



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(MUMBAI)

A
Project Report On

“Blood Bank Management”

Submitted by

Pranita Jejurkar
Rashmi Shewale
Sanjana Khairnar
Aishwarya Sonawane

Guided by – Ms.P.N.Patil



Department of Computer Technology
MVP's Rajarshi Shahu Maharaj Polytechnic,
Udoji Maratha Boarding Campus, Gangapur Road, Nashik-13
(2016-2017)

MVP's Rajarshi Shahu Maharaj Polytechnic, Nashik
Udoji Maratha Boarding Campus, Gangapur Road, Nashik-13
(2016-2017)
Department of Computer Technology



CERTIFICATE

This is to certify that project entitle “Blood Bank Management” has been carried out by following students from MVP's Rajarshi Shahu Maharaj Polytechnic, Nashik of the Diploma in the Computer Technology Department of MSBTE, Mumbai during academic year 2016-17.

Group Members:

- 1) Pranita Jejurkar
- 2) Rashmi Shewale
- 3) Sanjana Khairnar
- 4) Aishwarya Sonawane

Date:

Place:

Internal Examiner

External Examiner

Guide
(Ms. P.N.Patil)

H.O.D
(Mrs. A.N.Birari)

Principal
(Prof. B.N.Rajole)

ACKNOWLEDGEMENT

Inspiration and guidance are invaluable in every aspect of life, especially in the field of education, which I have received from our respected H.O.D. **Mrs.A.N.Birari** who has guided me in the first two phases of dissertation work and gave earnest co-operation whenever required. I would like to express my sincere gratitude towards her.

I am pleased to announce that my presentation of the project as well as the report would not have been completed without the able guidance and complete support of **Ms.P.N.Patil** those helped me at each and every step in every possible way. She always provided me with access to the latest technology and facilities and encouragement at every point and took active participation in the achievement of my objective. Heartfelt my foremost thanks go to dissertation guide and help of my well-wishers and colleagues.

At last, I would like to take this opportunity to convey thanks to all my staff members, who directly or indirectly encouraged and helped me to complete my work on time and contributed their valuable time in helping me to achieve success in the work of project.

INDEX

Sr.no.	Name of topic	Page no.
	Abstract	I
1	Introduction	1-3
2	Literature Survey	4-6
3	Requirement Analysis	7
4	System Description	8-13
5	Modeling and Designing	14-34
6	Testing	35-39
7	Conclusion and Future Scope	40
	Cost	41-42
	Result	43-44
	References	45
	Appendix-1	
	Appendix-2	

List of the figures:-

Fig.no.	Name of figure	Page no.
1.1	Incremental model	2
2.1	Screenshot of Blood Banks India Directory of Nagaland State	4
2.2	Screenshot of Blood Banks India Directory of Maharashtra State	4
2.3	Screenshot of Blood Banks India Directory	5
2.4	Screenshot of Nepal Blood Donors	5
2.5	Screenshot of Donate Blood Save Life's	6
4.1	System Architecture	8
5.1	System Flow diagram	14
5.2	ER Diagram	15
5.3	DFD Level I	16
5.4	DFD Level II	17
5.5	Classification of UML	18
5.6	Class Diagram	19
5.7	Use-case diagram for Admin	20
5.8	Use-case diagram for Blood Bank	21
5.9	Use-case diagram for Donor	22
5.10	Use-case diagram for Patient	23

5.11	Sequence diagram for Admin	24
5.12	Sequence diagram for Blood Bank	25
5.13	Sequence diagram for Donor	26
5.14	Sequence diagram for Patient	27
5.15	Activity diagram for Admin	28
5.16	Activity diagram for Blood Bank	29
5.17	Activity diagram for Donor	30
5.18	Activity diagram for Patient	31
5.19	Collaboration diagram for Admin	32
5.20	Collaboration diagram for Blood Bank	33
5.21	Collaboration diagram for Donor	33
5.22	Collaboration diagram for Patient	34
6.1	Snapshot of Search Option	38
6.2	Snapshot of Donor Registration	39

List of the Tables:-

Table no.	Title of Table	Page no.
6.1	Donor Login Test Cases	35
6.2	Donor Registration Test Cases	35
6.3	Blood Bank Test Cases	36
6.4	Search(Donor Request) Test Cases	37
6.5	Admin Test Cases	37

Abstract

This project is aimed to developing an online Blood Donation Information. The entire project has been developed keeping in view of the distributed client server computing technology, in mind. The Blood Donation Agent is to create an e-information about the donor and organization that are related to donating the blood. Through this application any person who is interested in donating the blood can register himself in the same way if any organization wants to register itself with this site that can also register. More over if any general consumer wants to make request blood online he can also take the help of this site. Admin is the main authority who can do addition, deletion, and modification if required. The project has been planned to be having the view of distributed architecture, with centralized storage of the database. The application for the storage of the data has been planned.

Keywords: Distributed Client Server, Distributed Architecture, Centralized Storage, Storage.

Chapter 1

Introduction

1.1 Overview

The proposed Blood Bank management system helps the people who are in need of a blood by giving them all details of blood group availability or regarding the donors with the same blood group. The people in need of blood can search for the donors by giving their blood group and city name. It saves time as he can search donors online without going anywhere. Using this system user can get blood in time and can save his relative or friend life. Our website work 24x7 so user can get information of blood donor any time. Blood donor can also get registered and save life of other person. The main benefit of this system is the information of available blood group.

When blood is need in the operation then people have very less time to get the blood available so if he get the information like who can give him blood in time in his city is lifesaving. And here our system work, whenever a person need blood he get information of the person who has the same blood group he needs.

Centralized Storage:

A centralized storage is a storage that is located, stored, and maintained in a single location. This location is most often a central computer or database system, for example a desktop or server CPU, or a mainframe computer. In most cases, a centralized database would be used by an organization or an institution.^[3]

Distributed Architecture:

In a distributed architecture, components are hosted on different platforms and communicate through a network. Distributed architecture is a field of computer science that studies distributed systems. A distributed system is a model in which components located on networked computers communicate and coordinate their actions by passing messages.^[4]

Distributed Client Server:

A Client-Server Architecture consists of two types of components: clients and servers. A server component perpetually listens for requests from client components. When a request is received, the server processes the request, and then sends a response back to the client. Servers may be further classified as stateless or stateful.

Clients of a stateful server may make composite requests that consist of multiple atomic requests. This enables a more conversational or transactional interactions between client and server.^[5]

Storage:

Computer data storage, often called storage or memory, is a technology consisting of computer components and recording media used to retain digital data. It is a core function and fundamental component of computers.^[6]

1.3 Applying Software Engineering Approach

The goal of system design is to produce a model or representation that exhibit, commodity and delight. It provides information about the application domain for the software to be built. It fully describes the internal details of each software. Here are some advantages of incremental model:-

1. Each iteration passes through the requirements, design, coding and testing phases.
2. Software will be generated quickly during the software life cycle.
3. It is flexible and less expensive to change requirements and scope.
4. Customer can respond to each built and errors are easy to be identified.
5. Easier to test and debug during a smaller iteration.

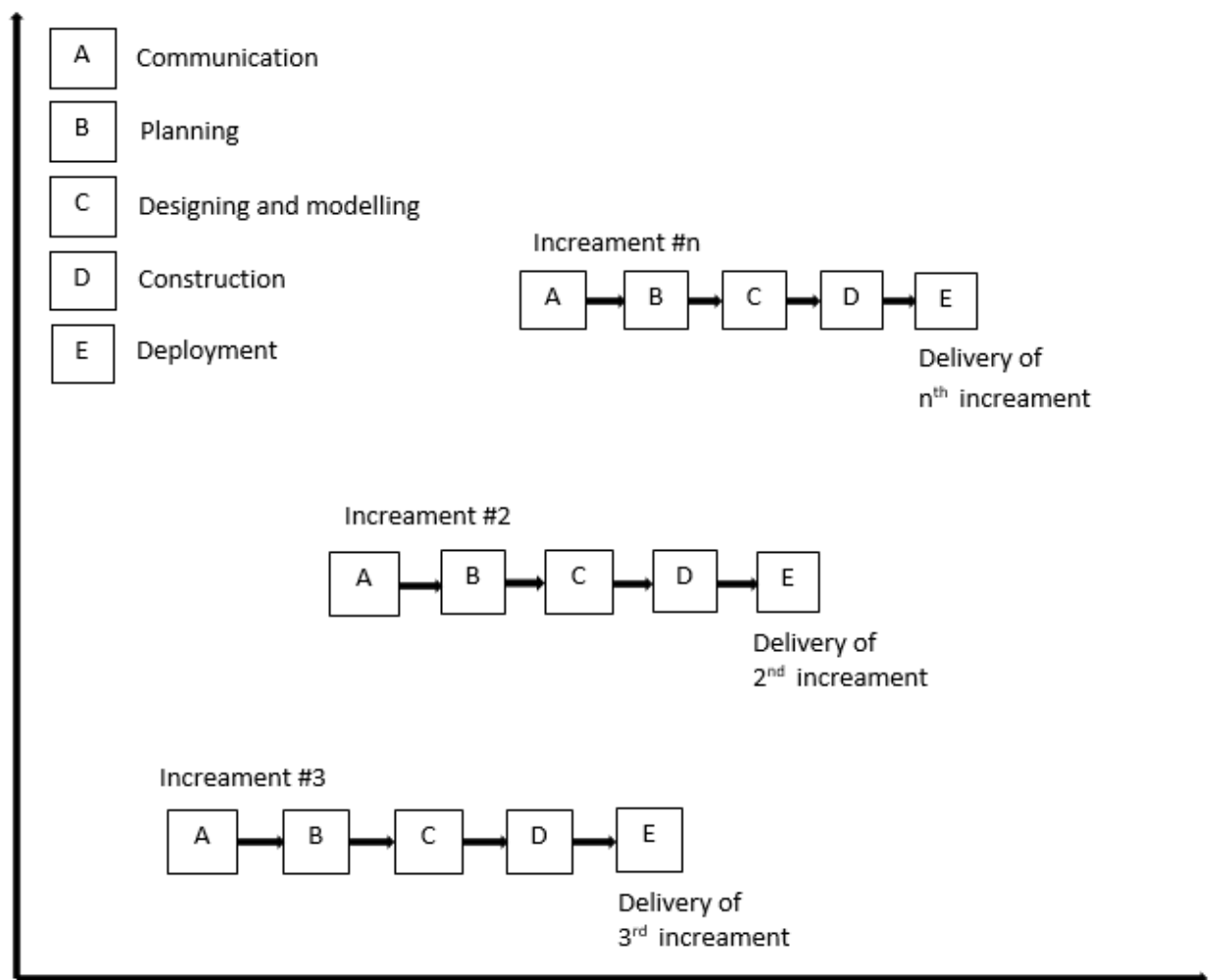


Fig.1.1: Incremental Model

1.3.1 Communication

This is the first step where the user initiates the request for a desired software product. He contacts the service provider and tries to negotiate the terms. He submits his request to the service providing organization in writing.

1.3.2 Planning and Requirement Analysis

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry.

1.3.3 Designing and Modeling

Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS.

1.3.4 Construction

This step is also known as programming phase. An estimate says that 50% of whole software development process should be tested. Software testing is done while coding by the developers and thorough testing is conducted by testing experts at various levels of code such as module testing, program testing, product testing, in-house testing and testing the product at user's end.

1.3.4 Deployment

Once the product is tested and ready to be deployed it is released formally in the appropriate market. The product may first be released in a limited segment and tested in the real business environment (UAT).

Chapter 2

Literature Survey

2.1 Blood Donation app ‘National Blood Banks Directory’ was developed in the year 2016 for the purpose of donating and receiving the blood. This app contains state/city wise list of Blood banks. This app is useful for particular area not universally. In fig. 2.1.1. Telephone is not mentioned for contact. It is very important for donor to know basic details about the receiver and also to contact with the receiver the contact number is required. So the contact no plays important role in Blood bank System. In fig. 2.1.2. Telephone is mentioned for contact.^[7]

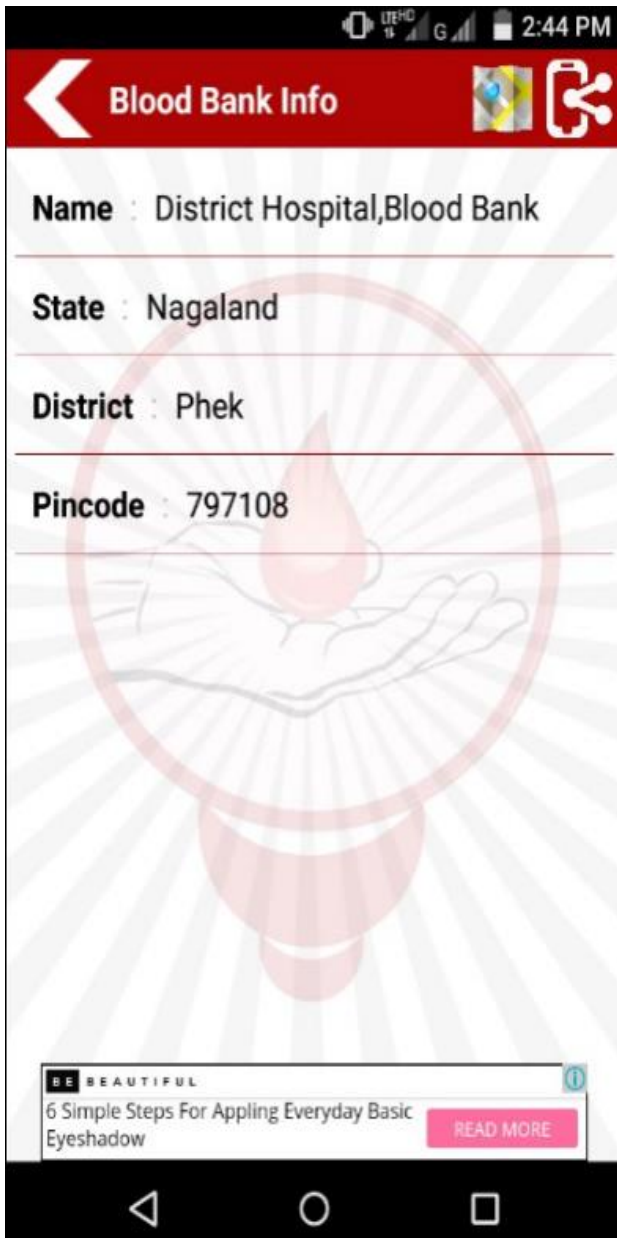


Fig.2.1: Screenshot of Blood Banks India Directory for Nagaland State



Fig.2.2: Screenshot of Blood Banks India Directory for Maharashtra State

2.2 In the Fig.2.2.1 the app ‘Blood Banks India Directory’ displays only the Blood Banks which helps the user to communicate with only blood bank but the user should allow to communicate with another user.^[7] In the Fig.2.2.2 the app ‘Nepal Blood Donors’ requires email ID and the password or our Facebook account which loses security. Therefore our app allows user to create there new login details. This helps to maintain a security.^[8]



Fig.2.3: Screenshot of Blood Banks India Directory

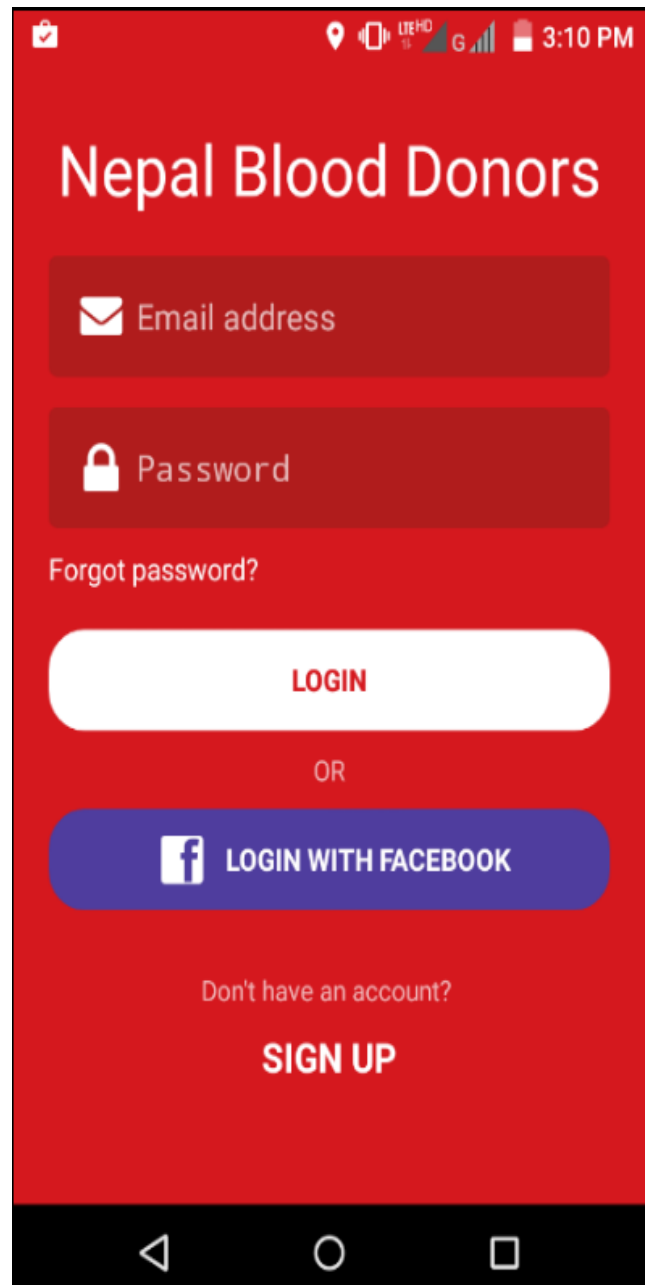


Fig.2.4: Screenshot of Nepal Blood Donors

2.3. Some apps have location tracking system which may not be useful when donor's GPS is off. If donors GPS is off the receiver will not be able to communicate with the donor and in the case if the GPS is on there are rare conditions to find the donor as shown in Fig.2.3.1. and the GPS reduce the battery.

