****

**INDO ASIAN ACADEMY DEGREE COLLEGE**

**Kalyan Nagar, Bengaluru – 560043**

**(Year of Submission-2019)**

TITLE OF THE PROJECT WORK

**ORGAN & BLOOD DONATION MANAGEMENT SYSTEM**

**PROJECT REPORT SUBMITTED TO BANGALORE UNIVERSITY,**

**IN PARTIAL FULFILLMENT OF THE REQUIREMETNS FOR**

**THE AWARD OF THE DEGREE OF**

**(BCA - BACHELOR OF COMPUTER APPLICATIONS )**

**Submitted**

**By**

**ABDUL HAKIM (17P4SB7003)**

**BHARATH KS (17P4SB7011)**

**ASHOK REDDY V (17P4SB7009)**

**Under the Guidance of**

**Prof.Hidayathulla M.R.,**

**HOD,Department of Computer Science and Applications**

**(Name of the Guide)**

**Mrs. Keerthi**

**Department of Computer Science and Applications**

.



**CERTIFICATE**

This is to certify that this project entitled “**ORGAN AND BLOOD DONATION MANAGEMENT SYSYTEM”**

submitted in partial fulfilment of the degree of BACHELOR OF COMPUTER APPLCATIONS (BCA) for the academic year 2019-20 is done by Mr/Ms**. ABDUL HAKIM, BHARATH KS, ASHOK REDDY V**, bearing a Reg No. **17P4SB7003, 17P4SB7011, 17P4SB7009** is an authentic work carried out by him/her . The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

Prof. Hidayathulla M.R. Guide Name : Mrs. Keerthi

HOD, Department of Computer Science Department of Computer Science

Internal Examiner External Examiner

Place: INDO ASIAN ACADEMY DEGREE COLLEGE

Date:



**DECLARATION**

I hereby declare that the project, entitled **“ORGAN AND BLOOD DONATION MANAGEMENT SYSYTEM”**, submitted for the award Of BCA-Bachelor of Computer Applications is my original work and the Project has not formed the basis for the award of any Degree, Diploma, Fellowship or similar other titles. It has not been submitted to any University or Institution for the award of any degree or diploma.

Signature of the student:

**Name:**

**Abdul Hakim - 17P4SB7003**

**Bharath ks - 17P4SB7011**

**Ashok Reddy v - 17P4SB7009**

Place: INDO ASIAN ACADEMY DEGREE COLLEGE

Date:

****

**ACKNOWLEDGEMENT**

I take this opportunity to express my sincere thanks to our Founder Chairman **Prof. Dr. T. Ekambaram Naidu** for giving me the opportunity to pursue my BCA course.

It gives me immense pleasure to express my deep sense of gratitude to **Prof. SupriyoGuhoyRoy** , CEO, Indo Asian Academy group of Institutions

I would like to express my gratitude to **Prof. N. Bharathi**, Principal, Indo Asian Academy Degree College for her consistent co-operation and encouragement throughout the programme.

I would like to express my sincere thank to **Prof. Hidayathulla M.R.,** HoD, Department of Computer Science and Applications for giving me this opportunity to undertake this project.

I express my heartfelt thanks forever to **Mrs. Keerthi** Project guide for suggesting this work, invaluable and untiring guidance, encouragement, Unfailing support and valuable criticism throughout my project work.

I wish to record sincere thanks to all my colleagues, friends and my family members whose blessings made this task possible for me.

**CONTENTS**

**1. Introduction**

1.1 Overview of Project

1.2 Objectives

1.3 Modules

**2. Literature Survey**

2.1 Existing System

2.1.1 Drawbacks of Existing System

2.2 Proposed System

2.2.1 Merits of Proposed System

2.3 Overview of Software used

2.3.1 Introduction of Microsoft Visual Basic 6.0

2.3.1.1 The Programming Process

2.3.1.2Understanding Program Maintenance

2.3.1.3 Visual Basic is a Smart Language

2.3.1.4 Tool Box Control

2.3.1.5 Adding Control

2.3.1.6 Removing Control

2.3.1.7 Intrinsic Control

2.3.1.8 Label Control

2.3.1.9 Command Bottom

2.3.1.10 Text Control

2.3.1.11 what does VB 6.0 has

2.3.1.12 Visual Basic with Databases

2.3.1.13 Active X Data Object Components

2.3.2 Introduction of Oracle SQL\*Plus

**3. Design**

3.1 Requirement Analysis

3.2 Data Flow Diagram

3.3 Entity-Relationship Diagram

3.4 Database Design

**4. Coding**

**5. Testing**

**5**.1 Introduction to Testing

5.1.1 White Box Testing

5.1.2 Black Box Testing

5.1.3 Application Strategies

5.1.4 Unit Testing

5.1.5 Integration Testing

5.1.6 Validation Testing

5.1.7 System Testing

5.2 Multi-User System

5.3 Testing Results

5.3.1 Job seeker Module Testing

5.3.2 Client Module Testing

5.3.3 Administration Module Testing

**6. User Manual**

6.1 Screen Shots

**7. Conclusion**

**8.Future Enhancement**

**9. Bibliography**

**Chapter 1**

**Introduction**

**The Beginning** :-

**Organ donation** is the process when a person allows an organ of their own to be removed and transplanted to another person, legally, either by consent while the donor is alive or dead with the assent of the next of kin.

**Blood donation** occurs when a person voluntarily has blood drawn and used for [transfusions](https://en.wikipedia.org/wiki/Blood_transfusion) and/or made into biopharmaceutical medications by a process called [fractionation](https://en.wikipedia.org/wiki/Blood_fractionation) (separation of whole-blood components). Donation may be of whole blood, or of specific components directly (the latter called apheresis). [Blood banks](https://en.wikipedia.org/wiki/Blood_bank) often participate in the collection process as well as the procedures that follow it.

* 1. **Overview of Project**

The project entitled “**ORGAN AND BLOOD DONATION MANAGEMENT SYSYTEM**” is designed and developed using visual basic 6.0 for maintaining activities like , donor details, blood donation, organ donation, user details, request details, adding new donors, adding new users, admin privilege etc

* 1. **Objectives**

The primary **objective** is to promote awareness of life-saving solid **organ transplants**. The secondary **objective** is to promote awareness of tissue and life-enhancing **transplants**. To raise awareness that individuals can save lives and improve the health of others by **donating blood**.

It manages all the information about donor, organ, blood, user details.

The purpose of the project is to build an application program to reduce the manual work for managing the donor details, users, organs,bloods. It tracks all the details about the Users, Donors.

* 1. **Modules**

In this application we have two main modules

* Admin Functionalities
* User details
* Organ details
* Blood details
* Request details
* Adding new users

* User Functionalities
* Updating organ donors
* Updating blood donors
* Accepting donor service request

**Chapter 2**

**Literature Survey**

Everything on the earth either living or non-living has an evolution cycle. Like scientist has proved that human has completed an evolution cycle from ape to man. In same manner the computer has an evolution cycle, as technology is improving the computers are becoming more and more efficient. Similarly as the computer architecture is improving day by day, it also requires upgrading the previous systems accordingly.

As we have developed this application for the first time. Similar application’s of different companies are present in the market and we tried to study those, a brief discussion was carried out with the project guide to know more details about existing system. After discussion the various deficiencies in the system was identified these points are noted down and discussed with project guide again and decide where to make modification over existing system.

Since we are implementing our application in the new envoirnment and using the present technologies so as to make it more efficient, there was a need to understand the application’s environment and also about the new technologies. And while implementing the application various thing has been taken into consideration like application will also be usable for those end users who don’t understand english.

**2.1 Existing System:-**

The existing system is not automated fully as only the accounts are automated with the accounting software. The Customer details are maintained in excel which will take more time to search from the long lists. As the current data is maintained through ledgers and excel sheet so there is no proper reports for facilitation of the administration of the showroom.

**2.1.1 Disadvantages of the Existing System**

* Time consuming.
* Updating and Retrieval tasks are very tedious.
* Highly error prone.
* Lack of reports
* Vehicle Sales & customer details are maintained in excel Sheet which leads to the inaccurate information.
* Customer data are not maintained properly which leads the follow-ups slow.

**2.2 Proposed** **System**

The proposed system overcomes the disadvantages of the existing system and records the entire activities happening. The reports generated will enhance the efficiency of the application.

**2.2.1 Advantages of the proposed System**

* Security of data.
* Ensure data accuracy.
* User friendly and interactive.
* Entire activities are recorded through the system.
* Retrieval and Updating task can be performed much faster and easier.
* Donor Data is maintained.
* Reports generated will be more useful for management to take the quick business decisions.
* Donor database is maintained which will be helpful for intimating the donor details
* New donors can be added to the existing donors which will be an added advantage of this system.

**2.3 Overview of languages used**

**2.3.1 Introduction to MICROSOFT VISUAL BASIC 6.0**

Visual Basic 6 is Microsoft's latest and greatest version of the Visual Basic programming language. Although writing programs can be a tedious chore at times, Visual Basic reduces the effort required on your part and makes programming enjoyable. Visual Basic makes many aspects of programming as simple as dragging graphic objects onto the screen with your mouse.

Microsoft based Visual Basic on a *programming language* written for beginners called *BASIC.*

BASIC has been around for more than 35 years in one form or another. The original language designers wanted to develop a programming language that beginners could use. With BASIC, new programmers could become proficient right away.

*BASIC* stands for *Beginner's All-purpose Symbolic Instruction Code.* That's some abbreviation!

A *programming language* is a set of commands and command options, called *arguments that* you use to give instructions to the computer. Computers cannot (yet) understand human languages because people deal well with ambiguous commands, and a computer cannot understand such ambiguity. A programming language must be more precise than a spoken language.

**2.3.1.1 The Programming Process**

Over time you'll find your own way of writing programs that works best for you.

Nevertheless, you'll generally follow these standard set of steps when creating your Visual Basic programs:

1. Decide what your application is to do by creating an overall design.
2. Create the visual portion of your application (the screens and menus that your users will interact with).
3. Add Visual Basic programming language code to tie the visual elements together and to automate the program.
4. Test your application to locate and remove any bugs you find.
5. Compile your tested application and distribute the compiled application to your users.

**2.3.1.2 Understanding Program Maintenance**

Bugs are not the only reason that you will work on a program after you think you're completely done with it. *Program maintenance* is necessary because requirements change, companies change, and laws change. You must also change the programs you write so that they remain viable programs; you will need to update your program periodically to reflect changes that impact the program. In addition, users will think of new things that they want the program to do.

*Program maintenance* is the term used for the updating of a program after the program is put into use. This update may be a result of a user's request or a change in the way the program needs to operate.

It is said that a program is written once and modified many times. The more program maintenance you perform, the more likely that your program will be up-to-date and in use. You may want to release new versions of your program so that users can, with a different version number on the opening screen that you place there, keep track of the latest version installed on their system.

**2.3.1.3 Visual Basic Is a Smart Language**

The main reason why Visual Basic is so popular and powerful is the same reason behind the success of Windows. Microsoft took a complex technology (writing computer programs) and made it easier to use through a graphical interface. Suppose you have to write a program for your company. In a visual programming environment, you can quickly design the windows that the user sees by drawing and arranging them just as you would lay out elements for a newspaper.

In a text-based programming system, you control the user interface through program language commands. Common sense tells you that the visual programming method is easier for newcomers to learn and requires less time to maintain. In this case, the old adage “a picture is worth a thousand words” truly applies. However, do not let me give you the impression that Visual Basic is just another pretty interface.

Another key concept of Visual Basic is the ability to create and use self-contained components, or *objects*.

**2.3.1.4 Tool Box Controls**

The toolbox contains all the custom control needed for a visual basic application .several advanced controls can be added to the project from the menu bar tool , **Project ­– Components.** Each control has its own properties, Events and methods. The properties of a controls can be set either through code or in the properties window user – defined controls can also be created and added to the tool box. Controls work with multimedia and internet too.

**2.3.1.5 Adding controls**

Controls can be added to a form in two ways:

\* Double-click on a control in the toolbox and it appears on the form.

\* Click on the control icon. Move the mouse over the form, click and draw the control to the required size.

**2.3.1.6 Removing controls**

To remove a control from the form

\*Select the control.

\*Press the delete key.

Controls can be placed anywhere on the form. Their size can be changed by using the sizing handle which appear when a control in selected.

**2.3.1.7 Intrinsic controls**

The default controls which are automatically displayed in a toolbox when a form is loaded are known as intrinsic controls. The intrinsic controls in visual basic are very powerful, since objects such as buttons, labels, text boxes, and combo or list boxes can be added to a form easily and coded.

Some common intrinsic controls which are described below are:

1. Label
2. Command button
3. Text box
4. Frame
5. Option button
6. Picture box
7. Image box

**2.3.1.8 Label**

Label control allows the user to display text on a form, which does not change.

For example Label1.caption=”Name;”

Generally labels are used to print captions for other controls. Text printed on labels cannot be edited during runtime.

**2.3.1.9 Command buttons**

Commands buttons is one of the most powerful and frequently used controls in windows application. It is used to invoke response from the user or to invoke special functions on the form.

**2.3.1.10 Textbox control**

Textbox control is one of the most popular and versatile mechanism used to display and enters text in a window user interface. It behaves like a mini text editor providing all the basic text editing facilities.

**Basic text properties**

The basic properties of a textbox determine the appearance and functionality of a textbox control. Some of them are as described below:

**Enabled:** Indicates whether the user can interact with the control or not.

**Index:** Determine the control array index.

**Locked:**  Boolean. Indicates whether the user can type in the textbox or not.

**Max Length:** Indicates the maximum number of characters input in the textbox. Default values are 0, which means any number of characters. The maximum of text can be typed in a textbox is nearly **64 kb.**

Max length property can be set to a specific value restricting the user to type only that many character

.

**Multi Line:** Specifies whether the textbox will hold a single line or multiple lines.

**Password char:** Specifies the masking character for text displayed in the textbox. Suppose the password is set to “\*”, the user than sees an asterisk in the place of every character typed in that textbox.

**Scrollbars:** Indicates horizontal scrollbars, vertical scrollbars or both for the textbox. This property is used with multiline property. Multiline textboxes can have horizontal, vertical or both scroll bars. If a horizontal scrollbar is attached to a text box, the text will not wrap automatically. The user must press enter to start new line.

**Text:** Specifies the text in the textbox.

**Alignment:** Alignment of text in a textbox can be Left-justified, centered or right-justified.

**SelText:** Returns the selected text. A selected text can be converted to uppercase by the following code:

Text1. SelText= Ucase(Text1.SelText)

**SelStart:** Specifies the position of the first character of the selected text.

**SetLength:** Returns the length of the selected text. The most common use of these properties is

to select a piece of text at runtime to select the text in a textbox.

Text1.selstart=1

Text1.selLength=Len (Text1)

**Click:** Click event is called when the text box is clicked with the mouse.

**Key press:** Occurs every time a key is pressed and stores the Ascii code of the key pressed in keyascii(an integer type). This event can be used to restrict the user to type only certain characters.

Controls are elements you can use when designing a user interface.

Visual Basic controls enable you to add features to your programs without you having to be involved in the details of how these features work. For example, receiving input from a user of your program is as simple as drawing a control that accepts input. This is a great advantage of visual programming languages—you can concentrate on what you want your program to do, not how to get the programming language to do it.

**2.3.1.11What does VB6.0 have….**

Just about everything we could write about Visual Basic is, from language reference to ADO database handling, from creating Web browsers to dragging and dropping data across applications, from email applications to multimedia players, from creating ActiveX controls and ActiveX

Documents to setup programs, it’s all here.

