# CERTIFICATE

I certify that this BSc industrial project report entitled <<web based blood bank management system for north Gondar>> blood bank by:

**Name**  **Signature**

Getachew Asmamaw

Tsehay Dawit

Abebe Sisay

Elsabet Sleshi

is approved by me for submission. I certify further that, to the best of my knowledge, the report represents work carried out by the students.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date Mr.Worku Abebe

**Name of coordinator Date Signature**

Mr. Setegn Asnakew \_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_

**Examining committee members’ Signature**

1.

2.

3.

It is approved that this project has been written in compliance with the formatting rules laid down by the university.

# DECLARATION

This is to declare that this project work is done under the supervision of Mr.Worku Abebe, and having the title <<Web based blood bank management system for North Gondar>>is sole contribution of:

Abebe Sisay GUR /1078/05

Elsabet Sleshi GUR/1152/05

Getachew Asmamaw GUR/11175/05

Tsehay Dawit GUR/1319/05

No part of the project work has been reproduced illegally (copy and paste) which can be considered as plagiarism. All referenced part has been used to argue the idea and have been cited properly. The project team will be responsible and liable for any consequence if violation of this declaration is proven.

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Group members:

Full name Signature

1. Getachew Asmamaw \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. Abebe Sisay \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3. Tsihay Dawit \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4. Elsabet Sleshi \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# ACKNOWLEDGMENT

First of all we want to thank the almighty ‘god’ keep our safe, staying power and energy starting from to get in to the friendship up to the day we have finished this project successfully. Secondly we would want to express out our deepest appreciation and gratitude to our adviser Mr.Worku Abebe for his advice and constrictive commit in our project what we have to do timely and efficiently. And thirdly we would like to thank information system academic staff for their guide line preparation which help us as framework to develop our project and Mr.Demeke Tilahun North Gondar blood bank manager who provide the information for to support, encouragement, comments and gives use full information in any time when we want to ask information he gives happiness face. Also like to thank his colleges that give help in requirement gathering time she has given more time and gave full information.

TABLE OF CONTENTS

[CERTIFICATE i](#_Toc453168472)

[DECLARATION ii](#_Toc453168473)

[ACKNOWLEDGMENT iii](#_Toc453168474)

[TABLE OF CONTENTS iv](#_Toc453168475)

[ACRONYMS x](#_Toc453168476)

[ABSTRACT xi](#_Toc453168477)

[CHAPTER ONE 1](#_Toc453168478)

[INTRODUCTION 1](#_Toc453168479)

[1.1 BACKGROUND OF THE PROJECT 1](#_Toc453168480)

[1.2 STATEMENT OF THE PROBLEM 2](#_Toc453168481)

[1.3 OBJECTIVES 3](#_Toc453168482)

[1.3.1 General Objective of the project 3](#_Toc453168483)

[1.3.2 Specific Objective of the project 3](#_Toc453168484)

[1.4 SCOPE AND LIMITATION OF THE PROJECT 3](#_Toc453168485)

[1.4.1 Scope of the project 3](#_Toc453168486)

[1.4.2 Limitation of the project 4](#_Toc453168487)

[1.5 SIGNIFICANCE OF THE PROJECT 4](#_Toc453168488)

[1.6 BENEFICIARIES OF THE PROJECT 5](#_Toc453168489)

[1.7 FEASIBILITY ANALYSIS 6](#_Toc453168490)

[1.7.1 Technical feasibility 6](#_Toc453168491)

[1.7.2 Operational feasibility 6](#_Toc453168492)

[1.7.3 Schedule feasibility 6](#_Toc453168493)

[1.7.4 Economical feasibility 7](#_Toc453168494)

[1.8 SYSTEM DEVELOPMENT METHODOLOGIES 9](#_Toc453168495)

[1.8.1 Data modeling techniques 9](#_Toc453168496)

[1.8.1.1 Data analysis methodology 10](#_Toc453168497)

[1.8.1.2 Data design methodology 10](#_Toc453168498)

[1.8.1.3Development tools 11](#_Toc453168499)

[1.9 TASK BREAKDOWN AND DELIVERABLES 12](#_Toc453168500)

[1.10 TIME SCHEDULE OF THE PROJECT 14](#_Toc453168501)

[CHAPTER TWO 15](#_Toc453168502)

[SYSTEM ANALYSIS 15](#_Toc453168503)

[2.1 INTRODUCTION 15](#_Toc453168504)

[2.2 EXISTING SYSTEM 15](#_Toc453168505)

[2.2.1 Existing System Description 15](#_Toc453168506)

[2.2.2 Problem of Existing System 15](#_Toc453168507)

[2.2.3 Alternative Solutions 16](#_Toc453168508)

[2.2.4 The Existing system performance is weak in case of following reasons 16](#_Toc453168509)

[2.2.5 Players of the existing system 17](#_Toc453168510)

[2.2.6 Forms and other Existing Systems Documents 17](#_Toc453168511)

[2.2.4.1 Donor enrollment form 18](#_Toc453168512)

[2.2.7 Business rules of existing system 18](#_Toc453168513)

[2.3 Requirement Gathering 19](#_Toc453168514)

[2.3.1 Requirement Gathering Methodologies 19](#_Toc453168515)

[2.3.1.1 Interview 20](#_Toc453168516)

[2.3.1.2 Record review 20](#_Toc453168517)

[2.3.1.3 Observation 20](#_Toc453168518)

[2.4 PROPOSED SYSTEM 20](#_Toc453168519)

[2.4.1 Overview of proposed system 20](#_Toc453168520)

[2.5 REQUIREMENT SPECIFICATION 21](#_Toc453168521)

[2.*5.*1 Functional requirement 21](#_Toc453168522)

[2.5.2 Nonfunctional Requirement 22](#_Toc453168523)

[*2.5.2*.1 User interface and human factor 22](#_Toc453168524)

[*2.5.2*.2 Documentation 22](#_Toc453168525)

[*2.5.2*.3Performance characteristics 22](#_Toc453168526)

[*2.5.2*.4 Error handling 22](#_Toc453168527)

[*2.5.2*.5 Quality Issues 22](#_Toc453168528)

[*2.5.2*.6 System Modifications 23](#_Toc453168529)

[*2.5.2*.7 Physical environment 23](#_Toc453168530)

[*2.5.2*.8 Security 23](#_Toc453168531)

[2.6 PLAYERS OF THE NEW SYSTEM 24](#_Toc453168532)

[2.7 SYSTEM MODEL 24](#_Toc453168533)

[2.7.1 Scenario 24](#_Toc453168534)

[2.7.2 Use Case Diagrams 26](#_Toc453168535)

[2.7.2.1 Essential Use case Diagram 26](#_Toc453168536)

[2.7.2.2 Proposed system Use case Diagram 28](#_Toc453168537)

[2.7.2.3 Use case documentation 29](#_Toc453168538)

[2.8 ACTIVITY DIAGRAM 44](#_Toc453168539)

[2.9 OBJECT MODEL 56](#_Toc453168540)

[2.9.1. Data Dictionary 56](#_Toc453168541)

[2.9.2. Class diagram 58](#_Toc453168542)

[2.9.3 Dynamic Modeling 59](#_Toc453168543)

[2.9.3.1 State chart diagrams 59](#_Toc453168544)

[2.9.3.2 Sequence Diagram 65](#_Toc453168545)

[2.10 USER INTERFACE DESIGN 79](#_Toc453168546)

[CHAPTER THREE 83](#_Toc453168547)

[SYSTEM DESIGN 83](#_Toc453168548)

[3.1 INTRODUCTION 83](#_Toc453168549)

[3.2 PURPOSE OF THE SYSTEM DESIGN 83](#_Toc453168550)

[3.3 DESIGN GOALS 83](#_Toc453168551)

[3.4 EXISTING SOFTWARE ARCHITECTURE 84](#_Toc453168552)

[3.5 PROPOSEDSOFTWARE ARCHITECTURE 84](#_Toc453168553)

[3.6 SUBSYSTEM DECOMPOSITION 84](#_Toc453168554)

[3.7 COMPONENT DIAGRAM 86](#_Toc453168555)

[3.8 DEPLOYMENT DIAGRAM 88](#_Toc453168556)

[3.9 PERSISTENCE DATA MANAGEMENT 88](#_Toc453168557)

[3.9 ACCESS CONTROL 90](#_Toc453168558)

[3.10 DETAILED CLASS DIAGRAM 91](#_Toc453168559)

[3.11 PACKAGE DIAGRAM 92](#_Toc453168560)

[CHAPTER FOUR 93](#_Toc453168561)

[IMPLEMENTATION AND TESTING 93](#_Toc453168562)

[4.1 INTRODUCTION 93](#_Toc453168563)

[4.2 CODING 93](#_Toc453168564)

[4.3 FINAL TESTING OF THE SYSTEM 96](#_Toc453168565)

[4.3.1 Black box testing 96](#_Toc453168566)

[4.3.2 White box testing 96](#_Toc453168567)

[4.4 USER MANUAL PREPARATION 98](#_Toc453168568)

[4.5 TRAINING 98](#_Toc453168569)

[4.6 INSTALLATION 98](#_Toc453168570)

[4.7 STARTUP STRATEGY 98](#_Toc453168571)

[CHAPTER FIVE 98](#_Toc453168572)

[CONCLUSION AND RECOMMENDATION 98](#_Toc453168573)

[5.1 CONCLUSION 98](#_Toc453168574)

[5.2 RECOMMENDATION 99](#_Toc453168575)

[5.3 FUTURE IMPROVEMENT 100](#_Toc453168576)

[APPENDICES 101](#_Toc453168577)

[Appendix I 101](#_Toc453168578)

[Appendix II 102](#_Toc453168579)

[REFERENCES 103](#_Toc453168580)

**List of Tables**

[Table 1 Lists of Hardware and Software costs 9](#_Toc453168186)

[Table 2 development tool 11](#_Toc453168187)

[Table 3 Task Breakdown and Deliverables 14](#_Toc453168188)

[Table 4 business rule1 18](#_Toc453168189)

[Table 5 business rule2 18](#_Toc453168190)

[Table 6 business rule3 19](#_Toc453168191)

[Table 7 business rule4 19](#_Toc453168192)

[Table 8 business rule5 19](#_Toc453168193)

[Table 9 business rule6 19](#_Toc453168194)

[Table 10 Use case description for Login 29](#_Toc453168195)

[Table 11 Use case description for manage account 30](#_Toc453168196)

[Table 12 Use case description for post information 31](#_Toc453168197)

[Table 13 Use case description for view report 32](#_Toc453168198)

[Table 14 Use case description for Approve 33](#_Toc453168199)

[Table 15 Use case description for view comment 34](#_Toc453168200)

[Table 16 Use case description for Hospital registration 35](#_Toc453168201)

[Table 17 Use case description for donation request 36](#_Toc453168202)

[Table 18 Use case description for generate report 37](#_Toc453168203)

[Table 19 Use case description for Give comment 38](#_Toc453168204)

[Table 20 Use case description for blood request 39](#_Toc453168205)

[Table 21 Use case description for collect blood 40](#_Toc453168206)

[Table 22 Use case description for donor registration 41](#_Toc453168207)

[Table 23 Use case description for blood distribution 42](#_Toc453168208)

[Table 24 Use case description for Manage stock 42](#_Toc453168209)

[Table 25 Data dictionary for Admin 56](#_Toc453168210)

[Table 26 Data dictionary for Donor 56](#_Toc453168211)

[Table 27 Data dictionary for Nurse 57](#_Toc453168212)

[Table 28 Data dictionary for Hospital 57](#_Toc453168213)

[Table 29 Data dictionary for Inventory manager 57](#_Toc453168214)

[Table 30 Access control 90](#_Toc453168215)

[Table 31 Test case to register 96](#_Toc453168216)

[Table 32 Test case to login 97](#_Toc453168217)

List of Figures

[Figure 1 Task Breakdown 13](#_Toc453171687)

[Figure 2 Project time schedule 14](#_Toc453171688)

[Figure 3 donor enrollment form 18](#_Toc453171689)

[Figure 4 Essential Use case Diagram 27](#_Toc453171690)

[Figure 5 Proposed system use case diagram 29](#_Toc453171691)

[Figure 6 Activity diagram for login. 44](#_Toc453171692)

[Figure 7 Activity diagram for Manage account 45](#_Toc453171693)

[Figure 8 Activity Diagram for Hospital registration 46](#_Toc453171694)

[Figure 9 Activity diagram for post new information 47](#_Toc453171695)

[Figure 10 Activity diagram for View report 48](#_Toc453171696)

[Figure 11 Activity diagram for Approve 49](#_Toc453171697)

[Figure 12 Activity diagram for View comment 49](#_Toc453171698)

[Figure 13 Activity diagram for Donation request 50](#_Toc453171699)

[Figure 14 Activity diagram for give Comment 51](#_Toc453171700)

[Figure 15 Activity diagram for Send blood request 52](#_Toc453171701)

[Figure 16 Activity diagram for Collect blood 53](#_Toc453171702)

[Figure 17 Activity diagram donor registration system 53](#_Toc453171703)

[Figure 18 Activity diagram for blood distribution 54](#_Toc453171704)

[Figure 19 Activity diagram for manage stock 55](#_Toc453171705)

[Figure 20 Class diagram 58](#_Toc453171706)

[Figure 21 State diagrams for Login 59](#_Toc453171707)

[Figure 22 State Diagram for donor registration 60](#_Toc453171708)

[Figure 23 State Diagram for view information 61](#_Toc453171709)

[Figure 24 State Diagram for add blood 62](#_Toc453171710)

[Figure 25 State Diagram for add new information 63](#_Toc453171711)

[Figure 26 State Diagram for approve request 64](#_Toc453171712)

[Figure 27 Sequence Diagram for login 65](#_Toc453171713)

[Figure 28 Sequence Diagram for Manage account 66](#_Toc453171714)

[Figure 29 Sequence Diagram for post information 67](#_Toc453171715)

[Figure 30 Sequence Diagram for View report 68](#_Toc453171716)

[Figure 31 Sequence Diagram for Approve 69](#_Toc453171717)

[Figure 32 Sequence Diagram for View comment 70](#_Toc453171718)

