**HEALTH CARE MANAGEMENT SYSTEM FOR DOCTOR**



A Course Project Report in the course

**Object Oriented Programming Concepts through java**

**Computer Science & Engineering**

**By Batch-01**

18K41A0505 AREEFA

18K41A0527 K. HARSHITH

18K41A0528 K. PRAGATHI

18K41A0529 K. RISHITHA

**Under the Guidance of**

Mr. K. Ravi Chythanya

Assoc. Prof

**Submitted on**

 APRIL, **2020**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**CERTIFICATE**

This is to certify that the **Object Oriented Programming through Java - Course Project** Report entitled **“** Health Care Management System For Doctor**”** a record of bonafide work carried out by the student(s) Areefa, K. Harshith, K. Pragathi, K. Rishitha bearing Roll No(s) 18K41A0505, 18K41A0527, 18K41A0528, 18K41A0529 during the academic year **2019-20** in partial fulfillment of the award of the degree of ***Bachelor of Technology*** in **Computer Science & Engineering** by the Jawaharlal Nehru Technological University, Hyderabad.

**Lab In-charge Head of the Department**

**ACKNOWLEDGEMENT**

I wish to take this opportunity to express my sincere and deep sense of respect to our beloved principal **Dr. V Mahesh, S R Engineering College** for providing an excellent academic atmosphere in the institution.

I express heartfelt thanks to the Head of Department, Computer Science & Engineering **Mr. A. Srinivas** for providing me with necessary infrastructure and there by giving me freedom to carry out the work.

I also thank the other staff members and friends who assisted me. Finally, I thank my parents who inspired me always to do the best.

ARREFA 18K41A0505

K. HARSHITH 18K41A0527

K. PRAGATHI 18K41A0528

K. RISHITHA 18K41A0529

**TABLE OF CONTENTS**

**TITLE PAGENO.**

1. **OBJECTIVE OF THE PROJECT 1**

1. **DEFINITIONS OF THE ELEMENTS USED 2-10**
2. **DESIGN 11-16**
   1. SCREENS
3. **IMPLEMENTATION 17-74**
   1. CODE
4. **RESULT SCREENS 75-83**
   1. MODULE : STAFF
   2. MODULE: DOCTOR ACCESS
5. **CONCLUSION 84**

**ABSTRACT**

The keystone concern that avails between a patient-doctor relation is the utmost health care which usually the patient expects and at the same time doctor puts all his efforts to meet the expectations relatively. Health care of the patient comprises gathering the data of the diagnosis to understanding the patient’s health issue, planning the treatment to accomplishing the compliance and healing the problem at the same time guiding for a suitable diet and providing the extreme support. The major hectic flaw comes in maintaining and managing the patient’s health records. In many hospitals where patient’s approaches for the routine course for their treatment of their health issue, face problems in maintaining all the reports and doctor prescriptions. At every appointment, they need to carry their health book or repeat detailing about their issue to the particular doctor. Thus, this is creating a traumatic confusion both for the patient and the doctor.

There is more scope to make these health care strategies hand in hand in a smart. Hence, the Smart- Health Care ManagementSystem portal that acts as a medium between the patient and doctor ensuring the strategies of health care. It helps doctors to access the patient’s health records easily on one platform.

Thisapplication proposed to be implemented for an Online health care management system should implement services for two types of users like Doctors, Staff. The staff is the ones at the registration desk in the hospital who are involved in the process for New Patient Registration. This health care portal would create a feasible access to the health profiles of every individual patient and overall details of the treatment including diet. Therefore a feasible platform for maintaining health records and smart approach alike from traditional doctor consultancy.

1. **OBJECTIVE OF THE PROJECT**

The main aim of this project is to design an effective application portal for ensuring health care The main aim of this project is to design an effective application portal for ensuring health care management strategies in a clinic where the patients' health issues and the records of the treatment can be maintained so that a proper improvement in health can be assessed easily for the doctors and the patient need not to detail about his/her health issue to the doctor every time specifically.

The health care management portal helps to minimize all the concerns and confusion that generally arise between a doctor and patient. An individual patient’s complete details regarding his/her health problem and diagnosis via treatment plans will be on whole on a single platform and helps the doctor to meet the expectations of the patient.

The health care management portal must include the modules for both the doctor and the staff by supporting the following strategies:

* Allow New patients to register by selecting the doctors from the doctors available list.
* Doctors to be able to view the patients' list.
* Easy access for a doctor to get his/ her patient’s health issue history and about the previous appointment prescription by him/her by their ID.
* Display of particular patient profile based on their Id such that no need for the patient to explain about his/her health issue again and again.
* The patient’s profile must be editable only by the doctor.
* For the next appointment, the profile will be updatable (To add any other health issues or medicines)
* The patient to be guided about his health care and diet.
* The printed diet prescription avoids the conflict by doctor’s hand-writing.

Thus, the main objective of this project is to integrate the doctor-patient relationship regarding health care on a single platform. This even manages the patient registration process as an additional service.

1. **DEFINITIONS OF ELEMENTS USED IN THE PROJECT**

We used the Eclipse IDE to build this application. In which we approached the Window Builder to design our screens involved relatively including the elements and Swing concepts. Files are the medium that we used in storing the data and retrieving the data respectively in an efficient way.

**SWINGS:**

It is used to create window based application and is built on the top of AWT (Abstract Windowing Toolkit) API (Application Programming Interface) and is entirely written in java.

**The hierarchy of java swing API is given below:**

****

**CONTAINERS:**

The Container is a component in AWT that can contain another components like buttons, textfields, labels etc. The classes that extends Container class are known as container such as Frame, Dialog and Panel.

* **Frames:**

In java, a frame is a window that has nice borders, various buttons along the top border, and other features. A frame is a container object, so GUI components can be placed in it.

* **JPanel:**

JPanel, a part of the Java Swing package, is a container that can store a group of components. The main task of JPanel is to organize components, various layouts can be set in JPanel which provide better organization of components, however, it does not have a title bar. It inherits the JComponents class.

* **JScrollPane:**

A JscrollPane is used to make a scrollable view of a component. When screen size is limited, we use a scroll pane to display a large component or a component whose size can change dynamically.

* **JLayeredPane:**

The JLayeredPane class is used to add depth to swing container. It is used to provide a third dimension for positioning components and divide the depth-range into several different layers. A layered pane is a Swing container that is used to hold the various components using the concept of layers. The components present in the upper layer overlaps the components present in the lower layer. The layered pane is created using the JLayeredPane class. the only constructor of this class is JLayeredPane ().

**LAYOUT - Absolute Layout:**

The LayoutManagers are used to arrange components in a particular manner. LayoutManager is an interface that is implemented by all the classes of layout managers

**COMPONENTS:**

The javax.swing package provides classes for java swing API such as JButton, JTextField, JTextArea, JRadioButton, JCheckbox, JLabel, JTextField, etc. Below are the ones which we have used:

* **JLabel:**

JLabel is a class of java Swing. JLabel is used to display a short string or an image icon. JLabel can display text, image or both. JLabel is only a display of text or image and it cannot get focus. JLabel is inactive to input events such a mouse focus or keyboard focus. By default labels are vertically centered but the user can change the alignment of the label.

Constructors of JLabel are:

**JLabel():** creates a blank label with no text or image in it.

**JLabel(String s):** creates a new label with the string specified.

**JLabel(Icon i):** creates a new label with an image on it

* **JCheckBox:**

JCheckBox is a part of the Java Swing package. JCheckBox can be selected or deselected. It displays its state to the user. JCheckBox is an implementation of the checkbox . JCheckBox inherits the JToggleButton class. The JCheckBox class is used to create a checkbox. It is used to turn an option on (true) or off (false).

Constructors of JCheckBox are:

1. **JCheckBox()**: creates a new checkbox with no text or icon.
2. **JCheckBox(Icon i** : creates a new checkbox with the icon specified.

* **JTable:**

The JTable class is a part of Java Swing Package and is generally used to display or edit two-dimensional data that is having both rows and columns. It is similar to a spreadsheet. This arranges data in a tabular form.

The constructors of JTable are:

1. **JTable():**A table is created with empty cells.
2. **JTable(int rows, int cols):**Creates a table of size rows \* columns.

* **JSeparator:**

JSeparator is a part of the Java Swing framework. It is used to create a dividing line between two components. More specifically, it is mainly used to create dividing lines between menu items in a JMenu. In JMenu or JPopupMenu addSeparartor function can also be used to create a separator.

The constructor of JSeparator are:

1. **separator()**: Creates a new horizontal separator.
2. **JSeparator(int o)**: Creates a new separator with the specified horizontal or vertical orientation.

* **JTextField:**

JTextField is a part of javax.swing package. The class JTextField is a component that allows editing of a single line of text. JTextField inherits the JTextComponent class and uses the interface SwingConstants.

The constructor of JTextField are :

1. **JTextField()** : constructor that creates a new TextField
2. **JTextField(int columns)**: constructor that creates a new empty TextField with the specified number of columns.
3. **JTextField(String text)**: constructor that creates a new empty text field initialized with the given string.

* **JButton:**

The JButton class is used to create a labeled button that has platform independent implementation. The application result in some action when the button is pushed. It inherits AbstractButton class.

Constructors of JButton are:

1. **JButton():**It create a new button without any text.
2. **JButton(String s):**It create a new button with specific text.

* **JRadioButton:**

The JRadioButton class is used to create a radio button. It is used to choose one option from multiple options. It is widely used in exam systems or quiz. It should be added to ButtonGroup to select one radio button only.

Constructor of JRadioButton are:

**JRadioButton() :** Creates a unselected RadioButton with no text.

* **JTextArea:**

JTextArea is a part of java Swing package . It represents a multi-line area that displays text. The text in JTextArea can be set to different available fonts and can be appended to a new text. A text area can be customized to the need of the user .

Constructors of JTextArea are:

1. **JTextArea() :**constructs a new blank text area .
2. **JTextArea(String s):** constructs a new text area with a given initial text.

* **JPasswordField:**

PasswordField is a part of javax.swing package . The class JPasswordField is a component that allows editing of a single line of text where the view indicates that something was typed by does not show the actual characters. JPasswordField inherits the JTextField class in javax.swing package.

Constructors of JPasswordField:

1. **JPasswordField()**: constructor that creates a new password field
2. **JPasswordField(int columns)**: constructor that creates a new empty PasswordField with a specified number of columns.

**JPasswordField(String Password)**: constructor that creates a new empty Password field initialized with the given string.

* **JScrollBox:**

The object of the Scrollbar class is used to add a horizontal and vertical scrollbar. The scrollbar is a GUI component that allows us to see an invisible number of rows and columns.

The constructor of JScrollBar are:

**JScrollBar():** Creates a vertical scrollbar with the initial values.

**ACTIONLISTENER:**

* **KeyListener:**

The Java KeyListener is notified whenever you change the state of the key. It is notified against KeyEvent. The KeyListener interface is found in java.awt.event package. It has three methods.

## Methods of KeyListener interface:

The signature of 3 methods found in KeyListener interface are given below:

1. **public abstract void** keyPressed(KeyEvent e).
2. **public abstract void** keyReleased(KeyEvent e).
3. **public abstract void** keyTyped(KeyEvent e).

* **Mouse Listener:**

The Java MouseListener is notified whenever you change the state of the mouse. It is notified against MouseEvent. The MouseListener interface is found in java.awt.event package. It has five methods.

## Methods of MouseListener interface:

The signature of 5 methods found in the MouseListener interface are given below:

1. **public abstract void** mouseClicked(MouseEvent e).
2. **public abstract void** mouseEntered(MouseEvent e).
3. **public abstract void** mouseExited(MouseEvent e).
4. **public abstract void** mousePressed(MouseEvent e).
5. **public abstract void** mouseReleased(MouseEvent e).

* **Iterator:**

It is a **universal** iterator as we can apply it to any Collection object. By using the Iterator, we can perform both read and remove operations. It is an improved version of Enumeration with the additional functionality of the remove-ability of an element. Iterator must be used whenever we want to enumerate elements in all Collection framework implemented interfaces like Set, List, Queue, Deque and also in all implemented classes of Map interface. The iterator is the **only** cursor available for the entire collection framework.

Iterator object can be created by calling *iterator()* method present in the Collection interface.

**IO STREAMS:**

The java.io package contains nearly every class you might ever need to perform input and output (I/O) in Java. All these streams represent an input source and an output destination. The stream in the java.io package supports many data such as primitives, objects, localized characters, etc.

## Byte Streams:

## Java byte streams are used to perform input and output of 8-bit bytes. Though there are many classes related to byte streams the most frequently used classes are, FileInputStream and FileOutputStream.

## FileInputStream:

## This stream is used for reading data from the files. Objects can be created using the keyword new and there are several types of constructors available.

## FileOutputStream:

## FileOutputStream is used to create a file and write data into it. The stream would create a file, if it doesn't already exist, before opening it for output.

## Character Streams:

## Java Byte streams are used to perform input and output of 8-bit bytes, whereas Java Character streams are used to perform input and output for 16-bit Unicode. Though there are many classes related to character streams but the most frequently used classes are, FileReader and FileWriter. Though internally FileReader uses FileInputStream and FileWriter uses FileOutputStream but here the major difference is that FileReader reads two bytes at a time and FileWriter writes two bytes at a time.

**EXCEPTIONS HANDLING IN JAVA:**

1. **Checked Exception**:The classes which directly inherit Throwable class except RuntimeException and Error are known as checked exceptions e.g. IOException, SQLException, etc. Checked exceptions are checked at compile-time.
2. **Unchecked Exception:** The classes which inherit RuntimeException are known as unchecked exceptions e.g. ArithmeticException, NullPointerException, ArrayIndexOutOfBoundsException, etc. Unchecked exceptions are not checked at compile-time, but they are checked at runtime.
3. **Error:**Error is irrecoverable e.g. OutOfMemoryError, VirtualMachineError, AssertionError etc.

**SERIALIZATION:**

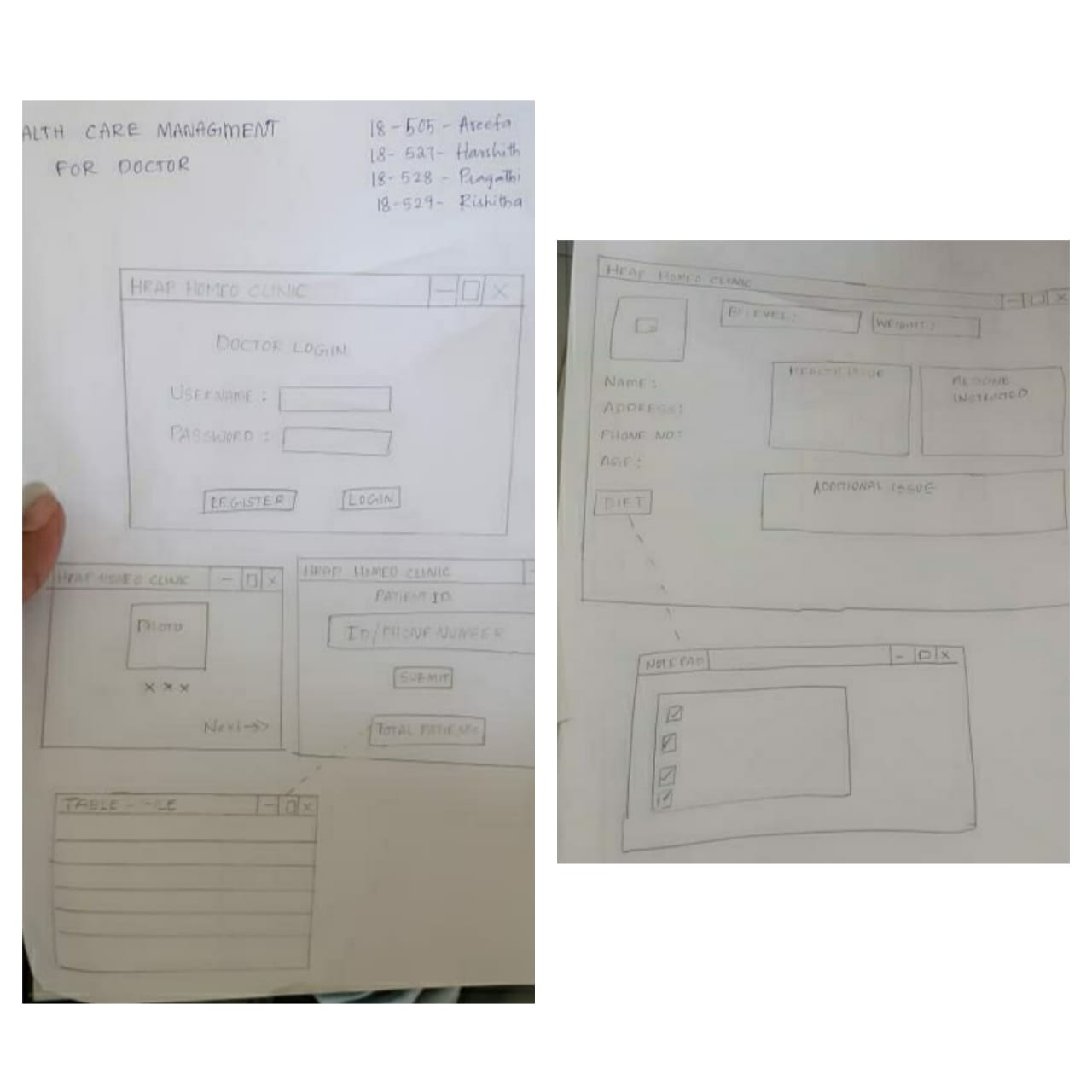
**Serialization in Java** is a mechanism of writing the state of an object into a byte-stream.  
The reverse operation of serialization is called deserialization where the byte-stream is converted into an object. The serialization and deserialization process is platform-independent, it means you can serialize an object in a platform and deserialize in different platforms.

