

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
“Jnana Sangama”,Belagavi-18,Karnataka



DATA BASE MANAGEMENT SYSTEM
MINI PROJECT REPORT

On

Pharmacy Management System

Submitted in partial fulfilment of the requirements for Database Management

System Laboratory

Bachelor of Engineering

in

Computer Science & Engineering

by

B R Nithyashree

(1CG15CS011)

Under the guidance of

Smt.Geetha J K

Assistant Professor

Sri.Harish T A

Assistant Professor



Partnering in Academic Excellence

Channabasaveshwara Institute of Technology

(Affiliated to VTU, Belagum & Approved by AICTE, New Delhi-NAAC

Accredited & ISO 9001:2015 Certified Institution)

NH 206 (B.H.Road),Gubbi, Tumkur-572 216, Karnataka

Channabasaveshwara Institute of Technology

(Affiliated to VTU, Belagum & Approved by AICTE, New Delhi-NAAC
Accredited & ISO 9001:2015 Certified Institution)
NH 206 (B.H.Road) Gubbi, Tumkur-572 216, Karnataka



CERTIFICATE

Certified that the mini project work entitled **Pharmacy Management System** is a bonafide work carried out by **& B R Nithyashree** bearing USN **1CG15CS011** in partial fulfillment of 5th Semester **Database Management System Laboratory (15CSL58)** in **Computer Science and Engineering**, of the **Visvesvaraya Technological University**, Belagavi, during the academic year 2017 - 2018. It is certified that all corrections / suggestions indicated for the internal assessment have been incorporated in the report deposited in the departmental library. The project has been approved as it satisfies the academic requirements in respect of mini project work prescribed for the said degree.

Smt.Geetha J K
Assistant Professor
CSE,CIT

Sir.Harish T R
Assistant Professor
CSE,CIT

Name of the Examiners

1.

2.

Signature with date

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the gratitude to the person, whose constant guidance and encouragement crowned my efforts with success.

First of all, I am thankful to the Director , **Dr.Suresh** and Principal **Dr.** , CIT Gubbi for providing me an opportunity to work on a project in this institution

I would also like to thank **Dr. Shantala**, H.O.D, Department of Computer Science for being kind enough to provide me encouragement & support to do a project involving a realistic approach.

Lastly, I express my sincere and heartiest gratitude to **Smt. Geetha J K & Sir.Harsish T R** for active guidance and encouragement in the completion of this project .They were instrumental in making me understand the exact requirements and also provided continuous supervision of the project.

B R Nithyashree
(1CG15CS011)

ABSTRACT

The main aim of the project is the management of the database of the pharmaceutical shop. This project is insight into the design and implementation of a Pharmacy Management System. This is done by creating a database on the available medicines in the shop. The primary aim of Pharmacy Management System is to improve accuracy and enhance safety and efficiency in the pharmaceutical store. The aim this project is to develop software for the effective management of a pharmaceutical store. We have developed this software for ensuring effective policing by providing statistics of the drugs in stock. Pharmacy Management system is useful to maintain correct database by providing an option to update the drugs in stock. It is used to manage most pharmacy related activities in the pharmacy.

Pharmacy Management System is a management system that is designed to improve accuracy and to enhance safety and efficiency in the pharmaceutical store. This program can be used in any pharmaceutical shops having a database to maintain. The software used can generate reports, as per the user's requirements. Using this pharmacy management system user is also able to generate report within a specified period of time. The system allows the user to enter a manufacturing and expiry date for a particular product or drug during opening stock and sales transaction. The software can print invoices, bills, receipts etc. It can also maintain the record of supplies sent in by the supplier. The system will also give report showing the list of products expiry after a specified date before the product eventually expires. The system services and goals are established by consultation with system user. It also involves manual entry upon arrival of new batches of drugs and upon drug movement out of the pharmacy for a certain period. Pharmacy management system is being build. Pharmacy management system is robust, integrated technology. every month, the pharmacist may want to generate report for the movement of drugs in and out of the pharmacy, getting information about the drugs e.g. expiry date, date purchased, number of drug type left, location of a drug in the pharmacy. Pharmacy management system deals with the maintenance of drugs and consumables in the pharmacy unit. This pharmacy management system is user friendly.

TABLE OF CONTENTS

CHAPTERS	Page Number
CHAPTER 1: INTRODUCTION	1
1.1 OVERVIEW	
1.2 PROBLEM STATEMENT	
1.3 OBJECTIVES	
1.5 COMPUTER GRAPHICS	
1.6 OPEN GL	
1.7 APPLICATIONS OF COMPUTER GRAPHICS	
CHAPTER 2: LITERATURE SURVEY	6
2.1 HISTORY OF COMPUTER GRAPHICS	
2.2 RELATED DATA STRUCTURE	
CHAPTER 3: SYSTEM REQUIREMENTS	8
3.1 DESIGN CONSTRAINTS	
CHAPTER 4: DESIGN	9
4.1 PROPOSED SYSTEM	
4.2 DATA FLOW DIAGRAM	
4.3 FLOW DIAGRAM	
CHAPTER 5: IMPLEMENTATION	12
5.1 MODULE DESCRIPTION	
5.2 HIGH LEVEL CODE	
CHAPTER 6: SNAPSHOTS	17
CHAPTER 7: CONCLUSION AND FUTURE ENHANCEMENT	19
BIBILOGRAPHY	20

LIST OF FIGURES

NAME OF THE FIGURE

PAGE NUMBER

Fig 1.1 OpenGL State Machine Model	4
Fig 4.1 Dataflow Diagram	11
Fig 4.2 Flow of the Program	12
Fig 6.1 User Interaction Controls	18
Fig 6.2 Welcome Screen	18
Fig 6.3 Game in Progress	19
Fig 6.4 Output Screen showing Score	19

INTRODUCTION

1. INTRODUCTION

1.1 ABOUT THE PROJECT

This project entitled with Pharmacy Management System. This application developed using php technology as front end and backend is MYSQL.

Pharmacy Management system are employed in regulatory control and drug management ,community pharmacy, hospital pharmacy, the pharmaceutical industry, academic activities, training of other health workers, and research. In all these fields, their aim is to ensure optimum drug therapy, both by contributing to the preparation, supply and control of medicines and associated products, and by providing information and advice to those who prescribe or use pharmaceutical products.

Pharmacy management system can make the work easier by giving the details of the medicine when its name is entered. A computer gives the details of the medicine like rate of medicine. It becomes very difficult in big medical stores to handle the details of all the medicines manually, so by using this pharmacy manage system we can maintain the records of all the medicines.

A pharmacy information system must retrieve process and update the information it obtains for safe and effective use of drugs. It is used to manage drug usage in the patient health care process and to communicate a large volume of information to pharmacy and pharmaceutical firms. The pharmacy information system is normally used to support activities and the inventory. Pharmaceutical companies had developed some relationship, on marketing issues, with hospital pharmacies.

1.2 Modules Description:

It has been modularized into following modules.

- Administrator Module
- Pharmacist Module
- Cashier Module
- Manager Module

Administrator Module:

- **Pharmacist**

Here admin can add new pharmacist and can view the pharmacists list. And he has the rights to delete Pharmacist Details.

- **Manager**

Here admin can add new Manager and can view the Details Of Managers And he has the rights to delete Manager Details.

- **Cashier**

Here admin can add new cashier and can view the Details Of Cashier And He has the rights to delete Cashier Details.

Pharmacist Module

Pharmacist Module used to add and view the prescription list.

He can maintain stock details and add new Medicines.

This module gives the following details:

- Prescription
- Stock

Cashier Module

Cashier Module can collect the bills and cash. This Modules maintain the all Process of payment details.

This module gives the following details

- Process payment (include invoice_no , payment_type , total amount)

Manager Module

Manager Module used to view the users Details and Prescription Details.

And Can Add new medicines ,View Medicine details.

This module gives the following details

- View Users
- View Prescription details
- Manage Medicine Details

2. SYSTEM STUDY

2.1 EXISTING SYSTEM

Hospital pharmacy supervision is to ensure that medicines are accessed and used safely by patients and professionals both within the environments of a hospital and beyond. Pharmacy involves information processing, which means retrieving information from one file and using it to compare, update, or display information from another file. The pharmacist could play an essential role in providing accurate data for managing patient care. Information systems are now necessary to help pharmacists to perform their expanding list of daily tasks efficiently.

