# CHAPTER FOUR

# SYSTEM DESIGN



## **Overview of system design**

After investigating the current working system, we were able to identify the critical problems clearly observed (that was explained in the drawback of the existing system in the description existing system). 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. Design is process of describing, organizing, and structuring system components at architectural design level and detailed design level. Design converts functional models from analysis into models that represent the solution. Our project which is web based drug store management system which deals with the problems of managing drug data and generating different report for the users. It provides more efficient, reliable, time, resources saving system.

## **Purpose of the system**

This document describes the design issues of the overall system. It provides the complete architectural overview of the proposed system. It is intended to capture and express the significant architectural decisions which have been made on the system.

## **Design goal**

The Design Goals specify the qualities of the system that should be achieved and addressed during the design of the system. The design goals for the system are grouped into four categories. These are:

* Performance
* Dependability
* Maintenance
* End user

### **Performance**

In order for the web based drug store management system is to make centralized system that the system to give the services more than two users concurrently the system should meet the following performance criteria’s.

* **Response time**: - Depending on the network connection that the user machine has the system is going to interact and respond to user’s request in a maximum of a second, if the user is just viewing the pages, but if the user’s request requires the processing of the data base, like searching for drug data, is going to take an average of 1-5 seconds of communication latency with the server system.
* **Memory**:-The client system requires an average of 10-15 megabits of RAM memory to be loaded on a user’s web browser. The server system is going to require up to 40 GB of memory to store all the data and other components of the system.

### **Dependability**

Web based drug store management system should achieve the following dependability characteristics in order to resist crash and be available and reliable.

* **Robustness**: - since the system is a web based system, that mainly use a menu driven entry there wouldn’t be an input problem by the user side. But for the server side there might be an error during the process of entering a data. In this time the system will provide an error page and the system will continue without failure or crush.
* **Availability**: - as long as there is an internet connection the system will be available 7 days a week and 24 hours a day for authenticated users.
* **Security**: - the system will provide a user name and password that will manage their own page according to their level of access.
* **Reliability**: the information provided by the system is as reliable as it is presented on the web page interface, and this is maintained by the persistent database.

### **Maintenance**

In time of failure or need modification the system need to be maintained. To be maintainable the system should meet the following maintenance criteria.

* **Extensibility**: - if it is needed to add new functionality to the system, this must be achieved by only making a separate page and integrate this page with the existing system.
* **Modifiability**: - if in the system, some functionality requires to be modified, this modification must be done specifically to that function or page without affecting the overall system organization.
* **Portability**: - the system is developed to be viewed and retrieved from any web browser regardless of their version and platform it resides in it.
* **Readability**: - the system code can be viewed by clicking on the current web page and choose “view the source code” option.

### **End user**

From the user point of view the system should provide the following end user criteria’s so that the system can achieve at least 90% usability by the user.

* **Utility**: - in order to help the user, to easily understand and interact with the system, the system must provide the following utilities.
* Mouse over tips
* Help menu
* **Usability**: to enhance the usability of the system, the system should be designed incorporating the following usability concepts
* Consistent page pattern
* Less overcrowded interface.

### **Backup and Recovery**

We have used backup mechanisms such as removable flash disks, CDs and hard disks. Because the data might lose due to computer viruses or power fluctuation at the time of formatting our computer. And also we use sql server backup mechanism by importing data to other working drives into the computer or flash disk.

### **No Redundancy**

The proposed system will avoid repetition of data anywhere in the database.

### **Priorities of design goals**

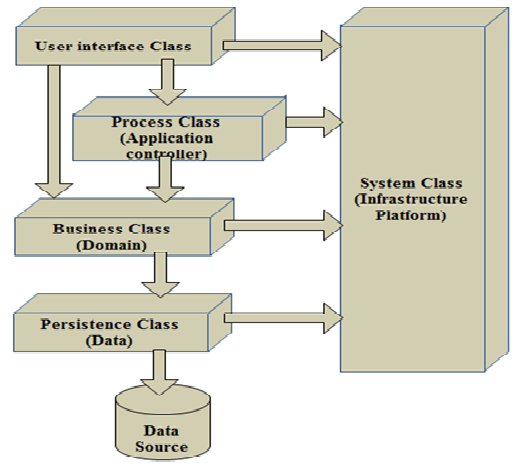
The design goals of the DMS are prioritized as follows

