



# BLOOD BANK

1Deepak Kumar, 2Deepak, 3Pramod Kumar, 4Ankit bhandari, 5Ankit Gupta

1student, 2student

1AKTU Lucknow,

2AKTU Lucknow

## INTRODUCTION

This project is focusing the preparation of such a Portal or Website, which intends to provide information about Online Blood Bank Management, Doctors and Camps.

The reason behind the creation of a website named **SaveLifeFoundation.Com** where a user or Patient can access or get information about Blood Banks in their locality by just visiting this site with a single click of a mouse. So they get the desired information about doctors or Blood Banks in emergency. You can take appointment with doctor or make a schedule with Blood Banks by sending them mail or by call them on their contact number through the contact number which will appear in a display in their page.

**The main features of this Website would be follows as:**

1. Collection of Information about the Blood Banks and Doctors.
2. Collection of information about the Users or Patients.
3. Creation of User Account with collecting the suitable information from the user.
4. Creation of Blood Bank by collecting the suitable information from the user and registers them with User ID so User can Change their information by their User ID which are posted on website.

5. Creation of Blood Bank with collecting the suitable information from the Blood Bank authorities and Register them so they can edit the data with their Blood Bank ID which they posted on website, so the details will be displayed to the user when they access the information of Blood Bank.
6. Creation of Administrator account so they can login on the website using their Admin ID and Password.
7. It will display the various services when user selecting an option from the Dropdown List or just clicking a link of the particular service.
8. There is facility to receive/send mail from to Blood Bank or any one and take appointment or get the message suggestion.

The present approach is given the Information Technology. IT is a new and fresh integrated approach and a key to unlock all the intricate problems of our modern era.

This technology tends to cater all our needs. It is not only an 'Integrated Approach', but also job-oriented & concrete device to aware us of the up-to-date skill and knowledge.

This branch of science plays a vital and significant role in catering information, recreation & what not? IT is instructive as well. It is an instrument of bringing us to a realm of technology. It has entirely revolutionized the vista of the modern world. It has made the modern life easier, softer, better, happier and more prosperous.

It has worked a magic wand in the spheres of industry, medicine, engineering, agriculture, transportation, electronics & electrical devices. IT has boosted progress beyond description. It has given us the safest & the most concrete means of instruction i.e. Computer, Internet, E-mail, E-commerce, Website and a lifeline for most of our problems.

IT affords us of whatever we could not dream of. All these devices under IT have drastically changed communication that now links the world within the twinkling of an eye.

IT has to a larger extent, played a role in bettering the global economy. Although there are many companies offering these services but still quality work is always in denied.

### 1.1 Purpose

The purpose of this document is to describe the external requirements for **SaveLifeFoundation.com** web site.

### 1.2 Scope

This document is the only one that describes the requirements of the system. It is meant is the only one that describes the requirements of the system. It is meant for use by the developers and will be the basis for validating the final delivery system. Any changes made to the requirements in the future will have to go through a formal change approval process.

### 1.3 Applicability

This Website can be used in the universities, colleges and Hospitals which opt for **SaveLifeFoundation.com** website.

## OBJECTIVE

The objective of this project is to provide facility of online information of Blood Banks and Doctors to the Users.

In an era where information is most important to the persons, institutes and organizations, as most of the money spent by these groups to gather the information and process the data for its effective storage of information in a way that information remains effective, efficient and non-redundant, there was little importance given to the way the information was stored and retrieved as per their needs. Not only the information is ineffective and inefficient, it also gets redundant, thus making job of the System Administrator more and more difficult.

In earlier age there is no any medium to know the information about Blood Banks and Doctors so they can approach them. To overcome this problem we are going to develop a website from that the user can get the information of Blood Bank and Doctor in their locality or anywhere in India so they can contact the Blood Bank or Doctor and take appointment. This web portal is used to make his work very easy. Moreover this approach increases efficiency in Data and effectiveness in information stored.

The objective of this project is to provide facility of online information of Blood Donors, Blood Request and Blood Management to the Users. To design a software package, which helps the user or patients to retrieve the information of the blood or Doctors who has posted their details on the portal? Hence the software is known as "**SaveLifeFoundation.Com**".

**SaveLifeFoundation.Com** part would enable users or patients gather information of particular Blood Request, Blood Donate and Doctors suggestions. At the end of this part of project we will able to achieve following goals.

- This will provide the facility of SEARCH by which users will able to search Blood Bank by their location.
- User will able to register them and get an id and password. This facility provide user to login by their User ID and Password.
- Blood Bank will register them and post their data on the website that would be available to the users or patients. Later they can edit the data which they posted on the website by using their BLOOD BANK ID and password.
- Administration will able to achieve use special function like make the status of Blood Bank or User is active or not active. They grant privileges to the user according to their position. Alter the rates of posting made by Blood Bank or User queries will be solved by the admin.
- Administrator allows the User, Blood Bank, Doctor, and Camp to change their Password or any information which they submit earlier.
- User will able to search the blood donation camp.
- Registered user will able to post the donate/request blood and any user will able to search available services.
- All type of information like Blood Bank details or Doctor's details will be available for users. Users will able take appointment from the doctor or according to his needs.
- User will able to register them to use some prohibited features like mail facility through which they can mail to anyone or Donor to take appointment.
- Administration will able to control the activity of blood.

The software package will enable the Administrator to do administration on the data which are posted by different users, doctors or Blood Banks more efficiently, effectively with no more redundancy in data. The software is build using .NET as Front End and SQL Server 2008 as Database.

## **.NET**

.NET comes with a Developer environment, Visual Studio 2010, which can cope equally well with C++, C# and VISUAL BASIC .Net as well as ASP.NET code.

## **ADVANTAGES OF .NET**

- Object Oriented Programming
- Good Design
- Language Independent
- Better Support for dynamic pages
- Efficient Data Access
- Code Sharing
- Improved Security
- Support for Web Services

## **SQL SERVER 2008**

SQL Server Express is the 2008 version of SQL SERVER that replaces MSDE (Microsoft Server Data Engine). The version does not have the strict limitations MSDE had.

Due to the fact SQL Server 2008 now host the CLR, this means that you can now avoid building database aspects of your application using the T-SQL programming language. Instead, you can now build items such as your stored procedures, triggers and even data types & any of the .NET compliant languages, such as C#.

**SaveLifeFoundation.Com** part would enable users or patients gather information of particular Blood Bank or Doctor. At the end of this part of project we will able to achieve following goals.

## PROJECT CATEGORY

In this software information is handled by using SQL SERVER 2008 for storing records in a database, as backend. ASP.NET with C# is used as front end. HTML used as hypertext markup language and CSS is used for presentation of Web pages as language, and Ajax is used to make the page attractive. Caching is used to make less traffic on the Server. Project categorized in a web based system, which keeps all the information about Blood Banks, Users, Doctors spreading all over the India so any one can get the details of Doctors by single click of Mouse in their locality or anywhere in India.

### TOOLS/PLATFORM, LANGUAGES TO BE USED

Software and hardware specification as the name suggests, tells us about the various characteristics of the software and the hardware environment used i.e. the development environment used. Here we specify various software languages, supporting tools that have been used for the development of the system. These tools and the languages have been used because of their relative ease of understand and personal interest of the team developing the project.

### HARDWARE REQUIREMENTS SPECIFICATIONS

#### HARDWARE

<b>PROCESSOR</b>	<b>: Intel Dual Core or Above</b>
<b>HARD DISK DRIVE</b>	<b>: 320 GB SATA or Above</b>
<b>RAM</b>	<b>: 2 GB SD DDR RAM OR Above</b>
<b>CACHE</b>	<b>: 2 MB L2 CACHE</b>

#### TOOLS

<b>PLATFORM</b>	<b>: WINDOWS 7 or above</b>
<b>WEB TECHNOLOGIES</b>	<b>: HTML, CSS, C#, ASP.NET</b>
<b>DOCUMENTATION TOOLS</b>	<b>: MS-WORD</b>
<b>WEB SERVER</b>	<b>: IIS</b>
<b>DATABASE</b>	<b>: SQL SERVER 2008</b>

## **WORKING ENVIRONMENTS**

### **ASP.NET**

ASP.NET is an Object Oriented Programming Language. It was developed by Bjarne Stroustrup at AT&T Bell Laboratories in Murry Hill, New Jersey, and USA in early 1980. ASP.NET is an extension of C with a major addition of the class construct feature of Simula 67.

The most important facilities that ASP.NET adds on to C are Classes, Inheritance, Method Overloading and Operator Overloading. These features enable creating of abstract data types, inherit properties from existing data types and support polymorphism, there was a making ASP.NET a truly Object Oriented Language.

The Object Oriented Features in ASP.NET allow Programmers to build large programs with clarity, extensibility and easy to maintenance, incorporating the spirit and efficiency of C.

### **APPLICATION OF ASP.NET**

ASP.NET is a versatile language for handling very large programs. It is suitable for virtually any programming task including development of editors, compilers, database communication systems and any real life application systems.

Since, ASP.NET allows us to create hierarchy related objects; we can build special object oriented libraries which can be used later by many programmers.

ASP.NET programs are easily maintainable and expandable when a new features needs to be implemented, it is a very easy to add to the existing structure of the object.

### **Components of ASP .NET**

.NET framework has two main components. They are:

- Common Language Runtime
- .NET class library

### **Common Language Runtime**

The Common Language Runtime (CLR) is the environment where all programs in .NET are run. It provides various services, like memory management and thread management.

## ASP .NET Class Library

.NET comes with thousands of classes to perform all important and not-so-important operations. Its library is completely object oriented, providing around 5000 classes.

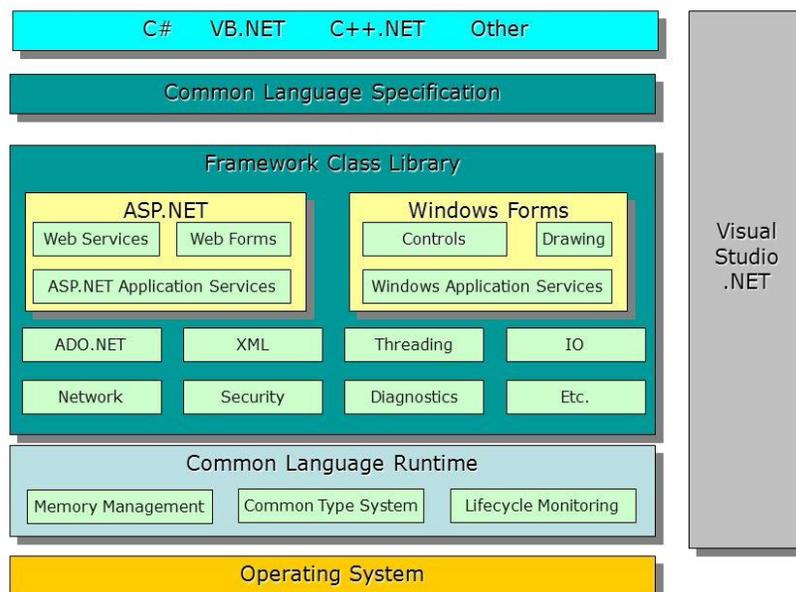
The following are the main areas that are covered by Class library.

- Data Structures
- IO management
- Windows and Web Controls
- Database access
- Multithreading
- Remoting
- Reflections

Also the library is common for all types of applications. The following are different types of applications that can make use of .NET class library.

- Console applications.
- Windows GUI applications.
- ASP.NET applications – web applications.
- XML Web services.
- Windows services.

## .NET Framework



### (Components of .NET Framework)

## Features of ASP .NET

The following are major features of .NET. We will use these features throughout journey. Here is just a brief introduction to all key features of .NET **Assemblies**

An assembly is either a .DLL or .EXE that forms a part of an application. It contains MSIL code that is executed by CLR

It is the unit on which permissions are granted. Every assembly contains a version.

Assemblies contain interfaces and classes. Every assembly contains assembly metadata, which contains information about assembly. CLR uses this information at the time of executing assembly.

Assemblies may be either private, which are used only by the application to which they belong or Global assemblies, which are used by any application in the system.

Two assemblies of the same name but with different versions can run side-by-side allowing applications that depend on a specific version to use assembly of that version.

The four parts of an assembly are:

- Assembly Manifest
- Type metadata
- MSIL
- Resources
- Common Type System

Common Type System (CTS) specifies the rules related to data types that languages must follow

## Cross-language Interoperability

.NET provides support for language interoperability. However, it doesn't mean every program written in a language can be used by another language. To enable a program to be used with other languages, it must be created by following a set of rules called Cross Language Specifications (CLS).

## Visual Studio.NET

VS.NET is the application development tool to develop applications for .NET. It supports development of all types of applications that .NET supports. It also provides support for VB.NET ,C#, Visual C++.Net and Visual J# languages. VS.NET is a single environment that provides all tools required to develop and debug applications.

## Languages supported

VS.NET supports application development using the language of your choice. It also allows mixed language solutions.

## Intelligence

Intelligence provides options that make programming in VS.NET easier than ever before.

# **INTRODUCTION TO MICROSOFT SQL SERVER 2008**

The SQL Server product is primarily divided into:

- **SQL Server Tools**
- **SQL Server Client Tools**

## **1. SQL Server**

Microsoft the No. one company in the software that produces the most widely used, front end and Server based Multi user RDBMS. The SQL Server is a program installed on the Server's hard disk driver. This program must be loaded in RAM so that it can process user requests. The SQL Server product is either called SQL Server Professional or SQL Server Enterprise.

The SQL Server takes care of the following:

- Updating the database.
- Retrieving information from the database.
- Accepting query language statements.
- Enforcing security specifications.
- Enforcing data integrity specifications.
- Enforcing transaction consistency.
- Managing data sharing.
- Optimizing queries.
- Managing system catalogs.

## **SQL Server Client Tools:**

Once the SQL Server engine is loaded into the server's memory, users would have to log into the engine to get work done. Microsoft has several client-based tools that facilitate this. The client tool most commonly used for Commercial Application Development is Visual basic.

## What is SQL used for-

Using SQL one can create and maintain data manipulation objects such as table, views, sequence etc. These data manipulation objects will be created and stored on the server's hard disk drive, in a table space, to which the user has been assigned.

Once these data manipulation objects are created, they are used extensively in commercial applications.

## DML, DCL, DDL

In addition to the creation of data manipulation objects, the actual manipulation of data within these objects is done using SQL.

The SQL sentences that are used to create these objects are called DDL's or Data Definition Language. The SQL sentences used to manipulate data within these objects are called DML's or Data Manipulation Language. The SQL sentences, which are used to control the behavior of these objects, are called DCL's or Data Control Language.

# SYSTEM ANALYSIS AND SPECIFICATION

## SYSTEM ANALYSIS

System analysis is the process of studying the business processes and procedures generally referred to as business systems, to see how they can operate and whether improvement is needed. This may involve examined data movements and storage, technology used in the system, programs that control the system, people providing inputs, doing the processing and receiving the inputs.

## INVESTIGATION PHASE

The investigation phase is also known as fact finding stage or the analysis of the current system. This is a detailed study conducted with the purpose of wanting to fully understand the existing system and to identify the basic information requirement. Various techniques may be used in fact finding and all the fact must be recorded.

## INVESTIGATION

As it was essential for us to find out more about the present system we use the following method to gather the information.

- Observations: necessary to see the way the system works.

- Document sampling: These are all documents that are used in the system. They are necessary to check all the data that enters the system and the output taken from system.

## ANALYSIS OF THE INVESTIGATION

### Strength of the System

- **No complex equipment:** The equipment that is used is very simple and no special skills have to be mastered to be able to operate the systems. Therefore, no training is required for employees.
- **Low Cost:** There is low cost spend in maintaining the system other than buying the necessary office equipment and the software.

### Constraints and limitations

The constraints and limitations within a system are draw back that occur during the implementation of the system. These limitations and constraints can crop up in all most every system the most important fact is to find a way to overcome these problems. Soft- ware design is the first of three technical activities that are required to build and verify the software.

When I started working on system design, I face different types of problems many of these are due to constraints imposed by the user or limitations of hardware and software available.

## PROJECT PLANNING

### GANTT CHART & PERT CHART

**(1) Gantt Chart** – Gantt charts are a project control technique that can be used for several purposes, including scheduling, budgeting, and resource planning. A Gantt chart is a bar chart, with each bar representing an activity. The bars are drawn against a time Line. The length of each bar is proportional to the length of time planned for the activity.

A Gantt chart helps in scheduling the activities of a project, but it does not help in identifying them. Gantt charts take different forms depending on their intended use. They are best for resource scheduling.

When creating a software project schedule, the planner begins with a set of tasks. If automated tools are used, the work break down is input as a task network or task outline. Effort, duration and start date are then input for each task. In additional tasks may be assigned to specific individuals.

As a consequence of this input a timeline chart also called a Gantt chart, is generated. A timeline chart can be developed for the entire project. Alternatively, separated charts can be developed for each project function or for each individual working on the project.

<b>WORK TASKS</b>	<b>ESTIMATED TIME (Week)</b>
Problem Evaluation (PE)	1
Calculating Fact (CF)	1
Analysis (A)	3
Identify Need & Benefits (INB)	1
Feasibility (F)	1
Design Layout (DL)	2
Detail Design (DD)	3
Testing of Design (TD)	2
Coding (C)	5
Testing of Code (TC)	5
Code Optimization (CO)	1
Implementation (I)	1

**GANTT CHART**

<b>Tasks Identified</b>	<b>Planned start</b>	<b>Actual start</b>	<b>Planned Complete</b>	<b>ACTUAL COMPLETE</b>
Preliminary Study	Wk1, d1	Wk1, d2	Wk4, d6	Wk5, d1
Definition of Desired Output/Input and Controls	Wk2, d1	Wk2, d3	Wk3, d4	Wk3, d4
Research on availability of desired system and development environment	Wk3, d2	Wk3, d4	Wk4, d4	Wk4, d6
Creation of Review documents and fixing the Scope of documents	Wk4, d5	Wk5, d1	Wk6, d5	Wk7,d2
Preliminary Designing	Wk6, d5	Wk7, d1	Wk9, d3	Wk10, d2
Detailed Designing	Wk8, d1	Wk8, d4	Wk16, d6	Wk17, d5
Coding	Wk15, d2	Wk16, d1	Wk22, d6	Wk23, d4
Testing	Wk22, d1	Wk22, d3	Wk24,d6	Wk25, d2
Implementation	Wk25,d1	Wk25, d2	Wk26, d6	Wk26, d6

**PERT CHART:-** Scheduling of a software project does not differ greatly from scheduling of any multitasking engineering effort. Therefore generalized project scheduling tools and techniques can be applied with little modification to software project. PERT (program evaluation and review technique) is project-scheduling methods that can be applied to software development. PERT techniques are driven by information already developed in eastern project planning activities.

PERT provide quantitative tools that allows the software planner to determine the critical path -The chain of task determines the duration of the project; establish, "most likely " time estimates for individual task applying statistical models; and calculate: "boundary time " that define a time "window for a particular task.

Boundary time calculations can be very useful in software project scheduling. PERT have been implemented in a wide variety of automated tools that are available for the personal computer. Such tools are easy to use and make the scheduling methods described previously available to very software project manager.

Now the Pert chart of the above data is as follows:

### PERT CHART

Tasks Scheduled	Time Scheduled ( 1 Unit = 1 Week)													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Preliminary Investigation														
Definition of Desired Output/Input and Controls														
Research on availability of desired system and development environment														
Creation of Review documents and fixing the Scope of documents														
Preliminary Designing														
Detailed Designing														
Coding														
Testing														
Implementation														



## SYSTEM DESIGN

### DATA FLOW DIAGRAM

A **data-flow diagram (DFD)** is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing (structured design).

On a DFD, data items flow from an external data source or an internal data store to an internal data store or an external data sink, via an internal process.

A DFD provides no information about the timing of processes, or about whether processes will operate in sequence or in parallel. It is therefore quite different from a flowchart, which shows the flow of control through an algorithm, allowing a reader to determine what operations will be performed, in what order, and under what circumstances, but not what kinds of data will be input to and output from the system, nor where the data will come from and go to, nor where the data will be stored (all of which are shown on a DFD).

There are different notations to draw data-flow diagrams, defining different visual representations for processes, data stores, data flow, and external entities.

Data flow diagram ("bubble charts") are directed graphs in which the nodes specify processing activities and the arcs specify data items transmitted between processing nodes.

The following table lists the important elements of DFDs.

Symbol	Stands For
Data process	Data processing
Data flow	Data flow or the exchange of data between processes
Data store	Data storage
Actor	Object producing and consuming data

#### **Data process**

A data process transforms data values.



#### **DATA STORE**

**ACTOR**

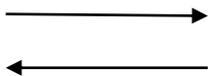
An actor produces and consumes data, driving the DFD. Actors lie on the boundary of the diagram; they terminate the flow of data as sources and sinks of data. They are also known as terminators. Data flows between an actor and a diagram are inputs to and outputs of the diagram. The system interacts with people through the actor.



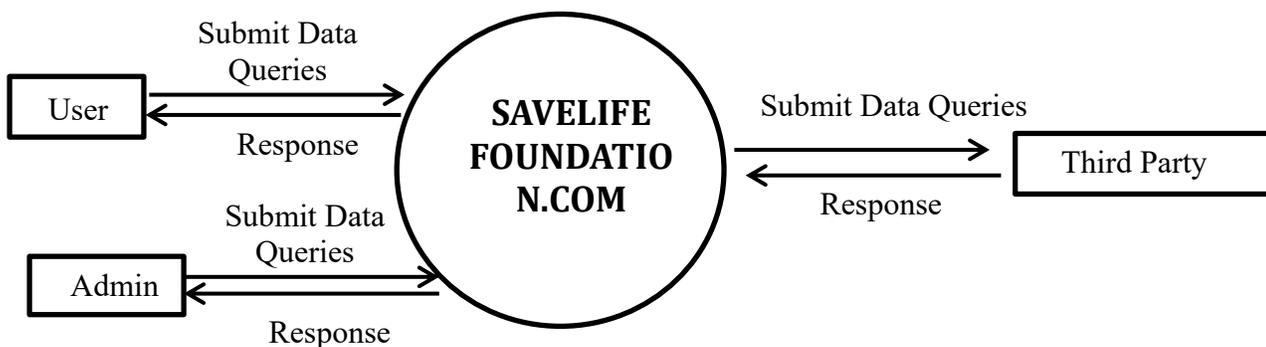
**DATA FLOW**

A data flow moves data between processes or between processes and data stores. As such, it represents a data value at some point within a computation and an intermediate value within a computation if the flow is internal to the diagram. This value is not changed.

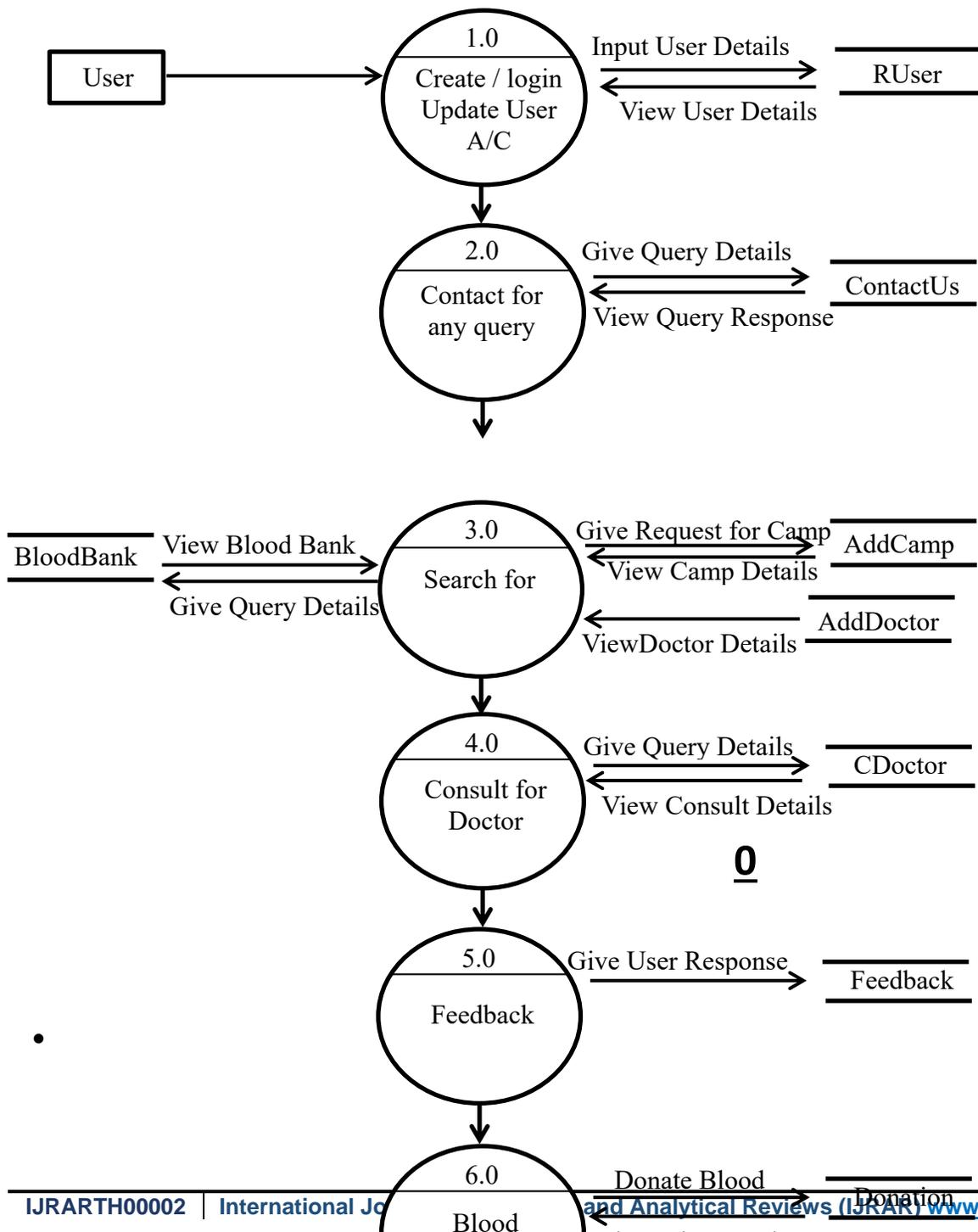
The names of input and output flows can indicate their roles in the computation or the type of the value they move. Data names are preferably nouns. The name of a typical piece of data, the data aspect, is written alongside the arrow.