[Figure 33 Sequence Diagram for Hospital Registration 71](#_Toc453171719)

[Figure 34 Sequence Diagram for Donation request 72](#_Toc453171720)

[Figure 35 Sequence Diagram for give comment 73](#_Toc453171721)

[Figure 36 Sequence Diagram for Send blood request 74](#_Toc453171722)

[Figure 37 Sequence Diagram for Collect blood 75](#_Toc453171723)

[Figure 38 Sequence Diagram for donor registration 76](#_Toc453171724)

[Figure 39 Sequence Diagram for blood distribution 77](#_Toc453171725)

[Figure 40 Sequence Diagram for Manage stock 78](#_Toc453171726)

[Figure 41 User interface prototype for login 79](#_Toc453171727)

[Figure 42 User interface prototype for create account 80](#_Toc453171728)

[Figure 43 User interface prototype for comment 80](#_Toc453171729)

[Figure 44 User interface prototype for donor registration 81](#_Toc453171730)

[Figure 45 Home page 82](#_Toc453171731)

[Figure 46 Proposed software architectures 84](#_Toc453171732)

[Figure 47 Subsystem decomposition 85](#_Toc453171733)

[Figure 48 Component diagram 87](#_Toc453171734)

[Figure 49 Deployment diagram 88](#_Toc453171735)

[Figure 50 Persistent data management 89](#_Toc453171736)

[Figure 51 Detailed class diagrams 91](#_Toc453171737)

[Figure 52 Package diagram 92](#_Toc453171738)

# ACRONYMS

FMOH: Federal ministry of health

CDC: Central disease control and prevention

ERCS: Ethiopia Red Cross society

OOSD: Object oriented system development methodology

Reg.no: Registration number

Hep b: hepatitis b

Hep c: hepatitis c

ABO: blood group

Exp.date: Expired date

Fig : Figure

Php: Php: Hyper Text Preprocessor

Admin: Administrator

DB: Database

GUI: Graphical User Interface

HTML: Hypertext markup Language

INFO: Information

MS: Microsoft

MYSQL: Structured Query Language

CSS: Cascading style sheet

WBBMS: Web based blood bank management system

OS: Operating System

CDROM: Compact Disk Read Only Memory

# ABSTRACT

The intension of this project is developing a supplemental blood bank system to North Gondar Zone.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. 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 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 requesters) 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**.**

# CHAPTER ONE

# INTRODUCTION

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

The system manage blood transfusion process starting from donor registration, blood collection, blood testing, storing blood in blood stock and distribution of blood for the hospitals through patient name.

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 through patient name to get blood from blood bank. Our project can manage blood types, quantity and expiry dates for each category of blood that stored in blood transfusion unit. System shows total amount of blood inside the stock for efficiently management of blood bank in addition to that, this system show to the inventory manger which blood types are expired.

# 1.1 BACKGROUND OF THE PROJECT

In Ethiopia blood bank service introduced in 1969 on yikatet12 hospitals. Ethiopian blood bank supported by FMOH and CDC Separated from ERCS since 2004.

Blood bank system in Ethiopia has one main bank in Addis Ababa and 26 sub branch around all Ethiopia. North Gondar blood bank is one of 26 blood bank branches.

North Gondar blood bank was established by federal government and Amhara regional state since in 2006 E.C. this organization was begin giving small capacity of blood service. The aim of the organization is to provide efficient service to user, to increase the capacity to provide blood for the recipient, to increases the number blood donors by teaching the society and facilitating blood donating service for the blood donors. From time to time the capacity of it is increasing but the organization was working with manual system. Since the organization is using manual system this makes the employee to fail with data redundancy and erroneous data storing. Generally this organization was established to save the life of people who are affected by these problem like accidents, cancer, sickle cell, premature and surgery.

Ethiopian people’s are donating and accepting blood time to time, there has been a good rise in the number of people who donate blood but the system is very old and paper based, therefore we are very interested to make computerized the system “North Gondar blood bank management system”.

# 1.2 STATEMENT OF THE PROBLEM

In the current system documenting, writing, finding and searching of the specific information of the blood is done manual. The administrators cannot manage the blood stock starting from the blood collection, to blood screening, processing, storage, distribution and lastly transfusion through this system. Moreover, there is no logging function available So that each process or workflow cannot be traced from the database. The system not showed the expired blood. These types of system make the worker to document erroneous and redundancy information and it also consume the time of worker for completing specific task. So we have identified the following problem in the existing system.

* Document Mismanagement: Chances for losing the vital information related to hospital, blood and blood donors.
* Difficult to know amount of blood in stock.
* Error handling is not efficient
* Requires More Resources: Consumes more resources and costs such as paper, pen.
* It becomes tedious for a person to search blood in case of emergency.
* There is also no centralized database used to keep the donors' records.
* Security is less.
* The system not efficient, reliable, available and difficult to get data fast from the paper.
* The system does not show the inventory manager when the blood expired.
* Difficult to prepare organized report.
* Difficult to give pack number to collected blood.
* Difficult to know fit blood during blood transfusion service.
* Difficult for making comment the system user to blood bank services.
* Less awareness among people about blood donation and blood transfusion services.
* Difficult to know how match time have donor donated blood before.

# 1.3 OBJECTIVES

## 1.3.1 General Objective of the project

The general objective of this project is to develop web based Blood Bank management system for North Gondar.

## 1.3.2 Specific Objective of the project

* Gathering requirements.
* Identifying and defining of the problem that the existing system have.
* Analyzing the existing system.
* Identifying functional and nonfunctional requirements.
* Design interactive user interfaces for the North Gondar web based blood bank management system to users.
* Create a database to register blood donor, hospital and to store personal profiles for individual donor and hospital of the blood bank.
* Coding and testing.
* Implementation of the new system.
* Prepare the documentation and train the users.

# 1.4 SCOPE AND LIMITATION OF THE PROJECT

## 1.4.1 Scope of the project

* Donor registration, adds new blood, and hospital request.
* Keep the record of all donors.
* Search, update and discard the expired blood from database.
* Shows how much blood inside the stock and separates blood by their blood group.
* Gives pack number to the collected blood.
* Store screened unscreened and infectious blood separately.

## 1.4.2 Limitation of the project

Our project will not include the following.

* Not support all nation and nationalities of Ethiopian language it only use English language.
* The system cannot help people that have visual impairment because of lack of sound sensor machine.

# 1.5 SIGNIFICANCE OF THE PROJECT

The current system is not using computerized data processing System so making the system online will give many benefits from the following four aspects. These are:

**For Blood donors**

* + It provides the unique identification number easily at the time of blood donation camp which helps him for the future correspondence.
  + The system gives the unique user id and password for those donors who are applying online. This feature helps administrator to collect the information of all the donors area wise and blood group wise.
  + Donors can view the blood donation camp organizing at the different places.
  + Donor can check the status of the particular blood group just on one click sitting at home**.**

**For blood seekers**

* Seeker can get the information of the desired blood group from the central inventory
* Seeker can get the list of donors’ area wise, blood group wise if the desired blood group is not available in the central inventory.
* Seeker can get the information of that blood group which is not fit for blood transfusion.

**For the organization**

* 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 can be generated of donors, seekers, total consumption of the blood units and overall report.
* The organization can get the information which blood is in demand but rarely available and which blood group is rarely in demand but plenty in stock.
* Can get the information which is maximum cause for which the blood units are required like accidental cases, heart surgery, delivery cases.
* Can view the list of discarded blood units, they can also view the reason for which the blood units are discarded.
* Can view the central inventory as it shows the total account number of units of the particular blood group.

**For The Developers**

* When we develop this system our performance will increase.
* When we gather information by interview method our communication skill will be improved.
* Develop our team knowledge and skills how to conduct and prepare a website.
* Our problem solving skill will be increase.

# 1.6 BENEFICIARIES OF THE PROJECT

There are different bodies that will be benefited from this system. The main beneficiary of this system is North Gondar blood bank office in which, the system will be changed to web based blood bank system.

Developing web based blood banking system provides Benefit for the following users:-

* Patient:-It helps patients by giving services that are needed efficiently and effectively, patient can see the blood available online and request for it if he/she want to take.
* Nurses:-Nurses are helpful from this by registering donors and collected blood registration web basedly.
* Administrator:-Can administer the system simply by creating, updating and deleting account.
* Hospital:-Hospital can send the blood request it wants online. So, the hospital is beneficiary from this web by saving time and cost.
* Inventory manager:-Can manage the stock in a convenient way because he/she do activities by automated system.
* Donor:-Donor can send donation request online. So, he/she is beneficiary by saving their times that is waste for donation.

# 1.7 FEASIBILITY ANALYSIS

To bring the successful completion of this project goals and objectives the feasibilities issues listed below has determined the project viability or the discipline of planning, organizing, and managing resources.

## 1.7.1 Technical feasibility

As the system has been built by concentrating on the graphical user interface concepts, the application can also be handled very easily with a novice user. This involves questions such as whether the most of the technology need for the system has exists in the North Gondar blood bank. Because the system is too customize according to North Gondar blood bank information transferring system. The system is self-explanatory and does not need any extra sophisticated training.

## 1.7.2 Operational feasibility

The system will provide adequate through put at desire time to the user and also give the need information in a timely usefully formatted way. The system also has security to gives access privilege providing account for an authorize person. This system provides help description to the user about how to use the system. And other technical modification on the system is done by the developers.

## 1.7.3 Schedule feasibility

The project is intending to be with in fixed time interval. Unless and otherwise we may fall in difficulty in cost estimation and as well as we may fall in the financial problem. To overcome such problems we have fixed the time interval in which we should finish the development of the whole time among different section of project.

## 1.7.4 Economical 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 advertization 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 advertization 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**. **Cost Reduction and Avoidance**: - To calculate the following things will be consider.

* Total number of workers in north Gondar blood bank of the existing system= 15
* Average Salary of each workers per month = 1500.00Birr
* Total money required for payment per year= 15\*1500\*12= **270,000.000Birr**
* **Total** stationary materials like( pen ,print and paper ) expense daily in the existing system =150 birr
* Average monthly expense= 30\*150=4500.00birr
* Total stationary expense for year=4500\*12=**54000.00birr**
* **Total cost** = Total stationary expense per year + Total money required for payment per year=270,000.000Birr+54000.00birr=**324,000.00birr**
* Average Number of workers needed when the new system is deployed= 8
* Average salary of each of them per month = 1800.00Birr
* Total money required for payment per year= 8\*1800\*12= 172,800.00Birr
* And total stationary expense per year in the new system= 15000.00birr
* **Total cost** required per year in the new system=172,800 + 15000=**187,800.00birr**
* Difference between before and after deployment money required for expense
* Cost Reduction and Avoidance= 324,000.00birr -187,800.00Birr= **136200.00Birr**

**Net Profit** == **136200.00Birr**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Stationary Costs   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Name | Unit | Amount | Unit price | Total price | | Pens | Piece | 20 | 4birr | 80 birr | | copy | - | - | 200 | 200 | | Note book | 1 | 1 | 45 | 45 | | Paper | Mass | 1packet | 120 | 120 birr | | transportation | - | 10 | 12 | 120 | | **Total** | | | | 565 birr | |

Table 1 Lists of stationary cost

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Types of cost | Name | Quantity | Unit Price | Total price |
| Hardware cost | Computer | 1 | Free | Free |
| Flash(8GB) | 1 | 150 | 150 |
| CD ROM | 1 | 10 | 10 |
| Print | - | 200 | 200 |
| Other material | - | 300 | 300 |
| Software cost | Xampp server | 1 | Free | Free |
| Notepad++ | 1 | Free | Free |
| Edraw max | 1 | Free | Free |
| Microsoft office 2007 | 1 | Free | Free |
| Microsoft vision | 1 | Free | Free |
| Window 7 OS | 1 | Free | Free |
| Total cost | -- | -- | -- | 660 |

Table 1 Lists of Hardware and Software costs

# 1.8 SYSTEM DEVELOPMENT METHODOLOGIES

## 1.8.1 Data modeling techniques

This project involves building a dynamic web-based blood bank management system. In order to achieve our project, an appropriate software design methodology would be chosen **iterative data model.**

Typically iterative development is used in conjunction with incremental development in which a longer software development cycle is split into smaller segments that build upon each other.

We choose this iterative data model because of it:-

* Building and improving the product step by step.
* Can get the reliable user feedback.
* Less time is spent on documenting and more time is given for designing.
* Can only create a high-level design of the application before we actually begin to build.

The phases which come under the iterative data model method are as follows:

* Requirement gathering
* Requirement Analysis
* Designs
* Implementation and testing

### 1.8.1.1 Data analysis methodology

After gathering different information from stakeholders the project team has analyzed requirements by using Unified Modeling Language models like use case diagram, sequence diagram and class diagram. Since:-

* UML is a modeling language widely used to visualize the object oriented designs.
* UML makes it easy to visualize the software design.
* UML diagrams can be easily decoded and converted into most of the popular object oriented programming languages.

### 1.8.1.2 Data design methodology

The team plan to use the object oriented system analysis and design Development Methodology for the development of the system among the different methodologies. Because it is better way to construct, manage and assemble objects that are implemented in our system.

We used OOSAD because of the following important features:

**Increase reusability**: - the object oriented provides opportunities for reuse of blood bank information easily updating or modifying by the form.

**Extensibility: -**when you to need to add new feature to the blood bank system you only need to make changes in one part of the applicable class.

**Improved quality**: -blood bank information management system improved quality and efficiency to blood transfusion services effectively by accept system user feedback and also easily managing the blood stock.

**Real-World Modeling:**  Object-oriented systems tend to model the real world in a more complete fashion than do traditional methods.

**Reduced maintenance cost**: - Software organizations currently spend significant resources maintain operating system by developing manual documentation of our system so the object oriented development methods helps us to overcome this problem.

**Managed complexity**:-The object oriented methods solve our system complexity in the following way, by design our system software expectation that it will need to be modified and being able to respond quickly when our system environment changed.

