

**DEBRETABOR UNIVERSITY**

**FACULITY OF TECHNOLOGY**

**DEPARTMENT OF COMPUTER SCINCE**

**PROJECT DOCUMENTATION**

**ON**

**”** WEB BASED PHARMACY MANAGEMENT SYSTEM FOR RED CROSS PHARMACY DEBRE TABOR BRANCH”

A DOCUMENTATION SUBMITTED TO THE FACULTY OF TECHNOLOGY DEPARTMENT OF COMPUTER SCIENCE IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF BSC IN COMPUTER SCINCE

June, 2016

**BY**

**NAME ID**

1. Amsalu Workie Comp/1306/05

2. Azeze Derso Comp/1311/05

3. Admasu Nigusie Comp/1303/05

4. Amare Tilaye Comp/1305/05

5. Helen Birhane Comp/1361/05

Advisors Name

Main \_advisor M.r Mersha N.

Co\_advisor M.r Dharnasi Prasad.

# 

**Approved by Board of Examiners**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Chairman Signature**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Main Advisor: Signature**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Co Advisor: Signature**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Examiner: Signature**

-----------------------------------------------------------------------------------------

# Acknowledgement

First of all we would like to thank our creator, God for his uncountable support for the completion of this project. Secondly we are thankful to our Main \_advisor M.r Mersha N. and Co\_advisor Mr. Dharnasi Prasad. for them encouragement, guidance and support from the initial to the final level enabled us to develop an understanding of the project and complete it on time. We also like to thank red cross Pharmacy workers for their help and cooperation during our interview. Lastly, we offer our regards and blessings to all of those who supported us in any respect during the completion of the project.

Tables of Contents Pages

[CHAPTER ONE…………………………………………………………………………………………………………………………………….1](#_Toc452629869)

[1.1 Introduction… 1](#_Toc452629870)

[1.2. Background of organization 2](#_Toc452629871)

[1.3. Statement of the problem 3](#_Toc452629874)

[1.4. Objectives of the projects 3](#_Toc452629875)

[1.4.1. General Objective 3](#_Toc452629876)

[1.4.2. Specific Objectives 3](#_Toc452629877)

[1.5. Scope of the project 4](#_Toc452629879)

[1.6. Limitation of the project 4](#_Toc452629880)

[1.7. Significance of the project 4](#_Toc452629881)

[1.8. Beneficiaries of the project 5](#_Toc452629882)

[1.9. Methodology of the Project 5](#_Toc452629883)

[1.9.1 Traditional methodology 5](#_Toc452629884)

[1.9.2 Modern methodology 6](#_Toc452629885)

[1.9.3 Techniques 6](#_Toc452629886)

[1.9.4 Tools 7](#_Toc452629887)

[1.10. Feasibility study 7](#_Toc452629889)

[1.10.1. Operational feasibility 8](#_Toc452629895)

[1.10.2. Technical feasibility 8](#_Toc452629897)

[1.10.3 Economic feasibility 8](#_Toc452629898)

[1.11. Work break down and deliverables 9](#_Toc452629899)

[1.12. Budget breakdown 9](#_Toc452629900)

[CHAPTER TWO 10](#_Toc452629902)

[SYSTEM REQUIREMENT SPECIFICATION AND ANALYSIS 10](#_Toc452629903)

[2.1 Introduction/ overview 10](#_Toc452629904)

[Part one: - System requirement specification………………………………………………………………………………………..10](#_Toc452629905)

[2.2 Description of the existing System 10](#_Toc452629906)

[2.3 Supplementary Specification 11](#_Toc452629907)

[2.4. Business Rule of the organization 11](#_Toc452629908)

[2.5Requirement elicitation model 12](#_Toc452629909)

[2.5.1 Essential use case modeling 12](#_Toc452629910)

[2.5.2 Essential user interface (UI) prototype. 13](#_Toc452629911)

[Part two: System analysis 13](#_Toc452629912)

[2.6 Description of the proposed system 13](#_Toc452629913)

[2.7Functional andNon-functional Requirement 13](#_Toc452629914)

[2.7.1 Functional requirement 15](#_Toc452629915)

[2.7.2 Non Functional requirement 15](#_Toc452629916)

[Performance Characteristics 15](#_Toc452629917)

[Error Handling 15](#_Toc452629918)

[Physical Environment 15](#_Toc452629919)

[2.8 The proposed system models 16](#_Toc452629920)

[2.8.1 System use case model 16](#_Toc452629921)

[2.8.2 Sequence diagrams 26](#_Toc452629922)

[2.8.3 Conceptual class model 32](#_Toc452629923)

[2.8.4 User interface (UI) flow diagram 33](#_Toc452629924)

[2.8.5 Activity diagram 34](#_Toc452629925)

[2.8.6 User interface prototype 40](#_Toc452629929)

[CHAPTER THREE: DESIGN SPECIFICATION 43](#_Toc452629932)

[3.1 Introduction /overview 43](#_Toc452629933)

[3.2 System architecture 43](#_Toc452629934)

[3.2.2 Describing an Architecture using UML 45](#_Toc452629935)

[3.2.3 Detailed Design of the System 50](#_Toc452629936)

[CHAPTER FOUR 57](#_Toc452629937)

[Implementation 57](#_Toc452629938)

[4.1 Overview of programming language used 57](#_Toc452629939)

[4.2 Algorithms used 57](#_Toc452629940)

[4.3 sample code… 58](#_Toc452629941)

[CHAPTER FIVE 60](#_Toc452629942)

[5. TESTING 60](#_Toc452629943)

[5.1 Overview of Testing 60](#_Toc452629944)

[5.2 Sample Test…. 61](#_Toc452629945)

[5.3Unit Testing….. 63](#_Toc452629946)

[5.4 Acceptance Testing 63](#_Toc452629947)

[CHAPTER SIX 64](#_Toc452629948)

[CONCLUSION AND RECOMMENDATIONS 64](#_Toc452629949)

[6.1 Conclusion….. 64](#_Toc452629950)

[6.2 Recommendation and Future Enhancement 64](#_Toc452629951)

[**Future Enhancement** 64](#_Toc452629952)

[References 65](#_Toc452629953)

[Appendix 66](#_Toc452629957)

[Glossary of Terms 70](#_Toc452629958)

**List of table’s page**

Table 1.1.Development tools …………….……………………………………………….………7

Table1.2.work brake down and deliverables ...................................................................…...........9

Table 1.3.budget break down……………………………………………………………………...9

Table 2.1.Use case description for login………………………………………………………....19

Table 2.2. Use case description for employee registration………………………………………20

Table 2.3. Use case description for delete employee………………………………………..…...21

Table 2.4. Use case description for drug registration…………………………………………....22

Table 2.5.Use case description for check expired date…………………………………………..23

Table 2.6. Use case description for register customer…………………………………………...24

Table 2.7.Use case description for print use case…………………………………….………….25

**List of figures page**

Figure 2.1.essential use case diagram………………………………………………..……..……12

Figure 2.2.Use case diagrams………………………………………….………………….……..18

Figure 2.3.Sequence diagram for login…………………………………………………………. 26

Figure 2.4.Sequence diagram for delete employee…………………………………………..…..27

Figure 2.5.Sequence diagram for register drug………………………………………………......28

Figure 2.6.Sequence diagram for sale drug……………………………………………..….……29

Figure 2.7.Sequence diagram for check expired date………………………………………….30

Figure 2.8.Analysis level class diagram………………………………………………………..31

Figure 2.9.Diagram for user interface flow diagram……………………………………….......32

Figure 2.10.Activity diagram for login…………………………………………………….…….33

Figure 2.11.Activity diagram for register drug…………………………………………….…….34

Figure 2.12.Activity diagram for register employee……………………………………………..35

Figure 2.13.Activity diagram for delete employee………………………………………………36

Figure 2.14.Activity diagram for check expired date……………………………………………37

Figure 2.15.Activitty diagram for sale drug…………………………………………………......38

Figure 2.16.User interface prototype for login…………………………………………………..39

Figure 2.17.User interface prototype for employee register…………………………………….40

Figure 2.18.User interface prototype for drug register………………………………………….41

Figure 3.1.System architecture model for the new system…………………………………..….43

Figure 3.2.Class type architecture…………………………………………………………..…....44

Figure 3.3.Component diagram……………………………………..,,…………………….……46

Figure 3.4.Deployment diagram…………………………………………….…………………...47

Figure 3.5.Persistent diagram…………………………………………...……………………….48

Figure 3.6.State chart diagram for login…………………………………………………..……..49

Figure 3.7.State chart diagram for register drug…………………………………………………50

Figure 3.8.State chart diagram for delete employee…………………………………,………….51

Figure 3.9.State chart diagram for delete drug…………………………………………….….....53

Figure 3.10.user interface design for login…….………………………………………….….....53

Figure 3.11. User interface design for employee registration.…………………………….….....53

Figure 3.12. User interface design for change password.…………………………….…...........54

Figure 3.13. User interface design for create account.…………………………….……….....54

Figure 5.1.Sample test for login page.…………………………….………………………….....59

Figure 5.2.Sample test for create account…………………....…………………………….….....60

Figure 5.3.Unit test for expired test of drug.…………………………….…................................61

**ABSTRACT**

The main objective of this project is to develop a pharmacy management system that enable to register, check item and check expire date of drug. The implementation of this project is by using PHP, html and some java script code. This project is insight into the design and implementation of a Pharmacy Management System. The primary aim of is to improve accuracy and enhance safety and efficiency in the pharmaceutical store. Today management is one of the most essential features of all form. Management provides sophistication to perform any kind of task in a particular form. This is pharmacy management system; it is used to manage most pharmacy related activities in the pharmacy.

# CHAPTER ONE

## 1.1 Introduction

Technology is spreading its wing in almost every walks of human life activities. Now a day it is better if every activity is done using new technology in order to fulfill the need of human being, Organization and Enterprise. As today’s world there are many organizations and each organizations needs to be preferable, computable and work on fastest way in order to satisfy users interest. I.e. they should have facilitated their activities in computerized way. Hence Red Cross Pharmacy management system is a system that is designed to improve accuracy and to enhance the performance of the task in the pharmacy. It is a computer based system which helps to the employee inside the pharmacy to facilitate the activity of the pharmacy in a simple way. In the pharmacy there are two places in which the drug are available. Those are stock and store. The stock is the place in which the drug that needs to be sold is stored. And the store is the place in which the new bought drug is stored.

At present manual system is being utilized in the pharmacy. It requires the pharmacist to manually monitor each drug that is available in the pharmacy. This usually leads to mistakes as the workload of the pharmacist increases. Red Cross pharmacy uses manual system to manage the pharmacy. But managing the pharmacy manually is bulky and tedious due to:- wastage of time, work overload, loss of data, data redundancy and the like problems. to avoid this problem Debre tabor pharmacy tries to change the manual system in to computerized system to overcome or reduce the problem of the manual system by developing a system that register customers, register drugs, view the available drugs, remove the unwanted drugs, differentiate expired drugs from normal/new drugs ,update drugs and search drugs .

## 

## 

## 

## 1.2. Background of organization

## Red Cross pharmacy was established for minimizing the shortage of drugs in the country and it expands their branch in different regions. Among those Debretabor Red Cross pharmacy is the one branch that found in Amhara region at south Gonder this institution was established in 1993 E.C but it doesn’t continue its work until 2003 E.C because of awareness of the society due to the usage of drugs. Then, in 2004E.C it starts its service in case of wideness of towns and establishment of university in Debretabor, know at the time it runs its service by using manual system. The pharmacy buys drugs from Ethiopia Drug Organization.

## Red Cross pharmacy uses the manual system in order to manage the stock item. These are shelf drug control, First in First Out, and pharmacological order of drug. The new system will generally change this manual system in to computerized system. Careful management of pharmaceuticals is directly related to a country’s ability to address public health concerns. There are shortages of pharmacist in Debretabor town to solve people’s problems in health care.Red Cross pharmacy has the following functions to the societies and community’s Such as:-supplement of drug to the society, Increase the access of drug in the town and customize the need of customers. Pharmaceutical management represents the whole set of activities aimed at ensuring the timely availability and appropriate use of safe, effective, quality medicines and related products and services in any health care .

## 

## 1.3. Statement of the problem

Managing a very large pharmacy with records on papers will be tedious and difficult to keep track of inventory with regards to the drugs in the stock inside the pharmacy. since it is manual system quantity of drugs available based on the categories and their functions can’t be easily known, difficulty of getting full information about drugs when needed immediately, difficult to identify which drugs are expired, the most sensitive data may lost, not efficient enough to the customers, Preparing report for each drug takes long time and also most of the time redundant data may occur [2].

## 1.4. Objectives of the projects

### 1.4.1. General Objective

The general objective of this project is to design and develop web Based Pharmacy Management System for Red Cross pharmacy in Debre tabor branch.

### 1.4.2. Specific Objectives

## To analysis the existing system.

* To develop computerized recordable system.
* To delete or remove expired drugs.
* To generate report with in short period of time.
* New store item registration.
* To identify expired date of drug.

## 

## 1.5. Scope of the project

The scope of the project is listed in the following:-

* Check availability items in the data base.
* Checks expired date of the drug.
* Register drug.
* Register customer.
* View drug.
* Delete drug.
* Validate input.
* Search drug.
* Generate report.
* Employee registration.
* Delete employee.
* Manage account.

## 1.6. Limitation of the project

* The system applies to only Red Cross pharmacy debre tabor branch.
* The system unable to pay salary to the employee who has the job in the pharmacy.
* Can’t order to take medicine, it only recommend for the entered medical information.
* The system organization does not have any interaction with other organization system.
* The system does not have any physical control mechanism.

## 1.7. Significance of the project

* Better stock management.
* It provides efficient, flexible and reliable items’ storing, locating and distributing.
* Provide better data stock in the systems.
* To minimize the workload of the employees.
* Enhance best controlling method for the drug.

## 

## 1.8. Beneficiaries of the project

Here the team member described the benefits that are expected to gain after the development of the system.

**To the team members:-**Having knowledge how real life problem should be solved.

**To the pharmacist: -** Decrease more time consumption, increasing job satisfaction by eliminating complex tasks and helping the pharmacists by decreasing the workload.

**To the customer:-**Will have more confidence or trust about the drug.

**To the manager:-**To control the activity which is done in the pharmacy in simple manner.

## 1.9. Methodology of the Project

In our project the team member will use Object Oriented Software Development Methodology.

To gather an accurate data from customers and the concerned body our team will use the

Following traditional and modern fact finding techniques:-

### 1.9.1 Traditional methodology

**Interview**: - This is one of data collection method that enables to gather information from the organization directly in the form of asking question and getting answers for those questions.