For serializing the object, we call the **writeObject()** method ObjectOutputStream, and for Deserialization we call the **readObject()** method of ObjectInputStream class.

1. **DESIGN**
   1. **SCREENS:**

**EXPECTATION**

First we had drawn the models of the screens that needed to be implemented as part of the application.

****These are just rough sketches of frames that we planned.

**Fig 3.1.i: Rough designs of frames**

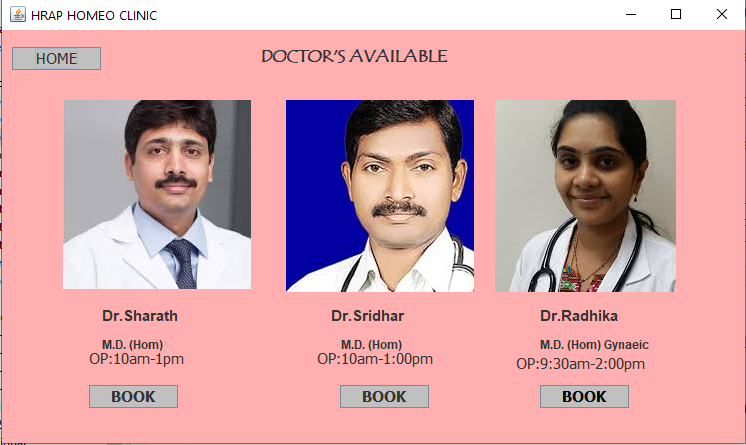
**ORIGINAL**

1. **HOME PAGE:**

**Fig 3.1.i: Home page**

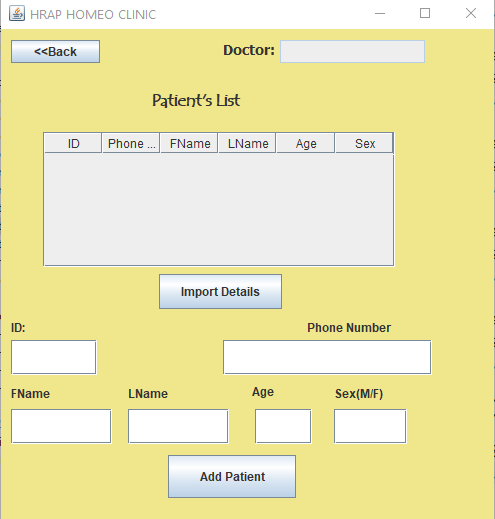
**MODULE WISE SCREENS:**

1. **STAFF (**If the patient is new then, he/she will be registered at the registration desk)

* New patients must be able to select the doctor to whom they want to select.

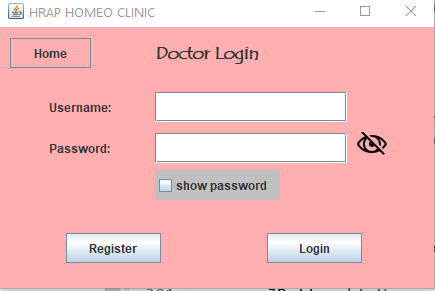
**Fig 3.1.ii: Doctor’s display**

* New patient registration by giving the details:

****

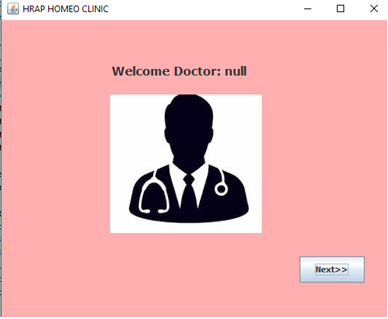
**Fig 3.1.iii: New Patient Register**

1. **DOCTOR LOGIN**

****

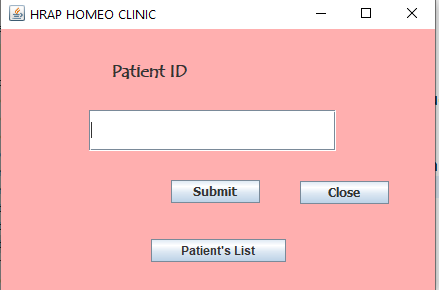
**Fig 3.1.iv: Doctor Login**

1. **Doctor Photo Display:**

****

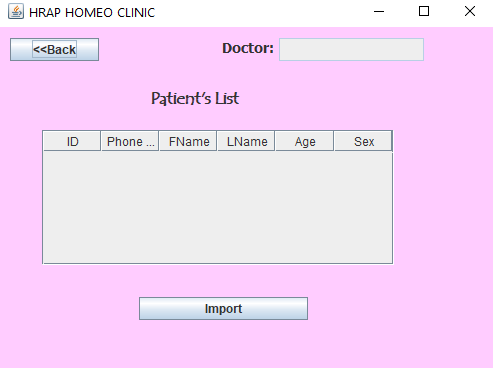
**Fig 3.1.v: Doctor Display (when login is successful)**

1. **Patient Login:**

****

**Fig 3.1.vi: Patient Login**

1. **Total Patients’ List Display:**

****

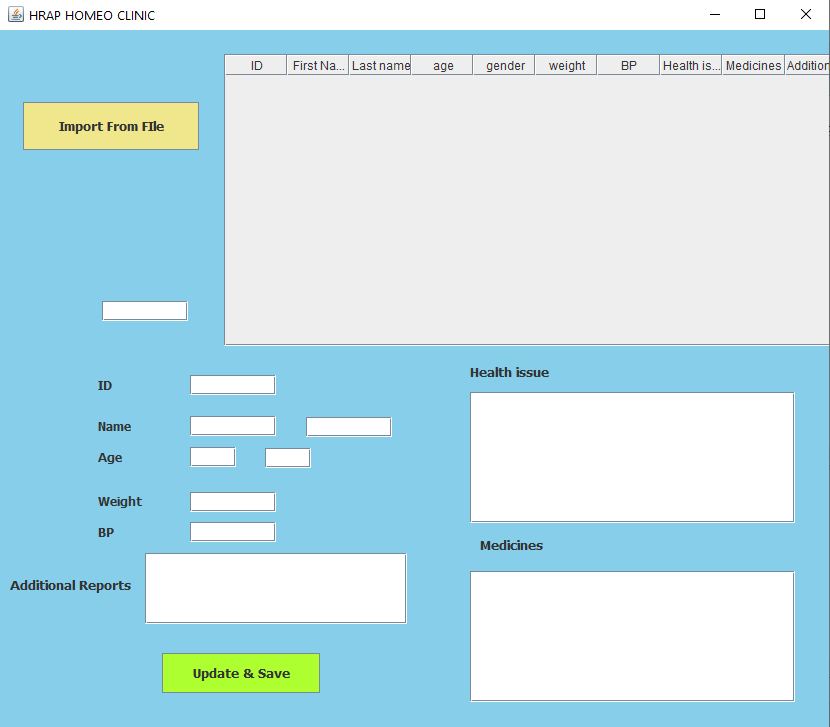
**Fig 3.1.vii: Patient List Display**



1. **Particular Patient Profile**

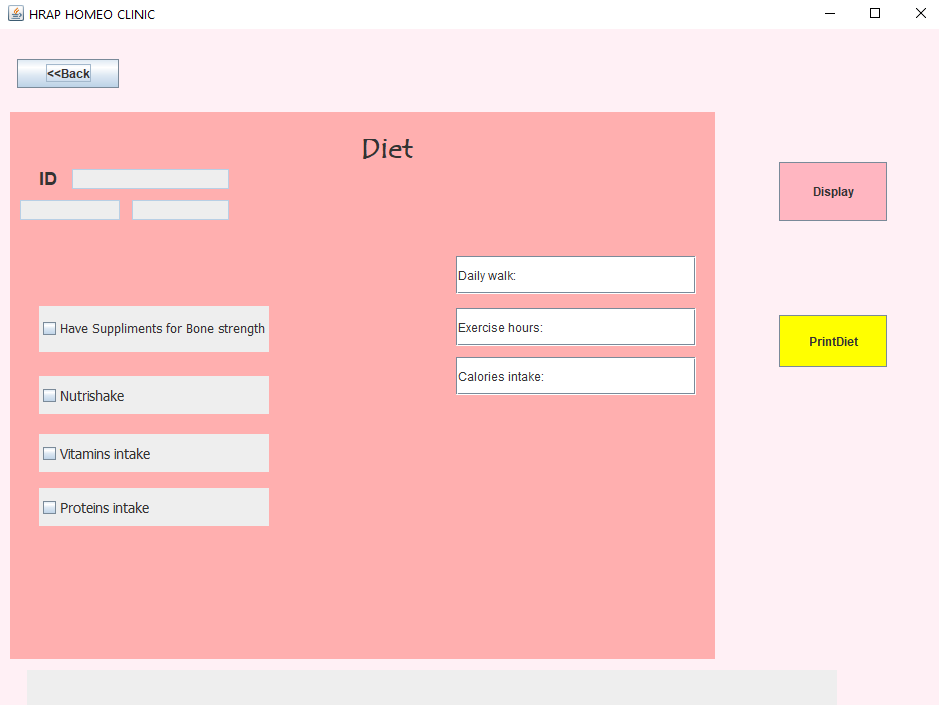
**Fig 3.1.viii: Patient Profile**

1. **Update Patient Profile:**



**Fig 3.1.ix: Update Patient Profile**

1. **DIET:**

****

**Fig 3.1.x: Diet (Set)**

1. **IMPLEMENTATION**
   1. **CODE:**

**4.1.i. Home Page:**

**package** Project;

**import** java.awt.EventQueue;

**import** java.awt.Image;

**import** javax.swing.ImageIcon;

**import** javax.swing.JFrame;

**import** java.awt.Color;

**import** javax.swing.JLabel;

**import** java.awt.Font;

**import** javax.swing.JButton;

**import** java.awt.event.ActionListener;

**import** java.awt.event.ActionEvent;

**public** **class** HomePage {

**private** JFrame frmHrapHomeoClinic;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

HomePage window = **new** HomePage();

window.frmHrapHomeoClinic.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** HomePage() {

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

**private** **void** initialize() {

frmHrapHomeoClinic = **new** JFrame();

frmHrapHomeoClinic.getContentPane().setBackground(Color.***PINK***);

frmHrapHomeoClinic.getContentPane().setForeground(Color.***WHITE***);

frmHrapHomeoClinic.setTitle("HOME PAGE");

frmHrapHomeoClinic.setBounds(100, 100, 700, 453);

frmHrapHomeoClinic.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frmHrapHomeoClinic.getContentPane().setLayout(**null**);

frmHrapHomeoClinic.setVisible(**true**);

JLabel lblNewLabel = **new** JLabel("");

Image img=**new** ImageIcon(**this**.getClass().getResource("/8.jpeg")).getImage();

lblNewLabel.setIcon(**new** ImageIcon(img));

lblNewLabel.setBounds(211, 77, 291, 217);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel);

JLabel lblNewLabel\_1 = **new** JLabel("HRAP HOMEO CLINIC");

lblNewLabel\_1.setFont(**new** Font("Verdana", Font.***BOLD*** | Font.***ITALIC***, 25));

lblNewLabel\_1.setBounds(174, 11, 350, 55);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_1);

JButton btnNewButton = **new** JButton("New Patient?");

btnNewButton.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frmHrapHomeoClinic.setVisible(**false**);

**new** DoctorsList();

}

});

btnNewButton.setForeground(Color.***BLACK***);

btnNewButton.setBackground(Color.***LIGHT\_GRAY***);

btnNewButton.setFont(**new** Font("Tahoma", Font.***BOLD*** | Font.***ITALIC***, 15));

btnNewButton.setBounds(110, 322, 175, 42);

frmHrapHomeoClinic.getContentPane().add(btnNewButton);

JButton btnNewButton\_1 = **new** JButton("DOCTOR'S LOGIN");

btnNewButton\_1.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frmHrapHomeoClinic.setVisible(**false**);

**new** DoctorLoginPage();

}

});

btnNewButton\_1.setForeground(Color.***BLACK***);

btnNewButton\_1.setBackground(Color.***LIGHT\_GRAY***);

btnNewButton\_1.setFont(**new** Font("Tahoma", Font.***BOLD*** | Font.***ITALIC***, 15));

btnNewButton\_1.setBounds(403, 321, 185, 44);

frmHrapHomeoClinic.getContentPane().add(btnNewButton\_1);

}

}

**MODULE – STAFF (Register the new patient)**

**4.1. ii. New Patient Register – Choice for patient to select doctor**

**(Doctors Available Display Frame)**

**package** Project;

**import** java.awt.EventQueue;

**import** java.awt.Graphics;

**import** java.awt.Image;

**import** java.awt.image.BufferedImage;

**import** java.awt.image.ImageObserver;

**import** java.io.IOException;

**import** javax.imageio.ImageIO;

**import** javax.swing.ImageIcon;

**import** javax.swing.JFrame;

**import** java.awt.Color;

**import** javax.swing.JLabel;

**import** java.awt.Font;

**import** javax.swing.JButton;

**import** java.awt.event.ActionListener;

**import** java.awt.event.ActionEvent;

**import** java.awt.Button;

**public** **class** DoctorsList {

**private** JFrame frmHrapHomeoClinic;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

DoctorsList window = **new** DoctorsList();

window.frmHrapHomeoClinic.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** DoctorsList() {

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

**private** **void** initialize() {

frmHrapHomeoClinic = **new** JFrame("HRAP HOMEO CLINIC");

frmHrapHomeoClinic.getContentPane().setBackground(Color.***PINK***);

frmHrapHomeoClinic.getContentPane().setLayout(**null**);

frmHrapHomeoClinic.setVisible(**true**);

String name;

JLabel lblNewLabel = **new** JLabel("");

Image img=**new** ImageIcon(**this**.getClass().getResource("/p3.jpeg")).getImage();

lblNewLabel.setIcon(**new** ImageIcon(img));

lblNewLabel.setBounds(61, 70, 188, 189);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel);

JLabel lblNewLabel\_1 = **new** JLabel("Dr.Sharath");

lblNewLabel\_1.setFont(**new** Font("SansSerif", Font.***BOLD***, 15));

lblNewLabel\_1.setBounds(100, 270, 98, 31);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_1);

JLabel lblNewLabel\_2 = **new** JLabel("OP:10am-1pm");

lblNewLabel\_2.setFont(**new** Font("Tahoma", Font.***PLAIN***, 15));

lblNewLabel\_2.setBounds(87, 313, 111, 31);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_2);