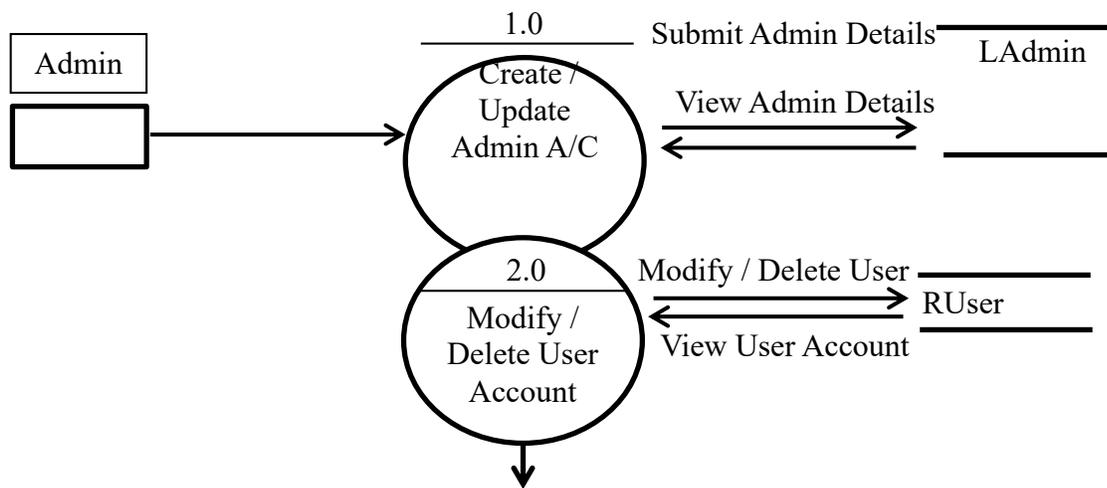


**CONTEXT LEVEL DFD**

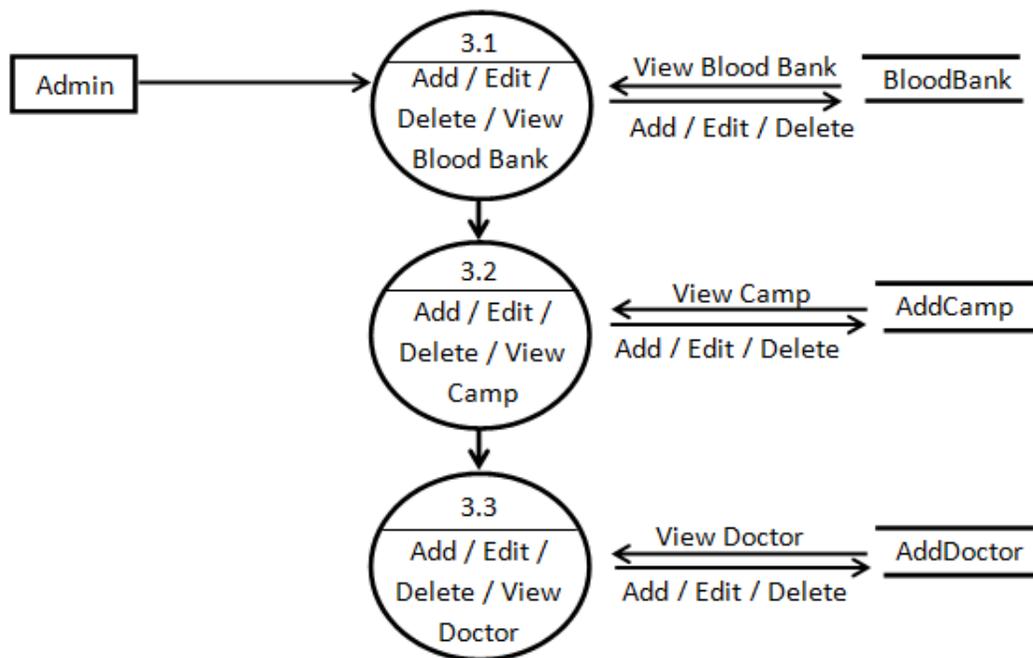


### 0 LEVEL DFD FOR USER 1.0

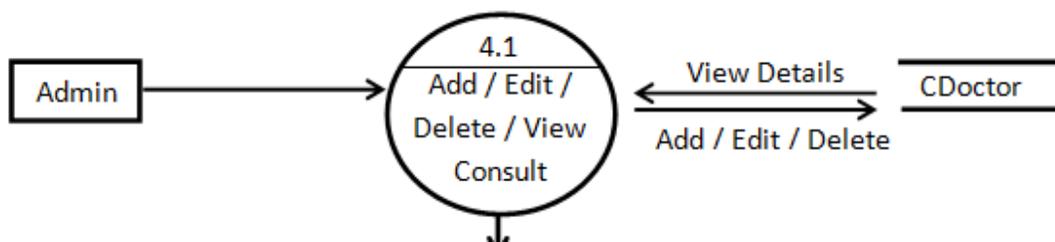




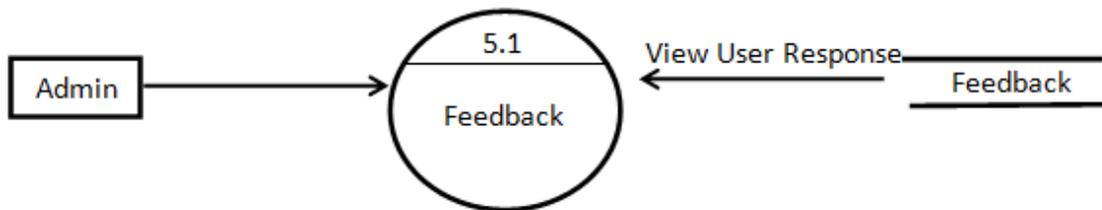
**1 LEVEL DFD FOR ADMIN 3.0**



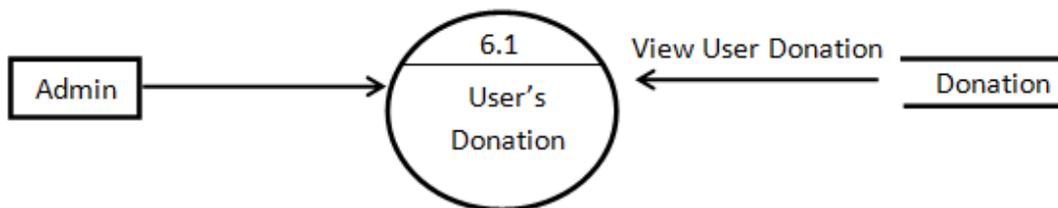
**1 LEVEL DFD FOR ADMIN 4.0**



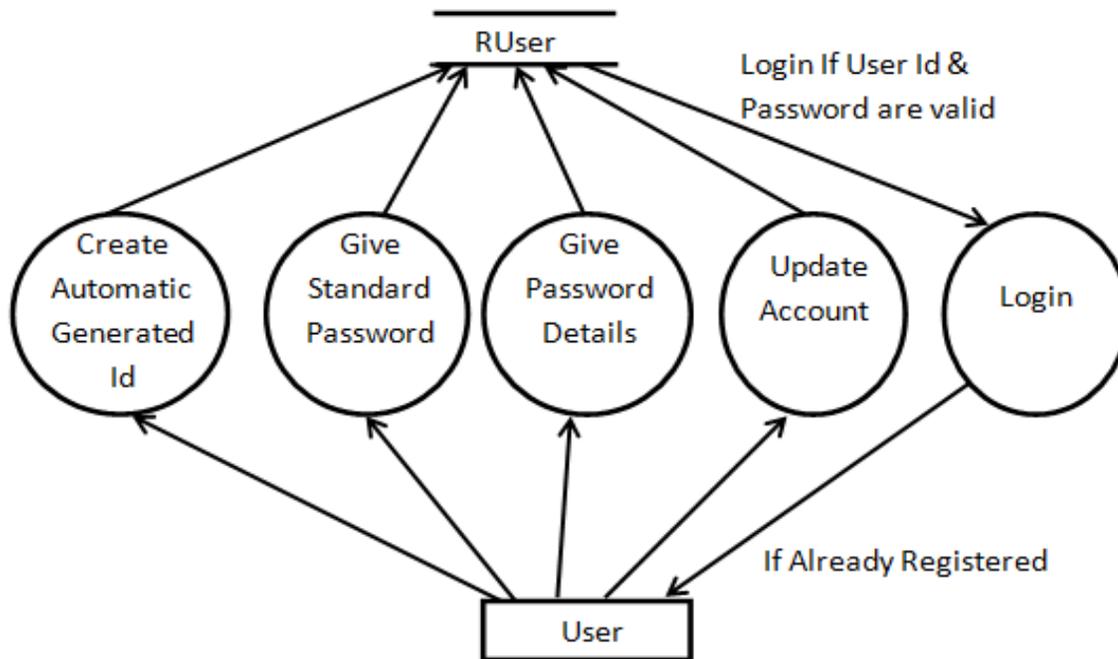
### 1 LEVEL DFD FOR ADMIN 5.0



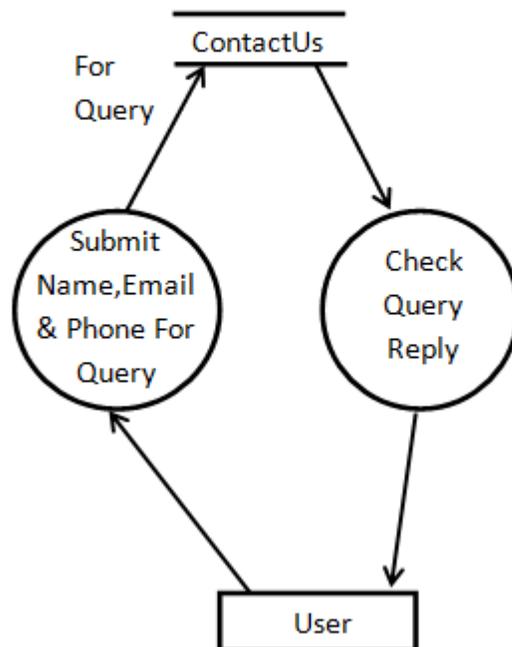
### 1 LEVEL DFD FOR USER 6.0



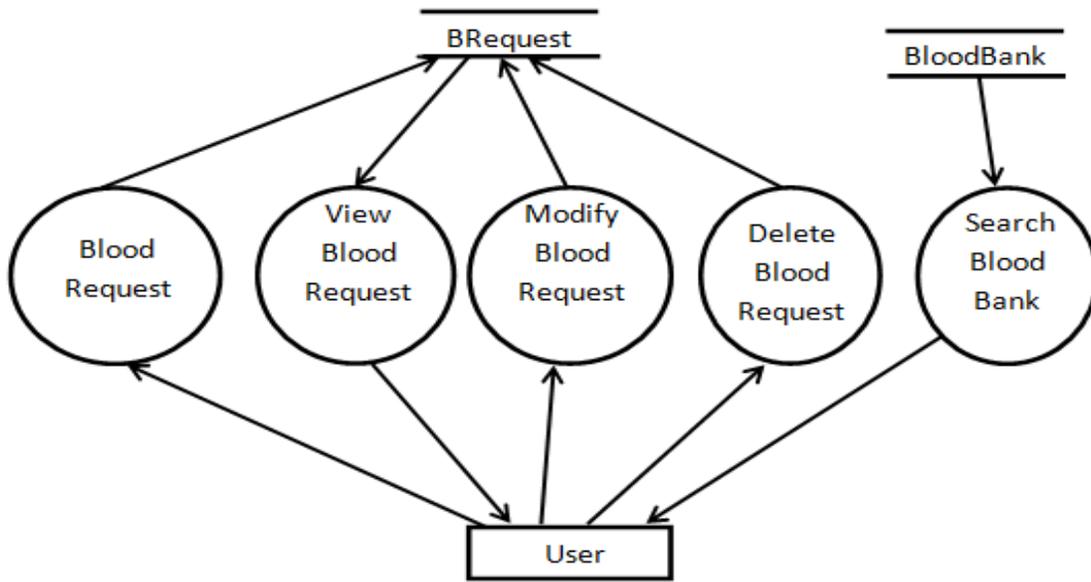
## 2 LEVEL DFD FOR USER REGISTRATION



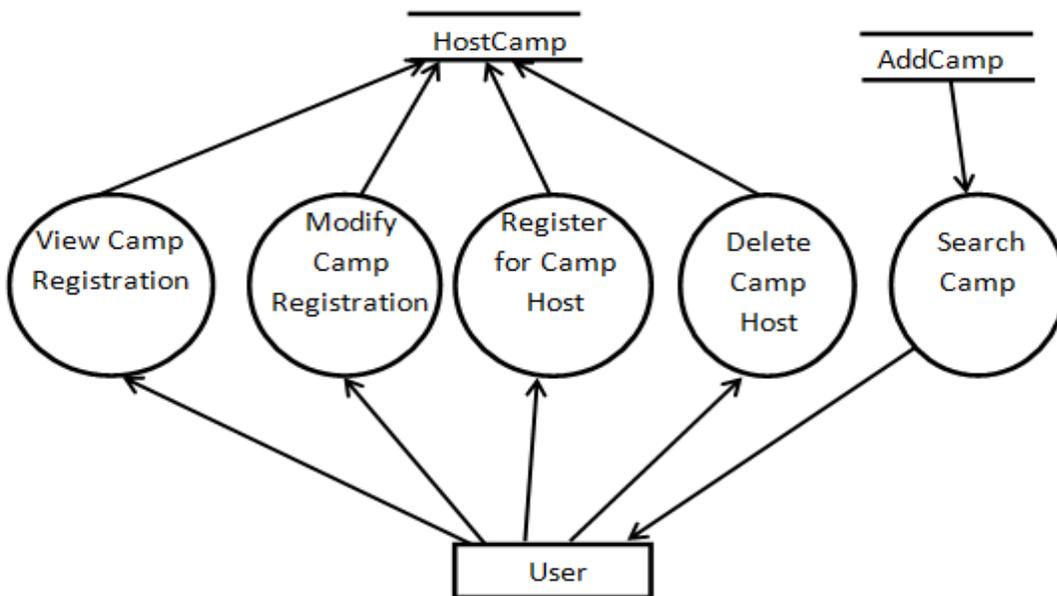
## 2 LEVEL DFD FOR USER QUERY



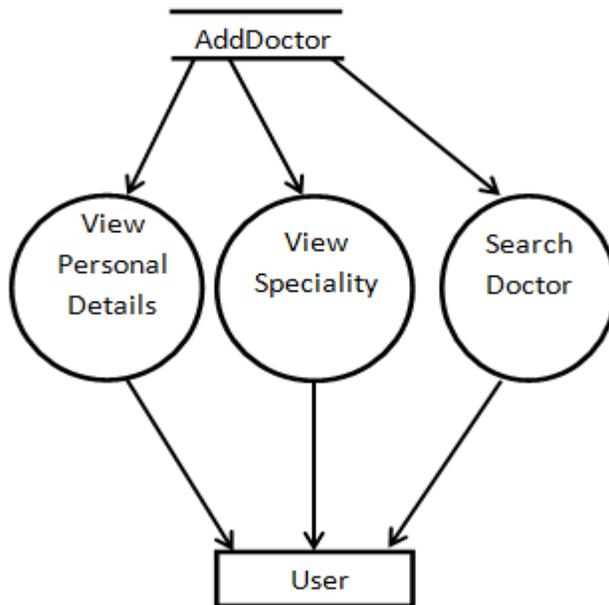
## 2 LEVEL DFD FOR USER BLOOD BANK



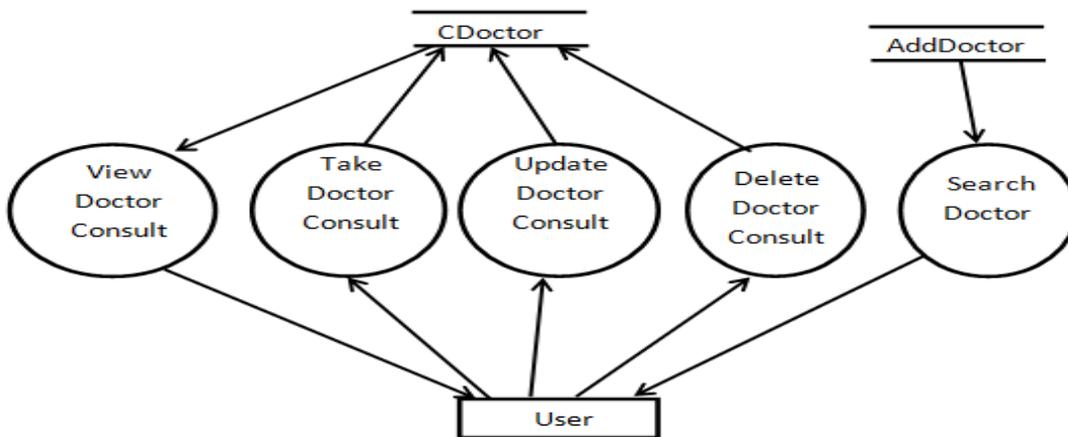
## 2 LEVEL DFD FOR USER CAMP



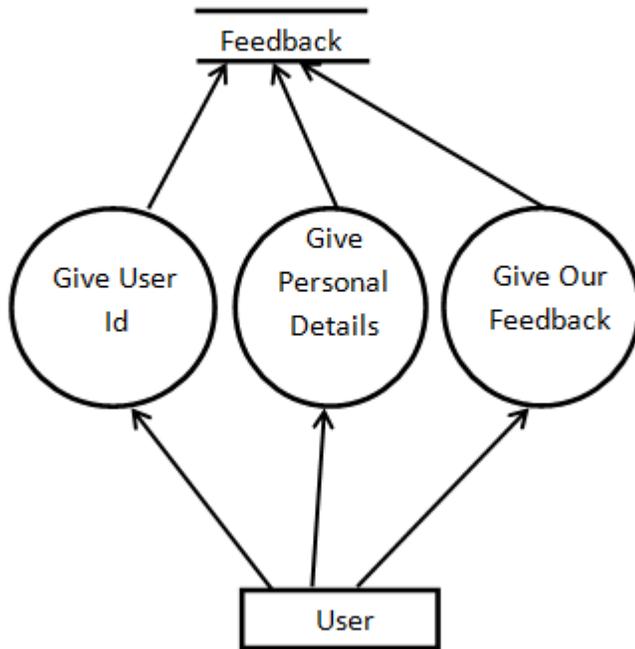
## 2 LEVEL DFD FOR USER DOCTOR



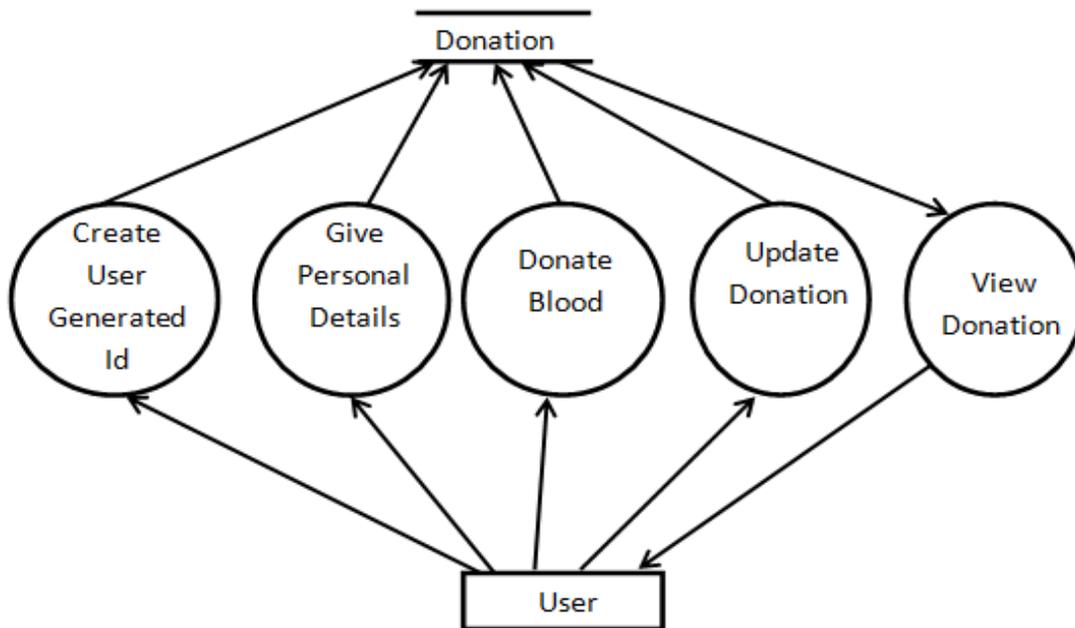
## 2 LEVEL DFD FOR USER CONSULT



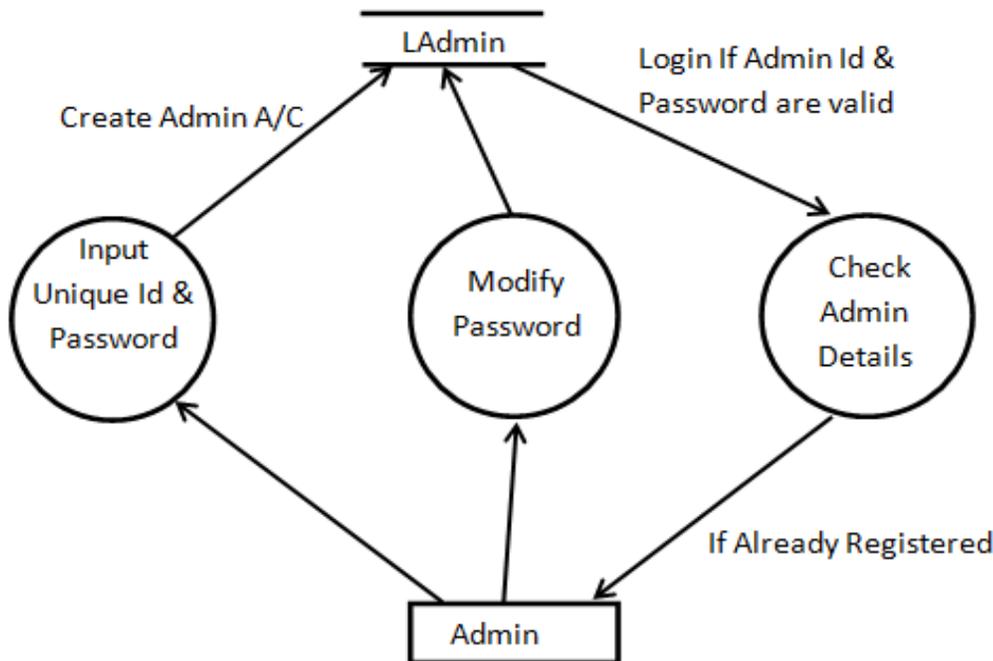
## 2 LEVEL DFD FOR USER FEEDBACK



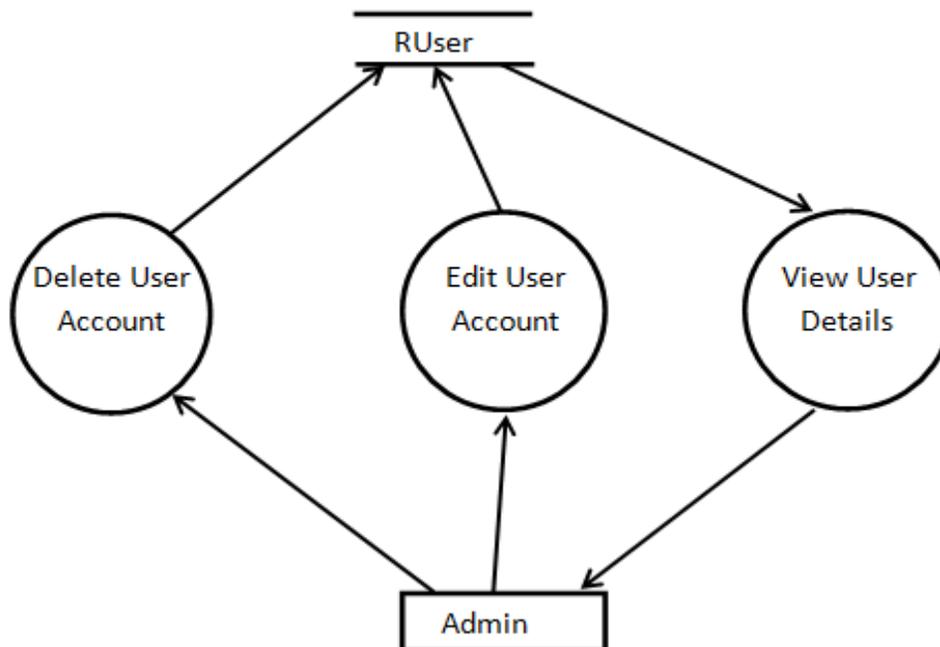
## 2 LEVEL DFD FOR USER DONATION



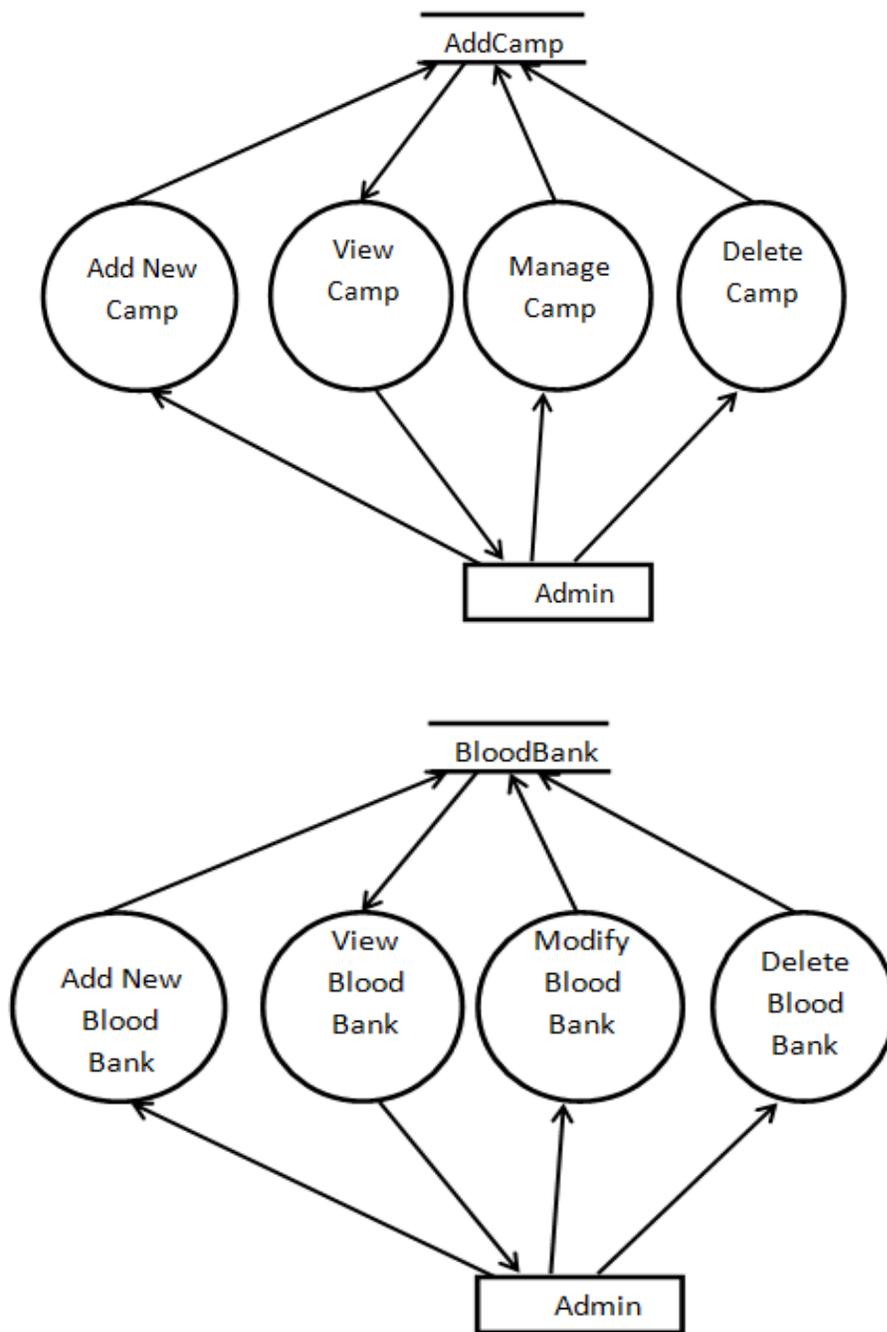
## 2 LEVEL DFD FOR ADMIN REGISTRATION



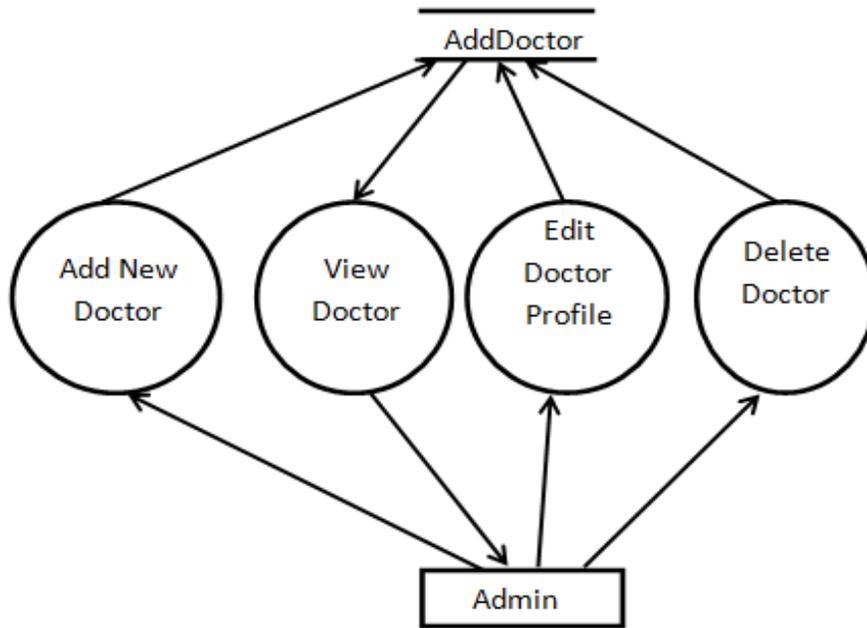
## 2 LEVEL DFD FOR ADMIN MANAGE USER



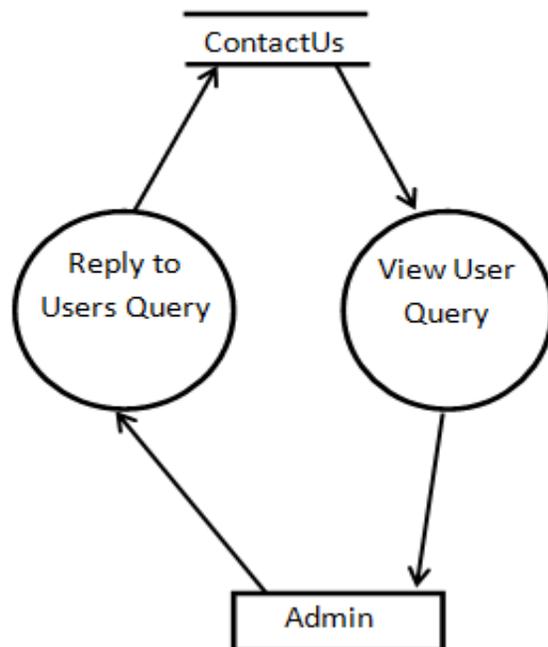
## 2 LEVEL DFD FOR ADMIN CAMP



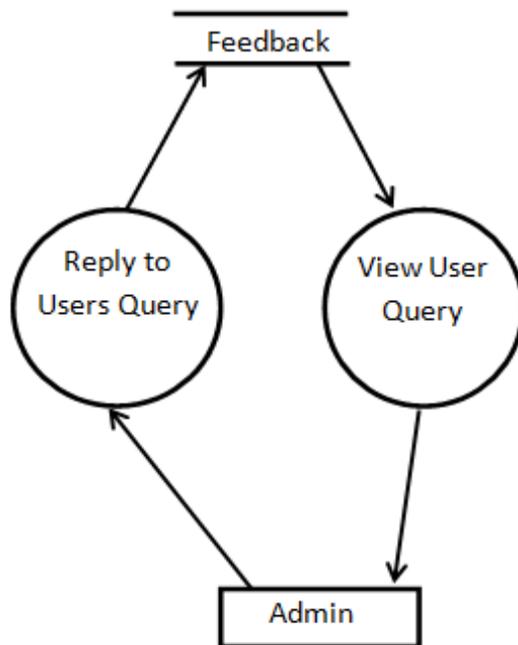
## 2 LEVEL DFD FOR ADMIN DOCTOR



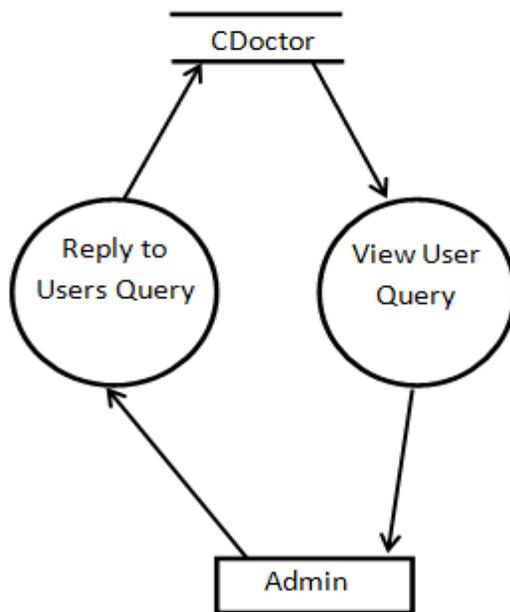
## 2 LEVEL DFD FOR ADMIN CONTACTUS



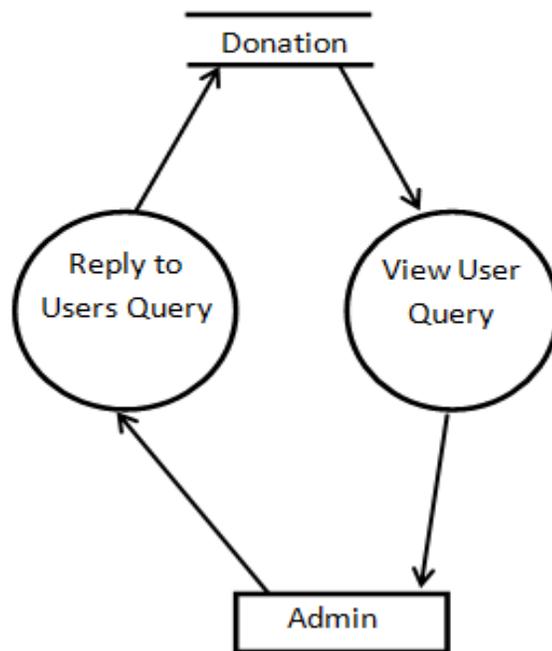
## 2 LEVEL DFD FOR ADMIN FEEDBACK



## 2 LEVEL DFD FOR ADMIN CONSULT



## 2 LEVEL DFD FOR ADMIN DONATION



## ENTITY RELATIONSHIP DIAGRAM

An entity relationship diagram is a graphical representation of entities and their relationships to each other, typically used in computing regarding the data within database or information system.

Entity Relationship Diagrams have three different components:

- Entities
- Attributes
- Relationship

### ➤ Entities

Entities are the principal data object about which information is to be collected. Entities are usually recognizable concepts, either concrete or abstract, such as person, places, things, or events, which have relevance to the database. An entity is analogous to a table in the relational model.

Entities defined in this system are:

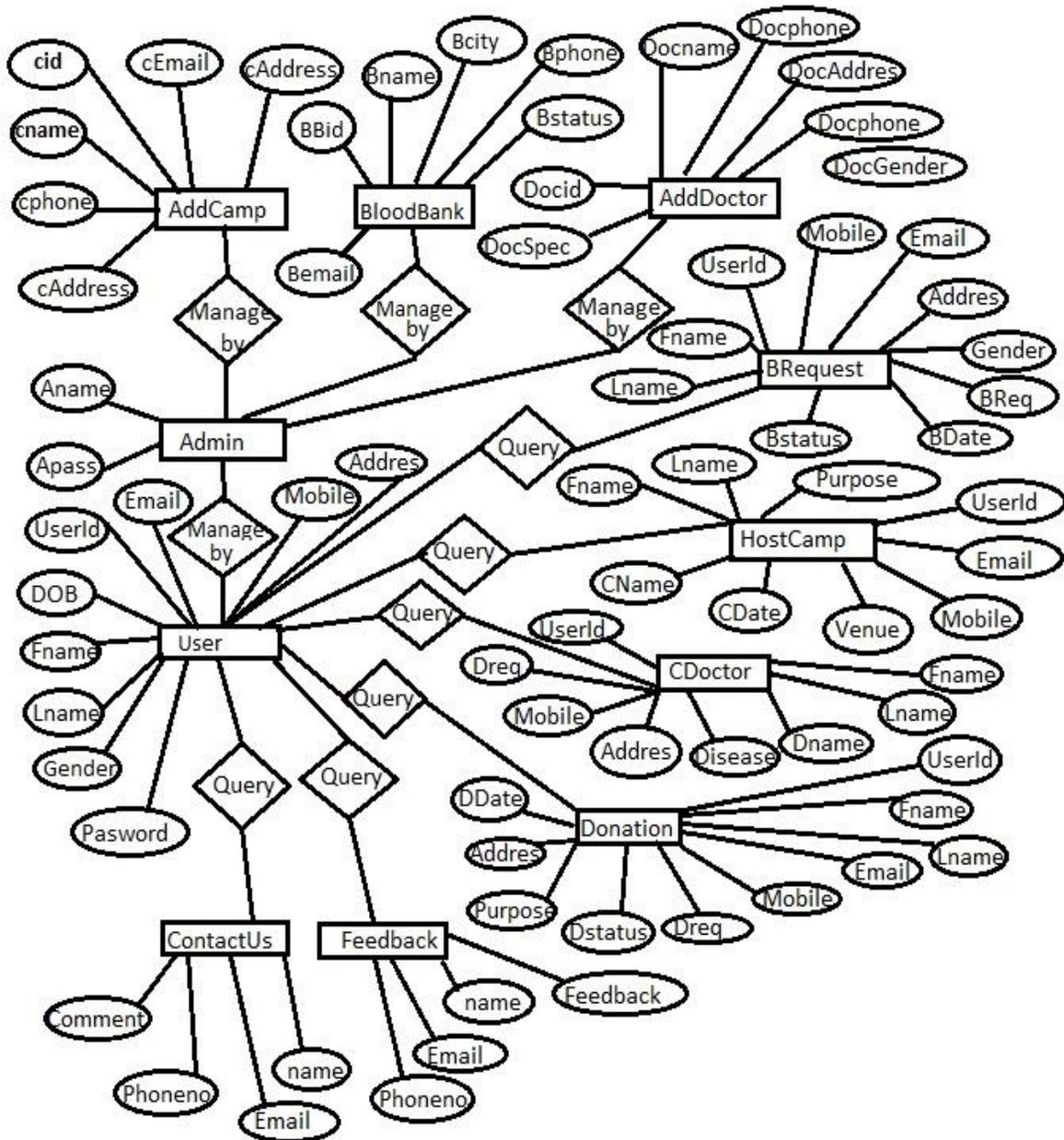
- Employee
- Project etc.

### ➤ Attributes

A data attribute is a characteristic common to all or most instances of a particular entity. An attribute or combination of attributes that uniquely identifies one and only one instance of an entity is called a primary key or identifier.

- Attributes of User are : - ID, Password, Address, Mobile, Email etc.
- Attributes of Project are: - ID, Project Name, Deadline etc.

**Relationship:** A data relationship is a natural association that exists between one or more entities. The connectivity of a relationship describes the mapping of associated entity instances in the relationship.



**CLASS DIAGRAM**

Class Name: LAdmin		
Attributes	Data Types	Descriptions
Aname	varchar(30) primary key	Name of Admin
Apass	varchar(30)	Admin Password
Operation : Login, New Admin		

Class Name: RUser		
Attributes	Data Types	Descriptions
UserId	int identity primary key	User Id
Fname	varchar(50)	User First Name
Lname	varchar(50)	User Last Name
DOB	varchar(50)	User Date of Birth
Gender	varchar(50)	User Gender
Blood_Group	varchar(30)	User Blood Group
Email	varchar(50)	User Email
Addres	varchar(50)	User Address
Mobile	varchar(10)	User Contact
Pasword	varchar(20)	User Password
Operation : New User, Update, Delete		

Class Name: AddDoctor		
Attributes	Data Types	Descriptions
Docid	varchar(50) primary key	Doctor Id
Docname	varchar(50)	Doctor Name
Docphone	varchar(10)	Doctor Phone Number
DocEmail	varchar(50)	Doctor Email Id
DocAddress	varchar(80)	Doctor Address
DocGender	varchar(40)	Doctor Gender
DocSpec	varchar(100)	Doctor Speciality
Operation : Add New Doctor,Delete,Search, Update Doctor		

Class Name: AddCamp		
Attributes	Data Types	Descriptions
cid	varchar(50) primary key	Camp Id
cname	varchar(50)	Camp Name

cphone	varchar(10)	Camp Phone Number
cEmail	varchar(50)	Camp Email Id
cAddress	varchar(80)	Camp Address
cOrganise	varchar(40)	Camp Gender
cFacility	varchar(100)	Camp Speciality
Operation : Add New Camp,Delete,Search, Update Camp		

<b>Class Name: BReqst</b>		
<b>Attributes</b>	<b>Data Types</b>	<b>Descriptions</b>
UserId	varchar(30)	User Id
Fname	varchar(50)	User First Name
Lname	varchar(50)	User Last Name
Mobile	varchar(10)	User Phone Number
Email	varchar(50)	User Email Id
Addres	varchar(50)	User Address
Gender	varchar(50)	User Gender
BReq	varchar(30)	User Blood Request
BDate	varchar(30)	User Blood Request Date
Bstatus	varchar(40)	User Blood Status
Operation : Add New Blood Request,Delete,Search, Update Blood Request		

<b>Class Name: BloodBank</b>		
<b>Attributes</b>	<b>Data Types</b>	<b>Descriptions</b>
BBid	varchar(50) primary Key	Blood Bank Id
Bname	varchar(50)	Blood Bank Name
Bcity	varchar(50)	Blood Bank City
Baddress	varchar(80)	Blood Bank Address
Bphone	varchar(10)	Blood Bank Phone
Bemail	varchar(40)	Blood Bank Email Id
Bstatus	varchar(100)	Blood Bank Status

Operation :	Add New Blood Bank,Delete,Search, Update Blood Bank
-------------	---

<b>Class Name: CDoctor</b>		
Attributes	Data Types	Descriptions
UserId	varchar(30) primary key	User Id
Fname	varchar(50)	User First Name