**Document analysis**: - To get historical information of the organization activities and to know the

Organization rules and regulations the team will analyze documents which are relevant to the new

System.

**Observation**:-To get first hand accurate information about how the existing system works the

Team will observe the current system directly. We are observing the situation encountered in the

Organization .The pharmacist gives the drug for customers by seeing the receipt paper come from

The other service.

**Questioner:-**To get full information about the pharmacy the team member want to prepare the

Questions by the document form, and then the manager gives detail information about the

Pharmacy, what actions done there by the written form.

### 1.9.2 Modern methodology

The team member will use prototype development modeling, because of the following reason:-

* Users are actively involved in the development
* Since in this methodology a working model of the system is provided, the users get a better understanding of the system being developed.
* Errors can be detected much earlier.
* Quicker user feedback is available leading to better solutions.
* Missing functionality can be identified easily

System design specifies how the system will accomplish. System design is the structural implementation of the system analysis.

Project planning

Requirement gathering

Implementation and system testing

Project development and interface design

### 1.9.3 Techniques

* UML (Unified Modeling Language).
* CRC (Class Responsibility Collaboration).

### 1.9.4 Tools

During system development the team member will use the following system development tools.

|  |  |
| --- | --- |
| Development tool | Tool name |
| **Software tool** | HTML and PHP:-for designing web interface. |
| Wamp server 2.0:-for database. |
| E draw 7.9:-For designing UML diagrams. |
| Microsoft power point 2007:-for presenting the document. |
| Microsoft office word 2007:- for documenting. |
| **Hardware tool** | 8 GB flash. |
| 700 MB CD |
| 4096MB RAM size, Intel(R) core (TM) i3-2328M CPU @ 2.20GHz (4cpus) processor , Intel(R) HD 3000 Personal computer |

Table 1.1 Development tool

## 1.10. Feasibility study

## Feasibility study is essential to evaluate the cost and benefits of the new system. On the basis of the feasibility study decision is taken on whether to proceed or to cancel the project. Need of the feasibility study:-

# It used to determine/finds out the problem of the existing system.

# To determine all goals of the new system.

# It finds all possible solutions of the problems of the existing system.

# Determines the potential of the existing system.

### 

### 1.10.1. Operational feasibility

# The new system will increase and improve the activity of customers by replying the reliable data, response time, and increase efficiency of work in the pharmacy. The project is welcomed with great pleasure by the organization. The employees are more cooperated to give needed information. This indicates that the project is operationally feasible.

### 1.10.2. Technical feasibility

The proposed system doesn’t require much technical expertise. The system to be developed by using technologically system development techniques such as PHP, Java script, CSS and Wamp server 2.0 Database without any problems.

### 1.10.3 Economic feasibility

The purpose of economic feasibility is to identify the financial benefits and costs associated with the development project. The development cost is less when it is compared to the benefit that the project will bring to the organization. Hence the project is economically feasible.

**Cost benefit analysis:** The system which we are going to develop will have economic benefit. Those economic benefits may be tangible or intangible

**Tangible benefit:** This means the concrete benefit that can be expressed in terms of dollars or birr. So the system proposed to develop will decrease a lot of birr that was expensive to buy the hard copy document material such as paper, pencil, rubber, and so on. But after the system developed the data can be managed with one person and in one computer in very short time.

**Intangible benefit:** Those benefits that cannot be expressed in terms of birr or dollar Intangible benefit that the system will give is the following:

* Give more readable, reliable, easily manageable, and database which contains all employees track.
* Time requires for adapting new system
* The proper and ordered files of employees which has stability means which is not easily lost.
* Saving loss of documents

## 1.11. Work break down and deliverables

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Task | Starting date | Finishing date | Expected output |
| 1 | Title selection | October 1 | October 7 | Title submission |
| 2 | Gather information | October 8 | October 15 | Start proposal |
| 3 | Requirement planning | October 16 | November 13 | Requirement analysis document |
| 4 | System analysis | November 14 | December 21 | System analysis document |
| 5 | System design | December 22 | January 3 | System design document |
| 6 | Final documentation | | January 4 | Final documentation submission |
| 7 | System implementation | February 1 | June 10 | Complete implementation |
| 8 | Testing | June 11 | June 20 | Test |
| 9 | Final project | | June 21 | Final project submission |

Table 1.2 work break down and deliverables

## 1.12. Budget breakdown

## To complete our project starting down from the beginning up to the end of this project we planned the following cost list.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Resource | Quantity | Unit price/quantity | Amount of price |
| 1 | Pen | 12 | 5 | 60 |
| 2 | paper | 1 | 100 | 100 |
| 3 | Color print | 5 | 10 | 50 |
| Normal print | 80 | 2 | 160 |
| Copy | 450 | 50 | 225 |
| 4 | CD | 5 | 15 | 75 |
| 5 | Transport | 10 | 15 | 150 |
| 6 | Flash | 2 | 150 | 300 |
| 7 | Others | - | - | 2000 |
|  | Total | - | - | 3,020 |

Table 1.3Budget breakdown

# 

# CHAPTER TWO

# SYSTEM REQUIREMENT SPECIFICATION AND ANALYSIS

## 2.1 Introduction/ overview

Under these chapter describes the overview of the existing system, overview of the proposed system functional and non-functional requirement of the system and modeled using unified modeling language (UML) such as use case documentation for each use case identified in the use case diagram, sequence diagram, activity diagram, analysis level class diagram or conceptual modeling, user interface prototyping and supplementary specifications.

The proposed system documentation involves the requirement elicitation, where the team members gather requirements of the system from user, analysis of the requirement that focuses on the functional, object and dynamic modeling (state diagram), system design and object design. The project is intended to advocate for the need of Red Cross pharmacy to use facilitated computerized drug information system. As a result the team member believed that the user will have the expected satisfaction of the service provided by the pharmacy. Therefore, the team member recommends Red Cross pharmacy manual system to the computerized system to facilitate their service and benefit of the organization.

# Part one: - System requirement specification

## 2.2 Description of the existing System

The current system of Red Cross pharmacy information management is manual system. That means checking expired date and availability of drugs is done by checking every drug inside the Pharmacy. This leads loosing time and resource of the organization. An existing system compromises different players to carry out its job. Those are pharmacist, manager, store coordinator and cashier [5].

**Pharmacist: The** customer comes with the ordered prescription then the pharmacist looks that order of drug and gives the drug accordingly. The customer gets his/her requested service from the pharmacist.

**Manager: The** manager gets reports from the pharmacist, store man, and cashier. The reports help the manager to see how services are given to the client.

**Store coordinator :**Store man is responsible to register the drugs that buy from the private sectors or from the governmental association, and also control the drug that are goes out to the stock.

**Cashier:** The cashier receives the cost of the drug from the customer ordered by the pharmacist.

## 2.3 Supplementary Specification

The Supplementary Specifications capture the system requirements that are not readily captured in the use cases of the use-case model. Such requirements include:-

* Legal and regulatory requirements and application standards.
* Quality attributes of the system to be built, including usability, reliability, performance, and supportability requirements.
* Other requirements such as operating systems and environments, compatibility requirements, and design constraints and the other Supplementary specifications are the business rules .The business rules is a principle or a policy in which the proposed system operates accordingly. It deals with access control issue.

## 2.4. Business Rule of the organization

The existing system has its own mechanism in which its customers are treated. These include:

**BR1**: The pharmacist must treat customers in good manner and should address customer’s request.

**BR2**: The cashier should receive the price of medicine honestly from customers and he/she should generate report for manager.

**BR3**: Manager should control the entire activity in the stock and should receive clear and appropriate report from the workers of the pharmacy.

**BR4**: Sold drug should order in their identifiable type to facilitate searching requested drug.

**BR5**: Manager should control the overall information from any biases properly.

**BR6:** Forms should contain stock information appropriately.

**BR7**: Pharmacist doesn’t sell the expired drug.

## 2.5Requirement elicitation model

### 2.5.1 Essential use case modeling

Essential use cases are of primary importance early in a project’s analysis phase. Their purpose is to document the business process that the Application must support without bias to technology and implementation. The narrative in the Essential use case is to be expressed in the language of the application domain of users. Essential Use cases should achieve the following goals: Serve as an effective communication tool between users and analysts, is diagrammed using the standards documents, Be documented in text format using the standard "Essential Use case Specification Template”. 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 existing system.

Drug manager

Store man

Pharmacist

Casher

Register drug

Distribute drug

Check availability

Print receipt

Figure 2.1 Essential use case diagrams.

### 2.5.2 Essential user interface (UI) prototype.

Forms and Other Documents of the Existing Systems as mentioned before, forms are used for different purposes. Also they are the backbone of the system. The existing system uses deferent types of forms. Some of the forms are attached on appendix.

# Part two: System analysis

## 2.6 Description of the proposed system

By carefully analyzing and observing the problem of existing system we came up with a solution that the current manual system should be computerized. The computerized system will eliminate/reduce the problem on time, work load and complexity on storing drugs information. The system will include a database for recording drugs that facilitate fast information retrieval, modifying, inserting and deleting. It also includes an attractive user interface that facilitates accessing the database and recording drugs easily.

The system allows the user to enter expiry date for a particular product or drug during opening stock and sales transaction. It also involves arrival of new batches of drugs, getting information about the drugs e.g. expiry date, number of drug type left, and location of a drug in the pharmacy. Players represent external entities that interact with the system. Players manage and perform the Systems functionality and also players can access the proposed system at any time, improving the efficiency of the system by ensuring effective managing of services and activities, Generating report, reducing the employees’ workload in the organization.

## 

## 2.7 Functional and Non-functional Requirement

### 2.7.1 Functional requirement

Functional requirement is concerned with actual performance of the system that is going to be developed. Functional requirements describe the functionality or service provided by the new system. The functional requirement is the services that are provided by the system. It also describes the interactions between the system and the user. The new system is expected to provide the following functionality

**Input requirement**

* The system should register drug.
* The system should verify the requested information.
* The system should check Store items by item name.
* Each input item information must include item id, item name, code, quantity, manufactured company, and expiry date.

**Output requirement**

* The system generates a report.
* The system should store all the data related with all the tasks performed into a database.
* The system should display store item that are reach to expired date.
* The system should display employee information to the manager.

### 2.7.2 Non Functional requirement

Nonfunctional requirement describe visible aspects of the system that are not directly related to the system. Unlike functional requirement, non-functional requirement deals with additional quality of the system such as performance, error Handling, usability, availability and security matter.

Some of the non-functional Requirements are:

### Performance Characteristics

The system should provide response for the users with less time than the previous system. If a user follows the correct way of execution the system will stay safe if not the system will respond to that action. It is expected that the system will serve many clients at a time.

### 

### Error Handling

The system handles errors by giving error-message and warning to the user. The error handling can be seen in different aspects. The system should check the validity of a user during log in. Any user can view the different part of the system based on the role assignment that the system provides. The system should handle errors which are occurred due to invalid inputs of the user then displays error message and additional information’s on how to correct it

**Quality Issues**

**Usability**: - the user interface shall be simple for the user on how to use the system and easy to learn.

**Availability:** The product is available at all time with the availability of power. The system shall be available for twenty four hours.

**Reliability:** - The application will run without failure. The system will be reliable in all aspects since all activities and operations are performed by machines.

### Physical Environment

Since Debre Tabor Red Cross pharmacy is currently growing in technology issues from time to time. Its main data center may afford server machine to deploy the system effectively. The system is expected to withstand the following external factor.

* Less processer speed of personal computer that are caused by dust and other unnecessary things happened in client computers for accessing the system.
* Less power that causes the system fail or stop functionality.
* The server and the other devices in which system installed should kept in a secured and air conditioned rooms

**Security Issues**

The system is secured from malicious users from accessing the database because most of the information is stored in the server. The authentication is done through password protection in database manipulation. This means that before entering to the database the system will request user name and password. This will prevent unauthorized data modification on the database.

* The system follow a role based security which means the access level and privilege for each Users are set by the system administrator.
* The system has authentication mechanism (Username and password).

**Maintainability**

Through time there are always changes when the user’s needs another additional functionality, when the system administrator identifies the system need to be modified, while the organizations work style is changed and depending on different reasons. Some of the user interface and basic modification can be performed by the collection of developers but the system developers are the right persons to update the system.

## 2.8 The proposed system models

### 2.8.1 System use case model

Use Case represents interaction between a user (human or machine) and the system. Use case components:

**Actor**: is a person, or external entity that plays a role in one or more interaction with the System.

**Use case**: describes a sequence of actions that provides something of measurable value to an actor and is drawn as a horizontal ellipse.

**System boundary**: indicates the scope of the system project. Anything within the box represent functionalities in side with in scope.