JButton btnNewButton = **new** JButton("BOOK");

btnNewButton.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frmHrapHomeoClinic.setVisible(**false**);

**new** PatientDetailsPage1("Dr.Sharath");

}

});

btnNewButton.setBackground(Color.***LIGHT\_GRAY***);

btnNewButton.setFont(**new** Font("Tahoma", Font.***BOLD***, 15));

btnNewButton.setBounds(87, 355, 89, 23);

frmHrapHomeoClinic.getContentPane().add(btnNewButton);

JLabel lblNewLabel\_3 = **new** JLabel("");

Image img1=**new** ImageIcon(**this**.getClass().getResource("/doc.jpg")).getImage();

lblNewLabel\_3.setIcon(**new** ImageIcon(img1));

lblNewLabel\_3.setBounds(284, 70, 188, 192);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_3);

JLabel lblNewLabel\_4 = **new** JLabel("Dr.Sridhar");

lblNewLabel\_4.setFont(**new** Font("SansSerif", Font.***BOLD***, 15));

lblNewLabel\_4.setBounds(329, 273, 98, 24);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_4);

JLabel lblNewLabel\_5 = **new** JLabel("OP:10am-1:00pm");

lblNewLabel\_5.setFont(**new** Font("Tahoma", Font.***PLAIN***, 15));

lblNewLabel\_5.setBounds(315, 313, 128, 31);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_5);

JButton btnNewButton\_1 = **new** JButton("BOOK");

btnNewButton\_1.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frmHrapHomeoClinic.setVisible(**false**);

**new** PatientDetailsPage1("Dr.Sridhar");

}

});

btnNewButton\_1.setBackground(Color.***LIGHT\_GRAY***);

btnNewButton\_1.setFont(**new** Font("Tahoma", Font.***BOLD***, 15));

btnNewButton\_1.setBounds(338, 355, 89, 23);

frmHrapHomeoClinic.getContentPane().add(btnNewButton\_1);

JLabel lblNewLabel\_6 = **new** JLabel("DOCTOR'S AVAILABLE");

lblNewLabel\_6.setFont(**new** Font("Tempus Sans ITC", Font.***BOLD***, 18));

lblNewLabel\_6.setBounds(258, 11, 225, 30);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_6);

JLabel lblNewLabel\_7 = **new** JLabel("");

Image img2=**new** ImageIcon(**this**.getClass().getResource("/p2.jpeg")).getImage();

lblNewLabel\_7.setIcon(**new** ImageIcon(img2));

lblNewLabel\_7.setBounds(493, 70, 181, 192);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_7);

JLabel lblNewLabel\_8 = **new** JLabel("Dr.Radhika");

lblNewLabel\_8.setFont(**new** Font("SansSerif", Font.***BOLD***, 15));

lblNewLabel\_8.setBounds(538, 275, 109, 21);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_8);

JLabel lblNewLabel\_9 = **new** JLabel("OP:9:30am-2:00pm");

lblNewLabel\_9.setFont(**new** Font("Tahoma", Font.***PLAIN***, 15));

lblNewLabel\_9.setBounds(514, 323, 155, 21);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_9);

JButton btnNewButton\_2 = **new** JButton("BOOK");

btnNewButton\_2.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frmHrapHomeoClinic.setVisible(**false**);

**new** PatientDetailsPage1("Dr.Radhika");

}

});

btnNewButton\_2.setForeground(Color.***BLACK***);

btnNewButton\_2.setBackground(Color.***LIGHT\_GRAY***);

btnNewButton\_2.setFont(**new** Font("Tahoma", Font.***BOLD***, 15));

btnNewButton\_2.setBounds(538, 355, 89, 23);

frmHrapHomeoClinic.getContentPane().add(btnNewButton\_2);

JLabel lblNewLabel\_10 = **new** JLabel("M.D. (Hom) Gynaeic");

lblNewLabel\_10.setBounds(538, 304, 109, 21);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_10);

JLabel lblNewLabel\_10\_1 = **new** JLabel("M.D. (Hom)");

lblNewLabel\_10\_1.setBounds(339, 304, 89, 21);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_10\_1);

JLabel lblNewLabel\_10\_1\_1 = **new** JLabel("M.D. (Hom)");

lblNewLabel\_10\_1\_1.setBounds(100, 304, 89, 21);

frmHrapHomeoClinic.getContentPane().add(lblNewLabel\_10\_1\_1);

JButton btnBack = **new** JButton("HOME");

btnBack.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frmHrapHomeoClinic.setVisible(**false**);

**new** HomePage();

}

});

btnBack.setFont(**new** Font("Tahoma", Font.***PLAIN***, 15));

btnBack.setBackground(Color.***LIGHT\_GRAY***);

btnBack.setBounds(10, 17, 89, 23);

frmHrapHomeoClinic.getContentPane().add(btnBack);

//frmHrapHomeoClinic.getContentPane().add(button);

frmHrapHomeoClinic.setTitle("HRAP HOMEO CLINIC");

frmHrapHomeoClinic.setBounds(100, 100, 759, 452);

frmHrapHomeoClinic.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}}

**4.1. iii. Patient Register (Entry Details)**

**package** Project;

**import** java.awt.EventQueue;

**import** javax.swing.JFrame;

**import** javax.swing.JScrollPane;

**import** javax.swing.JTable;

**import** javax.swing.table.DefaultTableModel;

**import** javax.swing.JTextField;

**import** javax.swing.JButton;

**import** java.awt.event.ActionListener;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.File;

**import** java.io.FileNotFoundException;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** java.lang.System.Logger.Level;

**import** java.util.logging.Logger;

**import** java.awt.event.ActionEvent;

**import** java.awt.Color;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** java.awt.Font;

**public** **class** PatientDetailsPage1 {

**protected** **static** **final** java.util.logging.Level ***Level*** = **null**;

JFrame frame;

**private** JTable table;

**private** JTextField first;

**private** JTextField second;

**private** JTextField third;

**private** JButton btnImportx;

**private** JTextField textField;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

PatientDetailsPage1 window = **new** PatientDetailsPage1();

window.frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** PatientDetailsPage1() {

initialize();

}

String name;

**private** JTextField fourth;

**private** JTextField fifth;

**private** JLabel lblNewLabel\_2;

**private** JLabel lblNewLabel\_3;

**private** JLabel lblNewLabel\_4;

**private** JLabel lblNewLabel\_5;

**private** JTextField sixth;

**private** JTextField textField\_1;

**private** JTextField textField\_3;

**public** PatientDetailsPage1(String name) {

**this**.name=name;

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

**private** **void** initialize() {

frame = **new** JFrame("HRAP HOMEO CLINIC");

frame.getContentPane().setBackground(**new** Color(240, 230, 140));

frame.setBounds(100, 100, 509, 530);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.getContentPane().setLayout(**null**);

frame.setVisible(**true**);

JScrollPane scrollPane = **new** JScrollPane();

scrollPane.setBounds(42, 103, 352, 135);

frame.getContentPane().add(scrollPane);

table = **new** JTable();

table.setModel(**new** DefaultTableModel(

**new** Object[][] {

},

**new** String[] {

"ID", "Phone Number", "FName", "LName", "Age", "Sex"

}

));

scrollPane.setViewportView(table);

first = **new** JTextField();

first.setBounds(10, 311, 86, 35);

frame.getContentPane().add(first);

first.setColumns(10);

second = **new** JTextField();

second.setColumns(10);

second.setBounds(222, 311, 209, 35);

frame.getContentPane().add(second);

third = **new** JTextField();

third.setColumns(10);

third.setBounds(10, 380, 101, 35);

frame.getContentPane().add(third);

JButton btnRun = **new** JButton("Add Patient");

btnRun.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

DefaultTableModel model=(DefaultTableModel)table.getModel();

model.addRow(**new** Object [] {Integer.*parseInt*(first.getText()), second.getText(), third.getText(),fourth.getText(),fifth.getText(),sixth.getText()});

**try** {

**if**(name.equals("Dr.Sridhar")){

FileWriter fw = **new** FileWriter("profile2",**true**);

BufferedWriter bw = **new** BufferedWriter(fw);

//bw.newLine();

bw.write(first.getText());

bw.write(" ");

bw.write(third.getText());

bw.write(" ");

bw.write(fourth.getText());

bw.write(" ");

bw.write(fifth.getText());

bw.write(" ");

bw.write(sixth.getText());

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.newLine();

bw.close();}

**else** **if**(name.equals("Dr.Sharath")) {

FileWriter fw = **new** FileWriter("profile1",**true**);

BufferedWriter bw = **new** BufferedWriter(fw);

//bw.newLine();

bw.write(first.getText());

bw.write(" ");

bw.write(third.getText());

bw.write(" ");

bw.write(fourth.getText());

bw.write(" ");

bw.write(fifth.getText());

bw.write(" ");

bw.write(sixth.getText());

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.newLine();

bw.close();

}

**else** **if**(name.equals("Dr.Radhika")) {

FileWriter fw = **new** FileWriter("profile3",**true**);

BufferedWriter bw = **new** BufferedWriter(fw);

bw.write(first.getText());

bw.write(" ");

bw.write(third.getText());

bw.write(" ");

bw.write(fourth.getText());

bw.write(" ");

bw.write(fifth.getText());

bw.write(" ");

bw.write(sixth.getText());

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.write("Null");

bw.write(" ");

bw.newLine();

bw.close();

}

}**catch**(IOException ex) {

Logger.*getLogger*(PatientDetailsPage1.**class**.getName()).log(***Level***,**null**,ex);

ex.printStackTrace();

}

first.setText(" ");

second.setText(" ");

third.setText(" ");

fourth.setText(" ");

fifth.setText(" ");

sixth.setText(" ");

}

});

btnRun.setBounds(167, 426, 128, 43);

frame.getContentPane().add(btnRun);

JButton btnSave = **new** JButton("<<Back");

btnSave.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

**try** {

**if**(name.equals("Dr.Sharath")) {

FileWriter fw = **new** FileWriter("patients1");

BufferedWriter bw = **new** BufferedWriter(fw);

**for**(**int** i=0;i<table.getRowCount();i++) {

**for**(**int** j=0;j<table.getColumnCount();j++) {

bw.write(table.getValueAt(i, j).toString()+" ");

}

bw.newLine();

} bw.close();

fw.close();

frame.setVisible(**false**);

**new** HomePage();}

**else** **if**(name.equals("Dr.Sridhar")) {

FileWriter fw = **new** FileWriter("patientssridhar");

BufferedWriter bw = **new** BufferedWriter(fw);

**for**(**int** i=0;i<table.getRowCount();i++) {

**for**(**int** j=0;j<table.getColumnCount();j++) {

bw.write(table.getValueAt(i, j).toString()+" ");

}

bw.newLine();

} bw.close();

fw.close();

frame.setVisible(**false**);

**new** HomePage();

}

**else** **if**(name.equals("Dr.Radhika")) {

FileWriter fw = **new** FileWriter("patients2");

BufferedWriter bw = **new** BufferedWriter(fw);

**for**(**int** i=0;i<table.getRowCount();i++) {

**for**(**int** j=0;j<table.getColumnCount();j++) {

bw.write(table.getValueAt(i, j).toString()+" ");

}

bw.newLine();

} bw.close();

fw.close();

frame.setVisible(**false**);

**new** HomePage();

}

}

**catch**(IOException ex) {

Logger.*getLogger*(PatientDetailsPage1.**class**.getName()).log(***Level***,**null**,ex);

ex.printStackTrace();

}

}

});

btnSave.setBounds(10, 11, 89, 23);

frame.getContentPane().add(btnSave);

JLabel lblNewLabel = **new** JLabel("Patient's List");

lblNewLabel.setBackground(**new** Color(219, 112, 147));

lblNewLabel.setFont(**new** Font("Tempus Sans ITC", Font.***BOLD***, 17));

lblNewLabel.setBounds(150, 55, 145, 32);

frame.getContentPane().add(lblNewLabel);

btnImportx = **new** JButton("Import Details");

btnImportx.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

**try** {

**if**(name.equals("Dr.Sharath")) {

System.***out***.println("Hii");

FileReader fr = **new** FileReader("patients1");

BufferedReader br = **new** BufferedReader(fr);

DefaultTableModel model = (DefaultTableModel)table.getModel();

Object[] lines = br.lines().toArray();

**for**(**int** i=0;i<lines.length;i++) {

**if**(i==(lines.length)-1) {

System.***out***.println(lines[i].toString());

String str=lines[i].toString();

System.***out***.println(str);

//String s[]=lines

**int** len=str.length();

System.***out***.println(len);

**int** j=0;

**int** a[] = **null**;

**for**(**int** k=0;k<6;k++)

{

System.***out***.println(str.charAt(k));

}

}

String[] row= lines[i].toString().split(" ");

model.addRow(row);

}}

**else** **if**(name.equals("Dr.Sridhar")) {

FileReader fr = **new** FileReader("patientssridhar");

BufferedReader br = **new** BufferedReader(fr);

System.***out***.println("Hiiloo");

DefaultTableModel model = (DefaultTableModel)table.getModel();

Object[] lines = br.lines().toArray();

**for**(**int** i=0;i<lines.length;i++) {

String[] row= lines[i].toString().split(" ");

model.addRow(row);

}

}

**else** **if**(name.equals("Dr.Radhika")) {

FileReader fr = **new** FileReader("patients2");

BufferedReader br = **new** BufferedReader(fr);

DefaultTableModel model = (DefaultTableModel)table.getModel();

Object[] lines = br.lines().toArray();

**for**(**int** i=0;i<lines.length;i++) {

String[] row= lines[i].toString().split(" ");

model.addRow(row);

}

}

} **catch** (FileNotFoundException e1) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

}

});

btnImportx.setBounds(158, 245, 123, 35);

frame.getContentPane().add(btnImportx);

textField = **new** JTextField();

textField.setEditable(**false**);

textField.setText(name);

textField.setBounds(279, 11, 145, 23);

frame.getContentPane().add(textField);

textField.setColumns(10);

JLabel lblDoctor = **new** JLabel("Doctor:");

lblDoctor.setFont(**new** Font("Tahoma", Font.***BOLD***, 14));

lblDoctor.setBounds(222, 9, 73, 23);

frame.getContentPane().add(lblDoctor);

fourth = **new** JTextField();

fourth.setColumns(10);

fourth.setBounds(127, 380, 101, 35);

frame.getContentPane().add(fourth);

fifth = **new** JTextField();

fifth.setColumns(10);

fifth.setBounds(254, 380, 57, 35);

frame.getContentPane().add(fifth);

JLabel lblNewLabel\_1 = **new** JLabel("ID:");

lblNewLabel\_1.setBounds(10, 291, 46, 14);

frame.getContentPane().add(lblNewLabel\_1);

JLabel lblNewLabel\_1\_1 = **new** JLabel("Phone Number");

lblNewLabel\_1\_1.setBounds(306, 291, 88, 14);

frame.getContentPane().add(lblNewLabel\_1\_1);

lblNewLabel\_2 = **new** JLabel("FName");

lblNewLabel\_2.setBounds(10, 357, 46, 14);

frame.getContentPane().add(lblNewLabel\_2);

lblNewLabel\_3 = **new** JLabel("LName");

lblNewLabel\_3.setBounds(127, 357, 46, 14);

frame.getContentPane().add(lblNewLabel\_3);

lblNewLabel\_4 = **new** JLabel("Age");

lblNewLabel\_4.setBounds(251, 355, 46, 14);

frame.getContentPane().add(lblNewLabel\_4);

lblNewLabel\_5 = **new** JLabel("Sex(M/F)");

lblNewLabel\_5.setBounds(333, 357, 61, 14);

frame.getContentPane().add(lblNewLabel\_5);

sixth = **new** JTextField();

sixth.setColumns(10);

sixth.setBounds(333, 380, 73, 35);

frame.getContentPane().add(sixth);

textField\_1 = **new** JTextField();

textField\_1.setColumns(10);

textField\_1.setBounds(10, 251, 23, 23);

frame.getContentPane().add(textField\_1);

textField\_3 = **new** JTextField();

textField\_3.setColumns(10);

textField\_3.setBounds(52, 249, 23, 23);

frame.getContentPane().add(textField\_3);

}

}

**MODULE – DOCTOR**

**4.1.iv. Doctor’s Login**

**package** Project;

**import** java.awt.event.\*;

**import** java.io.EOFException;

**import** java.io.File;

**import** java.io.FileInputStream;

**import** java.io.FileNotFoundException;

**import** java.io.FileOutputStream;

**import** java.io.IOException;

**import** java.io.ObjectInputStream;

**import** java.io.ObjectOutputStream;

**import** java.io.Serializable;

**import** java.io.StreamCorruptedException;

**import** javax.swing.\*;

**import** java.awt.Font;

**import** java.awt.HeadlessException;

**import** java.util.Date;

**import** java.util.Iterator;

**import** java.util.ArrayList;

**import** java.util.Calendar;

**import** java.awt.Color;

**import** java.awt.EventQueue;

**import** java.awt.SystemColor;

**import** java.awt.Window;

**class** Details **implements** Serializable{

**private** String user,pwd;

Details(String user,String pwd){

**this**.user=user;

**this**.pwd=pwd;

}

**public** String getPwd() {

**return** pwd;

}

**public** String getName() {

**return** user;

}

}

**public** **class** DoctorLoginPage {

JFrame frame;

**private** JTextField textField;

**private** **static** JLabel *icon1*;

**private** **static** JLabel *icon2*;

FileInputStream fin;

ObjectInputStream ois;

FileOutputStream fout;

ObjectOutputStream oos;

File file;

**private** JPasswordField passwordField;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

DoctorLoginPage window = **new** DoctorLoginPage();

window.frame.setVisible(**true**);

*icon2*.setVisible(**false**);

*icon1*.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** DoctorLoginPage() {

initialize();

**this**.*icon1*.setVisible(**true**);

}

/\*\*

\* Initialize the contents of the frame.