Here’s some of what we’ll see:

**•** ActiveX controls

**•** ActiveX documents

**•** ADO, DAO, and RDO database applications

**•** Multimedia AVI, MPG, WAV, and MID players

**•** CD players that play CDs from the computer’s CD-ROM drive

**•** Bitmapped menu items

**•** Full Web browsers

**•** Pie charts, line charts, bar charts, and others

**•** Code clients that call methods in programs like Microsoft Excel

**•** Code components (OLE automation servers)

**•** Graphics animation

**•** Applications that use the Windows Common Dialogs

**•** Customized toolbars with embedded controls like combo boxes

**•** Data entry forms

**•** Database editing applications

**•** Direct connections to the Windows API

**•** Direct connections to code written in Visual C++

**•** Drag/drop operations

**•** Graphics applications that draw arcs, circles, rectangles, lines, and more

**•** Email applications

**•** Error handlers

**•** Applications that use the printer

**•** Word processor applications

**•** File handlers for text and binary data

**•** FTP applications

**•** Dialog boxes

**•** Windows Help files

**•** MDI applications

**•** Pop-up menus activated with right mouse clicks

**•** Application deployment

**•** HTTP applications

**•** Image handling: blur, emboss, engrave, flip, sweep, stretch images, and more

**•** OLE applications

**•** Applications that use the Windows Registry

**•** List views and tree views

**•** Applications that create controls at runtime

**•** Mouse capture

**•** OLE drags (dragging data between applications)

**•** Online user registration

**•** Picture clip applications

**•** Setup programs

**•** Screen capture

**•** Spreadsheets

**•** Status bars and toolbars

**•** Tab strips, progress bars, and others

**2.3.1.12 Visual Basic with Databases**

A **database** is a collection of **information**. This information is stored in a very structured manner. By exploiting this known structure, we can access and modify the information quickly and correctly.

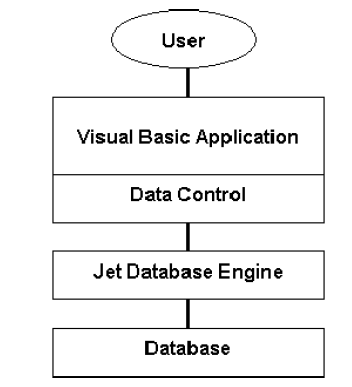
For database management, Visual Basic application acts as a **front**-**end** to the database. This means the Visual Basic application provides the **interface** between the user and the database. This interface allows the user to tell the database what he or she needs and allows the database to respond to the request displaying the requested information in some manner.

A Visual Basic application cannot directly interact with a database. There are two intermediate components between the application and the database: the **data control** and the **database engine**:

The **data control** is a Visual Basic object that connects the application to the database via the database engine. It is the conduit between the application and the engine, passing information back and forth between the two.

The **database engine** is the heart of a Visual Basic database management system. It is the actual application that does the management. Having this engine saves programmers a lot of work. The database engine native to Visual Basic is known as the **OLE DB** provider. It is the same provider used by Oracle SQL Plus for database management. Hence, it is primarily used to

work with SQL databases, but it can also work with others.



**OLE DB Provider for Oracle**

**2.3.1.13ActiveX Data Object Components**

The first step in editing an ADO database is to open that database, which is called a *data source* in ADO terminology, by setting up a Connection object.

To use that and other ADO objects in code, use the Project🡪|References item, select the Microsoft ActiveX Data Objects Library item, and click on OK, adding the ADO Object Library to your program.

Now create a new ADO Connection object with the Connection object’s **Open** method:

Here are the arguments for this method:

**•** *Connection String*—String containing connection information.

**•** *UserID*—String containing a username to use when establishing the connection.

**•** *Password*—String containing a password to use when establishing the connection.

**•** *OpenOptions*—if set to **adConnectAsync**, the connection will be opened asynchronously.

Here are the possible values for the *Type* argument:

**• dbOpenKeyset**—Opens a dynaset-type Recordset object, which is like an ODBC keyset cursor.

**• dbOpenDynamic**—Opens a dynamic-type Recordset object, which lets the application see changes made by other users.

**• dbOpenStatic**—Opens a static-type Recordset object.

**• dbOpenForwardOnly**—Opens a forward-only-type Recordset object, where you can only use

**MoveNext** to move.

Here are the possible values for the *LockType* argument:

**• AdLockReadOnly—**The default; read-only.

**• adLockPessimistic**—Pessimistic locking, record by record.

**• adLockOptimistic**—Optimistic locking, record by record.

**• adLockBatchOptimistic**—Optimistic batch updates.

Here are the possible values for the *Options* argument:

**• adCmdText**—Provider should evaluate *Source* as a definition of a command.

**• adCmdTable**—ADO should generate an SQL query to return all rows from the table named in *Source.*

**• adCmdTableDirect**—Provider should return all rows from the table named in *Source*.

**• adCmdStoredProc**—Provider should evaluate *Source* as a stored procedure.

**• adCmdUnknown**—Type of command in the *Source* argument is not known.

**• adCommandFile**—Record set should be restored from the file named in *Source*.

**• adExecuteAsync**—*Source* should be executed asynchronously.

**• adFetchAsync**—After the initial quantity specified in the **Cache Size** property is fetched, any remaining rows should be fetched asynchronously.

**2.3.2 Introduction to Oracle SQL\*Plus**

**Introduction and Objectives**

SQL\*Plus is the command-line interface to the Oracle database. Its fundamental reason for existence is to allow you to enter and execute ad hoc SQL statements and PL/SQL code blocks. This chapter explains what SQL\*Plus is, how it relates to other Oracle tools (as well as the database), and why you should master it. At the end of the chapter I’ll introduce you to the sample data, which is used for many of the examples in this book. If you like, you can load that data into your database and test out each example as you go through this book.

**Definitions**

These words are used often in SQL so you will want to become familiar with them before using the program and this tutorial.

**Relational Database**:

In relational databases such as SQL, data is stored in tables made up of one or more fields

(SQL calls a column afield). The data stored in each1column must be of a single data type

such as Character, Number or Date. A collection of values from each column of a table is called are cordon arrow in the table .Different tables can have the same column in common. This feature is used to explicitly specify a relationship between two tables. Values appearing in column A in one table are shared with another table

**Table**:

Tables are the main units of data storage in a database. A table is a collection of data about a specific topic; it is made up of one of more fields.

**Field**:

A field is a column in a table and defines a data type for a set of values in a table. For example, a mailing list table might include fields for first name, last name, address, city, state, zip code, and telephone number.

**Record**:

A record in a row in a table and is a set of values defined by fields. In a mailing list table, each record would contain the data for one person as specified by the intersecting fields.

**Data type**:

Data types are the properties of each field. A field only has one data type, such as Character, Number or Date.

**Primary Key:**

A primary key is a value that can be used to identify a unique record in a table.

**Design View**:

It provides the tools for creating fields in a table.

**Datasheet View**:

It allows you to update, edit, and delete information from a table.

**Chapter 3**

**Design**

**3.1 Requirement Analysis**

**Application Requirements**

|  |
| --- |
| **Operating system** Windows 7 |
| **Front End** Microsoft Visual Basic 6.0 |
| **Database** SQL\*Plus |
| **Connectivity** ActiveX Data Object |
| **Report** Data Report |

**Hardware Requirements**

|  |
| --- |
| **RAM** 1GB or higher |
| **Hard Disk** 10GB or higher |
| **Computer Processor** CORE i3 |
| **Clock Speed** 700MHZ Processor |

**3.2 Data flow diagram notations**

The dataflow diagram (DFD**)** is the one of the most important modeling tools. It shows the use of the data pictorially. DFD represents the flow of data between different transformations and process in the system. The dataflow shows logical flow of the data. Different notations used in DFD are:

**Functional Processing**

It is represented by an diamond. This notation specifies the processing or main transactions.

**Data Flow**

An arrow line represents it and name of the data is specified by the side of

the line as label. This arrow line is used to indicate data movement.

**Source or Sink**

It is represented by a rectangle .It is used for specifying from where data is coming and where it will reach.

**Login form**

**Login**

**Admin**

**User**

**Low Level DFD**

**Level 1 DFD Admin**

Login

Login Success

Donor Details

Request details

Organ Details

Blood Details

Admin

User Details

Add Users

**Level 1 DFD Employee**

Login

Login Success

Add new Blood donors

Add new Organ donors

User

Accept requests

**Level 2 DFD of Adminstrator**

Login

Login Success

Admin

User

User Details

Donor Details

Blood Donor Details

Organ Donors Details

Request Details

Add User

Add organ donors

Add blood donors

Accept request Details

**3.3 E-R Diagram**

PASSWORD

USERNAME

**LOGIN**

USER

ADMIN

ACCEPT REQUEST

ADD ORGAN DONORS

BLOOD DONOR DETAILS

ORGAN DONOR DETAILS

USER DETAILS DETAILS

ADD DONORS

ADD BLOOD DONORS

**FUNCTIONALITIES**

NEW USER

REQUEST DETAILS

**FUNCTIONALITIES**

**3.4 Data Base Design**

**3.4.1 Introduction to Database Design**

Data base design is concerned with the data focus from the perspective of the system designer. The end

product is called a database schema, a technical blueprint of database.

Database design translates the data models that were developed for the system users during the definition

phase in to data structures supported by the chosen database technology. The goals of database design are

as follows:

* A database should provide for the efficient storage, update and retrieval of data.
* A database should be reliable-the stored data should have high integrity to promote user trust in that data.
* A database should be adaptable and scalable to new and unforeseen requirements and applications.

**3.4.2 Tables Created**

**Table 1: Admin Login**

|  |  |
| --- | --- |
| **Field Name** | **Data Type** |
| Username | Text |
| Password | Number or Text |

**Table 2: User Login**

|  |  |
| --- | --- |
| **Field Name** | **Data Type** |
| Username | Text |
| Password | Number or Text |

**Table 3: User Details**

|  |  |
| --- | --- |
| **Field Name** | **Data Type** |
| UserName | Text |
| Userid | Text and Number |
| Gender | Text |
| Password | Text |
| Address | Text |
| Email | Text |
| Contact No | Number |

**Table 4: Blood Donor Details**

|  |  |
| --- | --- |
| **Field Name** | **Data Type** |
| Name | Text |
| Age | Text |
| DateOfBirth | Date |
| Blood Type | Text |
| Weight | Number |
| Email id | Text |
| Address | Text |
| City | Text |
| Pincode | Number |
| Mobile | Number |

**Table 4: Organ Donor Details**

|  |  |
| --- | --- |
| **Field Name** | **Data Type** |
| DonorName | Text |
| Age | Text |
| DateOfBirth | Date |
| Gender | Text |
| Email id | Text |
| Address | Text |
| City | Text |
| Pincode | Number |
| Mobile | Number |
| Organ | Text |

**Table 4: Request Details**

|  |  |
| --- | --- |
| **Field Name** | **Data Type** |
| Name | Text |
| Email id | Text |
| Address | Text |
| City | Text |
| Pincode | Number |
| Mobile | Number |
| Organ | Text |
| Blood | Text |

**Chapter 4**

**Coding**

**Form 1**

Private Sub Command1\_Click()

Label5.Visible = True

ProgressBar1.Visible = True

Timer1.Enabled = True

End Sub

Private Sub Form\_Load()

ProgressBar1.Max = 3

ProgressBar1.Value = 0

End Sub

Private Sub Timer1\_Timer()

If ProgressBar1.Value < ProgressBar1.Max Then

ProgressBar1.Value = ProgressBar1.Value + 1

Else

Timer1.Enabled = False

login.Show

End If

End Sub

**Login**

Private Sub Command1\_Click()

user\_login.Show

End Sub

Private Sub Command2\_Click()

admin\_login.Show

End Sub

Private Sub Command3\_Click()

create\_account.Show

End Sub

Private Sub Command5\_Click()

End

End Sub

**Admin Login**

Private Sub Command1\_Click()

If apassword.Text = "" Then

MsgBox "Enter valid 'password'", vbCritical

ElseIf apassword.Text = "iladmin00" Then

MsgBox "Welcome Mr.Admin", vbInformation

apassword.Text = ""

admin\_home.Show

Else

MsgBox "Invalid Password", vbCritical

apassword.Text = ""

End If

End Sub

Private Sub Command2\_Click()

apassword.Text = ""

login.Show

End Sub

Private Sub Command3\_Click()

a = InputBox("Enter your Tag")

If a = "iladmin00" Then

apassword.Text = a

MsgBox "Password is 'iladmin00'", vbInformation

Else

MsgBox "Incorrect", vbCritical

End If

End Sub

**Admin Functionalities**

**Admin Home**

Private Sub Command1\_Click()

login.Show

End Sub

Private Sub Command2\_Click()

user\_details.Show

End Sub

Private Sub Command3\_Click()

blood\_details.Show

End Sub

Private Sub Command4\_Click()

organ\_details.Show

End Sub

Private Sub Command5\_Click()

request\_details.Show

End Sub

Private Sub Command6\_Click()

create\_account.Show

End Sub

**User Details**

Dim con As New ADODB.Connection

Dim cmd As New ADODB.Command

Dim rs As New ADODB.Recordset

Dim cn As String

Dim d As String

Private Sub Command1\_Click()

cmd.ActiveConnection = con

con.CursorLocation = adUseClient

rs.Open "select \* from user\_account", con, adOpenForwardOnly, adLockOptimistic

Set details.DataSource = rs

Label14.Caption = rs.RecordCount

Command4.Enabled = True

Command2.Enabled = True

Command6.Enabled = True

Command7.Enabled = True

Command8.Enabled = True

Command9.Enabled = True

Command1.Enabled = False

End Sub

Private Sub Command2\_Click()

DataReport1.Show

End Sub

Private Sub Command4\_Click()

d = MsgBox("Are you sure you want to remove this User?", vbQuestion + vbYesNo, "DELETE?")

If d = vbYes Then

rs.Delete

MsgBox "User Removed Successfully", vbInformation

End If

End Sub

Private Sub Command5\_Click()

If Command1.Enabled = False Then

con.Close

Command1.Enabled = True

Command4.Enabled = False

Command2.Enabled = False

Command6.Enabled = False

Command7.Enabled = False

Command8.Enabled = False

Command9.Enabled = False

con.Open

admin\_home.Show

Else

Command1.Enabled = True

Command4.Enabled = False

Command2.Enabled = False

Command6.Enabled = False

Command7.Enabled = False

Command8.Enabled = False

Command9.Enabled = False

admin\_home.Show

End If

End Sub

Private Sub Command6\_Click()

rs.MoveFirst

End Sub

Private Sub Command7\_Click()

rs.MoveNext

If rs.EOF Then

rs.MoveFirst

ElseIf rs.BOF Then

rs.MoveFirst

End If

End Sub

Private Sub Command8\_Click()

rs.MovePrevious

If rs.EOF Then

rs.MoveFirst

ElseIf rs.BOF Then

rs.MoveFirst

End If

End Sub

Private Sub Command9\_Click()

rs.MoveLast

If rs.EOF Then

rs.MoveFirst

End If

End Sub

Private Sub Form\_Load()

cn = "Provider=MSDAORA.1;Password=open;User ID=scott;Persist Security Info=True"

con.Open cn

End Sub

**Blood Donor Details**

Dim con As New ADODB.Connection

Dim cmd As New ADODB.Command

Dim rs As New ADODB.Recordset

Dim cn As String

Dim d As String

Private Sub Command1\_Click()

cmd.ActiveConnection = con

con.CursorLocation = adUseClient

rs.Open "select \* from blood\_donors", con, adOpenForwardOnly, adLockOptimistic

Set details.DataSource = rs

Label14.Caption = rs.RecordCount

Command4.Enabled = True

Command2.Enabled = True

Command6.Enabled = True

Command7.Enabled = True

Command8.Enabled = True

Command9.Enabled = True

Command1.Enabled = False

End Sub

Private Sub Command2\_Click()

DataReport3.Show

End Sub

Private Sub Command4\_Click()

d = MsgBox("Are you sure you want to remove this Donor?", vbQuestion + vbYesNo, "DELETE?")

