity. But apart from that it has no appreciable uses so far. The government has spent a lot of taxpayer's money on a system they have themselves made obsolete. This has to change. Instead of aadhar being a mere ID proof, this system makes it such that it can be used even by the government to identify the common man whenever required. A common card to serve all purposes. A card that can act as a Criminal identification device, a medical insurance card, a banking card, a vehicle license, a cadaver identification service, a lost child finding service,….., in other words an all-purpose governmental use card.

The present day aadhar service is limited to a mere identification service because it lacks some key features and hence they can be replicated with ease. We can actually download and print multiple copies of the card online. We intend to put an end to that. The system we propose adds a DNA database and a facial recognition feature to the existing system.

CHAPTER 1

Introduction

Proposed System

This proposal for a slightly higher security, more durability and a spoof proof system is actually a quite simple and yet a quite effective upgrade to the existing system. It is of no doubt that a database containing DNA may take time to build. But nevertheless once such a system is put in place, it will be beneficial in long run.

Existing System

There is an already existent aadhar card. An obsolete ID card that is of no more use that a college ID card. The college ID card can get you discounts at certain places. But an aadhar card won't do that even. We plan on putting a presently out of use system that was started as an innovative initiative but went off rails, back on track.

The Aadhaar Card system already exists. But it is now merely an ID proof. The government giving freedom of printing it online, poses the greatest threat to its intended use as "ID Proof", as anyone having access to a mobile phone may get hands on a digital copy of this ID proof, one that was supposed to be Unique. And Digital copies make it easier to spoof.

Experts say that it is very easy to fake a fingerprint of a person, even iris can be simulated. Hence, it is not 100% reliable.

Advantages

Advantages of a universal DNA Database:

- Faster Identification of criminals or victims based on any DNA evidence left at the scene of crime.
- Easier Identification of cadavers found in several unsolved cases.
- Easier Identification of lost children.
- Finding organ acceptors in case of the death of a possible organ donor.

Advantages of a universal Facial recognition database:

- Faster Identification of criminals.
- Reduction of crime rates.

Advantages of Proposed System as a whole:

- A single card for all governmental purposes.
- Identification of stolen vehicles.
- Reduces expenditure in criminal investigation.
- Reduces the annual losses to life and property due to crimes.
- Reduce expenditure incurred by all governmental organization for issuing separate documents.
- All data pertaining to a person can now be obtained in one location and is available in digital format globally.

Disadvantages

- Creating a DNA Database and a facial recognition database will be expensive and time consuming.
- Such a database will require too much security and adequate man power to sustain and maintain.
- Facial recognition must be updated regularly for high accuracy.

Application

- Can be used by all governmental organizations for their functioning.

CHAPTER 2
SYSTEM REQUIREMENTS

Software Requirements

- A System that has an operating system that can run a Java Runtime Environment.
- Java Runtime Environment.
- An Oracle Server.

Hardware Requirements

- Any device with RAM at least 2GB or higher.
- Processor above Core I3 3rd Generation.
- An effective cooling station in case only minimum requirements are met.