### 1.8.1.3Development tools

The team plan to use the following software Development tools for the development of the system among different available tools.

|  |  |  |
| --- | --- | --- |
| **Activities** | | **Tools/program** |
| Documentation | | MS word 2007/8 |
| Design | | Edrawmax |
| Editing | | Paint |
| Client side script | | java script |
| Database server | | MYSQL server |
| Interface | | Html |
| Presentations | | PowerPoint |
| Web server | Apache Wamp server | |
| Implementation | html, php, notepad++,CSS | |

Table 2 development tool

# 1.9 TASK BREAKDOWN AND DELIVERABLES

The work breakdown structure is a view into the project which shows what work the project encompasses. It is a tool which helps to easily communicate the work and processes involved to execute the project. The project team uses the work breakdown structure to develop the project schedule, resource requirements and costs. Each part of tasks of the project will be delivered by collaborating with each of the group members. Each group members has their own responsibility to complete the project successfully.

Figure 1 Task Breakdown

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Members name | Email | Activity |
| 1 | GetachewAsmamaw | Getachewasmamaw10@gmail.com | In all activity |
| 2 | Tsehay Dawit | Sunnyedave.17@gmail.com | In all activity |
| 3 | Abebe Sisay | abebe.sisay@yahoo.com | In all activity |
| 4 | Elsabet Sleshi | Sleshielsabet@gmail.com | In all activity |

Table 3 Task Breakdown and Deliverables

# 1.10 TIME SCHEDULE OF THE PROJECT

The proposed system can be implemented in an acceptable timeframe given below.



Figure 2 Project time schedule

# CHAPTER TWO

# SYSTEM ANALYSIS

# 2.1 INTRODUCTION

In this chapter this project discuss about topics like how the current system is operating and how the proposed application intend to replace the existing system. The different requirement analysis techniques are also discussed under this chapter with their results found from those techniques. The major activities performed or identified are: modeling the function of the use case, identifying actors, identifying use case, constructing use case model and use case scenarios and final designing of user interface.

# 2.2 EXISTING SYSTEM

## 2.2.1 Existing System Description

Based on the data the team gathered the current blood bank of the North Gondar is a manual. In the existing system documenting, writing, finding and searching of the specific information of the blood is done manual. Moreover, there is no logging function available So that each process or workflow cannot be traced from the database. The system will not show the expired blood. These types of system make the worker to document erroneous and redundancy information and it also consume the time of worker for completing specific task.

## 2.2.2 Problem of Existing System

Due to the manual means been used by the blood bank, keeping information about donors and receivers, a lot problems are encountered which includes:

* The processes of donating require that the donors must be come and register at the blood bank.
* Less security.
* Wastage of resource and consuming storage space and time.
* Difficult to search, retrieve, update and delete the data about donors and other users of the system.
* The absence of electronic data storing mechanism it requires huge storage space.
* The current process requires high human-power.

## 2.2.3 Alternative Solutions

In order to overcome the current system problems that exist in the functioning of blood banking system, our project team members have put down alternative options. These are:-

* Changing the structure of manual system in to organized manner.
* Completely change manual system in to computerized and web based system without affecting the structure of the system.

The new system is designed to solve problem affecting the manual system in use. It is design to be used web based thereby relieving both donors, hospitals and staff from much stress as experienced from the manual system. This will do the analyzing and storing of information either automatically or interactively, it will make use of online access to internet. The new system gives full system functionality that is needed by system user to perform system functionality.

The proposed system will also have some other features like:-

* Accuracy in handling of data.
* Fast rate of operation and excellent response time.
* Flexibility it can be accessed at any time.
* Better storage and faster retrieval system.
* Accessibility from anywhere.

The above listed points are described what the project teams are proposed to do and since the proposed system is web based the donors and hospital can perform task efficiently and effectively.

## 2.2.4 The Existing system performance is weak in case of following reasons

**Efficiency:** due to the manual operation most of activities are likely to wastage of resources like papers, man power, time etc. to produce the corresponding outputs. This makes the current system ineffective while utilizing resources.

**Services:** the main users of the current system are hospitals and donor. The services given to the user are not flexible, reliable and expandable i.e. the users must there in the blood bank to get the services given by the system. Those services given by the system are limited to a particular area.

**Economical:** North Gondar blood bank performs all of its tasks manually which requires lot of man power to do that tasks, and needs huge amount of papers to store data manually. Consume money for advertization. This manual system spends more money for human resource, purchasing papers and other materials.

**Controlling:** All the records associated with the manual system are recorded and stored manually. The security that the system provides for the privacy of those records is not in good manner. The system shouldn’t provide sufficient protection for access and manipulation of the records associated with the system.

## 2.2.5 Players of the existing system

Players represent external entities that interact with the system. Due to this we will deal only with persons involved on those services or persons who have responsible for this work.

Here are the players involved:

* Donors:-people who are donating their blood to the system to save life.
* Nurse:-are persons who collect blood that the donors donate by packed it.
* Manager:-are office workers who manage different activities. This includes control blood transfusion, collect report from other workers, and announce new information.
* Hospital:-The organization which send blood request when patients needed blood.
* Inventory manager: -is a person who manages blood stock.

## 2.2.6 Forms and other Existing Systems Documents

Existing system uses different forms and reports to manipulate different records associated with different activities. Such as, donor registration form, hospital requests form through patient name. Some sample forms shown below.

### 2.2.4.1 Donor enrollment form

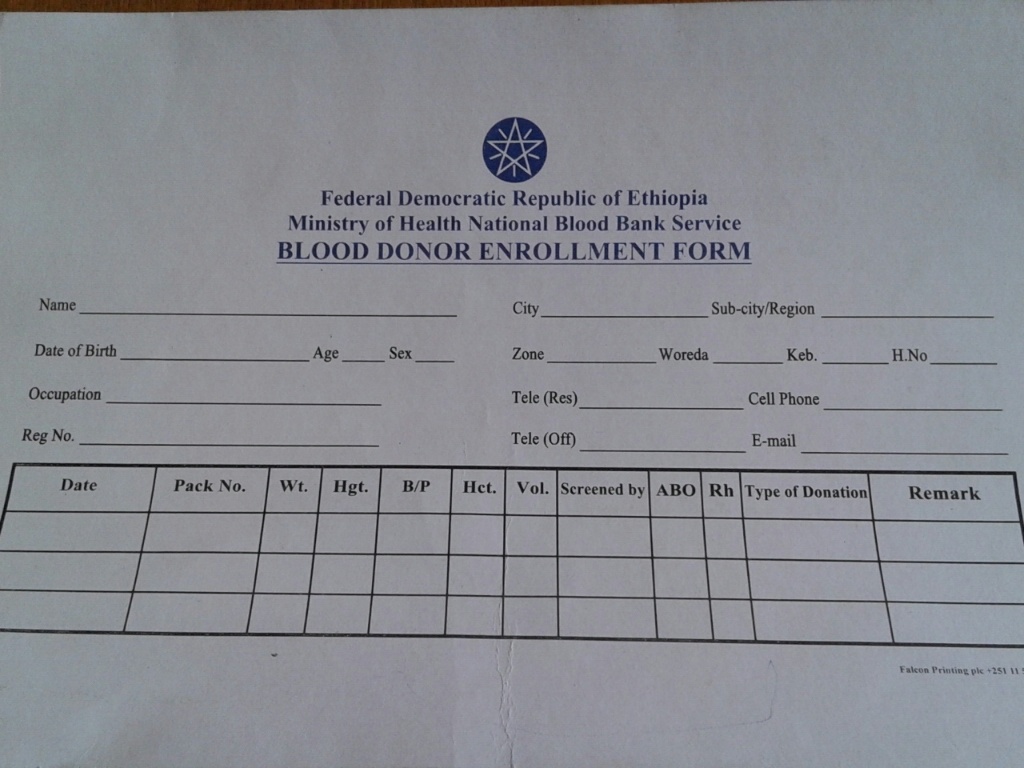


Figure 3 donor enrollment form

## 2.2.7 Business rules of existing system

Business rules are statements about the organization’s way of doing business. Organizations have policies in order to satisfy the business objective, satisfy customers, and make good use of resources, and conform to laws or general business conversions.

Basic business rule of the blood bank are:-

|  |  |
| --- | --- |
| Name | Registration |
| ID | BR1 |
| Description | The donor and hospital must be registered |

Table 4 business rule1

|  |  |
| --- | --- |
| Name | Test blood |
| ID | BR2 |
| Description | The lab technician must test the newly added blood. |

Table 5 business rule2

|  |  |
| --- | --- |
| Name | Blood request |
| ID | BR3 |
| Description | Hospital that send blood request must be members of the blood bank. |

Table 6 business rule3

|  |  |
| --- | --- |
| Name | Blood donation |
| ID | BR4 |
| Description | The donor must fulfill the required criteria to donate blood means their health status, age, weight etc. |

Table 7 business rule4

|  |  |
| --- | --- |
| Name | Distribute blood |
| ID | BR6 |
| Description | The inventory manager must distribute the collected blood. |

Table 8 business rule5

|  |  |
| --- | --- |
| Name | Collect blood |
| ID | BR7 |
| Description | The nurse must collect blood |

Table 9 business rule6

# 2.3 Requirement Gathering

## 2.3.1 Requirement Gathering Methodologies

During gathering and collecting necessary data and information needed for system analysis, the system use two major fact-finding techniques from both primary source and secondary source. In primary sources the project team uses the methodologies such as interview and observation. In secondary source the system obtain data from different document.

### 2.3.1.1 Interview

The team interviewed some staff members and manager of the North Gondar blood bank for necessary information. This information helps us to identify the blood bank problems and also to understand the current system. So, the project team analyzed the information of the blood bank and obtained some basic concepts on how the blood bank is managed in the current system.

Beginning with the basics, the team ask questions which describe and provide a background of fundamental details about the system.

### 2.3.1.2 Record review

This was the most beneficial technique for us while making our database. We studied the existing file structure, documents used and generated in the organization. For every files registered we tried to identify its significance need and relation with other data item. For example, flier papers and manuals prepared by the organization are some of records we have used.

### 2.3.1.3 Observation

In observation to find the facts the team observed all the activities and transaction usage of files and documents. Observation helped us in finding out the actual way functioning apart from the ideal or desired. This method is most useful when the team need to actually observe how documents are handled, how processes are carried out and whether specified steps are actually followed. While the team has observed the existing system, there have been improper handling of files that are too difficult to access it, and these make the decisions invalid. The teams know the way donating blood to collectors, at that time the team understands how much the donor and collector challenged with many processes.

# 2.4 PROPOSED SYSTEM

## 2.4.1 Overview of proposed system

The proposed web based North Gondar Blood bank management system that is to be developed is to minimize the problem of current system which is described in the problem statement. The system should be effective at the time of registration/insertion/, update, access, search and generate report. The proposed system that attempts to replace the manual system has the following functions. The system has high database security. Since it cannot be accessed by unauthorized user .But, each of member North Gondar Blood bank management system has its own privilege to do any operation on the database. The proposed system will also provide easily resource controlling mechanism and create simple communication between users and customers.

Generally the proposed system will able to minimize the existing problems and resource consumption because the blood bank activities are based on manual system.

# 2.5 REQUIREMENT SPECIFICATION

The functional and nonfunctional requirement of the proposed system is identify from requirement use cases associated with each actor and use case interaction. This enables the project team to have a clear idea about the structure and content of the information system.

## 2.*5.*1 Functional requirement

Functional requirements capture the intended behavior of the system. The proposed system is designed to do the following functionalities:

* Retrieve blood donors and hospitals from a database; the user/employee searches this information whenever it is required.
* Renew blood donors, blood group, hospital and blood testing information; administrator can update this information by assigning the new value to the old value.
* Discard old information from a database and infectious blood.
* Store new donors, hospital, blood group record in database, this is done by only privileged users of the bank.
* Generate of reports about the work of north Gondar blood bank management system by date per years per division of resource in a given time.
* Register and Keep record of donors and recipients in the database.

## 2.5.2 Nonfunctional Requirement

The non-functional requirement of the system deals with how well the system provides service to the user [1].

### *2.5.2*.1 User interface and human factor

Since users of the system will interface with the software to be deployed on a personal computer.

* Users can easily input and retriever customers profile and history
* Users easily navigate and access our web page by using user interface designed
* The system must be compatible with any environment and user friendly

### *2.5.2*.2 Documentation

Documentation will help the project team to make a knowledge management and also used for users to guide how to operate the system. Therefore it is a necessary requirement and it helps for maintenance purpose. The documentation is including proposal, project report, and final document.

### *2.5.2*.3 Performance characteristics

Since the system is going to be accessed by different users with different needs, it should be capable of handling and processing their queries quickly. Besides the software, Hardware will also be a great factor in the systems’ performance. Generally, the system should be able to handle many users and it will be responsive.

### *2.5.2*.4 Error handling

Errors could rise from users and from the system. Errors that occurred from the wrong doing of users will be handled by appropriate exception handling mechanisms. Generally, the system must handle the error and should display error message if the user inputted the characters that are mismatched to corresponding data types.

### *2.5.2*.5 Quality Issues

Since the system to be developed is web based and used different latest software when it will develop, the system should a fast and efficient service to all users. Adaptability, availability, flexibility, and reliability are the key issues of this requirement. Use suitable software and hardware to develop system, will able to achieve this requirement.

* **Availability:-**the system shall be available at any time for those who want to use it but it may be out of use when the power is turn off and if there is no connection.
* **Usability:-**

By training users to become familiar with the system and by designing user friendly interface, the end users are able to place an order within few response times. The system shall have a help support.

* **No Redundancy:-**The proposed system can be avoided reputation of data anywhere in the database.

### *2.5.2*.6 System Modifications

The System modification can be achieve easily because the system is going to be designed using an object oriented approach. The team would recommend that further work should be done on the system in order to make the system perform better like official website of others. For example currently, the system will not perform different function.

### *2.5.2*.7 Physical environment

The server must be put on a place that has high security room. And the client must put in the local area it must connect with server.

### *2.5.2*.8 Security

The administrator should provide high security interface for user and protect their personal data. The system administrator should regular evaluate risk, monitoring abnormal status and backup data. We are going to develop a secured database. There are different categories of users namely Administrator, Nurse, Lab technician, Donor who will be viewing either all or some specific information from the database. Depending upon the category of user the access rights are decided. It means if the user is an administrator then he can be able to modify the data, append etc. All other users only have the rights to retrieve the information about database.