\*/

**void** initialize() {

frame = **new** JFrame("HRAP HOMEO CLINIC");

frame.getContentPane().setBackground(Color.***PINK***);

frame.setBounds(100, 100, 450, 300);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.getContentPane().setLayout(**null**);

frame.setVisible(**true**);

ArrayList<Details> al = **new** ArrayList<Details>();

JLabel lblNewLabel\_1 = **new** JLabel("Doctor Login");

lblNewLabel\_1.setFont(**new** Font("Tempus Sans ITC", Font.***BOLD***, 18));

lblNewLabel\_1.setBounds(155, 11, 145, 30);

frame.getContentPane().add(lblNewLabel\_1);

JLabel lblNewLabel = **new** JLabel("Username:");

lblNewLabel.setBounds(49, 73, 72, 14);

frame.getContentPane().add(lblNewLabel);

JLabel lblPassword = **new** JLabel("Password:");

lblPassword.setBounds(49, 114, 72, 14);

frame.getContentPane().add(lblPassword);

textField = **new** JTextField();

textField.setBounds(155, 65, 192, 30);

frame.getContentPane().add(textField);

JPasswordField passwordField = **new** JPasswordField();

passwordField.setBounds(155,106,192,30);

frame.getContentPane().add(passwordField);

JButton btnRegister = **new** JButton("Register");

btnRegister.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

String user=textField.getText();

String pwd=passwordField.~~getText~~();

al.add(**new** Details(user,pwd));

textField.setText("");

passwordField.setText("");

**try** {

file=**new** File("project.txt");

**if**(file.exists()) {

oos=**new** ObjectOutputStream(**new** FileOutputStream(file,**true**)) {

**protected** **void** writeStreamHeader() **throws** IOException{

reset();

}

};

}

**else**

oos=**new** ObjectOutputStream(**new** FileOutputStream(file));

oos.writeObject(al);

JOptionPane.*showMessageDialog*(frame,"Registration Successfull");

al.clear();

oos.close();

}

**catch**(Exception e1) {

JOptionPane.*showMessageDialog*(frame,"Registration unsucessfull");

}

}

});

btnRegister.setBounds(66, 206, 95, 30);

frame.getContentPane().add(btnRegister);

JButton btnPassword = **new** JButton("Login");

btnPassword.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

// **TODO** Auto-generated method stub

//String user=textField.getText();

//String pwd=passwordField.getText();

**boolean** flag = **false**;

**try** {

fin = **new** FileInputStream("project.txt");

ois= **new** ObjectInputStream(fin);

ArrayList<Details> al1;

String pwd = passwordField.~~getText~~();

String user = textField.getText();

**try** {

**while**((al1=(ArrayList)ois.readObject())!=**null**){

Details e1 = al1.get(0);

**if**(pwd.equals(e1.getPwd())&&user.equalsIgnoreCase(e1.getName())){

//System.out.println("1");

flag = **true**;

**break**;

}

}

}

**catch**(EOFException a) {

}

//System.out.println(flag);

**if**(flag==**true**){

JOptionPane.*showMessageDialog*(frame,"Login Successfull:)");

frame.setVisible(**false**);

DoctorDisplayPhotoPage a=**new** DoctorDisplayPhotoPage(user);

}

**else**{

//System.out.println("2");

JOptionPane.*showMessageDialog*(frame,"Login Failed");

}

passwordField.setText("");

textField.setText("");

} **catch**(Exception er){

er.printStackTrace();

}

}

});

btnPassword.setBounds(267, 206, 95, 30);

frame.getContentPane().add(btnPassword);

JCheckBox chckbxNewCheckBox = **new** JCheckBox("show password");

chckbxNewCheckBox.setBackground(Color.***LIGHT\_GRAY***);

chckbxNewCheckBox.setBounds(155, 143, 125, 30);

chckbxNewCheckBox.addActionListener(**new** ActionListener() {

@Override

**public** **void** actionPerformed(ActionEvent e) {

// **TODO** Auto-generated method stub

**if**(chckbxNewCheckBox.isSelected()) {

passwordField.setEchoChar((**char**)0);

*icon1*.setVisible(**false**);

*icon2*.setVisible(**true**);

}

**else** {

passwordField.setEchoChar('\*');

*icon2*.setVisible(**false**);

*icon1*.setVisible(**true**);

}

}

});

frame.getContentPane().add(chckbxNewCheckBox);

*icon1* = **new** JLabel("");

*icon1*.setIcon(**new** ImageIcon("C:\\Users\\AREEFA\\eclipse-workspace\\18K41A0505\\Assignment\\images\\eye9.png"));

*icon1*.setBounds(357, 105, 46, 23);

*icon1*.addMouseListener(**new** MouseListener() {

@Override

**public** **void** mouseClicked(MouseEvent e) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mousePressed(MouseEvent e) {

// **TODO** Auto-generated method stub

*icon2*.setVisible(**true**);

*icon1*.setVisible(**false**);

passwordField.setEchoChar('\*');

}

@Override

**public** **void** mouseReleased(MouseEvent e) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mouseEntered(MouseEvent e) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mouseExited(MouseEvent e) {

// **TODO** Auto-generated method stub

}

});

frame.getContentPane().add(*icon1*);

*icon2* = **new** JLabel("");

*icon2*.setIcon(**new** ImageIcon("C:\\Users\\AREEFA\\eclipse-workspace\\18K41A0505\\Assignment\\images\\in9.png"));

*icon2*.setBounds(357, 105, 46, 23);

*icon2*.addMouseListener(**new** MouseListener() {

@Override

**public** **void** mouseClicked(MouseEvent e) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mousePressed(MouseEvent e) {

// **TODO** Auto-generated method stub

*icon1*.setVisible(**false**);

*icon2*.setVisible(**true**);

passwordField.setEchoChar((**char**)0);

}

@Override

**public** **void** mouseReleased(MouseEvent e) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mouseEntered(MouseEvent e) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mouseExited(MouseEvent e) {

// **TODO** Auto-generated method stub

}

});

frame.getContentPane().add(*icon2*);

JButton btnHome = **new** JButton("Home");

btnHome.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frame.setVisible(**false**);

**new** HomePage();

}

});

btnHome.setBackground(Color.***PINK***);

btnHome.setBounds(10, 11, 81, 30);

frame.getContentPane().add(btnHome);

}

}

**4.1.v. Doctor Photo Display Page**

**package** Project;

**import** java.awt.EventQueue;

**import** javax.swing.ImageIcon;

**import** javax.swing.JFrame;

**import** java.awt.Color;

**import** javax.swing.JLabel;

**import** java.awt.BorderLayout;

**import** java.awt.Font;

**import** java.awt.Image;

**import** javax.swing.JButton;

**import** java.awt.event.ActionListener;

**import** java.awt.event.ActionEvent;

**public** **class** DoctorDisplayPhotoPage {

**private** JFrame frame;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

DoctorDisplayPhotoPage window = **new** DoctorDisplayPhotoPage();

window.frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** DoctorDisplayPhotoPage() {

initialize();

}

String docname;

**public** DoctorDisplayPhotoPage(String docname) {

**this**.docname=docname;

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

**private** **void** initialize() {

frame = **new** JFrame();

frame.getContentPane().setBackground(Color.***PINK***);

frame.getContentPane().setLayout(**null**);

frame.setVisible(**true**);

JLabel lblNewLabel = **new** JLabel("Welcome Doctor :"+docname);

lblNewLabel.setFont(**new** Font("Tahoma", Font.***BOLD***, 14));

lblNewLabel.setBounds(176, 47, 317, 24);

frame.getContentPane().add(lblNewLabel);

frame.setBounds(100, 100, 603, 444);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

JLabel lblNewLabel\_1 = **new** JLabel("HRAP HOMEO CLINIC");

**if**(docname.equalsIgnoreCase("sridhar")) {

Image img1=**new** ImageIcon(**this**.getClass().getResource("/doc.jpg")).getImage();

lblNewLabel\_1.setIcon(**new** ImageIcon(img1));

lblNewLabel\_1.setBounds(168, 93, 208, 190);

frame.getContentPane().add(lblNewLabel\_1);}

**else** **if**(docname.equalsIgnoreCase("sharath")) {

Image img1=**new** ImageIcon(**this**.getClass().getResource("/p3.jpeg")).getImage();

lblNewLabel\_1.setIcon(**new** ImageIcon(img1));

lblNewLabel\_1.setBounds(168, 93, 208, 190);

frame.getContentPane().add(lblNewLabel\_1);

}

**else** **if**(docname.equalsIgnoreCase("radhika")) {

Image img1=**new** ImageIcon(**this**.getClass().getResource("/p2.jpeg")).getImage();

lblNewLabel\_1.setIcon(**new** ImageIcon(img1));

lblNewLabel\_1.setBounds(168, 93, 208, 190);

frame.getContentPane().add(lblNewLabel\_1);

}

JButton btnNewButton = **new** JButton("Next>>");

btnNewButton.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frame.setVisible(**false**);

**new** PatientLoginPage(docname);

}

});

btnNewButton.setBounds(439, 319, 89, 38);

frame.getContentPane().add(btnNewButton);

}

}

**4.1. vi. Patient ID Entry**

**package** Project;

**import** java.awt.EventQueue;

**import** javax.swing.JFrame;

**import** javax.swing.JLabel;

**import** javax.swing.JTextArea;

**import** java.awt.Color;

**import** javax.swing.JTextField;

**import** javax.swing.JButton;

**import** java.awt.event.ActionListener;

**import** java.io.FileInputStream;

**import** java.io.IOException;

**import** java.io.ObjectInputStream;

**import** java.io.Serializable;

**import** java.util.ArrayList;

**import** java.util.Iterator;

**import** java.util.Scanner;

**import** java.awt.event.ActionEvent;

**import** java.awt.Font;

**public** **class** PatientLoginPage{

JFrame frame;

JTextField textField;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

PatientLoginPage window = **new** PatientLoginPage();

window.frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** PatientLoginPage() {

initialize();

}

String docname;

**public** PatientLoginPage(String docname) {

**this**.docname=docname;

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

**void** initialize() {

frame = **new** JFrame("HRAP HOMEO CLINIC");

frame.getContentPane().setBackground(Color.***PINK***);

frame.setBounds(100, 100, 450, 300);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.getContentPane().setLayout(**null**);

frame.setVisible(**true**);

JLabel lblNewLabel = **new** JLabel("Patient ID");

lblNewLabel.setFont(**new** Font("Tempus Sans ITC", Font.***BOLD***, 18));

lblNewLabel.setBounds(170, 21, 89, 41);

frame.getContentPane().add(lblNewLabel);

textField = **new** JTextField();

textField.setBounds(88, 81, 247, 41);

frame.getContentPane().add(textField);

textField.setColumns(10);

JButton btnNewButton = **new** JButton("Submit");

btnNewButton.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

btnNewButton.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

String id = textField.getText();

frame.setVisible(**false**);

**new** SampleFile(id,docname);

}

});

btnNewButton.setBounds(170, 151, 89, 23);

frame.getContentPane().add(btnNewButton);

JButton btnNewButton\_1 = **new** JButton("Patient's List");

btnNewButton\_1.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frame.setVisible(**false**);

**if**(docname.equalsIgnoreCase("sharath")) {

docname="Dr.Sharath";

}

**else** **if**(docname.equalsIgnoreCase("sridhar")) {

docname="Dr.Sridhar";

}

**else** **if**(docname.equalsIgnoreCase("radhika")) {

docname="Dr.Radhika";

}

**new** PatientDetailsPage(docname);

}

});

btnNewButton\_1.setBounds(150, 210, 135, 23);

frame.getContentPane().add(btnNewButton\_1);

JButton btnClose = **new** JButton("Close");

btnClose.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frame.setVisible(**false**);

**new** DoctorLoginPage();

}

});

btnClose.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

btnClose.setBounds(299, 152, 89, 23);

frame.getContentPane().add(btnClose);

}

}

**4.1. vii. Patient List Display for doctor**

**package** Project;

**import** java.awt.EventQueue;

**import** javax.swing.JFrame;

**import** javax.swing.JScrollPane;

**import** javax.swing.JTable;

**import** javax.swing.table.DefaultTableModel;

**import** javax.swing.JTextField;

**import** javax.swing.JButton;

**import** java.awt.event.ActionListener;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.File;

**import** java.io.FileNotFoundException;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** java.lang.System.Logger.Level;

**import** java.util.logging.Logger;

**import** java.awt.event.ActionEvent;

**import** java.awt.Color;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** java.awt.Font;

**public** **class** PatientDetailsPage{

**protected** **static** **final** java.util.logging.Level ***Level*** = **null**;

JFrame frame;

**private** JTable table;

**private** JButton btnImportx;

**private** JTextField textField;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

PatientDetailsPage window = **new** PatientDetailsPage();

window.frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** PatientDetailsPage() {

initialize();

}

String name;

**public** PatientDetailsPage(String name) {

**this**.name=name;

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

**private** **void** initialize() {

frame = **new** JFrame("HRAP HOMEO CLINIC");

frame.getContentPane().setBackground(**new** Color(255, 204, 255));

frame.setBounds(100, 100, 509, 530);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.getContentPane().setLayout(**null**);

frame.setVisible(**true**);

JScrollPane scrollPane = **new** JScrollPane();

scrollPane.setBounds(42, 103, 352, 135);

frame.getContentPane().add(scrollPane);

table = **new** JTable();

table.setModel(**new** DefaultTableModel(

**new** Object[][] {

},

**new** String[] {

"ID", "Phone Number", "FName", "LName", "Age", "Sex"

}

));

scrollPane.setViewportView(table);

JButton btnSave = **new** JButton("<<Back");

btnSave.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

**try** {

**if**(name.equals("Dr.Sharath")) {

FileWriter fw = **new** FileWriter("patients1");

BufferedWriter bw = **new** BufferedWriter(fw);

**for**(**int** i=0;i<table.getRowCount();i++) {

**for**(**int** j=0;j<table.getColumnCount();j++) {

bw.write(table.getValueAt(i, j).toString()+" ");

}

bw.newLine();

} bw.close();

fw.close();

frame.setVisible(**false**);

name="sharath";

**new** PatientLoginPage(name);}

**else** **if**(name.equals("Dr.Sridhar")) {

FileWriter fw = **new** FileWriter("patientssridhar");

