WEB BASED BLOOD BANK RECORD MANAGEMENT SYSTEM FOR DEBREMARKOS TOWN

A PROJECT REPORT

Submitted by

1. **KEBADU DAGNE**   **TER/4669/07**
2. **BINALFEW DEBAS**   **TER/4652/07**
3. **ENDALEW BIAZEN**   **TER/4659/07**
4. **SEWAREG ALEMU**   **TER/4685/07**

In partial fulfillment for the award of the degree of

BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY

Under the guidance of

NIGUSSIE ABATE (MSC)

ADVISOR SIGNATURE

DEPARTMENT OF INFORMATION TECHNOLOGY

INSTITUTE OF TECHNOLOGY

DEBRE MARKOS UNIVERSITY

DEBRE MARKOS, ETHIOPIA

JANUARY 17, 2010 E.C

CERTIFICATE

This is to certify that the project entitled "web based blood bank management system" has been carried out by: Kebadu Dagne, Binalfew Debas, Endalew Biazen, and Sewareg Alemu under the guidance and supervision of Nigussie Abate, for the fulfillment of the requirement for the award of degree of Bachelor science in Department Information Technology.

Project Advisor: Head of Department:

INS:-Nigussie Abate.   Ms.Ayalneh Betew.

Signature: _____   Signature: _____

Examiners: Name: Signature:

    1. _____   _____

    2. _____   _____

    3. _____   _____

    4. _____   _____

    5. _____   _____

    6. _____   _____

Contents

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENT

ABSTRACT

The intension of this project is developing a supplemental web based blood bank system for Debremarkos town.A blood bank information management system holds information about blood groups, blood donors and keeps samples of blood gathered as a result of blood donation and the expired blood types.

However, the current system is limited on manual work. This has limitation on controlling the work securely, for declaring the result on time, and has a high consumption on resources.Web based Blood Bank management System is an online portal to facilitate the co-ordination between supply and demand of blood. The whole purpose of the project is bringing an online edge for blood donors and patients (blood seekers) and to create an interactive way of bringing the Blood donors and Blood requesters.

The system gives a lot of information about Blood groups, donation methods, frequency of blood donation and the details of the coming blood donation camps. Besides, extra features in the system such as security protection by using a password, Generating reports, reminders of blood stock shortage and workflow tracking can even enhance the efficiency of the management in the blood banks.

Abbreviation and definition

Abbreviation	Description	Abbreviation	Description
WBBBms	Web Based Blood Bank Management System	Etc.	Etcetera
DM	Debremarkos	HBV	Hepatitis B Virus
ABO	Blood Types A,B And O	HCV	Hepatitis C Virus
AID	Admin Identification	HIV	Human Immunodeficiency
BGroup	Blood Group	NBTS	National Blood Transfusion Services
BWeight	Body Weight	HTML	Hypertext Markup Language
BType	Blood Type	ID	Identification
CSS	Cascading Style Sheets	Lab	Laboratory
DBMS	Database Management System	NACO	National Aids Control
DID	Donor Identification	Pack no.	Pack Number
DOB	Date Of Birth	PHP	Hypertext Preprocessor
DOReg	Date Of Registration	RID	Recipient Identification
DOReq	Date Of Request	SDLC	Software Development Life Cycle
DsID	Disease Identification	SID	Staff Identification
E.C	Ethiopian Colander	G.C	Gregorian Calendar
BR	Business Rule		

Table 1: Abbreviation and Definition

CHAPTER ONE

INTRODUCTION

Introduction

Blood is a specialized body fluid in humans that delivers necessary substances such as nutrients and oxygen to the cells and transports metabolic waste products away from those same cells. Human blood is an element of human life. [1]

The National Blood Transfusion Services (NBTS) was established in 1969 by Ethiopian Red Cross society since 2004 it has been transferred to Federal Ministry of Health Ethiopia, and entrusted with the responsibility of managing the Blood donors, collection, testing and transfusion of blood and blood products in Ethiopia. Its main center is located in Addis Ababa and it has also the responsibility to oversee, support and monitor the activities of regional blood bank in the country which are administratively under their respective regional health bureaus. Blood banks in Ethiopia have one main place in Addis Ababa and it has many sub branches around all Ethiopia. [2]

Debremarkos blood bank is one of these and it was established by the federal government and Amara regional state since in 2005 E.C This organization Provide blood for 14 client hospitals, these client hospitals Receive blood by full filling certain forms and getting permission from blood bank's manager. This organization was begun giving small capacity of blood service. The aim of the organization is to provide efficient service to user, to increase the capacity for providing blood to the recipient, to increases the number of blood donors by teaching the society and facilitating blood donating services. From time to time the capacity of this bank is increasing but the bank was working with manual system and this makes the employee to fail with data redundancy and erroneous data storing. Generally this organization has established to save the life of people who are affected by these problems like accidents, cancer, sickle cell, premature surgery. [7]

In Debremarkos Town donating and seeking blood increases from time to time, there has been a good rise in the number of people who donate blood but the system is paper based, therefore we are very interested to computerize the system "Web Based Blood Bank management system for Debremarkos Town (WBBBms)".

1.2. Background of the project

The project that is going to be developed about Web based blood bank management system for Debremarkos town. The main purpose of the project is to handle and maintain Web Based blood bank management system and also provide efficient transfusion services.

The system manages the blood transfusion process starting from donor registration, storing blood in blood stock and distribution of blood for the client hospitals.

This project is intended to provide information about the availability of blood in emergency conditions their respective locations. Reservation date of Blood donation through online and maintaining the status of donors for Gathering Volunteers for blood bank and hospitals send request to get blood from blood bank. This project can manage blood types, quantity and expiry dates for each category of blood. The System shows the total amount of blood inside the stock for efficient management of blood bank. The system is mainly going to develop web based blood donor reservation, blood expiration management, and fair distribution of blood to their client and manage the system as whole.

The web based database application system will be used for the web based blood bank management systems a means to advertise the blood donation events to the public in order to rise up the public awareness on the events and at the same time allows the public to make online reservation on their desire session, This project has a plane to automate the blood and donor management system for the Debremarkos blood bank management system to produce better and efficient record management.

1.3. Problem Statement

In the current system documenting, writing, finding and searching of the specific information of the blood bank is done manually. Employer cannot manage the blood bank efficiently starting from the donor registration, blood screening, processing, and storage and distribution information. Moreover, there is no logging function available So that each process or workflow cannot be traced from the database. The current system cannot shows the expired date of blood, difficult for making comments about blood bank services, it becomes tedious for a seeker to search blood in case of emergency, and it is difficult to know availability of blood in stock. These types of system make the workers to document erroneous and redundancy information. The current system is also consumed the time of worker for completing specific task.

The need to improve performance and the urgency to solve the above stated problems contribute to undertake this project into consideration.

1.4. Objectives of the project

1.4.1. General Objectives

The main objective of this project is to develop a web based Blood Bank Management System to facilitate the blood transfusion process.

1.4.2. Specific Objective

To achieve the general objective mentioned above the following are specific objective:

➢ To allow the public to make online reservation.

➢ Enable seekers to see available blood online at home without going far apart.

➢ Enable seekers to know the detail information about each blood type   in their home.

➢ To provide authentic and authorized features to the system where private and confidential data can only be viewed by authorized user.

➢ To improve the efficiency of the blood store management by alerting the blood store manager when the drug expired.

➢ To improve information sharing between the organization and the users.

➢ To provide easy way of commenting the system.

➢ To avoid multiple records of the same data.

1.5. Scope of the Project

Blood bank management system has much functionality. But this project only focus on the information handling activities related to blood donors and blood distribution, managing the collected blood information's as well as how to distribute blood from the blood bank to different Clients such as hospitals.

The scope of proposed system will focus on the following main tasks

✓ The system used to register, update, view and block users

✓ Generating general report.

✓ Creating account for workers

✓ Advertising the organization services.

✓ Shows accessibilities and availabilities of blood inside the stock by their blood group.

✓ Recording blood components and blood donation.

✓ Handling information with related to blood donors, seekers and distribution information.

✓ Managing the collected blood data and distribute blood from the blood Bank to different place such as hospital and other health centers**.**

✓ Store screened blood information and discard expired blood.

1.6. Limitation of the project

Some of the main limitations of the project are as follows:

➢ The system does not support languages other than English language.

➢ The system cannot help people with visual impairment.

➢ The system is accessible when the internet is available.

➢ The system can't be receive or donate blood online.

1.7. Significance of the Project

The current system is not using computerized data processing system. So making the system web based will give benefits from the following four aspects.

♥ **For Blood donors**

  ➢ It provides the unique identification number easily at the time of blood donation camp which helps the user for the future correspondence.
  ➢ Donors can view the blood donation camp organizing at different places.
  ➢ Donor can check the status of the particular blood group just on one click sitting at home**.**
  ➢ Donors can make reservation Online.

♥ **For blood seekers**

  ➢ Seeker can get the information of the desired blood group from the central inventory.
  ➢ Seeker can see available Blood on the database.
  ➢ Seeker can get the information of the blood groups which is or not fit for blood transfusion inquests.
  ➢ Seekers can save time, effort and money.

♥ **For Blood bank center**

  ➢ The camp is getting rid from manual procedure. Now they to do the entries in the information system.
  ➢ The probability of error should be minimal.
  ➢ Information retrieval should be precise and effective.
  ➢ Report about donors, seekers, total consumption of the blood units and overall report can be generated.
  ➢ Can get the information which is maximum cause for which the blood units are required such as accidental cases, heart surgery, delivery cases etc.
  ➢ Can view the list of discarded blood units, they can also view the reason for which the blood units are discarded.

♡ **For The Developers**

- ➢ While developing the system
  - ✓ Developer's team and skills will be improved knowledge how to conduct and prepare a web based system.
  - ✓ Developer's team problem solving skill will be increase.

1.8. System Requirements

For the purpose of the development of this project the team members used different software and hardware tools which can be identified as hardware requirement and software requirement.

1.8.1 Hardware Requirement

This project used the following hardware requirements. The following hardware requirements are needed at minimum to develop the project

- ➢ Computer: used to write proposal, documentation, develop the system. The computer team project used has the following specifications:-
  - ✓ Processor–core i3dual core @ 2.4 GHz.
  - ✓ RAM-4 GB.
  - ✓ Hard Drive 220 GB.
  - ✓ Monitor LCD-Desktop.
- ➢ Flash: to store data.
- ➢ Disks (CD, DVD):-necessary for the movement of relevant data and for backup and recovery mechanism.
- ➢ Printer: to print documentations.
- ➢ Stationeries (pen, paper):for writing all necessary documentations associated with the project
- ➢ Note book:-to take notes during data collection and for other documentations.

1.8.2. Software Requirement

Software Requirements are descriptions of the services that a software system must provide and the constraints under which it must operate. Since; there are many software tools for developing any projects. This system or project also used much software from start to end.

- ✓ Wamp server: to provide MySQL for creating and manipulating databases and PHP to design user interface from the front end of software.
- ✓ It also provides the latest releases of php MyAdmin, HTML, CSS and JAVA SCRIPT embedded inside it.
- ✓ Microsoft office 2010 and Notepad++: to write on any necessary documents about the project.
- ✓ Paint: for editing.
- ✓ Web Browser:-is Language Interpreter that used to understand client side application.
- ✓ Anti-Virus Software: -used to keep secure, scan, fix Flash Disk and to prevent data destruction and corruption
- ✓ Microsoft Office Visio 2007: to design Sequence Diagram, Class Diagram Activity diagram and Use case Diagram of our system.
- ✓ Adobe Photoshop: for editing images.

1.8.3. Programming Language

Now a day's many programming languages are used to develop many projects. But we select the PHP programming language due to the following reason:-

- ➢ PHP: Hypertext Preprocessor is a widely used, general-purpose scripting language that was originally designed for web development to produce dynamic web pages.
- ➢ Easy to understand-When compared with other scripting languages, PHP can be understood easily because it has simple techniques and features.

- ➢ Integration-it is easy to integrate popular web applications using this scripting language.
    - ✓ Database tool: MYSQL
- ➢ Because of its unique storage engine architecture MYSQL performance is very high.

➢ Most of the time MYSQL is compatible with PHP programming language and MYSQL portable.

➢ We are familiar with MYSQL, so we select it to manage database system.

➢ Generally PHP is Clear and easy to understand, OS independent, Easier to fix problems, operates much faster than other scripting languages, Easy to learn and Open source.

❖ Additional programming languages:-

➢ CSS:-for styling the web page.

➢ Ajax:-For the client-side browser to communicate with the server (for example: retrieve data from a database) without having to perform a page refresh.

➢ HTML:-to display content.

➢ Java script: for client side scripting (interpreted by the browser).

1.9. Methodology

1.9.1. Data collection approaches

Data source for this project is the office of Debremarkos blood bank and representatives of the office. We have used different methods to collect data. Data collection is the most important part of the project to find the main requirement of the system and to understand how the system does.

❖ **Interviews**:-This is one of the methods used for the collection of data in which the project designers are asking different questions to Debremarkos blood bank organization Manager, Ato Kefale Gebeyehu, and employees for obtaining the required information and data.

