



WOLKITE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF INFORMATION TECHNOLOGY

FINAL PROJECT DOCUMENTATION

PROJECT TITLE: **COMPUTER LABORATORY MONITORING AND
MANAGEMENT SYSTEM FOR COLLEGE OF INFORMATICS**

Prepared By:-

<u>Name</u>	<u>ID</u>
1. Anwar Esliman.....	CIR/170/10
2. Markos Mengistu.....	CIR/202/10
3. Merihun Massebo.....	CIR/207/10

ADVISOR: MR. MOLALIGN CHECKLE

Wolkite University, Ethiopia

May 31, 2021

WOLKITE UNIVERSITY
COLLEGE OF COMPUTING AND INFORMATICS
DEPARTMENT OF INFORMATION TECHNOLOGY
COMPUTER LABORATORY MONITORING AND
MANAGEMENT SYSTEM FOR COLLEGE OF
INFORMATICS

SUBMITTED TO THE DEPARTMENT OF INFORMATION TECHNOLOGY OF
WOLKITE UNIVERSITY IN PARTIAL FULFILLMENT OF REQUIREMENTS FOR
THE DEGREE OF BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY

By:-

<u>Name</u>	<u>ID</u>
1. Anwar Esliman.....	CIR/170/10
2. Markos Mengistu.....	CIR/202/10
3. Merihun Massebo.....	CIR/207/10

ADVISOR: MR. MOLALIGN CHECKLE

Wolkite University, Ethiopia

May 31, 2021

DECLARATION

This is to declare that this project work is done under the supervision of mr. Molalign Checkle and having the title computer laboratory monitoring and management system for the college of informatics is the sole contribution of Anwar Esliman, Markos Mengistu and, Merihun Messbo. No part of the project work has been reproduced illegally (copy and paste) which can be considered plagiarism. All referenced parts have been used to argue the idea and have been cited properly. We will be responsible and liable for any consequence if a violation of this declaration is proven.

Date: _____

Group Members

No	NAME	Signature
1	Anwar Esliman	
2	Markos Mengistu	
3	Merihun Messbo	

APPROVAL FORM

This is to confirm that the project computer laboratory monitoring and management system for the college of informatics was submitted to Wolkite University, College of Computing and Informatics Department of information technology by Anwar Esliman, Markos Mengistu and Merihun Messbo is approved for submission.

Advisor Name	Signature	Date
-----	-----	-----
Department Head Name	Signature	Date
-----	-----	-----
Examiner 1 Name	Signature	Date
-----	-----	-----
Examiner 2 Name	Signature	Date
-----	-----	-----
Examiner 3 Name	Signature	Date
-----	-----	-----

ACKNOWLEDGMENT

First and foremost and above all our biggest thanks would be to Almighty GOD because nothing could be possible without his free will and the completion of this project is supported by him. Secondary we would like to thank the Excellency of our advisor MS. Molalign for she's a restless edition of our documentation, input to the quality of this document, heart full guidance, valuable advice, and providing to execute this project. Thirdly we would like to thank Wolkite University College of Informatics dean Mr. Korbza., Information technology department head Mr. Amanuel. And other College of Informatics all departments full willingness of interview, patience in answering to our numerous questions, giving documents and reading materials that help us to precede our project. Finally, we are developing a Computer laboratory monitoring and management system project not only for marks but also to solve the problems of computer laboratory monitoring and management systems of Wolkite University.

TABLE OF CONTENTS

ACKNOWLEDGMENT.....	iii
List of Table.....	viii
List of figure	ix
ABBREVIATION OR ACRONYM.....	ix
ABSTRACT.....	x
CHAPTER ONE	1
1. Introduction.....	1
1.1 Organizational Background	1
1.2 Motivations.....	2
1.3 Statement of the problem	2
1.4 Objective of the project.....	3
1.4.1 General objectives	3
1.4.2 Specific objective	3
1.5 Feasibility Study of the new system.....	4
1.5.1 Operational feasibility	4
1.5.2 Technical feasibility	4
1.5.3 Economic feasibility	4
1.5.4. Legal feasibility	5
1.6 Scope and limitation of the project	5
1.6.1 Scope of the project	5
1.6.2 Limitation	6
1.7 Significance of the project.....	6
1.8 Beneficiaries of the Project	6
1.9 Data Collection Methodology	7
1.9.1 Data sources.....	7
1.9.2 Data Collection Method.....	7
1.9.4 System Analysis and Design Techniques	9
1.9.4.1 Object Oriented Analysis (OOA)	9
1.9.4.2 Object Oriented Design (OOD).....	9

1.10 System Requirement	9
1.10.1 Hardware Development tool and technology requirement	9
1.10.2 Software Development tool and technology requirement	10
1.10.3 Programming language & database tools	11
1.11 Organization of the project.....	12
CHAPTER TWO	13
2. DESCRIPTION OF THE EXISTING SYSTEM	13
2.1. Introduction of Existing System.....	13
2.2 Users of Existing System	13
2.3 Major Functions of the Existing.....	14
2.4. Forms and Other Documents of the Existing Systems.....	15
2.5. Drawbacks of the Existing System.....	16
2.6. Business Rules of the Existing System	16
CHAPTER THREE	18
3 PROPOSED SYSTEM	18
3.1. Introduction of Proposed System	18
3.2. System Requirement Specification	19
3.2.1 Functional requirement	19
3.2.2. Non-functional requirements.....	20
3.2.2.1. User Interface and Human Factors	20
3.2.2.2. Hardware consideration.....	21
3.2.2.3. Security Issues	21
3.2.2.4. Performance Consideration	21
3.2.2.5. Error Handling and Validation	22
3.2.2.7. Back and recovery	22
3.2.2.8. Physical Environment.....	22
3.2.2. 9. Resource Issues.....	22
3.2.2.10. Documentation.....	22
CHAPTER FOUR.....	23
4. SYSTEM ANALYSIS	23

4.1. Introduction of System analysis	23
4.1. System Model.....	23
4.1.1 Use case diagram	24
4.1.3 Use case Descriptions	28
4.1.4. Use case Scenario	37
4.2. Object model	39
4.2.1. Class Diagram.....	39
4.2.2. Data Dictionary.....	41
4.3. Dynamic Model.....	44
4.3.1. Sequence Diagram	44
4.3.2. Activity Diagram	51
4.3.3. State Chart Diagram	58
CHAPTER FIVE	62
SYSTEM DESIGN	62
5.1. System Overview	62
5.2. Design goals	62
5.2.1 Performance and portability.....	63
5.2.2 Maintenance.....	63
5.2.3 End user	63
5.2.4 Dependability.....	63
5.2.5 Priorities of Design Goal	64
5.3. Current System Architecture	64
5.4. Architecture of the proposed System	65
5.4.1. Sub-system decomposition.....	66
5.4.2. Hardware/Software mapping (Deployment design)	70
5.4.3. Detailed Class Diagram.....	72
5.4.5. Persistent data storage and management	73
5.4.6. Access Control and Security	74
5.5 Packages.....	75
5.6. User interface design.....	76

6.6. Implementation of User Interface	83
CHAPTER SEVEN	84
7. CONCLUSION AND RECOMMENDATION.....	84
7.1. Conclusion	84
7.2. Recommendation	85
8. Reference	86
9. APPENDICES	87
Appendix I: prepare agreement format	87
Appendix II: Sample Source Code.....	88

List of Table

Table 1: Actors with their use cases	26
Table 2: Use Case identification number	28
Table 3: Use Case description for Login	29
Table 4: Use Case description for create account	30
Table 5: Use Case description for manage employee or user	31
Table 6: Use Case description for manage equipment.....	32
Table 7: Use Case description for request feedback.....	33
Table 8: Use Case description for delete user.....	34
Table 9: Use Case description for Update user account	35
Table 10: Use Case description for Registration of user	36
Table 11: Computer data dictionary	41
Table 12: Assistance data dictionary	41
Table 13: Attendant data dictionary.....	42
Table 14: Equipment data dictionary.....	42
Table 15: Register data dictionary	42
Table 16: Student data dictionary	43
Table 17: User account data dictionary	43
Table 18: User data dictionary.....	44

List of figure

Figure 1: prepare agreement format	16
Figure 2: The Proposed System Use Case Diagram.....	27
Figure 3: Class diagram for Computer laboratory monitoring and management system for College of computing and informatics	40
Figure 4: Sequence diagram for Login.....	45
Figure 5: Sequence diagram for Create account.....	46
Figure 6: Sequence diagram for manage Employee and user.....	47
Figure 7: Sequence diagram for attendants manage laboratory and students/user	48
Figure 8: Sequence diagram for feed back	49
Figure 9: Sequence diagram for registration of user	50
Figure 10: Activity Diagram for Login and Create Account.....	51
Figure 11: Activity Diagram for update user and equipment	52
Figure 12: Activity Diagram for search user	53
Figure 13: Activity Diagram for Registration of user	54
Figure 14: Activity Diagram for generate report.....	55
Figure 15: Activity diagram for request	56
Figure 16: Activity diagram for send notice.....	57
Figure 17: UML state chart diagram for login form.....	58
Figure 18: UML State Diagram Registrations.....	59
Figure 19: UML state chart diagram for announcement	60
Figure 20: UML state chart diagram for Generate report.....	61
Figure 21: System Architecture Diagram of CLMMCCI.....	66
Figure 22: Sub-system decomposition	69
Figure 23: Hardware/Software mapping (Deployment design).....	71
Figure 24: Detailed class diagrams.....	72
Figure 25: persistence diagram Computer laboratory monitoring and management system	73
Figure 26: Package Diagram	75
Figure 27: homepage interface	76
Figure 28: login interface	77

ABBREVIATION OR ACRONYM

Acronyms	Description
CLMMSI	Computer Laboratory Monitoring and Management System for College of Informatics
CMS	Classroom Management Software
DBMS	Data Base Management System
HTTP	Hypertext Transfer Protocol
MYSQL	Structural query Language
WKU	Wolkite University

ABSTRACT

The purpose of this project is to develop a computerized Computer laboratory monitoring and management system that can be used to revolutionize the traditional management system that currently implemented in a majority of the system are computerized systems so we developed the computer laboratory monitoring and management system through online by developing web-based system. This computerized computer laboratory monitoring and management system is designed to assist the business routine in terms of having better management as well as easier to handle daily computer laboratory operation of computer laboratory management system in College of Informatics. This system is designed for the College of Informatics laboratory management system. The chosen methodology for this project is iterative model methods. Therefore **iterative model** is a particular implementation of a **software development** life cycle (SDLC) that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. The computer laboratory management system described in this document has been designed to fill a specific laboratory with the ability to offer their managers and management without having to invest large amounts of time and designed specifically for them. The highly customizable system allows the laboratory admin to easily manage the site content, most importantly the student logs, through a very intuitive graphical interface.

CHAPTER ONE

1. Introduction

Computer laboratory monitoring and management system is still requiring more development of information technology solutions and applications to improve the management system of the College and legal documents of the laboratory.

Web-based computer laboratory monitoring and management systems to help the College Of Computing And Informatics staffs and managers to perform their tasks like informing damaged computers to managers, managing lab attendants, and other services, etc. The main purpose of a web-based laboratory management system is to reduce the effort of the candidate and create a smooth working environment for the College Of Computing And Informatics (1).

1.1 Organizational Background

“Wolkite University (WKU) is one of the third generation higher institutions that have been founded in 2012. It is established for the purpose of providing and promoting higher education learning, research, and outreach programs in the country to ensure the realization of the national vision of reaching the level of middle income countries by 2020. The University is located in the Southern Nation Nationalities Regional State, in Guraghe zone, 158 km southwest of the capital city, Addis Ababa, on the way to Jima. In November 2009 the late prime minister, his Excellency Mr. Meles Zenawi, laid the foundation stone of the University in a plain landscape which is quite ideal for academic pursuit. It is situated at Gubreye sub-city, 14 km away from Wolkite town, of the gubreye-butajira road. The major link road to the University is a direct route to Wolkite-Jimma, Wolkite-Hossana and wolkite-Butajira.

And the college of computing and informatics was established in 2011/12 with three departments (Computer Science, Information Technology and Information Systems) and department of Software Engineering was opened in 2013/14. Currently, the college has a total of 130 staffs and 1429 undergraduate students.

The vision of the College is to serve students and the community through excellent learning programs, by conducting relevant and effective research and community engagement projects.

The College commits itself to the achievement of academic excellence in the field of computing and informatics. Our graduates are in big demand from industry and from higher education institutions throughout the country. The college is well recognized in the country due to the performance of its graduates in their area of work. The college will always strive to succeed the delivery of quality education, problem solving research undertakings and community services.

1.2 Motivations

When we stayed for the last three years at Wolkite University College Of Computing And Informatics Computer Laboratory has many problems, Such as storing the computer laboratory information is only done manually whereby the records of all laboratory equipment's used and allocated to assistant are recorded in a piece of paper or spreadsheet.

The group is motivated in order to solve the problems in manual computer laboratory monitoring and management system and automate the system. We are interested to maintain the computer laboratory easily by computer system and reduce paper work.

1.3 Statement of the problem

College Of Computing And Informatics laboratory management system uses a manual system to monitoring and manage computer laboratory.

The existing system is using a manual to plan laboratory agreement, delegate lab assistance too many problems encountered in using a manual system. Because of manual method, there is a lack of protection of data and therefore more man resources required so this leads to time taking and requires manual calculations. It also uses a lot of resources, such as paper and pens. This causes difficulty in searching a document from a corpus and there is some data loss and damage because of paper.

In the current scheme, if a student's computer fails, they must go to the department and personally report their issues. Students waste their time as a result of this because department administrators cannot be present at the time. As a result of the managers' workload, students cannot have an instant response to their questions.

As the industrial project, we observed when dealing with the existing manual system research has the following problems such as:

- ✚ Resource consumption and needs resources like manpower, cost, and time, documents to fill user's or customer's information, and others.
- ✚ Data redundancy or some data's are duplicated because the system is manual and paper-based.
- ✚ Lack of effective and efficient service for the monitoring and management system.
- ✚ The system is costly; the system takes a lot of hard copies and other instruments.
- ✚ The existing system does not create a smooth workflow between managers of the laboratory and laboratory assistance or employees.
- ✚ The data handling and manipulation mechanism is not secured and well organized.
- ✚ Poor organizational and record-keeping problems that system from the manual service.

1.4 Objective of the project

1.4.1 General objectives

The general objectives of this project to develop web based monitoring and management systems for computer laboratory.

1.4.2 Specific objective

To achieve general objective of the project we have specifically categorized objective to meet the final goal into the following sub-divisions with some activities.

- ✚ To study the existing system of computer laboratory monitoring and management system.
- ✚ To identify problems with the existing system on the organization.
- ✚ To propose a solution to the identified problem.
- ✚ To design web based system for computer laboratory monitoring and management system.
- ✚ To implement the designed system.

✚ To test the project on the organization.

✚ To deploy the project on the organization.

1.5 Feasibility Study of the new system

A feasibility study is a test of system according to its workability, impact on the college, ability to meet user needs and effective use of resources. The main objective of the feasibility study is to test the technical, operational, legal and economic feasibility of the new system

1.5.1 Operational feasibility

The system will operate on any kind of platform without any failure, any user can operate the system because of this and other reason our system is operational feasible.

Operational feasibility is mainly concerned with issues like whether the system will be used if it is developed and implemented .The new system is operationally feasible since the system developed by simple languages and has better user interface, users can easily use the system i.e. the system member or team can provide training and to new users. Operationally, the systems give capable of producing quick and qualitative services to user.

1.5.2 Technical feasibility

The proposed system can be technically feasible because the technical resources needed to develop, install and to operate can be done by the developer .The proposed system can be easily maintained and repaired without requiring high exporters or technical assistants. This system is easily upgraded to provide the necessary information for the users.

1.5.3 Economic feasibility

After the proposed system is finished it will requires less amounts of cost budget .and also it save time and resource.

The current system use paper, pen and other costs due to improper mechanism to deal with the user information. For the system if the expected benefits equal or exceed the expected costs, the system can be judged to be economically feasible. Economic feasibility can be seen in two ways:

Tangible benefits:-

- ✚ Decrease response time and reduce processing error.
- ✚ Increased service and reduced extra resource.
- ✚ Error Reduction
- ✚ Increase Speed of activity

Intangible benefits:-

- ✚ Better service to student and Reduce Resource Consumption Increase, Security, reliability and trust.

1.5.4. Legal feasibility

The system will not have any conflict with the rule and regulation of university policy. The system is legally acceptable since it respects university's rule and regulation and other constitutional laws of the country. Perform tasks effectively and efficiently.

1.6 Scope and limitation of the project

1.6.1 Scope of the project

The scope of the project covers the development of a computer laboratory monitoring and management system for the College Of Computing And Informatics.

The system would be beneficial to the Computing and Informatics College, which is responsible for the computer lab and other technical equipment at Wolkite University College of Computing and Informatics.

The proposed system enables the students, lab assistants, lab attendants, and department head to communicate online easily sending requests and notifications.

The proposed system performs different tasks some of these are:-

- ✚ Reporting the damaged computer to the laboratory manager.
- ✚ Registration of students who uses computer laboratory.
- ✚ The Department Head can print out the attendance of the student; lab equipment and report information from attendants.
- ✚ Manage the laboratory attendants and assistances.
- ✚ Department Head upload/post notice.

- ✚ Student, Assistant, and attendant view notice.
- ✚ Registration of computers in the database with its model.
- ✚ The Department Head to monitor and track students in using the computer.
- ✚ The Department Head can add, edit and delete students, attendants, and assistant information.

1.6.2 Limitation

- ✚ The system by itself does not check the damaged computers and the computer laboratory equipment.
- ✚ The systems do not have digital Signature processing

1.7 Significance of the project

The project is aimed at improving the computer laboratory management system and make service delivered an effective way to College Of Computing And Informatics computer laboratory. It solved the major problems of service delivery system in the management systems of in computer laboratory.

The main significant of this project is to help the Computer laboratory worker to perform their work: -Faster, Efficiently, Accurately, Secure with wise use of information resources, available, Technological advancement of the management system and Protect unauthorized access.

1.8 Beneficiaries of the Project

After accomplishing this system there are a lot of beneficiaries groups like:-

1. **University:** In the manual system, there is a loss of materials like paper, a pen which costs much, uses more manpower, and also takes more time. But our system reduces the loss of costly materials and manpower. It has great importance for all users who uses the computer laboratory monitoring and management system because the system facilitates their job effectively and efficiently.
2. **Department Head:** - our system is used for the Department Head to manage and monitor every activity of the computer laboratory. For example to follow up the work of lab assistance, lab attendants, students, and types of equipment of the laboratory.

3. **Lab Attendants:** - in the current situation there are no lab attendants in computing and Informatics College but for the future, it may present; this system is used for lab attendants to monitor and manage the laboratory and to report any challenges simply for the lab manager.
4. **Students:** - students are also beneficiaries of this system. If anything is happening in the laboratory, they should report to lab assistance, to lab attendant, and also to the lab manager by using this system and to use laboratory.
5. **Team Member:** - by developing this system and by solving the problem of computer laboratory monitoring and management system they will get great satisfaction and they will graduate with grade distinction..
6. **Lab Assistant:** - by developing this system and by solving the problem of computer laboratory monitoring and management system lab assistants can easily assist the students

1.9 Data Collection Methodology

The Methodology describes how the project team is going to implement the proposed system. To do that first of all the project team have identified the main parts of the system basically the project would be on developing a suitable computer laboratory monitoring system for users.

In order to do that we use the following methods.

1.9.1 Data sources

The main data source for this project was the College Of Computing And Informatics. Since the monitoring and management system is not structured and well organized: we used the interview method to gather information from Computing and Informatics College laboratory manager, Department Head or College Dean.

1.9.2 Data Collection Method

We have used different methods to collect data. Data collection is the most important part of the project to find the main requirement of the system and to understand how the system does.

A primary source:-In primary data collection, we collect the data by the following methods, such as interview and observation.

- + **Interview:** - The reason why we select interview is, it permits clarification of questions, and also we have to interview the head of the college to know more about how the computer lab works.
- + **Observation:** - To gather relevant information the project team observes how the current system works. As we observe computer laboratory data is recorded on paper and put on the shelf.

Secondary sources:-are data that already exists. It is a document created at a later time than the event being researched, by someone who did not experience the said event.

- + **Document analysis:** - To get more information about the project we used earlier documents that help us to develop this project. During the analysis of documents, we considered those documents which can bring more features to the project relevant to develop the proposed system
- + **Internet:** Internet helps us to see the available samples and to download different types of tutorials which help us in developing the system.

1.9.3 System Testing Methodology

Is defined as strategies and testing types used to certify that the Application under Test meets client expectations.... Examples of Testing Methodologies are:-

Unit testing: - performed on each module or block of code during development. Unit Testing is normally done by the programmer who writes the code.

Integration testing: - done before, during and after integration of a new module into the main software package. This involves testing of each individual code module. One piece of software can contain several modules which are often created by several different programmers. It is crucial to test each module's effect on the entire program model.