CHAPTER 3

LIST OF MODULES

3.1 UI (User Interface)

3.1.1 EntryUI
3.1.2 SearchUI

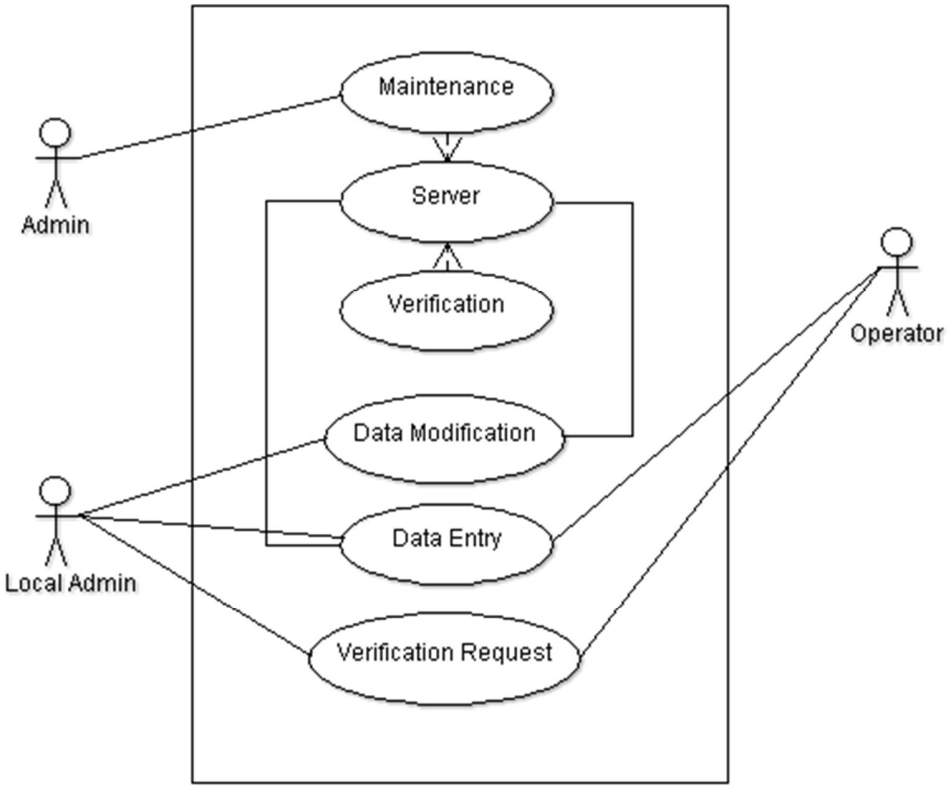
3.2 Biometrics Processing

3.2.1 DNA File Processing
3.2.2 Fingerprint Detection and Processing
3.2.3 Iris Recognition
3.2.4 Facial Pattern Recognition