❖ **Observation**:-This is another type of method for collecting data and information in which could witness the actual events which will happen in the organization. In this method all team members have observed and note down the events from that observation.

❖ **Document Analysis:-**The team member also collected certain relevant information from written documents in the blood bank. Not only that but also we tried to review other relevant documents to develop this project.

1.9.2 The system analysis and design approaches

The goal of this section is to provide the basic overview of the system that we are going to develop. The system analysis and design approaches for this project we used the object oriented system analysis & design. Because

- ➢ It provides code and function reuse through the concepts of inheritance, polymorphism, encapsulation, modularity, coupling and cohesion.
- ➢ To design the system the project team has choose Object Oriented Modeling techniques and Unified modeling language tools.

- ➢ Understanding of the structure is easy because object oriented modeling and tools used to represent real world entities.
- ➢ Modification of the object implementation is easy because objects are loosely coupled.

1.9.3 The system development model

There are various software development life cycle models defined and designed which are followed during the software development process. These models are also referred as "Software Development Process Models". Each process model follows a Series of steps unique to its type to ensure success in the process of software development. But the proposed system follows Waterfall Model. Because the waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap. In "The Waterfall" approach, the whole process of software development is divided into separate phases and, the outcome of one phase acts as the input for the next phase sequentially.

1.10. Feasibility study

Preliminary investigation examine project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Economical, Technical, Operational and Legal feasibility for adding new modules and debugging old running system. There are aspects in the feasibility study portion of the preliminary investigation:

- ❖ Technical Feasibility
- ❖ Operational Feasibility
- ❖ Economic Feasibility
- ❖ Legal feasibility

1.11.1. Technical Feasibility

The proposed system can be easily maintained and repaired; technically, the system will be powerful to be applied by low skilled users as much as possible. There is no need for the developer involvement in almost all implementation of the entire system. It is easily accessible by the people who can easily understand natural languages.

1.11.2. Operational Feasibility

The proposed system will provide best services for customers and user and it will be highly secure. The system will also be on behalf of origination's goal and user satisfaction, because the system will be possible to run and use in the organizations LAN internet. So the system will be operationally feasible or it will be operationally acceptable to users. The system give better user interface registration form and storage of user information, easy updating, deletion and modification etc.

1.10.3. Economic Feasibility

As cost/benefit analysis, show the new system is developed using minimum cost and it give a lot of benefits such as advancing the services of the system, decreasing the work load of the users. The organization does not using any media advertises because it makes information online and every one can get the information from the site.

Tangible benefits: are benefits derived from the creation of an information system that can be measured in money and with consistency.

The team has identified the following:-

- ➢ Increase in flexibility of the modifying blood bank information system.

➤ Provide higher data backup by designing database for the blood bank.

➤ Reduce resource requirements or unnecessary wastage of resource in blood bank like paper, pen and decrease payment for advertises gives for TV, Radio etc.

➤ Increase the speed of activities during searching information.

➤ Increase blood bank management system performance

➤ More timely information for advertising blood transfusion services 24 hour.

Intangible benefits: are benefits derived the creation of an information system that cannot be easily measured in money and consistency.

The intangible benefits of the new system are:

➤ Increase in accuracy of blood bank detail information.

➤ Faster decision making by searching records from data base.

➤ Increase security by providing authorized user can access.

➤ Reduce work load of the organization system users faster decision making by searching records from data base.

➤ Error reduction during filling necessary information about blood donation process.

➤ Increase efficiency of blood bank management system performance.

1.10.4. Legal feasibility

The proposed system has no any conflict with any government directives, because it gave services for the people effectively and efficiently so the organization is profitable and the system is politically feasible.

CHAPTER TWO

SYSTEM ANALYSIS

Introduction

To obtain all the required information for the system development, interview, document review and observation were conducted. The UML modeling technique was used to model the analysis phase of the system. The system analysis, modeling deals with analyzing the proposed system. It includes the system use case diagrams, sequence diagrams, activity diagrams, analysis class diagram and their descriptions. After identifying the actors and use cases, the use cases are developed and textual descriptions are stated. The Sequence diagram depicted based on the use cases which are developed for the proposed system. Activities will be represented by the activity diagrams.

2.1. Overview of the Existing System

As we described in the first chapter, we used the interview (specially, DMBB manager), documentation, and observation together the information required in the current system so based on the above data Collection techniques. We study the background of the organization.
Generally the overall activities of existing system the donor goes to the blood bank and reach to the receptionist nurse then nurse ask some questions about her/his willingness and motivate to

full fill questioners. Then the donor goes to Nurse to donate blood, while nurse test about his/her healthiness (i.e. weight, blood pressure etc.), donor gets counseling and refreshment. If the donor healthy the nurse receive blood. After donation the donor get some advice.

The nurse transferred blood to the laboratory class to check by the lab technician about his/her blood type (A, B, AB, O etc.), blood purity (hepatitis A, B, HIV and syphilis).if the blood is pure stored in stock otherwise discarded. If the donor wants to know about his/her blood profile gets from lab technician. Then the lab technician transfer donor's profile report to data encoder.

When the client hospital (seeker) wants blood they get blood from lab technicians.

Based on the analysis investigated so far, the problems of the existing systems are stated.

➢ Use more professional's human power for more awareness on community as a result there will be a spaced out on organization service.

➢ Because of the lack of Budget, they does not using any advertisements like TV, radio or magazine.

➢ They only decide when and where the reservation for blood donation can behold.

➢ It is time consuming Difficulty in Maintenance of Records

➢ It leads to error prone results; there is high data Redundancy and data Inconsistency.

➢ Editing of data becomes a tedious job.

➢ It lacks of data security, Percentage of accuracy is less.

2.1.1. Users of the existing system

Users are entities that interact with the system .It concerns only in Blood Bank management system in Debremarkos town. There are many basic beneficiaries which can get benefits from the DM blood bank Services. Which are:-

❖ **Blood Donors:** person who wants to donate the blood voluntarily at the blood donation camp.

❖ **Blood Seekers:** An Organization who wants the blood from the blood bank due to various reasons like accidents, surgeries, delivery and many more.

❖ **Blood bank:** staff people which are working in the blood bank which includes staff member, operator, blood bank in charge, head of pathological department.

❖ **Nurses**: -check donor healthiness and received blood.

❖ **Manager**: -managing, supervising, budget all of action for the overall activity of the system.

❖ **Lab Technician**:-test blood, give blood for client hospitals and manage the sock.

❖ **Data Encoder**: - Register all the donor profile and send report for manager.

❖ The **Receptionist-Nurse** Register blood donors give pre donation information and motivate donors to full fill questionnaires.

2.2. System Requirement Specification

The purpose of requirement is used to prepare requirement specification before starting the actual design. This enables the designer to have a clear idea about the structure and content of the information system.

2.2.1. Functional Requirements

Functional requirements explain what has to be done by identifying the necessary task, action or activity that must be accomplished. A function is described as a set of inputs, processing, and outputs. Functional requirements may be technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish.

The system provides the following basic functionality:-

1. Advertisement and announcement: The system to promoting about the organization what is there services, how they treat customers and others. Like other advertising media, it frequently involves a publisher, who integrates advertisements into its online content and users of the system who has an account can be visited published information through internet access.

2. Member signup and staff member registration: This BBMS allows the users to store his/her details in to the system to gain the system services.

3. Online reservation and appointment: The system allows to public can make online reservation on their desired session and date. The blood centers' administrators can then manage their appointments.

4. Blood request and Cross matching: BBMS allows the user to request for blood and blood transfusion for which the cross matching using the appropriate technique can be carried out and the results can further be processed and analyzed by the experts to issue the blood. Also various reports for blood requisition and the cross matching can be generated at run time.

5.Searching functionality: functionality in order to allow normal and privileged users to search the details of a given donor, Develop a searching blood group, and if necessary a type of disease common which causes one to need the donated blood.

6. No installation: As it is a web application, it prevents users from any kind of hindrances faced during installation or up-gradation of application. User simply needs a browser to access the application.

7. Report generation: Various comprehensive reports can be generated any time by the end user to measure the performance parameters in the blood bank and also to analyses the inventory and other aspects in blood bank.

2.2.2. Non Functional Requirements

Non-functional requirement describe visible aspects of the system that are not directly related to the system. Non-functional requirement deals with additional quality of the system such as performance, cost benefits, documentation, new information preserving and security matter.

The following are the non-functional requirements associated with the new system:

➤ **Availability**: The system which is called web based blood bank management system for Debremarkos town is available all the time if internet connection is reliable.

➤ **Security**: The system allows only authorized user to login into the system and Sensitive data is accessed and changed by authorized body (i.e. we use password encryption method like md5).

➤ **Usability**: The system will be easy to be used by all People who can read and write English language.

➤ **Performance**: The system performs its task efficiently and effectively because the team project will uses advanced programing language, less number of iterations for a given task and optimized query to develop the system.

> **Modifiability**: The system can be modified easily by the authorized body; since the system is developed with user friendly programming language which is PHP.

2.2.2.1. Technical Requirement

Technical requirements are the technical issues that must be considered to successfully complete a project. These are aspects such as performance, reliability, availability that the project must meet on in order to proceed with a project.

Generally these are technical requirements to complete the project.

- ✓ The interface of the system should be user friendly (easy to use).
- ✓ The interface should display error message if it detects invalid input.
- ✓ The system should deny unauthorized accesses to the system domain.
- ✓ The system should provide help for the user.
- ✓ Requires an expert to use the system.
- ✓ Training the users to access the system.

2.2.3. Business rules

The Blood Bank's core functions include blood collection, blood grouping, infectious testing, component preparation, and blood components disposition. Currently, all the data and information exchange and processing of the functions of Blood bank is done manually. Only Access Database is used to keep records of donor's, Recipient's, and hospital's information in the current system. Information is highly exposed to error, incompleteness, lose as well as damage.

Generally the following business rules (BR) are used in the project:-

BR1.Donors must be at least 18 years old or at most 65 years old.

BR2.If donor have desire to "give back", he/she can be donate his/her blood to the community in every 3 Months.

BR3.No donation if the Donor have any disease and not in proper health condition.

BR4.The blood in the blood stock is expired after 35 days.

BR5. If donor has a temperature above 37.5 C, donor may not donate.

BR6.Persons who is pregnant is not eligible to donate wait 6 weeks after giving birth.

BR7. Donors should not give blood if they have AIDS or have ever had a positive HIV test.

BR8. If donor had hepatitis, donors are not eligible to donate blood.

BR9.If weight of donor is between (45-50) kg can donate blood up to 350ml and if greater than 50kg can give 450ml at a time.

2.2.4. Change cases

Change cases are used to describe new potential requirements for a system or modifications to existing requirements. These are modeled in a simple manner. Describe the potential change to the existing requirements, indicate the likeliness of that change occurring, and indicate the potential impact of that change.

The system is ready to change if the organization has been open different branches and also if new material has been imported to the camp, in this case the system is easily scalable and ready to be changed. [6]

2.2.5. Constraints

A constraint is a restriction on the degree of freedom you have in providing a solution. Constraints are effectively global requirements, such as limited development resources or a decision by senior management that restricts the way you develop a system. Constraints can be economic, political, technical, or environmental and pertain to the project resources, schedule, target environment, or to the system itself. Some of them are:

1. Lack of transport to gathering a requirement we should be the blood bank and we could not do it move and back to as much as the process needs.

2. They are busy the laboratory technologists are too much busy on the blood bank laboratory work and they have not time to discuss for a time.

3. Expression Through the interview and gathering information we cannot understand some specific terminologies or using their using words.

4. Hard to clarify some of their domain knowledge is hard to articulate and not clear on speech.

2.3. System requirement analysis

The purpose of System Requirements Analysis is to obtain a thorough and detailed understanding of the business need as defined in Project Origination and captured in the Business Case, and to break it down into discrete requirements, which are then clearly defined, reviewed and agreed upon with the Customer Decision-Makers. During System Requirements Analysis, the framework for the application is developed, providing the foundation for all future design and development efforts. [8]

Systems Requirement Analysis gives the professional systems engineer the tools to set up a proper and effective analysis of the resources, schedules and parts that will be needed in order to successfully undertake and complete any large, complex project.

2.3.1. Actor and Use case Identification

Based on the findings of existing system assessment, the system process is modeled and use cases and actors are identified. We relate actors with the corresponding use cases as shown in figure 1.

2.3.1.1.1 Actors of the system

Actors: An actor represents anything or anyone that interacts with the system. This may include people (not just the end user), external systems, and other organizations. Actors are always external to the system being modeled; they are never part of the system.

The actors that will participate in the system are listed below:

➢ The **System Administrator**-is the one who maintains and manage account for users and manages the system.

➢ The **Blood donor**-is a person who reserves donation.

> ➢ **BB Manager-**is the one who is responsible for blood management and blood distribution.
> ➢ The **Blood Seeker**- is the user that sends requests to receive blood from the blood bank.
> ➢ **Lab Technicians**-a technician that discards expired blood from the database.
> ➢ The **Data Encoder**-generate report and register blood details in to the database.