If d = vbYes Then

rs.Delete

MsgBox "Donor Removed Successfully", vbInformation

End If

End Sub

Private Sub Command5\_Click()

If Command1.Enabled = False Then

con.Close

Command1.Enabled = True

Command4.Enabled = False

Command2.Enabled = False

Command6.Enabled = False

Command7.Enabled = False

Command8.Enabled = False

Command9.Enabled = False

con.Open

admin\_home.Show

Else

Command1.Enabled = True

Command4.Enabled = False

Command2.Enabled = False

Command6.Enabled = False

Command7.Enabled = False

Command8.Enabled = False

Command9.Enabled = False

admin\_home.Show

End If

End Sub

Private Sub Command6\_Click()

rs.MoveFirst

End Sub

Private Sub Command7\_Click()

rs.MoveNext

If rs.EOF Then

rs.MoveFirst

ElseIf rs.BOF Then

rs.MoveFirst

End If

End Sub

Private Sub Command8\_Click()

rs.MovePrevious

If rs.EOF Then

rs.MoveFirst

ElseIf rs.BOF Then

rs.MoveFirst

End If

End Sub

Private Sub Command9\_Click()

rs.MoveLast

If rs.EOF Then

rs.MoveFirst

End If

End Sub

Private Sub Form\_Load()

cn = "Provider=MSDAORA.1;Password=open;User ID=scott;Persist Security Info=True"

con.Open cn

End Sub

**Organ Donor Details**

Dim con As New ADODB.Connection

Dim cmd As New ADODB.Command

Dim rs As New ADODB.Recordset

Dim cn As String

Dim d As String

Private Sub Command1\_Click()

cmd.ActiveConnection = con

con.CursorLocation = adUseClient

rs.Open "select \* from organ\_donors", con, adOpenForwardOnly, adLockOptimistic

Set details.DataSource = rs

Label14.Caption = rs.RecordCount

Command1.Enabled = True

Command2.Enabled = True

Command4.Enabled = True

Command6.Enabled = True

Command7.Enabled = True

Command8.Enabled = True

Command9.Enabled = True

Command1.Enabled = False

End Sub

Private Sub Command2\_Click()

DataReport2.Show

End Sub

Private Sub Command4\_Click()

d = MsgBox("Are you sure you want to remove this Donor?", vbQuestion + vbYesNo, "DELETE?")

If d = vbYes Then

rs.Delete

MsgBox "Donor Removed Successfully", vbInformation

End If

End Sub

Private Sub Command5\_Click()

If Command1.Enabled = False Then

con.Close

Command1.Enabled = True

Command4.Enabled = False

Command2.Enabled = False

Command6.Enabled = False

Command7.Enabled = False

Command8.Enabled = False

Command9.Enabled = False

con.Open

admin\_home.Show

Else

Command1.Enabled = True

Command4.Enabled = False

Command2.Enabled = False

Command6.Enabled = False

Command7.Enabled = False

Command8.Enabled = False

Command9.Enabled = False

admin\_home.Show

End If

End Sub

Private Sub Command6\_Click()

rs.MoveFirst

End Sub

Private Sub Command7\_Click()

rs.MoveNext

If rs.EOF Then

rs.MoveFirst

ElseIf rs.BOF Then

rs.MoveFirst

End If

End Sub

Private Sub Command8\_Click()

rs.MovePrevious

If rs.EOF Then

rs.MoveFirst

ElseIf rs.BOF Then

rs.MoveFirst

End If

End Sub

Private Sub Command9\_Click()

rs.MoveLast

If rs.EOF Then

rs.MoveFirst

End If

End Sub

Private Sub Form\_Load()

cn = "Provider=MSDAORA.1;Password=open;User ID=scott;Persist Security Info=True"

con.Open cn

End Sub

**Request Details**

Dim con As New ADODB.Connection

Dim cmd As New ADODB.Command

Dim rs As New ADODB.Recordset

Dim cn As String

Dim d As String

Private Sub Command1\_Click()

cmd.ActiveConnection = con

con.CursorLocation = adUseClient

rs.Open "select \* from request\_details", con, adOpenForwardOnly, adLockOptimistic

Set details.DataSource = rs

Label14.Caption = rs.RecordCount

Command4.Enabled = True

Command2.Enabled = True

Command6.Enabled = True

Command7.Enabled = True

Command8.Enabled = True

Command9.Enabled = True

Command1.Enabled = False

End Sub

Private Sub Command2\_Click()

DataReport4.Show

End Sub

Private Sub Command4\_Click()

d = MsgBox("Are you sure you want to remove this Request?", vbQuestion + vbYesNo, "DELETE?")

If d = vbYes Then

rs.Delete

MsgBox "Request Removed Successfully", vbInformation

End If

End Sub

Private Sub Command5\_Click()

If Command1.Enabled = False Then

con.Close

Command1.Enabled = True

Command1.Enabled = False

Command4.Enabled = False

Command2.Enabled = False

Command6.Enabled = False

Command7.Enabled = False

Command8.Enabled = False

Command9.Enabled = False

con.Open

admin\_home.Show

Else

Command1.Enabled = True

Command4.Enabled = False

Command2.Enabled = False

Command6.Enabled = False

Command7.Enabled = False

Command8.Enabled = False

Command9.Enabled = False

admin\_home.Show

End If

End Sub

Private Sub Command6\_Click()

rs.MoveFirst

End Sub

Private Sub Command7\_Click()

rs.MoveNext

If rs.EOF Then

rs.MoveFirst

ElseIf rs.BOF Then

rs.MoveFirst

End If

End Sub

Private Sub Command8\_Click()

rs.MovePrevious

If rs.EOF Then

rs.MoveFirst

ElseIf rs.BOF Then

rs.MoveFirst

End If

End Sub

Private Sub Command9\_Click()

rs.MoveLast

If rs.EOF Then

rs.MoveFirst

End If

End Sub

Private Sub Form\_Load()

cn = "Provider=MSDAORA.1;Password=open;User ID=scott;Persist Security Info=True"

con.Open cn

End Sub

**Adding New User**

Dim cn As String

Dim con As New ADODB.Connection

Private Sub Command1\_Click()

If nametxt.Text = "" Or useridtxt.Text = "" Or passwordtxt.Text = "" Or addresstxt.Text = "" Or mobiletxt.Text = "" Then

MsgBox "Fill all the fields", vbCritical

ElseIf Check1.Value = 0 Then

MsgBox ("Confirm the entered details are true")

ElseIf Len(passwordtxt.Text) < 8 Then

Shape10.Visible = True

MsgBox "Weak password (minimum of 8 character)", vbExclamation

Else

Adodc1.RecordSource = "select \* from user\_account"

Adodc1.Recordset.AddNew

Adodc1.Recordset(0) = nametxt.Text

Adodc1.Recordset(1) = useridtxt.Text

Adodc1.Recordset(2) = passwordtxt.Text

If male.Value = True Then

Adodc1.Recordset(3) = male.Caption

ElseIf female.Value = True Then

Adodc1.Recordset(3) = female.Caption

ElseIf other.Value = True Then

Adodc1.Recordset(3) = other.Caption

End If

Adodc1.Recordset(4) = addresstxt.Text

Adodc1.Recordset(5) = emailtxt.Text

Adodc1.Recordset(6) = mobiletxt.Text

Adodc1.Recordset.Update

MsgBox "Your account is sucessfully created", vbInformation

nametxt.Text = ""

useridtxt.Text = ""

passwordtxt.Text = ""

addresstxt.Text = ""

emailtxt.Text = ""

mobiletxt.Text = ""

Shape10.Visible = False

male.Value = False

female.Value = False

other.Value = False

Check1.Value = 0

admin\_home.Show

End If

End Sub

Private Sub Command2\_Click()

user\_details.Show

End Sub

Private Sub Form\_Load()

cn = "Provider=MSDAORA.1;Password=open;User ID=scott;Persist Security Info=True"

End Sub

Private Sub gendertxt\_keypress(Ascii As Integer)

If Ascii = 13 And gendertxt.Text <> "" Then

addresstxt.SetFocus

ElseIf (Ascii < 65 And Ascii <> 8 And Ascii <> 32) Or (Ascii > 90 And Ascii < 97) Or (Ascii > 122) Then

Ascii = 0

MsgBox "Enter Letter Only", vbInformation

End If

End Sub

Private Sub mobiletxt\_keypress(Ascii As Integer)

If Ascii = 13 And mobiletxt.Text <> "" Then

mobiletxt.SetFocus

ElseIf (Ascii < 48 And Ascii <> 8) Or Ascii > 57 Then

Ascii = 0

MsgBox "Enter Numbers Only", vbInformation

End If

End Sub

Private Sub nametxt\_keypress(Ascii As Integer)

If Ascii = 13 And nametxt.Text <> "" Then

useridtxt.SetFocus

ElseIf (Ascii < 65 And Ascii <> 8 And Ascii <> 32) Or (Ascii > 90 And Ascii < 97) Or (Ascii > 122) Then

Ascii = 0

MsgBox "Enter Letter Only", vbInformation

End If

End Sub

**User Login**

Dim a

Dim con As New ADODB.Connection

Dim cmd As New ADODB.Command

Dim cn As String

Dim b As String

Private Sub Command1\_Click()

If Text1.Text = "" And Text2.Text = "" Then

MsgBox "Enter valid 'username' and 'password'", vbExclamation

ElseIf Text1.Text = "" Then

MsgBox "Invalid 'Username'", vbInformation

ElseIf Text2.Text = "" Then

MsgBox "Invalid 'Password'", vbInformation

Else

cmd.ActiveConnection = con

con.CursorLocation = adUseClient

ulAdodc1.RecordSource = "select \* from user\_account where username='" + Text1.Text + "' and password='" + Text2.Text + "'"

ulAdodc1.Refresh

If ulAdodc1.Recordset.EOF Then

MsgBox "Invalid 'Username' and 'Password'", vbInformation

Else

MsgBox "Welcome " + Text1.Text + "", vbInformation

home.uname.Caption = "[ " + Text1.Text + " ]"

home.Show

Text1.Text = ""

Text2.Text = ""

End If

End If

End Sub

Private Sub Command2\_Click()

Text1.Text = ""

Text2.Text = ""

login.Show

End Sub

Private Sub Command3\_Click()

If Text1.Text = "" Then

MsgBox "Please enter the 'Username' for 'password'", vbInformation

Else

a = InputBox("Enter your UserID")

End If

cmd.ActiveConnection = con

con.CursorLocation = adUseClient

ulAdodc1.RecordSource = "select \* from user\_account where userid='" + a + "'"

ulAdodc1.Refresh

If ulAdodc1.Recordset.EOF Then

MsgBox "Invalid UserID", vbInformation

Else

ulAdodc1.RecordSource = "select password from user\_account where username='" + Text1.Text + "'"

Text2.Text = ulAdodc1.Recordset(2)

MsgBox "Password is " + Text2.Text + "", vbInformation

End If

End Sub

Private Sub Form\_Load()

cn = "Provider=MSDAORA.1;Password=open;User ID=scott;Persist Security Info=True"

con.Open cn

End Sub

Private Sub Text1\_keypress(Ascii As Integer)

If Ascii = 13 And Text1.Text <> "" Then

Text2.SetFocus

ElseIf (Ascii < 65 And Ascii <> 8 And Ascii <> 32) Or (Ascii > 90 And Ascii < 97) Or (Ascii > 122) Then

Ascii = 0

MsgBox "Enter Letter Only", vbInformation

End If

End Sub

**User Home**

Private Sub Command1\_Click()

login.Show

End Sub

Private Sub Command2\_Click()

organ\_reg.Show

End Sub

Private Sub Command3\_Click()

blood\_reg.Show

End Sub

Private Sub Command4\_Click()

request\_reg.Show

End Sub

**Organ Donor Registeration**

Dim cn As String

Dim con As New ADODB.Connection

Private Sub Command2\_Click()

home.Show

End Sub

Private Sub Form\_Load()

cn = "Provider=MSDAORA.1;Password=open;User ID=scott;Persist Security Info=True"

End Sub

Private Sub ocancel\_Click()

oname.Text = ""

oage.Text = ""

odob.Text = ""

oaddress.Text = ""

ocity.Text = ""

opincode.Text = ""

omobile.Text = ""

oemail.Text = ""

male.Value = False

female.Value = False

other.Value = False

Eye.Value = Unchecked

Tooth.Value = Unchecked

Lungs.Value = Unchecked

Liver.Value = Unchecked

Kidney.Value = Unchecked

skin.Value = Unchecked

home.Show

End Sub

Private Sub oage\_keypress(Ascii As Integer)

If Ascii = 13 And oage.Text <> "" Then

odob.SetFocus

ElseIf (Ascii < 48 And Ascii <> 8) Or Ascii > 57 Then

Ascii = 0

MsgBox "Enter Numbers Only", vbInformation

End If

End Sub

Private Sub ocity\_keypress(Ascii As Integer)

If Ascii = 13 And ocity.Text <> "" Then

opincode.SetFocus

ElseIf (Ascii < 65 And Ascii <> 8 And Ascii <> 32) Or (Ascii > 90 And Ascii < 97) Or (Ascii > 122) Then

Ascii = 0

MsgBox "Enter Letter Only", vbInformation

End If

End Sub

Private Sub omobile\_keypress(Ascii As Integer)

If Ascii = 13 And omobile.Text <> "" Then

oemail.SetFocus

ElseIf (Ascii < 48 And Ascii <> 8) Or Ascii > 57 Then

Ascii = 0

MsgBox "Enter Numbers Only", vbInformation

End If

End Sub

Private Sub oname\_keypress(Ascii As Integer)

If Ascii = 13 And oname.Text <> "" Then

oage.SetFocus

ElseIf (Ascii < 65 And Ascii <> 8 And Ascii <> 32) Or (Ascii > 90 And Ascii < 97) Or (Ascii > 122) Then

Ascii = 0

MsgBox "Enter Letter Only", vbInformation

End If

End Sub

Private Sub opincode\_keypress(Ascii As Integer)

If Ascii = 13 And opincode.Text <> "" Then

omobile.SetFocus

ElseIf (Ascii < 48 And Ascii <> 8) Or Ascii > 57 Then

Ascii = 0

MsgBox "Enter Numbers Only", vbInformation

End If

End Sub

Private Sub osubmit\_Click()

If oname.Text = "" Or oage.Text = "" Or odob.Text = "" Or oaddress.Text = "" Or ocity.Text = "" Or opincode.Text = "" Then

MsgBox "Fill all the Fields", vbInformation

ElseIf Eye.Value = Unchecked And Tooth.Value = Unchecked And Lungs.Value = Unchecked And Liver.Value = Unchecked And Kidney.Value = Unchecked And skin.Value = Unchecked Then

