

BLOOD BANK MANAGEMENT SYSTEM

SOFTWARE REQUIREMENTS SPECIFICATION

VERSION 1.0

DATE: 22/04/2019



GROUP NO-19

NAME	ROLL NO
KUMAR SHUBHAM	127
PRERANA CHAKRABORTY	122
NILADRI PAUL	123
KAZI NAZMUL HAQUE	124
KUNAL KUMAR	125
SUBHANKAR SAHA	126
AMITABH SUMAN	121

Contents

[BLOOD BANK MANAGEMENT SYSTEM]	1-2
1. Product Description	3
1.1 Purpose	3
1.2 Scope	3
1.3 Assumptions	3
1.4 Constraints	3
2. Functional Requirements	4
3. Non-Functional Requirements	4
3.1 Performance Requirements	4
3.2 Safety Requirements	5
3.3 Security Requirements	5
4. Interface Requirements	5
4.1 User Interfaces(Front End)	6-10
4.2 User Interfaces(Back End)	11
4.3 Hardware Requirement	11
4.4 Software Requirement	12
5. Diagrams	12
5.1 Use Case Diagram	12
5.2 ER Diagram	13
5.3 Data Flow Diagram	14-16
6. Summary & Conclusion	17
7. Future Recommendation	17
6. Glossary	18
8. Bibliography	19

1. Project Description

1.1 Purpose

The BLOOD BANK MANAGEMENT SYSTEM provide blood donation service to the city. Blood Bank Management System (BBMS) is a browser based system that is designed to store, process, retrieve and analyze information concerned with the administrative and inventory management within a blood bank. This project aims at maintaining all the information pertaining to blood donors, different blood groups available in each blood bank and help them manage in a better way. Aim is to provide transparency in this field, make the process of obtaining blood from a blood bank hassle free and corruption free and make the system of blood bank management effective.

1.2 Scope

No other area of the clinical laboratory has more paper work associated with its individual services than does the Blood Bank. As of now, most of the work in many Private Blood Banks are done manually. Manual system has many limitations. So this product will have a great scope and a substantially large target audience. Also, the system can be extended to be used for maintaining the records of the hospital, organ donation and other similar fields.

1.3 Assumptions

- The end user should have a basic knowledge of English and computer usage.

1.4 Constraints

- System works smoothly for single servers
- Users should have basic knowledge of computer.
- Only English is used in GUI.
- Internet Connection is needed.

2.Functional Requirement

2.1 Donor Management

- The system allows data collection from donor .
- Information about the donor is stored in database with image.
- All donor related information are viewable in tabular form.
- The system provides easy edit or adding details for various sections of the donor form.

2.2 Request Management

- Captures patient personal information as well as the hospital where blood is required.
- The system allows for reserving a unit for 24 hours for a patient
- The blood component issued, the payment made as well as link to the final bill is available when the patient page is opened.

2.3 Search Management

- Provides filterable selections for donor selections
- Excel view of all donors and requesters.
- The reports are highly configurable and can be configured to display data as per requestor or donor requirements.
- The results displayed in search is highly configurable , Custom links can be added in the search results to allow easier navigation and accessibility.

3.Non-Functional Requirements

3.1 Performance Requirements

- The software is expected to have reasonably short response time.
- It should be able to log-in and feed the records related to camp, inventory, client, partner etc. and give us accurate records.

3.2 Safety Requirements

- The system should be capable of gracefully recovering from earlier crashes and continuing the blood bank process.

3.3 Security Requirements

- The system shall provide unique id to all the employees so that they can do their own work without intervening into another manger's work.
- Serial attacks will be avoided by maintaining a minimum time gap between successive invalid log-in attempts.

4. Interface Requirements

4.1. User Interfaces (Front End)

The front end is an interface between the user and the back end. The front and back ends may be distributed amongst one or more systems.