2.3.1.1.2. Use cases of the system

The Use case (UC) represents functionality provided by a system unit and expressed by sequence of message exchange by the system unit and one or more actors of the system. The following use cases have been identified for the proposed system specification.

Use case Name	Use Case ID	Uses/Includes
Create Account	UC-01	Login
Discard Blood	UC-02	Login
Distribute Blood	UC-03	Login
Generate Report	UC-04	Login
Give comment	UC-05	Login
Login	UC-06	-------
Logout	UC-07	-------
Block Account	UC-08	Login
Receive seeker request	UC-09	Login
Add blood Details	UC-10	Login
Register donor	UC-11	Login
Response Request	UC-12	Login
Request blood	UC-13	Login
Reserve donation	UC-14	Login
Update Account	UC-15	Login
View Account	UC-16	Login
View Blood Details	UC-17	Login
View Comments	UC-18	Login
View donor Profile	UC-19	Login
Retrieve Report	UC-20	Login
View Reservation	UC-21	Login
View Seeker Profile	UC-22	Login

Table 2: Use case Identification

2.3.1.1. Use case Diagram

The use case diagram is concerned with the interaction between the system and actors (objects outside the system that interact directly with it). It presents a collection of use cases and their corresponding external actors. A use case is a generic description of an entire transaction involving several objects of the system. Use cases are represented as ellipses, and actors are depicted as icons connected with solid lines to the use cases they interact with.

A use case diagram is helpful in visualizing the context of a system and the boundaries of the system's behavior. Each use cases in the use case diagram can also be described using a narrative form. [6]

A Use Case represents a discrete unit of interaction between a user and the system. A use case diagram contains four components.

1. **Boundary**:-which defines the system of interest in relation to the world around it.

2. **Actors**:-usually individuals involved with the system defined according to their roles.

3. **Use cases**: -which the specific roles are played by the actors within and around the system.

4. The **relationships** between the actors and the use cases as depicted in the following figure.

THE SYSTEM MODEL

ADMIN

Block Account

View Comment

Update Account

Generate Report

Create Account

View Account

Give Comment

Reserve Donation

Discard Blood

BLOOD DONOR

LAB TECHNITIAN

LOGOUT

LOGIN

Include

Exclude

Include

Recieve Request

Response Request

Retrive Report

Distribute Blood

Register Blood

View Blood

Send Request

BB Manager

DATA ENCODER

BLOOD SEEKER

2.3.1.2. Use case description

The following consecutive tables show the use case description for each of the use cases. Each table contains the use case name, the actor which initiates and interacts with the use case, description of the use case and typical course of events that show the interaction between the actor and the use case which enable the team to easily depict the functions of the proposed system.

Table 3: Use case description for login

Use case ID	UC-06	
Use case name	Login	
Actors	Admin, Blood Donor, Blood Seeker ,Lab Technicians, BB Manager, data encoder	
Description	It is authenticating bridge that allows user to login to the system.	
Goal	To be accessed by an authorized and trust system user.	
Precondition	User should be registered before perform any activity.	
Basic course of Action	Actor action Step1:user activate the system Step3: user enters user name and password Step6: Use case End.	System response Step2:system show login interface Step4: the system check the authentication of user name and password Step5:system display user page
Alternative course of action	A. If the username and password is invalid. 1. The system displays error message. 2. The system continues at step 2 to fill user name and password again.	
Post condition	System transfer control to user main screen to precede actions.	
Include	------	

Exit condition	Logout

Table 4: Use case description for manage account

Use case ID	UC-08	
Use case name	Manage Account	
Actors	Administrator	
Description	This activity is performed when the admin want to manage the users account	
Goal	To control the system worker.	
Precondition	The System administrator must login to control the account	

Basic course of Action	**Actor action**
	
	Step1: admin enter user name and password
	
	Step4:admin select
	   ➤ create account
	   ➤ Update account.
	   ➤ view account
	If create account
	
	Step6: admin enter user account information.
		System response
		
		Step2:the system check the authentication of user name and password
		
		Step3:the system display admin page
		Step5: System display create account page.
		
		Step7: system check creates user account information.
		Step8: System creates user account.

Alternative course of action	A.Invalid information entry. 1. The system displays error message. 2. Go to step 6to fill again.	
Post condition	The system admin successfully create, update and delete the account!	
Include	Login	
Exit condition	Logout	

Table 5: Use case description for Store blood detail

Use case ID	UC-10
Use case name	Add blood detail
Actor	Data Encoder
Description	A Person who Enter blood details to the database.
Goal	It allows user to store the blood detail of the system database.
Precondition	The system User must be logged in successfully.
Basic course of Action	<table><tr><td>Actor action</td><td>System response</td></tr><tr><td>1.User clicks on "Add blood detail" menu</td><td>2.The system displays "Add blood detail" form with "submit" button</td></tr><tr><td>3. User fills the Blood information. 4.Then clicks on "submit" button 8.Use case ends</td><td>5.The system validates the form 6. The system records the form into "blood detail" table of the system's database. 7. The system acknowledges user by displaying "the blood detail was successfully added on database! "Message.</td></tr></table>
Alternative course of action	If the values are invalid the system is return to step 5 to fill the appropriate values of the blood details.
Post condition	The blood detail exists on blood detail table of the system database.
Include	Login
Exit condition	Log Out

Table 6: Use case description for seeker profile

Use case ID	UC-22	
Use case name	View Seeker Profile	
Actors	BB Manager	
Description	It allows user to search other seekers detail from the seekers profile table of system's database.	
Goal	To know all blood Seeker profiles.	
Precondition	User must be logged in successfully.	
Basic course of Action	Actor action 1.User clicks on "view Seeker" menu 3. User fills ID numbers 4.User clicks on "search" button 6.the system search the data from "seeker details" table of the system's database	System response 2.The system displays "search" window with "search" button 5.The system validates the form 7.Then the system display the result on search window 8.Use case ends
Alternative course of action	A.6: If the form is invalid the system is return to step 2 of basic course of action and indicate user where he/she was made an error.	
Post condition	User views the result on search window.	
Include	Include	
Exit condition	Logout	

Table 7: Use case description for Reserve Donation

Use case ID	UC-14
Use case name	Reserve Donation
Actors	Donor
Description	Describes how a blood donor reserve donation time slot.
Goal	It allows user to Reserve a Donation with an organization by filling and sending to the System.
Pre-conditions	User must be logged in successfully.
Basic course of Action	<table><tr><td>Actor Action</td><td>System Response</td></tr><tr><td>1.User clicks on "Reserve Donation "menu 3. User answers the question by choosing the choices. 4.User clicks on "next" button 7. User fills all personal information. 8.User clicks on "send" button 11. The system acknowledges user by displaying "thank you for your support" message and displaying the date and time interval. 12.Use case ends</td><td>2. The system display questioner window which has an eligibility question within yes or no choice. 5.The system checks whether the donor is eligible or not 6.The system displays "Reserve "form with "send" button 9.The system validates the form 10. The system submits the form into "Reserve" table of the system's database.</td></tr></table>
Post Condition	Appointment Successfully Send.

Include	Include
Exit condition:	Logout

Table 8: Use case description for view comment

Use case ID	UC-18	
Use case name	View comment	
Actors	Administrator	
Description	Admin can see the comments that are submitted from the users (Blood Donor, Blood Seeker).	
Goal	To view system user feedback about the system.	
Pre-Condition	User must be logged in successfully.	
Basic course of Action	Actor action Step1: admin enter user name and password Step4:admin select view comment link Step6: admin view comment. Step7. Use case end.	System response Step2: the system checks the authentication of user name and password. Step3:system display admin page Step5:system display comment records
Post Condition	Admin views the submitted comments.	
Alternative Course Of Actions	A. If fail to view comment 1. The system displays error message. 2. Go to step4to view comment again.	
Include	Login	
Exit condition	Logout	

2.3.2. Sequence Diagram

Sequence diagrams are used to show how objects interact in a given situation. An important characteristic of a sequence diagram is that time passes from top to bottom: the interaction starts near the top of the diagram and ends at the bottom. [5]

➢ **The boxes** across top of the diagram represent classifiers or their instances; typically use cases, objects, classes or actors.

➢ **The solid lines** hanging from the boxes are called objects lifelines, representing the life span of object during the scenario being modeled.

➢ **Messages** indicate as labeled arrows, when the source and the target of a message is an object or class label the signature of the method invoked in response to the message.

However, if either the source or target is the human actor, then the message is labeled with brief text describing the information is available.

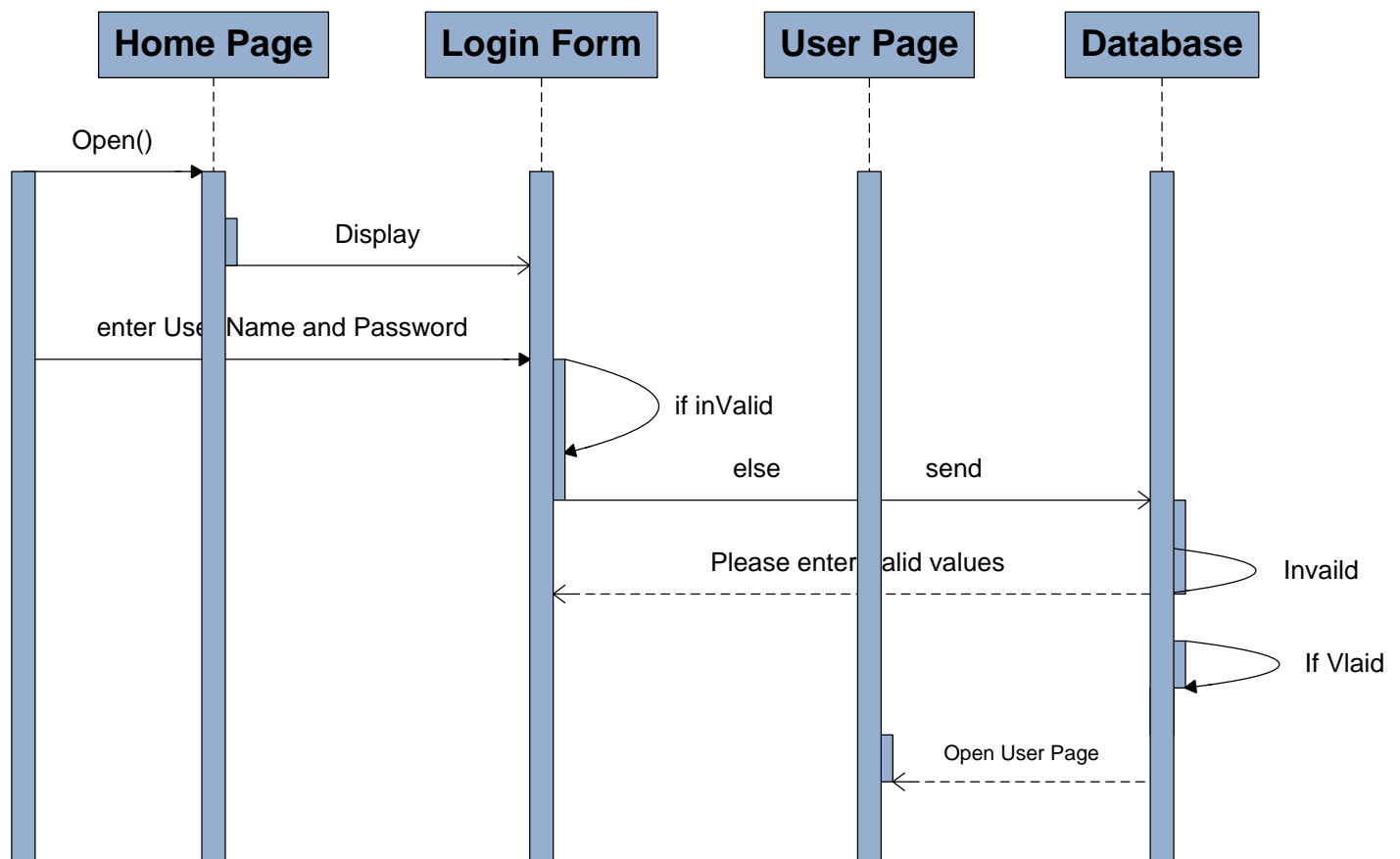Generally, a sequence diagram shows object interactions arranged in time sequence.
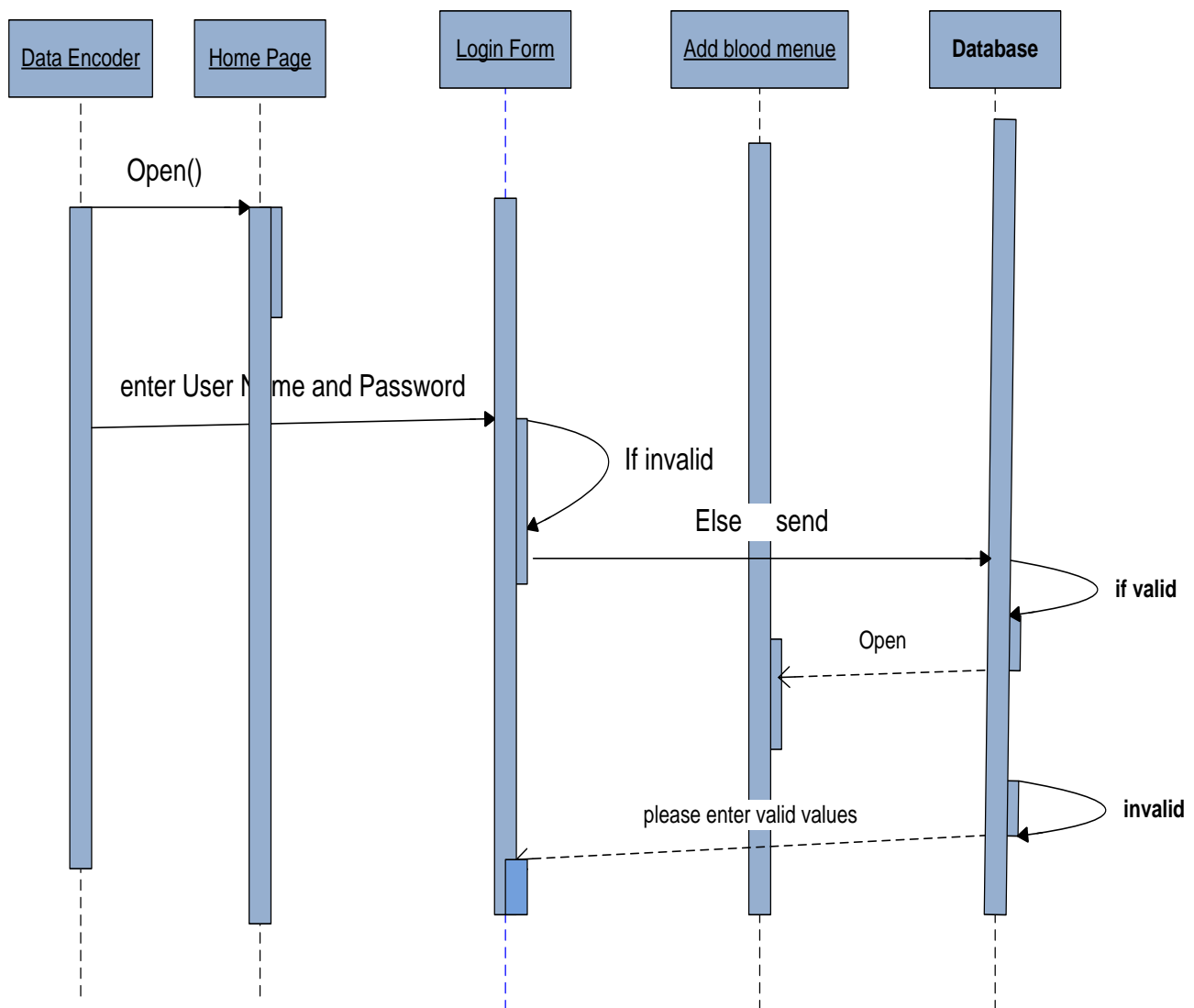
Figure 2: sequence diagram for login