MsgBox "Select you choice", vbInformation

ElseIf oage.Text < 18 Then

MsgBox "Above 18+ only", vbInformation

ElseIf male.Value = False And female.Value = False And other.Value = False Then

MsgBox "Select the gender", vbInformation

ElseIf Len(opincode.Text) < 5 Then

MsgBox "Enter Valid AreaPincode", vbInformation

ElseIf Len(omobile.Text) < 10 Then

MsgBox "Enter Valid Mobile Number", vbInformation

ElseIf Eye.Value = Checked And Tooth.Value = Checked And Lungs.Value = Checked And Liver.Value = Checked And Kidney.Value = Checked And skin.Value = Checked Then

MsgBox "choose one donate option", vbInformation

Else

ODAdodc1.RecordSource = "select \* from organ\_donors"

ODAdodc1.Recordset.AddNew

ODAdodc1.Recordset(0) = oname.Text

ODAdodc1.Recordset(1) = oage.Text

ODAdodc1.Recordset(2) = odob.Text

If male.Value = True Then

ODAdodc1.Recordset(3) = male.Caption

ElseIf female.Value = True Then

ODAdodc1.Recordset(3) = female.Caption

ElseIf other.Value = True Then

End If

ODAdodc1.Recordset(3) = other.Caption

ODAdodc1.Recordset(4) = oaddress.Text

ODAdodc1.Recordset(5) = ocity.Text

ODAdodc1.Recordset(6) = opincode.Text

ODAdodc1.Recordset(7) = omobile.Text

ODAdodc1.Recordset(8) = oemail.Text

If Eye.Value = Checked Then

ODAdodc1.Recordset(9) = Eye.Caption

ElseIf Tooth.Value = Checked Then

ODAdodc1.Recordset(9) = Tooth.Caption

ElseIf Lungs.Value = Checked Then

ODAdodc1.Recordset(9) = Lungs.Caption

ElseIf Liver.Value = Checked Then

ODAdodc1.Recordset(9) = Lungs.Caption

ElseIf Kidney.Value = Checked Then

ODAdodc1.Recordset(9) = Kidney.Caption

ElseIf skin.Value = Checked Then

ODAdodc1.Recordset(9) = skin.Caption

End If

ODAdodc1.Recordset.Update

MsgBox "Donor Details Successfully Registered", vbInformation

oname.Text = ""

oage.Text = ""

odob.Text = ""

oaddress.Text = ""

ocity.Text = ""

opincode.Text = ""

omobile.Text = ""

oemail.Text = ""

male.Value = False

female.Value = False

other.Value = False

Eye.Value = Unchecked

Tooth.Value = Unchecked

Lungs.Value = Unchecked

Liver.Value = Unchecked

Kidney.Value = Unchecked

skin.Value = Unchecked

home.Show

End If

End Sub

**Blood Donor Registeration**

Dim cn As String

Dim con As New ADODB.Connection

Private Sub bcancel\_Click()

bname.Text = ""

bage.Text = ""

bdob.Text = ""

bweight.Text = ""

baddress.Text = ""

bcity.Text = ""

bpincode.Text = ""

bmobile.Text = ""

bemail.Text = ""

Option1.Value = False

Option2.Value = False

Option3.Value = False

Option4.Value = False

Option5.Value = False

Option6.Value = False

Option7.Value = False

Option8.Value = False

home.Show

End Sub

Private Sub bsubmit\_Click()

If bname.Text = "" Or bage.Text = "" Or bdob.Text = "" Or bweight.Text = "" Or baddress.Text = "" Or bcity.Text = "" Or bpincode.Text = "" Or bmobile.Text = "" Then

MsgBox "Fill all the Fields", vbInformation

ElseIf Option1.Value = False And Option2.Value = False And Option3.Value = False And Option4.Value = False And Option5.Value = False And Option6.Value = False And Option7.Value = False And Option8.Value = False Then

MsgBox "Select the Blood Group", vbInformation

ElseIf bage.Text < 18 Then

MsgBox "Above 18+ only", vbInformation

ElseIf bweight.Text < 50 Then

MsgBox "Weight lessthen 50lbs not elegible", vbInformation

ElseIf Len(bpincode.Text) < 5 Then

MsgBox "Enter Valid AreaPincode", vbInformation

ElseIf Len(bmobile.Text) < 10 Then

MsgBox "Enter Valid Mobile Number", vbInformation

Else

BDAdodc1.RecordSource = "select \* from blood\_donors"

BDAdodc1.Recordset.AddNew

BDAdodc1.Recordset(0) = bname.Text

BDAdodc1.Recordset(1) = bage.Text

BDAdodc1.Recordset(2) = bdob.Text

If Option1.Value = True Then

BDAdodc1.Recordset(3) = Option1.Caption

ElseIf Option2.Value = True Then

BDAdodc1.Recordset(3) = Option2.Caption

ElseIf Option3.Value = True Then

BDAdodc1.Recordset(3) = Option3.Caption

ElseIf Option4.Value = True Then

BDAdodc1.Recordset(3) = Option4.Caption

ElseIf Option5.Value = True Then

BDAdodc1.Recordset(3) = Option5.Caption

ElseIf Option6.Value = True Then

BDAdodc1.Recordset(3) = Option6.Caption

ElseIf Option7.Value = True Then

BDAdodc1.Recordset(3) = Option7.Caption

ElseIf Option8.Value = True Then

BDAdodc1.Recordset(3) = Option8.Caption

End If

BDAdodc1.Recordset(4) = bweight.Text

BDAdodc1.Recordset(5) = baddress.Text

BDAdodc1.Recordset(6) = bcity.Text

BDAdodc1.Recordset(7) = bpincode.Text

BDAdodc1.Recordset(8) = bmobile.Text

BDAdodc1.Recordset(9) = bemail.Text

BDAdodc1.Recordset.Update

MsgBox "Donor Details Successfully Registered", vbInformation

bname.Text = ""

bage.Text = ""

bdob.Text = ""

bweight.Text = ""

baddress.Text = ""

bcity.Text = ""

bpincode.Text = ""

bmobile.Text = ""

bemail.Text = ""

Option1.Value = False

Option2.Value = False

Option3.Value = False

Option4.Value = False

Option5.Value = False

Option6.Value = False

Option7.Value = False

Option8.Value = False

home.Show

End If

End Sub

Private Sub Form\_Load()

cn = "Provider=MSDAORA.1;Password=open;User ID=scott;Persist Security Info=True"

End Sub

Private Sub bname\_keypress(Ascii As Integer)

If Ascii = 13 And bname.Text <> "" Then

bage.SetFocus

ElseIf (Ascii < 65 And Ascii <> 8 And Ascii <> 32) Or (Ascii > 90 And Ascii < 97) Or (Ascii > 122) Then

Ascii = 0

MsgBox "Enter Letter Only", vbInformation

End If

End Sub

Private Sub bage\_keypress(Ascii As Integer)

If Ascii = 13 And bage.Text <> "" Then

bdob.SetFocus

ElseIf (Ascii < 48 And Ascii <> 8) Or Ascii > 57 Then

Ascii = 0

MsgBox "Enter Numbers Only", vbInformation

End If

End Sub

Private Sub bweight\_keypress(Ascii As Integer)

If Ascii = 13 And bweight.Text <> "" Then

baddress.SetFocus

ElseIf (Ascii < 48 And Ascii <> 8) Or Ascii > 57 Then

Ascii = 0

MsgBox "Enter Numbers Only", vbInformation

End If

End Sub

Private Sub bcity\_keypress(Ascii As Integer)

If Ascii = 13 And bcity.Text <> "" Then

bpincode.SetFocus

ElseIf (Ascii < 65 And Ascii <> 8 And Ascii <> 32) Or (Ascii > 90 And Ascii < 97) Or (Ascii > 122) Then

Ascii = 0

MsgBox "Enter Letter Only", vbInformation

End If

End Sub

Private Sub bpincode\_keypress(Ascii As Integer)

If Ascii = 13 And bpincode.Text <> "" Then

bmobile.SetFocus

ElseIf (Ascii < 48 And Ascii <> 8) Or Ascii > 57 Then

Ascii = 0

MsgBox "Enter Numbers Only", vbInformation

End If

End Sub

Private Sub bmobile\_keypress(Ascii As Integer)

If Ascii = 13 And bmobile.Text <> "" Then

bemail.SetFocus

ElseIf (Ascii < 48 And Ascii <> 8) Or Ascii > 57 Then

Ascii = 0

MsgBox "Enter Numbers Only", vbInformation

End If

End Sub

**Request Registration**

Dim cn As String

Dim con As New ADODB.Connection

Private Sub Form\_Load()

cn = "Provider=MSDAORA.1;Password=open;User ID=scott;Persist Security Info=True"

End Sub

Private Sub rcity\_keypress(Ascii As Integer)

If Ascii = 13 And rcity.Text <> "" Then

rpincode.SetFocus

ElseIf (Ascii < 65 And Ascii <> 8 And Ascii <> 32) Or (Ascii > 90 And Ascii < 97) Or (Ascii > 122) Then

Ascii = 0

MsgBox "Enter Letter Only", vbInformation

End If

End Sub

Private Sub rmobile\_keypress(Ascii As Integer)

If Ascii = 13 And rmobile.Text <> "" Then

remail.SetFocus

ElseIf (Ascii < 48 And Ascii <> 8) Or Ascii > 57 Then

Ascii = 0

MsgBox "Enter Numbers Only", vbInformation

End If

End Sub

Private Sub rpincode\_keypress(Ascii As Integer)

If Ascii = 13 And rpincode.Text <> "" Then

rmobile.SetFocus

ElseIf (Ascii < 48 And Ascii <> 8) Or Ascii > 57 Then

Ascii = 0

MsgBox "Enter Numbers Only", vbInformation

End If

End Sub

Private Sub rname\_keypress(Ascii As Integer)

If Ascii = 13 And rname.Text <> "" Then

raddress.SetFocus

ElseIf (Ascii < 65 And Ascii <> 8 And Ascii <> 32) Or (Ascii > 90 And Ascii < 97) Or (Ascii > 122) Then

Ascii = 0

MsgBox "Enter Letter Only", vbInformation

End If

End Sub

Private Sub rorgan\_Click()

eye.Enabled = True

tooth.Enabled = True

lungs.Enabled = True

liver.Enabled = True

kidney.Enabled = True

skin.Enabled = True

Option1.Enabled = False

Option2.Enabled = False

Option3.Enabled = False

Option4.Enabled = False

Option5.Enabled = False

Option6.Enabled = False

Option7.Enabled = False

Option8.Enabled = False

Option1.Value = False

Option2.Value = False

Option3.Value = False

Option4.Value = False

Option5.Value = False

Option6.Value = False

Option7.Value = False

Option8.Value = False

End Sub

Private Sub rblood\_Click()

eye.Enabled = False

tooth.Enabled = False

lungs.Enabled = False

liver.Enabled = False

kidney.Enabled = False

skin.Enabled = False

Option1.Enabled = True

Option2.Enabled = True

Option3.Enabled = True

Option4.Enabled = True

Option5.Enabled = True

Option6.Enabled = True

Option7.Enabled = True

Option8.Enabled = True

eye.Value = False

tooth.Value = False

lungs.Value = False

liver.Value = False

kidney.Value = False

skin.Value = False

End Sub

Private Sub rcancel\_Click()

rname.Text = ""

raddress.Text = ""

rcity.Text = ""

rpincode = ""

rmobile.Text = ""

remail.Text = ""

Option1.Enabled = False

Option2.Enabled = False

Option3.Enabled = False

Option4.Enabled = False

Option5.Enabled = False

Option6.Enabled = False

Option7.Enabled = False

Option8.Enabled = False

Option1.Value = False

Option2.Value = False

Option3.Value = False

Option4.Value = False

Option5.Value = False

Option6.Value = False

Option7.Value = False

Option8.Value = False

eye.Enabled = False

tooth.Enabled = False

lungs.Enabled = False

liver.Enabled = False

kidney.Enabled = False

skin.Enabled = False

eye.Value = False

tooth.Value = False

lungs.Value = False

liver.Value = False

kidney.Value = False

skin.Value = False

home.Show

End Sub

Private Sub rsubmit\_Click()

If rname.Text = "" Or raddress.Text = "" Or rcity.Text = "" Or rpincode = "" Or rmobile.Text = "" Then

MsgBox "Fill all the Fields", vbInformation

ElseIf eye.Enabled = False And tooth.Enabled = False And lungs.Enabled = False And liver.Enabled = False And kidney.Enabled = False And skin.Enabled = False And Option1.Enabled = False And Option2.Enabled = False And Option3.Enabled = False And Option4.Enabled = False And Option5.Enabled = False And Option6.Enabled = False And Option7.Enabled = False And Option8.Enabled = False Then

MsgBox "Select an option", vbInformation

ElseIf eye.Value = False And tooth.Value = False And lungs.Value = False And liver.Value = False And kidney.Value = False And skin.Value = False And Option1.Value = False And Option2.Value = False And Option3.Value = False And Option4.Value = False And Option5.Value = False And Option6.Value = False And Option7.Value = False And Option8.Value = False Then

MsgBox "Select an option", vbInformation

ElseIf Len(rpincode.Text) < 5 Then

MsgBox "Enter Valid AreaPincode", vbInformation

ElseIf Len(rmobile.Text) < 10 Then

MsgBox "Enter Valid Mobile Number", vbInformation

Else

RDAdodc1.RecordSource = "select \* from request\_details"

RDAdodc1.Recordset.AddNew

RDAdodc1.Recordset(0) = rname.Text

RDAdodc1.Recordset(1) = raddress.Text

RDAdodc1.Recordset(2) = rcity.Text

RDAdodc1.Recordset(3) = rpincode.Text

RDAdodc1.Recordset(4) = rmobile.Text

RDAdodc1.Recordset(5) = remail.Text

If eye.Value = True Then

RDAdodc1.Recordset(6) = eye.Caption

ElseIf tooth.Value = True Then

RDAdodc1.Recordset(6) = tooth.Caption

ElseIf lungs.Value = True Then

RDAdodc1.Recordset(6) = lungs.Caption

ElseIf liver.Value = True Then

RDAdodc1.Recordset(6) = liver.Caption

ElseIf kidney.Value = True Then

RDAdodc1.Recordset(6) = kidney.Caption

ElseIf skin.Value = True Then

RDAdodc1.Recordset(6) = skin.Caption

ElseIf Option1.Value = True Then

RDAdodc1.Recordset(7) = Option1.Caption

ElseIf Option2.Value = True Then

RDAdodc1.Recordset(7) = Option2.Caption

ElseIf Option3.Value = True Then

RDAdodc1.Recordset(7) = Option3.Caption

ElseIf Option4.Value = True Then

RDAdodc1.Recordset(7) = Option4.Caption

ElseIf Option5.Value = True Then

RDAdodc1.Recordset(7) = Option5.Caption

ElseIf Option6.Value = True Then

RDAdodc1.Recordset(7) = Option6.Caption

ElseIf Option7.Value = True Then

RDAdodc1.Recordset(7) = Option7.Caption

ElseIf Option8.Value = True Then

RDAdodc1.Recordset(7) = Option8.Caption

End If

RDAdodc1.Recordset.Update

MsgBox "Request Successfully Registered", vbInformation

rname.Text = ""

raddress.Text = ""

rcity.Text = ""

rpincode = ""

rmobile.Text = ""

remail.Text = ""

Option1.Enabled = False

Option2.Enabled = False

Option3.Enabled = False

Option4.Enabled = False

Option5.Enabled = False

Option6.Enabled = False

Option7.Enabled = False

Option8.Enabled = False

Option1.Value = False

Option2.Value = False

Option3.Value = False

Option4.Value = False

Option5.Value = False

Option6.Value = False

Option7.Value = False

Option8.Value = False

eye.Enabled = False

tooth.Enabled = False

lungs.Enabled = False

liver.Enabled = False

kidney.Enabled = False

skin.Enabled = False

eye.Value = False

tooth.Value = False

lungs.Value = False

liver.Value = False

kidney.Value = False

skin.Value = False

home.Show

End If

End Sub

**Chapter 5**

**Testing**

**5.1 Introduction to Testing**

Testing goes side by side with the implementation that is aimed at ensuring that the system works accurately and efficiently before the live operation is performed .The common view of testing held by the user is process of executing a program with explicit intention of handling errors. The application which has been developed has to be tested to prove its validity. Testing is considered to be the least creative phase of the whole cycle of system design. In the real sense it is the phase, which helps to bring out the creativity of the other phases, and makes it shine. The Organ and Blood Donation Management System was tested using the following two techniques of application testing.