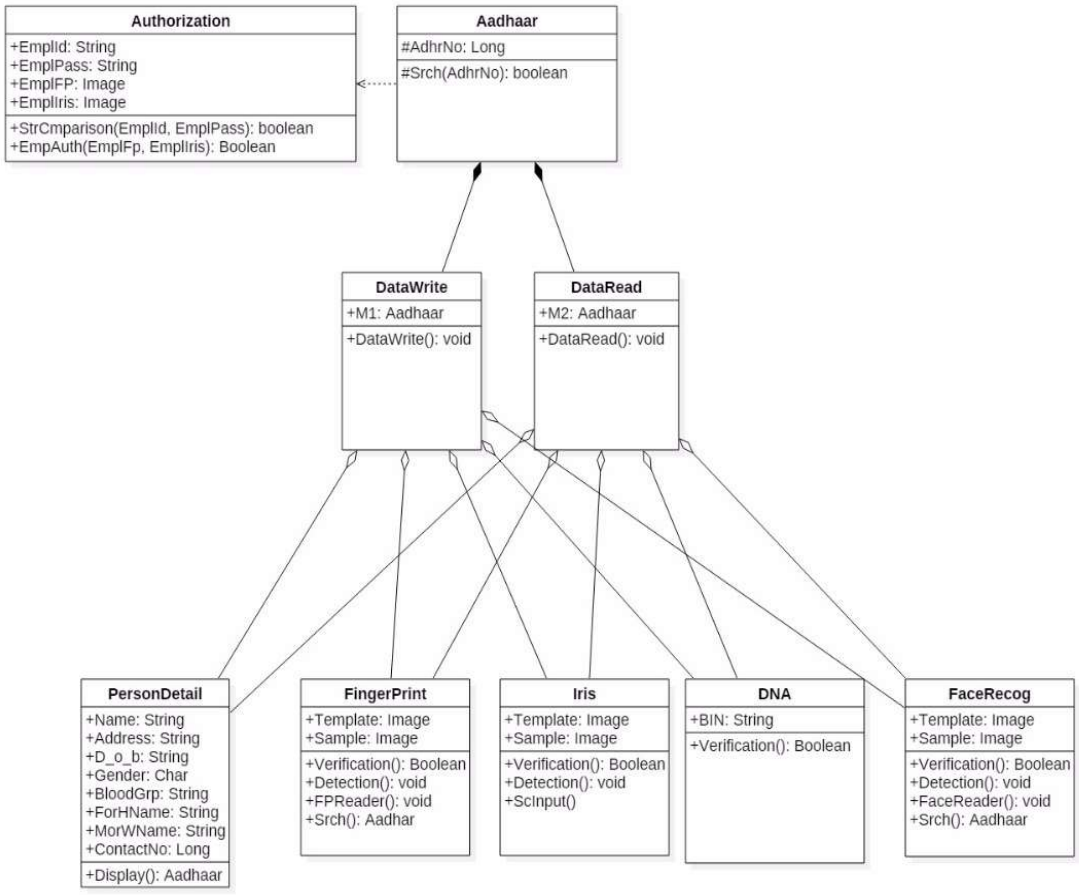
3.3 Oracle Database

CHAPTER 4
Static Diagrams

4.1 USE CASE DIAGRAM



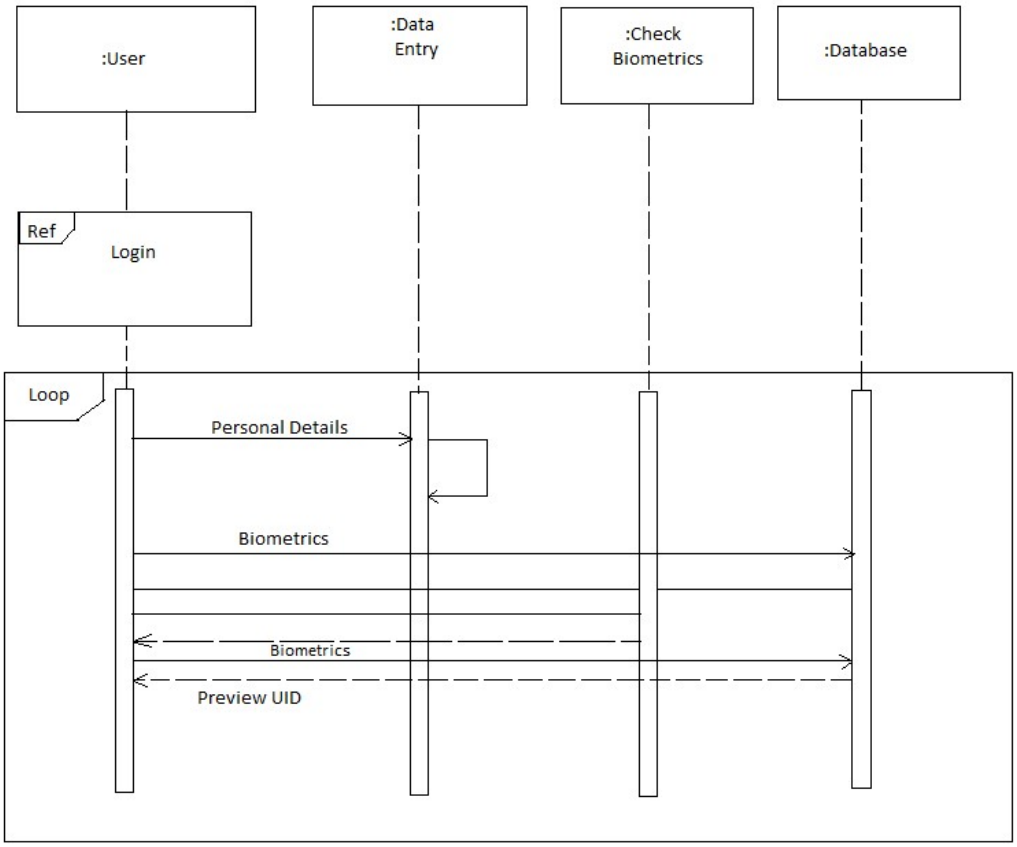
4.2 CLASS DIAGRAM



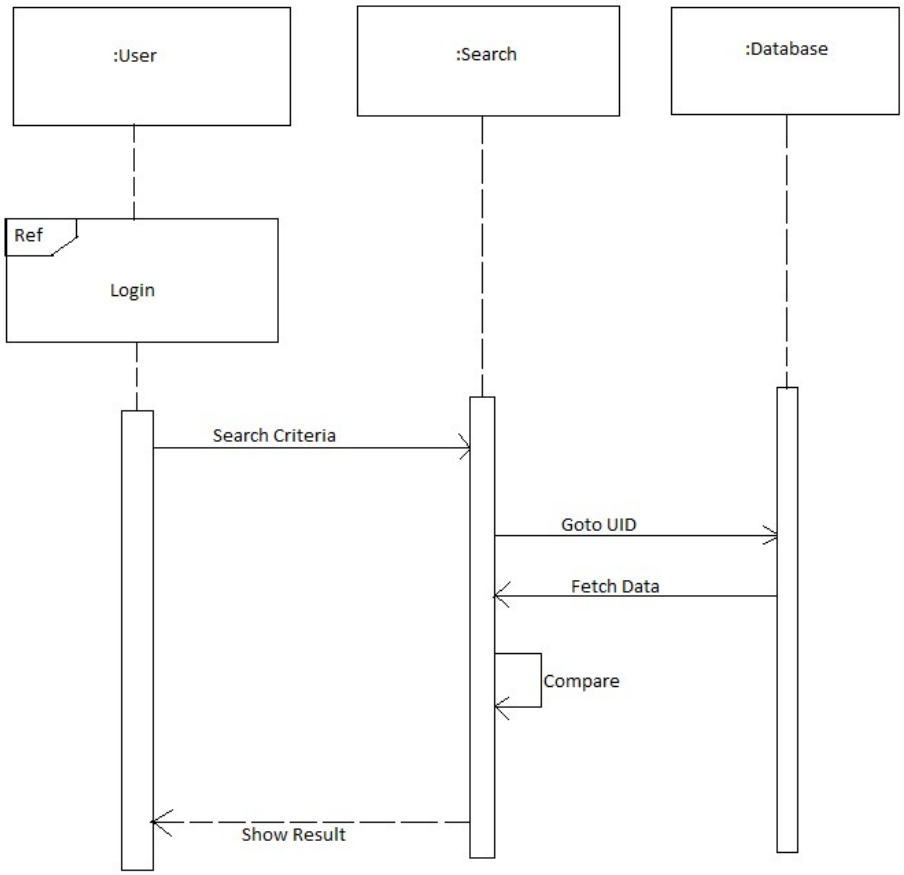
CHAPTER 5
Behavioral Diagrams

5.1 SEQUENCE DIAGRAMS

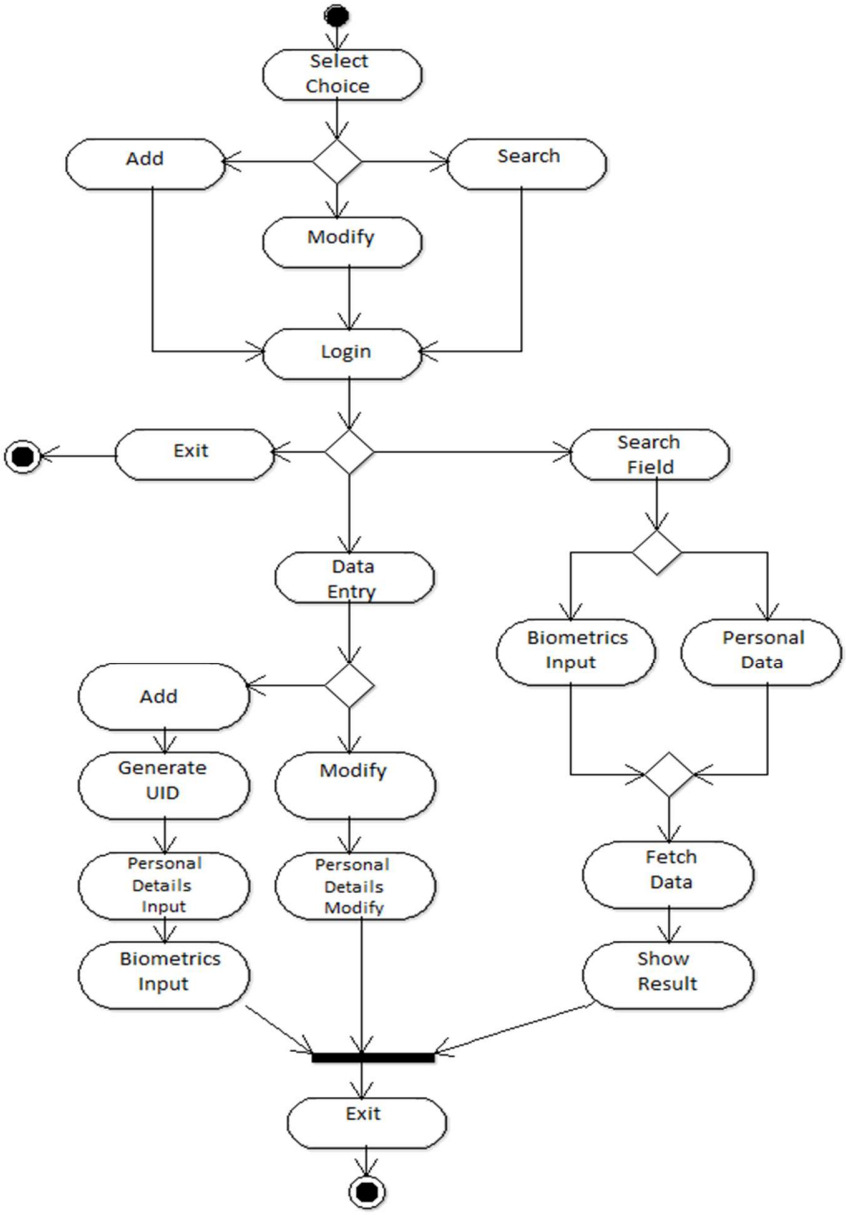
5.1.1 Addition



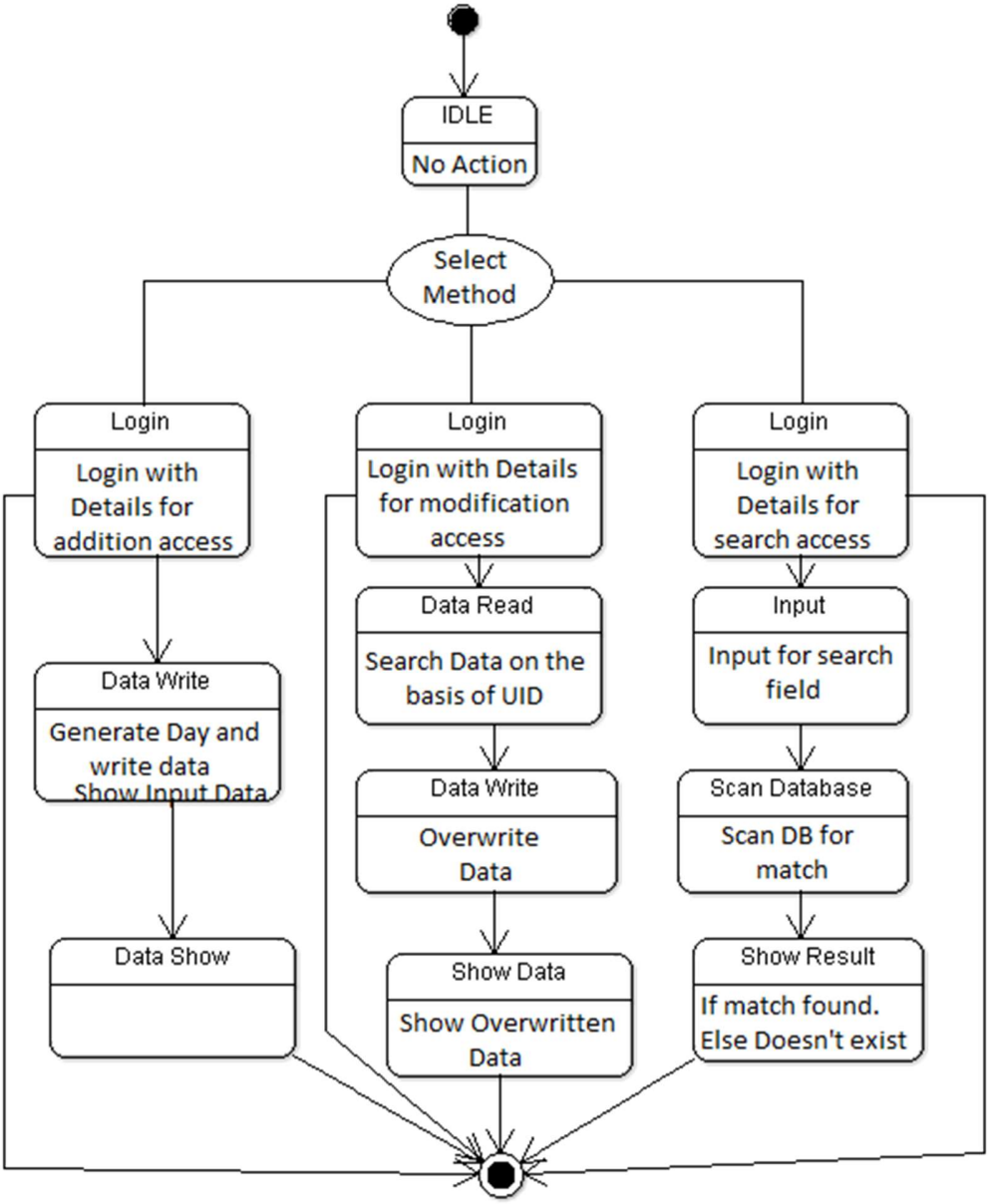
5.1.2 Search


:User
:Search
:Database
Ref
Login
Search Criteria
Goto UID
Fetch Data
Compare
Show Result

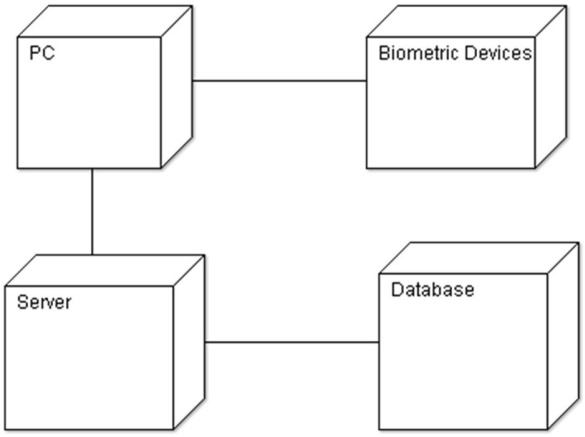
5.2 ACTIVITY DIAGRAM



5.3 STATE MACHINE DIAGRAM



5.4 DEPLOYMENT DIAGRAM



CHAPTER 6

IMPLEMENTATION TOOLS

The set of tools we used to implement this project are follows

5.1 ARGOUML Software:

ArgoUML is an UML diagramming application written in Java and released under the open-source Eclipse Public License. By virtue of being a Java application, it is available on any platform supported by Java SE. According to the official feature list, ArgoUML is capable of the following:

- All 9 UML 1.4 diagrams are supported.
- Closely follows the UML standard.
- Platform independent – Java 1.5+.
- Click and Go! with Java Web Start (no setup required, starts from your web browser).
- Standard UML 1.4 Metamodel.
- XMI support.
- Export diagrams as GIF, PNG, PS, EPS, PGML and SVG.
- Available in ten languages: EN, EN-GB, DE, ES, IT, RU, FR, NB, PT, ZH.
- Advanced diagram editing and zoom.
- Built-in design critics provide unobtrusive review of design and suggestions for improvements.
- Extensible modules interface.
- OCL support.
- Forward engineering (code generation supports C++ and C#, Java, PHP 4, PHP 5, Ruby and, with less mature modules, Ada, Delphi and SQL).
- Reverse engineering / JAR/class file import.