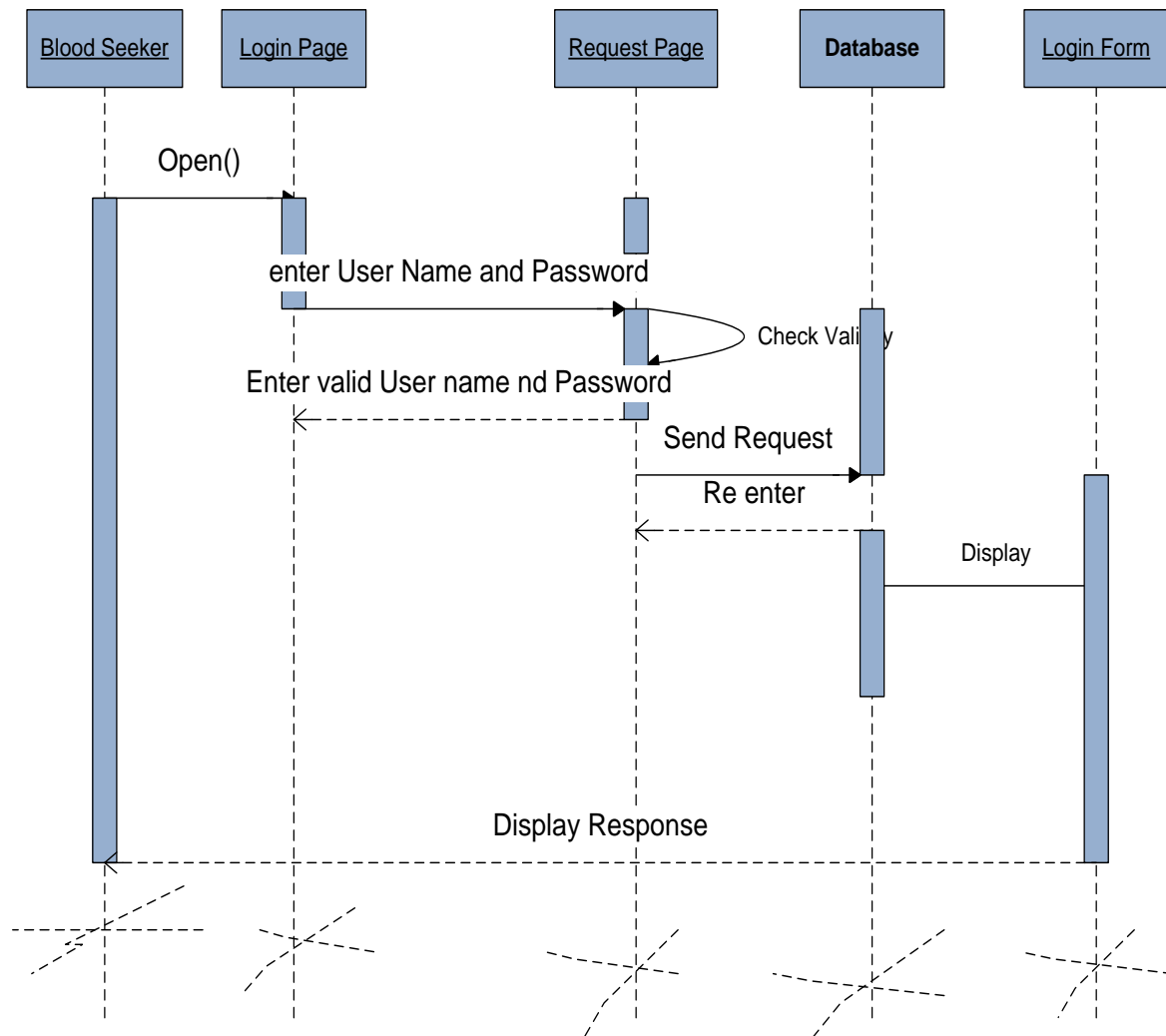Figure 3: sequence diagram for Add blood detail
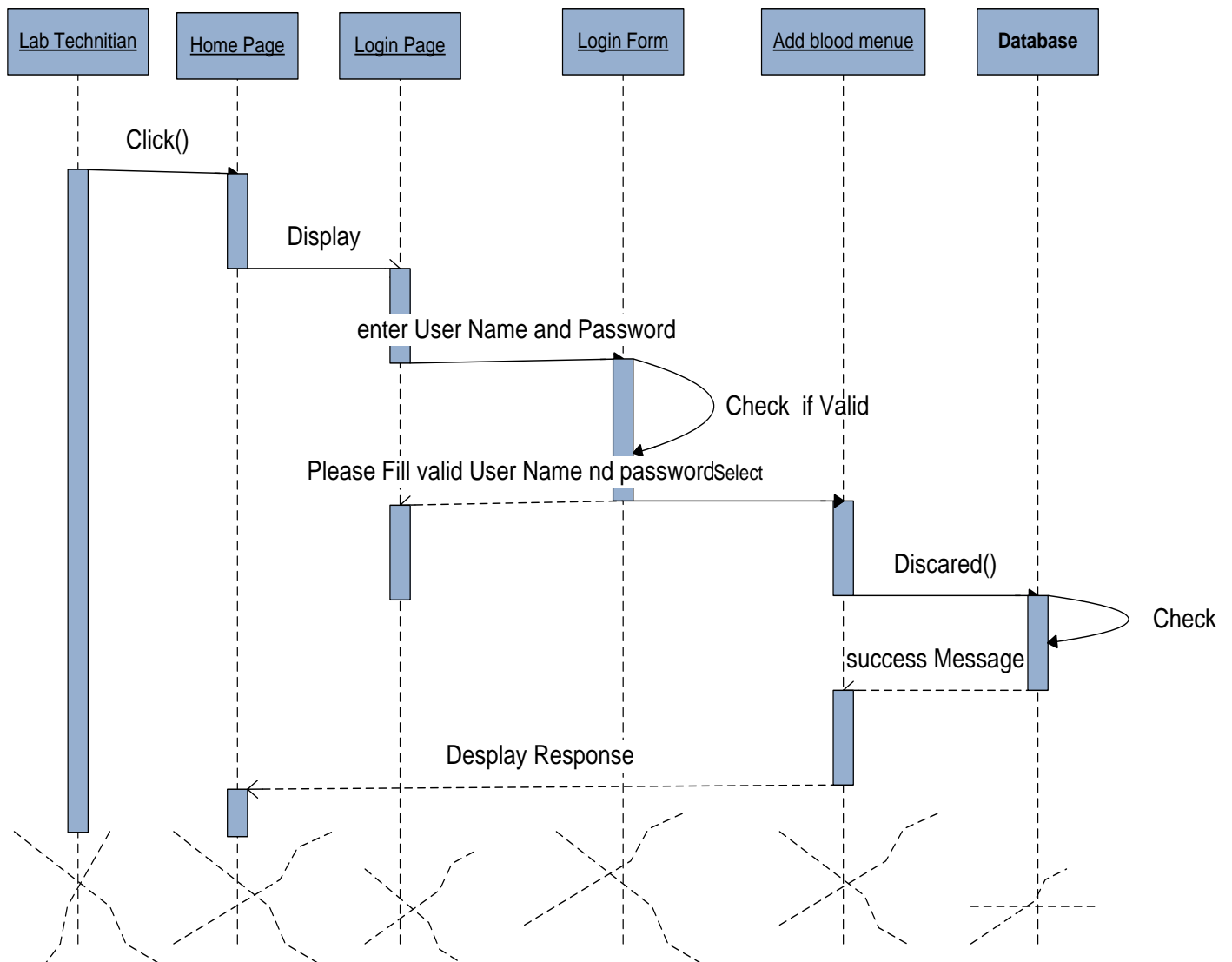

Figure 4: sequence diagram for request blood

Figure 5: sequence diagram for Discard blood detail
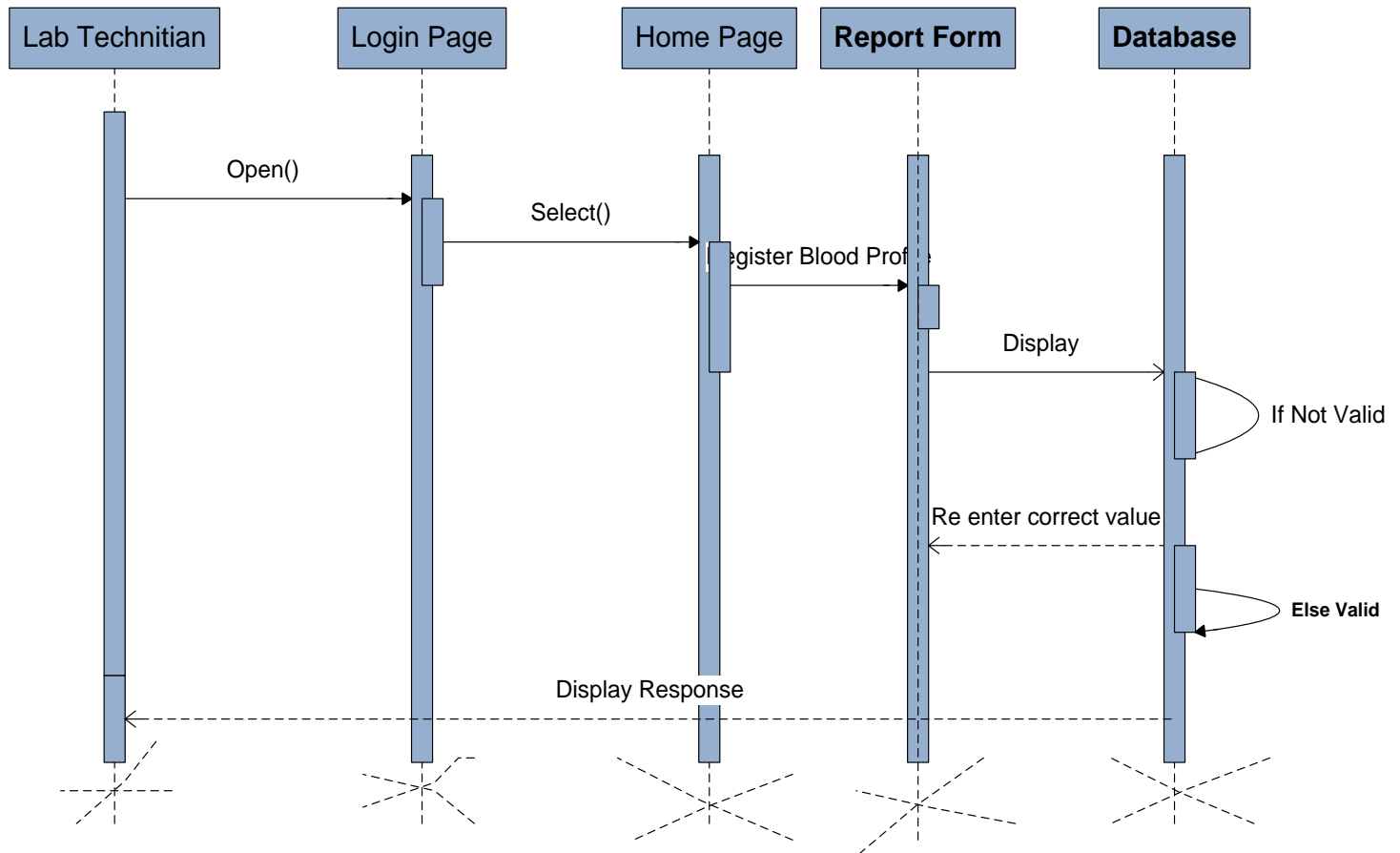
Figure 6: sequence diagram for generate report