2.2 PROPOSED SYSTEM

Hospital pharmacy supervision is to ensure that medicines are accessed and used safely by patients and professionals both within the environments of a hospital and beyond. Pharmacy involves information processing, which means retrieving information from one file and using it to compare, update, or display information from another file. The pharmacist could play an essential role in providing accurate data for managing patient care. Information systems are now necessary to help pharmacists to perform their expanding list of daily tasks efficiently. The pharmacy information system collects ,stores and manages information related to drugs and supervises the use of drugs in line with patient care. The pharmacy information system can rely on management information system to track and dispense drugs to hospitals and health care organizations. Pharmacy information systems are also among the most widely used clinical information systems today.

SYSTEM CONFIGURATION

3. SYSTEM CONFIGURATION

3.1 Hardware configuration

RAM : 1 GB and above

Hard Disk: 60 GB and above

3.2 Software configuration

Front end: Sublime/php

Back end: Mysql

Server : XAMP

3.3 SOFTWARE FEATURES:

Design Features

Open Source:

PHP is freely available for use. The community of open source PHP developers provides technical support and is constantly improving updating the core PHP functionalities. PHP is available at free of cost under PHP General Public License and most of its associative required software's like MySQL, Text Editors and Apache Server are also freely available, so it proves very cost effective for the developers.

Cross-Platform:

PHP provides high compatibility with leading operating systems and web servers such as thereby enabling it to be easily deployed across several different platforms .PHP scripts can run across operating systems such as Linux, Windows, Solaris, OpenBSD, Mac OSX etc and also provide support for all major web servers such as Apache, IIS, iPlanet etc.

Power:

Several web tasks can now be easily perform using PHP. For example now we can develop from small websites to giant business and organizational websites, informative forums, chatting platforms, CRM solutions, e-commerce shopping carts, community websites, e-business, shopping carts and gigantic database driven sites.

User Friendly:

Designed in a user friendly manner, PHP gives more flexibility than C, C++ and ASP and overall helps in increasing traffic to the site.

Quick:

PHP is designed to work well with the web, and so things like accessing the GET and POST and working with HTML and URLs are built-ins in the PHP language. This makes it really concise and straightforward to make a website.

Extensions:

Being an open source language, a large number of libraries and extensions, to extend its core functionalities, are available for download. The source code of PHP can be modified to include custom created extensions and components thereby increasing its extensibility.

Easy Deployment:

There are many hosting companies that will, for a few dollars a month, give you a server running PHP so you can make a website really easily.

Automatically Refreshes:

Nowadays developing dynamic websites are in the huge demand due to its specific characteristics like it automatically refreshes and does not need to make much changes manually.

Community Support:

A huge advantage that PHP offers is its community. If you are looking for a particular script, chances are another user has already created something similar. Check within the PHP community for availability. Likewise, if you have created a function that others might enjoy, be sure to post the code for others.

Other Tools:

If you need to access other web based tools like Google maps (which is always advisable for a business website), or any other, PHP makes it easy to access.

Talent Availability:

You can hire PHP programmers more easily than any other language programmers since so many people know the language.

Mysql is a powerful database. It's very good and free of charge. Many developers in the world selected Mysql and php for developing their website.

The Mysql database has become the world's most popular open source database because of its consistent fast performance, high reliability and easy of use. It's used in more than 6 million installations ranging from large corporations to specialized embedded applications on every continent in the world.

Not only is Mysql the world's most popular open source database, it's also become the database of choice for a new generation of applications built on the LAMP stack(Linux, Apache, Mysql, php / Perl / python.) Mysql runs on more than 20 platforms including Linux, Window, OS/X, HP-UX, AIX, Netware, giving you the kind of flexibility that puts you in control.

Whether you're new to database technology or an experienced developer or DBA, Mysql offers a comprehensive range of certified software, support, training and consulting to make you successful .MYSQL is a free, widely used SQL engine. It can be used as a fast database as well as a rock-solid DBMS using modular engine architecture.

MYSQL Language

DDL (Data Definition Language) refers to the CREATE, ALTER and DROP statements.

DDL allows adding / modifying / deleting the logical structures which contain the data or which allow users to access / maintains the data (databases, tables, keys, views...). DDL is about "metadata".

DCL is used to grant / revoke permissions on databases and their contents. DCL is simple, but MYSQL's permissions are rather complex. DCL is about security.

Mysql Homepage

1. Scalability and Flexibility

The Mysql database server provides the ultimate in scalability, sporting the capacity to handle deeply embedded applications with a footprint of only 1MB to running massive data warehouses holding terabytes of information.

2. High Performance

A unique storage-engine architecture allows database professional to configure the Mysql database server specifically for particular applications, with the end result being amazing performance results.

3. High Availability

Rock-solid reliability and constant availability are hallmarks of Mysql, with customers relying on Mysql to guarantee around-the-clock uptime.

4. Robust Transactional Support

Mysql offers one of the most powerful transactional database engines on the market. Features include complete ACID (atomic, consistent, isolated, durable) transaction support, unlimited row-level locking, distributed transaction capability, and multi-version transaction support where readers never block writers and vice-versa.

5. Web and Data Warehouse Strengths

Mysql is the de-facto standard for high-traffic website because of its high-performance query engine, tremendously fast data inserts capability, and strong support for specialized web functions like fast full text searches.

6. Strong Data Protection

Because guarding the data assets of corporations is the number one job of database professionals, Mysql offers exceptional security features that ensure absolute data protection.

7. Comprehensive Application Development

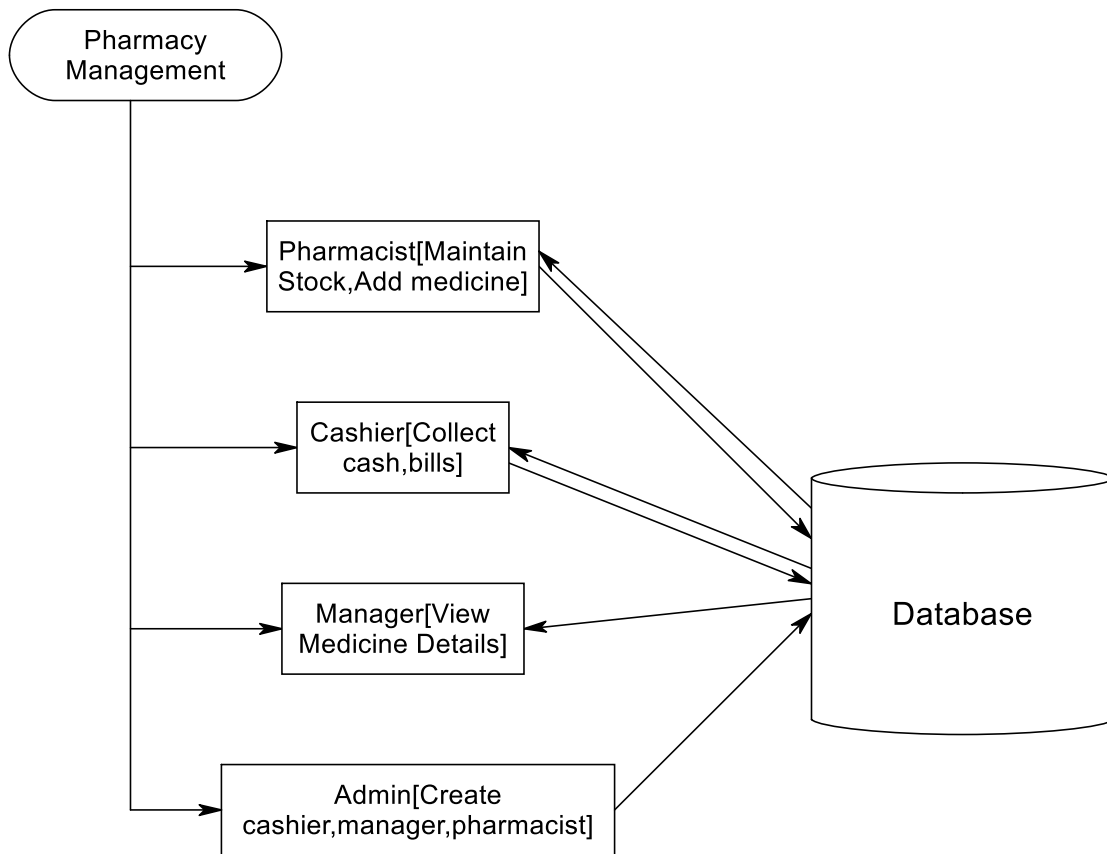
One of the reasons Mysql is the world's most popular open source database is that it provides comprehensive support for every application development need.