**Actor identification** In the use cases an actor interacts with the system to perform a piece of meaningful work that helps them to achieve a goal and has access to define their overall role in the system and the scope of their action. Depending on the above explanation actors in this system are the following:

**Manager**: Controls the overall activity in the shop.

**Pharmacist**: Manages the drug information in the stock.

**Store coordinator**: Manages the outgoing and incoming drug from the stock.

**Cashier:** Collect the price of the sold items and generate report to the manager.

**Use case identification**

Each Use Case describes the functionality to be built in the proposed system, which can include another Use Case's functionality or extend another Use Case with its own behavior. The most important and basic use cases of this system are the following:-

* Login
* Register employee
* View employee
* Delete employee
* Register drug
* Register Customer
* View drug
* Delete drug
* Check expire date
* Check item
* Prepare prescription
* send prescription
* receive prescription

**Manager**

* Login
* Register employee
* Delete employee
* Manage account

**Pharmacist**

* login
* Register customer
* Check item
* Sale drug

**Store coordinator**

* Login
* Check expired date
* View drug
* Register drug
* Delete drug

**Cashier**

* login
* Print prescription
* Receive prescription

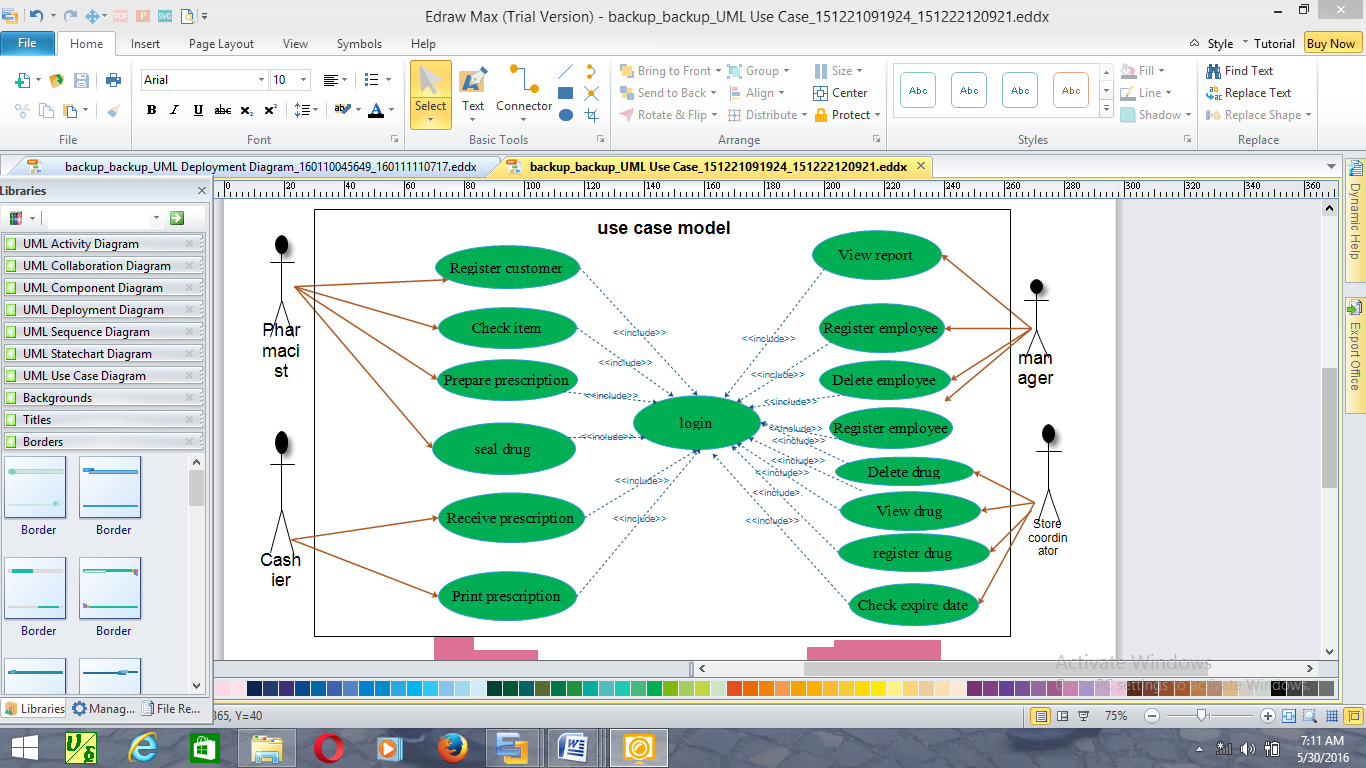


Figure 2.2use case diagram.

**Use Case Description**

Table 2.1 Use case description for login

|  |  |
| --- | --- |
| Name | Login |
| ID | UC1 |
| Actors | Manager, Pharmacist, Casher, Store coordinator |
| Description | In order to get into or access the system |
| Pre-condition | 1. The Manager, Store coordinator, pharmacist, or Cashier must open  the system |
| Flow of events | 1. Open the system.  2. Click on login link.  3. Login form displayed.  4. Select account type and enter user name and password.  5. Click on the login button.  6. System verifies in the account database.  7. Main form displayed.  8. End of use case. |
| Post condition | 1. Access the system |

Table 2.2 Use case description for employee registration

|  |  |
| --- | --- |
| Name | Employee Registration |
| ID | UC2 |
| Actor’s | Manager |
| Description | Register the information of the workers in the pharmacy |
| Pre-condition | 1. Initiate the system  2. Have user name and password |
| Flow of event | 1. The manager opens the system.  2. The manager log to his or her page  3. The manager click on the register employee link.  4. The system displays the register employee form.  5. The manager inserts the necessary information of the employee.  6. The manager click on the register button.  7. Then the system will generate successfully message  8. End of use case. |
| Post condition | 1. Access the system  2. Close the system |

Table 2.3 Use case description for delete employee

|  |  |
| --- | --- |
| Name | Delete Employee |
| ID | UC3 |
| Actor’s | Manager |
| Description | Delete the employee when it is necessary. |
| Pre-condition | 1. Initiate the system.  2. Have user name and password |
| Flow of event | 1. The manager log to his or her page.  2. The manager click on delete employee link.  3. The system displays the delete employee form.  4. The manager enters the IDNO of the employee.  5. Click on the delete button.  6. Then the system will generate successfully message  7. End of use case |
| Post condition | 1. Return to home page or  2. Close the system |

Table 2.4 Use case description for drug registration

|  |  |
| --- | --- |
| Name | Register Drug |
| ID | UC 4 |
| Actors | Store coordinator |
| Description | Registering the new drug from the store in to the data base. |
| Pre-condition | 1. Initiate the system.  2. Have user name and password. |
| Flow of event | 1. The Store coordinator opens the system.  2. The Store coordinator log to his or her page.  3. The Store coordinator click on Register drug link.  4. The system displays the register drug form.  5. The Store coordinator will enter the attributes of the drug.  6. Then click on submit.  7. Then the system will generate successfully message.  8. End of use case |
| Post condition | 1. Return to home page or  2. Close the system |

Table 2.5 Use case description for check expired date

|  |  |
| --- | --- |
| Name | Check Expired Date |
| ID | UC 5 |
| Actors | Store coordinator |
| Description | In order to check the drug expired date. |
| Pre-condition | 1. Initiate the system.  2. Have user name and password. |
| Flow of event | 1. Open the system.  2. The Store Coordinator log to his or her page.  3. The Store coordinator click on check expired date link.  4. Then the form will be displayed.  5. The Store coordinator enters the expired date of the drug.  6. Then Store coordinator clicks on search button.  7. The system displays the list of the dug that is inserted in its date.  8. The Store coordinator click on the clear button.  9. Then the system will response successfully message.  10. End of use case. |
| Post Condition | 1. Return to home page or  2. Close the system |

Table 2.6 Use case description for register customer

|  |  |
| --- | --- |
| Name | register customer |
| ID | UC 6 |
| Actors | Pharmacist |
| Description | To purchase the drug to the customer |
| Pre-condition | 1. The customer brings with his/her prescription |
| Flow of event | 1. The pharmacist opens the system.  2. The home page will be displayed.  3. The pharmacist inserts user name and password.  4. The system will verify the user name and password.  5. The pharmacist click on check list link.  6. Then the system displays the drug list.  7. Then return to sale drug link.  8. Click on sale drug link.  9. Enter the necessary information of the customer and the drug.  10. Then click on the load button.  11. Then the system will response successfully message.  12. End of use case. |
| Post condition | 1. Return to home page or  2. Close the system |

Table 2.7 Use case description for print use case

|  |  |
| --- | --- |
| Name | Print |
| ID | UC 7 |
| Actors | Cashier |
| Description | Printing soled drug for the customer |
| Pre-condition | 1. There must be list of drug that must be inserted by the pharmacist. |
| Flow of event | 1. Open the system.  2. The home page will be displayed.  3. The cashier inserts user name and password with their account type.  4. The system will verify the user name and password.  5. Then the system display his/her page.  6. The cashier click on fetch drug link.  7. The list of drugs with corresponding quantity and price.  8. The cashier calculates the total price.  9. Then click to print.  10. Then the system will display response.  11. End of use case. |
| Post condition | 1. Return to their appropriate page.  2. Close the system |

### 2.8.2 Sequence diagrams

The sequence diagram is used primarily to show the interactions between objects in the

Sequential order that those interactions occur.

To use our system it has the following sequence of events:-

1. All actors visit the home page and the home page display
2. Click on login link.
3. Then the login form is displayed.
4. User inserts username and password.
5. Click on login button.
6. Then the home page displayed
7. Then the manager creates account for all actors
8. Users can access the page that you want by selecting their position.

Figure

Home page

Login link

Login form

Validater

Database

Visit home page ()

t

r

a

y

a

g

a

i

n

(

)

Fig 3.2 Sequence Diagram for login

All actors

Basic course of

Action for login

1. All actor visit the home page

and home page display

Click on link ()

Send ()

Display login form ()

Enter user name

And password ()

Submit ()

v

a

l

i

d

a

t

e

Display the target

Page ()

Continue ()

Check ()

2. Click on the login link

3. Then the login form

Displayed

4. User inserts user name

And password

5. Click on login button

6. If the user are not insert

Valid user name and

Password system display

Error message

7. If the users are insert

Valid user name and

Password system display

A successful message

U

s

e

r

C

l

i

c

k

o

n

l

o

g

i

n

(

)

Home page

Delete

Employee

Delete

Employee

Form

Validater

Database

Visit home page ()

Invalid id

t

r

a

y

a

g

a

i

n

(

)

Manager

Basic course of

Action for delete

Employee

1. Manager visit the home

Page and home page display

Click on link ()

Send ()

Display delete

Employee form ()

Insert employee id ()

Send ()

v

a

l

i

d

a

t

e

Display

Successfully delete

Message

Send ()

Check ()

2. The manager click on the

Delete employee link

3. Then delete employee

Form displayed

4. Insert employee id

5. Click on delete

6. If employee delete

Correctly system display

Successful delete message

7. If it is not delete the

System display a

Message to re tray again

c

l

i

c

k

o

n

d

e

l

e

t

e

(

)

Figure 2.4 sequence diagram for delete employee

Home page

Register

Drug link

Register

Drug form

Validater

Database

Visit home page ()

r

e

e

n

t

e

r

**Store**

**Coordinator**

Basic course of

Action for register

Drug

1. Store coordinator visit the

Home page and home page

Display

Click on link ()

Send ()

Display drug

Register form ()

Fill the form ()

Submit ()

v

a

l

i

d

a

t

e

Display

Successfully

Register message

Send ()

Check ()

2. The store coordinator click on

the register drug link

3. Then register drug form

Displayed

4. Fill the form and submit

5. The system validate the

Input

6. If the input is correct then

the system display

Successful message

7. If the input is not correct

The system displays refill

The form

Figure 2.8 sequence diagram for register drug

Figure 2.5 sequence diagram for register drug

Figure 5

Figure 2.6 sequence diagram for sale drug

Home page

Sale drug

Link

Sale drug

Form

Validater

Database

Visit home page ()

f

i

l

l

c

o

r

r

e

c

t

l

y

t

r

a

y

a

g

a

i

n

(

)

Pharmacist

Basic course of

Action for sale drug

1. Pharmacist visit the home

Page and home page display

Click on link ()

Send ()

Display sale drug

Form ()

Insert customer

Prescription ()

Send ()

v

a

l

i

d

a

t

e

Display

Successfully

Message

Send ()

Check ()

2. The pharmacist click on the

Sale drug link

3. Then sale drug form

Displayed

4. Insert customer

Prescription

5. Click on sale button

6. If the field are not

Correctly fill system display

Successful error message

7. If the field are correctly

Fill system display a

Successful message

c

l

i

c

k

o

n

s

a

l

e

b

u

t

t

o

n

(

)

2.9 sequence diagram for sale drug

Home page

Check

Expired

Link

Expired list

Validater

Database

Visit home page ()

Store

Coordinator

Basic course of

Action for check

Expired date

1. Store coordinator visit the

Home page and home page

Display

Click on link ()

Send ()

d

i

s

p

l

a

y

i

f

t

h

e

r

e

e

x

i

s

t

(

)

Send ()

Check ()

Display successful

Message ()

2. click on the check expired

Link

3. Then the system checks if

There is expired drug or not

4. The system displays the

List if there is an expired

Drug

5. Click on clear button

6. Then the system display

A successful message

Click on clear ()

Figure 2.7 Sequence diagram for check expired date

### 2.8.3 Conceptual class model

Class diagram is static model that shows the classes and the relationships among classes that remain constant over the time. Class is the main building block of class diagram, which stores and manages information in the system. A class diagram describes the type of objects in the system and the various kinds of static relationships that exist among them. That shows the class of the system. Their interrelation ships (including inheritance, aggregation and association) and the operations and attributes of the class and constraints those apply to the way objects are operations and attributes of the class and constraints that apply to the way objects are connected.

Figure 2.8Analysis level class diagram

**Drug**

Drug\_name:var char(8)

Company\_name:var char(10)

Batch\_number:int

quantity:int

unit price:int

shelf\_number:int

expiry\_date:date

register drug()

sale drug()

check expire date()

**Account**

account id:int

user Name:var char(12)

password:var char(14)

confirm password:var char(12)

position:var char()

Create Account()

Delete Account()

Update Account()

change password()

**Employee**

Salary: float

emp\_id:int

F\_name:var char(12)

L\_name:var char(13)

Register employee()

Delete employee()

View employee():string

**Customer**

First Name:var char(12)

Last Name:var char(15)

age:int

sex:var char(10)

customer id:int

Register customer

():string

search()

**Sold Drug**

Sell:var char(12)

customer id: int

employee id:int

batch number:int

quantity:int

date:date

status:var char

insert()

view()

print():string

\*

1

1

1

\*

1

\*

1

view

manages

register

1

\*

\*

view

save

given to

takes

managed by

\*

**manager**

manager\_id:int

F\_name:var char

(8)

L\_name:var char

(9)

sex:var char(8)

age:int

insert()

view()

1

\*

1

1

1

# 

### 2.8.4 User interface (UI) flow diagram

User interface-flow diagrams are typically used for one of two purposes. First, they are used to model the interactions that users have with our software. Second enable to the user to gain a high-level overview of the user interface for our application. This overview is effectively the combination of all the behavioral views derived from your use cases, the result being called the architectural view of your user interface