2.3.3. Activity Diagram

Active diagrams are used to model the flow of an object as it moves from state to state at different points in the flow of control. It is essentially a flow chart that emphasizes the activity that takes place over time. Activity diagrams can be used to model higher-level business process at the business unit level, or to model low-level internal class actions. It is "Less technical" in appearance, compared to sequence diagrams, and business-minded people tend to understand them more quickly. [4]

Table 9: Activity diagram Description

No	Symbol	Name	Function
1.		Initial state	To start the activity.
2.		Decision activity	To make validation.
3.		Final state	To terminate from the activity.
4.		Arrow	To show the flow of activity between two activity.
5.		Activity	Set of activities that the user and system performing.

{USER}

User Open Home Page

Logon Page is Displayed

Users Fill User Name And Password

[Check Validity]

[Not Valid]

~ 34 ~

[If Valid]

Display Requested actorPage
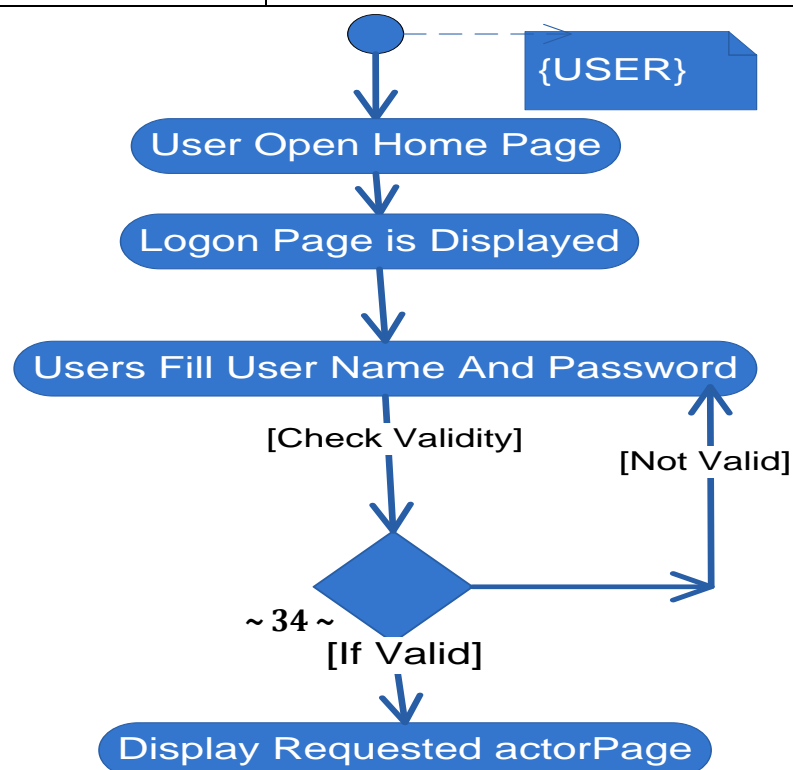
Figure 7: activity diagram for login
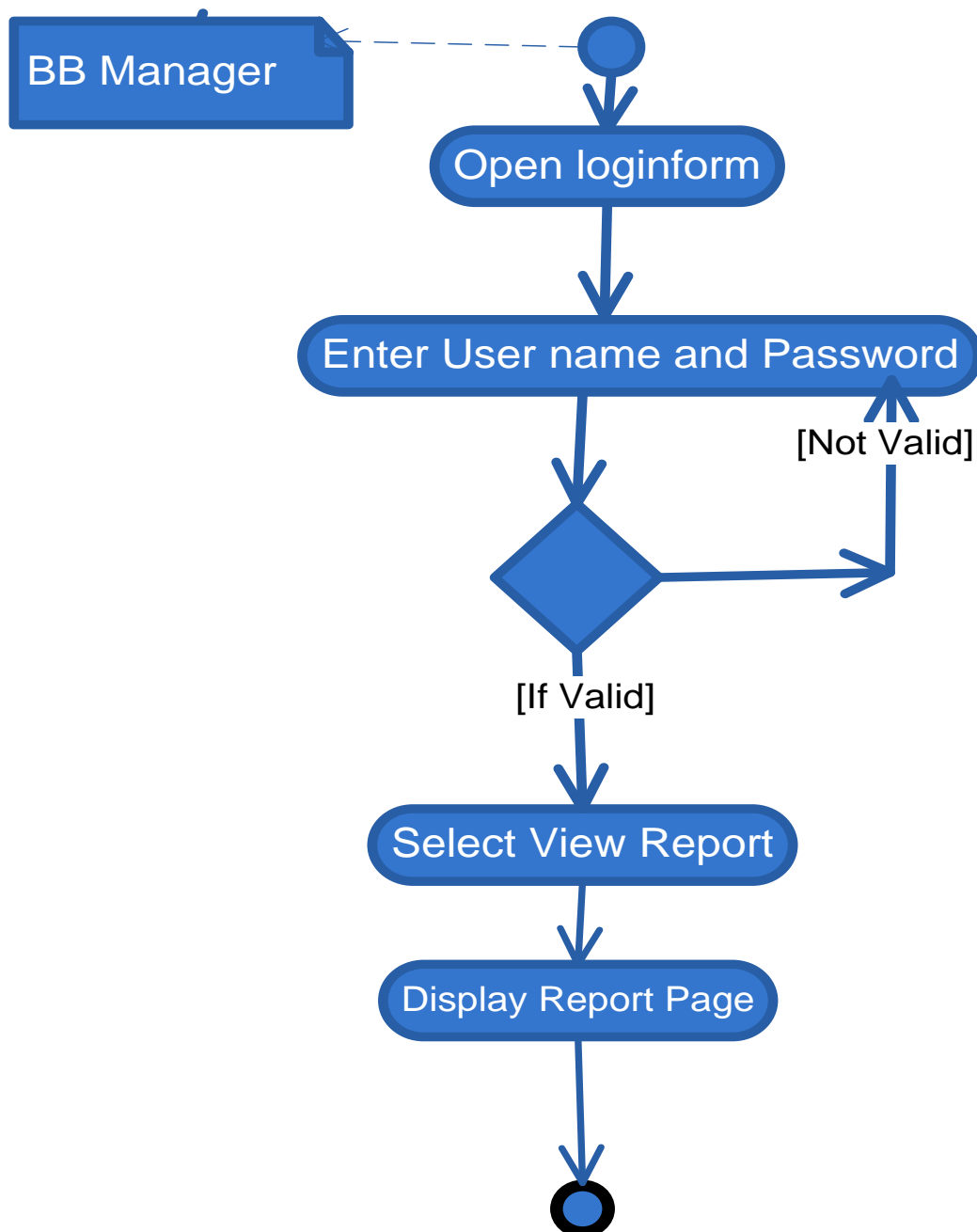
Figure 8: Activity diagram for view report
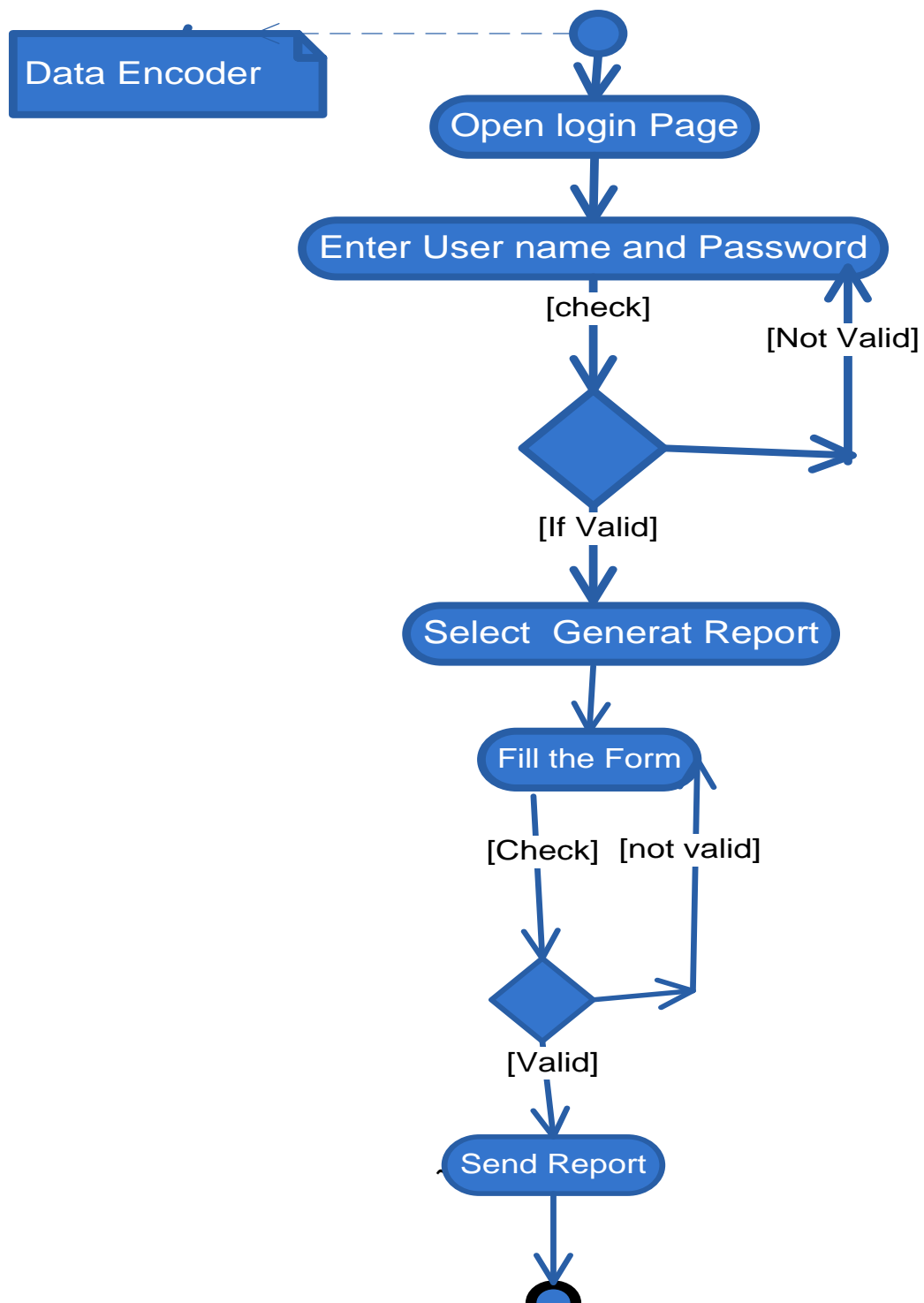
Figure 9: activity diagram for generate report

Blood Seeker

Open login Page

Enter User name and Password

[check]   [Not Valid]

[If Valid]

Select Request Blood

Fill the Form

[Check] [not valid]

[Valid]