Lname	varchar(50)	User Last Name
Mobile	varchar(10)	User Phone Number
Email	varchar(50)	User Email Id
Address	varchar(50)	User Address
<b>Class Name: Donor</b>		
Attributes	Data Types	Descriptions
UserId	varchar(30) primary key	User Id
Request	varchar(30) primary key	User Request
Fname	varchar(50)	User First Name
Lname	varchar(50)	User Last Name
Mobile	varchar(10)	User Phone Number
Address	varchar(50)	User Address
Gender	varchar(50)	User Gender
Grade	varchar(50)	User Grade
Name	varchar(30)	Camp Name
Purpose	varchar(30)	Purpose of Camp
Status	varchar(40)	Status of Camp
Date	varchar(20)	Camp Date
<b>Operation :</b> Add New Blood Donation,Delete,Update Blood Request		
<b>Operation :</b> Add New Host Camp,Delete, Update Host Camp		

<b>Class Name: ContactUs</b>		
Attributes	Data Types	Descriptions
Name	varchar(30)	User Name
Email	varchar(30)	User Email Id
Phone No	varchar(30)	User Phone No
Comment	varchar(30)	User Query for Any Suggestion
<b>Operation :</b> Add New Query,Get Response		

<b>Class Name: Feedback</b>		
Attributes	Data Types	Descriptions

<b>Name</b>	<b>varchar(30)</b>	<b>User Name</b>
<b>Email</b>	<b>varchar(30)</b>	<b>User Email Id</b>
<b>PhoneNo</b>	<b>varchar(30)</b>	<b>User Phone No</b>
<b>Feedback</b>	<b>varchar(30)</b>	<b>User Suggestion</b>
<b>Operation :</b>	<b>Give Feedback</b>	

## **DATABASE DESIGN**

### **SAVELIFEFOUNDATION.COM DATABASE**

create database **SaveLifeFoundation**

use SaveLifeFoundation

#### **1. create table RUser**

```
(  
  UserId int identity primary key,  
  FName varchar(50),  
  LName varchar(50),  
  DOB varchar(50),  
  Gender varchar(50),  
  Blood_Group varchar(30),  
  Email varchar(50),  
  Addres varchar(50),
```

```
Mobile varchar(10),  
Pasword varchar(20),  
);  
select * from RUser;
```

## 2. create table **BloodBank**

```
(  
BBid varchar(50) primary Key,  
Bname varchar(50),  
Bcity varchar(50),  
Baddress varchar(80),  
Bphone varchar(10),  
Bemail varchar(40),  
Bstatus varchar(100),  
);  
select * from BloodBank;
```

## 3. create table **BRequest**

```
(  
UserId varchar(30) ,  
Fname varchar(50),  
Lname varchar(50),  
Mobile varchar(10),  
Email varchar(50),  
Addres varchar(50),  
Gender varchar(50),  
BReq varchar(30),  
BDate varchar(30),  
Bstatus varchar(40),  
);  
select * from BRequest
```

## 4. create table **LAdmin**

```
(
```

```
Aname varchar(30) primary key,
```

```
Apass varchar(30),
```

```
);
```

```
select * from LAdmin
```

```
insert into LAdmin values('Ravi','admin');
```

#### 5. create table **AddDoctor**

```
(
```

```
Docid varchar(50) primary Key,
```

```
Docname varchar(50),
```

```
Docphone varchar(10),
```

```
DocEmail varchar(50),
```

```
DocAddress varchar(80),
```

```
DocGender varchar(40),
```

```
DocSpec varchar(100),
```

```
);
```

```
select * from AddDoctor;
```

#### 6. create table **AddCamp**

```
(
```

```
cid varchar(50) primary Key,
```

```
cname varchar(50),
```

```
cphone varchar(10),
```

```
cEmail varchar(50),
```

```
cAddress varchar(80),
```

```
cOrganise varchar(60),
```

```
cFacility varchar(100),
```

```
);
```

```
select * from AddCamp;
```

#### 7. create table **Donation**

```
(
```

```
UserId varchar(30) primary key,
```

```
Fname varchar(50),
```

```
Lname varchar(50),  
Mobile varchar(10),  
Email varchar(50),  
Addres varchar(50),  
Gender varchar(50),  
DReq varchar(30),  
DDate varchar(30),  
Purpose varchar(70),  
Dstatus varchar(40),  
);  
select * from Donation;
```

#### 8. create table **CDoctor**

```
(  
UserId varchar(30) primary key,  
Fname varchar(50),  
Lname varchar(50),  
Mobile varchar(10),  
Email varchar(50),  
Addres varchar(50),  
Gender varchar(50),  
Dname varchar(30),  
Dreq varchar(30),  
Disease varchar(70),  
);  
select * from CDoctor;
```

#### 9. create table **HostCamp**

```
(  
UserId varchar(30) primary key,  
Fname varchar(50),  
Lname varchar(50),  
Mobile varchar(10),  
Email varchar(50),
```

```
City varchar(50),  
Gender varchar(50),  
CName varchar(30),  
Purpose varchar(30),  
Venue varchar(70),  
CDate varchar(20),  
);  
select * from HostCamp;
```

#### 10. create table **ContactUs**

```
(  
name varchar(30),  
Email varchar(30),  
Phoneno varchar(30),  
comment varchar(30),  
);  
select *from ContactUs;
```

#### 11. create table **Feedback**

```
(  
name varchar(30),  
Email varchar(30),  
Phoneno varchar(30),  
Feedback varchar(30),  
);  
select *from Feedback;
```

## MODULES AND SUB-MODULES

- ❖ **Free Account Screen**
- ❖ **User Login**
  - **Search Blood Bank**
  - **Search Doctor**
  - **Search Camp**
  - **Consult Doctor**
  - **Blood Request**
  - **Blood Donate**
  - **Feedback**
  - **Contact Us**
- ❖ **Admin Login**
  - **Blood Bank Management**
    - **Add New Blood Bank**
    - **Update Blood Bank**
  - **Doctor Management**
    - **Add New Doctor**
    - **Update Doctor**
  - **Camp Management**
    - **Add New Camp**
    - **Update Camp**
  - **User Management**
    - **Add New User**
    - **Update User**
  - **Manage Contact Us**
  - **Manage Doctor Consult**
  - **Manage Feedback**
  - **Manage Blood Request**
  - **Manage Donation**

---

## DESCRIPTION OF SUB-MODULES WITH PROCESS LOGIC

### **HOME MODULE:**

#### **SPECIALITY:**

The Specialty page is used to check all the details about specialties such as Online Blood Request, Host Camp, Consult Doctor, Blood Donate etc.