System testing: - done by a professional testing agent on the completed software product before it is introduced to the market.

1.9.4 System Analysis and Design Techniques

In this project, the team used object-oriented system development methodology (OOSD) and it has two phases.

1.9.4.1 Object Oriented Analysis (OOA)

During this phase the team used to model the function of the system (use case modeling), find and identify the business objects, organize the objects and identify the relationship between them and finally model the behavior of the objects.

Analysis tools

-  Analysis class diagram
-  Use case diagram
-  Sequence diagrams
-  Activity diagrams

1.9.4.2 Object Oriented Design (OOD)

During this phase the team used EdrawMax software to refine the use case model, and to reflect the implantation environment, model object interactions and behavior that support the use case scenario and finally update object model to reflect the implementation environment.

Design tools

-  Deployment diagram
-  Design class diagram

1.10 System Requirement

System requirements are the hardware and software components of a computer system that are required to do the proposed system and simply identify tools and methodology. There are different System requirements:-

1.10.1 Hardware Development tool and technology requirement

Hardware requirements are the touchable and visible components that are necessary to develop a system.

- ✚ **Computers:** To do different activity or applications.
- ✚ **Flash disk (8GB):** To store data. The data in the computer may be lost because of different cases and to take available data from different sources.
- ✚ **CD (700MB):**-To store data and take back up.
- ✚ **Pen and Paper:**-To design the system before typing on the computer
- ✚ **Mobile:**-for idea exchanging

1.10.2 Software Development tool and technology requirement

The software requirements specification is the single most important document in the software development process. It provides the basis for development as well as for validation. The following are software requirement:

- ✚ **XAMP Server:** - is a free and open source cross-platform web server solution stack package, consisting mainly of the Apache HTTP Server, MySQL database, and interpreters for scripts written in the PHP and Perl programming languages.
- ✚ **MySQL:** - for storing the data because:
 - Most of the time MYSQL is compatible with PHP programming language and MYSQL portable.
 - MYSQL it is an open source database.
 - Its performance is high.
 - Supports large number of surrounded applications which makes MYSQL very flexible.
- ✚ **Editor:**-Notepad++, notepad is a free portable handy and easy-in-use code editor that is primarily intended for quick and easy creation and editing of PHP, HTML, CSS, JavaScript files.
- ✚ **Edraw Max and Visual paradigm editor:**-For drawing different UML diagrams.
- ✚ **Browser:** - Browsers for running the code like chrome, Firefox, opera, UC browser, and Internet explorer used to run the program.
- ✚ **Microsoft office word 2016:**-For writing documentation.
- ✚ **Microsoft PowerPoint 2016:**-For presentation.
- ✚ **Window 10:**-An operating system.

1.10.3 Programming language & database tools

A programming language is a formal computer language designed to communicate instructions to a machine, particularly a computer. The user interface (front end) of the application developed in HTML, CSS and Java Script to develop the presentation layer. That means the client side programming language is HTML, CSS and Java Script to display the system in the web browser. Database layer (back end) programming language is apaches server (MYSQL).

Front –end programing: - are programming Languages that run on browser the user see or directly interact with it as with client-side technology.

✚ **Web based (PHP):**-a server is scripting language, and a powerful tool for making dynamic and interactive Web pages. It is a programming language that allows web developers to create dynamic content that interacts with databases. And it also enables our software development.

- ✓ **Easy to understand:** when compared with other scripting languages, PHP can be understood easily because it has simple techniques and features.
- ✓ **Increased efficiency & usability:** PHP provides incomparable efficiency and usability when it is used for the development of websites.
- ✓ **Data processing:** a website that has been developed using the PHP functions has fast features of data processing.
- ✓ **Cost advantages:** PHP based websites are affordable to develop, design, modify and customize.

✚ **HTML:** - is a client side programming language for developing the client side or the user interface.it tells the browser how things are supposed to be displayed and used for different. The reason why we use HTML is editing and running code is fast; allow the addition of dynamic and interactive activities to web pages.

✚ **CSS:** - cascading style sheet performs the designs of the interface and performs the design layouts.

Back-end programing: - are programming Languages that run on browser the user doesn't see it or directly interact with it as with client-side technology. For Example PHP, Perl, Python, MYSQL Etc. but we are used PHP language.

1.11 Organization of the project

This project document contains seven chapters including this chapter. Chapter two assesses and describes concepts with regard to description of the existing system, aiming to give a general view to the reader of the document about the Existing System in the computer laboratory monitoring and management system. Chapter three it describes about Proposed System in computer laboratory monitoring and management system. In chapters four and five, we presented the analysis and design of the developed system respectively. Chapter six it describes about it describes about Implementation and Testing. Chapter seven it describes about it describes about Conclusion and Recommendation in the computer laboratory monitoring and management system in the remaining chapters, prototype development and conclusion and recommendations are briefly explained.

CHAPTER TWO

2. DESCRIPTION OF THE EXISTING SYSTEM

2.1. Introduction of Existing System

This chapter describes the existing systems of the College of computing and informatics Lab Management system. The chapter discusses reasons behind studying the existing system; identify problems of the existing system that related to time, space and resource, the major players of the existing system.

In the existing system students take lab properties under the current scheme, department heads plan an agreement format for them. This prepared agreement format is offered to lab assistants who are assigned to students. The lab assistance is delegated to each laboratory by the department, and he or she takes over the key and receives a list of lab resources. After taking the list of lab properties Lab assistant assigns lab properties for each student by signing the agreement format i.e. every student must sign the agreement format for his/her taking of lab properties, then the format is submitted to the department in case of resource lost. Lab assistance checks the lab resource daily for their existence. If a computer fails in the lab, students can report it to lab assistance; if the lab assistance can troubleshoot it, he/she can, if not he/she reports it to the department. In the existing system the department and students' communication is in the form of an advertisement on the wall in the form of paper, and the lab rules and guidelines are in the form of paper.

For the college of computing and informatics, currently there is no lab attendant. The lack of a lab attendant makes it difficult to keep track of anything going on in the lab.

2.2 Users of Existing System

Users represent anything or anyone who can participate in the system. This may include people that are external to the system. In the existing system of computer laboratory monitoring and management system there are different players namely:

Department head

- ✚ Responsible for controlling the laboratory assistances and coordinating them.
- ✚ Manage all computer laboratories.
- ✚ Prepare and manage computer information.

Lab assistance

- ✚ The responsibility of lab assistance is controlling the lab properties and reporting the material at the end of the year.
- ✚ The laboratory assistance manages the laboratory and manages students.
- ✚ Coordinates with manager to increase services effectiveness.

Students or Customers

- ✚ The responsibility of a student is taking the lab property by signing the agreement format and protecting them from damage.
- ✚ The students get services from computer laboratory.
- ✚ Use computer laboratory.

2.3 Major Functions of the Existing

There are many functions the existing system gives. These are:

- ✚ Providing the computer and computer laboratory rooms for students who belongs to College of informatics.
- ✚ There is process which the assistants fills the manual form and scheduling system to inform their existence at a time.
- ✚ In the existing system there is no any kind of forms which helps to user to search keywords and login forms to the administrator to enter his/her username and password because it is manual.

The input is user order face to face contact in the laboratory.

- ✚ Preparing schedule for laboratory rooms.
- ✚ Department prepares agreement format for students for taking of lab properties.
- ✚ Lab assistance prepares reports to the department after end of the year.
- ✚ Lab assistance report to department when there is failurity of properties in lab.
- ✚ Managing and control lab properties.
- ✚ Assigning lab assistance for laboratory.

2.4. Forms and Other Documents of the Existing Systems

Most of the existing system is paper-based that is time and labor consuming works. Many forms of used in this existing system are drawn by pen. After drawing thy start fill the data concerning that forms what they need.

Preparing agreement form in the existing system

WOLKITE UNIVERSITY
COLLEGE OF COMPUTING AND INFORMATICS
DEPARTEMENT OF INFORMATION TECHNOLOGY
Computer Laboratory services
Property Management form
Lab 1

Purpose: Material ownership lab
Group leader Name: Lidet Wudma
Group : All IT 4th . year Student 2013 E.C
Date: 19/12/2013

S.No	Item Name	Quantity	Serial number 1	Serial number 2	Serial number 3	Remark
1	Desktop computer	1	J060521			
2	Network cable	1				
3	Keyboard	1				
4	Mouse	1				
5	Seat chair	2				
6	Computer table	31				
7	Divider	1				
8	Staplizer	1				
9	Switch	1				

Member list:

1. Lidet Wudma
2. Tishan GORHU
3. Mariet Negewid
4. _____

signature: _____
signature: _____
signature: _____
signature: _____

Figure 1: prepare agreement format

2.5. Drawbacks of the Existing System

Any information about the laboratory (schedule, agreement format preparation) is prepared manually in the current procedure, which is tedious and exhausting for all department heads. no centrally managed database records information about the college, lab assistants, students who work in the laboratory.

After the end of the year, the lab assistant submits manual reports to the department head. This is tedious for lab assistants and department managers, and it adds to their workload. The current processes have several problems, which we must consider as a result of manual operation execution.

When a problem occurred in the laboratory, the students go to the lab assistant's office and communicate with them manually. The student does not get immediate feedback because of they may not available at that time. Also, lab assistants go to the department head and inform these problems to troubleshoot the problems.

For the college of computing and informatics, currently there is no lab attendant. The lack of a lab attendant makes it difficult to keep track of anything going on in the lab.

In the existing system, if there is new information, it is posted on the wall in the current scheme, which causes fading due to natural factors such as rain and winds. As a result, reposting the new details necessitates the use of other expensive items, such as paper for printing, which depletes money.

2.6. Business Rules of the Existing System

The business rule is a principle or a policy in which the proposed system operates accordingly. It deals with access control issues. It often pertains to access control issues, operating policies and principles of the organization. The organization has the following principles in the existing system which includes:-

Business Rules For students

BR1: Students are not allowed to download other things rather than learning materials they occupy computer memory.

BR2: The Computer is an expensive sensitive machine. Treat all computer parts with cautious care.

BR3: It is forbidden in the computer laboratory to use social media and pornography films.

BR4: In the laboratory, the computer is assigned within 1 to 3 group arrangements so one group takes two/three computers with numbers of students.

BR5: In the laboratory before taking of computers there is format prepared by the department. So every must have to sign these agreement formats.

BR6: Students, who get problems, identify and provide request for lab assistance.

BR7: Every student may sign the agreement format with their departments.

Business Rules For assistance

BR1: The assistance must take the lab resource list for controlling.

BR2: Checking and cleaning of all computer accessories.

BR3: The assistance must solve the problem that the student faced.

BR4: The assistance must prepare the report after end of the year.

BR5: Check the lab resources existence weekly.

Business Rules For department

BR1: The department must control lab assistance.

BR2: The department may prepare agreement format.

BR3: The department must request for ICT director when there is some problem occurred.

BR4: The department may prepare rules and regulations of laboratory.

BR5: The department must prepare the lab schedule.

CHAPTER THREE

3 PROPOSED SYSTEM

3.1. Introduction of Proposed System

The existing system has its problem and drawbacks. Because of these problems and drawbacks of the current system practiced in the organization, we have proposed problem-Solving systems named as “Web-based Laboratory monitoring and management system for College of Computing and informatics” which will minimize and eradicate those difficulties and drawbacks of the existing system.

The main aim of the proposed system is to implement a web-based Laboratory monitoring and management system for Computing and Informatics College which allows students, lab assistance, lab attendants, and Department head as well as college Dean can easily communicate with each other.

Generally, we proposed Wolkite University computing and Informatics College Laboratory monitoring and management system that allows the computing and informatics college students to request incidence to the laboratory assistants when there is failures occurred in the laboratory, and then the lab assistance troubleshoots it. If the lab assistant cannot troubleshoot it, he/she sent a request to department Heads.

In the Proposed system, the lab assistance assists the assigned course by the department to the student. The proposed system would have high data protection, the ability to organize all information in a single client-server system, and a simple way to document and view item information through its well-organized user-friendly interface.

There is a lab attendant for the college of computing and informatics in the proposed system. The lab attendant is manages all of lab equipment and users, as well as reporting damaged computers to the department head, communicating with the lab through short messaging, and viewing notices.

3.2. System Requirement Specification

System requirement specification (SRS) is a description of a system to be developed, laying out functional and non-functional requirements, and may include a set of use cases that describe interactions of users with the system. The system requirement specification is requirement that is required for the project development. To derive the requirements we need to have clear and thorough understanding of the products to be developed or being developed. This is achieved and refined with detailed and continuous communications with the project team and customer until the completion of the software.

3.2.1 Functional requirement

Functional requirements describe the interactions between the system and its Environment independent of its implementation. The environment includes the user and any other external system with which the system interacts. The system should provide how the system should react to particular inputs and how the systems behave in particular situations. The following are the functional requirements for our proposed system:-

Performance requirement

- ✚ Computer performance is characterized by the amount of useful work accomplished by a computer system compared to the time and resources used.

User management

- ✚ The system verifies user account to login into the system by authenticating their information.
- ✚ Lab attendants can report the lab operations for the department.
- ✚ The system handles user information.
- ✚ The system enables Users/Customers to change and updates their profiles.
- ✚ The system should store computer information in the database.
- ✚ Data storage and retrieval: All the user file record information should be kept properly in a well-organized database so that retrieving these files will easily and faster.
- ✚ Searching, inserting, updating and deleting records of computer laboratory user.

Storage requirements

The application that we are going to develop will store all its data in the **MYSQL** database which is the backend or panel of the website.

Functional requirements are the intended behavior of the system. This behavior may be expressed services, tasks or, functions that the system is required to perform.

Login: - to identify the authorized person to use the system.

Create an account: - the system will support to creating an account for student, assistants and laboratory attendants by a manager.

Update account: - the system will support to updating the account for computer laboratory user.

Register: - the system will register the computer laboratory information in the database

3.2.2. Non-functional requirements

Nonfunctional requirements describe user-visible aspects of the system that are not directly related to the functional behavior of the system. Nonfunctional requirements include quantitative constraints, such as response time (i.e., how fast the system reacts to user commands) or accuracy (i.e., how precise are the system's numerical answers). The non-functional requirements of our proposed computer laboratory monitoring and management system will list as follows:-

3.2.2.1. User Interface and Human Factors

The interface of the proposed system is simple to understand; easy to use and user-friendly interface and users of the system easily use and perform their task.

We have deeply studied our system user's status about their goals, skills, preferences, and tendencies, so that we point up the following considerations when designing our interface:

Interface simplicity: when our system users use our systems they will not get difficulty when using it in terms of interface.

Conventionality: we design our system by considering universal naming, alignment, symbols and icons which enable our users to use our system without requiring any learning time.

Using attractive and mining full colors: in order to make our users to be stick to our system when he or she uses our system without frustration and stress.

Any users who have basic computer skill will use our proposed system.

3.2.2.2. Hardware consideration

- ✚ The hardware that is consumed by the new system is basically a computer machine that is used for all the operation of the system. The Software product to be developed should run on existing standard computers. The system will be portable that can run on any type of computer and it supports any type of browsers.
- ✚ This Website opens in any personal computer supporting windows 7, Windows8, Windows XP and other smart phone.

3.2.2.3. Security Issues

We worry in security on greater extent to enable our users to trust our system. No one can access our system unless he or she is privileged to use our system services. We encrypted our password using md5 so that no one can see original password.

- ✚ Password entry shouldn't be less than 8 characteristics in order to prevent simple guess and expertise hackers attack.
- ✚ Then password entry should be encrypted using md5 before arriving to database.
- ✚ We use SQL injection preventer such as prepared statement in order to forbid arriving user input directly in an SQL query to disable execution of malicious SQL commands in the database.

3.2.2.4. Performance Consideration

The system will be accessed by many users or students inside the college. So it should be more given emphasis for the speed to access the system. The constraint characteristics are also one of performance characteristics of the system, the constraints can be user's computer low RAM, Network or connectivity failure, and others

3.2.2.5. Error Handling and Validation

When the users of the system interact with the system errors may appear. To control these In accuracies, the system will display different messages. Data errors which are entered in to the system may not be completely avoided but they can be minimized. The system handles exceptions input mismatch exception such as interchanging numeric and characters, username and Password mismatch by displaying alerts.

3.2.2.6. Quality Issues

Our system should be consistently good in quality or performance and able to be trusted. It also should be obtainable or accessible, capable of being made use of and at hand. Also it should have the ability to cope with errors during execution and cope with erroneous input.

3.2.2.7. Back and recovery

Our system should take data using portable storage device and store in cloud so that it may be used to restore the original data after a data loss event.

3.2.2.8. Physical Environment

We deploy proposed system in order to plan for and manage the transition and capabilities into operational use. We will install our system on Wolkite university server.

3.2.2. 9. Resource Issues

Time; it consumes much time in order to gather, analyze, design, implement and test, money; it consumes a lot of money during gathering information, analyzing and testing, effort; it uses our effort with a whole. Large sized smart phones, tablets and computers are required.

3.2.2.10. Documentation

Lab monitoring and management will provide two types of system documentation. These are internal and external documentation. System documentation addresses programmers, system developers, owners, and users.

Internal Documentation: This contains program source code.

 External Documentation: This also contains, use case models, the class and interaction diagrams (activity, sequence, and state diagrams).

CHAPTER FOUR

4. SYSTEM ANALYSIS

4.1. Introduction of System analysis

The system analysis modeling deals with analyzing the proposed system. It includes the system use case diagrams, the use case descriptions (scenarios), sequence diagrams, activity diagrams, analysis class diagram. After identifying the actors and use cases, the use cases are developed and textual descriptions (scenarios) are stated. The Sequence diagram is depicted based on the use cases which are developed for the proposed system. Activities will be represented by the activity diagrams (2).

4.1. System Model

System Model is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system. It is about representing a system using some kind of graphical notation, which is now almost always based on notations in the Unified Modeling Language (UML). Models help the analyst to understand the functionality of the system; they are used to communicate with customers.

Models can explain the system from different perspectives: An external perspective, where you model the context or environment of the system, An interaction perspective, where you model the interactions between a system and its environment, or between the components of a system, A structural perspective, where you model the organization of a system or the structure of the data that is processed by the system, A behavioral perspective, where you model the dynamic behavior of the system and how it responds to events.

Five types of UML diagrams that are the most useful for system modeling: Activity diagrams; which show the activities involved in a process or in data processing, Use case diagrams; which show the interactions between a system and its environment, Sequence diagrams; which show interactions between actors and the system and between system components, Class diagrams; which show the object classes in the system and the associations between these classes, State diagrams; which show how the system reacts to internal and external events.

4.1.1 Use case diagram

A use case diagram is a set of use cases for a system, the actors of these use cases, the relations between the actors and these use cases, and the relations among the use cases. The UML (unified modeling language) notation for a use case diagram is containing different shapes and each shape has its own meaning.

-  An oval represents a use case,
-  A man like figure represents an actor,
-  A line between an actor and a use case represents that the actor initiates and/or participates in the process.

Use cases

Use case diagram at its simplest representation of a user's interaction with the system and depicting the specifications of a use case.

Informally speaking, a use case is a story or a case of using a system by some users to carry out a process. A bit more precisely speaking, a use case describes the sequence of events of some types of users, called Actors, using some part of the system functionality to complete a process.

Actor

An actor represents a coherent set of roles that are entities external to the system can play in using the system, rather than representing a particular individual. An actor represents a type of users of the system or external systems that the system interacts with.

System Admin

Department Head

Laboratory attendants

Assistance

Student

A use case describes a sequence of action that provides a measurable value to an actor and draw as a horizontal ellipse.

-  An actor is a person, organization or external system that plays role in one or more interactions with the system and draw as stickman figure.

Relationship between actor and use cases exists whenever an actor is involved with interaction described by a use case and modeled as a line connected with use case and actors

Use case could be organized using the following relationships

Generalization: Generalization between use cases is similar to generalization between classes. Child use case inherits and behavior of the parent uses case and may override the behavior of parents.

Association: An association between an actor and a use case indicate that the actor and use case somehow interact or communicate with each other.

Include: Use case include is a direct relationships between two use case which is used to show that behavior of the included use case (the addition) is inserted in to the behavior of the including (the base) use case. It is used simplify large use case by splitting in to several use cases and common parts of the behavior of two or more use cases.

Extend: Extend use case typically defines optional behavior that is not necessarily meaningful by itself.