Send Report
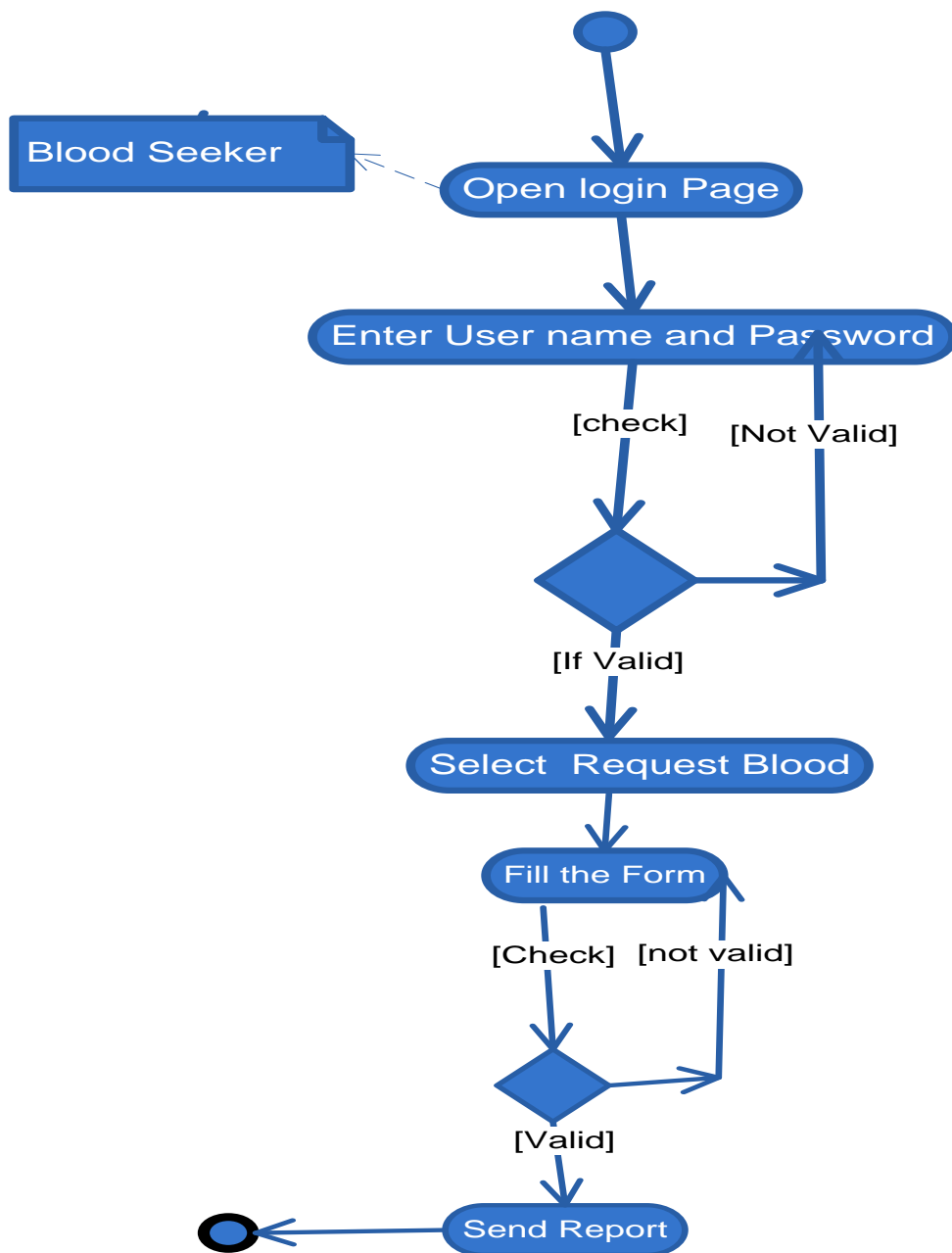
Figure 10: activity diagram for request blood
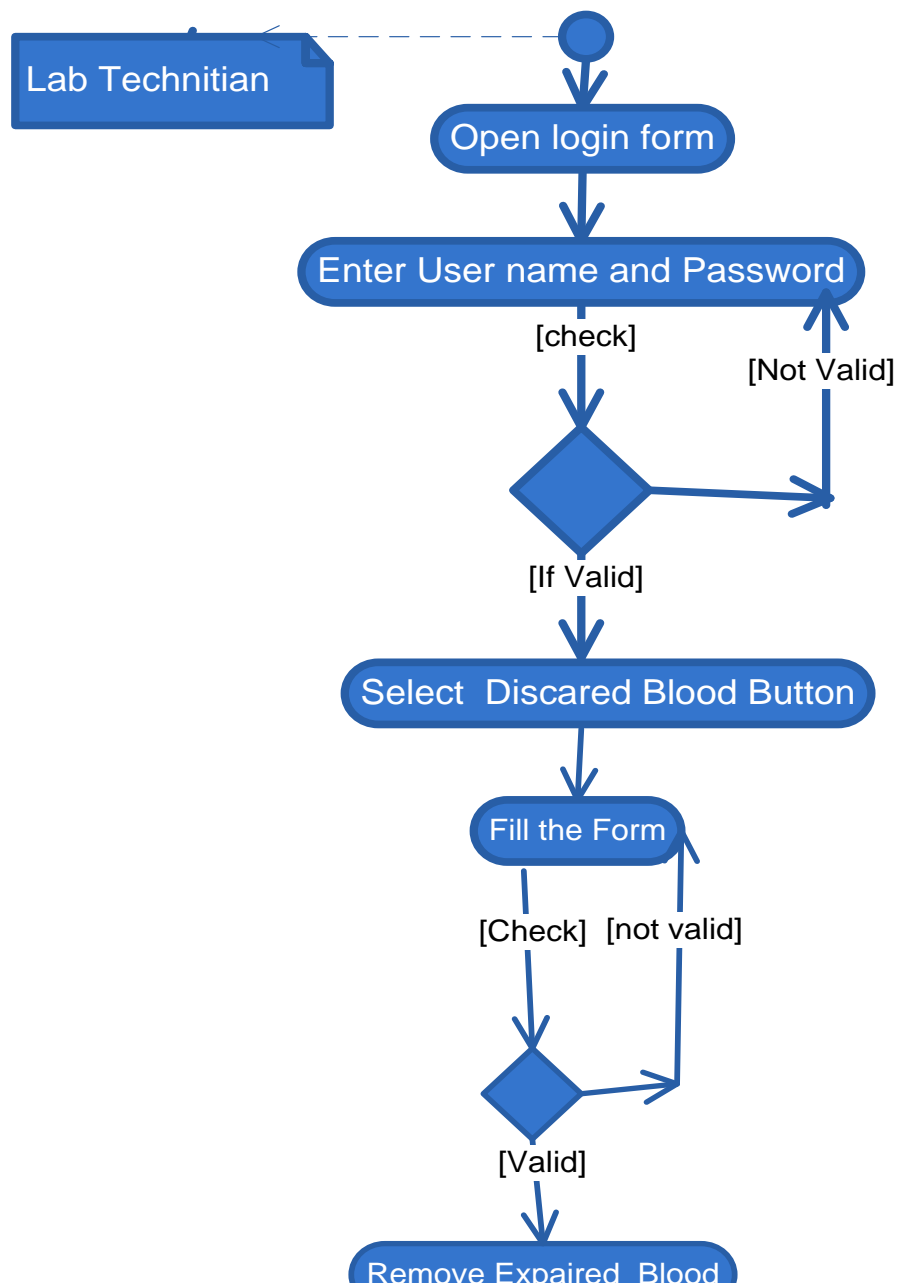
Figure 11: Activity diagram for Discard blood detail

2.3.4. Analysis Class Diagram

UML class diagrams are the mainstay of object-oriented modeling. Class models show the classes of the system, their interrelationships (including inheritance, aggregation, and association), and the operations and attributes of the classes. Class diagrams are used for a wide variety of purposes, including both conceptual/domain modeling and detailed structural design modeling.

Classes are depicted as boxes with three sections: the top one indicates the name of the class, the middle one lists the attributes of the class, and the third one lists the methods. By including both an attribute and a method box in the class. Another approach would be to have two sections, one for the name and one for the responsibilities. [4]

Figure 12: Analysis class diagram
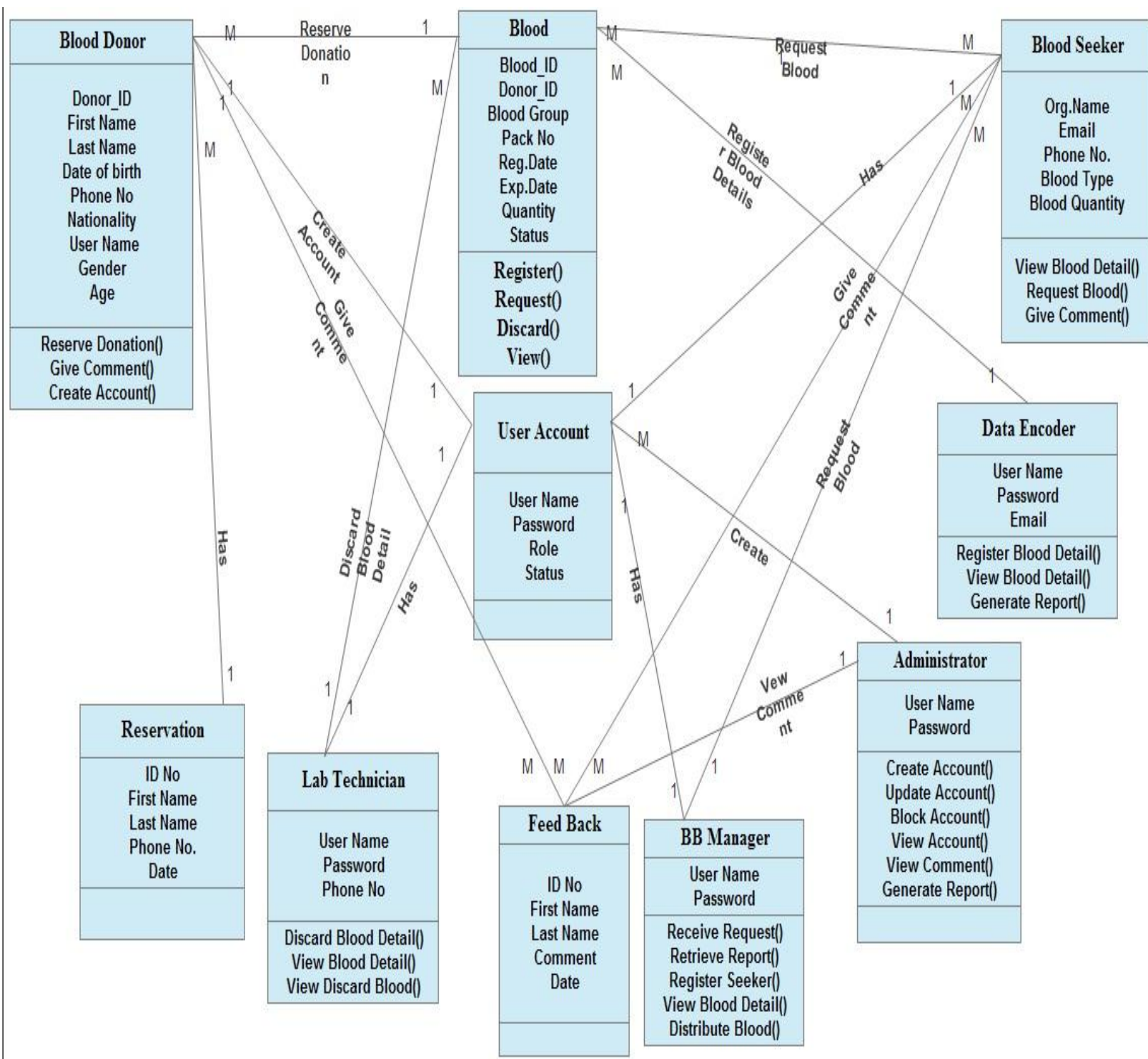
CHAPTER THREE

SYSTEM DESIGN

Introduction

The Design Phase seeks to develop detailed specifications that emphasize the physical solution to the user's information technology needs. The system requirements and logical description of the entities, relationships, and attributes of the data that were documented during the Requirements Analysis Phase are further refined and allocated into system and database design specifications that are organized in a way suitable for implementation within the constraints of a physical environment. (e.g., like computer, database, facilities etc.).

During the Design Phase, the initial strategy for any necessary training is also begun. Estimates of project expenses are updated to reflect actual costs and estimates for future phases. In addition, the work planned for future phases is redefined, if necessary, based on information acquired during the Design Phase. [6]

3.1. Design Goals of the system

The design of the system is highlighted as follows, in terms of performance, maintenance, end user criteria and security

> ➢ Performance

The system will complete the task quickly to allow easy input of data and to retrieve data from the server. The system will be accessible from any computer with internet and will be accessible anytime a user would want to use the program. In terms of performance the system will help the user quickly accomplish their task.

> ➢ Maintenance

It does not need expertise person but just it needs any professional person in IT who can use database and internet. The system will require minimal upkeep and maintenance to stay working at optimal levels.

➢ End user

This project is very simple to use. Anyone who can read English language can use the system, because, to use the system only clicking a button, it does not need to write commands and to think how to use it. This program will have a well-defined and easily understood interface. The processes will be easy to understand and use by a user of any level.

➢ Security requirement

Since the system will require strong security features to protect that valuable information. To use it, it needs user name and password when they try to login to the system. So it is very secured system. In general to make the system asset safe, the system will follow a role based security which means the access level and privilege for each users of system is predefined by the system administrator. The end user of the system most of the time will be asked for authorization login.

3.2. Design Class Diagram

The class diagram represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application. The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The classes diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages. The class diagram shows a collection of classes, interfaces, associations, collaborations and constraints. It is also known as structural diagram for this the team developed the following class diagram.