**Process Logic:** Specialty displays the BBid primary key, Bname, Bcity, Baddress, Bemail, Bstatus informations. By this any user can find that which Blood Bank are available or unavailable.

#### **CREATE FREE ACCOUNT:**

Any user can create his/her account before using Blood Bank services.

**Process Logic:** Any user create his/her new account to fill UserId primary key, Fname, Lname, DOB, Gender, Blood\_Group, Email, Addres, Mobile, Pasword.

**LOG IN:** Already created user can login after enter UserId and Pasword.

**Process Logic:** If already created user submit correct user id and user password then he/she can take go in User dashboard.

### **ADMIN MODULE:**

#### **ADD NEW BLOOD BANK:**

This module is used to add new blood bank for any type of blood help.

**Process Logic:** This page is used to add new blood bank. It contains BBid primary key, Bname, Bcity, Baddress, Bemail, Bstatus information.

**UPDATE BLOOD BANK INFO:** In this module admin can update blood bank information.

**Process Logic:** In this admin can edit or delete any information of any blood bank.

#### **ADD NEW DOCTOR:**

This module is used to add new doctor for any doctor specialty.

**Process Logic:** This page is used to add new doctor. It contains Docid primary key, Docname, Docphone, DocEmail, DocAddress, DocGender, DocSpec, information.

**UPDATE DOCTOR INFO:** In this module admin can update doctor information.

**Process Logic:** In this admin can edit or delete any information of any doctor.

#### **ADD NEW CAMP:**

This module is used to add new camp for blood donation.

**Process Logic:** This page is used to add new camp. It contains cid primary key, cname, cphone, cEmail, cAddress, cOrganise, cFacility information.

**UPDATE CAMP INFO:** In this module admin can update camp information.

**Process Logic:** In this admin can edit or delete any information of any camp.

**ADD NEW USER:**

This module is used to add manage user for using blood bank.

**Process Logic:** This page is used to add new camp. It contain UserId primary key, Fname, Lname, DOB, Gender, Blood\_Group, Email, Addres, Mobile, Pasword.

**UPDATE CAMP INFO:** In this module admin can update camp information.

**Process Logic:** In this admin can edit or delete any information of any user.

**CONTACT US:**

This Module is used by the administrator to see the details of the contact us.

**Process Logic:** This page contains a grid view so the details of each user's query. After seeing user's query admin can solve the query.

**CONSULT DOCTOR:**

This Module is used by the administrator to see the details of the consult required by us.

**Process Logic:** This page contains a grid view so the details of each user's consult. After seeing user's consult admin can solve the query.

**FEEDBACK:**

This Module is used by the administrator to see the details of the feedback.

**Process Logic:** This page contains a grid view so the details of each user's feedback. After seeing user's feedback admin can improve the response.

**BLOOD REQUEST:**

This Module is used by the administrator to see the details of the blood request.

**Process Logic:** This page contains a grid view so the details of each user's can request of blood. After seeing user's donation admin can accept the blood request of user.

**DONATION:**

This Module is used by the administrator to see the details of the blood donation.

**Process Logic:** This page contains a grid view so the details of each user's donation. After seeing user's donation admin can accept the blood donation request of user.

**USER MODULE:****BLOOD BANK SEARCH:**

This page contains the information about blood bank.

**Process Logic:** This page contains the information of blood bank near by location.

**DOCTOR SEARCH:**

This page contains the information about doctor.

---

**Process Logic:** This page contains the information of doctor near by location.

### **CAMP SEARCH:**

This page contains the information about camp.

**Process Logic:** This page contains the information of camp and host camp near by location.

### **CONSULT DOCTOR :**

Through This module any user can consult to doctor for any query.

**Process Logic:** This page contains name, email, phoneno and disease given by any user. So any user can do any query to doctor for any problem.

### **BLOOD REQUEST :**

Through This module any user can request blood.

**Process Logic:** This page contains UserId primary key ,Fname ,Lname ,Mobile ,Email ,Addres ,Gender ,BReq ,BDate ,Bstatus given by any user. So any user can required blood for any user required

### **BLOOD DONATE :**

Through This module any user can donate blood.

**Process Logic:** This page contains UserId primary key, Fname, Email, Purpose given by any user. So any user can donate blood for any user required.

### **FEEDBACK :**

Through This module any user can give feedback to admin for any query.

**Process Logic:** This page contains name, email, Phoneno and comment given by any user. So any user can do any query to admin for any problem.

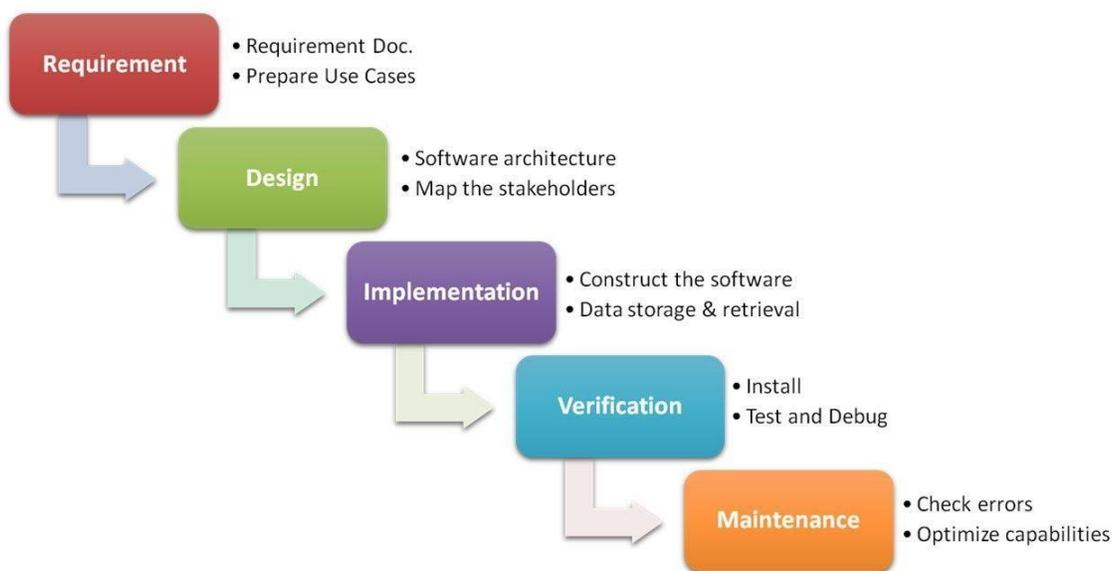
### **CONTACT US :**

Through This module any user can contact to admin for any query.

**Process Logic:** This page contains name, email, Phoneno and feedback given by any user. So any user can do any query to admin for any problem.

## **SYSTEM DEVELOPMENT LIFE CYCLE**

SDLC is an acronym for Software Development Life Cycle. It is also sometimes referred to as System Development Life Cycle. In simple words it the process, methods or a set of methodologies applied to create or alter software projects. Each of these methodologies defines unique way to create a new software module or program.



Following are some model which is used for Software or system Development Life Cycle

- Waterfall Model
- Build and Fix Model
- Spiral Model
- Prototype Model
- Interactive Enhancement Model

## FEASIBILITY STUDY

The feasibility study is carried out to test whether the proposed system is worth being implemented. Feasibility study is a test of system proposed regarding its work ability, its impact on the organization ability to meet user needs and effective use of resources. It is usually carried out by a small number of people who are familiar with the information system techniques, understand the part of the business or organization that will be involved or effected by the project and are skilled in the system analysis and design process.

The key consideration involve in the feasibility study are:

1. **Technical:** Does the company have the technological resources to undertake the project? Are the processes and procedures conducive to project success?
2. **Economic:** Given the financial resources of the company, is the project something that can be completed? The economic feasibility study is more commonly called the cost/benefit analysis.
3. **Schedule Feasibility** - Does the company currently have the time resources to undertake the project? Can the project be completed in the available time?

## SYSTEM SECURITY MEASURES

In order to have good security the following functions are presented in the project.

- User Name and password authentication is given to the system so that only valid users can access the system.

- Administrator can only create user names and password.
- Only administrator can delete important records from the system.
- Regular back up of database is provided so that if system breaks the records can be retrieved.
- A primary key and foreign key concept is implemented for avoiding incorrect data entry or intentional or accidental delete or modification of data.

## **FUTURE SCOPE OF APPLICATION**

As many business experts growth out of its operation the same stands true for computerized maintenance it has to adapt dynamically to changing structure and environment. Expansion is for betterment of organization. Expansion means addition of services to machine and new system. Automation will help efficient working and better performance. Actually the environment is taking new dimensions every today and tomorrow.

According to present scenario, the demand for the Web Portal DoctorsIndia.Com has already increased and has already proven it's worth. It is effective as time saving efforts. Quick glance and Retrieve the Any information of Doctors, Hospitals or Nursing Home any any location of India by just a single click of Mouse Button. So the information of the Doctors, Hospitals and Nursing Home in such a manner that

User can efficiently retrieve the information whenever they need or anywhere in India. With. These this site also contains the Information regarding symptoms of Disease so incase there is no any Doctor they easily recognize which type of problem they have so they can take medicine from the nearest Medicine Store and recover from the disease with minimum expense of money because they no need to contact a doctor in the case of emergency any th  
ey take First Aid when situation is very critical.

So I think there is excellent future scope of my application that will ease the working of any kind of system. It has been designed developed keeping in mind every minute requirement, which it can address in every possible way.

## BIBLIOGRAPHY

- ✓ Project Guide Booklet IGNOU
  - ✓ ASP.Net 2.0 Black Book Dreamtech Press
  - ✓ Visual C# 2012 Wrox Publication
  - ✓ Professional ASP.Net Wrox Publication
  - ✓ Introduction to Software Engineering IGNOU
  - ✓ Software Engineering *Roger S. Pressman*
  - ✓ Software Engineering Techmax Publication
  - ✓ [www.msdn.microsoft.com](http://www.msdn.microsoft.com)
  - ✓ [www.support.microsoft.com](http://www.support.microsoft.com)
  - ✓ [www.wikipedia.com](http://www.wikipedia.com)
  - ✓ [www.developer.com/net](http://www.developer.com/net)
  - ✓ [www.w3schools.com/sql](http://www.w3schools.com/sql)
- 
-

## .NET

.NET comes with a Developer environment, Visual Studio 2010, which can cope equally well with C++, C# and VISUAL BASIC .Net as well as ASP.NET code.

### ADVANTAGES OF .NET

- Object Oriented Programming
- Good Design
- Language Independent
- Better Support for dynamic pages
- Efficient Data Access
- Code Sharing
- Improved Security
- Support for Web Services

### SQL SERVER 2008

SQL Server Express is the 2008 version of SQL SERVER that replaces MSDE (Microsoft Server Data Engine). The version does not have the strict limitations MSDE had.

Due to the fact SQL Server 2008 now host the CLR, this means that you can now avoid building database aspects of your application using the T-SQL programming language. Instead, you can now build items such as your stored procedures, triggers and even data types & any of the .NET compliant languages, such as C#.

**SaveLifeFoundation.Com** part would enable users or patients gather information of particular Blood Bank or Doctor. At the end of this part of project we will able to achieve following goals.

## SYSTEM ANALYSIS

System Analysis and design refers to the process of examining a business situation with the intent of improving it through better procedures and methods. System analysis is a process of gathering and interpreting facts, diagnosis problems and using the information to recommend improvement to the

system. In brief we can say Analysis specifies what system should do. System Analysis is a management technique, which helps us in designing a new system or improving an existing system.

In this project we can have dealt with four basic elements of System Analysis in the following ways:

### **Outputs:**

In this project I have aim to achieve through my goal and what the application intends to perform. In other words, I have to determine what the objectives or goals are, what do we intend to achieve, what is the purpose of our work, i.e. what is the main aim behind the system. Defining the aim is very vital in system work.

My project can perform all the basic operations of Online Submission of User data and Registration for the Blood and Blood Bank and managing the records of the users, Camp, Doctor with the Administrator Id and Password and storing all the data in the database.

### **Inputs:**

In this project, once we know the output, we can easily determine what the inputs should be. Sometimes, it may happen that the required information may not be readily available in the proper form. This may be because of the existing forms are not properly designed. In my project, user must know all the required data such as for User need User's UserId, Fname, Lname, DOB, Gender, Blood\_Group, Email, Address, Mobile, Password, etc. for Admins need BBid, Bname, Bcity, Baddress, Bphone, Bemail, Bstatus for blood bank as the system will ask for this data when adding a record to the database.

The essential elements of inputs as far as this project is concerned are:

- Accuracy
- Timeliness
- Proper Format
- Economy

### **Process:**

In this section of the project, I have deal with the matter that how inputs are converted to outputs by the code. Basically, the data that has been feed to the system is stored in the form of tables and when the submit button is pressed the data is stored in the database and when the administrator allowed successful registration takes place. Also, the coding section can be referred to find the details about the processes by which the input gets converted into the required outputs.

## **IDENTIFICATION OF NEED**

The success of the system depends largely on how accurately a problem is defined, thoroughly investigated, and properly carried out through the choice of solution. Users need identification and analysis is concerned with what the user needs rather than what he wants. Until the problem has been identified, defined, and evaluated the analyst shouldn't think about solutions and whether the problem is worth solving or not. This step is intended to help the user and the analyst understand the real problem rather than its symptoms.

### **Information Gathering**

A key part of system development is gathering information. The analyst must know what information to get, where to find it, how to collect it, and how to make use of it. The proper use of tools for gathering information is the key to successful analysis. The information is gathered from different Blood Banks and Doctors.

Since the application to be developed is to be used on the Internet and there is no direct user available to interact so the information is gathered from the existing sites and portals as well as various print media.

## **PRELIMINARY INVESTIGATION**

The First Step in the System Development Life Cycle is the Preliminary Investigation to determine the feasibility of the system. The purpose of the preliminary investigation is to evaluate project requests. It is not a design study nor does it include the collection of details to describe the business system in all respect. Rather, it is the collecting of the information that helps committee members to evaluate the merits of the project request and make an informed judgment about the feasibility of the proposed project.

The Preliminary investigation should accomplish the following objectives.

- Clarify and understand the project request.
- Determine the size of the project.
- Assess costs and benefits of alternative approaches.
- Determine the technical and operational feasibility of alternative approaches.
- Report the findings to management, with recommendations outlining the acceptance or rejection of the proposal.

### **Conducting the Investigation**

The data that analysis collect during preliminary investigations are gathered through three primary methods, reviewing website documents, on-site observations and conducting interviews with different types of person, who are involve with it. In this project the administrative authorities of the Organization were questioned and a detailed report was developed on their requirements.

### **Reviewing Organization Documents**

The analyst conducting the investigation first learns about the organization involved in, or affected by the project. In “SAVELIFEFOUNDATION.COM”, it is necessary to know how the Registration for the users, blood bank, camp and doctors previously.

### **Onsite Observation**

Another important technique to collect data is on-site observation. The purpose of the on-site observation is to get as close as possible to the real system being studied. During on-site observation, Organization environment can be seen, workload, method of work and facilities provided by the organization to the users can be studied.

## **Conducting Interviews**

Interviews allow learning more about the nature of the project request and reasons for submitting it. Interviews should provide details that further explain the project and show whether assistance is merited economically, operationally and technically. In this project the different levels of staff of blood bank and users like patients were interviewed.

## **FEASIBILITY STUDY**

The feasibility study is carried out to test whether the proposed system is worth being implemented. Feasibility study is a test of system proposed regarding its work ability, its impact on the organization ability to meet user needs and effective use of resources. It is usually carried out by a small number of people who are familiar with the information system techniques, understand the part of the business or organization that will be involved or effected by the project and are skilled in the system analysis and design process. A feasibility study involves taking a judgment call on whether a project is doable. The two criteria to judge feasibility are cost required and value to be delivered. A well-designed study should offer a historical background of the business or project, a description of the project or service,

accounting statements, details of operations and management, marketing research and policies, financial data, legal requirements and tax obligations. Generally, such studies precede technical development and project implementation. The feasibility study is carried out to test whether the proposed system is worth being implemented.

The key consideration involve in the feasibility study are:

1. **Technical Feasibility**- Assessment is centered on the technical resources available to the organization. It helps organizations assess if the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. Technical feasibility also involves evaluation of the hardware and the software requirements of the proposed system.
2. **Economic Feasibility**- helps organizations assess the viability, cost, and benefits associated with projects before financial resources are allocated. It also serves as an independent project assessment, and enhances project credibility, as result. It helps decision-makers determine the positive economic benefits to the organizations that the proposed system will provide, and helps quantify them. This assessment typically involves a cost/benefits analysis of the project.
3. **Operational Feasibility**- this involves undertaking a study to analyze and determine whether our business needs can be fulfilled by using the proposed solution. It also measures how well the proposed system solves problems and takes advantage of the opportunities identified during scope definition. Operational feasibility studies also analyze how the project plan satisfies the requirements identified in the requirements analysis phase of system development. To ensure success, desired operational outcomes must inform and guide design and development. These include such designed-dependent parameters such as reliability, maintainability, supportability, usability, disposability, sustainability, affordability, and others.

## **TOOLS/PLATFORM, LANGUAGES TO BE USED**

Software and hardware specification as the name suggests, tells us about the various characteristics of the software and the hardware environment used i.e. the development environment used. Here we specify various software languages, supporting tools that have been used for the development of the system. These tools and the languages have been used because of their relative ease of understand and personal interest of the team developing the project.

## **HARDWARE REQUIREMENTS SPECIFICATIONS**

### **HARDWARE**

<b>PROCESSOR</b>	<b>: Intel Dual Core or Above</b>
<b>HARD DISK DRIVE</b>	<b>: 320 GB SATA or Above</b>
<b>RAM</b>	<b>: 2 GB SD DDR RAM OR Above</b>
<b>CACHE</b>	<b>: 2 MB L2 CACHE</b>

### **TOOLS**

<b>PLATFORM</b>	<b>: WINDOWS 7 or above</b>
<b>WEB TECHNOLOGIES</b>	<b>: HTML, CSS, C#, ASP.NET</b>
<b>DOCUMENTATION TOOLS</b>	<b>: MS-WORD</b>
<b>WEB SERVER</b>	<b>: IIS</b>
<b>DATABASE</b>	<b>: SQL SERVER 2008</b>

system functions and performance may be missing an initial description produced by an inexperienced user.

**A statement of the requirements for the implementation:**

## **SOFTWARE ENGINEERING PARADIGM APPLIED**

To solve actual problems in an industry setting, a software engineer or a team of engineers must incorporate a development strategy that encompasses the process, methods, and tools layers. The strategy is often referred to as a process model or a software engineering paradigm. A process model for software engineering is chosen based on the nature of the project and application, the methods and tools to be used, and the controls and deliverables that are required.

**There are numbers of paradigm available:**

- The Linear Sequential Model
- The Prototype Model
- The RAD Model
- The Incremental Model
- The Spiral Model
- The Component based development Model
- The Concurrent Development Model
- The Formal Methods Model
- The Fourth Generation Technique Model

The Model, which has been applied in my project, is the Rapid Application Development (RAD). For the same reason .NET is used as the programming tool because it provides us with the components that have already been made and we can use them as it is. This reduces the time and cost incurred on completing the project. While automating the work procedure of a “DoctosIndia.Com” no new classes were needed, hence RAD was the most suitable technique.

## **SYSTEM DESIGN**

Software Design is the first of three technical activities – Design, code generation, and test are required to build and verify the software. Each activity transforms information in manner that ultimately results in validated computer software. The design task produces a data design, an interface design and component design.

The design of an information system produces the details that clearly describe how a system will meet the requirements identified during system analysis. The system design process is not a step-by-step adherence of clear procedure and guidelines. When I started working on system design, I face different types of problems; many of these are due to constraints imposed by the user or limitation of hardware and software available. Some time it was quite difficult to likely problems is so great and no solutions are exactly similar however the following consideration I kept in mind during Design phase.

### **A. Design Objectives**

The primary objective of the design is to deliver the requirements as specified in the feasibility report. These are the some of the objective, which I kept mind.

**Practicality:** The system is quite suitable and is can operate by the people with average intelligence.

**Efficiency:** I tried to involve accuracy, timeliness and comprehensive of the system output.

**Cost:** It is desirable to aim for the system with a minimum cost subject to the condition that it must satisfy the entire requirement.

**Flexibility:** I have tried that the system should be modifiable depending on the changing needs of the user. Such modifications should not entail extensive reconstructing or recreation of software. It should also be portable to different computer.

**Security:** This is very important aspect which I followed in this designing phase and tried to covers the areas of the hardware reliability, fallback procedures and physical security of data.

### **Constraints:**

These are the following constraints:

- Hardware

- Software
- Budget
- Time Scale
- Interface with other system

### **Operations:**

Throughout the design process I considered and specify the requirements of each of these operational areas.

- User Raising Input
- Data Preparation
- Data Validation
- Processing
- Output Handling
- Action on Output

## **MODULARIZATION DETAILS**

### **HOME MODULE:**

#### **SPECIALITY:**

The Specialty page is used to check all the details about specialties such as Online Blood Request, Host Camp, Consult Doctor, Blood Donate etc.

**Process Logic:** Specialty displays the BBid primary key, Bname, Bcity, Baddress, Bemail, Bstatus informations. By this any user can find that which Blood Bank are available or unavailable.

#### **CREATE FREE ACCOUNT:**

Any user can create his/her account before using Blood Bank services.

**Process Logic:** Any user create his/her new account to fill UserId primary key, Fname, Lname, DOB, Gender, Blood\_Group, Email, Address, Mobile, Password.

**LOG IN:** Already created user can login after enter UserId and Password.

**Process Logic:** If already created user submit correct user id and user password then he/she can take go in User dashboard.

### **ADMIN MODULE:**

#### **ADD NEW BLOOD BANK:**

This module is used to add new blood bank for any type of blood help.

**Process Logic:** This page is used to add new blood bank. It contains BBid primary key, Bname, Bcity, Baddress, Bemail, Bstatus information.

**UPDATE BLOOD BANK INFO:** In this module admin can update blood bank information.

**Process Logic:** In this admin can edit or delete any information of any blood bank.

#### **ADD NEW DOCTOR:**

This module is used to add new doctor for any doctor specialty.

**Process Logic:** This page is used to add new doctor. It contains Docid primary key, Docname, Docphone, DocEmail, DocAddress, DocGender, DocSpec, information.

**UPDATE DOCTOR INFO:** In this module admin can update doctor information.

**Process Logic:** In this admin can edit or delete any information of any doctor.

#### **ADD NEW CAMP:**

This module is used to add new camp for blood donation.

**Process Logic:** This page is used to add new camp. It contains cid primary key, cname, cphone, cEmail, cAddress, cOrganise, cFacility information.

**UPDATE CAMP INFO:** In this module admin can update camp information.

**Process Logic:** In this admin can edit or delete any information of any camp.

#### **ADD NEW USER:**

This module is used to add manage user for using blood bank.

**Process Logic:** This page is used to add new camp. It contain UserId primary key, Fname, Lname, DOB, Gender, Blood\_Group, Email, Addres, Mobile, Pasword.

**UPDATE CAMP INFO:** In this module admin can update camp information.

**Process Logic:** In this admin can edit or delete any information of any user.

### **CONSULT DOCTOR:**

This Module is used by the administrator to see the details of the consult required by us.

**Process Logic:** This page contains a grid view so the details of each user's consult. After seeing user's consult admin can solve the query.

### **HOST CAMP:**

This Module is used by the administrator to see the details of the host camp required by us.

**Process Logic:** This page contains a grid view so the details of each user's hosted camp. After seeing user's hosted camp admin can solve the query.

### **BLOOD REQUEST:**

This Module is used by the administrator to see the details of the blood request.

**Process Logic:** This page contains a grid view so the details of each user's can request of blood. After seeing user's donation admin can accept the blood request of user.

### **DONATION:**

This Module is used by the administrator to see the details of the blood donation.