Table table for priority of design goals

|  |  |
| --- | --- |
| **Priority** | **Design goal** |
| 1 | End user |
| 2 | Performance |
| 3 | dependability |
| 4 | Maintenance |

## **Class type architecture diagram**

* **User interface Layer**:-provides user interface for accessing the system for programmer and end user. There are two categories of interface class \_user interface (UI) classes for end user and system interface (SI) class for programmer.
* **Controller/process Layer**:-the process layer implements business logic that involves collaborating with several domain classes or even other process classes in the system.
* **Business/Domain Layer**:-the class which uses to write the function which works as mediator to transfer data from process layer to persistence layer.
* **Persistence Layer**:-which is also class to get or set data to database queries back and forth. This layer only interacts with the database. The database queries or stored procedures will be written here to access the data from database or to perform any operation to the database.
* **System Layer**:-system classes provide operating-system-specific functionality for our application, isolating our software from the operating system (OS) by wrapping OS-specific feature, increasing the portability of our application.



## **Proposed system architecture**

### **Overview**

Nech Sar Primary Hospital drug store management system is a web based system, which is used to avoid the current manual system. Because of this, the system will have software architecture. In order to propose the DMS software architecture, we will user 3-tier architecture. Three-tier architecture allows any one of the three tiers to be upgraded or replaced independently. The user interface is implemented on a desktop PC and uses a standard graphical user interface with different modules running on the application server. The relational database management system on the database server contains the computer data storage logic. The middle tiers are usually multitier. In this case, we will have:

* **Presentation layer**: - Occupies the top level and displays information related to services available on a website. This tier communicates with other tiers by sending results to the browser and other tiers in the network. In order to display user data through user interface.
* **Application (business) layer**: -Also called the middle tier, logic tier, business logic or logic tier, this tier is pulled from the presentation tier. It controls application functionality by performing detailed processing. Also used to handle the data validation.
* **Data access layer**: - Houses database servers where information is stored and retrieved. Data in this tier is kept independent of application servers or business logic. Also used to communicate with the database by constructing SQL queries.

Figure 12 architectural diagram for new system

User

Local area network

Web server

Arba Minch Nech Sar Primary Hospital drugstore server

Database

AMNS drugstore

DMS

DMS admin

Server Side

Data Access Layer

## **Hardware/software mapping**

Nech Sar Primary Hospital web based drug store management system is a web-based application that will be accessed through WAN. The web server will run over xampp/wamp Server, The programming language used to develop this product will be PHP version 5.2.6 and some scripting language such as hypertext markup language (HTML), Java script (JS) and we have used MYSQL version 5.0.51b as the database management system.

Figure diagram for client server architecture

**Client/Presentation layer**



Data Access layer (MySQL database)

**Http**Php



Web server xampp

**Application layer**

## **State chart Diagram**

State chart diagram is one of the five UML diagrams used to model dynamic nature of a system. They define different states of an object during its lifetime. And these states are changed by events. So State chart diagrams are useful to model reactive systems. Reactive systems can be defined as a system that responds to external or internal events. State chart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered. So the most important purpose of State chart diagram is to model life time of an object from creation to termination. State chart diagrams are also used for forward and reverse engineering of a system. But the main purpose is to model reactive system. Following are the main purposes of using State chart diagrams:-

* To model dynamic aspect of a system.
* To model life time of a reactive system.
* To describe different states of an object during its life time.

Figure 14 state diagram for login



Figure state diagram for add feedback

****

Figure state diagram for create account

****

Figure state diagram for delete account

****

## **Class model**

Class diagrams basically represent the object oriented view of a system which is static in nature. It is generally used for development purpose. The class diagram is used to refine the use case diagram and define a detailed design of the system. The class diagram classifies the actors defined in the use case diagram into a set of interrelated classes. These functionalities provided by the class are termed “methods” of the class. Apart from this each class may have certain attributes that uniquely identify the class. This is the most widely used diagram at the time of system construction.



## **Deployment diagram**

It describes the physical architecture of the hardware and software in the system. They depict the software components, processors, and devices that make up the system’s architecture. A deployment modeling depicts a static view of the run-time configuration of processing nodes and components that run on those nodes. And it shows: -

* The hardware for the system.
* The software that is installed on the hardware.
* Depict the hardware/network infrastructure of an organization.
* Depict a major deployment configuration of a business application.