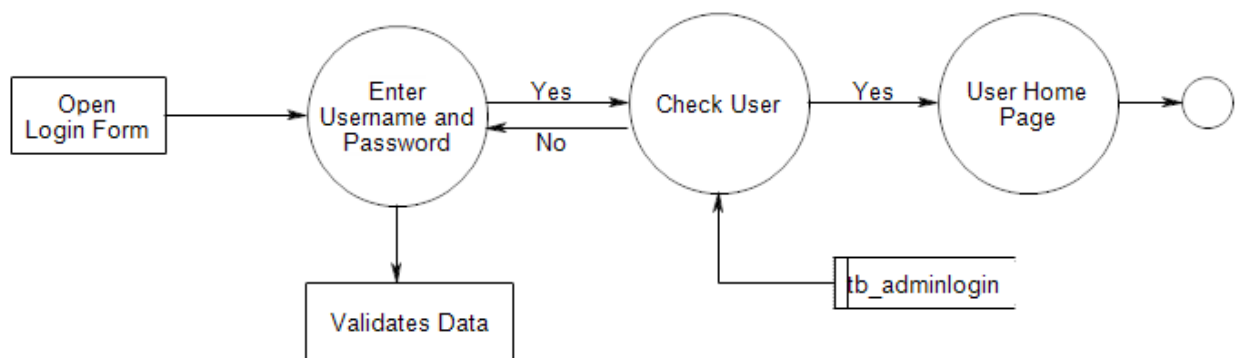
8. Management Ease

This rule holds true whether the platform is Microsoft Windows, Linux, or UNIX. Once installed, Self-management features like automatic space expansion, auto-restart, and dynamic configuration changes take much of the burden off already overworked database administrators.

4. SYSTEM DESIGN

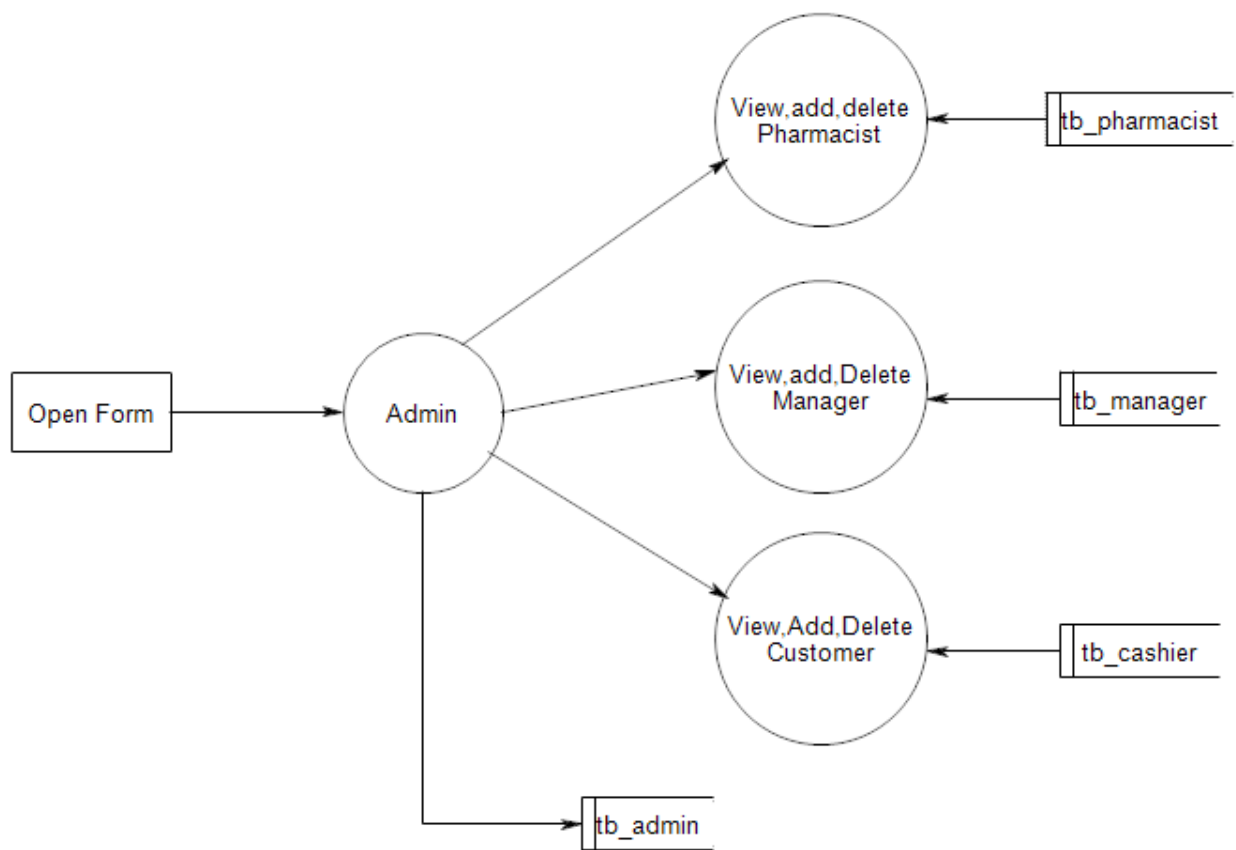
4.1 DATA FLOW DIAGRAM



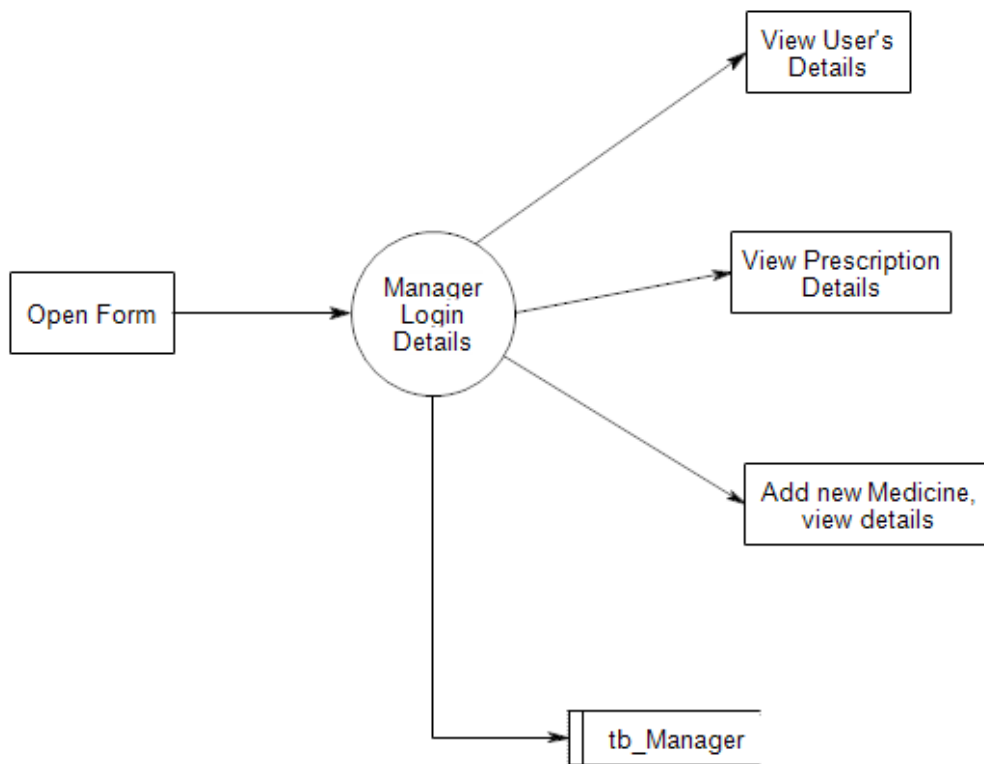


Admin Details Data Flow:

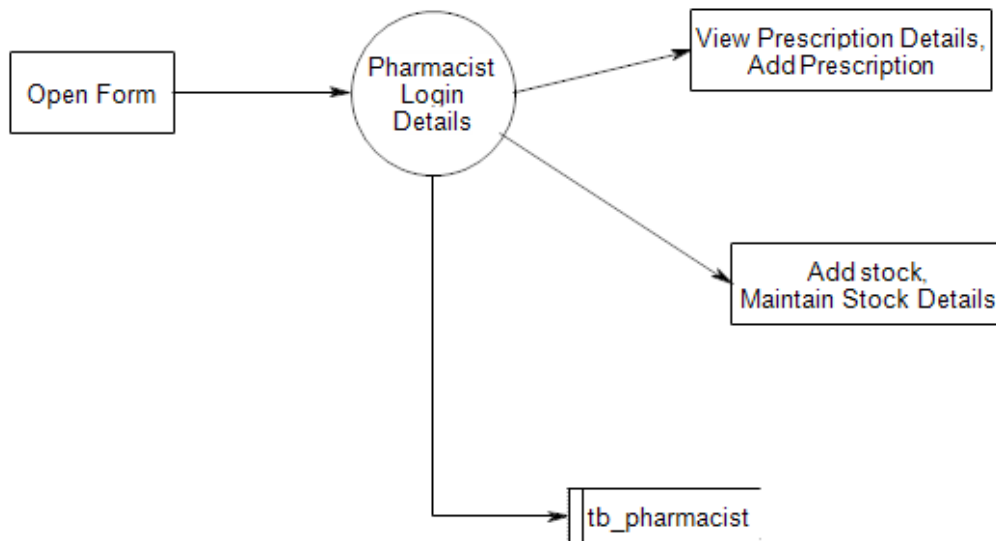
Level 1:



Level 2:



Level 3:

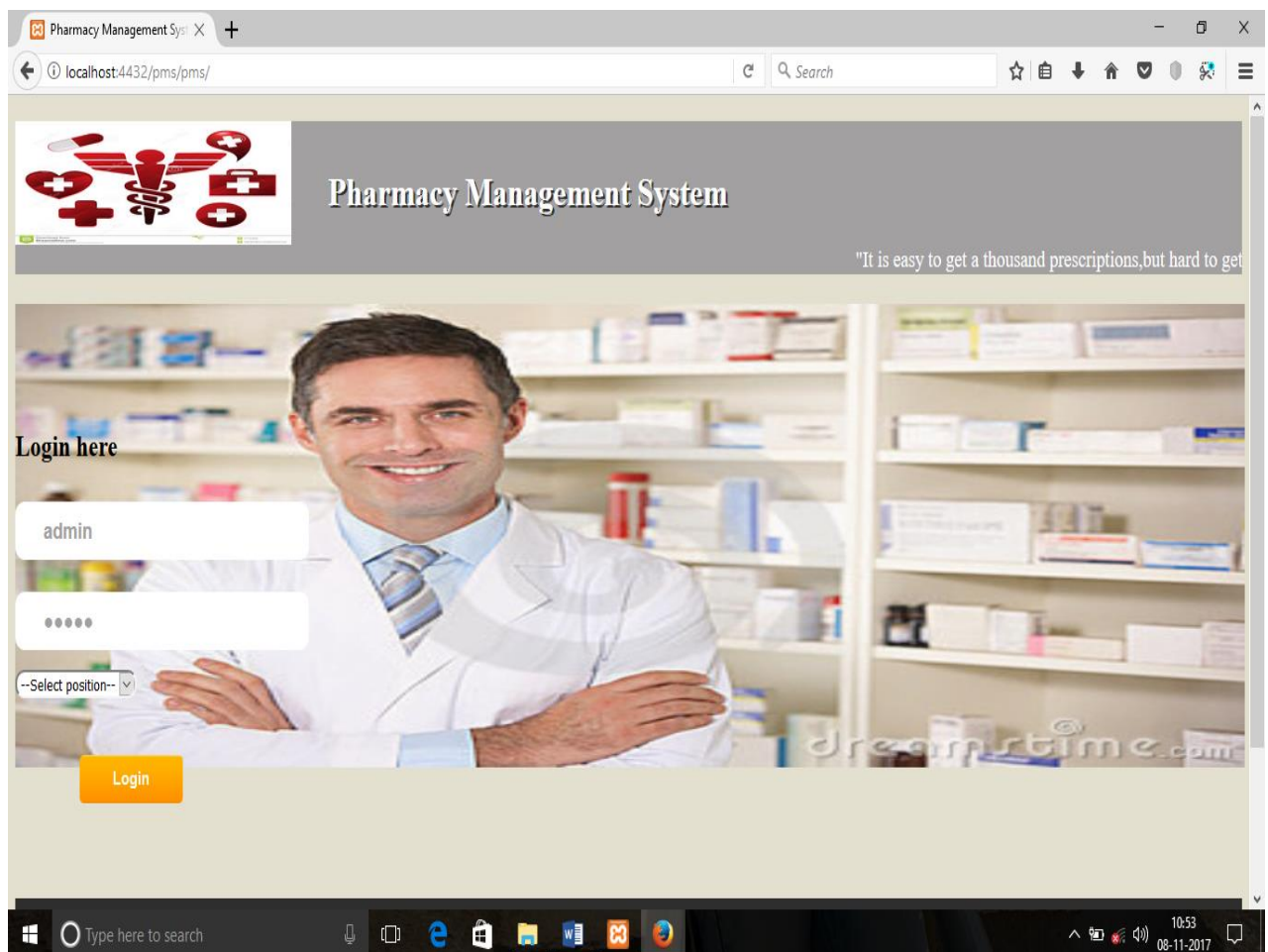


4.2 INPUT DESIGN:

Input design is the method by which valid data are accepted from the user. The valid data turn is stored as operational data in the database. Incorrect input data are the most common cause of errors in the data processing. The input design is carried out in such a way that the input screens are user friendly. The goals of designing input design are to make input data entry

as easy and error free. Input screen takes care to filter the valid data from being becoming an operational data at data entry phase.

Input design is the part of the overall system design that careful attention and is the most expensive phase. It is the point of most contact for the users with the system and so it is prone to errors.



4.3 DATA BASE DESIGN:

Table Design:

Table Name : admin

Column Name	Data Type	Description
Admin_id	Tinyint(5)	Primary Key
Username	Varchar(10)	username for login
Password	Varchar(10)	Password for login
Date	Datetime	Datetime of account created

Table Name : Cashier

Column Name	Data Type	Description
cashier_id	Tinyint(5)	Primary Key
first_name	Varchar(15)	Firstname of cashier

last_name	Varchar(15)	Lastname of cashier
Staff_id	Varchar(10)	Staffid of cashier
postal_address	Varchar(20)	Postal address of cashier
Phone	Varchar(12)	Phone number of cashier
Email	Varchar(20)	Email of cashier
Username	Varchar(10)	Username of cashier
Password	Varchar(10)	Password of cashier
Date	datetime	Date created

Table Name : Manager

Column Name	Data Type	Description
manager_id	Tinyint(5)	Primary Key
first_name	Varchar(15)	Firstname of manager
last_name	Varchar(15)	Lastname of manager
Staff_id	Varchar(10)	Staffid of manager
postal_address	Varchar(20)	Postal address of manager
Phone	Varchar(12)	Phone number of manager
Email	Varchar(20)	Email of manager
Username	Varchar(10)	Username of manager
Password	Varchar(10)	Password of manager
Date	datetime	Date created

Table Name : Pharmacist

Column Name	Data Type	Description
pharmacist_id	Tinyint(5)	Primary Key
first_name	Varchar(15)	Firstname of pharmacist
last_name	Varchar(15)	Lastname of pharmacist
Staff_id	Varchar(10)	Staffid of pharmacist
postal_address	Varchar(20)	Postal address of pharmacist
Phone	Varchar(12)	Phone number of pharmacist
Email	Varchar(20)	Email of pharmacist
Username	Varchar(10)	Username of pharmacist
Password	Varchar(10)	Password of pharmacist
Date	datetime	Date created