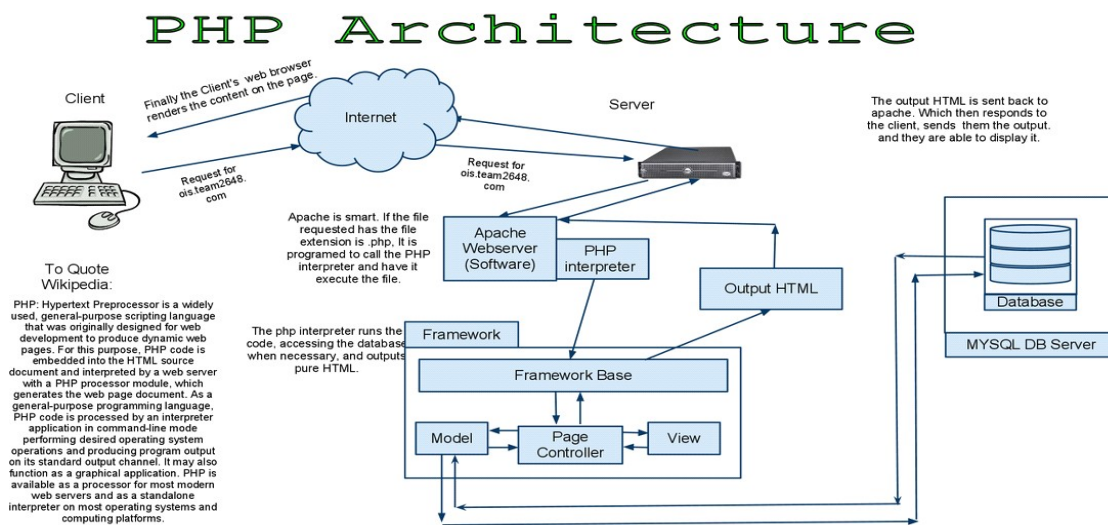
In network computing, *front end* can refer to any hardware that optimizes or protects network traffic. It is called application front-end hardware because it is placed on the network's outward-facing front end or boundary. Network traffic passes through the front-end hardware before entering the network.

In compilers, the front end translates a computer programming source code into an intermediate representation, and the back end works with the intermediate representation to produce code in a computer output language. The back end usually optimizes to produce code that runs faster. The front-end/back-end distinction can separate the parser section that deals with source code and the back end that generates code and optimizes .

These days, front-end development refers to the part of the web users interact with. In the past, web development consisted of people who worked with Photoshop and those who could code HTML and CSS. Now, developers need a handle of programs like Photoshop and be able to code not only in HTML and CSS, but also JavaScript or jQuery, which is a compiled library of JavaScript.

Most of everything you see on any website is a mixture of HTML, CSS, and JavaScript, which are all controlled by the browser. For example, if you're using Google Chrome or Firefox, the browser is what translates all of the code in a manner for you to see and with which to interact, such as fonts, colors, drop-down menus, sliders, forms, etc. In order for all of this to work, though, there has to be something to support the front-end; this is where the backend comes into play.

4.1.1 Architecture of Front End user:



4.1.2 Module UI Description

1. Donor login:-

Donor Log In

Email

Password

Not a DONOR ? [Click here to Register](#)

2.Donor Registration:

Donor Registration

Donor Name :