**Process Logic:** This page contains a grid view so the details of each user's donation. After seeing user's donation admin can accept the blood donation request of user

### **CONTACT US:**

This Module is used by the administrator to see the details of the contact us.

**Process Logic:** This page contains a grid view so the details of each user's query. After seeing user's query admin can solve the query.

### **FEEDBACK:**

This Module is used by the administrator to see the details of the feedback.

**Process Logic:** This page contains a grid view so the details of each user's feedback. After seeing user's feedback admin can improve the response.

### **USER MODULE:**

#### **BLOOD REQUEST :**

Through This module any user can request blood.

**Process Logic:** This page contains UserId primary key ,Fname ,Lname ,Mobile ,Email ,Addres ,Gender ,BReq ,BDate ,Bstatus given by any user. So any user can required blood for any user required.

#### **BLOOD DONATE :**

Through This module any user can donate blood.

**Process Logic:** This page contains UserId primary key, Fname, Email, Purpose given by any user. So any user can donate blood for any user required.

### **CONSULT DOCTOR :**

Through This module any user can consult to doctor for any query.

**Process Logic:** This page contains name, email, phoneno and disease given by any user. So any user can do any query to doctor for any problem.

### **HOST CAMP :**

Through This module any user can host the camp for the multiple purpose.

**Process Logic:** This page contains name, email, phoneno and venue given by any user. So any user can host the camp for blood donation,etc.

### **BLOOD BANK SEARCH:**

This page contains the information about blood bank.

**Process Logic:** This page contains the information of blood bank near by location.

**CAMP SEARCH:**This page contains the information about camp.

**Process Logic:** This page contains the information of camp and host camp near by location.

**DOCTOR SEARCH:**This page contains the information about doctor.

**Process Logic:** This page contains the information of doctor near by location.

### **DONOR SEARCH:**

This page contains the information about blood donor.

**Process Logic:** This page contains the information of blood donor nearby location.

### **BLOOD REQUEST STATUS:**

This page contains the information about blood request.

**Process Logic:** This page contains the status of blood requested that is request by user.

### **HOST CAMP STATUS:**

This page contains the information about host camp.

**Process Logic:** This page contains the status of host camp that is requested by user.

### **CONSULT DOCTOR STATUS:**

This page contains the information about consult with doctor.

**Process Logic:** This page contains the status of consult with doctor that is requested by user.

### **BLOOD DONOR STATUS:**

This page contains the information about blood donor.

**Process Logic:** This page contains the status of blood donor that is requested by user.

### **CONTACT US :**

Through This module any user can contact to admin for any query.

**Process Logic:** This page contains name, email, Phoneno and feedback given by any user. So any user can do any query to admin for any problem.

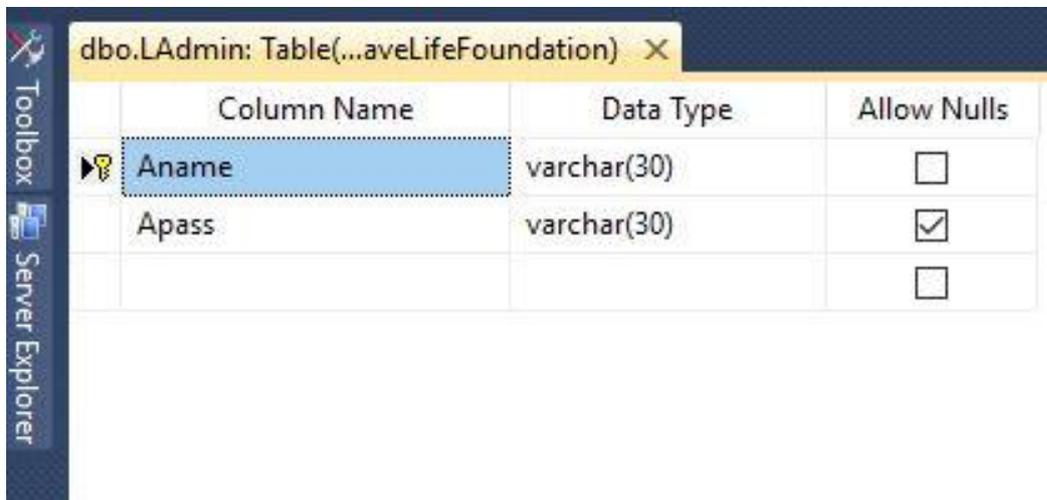
### **FEEDBACK :**

Through This module any user can give feedback to admin for any query.

**Process Logic:** This page contains name, email, Phoneno and comment given by any user. So any user can do any query to admin for any problem.

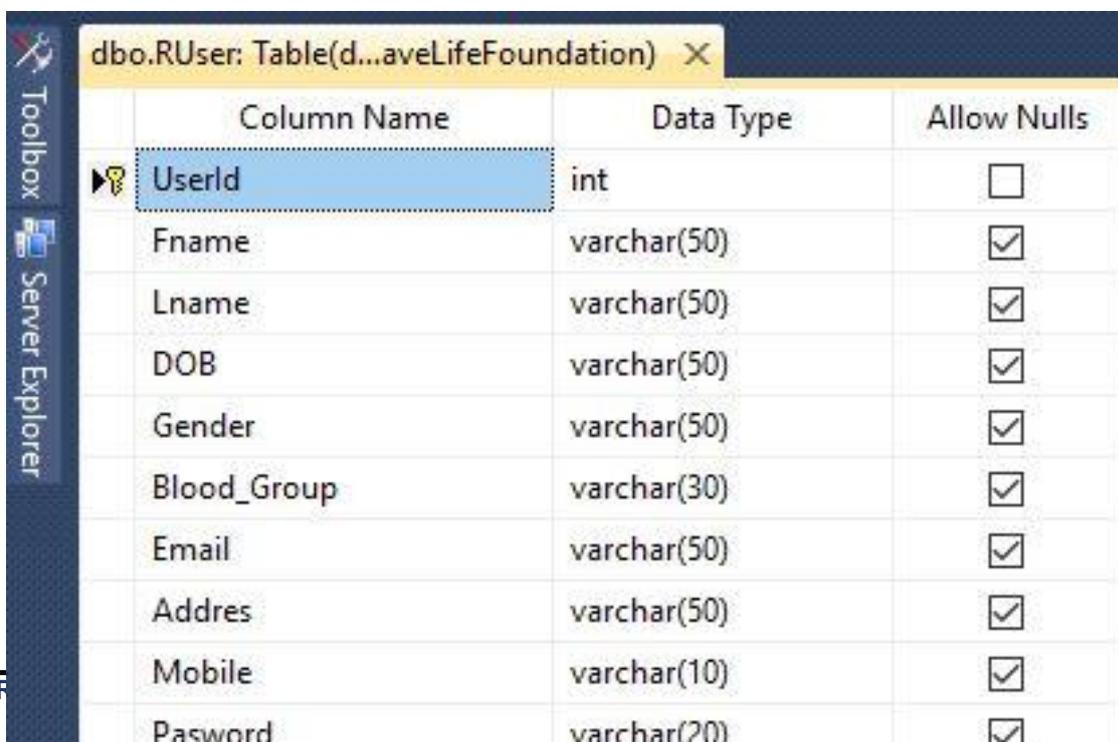
## **DATA INTEGRITY AND DATABASE DESIGN**

### **LAdmin:**



Column Name	Data Type	Allow Nulls
Aname	varchar(30)	<input type="checkbox"/>
Apass	varchar(30)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

### **RUser:**



Column Name	Data Type	Allow Nulls
UserId	int	<input type="checkbox"/>
Fname	varchar(50)	<input checked="" type="checkbox"/>
Lname	varchar(50)	<input checked="" type="checkbox"/>
DOB	varchar(50)	<input checked="" type="checkbox"/>
Gender	varchar(50)	<input checked="" type="checkbox"/>
Blood_Group	varchar(30)	<input checked="" type="checkbox"/>
Email	varchar(50)	<input checked="" type="checkbox"/>
Addres	varchar(50)	<input checked="" type="checkbox"/>
Mobile	varchar(10)	<input checked="" type="checkbox"/>
Pasword	varchar(20)	<input checked="" type="checkbox"/>

### AddCamp:

Column Name	Data Type	Allow Nulls
cid	varchar(50)	<input type="checkbox"/>
cname	varchar(50)	<input checked="" type="checkbox"/>
cphone	varchar(10)	<input checked="" type="checkbox"/>
cEmail	varchar(50)	<input checked="" type="checkbox"/>
cAddress	varchar(80)	<input checked="" type="checkbox"/>
cOrganise	varchar(60)	<input checked="" type="checkbox"/>
cFacility	varchar(100)	<input checked="" type="checkbox"/>

### AddDoctor:

Column Name	Data Type	Allow Nulls
Docid	varchar(50)	<input type="checkbox"/>
Docname	varchar(50)	<input checked="" type="checkbox"/>
Docphone	varchar(10)	<input checked="" type="checkbox"/>
DocEmail	varchar(50)	<input checked="" type="checkbox"/>
DocAddress	varchar(80)	<input checked="" type="checkbox"/>
DocGender	varchar(40)	<input checked="" type="checkbox"/>
DocSpec	varchar(100)	<input checked="" type="checkbox"/>

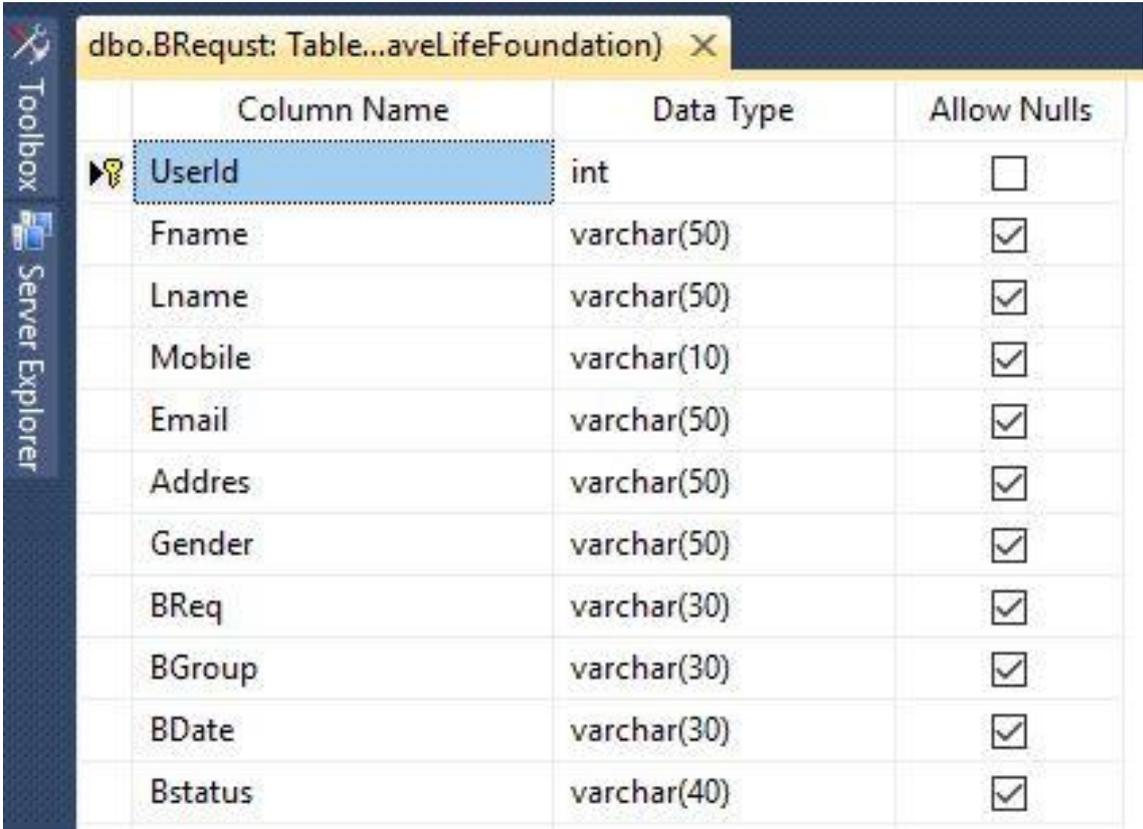
### Feedback:

Column Name	Data Type	Allow Nulls
name	varchar(30)	<input checked="" type="checkbox"/>
Email	varchar(30)	<input checked="" type="checkbox"/>
Phoneno	varchar(30)	<input checked="" type="checkbox"/>
Feedback	varchar(100)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

### BloodBank:

Column Name	Data Type	Allow Nulls
BBid	varchar(50)	<input type="checkbox"/>
BPhone	varchar(50)	<input type="checkbox"/>

## BRequest:



The screenshot shows the 'Server Explorer' pane on the left and a table structure window for 'dbo.BRequest'. The table has the following columns:

Column Name	Data Type	Allow Nulls
Userld	int	<input type="checkbox"/>
Fname	varchar(50)	<input checked="" type="checkbox"/>
Lname	varchar(50)	<input checked="" type="checkbox"/>
Mobile	varchar(10)	<input checked="" type="checkbox"/>
Email	varchar(50)	<input checked="" type="checkbox"/>
Addres	varchar(50)	<input checked="" type="checkbox"/>
Gender	varchar(50)	<input checked="" type="checkbox"/>
BReq	varchar(30)	<input checked="" type="checkbox"/>
BGroup	varchar(30)	<input checked="" type="checkbox"/>
BDate	varchar(30)	<input checked="" type="checkbox"/>
Bstatus	varchar(40)	<input checked="" type="checkbox"/>

### CDoctor:

Column Name	Data Type	Allow Nulls
UserId	int	<input type="checkbox"/>
Fname	varchar(50)	<input checked="" type="checkbox"/>
Lname	varchar(50)	<input checked="" type="checkbox"/>
Mobile	varchar(10)	<input checked="" type="checkbox"/>
Email	varchar(50)	<input checked="" type="checkbox"/>
Address	varchar(50)	<input checked="" type="checkbox"/>
Gender	varchar(50)	<input checked="" type="checkbox"/>
Dname	varchar(30)	<input checked="" type="checkbox"/>
Dreq	varchar(30)	<input checked="" type="checkbox"/>
Disease	varchar(70)	<input checked="" type="checkbox"/>
DStatus	varchar(50)	<input checked="" type="checkbox"/>

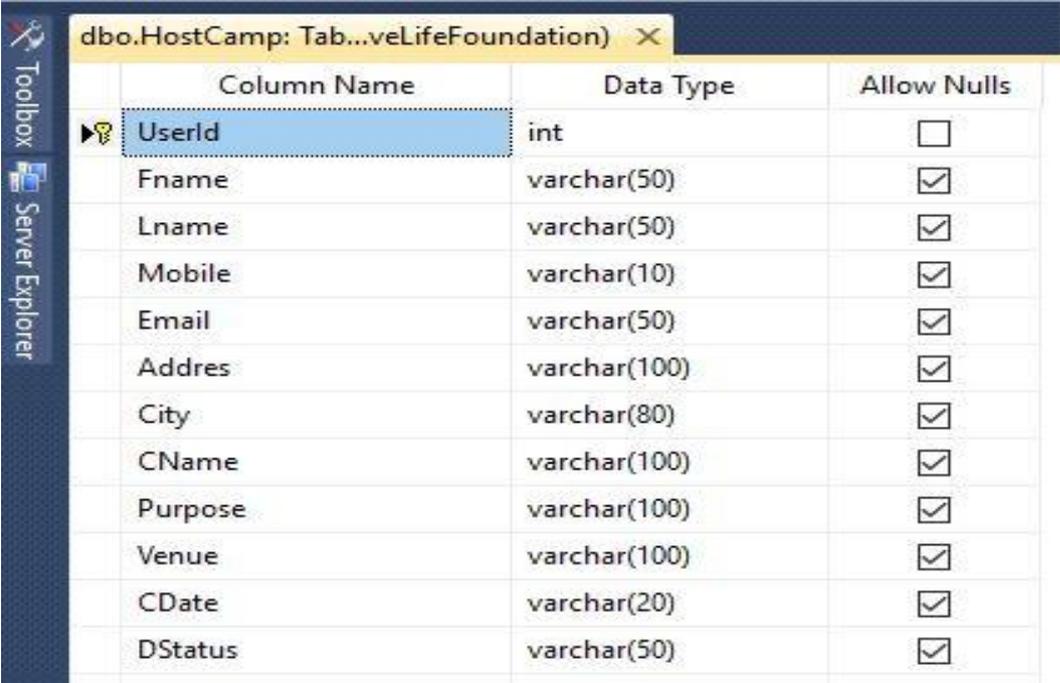
### Contact:

Column Name	Data Type	Allow Nulls
UserId	int	<input type="checkbox"/>
Name	varchar(50)	<input checked="" type="checkbox"/>
Email	varchar(50)	<input checked="" type="checkbox"/>
Mobile	varchar(10)	<input checked="" type="checkbox"/>
Query	varchar(100)	<input checked="" type="checkbox"/>
Request	varchar(100)	<input checked="" type="checkbox"/>

### Donation:

Column Name	Data Type	Allow Nulls
UserId	int	<input type="checkbox"/>

## HostCamp:



Column Name	Data Type	Allow Nulls
UserId	int	<input type="checkbox"/>
Fname	varchar(50)	<input checked="" type="checkbox"/>
Lname	varchar(50)	<input checked="" type="checkbox"/>
Mobile	varchar(10)	<input checked="" type="checkbox"/>
Email	varchar(50)	<input checked="" type="checkbox"/>
Address	varchar(100)	<input checked="" type="checkbox"/>
City	varchar(80)	<input checked="" type="checkbox"/>
CName	varchar(100)	<input checked="" type="checkbox"/>
Purpose	varchar(100)	<input checked="" type="checkbox"/>
Venue	varchar(100)	<input checked="" type="checkbox"/>
CDate	varchar(20)	<input checked="" type="checkbox"/>
DStatus	varchar(50)	<input checked="" type="checkbox"/>

## COMPLETE PROJECT CODE

### SaveLifeFoundation.Master

```

<%@ Master Language="C#" AutoEventWireup="true"
CodeBehind="Savelifefoundation.master.cs"
Inherits="Savelifefoundation.Savelifefoundation" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title>Home</title>
  <link rel="stylesheet" type="text/css" href="Style/Style.css"/>

```

```

<link href="https://fonts.googleapis.com/css?family=Lato|Open+Sans"
rel="stylesheet"/>
<script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
<link href="Style/nivo-slider.css" rel="stylesheet" type="text/css" />
<link href="Style/Default.css" rel="stylesheet" type="text/css" />
<script src="Scripts/Demo.js" type="text/javascript"></script>
<style type="text/css">

        .style2
        {
            font-size: small;
        }
</style>
</head>
<body>

<form id="form1" runat="server">
    <div id="wrapper">
        <div id="banner">
            <div class="slider-wrapper theme-default">

                <div id="nivo-slider" class="nivoSlider" style="width: 1449px;
height: 221px;" >
                    
                    
                    
                    
                </div>
            </div>
            <script type="text/javascript">

                $(window).load(function () {

                    $('#nivo-slider').nivoSlider();

                });

            </script>
        </div>
        <div id="nav">
            <ul>
                <li><a href= "Home.aspx">Home</a></li>
                <li><a href= "#">Login</a>
            </ul>
            <ul>
                <li><a href= "AdminLogin.aspx">Admin</a></li>
                <li><a href= "#">User</a>
            </ul>
            <ul>
                <li><a href= "RegisterUser.aspx">New User</a></li>
                <li><a href= "UserLogin.aspx">Registered User</a></li>
            </ul>
        </li>
        </ul>
        <li><a href= "WhyDonate.aspx">Why Donate Blood</a></li>
        <li><a href= "WhoNeeds.aspx">Who Needs Blood</a></li>
        <li><a href= "AboutUs.aspx">About Us</a></li>
        <li><a href= "#">Contact Us</a>
            <ul>
                <li><a href= "ContactUs.aspx">Ask Us</a></li>
                <li><a href= "QueryResponse.aspx">Query Response</a></li>
            </ul>

```







```

<li><a href= "AddDoctor.aspx">Add Doctors</a></li>
<li><a href= "AddCamp.aspx">Add Camp</a></li>
</ul>
</li>
<li><a href= "#">Management</a>
<ul>
<li><a href= "UserManage.aspx">User </a></li>
<li><a href= "BloodBankManage.aspx">Blood Bank </a></li>
<li><a href= "DoctorManage.aspx">Doctor</a></li>
<li><a href= "RBloodManage.aspx">Blood Request </a></li>
<li><a href= "DBloodManage.aspx">Blood Donation </a></li>
<li><a href= "CampManage.aspx">Camp </a></li>
<li><a href= "HostCampManage.aspx">Host Camp </a></li>
<li><a href= "ConsultManage.aspx">Consult Doctor</a></li>
<li><a href= "AContact.aspx">Contact Us</a></li>
</ul>
</li>
<li><a href= "Logout.aspx">Logout</a></li>
</ul>

```

```
</html>
```

## Style1.css

```

body
{
    font-family: 'Lato', sans-serif;
    font-family: 'Open Sans', sans-serif;

    background-repeat:repeat;
}

#wrapper
{
    width:auto;
    margin: 0 auto;
    padding: 10px;
    border: 3px solid #dedede;
    background-image:url('../Images/interior.jpg');
    background-color: #f1f1f1;
}

#banner
{
    text-align:justify;
    font-family:'Lucida Console';
    height: 223px;
    width:auto;

    border: 3px solid #FFFFFF;
    background-repeat:repeat-x;
    background-image:url('../Images/.jpg');
}

#nav
{
    margin:0px;
}

```

```
#nav ul
{
    padding:0px;
    list-style:none;
    height:50;
}

#nav li
{
    font-size:18px;
    float:left;
    position:relative;
    width:180px;
    padding:5px;
    height:50px;
    margin:0px;
}

#nav ul li
{
    float:left;
    height:38px;
    line-height:35px;
    text-align:center;
    width:170px;
    position:relative;
    padding:0px;
    margin:0px;
    border-collapse:collapse;
top: 0px;
left: 0px;
}

#nav ul li a
{
    color:#ffd800;
    text-decoration:none;
    font-family:Calibri;
    background:#0094ff;
    display:block;
    border-radius:8px;
    margin:1px;
}

#nav ul li a:hover
{
    background-color:Green;
    border-radius:8px;
}

#nav ul li ul
{
    display:none;
    padding:1px;
    margin:1px;
}

#nav ul li:hover >ul
```

```

        {
            clear:both;
            display:block;
            border-bottom-style:none;
        }

        #nav ul ul ul
    {
        margin-left:170px;
        top:0px;
        position:absolute;
    }

#contentarea
{
    font-family: 'Lato', sans-serif;
    font-family: 'Open Sans', sans-serif;
    float: left;
    width: 1400px;
    height:auto;
    margin: 20px 0 20px 0;
    padding:10px;
    border: 3px solid #E3E3E3;
}

#sidebar
{
    float: right;
    width: 240px;
    height: 397px;
    margin: 20px 10px 20px 10px;
    border: 3px solid #E3E3E3;
}

#footer
{
    clear: both;
    width: auto;
    height: 17px;
    margin: 88px 20px 20px 20px;
    text-align: left;
    background-color:#f1f1f1;
    color: #FFFFFF;
}

```

## **AddCamp.aspx.cs**

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;

```

```
namespace Savelifoundation
```

```
{
    public partial class AddCamp : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void Button1_Click(object sender, EventArgs e)
        {
            string scon = "trusted_connection=true; Database=SaveLifeFoundation";
            SqlConnection con = new SqlConnection(scon);
            string scom = "Insert into AddCamp
values(@cid,@cname,@cphone,@cEmail,@cAddres,@cOrganise,@cFacility)";
            SqlCommand com = new SqlCommand(scom, con);

            SqlParameter TuId = new SqlParameter();
            TuId.DbType = System.Data.DbType.String;
            TuId.ParameterName = "@cid";
            TuId.Value = TextBox1.Text;
            com.Parameters.Add(TuId);

            SqlParameter TuUname = new SqlParameter();
            TuUname.DbType = System.Data.DbType.String;
            TuUname.ParameterName = "@cname";
            TuUname.Value = TextBox2.Text;
            com.Parameters.Add(TuUname);

            SqlParameter TuMobile = new SqlParameter();
            TuMobile.DbType = System.Data.DbType.String;
            TuMobile.ParameterName = "@cphone";
            TuMobile.Value = TextBox3.Text;
            com.Parameters.Add(TuMobile);

            SqlParameter TuEmail = new SqlParameter();
            TuEmail.DbType = System.Data.DbType.String;
            TuEmail.ParameterName = "@cEmail";
            TuEmail.Value = TextBox4.Text;
            com.Parameters.Add(TuEmail);

            SqlParameter TuAddress = new SqlParameter();
            TuAddress.DbType = System.Data.DbType.String;
            TuAddress.ParameterName = "@cAddres";
            TuAddress.Value = TextBox5.Text;
            com.Parameters.Add(TuAddress);