**5.1.1 White Box Testing**

a. By using this technique it was tested that all the individual logical paths were executed at least once and every statement in the program was executed once during testing

b. All the logical decisions where tested on both their true and false sides

c. All the loops were tested with data in between the range and especially at the boundary values.

**5.1.2Black Box Testing**

a. By the use of this technique the missing functions were identified and placed in their position.

b. The errors in the interfaces were identified and corrected.

c. The errors in the database access were indentified.

d. This technique was along used to identify the initialization and termination errors and correct them.

**5.1.3 Application Testing Strategies**

Any application has to be tested with pre-planned strategies. As Roger Pressman states, the preparation for testing should start as soon as the design of the system start. To carry out the testing in an efficient manner certain amount of strategic planning has to be done. Any testing strategy must incorporate test planning, test case design, test execution and the resultant data collection evaluation.

Organ and Blood Donation Management System was tested with the help of the following testing strategies:

* + 1. **Unit Testing**

a. In the line of strategy the entire individuals function and modules were put to test independently

b.By following this strategy all the errors in coding were identified and corrected.

c.This method was applied in combination with the White Box and Black Box testing

d. Technique to find errors in each module.

e.The effort of specific combination of data on system operation was tested.

f.The following were the testes carried out for Graphical User Interface (GUI).

g.It was seen that the pages opens properly based on related menu based commands.

h.It was tested whether all relevant buttons, icons and other controls are available and properly displayed.

i.Are the names of the menu functions self explanatory?

j.Are data input messages intelligible?

**5.1.5Integration Testing**

This application testing strategy has two different approached namely the top down approach in which the integration is carried out from the top level module to the bottom and the bottom up approach in which the integration is carried out from the low level modules to the top.

The modules were tested using the bottom up approach by introducing stubs for the top –level functions .

This test was used to identify the errors in the interfaces, the error in passing the parameters between the function and to correct them.

**5.1.6 Validation Testing**

The main aim of this testing is to verify that the application system does what it was designed for. The system was tested to ensure that the purpose of automating the Organ and Blood Donation Management was fulfilled.

Alpha testing was carried out to ensure the validity of the system

**5.1.7 System Testing**

The idea of applying this testing strategy was to put the system to a series of tests to ensure that it performed well and exactly same under all condition.

The tests that the system was put were:

Recovery Testing.

Performance Testing.

The tests were made on each form for its correctness of accepting the data and storing the data into the respective tables in the desired form of data type. The algorithms were tested with the test data first and then with the real data. The tests were made for all types of constraints. The tests were done in presence of the user so that he/she is familiar with the system that is going to be introduced. During the testing each objective of the system was tested and found to be working correctly.

**5.2 MULTI-USER SYSTEM**

Database Locking Schemes: Whenever more than one person is accessing a record/s some type of process must be use to prevent the outer users from attempting to update the same record at the same time. This process is a locking scheme. In its simplest form, a locking scheme allows only one user at a time to update information in the database.

**5.3 TESTING RESULTS**

In this application we have 2 modules

* **Admin**
* **User**

The data are checked as per the data type specified in database.

For example

If there is a need to enter PhNo. user can’t enter the characters, he can enter only numbers up to 10 digits.

If you want to enter the User name of a Person, you can’t enter the number as it requires accepting the character data only as per the hard coding.

In Login form there are 2 roles specified.

**Admin** is the user who can access all the Modules in the application ,Except the Donor Registration

**User** is who can access only User Home , Register Donors, Register Requests.

**5.3.1 Setup Module Testing**

In this module all the primary keys are auto generated in all the sub modules,

The design, caption of the form, caption of the label and the alignment of the Text boxes, Combo Boxes are taken care. The initial values for the form is also consider in the form load events.

The Phone no., E-mail are tested well, if the e-mail id doesn’t have @ and . (domain) will not be considered.

Before saving the data into the database, the mandatory fields and the maximum length of the fields has been taken in to considered and test well.

The message is displayed for the user conformation while saving and deleting the Record from the database.

In case of search if the record is not found, A message is displayed so that the user can analysis the data present or not in the database.

While closing for the form done the record set will be disconnected from the database.

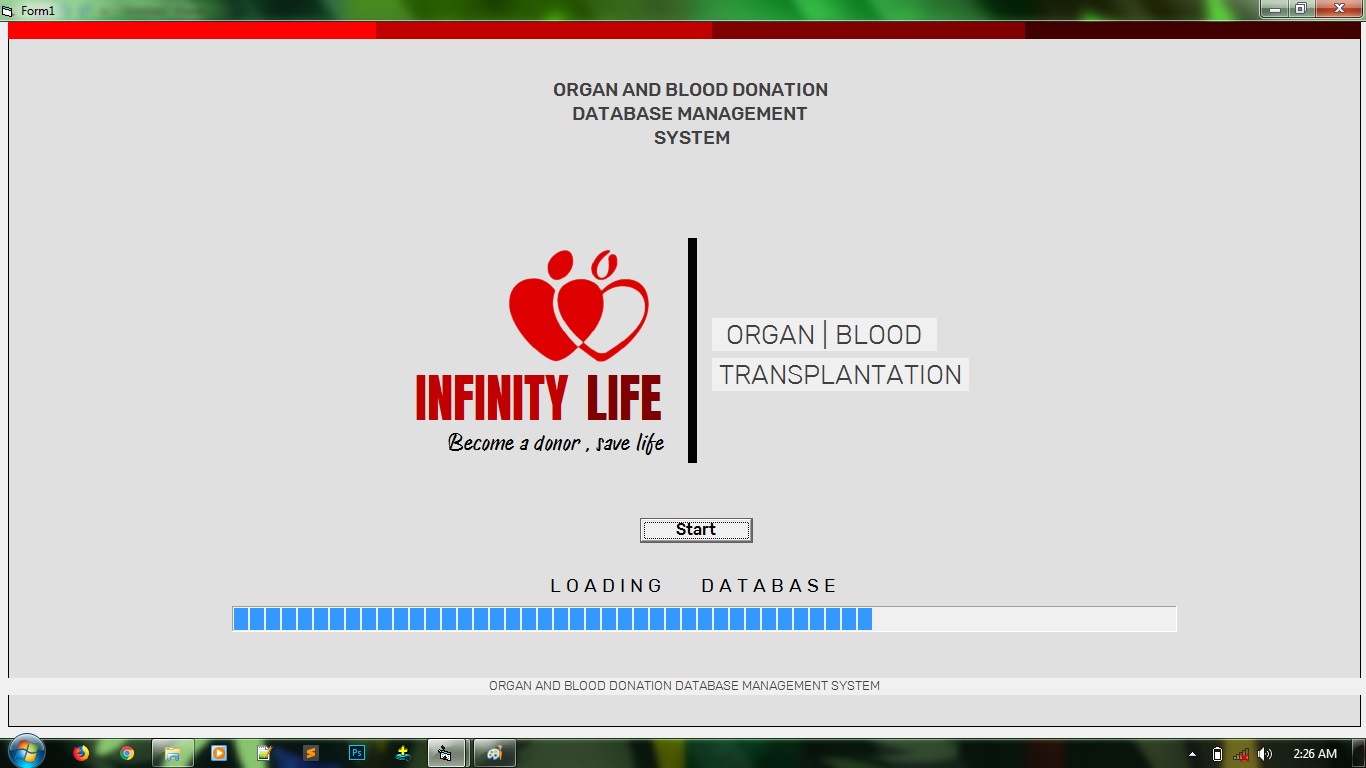
The Clear Command button clears the form and set the curser to the default command button or the textbox.

