COMPONENT MANAGEMENT SYSTEM

A PROJECT REPORT

Submitted by

PURNENDU KUMAR ROY (M120368CA)

In partial fulfilment for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

Under the guidance of

Mr. Dhanaraj K. J

(Assistant Professor, Dept. of Electronics and Communication Engineering)(External)

Dr. Vinod Pathari

(Assistant Professor, Dept. of Computer Science and Engineering)(Internal)

Department of Computer Science and Engineering,
National Institute of Technology, Calicut
NITC Campus PO, Calicut
Kerala, India 673601

JUNE 2015

DECLARATION

"I, hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or the other institute of higher learning, except where due acknowledgement has been made in the text."

Place: NIT Calicut

Date: March 16, 2015

Signature:

Name: Purnendu Kumar Roy

Reg. No.: M120368CA

CERTIFICATE

*This is to certify that the project report entitled: "**Component Management System**". Submitted by **Purnendu Kumar Roy** to the Department of Computer Science and Engineering, National Institute of Technology Calicut towards the partial fulfilment of the requirements for the award of the degree of Master of Computer Applications, is a bona fide record of the work carried out by him under our supervision and guidance.*

(Mr. Dhanaraj K. J) (Dr. Vinod Pathari)

Place:

Date:

Signature of Head of the Department

(Office seal)

ACKNOWLEDGEMENT

I take this opportunity to express my sincere gratitude to all individuals, directly or indirectly, who have contributed towards the completion of this work.

I would like to express my sincere gratitude to my external project guide, **Mr. Dhanaraj K.J**, Assistant professor, Dept. of Electronics and communication engineering for this exemplary guidance, monitoring and constant encouragement throughout the course of this project.

I would like to express my heartfelt gratitude to my internal project guide, **Dr. Vinod Pathari**, Assistant Professor, Dept. of computer science and engineering for their guidance, for this cordial support, valuable information and guidance, which helped me in completing this task through various stages.

I would also like to express my sincere thanks to **Mr. K.P. Venugopal**, Instructor, Dept. of Electronics and Communication Engineering for his support and cooperation throughout this work.

Lastly, I thank all my classmates and friends for their continuous support and encouragement.

Purnendu Kumar Roy

Table of Contents

ABSTRACT

Component **M**anagement **S**ystem (CMS) is a web application to improve the component management in different laboratories of Electronics and Communication Engineering Department of National Institute of technology Calicut. CMS will store all transaction details of component, user details and along with it also provide easy interface to search users and components. User can easily get no dues, as any pending items will be immediately displayed. This system provides GUI based Interface which is very user friendly.

To develop this web application PHP is used at front end with scripting like HTML, jQuery and JavaScript. MySQL database is used for storing data.

1. INTRODUCTION

1.1 Problem Definition

Component Management System (CMS) is a web application which automates the component management in department laboratories of Electronics and Communication Engineering of NITC and to be made available in the departmental website. The CMS site administrator should have the provision to add component, edit/delete/update existing components, search for components, view all components, issue/return of components and check no dues for students. Administrator should also have provision to create new user account, delete/update user's profile, and maintain the database from where they are purchasing components. Student should be able to search components and get the required component issued through the staff-in-charge. Student can request for the purchase of a component if it is not available in the store. CMS will generate report that includes minimum stock alerts, components issued to users/labs, request for components, component status, damaged components, issue no dues and view all capital items.

1.2 Background

The term "CMS" has been applied to file based system (paper based) where Administrator will record the transactions in stock register which is currently done manually which consumes lot of time and results in errors. Students sometimes purchase components from outside which are needed for the project, because they do not have the facility to know the components that are available in the store. This results in duplication in many occasions. There are a lot of components being used in electronics department by users.

There are mainly two types of components:

(a) Capital Items – These are main item which are issued to labs and used for testing. These are very costly items. E.g.: CRO, Multi meter etc.

(b) Consumable Items – These are items which will be used for day to day activities, they are not normally audited. E.g.: Resistor, Capacitor etc.

In CMS there are two actors:

(a) Administrator – Administrator is the store in charge who will manage whole system.

(b) Users- There are three type of users: Faculty, Staff, and Student.

Store-in-charge will record the transactions in a stock register, which is currently done manually and hence it consumes lot of time and may cause error. Student usually buy components from outside for their project. They do not have the facility to know the components that are available in the store and hence it causes duplication in many occasions.

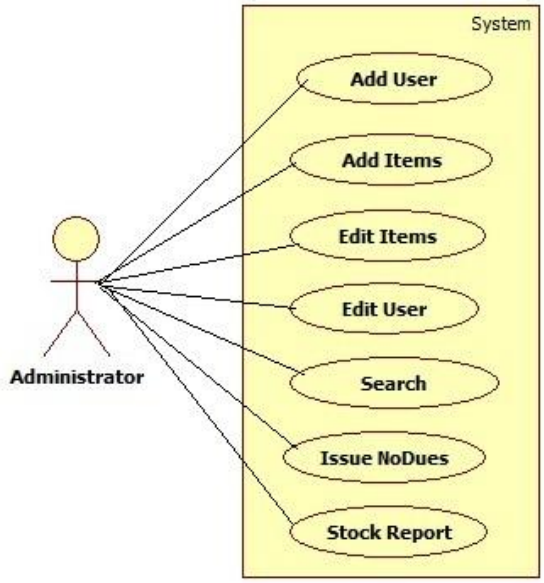
1.3 Motivation

Presently, there is no successful software for the management of components of ECED. This project is intended to bring this facility to the ECED department store for the better management of components. The CMS can benefit users in many ways. First if a student need a component there is no mechanism to search whether the component is available or not. The CMS may also become an improved online stock register where student/staff/faculty can have a better search mechanism and request issues. Potential benefit of CMS is that a student/faculty/staff can easily get no dues, as any pending items will be immediately displayed. The proposed application will help us to handle the stock and transaction details in an easy and better manner. Store-in-charge will handle all these manually. CMS may provide efficient way to deal with stock of components. This will help to reduce the man power, save the time and will be more accurate.

2. SYSTEM DESIGN

2.1 Use case Diagram

- Use case diagram - administrator (store-in-charge)



Brief description: Administrator will have to register first with particular user Id and Password, where the Administrator will maintain the database and he will have the provision to create/delete/update/edit users and components.

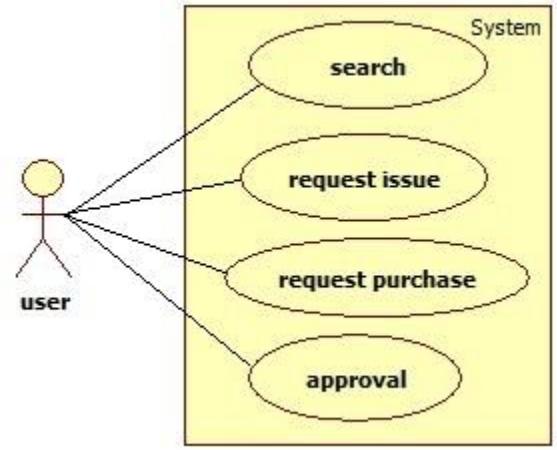
Precondition: Administrator should visit the CMS site and must register himself/herself.

Step by Step: **i)** Store-in-charge will register himself/herself through appropriate user id and password, where the user id must be unique.

ii) he/she will register other users and add/delete/update new components, issue of no dues will also done by the administrator.

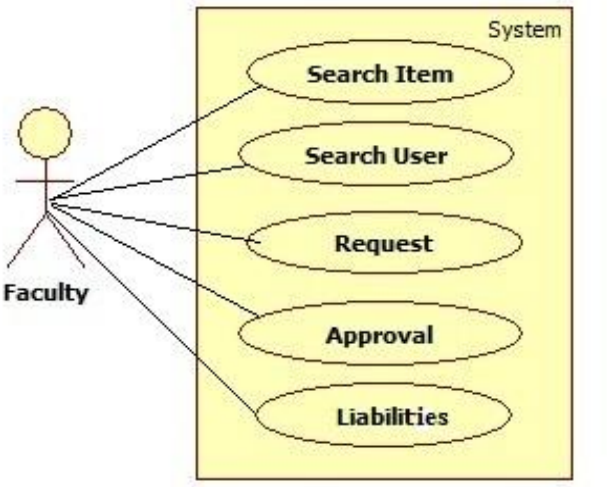
iii) Administrator will logout after completion of work.