Blood Donor

Donor_ID:varchar(20)
First Name:varchar(20)
Last Name:varchar(20)
Date of birth:varchar(20)
Phone No:Int
Nationality:varchar(20)
Blood Group:varchar(20)
User Name:varchar(20)
Gender:varchar(20)
Age:Integer

Reserve Donation()
Give Comment()
Create Account()

Blood

Blood_ID:varchar(20
Donor_ID:varchar(20
Blood Group:varchar(20
Pack No:varchar(20
Reg.Date:date
Exp.Date:date
Quantity:Integer
Status:varchar(20

Register()
Request()
Discard()
View()

Blood Seeker

Org.Name:varchar(20
Email:varchar(20
Phone No.:varchar(20
Blood Type:varchar(20
Blood Quantity:Integer

View Blood Detail()
Request Blood()
Give Comment()

Reserve Donation

Request Blood

Register Blood Details

Has

Create Account

Give Comment

Give Comment

Request Blood

User Account

User Name:varchar(20
Password:varchar(20
Role::varchar(20
Status:Int

Data Encoder

User Name:varchar(20
Password:varchar(20
Email:varchar(20

Register Blood Detail()
View Blood Detail()
Generate Report()

Have

Discard Blood Detail

Has

Create

Has

View Comment

Donation

ID No:varchar(20)
First Name:varchar(20)
Last Name:varchar(20)
Phone No.:Integer
Date:Date

Lab Technician

User Name:varchar(20)
Password:varchar(20)
Phone No.:Integer

Discard Blood Detail()
View Blood Detail()
View Discard Blood()

Feed Back

ID No:varchar(20)
First Name:varchar(20)
Last Name:varchar(20)
User Name:varchar(20)
Comment:varchar(200)
Phone No:varchar(20)
Date:date

BB Manager

User Name:varchar(20
Password:varchar(20

Receive Request()
Retrieve Report()
Register Seeker()
View Blood Detail()
Distribute Blood()

Administrator

User Name:varchar(20
Password:varchar(20

Create Account()
Update Account()
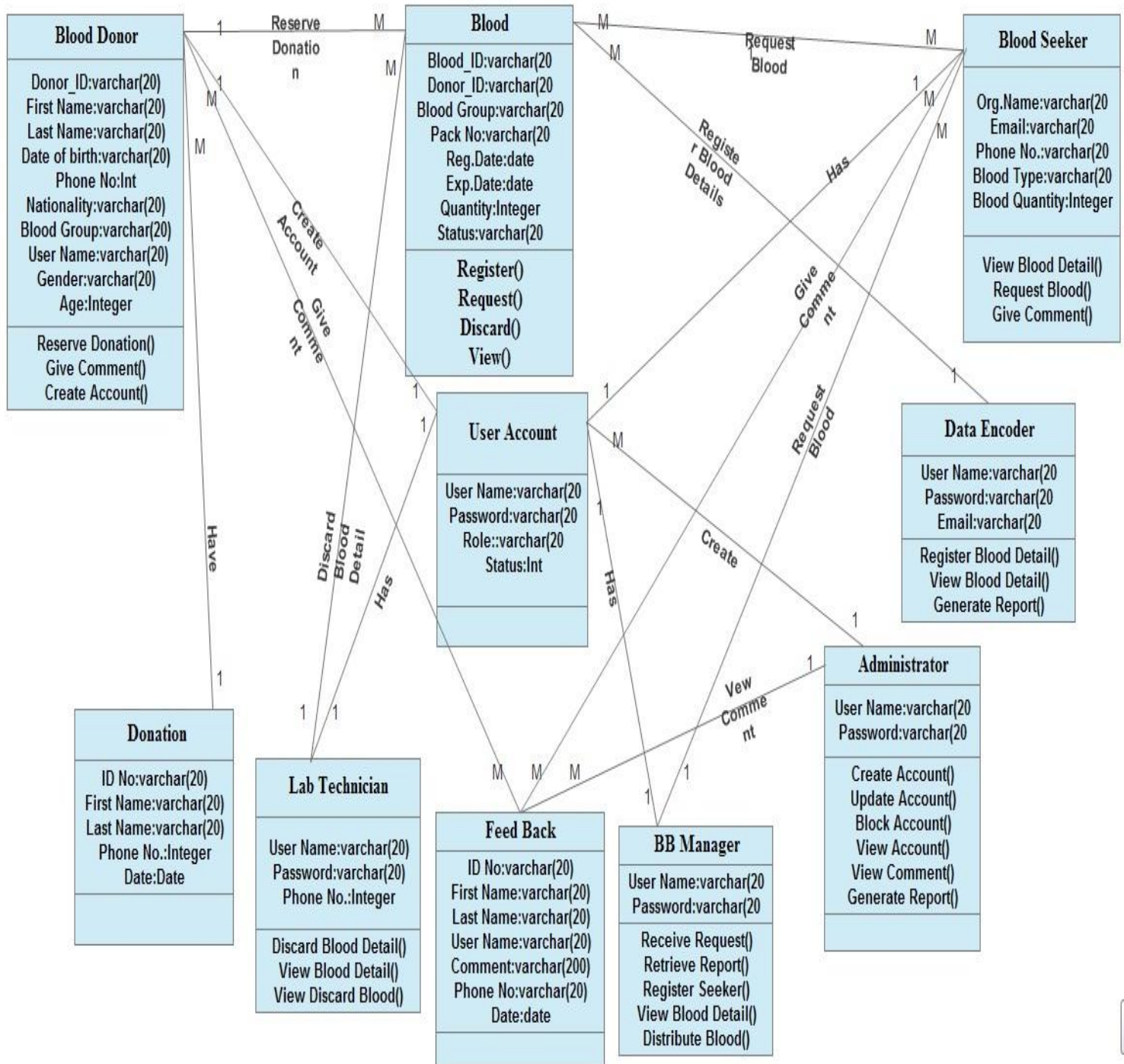Block Account()
View Account()
View Comment()
Generate Report()

Figure 13: Design class diagram

3.2.1. Description of design class diagram

Documenting class model is used to document the detail about a class as well as the reasoning behind any trade off .The following are the documentation of the class model for the WB blood bank management system for DM Town.

Blood donor class

Table 10: Description of Blood donor class

Attribute	Description	Type
First Name	Represents the first name of a blood donor	Varchar2(255)
Last Name	Represents the father name of a blood donor	Varchar2(255)
ID No	The unique identification of a blood donor	String
Date of Birth	Represents the birth date of a blood donor	Varchar2(255)
Nationality	The country whose donor come	String
Blood Group	The type of blood which is donor donated	String
Gender	The maleness or femaleness of a donor.	String
Age	The age of a donor specially between 18 up to 65	Integer
Phone No	Phone number of a donor which is contacted.	Integer
User Name	The authentication system of a donor to use the system.	String
Password	The authentication system of a donor to use the system in visible part.	Long
Method for the Blood donor class		
Method	Purpose	
Donate Blood	The main target of blood donor is donate blood with voluntary	
Send Request	Donor Can Make Request to donate blood.	
Give Comment	Donor can give comment about the system either appreciate or advice	
View Appointment	Nurse can give appointment to blood donor to donate blood.	
Login	Donor must be use login to keep security issue.	

Administrator class

Attribute	Description	Type
User name	Any user of the system must be login to make any activity.	Varchar2(255)
Password	It is a security method to control the system for admin.	Long
Address	The specific Location of administrator manage & control the system	Varchar2(255)
Role	The specific activity of the user to manage the system	Varchar2(255)
Methods for Administrator class		
Method	Purpose	
Create Account	Admin Create account For Users by giving user name and password for each user to keep system security.	
Talk Back up	Admin Can talk backup Before the system failed happen.	
Register Blood Seeker	Client hospitals are agreement with admin before using the system.	
View Blood	Admin can view all available bloods on the stock.	
View Comment	Comments send by users can manage, view, accept or reject the idea by Admin.	
View Report	Daily or monthly reports can view admin.	

Table 11: Description of Administrator class

3.3. Database design /Physical data model

Physical database design translates the logical data model into a set of SQL statements that define the database. For relational database systems, it is relatively easy to translate from a logical data model into a physical database. The physical design of the database optimizes performance while ensuring data integrity by avoiding unnecessary data redundancies. During physical design, you transform the entities into tables, the instances into rows, and the attributes into columns.

Rules for translation from logical to physical database design:

➤ Entities become tables in the physical database.

> ➢ Attributes become columns in the physical database. Choose an appropriate data type for each of the columns.

> ➢ Unique identifiers become columns that are not allowed to have NULL values. These are referred to as primary keys in the physical database. Consider creating a unique index on the identifiers to enforce uniqueness.

> ➢ Relationships are modeled as foreign keys.

☐	1	**User Name**	varchar(25)	latin1_swedish_ci	No	None	🖉 Change ⊖ Drop	🔑 Primary
☐	2	**Password**	varchar(25)	latin1_swedish_ci	No	None	🖉 Change ⊖ Drop	🔑 Primary
☐	3	**Role**	varchar(25)	latin1_swedish_ci	No	None	🖉 Change ⊖ Drop	🔑 Primary
☐	4	**Status**	int(11)		No	None	🖉 Change ⊖ Drop	🔑 Primary

Table 12: physical data model for User Account table

Table 13: physical data model for Blood table

☐	1	Blood_ID	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	2	Blood_Group	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	3	Donor_ID	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	4	Reg_Date	date		No	None	Change ⊖ Drop	Primary
☐	5	Exp_Date	date		No	None	Change ⊖ Drop	Primary
☐	6	Pack No	int(25)		No	None	Change ⊖ Drop	Primary
☐	7	Status	int(11)		No	None	Change ⊖ Drop	Primary
☐	8	Quantity	decimal(10,0)		No	None	Change ⊖ Drop	Primary

Table 14: Physical data model for Blood Donor table

☐	1	Donor_ID	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	2	First_Name	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	3	Last_Name	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	4	Blood_Group	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	5	Date Of Birth	date		No	None	Change ⊖ Drop	Primary
☐	6	Nationality	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	7	Gender	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	8	Age	int(25)		No	None	Change ⊖ Drop	Primary
☐	9	Phone No	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	10	User Name	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary
☐	11	PassWord	varchar(25)	latin1_swedish_ci	No	None	Change ⊖ Drop	Primary

3.4. Sample user interface

The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals what is often called user centered design.

Good user interface design facilitates finishing the task at hand without drawing unnecessary attention to it. The following interface design pictures describe the logical characteristics of some interfaces between the system and the users. So the sample interfaces are shown as follows.

Figure 14: Blood bank home page user interface

Figure 15: Blood bank Login page user interface

3.5. System Architecture (Deployment Diagram, component)

3.5.1. System Architecture of the system

The term system architecture is used to describe the overall design and structure of a computer network or system. A method is required to organize and connect these items together in a cohesive manner. The term is also used to describe complex computer_ software tools. Systems Architecture is a generic discipline to handle objects (existing or to be created) called "systems", in a way that supports reasoning about the structural properties of the objects or it is the conceptual model that defines the structure, behavior, and more views of a system.

There are three main components to any system architecture of the system theses includes: storage, connectivity, and user experience. It is important to note that system architecture must be flexible and able to meet changing needs quickly. Processing power is the brain of the system. To installing the correct allocation of processors to the system must be based on the software specifications, number of concurrent users, strength of the connection, and applications.

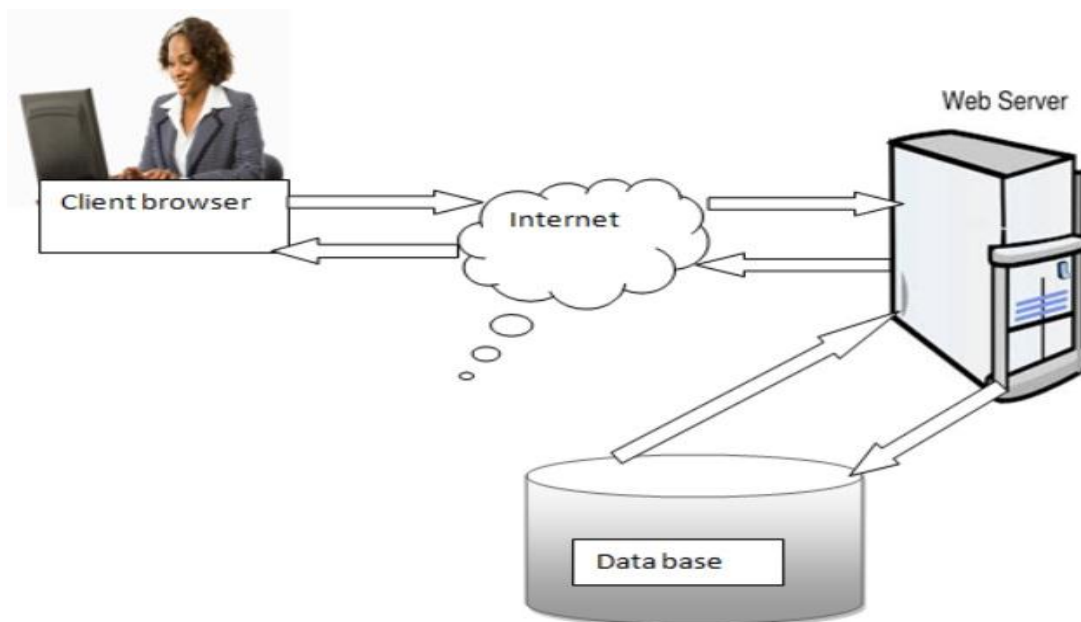The following architecture shows how the system looks like to implement.