**Chapter 6**

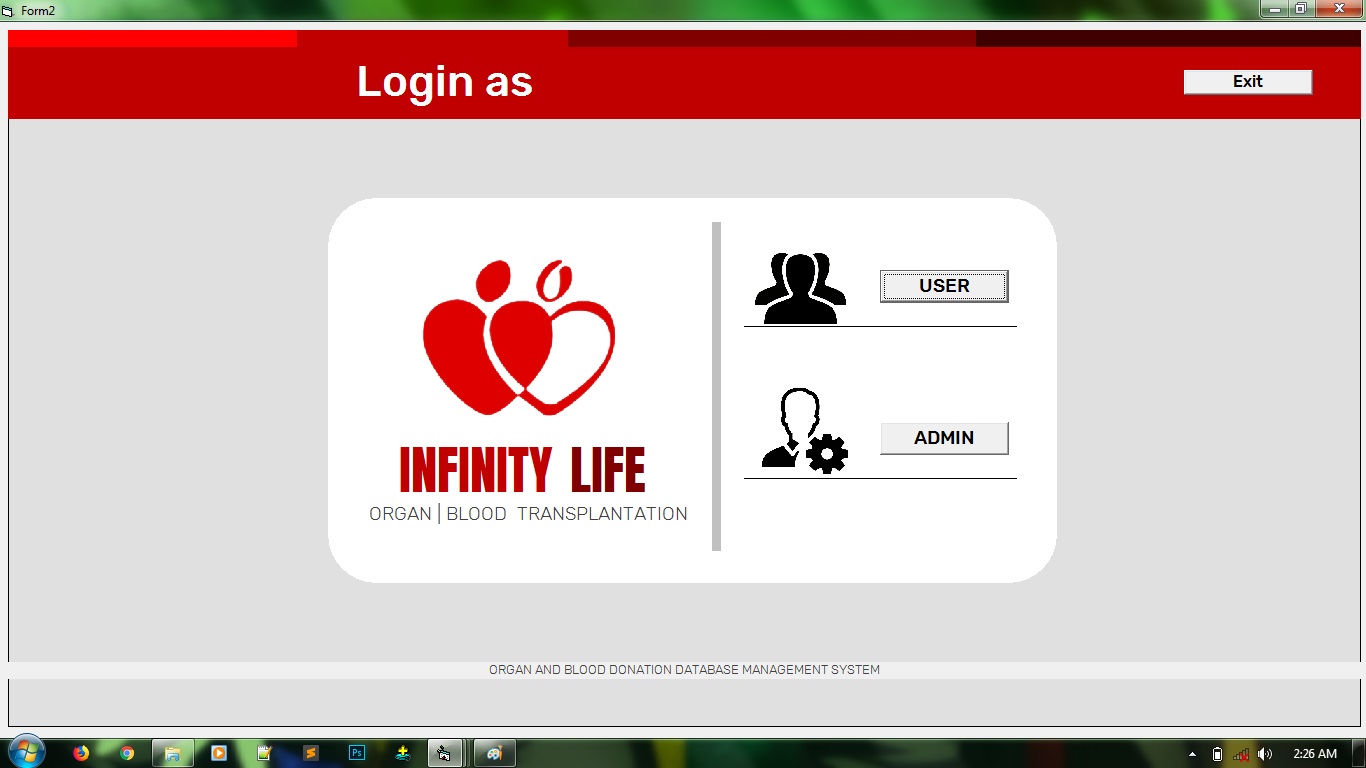
**User Manual**

**6.1 Screen Shorts**

**Welcome Page**

****

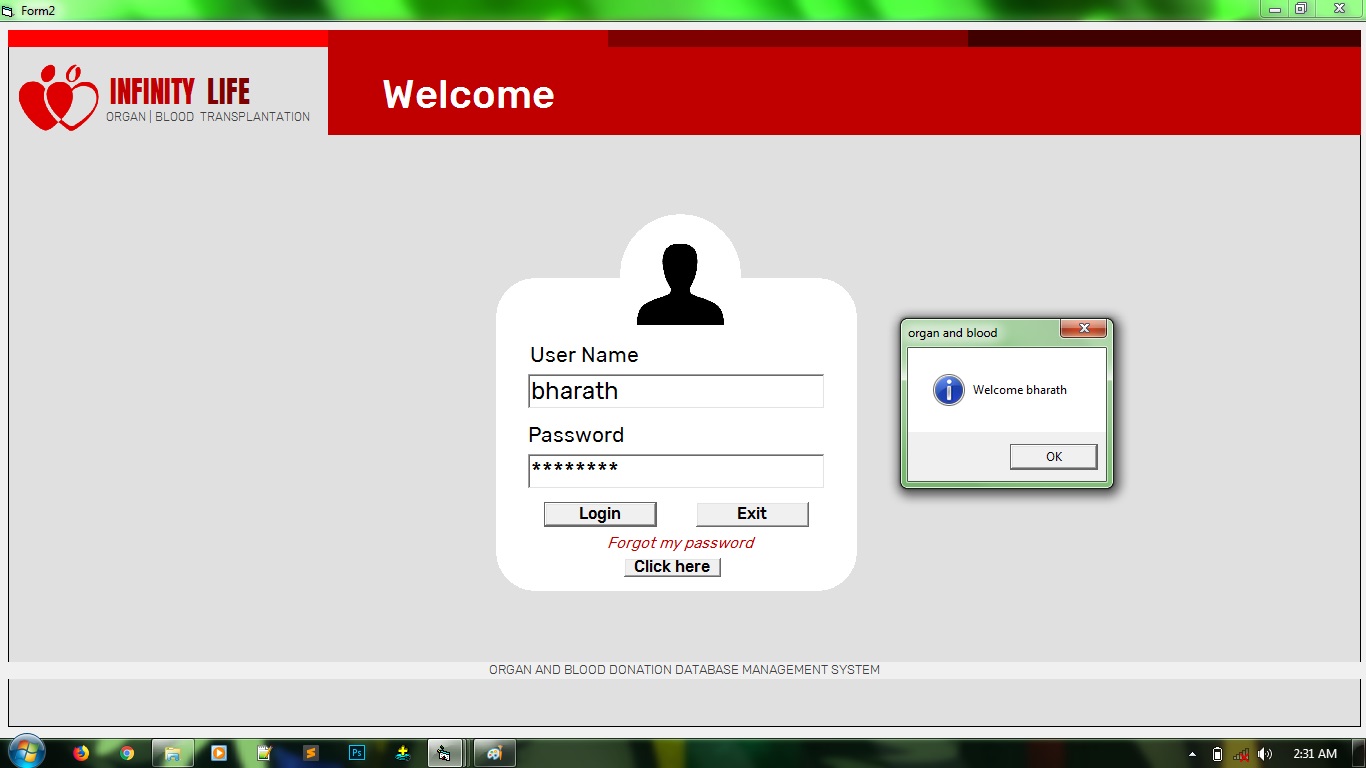
**Login Page**

****

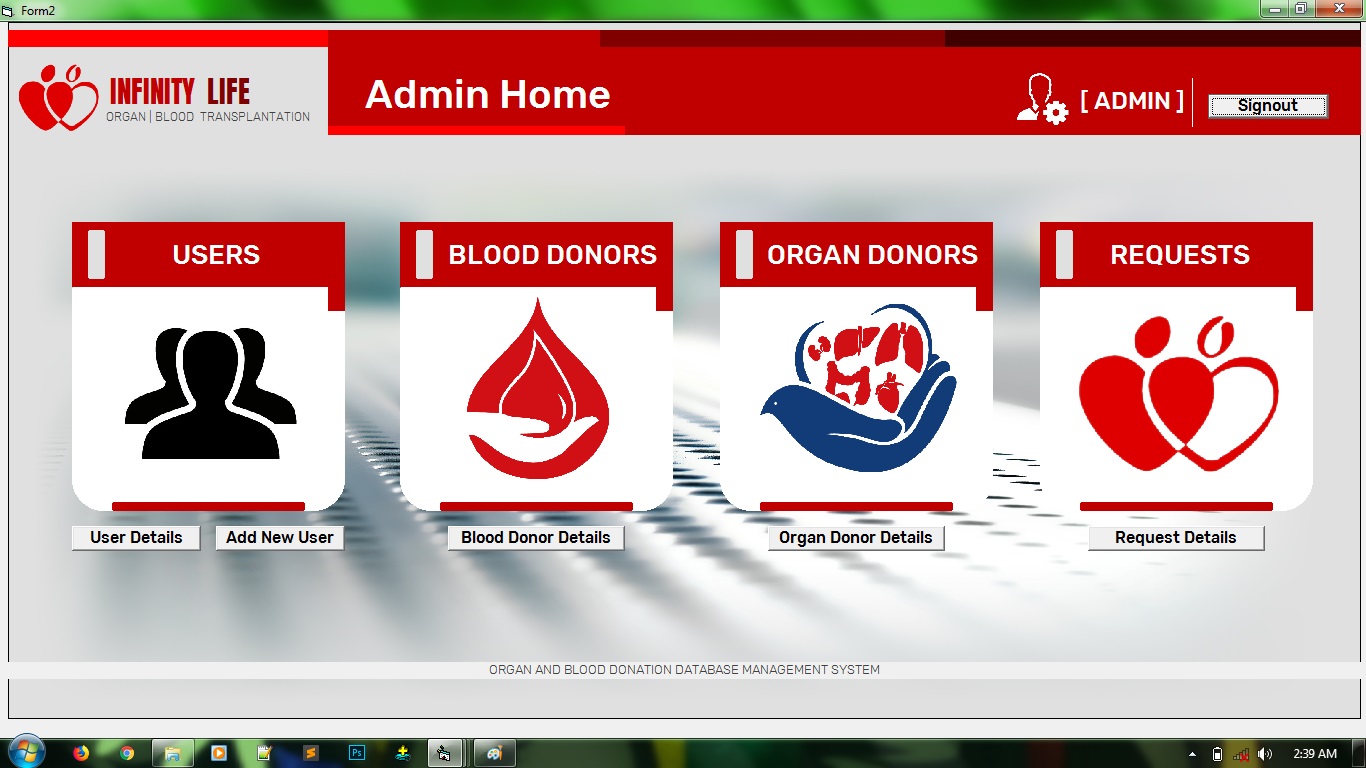
**Admin Login:-** In this form admin can login

****

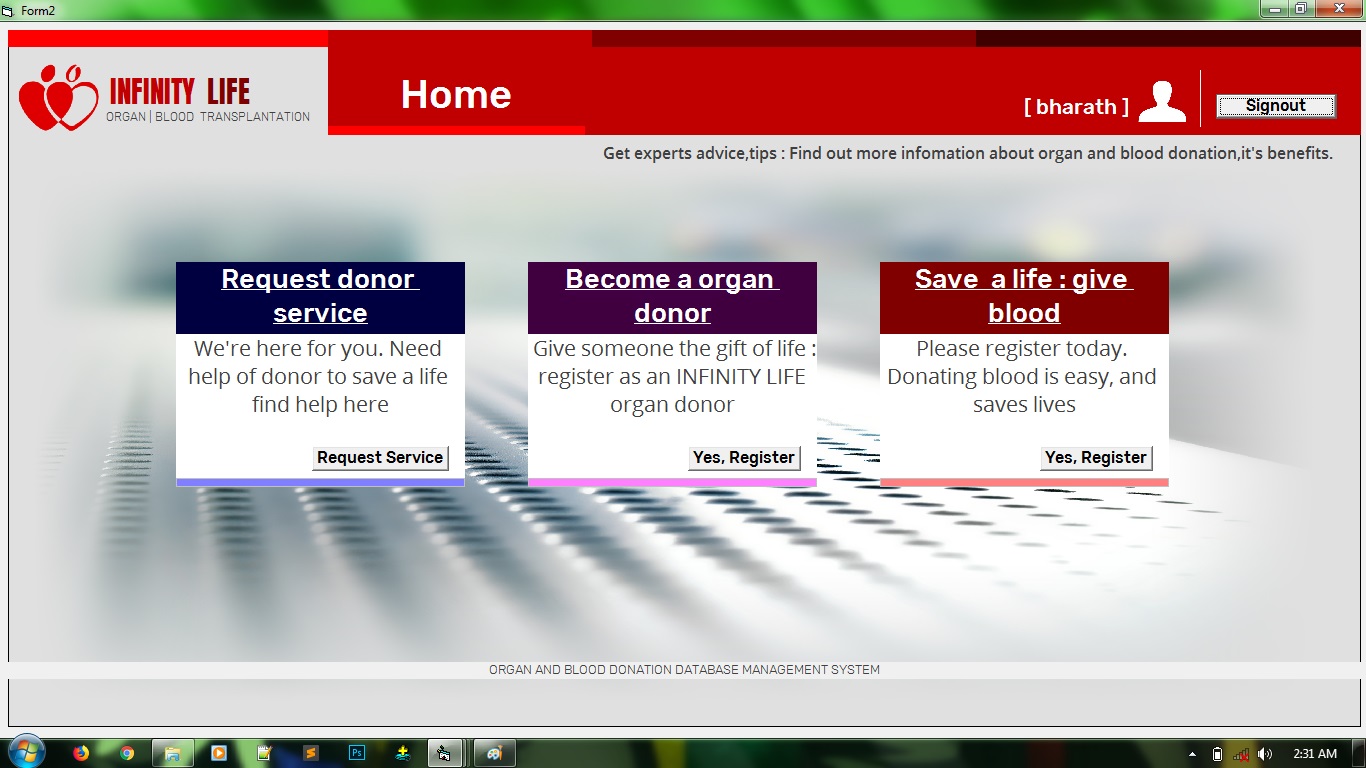
**User Login:-** In this form user login



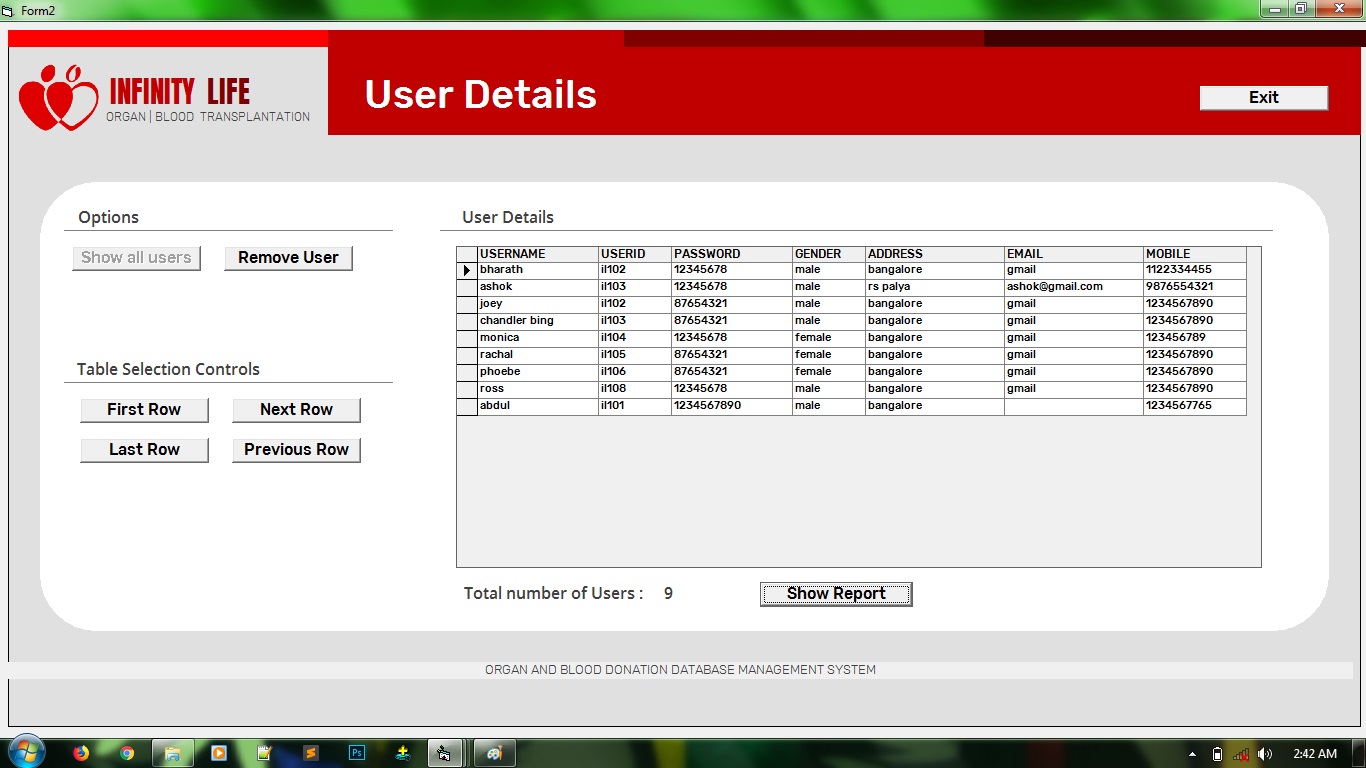
**Admin Functionalities:-** This Page contains the functionalities of admin



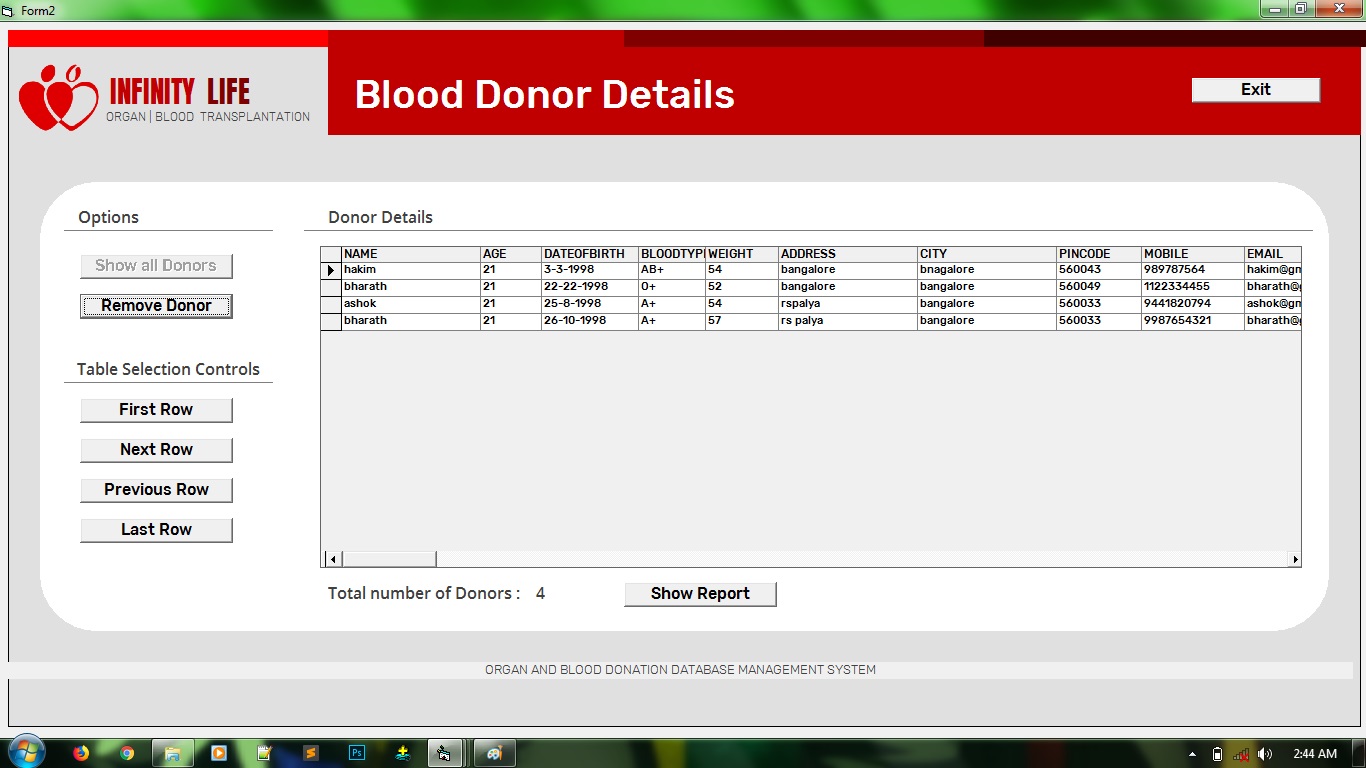
**User Functionalities:-** This Page contains the functionalities of user



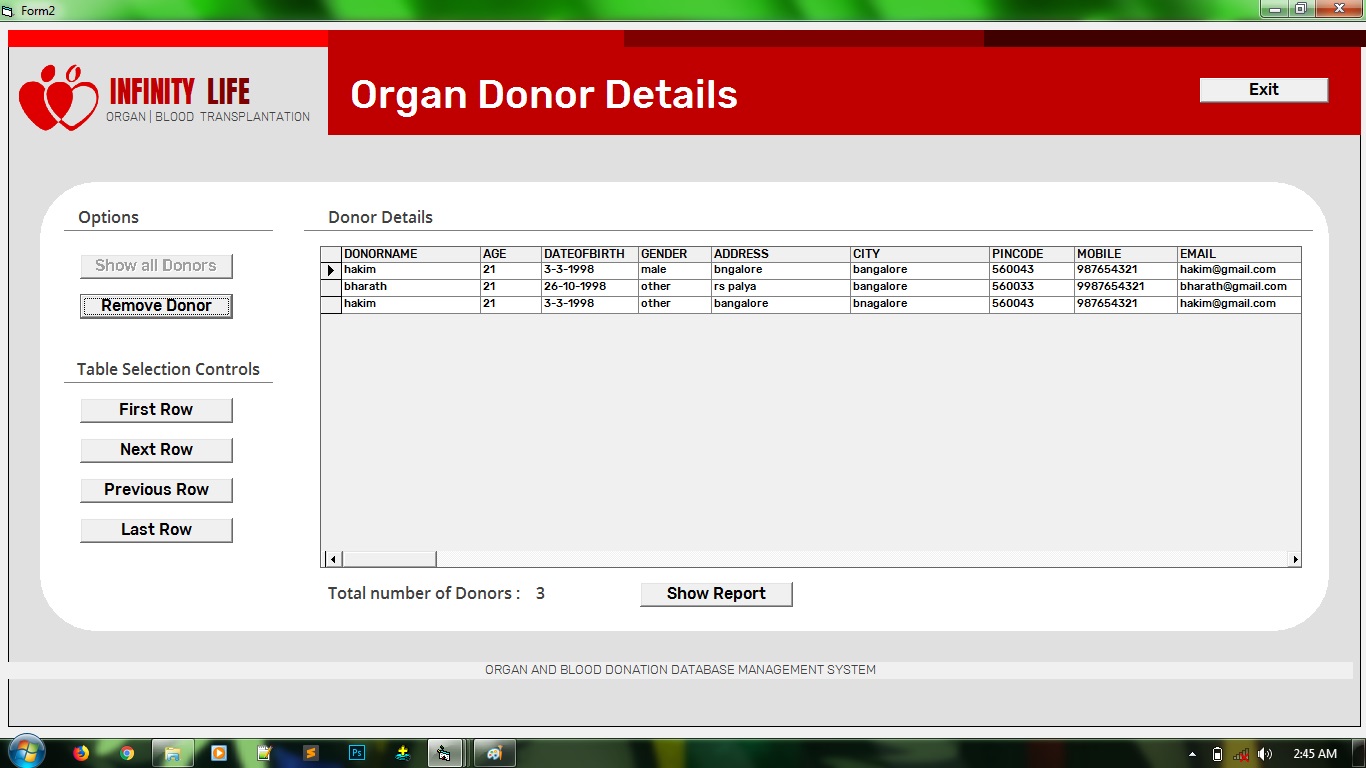
**User Details:-** This page contains all the details of the employees , only admin can add, delete