Actor	Use Case
Admin	 Log in  Create account  Update account  Delete account  Log out

Department Head	 Log in  Manage employee and user  View report, student record, Message  Post notice  Replay request  Monitor lab equipment  Manage lab  Log out
Attendants	 Log in  Monitor lab equipment  Manage lab  Report generate  View notice  Mange student  Log out
Assistant	 Log in  Assist practical class  View notice  Manage student  Log out
Student	 Log in  Send Request  Use computer lab  Log out

Table 1: Actors with their use cases

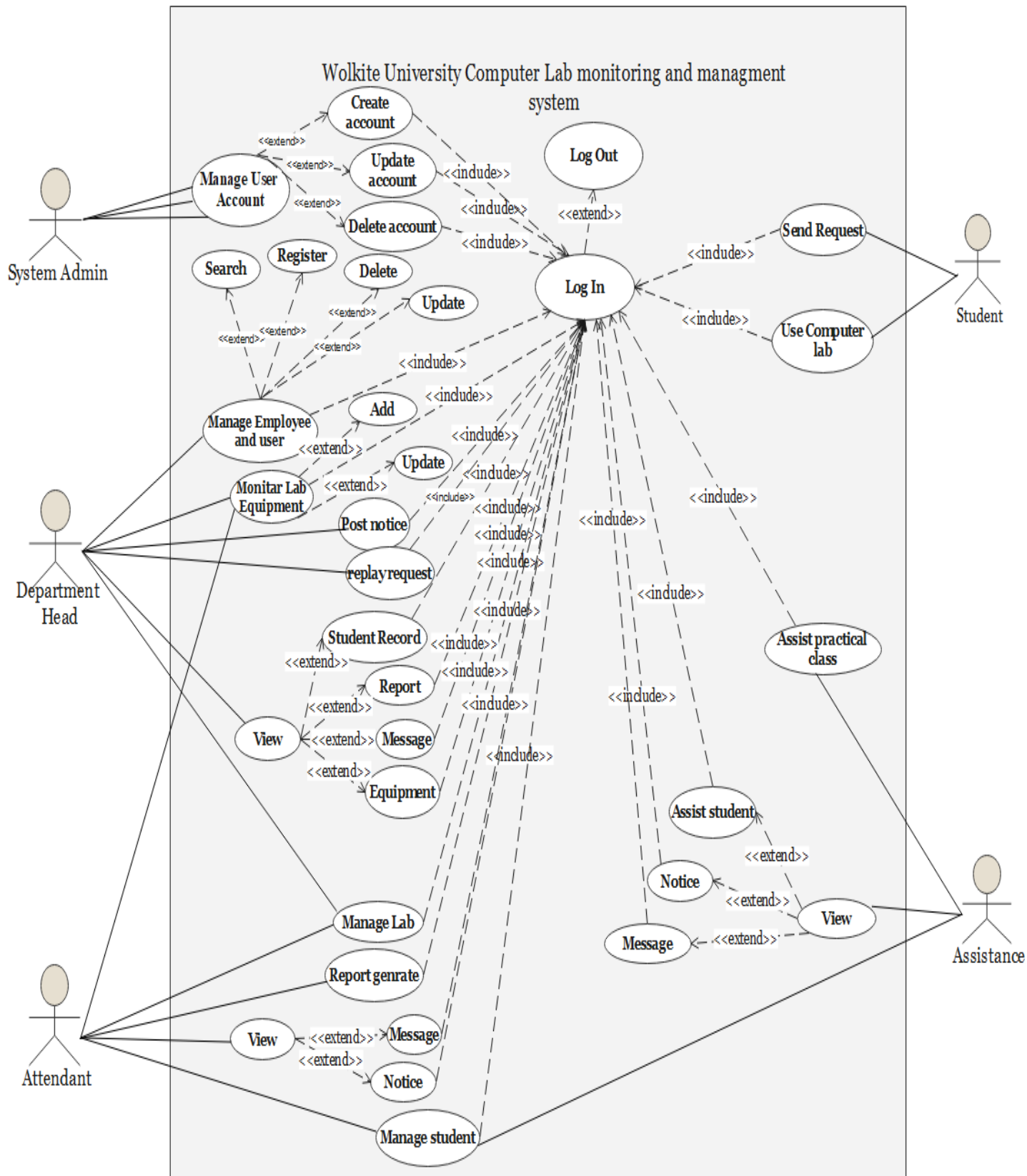


Figure 2: The Proposed System Use Case Diagram

4.1.3 Use case Descriptions

This part describes the pictorially represented use case documentations. As we have drawn the use case diagram for administrator to login and for the search box form, now we are going to define them each and individually (for the login form and for the search box form)

Use case identification No	Use case name	Use case description
USI 01	Login	Describe how user login to the system
USI 02	Create account	Describe how admin creates account for users
USI 03	Manage employee/user	Describe the admin how to manage laboratory attendants and users
USI 04	Monitor laboratory equipment's	Describe how the computer laboratory manager monitor the computer and computer based information
USI 05	send request	Describes the students how comment and request related to computer laboratory issue
USI 06	Delete user	Describes how the department head delete user
USI 07	Update account	Describes how department head update user and laboratory equipment
USI 08	Register user	Describes how the department head register user

Table 2: Use Case identification number

UCI 01	
UC Name	Login
UC Description	Login to the system by users own account in order to perform actions
Actor	System admin, Department head ,Attendants, Assistances, students
Precondition	The users should have an account
Normal Flow of event	1. The user activates the system. 2.The system display Home Page 3.The User Press Login Tab Then 4.Users Enters user name and password 5.Users clicks on login button 6. The system checks and validates the entered user name and password. 7.The system displays the user first page 8.Use case ends
Post condition	An authorized user entered to the system and can access the system and performs their actions
Alternative flow event	A7.The system informs to the user invalid user name and password A8. End.
Exceptional Flow of Event	E7. Error! Can't connect to server please try again. Message is displayed. E8. Use case ends.

Table 3: Use Case description for Login

	UCI 02
UC Name	Create Account
UC Description	Describes how an administrator creates account
Actor	Admin
Precondition	The administrator should successfully login into the system and know the information of users
Normal Flow of event	1.The Administrator click create account link 2. The system displays create account page. 3.The Administrator fills full information about users 4.The Administrator clicks on submit button 5 .The system validates the information interred by an Admin 6 .The systems register and create new account. 7 .The use case ends
Post condition	An Administrator creates Account successfully to the user
Alternative flow event	A6 The system displays “Fill all information correctly!” massage. A7 use case end
Exceptional Flow of Event	E6 The system displays “Error! Can’t connect to server please try again!” message. E7 use case end

Table 4: Use Case description for create account

	USI 03
UC Name	Manage employee or user
Use case description	Describes how Department Head manage user and employee
Actor	Department Head
Precondition	The Department Head first login to the system as manager
Normal Flow of event	1.The Department Head click manage employee and user link 2. The system displays manage employee and user page. 3.The Department Head fills full information about employee and users 4.The Department Head clicks on manage button 5 .The system validates the information entered by an Admin 6 .The system registers and manages employee and user account. 7 .The use case ends
Post condition	An Department Head successfully manages employee and user
Alternative flow event	The system displays “Fill all information correctly!” message

Table 5: Use Case description for manage employee or user

	USI 04
UC Name	Monitor laboratory equipment's
Use case description	Describes how Department Head monitor laboratory equipment
Actor	Department Head
Precondition	The Department Head must login into the system as administrator
Normal Flow of event	1.The Department Head click in to manage equipment link 2. The system display the laboratory equipment 3 .The Department Head fills full information about computer laboratory equipment 4. The system validates the information interred by an Admin 5.The system display and show the categories of equipment 6. The use case end
Post condition	The Department Head mange the equipment successfully

Table 6: Use Case description for manage equipment

	USI 05
UC Name	Send Request
Use case Description	Describes how students send request to the system
Actor	Students
Precondition	The students must be login into account as student
Normal Flow of event	1.Students wants to send request 2 Students login as student 3.The students click to send request link 4, The system displays request page 5. fill the request information 6. click send button 7.Use case end
Post condition	The Students feedback successfully

Table 7: Use Case description for request feedback

	USI 06
UC Name	Delete user
Use case description	Describes how the department head delete employee and computer laboratory user
Actor	department head
Precondition	department head must login into system as administrator
Normal Flow of event	1. department head wants to be delete employee or user 2.Login as department head 3. department head click to delete employee or user link 4.The system display delete user or employee page 5. fill the delete information depending on the grant 6. click delete button 7.Use case end
Alternative flow event	If the department head use incorrect user name or password incorrect system displays "incorrect user name or password " message
Post condition	User deleted

Table 8: Use Case description for delete user

	USI 07
UC Name	Update account
Use case description	Describes how manager update user
Actor	Administrator
Precondition	The administrator must login successfully
Normal flow of event	1.The administrator click update user 2.The system displays Update user form and also all records 3 The administrator fills the correct information for the specific updated data of user or directly press update button for specific row from displayed records 4. The system Displays the update form with previously registered information 5. The administrator files new information and clicks on Update button 6.The system checks the entry 7. The system Updates successfully 8. The use case ends
Post condition	A administrator Register Search user successfully from the DB

Table 9: Use Case description for Update user account

	USI 08
UC Name	Register user
Use case description	Describes how department head register computer laboratory user
Actor	department head
Precondition	The department head must login to the system.
Normal flow of event	1. The department head clicks register user/student link 2. The system displays register use form 3. The department head fills full information about the user 4. The department head clicks the register/save button 5. The system validates the information interred by the department head 6. The system registers successfully 7. The use case ends
Post condition	department head register user successfully
Alternative flow event	A6 The system displays “Fill all information correctly” messages A7 Use case ends
Exceptional flow event	E6 The system displays “Error! Can’t connect to server please try again!” message. E7 use case end.

Table 10: Use Case description for Registration of user

4.1.4. Use case Scenario

Scenario is an instance of a use case describing a concrete set of actions. Scenarios used as example for illustrating common use cases. In addition, we can describe a scenario using three fields:

Scenario for Login

Brief Description: This use case describes how a user logs into system

Actor: System Admin, students, Lab assistants, lab attendants, Department heads

Flow of Events: This use case starts when the actor wishes to log into the system

The actor students, Lab assistants and lab attendant have their own account in the site and the actors enter his/her name and password then the system validates the entered name and password and log the actor into the account. If there is error in username and password in validation process the system Says in valid username or password please try again. And if the entered username and password is valid the students and lab assistants open the system and access what they want.

Scenario for create account

Brief Description: This use case describes how a user create account for others to access system

Actor: System admin

Flow of Events: This use case starts when the actor wishes to create account for others

The actor System admin have their own account in the site

The System admin enters required information to create account like user name and password.

The system validates the entered name and password and creates account to desired user.

The Department head, students, lab attendant and lab assistants open the system and access what they want using their account.

Scenario for registering students

Brief Description: This use case describes how a Department heads register the students to access system

Actor: Department heads

Flow of Events: This use case starts when the Department heads wishes to register students to access the system

User types a user name of his or her choice and password. For confirmation User re-types the password Submit then System checks if the user name is not already in use and also System checks If the two passwords are identical then lastly, System registers the new users with the given parameters (user name, password)

Extensions:

If User name is already in use, the system asks the user to select another username and password which is not already in use. And also if the two passwords are different, the user requested to enter the same password only in the password text field and the system allows re-type the passwords which match with the above password.

Scenario for generate report

Brief Description: This use case describes how Lab attendants generate report to the department.

Actor: Lab attendant

Flow of Events: these use case stats when the lab attendant wishes to generate report to the Department.

The lab attendants have his own username and password to access the system. By using this Username and password the lab attendant access the system to generate the report. The lab attendant opens the system and click on generate report, then the form is displayed and the system says select appropriate required information to generate report. The lab attendant selects

the required information and select generate report button. Then lastly the system generates the report.

The lab attendants have his own username and password to enter the device. Through using this Username and password the lab attendant enter the device to produce the study. The lab attendant logs in and clicks on "Generate Report," which displays a form with the message "Pick sufficient necessary details to generate report." The lab attendant gathers the necessary data and s

Extensions

If the selected information is incorrect format or the selected information is not full, the system Says invalid selection to generate report, please try to select appropriate report information.

4.2. Object model

Object Model is a logical interface, software or system that is modeled through the use of object-oriented techniques. It enables the creation of an architectural software or system model prior to development or programming. In our system we use object model like class diagram and data dictionary

4.2.1. Class Diagram

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. In our project there will be class that makes the system like students, administrator, attendants etc.

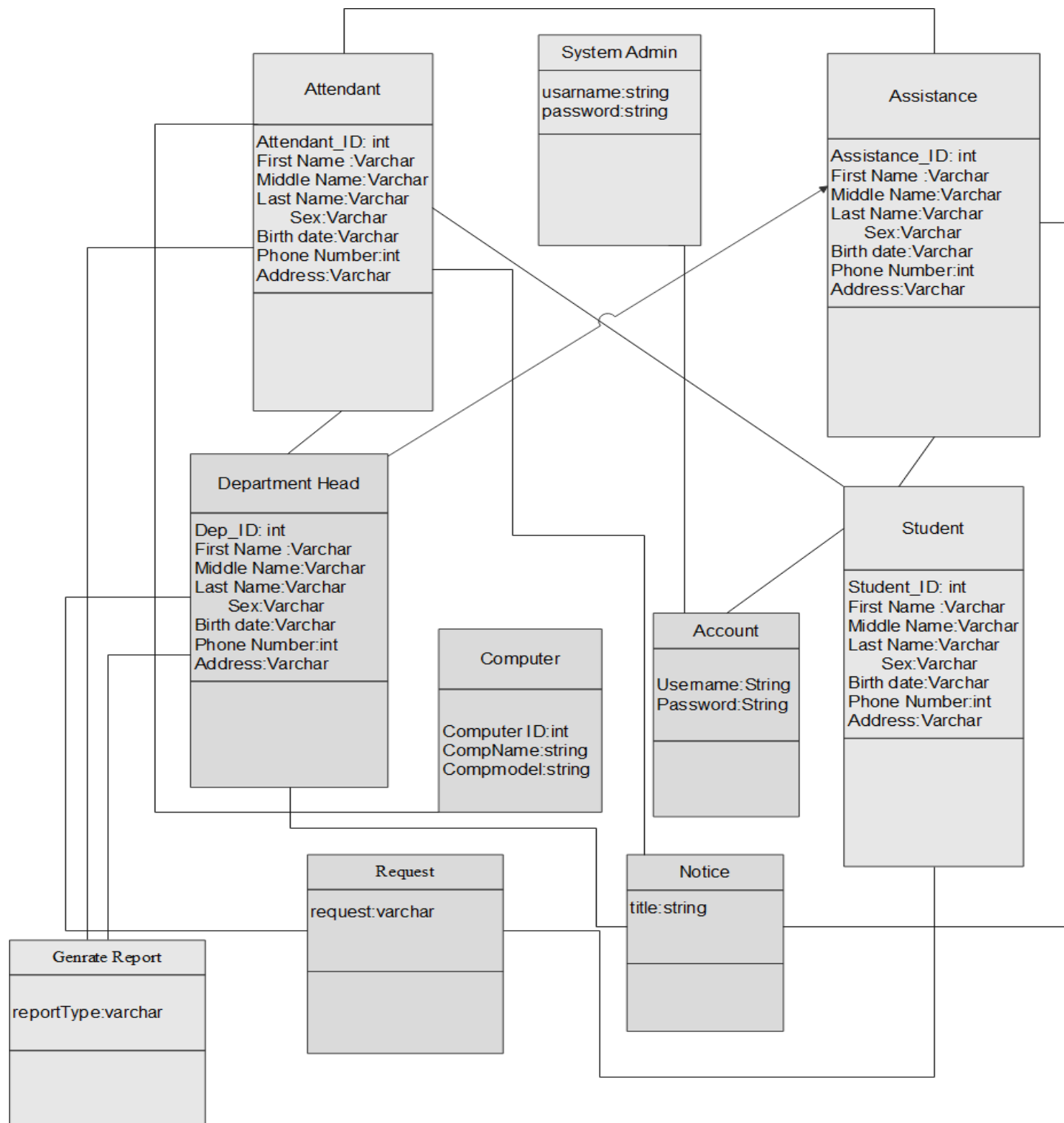


Figure 3: Class diagram for Computer laboratory monitoring and management system for College of computing and informatics

4.2.2. Data Dictionary

A data dictionary is a file or a set of files that includes a database's metadata. The data dictionary hold records about other objects in the database, such as data ownership, data relationships to other objects, and other data. In our system the following table describe data dictionary. The table 11 below illustrate data dictionary of our system.

Table 11: Computer data dictionary

Attribute	Data type	Data Size	Constraint	Key Constraint
COMPUTER_ID	Integer	10	NOT NULL	PRIMARY
COMPUTER_NAME	String	20	NOT NULL	
COMPUTER_RAM	Integer	10	NOT NULL	
COMPUTER_QUANTITY	Integer	10	NOT NULL	
COMPUTER_DEPARTMENT	String	20	NOT NULL	
COMPUTER_YEAR	String	20	NOT NULL	

Table 12: Assistance data dictionary

Attribute	Data type	Data Size	Constraint	Key Constraint
ASSIST_ID	Integer	11	NOT NULL	PRIMARY
ASSIST_NAME	String	50	NOT NULL	
ASSIST_EMAIL	String	30	Unique	
ASSIST_DEPTNAME	String	34	NOT NULL	
ASSIST_ASSIST	String	50	NOT NULL	

Table 13: Attendant data dictionary

Attribute	Data type	Data Size	Constraint	Key Constraint
ATTEND_ID	Integer	13	NOT NULL	PRIMARY
ATTEND_FIRSTNAME	String	13	NOT NULL	
ATTEND_LASTNAME	String	23	NOT NULL	
ATTEND_PHONE	String	23	NOT NULL	
ATTEND_SUBJECT	String	13	NOT NULL	

Table 14: Equipment data dictionary

Attribute	Data type	Data Size	Constraint	Key Constraint
EQUIPMENT_ID	Integer	10	NOT NULL	PRIMARY
EQUIPMENT_NAME	String	20	NOT NULL	
EQUIPMENT_MODEL	String	15	NOT NULL	
EQUIPMENT_RAM	String	10	NOT NULL	
EQUIPMENT_QUANTITY	Integer	12	NOT NULL	
EQUIPMENT_LABORATORY	String	16	NOT NULL	

Table 15: Register data dictionary

Attribute	Data type	Data Size	Constraint	Key Constraint
REGISTER_ID	Integer	10	NOT NULL	PRIMARY
REGISTER_FIRSTNAME	String	20	NOT NULL	
REGISTER_FATHERNAME	String	15	NOT NULL	
REGISTER_GFATHERNAME	String	10	NOT NULL	
REGISTER_EMAIL	String	23	NOT NULL	
REGISTER_PHONE	Integer	16	NOT NULL	
REGISTER_DATE	Date	20	NOT NULL	
REGISTER_SEX	String	23	NOT NULL	
REGISTER_YEAR	String	18	NOT NULL	
REGISTER_SEMESTER	String	23	NOT NULL	
REGISTER_COLLEGE	String	23	NOT NULL	
REGISTER_DEPARTMENT	String	30	NOT NULL	

REGISTER_CITY	String	23	NOT NULL	
REGISTER_REGION	String	23	NOT NULL	

Table 16: Student data dictionary

Attribute	Data type	Data Size	Constraint	Key Constraint
S_ID	Integer	10	NOT NULL	PRIMARY
F_NAME	String	40	NOT NULL	
L_NAME	String	40	NOT NULL	
M_NAME	String	40	NOT NULL	
DEPARTMENT	String	23	NOT NULL	
GENDER	String	10	NOT NULL	
B_DAY	Date	Date Size	NOT NULL	
AGE	Integer	30	NOT NULL	
CONTACT_NO	String	40	NOT NULL	
HOME_ADD	Text	Text	NOT NULL	
EMAIL	String	100	NOT NULL	
ACC_PASSWORD	String	15	NOT NULL	Unique Key

Table 17: User account data dictionary

Attribute	Data type	Data Size	Constraint	Key Constraint
ACCOUNT_ID	Integer	11	NOT NULL	PRIMARY
ACCOUNT_NAME	String	50	NOT NULL	
ACCOUNT_USERNAME	String	50	NOT NULL	
ACCOUNT_PASSWORD	Text	Text	NOT NULL	
ACCOUNT_TYPE	String	30	NOT NULL	
EMPID	Integer	11	NOT NULL	
ACC_STATUS	String	10	NOT NULL	

Table 18: User data dictionary

Attribute	Data type	Data Size	Constraint	Key Constraint
id	Integer	10	NOT NULL	PRIMARY
Name	String	50	NOT NULL	
Username	String	50	NOT NULL	
Password	String	50	NOT NULL	
Type	Integer	11	NOT NULL	
Status	Integer	11	NOT NULL	

4.3. Dynamic Model

A dynamic model represents the behavior of an object over time. It is used where the object's behavior is best described as a set of states that occur in a defined sequence.

4.3.1. Sequence Diagram

A sequence diagram in a unified modeling language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. The sequence diagrams for our automated computer laboratory monitoring and management system is clearly show the participating objects in the given use case as follows.

Sequence Diagram Notations

Actor: Represents a type of role where it interacts with the system and objects.

Lifelines: A lifeline is which depicts an individual participant in sequence diagram. It is located at the top of sequence diagram.

Messages: Communication between objects is depicted using messages. The message appears in sequential order on the lifeline. We represent the messages using arrows. There are many categories of messages: these are synchronous message, asynchronous message, create message, delete message, self-message, replay message and etc.