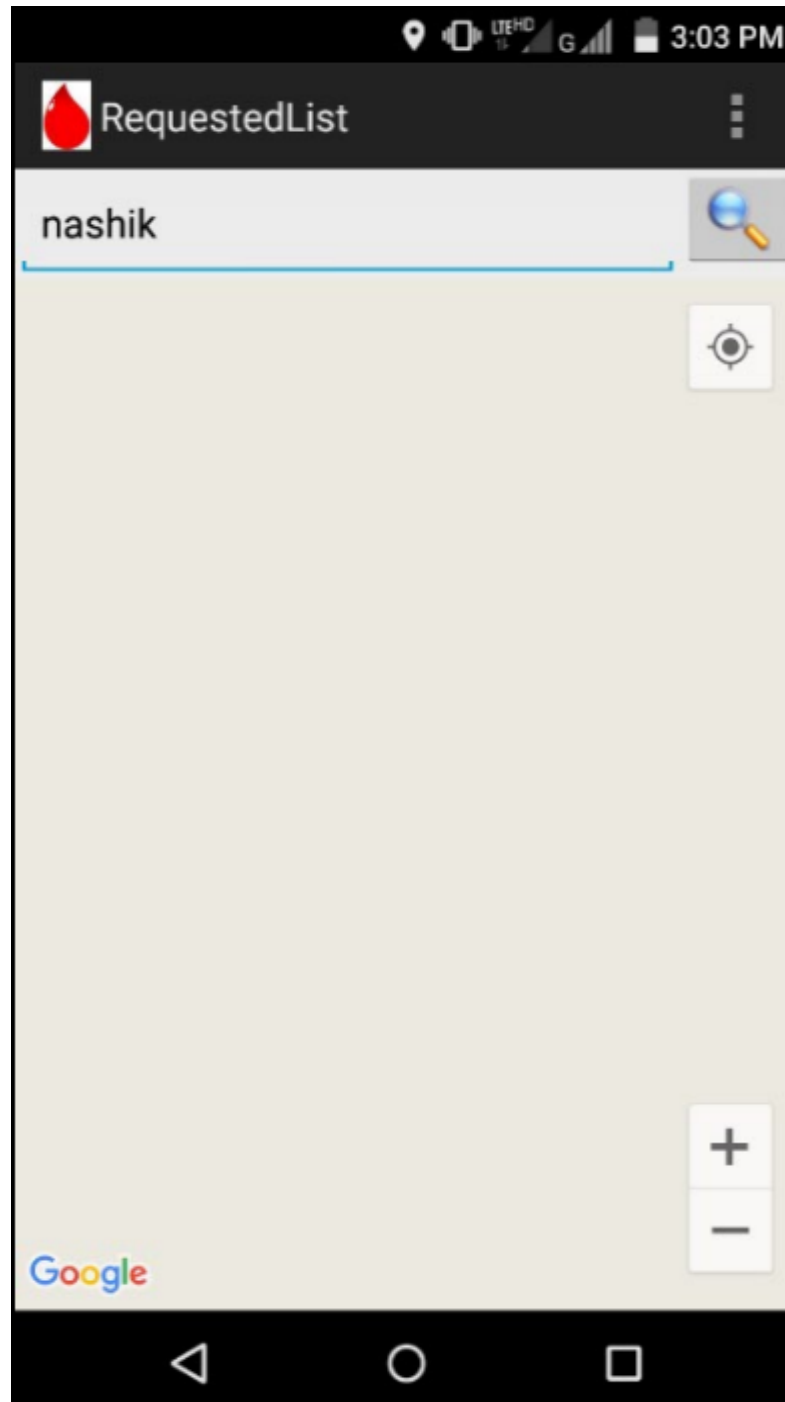


Fig.2.5: Screenshot of Donate Blood Save Life's

Chapter 3

Requirement Analysis

3.1 Functional Requirements:

3.1.1 Software Requirements:

- 1) Platform: Windows XP

Platform is any hardware used to host an application or service. ^[9]

- 2) Language: PHP

A programming language is a formal computer language designed to communicate instructions to a machine, particularly a computer. Programming languages can be used to create programs to control the behavior of a machine or to express algorithms.

- 3) Mobile Client: Android

Mobile server is a computer system (computer hardware and operating system), that responds to requests across a computer network to provide, or help to provide, a network service, while being easily portable in a laptop form factor.

- 4) IDE/Tool: Eclipse IDE

An integrated development environment (IDE) is a software application that provides comprehensive facilities to computer programmers for software development.

3.1.2 Hardware Requirements:

- 1) Processor: Pentium IV

A processor is the logic circuitry that responds to and processes the basic instructions that drive a computer.

- 2) RAM: 64 MB

Random-access memory (RAM) is a form of computer data storage which stores frequently used program instructions to increase the general speed of a system.

- 3) Storage: 20 GB

Computer data storage, often called storage or memory, is a technology consisting of computer components and recording media used to retain digital data. It is a core function and fundamental component of computers. ^[6]

- 4) Mobile Phone: Android Phone

A telephone with access to a cellular radio system so it can be used over a wide area, without a physical connection to a network.

- 5) Monitor: 15"

A computer monitor or a computer display is an electronic visual display for computers.

3.2 Non-Functional Requirements:

- 1) Person: A person is required for the transaction of blood.
- 2) Blood Bank: Blood bank update the user information with medical tests also can add new user and new bank.
- 3) Admin: Admin manages all the transactions between User, Blood Bank and donor which helps the app to execute effectively.

Chapter 4

System Description

4.1 Brief Description

The Blood Donation Agent is to create an e-Information about the donor and organization that are related to donating the blood. Through this application any person who is interested in donating the blood can register himself in the same way if any organization wants to register itself with this site that can also register. More over if any general consumer wants to make request blood online he can also take the help of this site. Admin is the main authority who can do addition, deletion, and modification if required. The project has been planned to be having the view of distributed architecture, with centralized storage of the database. The application for the storage of the data has been planned. Using the constructs of MS-SQL Server and all the user interfaces have been designed using the PHP technologies.

Working:

1. Donor will register through the app or donor can register through blood bank.
2. Donor has update the medical details by visiting the near blood bank.
3. If receiver has requested to donor, the donor will go to blood bank
4. Blood bank will check if the donor is eligible or not to donate the blood. And if the donor is eligible then and then only the blood can be donated by the donor.
5. After donating the blood, blood bank will update the date of donation so donor will not be able to donate the blood up to 3 months.
6. The receiver will sent request to the person with same blood by searching district wise and contact with that donor respectively.
7. Admin will handle all the transaction between the receiver, blood bank and donor.

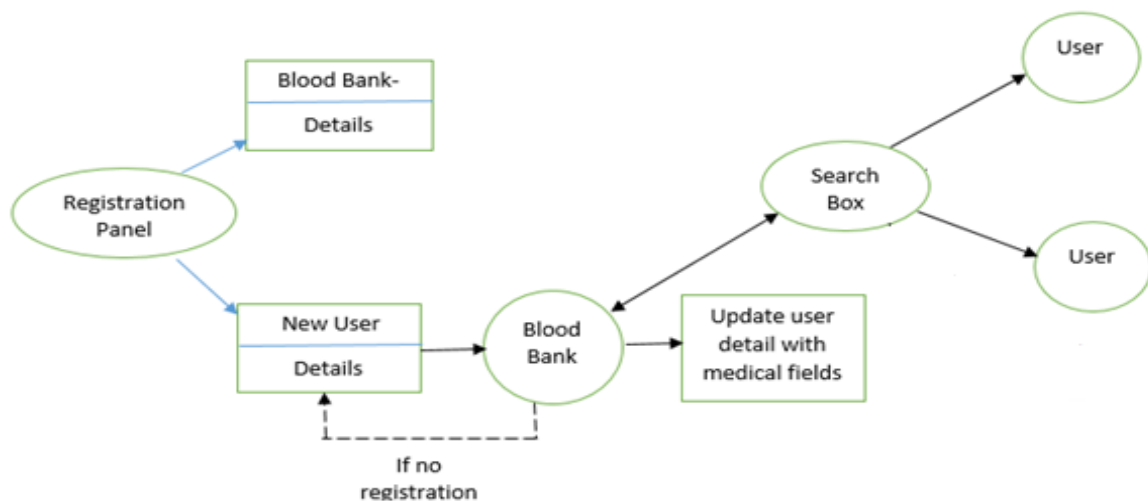


Fig.4.1: System Architecture

4.2 Modules of Blood Bank Management System

Online Blood Bank management system is to provide services for the people who are in need of blood by getting help from the donors who are interested in donating blood for the people. There are seven main modules in this system.

1. Admin
2. Donors
3. Donor Registration
4. Modifying Donor Information
5. Receivers
6. Donor Search
7. Life Saving Contacts

4.2.1 Admin

Admin can manage donors, bank & receivers. He can add or remove any user and bank from the system. After filling the details by the bank, details will be verified by the admin and then and then only the user id and password will be provided to the respected bank.

- Change Password
- Modify donor/bank details
- delete donor/bank details
- Logout

Whenever a user wants to change his / her password he can select the change password option. The system displays the form, which asks him for his old password and new password. The system then compares the old password with the existing password in the database and if they match then the password is set to the new password in the database.

4.2.2 Donors

From this module user can create their account, when user create his account the user create a user id and password, which identifies him uniquely. From this module user can search donor for blood and can also refer his friend to become a donor. Donor can also get information like when he donated blood or when he will be able to donate blood.

4.2.3 Donor Registration

In this module, people who are interested in donating blood get registered in my site and give his overall details related to him, i.e. he fills in a registration form by giving the total details such as name, address, city, gender, dob, blood group, telephone numbers, e-mail address, etc. He was also given two fields' username and password to fill such that he was a registered donor and he can enter the login form with his username and password and can modify his details if needed.

4.2.4 Modifying Donor Information:

The registered donor only is able to modify his details; no other person can modify his details as there was a login form which restricts others from entering the username and password providing high security for the details given by the donor. If at all the donor wants to modify his details, he was forced to give his username and password to enter in. After giving the username and password it checks for the donor whether he is an existing donor or not and if the username and password matches, he can then able to modify his total details. If the username and password do not exist then he gets a message as ‘Wrong ID and Password Entered, Try Again’.

Following links are available on donor module.

- Home
- Update Profile
- View Donation
- People in need
- Change Password
- Logout
- Search

4.2.5 Receivers

This module helps user to find blood group. When user click on find a blood group or search option system ask him to enter blood group, state, city, taluka in which who want search. After entering the blood group, system search for the availability of the blood group and give him the list of the donors who has the same blood group.

4.2.6 Donor Search:

The people who are in need of blood can search in our site for getting the details of donors having the same blood group and within the same city. They can directly click on the link search a donor and can select a city name as well as the blood group which he needs. He then gets the details of the donors who exist within the city and the same blood group that he has selected. If no match was are found for the city and group selected by him he gets a message ‘SORRY DONORS ARE NOT AVAILABE WITH THE FOLLOWING BLOOD GROUP AND AREA’.

4.2.7 Life Saving Contacts:

If at all the people in search of a donor doesn’t get any match for their area and group then they will be provided a service i.e. he will be given a Contact Person details for their nearby cities who have the details of many other donors with him. The people in search can call him and can get the details of the donors and can be provided services in this manner. But this life saving contact persons can be available only for a limited number of cities but not for all. These contact persons are the authorized persons of my blood bank.

4.3 Pseudo code for Blood Bank Management System

4.3.1 Client (Search module)

```

</script>
<Script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
<!--This link is important for dependent drop drop----->
<form method="post" enctype="multipart/form-data" id="registrationForm"
action="find.php">
<table class="table table-bordered table-responsive" style="color:#000">
<tr><td>Blood Group</td><td><select name="blood_group" class="form-control"
id="fetchval">
<option>-----Select Bloodgroup-----</option>
<?php
$q="select * from bloodgroup";
$r=mysqli_query($con,$q);
while($row=mysqli_fetch_array($r))
{
?>
<option value="<?php echo $row['bloodgroupname'] ?>"><?php echo
$row['bloodgroupname'] ?></option>
<?php
}
?></select>
</td></tr>
<tr>
<td>State</td>
<td>
<select class="c form-control validate[required]" name="state" id="state">
<option value="" selected="selected">---Select state---</option>
<?php
require 'config.php';
$sql="select DISTINCT state from statecity";
$result=mysqli_query($con,$sql) or die(mysqli_error());
while($row=mysqli_fetch_array($result))
{?>
<option value="<?php echo $row['state'];?>"><?php echo $row['state'];?></option>
<?php }
?>

</select>
</td>
</tr>
<tr>
<td>District</td>
<td>
<select class="form-control validate[required]" id="district" name="district">

```

```
<option value="" selected="selected">--Select District--</option>
```

```
</select>
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Taluka</td>
```

```
<td>
```

```
<select class="form-control validate[required]" name="city" size="1"
id="city">
```

```
<option value="" selected="selected" >--Taluka--</option>
```

```
</select>
```

```
</td>
```

```
</tr>
```

```
<tr><td colspan="2" align="center"><input type="submit" name="save" value="Search"
class="btn btn-success"> &nbsp;&nbsp;&nbsp;&nbsp;<input type="reset" name="reset"
value="Clear" class="btn btn-danger"></td></tr>
```

```
</table>
```

```
</form>
```

```
</div>
```

```
</div>
```

```
<?php /*?> <?php
```

```
if(isset($_POST['save']))
{
```

```
$query="select * from donor_registration where
blood_group='".$_POST['blood_group']."'and
state='".$_POST['state']."'and
district='".$_POST['district']."'and city='".$_POST['city']."'";
$res=mysqli_query($con,$query) or die(mysqli_error($con));
if(mysqli_num_rows($res)>0)
```

```
{
?>
```

```
<table class="table table-bordered">
```

```
<tr>
```

```
<th>Name</th>
```

```
<th>Gender</th>
```

```
<th>Age</th>
```

```
<th>Mobile No.</th>
```

```
<th>Blood Group</th>
```

```
<th>Email</th>
```

```
<th>Image</th>
```

```
</tr>
```

```

<?php
while($row=mysqli_fetch_array($res))
{
?>
<tr>
<td><?php echo $row['donor_name'] ?></td>
<td><?php echo $row['gender'] ?></td>
<td><?php echo $row['age'] ?></td>
<td><?php echo $row['mobile_no'] ?></td>
<td><?php echo $row['blood_group'] ?></td>
<td><?php echo $row['email'] ?></td>
<td></td>
</tr>
<?php
}
}
else
{
echo "<script>";
echo "alert('Result Not Found');";
echo "</script>";
}

}
?>
</table><?php */?>

</div>
<div class="col-md-3 w3ls-about-top-left-grid"></div>
<div class="clearfix"> </div>
</div>
</div>

```

Chapter 5

Modeling and Designing

5.1 System Flow Diagram

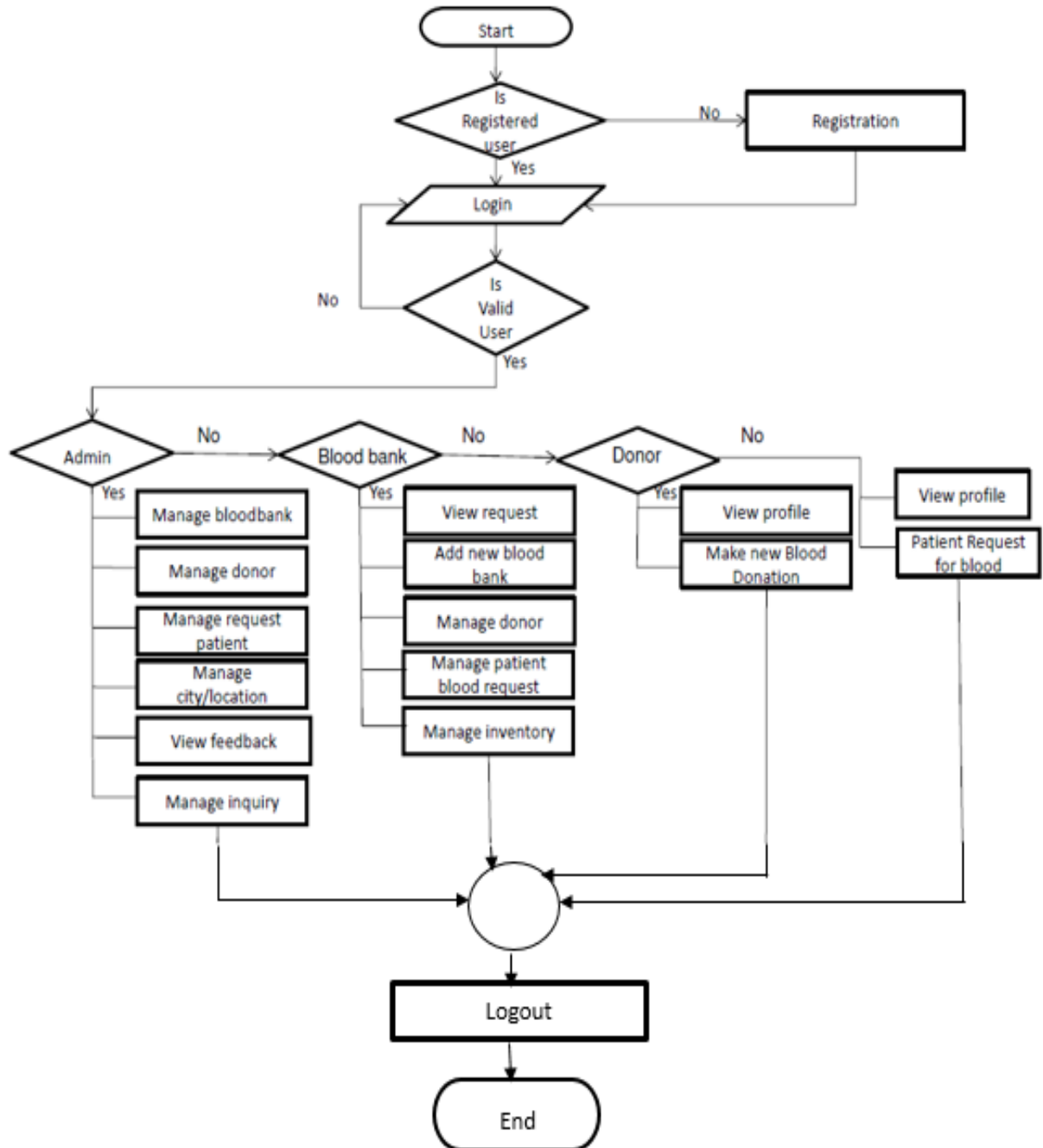


Fig.5.1: System Flow Diagram

5.2 ER Diagram

An entity relationship model, also called an ER diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within database or information system.