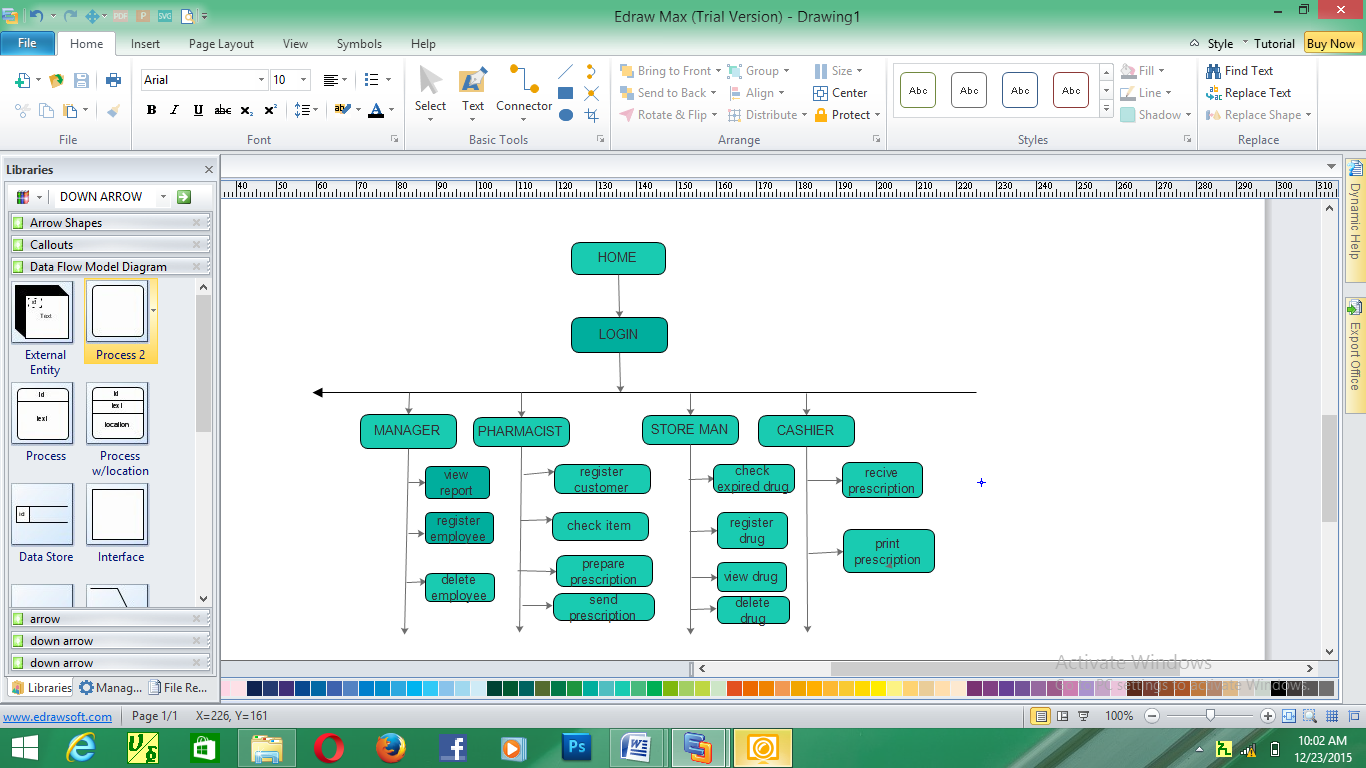
****

Figure 2.9 diagram for user interface flow diagram

### 2.8.5 Activity diagram

Activity diagram is another important diagram in UML to describe dynamic aspects of the

System. Activity diagram is basically a flow chart to represent the flow of information one

Activity to another activity.

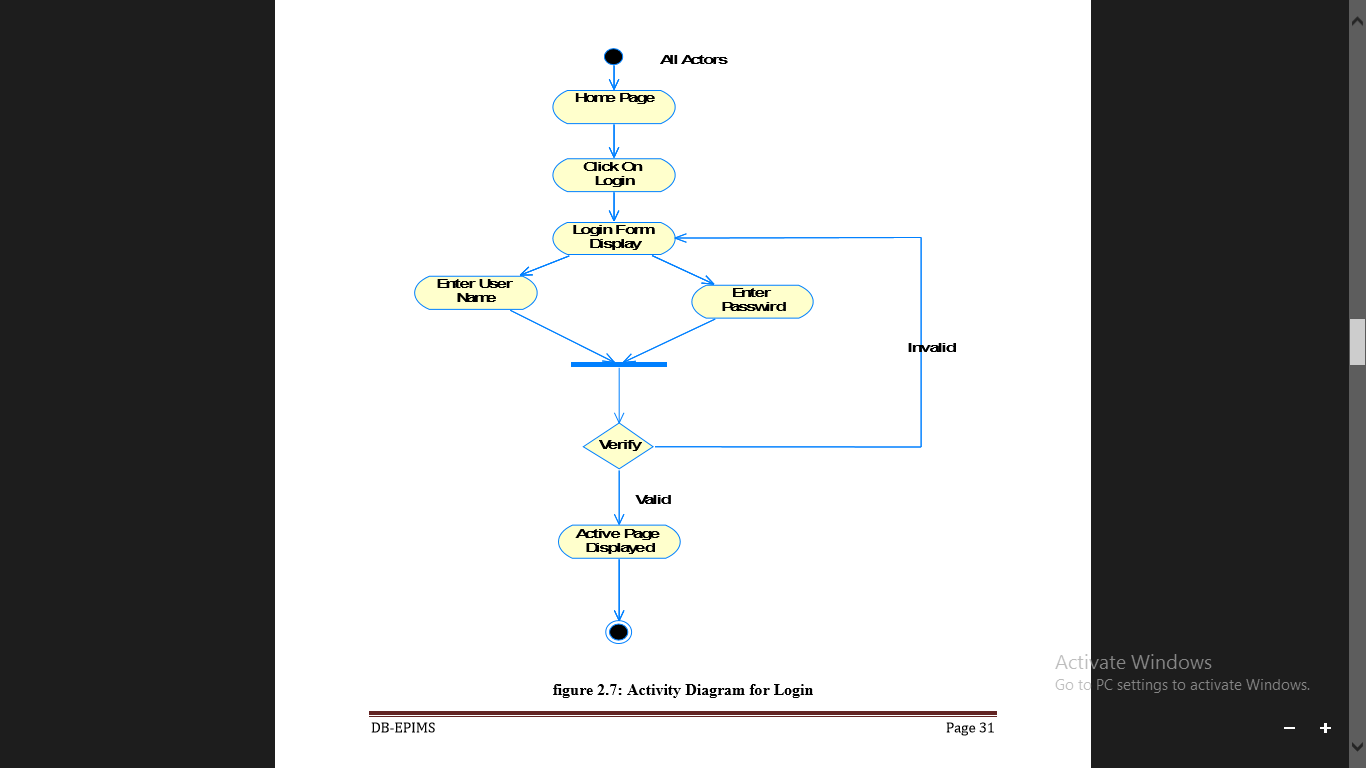


Figure 2.10 Activity diagram for login

# 

Figure2.11: Activity diagram for register drug

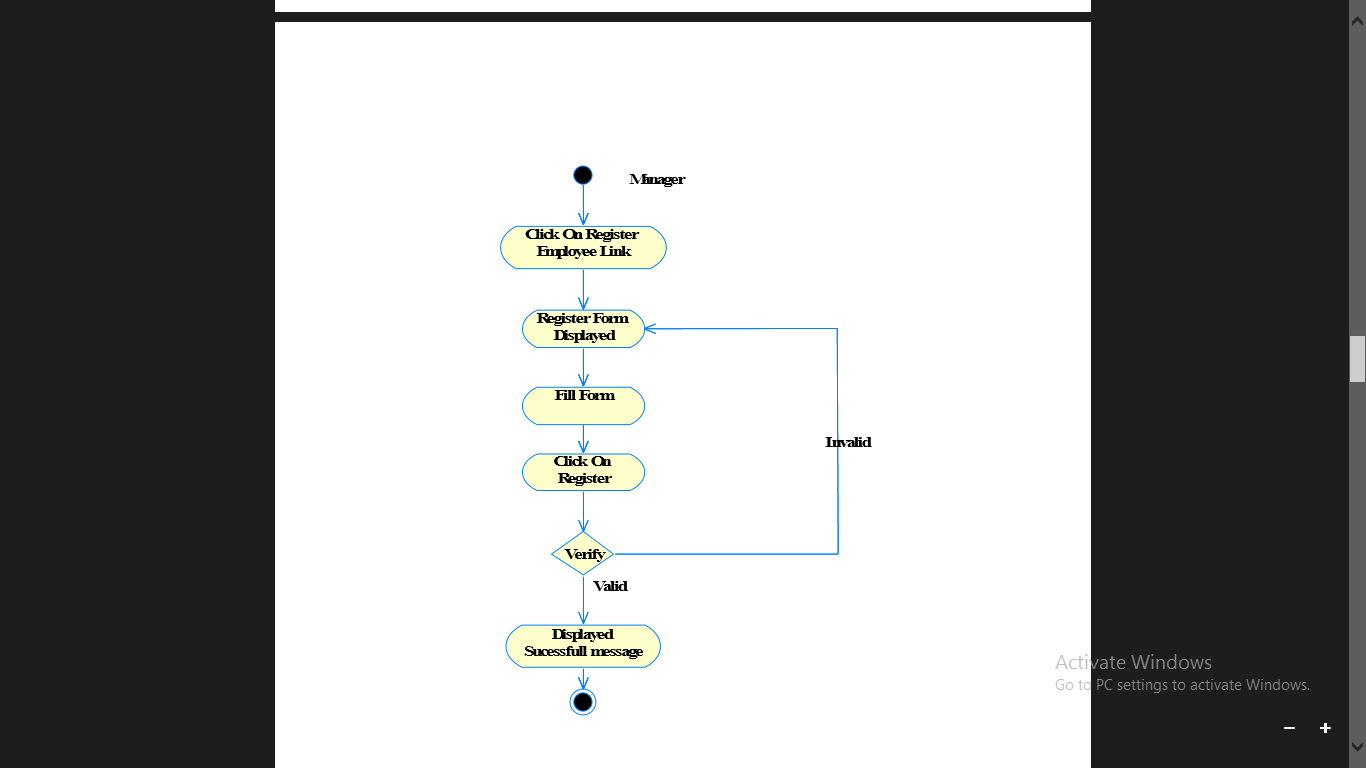
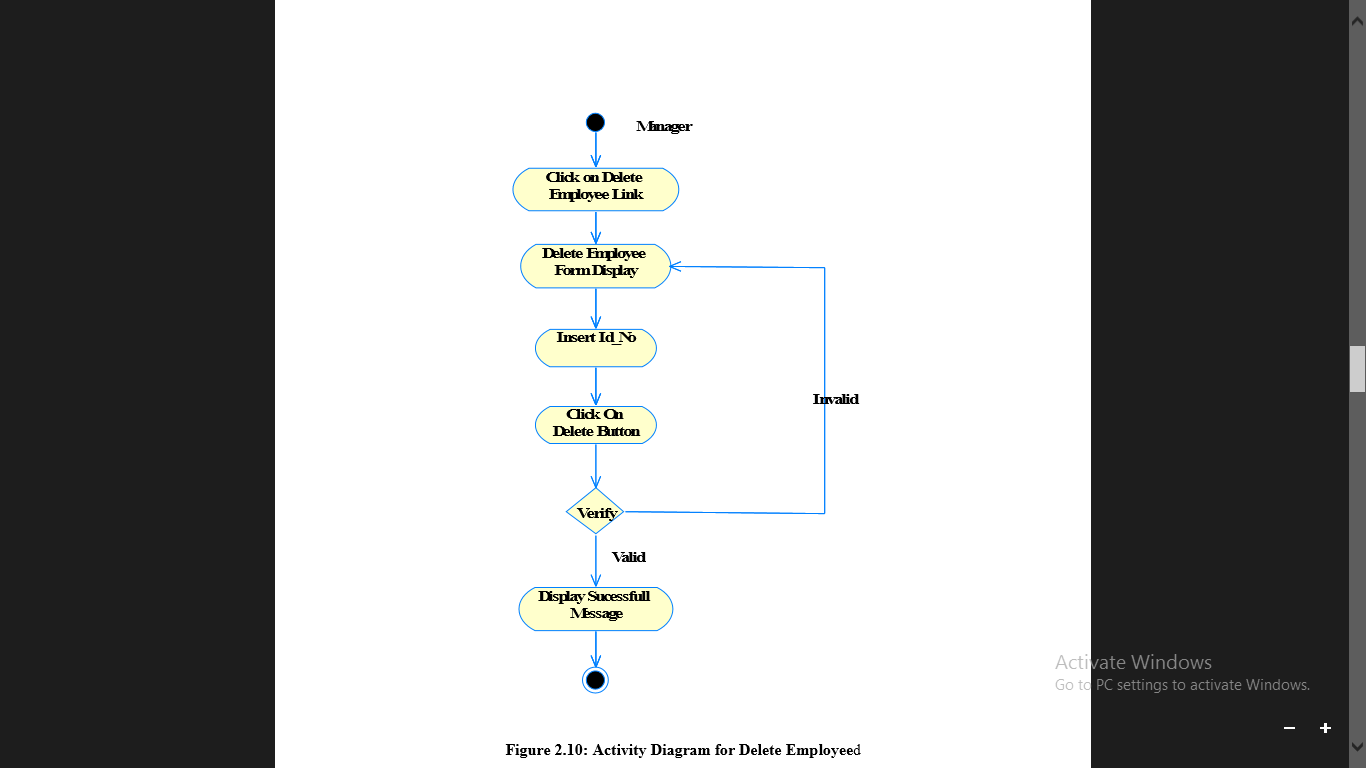


Figure 2.12: Activity diagram for register employee



# Figure 2.13: Activity diagram for delete employee

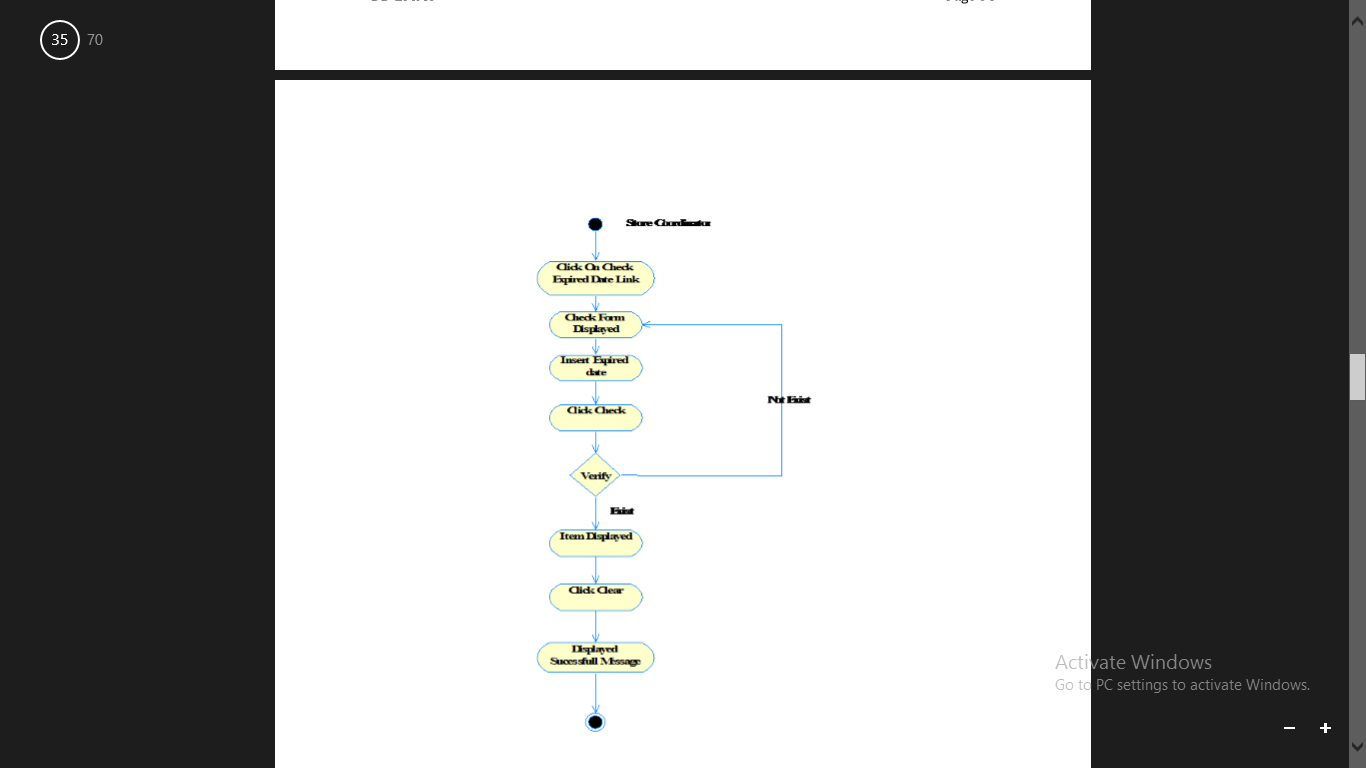


Figure 2.14: Activity diagram for check expired date

# 

# Figure 2.15: Activity diagram for sale drug

### 

### 2.8.6 User interface prototype

User interface prototyping is an iterative analysis technique in which users are actively involved in the making up of the UI for a system. UI prototyping has two purposes: First, it is an analysis technique because it enables us to explore the problem space our system addresses. Second, UI prototyping enables us to explore the solution space of our system. The user interface prototyping of this project is depicts as follow.