            SqlParameter TuOrganise = new SqlParameter();
            TuOrganise.DbType = System.Data.DbType.String;
            TuOrganise.ParameterName = "@cOrganise";
            TuOrganise.Value = TextBox6.Text;
            com.Parameters.Add(TuOrganise);

            SqlParameter TuFacility = new SqlParameter();
            TuFacility.DbType = System.Data.DbType.String;
            TuFacility.ParameterName = "@cFacility";
            TuFacility.Value = TextBox7.Text;
            com.Parameters.Add(TuFacility);

            con.Open();
            com.ExecuteNonQuery();
        }
    }
}
```



```

    <asp:BoundField DataField="cEmail" HeaderText="Email Id"
        SortExpression="cEmail">
    <HeaderStyle HorizontalAlign="Center" />
</asp:BoundField>
    <asp:BoundField DataField="cAddress" HeaderText="Address"
        SortExpression="cAddress">
    <HeaderStyle HorizontalAlign="Center" />
</asp:BoundField>
    <asp:BoundField DataField="cOrganise" HeaderText="City"
        SortExpression="cOrganise">
    <HeaderStyle HorizontalAlign="Center" />
</asp:BoundField>
    <asp:BoundField DataField="cFacility" HeaderText="Facility"
        SortExpression="cFacility">
    <HeaderStyle HorizontalAlign="Center" />
</asp:BoundField>
</Columns>
<EditRowStyle BackColor="#2461BF" />
<FooterStyle BackColor="#507CD1" ForeColor="White" Font-Bold="True" />
<HeaderStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
<PagerStyle BackColor="#2461BF" ForeColor="White" HorizontalAlign="Center"
/>
    <RowStyle BackColor="#EFF3FB" />
    <SelectedRowStyle BackColor="#D1DDF1" Font-Bold="True" ForeColor="#333333"
/>
    <SortedAscendingCellStyle BackColor="#F5F7FB" />
    <SortedAscendingHeaderStyle BackColor="#6D95E1" />
    <SortedDescendingCellStyle BackColor="#E9EBEF" />
    <SortedDescendingHeaderStyle BackColor="#4870BE" />
</asp:GridView>
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConflictDetection="CompareAllValues"
    ConnectionString="<?%$ ConnectionStrings:SaveLifeFoundationConnectionString7
%>"
    DeleteCommand="DELETE FROM [AddCamp] WHERE [cid] = @original_cid AND
    (([cname] = @original_cname) OR ([cname] IS NULL AND @original_cname IS NULL)) AND
    (([cphone] = @original_cphone) OR ([cphone] IS NULL AND @original_cphone IS NULL))
    AND (([cEmail] = @original_cEmail) OR ([cEmail] IS NULL AND @original_cEmail IS
    NULL)) AND (([cAddress] = @original_cAddress) OR ([cAddress] IS NULL AND
    @original_cAddress IS NULL)) AND (([cOrganise] = @original_cOrganise) OR
    ([cOrganise] IS NULL AND @original_cOrganise IS NULL)) AND (([cFacility] =
    @original_cFacility) OR ([cFacility] IS NULL AND @original_cFacility IS NULL))"
    InsertCommand="INSERT INTO [AddCamp] ([cid], [cname], [cphone], [cEmail],
    [cAddress], [cOrganise], [cFacility]) VALUES (@cid, @cname, @cphone, @cEmail,
    @cAddress, @cOrganise, @cFacility)"
    OldValuesParameterFormatString="original_{0}"
    SelectCommand="SELECT * FROM [AddCamp]"
    UpdateCommand="UPDATE [AddCamp] SET [cname] = @cname, [cphone] = @cphone,
    [cEmail] = @cEmail, [cAddress] = @cAddress, [cOrganise] = @cOrganise, [cFacility] =
    @cFacility WHERE [cid] = @original_cid AND (([cname] = @original_cname) OR ([cname]
    IS NULL AND @original_cname IS NULL)) AND (([cphone] = @original_cphone) OR
    ([cphone] IS NULL AND @original_cphone IS NULL)) AND (([cEmail] = @original_cEmail)
    OR ([cEmail] IS NULL AND @original_cEmail IS NULL)) AND (([cAddress] =
    @original_cAddress) OR ([cAddress] IS NULL AND @original_cAddress IS NULL)) AND
    (([cOrganise] = @original_cOrganise) OR ([cOrganise] IS NULL AND @original_cOrganise
    IS NULL)) AND (([cFacility] = @original_cFacility) OR ([cFacility] IS NULL AND
    @original_cFacility IS NULL))">
    <DeleteParameters>
        <asp:Parameter Name="original_cid" Type="String" />
        <asp:Parameter Name="original_cname" Type="String" />
        <asp:Parameter Name="original_cphone" Type="String" />

```



```

    <asp:BoundField DataField="UserId" HeaderText="User Id"
InsertVisible="False"
    ReadOnly="True" SortExpression="UserId" />
    <asp:BoundField DataField="Fname" HeaderText="First Name"
    SortExpression="Fname" />
    <asp:BoundField DataField="Lname" HeaderText="Last Name"
    SortExpression="Lname" />
    <asp:BoundField DataField="Mobile" HeaderText="Mobile"
    SortExpression="Mobile" />
    <asp:BoundField DataField="Email" HeaderText="Email"
SortExpression="Email" />
    <asp:BoundField DataField="Addres" HeaderText="Address"
    SortExpression="Addres" />
    <asp:BoundField DataField="City" HeaderText="City" SortExpression="City"
/>
    <asp:BoundField DataField="CName" HeaderText="Camp Name"
    SortExpression="CName" />
    <asp:BoundField DataField="Purpose" HeaderText="Purpose"
    SortExpression="Purpose" />
    <asp:BoundField DataField="Venue" HeaderText="Venue"
SortExpression="Venue" />
    <asp:BoundField DataField="CDate" HeaderText="Date"
SortExpression="CDate" />
    <asp:BoundField DataField="DStatus" HeaderText="Status"
    SortExpression="DStatus" />
</Columns>
<EditRowStyle BackColor="#2461BF" />
<FooterStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
<HeaderStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
<PagerStyle BackColor="#2461BF" ForeColor="White" HorizontalAlign="Center"
/>
<RowStyle BackColor="#EFF3FB" />
<SelectedRowStyle BackColor="#D1DDF1" Font-Bold="True" ForeColor="#333333"
/>
<SortedAscendingCellStyle BackColor="#F5F7FB" />
<SortedAscendingHeaderStyle BackColor="#6D95E1" />
<SortedDescendingCellStyle BackColor="#E9EBEF" />
<SortedDescendingHeaderStyle BackColor="#4870BE" />
</asp:GridView>
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<%$ ConnectionStrings:SaveLifeFoundationConnectionString10
%>"
    DeleteCommand="DELETE FROM [HostCamp] WHERE [UserId] = @UserId"
    InsertCommand="INSERT INTO [HostCamp] ([Fname], [Lname], [Mobile], [Email],
[Addres], [City], [CName], [Purpose], [Venue], [CDate], [DStatus]) VALUES (@Fname,
@Lname, @Mobile, @Email, @Addres, @City, @CName, @Purpose, @Venue, @CDate,
@DStatus)"
    SelectCommand="SELECT * FROM [HostCamp]"
    UpdateCommand="UPDATE [HostCamp] SET [Fname] = @Fname, [Lname] = @Lname,
[Mobile] = @Mobile, [Email] = @Email, [Addres] = @Addres, [City] = @City, [CName] =
@CName, [Purpose] = @Purpose, [Venue] = @Venue, [CDate] = @CDate, [DStatus] =
@DStatus WHERE [UserId] = @UserId">
    <DeleteParameters>
        <asp:Parameter Name="UserId" Type="Int32" />
    </DeleteParameters>
    <InsertParameters>
        <asp:Parameter Name="Fname" Type="String" />
        <asp:Parameter Name="Lname" Type="String" />
        <asp:Parameter Name="Mobile" Type="String" />
        <asp:Parameter Name="Email" Type="String" />
        <asp:Parameter Name="Addres" Type="String" />

```



```

    <asp:CommandField ShowDeleteButton="True" ShowEditButton="True" />
    <asp:BoundField DataField="UserId" HeaderText="User Id"
InsertVisible="False"
        ReadOnly="True" SortExpression="UserId" />
    <asp:BoundField DataField="Fname" HeaderText="First Name"
        SortExpression="Fname" />
    <asp:BoundField DataField="Lname" HeaderText="Last Name"
        SortExpression="Lname" />
    <asp:BoundField DataField="Mobile" HeaderText="Mobile"
        SortExpression="Mobile" />
    <asp:BoundField DataField="Email" HeaderText="Email"
SortExpression="Email" />
    <asp:BoundField DataField="Addres" HeaderText="Address"
        SortExpression="Addres" />
    <asp:BoundField DataField="Gender" HeaderText="Gender"
        SortExpression="Gender" />
    <asp:BoundField DataField="Dname" HeaderText="Doctor Name"
        SortExpression="Doctor Name" />
    <asp:BoundField DataField="Dreq" HeaderText="Request"
SortExpression="Dreq" />
    <asp:BoundField DataField="Disease" HeaderText="Disease"
        SortExpression="Disease" />
    <asp:BoundField DataField="DStatus" HeaderText="Status"
        SortExpression="DStatus" />
</Columns>
<EditRowStyle BackColor="#2461BF" />
<FooterStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
<HeaderStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
<PagerStyle BackColor="#2461BF" ForeColor="White" HorizontalAlign="Center"
/>
    <RowStyle BackColor="#EFF3FB" />
    <SelectedRowStyle BackColor="#D1DDF1" Font-Bold="True" ForeColor="#333333"
/>
    <SortedAscendingCellStyle BackColor="#F5F7FB" />
    <SortedAscendingHeaderStyle BackColor="#6D95E1" />
    <SortedDescendingCellStyle BackColor="#E9EBEF" />
    <SortedDescendingHeaderStyle BackColor="#4870BE" />
</asp:GridView>
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<?%$ ConnectionStrings:SaveLifeFoundationConnectionString20 %>"
    DeleteCommand="DELETE FROM [CDoctor] WHERE [UserId] = @UserId"
    InsertCommand="INSERT INTO [CDoctor] ([Fname], [Lname], [Mobile], [Email],
[Addres], [Gender], [Dname], [Dreq], [Disease], [DStatus]) VALUES (@Fname, @Lname,
@Mobile, @Email, @Addres, @Gender, @Dname, @Dreq, @Disease, @DStatus)"
    SelectCommand="SELECT * FROM [CDoctor]"
    UpdateCommand="UPDATE [CDoctor] SET [Fname] = @Fname, [Lname] = @Lname, [Mobile]
= @Mobile, [Email] = @Email, [Addres] = @Addres, [Gender] = @Gender, [Dname] =
@Dname, [Dreq] = @Dreq, [Disease] = @Disease, [DStatus] = @DStatus WHERE [UserId] =
@UserId">
    <DeleteParameters>
        <asp:Parameter Name="UserId" Type="Int32" />
    </DeleteParameters>
    <InsertParameters>
        <asp:Parameter Name="Fname" Type="String" />
        <asp:Parameter Name="Lname" Type="String" />
        <asp:Parameter Name="Mobile" Type="String" />
        <asp:Parameter Name="Email" Type="String" />
        <asp:Parameter Name="Addres" Type="String" />
        <asp:Parameter Name="Gender" Type="String" />
        <asp:Parameter Name="Dname" Type="String" />
        <asp:Parameter Name="Dreq" Type="String" />

```



```

</Columns>
<EditRowStyle BackColor="#2461BF" />
<FooterStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
<HeaderStyle BackColor="#507CD1" Font-Bold="True" ForeColor="White" />
<PagerStyle BackColor="#2461BF" ForeColor="White" HorizontalAlign="Center" />
<RowStyle BackColor="#EFF3FB" />
<SelectedRowStyle BackColor="#D1DDF1" Font-Bold="True" ForeColor="#333333" />
<SortedAscendingCellStyle BackColor="#F5F7FB" />
<SortedAscendingHeaderStyle BackColor="#6D95E1" />
<SortedDescendingCellStyle BackColor="#E9EBEF" />
<SortedDescendingHeaderStyle BackColor="#4870BE" />
</asp:GridView>
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConflictDetection="CompareAllValues"
    ConnectionString="<%"$ ConnectionStrings:SaveLifeFoundationConnectionString16 %">
    DeleteCommand="DELETE FROM [contact] WHERE [UserId] = @original_UserId AND
    (([Name] = @original_Name) OR ([Name] IS NULL AND @original_Name IS NULL)) AND
    (([Email] = @original_Email) OR ([Email] IS NULL AND @original_Email IS NULL)) AND
    (([Mobile] = @original_Mobile) OR ([Mobile] IS NULL AND @original_Mobile IS NULL))
    AND (([Query] = @original_Query) OR ([Query] IS NULL AND @original_Query IS NULL))
    AND (([Request] = @original_Request) OR ([Request] IS NULL AND @original_Request IS
    NULL))"
    InsertCommand="INSERT INTO [contact] ([Name], [Email], [Mobile], [Query],
    [Request]) VALUES (@Name, @Email, @Mobile, @Query, @Request)"
    OldValuesParameterFormatString="original_{0}"
    SelectCommand="SELECT * FROM [contact]"
    UpdateCommand="UPDATE [contact] SET [Name] = @Name, [Email] = @Email, [Mobile] =
    @Mobile, [Query] = @Query, [Request] = @Request WHERE [UserId] = @original_UserId
    AND (([Name] = @original_Name) OR ([Name] IS NULL AND @original_Name IS NULL)) AND
    (([Email] = @original_Email) OR ([Email] IS NULL AND @original_Email IS NULL)) AND
    (([Mobile] = @original_Mobile) OR ([Mobile] IS NULL AND @original_Mobile IS NULL))
    AND (([Query] = @original_Query) OR ([Query] IS NULL AND @original_Query IS NULL))
    AND (([Request] = @original_Request) OR ([Request] IS NULL AND @original_Request IS
    NULL))">
    <DeleteParameters>
        <asp:Parameter Name="original_UserId" Type="Int32" />
        <asp:Parameter Name="original_Name" Type="String" />
        <asp:Parameter Name="original_Email" Type="String" />
        <asp:Parameter Name="original_Mobile" Type="String" />
        <asp:Parameter Name="original_Query" Type="String" />
        <asp:Parameter Name="original_Request" Type="String" />
    </DeleteParameters>
    <InsertParameters>
        <asp:Parameter Name="Name" Type="String" />
        <asp:Parameter Name="Email" Type="String" />
        <asp:Parameter Name="Mobile" Type="String" />
        <asp:Parameter Name="Query" Type="String" />
        <asp:Parameter Name="Request" Type="String" />
    </InsertParameters>
    <UpdateParameters>
        <asp:Parameter Name="Name" Type="String" />
        <asp:Parameter Name="Email" Type="String" />
        <asp:Parameter Name="Mobile" Type="String" />
        <asp:Parameter Name="Query" Type="String" />
        <asp:Parameter Name="Request" Type="String" />
        <asp:Parameter Name="original_UserId" Type="Int32" />
        <asp:Parameter Name="original_Name" Type="String" />
        <asp:Parameter Name="original_Email" Type="String" />
        <asp:Parameter Name="original_Mobile" Type="String" />
        <asp:Parameter Name="original_Query" Type="String" />
        <asp:Parameter Name="original_Request" Type="String" />

```



```

        {
            color: #000000;
        }
        .style2
        {
            font-size: small;
        }
    </style>
</head>
<body>
    <form id="form1" runat="server">
    <div id="wrapper">
        <div id="banner">

            <asp:Image ID="Image1" runat="server" Height="226px"
                ImageUrl="~/Images/Web-Banner-emergency-generic_0.png" style="margin-
top: 0px"
                Width="1449px" />

        </div>
        <div id="nav">
            <ul>
                <li><a href= "UserDashboard.aspx">Home</a></li>
                <li><a href= "#">Service</a>
                <ul>
                    <li><a href= "DonateBlood.aspx">Donate Blood</a></li>
                    <li><a href= "BloodBrequest.aspx">Request Blood</a></li>
                    <li><a href= "ConsultDoctor.aspx">Consult Doctor</a></li>
                    <li><a href= "HostCamp.aspx">Host Camp</a></li>
                </ul>
                </li>
                <li><a href= "#">Search</a>
                <ul>
                    <li><a href= "SBloodDonor.aspx">Blood Donor</a></li>
                    <li><a href= "SearchBlodBank.aspx">Blood Bank</a></li>
                    <li><a href= "SearchDoctor.aspx">Doctor</a></li>
                    <li><a href= "SearchCamp.aspx">Camp</a></li>
                </ul>
                </li>
                <li><a href= "#">Status</a>
                <ul>
                    <li><a href= "SBRequest.aspx">Blood Request</a></li>
                    <li><a href= "SBDoante.aspx">Blood Donate</a></li>
                    <li><a href= "SCamp.aspx">Host Camp</a></li>
                    <li><a href= "SDoctor.aspx">Consult Doctor</a></li>
                </ul>
                </li>
                <li><a href= "Editlogin.aspx">Edit Profile</a></li>
                <li><a href= "feedback.aspx">Feedback</a></li>
                <li><a href= "Logout.aspx">Logout</a></li>
            </ul>
        </div>
        <div id="contentarea">

            <asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">
                <h3>
                    <span style="font-weight: normal"></h3>
                <h3>
                    </span></asp:ContentPlaceHolder>