Gender ☐ Male ☐ Female

Age

Mobile No

Blood Group

State

District

City

E-mail

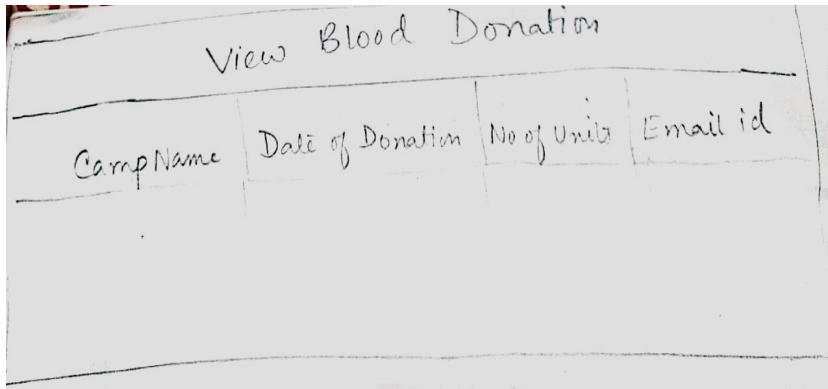
Password

Confirm Password

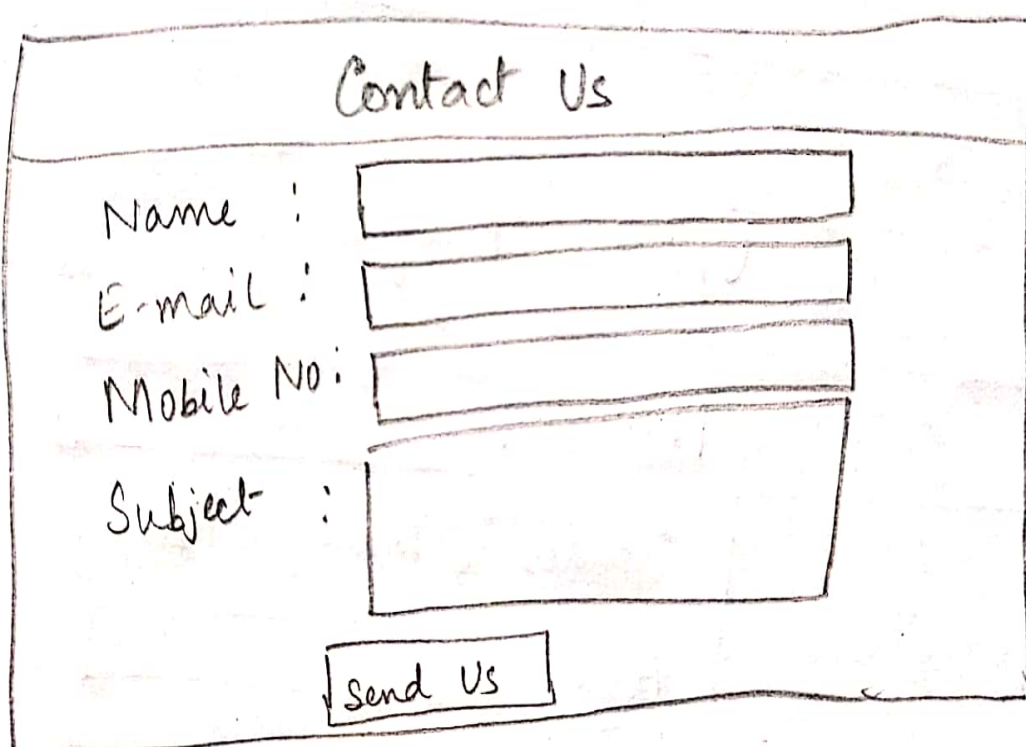
Upload Pic No file chose

3.Request for Blood:-

Requests For Blood				
Blood Group	Name	Mobile No	Email	Till Req. Date

4.View Blood Donation:-

Camp Name	Date of Donation	No of Units	Email id

5.Contact us:-

Contact Us	
Name :	<input type="text"/>
E-mail :	<input type="text"/>
Mobile No:	<input type="text"/>
Subject :	<input type="text"/>
<input type="button" value="Send Us"/>	

6. Blood Donated:-

A hand-drawn form titled "Blood Donated". It contains the following fields and controls:

- Select camp**: A dropdown menu with the word "Select" and a downward arrow.
- Date**: Three input fields for date selection. The first contains "JAN", the second contains "1", and the third is empty.
- No of Units**: A single-line text input field.
- Other Details**: A single-line text input field.
- Save**: A rectangular button at the bottom.

7. Profile Updation:-

A hand-drawn form titled "Update Profile". It contains the following fields and controls:

- Name**: A single-line text input field.
- Gender**: Two radio button options labeled "Male" and "Female".
- Age**: A single-line text input field.
- Mobile No**: A single-line text input field.
- Profile Picture**: A large square placeholder box on the left side.
- Update**: A rectangular button at the bottom right.