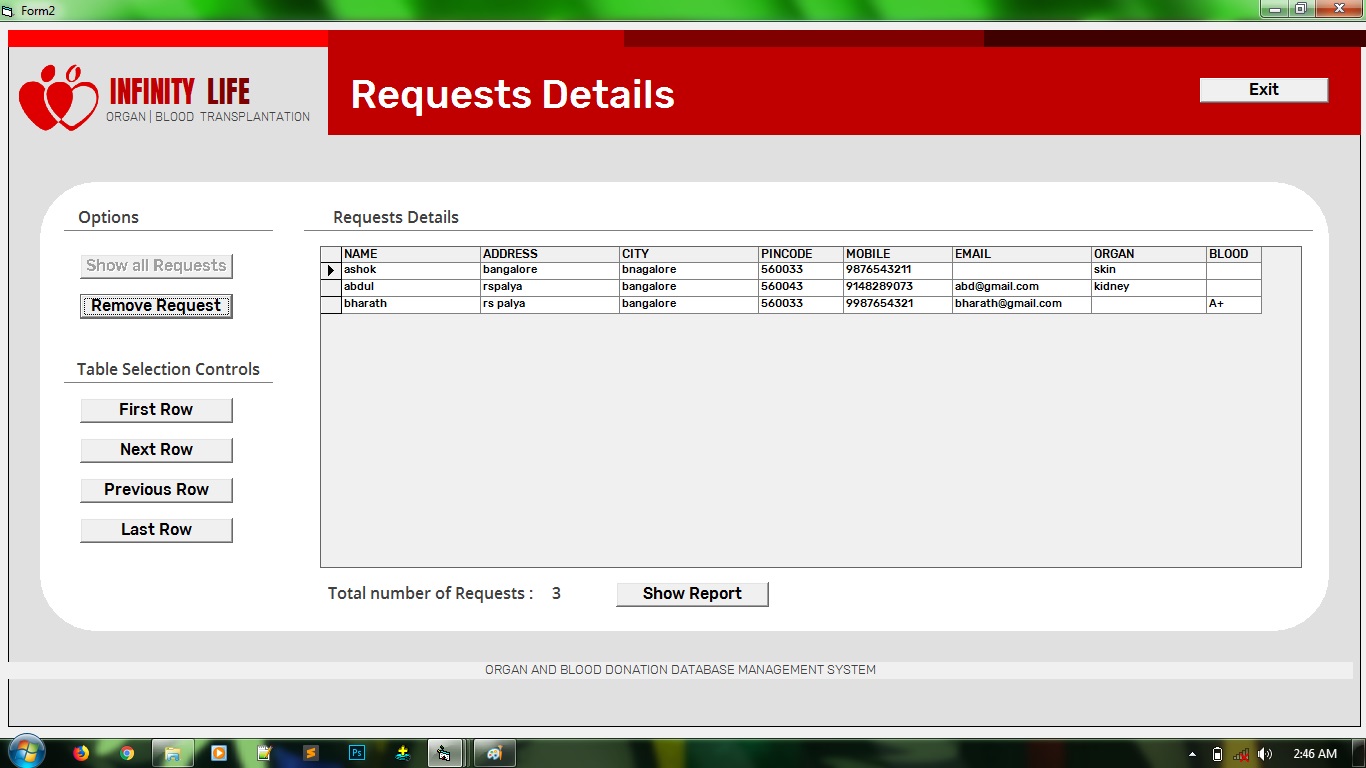
**Blood Donor Details:-** This page contains all the details of the Donors ,the admin can delete

****

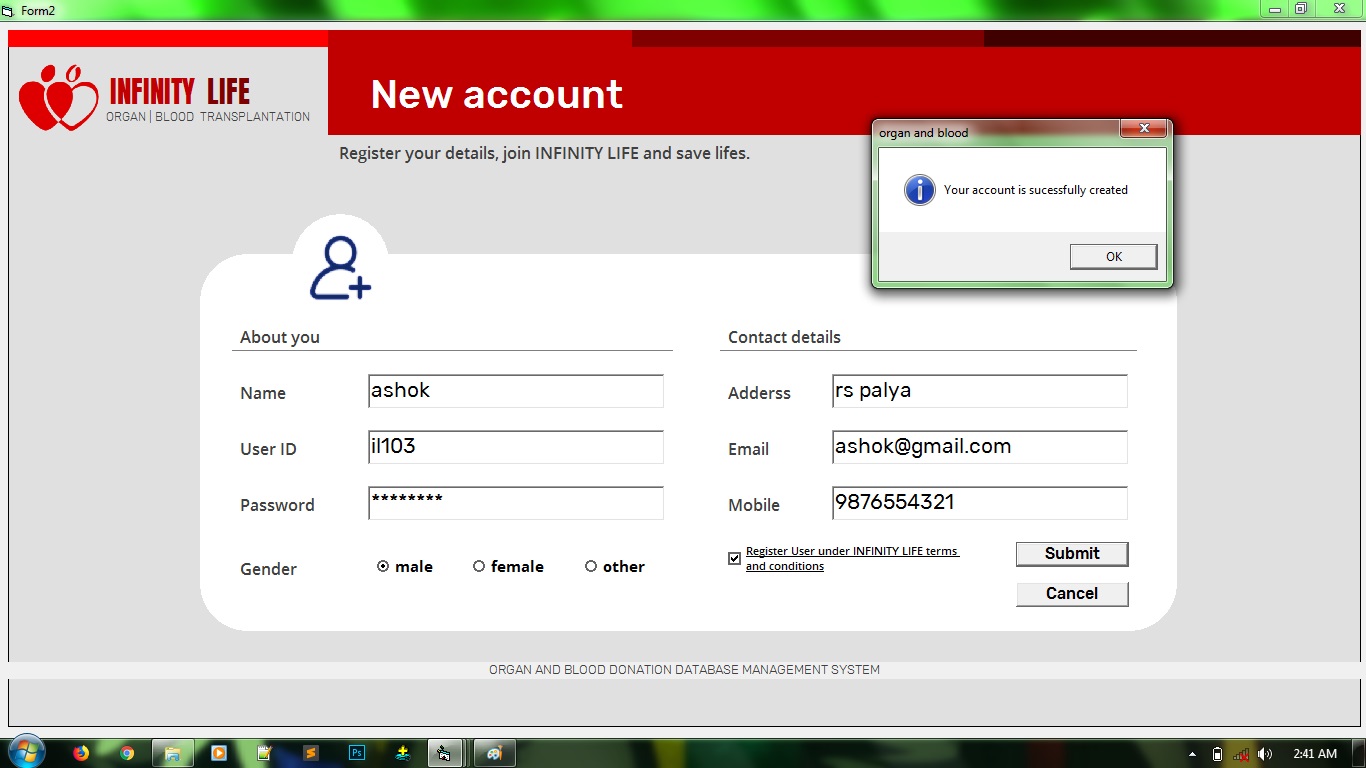
**Organ Donor Details:-**  This page contains all the details of the Donors ,the admin can delete



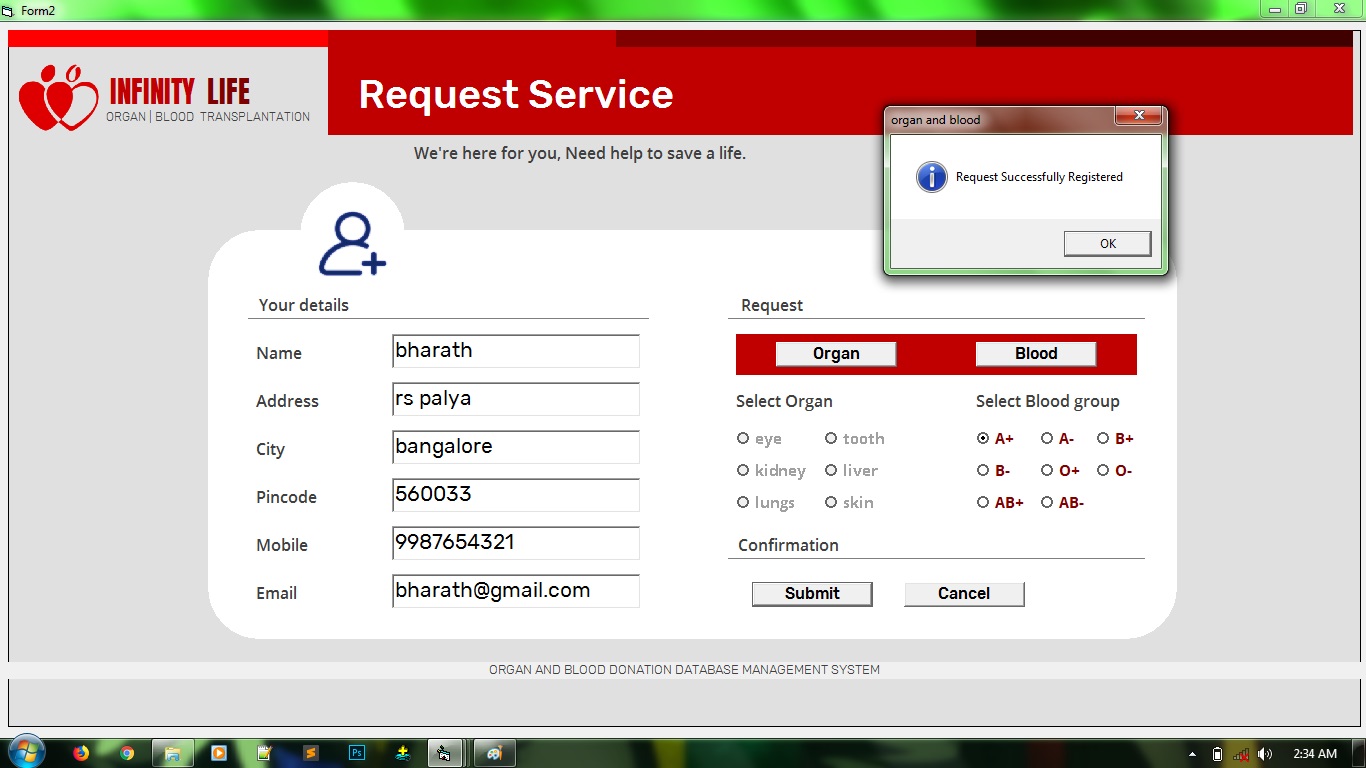
**Request Details:-** This page contains all the details of the Request ,the admin can delete



**Adding New User:-** In this page only admin can add new user to the existing list of users



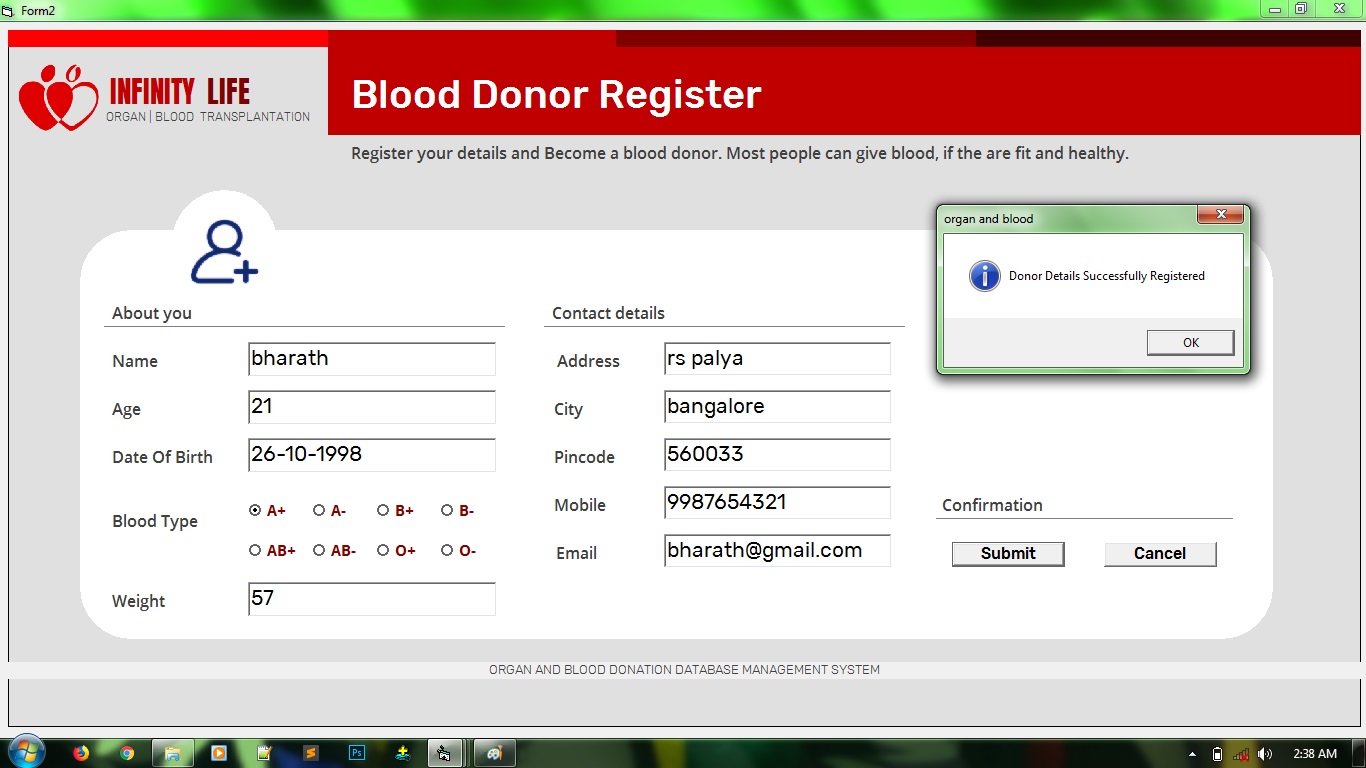
**Request Services :-** In this page the user can update the request for services