LOGIN

User name

Password 

Position



Cancel

LOGIN

# Figure 2.16: User interface prototype for login

**Employee registration form**

First name

Last name 

Employee id 

Sex 

Age 

Salary 

Address 

Position 

# Figure 2.17: User interface prototype for employee registration

Drug registration form

Drug Name

Company Name 

Batch Number 

Quantity 

Unit Price 

Shelf Number 

Expiry Date 

Cancel

Register

Figure 2.18 user interface prototype for drug registration

# CHAPTER THREE: DESIGN SPECIFICATION

## 3.1 Introduction /overview

Systems design is the transformation of the analysis model into a system design model. This chapter mainly concerned with the design part of the pharmacy management system. The purpose of this chapter is to provide an overview as to how to actually build the proposed system and to obtain the information needed to derive the actual implementation of our system. In addition to these systems design makes the implementation easy the design is very important. In this section we will see different types of system modeling techniques that will be used for the implementation of the system such as component modeling, deployment diagram, data base design and class mapping [4].

## 3.2 System architecture

The system architecture defines how pieces of the application interact with each other, and what functionality each piece is responsible for performing. There are three main classes ofapplication architecture. They can be characterized by the number of layers between the user and the data. The three types of application architecture are single-tier (or monolithic), two-tier, and n-tier, where n can be three or more. In a three-tier or a multi-tier architecture has client, server and database. Where the client request is sent to the server and the server in turn sends the request to the database. The database sends back the information/data required to the server which in turn sends it to the client. So our system has three tier architecture representations [2].

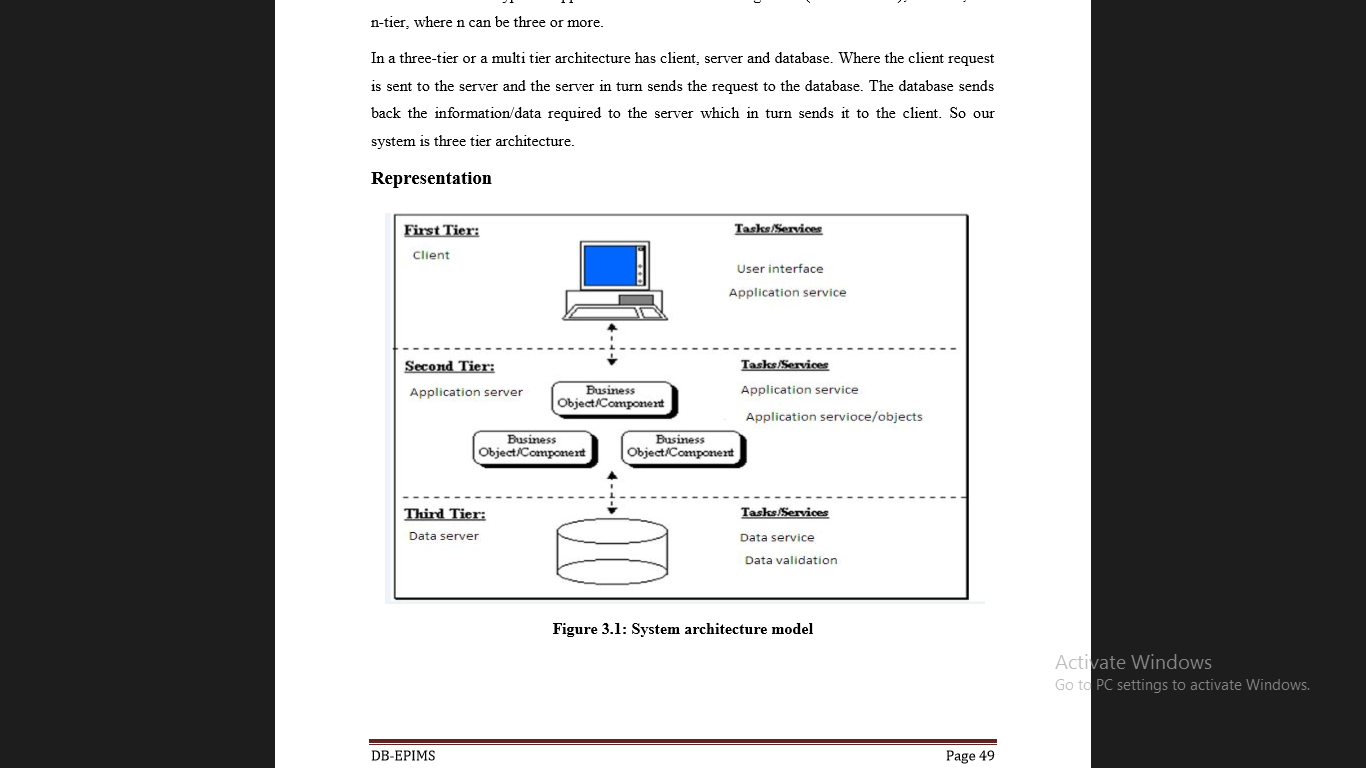


Figure 3.1 System architecture modelfor the new system

### 3.2.2 Describing an Architecture using UML

#### [3.2.2.1 Class Type Architecture](#_Toc297189556)

M

A

I

N

S

Y

S

T

E

M

User interface

Process (application, controller)

Domain/business rules

Persistence data

Database

Figure 3.2 Classtypearchitecture

The purpose of class type architecture is the static structure of how our software will be built the only difference with that of analysis class architecture is that it focuses on domain solution rather than problem domain and it introduce changes to our class model based on implementation technologies in the project we have six classes.

**User interface layer**

This layer wraps access to the logic of the system.  There are two categories of interface class – user interface (UI) classes that provide people access to the system and system interface (SI) classes that provide access to external systems to the system.  Graphical user interface (GUI) screens implement UI classes.

**Controller\process layer**

The process layer implements business logic that involves collaborating with several domain classes or even other process classes.

**Business\domain layer**

This layer implements the concepts pertinent to the business domain such as drugs focusing on the data aspects of the business objects, plus behaviors specific to individual objects.

**Persistence layer**

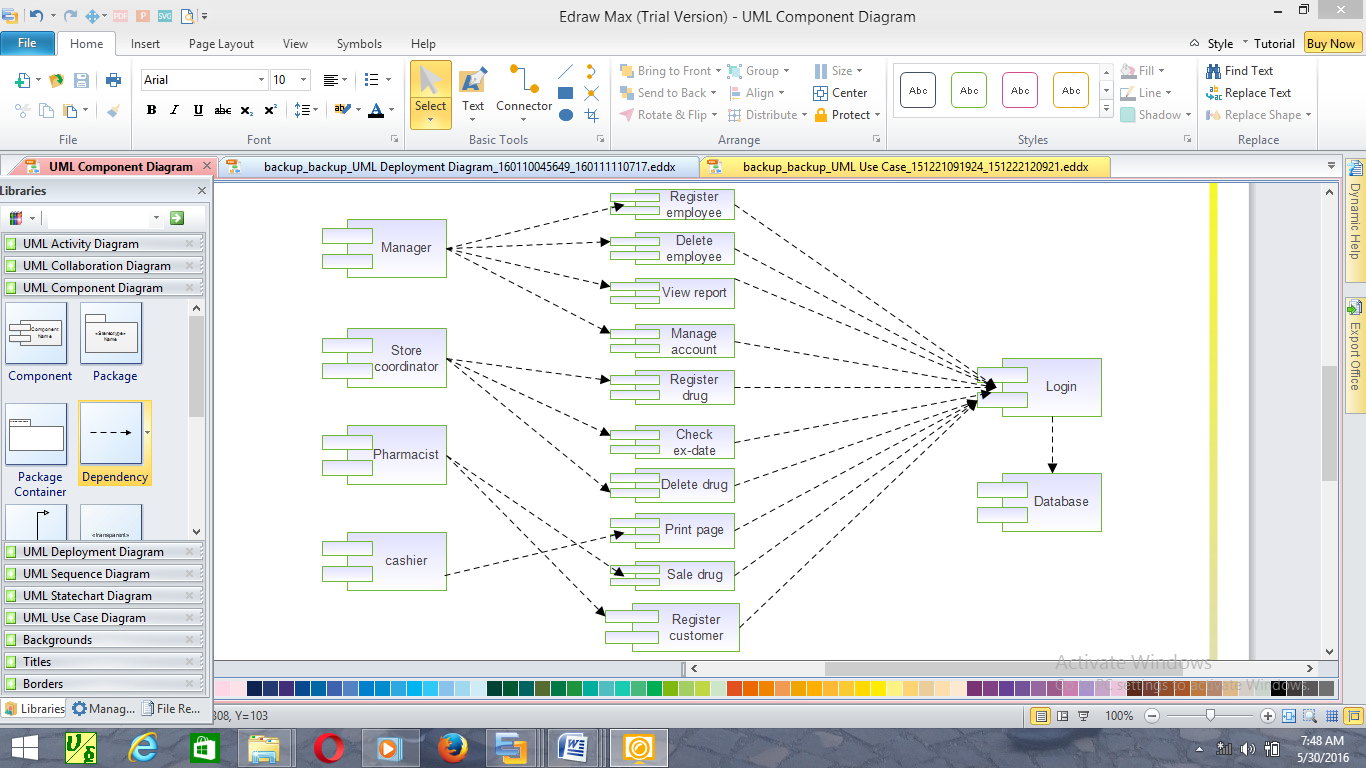
Persistence layers encapsulate the capability to store, retrieve, and delete objects/data permanently without revealing details of the underlying storage technology.

**System classes**

System classes provide operating-system-specific functionality for the applications, isolating the software from the operating system.

#### [3.2.2.2 Component Diagram](#_Toc297189557)

Component diagrams are often used to model high-level software components and how they interact. The interfaces between these components become clear as the model grows, which provides a much clear delineation of duties of each component. So from that point component diagrams are used to visualize the physical components in a system. These components are libraries, packages, files etc. 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. It does not describe the functionality of the system but it describes the components used to make those functionalities. A single component diagram cannot represent the entire system but a collection of diagrams are used to represent the whole.

Figure 3.3 component diagram

#### 3.2.2.3 Deployment Diagram

The deployment diagram shows how the software components, processes, and objects are deployed into the physical architecture of the system. It shows the configuration of the hardware units (e.g. Computers, communication devices, etc.) and how the software components are distributed across the units [4].