* The system shall provide high level of security by blocking anyone to view system secured page.
* The external security should be provided by given the login authentication.

# 2.6 PLAYERS OF THE NEW SYSTEM

A new system contains different players to carry out its job. Among that different player, the most common are administrator**.**

The major actors in the new system are:

* **Admin**: a person who create, delete, update account and perform other administration related tasks.
* **Donor**: a person, who is voluntarily to donate blood,
* **Hospita**l: send blood request through patient name
* **Nurse**: collected blood from donor and give pack number for the blood component.
* **Inventory manger:** accept safe blood from lab technician. Mange stock and he also processes the request from the person through hospitals

# 2.7 SYSTEM MODEL

## 2.7.1 Scenario

This describes a particular sequence of activities within a use case. Flow of event Describes how and when use case starts and ends [1].

1. Scenario name: **Login.**

The admin enter username and password and select login option, then the system display home page for the admin

2. Scenario name: **Manage account.**

First admin must login into the system and the system display admin page then the admin select one of the listed links, for example if he wants to create new account click to create account link and system displays create account form then the admin fill correctly needed information and then the system displays account successfully created message.

3. Scenario name: **Post info.**

First the admin must login into the system and the system displays admin page then admin select add info menu and the system display rich text box form when the admin fill the form and select submit button, then the system display successfully added message.

**4.** Scenario name: **Hospital Registration**

First the admin must login into the system and the system displays admin page then admin select hospital registration menu and the system display registration form when the admin fill the form and select register button, then the system display hospital successfully registered message.

**5.** Scenario name: **Donation request**

First donor login into the system and the system display donor page, donor select donation request link and the system display donation request form the donor fill the donation request form and select send button then the system generate successfully send message.

7. Scenario name: **View report.**

First the system users must login into the system and the system display admin page then admin select view report menu the system display view report form and admin enters report date and select submit button then the system displays viewed report message.

8. Scenario name: **Approve**

First the admin must login into the system and the system display admin page then admin select approve request menu and the system display requested information and check the requested information and if it is correct the system generate successfully approved message.

9. Scenario name: **View comment.**

First the system user must login into the system select view comment menu then the system display the feedback given by the customer.

10. Scenario name: **Give comment.**

The system display user page and the user select comment the system link

then the system display the contact us page with feedback form, when the customer or the visitor fill the required field select submit after that the system display thanks for give us comment message.

11. Scenario name: **blood request.**

First the hospital admin must login into the system and system display hospital page, hospital admin select blood request link and the system display request form and admin fill request form and select send button then the system generate your request is successfully sent message.

12. Scenario name**: collect blood**

First Nurse must login into the system and system display nurse page, the nurse select collect blood link and the system display collect blood form then nurse fill collected blood information, the system check the filled info and if it is correct it generate the blood is successfully recorded message.

13. Scenario name: **donor registration**

First Nurse must login into the system and system display nurse page, the nurse select donor registration link and the system display registration form then nurse fill donor information, the system check the filled info and if it is correct it generate the donor is successfully registered message.

14. Scenario name: **Distribution**

First inventory manager must login into the system and the system display inventory manager page, the manager select distribution link and the system display distribution form and manager search the matched blood if exist distribute and select distribute button then the system generate successfully distributed message.

15. Scenario name: **Manage stock**

First inventory manager must login into the system and the system display inventory manager page, the manager select manage stock link and the system display manage stock form and manager select update, register or discard blood link if the manager want to discard the expired blood select it by its pack number and select discard button then the system generate blood is successfully discarded message.

## 2.7.2 Use Case Diagrams

The second step is to construct the use case model which graphically represents the interaction of the system with the external environment. The major actors that plays major role out of the system are: - administrator, nurse, lab technician, inventory manager, hospital and donor the use cases incorporated in the system are:-Login, manage account, add new information, approve, view report, Register donor, register hospital, donation request, add new blood, collect blood, blood request, blood distribution and searching process.

The following figure specifies the use case model of the system.

### 2.7.2.1 Essential Use case Diagram

Essential use case modeling is a simplified abstract, generalized use case that captures the intentions of the user in a technology and implementation independent manner. It identifies use case and actors of the proposed system [3].

Post

Information

Viewreport

Approve

Request

Distribute

Manage

Stock

View

Comment

Manager

INVENTORY

MANAGER

Hospital

Registration

Donation

Request

Give

Comment

Blood

Request

Donor

Registration

Collect

Blood

Donor

Hospital

Nurse

Generate

Report

Figure 4 Essential Use case Diagram

### 2.7.2.2 Proposed system Use case Diagram

Post

Information

View report

Approve

Request

Distribute

Manage stock

View

Comment

Manage

Account

Admin

INVENTORY

MANAGER

Hospital

Registration

Donation

Request

Give

Comment

Blood

Request

Donor

Registration

Collect

Blood

Add new

Blood

Discard

Blood

Create

Account

Update

Account

Delete

Account

Donor

Hospital

Nurse

Login

<<Include>>

Extend

Extend

Extend

Extend

Extend

Generate

Report

Figure 5 Proposed system use case diagram

### 2.7.2.3 Use case documentation

The third step is to document each of the above use case events to determine the requirement use cases as described in the following section.

Table 10 Use case description for Login

|  |  |  |
| --- | --- | --- |
| Use case id | UC#1 | |
| Use case name | Login | |
| Actors | Admin, Donor, Hospital, nurse, Inventory manager. | |
| Description | The authentication for authorized users in the system to interact with the system Gondar web based blood bank. | |
| Goal | To be accessed by an authorized and trust system user | |
| Precondition | Any user must have user name and password. | |
| Basic flow of action | Actor action | System response |
| **Step1:**user activate the system  **Step3:**user enter user name and password | **Step2:**system show login interface  **Step4:** the system check the authentication of user name and password  **Step5:**system display user page |
| Post condition | System transfer control to user main screen to precede actions. | |
| Alternative 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. | |

Table 11 Use case description for manage account

|  |  |  |
| --- | --- | --- |
| Use case id | UC#2 | |
| Use case name | manage account | |
| Actor | Admin | |
| Precondition | The System administrator must login to control the account | |
| Description | This activity is performed when the admin want to manage the account | |
| Goal | To control the system users activity. | |
| Basic course of action | Actor action | System response |
| **Step1:** admin enter user name and password  **Step4:** admin select mange account page :-  1.create account  2.update account  3.delete account  If create account  **Step6:**admin enter user account information  If update account  **Step10:**admin enter user account information to be updated  If to delete account  **Step13:** admin enter user account information to be deleted | **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 created user account information  **Step8:**System creates user account  **Step9:** System display update account page.  **Step11:**system check updated user account information  **Step12:** System display delete account page.  **Step14:**system check deleted user account information |
| Post condition | The system admin successfully create, update and delete the account! | |
| Alternative course of action | 1. If invalid entry. 2. The system displays error message.   **2.** Go to **step 6** to fill again. | |

Table 12 Use case description for post information

|  |  |  |
| --- | --- | --- |
| Use case id | UC#3 | |
| Use case name | Post information | |
| Actor | Admin | |
| Description | Post new information to create awareness and use full information | |
| Goal | To post new information to the blood bank users and customers | |
| Precondition | The System admin must login to post new information | |
| Basic course of action | Actor action | System response |
| **Step1:** admin enter user name and password  **Step4:** admin select post information link  **Step6:**System admin enter post available information | **Step2:** the system check the authentication of user name and password  **Step3:**the system display main admin page  **Step5:** System display post information page.  **Step7:**system check posted information  **Step8:**System posted new information |
| Post condition | The system posted information will be viewed by an authorized users | |
| Alternative course of action | 1. If the new submitted information is not valid. 2. The system displays error message. 3. Go to**Step5** to post again. | |

Table 13 Use case description for view report

|  |  |  |
| --- | --- | --- |
| Use case id | UC#4 | |
| Use case name | View report | |
| Actors | Donor, Hospital, nurse, Inventory manager. | |
| Description | View the reported that is generated by admin | |
| Goal | To view about all activities that have been done in organization | |
| Precondition | The actors must be log in to the system | |
| Basic flow of action | Actor action | System response |
| **Step1:** user enter user name and password  **Step4:**user select view report link | **Step2:** the system check the authentication of user name and password  **Step3:**system display user page  **Step5:** system display report |
| Post condition | The admin view report | |
| Alternative action | **A.** If no new report found to be reported.  **1.** The system displays error message.  **2.** Go to **step4** to view report again. | |

Table 14 Use case description for Approve

|  |  |  |
| --- | --- | --- |
| Use case id | UC#5 | |
| Use case name | Approve | |
| Primary actors | Admin | |
| Description | For Approving those who send donation request to donate blood. | |
| Goal | Give decision for the donor appointment | |
| Precondition | Admin to approve must be view the donor donation request | |
| Basic flow of action | Actor action | System response |
| **Step1:**admin enter user name and password  **Step4:**admin select approve form  **Step6:** admin search request and  if the request is valid approve | **Step2:** the system check the authentication of user name and password  **Step3:**system display admin page  **Step5:**the system display approve  **Step7:**system check information  **Step8:**system display the request is approved |
| Post condition | Send for donor session date to donate the blood | |
| Alternative action | The request is may be disapproved | |

Table 15 Use case description for view comment

|  |  |  |
| --- | --- | --- |
| Use case id | UC#6 | |
| Use case name | View comment | |
| actor | Admin | |
| Description | Users can see the comments that are submitted from the user, customer and other parties. | |
| Goal | To view user feedback about the system. | |
| Precondition | Login to the system | |
| Basic flow of action | Actor action | System response |
| **Step1:** admin enter user name and password  **Step4:**admin select view comment link  **Step6:**admin view comment | **Step2:** the system check 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 action | **A.** If there is no comment.  **1.** The system displays error message.  **2.** Go to **step4** to view comment again. | |

Table 16 Use case description for Hospital registration

|  |  |  |
| --- | --- | --- |
| Use case id | UC#7 | |
| **Use case name** | **Hospital Registration** | |
| actor | **Admin** | |
| Description | **for registration to get access from the blood bank** | |
| Goal | **To access the system and to be the system member.** | |
| Precondition | Go to the site and register | |
| Basic flow of action | **Actor action** | **System response** |
| **Step1:** admin enter user name and password  **Step4:Admin select registration link**  **Step6:Admin fill hospital registration form** | **Step2:** the system check the authentication of user name and password  **Step3:**system display admin page  **Step5:** the system display hospital registration form  **Step7:**system check hospital registration information  **Step8:**system Display successfully registered |
| Post condition | If valid successfully register if not valid Alternate action | |
| Alternative action | A. If not correctly fill to registered  1. The system displays error message.  2. Go to **step5** to fill again registration information | |

Table 17 Use case description for donation request

|  |  |  |
| --- | --- | --- |
| Use case id | UC#8 | |
| Use case name Use case id | Donation request | |
| actor | Donor | |
| Description | The donor should be visit the blood bank makes appointment date for the blood donation purpose. | |
| Goal | To specify the donation date. | |
| Precondition | Want to make appointment date for the donation | |
| Basic flow of action | Actor action | System response |
| **Step1:**Donor activate the system  **Step3:**donor select donation request link  **Step5:**fill the donation request form and send | **Step2:**system display main page  **Step4:**system display donation request menu  **Step6:**system save the request in DB |
| Post condition | If Get appointment date to donate blood | |
| Alternative action | 1. If donor do not fill the form correctly to send donation request   1. The system displays error message.  2. Go to **step5** to fill again donation request. | |

Table 18 Use case description for generate report

|  |  |  |
| --- | --- | --- |
| Use case id | UC#9 | |
| Use case name Use case id | Generate report | |
| actor | Admin | |
| Description | Generating the report that of activities have been done. | |
| Goal | To generate the required report information for users and customers | |
| Precondition | Generating to click generate button | |
| Basic flow of action | Actor action | System response |
| **Step1:** admin enter user name and password  **Step4:Admin select generate report link** | **Step2:** the system check the authentication of user name and password  **Step3:**system display admin page  **Step5:**system check report  **Step6:**system display the result |
| Post condition | Display the generated report | |
| Alternative action | A. If fail to generate  1.the system display error message  2. Go to **step6** to check again. | |

Table 19 Use case description for Give comment

|  |  |  |
| --- | --- | --- |
| Use case id | UC#10 | |
| Use case name | Give Comment | |
| actors | System user and customer | |
| Description | Comment the blood bank system about any thing | |
| Goal | To give the weakness and strength of the system | |
| Precondition | User must have valid email address comment the system | |
| Basic flow of action | Actor action | System response |
| **Step1:**User initiate the system  **Step3:**user select feedback link  **Step5:**user write comment about the system | **Step2:**system display user page  **Step4:**system display feedback form  **Step6:**system check comment information  **Step7:**system display comment submitted |
| Post condition | User send comment to the system | |
| Alternative action | 1. When fill to send the comment if not valid   1. The system displays error message.  2. Go to **step4** to fill again comment. | |

Table 20 Use case description for blood request

|  |  |  |
| --- | --- | --- |
| Use case id | UC#11 | |
| Use case name | blood request | |
| Primary actors | Hospital | |
| Description | Sending request for required blood unit, blood group. with the patient name just for the identification of blood accepter | |
| Goal | Asking blood from the blood bank for the patient. | |
| Precondition | Hospital send request to blood bank. | |
| Basic flow of action | Actor | System response |
| **Step1:**hospital enter user name and password  **Step4:**hospital select send blood request link  **Step6:**user fill the blood request in the name of patient | **Step2:** the system check the authentication of user name and password  **Step3:**system display hospital page  **Step5:**system display blood request form  **Step7:**system check blood request information  **Step8:**system display inserted request record |
| Post condition | Indirectly accept the required blood thorough patient name | |
| Alternative action | 1. If the hospital fails to fill the form correctly.   1. The system displays error message.  2. Go to **step6** to fill again the request. | |