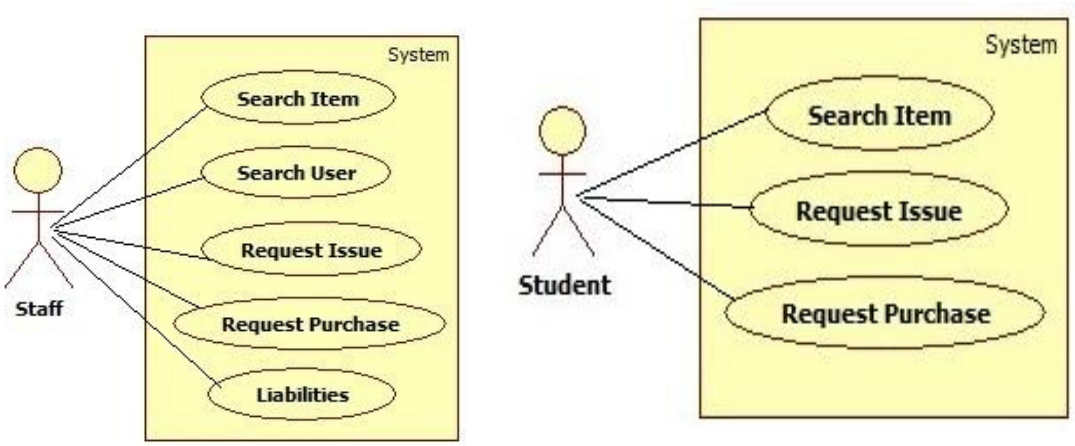
- Use case diagram - Users (Student, staff, and faculty)



Brief Description: Administrator will register Users and then they will access the CMS website. Users navigate to the login page and login with appropriate user Id, password and type of the user.

There are three type of user in CMS: Faculty, Staff and Student. Users have provision to search, request a component.





Precondition: User has logged into his account by giving appropriate user id, password and type.

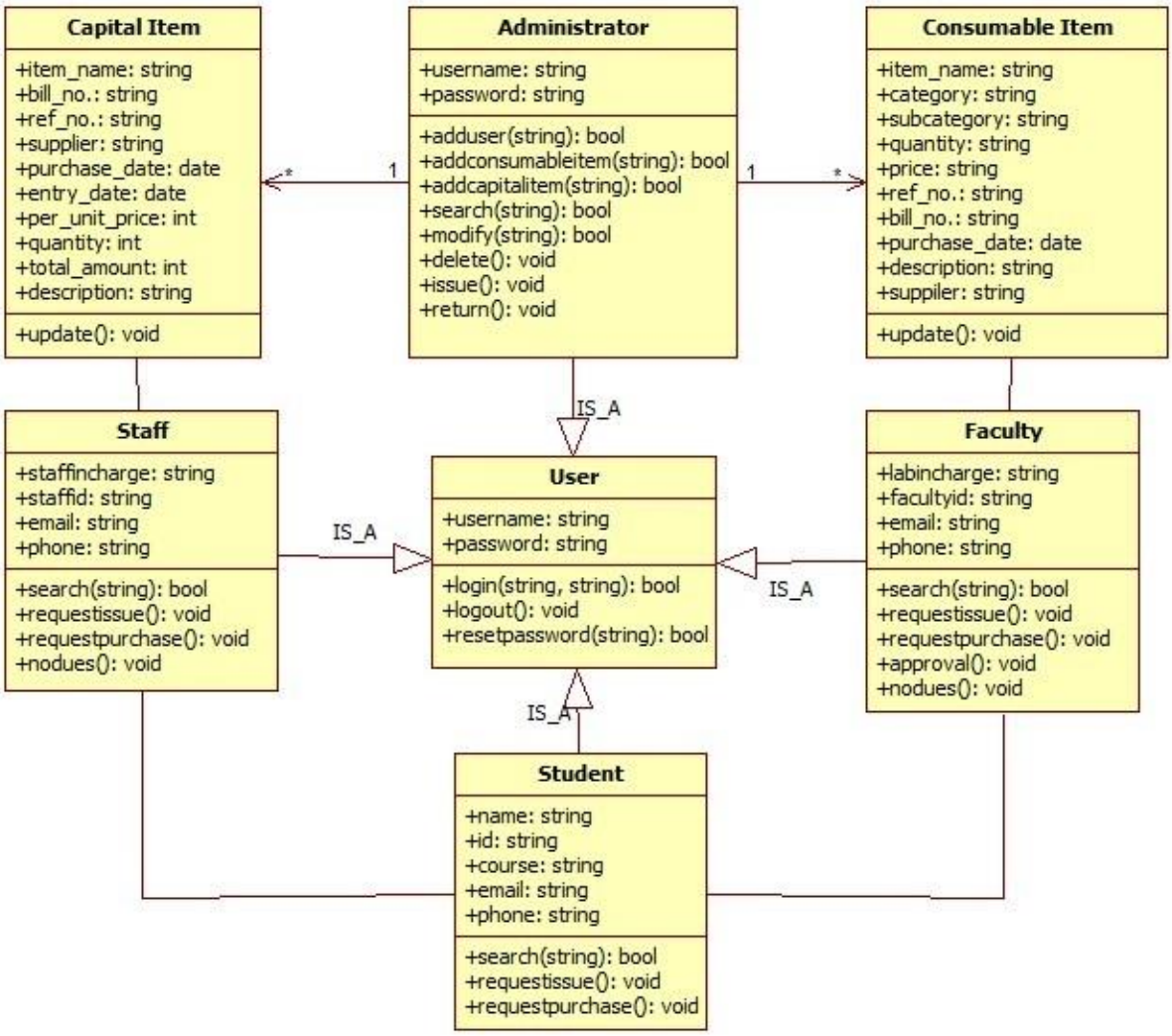
Step-By-Step: i) Faculty, student and staff can search an item. Student or Staff will request for an item.

ii) Store Keeper will approve the request and after that Faculty will have requested component and then student will take component from staff after approval from faculty.

iii) Faculty will also issue no dues for student whether he/she has any due or not.

iv) Faculty will exit after completion of work.

2.2 Class Diagram



Noun phrases: **i)** Administrator

ii) Faculty

iii) Staff

iv) Student

v) Capital items

vi) Consumable items

Here in this system there could be three types of users: Student, Faculty and staff. These entities share many of the properties and methods. So, we defined new class that is user and from this class student, faculty and staff inherits properties and methods. Hence user is basically an abstract class whose object directly can't be created. Administrator will be able to add, delete and update users, capital and consumable items. Users can search for the capital and consumable items. User can request issue and purchase for the component. If a request is made by student or staff, it should be approved by corresponding faculty.

2.3 System features

2.3.1 Add component/users

Case name	Add
Priority	High
Trigger use	Admin page
Pre-condition	Admin is logged in
Basic path :Index	Admin enter the details of component/user.
Post condition	The information will be stored in database.

2.3.2 Login page

Case name	Login
Priority	High
Trigger use	Home page
Precondition	User is logged in already
Basic path: login	User will be asked to enter the username and password and user type. If authenticated he will be directed back to the current page with access more service.
Alternate path: invalid username/password	If the username/password is not authenticated the user will be notified and will be asked to enter again.