8. Blood Search:-

Search

Select Blood Group

Select State

Select District

Select City

9. Request for blood form:-

Requests For Blood

Name :

Gender ☐ Male ☐ Female

Age

Mobile No

Select Blood Group

E-mail

Till Required Date

Hospital Name

Detail

4.2. User Interfaces (Back End):

The back-end has three parts to it: server, application, and database.

Everything that the customer sees on the webpage is the front-end, as we have explained before, but once that customer enters all of his or her information, such as their name, billing address, destination, etc, the web application stores the information in a database that was created previously on the server in which the website is calling for information.

The web application creates, deletes, changes, renames, etc items in the database. For example, when a customer purchases a ticket, that creates an item in the database, but when they have a change in their order or they wish to cancel, the item in the database is changed.

.In short, when a customer wants to buy a ticket, the backend operation is the web application communicating with the server to make a change in a database stored on said server. Technologies like PHP, Ruby, Python, and others are the ones backend programmers use to make this communication work smoothly, allowing the customer to purchase his or her ticket with ease.

4.3 Hardware Requirement

Processor	:	Intel Core Duo 2.0 GHz or more
RAM	:	1 GB or More
Harddisk	:	80GB or more
Monitor	:	15" CRT, or LCD monitor
Keyboard	:	Normal or Multimedia
Mouse	:	Compatible mouse