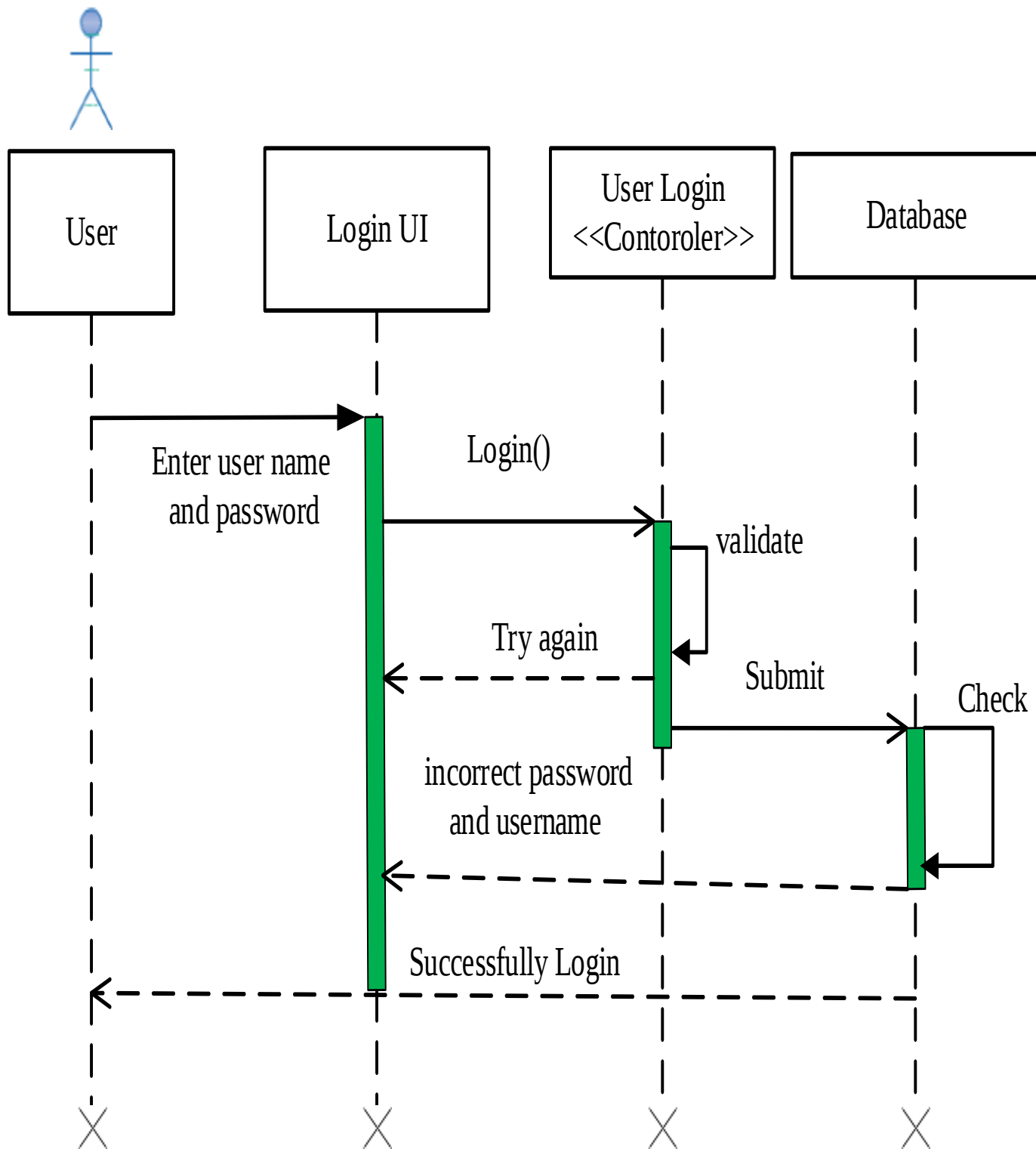


Figure 4: Sequence diagram for Login

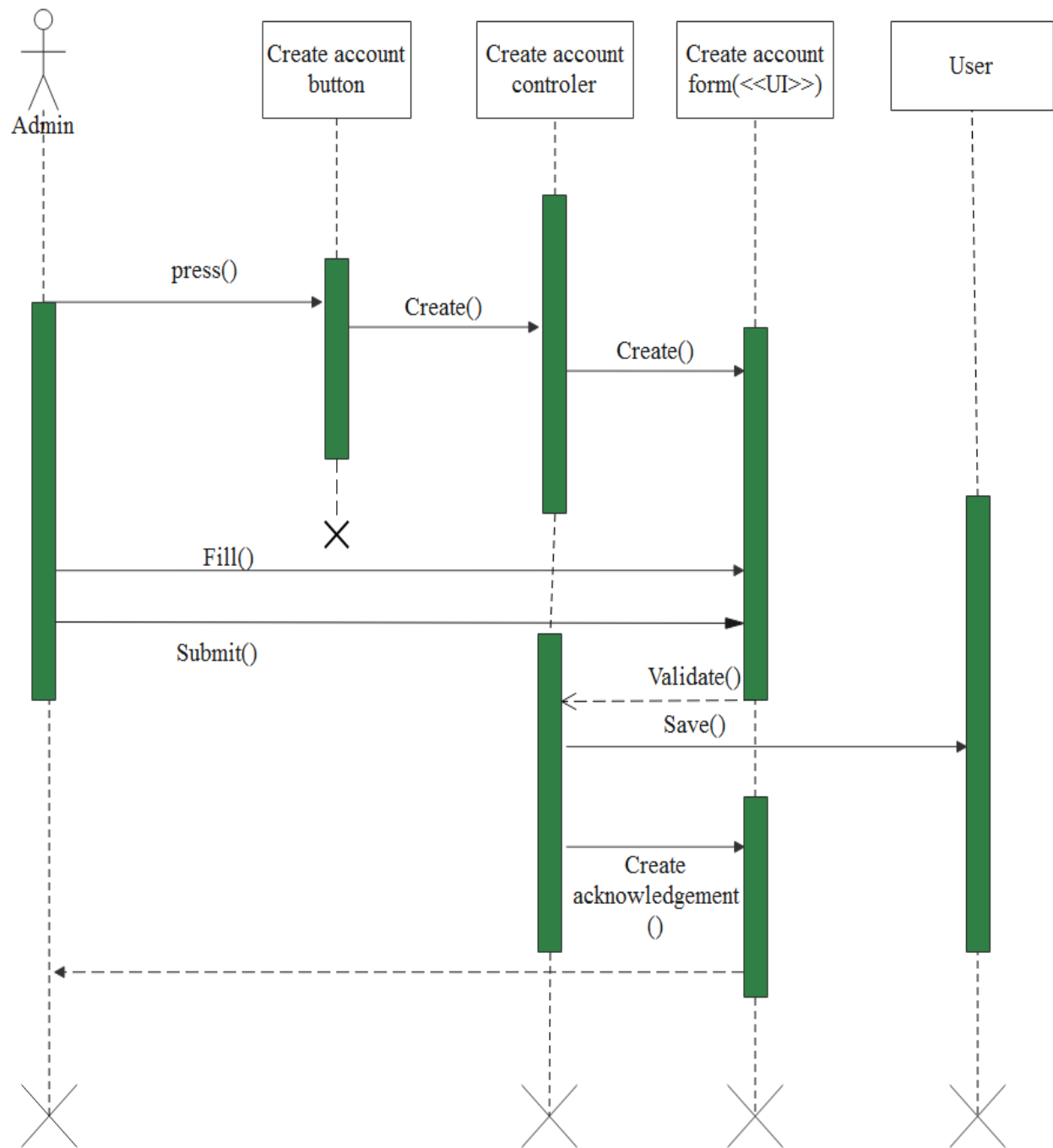


Figure 5: Sequence diagram for Create account

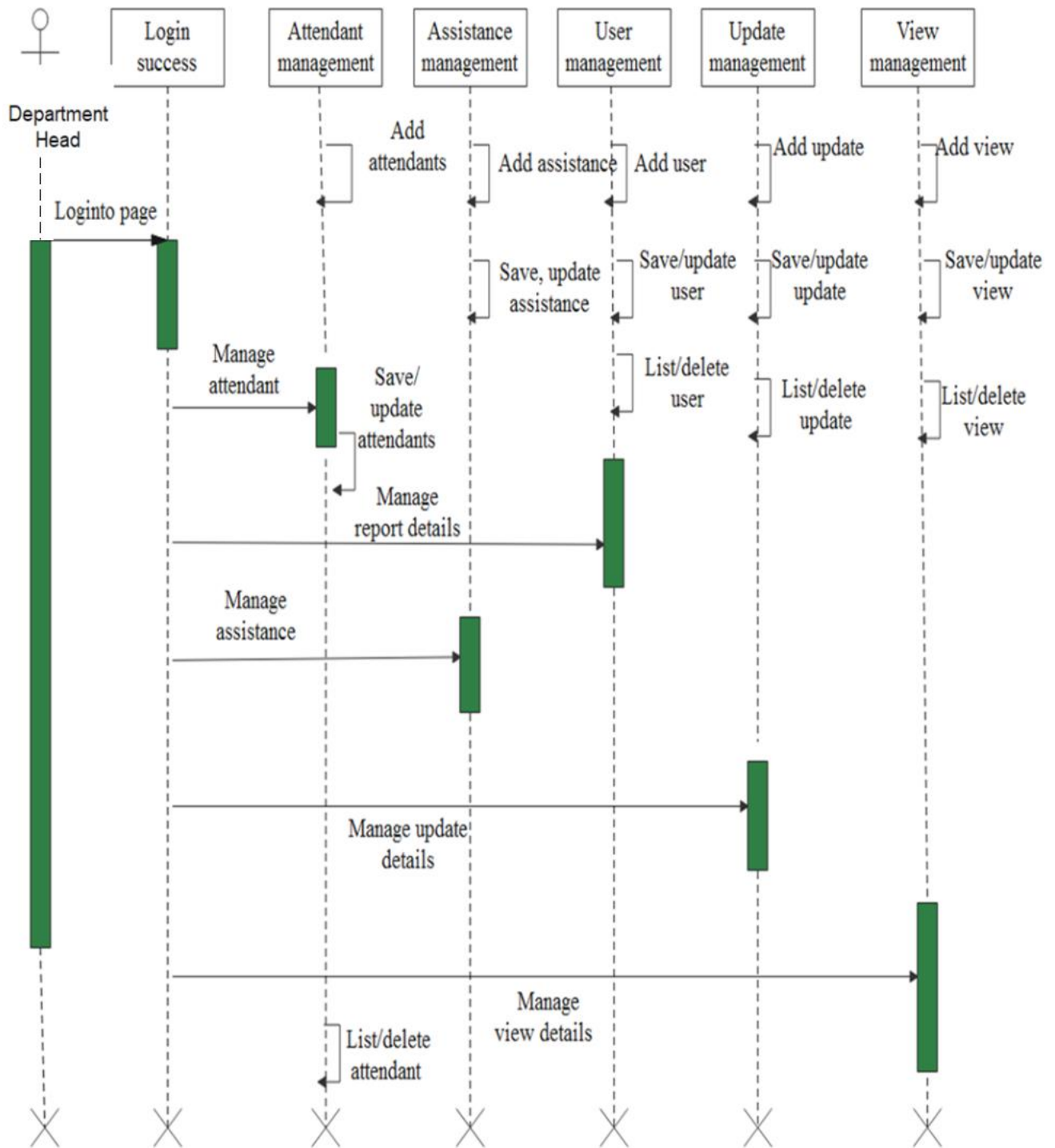


Figure 6: Sequence diagram for manage Employee and user

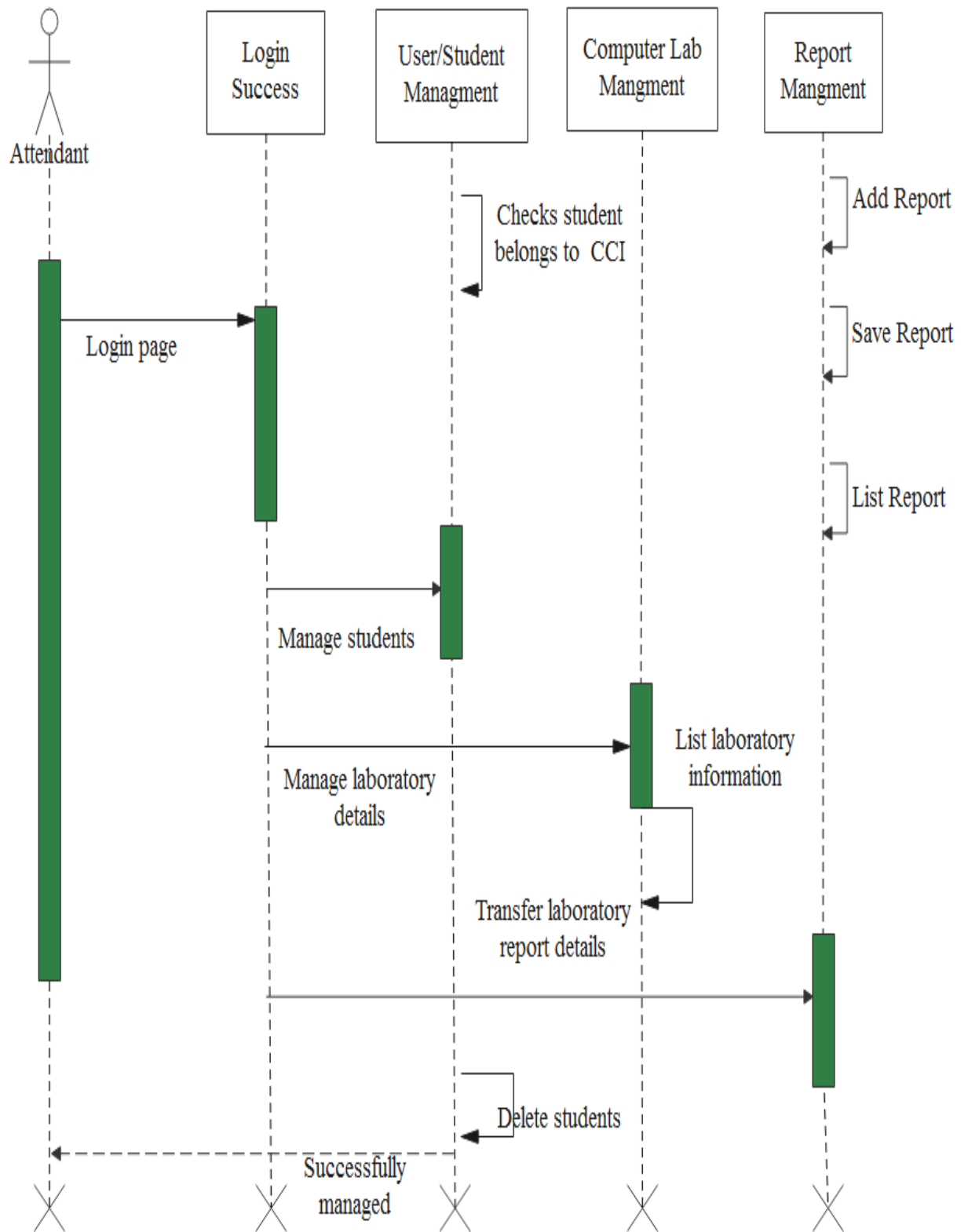


Figure 7: Sequence diagram for attendants manage laboratory and students/user

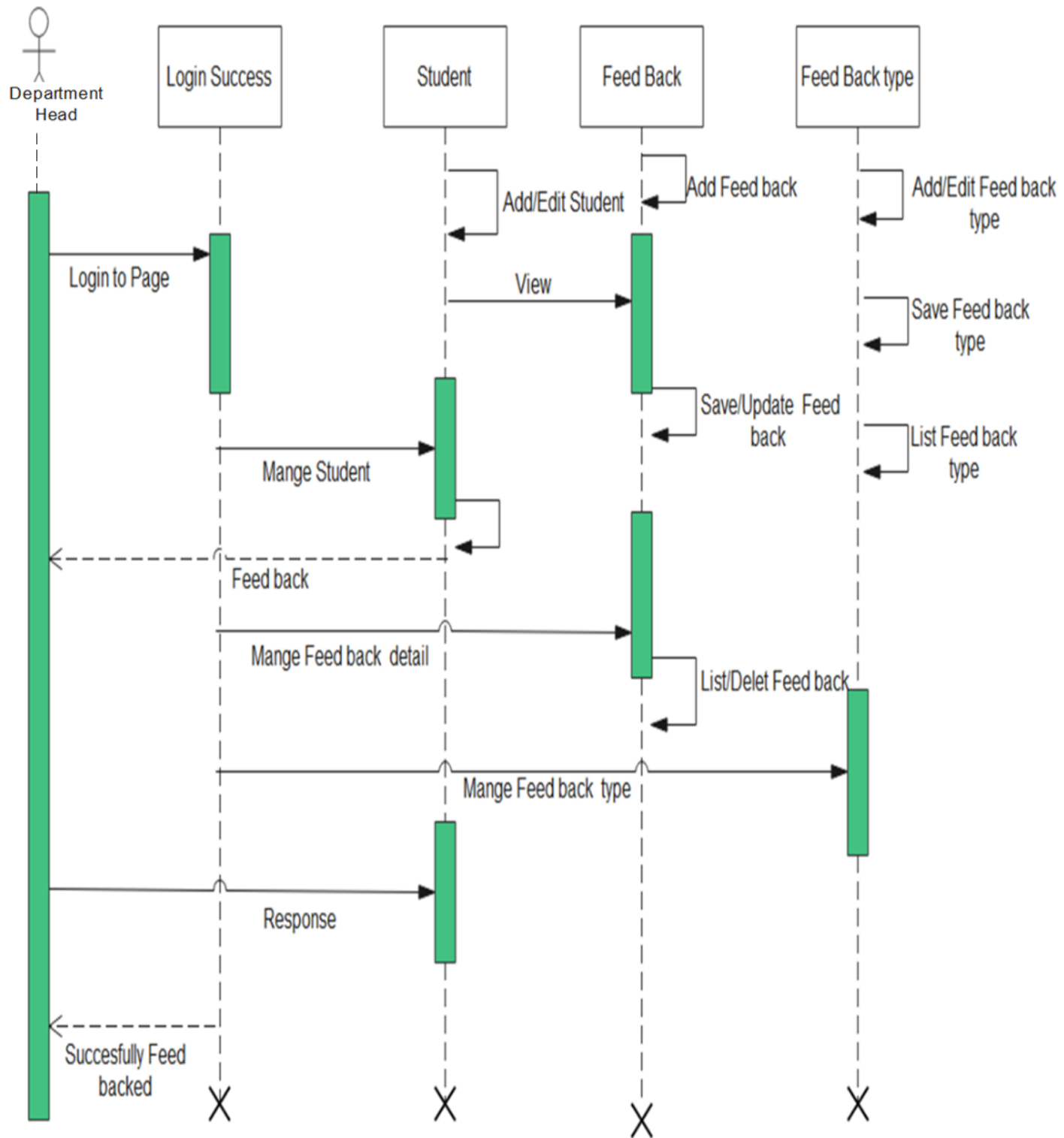


Figure 8: Sequence diagram for feed back

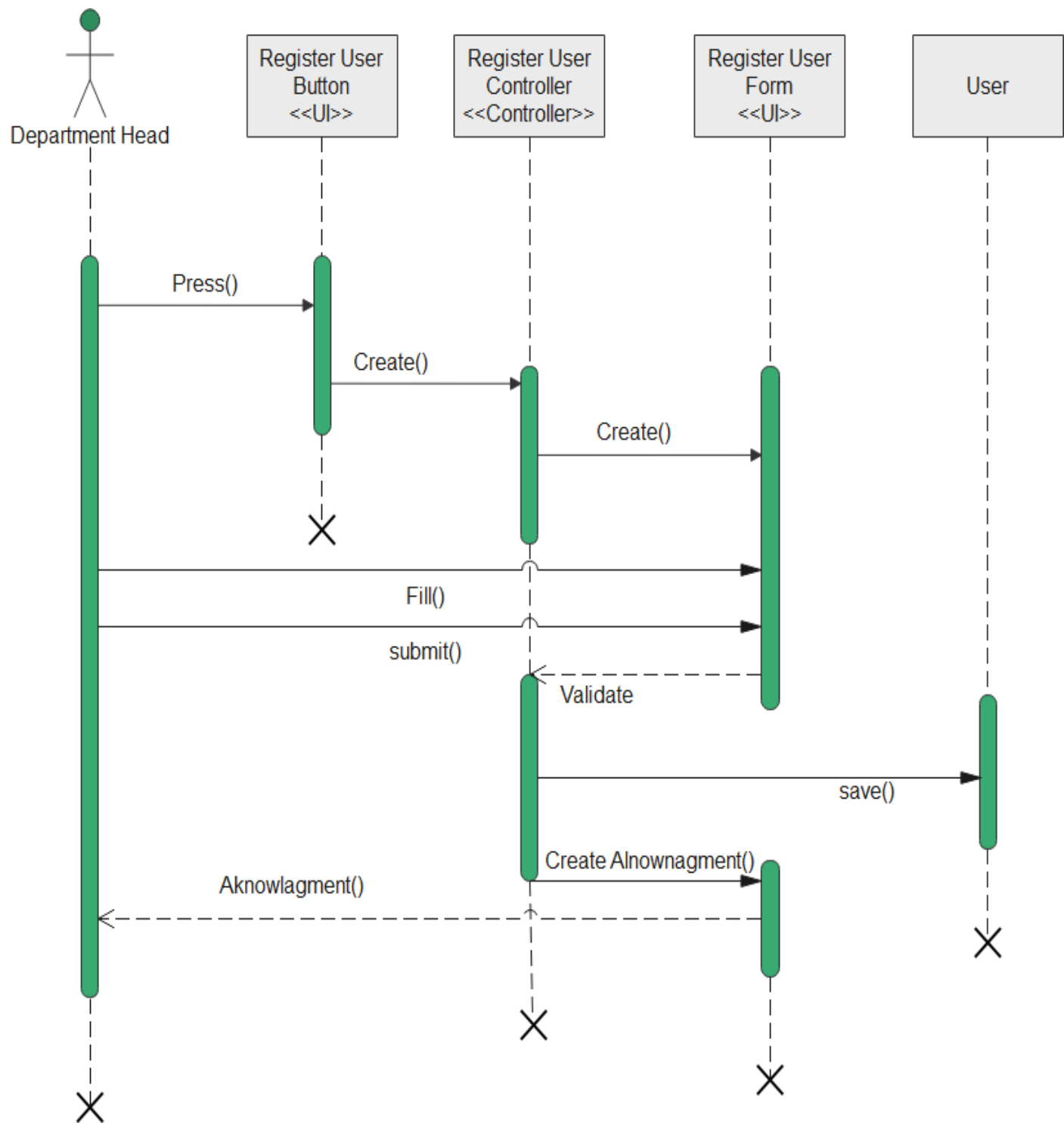


Figure 9: Sequence diagram for registration of user

4.3.2. Activity Diagram

Activity diagram describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. Figures that are depicted below are our system activity diagram.