Table 21 Use case description for collect blood

|  |  |  |
| --- | --- | --- |
| Use case id | UC#12 | |
| Use case name | collect blood | |
| Primary actors | Nurse | |
| Description | To Collect The Blood from voluntary donors by filling full information, | |
| Goal | To Collect The Blood from voluntary donors | |
| Precondition | Fill collected blood information | |
| Basic flow of action | Actor | System response |
| **Step1:**nurse enter user name and password  **Step4:**nurse select collect blood link  **Step6:** fill collect blood information  **Step8:**nurse print out record then give for donor | **Step2:** the system check the authentication of user name and password  **Step3:**system display nurse page  **Step5:**system display collect blood menu  **Step7:**system check collect blood information  **Step8:**system display collected blood record |
| Post condition | Submit collected blood information | |
| Alternative action | A. If nurse not enter valid entry to fill blood collection form  1. The system displays error message.  2. Go to **step6** to fill again the collected blood. | |

Table 22 Use case description for donor registration

|  |  |  |
| --- | --- | --- |
| Use case id | UC#13 | |
| Use case name | donor registration | |
| actor | Nurse | |
| Description | To register new donor and search the old donor for the blood collection mechanism | |
| Goal | To search the old blood donor. | |
| Precondition | Register new donor | |
| Basic flow of action | Actor | System response |
| **Step1:**nurse enter user name and password  **Step4:**nurse select view donor registration link  **Step6:**the volunteer old donor search() from donor registration and  If the new donor come fill the donor registration form  **Step8:**nurse view donor record information | **Step2:** the system check the authentication of user name and password  **Step3:**system display nurse page  **Step5:**system display donor registration form  **Step7:**the system check donor registration information  **Step8:**system display new donor information |
| Post condition | Register detail donor information | |
| Alternative action | **A.** If Nurse is filled invalid new donor registration information  1. The system displays error message.  2. Go to **step6** to register new donor or search old donor | |

|  |  |  |
| --- | --- | --- |
| Use case id | UC#14 | |
| Use case name | Distributed | |
| Primary actors | Inventory manager | |
| Description | **For Blood distribution.** blood accepter must be come using hospital blood donation request | |
| Goal | **To distribute the safe blood to the hospitals.** | |
| Precondition | Searching the blood fit from the stock | |
| Basic flow of action | Actor action | System responses |
| **Step1:** Enter User name & password  **Step4**:inventory manger select blood distribution link  **Step6:**search() from blood stock that fit with the blood accepter | **Step2:** the system check the authentication of user name and password  **Step3:**the system display inventory manager page  **Step5:**the system display distribution form  **Step7:**system check blood stock information  **Step8:**system display fit blood |
| Post condition | Distribute blood for the accepter | |
| Alternative action | A. If Filled entry to distribute blood is invalid  1. The system displays error message.  2. Go to **step6** to search fit blood from the stock for the distribution. | |

Table 23 Use case description for blood distribution

Table 24 Use case description for Manage stock

|  |  |  |
| --- | --- | --- |
| Use case id | UC#15 | |
| Use case name | Manage stock | |
| Primary actors | Inventory manager | |
| Description | Manage the amount of blood in stock too knows the amount of Blood per level and the expired blood. and also to insert new blood | |
| Goal | To manage the blood in the blood stock. | |
| Precondition | view the Details of the Blood Stock | |
| Basic flow of action | Actor | System response |
| **Step1**:enter user name and  Pass word  **Step4:**inventory manger select blood stock link  **Step6:**inventory manger select  1.register blood  2.discard blood  If select register blood  **Step7:**inventory manager fill blood registration form  If select discard blood  **Step11:**inventory manager check if blood expired  **Step12:**enter expired blood information and click discard | **Step2:** the system check the authentication of user name and password  **Step3:**the system display inventory manger page  **Steps5:**the system display mange stock form  **Step8:**system display blood registration form  **Step9:**system check register blood information  **Step10:**system display blood successfully registered  **Step13:**system display blood successfully discarded |
| Post condition | Add new blood to stock, view the Less amount of Blood per level and expired blood | |
| Alternative action | A. If invalid entry is filled to the inventory manger to add new blood or discard blood  1. The system displays error message.  2. Go to **step6** to fill again the add new blood or discard. | |

# 2.8 ACTIVITY DIAGRAM

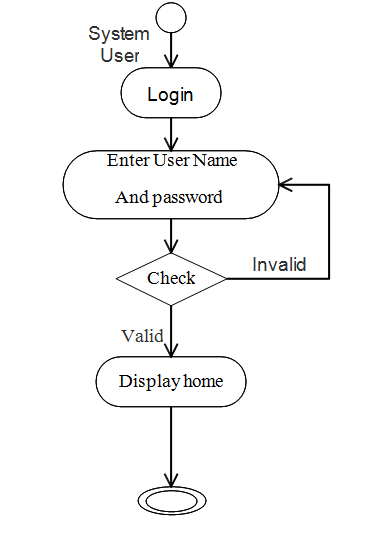


Figure 6 Activity diagram for login.

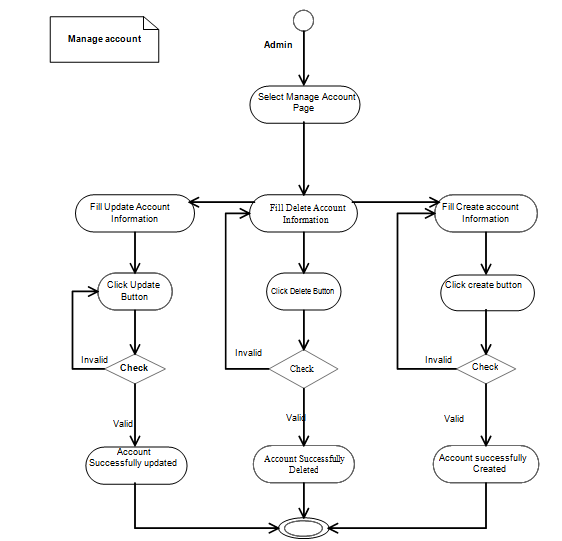


Figure 7 Activity diagram for Manage account

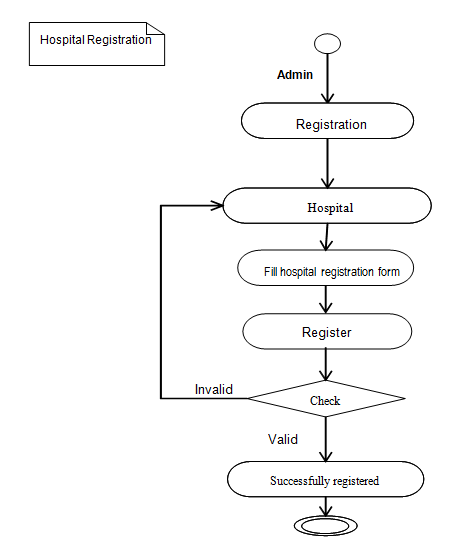


Figure 8 Activity Diagram for Hospital registration

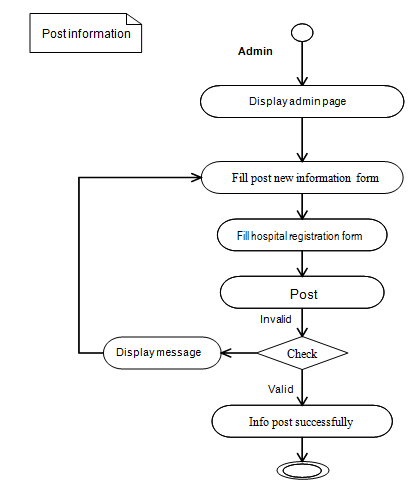


Figure 9 Activity diagram for post new information

## 

Figure 10 Activity diagram for View report

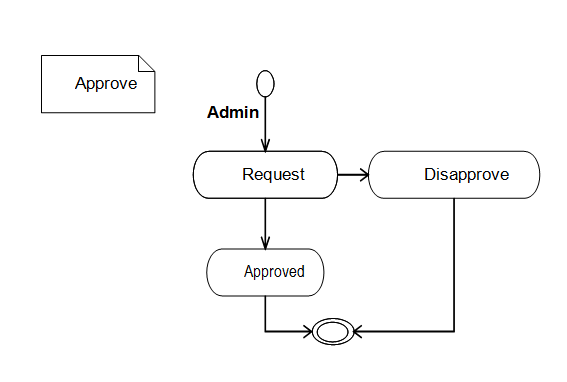


Figure 11 Activity diagram for Approve

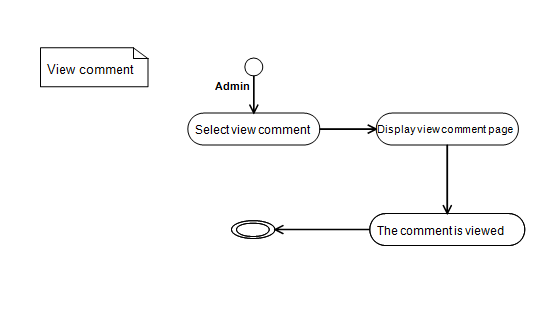


Figure 12 Activity diagram for View comment

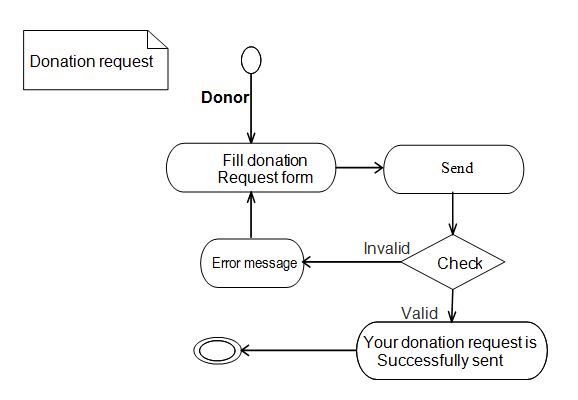


Figure 13 Activity diagram for Donation request

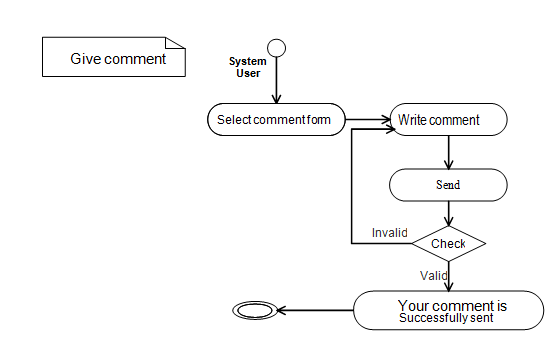


Figure 14 Activity diagram for give Comment

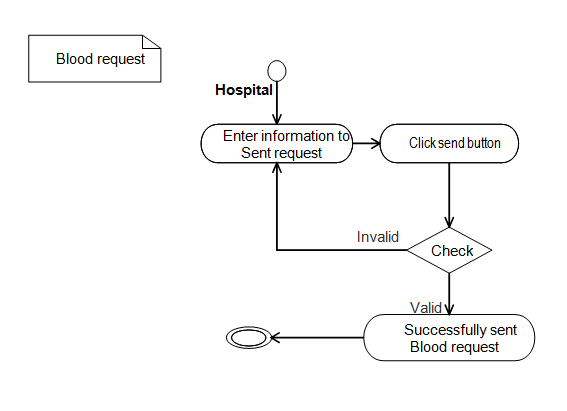


Figure 15 Activity diagram for Send blood request

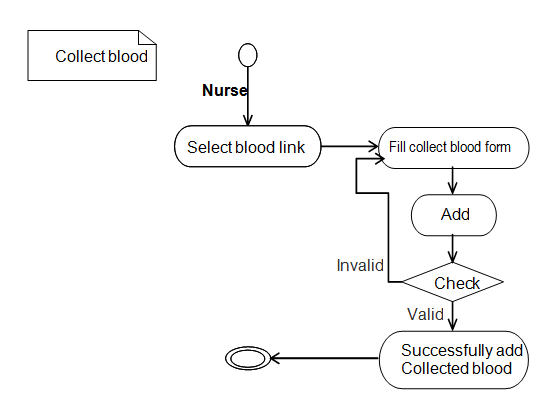


Figure 16 Activity diagram for Collect blood

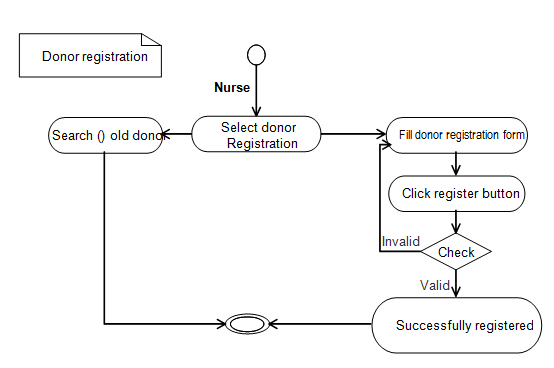


Figure 17 Activity diagram donor registration system

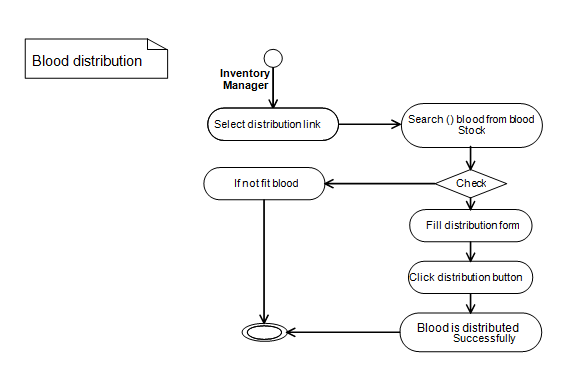


Figure 18 Activity diagram for blood distribution

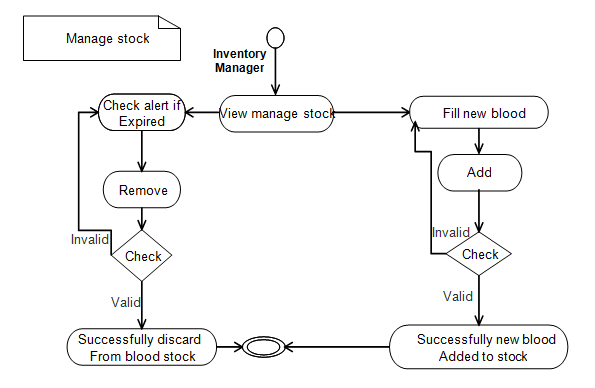
****

Figure 19 Activity diagram for manage stock

# 2.9 OBJECT MODEL

## 2.9.1. Data Dictionary

Table 25 Data dictionary for Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Primary/Foreign key | Field name | caption | Data type | Field size |
| P | ID | ID | string | 5 |
|  | Fname | First Name | Text | 20 |
|  | Lname | Last Name | Text | 20 |
|  | Age | Age | Integer | 2 |
|  | Email | Email | string | 11 |
|  | Ph.no | Phone number | string | 13 |