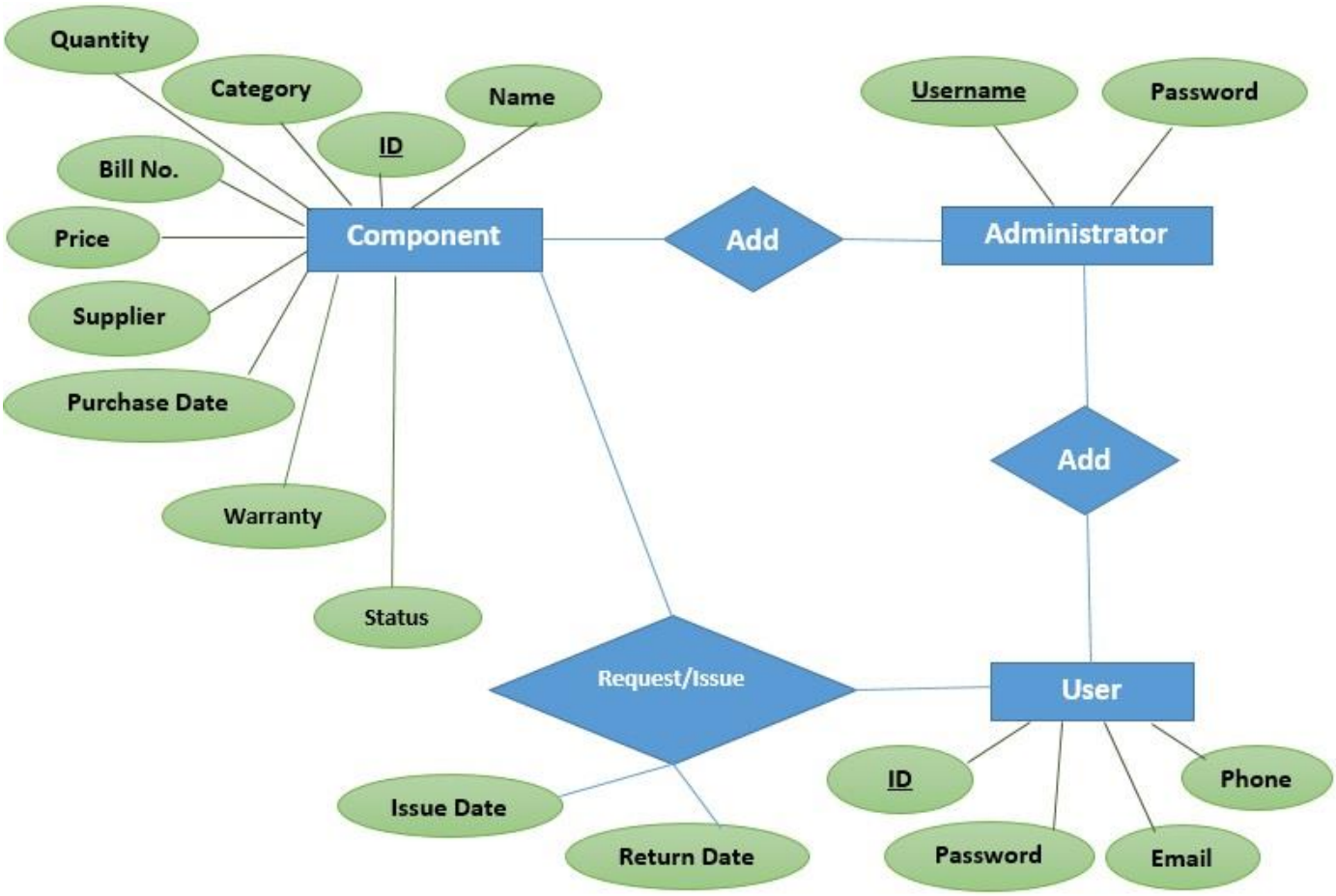
2.3.3 Search

case name	Search
Priority	High
Trigger use	Search box
Precondition	User is already logged in
Basic path: index	Gets activated when user hits the return key after entering some text in the search box.
Post condition	It will return a list of all data matching the search query, and user will have figure out the best result.

2.3.4 Issue

case name	Issue
Priority	High
Trigger user	Issue page
Precondition	Admin should be already logged in
Basic path : index	Admin enters the user details and issue the requested component with return date.
Post condition	This information will be stored in database.

2.4 E-R Diagram

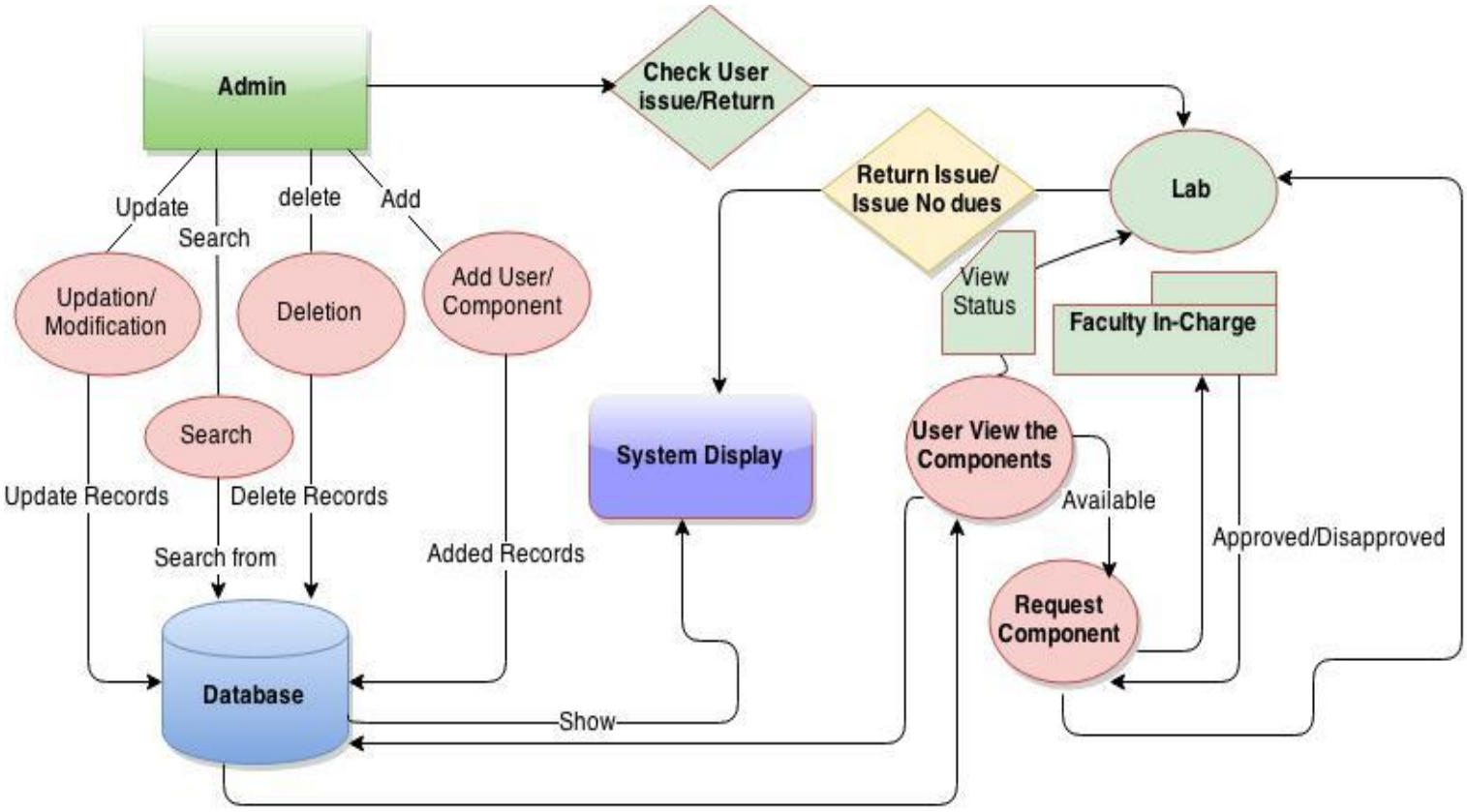


2.5 Database Design

There are 11 tables. Table Structures are given in Appendix-1

No.	Table name	Primary key
1	Capital	id
2	Consumable	id
3	Faculty	id
4	Staff	id
5	Student	roll
6	lab_det	lab, fac_id, staff_id
7	Lab_issue_consumable	id
8	Request_issue	id
9	User_issue_consumable	id
10	Consum_details	id
11	Discussion	qno