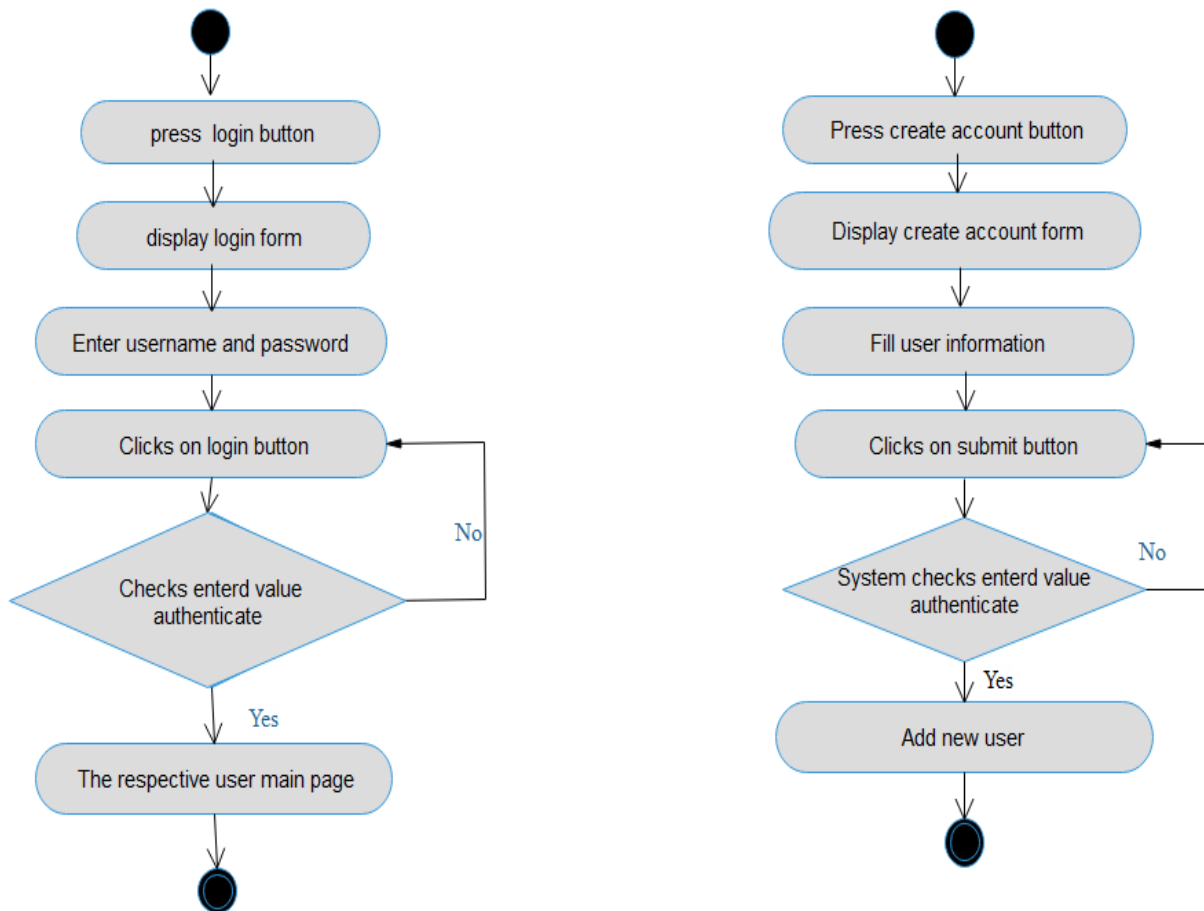


Figure 10: Activity Diagram for Login and Create Account

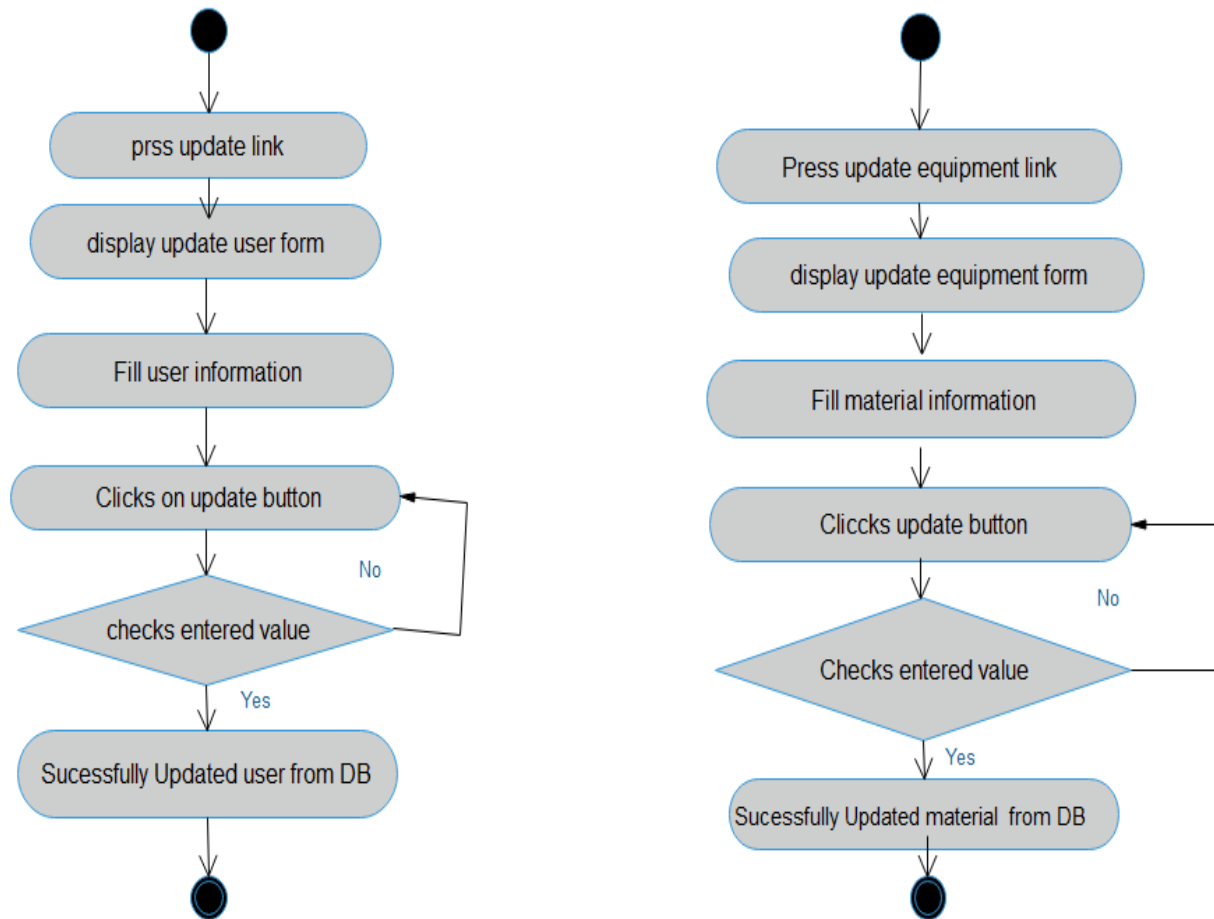


Figure 11: Activity Diagram for update user and equipment

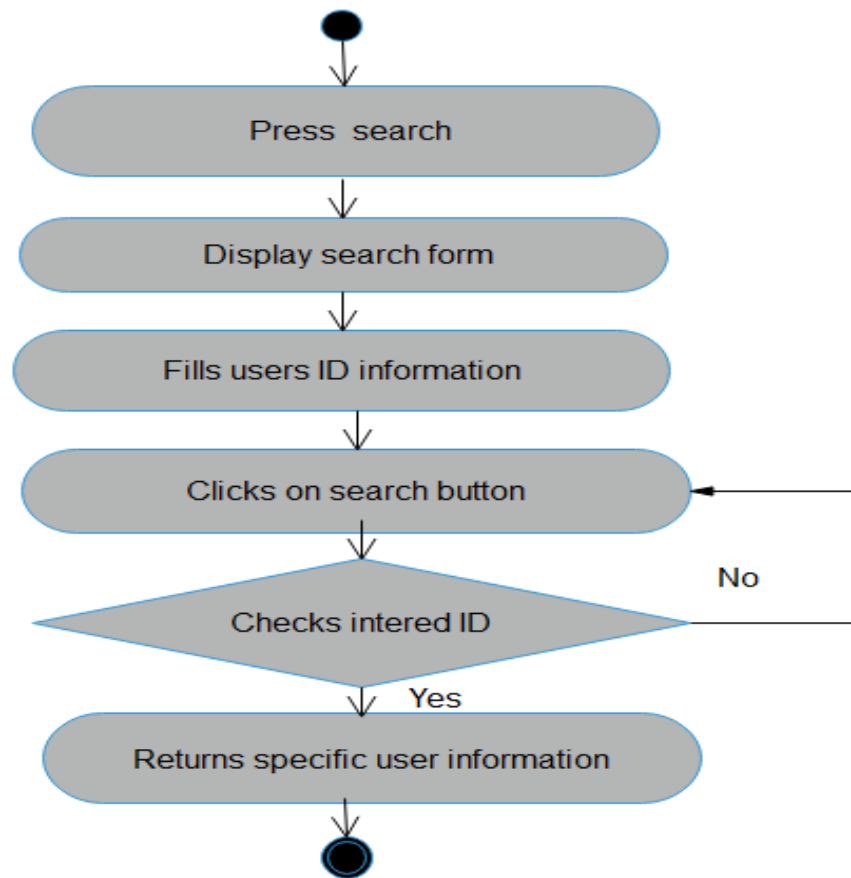


Figure 12: Activity Diagram for search user

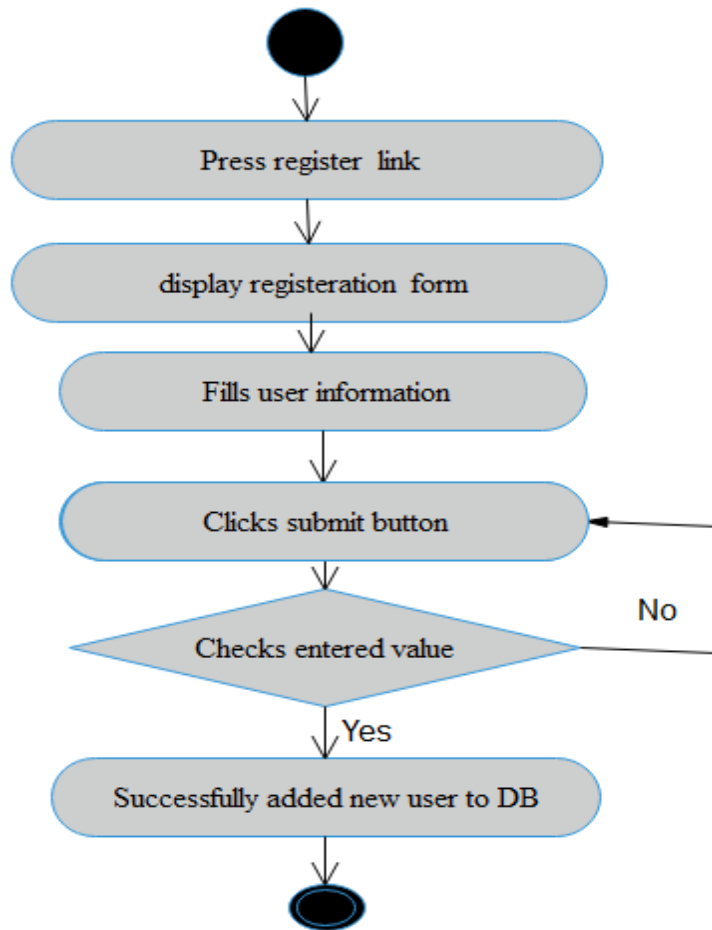


Figure 13: Activity Diagram for Registration of user

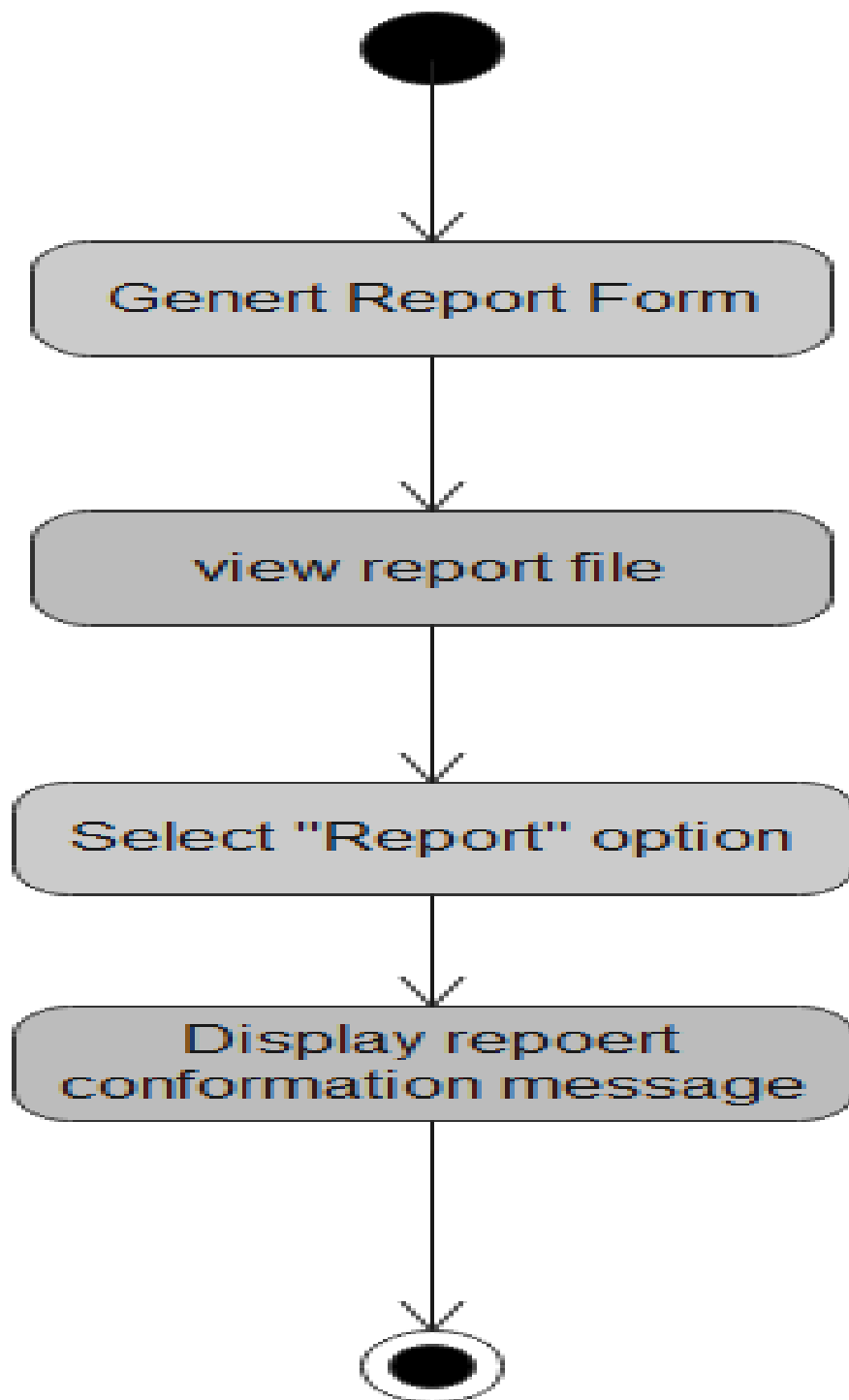


Figure 14: Activity Diagram for generate report

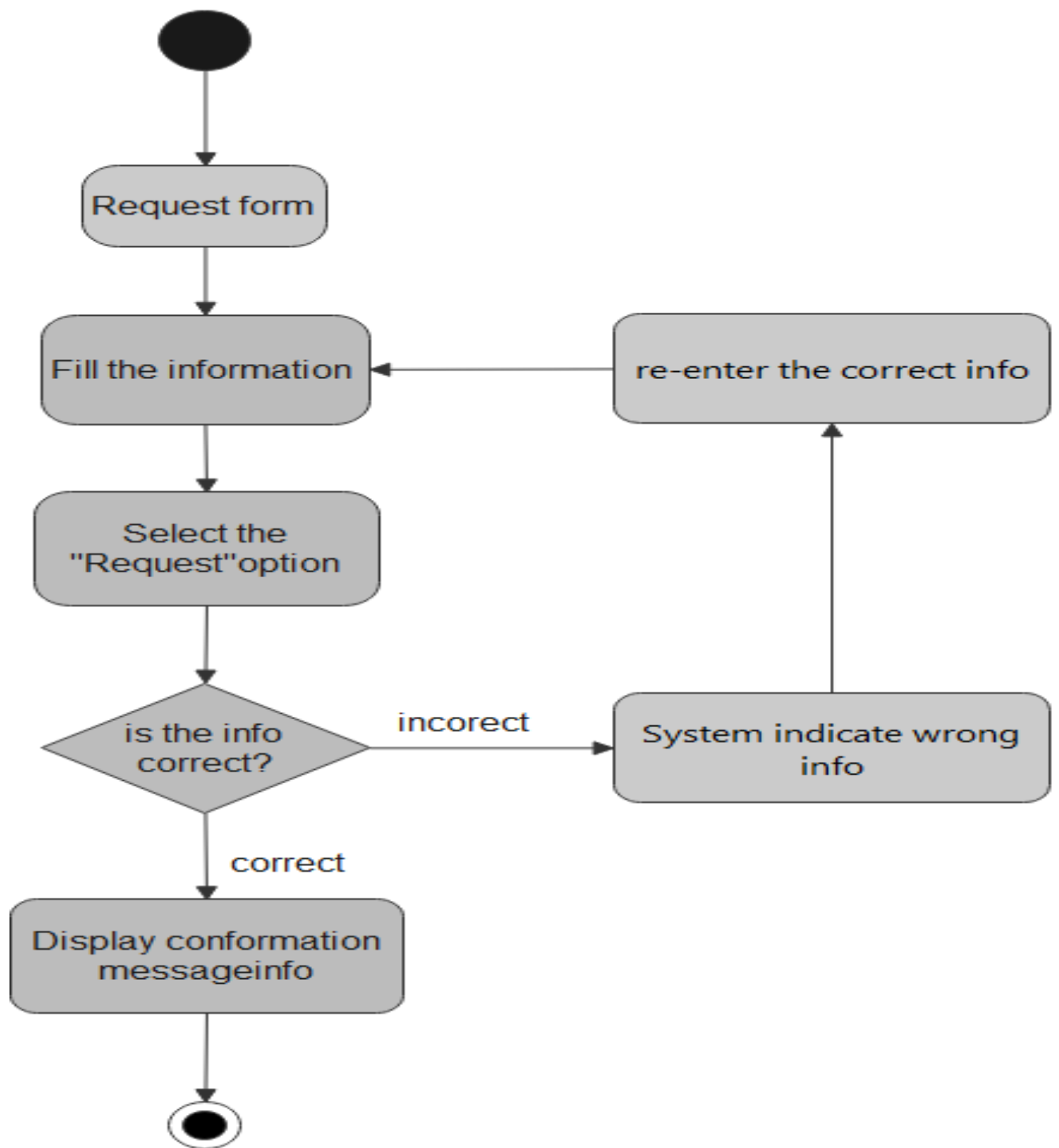


Figure 15: Activity diagram for request

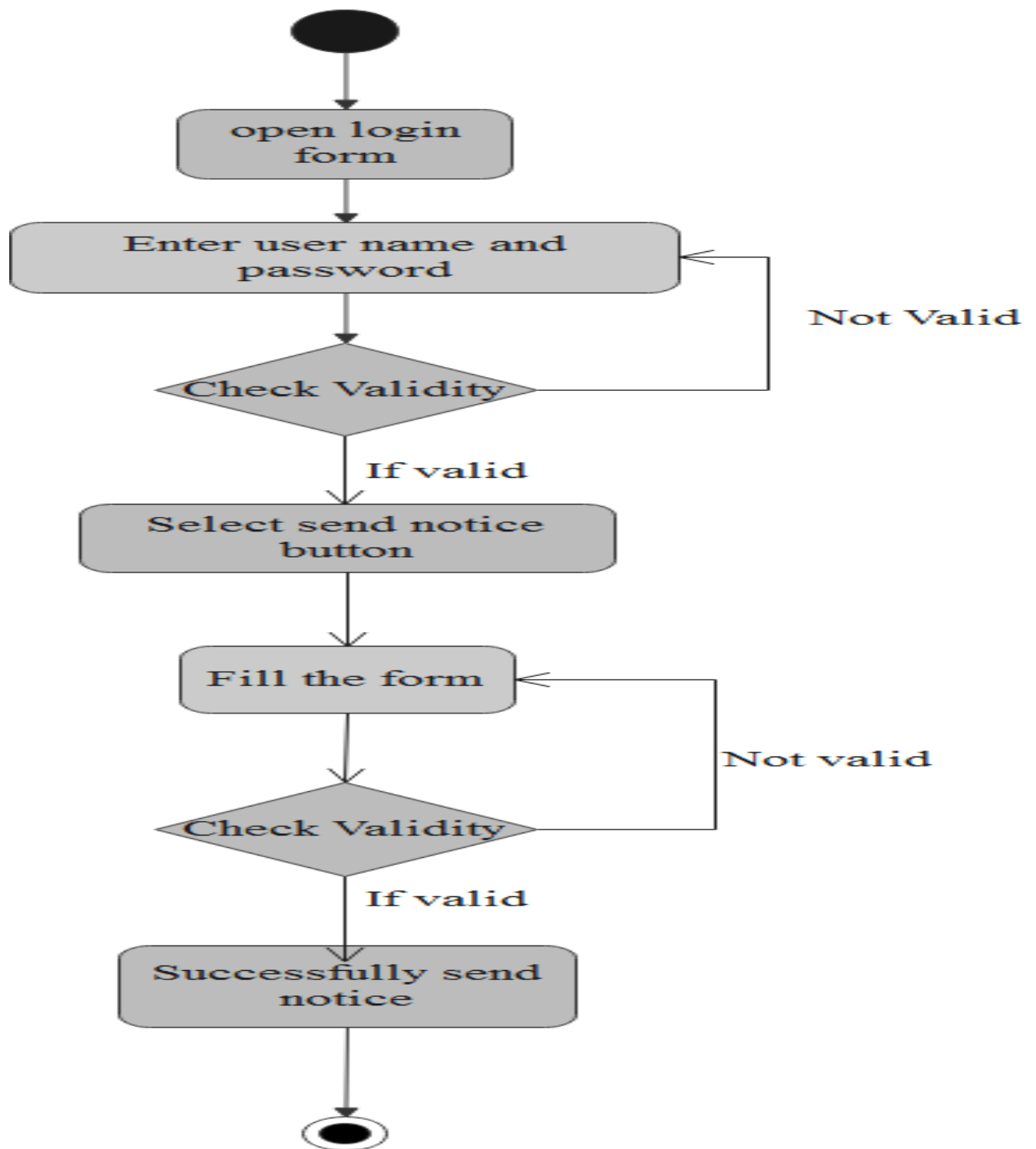


Figure 16: Activity diagram for send notice

4.3.3. State Chart Diagram

State chart diagram is used to describe the states of different objects in its life cycle. So the emphasis is given on the state changes upon some internal or external events. These states of objects are important to analyze and implement them accurately. The common model elements that state chart diagrams contain are:

- States
- Start and end states
- Transitions

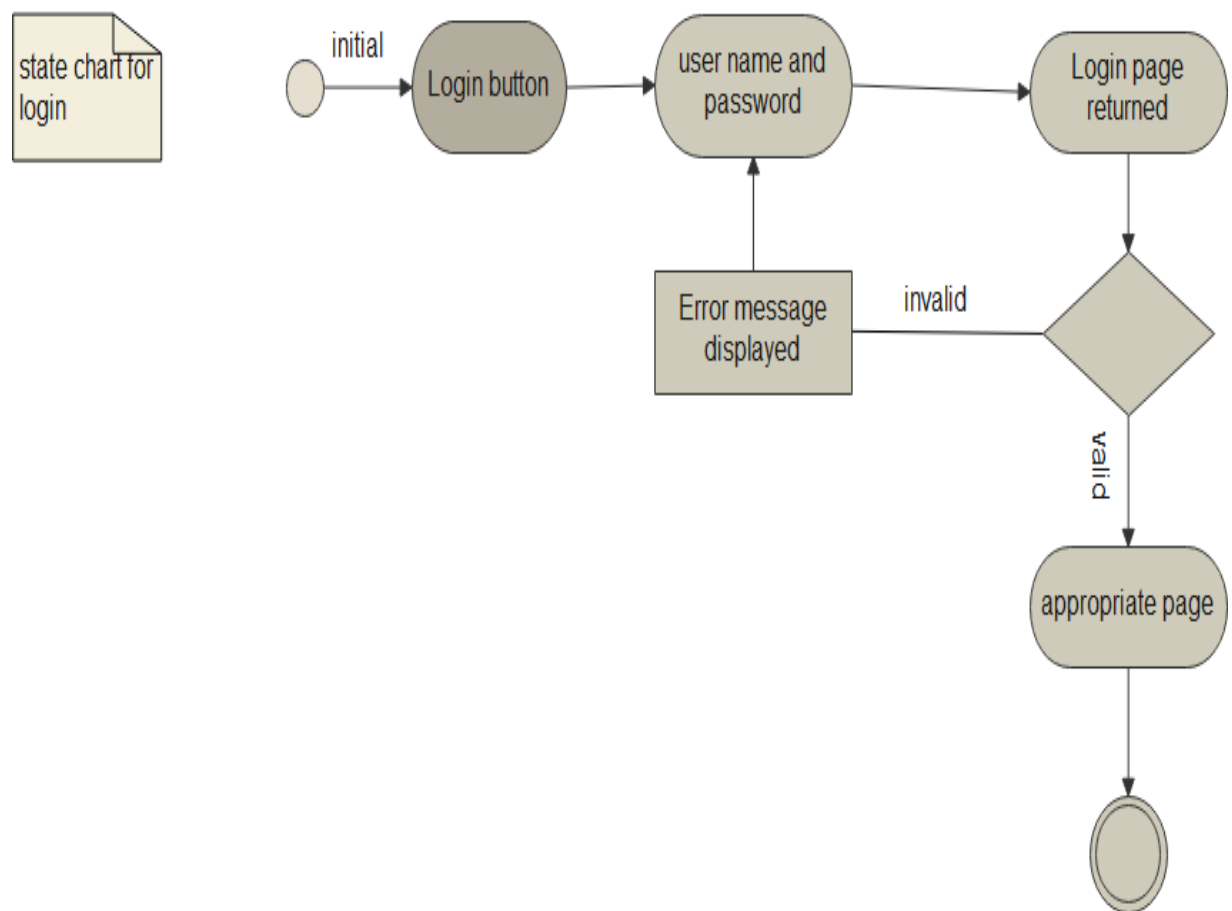


Figure 17: UML state chart diagram for login form

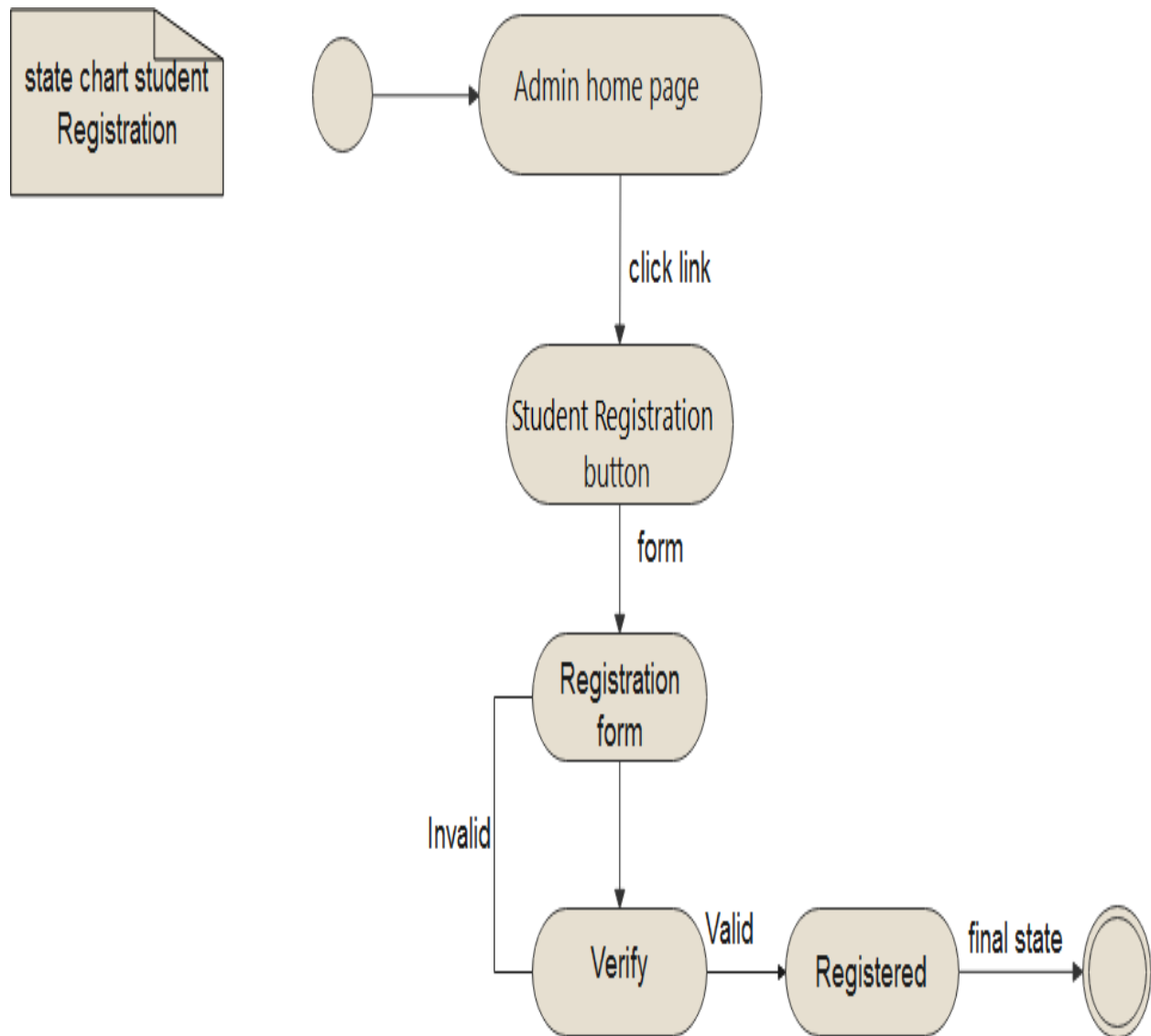


Figure 18: UML State Diagram Registrations

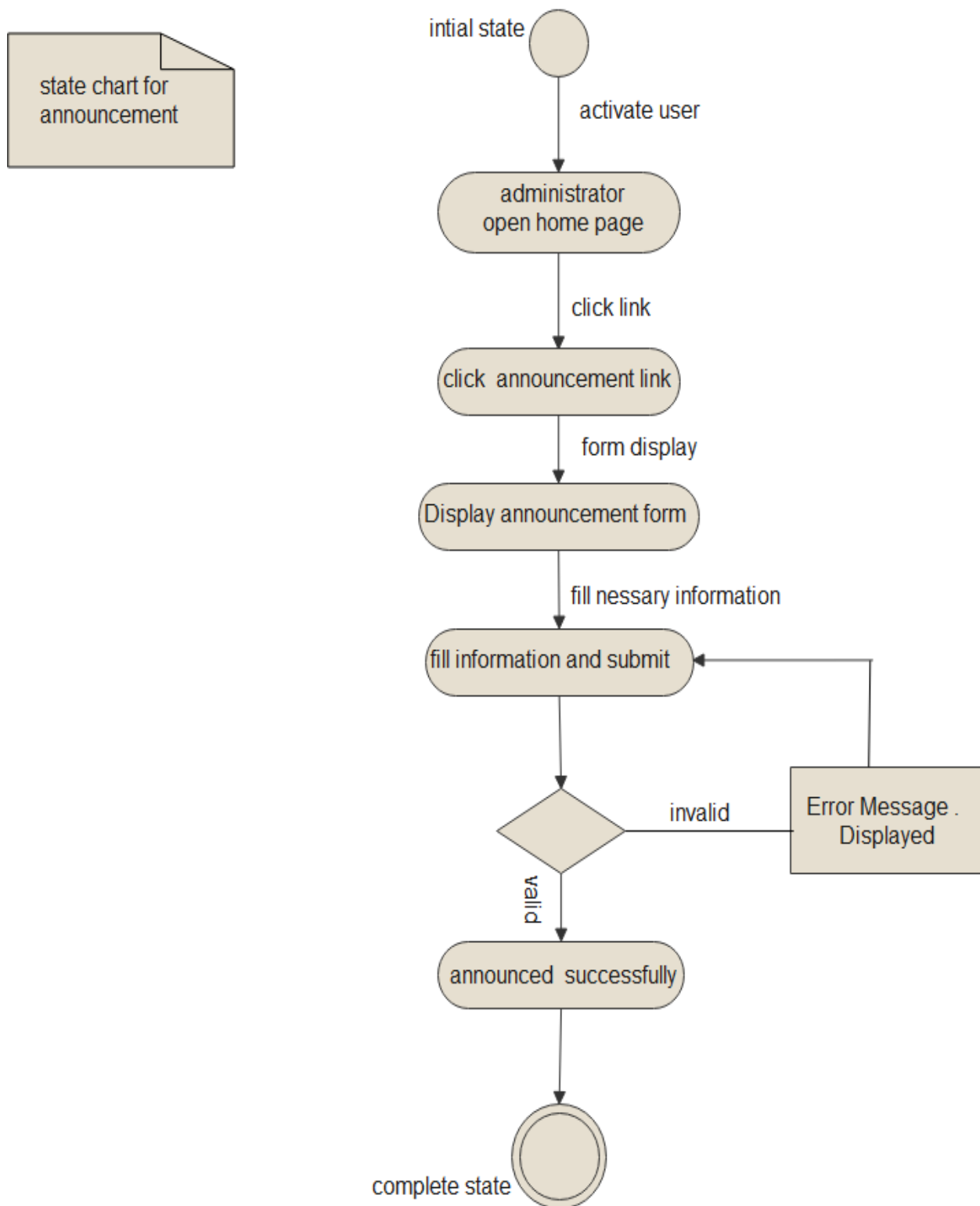


Figure 19: UML state chart diagram for announcement

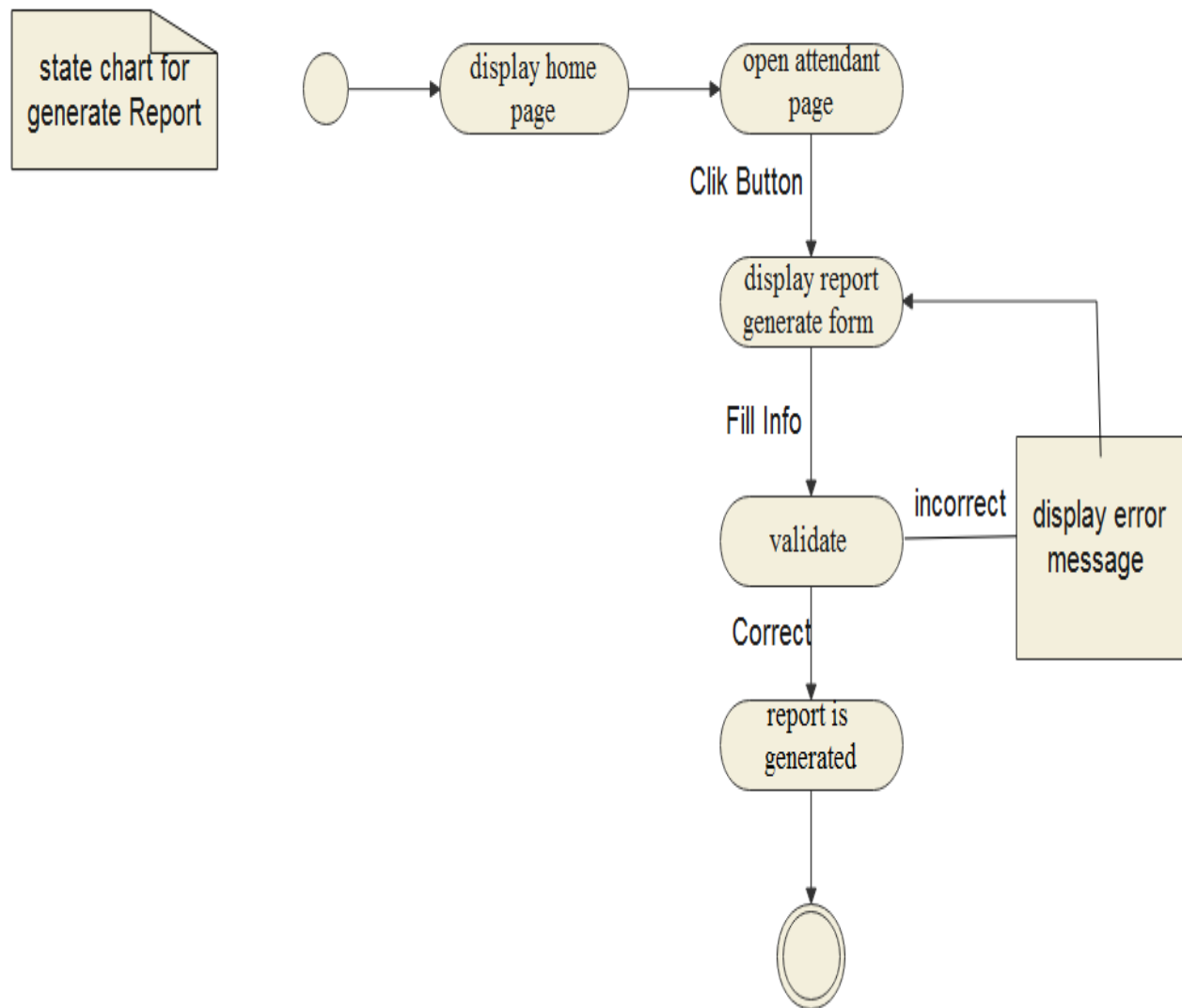


Figure 20: UML state chart diagram for Generate report

CHAPTER FIVE

SYSTEM DESIGN

5.1. System Overview

System design is the process of defining the components, modules, interfaces, and data for a system to satisfy specified requirements. Also system design is the process of creating or altering systems, along with the processes, practices, models, and methodologies used to develop them. Our system design is proposed solution for existing system problem. Our proposed system is divided into different categories in order to simplify the complexity of system when we providing a solution for the specific problem (3).

The following topics are included in this chapter

- ✚ Design goals
- ✚ System architecture
- ✚ Sub-system decomposition
- ✚ Hardware or Software mapping
- ✚ Persistent data storage and management

5.2. Design goals

The goal of the system is to minimize the time and resource loss for different application that was applied when the Wolkite University collages of computing and informatics workers and user done their work. This is also help to facilitate the work to accomplish specific tasks in short period of time i.e. a small number of people can do more efficient work in a limited period of time and resource. This design goals are derived from non-functional requirements of description the feature characteristics and attribute of the system as well as any constraints that may limit the boundary of the proposed solution. It is used to identify the quality that our system should focus on and to manage complexity by dividing the system into smaller manageable pieces. The objectives of design are to model the system with high quality of service to the system users. The following are the main goals of the proposed system: -

5.2.1 Performance and portability

Our system has best performance. It has fast speed when any user is using it in order to do privileged thing, also it has good responsiveness and stability. This leads to user satisfaction in our system because they will not annoy and get bothered when using our system rather they will do what they want. Also our system can be used in any operating systems other than the one in which it was created (i.e. windows) without requiring major rework so that it can run in the new environment.