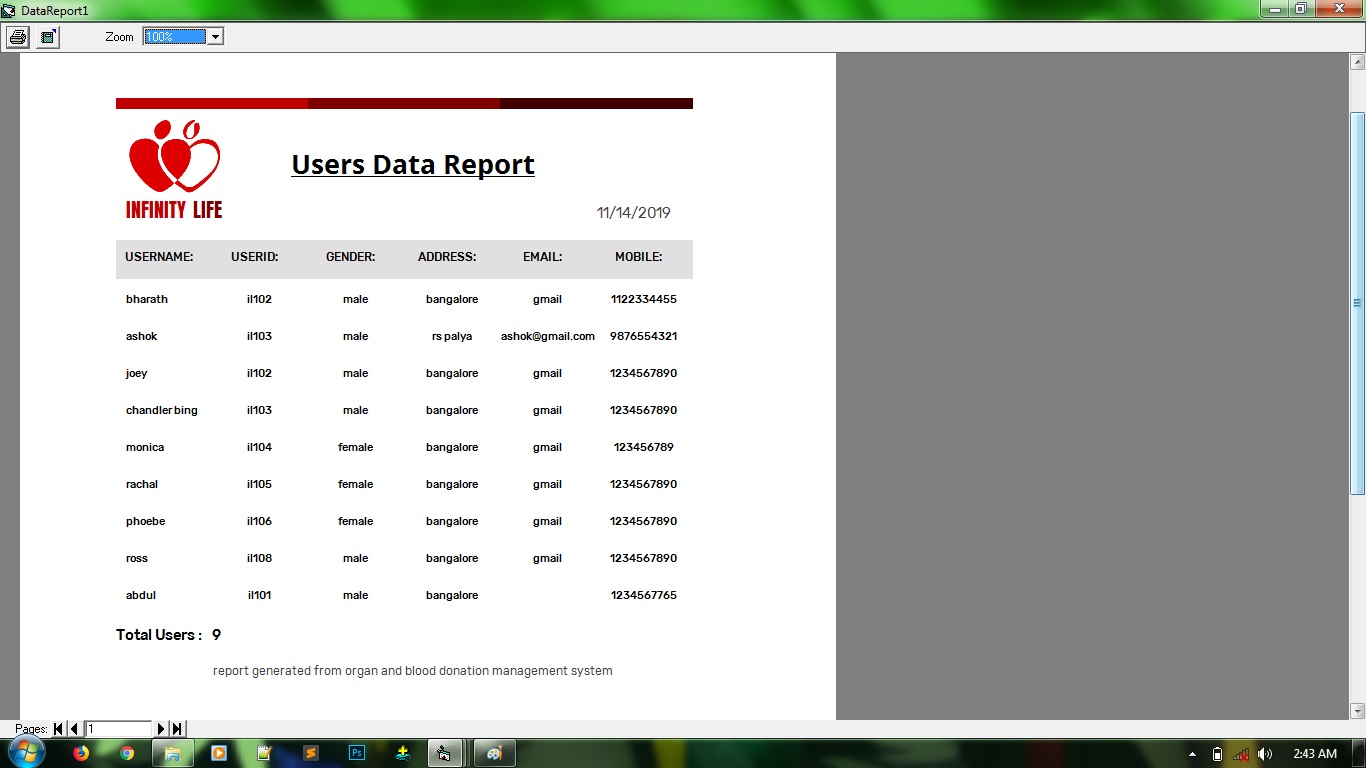
**Register Organ Donor :-** In this page the user can update the organ donor details



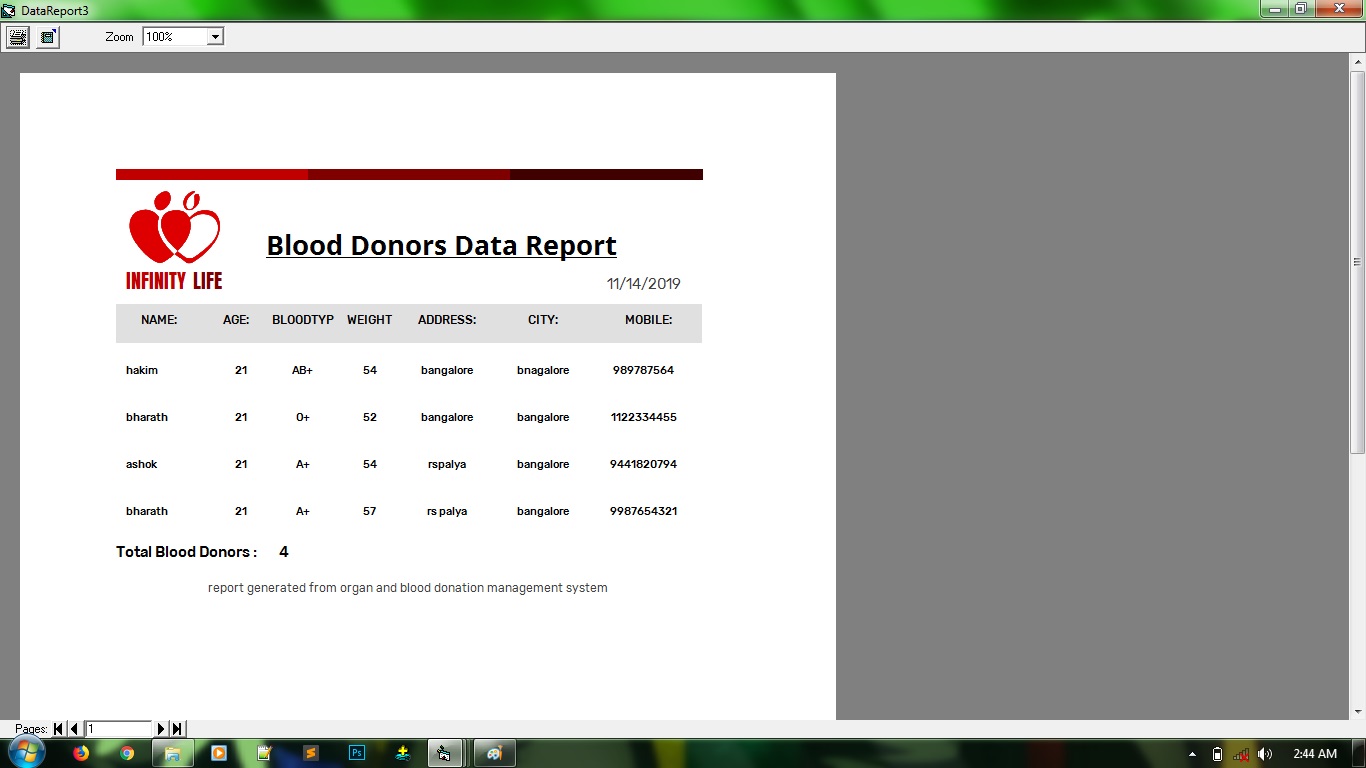
**Register Blood Donor :-** In this page the user can update the blood donor details



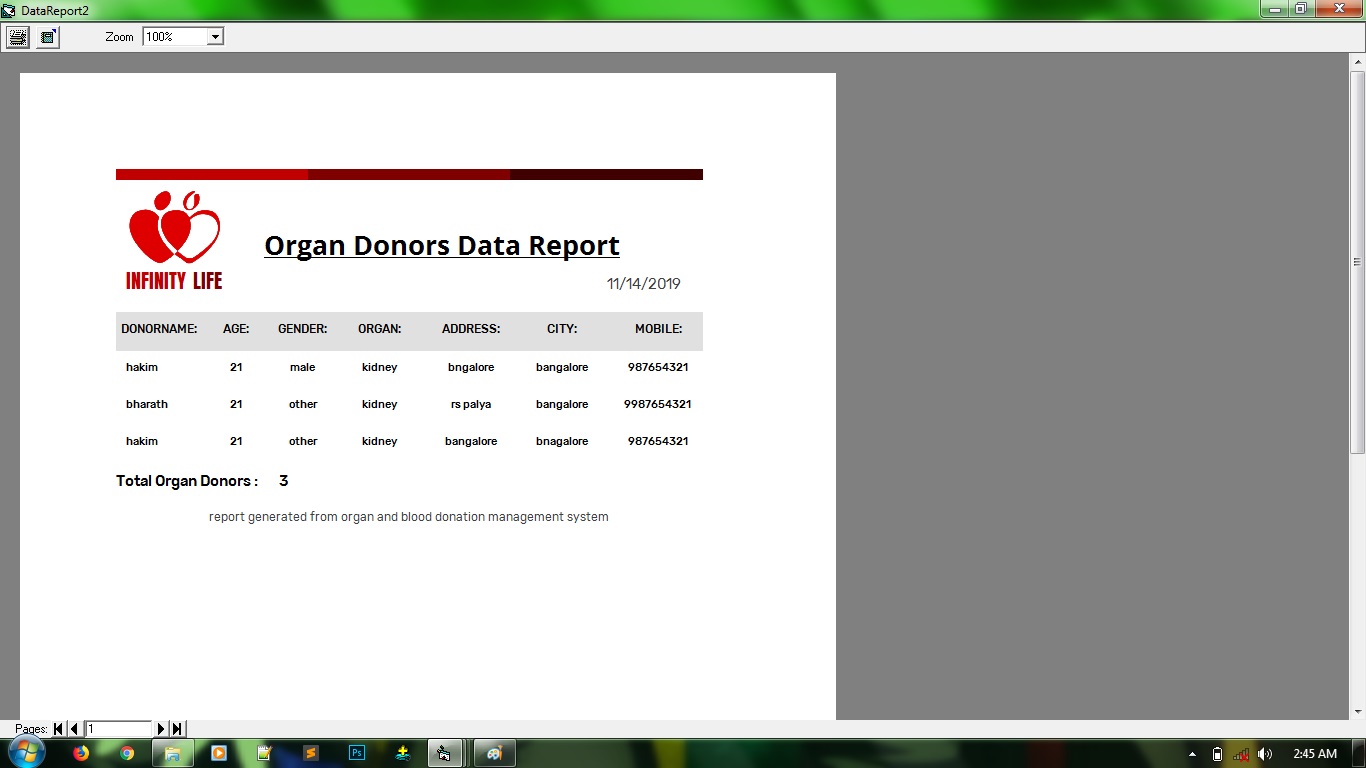
**User Details Report :-** In this page admin can view the user details and print them

****

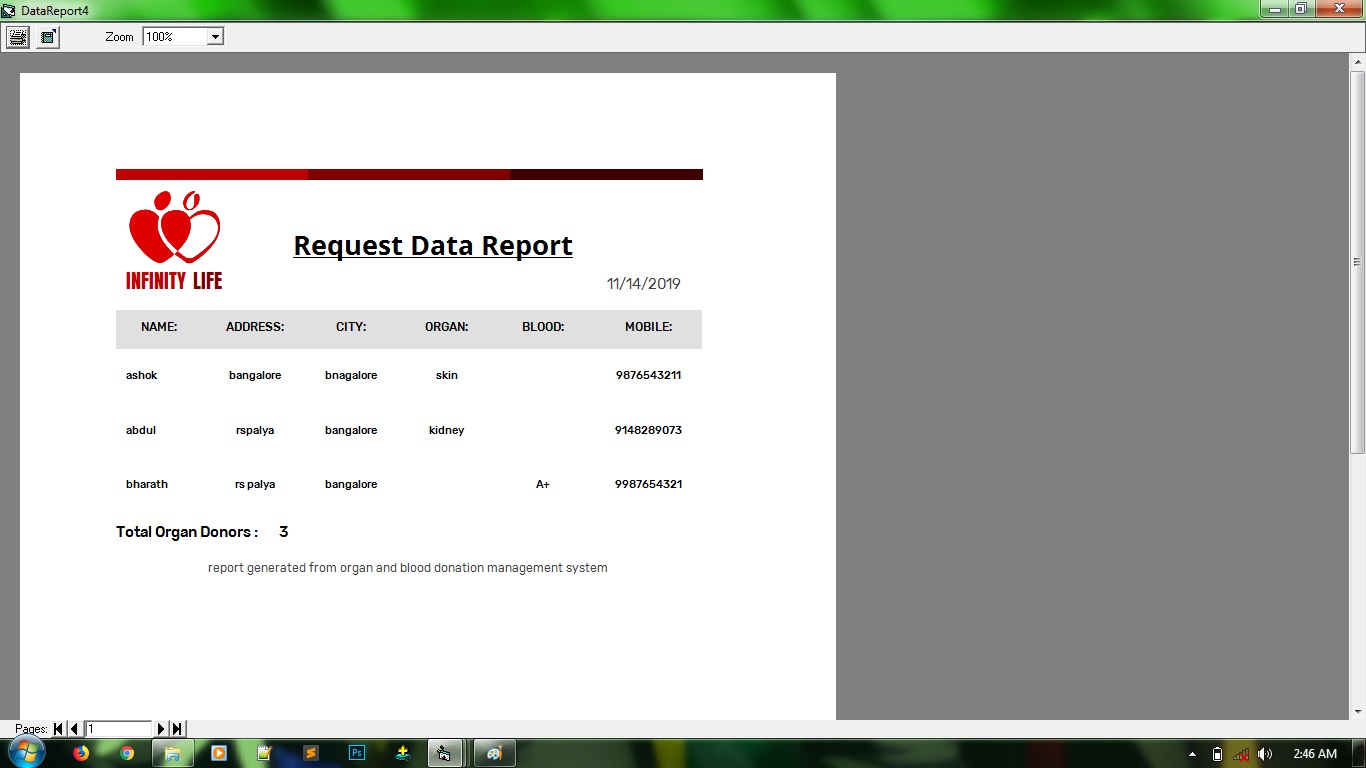
**Blood Donor Details Report :-** In this page admin can view the blood donor details and print them

****

**Organ Donor Details Report :-** In this page admin can view the organ donor details and print them

****

**Request Details Report :-** In this page admin can view the request details and print them

****

**Chapter 7**

**Conclusion**

Organ donors are needed to save thousands of lives every year. Agreeing to donate organs does not affect the quality of care a person receives. Transplantation processes start only after the donor is no longer living.

The decision to donate organs is yours. If you decide become an organ donor, put it in your advance health directives and make sure your family and health care agent know about your decision.

Organ transplant is the moving of an organ from one body to another, for the purpose of replacing the recipient`s damaged or failing organ with a working one from the donor site.

Organ donors can be living or deceased.

One thing is clear donating blood has many benefits to the donor as well as the recipient, so if you want to pick up a healthy habit head over to your local blood center and give the gift of life.

Blood donation is a one way to a person to a healthy life, this is because during the blood donation our body will be replaced with new blood cell which have a better protein, from this program awareness of importance of blood donation.

Each day about 60 people around the world receive an organ transplant, while another 13 die due to non-availability of organs.

The largest group waiting is from 18-14 years of age .

Organs and tissue from a single non-living donor can be used to benefit more than 50 people.

**Chapter 8**

**Future Enhancement**

There is a wide scope for future development of the application. The world of computer fields is not static it is always subject to change. Yet it will improve with further enhancements.

The project has been developed in a very short period of time and all the efforts have been taken so that this project is very efficient in its execution there still exists some scope of enhancement that can be added incorporate into the project

Help file can be included. The system as of now does not support any help facility for the user of the system.A help menu can be provided with a special function key and help command in main page itself.Help can be either introduced in a separate window,a reference to a printed manual or as one two line suggestion produced in a fixed screen location

Any other new requirements or services can be added easily without any major change in the existing application

**Chapter 9**

**Bibliography**

**Reference Book**

* Visual Basic 6 A Complete Reference
* Visual Basic Source Book Manual

**Websites**

* <http://www.google.com>
* <http://www.sourcecode.com>
* <http://tutorialspoint.com>
* <http://www.youtube.com>
* https://www.nhs.uk