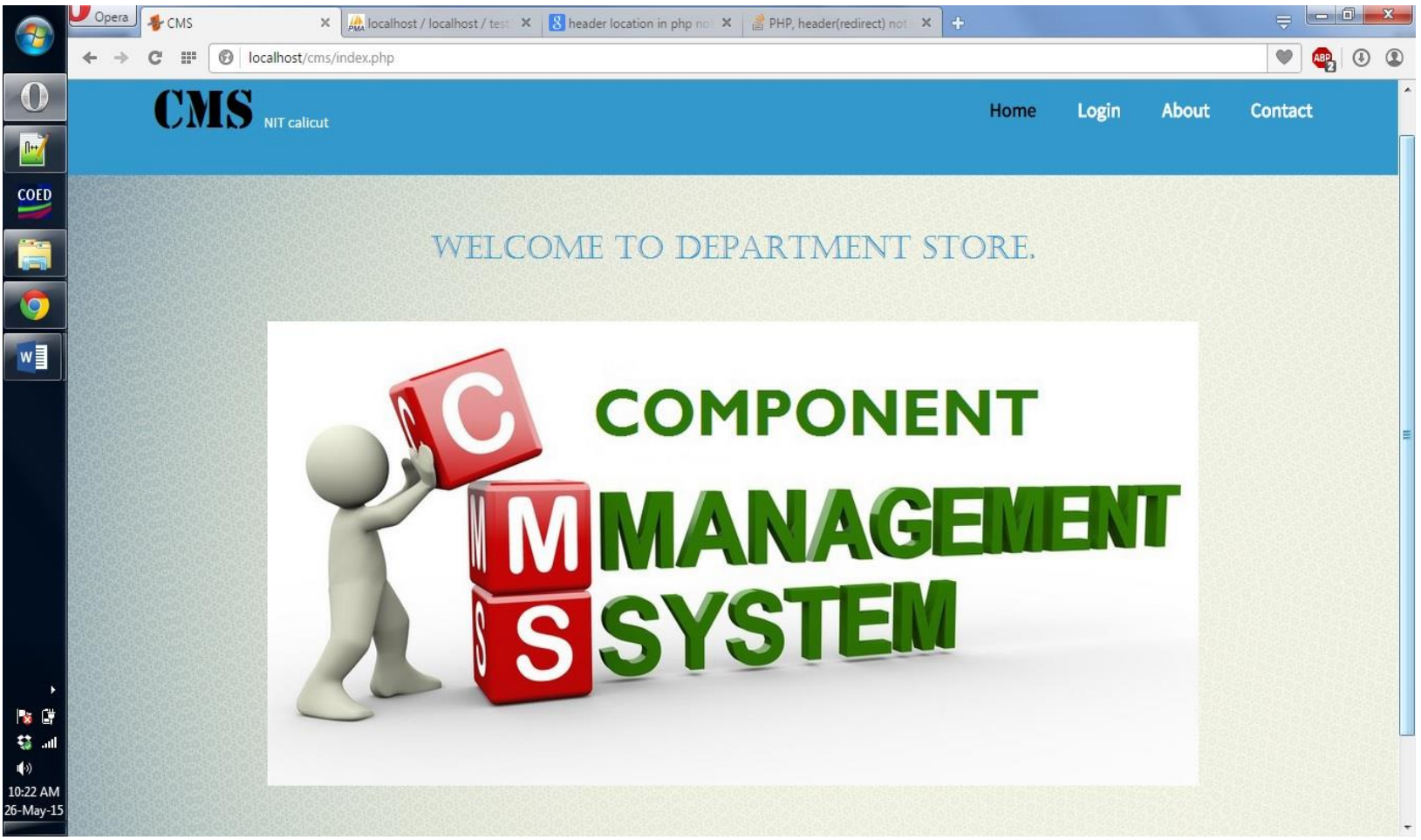
2.6 Data Flow Diagram



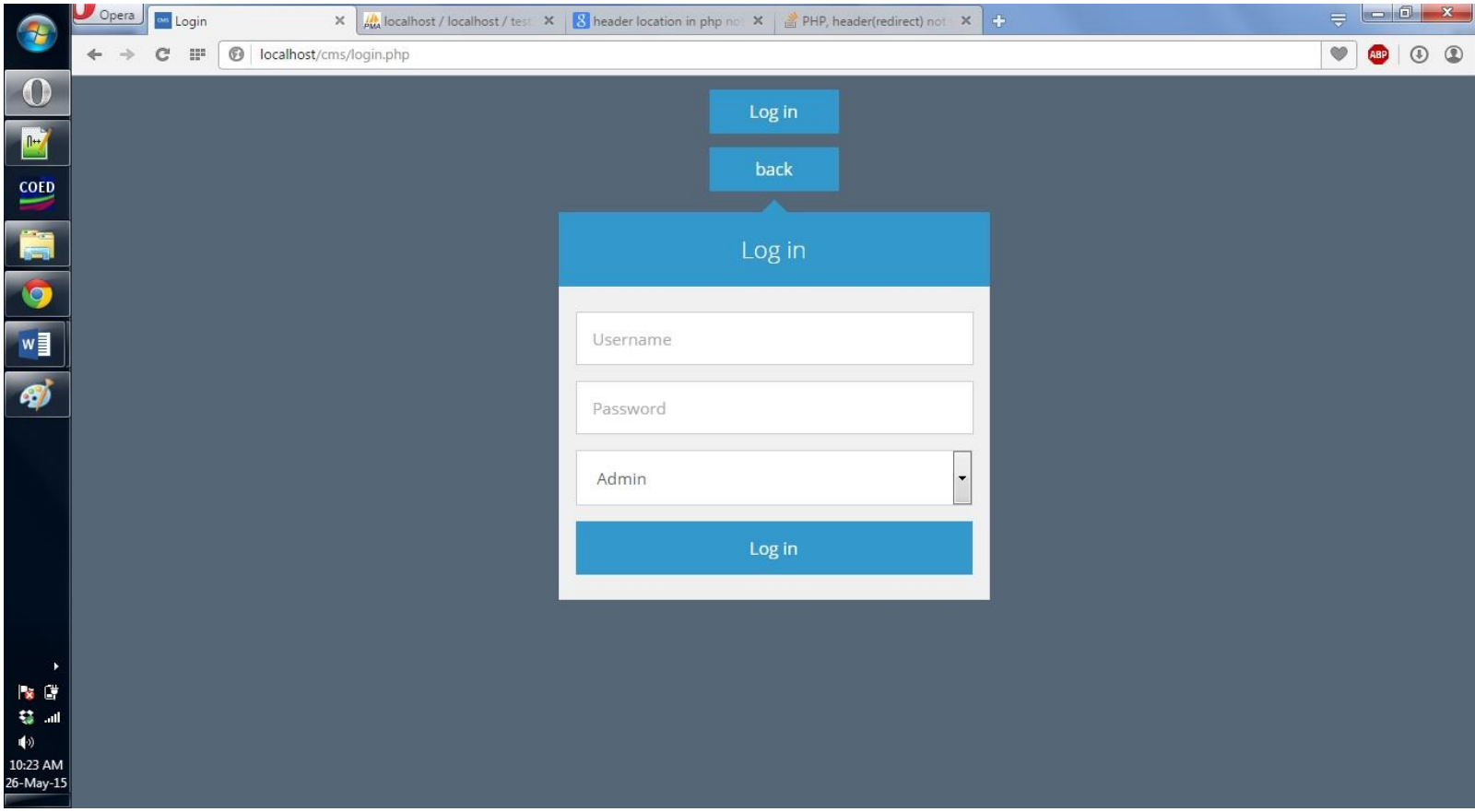
3. USER INTERFACE

3.1 Home Page

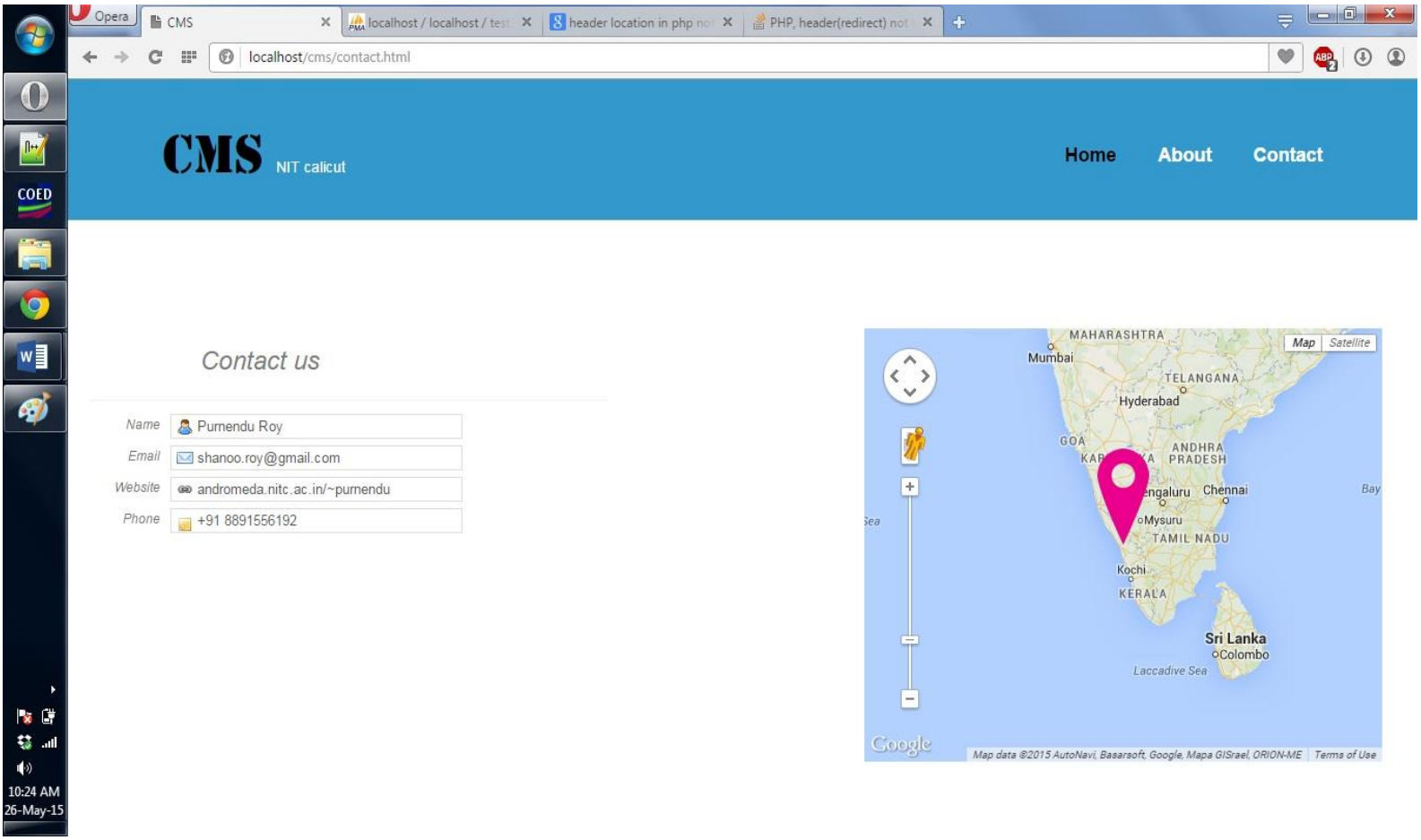
This will the CMS home page which is first page for the users. The home page is used to facilitate navigation to other pages on the site, by providing links to login page, about and Contact page for the users.