5.2 Oracle Database 12c:

5.3 Notepad ++:

Notepad++ is a free (as in "free speech" and also as in "free beer") source code editor and Notepad replacement that supports several languages. Running in the MS Windows environment, its use is governed by GPL License.

- Syntax Highlighting and Syntax Folding
- User Defined Syntax Highlighting and Folding: screenshot 1, screenshot 2, screenshot 3 and screenshot 4
- PCRE (Perl Compatible Regular Expression) Search/Replace
- GUI entirely customizable: minimalist, tab with close button, multi-line tab, vertical tab and vertical document list
- Document Map
- Auto-completion: Word completion, Function completion and Function parameters hint
- Multi-Document (Tab interface)
- Multi-View
- WYSIWYG (Printing)
- Zoom in and zoom out
- Multi-Language environment supported
- Bookmark
- Macro recording and playback
- Launch with different arguments

CHAPTER 7

SAMPLE CODING

Elements in EntryUI:

```
private datechooser.beans.DateChooserCombo dateChooserCombo1;
private javax.swing.JLabel imagearea;
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JButton jButton5;
private javax.swing.JButton jButton6;
private javax.swing.JButton jButton7;
private javax.swing.JComboBox<String> jComboBox1;
private javax.swing.JComboBox<String> jComboBox2;
private javax.swing.JLabel jLabel1;
private javax.swing.JPanel jPanel1;
private javax.swing.JTextField jTextField1;
private javax.swing.JTextField jTextField10;
private javax.swing.JTextField jTextField11;
private javax.swing.JTextField jTextField12;
private javax.swing.JTextField jTextField2;
private javax.swing.JTextField jTextField3;
private javax.swing.JTextField jTextField4;
```

```
    private javax.swing.JTextField jTextField5;
    private javax.swing.JTextField jTextField6;
    private javax.swing.JTextField jTextField7;
    private javax.swing.JTextField jTextField8;
    private javax.swing.JTextField jTextField9;
```

Generating Aadhaar Number:

```
private long genadhno() {
        Random rand = new Random();
        char[] digits = new char[12];
        digits[0] = (char) (rand.nextInt(9)+'1');
        for(int i = 1; i < 12; i++)
            digits[i] = (char) (rand.nextInt(10)+'0');
        return Long.parseLong(new String(digits));
    }
```

Input from User:

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_jButton3ActionPerformed
        fc1 = new JFileChooser();
        fc1.setFileSelectionMode(JFileChooser.FILES_ONLY);
        fc1.addChoosableFileFilter(new FileNameExtensionFilter("DNA File", "fasta"));
        fc1.setAcceptAllFileFilterUsed(false);
        int result = fc1.showOpenDialog(this);
        if (result == JFileChooser.APPROVE_OPTION)
            dna = fc1.getSelectedFile();
    }//GEN-LAST:event_jButton3ActionPerformed
```

```
    private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {

        fc2 = new JFileChooser();

        fc2.setFileSelectionMode(JFileChooser.FILES_ONLY);

        fc2.addChoosableFileFilter(new FileNameExtensionFilter("Images", "jpg", "png", "gif",
"bmp"));

        fc2.setAcceptAllFileFilterUsed(false);

        int result = fc2.showOpenDialog(this);

        if (result == JFileChooser.APPROVE_OPTION)

            img = fc2.getSelectedFile();

        try {

            image = ImageIO.read(img);

        } catch (IOException ex) {

            JOptionPane.showMessageDialog(this, "Invalid image!", "Error",
JOptionPane.ERROR_MESSAGE);

        }

        ImageIcon icon =new ImageIcon(image);

        imagearea.setIcon(icon);

    }


    private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {

        jTextField1.setText("");

        jTextField2.setText("House/Door No.");

        jTextField3.setText("Village/Town/City");

        jTextField4.setText("Street/Road/Lane");

        jTextField5.setText("Area/Locality/Sector");
```

```
        jTextField6.setText("District");
        jTextField7.setText("State");
        jTextField8.setText("");
        jTextField9.setText("");
        jTextField10.setText("");
        jTextField11.setText("");
        jTextField12.setText("");
        dna = null;
        image = null;
        img = null;
        imagearea.setIcon(null);
    }
```

Search using DNA

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt){
        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            try(Connection
con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:ORCL","SYSTEM","Prana
v123")){
                Statement stmt=con.createStatement();
                ResultSet rs = stmt.executeQuery("select * from aadhar");
                boolean flag = false;
                int i;
                while(!flag && rs.next()){
                    Clob c = rs.getClob(12);
                    Reader r = c.getCharacterStream();
```