4.4 Software Requirement

Front End : NotePad++

Wamp Server 3.1.7

Or

Visual Basic 2008 Express edition

With Sql Server Compact Edition

Microsoft SDK 3.0

Back End : MS Sql Server

Or

Wamp Server 3.1.7

Operation System : Windows

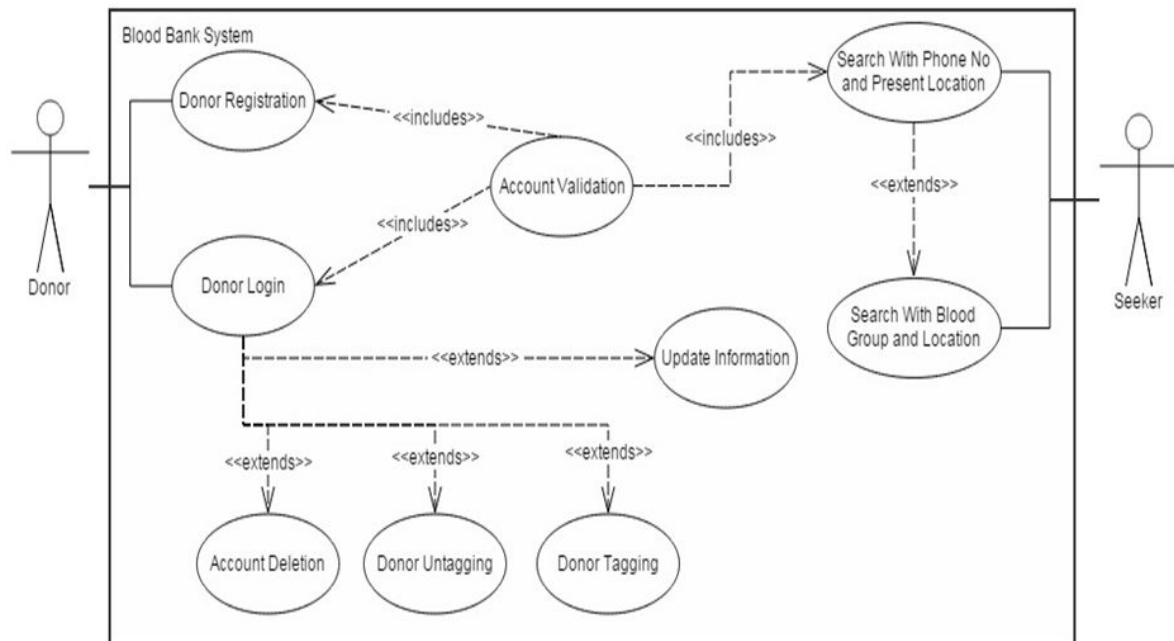
Or

Windows 8.1

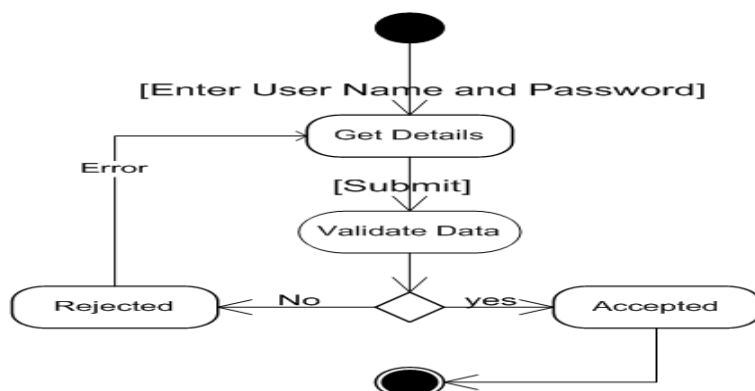
5. Diagrams

5.1 Use Case Diagram

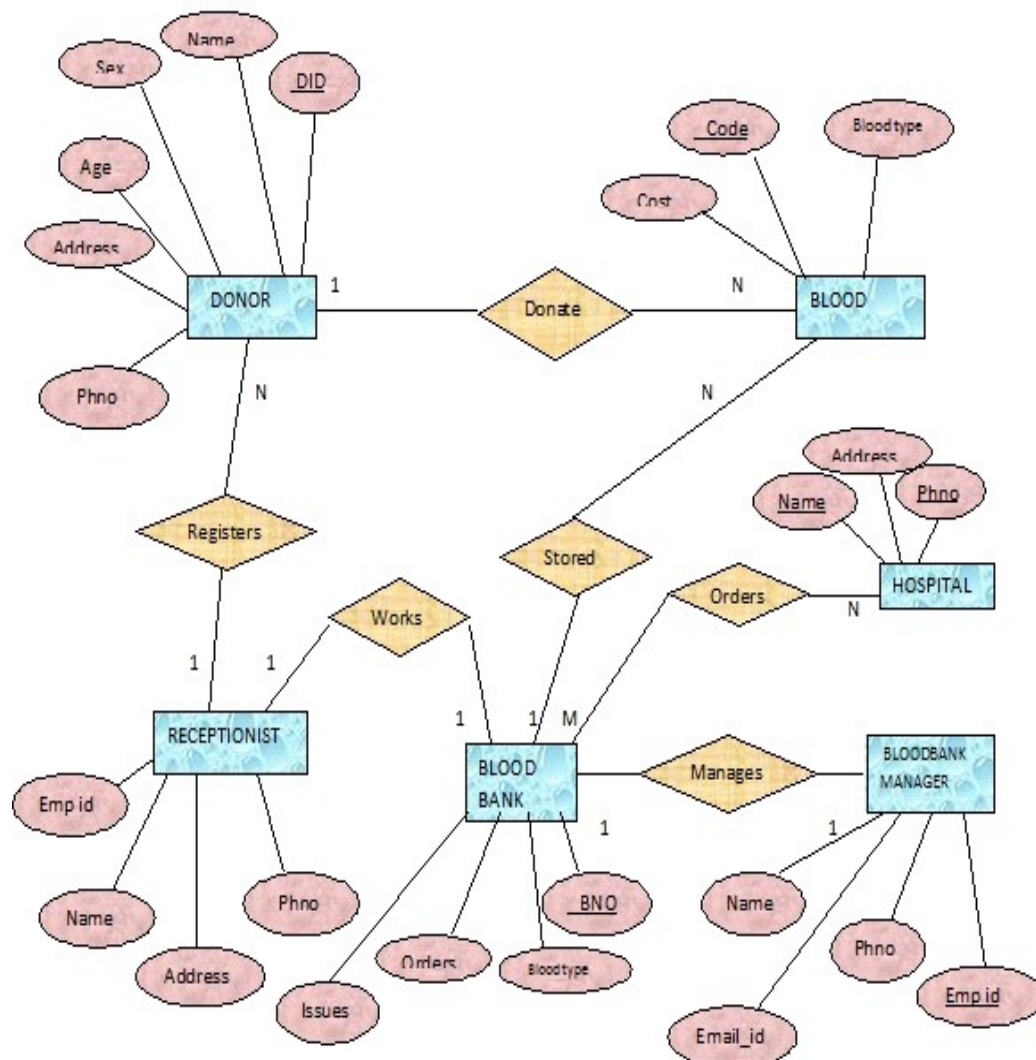
Use Case Diagram - Blood Bank



5.2 E-R Diagrams



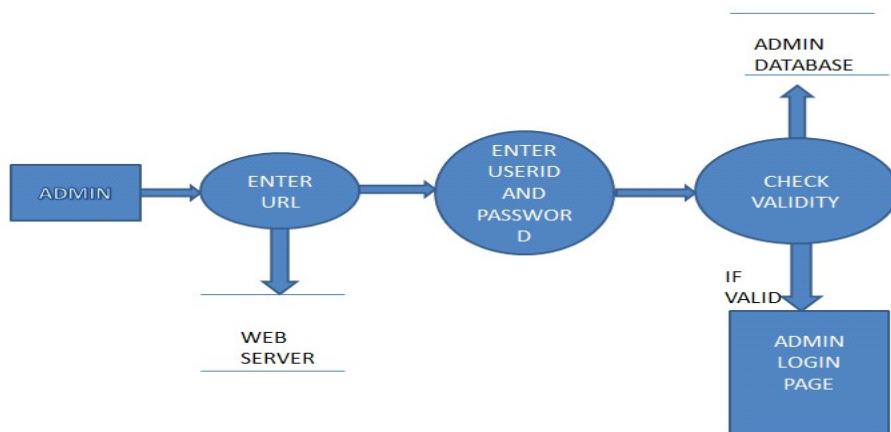
THE COMPLETE ER- DIAGRAM



5.3 Data flow diagram