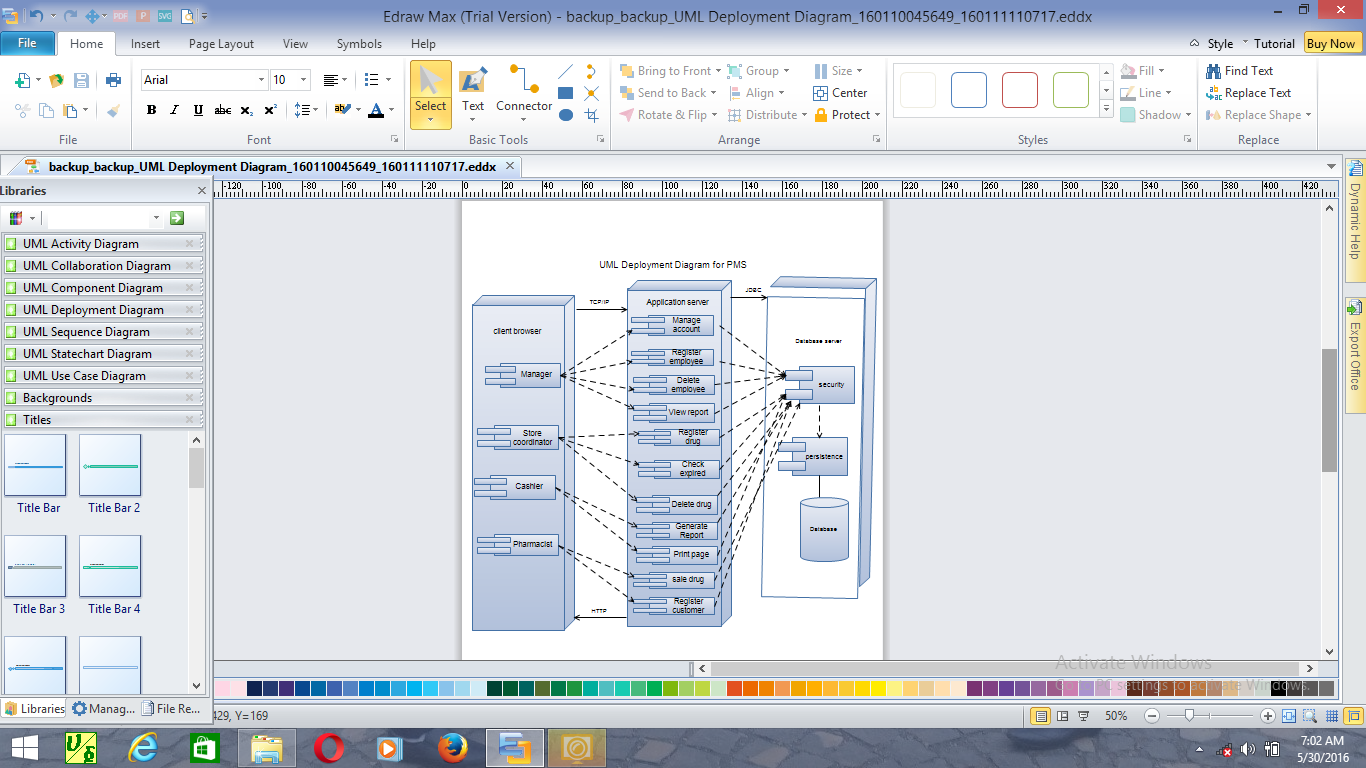
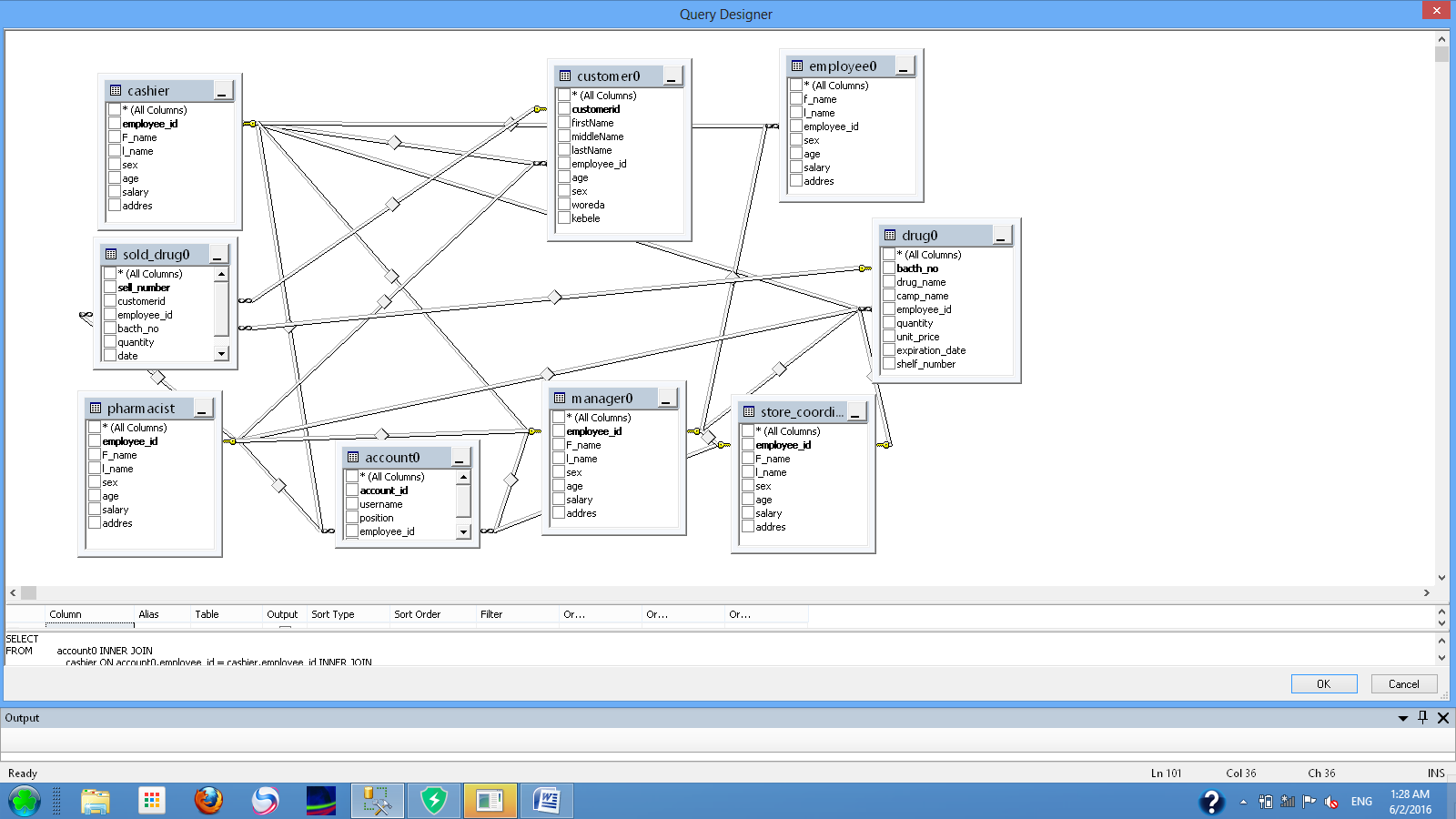


Figure 3.4 deployment diagram

#### [3.2.2.4 Persistent Diagram](#_Toc297189559)

Persistent Diagram is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity. It can be thought of as the logical design of the base data structures used to store the data [4].

Figure 3.5 persistent diagram

### [3.2.3 Detailed Design of the System](#_Toc297189560)

#### [3.2.3.1 State chart Diagram](#_Toc297189563)

A state chart diagram is a view of a state machine that models the changing behavior of a state. State chart diagrams show the various states that an object goes through, as well as the events that cause a transition from one state to another. The common model elements that state chart diagrams contain are: States, Start and end state, Transitions.

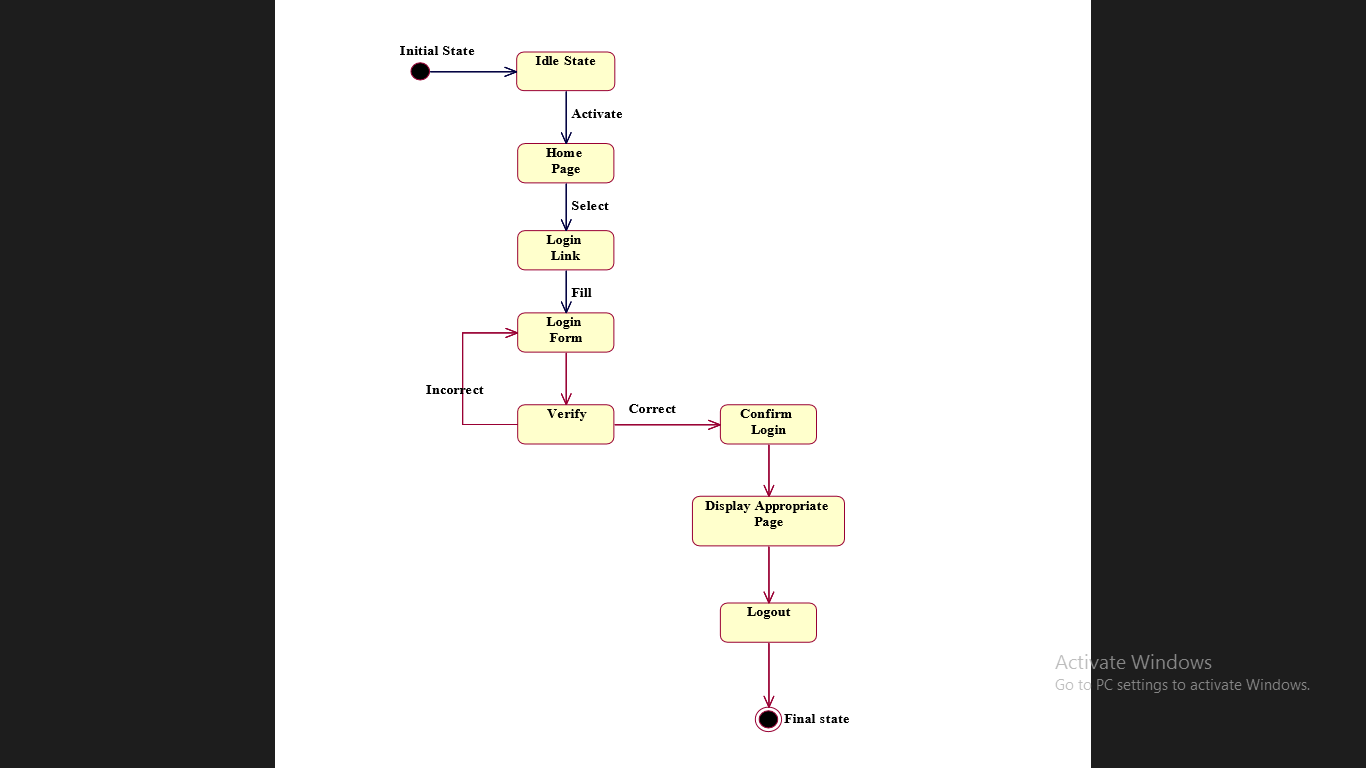


Figure 3.6 State chart digram for login

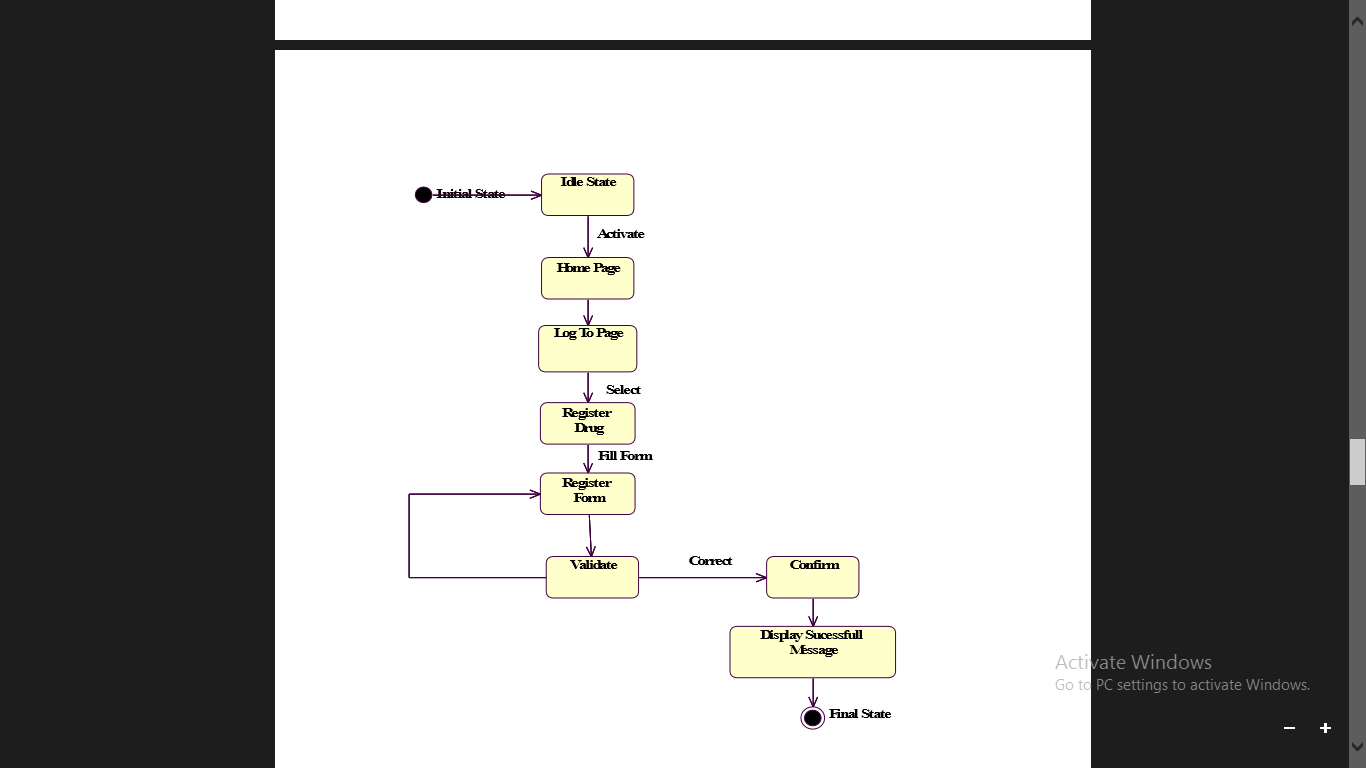


Figure 3.7 State chart digram for register drug

# 

# 

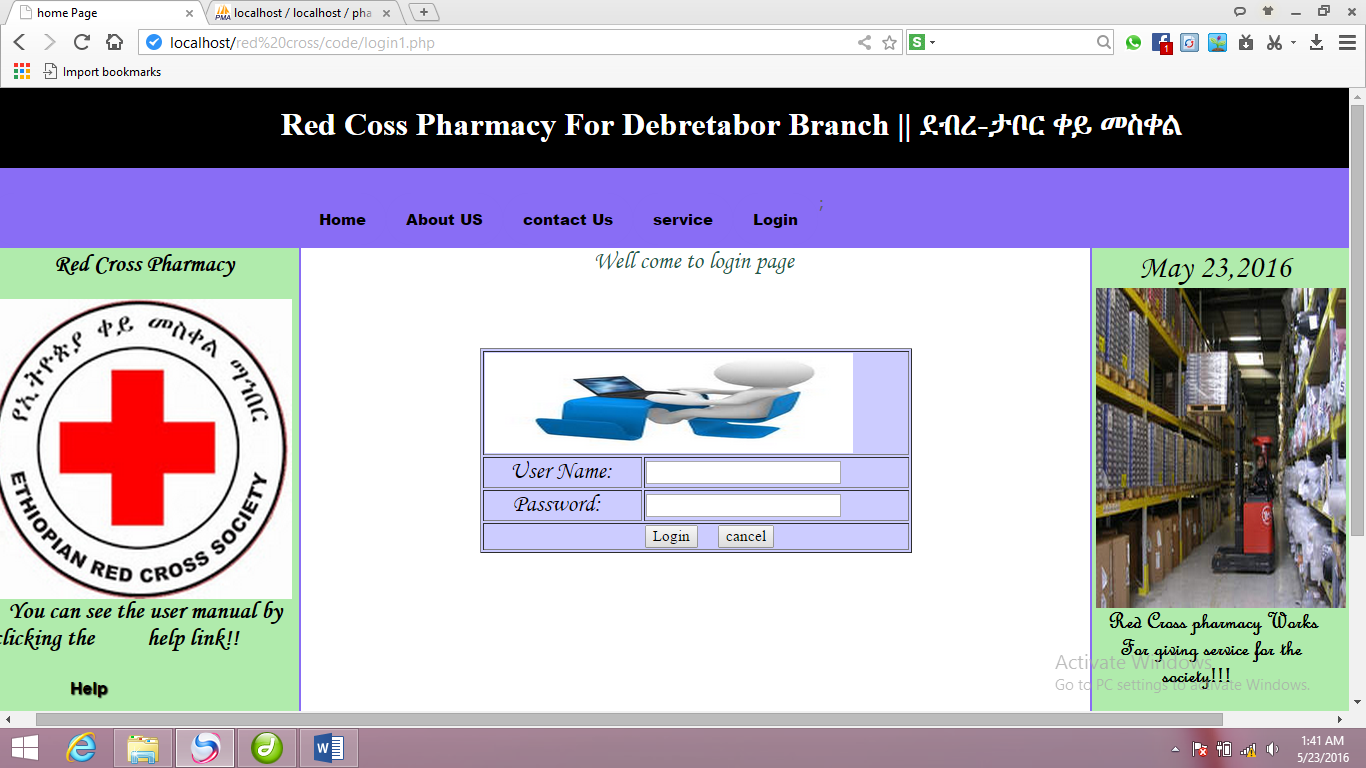
Figure 3.8 State chart diagramsfor delete employee