BufferedWriter bw = **new** BufferedWriter(fw);

**for**(**int** i=0;i<table.getRowCount();i++) {

**for**(**int** j=0;j<table.getColumnCount();j++) {

bw.write(table.getValueAt(i, j).toString()+" ");

}

bw.newLine();

} bw.close();

fw.close();

frame.setVisible(**false**);

name="sridhar";

**new** PatientLoginPage(name);

}

**else** **if**(name.equals("Dr.Radhika")) {

FileWriter fw = **new** FileWriter("patients2");

BufferedWriter bw = **new** BufferedWriter(fw);

**for**(**int** i=0;i<table.getRowCount();i++) {

**for**(**int** j=0;j<table.getColumnCount();j++) {

bw.write(table.getValueAt(i, j).toString()+" ");

}

bw.newLine();

} bw.close();

fw.close();

name="radhika";

frame.setVisible(**false**);

**new** PatientLoginPage(name);

}

}

**catch**(IOException ex) {

Logger.*getLogger*(PatientDetailsPage1.**class**.getName()).log(***Level***,**null**,ex);

ex.printStackTrace();

}

}

});

btnSave.setBounds(10, 11, 89, 23);

frame.getContentPane().add(btnSave);

JLabel lblNewLabel = **new** JLabel("Patient's List");

lblNewLabel.setBackground(**new** Color(219, 112, 147));

lblNewLabel.setFont(**new** Font("Tempus Sans ITC", Font.***BOLD***, 17));

lblNewLabel.setBounds(150, 55, 145, 32);

frame.getContentPane().add(lblNewLabel);

textField = **new** JTextField();

textField.setEditable(**false**);

textField.setText(name);

textField.setBounds(279, 11, 145, 23);

frame.getContentPane().add(textField);

textField.setColumns(10);

JLabel lblDoctor = **new** JLabel("Doctor:");

lblDoctor.setFont(**new** Font("Tahoma", Font.***BOLD***, 14));

lblDoctor.setBounds(222, 9, 73, 23);

frame.getContentPane().add(lblDoctor);

JButton btnImport = **new** JButton("Import");

btnImport.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

**try** {

**if**(name.equals("Dr.Sharath")) {

System.***out***.println("Hii");

FileReader fr = **new** FileReader("patients1");

BufferedReader br = **new** BufferedReader(fr);

DefaultTableModel model = (DefaultTableModel)table.getModel();

Object[] lines = br.lines().toArray();

**for**(**int** i=0;i<lines.length;i++) {

String[] row= lines[i].toString().split(" ");

model.addRow(row);

}}

**else** **if**(name.equals("Dr.Sridhar")) {

FileReader fr = **new** FileReader("patientssridhar");

BufferedReader br = **new** BufferedReader(fr);

System.***out***.println("Hiiloo");

DefaultTableModel model = (DefaultTableModel)table.getModel();

Object[] lines = br.lines().toArray();

**for**(**int** i=0;i<lines.length;i++) {

String[] row= lines[i].toString().split(" ");

model.addRow(row);

}

}

**else** **if**(name.equals("Dr.Radhika")) {

FileReader fr = **new** FileReader("patients2");

BufferedReader br = **new** BufferedReader(fr);

DefaultTableModel model = (DefaultTableModel)table.getModel();

Object[] lines = br.lines().toArray();

**for**(**int** i=0;i<lines.length;i++) {

String[] row= lines[i].toString().split(" ");

model.addRow(row);

}

}

} **catch** (FileNotFoundException e1) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

}

});

btnImport.setBounds(139, 270, 169, 23);

frame.getContentPane().add(btnImport);

}

}

**4.1. viii. Patient’s Profile**

**package** Project;

**import** java.awt.EventQueue;

**import** javax.swing.JFrame;

**import** javax.swing.JLabel;

**import** javax.swing.JTextArea;

**import** javax.swing.JTextField;

**import** java.awt.Font;

**import** java.awt.Color;

**import** javax.swing.JButton;

**import** java.awt.event.ActionListener;

**import** java.io.BufferedReader;

**import** java.io.FileInputStream;

**import** java.io.FileNotFoundException;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** java.util.Properties;

**import** java.util.Scanner;

**import** java.awt.event.ActionEvent;

**public** **class** SampleFile {

**private** JFrame frame;

**private** JTextField txtName;

**private** JTextField txtAge;

**private** JTextArea textArea;

**private** JTextField textField;

**private** JTextField txtBpLevels;

**private** JTextField txtWeight;

**private** JLabel label;

**private** JLabel lblNewLabel;

**private** JLabel lblNewLabel\_1;

**private** JTextArea txtrAdditionalInformationOf;

**private** JButton btnNewButton;

**private** JLabel lblNewLabel\_2;

**private** JButton btnSave;

**private** JLabel lblNewLabel\_3;

**private** JLabel lblNewLabel\_4;

**private** JLabel lblNewLabel\_5;

**private** JLabel lblNewLabel\_6;

String ID;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

SampleFile window = **new** SampleFile();

window.frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** SampleFile() {

initialize();

}

String id;

String fn,ln;

**private** JTextField textField\_1;

String docname;

**private** JTextField textField\_2;

**private** JTextField textField\_3;

**private** JLabel lblNewLabel\_7;

SampleFile(String id,String docname){

**this**.id=id;

**this**.docname=docname;

System.***out***.println(id+docname);

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

**private** **void** initialize() {

frame = **new** JFrame("HRAP HOMEO CLINIC");

frame.getContentPane().setBackground(**new** Color(255, 204, 204));

frame.setBounds(100, 100, 650, 670);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.getContentPane().setLayout(**null**);

frame.setVisible(**true**);

txtName = **new** JTextField();

txtName.setEditable(**false**);

txtName.setBounds(10,112,86,20);

frame.getContentPane().add(txtName);

txtName.setColumns(10);

txtAge = **new** JTextField();

txtAge.setEditable(**false**);

txtAge.setColumns(10);

txtAge.setBounds(10, 197, 42, 20);

frame.getContentPane().add(txtAge);

textArea = **new** JTextArea();

textArea.setEditable(**false**);

textArea.setBackground(**new** Color(255, 228, 181));

textArea.setBounds(169, 175, 425, 86);

frame.getContentPane().add(textArea);

textField = **new** JTextField();

textField.setEditable(**false**);

textField.setBackground(**new** Color(255, 218, 185));

textField.setBounds(169, 297, 425, 86);

frame.getContentPane().add(textField);

textField.setColumns(10);

txtBpLevels = **new** JTextField();

txtBpLevels.setEditable(**false**);

txtBpLevels.setBounds(269, 74, 92, 28);

frame.getContentPane().add(txtBpLevels);

txtBpLevels.setColumns(10);

txtWeight = **new** JTextField();

txtWeight.setEditable(**false**);

txtWeight.setColumns(10);

txtWeight.setBounds(490, 74, 104, 28);

frame.getContentPane().add(txtWeight);

label = **new** JLabel("New label");

label.setBounds(198, 215, 21, -57);

frame.getContentPane().add(label);

lblNewLabel = **new** JLabel("Health issues:");

lblNewLabel.setFont(**new** Font("Trebuchet MS", Font.***BOLD***, 14));

lblNewLabel.setBounds(169, 144, 129, 28);

frame.getContentPane().add(lblNewLabel);

lblNewLabel\_1 = **new** JLabel("Medices prescribed");

lblNewLabel\_1.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblNewLabel\_1.setBackground(**new** Color(240, 240, 240));

lblNewLabel\_1.setBounds(171, 272, 141, 14);

frame.getContentPane().add(lblNewLabel\_1);

txtrAdditionalInformationOf = **new** JTextArea();

txtrAdditionalInformationOf.setEditable(**false**);

txtrAdditionalInformationOf.setBackground(**new** Color(173, 216, 230));

txtrAdditionalInformationOf.setForeground(**new** Color(255, 99, 71));

txtrAdditionalInformationOf.setText("Additional information of reports:");

txtrAdditionalInformationOf.setBounds(10, 394, 528, 114);

frame.getContentPane().add(txtrAdditionalInformationOf);

btnNewButton = **new** JButton("Diet");

btnNewButton.setForeground(Color.***BLACK***);

btnNewButton.setBackground(**new** Color(60, 179, 113));

btnNewButton.setFont(**new** Font("Times New Roman", Font.***BOLD***, 14));

btnNewButton.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frame.setVisible(**false**);

**new** PanelDiet(id,fn,ln,docname);

}

});

btnNewButton.setBounds(10, 241, 77, 39);

frame.getContentPane().add(btnNewButton);

lblNewLabel\_2 = **new** JLabel("Patient's profile");

lblNewLabel\_2.setFont(**new** Font("Tempus Sans ITC", Font.***BOLD*** | Font.***ITALIC***, 18));

lblNewLabel\_2.setBounds(274, 11, 296, 55);

frame.getContentPane().add(lblNewLabel\_2);

btnSave = **new** JButton("Save");

btnSave.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frame.setVisible(**false**);

**new** PatientLoginPage(docname);

}

});

btnSave.setForeground(Color.***BLACK***);

btnSave.setFont(**new** Font("Times New Roman", Font.***BOLD***, 14));

btnSave.setBackground(Color.***GREEN***);

btnSave.setBounds(221, 581, 77, 39);

frame.getContentPane().add(btnSave);

lblNewLabel\_3 = **new** JLabel("BP:");

lblNewLabel\_3.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblNewLabel\_3.setBounds(224, 77, 35, 21);

frame.getContentPane().add(lblNewLabel\_3);

lblNewLabel\_4 = **new** JLabel("Weight:");

lblNewLabel\_4.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblNewLabel\_4.setBounds(426, 80, 54, 21);

frame.getContentPane().add(lblNewLabel\_4);

lblNewLabel\_5 = **new** JLabel("Name:");

lblNewLabel\_5.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblNewLabel\_5.setBounds(10, 80, 66, 21);

frame.getContentPane().add(lblNewLabel\_5);

lblNewLabel\_6 = **new** JLabel("Age:");

lblNewLabel\_6.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblNewLabel\_6.setBounds(10, 175, 35, 21);

frame.getContentPane().add(lblNewLabel\_6);

textField\_1 = **new** JTextField();

textField\_1.setEditable(**false**);

textField\_1.setColumns(10);

textField\_1.setBounds(10, 46, 86, 20);

frame.getContentPane().add(textField\_1);

ID=textField\_1 .getText();

JButton btnUpdate = **new** JButton("Update");

btnUpdate.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

frame.setVisible(**false**);

ID = textField\_1.getText();

**new** UpdateTable(ID,docname);

}

});

btnUpdate.setForeground(Color.***BLACK***);

btnUpdate.setFont(**new** Font("Times New Roman", Font.***BOLD***, 14));

btnUpdate.setBackground(Color.***RED***);

btnUpdate.setBounds(179, 519, 192, 39);

frame.getContentPane().add(btnUpdate);

textField\_2 = **new** JTextField();

textField\_2.setEditable(**false**);

textField\_2.setColumns(10);

textField\_2.setBounds(106, 112, 86, 20);

frame.getContentPane().add(textField\_2);

textField\_3 = **new** JTextField();

textField\_3.setEditable(**false**);

textField\_3.setColumns(10);

textField\_3.setBounds(75, 197, 54, 20);

frame.getContentPane().add(textField\_3);

lblNewLabel\_7 = **new** JLabel("Gender:");

lblNewLabel\_7.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblNewLabel\_7.setBounds(75, 175, 54, 21);

frame.getContentPane().add(lblNewLabel\_7);

String line="";

**try** {

**if**(docname.equalsIgnoreCase("sridhar")) {

FileReader fin=**new** FileReader("profile2");

BufferedReader br=**new** BufferedReader(fin);

Scanner sc = **new** Scanner(fin);

**while**(sc.hasNextLine()) {

line=sc.nextLine();

String[] row;

**for**(**int** i=0;i<line.length();i++) {

}

**if**(line.startsWith(id)) {

String s=line;

sc.close();

FileWriter fw = **new** FileWriter("profile22");

**int** len=s.length();

String[] str[];

**int** j=0;

**for**(**int** i=0;i<s.length();i++)

{

fw.write(s.charAt(i));

}

fw.close();

String lines = **null**;

FileInputStream fin1=**new** FileInputStream("profile22");

Scanner sc1 = **new** Scanner(fin1);

**while**(sc1.hasNext()){

lines=sc1.next();

String id=sc1.next();

String lname=sc1.next();

String age=sc1.next();

String sex=sc1.next();

String bp=sc1.next();

String wt=sc1.next();

String health=sc1.next();

String med=sc1.next();

String add=sc1.next();

textField\_1.setText(lines);

txtName.setText(id);

fn=id;

textField\_2.setText(lname);

ln=lname;

txtAge.setText(age);

textField\_3.setText(sex);

txtBpLevels.setText(bp);

txtWeight.setText(wt);

textArea.setText(health);

textField.setText(med);

txtrAdditionalInformationOf.setText(add);

System.***out***.println(lines);

}

sc1.close();

}}}**else** **if**(docname.equalsIgnoreCase("sharath")) {

FileReader fin=**new** FileReader("profile1");

BufferedReader br=**new** BufferedReader(fin);

Scanner sc = **new** Scanner(fin);

**while**(sc.hasNextLine()) {

line=sc.nextLine();

String[] row;

**for**(**int** i=0;i<line.length();i++) {

}

**if**(line.startsWith(id)) {

String s=line;

sc.close();

FileWriter fw = **new** FileWriter("profile11");

**int** len=s.length();

String[] str[];

**int** j=0;

**for**(**int** i=0;i<s.length();i++)

{

fw.write(s.charAt(i));

}

fw.close();

String lines = **null**;

FileInputStream fin1=**new** FileInputStream("profile11");

Scanner sc1 = **new** Scanner(fin1);

**while**(sc1.hasNext()){

lines=sc1.next();

String id=sc1.next();

String lname=sc1.next();

String age=sc1.next();

String sex=sc1.next();

String bp=sc1.next();

String wt=sc1.next();

String health=sc1.next();

String med=sc1.next();

String add=sc1.next();

textField\_1.setText(lines);

txtName.setText(id);

textField\_2.setText(lname);

txtAge.setText(age);

textField\_3.setText(sex);

txtBpLevels.setText(bp);

txtWeight.setText(wt);

textArea.setText(health);

textField.setText(med);

txtrAdditionalInformationOf.setText(add);

System.***out***.println(lines);

}

sc1.close();

}}

}**else** **if**(docname.equalsIgnoreCase("radhika")) {

FileReader fin=**new** FileReader("profile3");

BufferedReader br=**new** BufferedReader(fin);

Scanner sc = **new** Scanner(fin);

**while**(sc.hasNextLine()) {

line=sc.nextLine();

String[] row;

**for**(**int** i=0;i<line.length();i++) {

}

**if**(line.startsWith(id)) {

String s=line;

sc.close();

FileWriter fw = **new** FileWriter("profile33");

String[] str[];

**int** j=0;

**for**(**int** i=0;i<s.length();i++)

{

fw.write(s.charAt(i));

}

fw.close();

String lines = **null**;

FileInputStream fin1=**new** FileInputStream("profile33");

Scanner sc1 = **new** Scanner(fin1);

**while**(sc1.hasNext()){

lines=sc1.next();

String id=sc1.next();

String lname=sc1.next();

String age=sc1.next();

String sex=sc1.next();

String bp=sc1.next();

String wt=sc1.next();

String health=sc1.next();

String med=sc1.next();

String add=sc1.next();

textField\_1.setText(lines);

txtName.setText(id);

textField\_2.setText(lname);

txtAge.setText(age);

textField\_3.setText(sex);

txtBpLevels.setText(bp);

txtWeight.setText(wt);

textArea.setText(health);

textField.setText(med);

txtrAdditionalInformationOf.setText(add);

System.***out***.println(lines);

}

sc1.close();

}}

}

}

**catch**(IOException ioe){

ioe.printStackTrace();