Table name : Stock

Column Name	Data type	Description
Stock_id	Tinyint(5)	Primary key
Drug_name	Varchar(20)	Name of the drug
Category	Varchar(20)	Category drug belong to
Description	Varchar(50)	Description
Company	Varchar(20)	Name of the company
Quantity	Int(11)	Quantity
Cost	Int(11)	Cost
Date_supplied	date	Date of supplied

Table name : Prescription

Column Name	Data type	Description
Prescription_id	Tinyint(5)	Id of prescription(Primary key)
Drug_id	Tinyint(5)	id of drug (foreign key)
Drug_name	Varchar(20)	Name of drug
Strength	Varchar(10)	Weight of drug(gms)
Dose	Varchar(10)	Dose of the drug
Quantity	Int(5)	Quantity of drug

Table name : invoice

Column Name	Data Type	Description
Invoice_no	Tinyint(5)	Primary key
Customer_name	varchar(20)	Name of customer
Drug_id	Tinyint(5)	Drug id (foreign key)
Drug_name	Varchar(20)	Name of drug

quantity	Int(11)	Quantity of drug
Cost	Int(11)	Cost of drug

Table name : Payment_details

Column Name	Data type	Description
Payment_id	Tinyint(5)	Primary key
Invoice_no	Tinyint(5)	Foreign Key
Customer_name	Varchar(20)	Name of customer
Payment_type	Varchar(20)	Payment type
Total_ammount	Int(11)	Total of amount
Date	timestamp	Date

4.4 OUTPUT DESIGN:

The output design defines the output required and the format in which it is to be produced. Care must be given to present the right information.

The output is the most important and direct source of information to the user. Efficient, output design should improve the systems relationship with the user and helps in decision making. A major form of output is a hard copy from the printer. Printouts should be designed around the output requirement to the user. The standard that is maintained for output design is clear. Output provides a permanent copy of the results for the later consultation.

Output images.....

SYSTEM TESTING AND IMPLEMENTATION

5.1 SYSTEM TESTING

INTRODUCTION:

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. In fact, testing is the one step in the software engineering process that could be viewed as destructive rather than constructive.

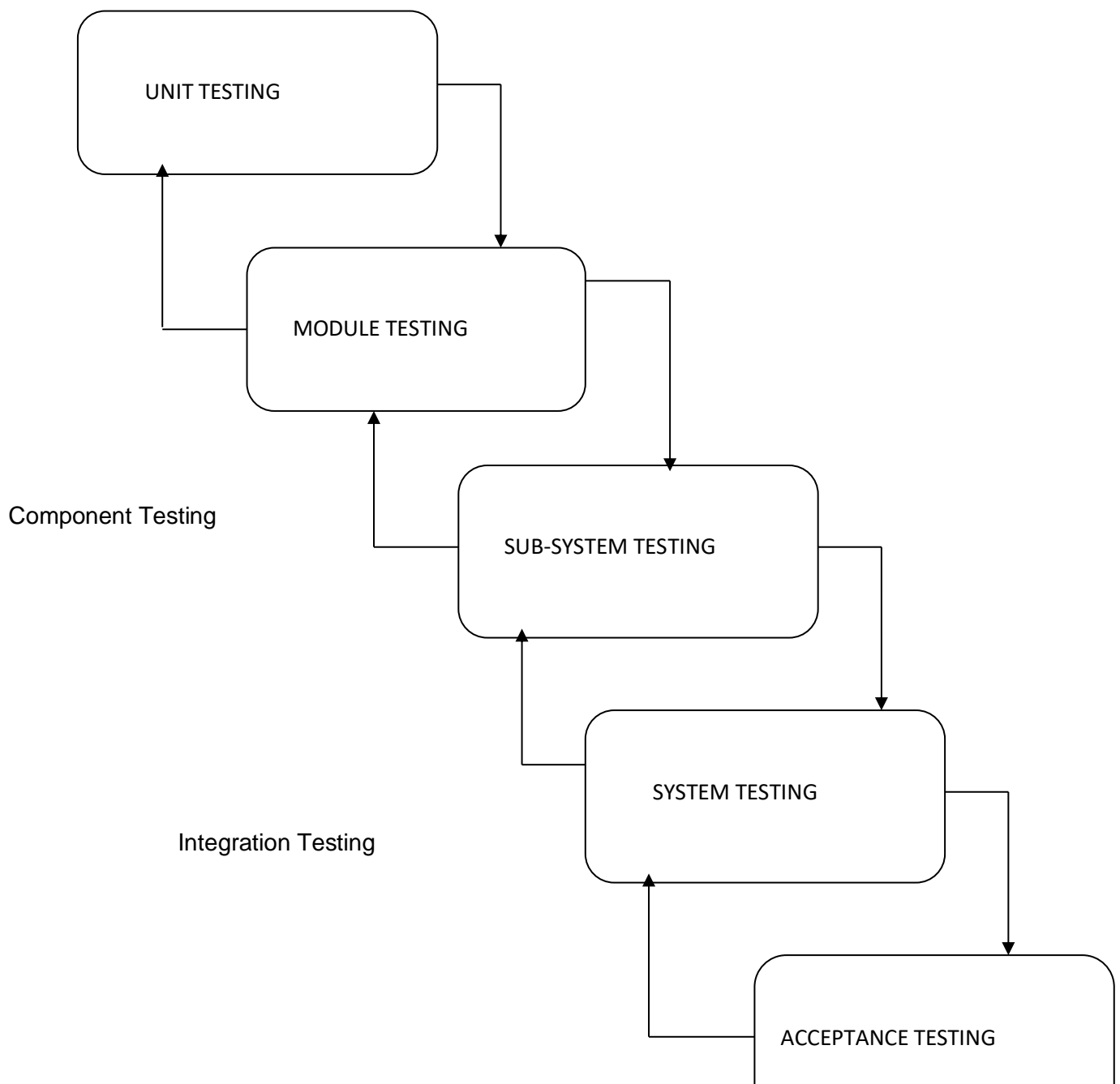
A strategy for software testing integrates software test case design methods into a well-planned series of steps that result in the successful construction of software. Testing is the set of activities that can be planned in advance and conducted systematically. The underlying motivation of program testing is to affirm software quality with methods that can economically and effectively apply to both strategic to both large and small-scale systems.

STRATEGIC APPROACH TO SOFTWARE TESTING

The software engineering process can be viewed as a spiral. Initially system engineering defines the role of software and leads to software requirement analysis where the information domain, functions, behaviour, performance, constraints and validation criteria for software are established. Moving inward along the spiral, we come to design and finally to coding. To

develop computer software we spiral in along streamlines that decrease the level of abstraction on each turn.

A strategy for software testing may also be viewed in the context of the spiral. Unit testing begins at the vertex of the spiral and concentrates on each unit of the software as implemented in source code. Testing progress by moving outward along the spiral to integration testing, where the focus is on the design and the construction of the software architecture. Talking another turn on outward on the spiral we encounter validation testing where requirements established as part of software requirements analysis are validated against the software that has been constructed. Finally we arrive at system testing, where the software and other system elements are tested as a whole.



UNIT TESTING

Unit testing focuses verification effort on the smallest unit of software design, the module. The unit testing we have is white box oriented and some modules the steps are conducted in parallel.

1. WHITE BOX TESTING

This type of testing ensures that

- All independent paths have been exercised at least once
- All logical decisions have been exercised on their true and false sides
- All loops are executed at their boundaries and within their operational bounds
- All internal data structures have been exercised to assure their validity.

To follow the concept of white box testing we have tested each form .we have created independently to verify that Data flow is correct, All conditions are exercised to check their validity, All loops are executed on their boundaries.

2. BASIC PATH TESTING

Established technique of flow graph with Cyclometric complexity was used to derive test cases for all the functions. The main steps in deriving test cases were:

Use the design of the code and draw correspondent flow graph.

Determine the Cyclometric complexity of resultant flow graph, using formula:

$$V(G)=E-N+2 \text{ or}$$

$$V(G)=P+1 \text{ or}$$

$$V(G)=\text{Number of Regions}$$

Where $V(G)$ is Cyclometric complexity,

E is the number of edges,

N is the number of flow graph nodes,

P is the number of predicate nodes.

3. CONDITIONAL TESTING

In this part of the testing each of the conditions were tested to both true and false aspects. And all the resulting paths were tested. So that each path that may be generate on particular condition is traced to uncover any possible errors.