Figure 16: System Architecture of the system

3.5.2. Deployment and component Diagram

Component diagrams and deployment diagrams are closely related. Component diagrams are used to describe the components and deployment diagrams shows how they are deployed in hardware. UML is mainly designed to focus on software artifacts of a system. But these two diagrams are special diagrams used to focus on software components and hardware components. So most of the UML diagrams are used to handle logical components but deployment diagrams are made to focus on hardware topology of a system. The purpose of deployment diagrams can be described as: [9]

➢ Visualize hardware topology of a system.

➢ Describe the hardware components used to deploy software components.

➢ Describe runtime processing nodes. For this project the project team developed the following deployment diagrams.
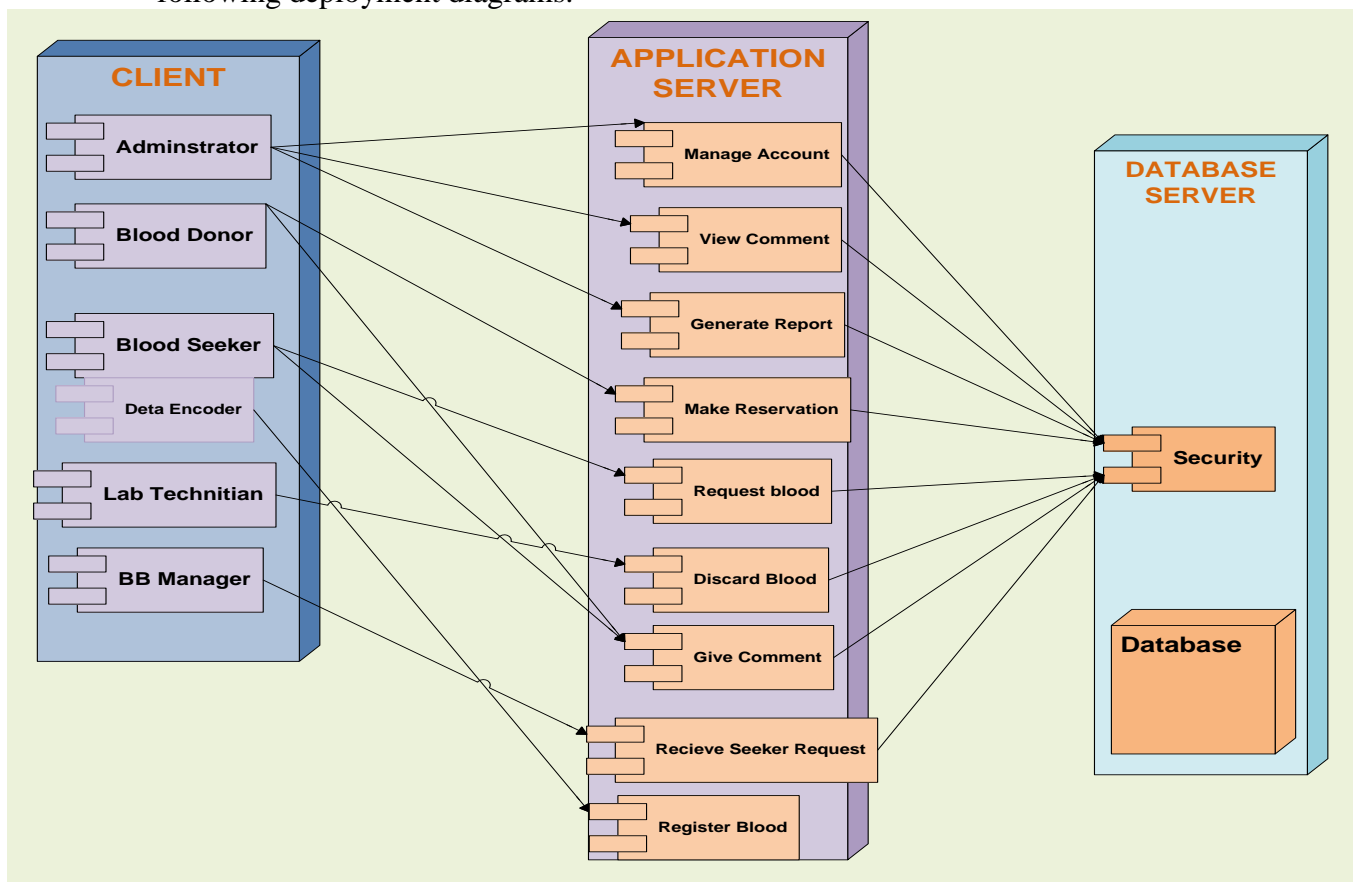
Figure 17: Deployment Diagram

CHAPTER FOUR

IMPLEMENTATION

Introduction

Systems implementation is the final stage of system development process of defining how the information system should be built ensuring that the information system is operational and used and ensuring that the information system meets quality.

The implementation document helps users on how to work with the system. It acts as a user manual and it helps users not to be the system confused with. It includes sample forms and some selected fragment code. It gives the users a brief over view of the system.

This document describes the project implementation for developing the web based blood bank management system. The project implements PHP, MySQL and standard HTML.The project will be capable of running on standard internet web browsers. The interface for the project will provide a good navigation to user of the system with nice look and feel.

4.1. Overview of the Programming Language

This project used php server side programming technology integrated MySQL database with a Programming language.

PHP is a scripting language that is often embedded into HTML to add functions HTML alone can't do. PHP allows you to collect process and utilize data to create a desired output. In short, it lets you interact with your pages. PHP is freely available for use. The community of open source PHP developers provides technical support and is constantly improving updating the core PHP functionalities.

PHP is available at free of cost under PHP General Public License and most of its associative required software's like MySQL, Text Editors and Apache Server are also freely available, so it proves very cost effective for the developers.

PHP provides high compatibility with leading operating systems and web servers such as thereby enabling it to be easily deployed across several different platforms.

PHP scripts can run across operating systems such as Linux, Windows, Solaris, and Mac OSX etc. and also provide support for all major web servers such as Apache, IIS, I Planet etc.

Generally

- ➢ PHP is completely free and Open Source.
- ➢ PHP can be easily embedded directly into HTML.
- ➢ Platform independent can run on Windows Linux or Mac servers.
- ➢ Run faster on the internet and easily integrate AJAX, Callback etc.
- ➢ Interfaces very easily with Apache/MySQL Lots of good books and on-line help.
- ➢ It's available with documentation in many languages.
- ➢ Easy to learn compared too many other scripting languages. It has a syntax that is easy to parse and is actually rather human-friendly.
- ➢ Pretty easy to access other web-based tools through PHP i.e. Google maps, etc.
- ➢ Lots of good source code out there to use and/or learn from, as well as many useful libraries for working with PDFs, graphics, etc.

4.2. Algorithm Used

In general PHP is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. Hypertext Preprocessor (PHP) is a server-side scripting language designed for web development but also used as a general-purpose programming language. For encryption, we use MD5 technique.

Pseudo code Algorithm

Fill the Login Form

Click the Login button

If (Form is filled)

       If (valid)

              Generate SQL select queries

              Connect to database

              Pass queries to database

If (any query fails)

              Display error message

              Else

Read session

If session exists on database, user is already logged in,

Display the page

Else

If they're correct

Create session ID

Store session ID on database

Display the page

End if

End if

       Else

              Display error message

              Ask the user to refill the form

       End if

4.3. Sample Code

CHAPTER FIVE

TESTING

Introduction

Developing software is a complex process. No matter how hard we try to eliminate all faults simply by going through the development phases which is requirements elicitation, requirement analysis, system design, and implementation, however through good practice we can make sure that the most series fault does not occur in the first place. In addition we need a separate testing phase, with the goal of elimination all remaining faults before release the system. Testing is the final phase of implementation. Testing is a process to show the correctness of the program. Testing is checking of the system workability in an attempt to discover errors and avoiding such errors from the system. In this the team members tested the entire system as a whole with all forms, code, modules. In this we tested all the functionalities in the System. All errors in the forms, functions, modules have been tested. [9]

5.1. Sample Test

To simplify the testing process, the project team followed the different types of tests that break the testing process up into the distinct levels. These types testing are unit testing, integration testing, acceptance testing and system testing. We have seen the following different testing strategies

5.1.1. Unit Testing

Unit testing is every module of the System is separately tested. It is often done by the programmer to test that the unit he/she has implemented is producing expected output against given input. Functional Test Specifications: that will be done in our proposed system the tests done as follow

- ➢ Test for login
- ➢ Test for user registration
- ➢ Test for user update user account
- ➢ Test for feedback

>   Authentication for login
>
>   Assumptions = Login into appropriate page
>   Test Data = User Name (empty, invalid user name, valid username).

5.1.2. Integration Testing

Integration testing is the activity of software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before acceptance testing. Thus, Integration testing is a logical extension of unit testing. In its simplest form, two units that have already been tested are combined into a component and the interface between them is tested. Here a component refers to an integrated aggregate of more than one unit.

In a realistic scenario, many units are combined into components, which are in turn aggregated into even larger parts of the program. The idea is to test combinations of pieces and eventually expand the process to test your modules with those of other groups. Integration testing can be done in a variety of ways but the following are two common strategies:

5.1.3. Acceptance Testing

Acceptance testing is the process of testing system prior to its delivery. A system is mainly developed for an end user normally a customer of the organization. A system is said to be accepted if and only if the user of the system is satisfied. In this perspective acceptance testing is widely used to prove that system performs as per the requirements. In acceptance testing the customers provides the input data to validate the system operation. It is also known as functional

testing, black-box testing, release acceptance, application testing, confidence testing, final testing and validation testing.

5.1.4. System Testing

It is the final step of testing. In this the team members tests the entire system as a whole with all forms, code, modules. This form of testing is popularly known as Black Box testing or System tests. In this the team members tests all the functionalities in the System. All errors in the forms, functions, modules are tested. [9]

CHAPTER SIX

CONCLUSIONS AND RECOMMENDATION

Introduction

This chapter is the last chapter for our project analysis phase. In this chapter we generalize our project idea and set the recommendation how the user use this system and what infrastructure are organization provide for the user.

6.1. Conclusions

The aim of this project is to develop computerized system for online blood bank management system in case of Debremarkos Town. This system is helpful for solving the work load of employee and blood screen able to check available blood and then request online. This project is developed by using object oriented system analysis method.

6.2. Recommendation and Future Enhancement

According to scope of the project the team develops web based application. Because of the time constraint we may have limitation which should be consideration in, but in the feature the team

believes that this system should be fully operationally by adding some functionality that are not included in the proposed system.

We also want to recommend this project can be expanded and more automated with additional functionalities by integrating with many new technologies. We recommend being included the following functionality:

- ➢ Online blood donation system
- ➢ Online testing human healthy
- ➢ By supporting all nation and nationalities languages.
- ➢ To do using digital signature technologies so as to make the system more secure.
- ➢ The system has to include video demonstration in order to support illustrate people to use the system easily.

EFERENCE

[1].S. A. Nwabueze et.al Perception of Blood Donation among Medical and Pharmaceutical Science Students of NnamdiAzikiwe University, Awka. Open Journal of Preventive Medicine.

[2]. WHO. National Blood Bank Service. [Cited 2014 November 20]; Available from:http://www.MOH.gov.et/nbbso.

[3]. https://www.sciencedirect.com/science/article/pii/S1877042815036940.

[4]. Donald Bell, IBM Global Services, UML basics: An introduction to the Unified Modeling Language, 2003

[5]. Mohammed Elkoutbi, Ismail Khriss and Rudolf K. Keller, User Interface PrototypingUsing UML Specifications, University of Montreal, Canada.

[6]. from OOSAD IT 2nd year handout

[7]. From Debremarkos Blood Bank Service.

[8]. https://its.ny.gov/sites/default/files/documents/systemreq.pdf

[9]Bharat blood bank,http://www.bharatbloodbank.com

[10]Jeevan blood bank bank,http://www.jeevan.org

APPENDIX

A.Questioners

- What is blood bank?
- When the organization was established?
- What is the role of organization?
- What are the organization services?
- How many employees worked under the organization?
- How the organization stored record?
- What are the problems on their works?
- Is the Organization currently uses computer system?
- What is the organization plan for the future?
- How the organization educates the blood donor?
- What is the organization structure?
- Who could request a blood?
- What are the criteria to request a blood?
- How the request will be handling in the organization?

B.Donor form

Federal Democratic Republic of Ethiopia
Ministry of Health National Blood Bank Service
BLOOD DONOR ENROLLMENT FORM

Name _____   City _____ Sub-city/Region _____

Date of Birth _____ Age _____ Sex _____   Zone _____ Woreda _____ Keb. _____ H.No _____

Occupation _____   Tele (Res)_____ Cell Phone _____

Reg No. _____   Tele (Off) _____ E-mail _____

Date	Pack No.	Wt.	Hgt.	B/P	Hct.	Vol.	Screened by	ABO	Rh	Type of Donation	Remark

Falcon Printing plc +251 11