Table 26 Data dictionary for Donor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Primary/Foreign key | Field name | Caption | Data type | Field size |
| P | ID | ID | String | 5 |
|  | Fname | First name | Text | 20 |
|  | Lname | Last name | Text | 20 |
|  | Age | Age | Integer | 2 |
|  | Date of birth | Date of birth | String | 10 |
|  | Sex | Gender | Text | 4 |
|  | Zone | Zone | String | 10 |
|  | Woreda | Woreda | Text | 10 |
|  | Kebele | Kebele | String | 12 |
|  | Ph.no | Phone number | string | 13 |
|  | City | City | String | 15 |
|  | Email | Email | String | 16 |

Table 27 Data dictionary for Nurse

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Primary/Foreign key | Field Name | Caption | Data Type | Field Size |
| P | ID | ID | String | 5 |
|  | Fname | First Name | Text | 13 |
|  | Lname | Last Name | Text | 14 |
|  | Email | Email | String | 18 |

Table 28 Data dictionary for Hospital

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Primary/Foreign key | Field Name | Caption | Data Type | Field Size |
| P | ID | ID | String | 5 |
|  | Hname | Hospital Name | Text | 10 |
|  | P name | Patient Name | Text | 14 |
|  | Fax | Fax | Integer | 18 |
|  | Ph.no | Phone number | String | 13 |

Table 29 Data dictionary for Inventory manager

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Primary/Foreign key | Field name | caption | Data type | Field size |
| P | ID | ID | string | 5 |
|  | Fname | First Name | Text | 20 |
|  | Lname | Last Name | Text | 20 |
|  | Age | Age | Integer | 2 |
|  | Email | Email | string | 11 |
|  | Ph.no | Phone number | string | 13 |

## 2.9.2. Class diagram

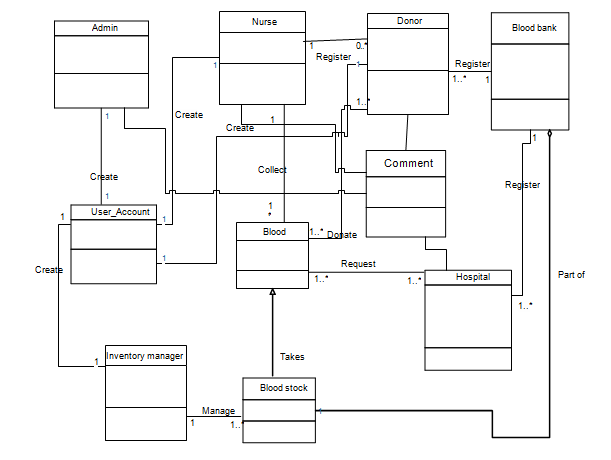


Figure 20 Class diagram

## 2.9.3 Dynamic Modeling

The dynamic model is used to express and model the behavior of the system over time. It includes support for state diagrams, sequence diagrams and extensions including business process modeling [1].

### 2.9.3.1 State chart diagrams

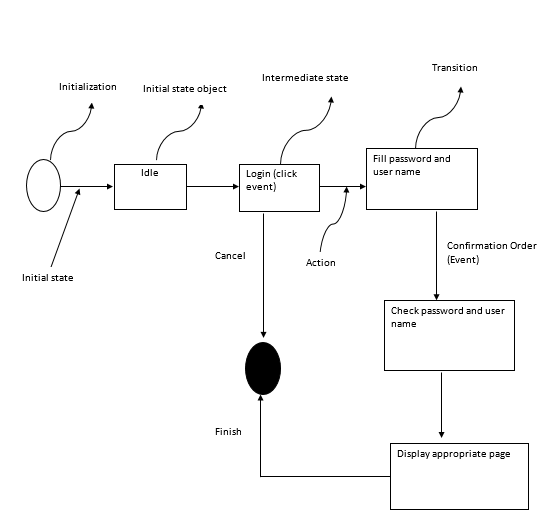


Figure 21 State diagrams for Login

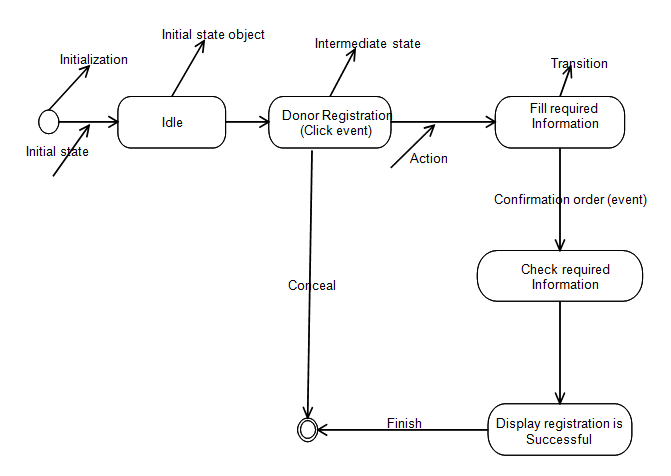


Figure 22 State Diagram for donor registration

### 

Figure 23 State Diagram for view information

### 

Figure 24 State Diagram for add blood

### 

Figure 25 State Diagram for add new information

### 

Figure 26 State Diagram for approve request

### 2.9.3.2 Sequence Diagram

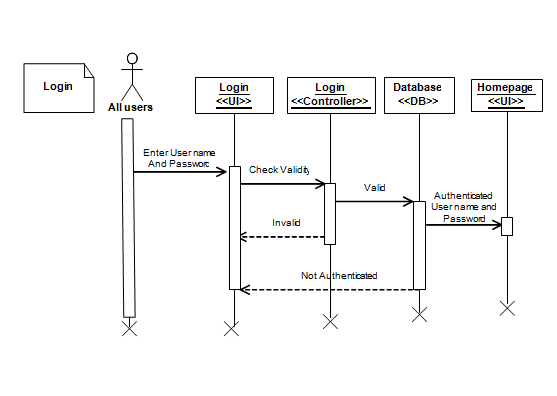
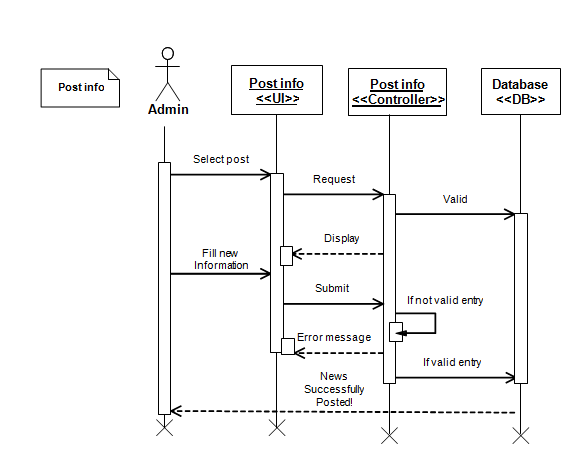
****

Figure 27 Sequence Diagram for login

Figure 28 Sequence Diagram for Manage account

**Manage**

**Account**

**<<UI>>**

**Manage**

**Account**

**<<Controller>>**

**Database**

**<<DB>>**

**Admin**

Request

Display

Valid

Create account

Account

Successfully

Created!