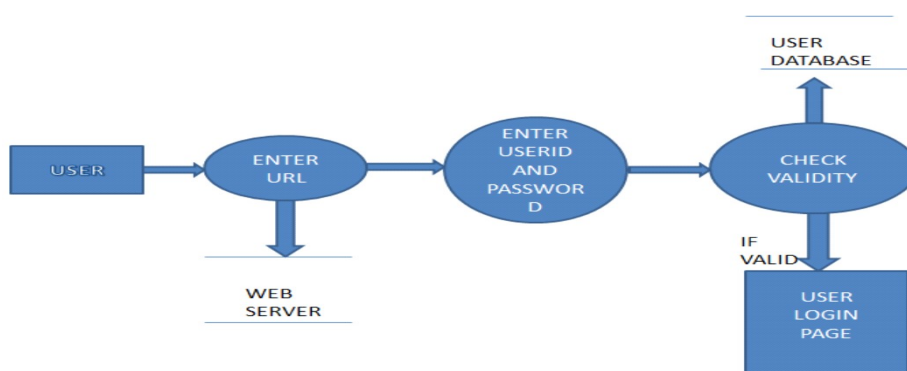
5.3.1 DFD FOR ADMIN LOGIN

After entering to the home page of the website , admin can choose the ADMIN LOGIN option where they are asked to enter username & password , and if he/she is a valid user then admin login page will be displayed.



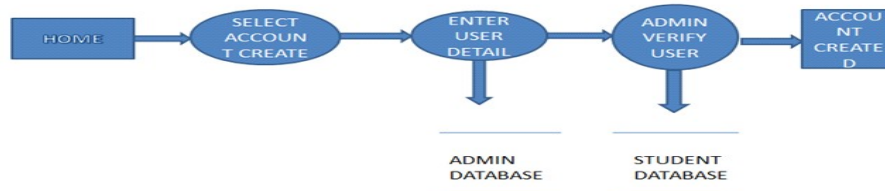
5.3.2 DFD FOR USER LOGIN

After entering to the home page of the website , student can choose the USER LOGIN option where they are asked to enter username & password , and if he/she is a valid user then a user login page will be displayed.