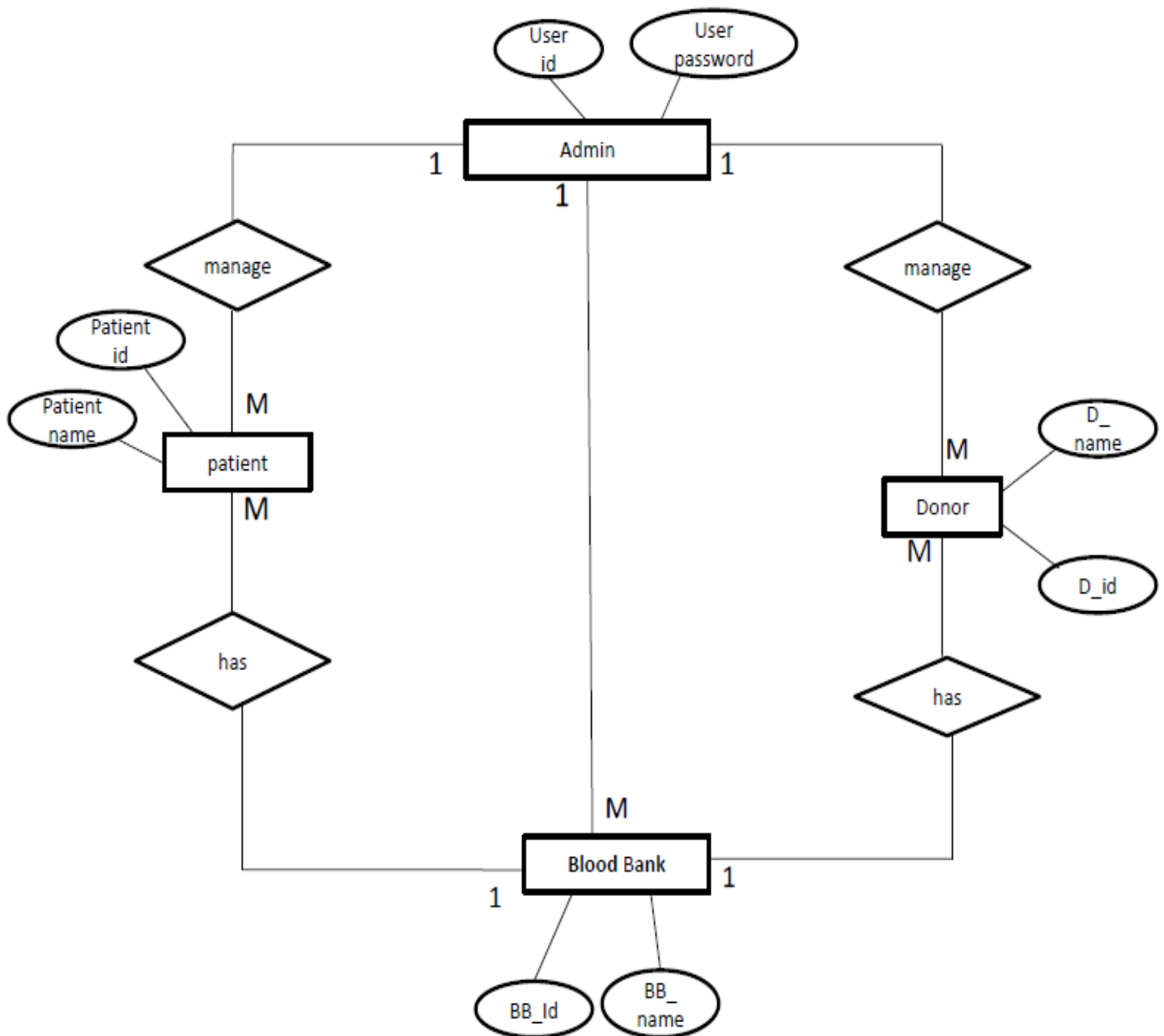


Fig.5.2: ER Diagram

5.3 Data Flow Diagram

A data flow diagram (DFD) is a graphical representation of “flow” of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of system, which can later be elaborated. DFD can also be used for the visualization of data processing.

5.3.1 DFD level-I

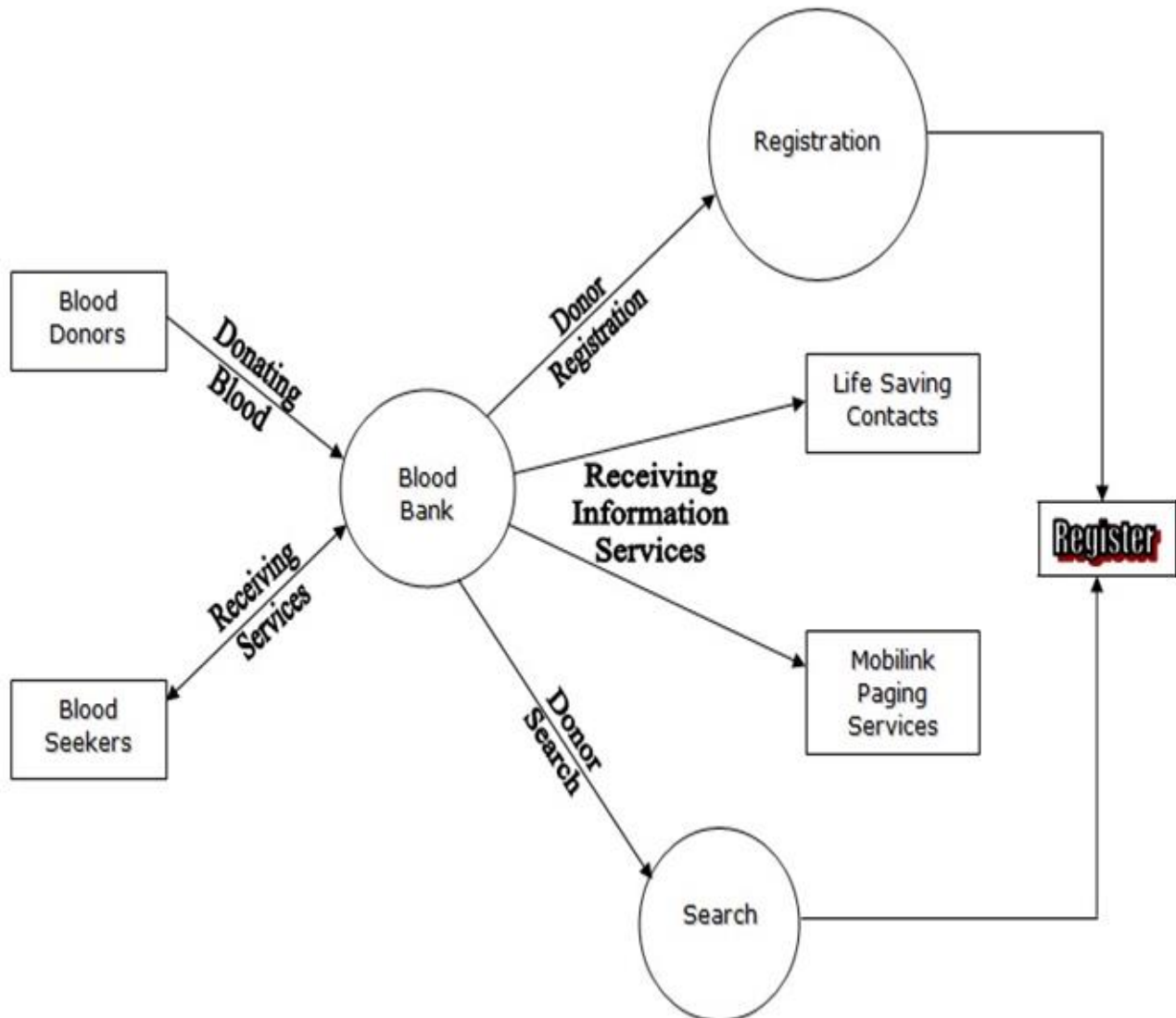


Fig.5.3: DFD Level-I

5.3.2 DFD level-II

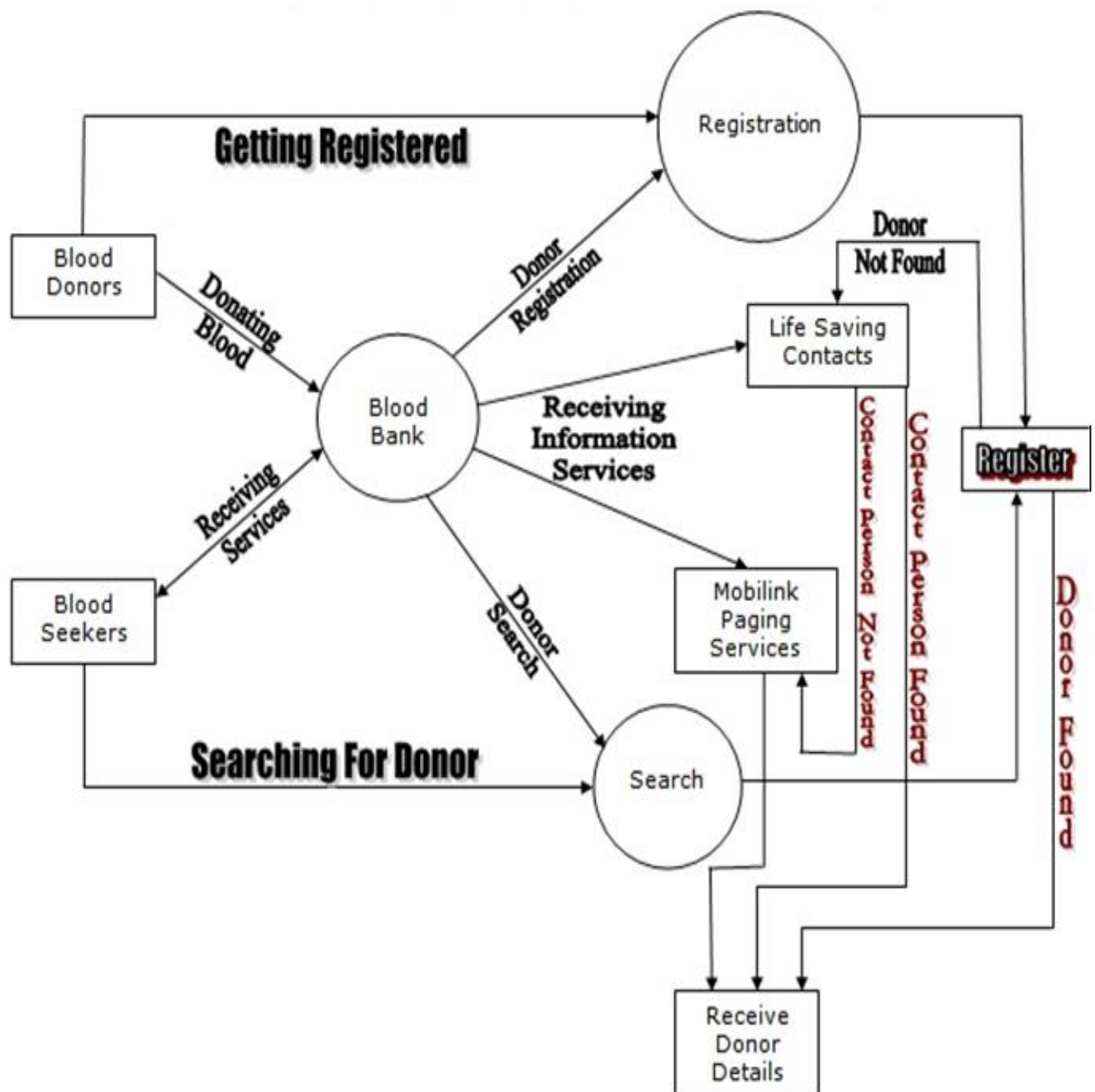


Fig.5.4: DFD Level-II

5.4 UML Diagrams

UML stands for Unified Modeling Language. The approach used by UML is called as object oriented approach for the development of models.

UML is used in converting reality with the help of simplest models. The major contributors to UML are of James Rumbaugh Ivar Jacobson and Grady Booch and the Rational Software Corporation.

Because of great contribution of the people and organization above, UML is accepted as a standard modeling language by OMG.

The UML modeling consist of following diagrams to model a software system and those diagrams are:

1. Object Diagram
2. Class Diagram
3. Use-Case Diagram
4. Sequence Diagram
5. Activity Diagram
6. Collaboration Diagram
7. Deployment Diagram

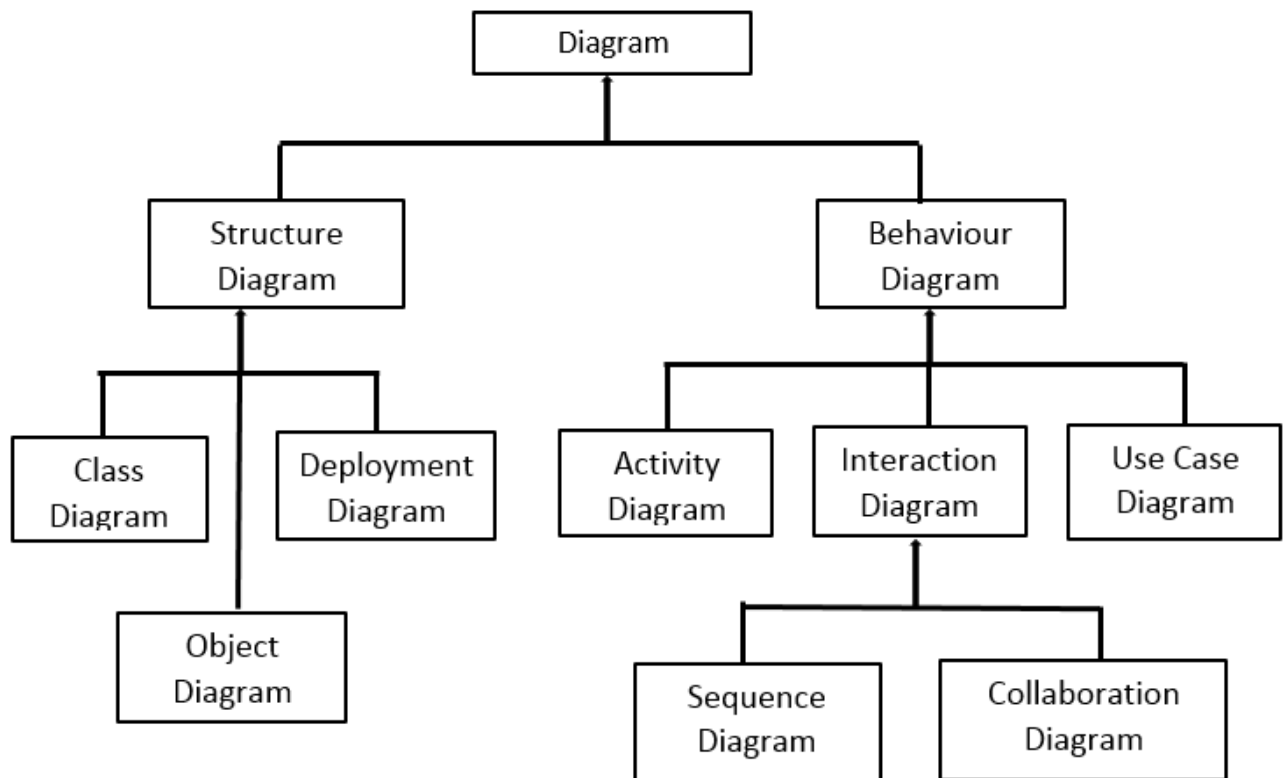


Fig.5.5 Classification of UML

5.4.1 Class Diagram

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. The class diagram is a static diagram. It represents the static view of an application. The class diagrams are widely used in the modelling of object oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages.