5.2.2 Maintenance

The system is better and is not highly suffered by damages therefore there is no need to think about its maintenance features simply the database /system administer can maintain it if it fails in some extent. It needs Administrators that manage the database and system aspects. It does not need expertise person instead it needs some professional person in using of computers who can use database and internet very well.

5.2.3 End user

This system has a well-defined and easily understood interface and the processes is easy to understand and useable by users at any level. Anyone who can read English can use the system, because, to use the system only navigating through the system parts by clicking a link is the only required thing from the system users.

5.2.4 Dependability

Our system needs to be highly dependable. Our system is robust and fault tolerant. Our system is handling sensitive data of the university, high emphasis given with regards to security. The proposed system should achieve the following dependability characteristics:

- Security: the system should be secured, i.e., not allow unauthorized users to access the database system.
- 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.
- Robustness: Since the system is a web-based system that mainly uses a menu driven access there would not be an input problem by the user side. But for the server side there might be an error during the process

5.2.5 Priorities of Design Goal

Performance vs. portability

The system selects performance compares to portability but this does not mean our system perform its task with only large sized devices system. It has fast speed when any user is using it in order to do privileged thing, also it has good responsiveness and stability. Also it performs its operation without depending specific type of operating system as a result of this users can install the system any place where ever it needed without bother a kind of plate form, they going to use.

Functionality vs. Usability

The system favors usability compared to functionality yet it doesn't mean that the system loses its proper functionality. Rather the developing team will endeavor to maintain the basic requirements of the system along with high usability. The system will avoid lavish functionalities, which are not a must for the system's existence, for the sake of encouraging basic system utilization

Security vs. Usability

Possible have lots of reason for selecting security instead of usability but again our system considers usability besides security. Relating security issues our system will drown line for each stakeholder which functionality of the system need to access by authenticate finally, record each users of the systems' information including the task they have done on that specific date, also securing data during transfer is one of task our system handles additional to others security matter.

5.3. Current System Architecture

Since system architecture describes conceptual model that defines the structure, behavior, and more views of a system. Currently no any system architecture found in our university related to computer lab monitoring and management system that we cannot able to model the structure, behavior and more views.

5.4. Architecture of the proposed System

The proposed software has three layers of architecture. The layers are the following: -

-  The application layer
-  Logic/ connector layer
-  Database layer

Application layer: Layer which provides graphical user interface and application specific entry forms to the user of the system. Application layer interact with logical layer through http protocol.

Logic layer: Layer that used to implement business rules and data rules, which keep the data structure consistent.

Database layer: This actual DBMS layer. Used to assist resource sharing and allow the client to be configured with the help of ODBC driver.

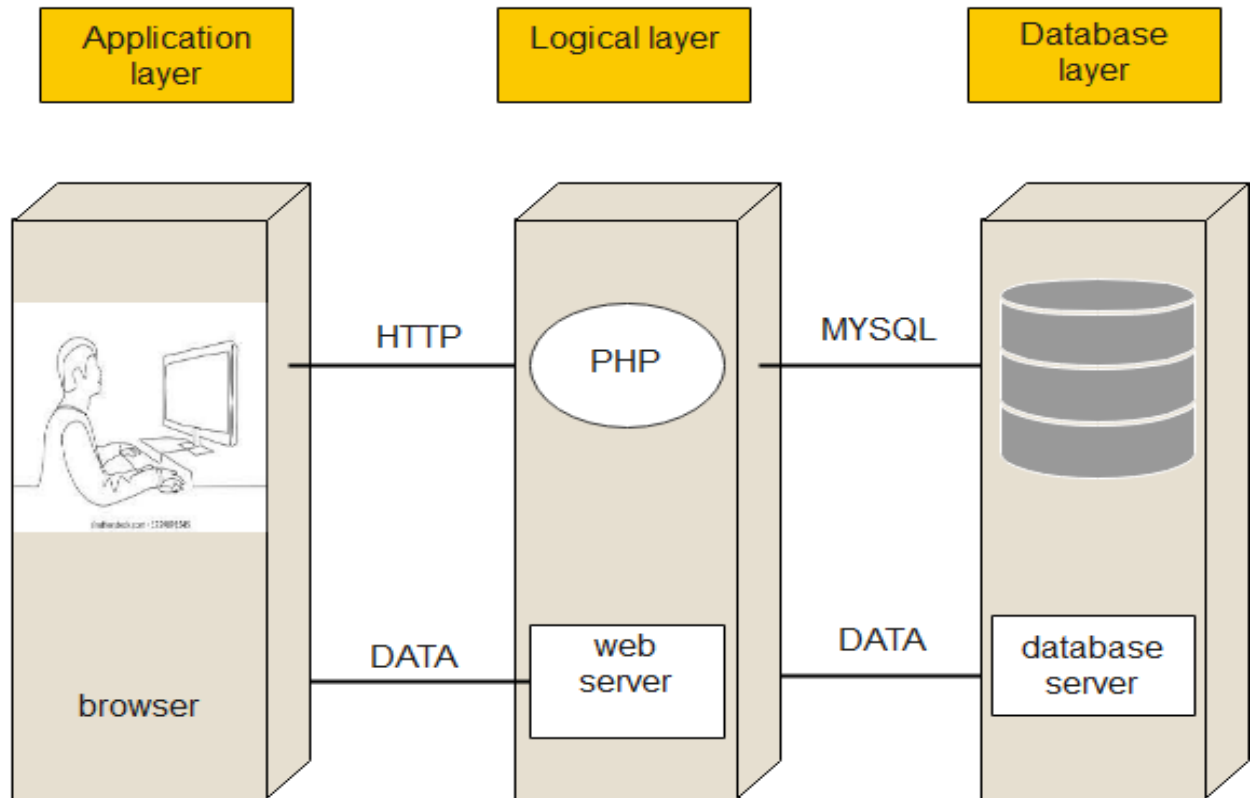


Figure 21: System Architecture Diagram of CLMMCCI

The above diagram shows the architecture of the proposed system is three tiers. The application layer communicates with the logic layer through HTTPS protocol to get service to their request from web server. ODBC is used as bridge to access data from the database layer to logic layer. Since we are using to PHP to implement the server side script that we use ODBC to connect it with the database server.

5.4.1. Sub-system decomposition

During system decomposition of computer laboratory monitoring and management system, we have just divided the general system in to smaller subsystem with strong coherence. The main purpose of the subsystem decomposition is reducing the complexity of the system. The following are sub systems.

✚ **Administrator subsystem:** this subsystem describes admin in our system it includes the operations of:

- ❖ Login
- ❖ Create account
- ❖ Update user
- ❖ Delete Account
- ❖ Log out

✚ **Department Head subsystem:**

- ❖ Login
- ❖ Replay request
- ❖ Delete user
- ❖ Search user
- ❖ Monitor lab equipment
- ❖ Manage attendant
- ❖ Register student and customer
- ❖ View report
- ❖ Post notice
- ❖ Logout

✚ **Attendant subsystem:** this subsystem handles manage and monitor computer and it includes operations of:

- ❖ Login
- ❖ Manage laboratory
- ❖ Manage Student
- ❖ Monitor computer
- ❖ Receive computer
- ❖ Report generate
- ❖ Logout

✚ **Assistance subsystem:** this subsystem handles of assist students and it includes operations of:

- ❖ Login
- ❖ Manage Student
- ❖ Tech technical LAB
- ❖ Make contact with Las
- ❖ Logout

✚ **Student subsystem:** this subsystem handles use of computer lab and includes operations of:

- ❖ Login
- ❖ Use computer lab
- ❖ Send Request
- ❖ Logout

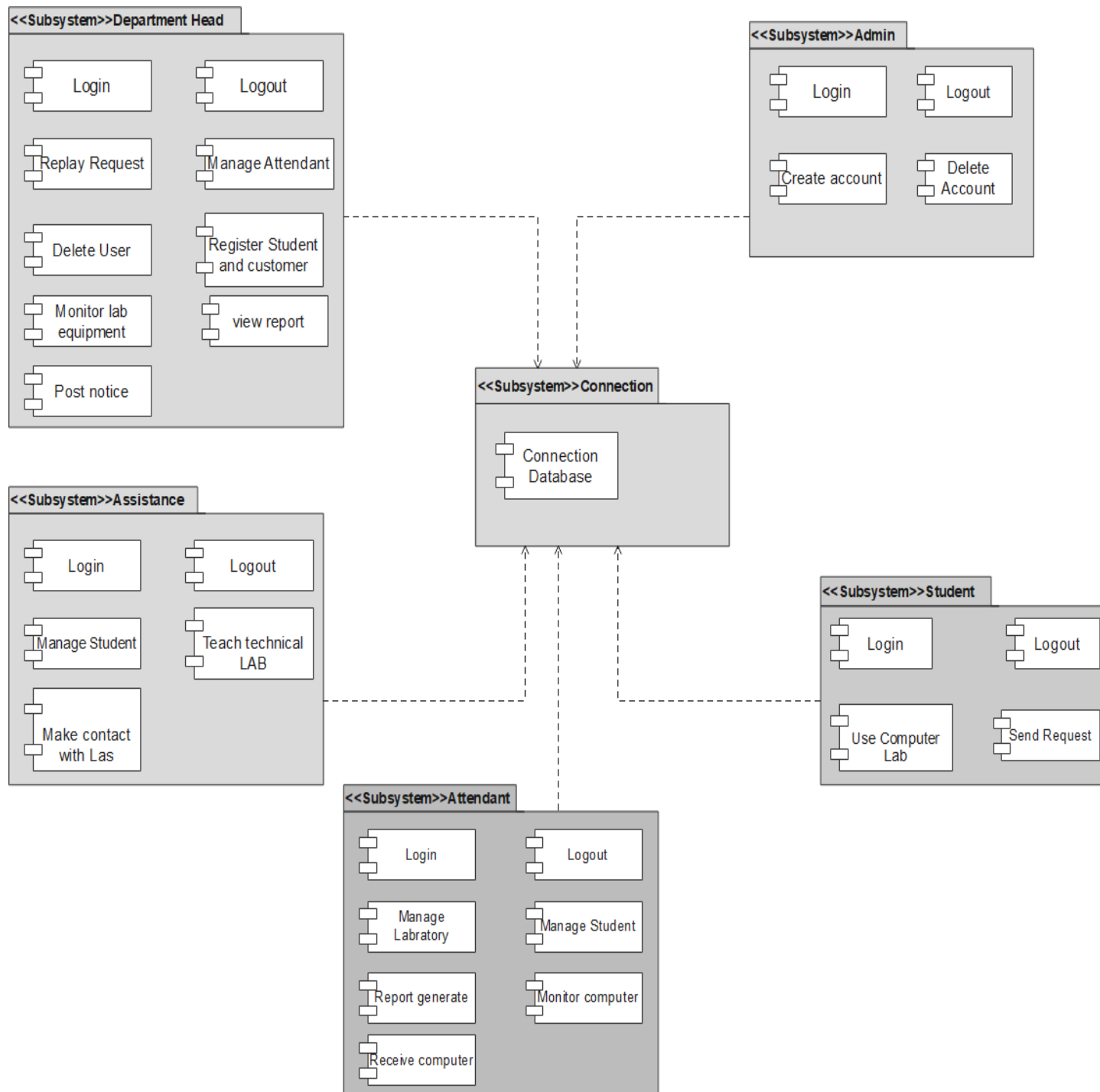


Figure 22:Sub-system decomposition

5.4.2. Hardware/Software mapping (Deployment design)

The system is built as a client-server application and communicates any other computers. The system is dependent component that runs on any computer node. UML deployment diagram illustrates the hardware /software mapping of the system Deployment diagram is implementation type diagram that describes the physical architecture of the hardware and software of the system. It is concerned with the physical deployment of the application. This includes the network layout and location of the components on the network. A deployment modeling depicts a static view of the run time configuration of process nodes and the components that run on these nodes. It shows:

-  The hardware for the system
-  The software that installed on the hardware
-  Middle ware used to connect different machines to one another

For our system we have chosen to adopt a three tires client-server approach in the deployment diagram. The tier are the user interface tier, an application server tier and database server tier.

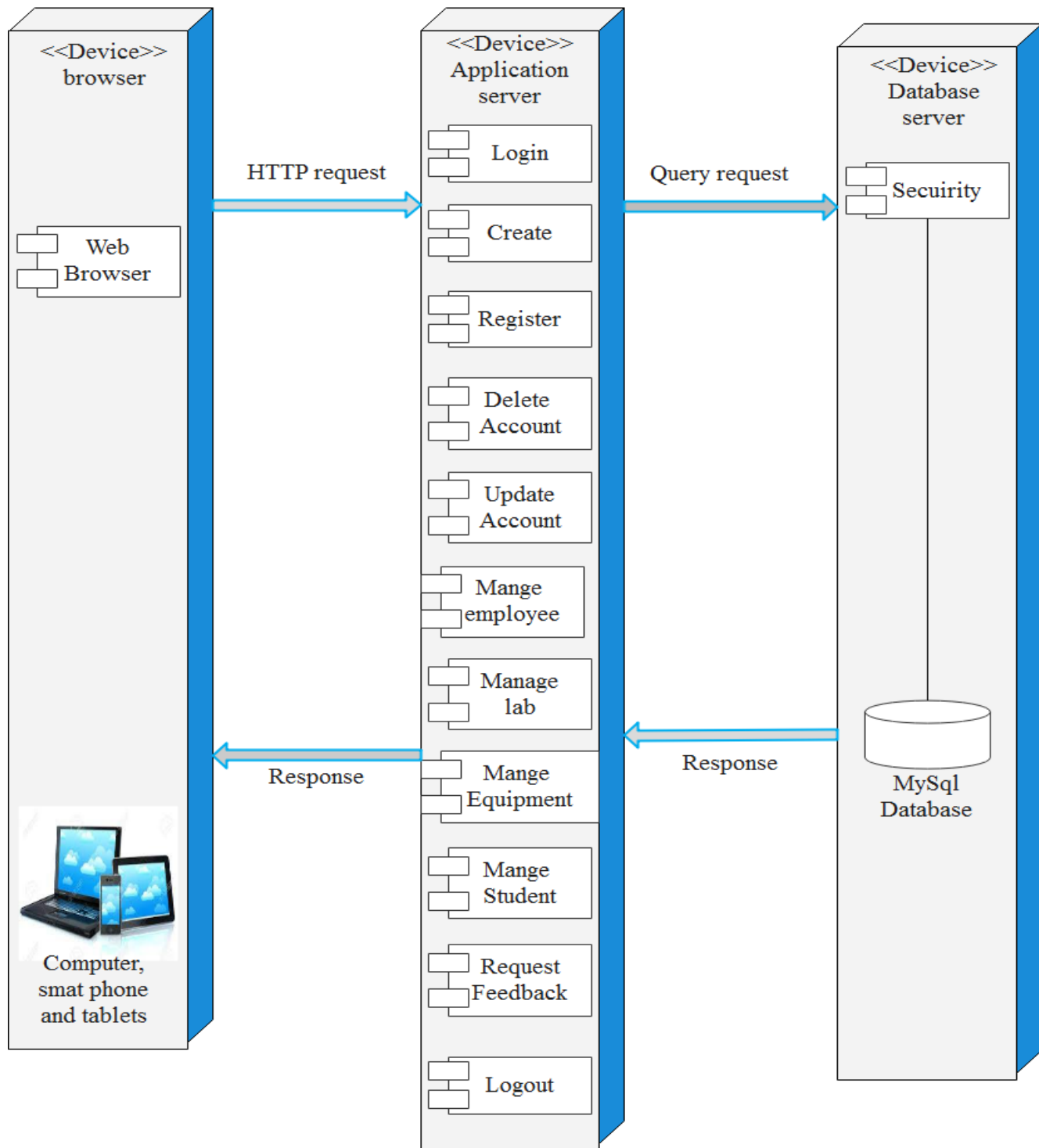


Figure 23: Hardware/Software mapping (Deployment design)

5.4.3. Detailed Class Diagram

Detail class diagram is class diagram that include attributes, methods, attribute data types, visibility of attributes and methods, inheritance, association, aggregation, composition, dependency, and municipality (cardinality and optimality).

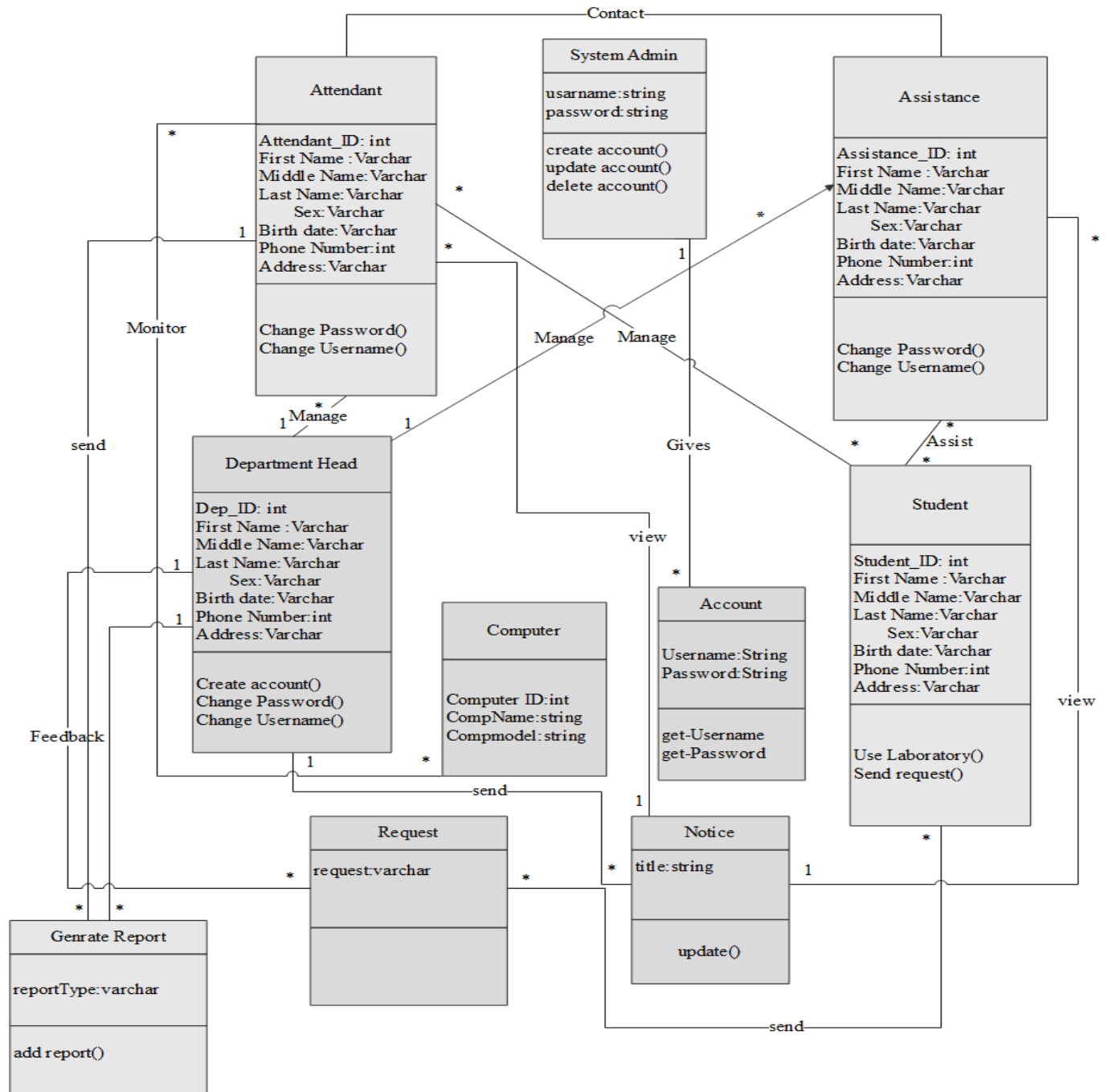


Figure 24: Detailed class diagrams

5.4.5. Persistent data storage and management

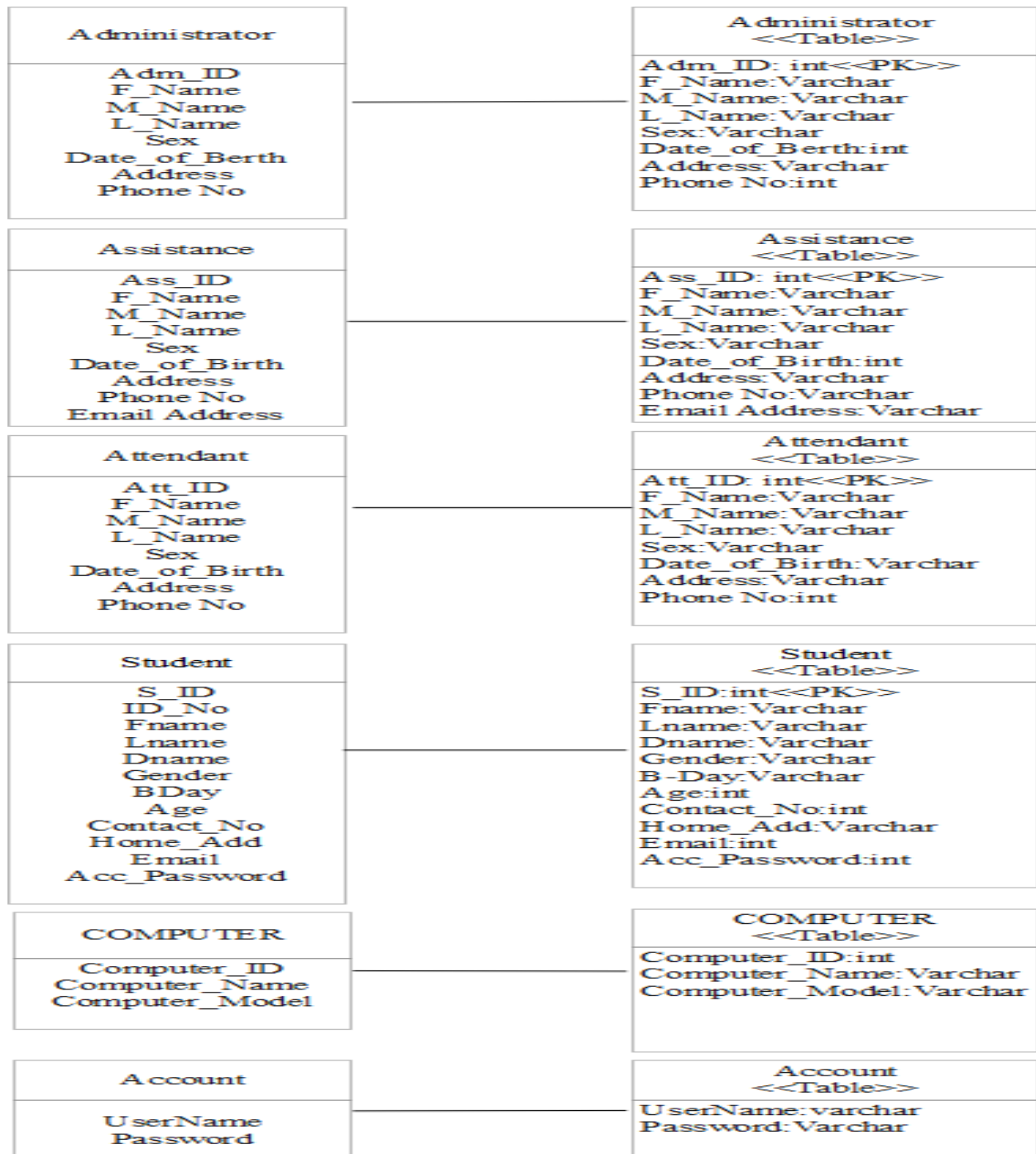


Figure 25: persistence diagram Computer laboratory monitoring and management system