3.2 Login Page



3.3 Contact Page



3.4 Admin page

This page includes all the works of the admin.

1. Add component:
Adding the details of the components such as name, category, description, price, purchase date, bill no., and ref.no.

2. Add user:
Adding the details of the student, faculty, and staff. Initially User id will be used as both username and password. Users can change their password.

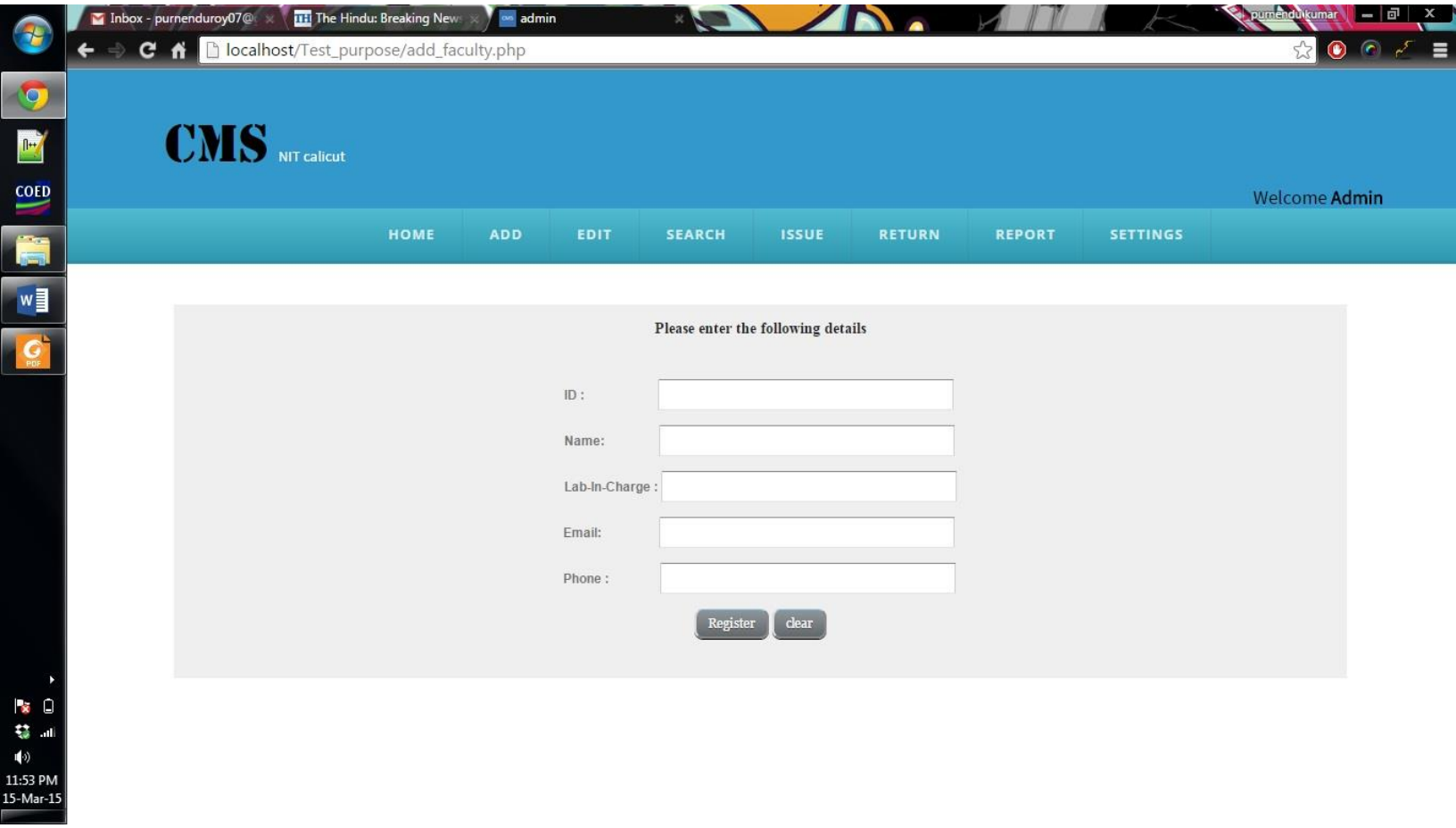
3. Search for a user or component:
Admin can search for registered user and components.

4. View all components and transaction:
There is a provision to view the details of all components that are available in the store, component issued to student, faculty and lab.

5. Update the user and component details:
admin can update the details of components and users.

6. Issue the components to users:
The page will display the components issue request made by the user. The page will show roll number (for students), faculty id (for faculty) and staff id for particular lab. Administrator can issue all or selected components to the user with return date (Only in the case of student and faculty). There is no return if the component is issued to lab. Administrator can edit the quantity of a particular component that is requested for issue by a user.

7. Return issued components from users:
The page will display roll number (for students) and Id (for faculty). It will display the components that are issued to the corresponding user. Administrator can mark the return of all or selected components.



8. Delete user and components:

Admin can delete the details of consumable components, students, faculties and staff.

9. Issue no dues:

The page will ask for unique id for users. It will display whether the user has any dues or not.

3.5 Faculty page

Faculty will have the following functionalities.

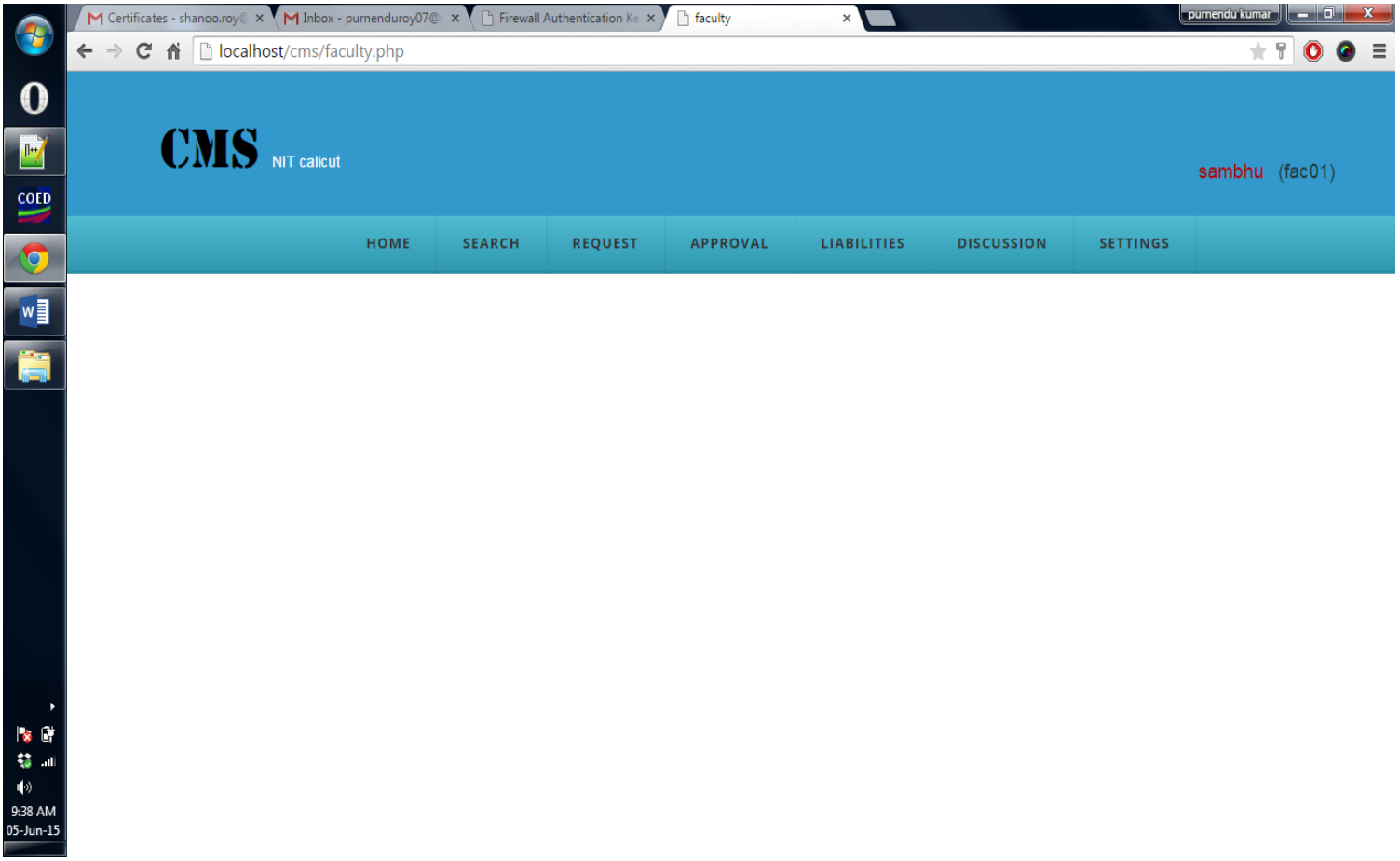
1. Search:
Faculty can search for the components that is available in the store. They can also search for the users.