# 

Figure 3.9 state chart diagrams for delete drug

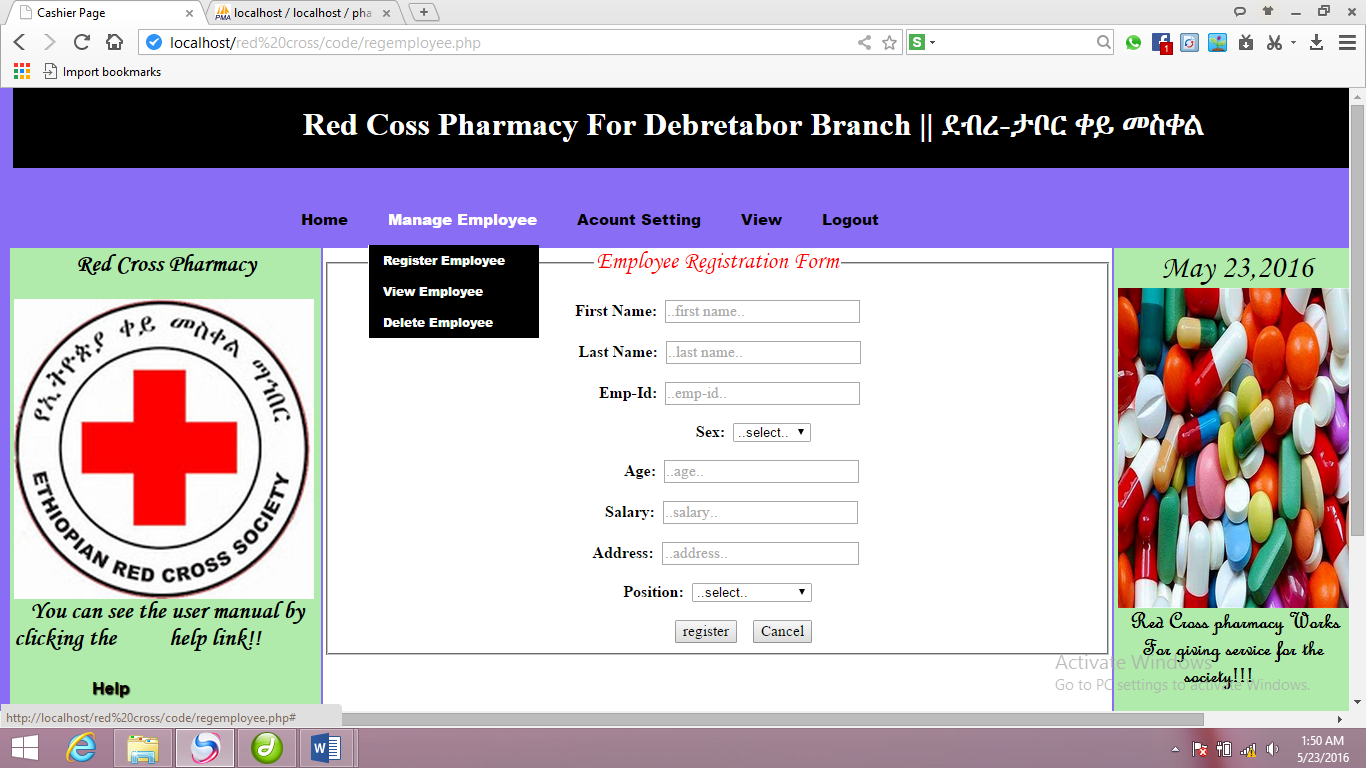
#### 3.2.3.2 User interface design

**User interface design for login page**

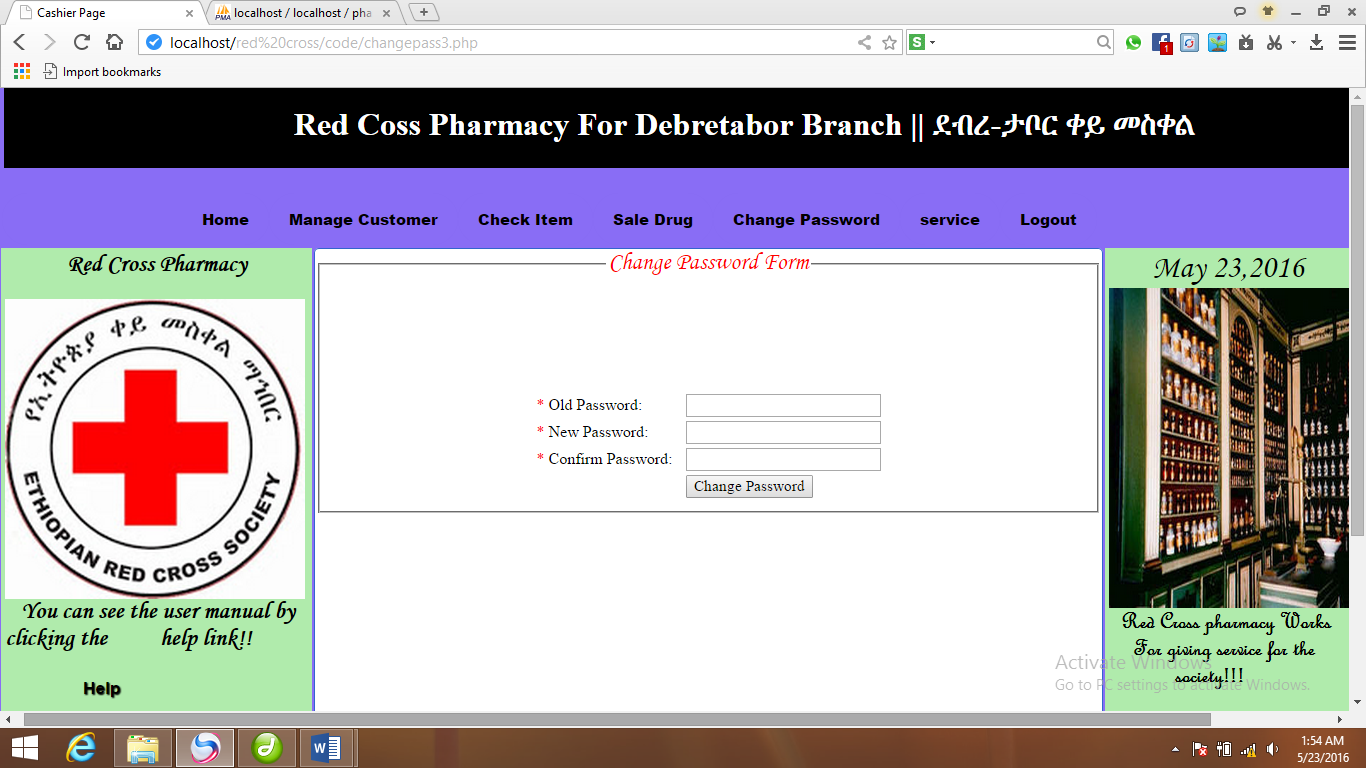


**Figure 3.10User interface design for login page**

**User interface design for employee registration**

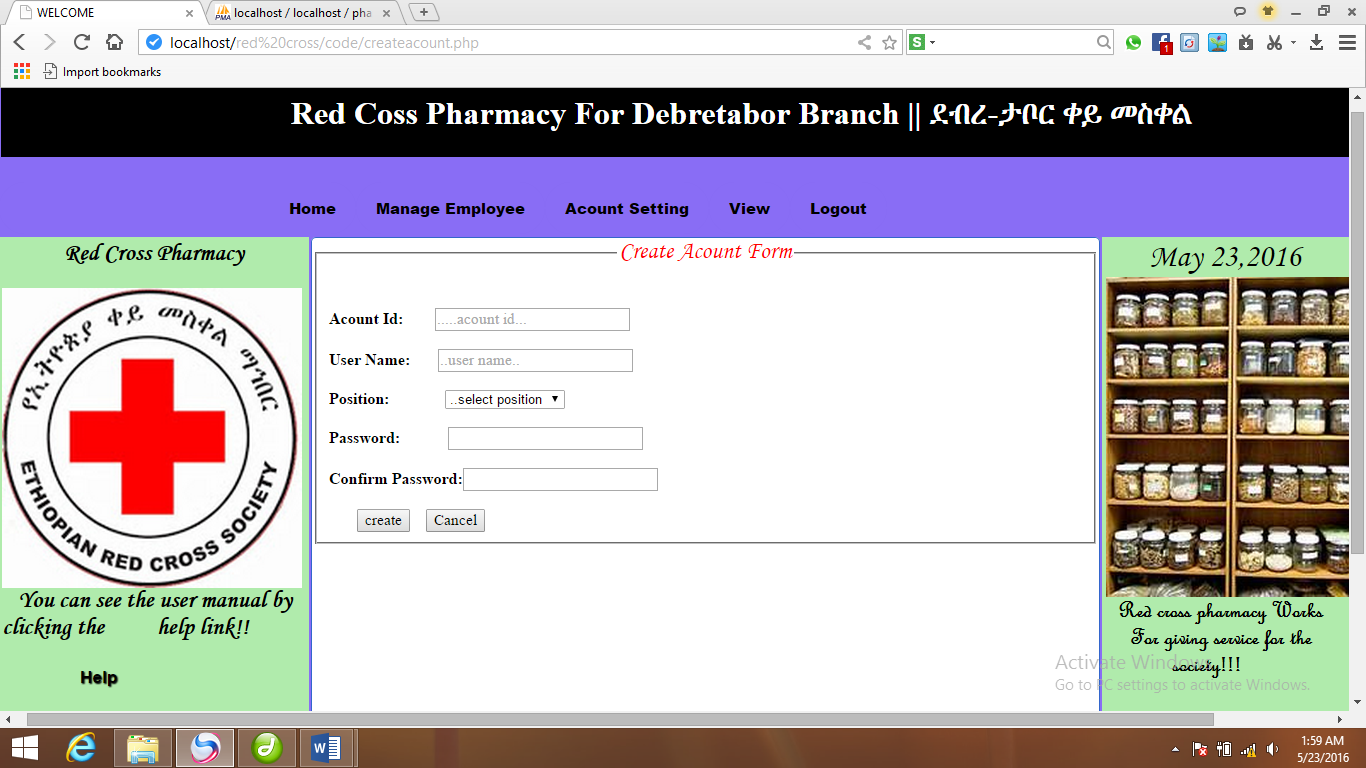
**Figure 3.11User interface design for employee registration**

**User interface design for change password**



**Figure 3.12User interface design for change password**

**User interface design for create account**

**Figure 3.13 User interface design for create account**

# CHAPTER FOUR

# Implementation

This chapter basically highlights the issues dealt with the implementation phases. Implementation is the phase where objectives of physical operations of the system turned into reality i.e. real working model. In this phase the coding convention has made it possible as it’s the real phase of objectivity to reality. Then the code is tested until most of the errors have been detected and corrected.

The goal of implementation is to introduce our system for the users in real sense that how they use this new system which is developed for their intended objectives.

## 4.1 Overview of programming language used

Choosing a programming language depends on your language experience and the scope of the application you are building. The choice of programming language to use for this project is PHP, java script and HTML. The implementation of this programming language is very simple, particularly as to the executable code.

## 4.2 Algorithms used

**Algorithm for checking whether the field is empty or not**

If field is empty

Display error Message enter the required value

**Algorithm for checking whether the field takes only Numbers or characters**

If user inputs incorrect data type

Display error message “enter the correct data type”

## 4.3 sample code

**Sample code for login**

<?php$conn=mysql\_connect("localhost","root","");

$db1=mysql\_select\_db("pharmacy",$conn);

if(isset($\_POST['login']))

{$uname=$\_POST['username'];

$password =$\_POST['password']; $query = "SELECT \* FROM account WHERE username = '{$uname}' AND password = '{$password}' ";

$result\_set=mysql\_query($query);

$row=mysql\_fetch\_array($result\_set);

$user=$row['position'];

if(mysql\_num\_rows($result\_set)>0)

{if($user==manager){session\_start();

$\_SESSION['accountid']=$row['accountid'];

echo "<script>window.location='manager.php';</script>";}else if($user==storecoordinator){session\_start();

$\_SESSION['accountid']=$row['accountid'];

echo "<script>window.location='storecoordinator.php';</script>";

}else if($user==pharmacist)

{session\_start();

$\_SESSION['accountid']=$row['accountid'];

echo "<script>window.location='pharmacist.php';</script>";}

else if($user==cashier){

session\_start();$\_SESSION['accountid']=$row['accountid'];

echo "<script>window.location='cashier.php';</script>";}}

else{echo" <font color='green'><p class='wrong'>User Name and password not match!</p></font>";

echo' <meta content="3;login1.php" http-equiv="refresh" />';} }

mysql\_close($conn);

?>

**Sample code for register drug**

<?php

$connection=mysql\_connect("localhost","root","");

$db1=mysql\_select\_db("pharmacy",$connection);

if(isset($\_POST['register']))

{$dname=$\_POST['dname'];$cname=$\_POST['cname'];$bnumber=$\_POST['bnumber'];

$quantity=$\_POST['quantity'];

$unitprice=$\_POST['price'];

$shelfno=$\_POST['shelfno'];

$exdate=$\_POST['edate'];

$res=mysql\_query("INSERTINTOdrug values('$accountid','$dname','$cname','$bnumber','$quantity','$unitprice','$shelfno','$exdate') ");

if(!$res){echo"<fontcolor='green'><pclass='wrong'>Insertion failed!!!</p></font>";echo'<metacontent="3;registerdrug.php"http-equiv="refresh" />';}else{echo"<fontcolor='green'><pclass='right'>Youhave succefully registered drug in to your data base!!!</p></font>";

echo' <metacontent="3;registerdrug.php"http-equiv="refresh" />'; }}mysql\_close($connection);?>

# 

# CHAPTER FIVE

# 5. TESTING

## 5.1 Overview of Testing

After a successful completion of developing the software, we must test it for its correct functionality of the system according to customer requirement and scope boundaries. Testing a code and other artifacts as we go along the development of the system help us to acquire the following advantage:

* If improves the quality of the software.
* It reduces the cost of the testing phases.
* It shows the programmers that they are making real progress.
* It reduces the number of faults that are linked to the program during the testing phase.