        </div>

```











```
        onclick="Button1_Click" />  
    <br />  
</asp:Content>
```

## **OPTIMIZATION OF CODE**

In this project I have proposed a method to enable aggressive, inter procedural optimization in a setting where code can be replaced at run-time. Code replacement involves both introducing a new module into the system and deal locating old code. Code purging deal locates replaced code, which is required in long running systems.

My approach, module merging, is simple and practical; we merge code modules and insert code to check for code replacement at the appropriate points. We show how to preserve the behavior of code purging. The net result is that merged modules preserve the original code replacement behavior, while enabling optimization across code replacement boundaries.

## VALIDATION CHECKS

At the culmination of integration testing, software is completely assembled as a package, interfacing errors have been uncovered and corrected, and a final series of software test-validation testing –may again. Validation can be defined in many ways, but a simple is that validation succeeds when software functions in a manner that can be reasonably expected by the customer.

Validation is important characteristics of a good project because many times it has been seen that running project may come to a halt stage or just because of ignorance of validation invalid or insufficient data may get enter in the database.

In this project I adopted many level of validations like in any entry form user is not supposed to leave any field blank.

As there is a form for students there are some fields which are mandatory like phone no, email id etc. and some fields are optional because it might be possible that customer does not have any middle name mean to say any unessential field which is not required in data storage.

Then in the second level of validation I check individual fields like the name cannot be numeric, phone number should not contain any alphabet, if by mistake any alphabets are entered then the message appears that phone number should be in Digits and in the same context I applied the Third level of validation that the phone number should be between 7 to 10 Digit and if user entered data in the form but not between 7 to 10 digits. Other fields have also been validated which can be found out in the coding.

We check validation in ASP.NET by using inbuilt control of Visual Web Developer:

- RequiredFieldValidator The  
RequiredFieldValidator control simply checks to see if something was entered into the HTML form element. It is a simple validation Control use to enforce a value-required rule.
- RegularExpressionValidator

Regular expressions constitute a language that can be used to find precisely defined patterns in strings. In the RegularExpressionValidator control, you define a regular expression for the pattern that is valid.

The Regular Expression Editor contains a list of commonly used regular expressions so that you can use the validator control without learning regular expression syntax.

- Range Validator

The RangeValidator control performs two functions: it ensures that the data a user enters is numeric, and it checks that the number is between the specified minimum and maximum values.

- CustomValidators.

Checks the user's entry using custom-coded validation tags.

**In our system, we enforce the following validation checks:**

- i. The user id should be unique.
- ii. UserID should have minimum 1 characters.
- iii. Password must be minimum 6 characters long.
- iv. Users can negotiate that they have successfully registered and logged in.
- v. Similarly, Doctors can register their information only when they have registered and successfully logged in.
- vi. Email address in registration must be in name@domain.xyz format.
- vii. Date of birth in registration form not more than current.
- viii. In case of online payment is made, the cheque/draft number should be numeric and should be entered compulsorily.

# **TESTING TECHNIQUES AND STRATEGIES**

It should be clear in mind that the philosophy behind is to find errors. Test cases are devised with this purpose in mind. A test case is a set of data that the system will process as normal input. There are two general strategies for testing software: Code Testing and Specification Testing. In Code Testing, the analyst develops the cases to execute every instructions and path in the program. Under specification testing, the analyst examines the program specification and then writes test data to determine how the program operates under specific conditions.

## **LEVELS OF TESTS**

### **UNIT TESTING:**

In unit testing the analyst tests the programs making up a system. For this reason, unit testing is sometimes called program testing. Unit testing gives stress on the modules independently of one another, to find errors. LifeCareDiagnosticLab.Com consists of modules to handle registration, modify or retrieve data and to respond to different types of inquiries or prints. The test cases needed for unit testing should exercise each condition and option.

### **SYSTEM TESTING:**

The important and essential part of the system development phase, after designing and developing the software is system testing. It cannot be said that every program or system design is perfect and because of lack of communication between the user and the designer, some error is there in the software development.

SYSTEM TESTING CONSISTS OF THE FOLLOWING FIVE STEPS:

- Program testing
- String testing
- System testing
- System documentation
- User Acceptance testing.

## **SPECIAL SYSTEM TESTS**

There are other 6 tests that fall under special category. They are:

**Peak load test:** It determines whether the system will handle the volume of activities that occur when the system is at the peak of its processing demand.

**Storage testing:** It determines the capacity of the system to store transaction data on a disk or in other files.

**Performance time testing:** It determines the length of time system used by the system to process transaction data.

**Recovery testing:** This testing determines the ability of user to recover data or restart system after failure.

**Procedure Testing:** It determines the clarity of documentation on operation and uses of system by having users do exactly what manuals request.

**Human Factors Testing:** It determines how the users will use the system when processing data or preparing reports.

## **IMPLEMENTATION**

A crucial in the system life cycle is the successful implementation of the new system design. Implementation includes all those activities that take place to convert from the old system to the new one.

The new system may be completely new, replacing an existing manual or automated system or it may be major modification to an existing system. In other case, proper Implementation becomes necessary so that a reliable system on the requirements of the organization can be provided.

Successful implementation may not guarantee improvement in the organization using the new system, but improper installation will prevent it. It has been observed that even the best system

cannot show good result if the analyst managing the implementation does not attend to every important detail. This is an area where the system analyst needs to work with utmost care.

The implementation discusses the aspects of implementation.

## **Training Personnel**

### **Conversation procedures**

### **Post-implementation Review**

The training of personnel involved with system. Even well designed system can succeed or fail because of the way they are operated and used. Therefore, the quality of training received by the personnel involved with the system in various capacities helps or hinders and may even prevent the successful implementation of management information system.

Those who are directly or indirectly related with the system development work must know in detail what their roles will be, how they can make efficient use of the system and what the system will or will not do for them. Both systems operations and users need training.

### **System Operations Training**

Running of the system successfully depend on the personnel working in the computer centre. They are responsible for providing the necessary support. Their training must ensure that they are to handle all possible operations, both routine and extra-ordinary in nature.

If the system calls for the installation of new equipment, such as new computer system, special terminals or different data entry machines, the operators should include such fundamentals as how to turn the equipment on and use it, how to power off and a knowledge of what constitutes normal operations. The operator should also be trained on different type of malfunctioning, how to recognize them and what steps should also be taken whenever they arise.

### **User Training**

User may be trained on use of equipment, particularly in the case where, e.g. a micro computer is in use and the individual involved is both operator and user. In such cases, user must be given training on how to operate the system also. Questions that may be trivial to the analyst, such as how to turn on a terminal, how to insert a diskette into a micro-computer or when it is safe to turn off equipment without danger of data loss are significant problems to new users who are not familiar.

In most of the cases user training deals with the operation of the system itself, with proper attention given to data handling techniques. It is imperative that users be properly trained in methods of entering transactions, editing data, formulating inquiries, deleting and inserting of records. No training is complete without familiarizing users with simple systems maintenance activities. Weakness in any aspect of training may lead to awkward situation that creates user frustration and errors.

### **Conversion Methods**

Conversion is the process of changing from the old system to the new one. It must be properly planned and executed. Four methods are common in use. They are parallel systems, Direct Conversion, Pilot system and system phase in. Each method should be considered in the light of the opportunities that it offers and problems that it may create. In general, system conversion should be accomplished in shortest possible time. Long conversion periods create for all persons involved including both analysts and users.

### **Parallel Systems**

The most secure method of converting from an old to new system is to run both systems in parallel. This method is safest one because it ensures that in case of any problem in using new system, the organization can still fall back to the old system without the loss of time and money.

### **The Disadvantages of parallel systems approach are :**

It doubles operating costs.

The new system may not get fair trial.

Direct Conversion.

This method converts from the old to the new system abruptly, sometimes over a weekend or even overnight. The old system is used until a planned conversion day, when it is replaced by the new system.

### **Pilot System**

Pilot approach is often preferred in the case of the new system, which involves new techniques or some drastic changes in the organization performance. In this method, a working version of the system is implementing in one part of the organization, such as a single work area or department.

### **Phase-In-Method**

This method is used when it is not possible to install a new system throughout an organization all at once. The conversion of files, training of personal or arrival of equipment may force the staging of the implementation over a period of time, ranging from weeks to months.

## **Post Implementation Review**

After the system is implementing and conversion is complete, a review should be conducted to determine whether the system is meeting expectations and where improvements are needed. A post implementation review measures the systems performance against predefined requirements. It determines how well the system continues to meet the performance specifications.

## **Evaluation**

Much of the management is decision making, according to one many approaches to management. While there are several views of constitutes management, according to the decision-oriented view, management mainly comprises the following:

- **Planning**
- **Organizing**
- **Directing**
- **Control**

Each one of these functions may be at the strategic, tactical or operational level. To illustrate this point we will use a series of examples of strategic, tactical and operational decision and the information needs, in each of these functional areas.

## **Planning**

Strategic level planning would call a lot of environment information like shifting markets, changing technology as well as internal information like core-competitive strength of the organization.

Tactical planning activities like vendor development make or buy decisions would call for cost and availability information pertaining to materials, production capacities both internal to the organization as well as outside. Operational planning like staff scheduling would need large amounts of internal information like schedules, attendance, up-times of equipment.

## **Organizing**

Strategic organizing would need external and internal data to decide on re-structuring as well as forge strategic partnerships. Tactical organizing would need changing wage level data of both the organization as well as that of competitors. Operational organizing would need data relating to skills and training requirements of the operational staff.

## **Co-ordination**

Strategic coordination would call for industry wide data corresponding to technology availability. Tactical coordination would call for planet wide and supplier wise bottleneck data, which reflects the deficiencies both inside the organization and outside. Operational coordination would require itemized break up of plant and machinery performance, failures etc.

### **Directing**

Strategic directing functions like introduction of office automation would call for detailed cost benefit analysis of new techniques. Tactical directing like innovating marketing strategy would call for detailed market and production data. Operational directing function would need data pertaining to the individual manager's detailed skills.

### **Control**

Strategic control decisions like total quality management would need detailed performance data and bench marketing data from outside the organization. Tactical control decisions like maintaining steady market share in the medium run would necessitate continuous monitoring of planet data. Operational control may call for techniques of statistical process control that involves the collection of substantial sampling information that must be collected and processed continuously during the entire production period.

In essence each and every area of managerial decision making would be it planning, organizing, co-ordinating, directing or control, it calls for substantial amounts of information processing. While these functions of management need information support for decision making there are subtle differences between the decisions that can significantly benefit from information equally there are decisions that are unlikely to be benefited substantially by the context was very ably pointed out by the pioneering decision.

Programmed decisions are those that can easily be automated, like the determination of optimal product, minimum cost production schedule, optimal sequencing of machines to minimize mean flow time etc. generally such decisions are characterized by large data and a few decision rule or algorithms that use the data in an automated fashion to arrive at an optimal plan.

Techniques of O-R like linear programming represent a typical example of this category of decisions that use formal data and algorithms. Naturally such decisions are easily programmed. In other words they can be represented as algorithmic procedures into ambiguous instructions whose step by step execution will lead the optimal result.

Since these algorithms are likely to be codified in the form of a computer program and run on a digital computer, they are programmable or programmed decisions. The key to such

programmability underlying structure of these decisions situation those permit an algorithmic translation. By no means, it is intended that such program decisions are unimportant, trivial or simple.

There are no value judgments to such programmed decision either. Many of the programmed decisions may need the most challenging algorithms involving the best brains available at the moment for their solution. Nevertheless they are translatable into algorithmic procedures. Information support for such program can be designed rather easily.

## **DATABASE SECURITY AND ACCESS RIGHTS**

Any software is susceptible to some security threat or the other. These threats can be mainly classified as accidental intentional or malicious security threats. Following measures have been taken to ensure unauthorized users are not able to access the site:

- 1) Only registered users are allowed to negotiate their details. They should firstly login for this purpose.
- 2) If somebody tries to access the admin login and fails for more than 5 times, an alarm mails sent to the administrator with the login details.
- 3) Any users cannot view or access the personal information of any other user. This is implemented by using session variables for tracking the users.
- 4) Also, a separate login page on the site for the system administrator is implemented so that the admin jobs can be done through the site itself, instead of accessing the database directly.
- 5) Any User can create personal id and password for submit his information.
- 6) A detail of any Blood Bank is maintained properly that which blood bank is active or passive.

## **COST ESTIMATION OF THE PROJECT**

Unlike traditional engineering disciplines where one has to budget for material the “raw material” used in software is mainly the engineer’s brainpower.

Thus the cost of a software project is directly proportional to the number of engineers need for the project. The problem of predicting how many engineers and other resources are needed for a given software project is known as software cost estimation.

Forecasting how many engineers will be needed is a difficult problem that is intimately tied to the problem of how to estimate the productivity of software engineers.

There are two parts to the forecasting problem.

- Estimating the difficulty of the task
- Estimating how many tasks each engineer can solve.

Clearly, to estimate difficulty of the task, one knows what the task is - i.e., what the requirements are. But, it is often difficult to specify the software requirements completely.

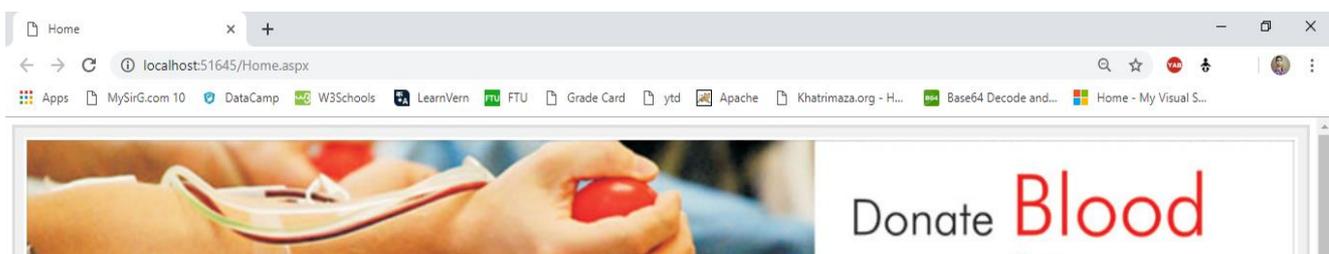
It was precisely such difficulties that motivated me to look at the evolutionary process as a alternative to the traditional waterfall model.

Incomplete and imprecise requirements hinder accurate cost estimation. The clearer and more complete the requirements, the easier it is to determine the resources required. But even when the requirements are clearer, estimating the number of engineers needed is a formidable task with inherits difficulties.

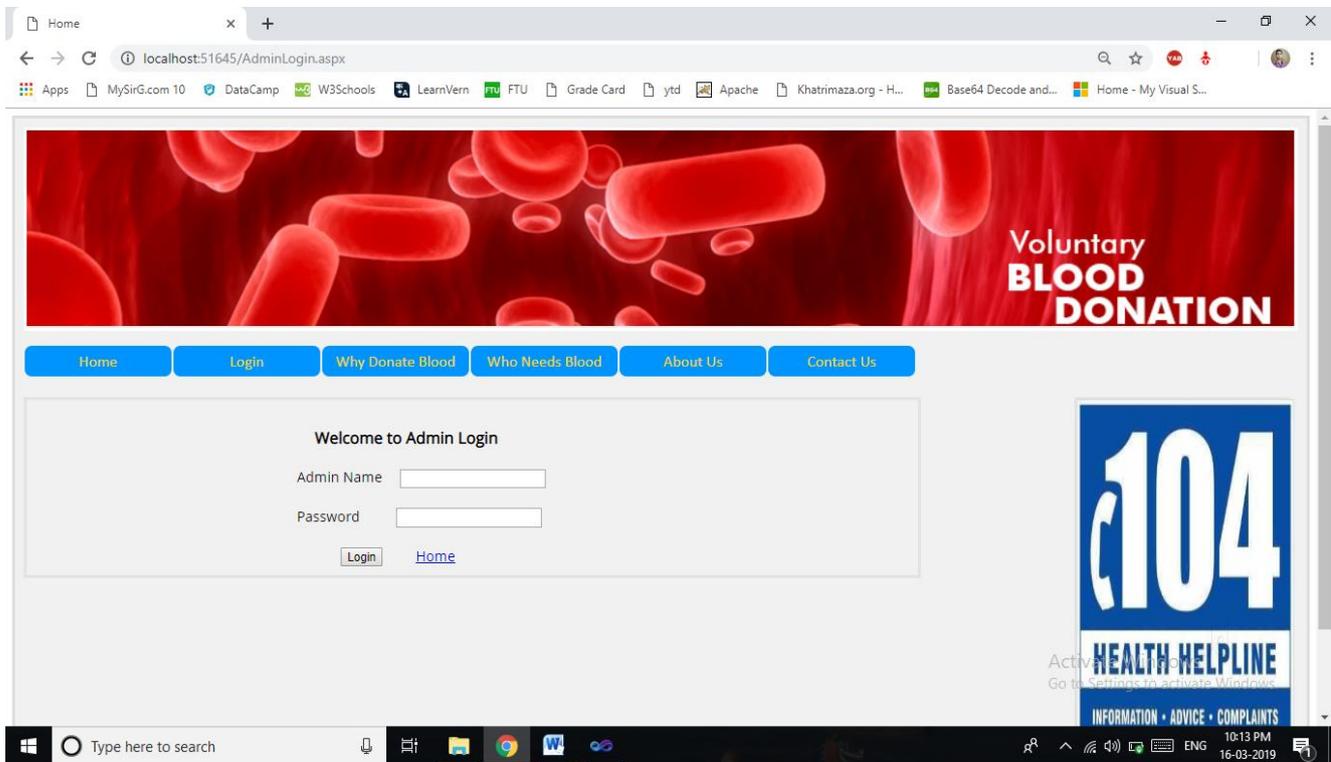
The best approach is to develop the resource requirements incrementally, revising current estimates, as more information becomes available.

## **OUTPUT SCREENS**

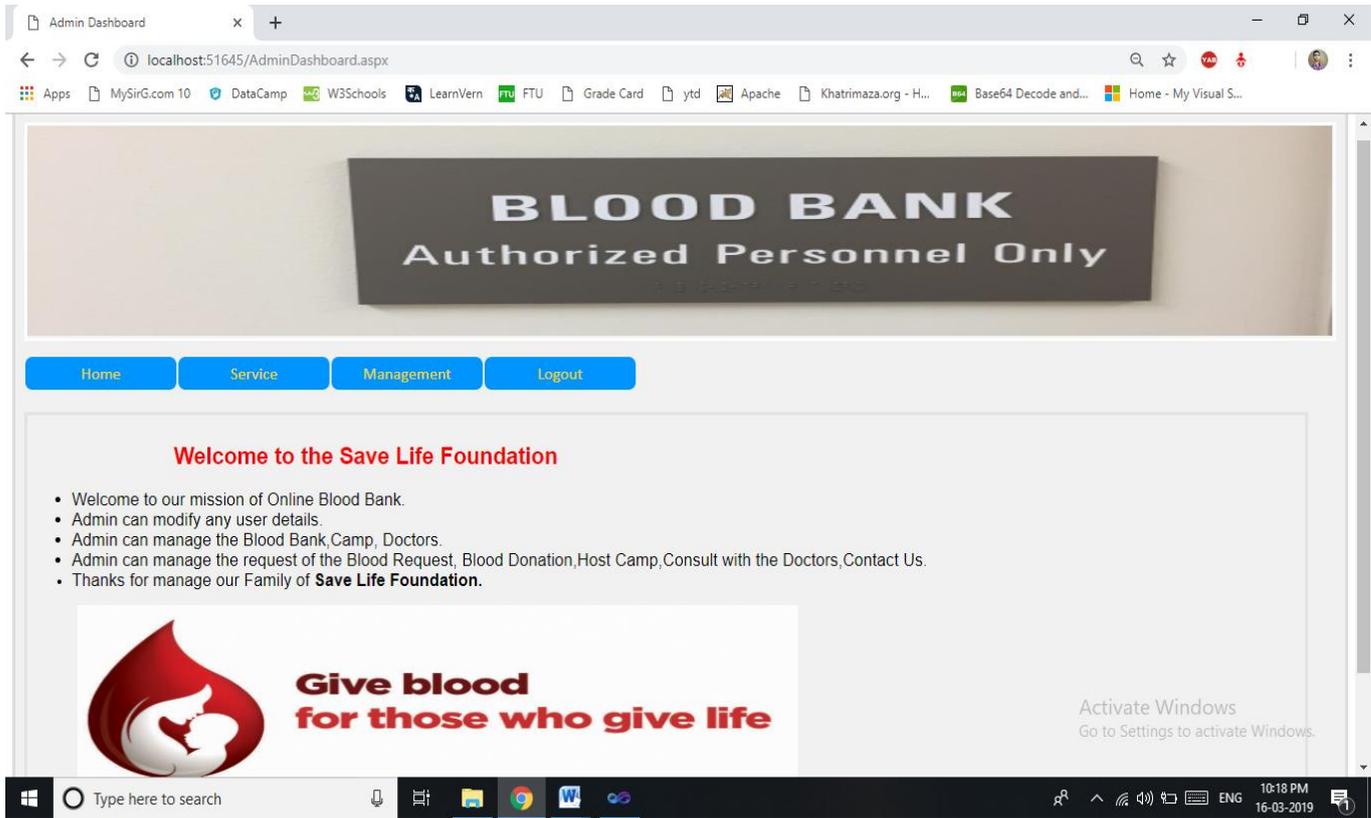
### **Home Screen-**



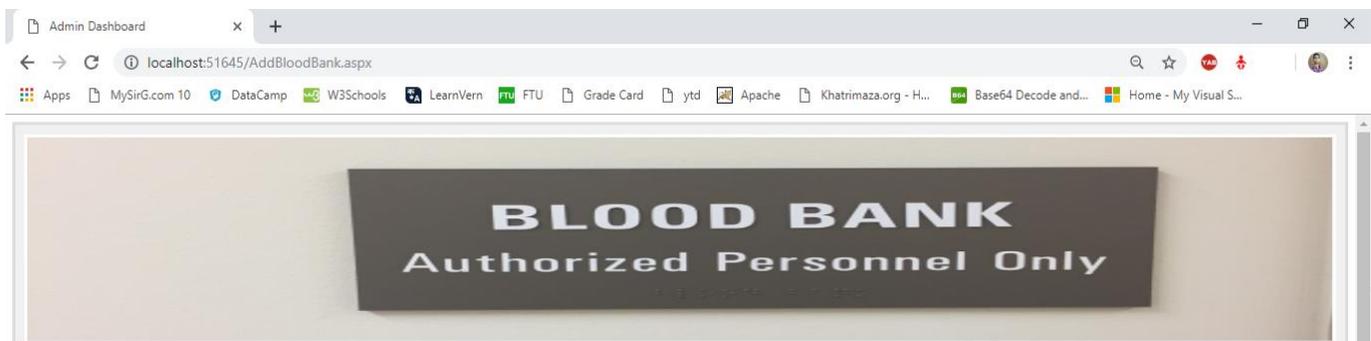
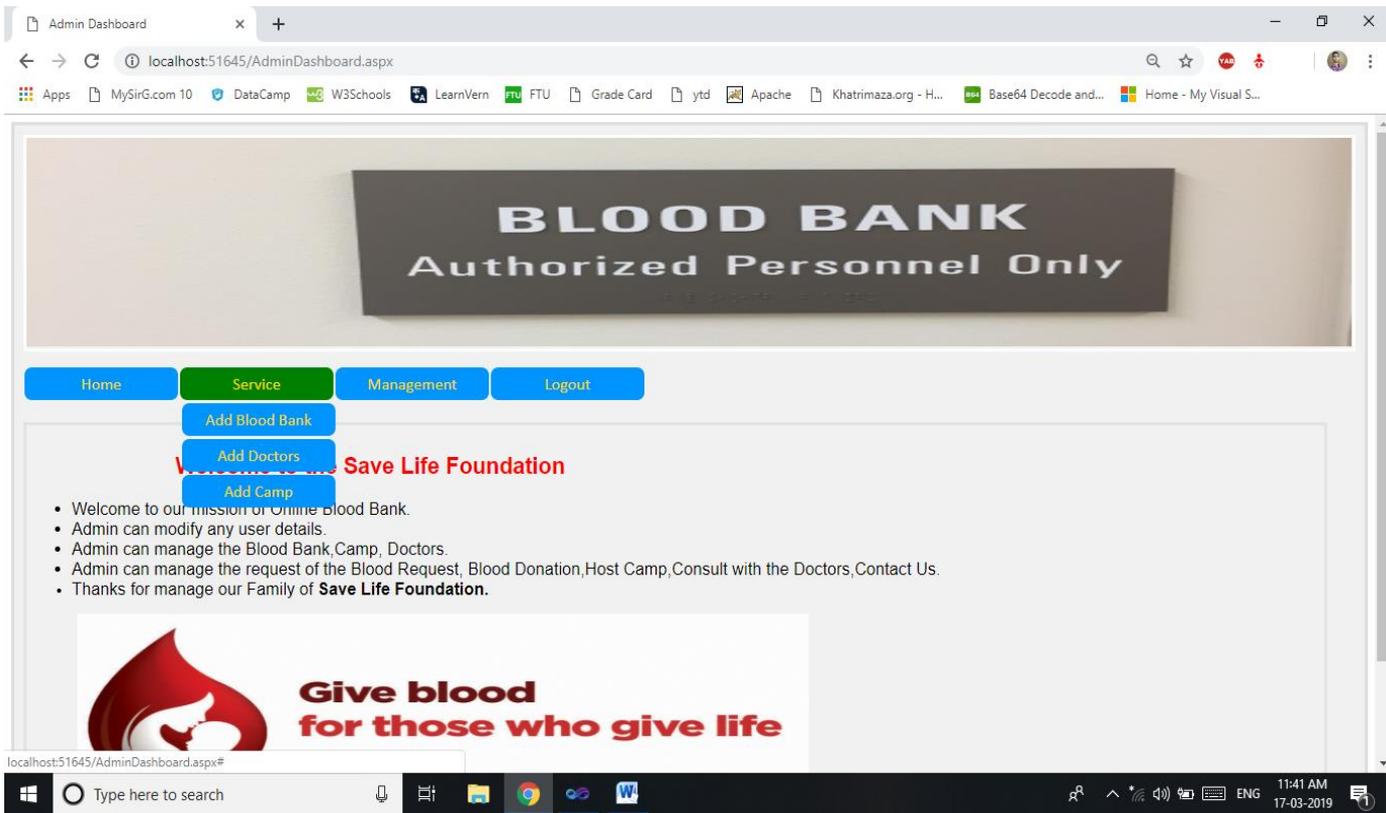
## Admin Login-



# AdminDashboard-



# Services-



## Add Doctor-

Admin Dashboard

localhost:51645/AddDoctor.aspx

Home Service Management Logout

### Add New Doctors

Doctor Id

Doctor Name

Mobile

Email Id

Address

Gender

Speciality

Activate Windows  
Go to Settings to activate Windows.

Type here to search

10:21 PM  
16-03-2019

## Add Camp-

Admin Dashboard

localhost:51645/AddCamp.aspx

Home Service Management Logout

### New Camp Registration

Camp Id

Camp Name

Mobile No

Email Id

Address

City

Activate Windows  
Go to Settings to activate Windows.

Type here to search

10:21 PM  
16-03-2019

# Management-

Admin Dashboard

localhost:51645/AdminDashboard.aspx

Home Service **Management** Logout

- User
- Blood Bank
- Doctor
- Blood Request
- Blood Donation
- Camp
- Host Camp
- Consult Doctor
- Contact Us

Welcome to the Blood Bank Foundation

- Welcome to our mission of Online Blood Bank
- Admin can modify any user details.
- Admin can manage the Blood Bank, Consult Doctors
- Admin can manage the request of the Blood Donation, Host Camp, Consult with the Doctors, Contact Us.
- Thanks for manage our Family of Sa

Give Blood for those who give life

11:41 AM 17-03-2019

# User Manage-

Welcome to Manage Users

	User Id	First Name	Last Name	DOB	Gender	Blood Group	Email Id	Address	Mobile	Password
<a href="#">Edit</a> <a href="#">Delete</a>	1	Ravi	kumar	20/07/1995	Male	A Negative	ravikushwah1175@gmail.com	Sangam Vihar	9999449258	1234
<a href="#">Edit</a> <a href="#">Delete</a>	2	Raju	singh	20/06/1994	Male	A Positive	Raju@gmail.com	Sangam Vihar	9999449254	1234
<a href="#">Edit</a> <a href="#">Delete</a>	3	mohan	kumar	04/06/1997	Male	B Positive	mohankumar123@gmail.com	Chirag Enclave, New Delhi-119972.	8800646385	1234
<a href="#">Edit</a> <a href="#">Delete</a>	4	Sanjay	Kumar	03/02/1996	Male	B Negative	sanjay121@yahoo.com	B-32, Nehru Place, New Delhi-110092.	9999449288	1234
<a href="#">Edit</a> <a href="#">Delete</a>	5	Klran	Kumari	06/04/2002	Female	O Positive	kiran123@gmail.com	c-14, Greater Kailash, New Delhi-119202.	9494949499	1234
<a href="#">Edit</a> <a href="#">Delete</a>	6	Sonam	Singh	06/4/1992	Female	O Negative	Sonam1234@gmail.com	Sangam Vihar, Dehi-110062	9999559258	1234
<a href="#">Edit</a> <a href="#">Delete</a>	7	Raju	singh	02/08/1999	Male	AB Positive	Raju1975@gmail.com	G-21, Nehru Place, New Delhi-110053.	9999443333	1234
<a href="#">Edit</a> <a href="#">Delete</a>	8	Pawan	kumar	06/08/1998	Male	AB Negative	pawan123@gmail.com	kalkaji, DDA Flat, New Delhi-110063.	9999449256	1234
<a href="#">Edit</a> <a href="#">Delete</a>	9	Sanjay	kumar	14-07-1998	Male	A Positive	Sanjaykumar@gmail.com	C-23, Malviya Nagar, New Delhi-110080	9999449268	1234

# Blood Bank Manage-

Welcome to Manage Blood Bank

	Bank Id	Name	City	Address	Phone No	Email Id	Status
<a href="#">Edit</a> <a href="#">Delete</a>	1	RBS Blood Bank	Delhi	C-14, Devli, New Delhi-62	8800646385	rbsblood@gmail.com	Available
<a href="#">Edit</a> <a href="#">Delete</a>	10	Health care	Noida	F-43, Sector-3, Noida, Up-32221,	6363634502	healthcare32@gmail.com	Available
<a href="#">Edit</a> <a href="#">Delete</a>	11	Ayur Healthcare	Gurgaon	D-32, Sector-23, Gurgaon, Haryana-23213.	8528521230	ayurhealthcare23@gmail.com	Available
<a href="#">Edit</a> <a href="#">Delete</a>	12	Biswas Healthcare	Gurgaon	C-23, Sector-33, Gurgaon, Haryana-23231.	8522254566	biswashealthcare@gmail.com	Available
<a href="#">Edit</a> <a href="#">Delete</a>	2	Indian Youth Foundation	Delhi	B-14, Khanpur, Delhi-110062.	9990427096	indianyouth12@yahoo.com	Available
<a href="#">Edit</a> <a href="#">Delete</a>	3	Sankalp	Delhi	A-14, Badarpur, New Delhi-110072.	8800646385	Sankalp@yahoo.com	Available
<a href="#">Edit</a> <a href="#">Delete</a>	4	Save Life Foundation	Delhi	C-1175, Sangam Vihar, New Delhi-110062	9999449258	ravikushwah1175@gmail.com	Available
<a href="#">Edit</a> <a href="#">Delete</a>	5	Holy Foundation	Noida	B-95, Sector-3, Noida-4021.	9990427098	HolyFoundation@holy.org	Available
<a href="#">Edit</a> <a href="#">Delete</a>	6	Madan Charitable	Noida	A-23, Golf City, Noida-4632.	8800646384	Madancharitable@gmail.com	Available
<a href="#">Edit</a> <a href="#">Delete</a>	7	KMB Foundation	Gurgaon	B-32, Sector-4, Gurgaon, Haryana	9990427780	kmb_12@gmail.com	Available
<a href="#">Edit</a> <a href="#">Delete</a>	8	Dr Lal Charitable Trust	Delhi	B-43, Mohan Estate, New Delhi-110082.	9990427088	drlal@gmail.com	Available