## **Component diagram**

Component diagrams show how the physical components of a system are organized. And also shows which component or objects will be accessed by whom and what type of security infrastructures it is using. It is a UML diagram that depicts the software components that comprise an application, system or enterprise. The components, their interrelationships, interactions and their public interfaces are depicted.



# **CHAPTER FIVE**

# **IMPLEMENTATION AND TESTING**



## **Overview**

Coding and Implementation refers to the Coding of all the design modules mentioned the design pattern of the system starting from requirement analysis to Design phase. After the due coding and testing the system will be implemented for the purpose it is designed and developed.

## **Database management**

In WBDMS database connection is very typical for the Web server to contact the database to get information as needed. PHP uses a technology called PHP Data Objects (PDO) to connect to the database.

* PDO (PHP data objects)

Classic PHP pages used PHP Data Objects (PDO) to access and modify databases. PDO is a programming interface used to access data. This method was efficient and fairly easy for developers to learn and implement. However, PDO suffered from a dated model for data access with many limitations, such as the inability to transmit data so it is easily and universally accessible. Coupled with the move from standard SQL databases to more distributed types of data (such as XML).

Here is the code that create a connection to the database is made.  
<?php

error\_reporting (E\_ALL ^ E\_NOTICE ^ E\_WARNING);

$con=mysql\_pconnect('localhost','root','')or die("cannot connect to server");

mysql\_select\_db('drug')or die("cannot connect to database");

?>

The above statement creates a connection to the database with a PDOConnection object. This object tells PHP where to go to get the data it needs. Since the data is stored in the same computer as the application, the SERVER is given as *localhost.*

## **Implementation detail**

### **Sample coding**

Sample coding for home page of Arba Minch Nech Sar Primary Hospital drugstore management system are shown below.

<?php

include("connect\_db.php");

//Start session

session\_start();

//Unset the variables stored in session

session\_destroy();

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<title> Arba Minch Nech Sar Primary Hospital </title>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link href="css/menu.css" rel="stylesheet" type="text/css" media="screen" />

<link href="css/menuCss.css" rel="stylesheet" type="text/css" media="screen" />

<link href="style/mystyle\_login.css" rel="stylesheet" type="text/css" />

<style>

#content {

height: auto;

}

#main{

height: auto;}

</style>

</head>

<body>

<center>

<table style="border:1px solid #3366CC;border-radius:5px " width="950px">

<tr>

<td colspan="2" height="60px" bgcolor="green">

<p align="left"><a href="contacts.php"><img width="1000px" height="100px" src="images/hd\_logo.jpg"></a>

<ul id="menu">

<ul>

<li class="active"><a href="index.php">Home</a></li>

<li ><a href="medicine.php">Medicine</a>

</li>

<li><a href="product.php">Verify</a></li>

<li><a href="contacts.php">Contact Us</a></li>

<li><a href="login.php">Login</a></li>

</ul>

</p>

</td>

</tr>

<tr>

<td valign="top" width="200px" bgcolor="green">

<br><br>

<table style="border:1px solid #3366CC;border-radius:5px" width="200px">

<tr>

<th bgcolor="#000000" style="border-radius:5px"><font color="white">Related Links</font></th>

</tr>

<tr><td></td></tr>

<tr>

<td bgcolor="#000000">

<a href="site.php"><font color="white">Site Map</font></a>

</td>

</tr>

<tr>

<td bgcolor="#000000">

<a href="about.php"><font color="white">About Us</font></a>

</td>

</tr>

<tr>

<td bgcolor="#000000">

<a href="developer.php"><font color="white">Developer</font></a>

</td>

</tr>

<tr>

<td>

</td>

</tr>

<tr><td></td></tr>

</table>

</td>

<td valign="top" style="border:1px solid #3366CC;border-radius:5px ">

<table><tr>

<!DOCTYPE html>

<html>

<head>

<title></title>

<meta http-equiv="content-type" content="text/html; charset=utf-8" />

<meta name="description" content="Made with beautiful, responsive image sliders in a few clicks. Awesome skins and animations. Image carousel" />