}

}

}

**4.1. ix. Update Patient’s Profile**

**package** Project;

**import** java.awt.EventQueue;

**import** javax.swing.JFrame;

**import** javax.swing.JTable;

**import** javax.swing.JScrollPane;

**import** javax.swing.table.DefaultTableModel;

**import** javax.swing.table.TableRowSorter;

**import** javax.swing.AbstractButton;

**import** javax.swing.JButton;

**import** java.awt.event.ActionListener;

**import** java.awt.event.ActionEvent;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** javax.swing.JTextField;

**import** javax.swing.RowFilter;

**import** javax.swing.JTextArea;

**import** java.awt.event.MouseAdapter;

**import** java.awt.event.MouseEvent;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.File;

**import** java.io.FileNotFoundException;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** java.util.logging.Level;

**import** java.util.logging.Logger;

**import** java.awt.Color;

**import** java.awt.Font;

**import** java.awt.event.KeyAdapter;

**import** java.awt.event.KeyEvent;

**public** **class** UpdateTable {

**protected** **static** **final** Level ***Level*** = **null**;

**private** JFrame frame;

**private** JTable table;

**private** JTextField textField;

**private** JTextField textField\_1;

**private** JTextField textField\_2;

**private** JTextField textField\_3;

**private** JTextField textField\_4;

**private** JTextField textField\_5;

**private** JTextField textField\_6;

**private** JTextField textField\_7;

String Id;

**private** JTextField textField\_8;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

UpdateTable window = **new** UpdateTable();

window.frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** UpdateTable() {

initialize();

}

String keyid;

String docname;

**private** JTextField textField\_9;

**private** JTextField textField\_10;

**public** UpdateTable(String ID,String docname) {

keyid=ID;

**this**.docname=docname;

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

**private** **void** initialize() {

frame = **new** JFrame("HRAP HOMEO CLINIC");

frame.getContentPane().setBackground(**new** Color(135, 206, 235));

frame.setBounds(100, 100, 845, 888);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.getContentPane().setLayout(**null**);

frame.setVisible(**true**);

JTextArea textArea = **null**;

JScrollPane scrollPane = **new** JScrollPane();

scrollPane.setBounds(224, 24, 625, 292);

frame.getContentPane().add(scrollPane);

table = **new** JTable();

table.addMouseListener(**new** MouseAdapter() {

@Override

**public** **void** mouseClicked(MouseEvent e) {

**int** selectedRow = table.getSelectedRow();

DefaultTableModel model = (DefaultTableModel)table.getModel();

textField.setText(model.getValueAt(selectedRow, 0).toString());

textField\_1.setText(model.getValueAt(selectedRow, 1).toString());

textField\_9.setText(model.getValueAt(selectedRow, 2).toString());

textField\_2.setText(model.getValueAt(selectedRow, 3).toString());

textField\_10.setText(model.getValueAt(selectedRow, 4).toString());

textField\_3.setText(model.getValueAt(selectedRow, 5).toString());

textField\_4.setText(model.getValueAt(selectedRow, 6).toString());

textField\_5.setText(model.getValueAt(selectedRow, 7).toString());

textField\_6.setText(model.getValueAt(selectedRow, 8).toString());

textField\_7.setText(model.getValueAt(selectedRow, 9).toString());

}

});

table.setModel(**new** DefaultTableModel(

**new** Object[][] {

},

**new** String[] {

"ID", "First Name", "Last name", "age", "gender", "weight", "BP", "Health issue", "Medicines", "Additional Reports"

}

) {

Class[] columnTypes = **new** Class[] {

String.**class**, String.**class**, String.**class**, String.**class**, String.**class**, String.**class**, String.**class**, String.**class**, Object.**class**, Object.**class**

};

**public** Class getColumnClass(**int** columnIndex) {

**return** columnTypes[columnIndex];

}

});

scrollPane.setViewportView(table);

JButton btnNewButton = **new** JButton("Import From FIle");

btnNewButton.setBackground(**new** Color(240, 230, 140));

btnNewButton.setFont(**new** Font("Tahoma", Font.***BOLD***, 13));

btnNewButton.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

//String filePath="C:\\Users\\AREEFA\\Desktop\\names.txt";

// File file=new File(filePath);

**try** {

**if**(docname.equalsIgnoreCase("sridhar")) {

FileReader fr=**new** FileReader("profile2");

BufferedReader br = **new** BufferedReader(fr);

DefaultTableModel model = (DefaultTableModel)table.getModel();

Object[] lines = br.lines().toArray();

**for**(**int** i=0;i<lines.length;i++) {

String[] row= lines[i].toString().split(" ");

model.addRow(row);

}}

**else** **if**(docname.equalsIgnoreCase("sharath")){

FileReader fr=**new** FileReader("profile1");

BufferedReader br = **new** BufferedReader(fr);

DefaultTableModel model = (DefaultTableModel)table.getModel();

Object[] lines = br.lines().toArray();

**for**(**int** i=0;i<lines.length;i++) {

String[] row= lines[i].toString().split(" ");

model.addRow(row);

}

}

**else** **if**(docname.equalsIgnoreCase("radhika")) {

FileReader fr=**new** FileReader("profile3");

BufferedReader br = **new** BufferedReader(fr);

DefaultTableModel model = (DefaultTableModel)table.getModel();

Object[] lines = br.lines().toArray();

**for**(**int** i=0;i<lines.length;i++) {

String[] row= lines[i].toString().split(" ");

model.addRow(row);

}

}

} **catch** (FileNotFoundException e1) {

// **TODO** Auto-generated catch block

Logger.*getLogger*(PatientDetailsPage.**class**.getName()).log(***Level***,**null**,e1);

e1.printStackTrace();

}

}

});

btnNewButton.setBounds(23, 72, 176, 48);

frame.getContentPane().add(btnNewButton);

JLabel lblNewLabel = **new** JLabel("ID");

lblNewLabel.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblNewLabel.setBounds(98, 344, 80, 23);

frame.getContentPane().add(lblNewLabel);

JLabel lblName = **new** JLabel("Name");

lblName.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblName.setBounds(98, 389, 46, 14);

frame.getContentPane().add(lblName);

JLabel lblNewLabel\_1\_1 = **new** JLabel("Age");

lblNewLabel\_1\_1.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblNewLabel\_1\_1.setBounds(98, 420, 46, 14);

frame.getContentPane().add(lblNewLabel\_1\_1);

JLabel lblNewLabel\_1\_1\_1 = **new** JLabel("Weight");

lblNewLabel\_1\_1\_1.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblNewLabel\_1\_1\_1.setBounds(98, 464, 46, 14);

frame.getContentPane().add(lblNewLabel\_1\_1\_1);

JLabel lblNewLabel\_1\_1\_1\_1 = **new** JLabel("Health issue");

lblNewLabel\_1\_1\_1\_1.setFont(**new** Font("Tahoma", Font.***BOLD***, 13));

lblNewLabel\_1\_1\_1\_1.setBounds(470, 327, 110, 30);

frame.getContentPane().add(lblNewLabel\_1\_1\_1\_1);

JLabel lblNewLabel\_1\_1\_1\_1\_1 = **new** JLabel("Medicines");

lblNewLabel\_1\_1\_1\_1\_1.setFont(**new** Font("Tahoma", Font.***BOLD***, 13));

lblNewLabel\_1\_1\_1\_1\_1.setBounds(480, 500, 120, 30);

frame.getContentPane().add(lblNewLabel\_1\_1\_1\_1\_1);

textField = **new** JTextField();

textField.setBounds(190, 345, 86, 20);

frame.getContentPane().add(textField);

textField.setColumns(10);

textField\_1 = **new** JTextField();

textField\_1.setColumns(10);

textField\_1.setBounds(190, 386, 86, 20);

frame.getContentPane().add(textField\_1);

textField\_2 = **new** JTextField();

textField\_2.setColumns(10);

textField\_2.setBounds(190, 417, 46, 20);

frame.getContentPane().add(textField\_2);

textField\_3 = **new** JTextField();

textField\_3.setColumns(10);

textField\_3.setBounds(190, 492, 86, 20);

frame.getContentPane().add(textField\_3);

JLabel lblNewLabel\_2 = **new** JLabel("Additional Reports ");

lblNewLabel\_2.setFont(**new** Font("Tahoma", Font.***BOLD***, 13));

lblNewLabel\_2.setBounds(10, 539, 134, 32);

frame.getContentPane().add(lblNewLabel\_2);

textField\_4 = **new** JTextField();

textField\_4.setColumns(10);

textField\_4.setBounds(190, 462, 86, 20);

frame.getContentPane().add(textField\_4);

JButton btnUpdate = **new** JButton("Update & Save");

btnUpdate.setBackground(**new** Color(173, 255, 47));

btnUpdate.setFont(**new** Font("Tahoma", Font.***BOLD***, 13));

btnUpdate.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

**int** i = table.getSelectedRow();

DefaultTableModel model = (DefaultTableModel)table.getModel();

**if**(i>=0) {

model.setValueAt(textField.getText(),i,0);

Id=textField.getText();

model.setValueAt(textField\_1.getText(),i,1);

model.setValueAt(textField\_9.getText(),i,2);

model.setValueAt(textField\_2.getText(),i,3);

model.setValueAt(textField\_10.getText(),i,4);

model.setValueAt(textField\_3.getText(),i,5);

model.setValueAt(textField\_4.getText(),i,6);

model.setValueAt(textField\_5.getText(),i,7);

model.setValueAt(textField\_6.getText(),i,8);

model.setValueAt(textField\_7.getText(),i,9);

JOptionPane.*showMessageDialog*(frame,"Updated Successfully");

**try** {

**if**(docname.equalsIgnoreCase("sridhar")){

FileWriter fw = **new** FileWriter("profile2");

BufferedWriter bw = **new** BufferedWriter(fw);

**for**(**int** k=0;k<table.getRowCount();k++) {

**for**(**int** j=0;j<table.getColumnCount();j++) {

bw.write(table.getValueAt(k, j).toString()+" ");

}

bw.newLine();

} bw.close();

fw.close();

frame.setVisible(**false**);

**new** SampleFile(Id,docname);

}

**else** **if**(docname.equalsIgnoreCase("sharath")){

FileWriter fw = **new** FileWriter("profile1");

BufferedWriter bw = **new** BufferedWriter(fw);

**for**(**int** k=0;k<table.getRowCount();k++) {

**for**(**int** j=0;j<table.getColumnCount();j++) {

bw.write(table.getValueAt(k, j).toString()+" ");

}

bw.newLine();

} bw.close();

fw.close();

frame.setVisible(**false**);

**new** SampleFile(Id,docname);

}

**else** **if**(docname.equalsIgnoreCase("radhika")) {

FileWriter fw = **new** FileWriter("profile3");

BufferedWriter bw = **new** BufferedWriter(fw);

**for**(**int** k=0;k<table.getRowCount();k++) {

**for**(**int** j=0;j<table.getColumnCount();j++) {

bw.write(table.getValueAt(k, j).toString()+" ");

}

bw.newLine();

} bw.close();

fw.close();

System.***out***.println("Done");

frame.setVisible(**false**);

**new** SampleFile(Id,docname);

}

}

**catch**(IOException ex) {

Logger.*getLogger*(PatientDetailsPage.**class**.getName()).log(***Level***,**null**,ex);

ex.printStackTrace();

}

}**else** {

JOptionPane.*showMessageDialog*(**null**, "Error");

}

}

});

btnUpdate.setBounds(162, 623, 158, 40);

frame.getContentPane().add(btnUpdate);

textField\_5 = **new** JTextField();

textField\_5.setColumns(10);

textField\_5.setBounds(470, 362, 325, 131);

frame.getContentPane().add(textField\_5);

textField\_6 = **new** JTextField();

textField\_6.setColumns(10);

textField\_6.setBounds(470, 541, 325, 131);

frame.getContentPane().add(textField\_6);

textField\_7 = **new** JTextField();

textField\_7.setColumns(10);

textField\_7.setBounds(145, 523, 262, 71);

frame.getContentPane().add(textField\_7);

JLabel lblNewLabel\_1\_1\_1\_2 = **new** JLabel("BP");

lblNewLabel\_1\_1\_1\_2.setFont(**new** Font("Tahoma", Font.***BOLD***, 12));

lblNewLabel\_1\_1\_1\_2.setBounds(98, 495, 46, 14);

frame.getContentPane().add(lblNewLabel\_1\_1\_1\_2);

textField\_8 = **new** JTextField();

System.***out***.println(keyid);

textField\_8.setText(keyid);

textField\_8.addKeyListener(**new** KeyAdapter() {

@Override

**public** **void** keyReleased(KeyEvent e) {

DefaultTableModel model=(DefaultTableModel)table.getModel();

String search = textField\_8.getText().toLowerCase();

TableRowSorter<DefaultTableModel> tr=**new** TableRowSorter<DefaultTableModel>(model);

table.setRowSorter(tr);

tr.setRowFilter(RowFilter.*regexFilter*(search));

}

});

textField\_8.setColumns(10);

textField\_8.setBounds(102, 271, 86, 20);

frame.getContentPane().add(textField\_8);

textField\_9 = **new** JTextField();

textField\_9.setColumns(10);

textField\_9.setBounds(306, 387, 86, 20);

frame.getContentPane().add(textField\_9);

textField\_10 = **new** JTextField();

textField\_10.setColumns(10);

textField\_10.setBounds(265, 418, 46, 20);

frame.getContentPane().add(textField\_10);

}

}

**4.1.x. Patient’s Recommended Diet**

**package** Project;

**import** java.awt.EventQueue;

**import** javax.swing.JFrame;

**import** javax.swing.JLayeredPane;

**import** javax.swing.JOptionPane;

**import** java.awt.CardLayout;

**import** javax.swing.JPanel;

**import** javax.swing.JLabel;

**import** javax.swing.JButton;

**import** javax.swing.JTextField;

**import** javax.swing.SwingConstants;

**import** java.awt.event.ActionListener;

**import** java.awt.print.PageFormat;

**import** java.awt.print.Printable;

**import** java.awt.print.PrinterException;

**import** java.awt.print.PrinterJob;

**import** java.awt.event.ActionEvent;

**import** javax.swing.JCheckBox;

**import** java.awt.Color;

**import** java.awt.Font;

**import** java.awt.Graphics;

**import** java.awt.Graphics2D;

**public** **class** PanelDiet {

**private** JFrame frame;

**private** JTextField textField;

**private** JTextField textField\_1;

**private** JTextField textField\_2;

**private** JTextField textField\_3;

**private** JTextField textField\_4;

**private** JTextField textField\_5;

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

PanelDiet window = **new** PanelDiet();

window.frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** PanelDiet() {

initialize();

}

String id,fn,ln,docname;

**public** PanelDiet(String id,String fn,String ln,String docname) {

**this**.id=id;

**this**.fn=fn;

**this**.ln=ln;

**this**.docname=docname;

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

**private** **void** initialize() {

frame = **new** JFrame("HRAP HOMEO CLINIC");

frame.getContentPane().setBackground(**new** Color(255,240,245));

frame.setBounds(100, 100, 960, 788);

//frame.setBackground(255,240,245);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.getContentPane().setLayout(**null**);

frame.setVisible(**true**);

JLayeredPane layeredPane = **new** JLayeredPane();

layeredPane.setBounds(10, 83, 705, 547);

frame.getContentPane().add(layeredPane);

layeredPane.setLayout(**new** CardLayout(0, 0));

JPanel panel = **new** JPanel();

panel.setBounds(27, 641, 810, 108);

frame.getContentPane().add(panel);

panel.setLayout(**null**);

JPanel panel1 = **new** JPanel();

panel1.setBackground(Color.***PINK***);

layeredPane.add(panel1, "name\_1555880615500");

panel1.setLayout(**null**);

JCheckBox chckbxNewCheckBox = **new** JCheckBox("Have Suppliments for Bone strength");

chckbxNewCheckBox.setFont(**new** Font("Tahoma", Font.***PLAIN***, 13));

chckbxNewCheckBox.setBounds(29,194, 230,44);

panel1.add(chckbxNewCheckBox);

JCheckBox chckbxOmegaTablets = **new** JCheckBox("Omega 3 Tablets daily twice");

chckbxOmegaTablets.setFont(**new** Font("Tahoma", Font.***PLAIN***, 13));

chckbxOmegaTablets.setBounds(29, 196, 230, 44);

panel1.add(chckbxOmegaTablets);