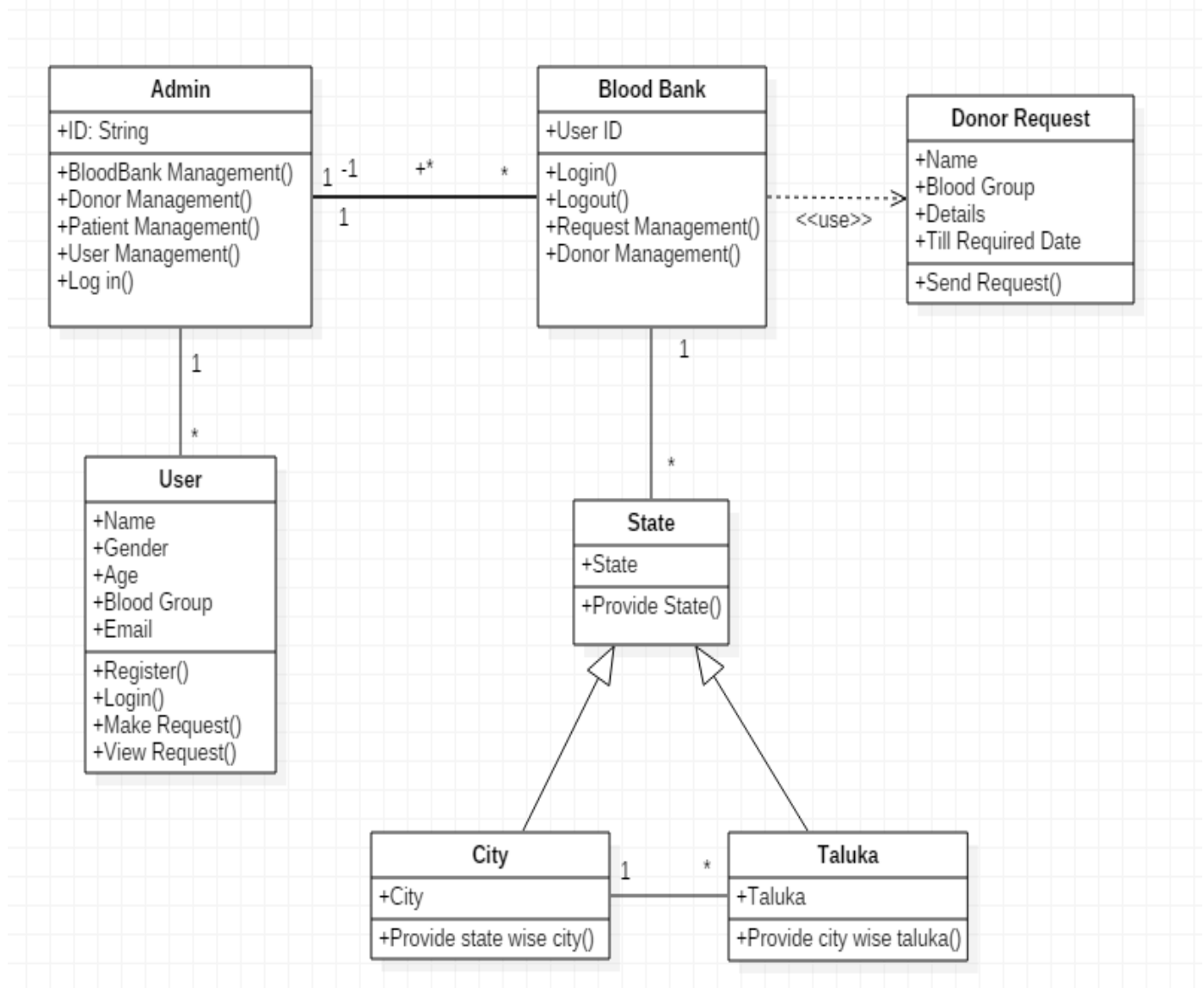


Fig.5.6: Class Diagram

5.4.2 Use-case Diagram

Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors). A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.

5.4.2.1 Admin

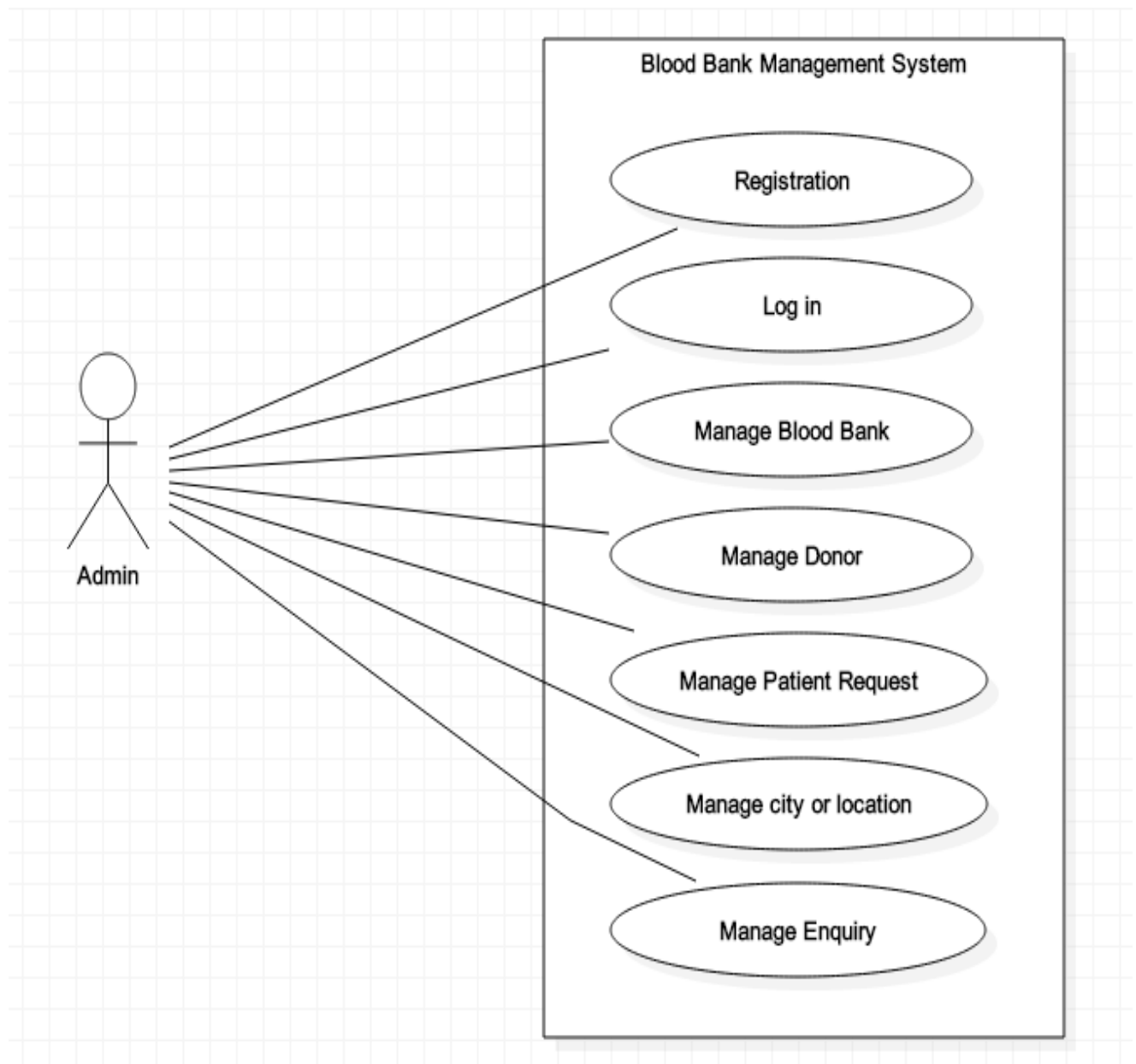


Fig.5.7: Use-Case Diagram for Admin

5.4.2.2 Blood Bank

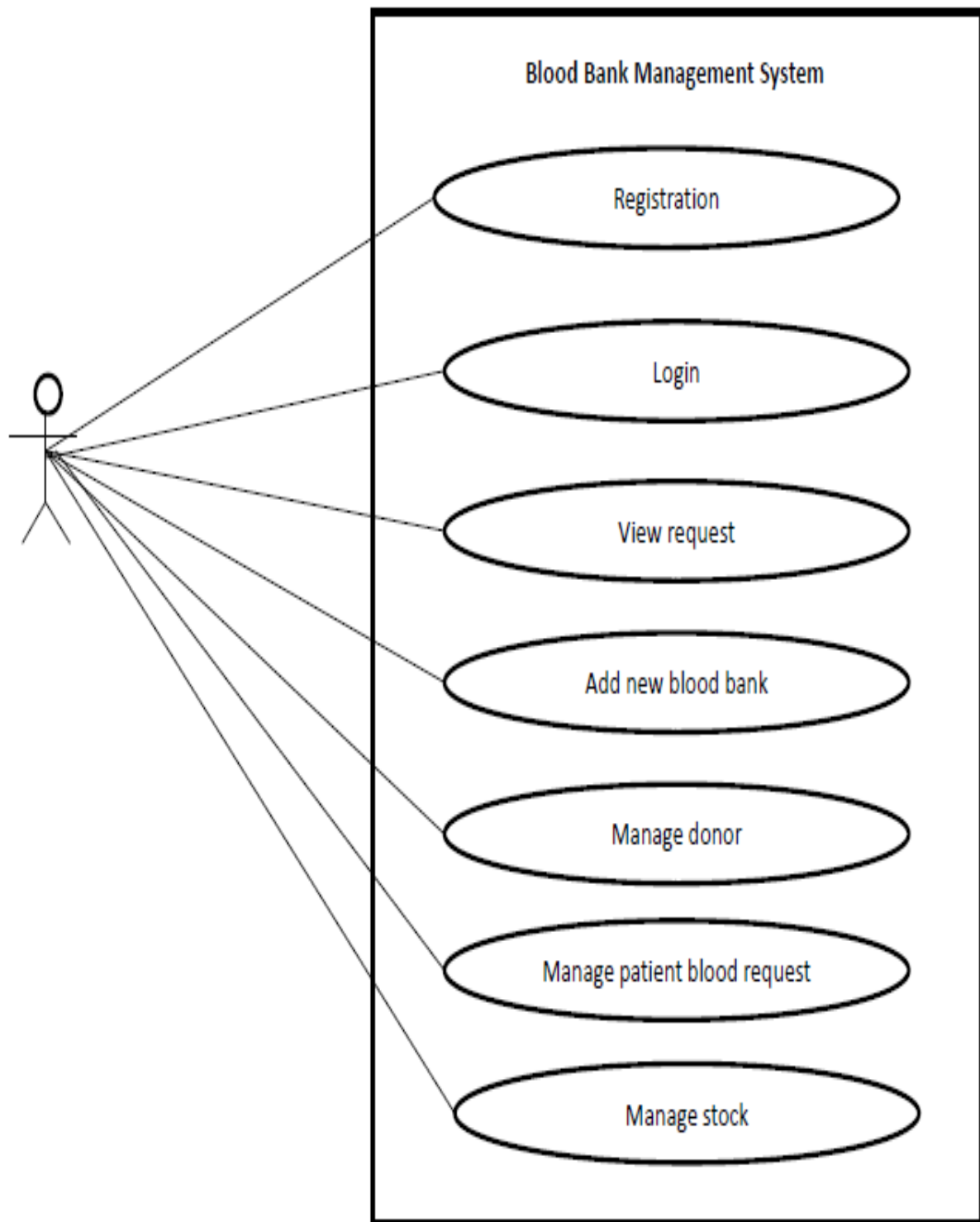


Fig.5.8: Use-Case Diagram for Blood Bank

5.4.2.3 Donor

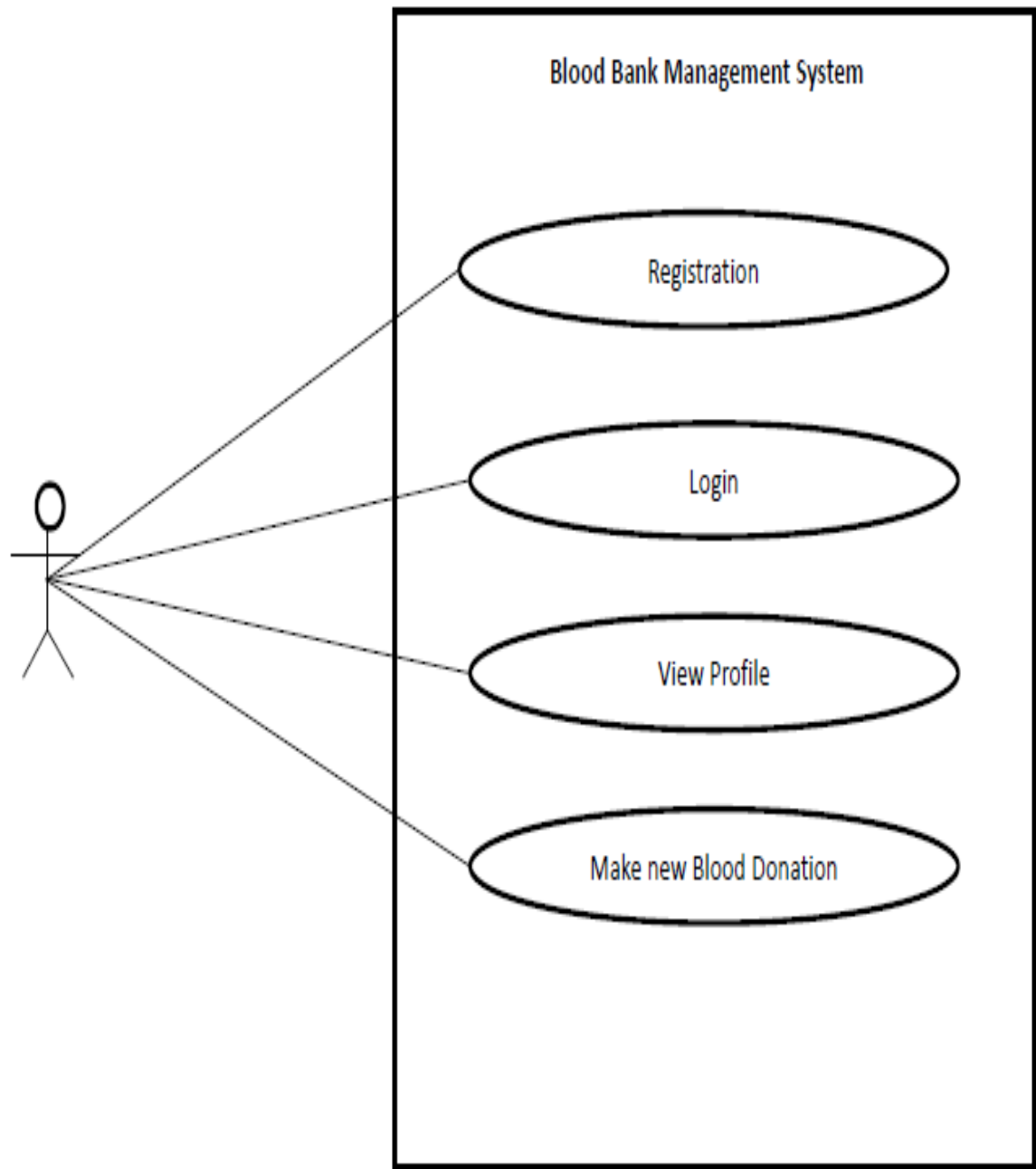


Fig.5.9: Use-Case Diagram for Donor

5.4.2.4 Patient

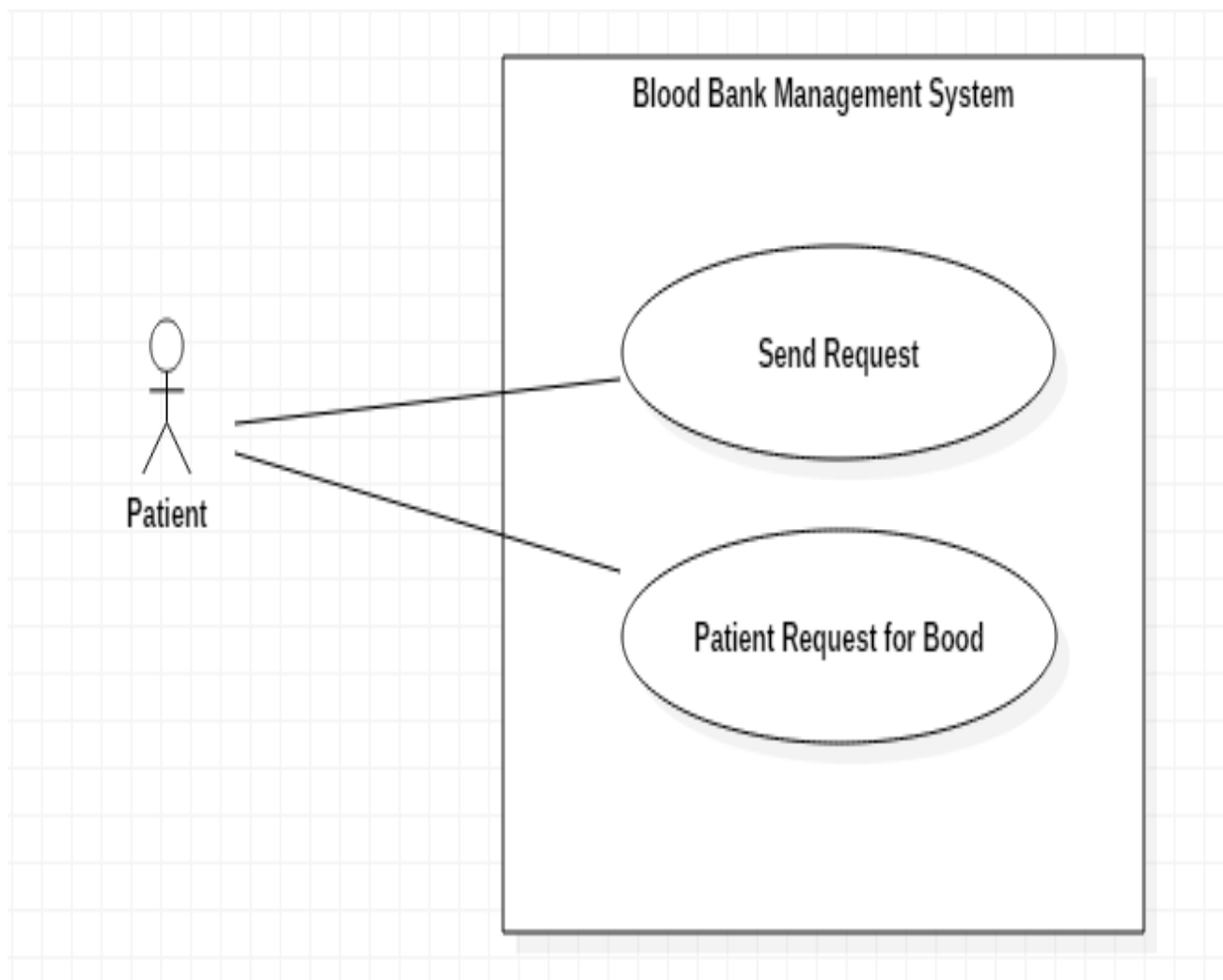


Fig.5.10: Use-Case Diagram for Patient

5.4.3 Sequence Diagram

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.