<!-- Start HEAD section --><!-- add to the <head> of your page -->

<link rel="stylesheet" type="text/css" href="engine1/style.css" />

<script type="text/javascript" src="engine1/jquery.js"></script>

<!-- End HEAD section -->

</head>

<body style="background-color:#d7d7d7;margin:0">

<!-- Start BODY section --><!-- add to the <body> of your page -->

<div id="wowslider-container1">

<div class="ws\_images"><ul>

<li><img src="data1/images/img1.jpg" alt="img1" title="img1" id="wows1\_0"/></li>

<li><img src="data1/images/img2.jpg" alt="img2" title="img2" id="wows1\_1"/></li>

<li><a href="#"><img src="data1/images/img3.jpg" alt="full screen slider" title="img3" id="wows1\_2"/></a></li>

<li><img src="data1/images/shr\_290913\_hospital2.jpg" alt="shr\_290913\_hospital2" title="shr\_290913\_hospital2" id="wows1\_3"/></li>

</ul></div>

<div class="ws\_bullets"><div>

<a href="#" title="img1"><img src="data1/tooltips/img1.jpg" alt="img1"/>1</a>

<a href="#" title="img2"><img src="data1/tooltips/img2.jpg" alt="img2"/>2</a>

<a href="#" title="img3"><img src="data1/tooltips/img3.jpg" alt="img3"/>3</a>

<a href="#" title="shr\_290913\_hospital2"><img src="data1/tooltips/shr\_290913\_hospital2.jpg" alt="shr\_290913\_hospital2"/>4</a>

</div></div><span class="wsl"><a href="#">image carousel</a></span>

<div class="ws\_shadow"></div>

</div>

<script type="text/javascript" src="engine1/wowslider.js"></script>

<script type="text/javascript" src="engine1/script.js"></script>

<!-- End BODY section -->

</body>

</html>

</tr></table>

</div>

<div id="footer" align="Center Arba Minch Nech Sar Primary Hospital Drugstore 2019. Copyright All Rights Reserved</div>

</td>

</tr>

<tr>

</tr>

</table>

</center>

</body>

</html>

Here is login page sample code

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<title> Arba Minch Nech Sar Primary Hospital Drugstore</title>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link href="css/menu.css" rel="stylesheet" type="text/css" media="screen" />

<link href="css/menuCss.css" rel="stylesheet" type="text/css" media="screen" />

<link href="style/mystyle\_login.css" rel="stylesheet" type="text/css" />

<style>

#content {

height: auto;

}

#main{

height: auto;}

</style>

</head>

<body>

<center>

<table style="border:1px solid #3366CC;border-radius:5px " width="950px">

<tr>

<td colspan="2" height="60px" bgcolor="green">

<p align="left"><a href="contacts.php"><img width="1000px" height="100px" src="images/hd\_logo.jpg"></a>

<ul id="menu">

<ul>

<li class="active"><a href="index.php">Home</a></li>

<li ><a href="medicine.php">Medicine</a>

</li>

<li><a href="product.php">Verify</a></li>

<li><a href="contacts.php">Contact Us</a></li>

<li><a href="login.php">Login</a></li>

</ul>

</div>

</p>

</td>

</tr>

<tr>

<td valign="top" width="200px" bgcolor="green">

<br><br>

<table style="border:1px solid #3366CC;border-radius:5px" width="200px">

<tr>

<th bgcolor="#000000" style="border-radius:5px"><font color="white">Related Links</font></th>

</tr>

<tr><td></td></tr>

<tr>

<td bgcolor="#000000">

<a href="site.php"><font color="white">Site Map</font></a>

</td>

</tr>

<tr>

<td bgcolor="#000000">

<a href="about.php"><font color="white">About Us</font></a>

</td>

</tr>

<tr>

<td bgcolor="#000000">

<a href="developer.php"><font color="white">Developer</font></a>

</td>

</tr>

<tr>

<td>

</td>

</tr>

<tr><td></td></tr>

</table>

</td>

<td valign="top" style="border:1px solid #3366CC;border-radius:5px " >

<table><tr>

<?php

include\_once 'connect\_db.php';

if(isset($\_POST['submit'])){

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

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

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

switch($position){

case 'Admin':

$result=mysql\_query("SELECT admin\_id, username FROM admin WHERE username='$username' AND password='$password'");

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

if($row>0){

session\_start();

$\_SESSION['admin\_id']=$row[0];

$\_SESSION['username']=$row[1];

header("location:http://".$\_SERVER['HTTP\_HOST'].dirname($\_SERVER['PHP\_SELF'])."/admin.php");