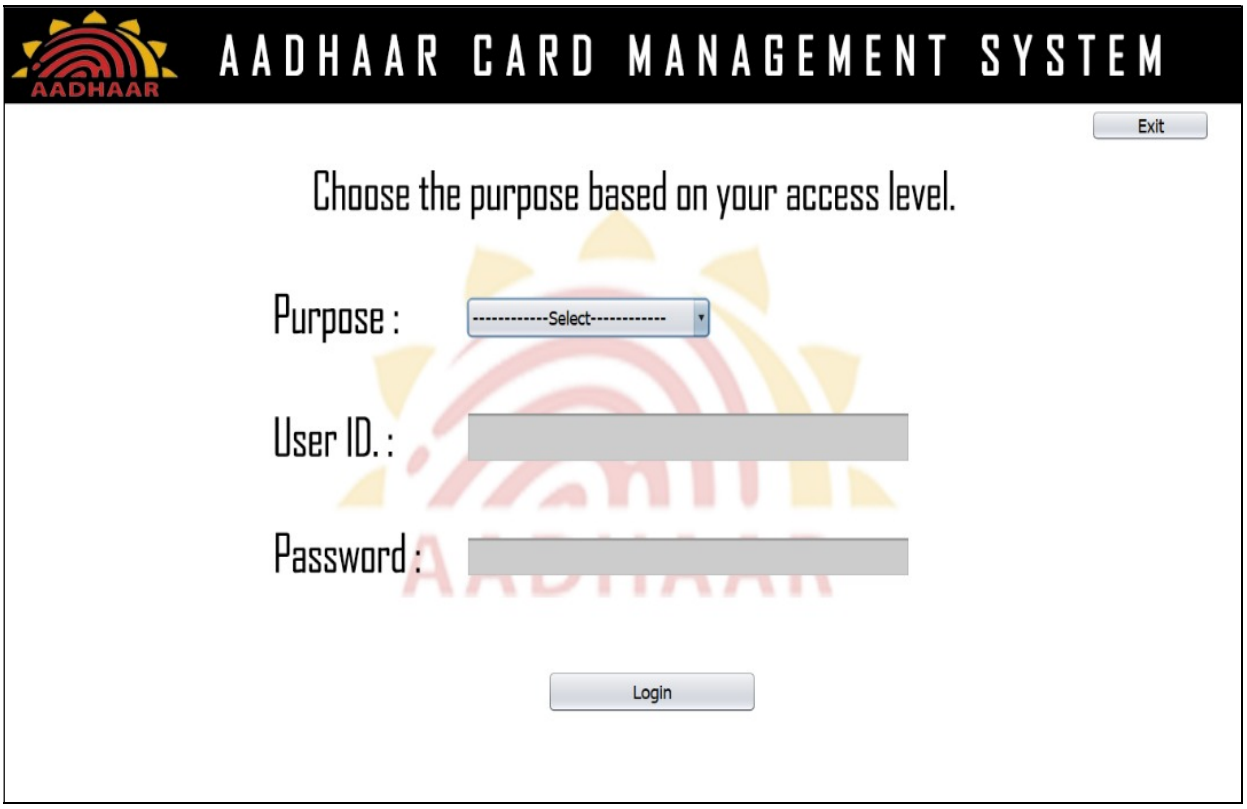
```
try {
    try (FileWriter fw = new FileWriter("tmp1.fasta")) {
        while((i = r.read()) != -1)
            fw.write((char)i);
    }
    File temp = new File("tmp1.fasta");
    Compare cmp = new Compare(dnaf,temp);
    flag = cmp.compare();
    temp.delete();
    if(flag){
        jTextField3.setText(String.valueOf(rs.getLong(1)));
        jTextField2.setText(String.valueOf(rs.getString(2)));
        jTextField6.setText(String.valueOf(rs.getString(3)));
        jTextField1.setText(String.valueOf(rs.getString(4)));
        jTextField5.setText(String.valueOf(rs.getLong(7)));
        jTextField4.setText(String.valueOf(rs.getString(13)));
        jTextField8.setText(String.valueOf(rs.getString(14)));
        jTextField9.setText(String.valueOf(rs.getString(15)));
        jTextField10.setText(String.valueOf(rs.getString(16)));
        jTextField11.setText(String.valueOf(rs.getString(17)));
        jTextField7.setText(String.valueOf(rs.getString(18)));
        Blob b = rs.getBlob(11);
        byte barr[] = b.getBytes(1, (int)b.length());
        try (FileOutputStream fout = new FileOutputStream("tmp2.jpg")) {
            fout.write(barr);
```

```
                    }
                    img1 = new File("tmp2.jpg");
                    Image image = ImageIO.read(img1);
                    ImageIcon icon =new ImageIcon(image);
                    imagearea.setIcon(icon);
                    img1.delete();
                  }
                } catch (IOException | SQLException ex) {
                   JOptionPane.showMessageDialog(this,"Server
Error!","Error!",JOptionPane.ERROR_MESSAGE);
                }
              }
              if(!flag)
                JOptionPane.showMessageDialog(this,"No data
found!","Error!",JOptionPane.ERROR_MESSAGE);
            } catch (SQLException ex) {
              JOptionPane.showMessageDialog(this,"Server
Error!","Error!",JOptionPane.ERROR_MESSAGE);
            }
        }catch (ClassNotFoundException ex) {
           JOptionPane.showMessageDialog(this,"Driver
Error!","Error!",JOptionPane.ERROR_MESSAGE);
        }
    }
```