2. Request for issue:
if a particular components is needed, then they can make a request for the issue of a particular component or components.

3. Request for purchase:
If a particular component is not available in the store, then faculty can make a request for the purchase of components. Administrator can see the purchase requests and purchase that item.

4. Approval:
If a student or lab staff made a component issue or purchase requests, then the corresponding faculty can see the requests and they can approve or disapprove the requests. Here the faculty can edit the quantity of a particular component also.

5. Liabilities:
Faculty can view the components issued to student/staff and the return date of the components, liabilities of those students whose requests were approved by admin.


Certificates - shanoo.roy@
Inbox - purnenduroy07@
Firewall Authentication Ke
faculty
purnendu kumar
localhost/cms/faculty.php
CMS
NIT calicut
sambhu (fac01)
HOME
SEARCH
REQUEST
APPROVAL
LIABILITIES
DISCUSSION
SETTINGS
COED
9:38 AM
05-Jun-15

3.6 Staff page

Staff page will have following functionalities.

1. Search Components:
They can search for a component that is available in the store or not.

2. Request for issue:
If a particular components is needed, then staff can make a request for the issue particular component. The request should be approved by the guide for students and faculty in charge for lab staff. Once the request is approved by faculty, student/staff can get the component from the store.

3. Request for purchase:
If a particular component is not available in the store, then student / staff can make a request for the purchase of components. The request should be approved by guide for student and faculty in charge for lab staff. Once the request is approved by faculty, administrator c the purchase requests and purchase that item.

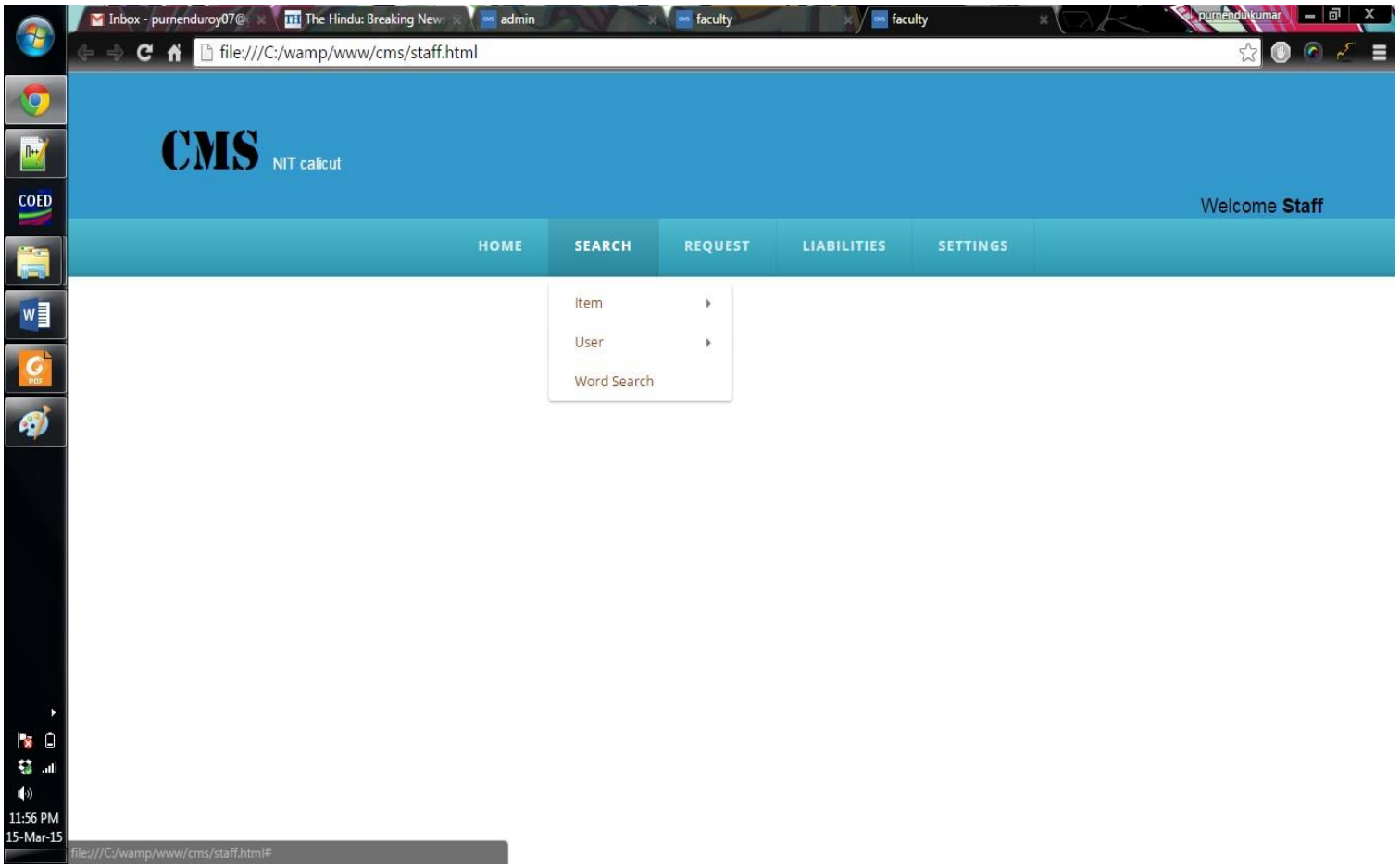
4. View all components:
There is a provision to view the details of all components that are available in the store, component issued to student and particular lab.

5. Update the user and component details:
Staff can update the details of components for the particular lab and users.

6. Issue the components to student:
The page will display the components issue request made by the student. The page will show roll number (for students), faculty id (for faculty) and staff id for particular lab. Staff can issue particular labs available components to the user with return date. There is no return if the component is issued to lab. Staff can edit the quantity of a particular component that is requested for issue by a student.

7. Liabilities:
Staff can view the components issued to student and the return date of the components.

8. Issue no dues:
The page will ask for unique id for student. It will display whether the user has any dues or not.


Inbox - purnenduroy07@
The Hindu: Breaking New
admin
faculty
faculty
purnendukumar
file:///C:/wamp/www/cms/staff.html
CMS
NIT calicut
Welcome Staff
HOME
SEARCH
REQUEST
LIABILITIES
SETTINGS
Item
User
Word Search
11:56 PM
15-Mar-15
file:///C:/wamp/www/cms/staff.html#