}else{

$message="<font color=red>Invalid login Try Again</font>";

}

break;

case 'Pharmacist':

$result=mysql\_query("SELECT pharmacist\_id, first\_name,last\_name,staff\_id,username FROM pharmacist WHERE username='$username' AND password='$password'");

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

if($row>0){

session\_start();

$\_SESSION['pharmacist\_id']=$row[0];

$\_SESSION['first\_name']=$row[1];

$\_SESSION['last\_name']=$row[2];

$\_SESSION['staff\_id']=$row[3];

$\_SESSION['username']=$row[4];

header("location:http://".$\_SERVER['HTTP\_HOST'].dirname($\_SERVER['PHP\_SELF'])."/pharmacist.php");

}else{

$message="<font color=red>Invalid login Try Again</font>";

}

break;

case 'Manager':

$result=mysql\_query("SELECT manager\_id, first\_name,last\_name,staff\_id,username FROM manager WHERE username='$username' AND password='$password'");

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

if($row>0){

session\_start();

$\_SESSION['manager\_id']=$row[0];

$\_SESSION['first\_name']=$row[1];

$\_SESSION['last\_name']=$row[2];

$\_SESSION['staff\_id']=$row[3];

$\_SESSION['username']=$row[4];

header("location:http://".$\_SERVER['HTTP\_HOST'].dirname($\_SERVER['PHP\_SELF'])."/manager.php");

}else{

$message="<font color=red>Invalid login Try Again</font>";

}

break;

}}

echo <<<LOGIN

<!DOCTYPE html>

<html>

<head>

<title> Arba Minch Nech Sar Primary Hospital Drugstore</title>

<link rel="stylesheet" type="text/css" href="style/mystyle\_login.css">

</head>

<body>

<div id="content">

<div id="main">

<section class="container">

<div class="login">

<h1>Login here</h1>

$message

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

<p><input type="text" name="username" value="" placeholder="Username"></p>

<p><input type="password" name="password" value="" placeholder="Password"></p>

<p><select name="position">

<option>--Select position--</option>

<option>Admin</option>

<option>Pharmacist</option>

<option>Manager</option>

</select></p>

<p class="submit"><input type="submit" name="submit" value="Login"></p>

</form>

</div>

</section>

</div>

<div id="footer" align="Center"> Arba Minch Nech Sar Primary Hospital Drugstore 2019. Copyright All Rights Reserved</div>

</div>

</body>

</html>

LOGIN;

?>

</tr></table>

</td>

</tr>

</table>

</center>

</div>

</div>

</div>

</body>

</html>

## **User manual preparation**

Preparing user manual is important for users of the system since it help them to understand how the system is used.

1. **Welcome page.**

All of this page’s links like home, about us, add feedback and product and service provided by drugstore can be accessed by any users expect the login part.

1. **Login**

This page gives authentication for the users to be accessed. Even the doctor could not access unless registered first by the admin. And all drug store staff also could not access without gain privilege from admin and unless uploaded first by admin. But if they once registered they can change the password into their own and use the system. In addition to this patient have to make registration for use system and to gain additional service from drugstore.

If the user success on the login page; after submission each user will get their own working place page. E.g. if the user is an admin, they will get the bellow snapshotted page. Such page is also hold for other user with their perspective job.

## **System testing**

### **Objective**

The main objectives of conducting testing on the system are:

* Testing that the system satisfies the functional requirements
* Testing that the system satisfies the non-functional requirements
* Testing the system’s weak point or failure condition

### **Scope**

This test plan covers a full system test of the DMS system. This includes operator and user procedures, as well as programs and job control. In addition to comprehensively testing multiprogramming functionality, external interfaces, security, recovery and performance will also be evaluated.

## **Feature to be tested or not to be tested**

### **Feature to be tested**

This section lists the functional requirements used for creating the test-case table, the test cases that were used to verify the interface table, and the results for the test-cases table.