5.4.3.1 Admin

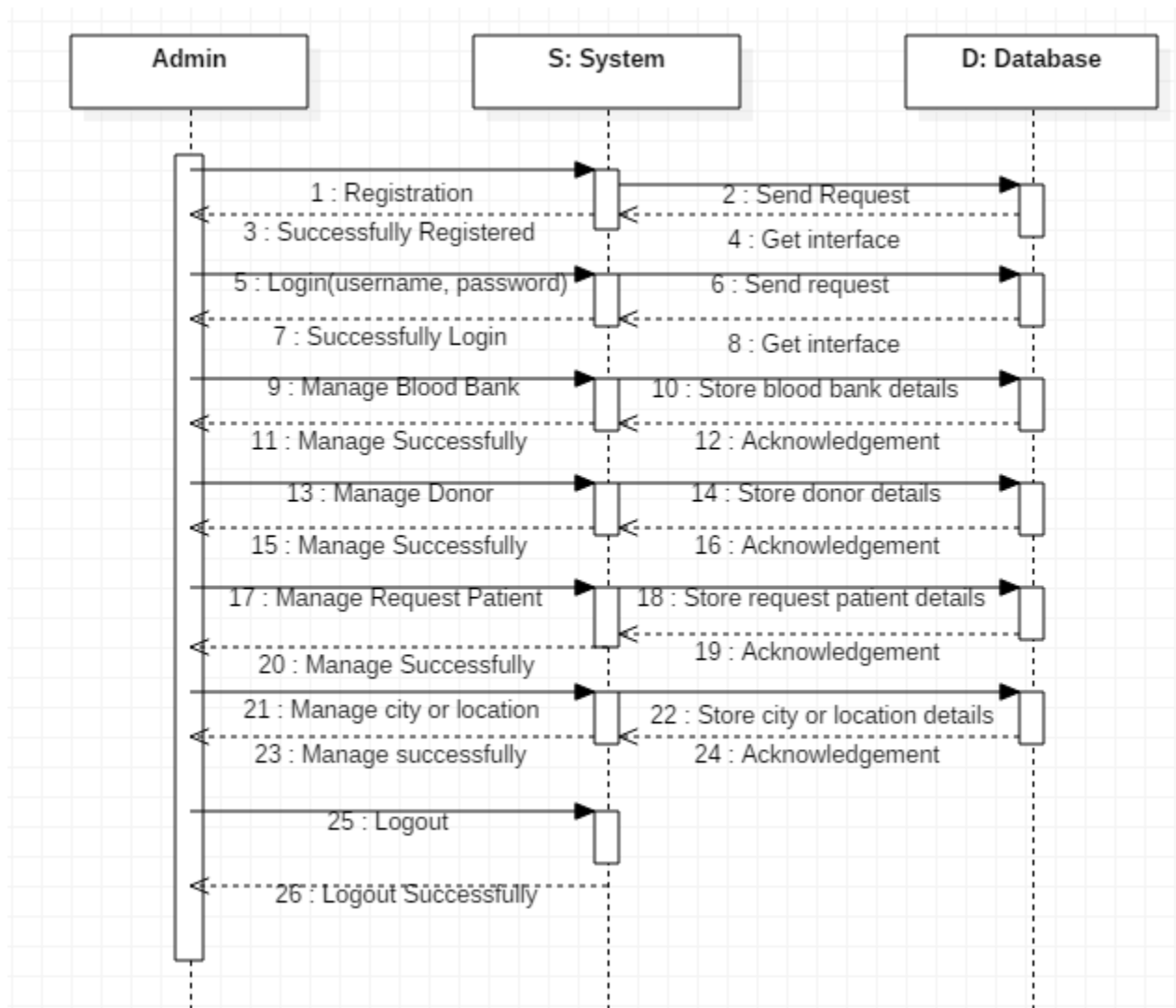


Fig.5.11: Sequence Diagram for Admin

5.4.3.2 Blood Bank

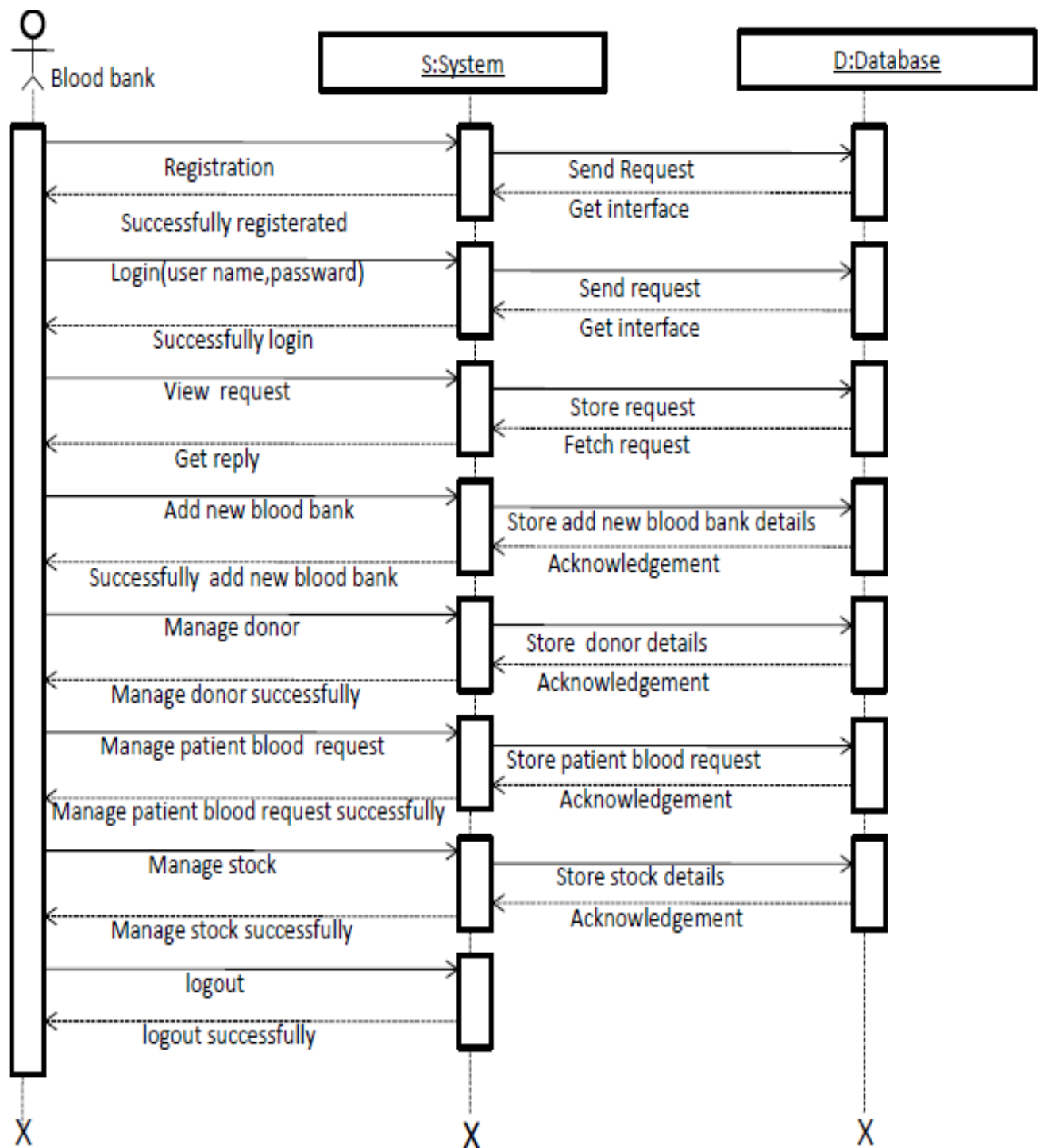


Fig.5.12: Sequence Diagram for Blood Bank

5.4.3.3 Donor

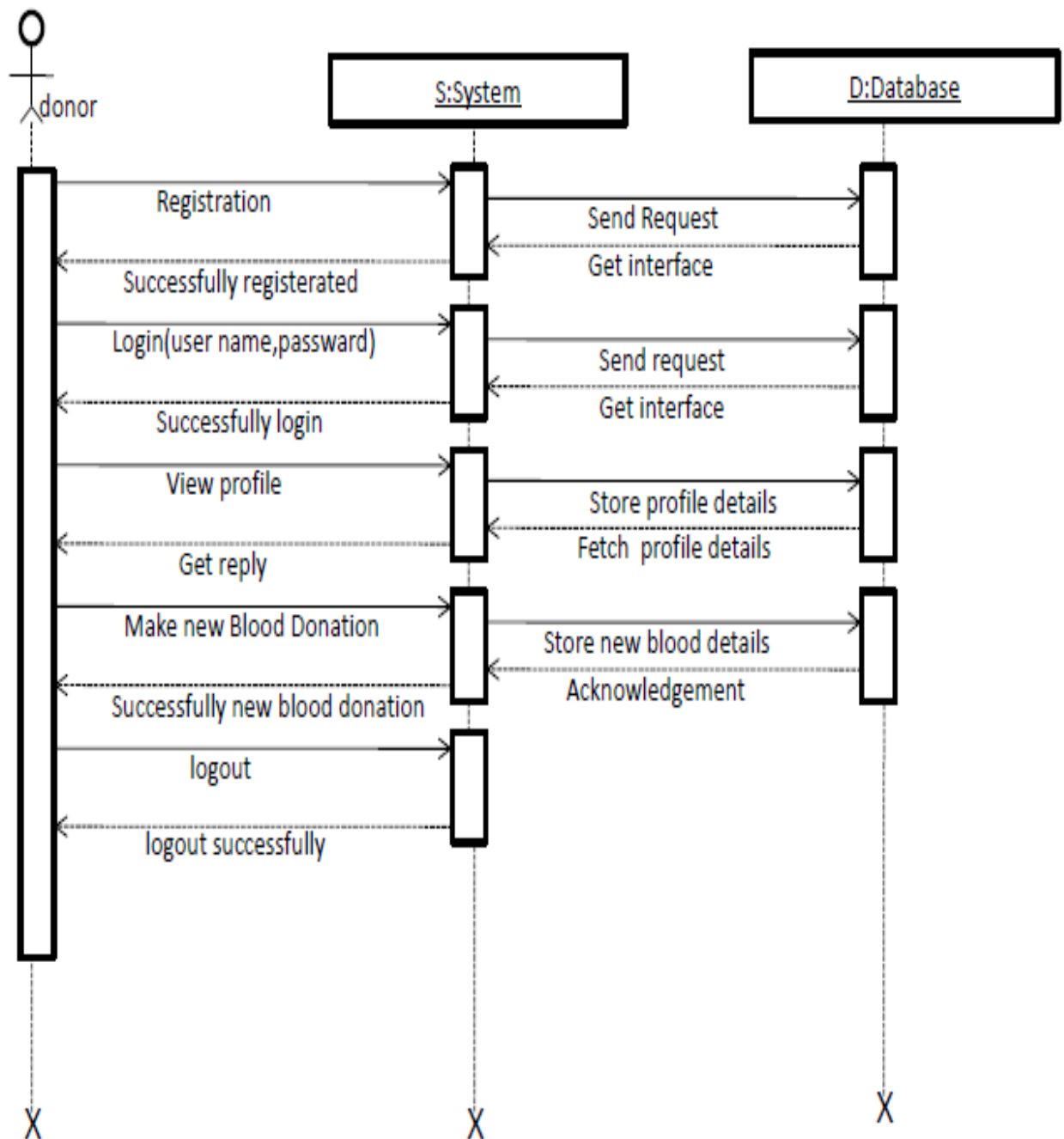


Fig.5.13: Sequence Diagram for Donor

5.4.3.4 Patient

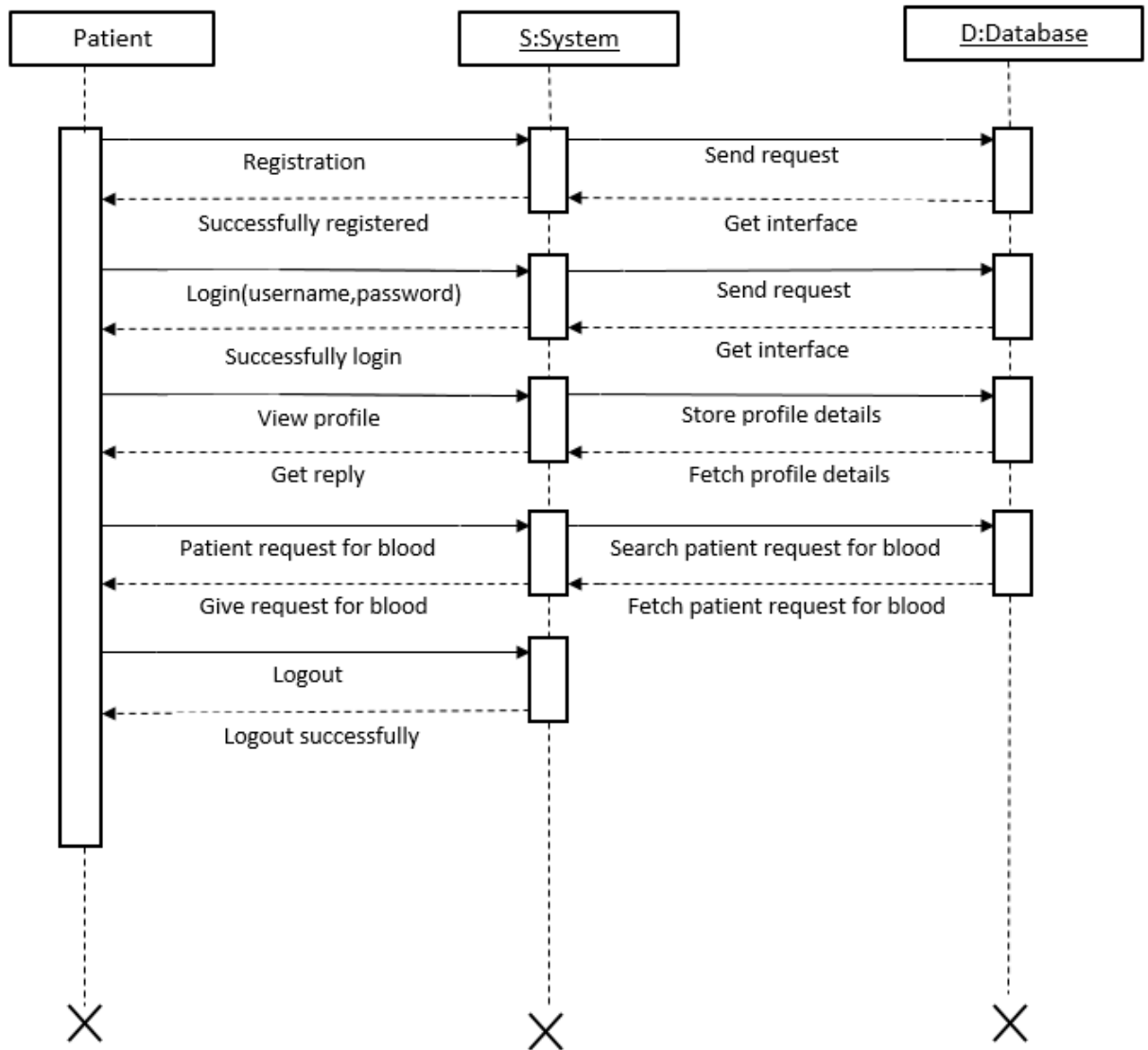


Fig.5.14: Sequence Diagram for Patient

5.4.4 Activity Diagram

Activity diagram is another important diagram in UML to describe dynamic aspects of the system. Activity diagram is basically a flow chart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. So the control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent. Activity diagrams deals with all type of flow control by using different elements like fork, join etc.