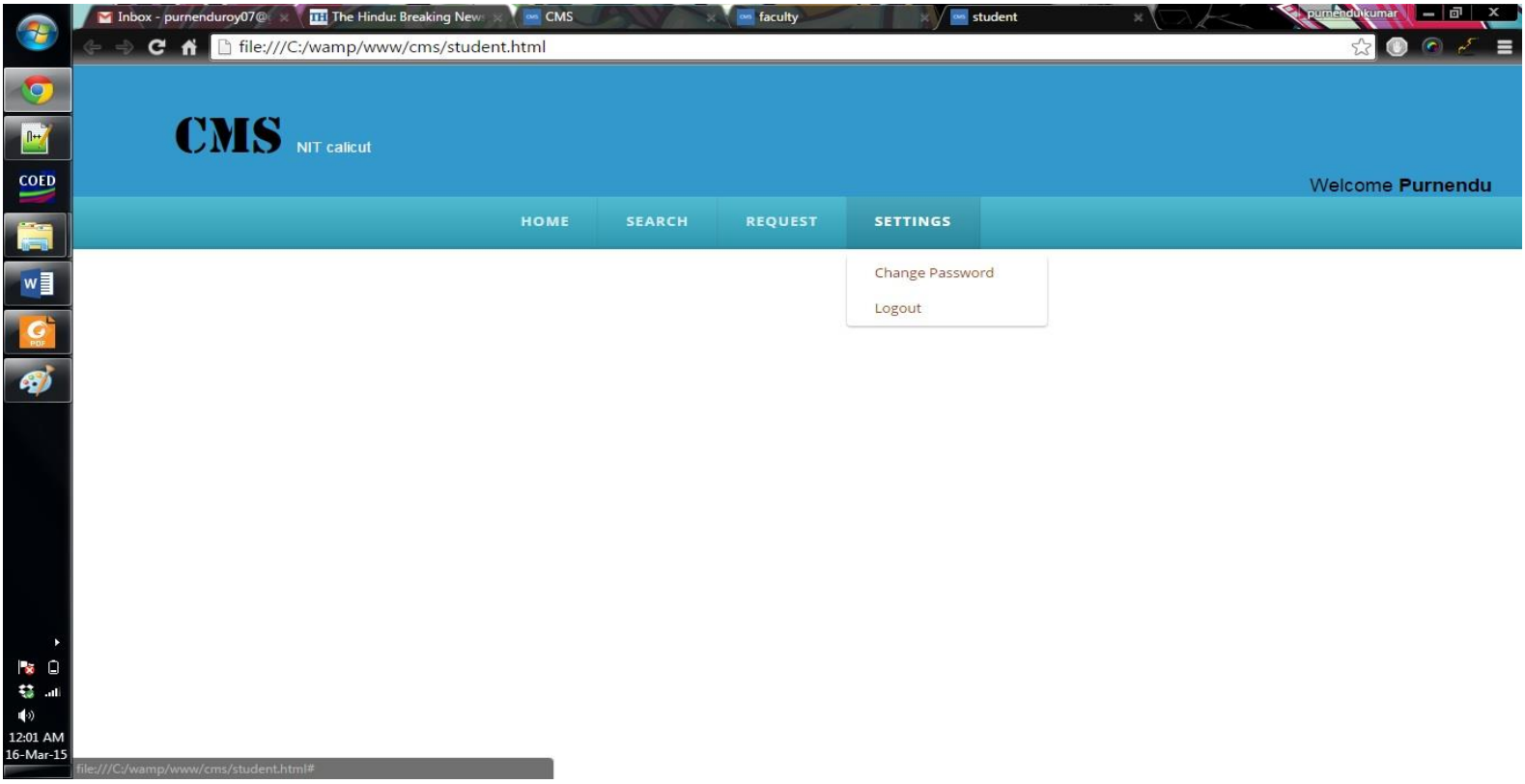
3.7 Student page

Student page will have following functionalities.

1. Search Components:
They can search for a component that is available in the store or not.

2. Request for issue:
If a particular components is needed, then student can make a request for the particular component. The request should be approved by the faculty. Once the request is approved by faculty, student can get the component from the particular lab where he/she has requested through the consent of lab staff.

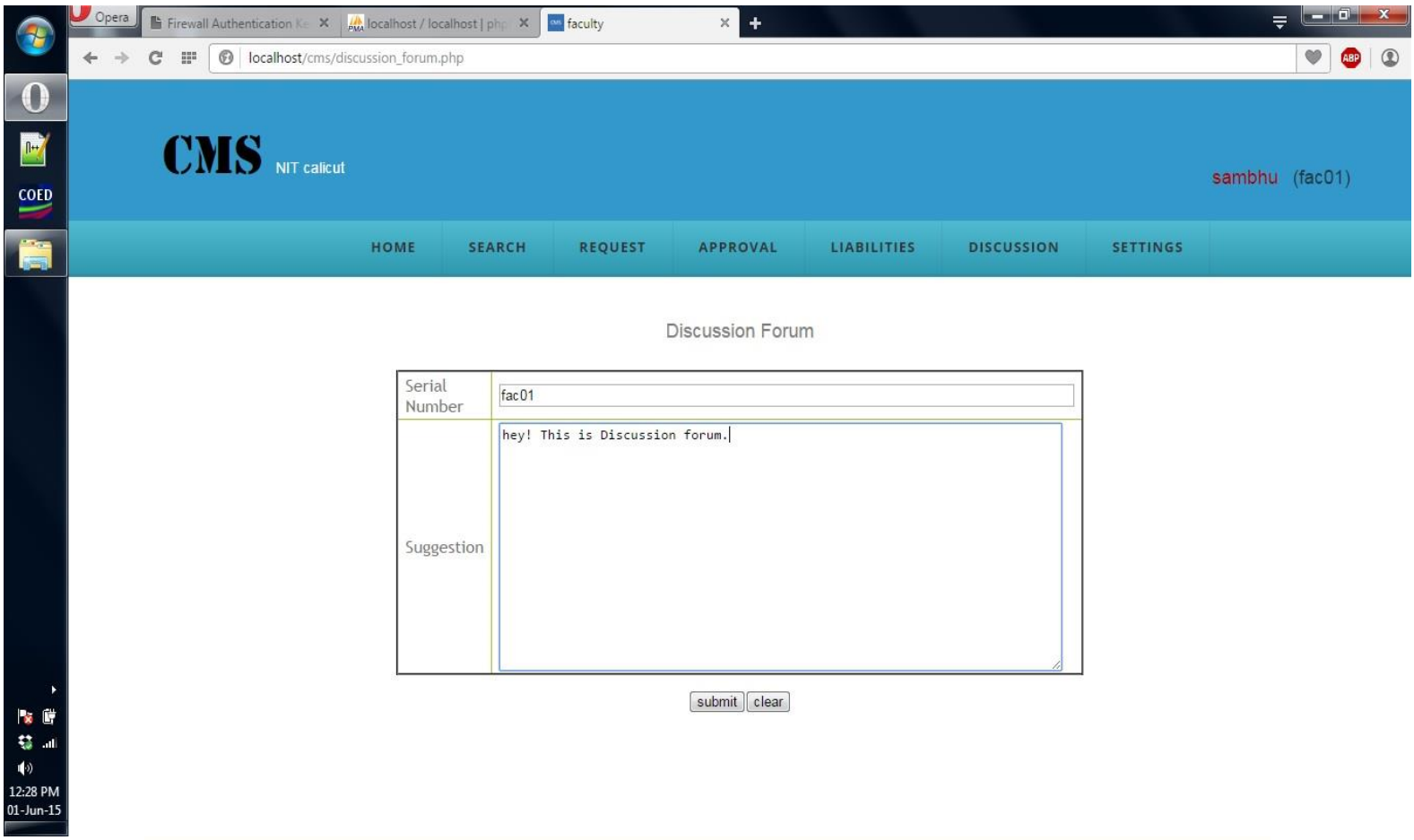
3. Request for purchase:
If a particular component is not available in the particular lab, then student can make a request for the purchase of components. The request should be approved by guide for student and faculty in charge for student.