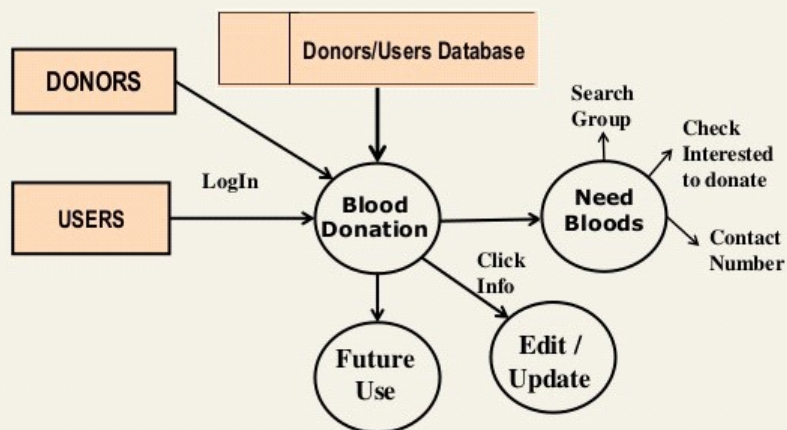
5.3.3 DFD FOR ACCOUNT CREATION

After the home page login there will be an option of CREATE AN ACCOUNT where after entering user detail ,if all the fields are filled then a request will be sent to the librarian who will approve him as a registered member of the website.



Data Flow Diagram

Diagram2 [Blood Donation process]



6.Summary and Conclusion

With the theoretical inclination of our syllabus it becomes very essential to take the utmost advantage of any opportunity of gaining practical experience that comes along. The building blocks of this Major Project "BLOOD BANK Management System" was one of these opportunities. It gave us the requisite practical knowledge to supplement the already taught theoretical concepts thus making us more competent as a computer engineer. The project from a personal point of view also helped us in understanding the following aspects of project development:

- The planning that goes into implementing a project.
- The importance of proper planning and an organized methodology.
- The key element of team spirit and co-ordination in a successful project.

The project also provided us the opportunity of interacting with our teachers and to gain from their best experience

7.Future Recommendation

BLOOD BANK MANAGEMENT is a software application to build such a way that it should suits for all type of blood banks in **future**.

One important future scope is availability of location based blood bank details and extraction of location based donor's detail, which is very helpful to the acceptant people. All the time the network facilities cannot be use. This time donor request does not reach in proper time, this can be avoid through adding some message sending procedure this will help to find proper blood donor in time. This will provide availability of blood in time.

8. Glossary

Term	Definition
Administrator	Director of the blood bank.
Client Database	Is a table that records all Client related details; Is related to the Inventory Database
Camp Event Manager	Is a Blood Bank representative who manages all camp related activities and updates the Stock Database; bridges the gap between the Donor and the Inventory Manager.
Client Manager	Is a Blood Bank employee responsible for handling and allocating client request; bridges the gap between the Client and the Inventory Manager.
Crash	A computer crash means that the computer itself stops working or that a program aborts unexpectedly.
Cross Matching	Cross checking the details provided by the client with the hospital to omit discrepancies.
Donor Database	Is a table that records all Donor related details; Is related to the Sample Database
Inventory	Database containing a complete records of all Blood Groups and their individual components.
Inventory Database	Is a table that records all Inventory related details.
Inventory Manager	Is a Blood Bank employee responsible for maintaining and updating the Inventory and the Stock Database.
Partner Companies	Organizations interested in organizing Blood Donation Camp.
Partner Database	Is a table that records all Partner related details.
Partner Manager	Is a Blood Bank employee responsible for managing companies requesting to organize blood donation camps is collaboration with the Blood Bank.
Sample Database	Is a table that records all collected Sample related details; Is related to the Inventory Database
Serial Attacks	Multiple unsuccessful attempts to unlock an account.
Software Requirement Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.

Stakeholder	Any person with an interest in the project who is not a developer.
Stock Database	Is a table that records all Stock related details such as the number of bags present, the amount of reagents present etc.
Transfusion	An act of transferring donated blood or it's component.

9. Bibliography

PHP Manual www.php.net/

<https://www.google.com>

<http://www.w3schools.com>

<http://www.indianbloodgroup.com>