4. DATA FLOW TESTING

This type of testing selects the path of the program according to the location of definition and use of variables. This kind of testing was used only when some local variable were declared. The *definition-use chain* method was used in this type of testing. These were particularly useful in nested statements.

INTEGRATION TESTING

Integration testing is a systematic technique for constructing tests to uncover error associated within the interface. In the project, all the modules are combined and then the entire programmer is tested as a whole. In the integration-testing step, all the error uncovered is corrected for the next testing steps

5.2 SYSTEM IMPLEMENTATION:

The Implementation Plan describes how the information system will be deployed, installed and transitioned into an operational system. The plan contains an overview of the system, a brief description of the major tasks involved in the implementation, the overall resources needed to support the implementation effort (such as hardware, software. facilities, materials, and personnel), and any site-specific implementation requirements. The plan is developed during the Design Phase and is updated during the Development Phase; the final version is provided in the Integration and Test Phase and is used for guidance during the Implementation Phase.

5.3 SYSTEM MAINTENANCE:

Every system has to perform **Maintenance** at some point or another. Whether it's just to upgrade a portion of the system or because of some problem with the system, it's an inevitable fact of operating system. And in many cases, maintenance requires taking your system for at least a few minutes.

6.CONCLUSION

It has been a great pleasure for me to work on this exciting and challenging project. This project proved good for me as it provided practical knowledge of programming in PHP web based application and MYSQL Server, and also about all handling procedure related with “**Pharmacy Management System**”. It also provides knowledge about the latest technology used in developing web enabled application and client server technology that will be great demand in future. This will provide better opportunities and guidance in future in developing projects independently.

7.FUTURE ENHANCEMENTS

Currently the Pharmacy Management System is to enhance and upgrade the existing system by increasing its efficiency and effectiveness. The software improves the working methods by replacing the existing manual system with the computer-based system. One of the main future enhancements in our system, we will include BAR CODE facility using the bar code reader, which will detect the expiry date and the other information about the related medicines.

8. BIBLIOGRAPHY

WEB SITES:

For php & Mysql installation:

<http://php.net/manual/en/install.php>

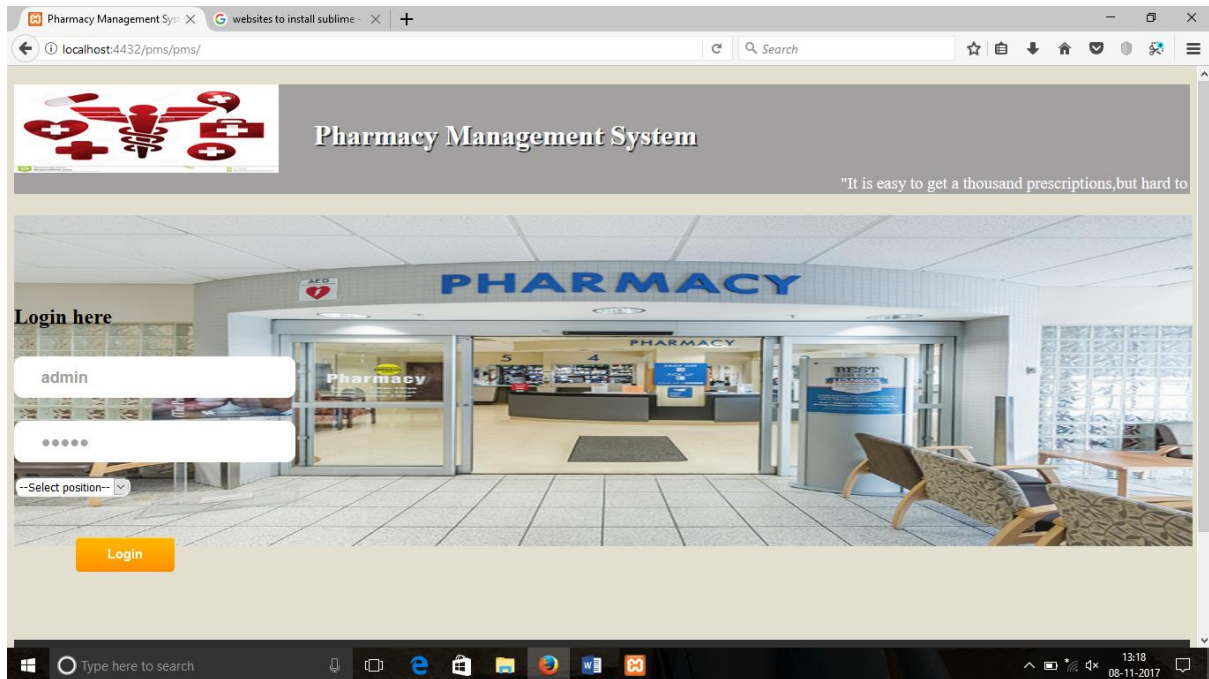
<http://php.net/manual/en/install.windows.php>

for Sublime installation:

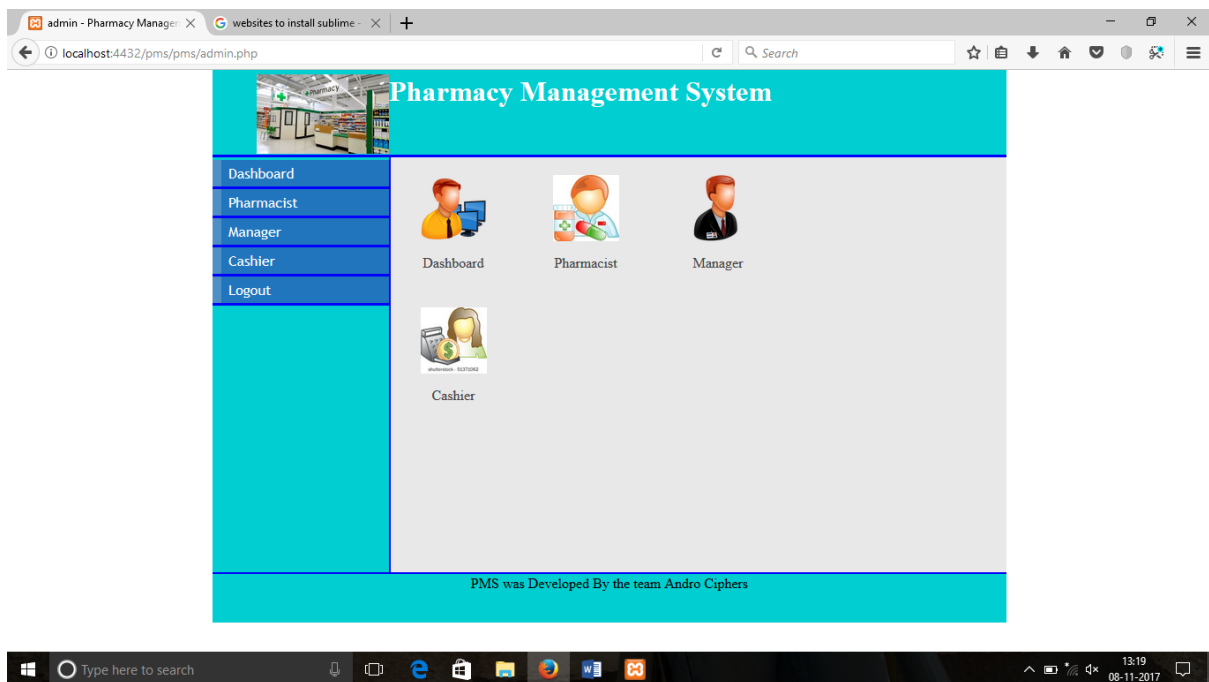
APPENDIX

A.SAMPLE FORM

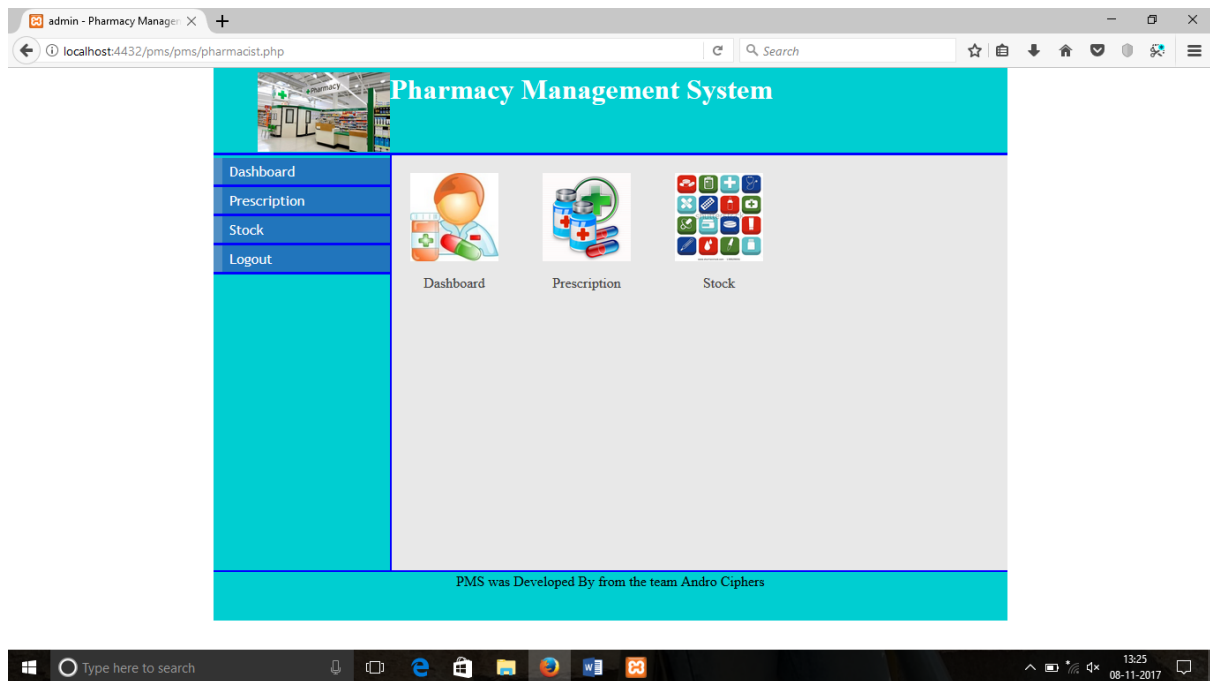
LOGIN PAGE:



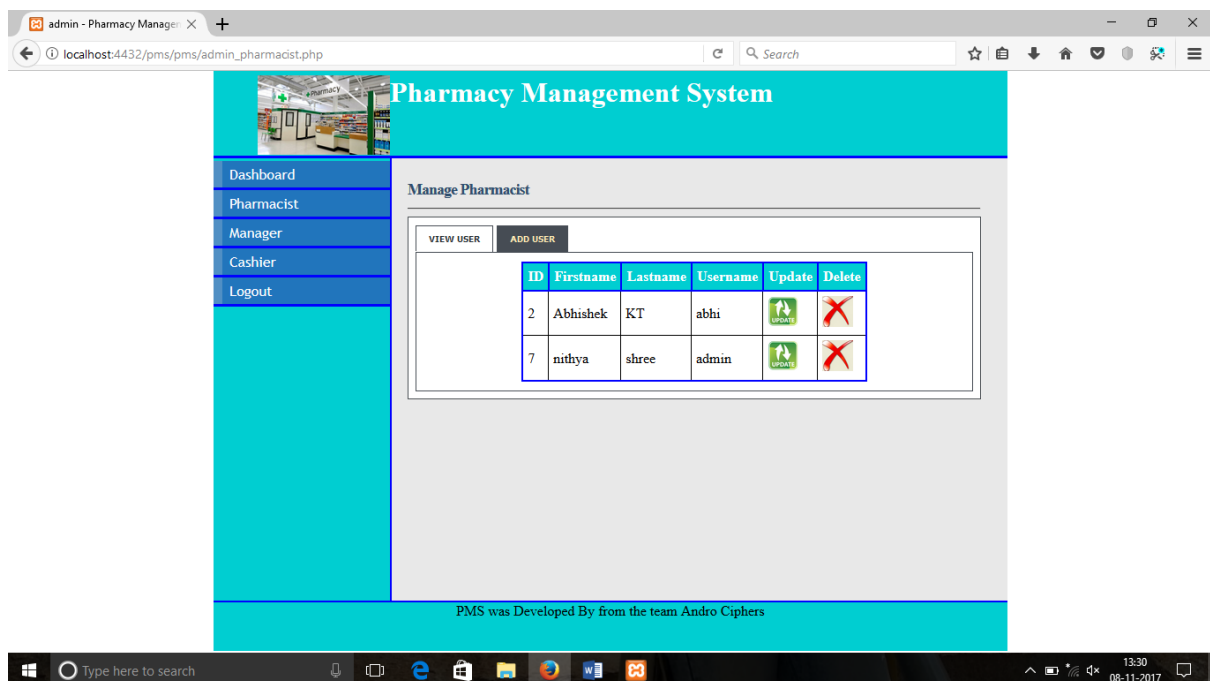
Admin Mainpage:



Manage pharmacist:



Add Pharmacist:



Manage Manager:

Add Manager:

The screenshot displays the 'admin - Pharmacy Manager' web application running on a browser at the URL `localhost:4432/pms/pms/admin_manager.php`. The interface features a teal sidebar with navigation links: Dashboard, Pharmacist, Manager, Cashier, and Logout. The main content area is titled 'Pharmacy Management System' and 'Manage Manager'. It includes a 'VIEW USER' button and an 'ADD USER' button. A table lists the current manager:

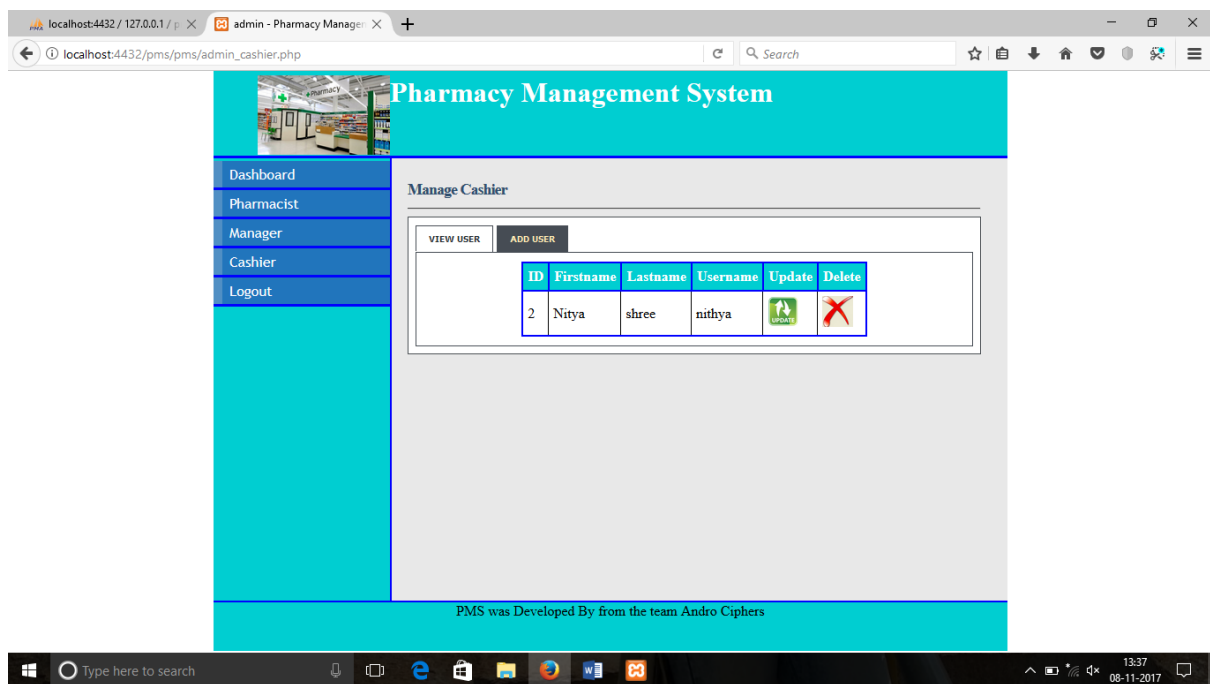
ID	Firstname	Lastname	Username	Update	Delete
1	Vinay	K T	vinay		

At the bottom of the page, a footer states: 'PMS was Developed By from the team Andro Ciphers'. The Windows taskbar at the bottom shows the time as 13:31 on 08-11-2017.