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

## 5.2 Sample Test

Testing performed on each modules or block of codes during development.

When we test the login authentication without fill the user name the system alerts please fill user name

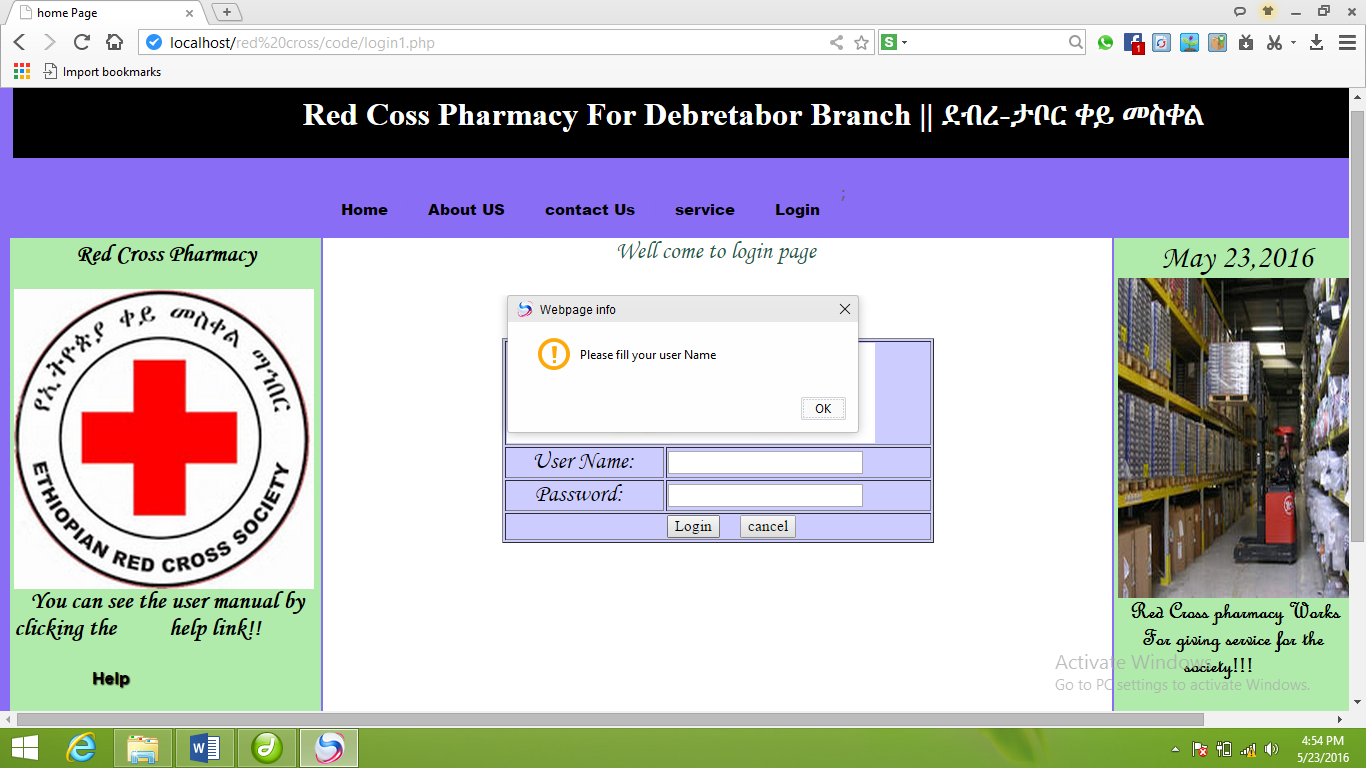


Figure 5.1 Sample test for login page

When we test whether account is created successfully or not by fill all text boxes with its data type then the system alerts account created successfully.

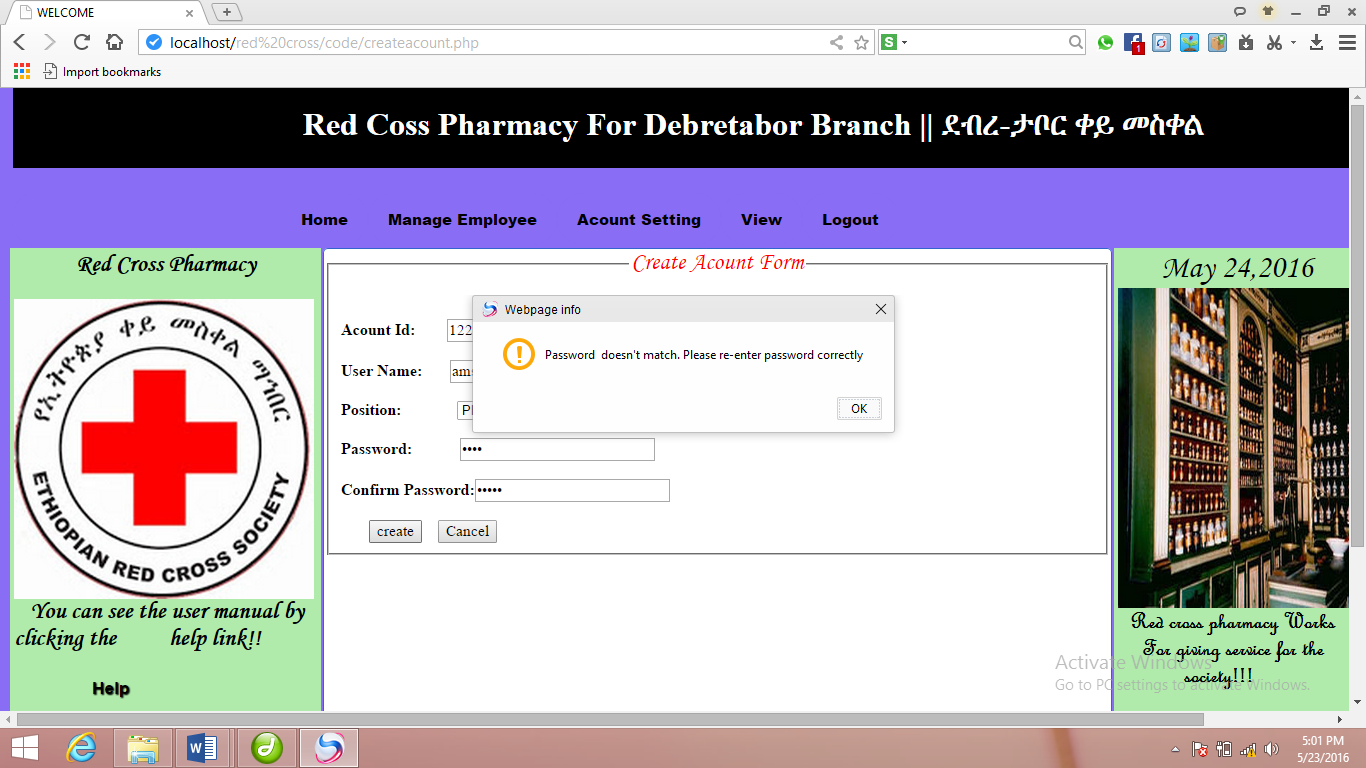


Figure 5.2 sample test for create account

5.3Unit Testing**.**To conduct test in this method we select sample code (one function or module) and run it separately to look its correct functionality. For example we can take module that checks whether the drug is expired or not according to given range and checking it.



Figure 5.3 unit test for expired date of drug

5.4 Acceptance Testing**.** Testing of the product done by the actual end users. During the interface design process, usability evaluation has an important role that includes interactive cycle of designing and evaluation. Used to demonstrate that the system can be ready to use the end use. It tested with the real data in real or simulated environment. Where product being delivered to customer and then customer execute the acceptance test see whether the expectation of the functionality meet/fulfill their requirements.

# CHAPTER SIX

# CONCLUSION AND RECOMMENDATIONS

## 6.1 Conclusion

Effective documentation of this software will take care of the basic requirements of the pharmacy management system because it is capable of providing easy and effective of information storage related to the activities happening in the stipulated area. With these, the objectives of the system design will be achieved.

In order to allow for the future expansion, the system has been designed in such a way that will be allowed the possible modification as it may deem necessary by the pharmacy management, whenever the idea arises.

## 6.2 Recommendation and Future Enhancement

Designing this software (Pharmacy management system) is not an easy task. It has been started from the requirement gathering and passes through so many other stages before completion.

Based on the benefits of this system and tremendous value it will add to customer-user satisfaction, the below recommendation will be considered;

It is recommended that the new system should be used with the necessary specifications of the system requirements and provision for an uninterrupted power supply should be made available throughout the hours of operation of the pharmacy to avoid power outage. There should also be basic computer knowledge for the users of the software.

It is recommended that the software would be improved especially in areas of accounting as it will be of great impact to the development of sell pharmacy.

**Future Enhancement**

The group members can plan for the future to maintain and deploy the system and to expand the functionality of the system in the administration. Such as:

* Payroll functions.

# 

# References

[1] American Society of Health-System Pharmacists. (2016). ASHP statement on the

[2]Barbara Griggs. *Green Pharmacy: The History and Evolution of Western Herbal Medicine, Second Edition.*

[3] Brian Wilkerson, **Object oriented design** 1st edition, 2004.Pharmacy technician’s role in pharmacy informatics. American Journal of Health-System Pharmacy, 71(3), 247–50.

[4]Charles E. Rosenberg, Morris J. Vogel. *The Therapeutic Revolution: Essays in the social history of American Medicine, Second Edition.*

[5] "Clinical Informatics 2014 Diplomates". American Board of Preventive Medicine. December 2013. Retrieved 7 January 2015.

[6] S.A Kalkar, **Software Engineering**, 3rd edition, 2007.

[7]Workers in red cross pharmacy.

# Appendix

**Interview Questions**

The main objective of this interview questions is to elicit requirement fromRed Crosspharmacy that will help for the development of a system. The interviewer records the respondents‟ response using pen, pencil and paper during managing, are there any problems?

1. How drug information management system is going on?

2.During managing, are there any problems?

3.Who is responsible for what?

4.When your organization has been established?

5.How much members are present in your company?

6.Is their work division among those members?

7.How can you differentiate outdated drugs from normal drug?

**Forms of the existing system**

የኢትዮጵያ ቀይ መስቀል ማህበር

Ethiopian Red Cross society

የእቃ መቀበያ ደረሰኝ ቅጽ

Goods receiving note NO 284049ላኪው ድርጅት………………………………………….. የያዘው ክፍል…………………………………………..

Sender org ordering Dept ያጓጓዢ ስም…………………………………………. Transported by

የማጓጓዣ ሰነድ ቁጥር………………………………….. የሰሌዳ ቁጥር………………………………………..

Waybill No plate No

የመሸኛ ሰነድ ፋይል ቁጥር…………………………… ቀን………………………………………………………….

Packing list No date

ደረሰኝ ቁጥር………………………………………………

Receipt No

የትዛዝ ቁጥር………………………………………………….

Order No

ተረካቢ ድርጅት………………………………………….

Receiving org

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Code No | Type of Goods | location | Measurement | quantity | Unit price | Total price |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

ፊርማ ……………………………………………………. ……………………………………….

አስረካቢ ተረካቢ

Figure 2.2 goods receiving for the existing system

የኢትዮጵያ ቀይ መስቀል ማህበር

Ethiopian Red Cross society

|  |  |
| --- | --- |
| ወጪ |  |
| ተመላሽ |  |

የዕቃ ማዉጫና መመለሻ ሠነድ ቅጽ No 111919

Issues or turn-in form

ከ ቀን

from Date

ለ የሂሳብ መደብ

To A/C No

የአሽከርካሪዉ ስም የሰሌዳ ቁጥር

Driver’s name plate No

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | የዕቃዉ አይነት  Type of Goods | መላኪያ  unit | የተሰጠ ብዛት  Amount | ያንዱ ዋጋ  Unit price | ጠቅላላ ዋጋ  Total price |
|
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

ስምና ፊርማ ላኪዉ ያጸደቀዉ ተቀባይ የመዝጋቢ ስም

Name and sign sender Approved by received by posted by

Figure 2.3 issues of turn in form of the existing system

የኢትዮጵያ ቀይ መስቀል ማህበር

Ethiopia Red Cross Society

………………………መድሀኒት ቤት /መደብር

…………………………….pharmacy/drug store

የእጅ በእጅ ሽያጭ ደረሰኝ Cash sales invoice

ከኢትዮጵያ ቀይ መስቀል ማህበር

From Ethiopia Red Cross Society

አድራሻ……………………………..ክ/ከተማ………………………ቀበሌ………

Address sub city kebele

የተ.እ.ታ.ቁጥር………………………………….

VAT No ………………………………………

መለያ ቁጥር……………………………………

Tin No………………………………………….

የተመዘገበበት ቀንና ዓ.ም…………….

Date of registration…………………

ለ………………………………………..

To……………………………………………….

አድራሻ……………………………..ክ/ከተማ………………………ቀበሌ………

Address sub city kebele

የተ.እ.ታ.ቁጥር………………………………….

VAT No ………………………………………

መለያ ቁጥር……………………………………

Tin No………………………………………….

የተመዘገበበት ቀንና ዓ.ም…………….

Date of registration……

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | የዕቃዉ አይነት  Type of Goods | መላኪያ  unit | የተሰጠ ብዛት  amount | ያንዱ ዋጋ  Unit price | ጠቅላላ ዋጋ  Total price |
|
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

የገንዘቡ ልክ በፊደል………………

የክፍያ ሁኔታ በጥሬ………….በቸክ…………ቼክቁጥር……………ቫዉቸር ቁጥር………….

የአዘጋጅ ስምና ፊርማ…………………የገንዘብ ተቀባይ ስምና ፊርማ

Figure 2.4 cash sales invoice of the existing system

# Glossary of Terms

UML - Unified Modeling Language.

CRC-Class Responsibility Collaboration.

OOSDM-Object Oriented Software Development Methodology

HTML-hypertext markup language.

PHP- Hypertext pre processor