5.4.4.1 Admin

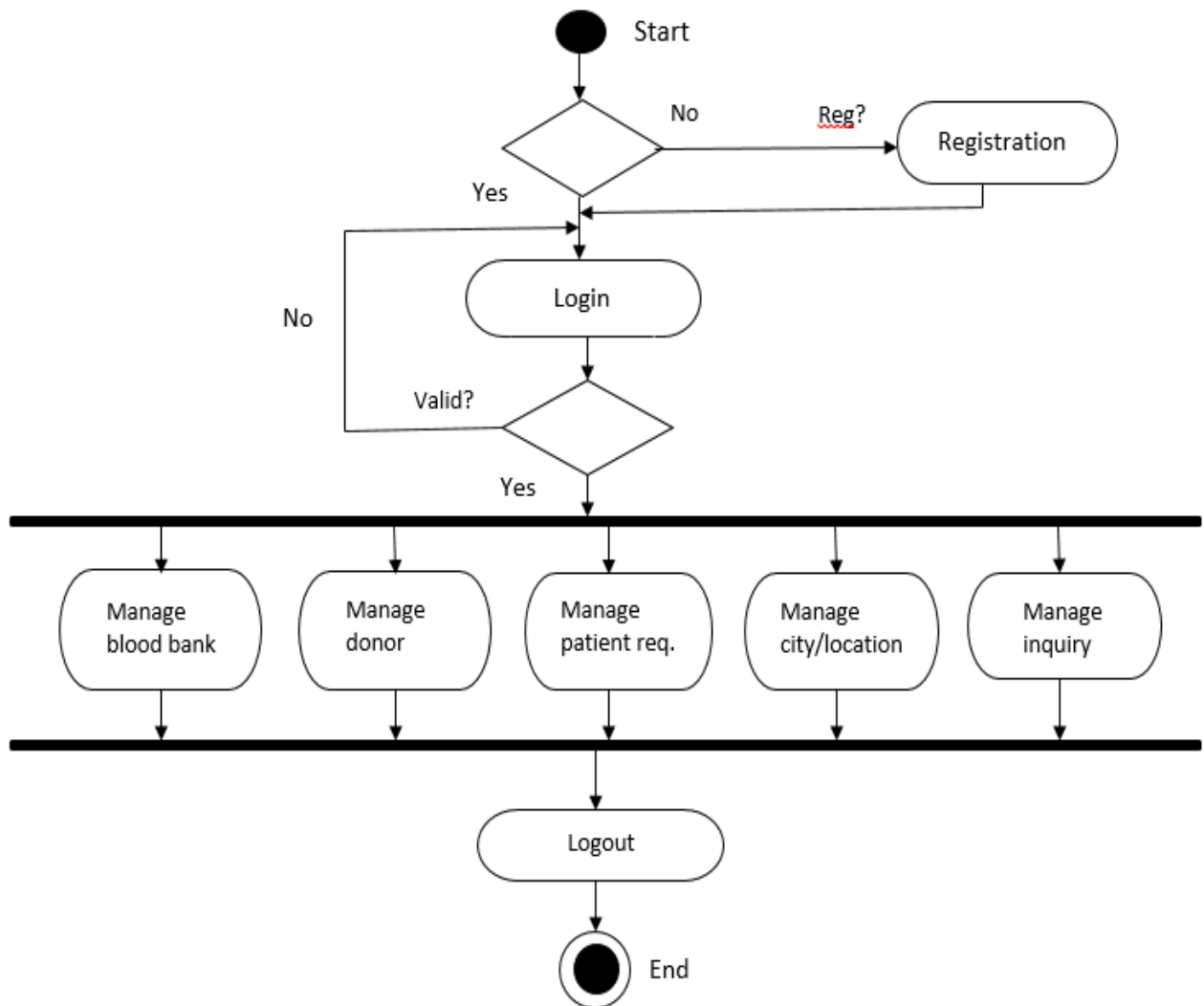


Fig.5.15: Activity Diagram for Admin

5.4.4.2 Blood Bank

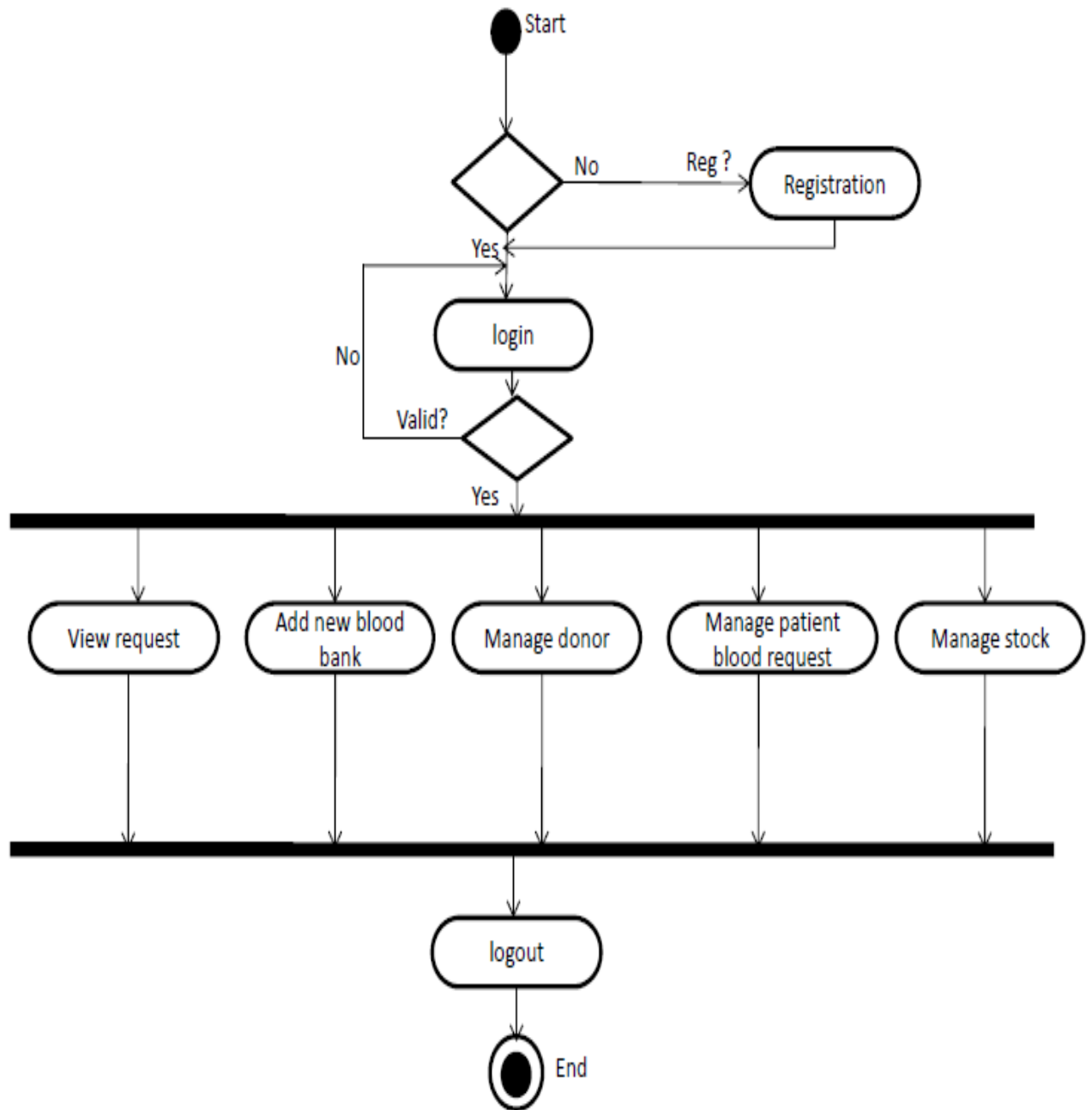


Fig.5.16: Activity Diagram for Blood Bank

5.4.4.3 Donor

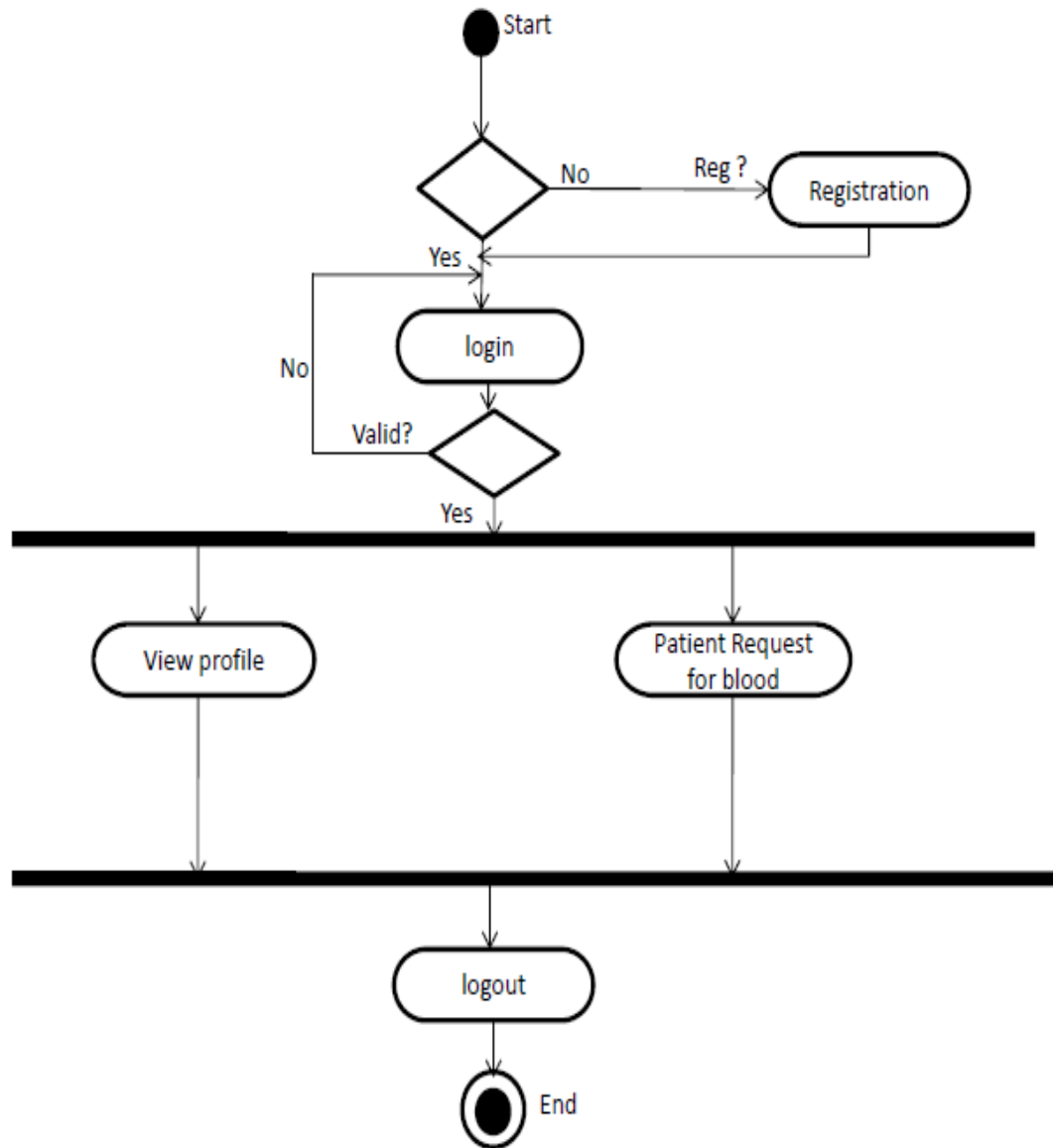


Fig.5.17: Activity Diagram for Donor

5.4.4.4 Patient

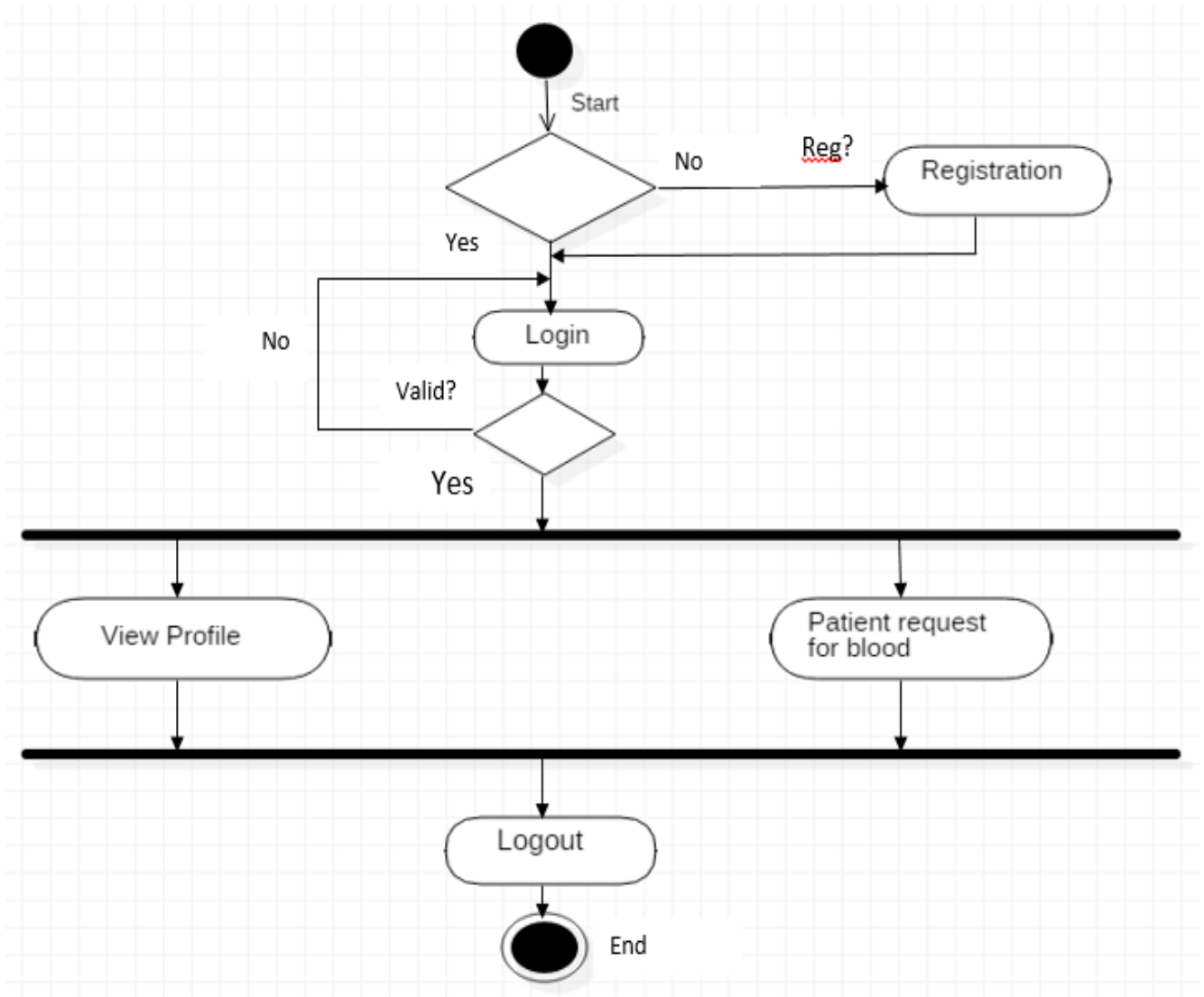


Fig.5.18: Activity Diagram for Patient

5.4.5 Collaboration Diagram

A collaboration diagram, also called a communication diagram or interaction diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). The second interaction diagram is collaboration diagram. It shows the object organization as shown below. Here in collaboration diagram the method call sequence is indicated by some numbering technique. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram.