Table List of Functional Requirements.

|  |  |
| --- | --- |
| Functional requirement Number | Functional requirement short description |
| FR01 | The DMS application shall have three types of authentication: Manager authentication, pharmacist authentication, and admin authentication. |
| FR02 | The DMS application shall be accessible to all the user to browse all their profile information. |
| FR03 | The manager shall be able to view the patient status they added to the DMS. |
| FR04 | The admin shall be able to add new/revised service item as well as to add/modify the employee information. |
| FR05 | The pharmacist shall be able to view all the medicine info. |
| FR06 | The manager shall be able to view all the report and comment from users. |
| FR07 | The pharmacist shall be able to check out expired drug. |
| FR08 | The pharmacist shall not be able to fill order form without providing valid information for all rows in the drug order form. |
| FR09 | The manager shall not be able to fill new medicine form if any of the columns in the drug registration form are left empty. |

### **Feature not to be tested**

* No other than mentioned above in section

## **Test case specification**

Table below shows the functional requirements used to write the test cases along with the test case numbers for each test case and a short description of the test cases.

Table Table for test case specification

|  |  |
| --- | --- |
| Test case no. | Test-case short description |
| TC01 | To test the Login/Authentication interface for the Admin. |
| TC02 | To test the Login/Authentication interface for the Pharmacist. |
| TC03 | To test the Login/Authentication interface for the Manager. |
| TC04 | To test the verifyinterface if it verify drug or not. |
| TC05 | To test, pharmacist can order replenished drug. |
| TC06 | To test, users can view their profile they add in the DMS system. |
| TC07 | To test, pharmacist can view patient information. |
| TC08 | To test, pharmacist can add new patient information. |
| TC09 | To test, admin can add new employee information. |
| TC10 | To test, manager can view all registered users. |
| TC11 | To test, manager/admin can view all user comment and report. |
| TC12 | To test that pharmacist check out expired drug. |
| TC13 | To test that pharmacist are not able to submit order form if the information in any of the fields is not valid. |
| TC14 | To test that manager are not able to submit medicine registration form if the information in any of the fields is left blank. |

## **Pass/Fail criteria**

This section lists the results that were produced by running the test cases. Table below lists the test cases that were used while testing the interface along with the expected result and the actual results for each test case.

Table Table for pass/fail criteria

|  |  |  |
| --- | --- | --- |
| Taste case number | Expected result | Actual result |
| TC01 | Pass | Pass |
| TC02 | Pass | Pass |
| TC03 | Pass | Pass |
| TC04 | Pass | Pass |
| TC05 | Pass | Pass |
| TC06 | Pass | Pass |
| TC07 | Pass | Pass |
| TC08 | Pass | Pass |
| TC09 | Pass | Pass |
| TC10 | Pass | Pass |
| TC11 | Pass | Pass |
| TC12 | Pass | Pass |
| TC13 | Pass | Pass |

## **Installation process**

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.

## **Start-up Strategy**

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

# CHAPTER SIX

# CONCLUSION AND RECOMMENDATION



## **Conclusion**

Providing health service to the society and managing societies’ health information task is one of the main activities to be done at Arba Minch Nech Sar Primary Hospital in past few years. So as to facilitative the working model in to an automated system, we developed a web based drugstore management system to serve the hospital with digitalized features and its friendly user interface to accomplish their tasks. The system facilitates tasks performing environment with several benefits for the employees with its remarkable features such as friendly user interface, minimum time of computation, efficient data handling and retrieval with security of the data. In addition to that it’s so quite easy in the system to add drug, register prescription of patient and to record expired drug detail. It also provide wide advantage for pharmacist when the drug in the store finished. Through various challenging, now we have come to the completion phase of this project. We believe our work will ensure a significant contribution to the unit of drugstore office in Arba Minch Nech Sar Primary Hospital and its future mission.

## **Recommendation**

According to scope of our project the team develops web application. Because of the time constraint we cannot do beyond to our scopes, but in the future the team believes that this system can be fully operational by having enough time and fully information. Finally the team would recommend that further work will do on the system in order to make the system performance better and can add further functionality.

# References

(n.d.). Retrieved from www.w3school.com

modern system anaysis and design. (2000). In A. H. Fred.

***2015.*** *Arba Minch University. www.amu.edu.et. [Online] Jan 17, 2015. [Cited: Dec 14, 2014.] http://www.amu.edu.et.*

*w3schools. www.w3schools.com. [Online] [Cited: Jan 11, 2015.] http://www.w3schools.com.*

*www.programmersheaven.com. [Online] programmersheaven. [Cited: January 10, 2015.] http://www.programmersheaven.com.*

*www.codeproject.com . [Online] codeproject. [Cited: January 12, 2015.] http://www.codeproject.com .*