JCheckBox chckbxNewCheckBox\_2 = **new** JCheckBox("Nutrishake");

chckbxNewCheckBox\_2.setFont(**new** Font("Tahoma", Font.***PLAIN***, 14));

chckbxNewCheckBox\_2.setBounds(29, 264, 230, 38);

panel1.add(chckbxNewCheckBox\_2);

JCheckBox chckbxNewCheckBox\_2\_1 = **new** JCheckBox("Vitamins intake");

chckbxNewCheckBox\_2\_1.setFont(**new** Font("Tahoma", Font.***PLAIN***, 14));

chckbxNewCheckBox\_2\_1.setBounds(29, 322, 230, 38);

panel1.add(chckbxNewCheckBox\_2\_1);

JCheckBox chckbxNewCheckBox\_2\_2 = **new** JCheckBox("Proteins intake");

chckbxNewCheckBox\_2\_2.setFont(**new** Font("Tahoma", Font.***PLAIN***, 14));

chckbxNewCheckBox\_2\_2.setBounds(29, 376, 230, 38);

panel1.add(chckbxNewCheckBox\_2\_2);

JLabel lblNewLabel\_4 = **new** JLabel("");

lblNewLabel\_4.setBounds(62, 449, 726, 44);

panel1.add(lblNewLabel\_4);

textField = **new** JTextField();

textField.setText("Daily walk:");

textField.setColumns(10);

textField.setBounds(446, 144, 240, 38);

panel1.add(textField);

textField\_1 = **new** JTextField();

textField\_1.setText("Exercise hours:");

textField\_1.setColumns(10);

textField\_1.setBounds(446, 196, 240, 38);

panel1.add(textField\_1);

textField\_2 = **new** JTextField();

textField\_2.setText("Calories intake:");

textField\_2.setColumns(10);

textField\_2.setBounds(446, 245, 240, 38);

panel1.add(textField\_2);

JLabel lblNewLabel\_1\_1 = **new** JLabel("ID");

lblNewLabel\_1\_1.setFont(**new** Font("Dialog", Font.***BOLD***, 18));

lblNewLabel\_1\_1.setBounds(29, 55, 33, 23);

panel1.add(lblNewLabel\_1\_1);

textField\_3 = **new** JTextField();

textField\_3.setText((String) **null**);

textField\_3.setEditable(**false**);

textField\_3.setColumns(10);

textField\_3.setBounds(62, 57, 157, 20);

panel1.add(textField\_3);

textField\_3.setText(id);

textField\_4 = **new** JTextField();

textField\_4.setText((String) **null**);

textField\_4.setEditable(**false**);

textField\_4.setColumns(10);

textField\_4.setBounds(10,88, 100, 20);

panel1.add(textField\_4);

textField\_4.setText(fn);

textField\_5 = **new** JTextField();

textField\_5.setText((String) **null**);

textField\_5.setEditable(**false**);

textField\_5.setColumns(10);

textField\_5.setBounds(122, 88, 97, 20);

panel1.add(textField\_5);

textField\_5.setText(ln);

JLabel lblNewLabel = **new** JLabel("Diet");

lblNewLabel.setFont(**new** Font("Tempus Sans ITC", Font.***BOLD***, 29));

lblNewLabel.setBounds(350, 27, 67, 21);

panel1.add(lblNewLabel);

JLabel lblNewLabel\_1\_1\_1 = **new** JLabel("Name");

lblNewLabel\_1\_1\_1.setFont(**new** Font("Dialog", Font.***BOLD***, 16));

lblNewLabel\_1\_1\_1.setBounds(29, 89, 62, 23);

panel1.add(lblNewLabel\_1\_1\_1);

JPanel panel2 = **new** JPanel();

layeredPane.add(panel2, "name\_1877305070300");

panel2.setLayout(**null**);

JLabel lblNewLabel\_1 = **new** JLabel("This is panel2");

lblNewLabel\_1.setHorizontalAlignment(SwingConstants.***CENTER***);

lblNewLabel\_1.setBounds(49, 60, 400, 132);

panel2.add(lblNewLabel\_1);

JPanel panel3 = **new** JPanel();

layeredPane.add(panel3, "name\_1897299946100");

panel3.setLayout(**null**);

JLabel lblNewLabel\_2 = **new** JLabel("This is panel3");

lblNewLabel\_2.setHorizontalAlignment(SwingConstants.***CENTER***);

lblNewLabel\_2.setBounds(48, 61, 376, 127);

panel3.add(lblNewLabel\_2);

JButton btn = **new** JButton("<<Back");

btn.addActionListener(**new** ActionListener() {

@Override

**public** **void** actionPerformed(ActionEvent e) {

// **TODO** Auto-generated method stub

frame.setVisible(**false**);

**new** SampleFile(id,docname);

}

});

btn.setBounds(17,30,102,29);

frame.getContentPane().add(btn);

JButton btnDisplay = **new** JButton("Display");

btnDisplay.setBackground(**new** Color(255, 182, 193));

btnDisplay.setBounds(779, 133, 108, 59);

frame.getContentPane().add(btnDisplay);

JButton btnPrintdiet = **new** JButton("PrintDiet");

btnPrintdiet.setBackground(**new** Color(255, 255, 0));

btnPrintdiet.setBounds(779, 286, 108, 52);

frame.getContentPane().add(btnPrintdiet);

btnPrintdiet.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

printRecord(panel1);

}

});

btnDisplay.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

// **TODO** Auto-generated method stub

String s1,s2,s3,s4,s5;

**if**(chckbxNewCheckBox.isSelected())

s1="Have Suppliments for Bone strength ,";

**else**

s1="";

**if**(chckbxOmegaTablets.isSelected())

s2="Omega 3 Tablets daily twice ,";

**else**

s2="";

**if**(chckbxNewCheckBox\_2.isSelected())

s3="Nutrishake ,";

**else**

s3="";

**if**(chckbxNewCheckBox\_2\_1.isSelected())

s4="Vitamins intake ,";

**else**

s4="";

**if**(chckbxNewCheckBox\_2\_2.isSelected())

s5="Proteins intake";

**else**

s5="";

lblNewLabel\_4.setText("Diet : "+s1+s2+s3+s4+s5);

}

});

}

**private** **void** printRecord(JPanel panel) {

//Create PrinterJob here

PrinterJob printerJob = PrinterJob.*getPrinterJob*();

//Set printer job name

printerJob.setJobName("Print Diet");

//Set Printable

printerJob.setPrintable(**new** Printable() {

@Override

**public** **int** print(Graphics graphics, PageFormat pageFormat, **int** pageIndex) **throws** PrinterException {

// **TODO** Auto-generated method stub

**if**(pageIndex>0) {

**return** Printable.***NO\_SUCH\_PAGE***;

}

//Make 2D Graphics to map content

Graphics2D graphics2D=(Graphics2D)graphics;

//set Graphics Translation

graphics2D.translate(pageFormat.getImageableX()\*2, pageFormat.getImageableY()\*2);

//This is a page scale. Default should be 0.3 I am using 0.5

graphics2D.scale(0.5, 0.5);

//Now paint panel as graphics2D

panel.paint(graphics2D);

//return if page exists

**return** Printable.***PAGE\_EXISTS***;

//return pageIndex;

}

});

//Store printerDialog as boolean

**boolean** returningResult = printerJob.printDialog();

//check if dialog is showing

**if**(returningResult) {

//use try catch exception for failure

**try** {

printerJob.print();

}**catch**(PrinterException printerException) {

JOptionPane.*showMessageDialog*(frame,"Print Error: "+printerException.getMessage());

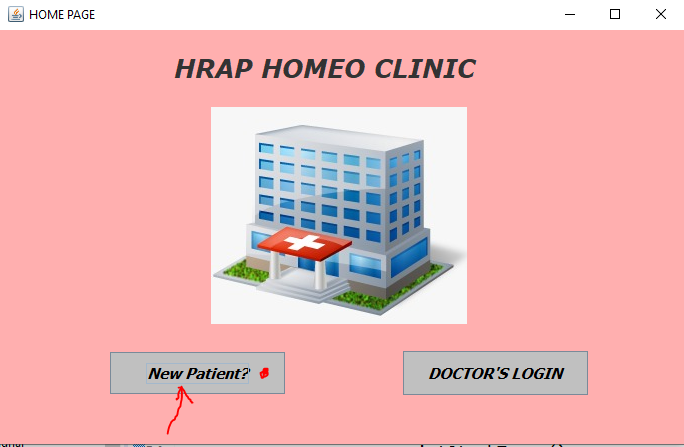
}

}

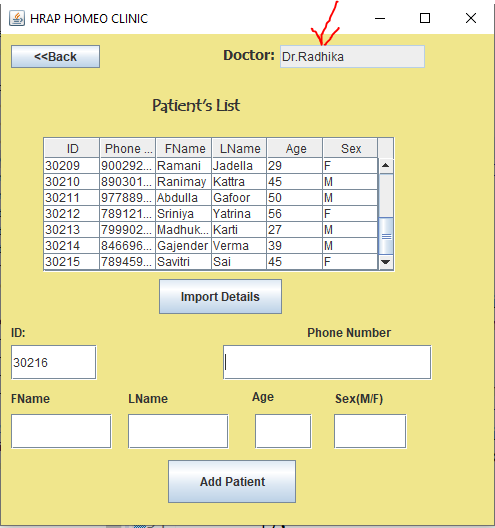
}

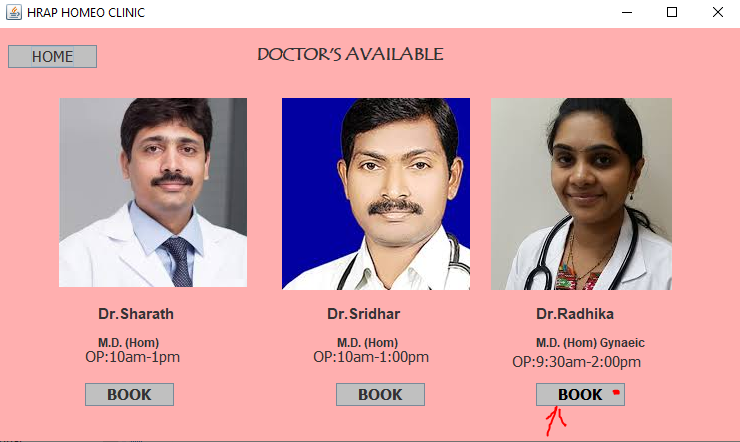
}

1. **RESULT SCREENS**
   1. **MODULE : STAFF (Register New Patient)**

****

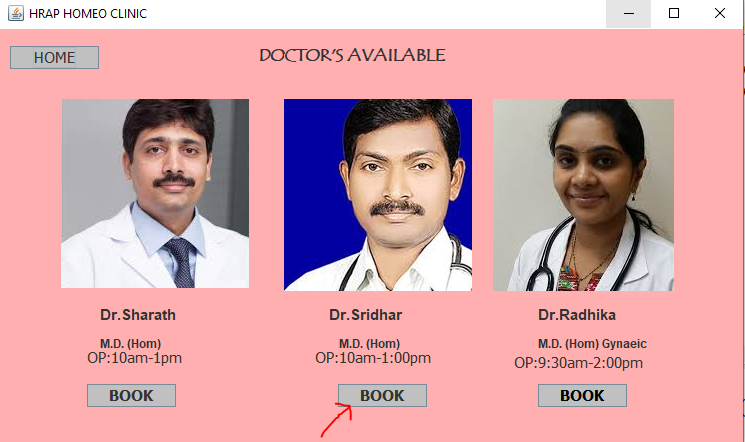
**Fig 5.1.i. Home Page (Go to New Register?)**



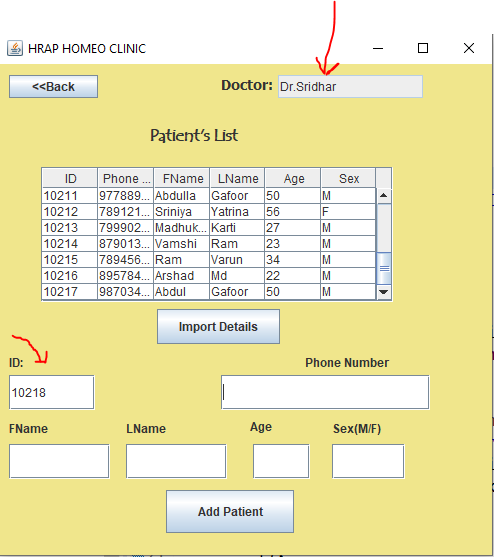
**Patient Choose Doctor**

**Fig 5.1.ii. New Patient selects the doctor**

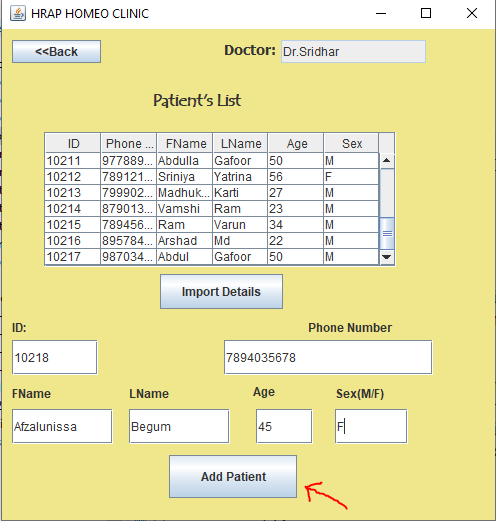
**Fig 5.1.iii. Staff register the patient (ID generates)**

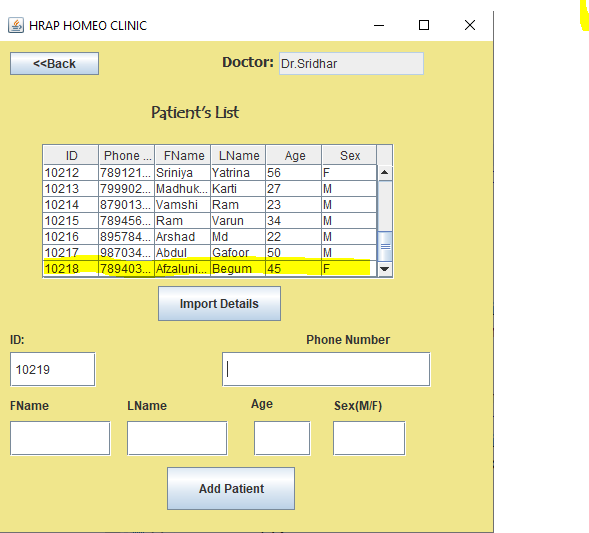
****

**Fig 5.1.iv. Another new patientient selecting other doctor**

****

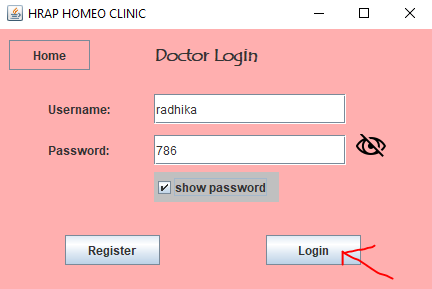
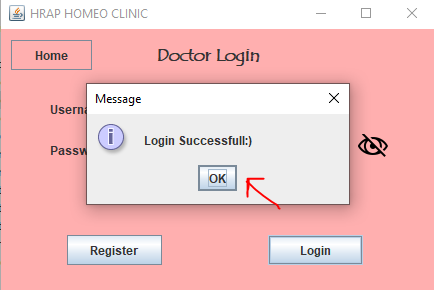
**Fig 5.1.v. Displays that patients registered under that particular doctor and generates ID**