5.4.5.1 Admin

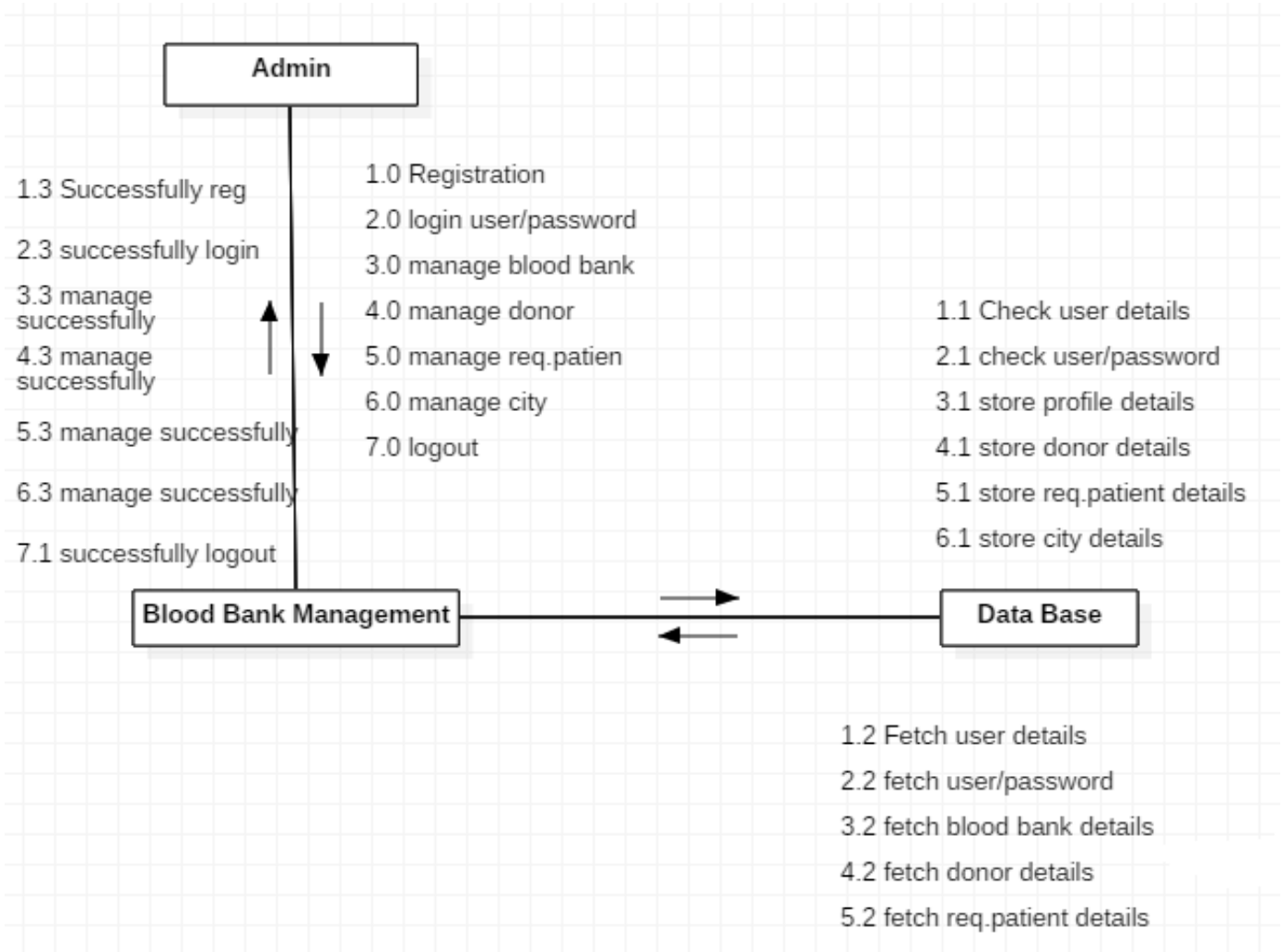


Fig.5.19: Collaboration Diagram for Admin

5.4.5.2 Blood Bank

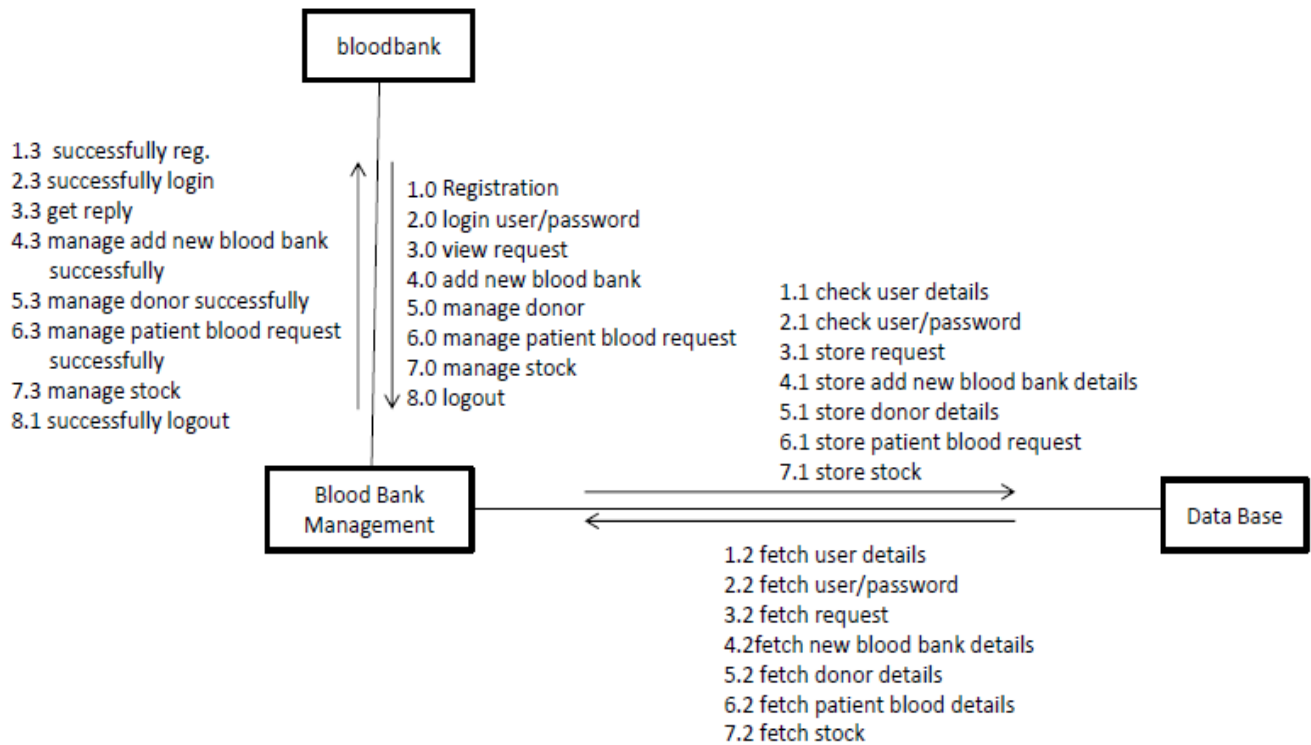


Fig.5.20: Collaboration Diagram for Blood Bank

5.4.5.3 Donor

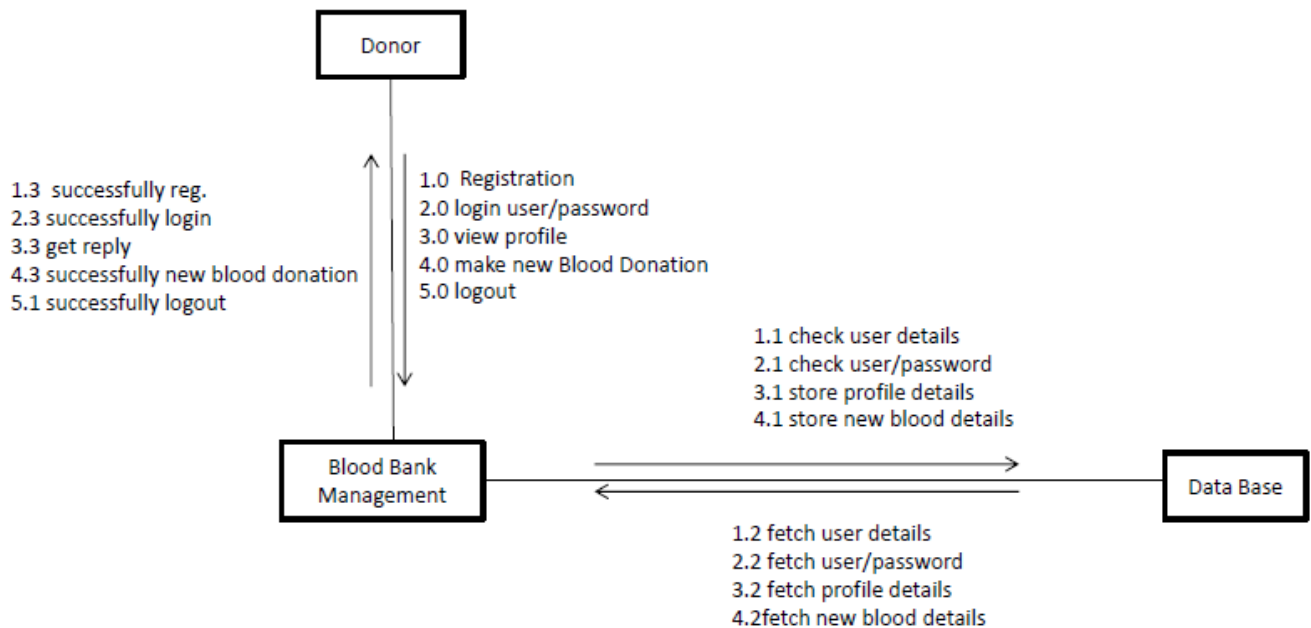


Fig.5.21: Collaboration Diagram for Donor

5.4.5.4 Patient

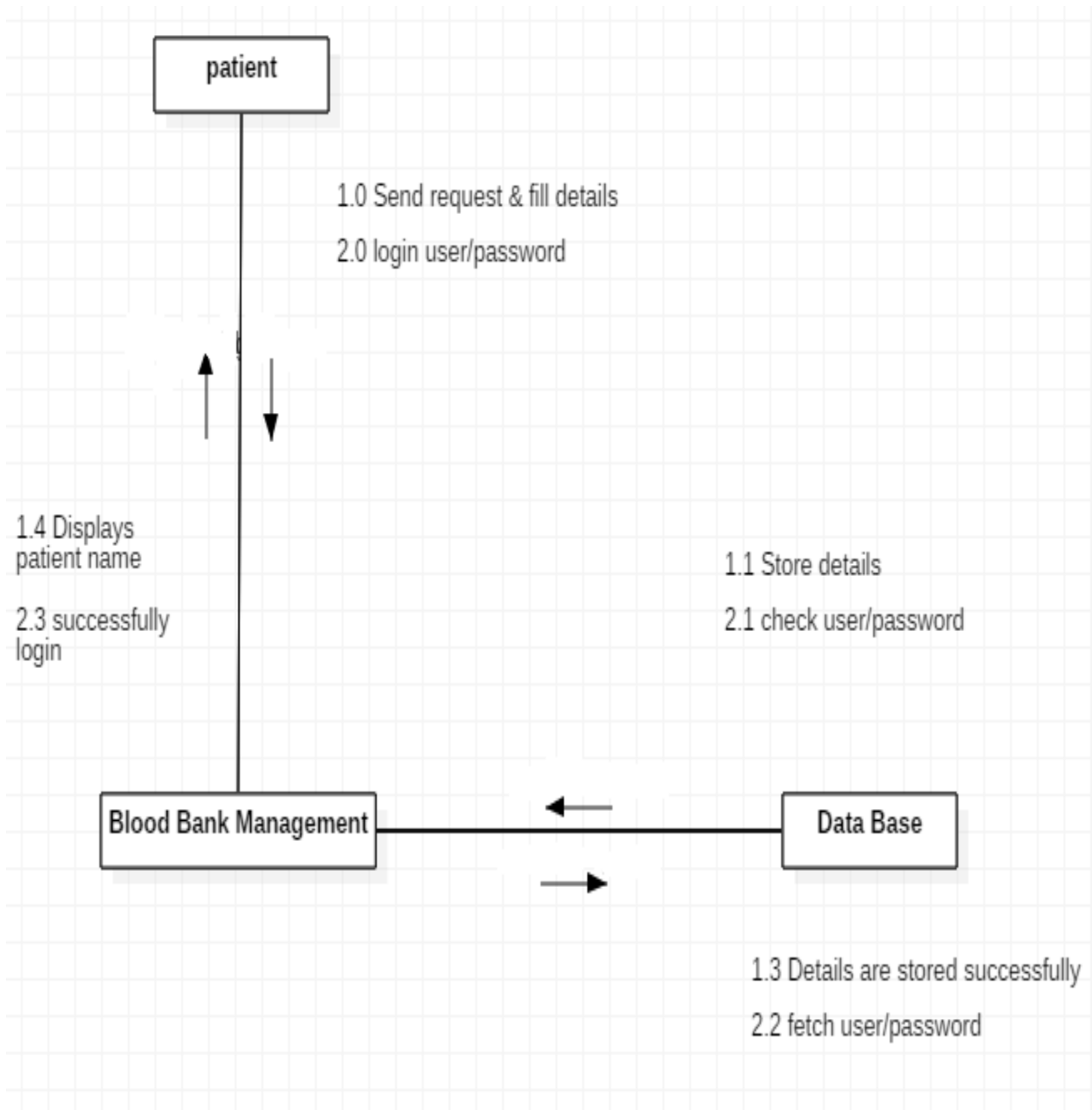


Fig.5.22: Collaboration Diagram for Patient

Chapter 6

Testing

6.1 Donor Login

Table 6.1 Donor Login Test Cases

Sr.no	Test Case	Test Data	Expected Result	Actual Result	Status
1	Whether clicking on submit button without user name and password it allows login or not.	Click on submit button	System does not allow user to login.	System displays message & resume to the same page.	Pass
2	Whether click on submit button with invalid user name and or password it displays the message or not.	Username:abc@gmail.com	It should display message 'Please fill up the username or password'	It displays message	Pass
3	Whether by clicking on submit button with correct username and password it logins or not.	Username:rashmishewale11@gmail.com	System allow user to login	System allow user to access application based on rights given to him.	Pass

6.2 Donor Registration

Table 6.2 Donor Registration Test Cases

Sr.no.	Test Case	Test Data	Expected Result	Actual Result	Status
1	Whether clicking on Donor Registration button with blank field it allows register or not.	null	System does not allow user to register	System displays message & resume to the same page.	Pass

2	Whether click on Clear button it removes all the fields or not.	Name: Rashmi Age:19	It should clear all the fields	It resets	Pass
3	Whether by clicking on Donor Registration button with all correct fields it registers donor or not	Name: Rashmi Age: 19	System should register to database	System allow to register	Pass

6.3 Blood Bank

Table 6.3 Blood Bank Test Cases

Sr.no.	Test Case	Test Data	Expected Result	Actual Result	Status
1	Whether clicking on Login button it login to correct blood bank account or not	Name: Bank of India	It should login to correct account	It Logins to correct account	Pass
2	After login, whether clicking on different menus, the specific pages gets open or not	Click on different menu	It should open the specific page.	It opens the specific page.	Pass
3	In Update Profile menu, after filling the details whether clicking on Update Profile button it gets update or not.	Type: Government License No: 1234	It should update the information.	It updates the information.	Pass
4	Whether in settings menu, after clicking on change password button the password changes or not.	Old Password:***** New Password: ***** Reenter Password: ***	It should change the password.	It changes the password.	Pass
5	After Logout, whether clicking on back button it opens the previous page or not.	Click on logout button	It should not open the previous page.	It doesn't open the previous page.	Pass

6.4 Search (Donor Request)

Table 6.4 Search (Donor Request) Test Cases

Sr.no.	Test Case	Test Data	Expected Result	Actual Result	Status
1	Whether clicking on search menu it displays the search donor form or not	Blood Group: A+ State:Maharashtra	It should display the search donor form	It displays the search donor form	Pass
2	Whether clicking on dropdown button of blood group & state it shows the sub menus or not	Blood Group:A+ A-, B+, B-, O+, O-, AB+, AB-	It should show the sub menus of blood group	It shows the sub menus of blood group	Pass
3	Whether clicking on dropdown button of district it shows the sub menus of that specific selected state or not	District: Nashik, Niphad, Aurangabad etc.	It should show the sub menus of that specific selected state	It show the sub menus of that specific selected state	Pass
4	After clicking on clear button it resets the details or not.	Click on clear button	It should resets the details	It resets the details	Pass

6.5 Admin

Table 6.5 Admin Test Cases

Sr.no	Test Case	Test Data	Expected Result	Actual Result	Pass
1	After verify the bank details from the Blood Group List it either approves or delete the bank	null	It should either approve or delete the bank after verify	It approves or delete the bank after verify	Pass
2	After adding new camp in Add Camp option, the camp displays at the client page or not	Click on add button	It should displays at the client page	It displays at the client page	Pass