3.8 Discussion Forum

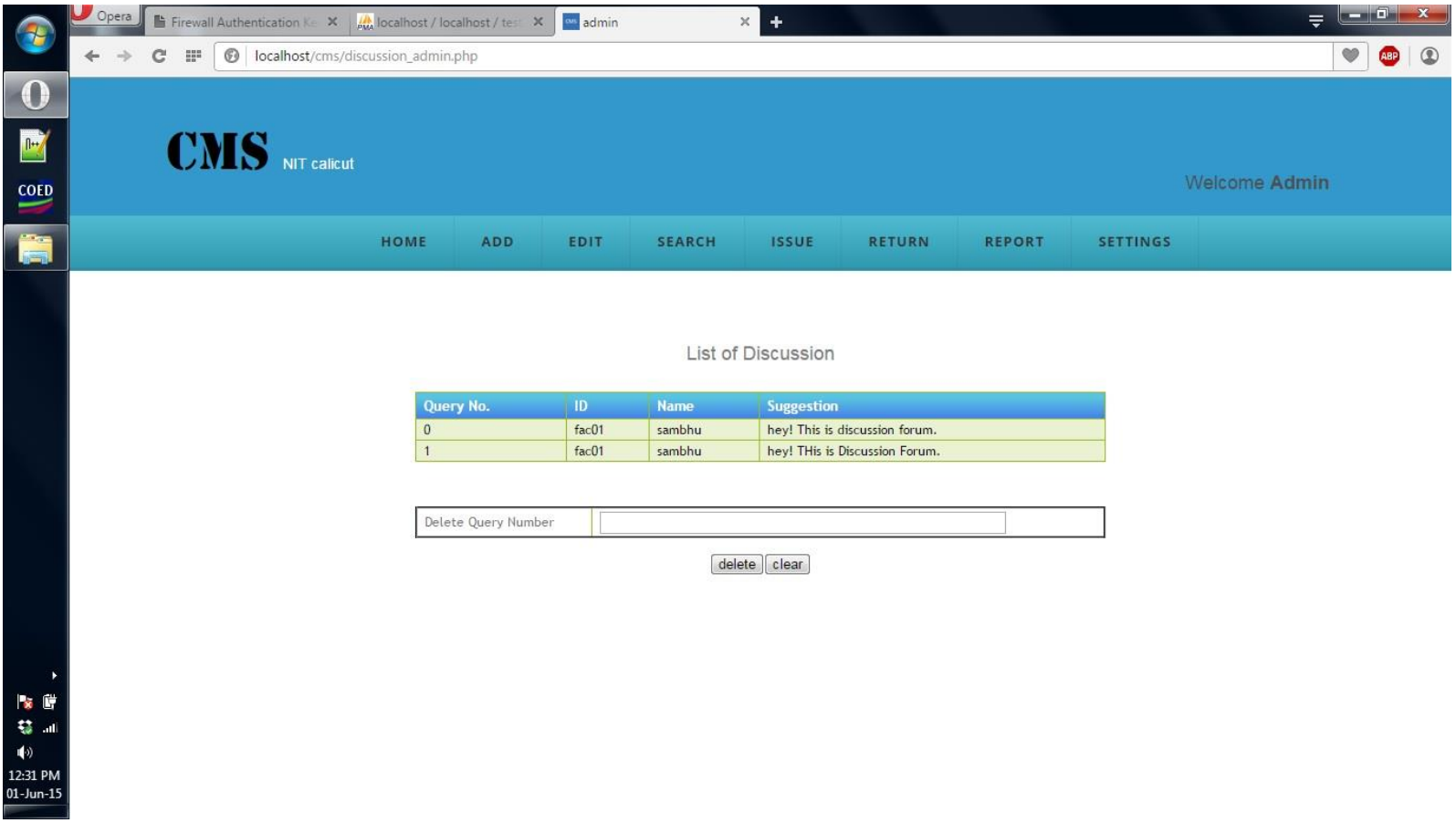
3.8.1 Users Discussion section

This page allows users to discuss with the administrator for any query.



3.8.2 Administrator discussion section

This page will show the list of discussion along with delete section where admin can delete any query from list of discussion.



4. SUMMARY OF WORK

4.1 Selected studies

The concept of Component Management System is not new. A similar project was already designed and implemented to address the component management system. The existing system has developed CMS in PHP language, HTML/CSS, JAVA SCRIPT, JQUERY and MYSQL as a database. And this system used Wamp server and it is a windows web development environment, it allows to create web application with PHP and MYSQL database. Alongside, PhpMyAdmin allows to manage easily the database. This approach builds on the familiarity among PHP programmers with standalone web containers such as Apache tomcat, which is the reference implementation.

4.2 Work Done

1. Understood the concept of the component management system.
2. Created login system for different types of user for component Management System.
3. Created all of the part of administrator for component Management System.
4. Created issue and return feature in CMS.
5. Created all of the part of faculty page such as search, request issue, approval issue etc.
6. Created staff section's features such as request issue, request purchase, issue request, return Issue, liabilities etc.
7. Created all the part of student page such as search, request issue, request purchase, view status, liabilities etc.
8. Created the 'change password' facility for the users and administrator.
9. Created contact page for users where users can easily get the location of department.
10. Created discussion forum for users and administrator.

5. CONCLUSION

Component Management System improves the management of components in the department store of Electronics and Communication Engineering Department. Users of the system can check the availability of the components without going to the store. This application helps to handle the stock and its transactions in an easy and efficient manner. This will helps to reduce the man power, saves time and will be more accurate. This helps for more efficient and error free component management.

REFERENCES

[1]. Timothy C. *Object Oriented Software Engineering*, Tata McGraw-Hill-2005 Edition04.

[2]. Navathe, Elmasri *Fundamentals of Database Systems*, Pearson-2013 Edition06.

[3]. Stock register of *Electronics and Communication Engineering* Department year-2015.

[4]. Report of Component Management System by Ramsy Razack year-2013.

[5]. http://www.virtualsplat.com/software-solution/store-management-software.asp Accessed on Jan 2015.

[6]. "IEEE P1233", Draft Guide to Developing System Requirements Specifications. 1993. http://www.cse.msu.edu/~chengb/RE-491/Papers/IEEE-SRS-practice.pdf Accessed on Feb 2015.

[7]. Jay Greenspan, David Wall, MySQL / PHP Database Applications, 2nd Edition Oct. 2003.

[8]. http://www.tutorialspoint.com/mysql/mysql-using-joins.htm Accessed on Jan 2015.

[9]. http://w3schools.invisionzone.com/index.php?showtopic=47997 Accessed on Feb 2015.

[10]. http://www.w3schools.com/php/default.asp Accessed on Feb 2015.

[11].http://stackoverflow.com/questions/16358361/php-headerredirect-not-working-on-live-server Accessed on May 2015.

Appendix-1

capital

Column	Type	Null
id	int(10)	No
name	varchar(50)	No
category	varchar(50)	No
description	varchar(100)	No
supplier	varchar(50)	No
unitprice	bigint(10)	No
bno	varchar(50)	No
refno	varchar(50)	No
manufacturer	varchar(50)	No
warranty	varchar(30)	No
lab	varchar(50)	No
pdate	date	No
edate	date	No
quantity	bigint(10)	No
tamount	bigint(10)	No
status	varchar(20)	No
remarks	varchar(100)	Yes

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	1	A	No	
bno	BTREE	Yes	No	bno	1	A	No	

consumable

Column	Type	Null
id	int(10)	No
type	varchar(30)	No
category	varchar(30)	Yes
subcat	varchar(30)	Yes
description	varchar(100)	Yes
quantity	bigint(10)	No
price	bigint(10)	No
pdate	date	No
edate	date	No
billno	varchar(50)	No
supplier	varchar(30)	Yes
manufacturer	varchar(30)	No
refno	varchar(50)	Yes
alert	bigint(10)	No
specid	int(10)	No
cq	bigint(10)	No
f1	varchar(50)	Yes
f2	varchar(50)	Yes
f3	varchar(50)	Yes
f4	varchar(50)	Yes
f5	varchar(50)	Yes
f6	varchar(50)	Yes
f7	varchar(50)	Yes
f8	varchar(50)	Yes
f9	varchar(50)	Yes
f10	varchar(50)	Yes

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	3	A	No	
specid	BTREE	No	No	specid	3	A	No	

faculty

Column	Type	Null
id	varchar(20)	No
name	varchar(30)	No
lab	varchar(50)	No
email	varchar(50)	No
phone	varchar(10)	No

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	4	A	No	

staff

Column	Type	Null
id	varchar(20)	No
name	varchar(20)	No
lab	varchar(50)	No
email	varchar(30)	No
phone	varchar(10)	No

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	4	A	No	

student

Column	Type	Null
roll	varchar(20)	No
name	varchar(30)	No
course	varchar(20)	No
email	varchar(50)	No
phone	varchar(10)	No

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	roll	5	A	No	

lab_det

Column	Type	Null
lab	varchar(50)	No
fac_id	varchar(50)	No
fac_in	varchar(50)	No
staff_id	varchar(50)	No
staff_in	varchar(50)	No

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	lab	4	A	No	
				fac_id	4	A	No	
				staff_id	4	A	No	
lab	BTREE	Yes	No	lab	4	A	No	
fac_id	BTREE	Yes	No	fac_id	4	A	No	

lab_issue_consumable

Column	Type	Null
id	int(10)	No
cid	int(10)	No
uid	varchar(50)	No
quantity	varchar(50)	No
idate	varchar(30)	No

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	

request_issue

Column	Type	Null
id	int(10)	No
uid	varchar(50)	No
utype	varchar(50)	No
cid	int(10)	No
quantity	bigint(10)	No
pname	varchar(50)	No
gname	varchar(50)	No
rdate	varchar(20)	No
status	varchar(20)	No

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	

user_issue_consumable

Column	Type	Null
id	int(10)	No
uid	varchar(50)	No
utype	varchar(50)	No
guide	varchar(50)	No
cid	int(10)	No
quantity	bigint(10)	No
idate	varchar(20)	No
rdate	varchar(20)	No

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	

consum_details

Column	Type	Null	Defaul
id	int(10)	No	
cid	int(10)	No	
price	bigint(15)	No	
quantity	bigint(15)	No	
pdate	date	No	
manufac	varchar(50)	No	
refno	varchar(50)	No	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null
PRIMARY	BTREE	Yes	No	id	3	A	No
cid	BTREE	No	No	cid	3	A	No

discussion

Column	Type	Null
qno	int(10)	No
id	varchar(20)	No
name	varchar(50)	No
suggestion	varchar(300)	No

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	qno	0	A	No	