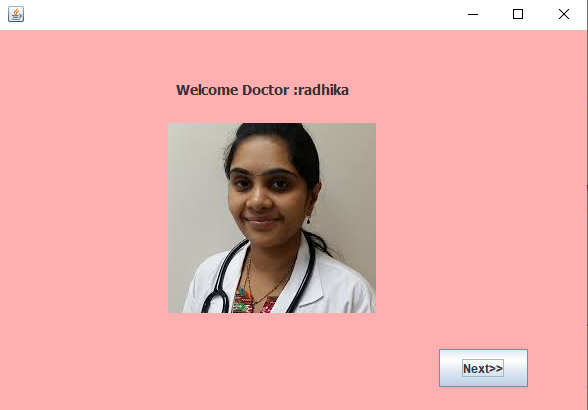
** Fig 5.1.vi. Staff enters the details of new patient and clicks add patient button**

**Fig 5.1.vii. That particular patient details added**

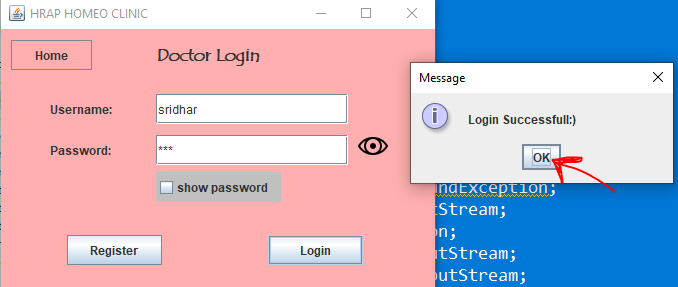
* 1. **MODULE : DOCTOR ACCESS**

****

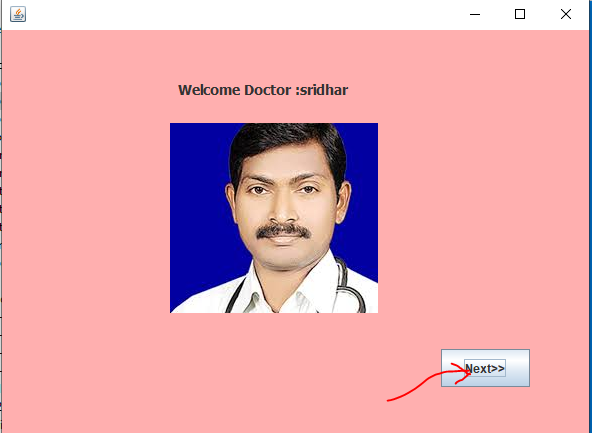
**Fig 5.2.i. Doctor Login(enter credentials) Fig 5.2.ii. Login Successful**

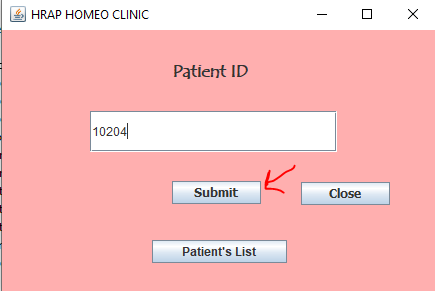
****

**Fig 5.2.iii. Display that respective doctor photo**

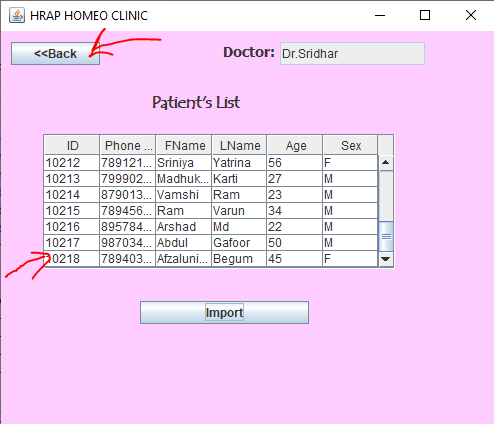
****

**Fig 5.2.iv. Another Doctor Login**

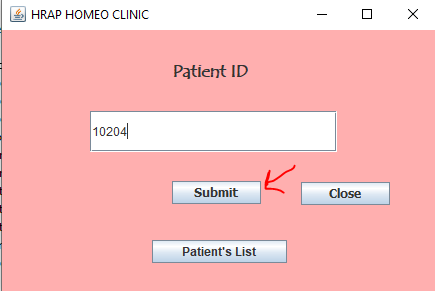
****

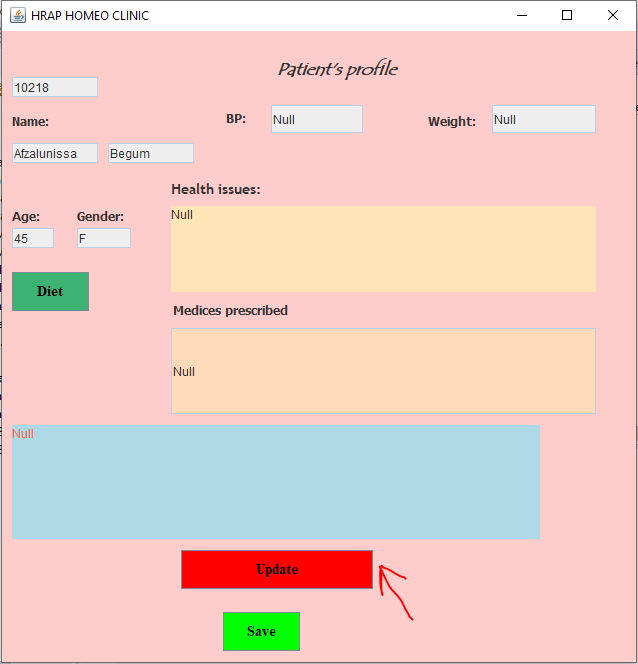
****

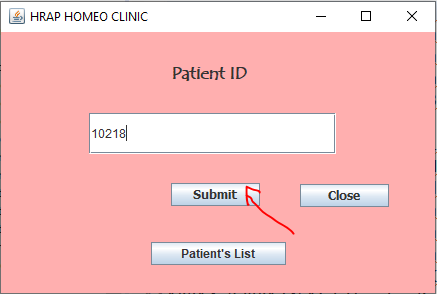
**Fig 5.2.v. Doctor Photo Display Fig 5.2.vi. Enter Patient ID**

****

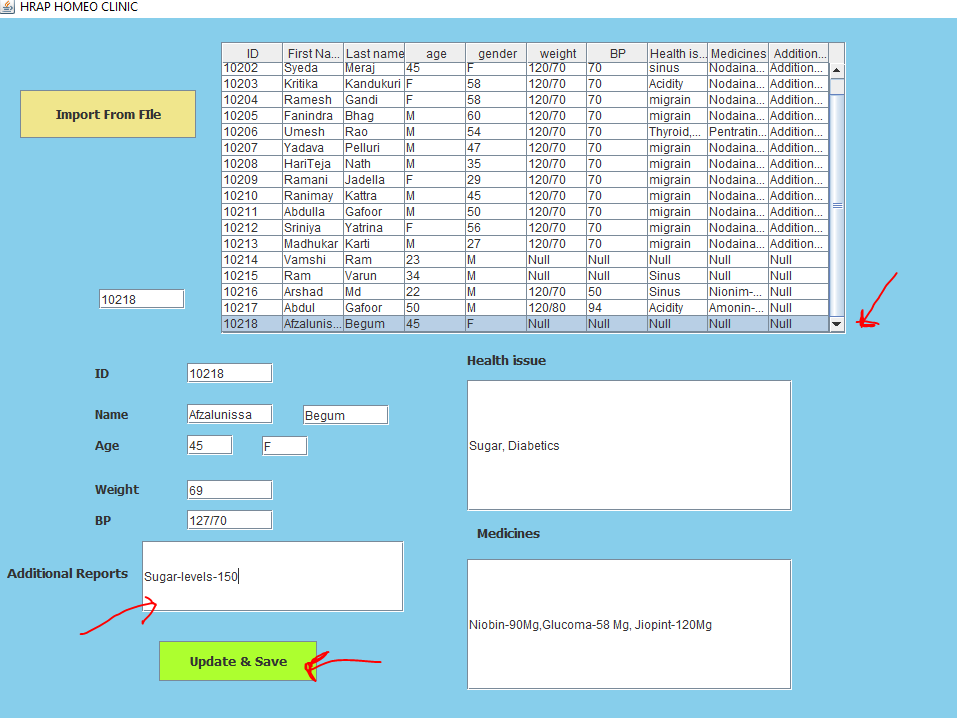
****

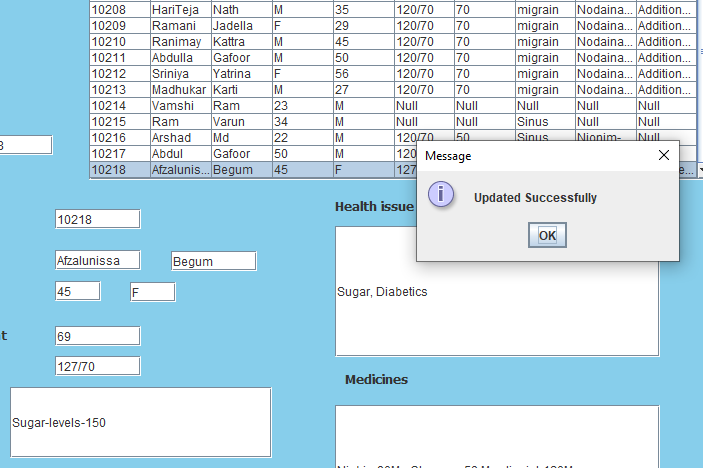
****

**Fig 5.2.vii. Old Patient Profile (generates)**

**Fig 5.2.viii.New Patient ID**

**Fig 5.2.ix. New Patient Profile (default have Null)**

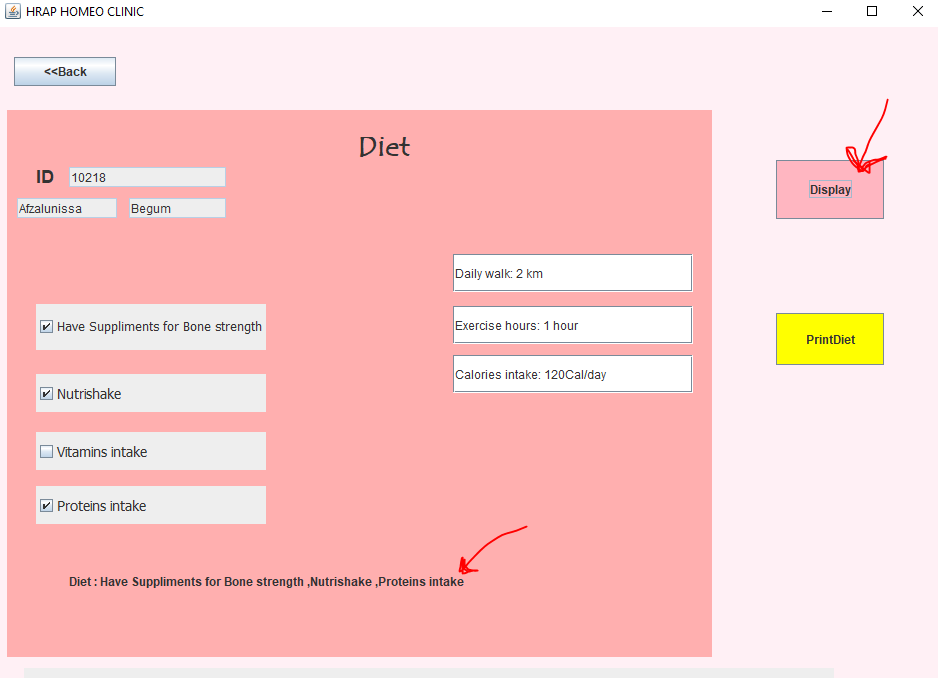
****

**Fig 5.2.x. Update profile (by selecting particular row)**

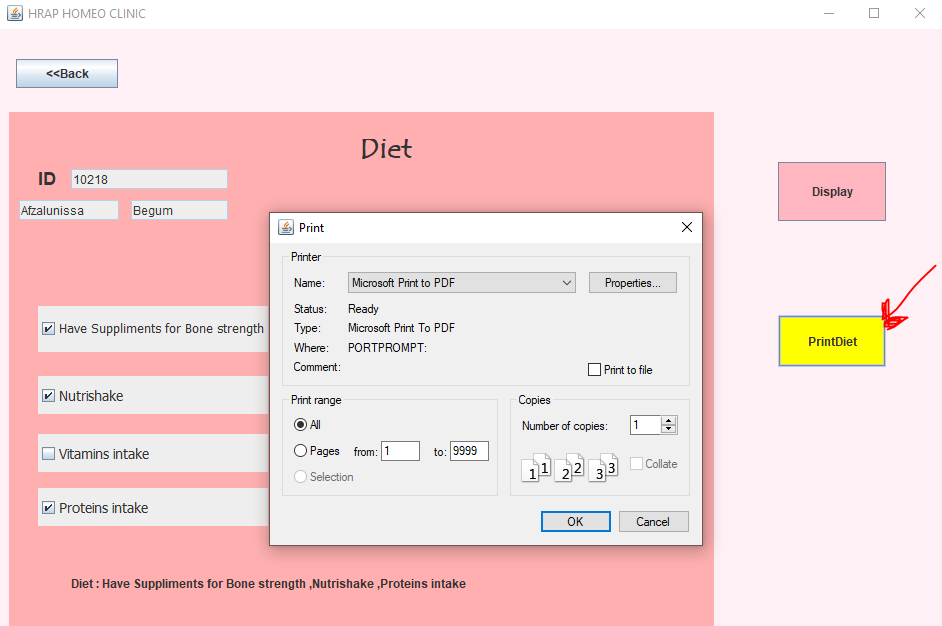
**Fig 5.2.xi. Updated Successfully**

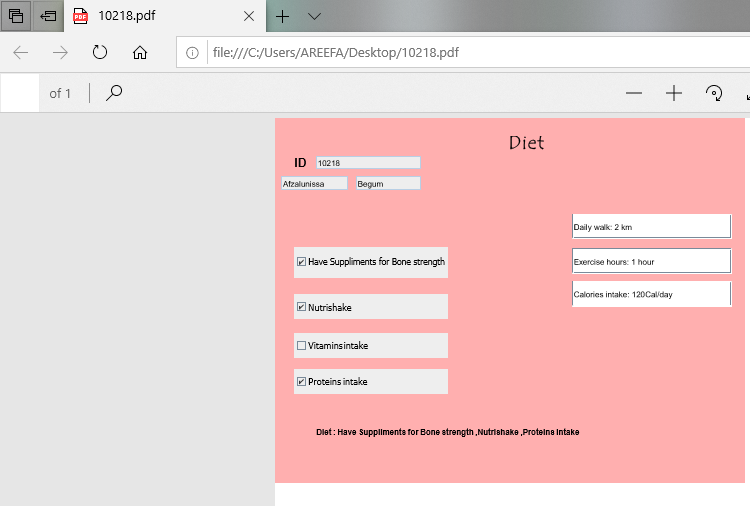
****

**Fig 5.2.xii. Updated Particular Patients Profile**

****

**Fig 5.2.xiii. That Patient Diet (Displayed)**

****

**Fig 5.2.xiv. That Patient Diet Exported to PrinterFig 5.2.xv. That Patient Diet Demo shown in Pdf(Printed)**

1. **CONCLUSION:**

The Smart – Health Care Management Portal for doctors is helpful to maintain the patient’s records and integrates both the doctor and the patient without any conflicts. By implementing this application portal in clinics, the proper management of the patient’s health related diagnosis and treatment is ensured. By this, patient need not to carry the previous appointment prescriptions or reports for the next appointment and need not to detail again about his/her health issue to the doctor.

Doctor can easily enter the patient’s particular ID and view the previous appointment history in the respective profile and make updates if any easily. Thus, there is no need of even writing the prescription onto the receipt by the doctor and this even avoids the conflicts by the doctor’s messy hand writing since, diet plan prescription would be given printed.

This application features will increase the efficiency, transparency. Additionally the same application can be utilized by the staff for registering the new patients by entering their details and generating ID’s. All the details of the patients are efficiently stored and retrieved from file. Thus, accuracy and memory management is also enhanced.

Thus, one java application can lead to solving the concerns between the patient and the doctor. At last, we would conclude that, health care management plays a major role in producing a happy outcome after a particular treatment and thus, health care management portal for doctors helps in implementing the above strategies.