## Doctor Manage-

Admin Dashboard

localhost:51645/DoctorManage.aspx

Home Service Management Logout

Welcome to Manage Doctors

Doctor Id	Name	Phone No	Email	Address	Gender	Speciality
<a href="#">Edit</a> <a href="#">Delete</a> 1	Rajesh Kumar	9999444452	rajeshkumar121@gmail.com	C-13,Khanpur,New Delhi-110062.	Male	Children
<a href="#">Edit</a> <a href="#">Delete</a> 2	Karan Kumar	8888444465	karankumar12@gmail.com	B-212,Neb Sarai,New Delhi-110082.	Male	Adults
<a href="#">Edit</a> <a href="#">Delete</a> 3	Ravi Kumar	9999444442	ravikushwah15@gmail.com	C-32,Kalkaji,New Delhi-110080	Male	Adults
<a href="#">Edit</a> <a href="#">Delete</a> 4	Raju	9999444488	raju1175@gmail.com	D-32,Mehrauli,New Delhi-110080.	Male	Children

Activate Windows  
Go to Settings to activate Windows.

## Blood Request Manage-

Admin Dashboard

localhost:51645/RBloodManage.aspx

Home Service Management Logout

Welcome to Manage Blood Request

User Id	First Name	Last Name	Mobile	Email	Address	Gender	Request	Group	Date	Status
<a href="#">Edit</a> <a href="#">Delete</a> 1	Ravi	Singh	9999444444	ravikumar1213@gmail.com	C-32,Sangam Vihar,New Delhi-110062.	Male	Yes	A Positive	28/02/2019	Accepted
<a href="#">Edit</a> <a href="#">Delete</a> 2	Ravi	kumar	9898454512	ravikushwah1175@gmail.com	C-32,Sangam Vihar,New Delhi-110062.	Male	Yes	A Positive	28/02/2019	Rejected

Activate Windows  
Go to Settings to activate Windows.

# Blood Donation Manage-

**BLOOD BANK**  
Authorized Personnel Only

Home Service Management Logout

Welcome to Manage Blood Donation

User Id	First Name	Last Name	Mobile	Email	Address	Gender	Request	Date	Purpose	Blood Group	Status
<a href="#">Edit</a> <a href="#">Delete</a> 1	Ravi	kumar	9999449252	ravikumar123@gmail.com	C-23,Sangam Vihar,New Delhi-110062.	Male	Yes	28/02/2019	For Brother	A Negative	Approved
<a href="#">Edit</a> <a href="#">Delete</a> 2	Ravi	kumar	9999449258	ravikushwah1175@gmail.com	C-32,Sangam Vihar,New Delhi-110062.	Male	Yes	28/02/2019	For Humanity	A Positive	Rejected
<a href="#">Edit</a> <a href="#">Delete</a> 4	Raju	Kumar	9999449256	raju123@gmail.com	C-23,Sangam Vihar,New Delhi-110062.	Male	Yes	07/02/2019	For Humanity	A Negative	Approved

Activate Windows  
Go to Settings to activate Windows.

# Camp Manage-

**BLOOD BANK**  
Authorized Personnel Only

Home Service Management Logout

Welcome to Manage Camp

Camp Id	Name	Phone No	Email Id	Address	City	Facility
<a href="#">Edit</a> <a href="#">Delete</a> 1	Sankalp Blood Donation	9990427096	Sankalp@gmail.com	B-32,Lodhi Colony	New Delhi	Blood Donation
<a href="#">Edit</a> <a href="#">Delete</a> 2	Youth Era	8885554442	youthera23@gmail.com	C-32,Sector-3	Noida	Blood Donation Awareness
<a href="#">Edit</a> <a href="#">Delete</a> 3	Meri Awaz	9696964525	meriawaz@gmail.com	C-232,Gupta Estate	Gurgaon	Blood Donation

Activate Windows  
Go to Settings to activate Windows.

## Contact Us Manage-

**BLOOD BANK**  
Authorized Personnel Only

Home Service Management Logout

Welcome to Manage Contact Us

Userid	Name	Email	Mobile	Query	Reply
<a href="#">Edit</a> <a href="#">Delete</a> 1	Ravi	ravikushwah1175@gmail.com	9944558855	I will update you	OK
<a href="#">Edit</a> <a href="#">Delete</a> 2	Raju	raju123@gmail.com	8800654565	Your Helpline Number	9999449258

Activate Windows  
Go to Settings to activate Windows.

## Consult Doctor Manage-

**BLOOD BANK**  
Authorized Personnel Only

Home Service Management Logout

Welcome to Manage Consult Doctors

User Id	First Name	Last Name	Mobile	Email	Address	Gender	Doctor Name	Request	Disease	Status
<a href="#">Edit</a> <a href="#">Delete</a> 1	Ravi	kumar	9990427096	ravikushwah1175@gmail.com	C-23,Sangam Vihar,New Delhi-110062.	Male	Ravi Kumar	28/02/19	Dengue	Accepted
<a href="#">Edit</a> <a href="#">Delete</a> 2	Ravi	kumar	9999444444	ravikushwah1175@gmail.com	C-23,Sangam Vihar,New Delhi-110062.	Male	Rajesh Kumar	28/02/2019	Sugar	Rejected

Activate Windows  
Go to Settings to activate Windows.

# Host Camp Manage-

**BLOOD BANK**  
Authorized Personnel Only

Home Service Management Logout

Welcome to Manage Host Camp

User Id	First Name	Last Name	Mobile	Email	Address	City	Camp Name	Purpose	Venue	Date	Status
<a href="#">Edit</a> <a href="#">Delete</a> 2	Ravi	kumar	9999434321	ravikushwah1175@gmail.com	C-32,Sangam Vihar,New Delhi-110062.	Delhi	Sankalp Trust	Blood Donation	C-14,Devli,New Delhi-62	28/02/2019	Approved

Activate Windows  
Go to Settings to activate Windows.

# User Login-

**Blood Bank**  
Overview

Home Login Why Donate Blood Who Needs Blood About Us Contact Us

Admin User New User Registered User

Welcome to Save Life Foundation

Save Life Foundation is an ISO 9001:2008 certified blood bank established in 2019, and is the country's largest and most modern blood bank with state-of-art technologies and service.

Our Blood Bank is committed to provide high quality services by assuring the supply of safe and quality blood and its products to the patients. The blood bank adopts most modern technology and equipment. We've fully automatic ELISA processor, blood grouping and cross matching and blood component equipments. All the records and standard operative procedures (SOPs) are maintained and followed as per National and International standards.

Our Blood Bank aims to meet the requirement of precious blood by organizing voluntary blood donation camps at education institutes, corporate offices and many other organizations and gatherings.

**104**  
HEALTH HELPLINE  
INFORMATION • ADVICE • COMPLAINTS

## New User-

Home Login Why Donate Blood Who Needs Blood About Us Contact Us

**Welcome to User Registration**

User Id

First Name

Last Name

DOB

Gender

Blood Group

Email

Address

Mobile

Password

[Home](#)

**104**  
HEALTH HELPLINE  
INFORMATION • ADVICE • COMPLAINTS

Type here to search 11:05 AM 17-03-2019

## Registered User Login-

Home Login Why Donate Blood Who Needs Blood About Us Contact Us

**Welcome to User Login**

Username

Password

[Sign Up](#)

**104**  
HEALTH HELPLINE  
INFORMATION • ADVICE • COMPLAINTS

Type here to search 11:06 AM 17-03-2019

# User Dashboard-

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

**Thank you to our Volunteers and Blood Donors!**

- Welcome to our member of our mission of Online Blood Bank. Our Blood Bank Initiative is done by **Save Life Foundation**.
- User can Donate the Blood, Request the Blood, Consult with Doctor and Host the Camp for Blood Donation.
- User can search the Blood Bank, Blood Donor, Camp, Doctors.
- User can check the status of the Blood Bank, Blood Donor, Camp, Doctors.
- Thanks for Being a part of our **Save Life Foundation**.

**World Blood Donor Day**  
"Celebrating Your Gift of Blood" 14 JUNE

**IT'S BETTER TO BE A MEMBER**

**Eligibility**

- ... should be normal and oral temperature should not exceed 37.5 degree Celsius.

# Services-

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home **Service** Search Status Edit Profile Feedback Logout

Donate Blood  
Request Blood  
Consult Doctor  
Host Camp

**Thank you to our Volunteers and Blood Donors!**

- Welcome to our member of our mission of Online Blood Bank. Our Blood Bank Initiative is done by **Save Life Foundation**.
- User can Donate the Blood, Request the Blood, Consult with Doctor and Host the Camp for Blood Donation.
- User can search the Blood Bank, Blood Donor, Camp, Doctors.
- User can check the status of the Blood Bank, Blood Donor, Camp, Doctors.
- Thanks for Being a part of our **Save Life Foundation**.

**World Blood Donor Day**  
"Celebrating Your Gift of Blood" 14 JUNE

**IT'S BETTER TO BE A MEMBER**

**Eligibility**

- ... should be normal and oral temperature should not exceed 37.5 degree Celsius.

# Donate Blood -

The screenshot shows a web browser window with the URL localhost:51645/DonateBlood.aspx. The page features a navigation bar with buttons for Home, Service, Search, Status, Edit Profile, Feedback, and Logout. The main content area is titled "Welcome to Blood Donate" and contains a registration form with the following fields: User Id (pre-filled with 5), First Name, Last Name, Mobile, Email, Address, Gender (dropdown), Blood Request (dropdown), Request Date (calendar icon), Purpose, Blood Group (dropdown), and Blood Status (dropdown with a note "[Status Shown By Admin]"). A "Donate" button is located at the bottom of the form. On the right side, there is a red banner that says "IT'S BETTER TO BE A MEMBER" and an "Eligibility" section with the following text: "A donor can again donate blood after 3 months of your last donation of blood. Pulse rate must be between 50 to 100mm". The Windows taskbar at the bottom shows the time as 11:10 AM on 17-03-2019.

# Blood Request-

The screenshot shows a web browser window with the URL localhost:51645/BloodBrequst.aspx. The page features a navigation bar with buttons for Home, Service, Search, Status, Edit Profile, Feedback, and Logout. The main content area is titled "Welcome to Blood Request" and contains a registration form with the following fields: User Id (pre-filled with 3), First Name, Last Name, Mobile, Email, Address, Gender (dropdown), Blood Request (dropdown), Blood Group (dropdown), Request Date (calendar icon), and Blood Status (text input with a note "Status Shown By Admin"). A "Request" button is located at the bottom of the form. On the right side, there is a red banner that says "IT'S BETTER TO BE A MEMBER" and an "Eligibility" section with the following text: "Donor must be 18-60 years age and having a minimum weight of 50Kg can donate blood. Donor's Hemoglobin level is 12.5%". The Windows taskbar at the bottom shows the time as 11:11 AM on 17-03-2019.

# Consult Doctor -

User Dashboard x +

localhost:51645/ConsultDoctor.aspx

Home Service Search Status Edit Profile Feedback Logout

### Welcome to Consult Doctor

User Id

First Name

Last Name

Mobile

Email

Address

Gender

Doctor Name

Request Date

Disease

Status  [Show By Admin]

Consult

IT'S BETTER TO BE A MEMBER

#### Eligibility

- minimum weight of 50Kg can donate blood.
- Donor's Hemoglobin level is 12.5% minimum.
- A donor can again donate blood after 3

© Copyright 2018 SaveLifeFoundation. All rights reserved.

Type here to search

11:12 AM 17-03-2019

# Host Camp-

User Dashboard x +

localhost:51645/HostCamp.aspx

Home Service Search Status Edit Profile Feedback Logout

### Welcome to Host Camp

User Id

First Name

Last Name

Mobile

Email

Address

City

Camp Name

Purpose

Venue

Date

Status  [Shown By Admin]

Host

IT'S BETTER TO BE A MEMBER

#### Eligibility

- Donor must be 18-60 years age and having a minimum weight of 50Kg can donate blood.
- Donor's Hemoglobin level is 12.5% minimum.

Type here to search

11:13 AM 17-03-2019

# Search -

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service **Search** Status Edit Profile Feedback Logout

Blood Donor  
Blood Bank  
Doctor  
Camp

**Thank you to our members and Blood Donors!**

- Welcome to our member of our mission. Our Blood Bank Initiative is done by **Save Life Foundation**.
- User can Donate the Blood, Request the Blood, Consult with Doctor and Host the Camp for Blood Donation.
- User can search the Blood Bank, Blood Donor, Camp, Doctors.
- User can check the status of the Blood Bank, Blood Donor, Camp, Doctors.
- Thanks for Being a part of our **Save Life Foundation**.

**World Blood Donor Day**  
"Celebrating Your Gift of Blood" 14 JUNE

**IT'S BETTER TO BE A MEMBER**

**Eligibility**

- Donor's Hemoglobin level is 12.5% minimum.
- A donor can again donate blood after 3 months of your last donation of blood.

# Blood Donor-

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

**Welcome to Search Blood Donor**

Blood Group:

User Id	Name	Address	Phone	Email
2	Ravi	C-32, Sangam Vihar, New Delhi-110062.	9999449258	9999449258

**IT'S BETTER TO BE A MEMBER**

**Eligibility**

years age and having a minimum weight of 50Kg can donate blood.

- Donor's Hemoglobin level is 12.5% minimum.

# Blood Bank -

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

Welcome to Search Blood Bank

City:

Bank Id	Bank Name	Address	Phone	Email
10	Health care	F-43, Sector-3, Noida, Up-32221,	6363634502	healthcare32@gmail.com
5	Holy Foundation	B-95, Sector-3, Noida-4021.	9990427098	HolyFoundation@holy.org
6	Madan Charitable	A-23, Golf City, Noida-4632.	8800646384	Madancharitable@gmail.com
9	Life Care	B-34, Pari Chowk, Noida, UP-32321.	8888445564	lifecare231@gmail.com

**IT'S BETTER TO BE A MEMBER**

**Eligibility**  
 Healthy, and not suffering from any transmittable diseases can donate blood.  
 • Donor must be 18-60 years age and having a minimum weight of

# Doctor-

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

Welcome to Search Doctor

Speciality:

Doctor Id	Name	Address	Phone	Email
2	Karan Kumar	B-212, Neb Sarai, New Delhi-110082.	8888444465	karankumar12@gmail.com
3	Ravi Kumar	C-32, Kalkaji, New Delhi-110080	9999444442	ravikushwah15@gmail.com

**IT'S BETTER TO BE A MEMBER**

**Eligibility**  
 years age and having a minimum weight of 50Kg can donate blood.  
 • Donor's Hemoglobin level is 12.5% minimum.

# Camp -

localhost:51645/SearchCamp.aspx

Blood stocks are critical. PLEASE DONATE BLOOD!

Home Service Search Status Edit Profile Feedback Logout

Welcome to Search Camp

City: New Delhi

Search

Camp Id	Camp Name	Address	Phone	Email	Facility
1	Sankalp Blood Donation	B-32, Lodhi Colony	9990427096	Sankalp@gmail.com	Blood Donation

IT'S BETTER TO BE A MEMBER

**Eligibility**  
Healthy, fit and not suffering from any transmittable diseases can donate blood.

- Donor must be 18-60 years age and having a minimum weight of

# Status-

localhost:51645/UserDashboard.aspx

Blood stocks are critical. PLEASE DONATE BLOOD!

Home Service Search Status Edit Profile Feedback Logout

Thank you to our Volunteers and Donors!

- Welcome to our member of our mission of Online Blood Bank Initiative is done by **Save Life Foundation**.
- User can Donate the Blood, Request the Blood, Consult with Doctor and Host the Camp for Blood Donation.
- User can search the Blood Bank, Blood Donor, Camp, Doctors.
- User can check the status of the Blood Bank, Blood Donor, Camp, Doctors.
- Thanks for Being a part of our **Save Life Foundation**.

**World Blood Donor Day**  
"Celebrating Your Gift of Blood" 14 JUNE

IT'S BETTER TO BE A MEMBER

**Eligibility**  
18-60 years age and having a minimum weight of 50Kg can donate blood.

- Donor's Hemoglobin level is 12.5% minimum.
- A donor can again

# Blood Request -

User Dashboard

localhost:51645/SBRequest.aspx

Apps MySirG.com 10 DataCamp W3Schools LearnVern FTU Grade Card ytd Apache Khatrimaza.org - H... Base64 Decode and... Home - My Visual S...

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

Welcome to Search Status of Blood Request

User Id  Search

Name	Mobile	Blood Group	Status	Date
Ravi	9999444444	A Positive	Accepted	28/02/2019

**IT'S BETTER TO BE A MEMBER**

**Eligibility**  
suffering from any transmittable diseases can donate blood.

- Donor must be 18-60 years age and having a minimum weight of 50Kg can donate blood

Type here to search

11:23 AM 17-03-2019

# Blood Donor-

User Dashboard

localhost:51645/SBDoante.aspx

Apps MySirG.com 10 DataCamp W3Schools LearnVern FTU Grade Card ytd Apache Khatrimaza.org - H... Base64 Decode and... Home - My Visual S...

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

Welcome to Search Status of Blood Donate

User Id  Search

Name	Mobile	Blood Group	Status	Date
Ravi	9999449258	A Positive	Rejected	28/02/2019

**IT'S BETTER TO BE A MEMBER**

**Eligibility**  
years age and having a minimum weight of 50Kg can donate blood.

- Donor's Hemoglobin level is 12.5% minimum.
- A donor can again

Type here to search

11:24 AM 17-03-2019

# Host Camp -

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

Welcome to Search Status of Host Camp

User Id

Name	Mobile	Camp Name	Date	Venue	Status
Ravi	9999434321	Sankalp Trust	28/02/2019	C-14,Devli,New Delhi-62	Approved

**IT'S BETTER TO BE A MEMBER**

**Eligibility**

- A donor can again donate blood after 3 months of your last donation of blood.
- Pulse rate must be

# Consult Doctor-

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

Welcome to Search Status of Consult Doctor

User Id

Name	Mobile	Doctor Name	Date	Disease	Status
Ravi	9990427096	Ravi Kumar	28/02/19	Dengue	Accepted

**IT'S BETTER TO BE A MEMBER**

**Eligibility**

- Donor must be 18-60 years age and having a minimum weight of 50Kg can donate blood.

# Login to Edit Profile -

User Dashboard

localhost:51645/Editlogin.aspx

Apps MySirG.com 10 DataCamp W3Schools LearnVern FTU FTU Grade Card ytd Apache Khatrimaza.org - H... Base64 Decode and... Home - My Visual S...

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

**Login To Edit Profile**

User Id

Password

Login To Edit

**IT'S BETTER TO BE A MEMBER**

**Eligibility**

- Donor's Hemoglobin level is 12.5% minimum.
- A donor can again donate blood after 3 months of your last

Type here to search

11:27 AM 17-03-2019

# Update Profile-

User Dashboard

localhost:51645/UpdateProfile.aspx

Apps MySirG.com 10 DataCamp W3Schools LearnVern FTU FTU Grade Card ytd Apache Khatrimaza.org - H... Base64 Decode and... Home - My Visual S...

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

**Welcome to Update Your Profile**

User Id

First Name

Last Name

DOB

Gender

Blood Group

Email

Address

Mobile

Password

Update [Home](#)

**IT'S BETTER TO BE A MEMBER**

**Eligibility**

should be normal and oral temperature should not exceed 37.5 degree Celsius.

© Copyright 2018 SaveLifeFoundation. All rights reserved.

Type here to search

11:28 AM 17-03-2019

## Feedback -

**Blood stocks are critical. PLEASE DONATE BLOOD!**

Home Service Search Status Edit Profile Feedback Logout

**Feedback**

Name

Email

Phone No

Feedback

Feedback

**IT'S BETTER TO BE A MEMBER**

**Eligibility**  
 healthy, fit and not suffering from any transmittable diseases can donate blood.

- Donor must be 18-60 years age and having a

## Why Donate Blood-

Home Login Why Donate Blood Who Needs Blood About Us Contact Us

**DONATION**

Blood is the living fluid that all life is based on. Blood is composed of 60% liquid part and 40% solid part. The liquid part called Plasma, made up of 90% water and 10% nutrients, hormones, etc. is easily replenished by food, medicines, etc. But the solid part that contains RBC (red blood cells), WBC (white blood cells) and Platelets take valuable time to be replaced if lost.

This is where you come in. The time taken by a patient's body to replace it could cost his/her life. Sometimes the body might not be in a condition to replace it at all.

As you know blood cannot be harvested it can only be donated. This means only you can save a life that needs blood.

Every year India requires 40 million units of 250cc blood out of which only a meager 500,000 of blood units are available.

Saving a life does not require heroic deeds. You could just do it with a small thought and an even smaller effort by saying "yes".

For all of these reasons and more, donating blood is a good idea. To find out more about where to donate blood locally, use [SaveLifeFoundation.Com](http://SaveLifeFoundation.Com)

**104**  
**HEALTH HELPLINE**  
 INFORMATION • ADVICE • COMPLAINTS

Copyright 2018 SaveLifeFoundation. All rights reserved.

# Who Needs Blood-

Home Login Why Donate Blood Who Needs Blood About Us Contact Us

Every 2 seconds someone needs blood. Your blood helps more than one life at a time. Accident victims, premature babies, patients undergoing major surgeries require whole blood, where your blood after testing is used directly. Patients suffering from trauma, anemia, and other surgeries require only red blood cells, which is separated from your blood. The procedure of splitting blood components is called Cytapheresis. Similarly blood platelets are used for cancer patients undergoing chemotherapy or for those undergoing treatment for dengue fever etc. Fresh frozen plasma is used for patients having massive transfusions, plasma is used for burns and cryoprecipitate is used for hemophilia.

Blood is needed at regular intervals and at all times as it has only finite time of storage. Red blood cells can be stored for about 42 days, fresh frozen plasma and cryoprecipitate for 365 days and blood platelets for 5 days.

Anyone above 18 years weighing more than 50 kgs (110 lbs) can donate blood.

104 HEALTH HELPLINE  
INFORMATION • ADVICE • COMPLAINTS

© Copyright 2018 SaveLifeFoundation. All rights reserved.

# About Us-

Home Login Why Donate Blood Who Needs Blood About Us Contact Us

Donate Blood Save Lives

'Blood Bank India' is the first product resulted out of the community welfare initiative called 'SAVELIFEFOUNDATION.COM' Project. Universally, 'Blood' is recognized as the most precious element that sustains life. It saves innumerable lives across the world in a variety of conditions. Once in every 2- seconds, someone, somewhere is desperately in need of blood. More than 29 million units of blood components are transfused every year. The need for blood is great - on any given day, approximately 39,000 units of Red Blood Cells are needed. Each year, we could meet only up to 1% (approx) of our nation's demand for blood transfusion.

Despite the increase in the number of donors, blood remains in short supply during emergencies, mainly attributed to the lack of information and accessibility. We positively believe this tool can overcome most of these challenges by effectively connecting the blood donors with the blood recipients.

104 HEALTH HELPLINE  
INFORMATION • ADVICE • COMPLAINTS

# Contact Us -

Home Login Why Donate Blood Who Needs Blood About Us Contact Us

**Welcome to Contact Us**

ID

Name

Email

Phone No

Query

Request

Contact Us

104 HEALTH HELPLINE  
INFORMATION • ADVICE • COMPLAINTS

Copyright 2018 SaveLifeFoundation. All rights reserved.

# Check Contact Us-

Home Login Why Donate Blood Who Needs Blood About Us Contact Us

**Welcome to Check Contact Us**

Mobile No

Search

104 HEALTH HELPLINE  
INFORMATION • ADVICE • COMPLAINTS

## **FUTURE SCOPE OF APPLICATION**

As many business expert's growth out of its operation the same stands true for computerized maintenance it has to adapt dynamically to changing structure and environment. Expansion is for betterment of organization. Expansion means addition of services to machine and new system. Automation will help efficient working and better performance. Actually the environment is taking new dimensions every today and tomorrow.

According to present scenario, the demand for the Web Portal **SaveLifeFoundation.Com** has already increased and has already proven it's worth. It is effective as time saving efforts. Quick glance and Retrieve the any information of Blood Bank, Doctor and Camps of any location of India by just a single click of Mouse Button. So the information of the Blood Bank, Doctor and Camps in such a manner that User can efficiently retrieve the information whenever they need or anywhere in India. With these, this site also contains the information regarding the importance of blood so in case there is no any information of blood they can easily recognize which type of problem they have so they can take information from the nearest Blood Bank and recover from the disease with minimum expense of money because they no need to contact a Admin in the case of emergency any they need to go to the nearest blood bank when situation is very critical.

So I think there is excellent future scope of my application that will ease the working of any kind of system. It has been designed developed keeping in mind every minute requirement, which it can address in every possible way.

## BIBLIOGRAPHY

- ✓ Project Guide Booklet IGNOU
- ✓ ASP.Net 2.0 Black Book Dreamtech Press
- ✓ Professional ASP.Net Wrox Publication
- ✓ Introduction to Software Engineering IGNOU
- ✓ Software Engineering Roger S. Pressman
- ✓ Software Engineering Techmax Publication
- ✓ www.msdn.microsoft.com
- ✓ www.support.microsoft.com
- ✓ www.wikipedia.com
- ✓ www.developer.com/net
- ✓ www.w3schools.com/sql
- ✓ www.tutorialspoint.com
- ✓ www.simplilearn.com
- ✓ www.c-sharpcorner.com