Fill user account

Information

Create

Error message

If valid entry

If not valid entry

**Manage**

**Account**

If you want to

Update account

Request

Display

Fill information

Update

If not valid

Error message

If valid

Successfully

Updated

Figure 29 Sequence Diagram for post information

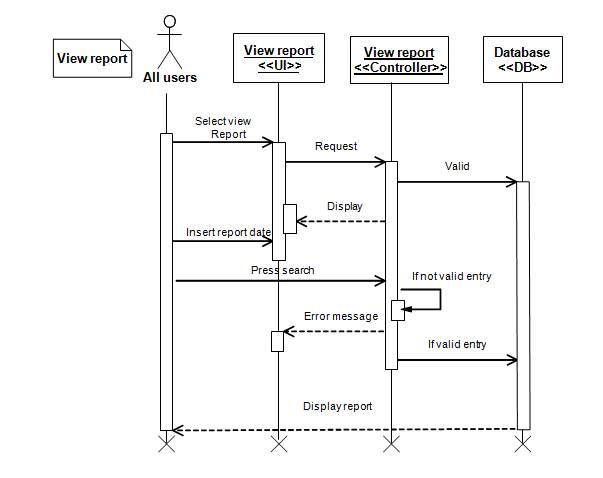


Figure 30 Sequence Diagram for View report

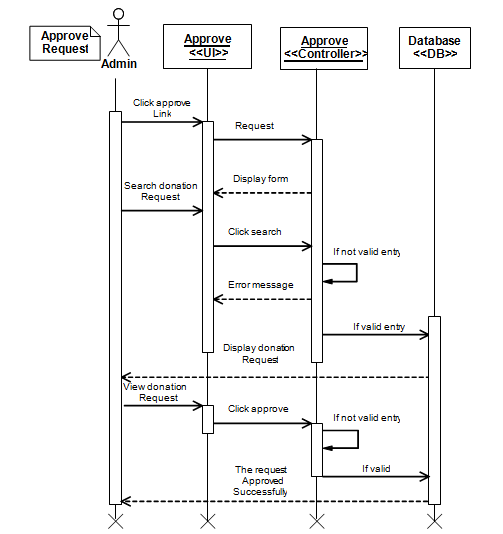


Figure 31 Sequence Diagram for Approve

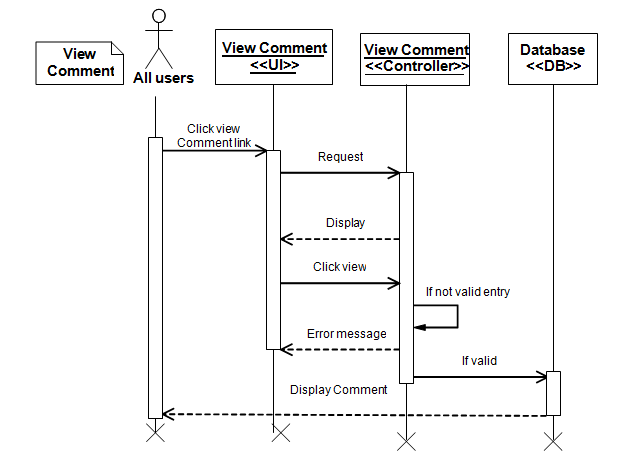
****

Figure 32 Sequence Diagram for View comment

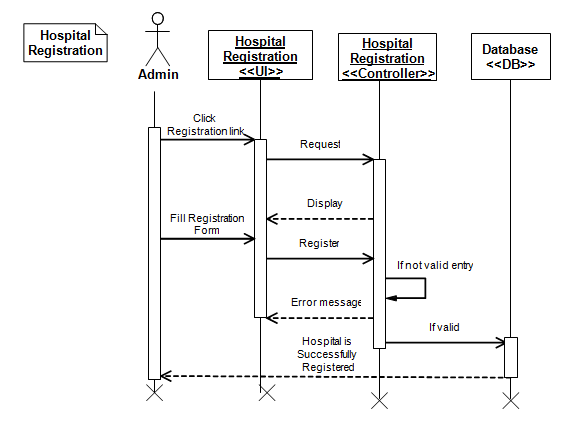
****

Figure 33 Sequence Diagram for Hospital Registration

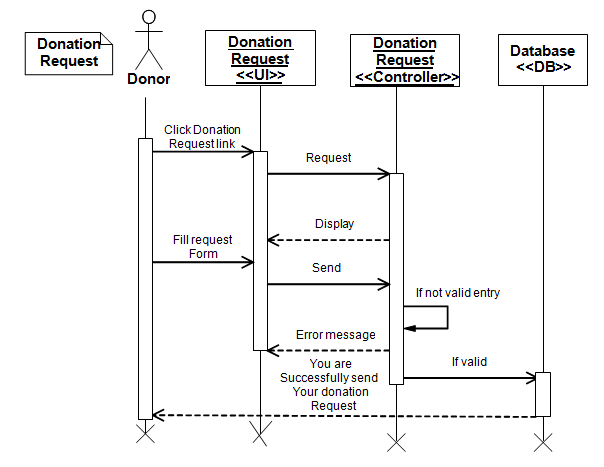


Figure 34 Sequence Diagram for Donation request

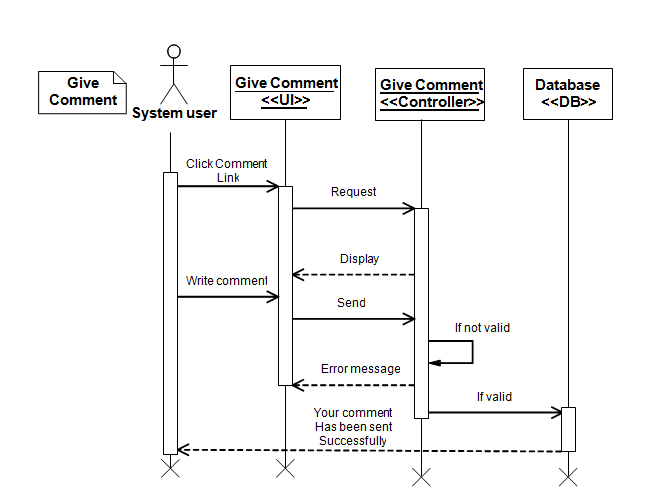


Figure 35 Sequence Diagram for give comment

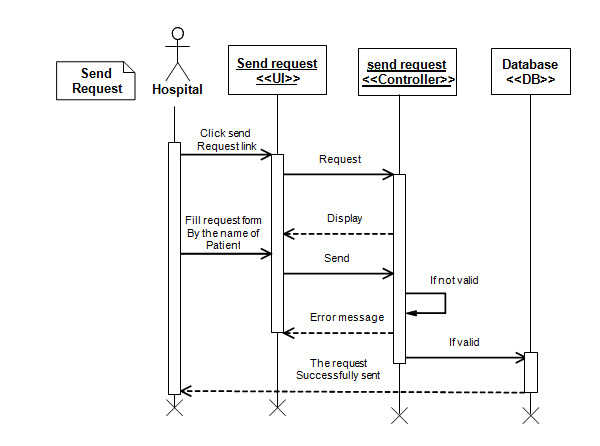


Figure 36 Sequence Diagram for Send blood request

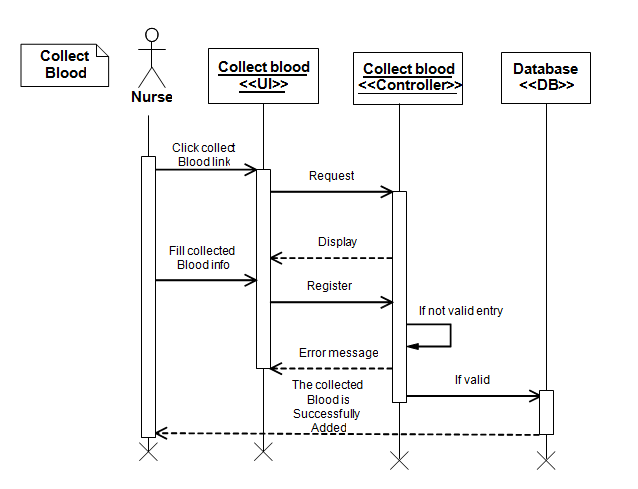


Figure 37 Sequence Diagram for Collect blood

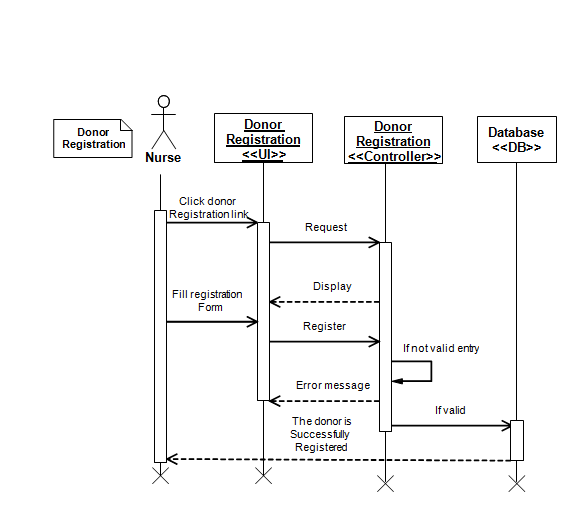


Figure 38 Sequence Diagram for donor registration

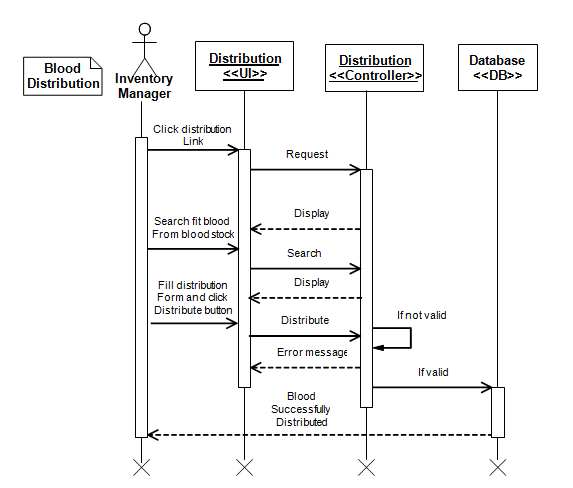


Figure 39 Sequence Diagram for blood distribution

**Manage stock**

**<<UI>>**

**Manage stock**

**<<Controller>>**

**Database**

**<<DB>>**

**Inventory**

**Manager**

Request

Display form

Valid

Select manage

Stock link

Blood

Successfully

Discarded from

Stock

View if blood is

Expired

Remove

Error message

If valid entry

If not valid

If you want to

Add new blood

Request

Display

Fill accepted

Blood form

Save

If not valid

Error message

If valid

New blood

Successfully

Added

**Manage**

**Stock**

Figure 40 Sequence Diagram for Manage stock

# 2.10 USER INTERFACE DESIGN

This is the design of the new system interface which includes the total layout of the main interface of the new system [1].

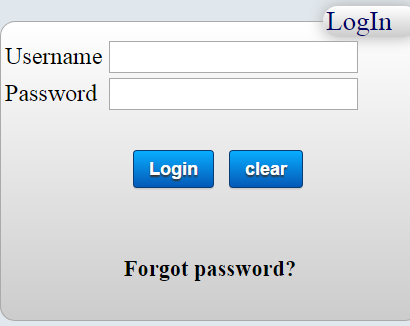


Figure 41 User interface prototype for login

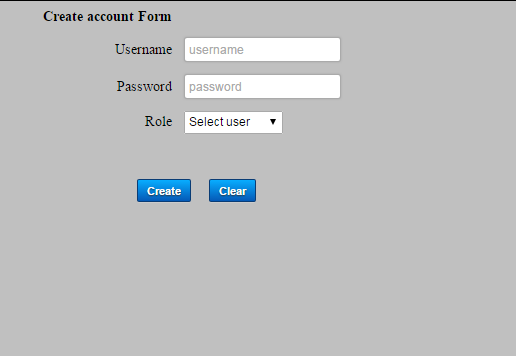


Figure 42 User interface prototype for create account

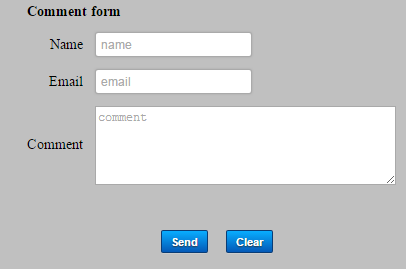


Figure 43 User interface prototype for comment

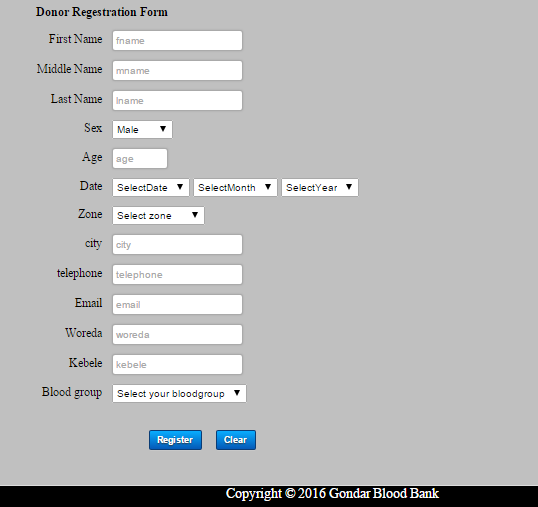


Figure 44 User interface prototype for donor registration

# 

Figure 45 Home page

# CHAPTER THREE

# SYSTEM DESIGN

# 3.1 INTRODUCTION

In this phase the overall procedures, activities and methods of execution during the implementation phase of the project are included. The following subtopics are discussed in this phase. These are component diagram, deployment diagram, and persistence diagram and user interface prototype of the project.

# 3.2 PURPOSE OF THE SYSTEM DESIGN

The main purpose of system design is to provide architecture design, design goals of the system, detailed class design and database design for the system. It is the process of defining and developing systems to satisfy specified requirements of the user.

Design converts functional models from analysis into models that represent the solution. This project is designed in a manner that solves the problems of the organization by minimizing the work load of the existing system and employee. It provides more efficient, reliable and time saving system.

# 3.3 DESIGN GOALS

The design part is very important so as to make the implementation very easy. The different types of the system modeling techniques that are used for the implementation of the system such as deployment and component modeling are show in detail. Not only the system modeling techniques but also some system design techniques such as system decomposition design are cover in detail in this phase. Some of the design goals are:-

# 3.4 EXISTING SOFTWARE ARCHITECTURE

The existing system of the North Gondar blood bank system is manual system and hence there is no Existing software architecture that will be considered. As a result, we only describe the software architecture of the newly proposed system [2].

# 3.5 PROPOSEDSOFTWARE ARCHITECTURE

Software architecture is structured into three layers by dotted lines. Each layer is an abstraction of functionality. The layer on the bottom offers data management functionality to the services layer [3]. And the services layer offers functionality to several clients on the Internet. Each layer is build onto the functionality of the next layer down the stack [2].



Figure 46 Proposed software architectures

# 3.6 SUBSYSTEM DECOMPOSITION

Subsystem decompositions will help reduce the complexity of the system. The subsystems can be considered as packages holding related classes/objects. These subsystems are further decomposed into other subsystems. The major subsystems identified are “Registration”, “Login”, “Screening”, “Donate Blood”, “Blood Distribution”, “Blood Collection” and “manage stock” subsystems. Users are classified in to roles. The “Login” subsystem authenticates a user to grant access based on the role of the user [1].

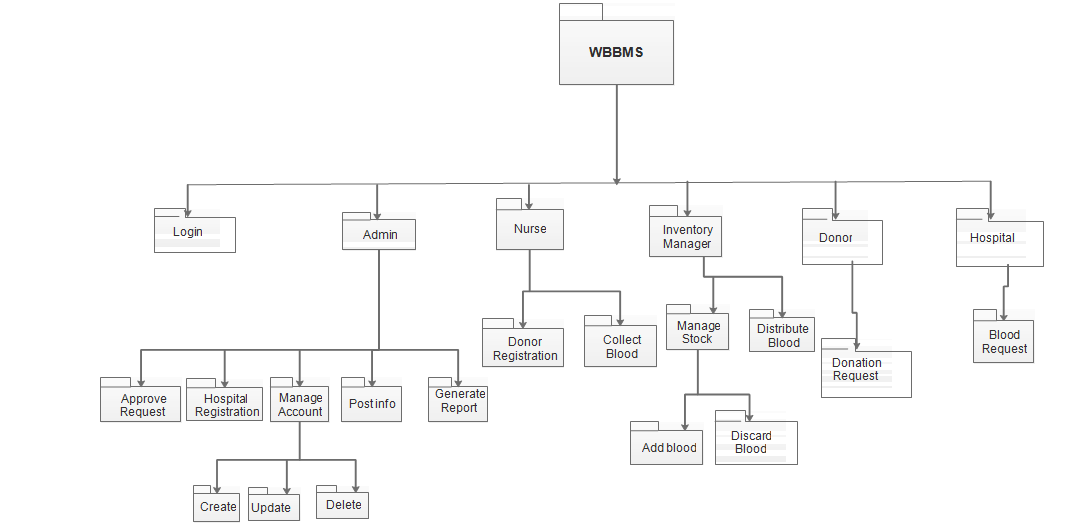


Figure 47 Subsystem decomposition

# 

# 3.7 COMPONENT DIAGRAM

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities. So from that point component diagrams are used to visualize the physical components in a system. Component diagrams can also be described as a static implementation view of a system. Static implementation represents the organization of the components at a particular moment. A single component diagram cannot represent the entire system but a collection of diagrams are used to represent the whole [2].

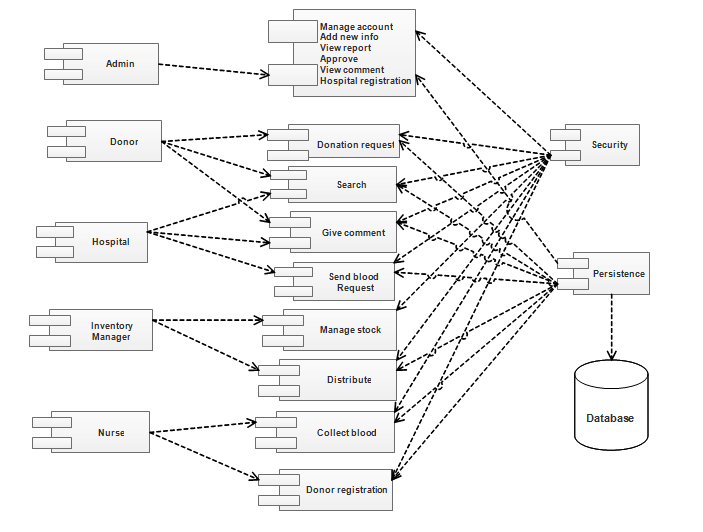


Figure 48 Component diagram

# 3.8 DEPLOYMENT DIAGRAM

The name Deployment itself describes the purpose of the diagram. Deployment diagrams are used for describing the hardware components where software components are deployed [2].

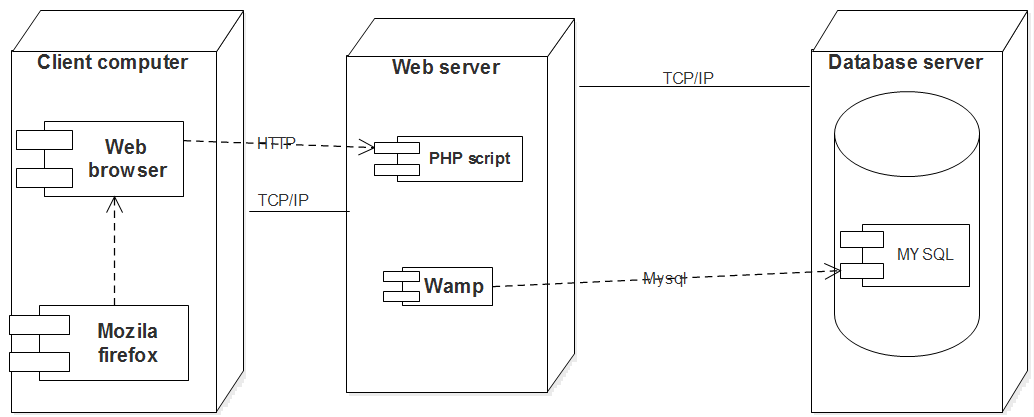


Figure 49 Deployment diagram

# 3.9 PERSISTENCE DATA MANAGEMENT

Persistence modeling is used to communicate the design of the database, usually the data base to both the users and the developers. It is also used to describe the persistence data aspect of the system [1]. The following tables indicate the persistence data management of the system.