6.6 Snapshots

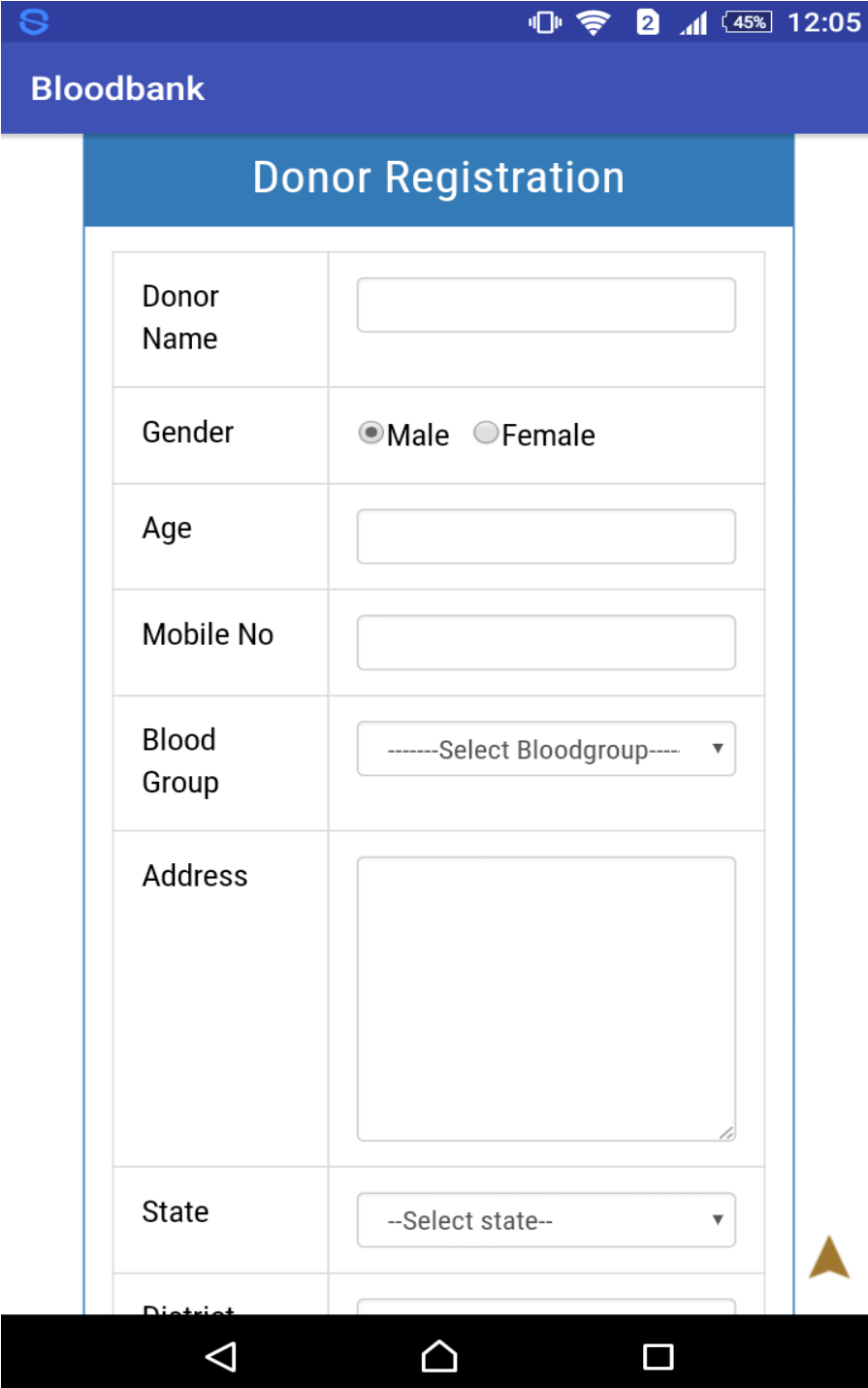
6.6.1 Snapshot of Search (Donor Request)

The image shows a mobile application interface for a Blood Bank Management System. The top status bar displays the time as 12:05, battery level at 45%, and signal strength. The app's header is blue with the text "Bloodbank" and contact information: "+040 185 999" and "mail@example.com". Below the header is a banner with the text "ONLINE BLOOD BANK" and a graphic of a red blood drop. The main content area is titled "SEARCH DONOR" in green. Below this is a "Search" form with a blue header. The form contains four dropdown menus for "Blood Group", "State", "District", and "Taluka", each with a placeholder text like "-----Select Bloodgroup-----". At the bottom of the form are two buttons: a green "Search" button and a red "Clear" button. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps icons.

Search	
Blood Group	-----Select Bloodgroup----- ▼
State	--Select state-- ▼
District	--Select District-- ▼
Taluka	--Taluka-- ▼
<div>Search Clear</div>	

Fig.6.1: Snapshot of Search Option

6.6.2 Snapshot for Donor Registration



The image shows a mobile application interface for a Blood Bank Management System. At the top, there is a status bar with various icons and the time 12:05. Below this is a blue header bar with the text "Bloodbank". The main content area has a blue title bar with the text "Donor Registration". Below the title bar is a form with several fields: "Donor Name" (text input), "Gender" (radio buttons for Male and Female), "Age" (text input), "Mobile No" (text input), "Blood Group" (dropdown menu with "-----Select Bloodgroup-----"), "Address" (text area), "State" (dropdown menu with "--Select state--"), and "District" (text input). The form is displayed on a mobile device screen, with a black navigation bar at the bottom.

Donor Name	<input type="text"/>
Gender	<input checked="" type="radio"/> Male <input type="radio"/> Female
Age	<input type="text"/>
Mobile No	<input type="text"/>
Blood Group	-----Select Bloodgroup----- ▼
Address	<input type="text"/>
State	--Select state-- ▼
District	<input type="text"/>

Fig.6.2: Snapshot of Donor Registration

Chapter 7

Conclusion and Future Scope

7.1 Conclusion

It has been a great pleasure for me to work on this exciting and challenging project. This project proved good for me as it provided practical knowledge of not only programming in PHP web based application and no some extent Windows Application and SQL Server, but also about all handling procedure related with **“Blood Bequeath Federal”**. It also provides knowledge about the latest technology used in developing web enabled application and client server technology that will be great demand in future. This will provide better opportunities and guidance in future in developing projects independently.

7.2 Future Scope

- Through online directory user can search any blood bank data by location.
- User can also message module.
- User can see this website and search blood bank.

Cost

Basic COCOMO compute software development effort as a function of program size. Program size is expressed in estimated thousands of source lines of code ([SLOC](#), [KLOC](#)).^[10]

COCOMO applies to three classes of software projects:

1. Organic projects - "small" teams with "good" experience working with "less than rigid" requirements
2. Semi-detached projects - "medium" teams with mixed experience working with a mix of rigid and less than rigid requirements
3. Embedded projects - developed within a set of "tight" constraints. It is also combination of organic and semi-detached projects.(hardware, software, operational, ...)

The basic COCOMO equations take the form

$$\text{Effort Applied (E)} = a_b(\text{KLOC})^{b_b} \text{ [man-months]}$$

$$\text{Development Time (D)} = c_b(\text{Effort Applied})^{d_b} \text{ [months]}$$

$$\text{People required (P)} = \text{Effort Applied} / \text{Development Time} \text{ [count]}$$

where, KLOC is the estimated number of delivered lines (expressed in thousands) of code for project. The coefficients a_b , b_b , c_b and d_b are given in the following table (note: the values listed below are from the original analysis, with a modern reanalysis^[4] producing different values):

Table: Analysis of values

Software project	a_b	b_b	c_b	d_b
Organic	2.4	1.05	2.5	0.38
Semi-detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

Basic COCOMO is good for quick estimate of software costs. However it does not account for differences in hardware constraints, personnel quality and experience, use of modern tools and techniques, and so on.

This software comes under embedded software project. The calculation is as follows:

$$\text{Effort Applied (E)} = 3.6(7)^{1.20} = 37.18$$

Here, a_b is 3.6 and b_b is 1.20 as mentioned in above table and KLOC i.e. lines of code is 7 which is expressed in thousand.

$$\text{Development Time (D)} = 2.5(37.18)^{0.32} = 7.95$$

Here, c_b is 2.5 and d_b is 0.32 as mentioned in above table.

$$\text{People required (P)} = 37.18/7.95 = 4.67$$

According to this calculation our project require 4 number of people.

Working Hours:

Table: Working Hours

Sr. No.	Work to perform	Date	Duration (In Hours)
1	Selection of Project	Daily	15
2	Flow chart preparation	Daily	15
3	Software Module Development	Daily	20
4	Module Testing	Daily	30
5	Project Report	Daily	20

Costing:

Table: Costing

Main Hours(From the above table)	100
Cost of one man power	Rs. 25/- per Hour
Rent of computer	Rs. 20/- per Hour
Stationary & other Expenses	Rs. 1500/-

Total Cost:

Table: Total Cost

Cost of total man power(100×25)	Rs. 2500/-
Total rent for computer for 9 months(100×20)	Rs. 7000/-
Stationary & other expenses	Rs. 1500/-
Total	Rs. 11,000/-

Total cost of the project is Rs.11,000.

Result

Result includes the information about how the operation executes and displays the result by giving various inputs. In this case after log in with specific user name and password the user gets login and the donor panel gets open in which the user can update profile, view request,etc.



DONOR LOGIN

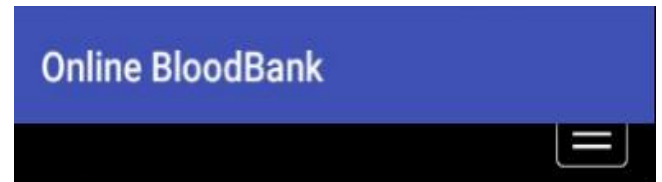
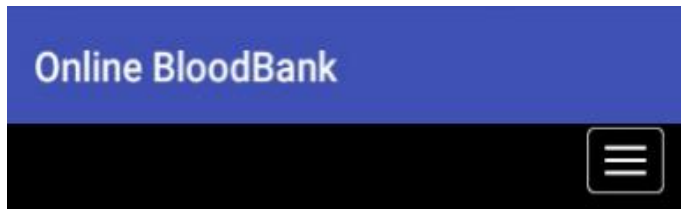
Fig. Screenshot of Donor Login



DONOR REGISTRATION

Fig. Screenshot after login

In the following screenshot of search option the user can search for blood group by various states, districts and taluka's which helps user to find the another donor area wise. Finding the donor by state/district/taluka wise helps user to find the donor fastly. Whereas in another screenshot after searching it shows the various donors with their detailed information.



Search	
Blood Group	O+ ▼
State	Maharashtra ▼
District	Nashik ▼
Taluka	Nashik ▼
<div>Search</div> <div>Clear</div>	

Request For Blood			
Sr	Name	Gender	Age
1	Aishwarya Sonawane	Female	18
2	Pranita	Female	20
3	korde	Male	20
4	Pranita	Female	19

Fig. Screenshot of search option

Fig. Screenshot after searching

REFERENCES

- [1] URL Address (https://www.google.co.in/?gfe_rd=cr&ei=dRvaWITjL9eR2ASs5ImQBw#q=w3schools&*)
- [2] URL Address (<http://getbootstrap.com/>)
- [3] URL Address (https://en.wikipedia.org/wiki/Centralized_database)
- [4] URL Address (https://en.wikipedia.org/wiki/Distributed_computing)
- [5] URL Address (https://en.wikipedia.org/wiki/Client-server_model)
- [6] URL Address (<http://searchstorage.techtarget.com/definition/storage>)
- [7] App “National Blood Banks Directory” on play store
- [8] App “Nepal Blood Donors” on play store
- [9] URL Address (https://en.wikipedia.org/wiki/Computing_platform)
- [10] URL Address (<https://en.wikipedia.org/wiki/COCOMO>)

Appendix: I

SRS-System requirement Specification

DDS - Design Document Specification.

UAT- User acceptance testing.

SDLC-Software Development Life Cycle

ER-Entity Relationship

DFD-Data Flow Diagram

UML-Unified Modelling Language

OMG- Object Management Group.

PHP-Pre-Hypertext Processor

SQL-Structure Query Language

Appendix: II

Problem Statement:

To develop an android app for Blood Bank Management System.

Abstract:-

This project is aimed to developing an online Blood Donation Information. The entire project has been developed keeping in view of the distributed client server computing technology, in mind. The Blood Donation Agent is to create an e-information about the donor and organization that are related to donating the blood. Through this application any person who is interested in donating the blood can register himself in the same way if any organization wants to register itself with this site that can also register. More over if any general consumer wants to make request blood online he can also take the help of this site. Admin is the main authority who can do addition, deletion, and modification if required. The project has been planned to be having the view of distributed architecture, with centralized storage of the database. The application for the storage of the data has been planned.

Literature Survey:

Blood Donation app ‘National Blood Banks Directory’ was developed in the year 2016 for the purpose of donating and receiving the blood. This app contains state/city wise list of Blood banks and other required information like address, geo-location, contact details, area pin-code, email address, website link, etc. But it is suitable for only particular area and it connects receiver to only donors or bank.

The app ‘Blood Donation’ works as above app but requires email ID and the password which loses security. Some apps have location tracking system which may not be useful when donor’s GPS is off. Some apps are useful for particular area not universally.

Objective & Scope of Proposed System:

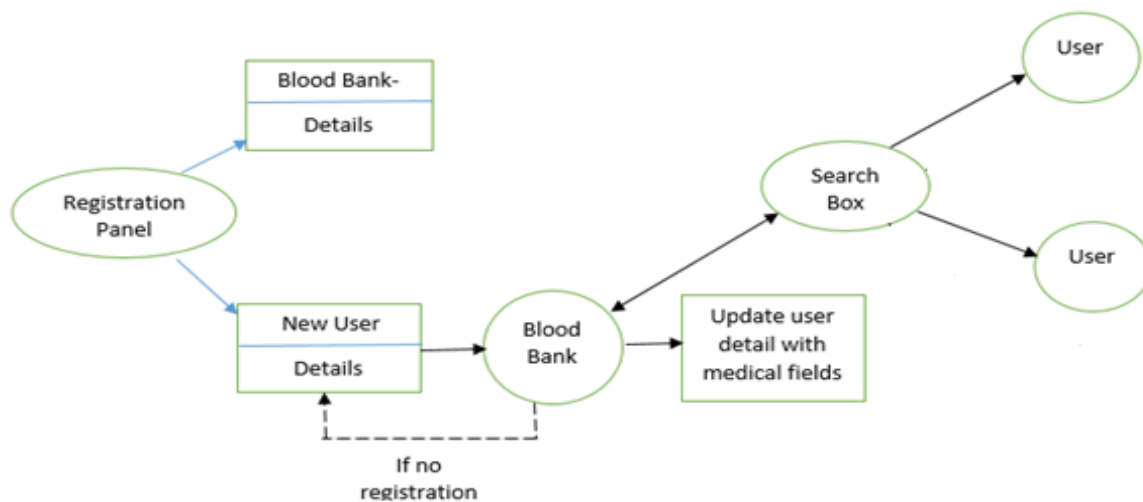
- 1) This project offers user to enter the data through simple and interactive forms.
- 2) The user is mainly more concerned about the validity of the data, whatever he is entering.
- 3) User is provided the option of monitoring the records he entered earlier.
- 4) Data storage and retrieval will become faster and easier to maintain because data is stored in a systematic manner and in a single database.
- 5) Easier and faster data transfer through latest technology associated with the computer and communication.
- 6) Through these features it will increase the efficiency, accuracy and transparency.

Proposed System:

The proposed Blood Bank management system helps the people who are in need of a blood by giving them all details of blood group availability or regarding the donors with the same blood group.

The people in need of blood can search for the donors by giving their blood group and city name. It saves time as he can search donors online without going anywhere. Using this system user can get blood in time and can save his relative or friend life. Our website work 24x7 so user can get information of blood donor any time. Blood donor can also get registered and save life of other person. The main benefit of this system is the information of available blood group. When blood is need in the operation then people have very less time to get the blood available so if he get the information like who can give him blood in time in his city is lifesaving. And here our system work, whenever a person need blood he get information of the person who has the same blood group he needs.

Block Diagram of the Proposed System:



Advantages & Disadvantages:

Advantages:

- 1) This is very helpful for the client to enter the desired information through so much simplicity.
- 2) There are checks on every stages of any new creation, data entry or updation so that the user cannot enter the invalid data, which can create problems at later date.
- 3) We can say that the project is user friendly which is one of the primary concerns of any good project.+
- 4) Allocating of sample results becomes much faster because at a time the user can see the records of last years.

Disadvantages:

We can implement it in future scope.

Application Areas:

- 1) Blood Donation Camp & Camp Organizer Management.
- 2) Blood requisition and issuance of blood.
- 3) Online transfer of blood from one blood bank to another.
- 4) Donor Management - Donor Registration, managing donor database, recording their physical and medical statistics.
- 5) Camp Wise Donor List and Printing of Donor Cards

Software, Hardware & Test Data Requirements:**Software Requirements:**

Table Software Requirements

Platform	Windows XP
Language	PHP
Mobile Client	Android
IDE/Tool	Eclipse IDE

Hardware Requirements:

Table Hardware Requirements

Processor	Pentium IV
RAM	64 MB
Storage	20 GB
Mobile Phone	Android Phone
Monitor	15"

Project Plan:

Table Project Plan

Month	Schedule	Project task
August	1 st Week	Project Inception
	3 rd Week	Project Feasibility Check
	4 th Week	Project Information Collection
September	1 st Week	Literature Survey
	2 nd Week	Report Writing
	3 rd & 4 th Week	To Document Abstract of Project
October	1 st & 2 nd Week	To Document Software Requirement Specification
	3 rd Week	Designing & Modelling
	4 th Week	To Integrate all modules of Project
December	3 rd Week	To implement Coding for Project
	4 th Week	Unit Testing & Integration Testing
January	1 st Week	To implement GUI interface
	2 nd Week	UML diagrams
February	1 st Week	Fixing any Bugs found
	2 nd Week	Conclusion

References

- [1] URL Address (https://www.google.co.in/?gfe_rd=cr&ei=dRvaWITjL9eR2ASs5ImQBw#q=w3schools&*)
- [2] URL Address (<http://getbootstrap.com/>)
- [3] URL Address (https://en.wikipedia.org/wiki/Centralized_database)
- [4] URL Address (https://en.wikipedia.org/wiki/Distributed_computing)
- [5] URL Address (https://en.wikipedia.org/wiki/Client-server_model)
- [6] URL Address (<http://searchstorage.techtarget.com/definition/storage>)
- [7] URL Address (https://en.wikipedia.org/wiki/Computing_platform)