Manage Cashier:

.....

AddCashier:



B. SAMPLE REPORT

View Users:

View prescription:

.....

Manage Stock:

.....

49

Cashier Dashboard:

.....

50

Manage Payments:

.....

C. Sample Coding

Admin Cashier

```
<?php
session_start();
include_once('connect_db.php');
if(isset($_SESSION['username'])){
$id=$_SESSION['admin_id'];
$username=$_SESSION['username'];
}else{
header("location:http://".$_SERVER['HTTP_HOST'].dirname($_SERVER['PHP_SELF'])."/index.php");
exit();
}
if(isset($_POST['submit'])){
$fname=$_POST['first_name'];
if (!preg_match("/^[a-zA-Z ]*$/",$fname))
{
$nameErr = "Only letters and white space allowed";
}
$lname=$_POST['last_name'];
$sid=$_POST['staff_id'];
$postal=$_POST['postal_address'];
$phone=$_POST['phone'];
$email=$_POST['email'];
$user=$_POST['username'];
$pass=$_POST['password'];
$sql1=mysql_query("SELECT * FROM cashier WHERE username='$user'")or
die(mysql_error());
$result=mysql_fetch_array($sql1);

if($result>0){
```

```

$message="<font color=blue>sorry the username entered already exists</font>";
}else{
$sql=mysql_query("INSERT INTO
cashier(first_name,last_name,staff_id,postal_address,phone,email,username,password,date)
VALUES('$fname','$lname','$sid','$postal','$phone','$email','$user','$pas',NOW())");
if($sql>0)
{ header("location:http://".$_SERVER['HTTP_HOST'].dirname($_SERVER['PHP_SELF'])."/
admin_cashier.php");
}else{
$message1="<font color=red>Registration Failed, Try again</font>";
}
}}
?>
<!DOCTYPE html>
<html>
<head>
<title><?php echo $username;?> - Pharmacy</title>
<link rel="stylesheet" type="text/css" href="style/mystyle.css">
<link rel="stylesheet" href="style/style.css" type="text/css" media="screen" />
<link rel="stylesheet" href="style/table.css" type="text/css" media="screen" />
<script src="js/function.js" type="text/javascript"></script>
<script src="js/validation_script.js" type="text/javascript"></script>
<!--<script>
function validateForm()
{

//for alphabet characters only
var str=document.form1.first_name.value;

var valid="abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ";
//comparing user input with the characters one by one
for(i=0;i<str.length;i++)
{

```


//charAt(i) returns the position of character at specific index(i)

//indexOf returns the position of the first occurrence of a specified value in a string. this method returns -1 if the value to search for never occurs

```
if(valid.indexOf(str.charAt(i))!=-1)
{
    alert("First Name Cannot Contain Numerical Values");
    document.form1.first_name.value="";
    document.form1.first_name.focus();
    return false;
}}
```

```
if(document.form1.first_name.value=="")
{
    alert("Name Field is Empty");
    return false;
}
```

//for alphabet characters only

var str=document.form1.last_name.value;

var valid="abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ";

//comparing user input with the characters one by one

for(i=0;i<str.length;i++)

```
{
    //charAt(i) returns the position of character at specific index(i)
```

//indexOf returns the position of the first occurrence of a specified value in a string. this method returns -1 if the value to search for never occurs

```
if(valid.indexOf(str.charAt(i))!=-1)
{
    alert("Last Name Cannot Contain Numerical Values");
    document.form1.last_name.value="";
    document.form1.last_name.focus();
    return false;
}}
```

```
if(document.form1.last_name.value=="")
{
alert("Name Field is Empty");
return false;
}

}
```

```
</script>-->
```

```
<style>
```

```
<style>#left-column { height: 477px;}
```

```
#main { height: 477px;}</style>
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div id="content">
```

```
<div id="header">
```

```
<h1><a href="#"></a> Pharmacy</h1></div>
```

```
<div id="left_column">
```

```
<div id="button">
```

```
<ul>
```

```
<li><a href="admin.php">Dashboard</a></li>
```

```
<li><a href="admin_pharmacist.php">Pharmacist</a></li>
```

```
<li><a href="admin_manager.php">Manager</a></li>
```

```
<li><a href="admin_cashier.php">Cashier</a></li>
```

```
<li><a href="logout.php">Logout</a></li>
```

```
</ul>
```

```
</div>
```

```
</div>
```

```
<div id="main">
```

```
<div id="tabbed_box" class="tabbed_box">
```

```
<h4>Manage Cashier</h4>
```

```
<hr/>
```

```
<div class="tabbed_area">
```

```
<ul class="tabs">
```

```
<li><a href="javascript:tabSwitch('tab_1', 'content_1');" id="tab_1"
class="active">View User</a></li>
```

```
<li><a href="javascript:tabSwitch('tab_2', 'content_2');" id="tab_2">Add
User</a></li>
```

```
</ul>
```

```
<div id="content_1" class="content">
```

```
<?php echo $message;
echo $message1;
```

```
/*
```

```
View
```

```
Displays all data from 'Cashier' table
```

```
*/
```

```
// connect to the database
```

```
include_once('connect_db.php');
```

```
// get results from database
```

```
$result = mysql_query("SELECT * FROM cashier")
or die(mysql_error());
```

```
// display data in table
```

```
echo "<table border='1' cellpadding='5' align='center'>";
```

```

        echo "<tr> <th>ID</th><th>Firstname </th> <th>Lastname </th> <th>Username
</th><th>Update </th><th>Delete</th></tr>";

```

```

// loop through results of database query, displaying them in the table

```

```

while($row = mysql_fetch_array( $result )) {

```

```

    // echo out the contents of each row into a table

```

```

    echo "<tr>";

```

```

    echo '<td>' . $row['cashier_id'] . '</td>';

```

```

    echo '<td>' . $row['first_name'] . '</td>';

```

```

        echo '<td>' . $row['last_name'] . '</td>';

```

```

        echo '<td>' . $row['username'] . '</td>';

```

```

        ?>

```

```

                                <td><a href="update_cashier.php?username=<?php echo
$row['username']?>"></a></td>

```

```

                                <td><a href="delete_cashier.php?cashier_id=<?php echo $row['cashier_id']?>"></a></td>

```

```

                                <?php

```

```

                                }

```

```

// close table>

```

```

echo "</table>";

```

```

?>

```

```

</div>

```

```

<div id="content_2" class="content">

```

```

    <!--Cashier-->

```

```

    <?php echo $message;

```

```

        echo $message1;

```

```

    ?>

```

```

        <form name="form1" onsubmit="return validateForm(validation_script.js);"
action="admin_cashier.php" method="post" >
            <table width="220" height="106" border="0" >
                <tr><td align="center"><input name="first_name" type="text"
style="width:170px" placeholder="First Name" required="required" id="first_name"
/></td></tr>
                <tr><td align="center"><input name="last_name" type="text"
style="width:170px" placeholder="Last Name" required="required" id="last_name"
/></td></tr>
                <tr><td align="center"><input name="staff_id" type="text"
style="width:170px" placeholder="Staff ID" required="required" id="staff_id"/></td></tr>
                <tr><td align="center"><input name="postal_address"
type="text" style="width:170px" placeholder="Address" required="required"
id="postal_address" /></td></tr>
                <tr><td align="center"><input name="phone" type="text"
style="width:170px"placeholder="Phone" required="required" id="phone" /></td></tr>
                <tr><td align="center"><input name="email" type="email"
style="width:170px" placeholder="Email" required="required" id="email" /></td></tr>
                <tr><td align="center"><input name="username" type="text"
style="width:170px" placeholder="Username" required="required" id="username"
/></td></tr>
                <tr><td align="center"><input name="password"
type="password" style="width:170px" placeholder="Password" required="required"
id="password"/></td></tr>
                <tr><td align="right"><input name="submit" type="submit"
value="Submit"></td></tr>
            </table>
        </form>
    </div>

</div>

```

</div>

</div>

<div id="footer" align="Center"> PMS was developed by the team of Andro Ciphers</div>

</div>

</body>

</html>