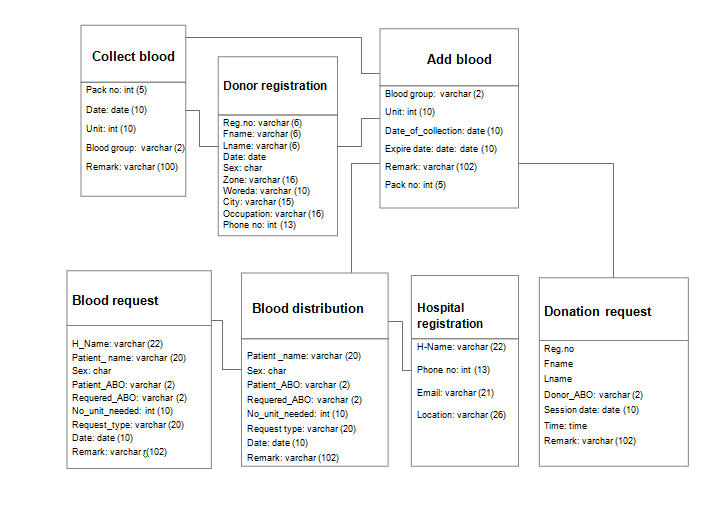


Figure 50 Persistent data management

# 3.9 ACCESS CONTROL

Access control and security describes the user model of the system in terms of access matrix. Upon start up, the system will display the user a login screen. Then the user will enter username and password. After the user entered the username and password, the system verifies whether the username and password entered are valid or not. If it is valid, the system will allow access to the application based on the privilege to which the user belongs. Accordingly to the following access control list is given for the system [2].

Table 30 Access control

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Actor | Request | Post info | Registration | Distribution | Collect blood | Stock management | comment |
| Donor | Send request Online for giving blood |  |  |  |  |  | View comment |
| Admin |  | adding new information | Register Hospital for being member | Control Distribution of Blood |  |  | View comment |
| Nurse |  |  | Register donor when donate blood |  | Collect blood from donors |  | View comment |
| Inventory manager |  |  |  | Distribute blood as ordered by Admin |  | Manage the Stock | View comment |
| Hospital | Send request online for receive blood |  |  |  |  |  | Give comment |

# 3.10 DETAILED CLASS DIAGRAM

The Class modeling diagram describes the attributes and operations of a class and also the constraints imposed on the system [1].

Nurse

Fname

: String

Lname

: String

Age

: Int

Sex: string

Phone \_ no

: Int

Register donor ()

Pack no ()

Inventory manager

Fname: string

Lname: string

Age: int

Sex: string

Phone \_ no: int

Email: string

Manage stock ()

Discard blood ()

Accept blood ()

Distribute ()

Blood

Blood type

Pack no

: Int

Info

: String

Add ()

Subtract ()

Admin

Fname: string

Lname: string

Age: int

Sex: string

Phone \_ no: int

Email: string

Approve ()

Manage Account ()

Add info ()

Blood bank

Name: string

Creation date: date

Email: string

Phone \_ no: int

Location: string

Accept ()

Store blood ()

Donor

Fname

: String

Lname

: String

Age

: Int

Date of birth

: Date

Occupation

: String

Sex: string

Reg.no

: String

Zone

: String

Woreda

: String

Kebele

: String

Phone\_ no

: Int

City

: String

Email

:string

Donation request ()

Hospital

H name: string

Patient name

Phone \_ no: int

Reason: string

Requirement date

Request blood ()

Blood stock

Stock \_ id: int

Date: date

Stock info ()

Update stock ()

1

0..\*

1...\*

1

1

1

.

.

\*

1..\*

1..\*

1

1..\*

Create

Create

Register

Register

Register

Part of

Takes

Manage

Donate

Collect

Request

1

1

1..\*

1..\*

1

1..\*

User Account

Username: string

Password: string

Set username ()

Set password ()

Get username ()

Get password ()

1

1

1

Create

1

1

Create

1

Figure 51 Detailed class diagrams

# 3.11 PACKAGE DIAGRAM

**WBBMS**

**Login**

**Subsystem**



All users

Login

System

Homepage



Nurse



Blood

Collecting



Donor

**Blood**

**Collection**



Donor

Donation

Date

Approve

**Blood**

**Donation**

Inventory

Manager



Order

Distributer

Distribute



Inventory

Manager



Stock



Stored

Blood



Nurse



Registration



Donor

**Distribution**

**Subsystem**

**Stock**

**Subsystem**

**Donor**

**Registration**

Figure 52 Package diagram

# CHAPTER FOUR

# IMPLEMENTATION AND TESTING

# 4.1 INTRODUCTION

In this phase what the group members have done is turning the physical design specification into working computer code, and then the code is tested until most of the errors have been detected and corrected. User sites are prepared for new system and user must come totally on the new system rather than the existing one to get there work done. There are some managerial activities in this, coding, testing, and installation.

# 4.2 CODING

The physical design specification created by the designers is turned in to working computer code.

**Sample code for login:**

<html>

<head>

<title>login</title>

<link rel="stylesheet" type="text/css" href="CSS/cs2.css" />

</head>

<body>

<form method="post" action="LoginServer.php">

<fieldset class="loginField" color="#E1E8ED">

<legend class="legend1">LogIn</legend>

<table cellspaccolspan="2"><?php if(isset($\_GET['fail'])) echo "<font color=\"red\"><center>login error</center></font>";?></td></tr>

<tr><td>Username</td><td><input type="text" name="username"></td></tr>

<tr><td>Password</td><td><input type="password" name="pass"></td></tr>

<tr><td <tr><td colspan="2"><?php if(isset($\_GET['select'])) echo "<font color=\"green\">please select privlage</font>";?> </td> </tr>

</table>

<table align="center" cellspacing="20">

<tr><td><input type="Submit" class="submit" value="Login"/></td>

<td><input type="reset" class="submit" value="clear"/></td></tr>

</table>

<h5 align ="center"> <a href="forgotpassword.php">Forgot password?</a></h5>

</fieldset>

</form>

</body>

</html>

<?php

session\_start();

include\_once('Db\_Cconnection.php');

$user\_name=$\_POST["username"];

$pass=$\_POST["pass"];

mysql\_select\_db("brc", $con);

$sql = "SELECT \* from account where Username ='$user\_name' and Password ='$pass'";

$check = mysql\_query($sql,$con);

if(mysql\_num\_rows($check)!=1){

header("Location: home.php?fail");

//session\_destroy();

}

else {

while($row = mysql\_fetch\_array($check))

{ $previlage=$row['Role']; $\_SESSION['user\_id']=$pass; $\_SESSION['user\_name']=$user\_name; //echo $\_SESSION['user\_name']; //$\_SESSION['E\_fname']=$user\_name; if($previlage=='Hospital'){ header("Location: Hospitalhome.php"); break; } else if($previlage=='Administrator'){ header("Location: adminp.php"); break; } else if($previlage=='Inventory'){ header("Location: Inventoryhome.php"); break; } else if($previlage=='Nurse'){ header("Location:NurseHomePage.php"); break; }} } mysql\_close();?>

# 4.3 FINAL TESTING OF THE SYSTEM

## ****4.3.1 Black box testing****

To test our system, the tester may use black box testing, if he/she has not enough time to check internal modules or codes. By looking only input /output or user interface, the tester can test our systems functionalities without looking the internal code. We used this testing technique for the following reasons:-

* This testing type is more effective on larger units of code
* Tester needs no knowledge of implementation, including specific programming languages
* Tester and programmer are independent of each other
* Testers are done from a user's point of view

## ****4.3.2 White box testing****

In this type of testing, skilled man in different programming languages tries to test the logic of our system. If the person who tests the system is not skilled, it is difficult to understand our systems functionality. If any failures occur while testing the system in all of the above testing methods, the team will take immediate correction where this fault occurred before jumping to next work. So, that it will meet the goal.

Table 31 Test case to register

|  |  |  |
| --- | --- | --- |
| Test Case 1: Register donor | | |
| Test case objective : To register donor | | |
| Test case description**:** Nurse enters basic info of the donors, and then presses register button. Client program contacts with server, server contacts with the database, and database checks for registration and sends message to nurse. | | |
| Requirements Verified**:** Yes | | |
| Test Environment**:** Apache Mysql server must be in running state, Database Should contain appropriate table and link must be established between server and client program. | | |
| Test Setup/Pre-Conditions**:** Apache server should be in running state and all fields should be filled. | | |
| Actions | | Expected Results |
| The nurse presses the register menu. | Displays registration page. | |
| Nurse should enter donors information | Displays success message | |
| If some fields are not filled the system display to fill the fields again. | | |

Table 32 Test case to login

|  |  |  |
| --- | --- | --- |
| Test Case 2: Login | | |
| Test case objective : To access the system | | |
| Test case description**:** System user enters Username and Password, then presses login button. Client program contacts with server, server contacts with the database, and database checks for authentication and displays System user page. | | |
| Requirements Verified**:** Yes | | |
| Test Environment**:** Apache Mysql server must be in running state, Database Should contain appropriate table and link must be established between server and client program. | | |
| Test Setup/Pre-Conditions**:** Apache server should be in running state and username and Password fields should be filled correctly. | | |
| Actions | | Expected Results |
| The System user should enter the correct user name and password to login. | Displays System user page. | |
| If user name and password are not filled correctly the system display to fill the user name and password again. | | |

# 4.4 USER MANUAL PREPARATION

No more manual preparation is needed for users, because the system developed is not software and it is not installed on a client computer. After the implementation has been completed, it is directly hosted on Network (server).

# 4.5 TRAINING

During the deployment of the system, the project group members will give short time training for the system administrators and clerks explaining how the system works and in what way they can manage the system developed.

# 4.6 INSTALLATION

Since the project is a web based System, there is no need to install it on a particular machine rather it will be hosted on a server.

# 4.7 STARTUP STRATEGY

Once the system has been published, the user can start and access his/her authorized page by entering the correct Username and Password with proper authentication and authorization processes.

# CHAPTER FIVE

# CONCLUSION AND RECOMMENDATION

# 5.1 CONCLUSION

This project has given us vast knowledge on the different computing technologies. We have learned a lot during the documentation and implementation of this system project. We have learned the way manual blood bank management system works and its different stages of manual blood bank management process.

We were also able to learn a lot of system analysis and design of the project, and all about object oriented concept with database. We came to know the different issues that come in the way of the development of the web based blood bank management system. Security was the main issue in the development of this project and we conclude that if these issues are taken into consideration, web based blood bank management system will become and real life system from just more a project.

# 5.2 RECOMMENDATION

Based on the above conclusions, we recommended that:

1. We develop this system for north Gondar only but we recommended to the next generations to develop for the national level.
2. This system uses user name and password as main security mechanism because of that almost all north Gondar people’s did not use digital signature identification so that we recommended to the next that it must have identification of digital signature of all allowable persons.
3. The government should encourage the habit of using web based systems and should provide all important infrastructure

# 5.3 FUTURE IMPROVEMENT

The system that we develop now is very limited in scope and there are some needs to add to make it very interactive and well functional system. So for the future we try to do by including the following functionality:

* 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.
* The system has to support Amharic language as result those people who cannot English language can use the system.

# APPENDICES

## Appendix I

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Student name | Student id. | Present | Absent | Date | Title discussed | Decision made |
| Getachew Asmamaw | 1175/05 |  |  | Oct 20/2015 | Requirement gathering | It has to be completed Oct 30/2015 |
| Tsehay Dawit | 1319/05 |  |  |  |  |  |
| Elsabet Sleshi | 1152/05 |  |  |  |  |  |
| Abebe  Sisay | 1078/05 |  |  |  |  |  |
| Getachew Asmamaw | 1175/05 |  |  | Oct31/2015 | About proposal | It has to be completed Nov 25/2015 |
| Tsehay Dawit | 1319/05 |  |  |  |  |  |
| Elsabet Sleshi | 1152/05 |  |  |  |  |  |
| Abebe  Sisay | 1078/05 |  |  |  |  |  |
| Getachew Asmamaw | 1175/05 |  |  | Nov 26/2015 | About requirement analysis | It has to be completed Dec 16/2015 |
| Tsehay Dawit | 1319/05 |  |  |  |  |  |
| Elsabet Sleshi | 1152/05 |  |  |  |  |  |
| Abebe  Sisay | 1078/05 |  |  |  |  |  |
| Getachew Asmamaw | 1175/05 |  |  | Dec 21/2015 | Model analysis | It has to be completed Jan 20/2016 |
| Tsehay Dawit | 1319/05 |  |  |  |  |  |
| Elsabet Sleshi | 1152/05 |  |  |  |  |  |
| Abebe Sisay | 1078/05 |  |  |  |  |  |
| Getachew Asmamaw | 1175/05 |  |  | Jan 4/2016 | System design | It has to be completed Jan20/2016 |
| Tsehay Dawit | 1319/05 |  |  |  |  |  |
| Elsabet Sleshi | 1152/05 |  |  |  |  |  |
| Abebe  Sisay | 1078/05 |  |  |  |  |  |

## Appendix II

Sample question during requirement gathering:

* Where does it performed?
* Who can donate blood?
* How often blood can be donated?
* What is the purpose of blood bank?
* What steps are performed to collect blood?
* Who is beneficiary of this blood?
* How long blood is being safe in stock?
* How patients can get blood from stock?

# REFERENCES

[1]Robert v, stumper .layette C. Teague**,** (1974/January 14), objects oriented system analysis and design with UML, (2nd edition),-[online available], [October22, 2015].

[2]Bernd bruegge, allenH.Dutoit, (1984/January 8), object oriented software engineering conquering complex and changing systems, (3rd edition),-[online available], [October15, 2015].

[3]http://casecomplete.com/use-case-diagrams,-[onlineavailable],[December 05, 2015].

[4] http://index.about.com/index Example of Sequence Diagram,-[online available], [December 15, 2015].

[5] http://index.about.com/index Examples Activity Diagram, -[online available], [December 05, 2015].

[6]Rada Musta, (1990/July 15), Textbook of Blood Banking and Transfusion Medicine, (1stedition),-[online available], [February 06, 2015].

[7] <http://nevonprojects.com/online-blood> bank-project/the software system is an online blood bank management,-[online available], [February 26, 2016].

.