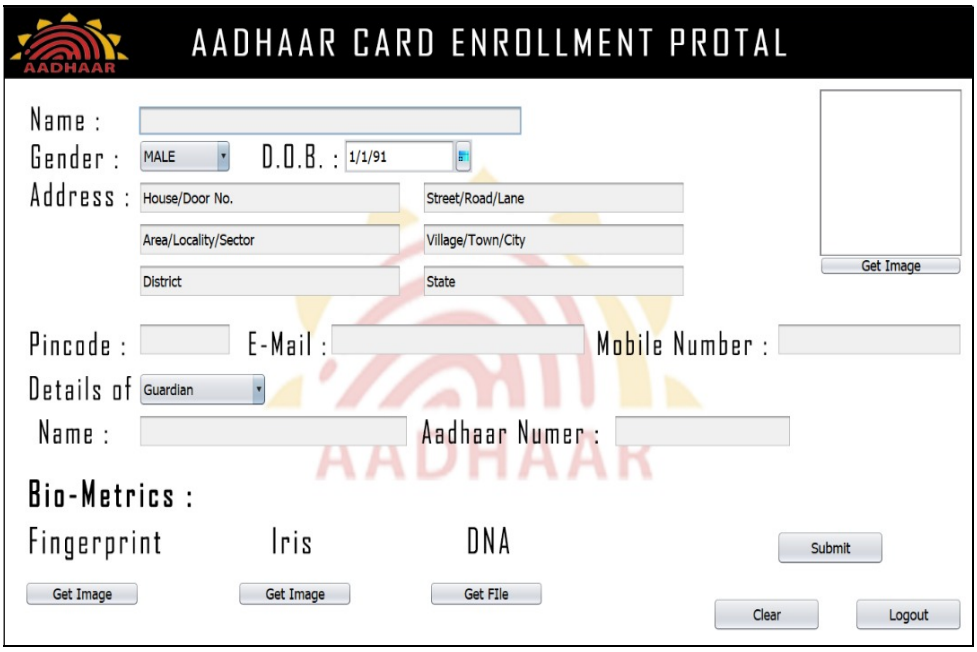
CHAPTER 8

SCREEN SHOTS

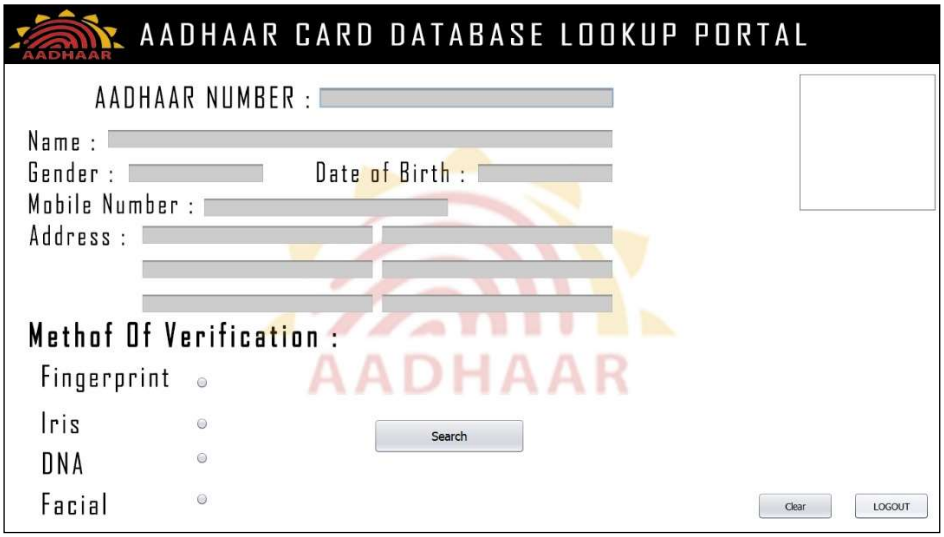
Login Page:



Enrollment Portal:



Database Lookup Portal:



CHAPTER 9

CONCLUSION

Considering the future prospects we plan to include the following features.

1. Undertake bulk orders from restaurants for special occasions and parties
2. Improve accuracy of order and delivery
3. Expanded reach
4. Release of a mobile application
5. Reduce expenses and increase revenue