5.4.6. Access Control and Security

In our project, different actors have access to different information and data. Access control and security specifies what the user can access or what cannot perform by some users. This access control is verified by ID and password. System administrator represents an authenticated user. In our system only authenticated user can access our system and only permitted user can use our system. The proposed system follows multi user system. In multi user system, different actors have access to different functionality and data. Then it must be having: -

- ✚ Confidentiality: Only authorized person can see the information. Private data is kept private; personal privacy is respected.
- ✚ Integrity: There are limits on who can change the data in this system.
- ✚ Availability: The system is available at all times to authorized users.

Actors	Operation			
	Manage account	Manage equipment	View	Request
Admin	-update -delete -search -create -register	-update -view -delete -search -add	-damaged computer	
Attendants		-search -view	-Announcement	
Assistants	-Update		-Assisting student --Announcement	
Students				Feedback request to admin

5.5 Packages

Package is an organized and functionality-based set of related interfaces and classes. Packages organize classes that belong to the same category or provide similar functionality. We categorized our system into laboratory management package, database management package, notification management package, account management package, security management package and report management package.

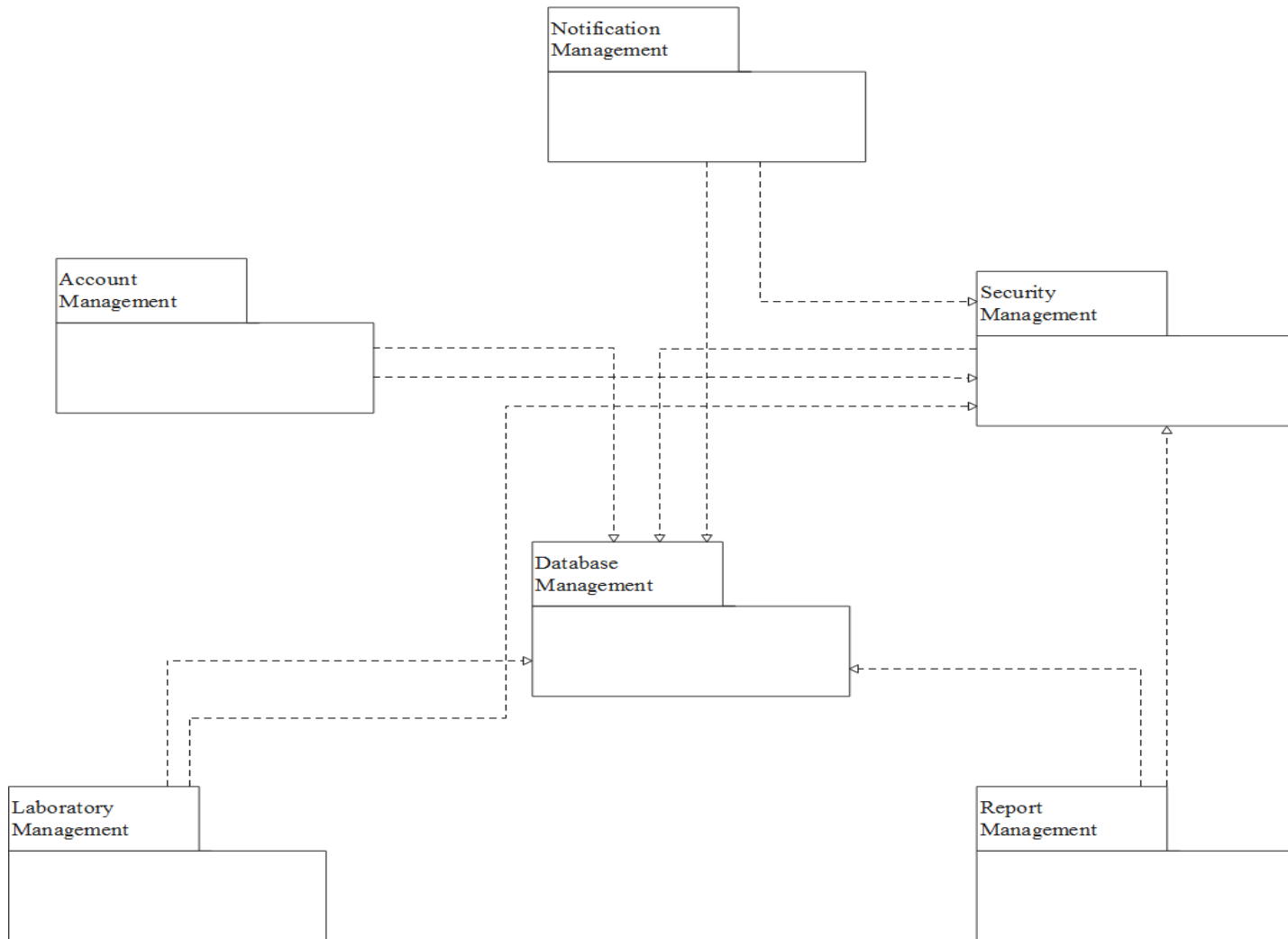


Figure 26: Package Diagram

5.6. User interface design

User interface design is the overall process of designing how a user will be able to interact with a system. The goal of user interface design is to make the user's interaction as simple and efficient as possible.



Figure 27: homepage interface



Figure 28: login interface

CHAPTER SIX

6. IMPLEMENTATION AND TESTING

6.1. Implementation of the Database

The implementation phase is where you install the DBMS on the required hardware, optimize the database to run best on that hardware and software platform, and create the database and load the data. Set up the users and security. Implement the backup regime.

We have use MySQL database management system for the implementation of the data base. Because MySQL is most stable, secure, reliable and higher performance than the rest; it takes lower time for transactions (accessing and processing) data in a database. Which means it's faster to process transactions. MySQL reveals that it can perform efficiently regardless of its workload.

6.2. Implementation of the Class Diagram

In domain modeling class diagrams, an implements relationship exists between two classes when one of them must implement, or realize, the behavior specified by the other.

The class that specifies the behavior is called the supplier, and the class that implements the behavior is called the client. An implements relationship can include those between interfaces and classes.

For example, an implements relationship connects an interface to a subsystem. The interface specifies the behaviors, and the subsystem implements the behaviors.

In class implementation the following activities are performed.

- ✓ We define attributes with the appropriate data type and access visibilities.
- ✓ We define all methods with appropriate return type, parameters and the corresponding data types and access visibility.

6.3. Configuration of the Application Server

Role Manager is intended to be deployed on only one server platform per installation.

The procedures in application configuration are the following:

- ✓ The application server system has network access to the database server host.
- ✓ The Role Manager software is accessible by the application server system.
- ✓ You know the application server's listener port and host name.
- ✓ If running on Web Sphere, a server, cell, and node to use for Role Manager have been created and configured.

We have been used XAMPP as application server because it is a simple, lightweight Apache distribution that is extremely easy for us to create a local web server for testing and deployment purposes. Since the basic job of all web servers is to accept requests from clients (visitor of web browser) and then send the response to that request (the components of the page that a visitor wants to see). And a web server is an essential part of any website, since our developed system is web-based system, we had used Apache web Server which is open-source software.

- ✓ In the configuration process its installation is straightforward and simple then set upped and run the configuration to start, stop, and configure the services by opening the dashboard.
- ✓ It is integrated with most popular operating systems. We use XAMPP for windows.
- ✓ It is automatically installed on our computer as we install it.
- ✓ It is free web server and supports many operating systems.
- ✓ It is used to easily run and test websites and web applications locally.

6.4. Configuration of Application Security

Since our system involves storing of some personal data, we put some security mechanisms like unauthorized person cannot login into the system because the system requires a user name and password. Web application security is the process of securing confidential data stored online from unauthorized access and modification. We have implemented all input validations properly in order to secure our system. Since the system developed to the users who may senior to

computer or may professional to computer so all inputs must be implemented easily and sample to use.

6.5. Testing

6.5.1. Test Case

A test case is usually a single step, or occasionally a sequence of steps, to test the correct behavior, functionality, and features of an application.

Testing is the final step in which the entire system as a whole with all forms, code, and modules are tested. It is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. Testing is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements. In this procedure we have tested all the functionalities of the System. All errors in the forms, functions, modules have been tested. Finally, System testing ensures that the entire integrated software system meets the desired requirements. It tests a configuration to ensure known and predictable results.

Features to be tested

When we test our project, we perform following testing.

- ✓ **Input output functions.** Checking what type of our system should be taken as input and produce as output and checking input is produce expected output.
- ✓ **Graphical user interface.** in case we consider standard position of interface, standard of interface by comparing with previously performed system and criteria written for graphical user interface.
- ✓ **Sub system communication.** Our system is decomposed into different module because of module system easily managed and error easily detected and fixed. But communication between them should be tested.
- ✓ **Data base transaction.** Ensuring transaction of our system database.
- ✓ **Security.** Identifying security of our system by identifying only identified user allowed access our system and ensuring password of each individual user not seen by another. This thing is the thing we consider on login capability.

- ✓ **User interface and database interaction.** Database is one that store data and user interface one that user data is entered, so during our test we try to identify data entered on user interface stored in database and another crud.

6.5.2. Testing Tools and Environment

A testing environment is a setup of software and hardware for the testing teams to execute test cases. In other words, it supports test execution with hardware, software and network configured.

Hardware testing tools

- ✓ Computer with windows platform
- ✓ Data cable
- ✓ Web browser

Software's testing tools

- ✓ Operating system
- ✓ Windows 10

6.5.3. Unit Testing

After a development of this project an individual program module or component has been tested to ensure that it works according to its predetermined specifications, operates within acceptable parameters, and meets the appropriate standards.

In this phase of testing, every module of the System is separately tested. Each modules of the system can be tested check the working of each classes, methods and attributes of the system. It is done by the programmer to test that the unit implemented is producing expected output against given input.

Sample we test in our system in unit testing are:

- ✓ Check whether entered input is validated correctly.
- ✓ Check how the sub procedures or functions are call correctly.
- ✓ Check if the correct output is produced for different inputs.
- ✓ Check the efficiency of the code with respect to the memory and CPU time.

- ✓ Check the input data that we write on the GUI must be submitted to the data base.
- ✓ Check the GUI can access the privileged data from the data base.
- ✓ Is password of the user hidden? This can be checked in database and performed using code.

6.5.4. System Testing

System testing, also referred to as system-level tests or system-integration testing, is the process in which a quality assurance (QA) team evaluates how the various components of an application interact together in the full, integrated system or application.

In this level of testing process, examined how the whole subsystems work together to achieve the desired goal user's requirements of the system. The goals of testing are to detect faults that can only be exposed by testing the entire integrated system or some major part of it.

6.5.5. Integration Testing

After completion of individual components; they were integrated and tested to ensure that they integrate well with each other and with the operating system, and other components as well to achieve the central objectives.

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. Integration testing is conducted to evaluate the compliance of a system or component with specified functional requirements.

In this level of testing, we have examined how the different procedures work together to achieve the goal of the system. Check the interaction between individual functionality which performs the specific task.

6.5.6. Acceptance Testing

Acceptance testing is also known as user acceptance testing (UAT), end-user testing, operational acceptance testing (OAT), acceptance test-driven development (ATDD) or field (acceptance) testing. Acceptance criteria are the criteria that a system or component must satisfy in order to be accepted by a user, customer, or other authorized entity

When the user enters invalid inputs or empty, the system notifies to the user to enter valid inputs. In order to secure our system, we have been performed the following activities: -

- ✓ All inputs were validated properly
- ✓ User accounts was assigned with necessary access privileges
- ✓ Sessions was implemented.

6.6. Implementation of User Interface

The user interface is user centered design (Place users in control of the interface) and the first impression of the system where user interacts with a computer other software system.

Regarding to the user interface, we have to apply the following.

- ✓ The user interface that we develop is user adjusted design. It means we have to make our user interface be attractive and eye catching for users by developing compatible, well-matched and friendly user interface.
- ✓ The user interface that we develop is consistent and dependable
- ✓ Reduce user's memory load –our user interface can be easily remembered for users in long period of usage and it does not require any retention to use
- ✓ The user interface is consistent and stable which does not create any confusion for users easily understandable with clear and steady navigation.

CHAPTER SEVEN

7. CONCLUSION AND RECOMMENDATION

7.1. Conclusion

The computers and computerization technology in the world is growing along the fast lane where Ethiopia is attempting to run along this lane in which there are achievements recorded. And as years, pass by, we are pretty much confident that extent progress will be made in our country. System development is one of the ways in which we can make progress by making a lot of tasks easier and the overall use of computers along with the developed systems is believed to lead to the peak of the technology standard. It is known that developing a system for an organization is not easy. But the teams have tried its best and developed interesting system web based computer laboratory monitoring and management system for college of computing and informatics. It is flexible, accurate and attractive with easy GUI approach. Generally, the team confidently can say that the software is completed successfully with negligible errors. Concussively, the issues raised in the summary of finding will be discussed in this conclusion part. There for, we conclude

- ✚ The program complete in modular forms and in its overall status where it consists of all the necessary features required for a system to link a system. The testes and evaluations made show that the system is full-fledged and complete.
- ✚ The developed system can conclusively be called user friendly in which the form developed are suitable for use by any user who will defiantly get accustomed to the system's environment easily.
- ✚ We can be also concluding that the graphical user interface is fully developed with all the necessary dropdown menu and textual representation with the use of the international language.
- ✚ The system is strongly secure with the use of identification user name and a password.
- ✚ The database of the system is properly organized with fulfilled tables on their connection is appropriately made with the right forms through the coding. All the necessary tables are created effectively and in a sufficient manner.

Finally the team expects the software will change the manual doing of action and we recommend Wolkite University computer laboratory to use these software.

7.2. Recommendation

Based on our experience in the process of this project undertaking and as the computer science and information technology professionals we recommend. It is essential to consider that the project development is not easy task since it requires time, skills, resource, budget and constraints.

In addition, we recommend the Organization to which the system is developed to strictly follow the user training requirements listed in the user training section for the respective users to effectively use the system.

Since developing well organized project is not simple and it takes a long period of time to complete the project, we recommend our department to group and organize the students and tell to them to select the title the one they want to select and give enough time to complete the project effectively.

We want to recommend any voluntary persons, who have the intension to update, change and modify this project by using new technology such as digital Signature processing or to check the damaged computers and the computer laboratory equipment. And others by using other interactive programming languages. In general we are hopeful to our recommendation will be get the spectators and performed for the next coming of our follower batch students.

8. Reference

1. **L. Chao.** *Developing Web-Based Computer Laboratory, USA: IdeaGroup, . 2003. .*
2. **S. head, Interviewee.,** 8 3 2012. : s.n., About Fixed asset transefer. [Interview]. .
3. **S. F. B. BAHARUM.** *"A WEB-BASED COMPUTER LAB MANAGEMENT SYSTEM:," . MALAYSIA, 2012 : s.n.*

Websites

1. www.tutorialspoint.com/index.html
2. www.w3schools.com/index.php
3. <http://www.slideshare.net/RudyFlores1/4-technical-feasibility>

Books

1. Essentials of System analysis and design (in analysis and design phase)
2. System analysis and Design methods (in analysis and design phase)
3. Janet Valade: PHP And Mysql For Dummies, 2nd Edition

9. APPENDICES

Appendix I: prepare agreement format

WOLKITE UNIVERSITY
COLLEGE OF COMPUTING AND INFORMATICS
DEPARTEMENT OF INFORMATION TECHNOLOGY
Computer Laboratory services
Property Management form
Lab 1

Purpose: Material ownership lab
Group leader Name: Lidet Wudama
Group : All IT 4th . year Student 2013 E.C
Date: 19/11/2013

S.No	Item Name	Quantity	Serial number 1	Serial number 2	Serial number 3	Remark
1	Desktop computer	1	J060521			
2	Network cable	1				
3	Keyboard	1				
4	Mouse	1				
5	Seat chair	2				
6	Computer table	31				
7	Divider	1				
8	Staplizer	1				
9	Switch	1				

Member list:

1. Lidet Wudama
2. T. Shon GOSHU
3. Mahied Negewo
4. _____

signature: _____
signature: _____
signature: _____
signature: _____

Figure 29: prepare agreement format

Appendix II: Sample Source Code

Log in Source Code

```
<?php

if(isset($_POST['btnLogin'])){

    $email = trim($_POST['user_email']);

    $upass = trim($_POST['user_pass']);

    $h_upass = sha1($upass);

    if ($email == " OR $upass == ") {

        message("Invalid Username and Password!", "error");

        redirect("Eminetuhome.php");

    } else {

        //it creates a new objects of member

        $user = new User();

        //make use of the static function, and we passed to parameters

        $res = $user::AuthenticateUser($email, $h_upass);

        if ($res==true) {

            message("Successfully Login!Wellcome To

            ".$_SESSION['ACCOUNT_TYPE'].".", "success");
```

```

if
($_SESSION['ACCOUNT_TYPE']=='Administrator'
||$_SESSION['ACCOUNT_TYPE']=='Student'
||$_SESSION['ACCOUNT_TYPE']=='Assistance'||
$_SESSION['ACCOUNT_TYPE']=='Attendant'||
$_SESSION['ACCOUNT_TYPE']=='DepartmentHead'){

    $_SESSION['ACCOUNT_ID']    = $_SESSION['ACCOUNT_ID'];

    $_SESSION['ACCOUNT_NAME']  = $_SESSION['ACCOUNT_NAME'] ;

    $_SESSION['ACCOUNT_USERNAME']          =
$_SESSION['ACCOUNT_USERNAME'];

    $_SESSION['ACCOUNT_TYPE']   = $_SESSION['ACCOUNT_TYPE'];

    $_SESSION['EMPID']          = $_SESSION['EMPID'];

    $singledft = new Defaults();

    $assignedSEM = $singledft->single_default_sem();

    $assignedAY = $singledft->single_default_ay();

    $_SESSION['SEMESTER']      =  $assignedSEM->LISTNAME;

    $_SESSION['AY']            =  $assignedAY->LISTNAME;


    if ($_SESSION['ACCOUNT_TYPE']=='Student'){

//if ($_SESSION['ACCOUNT_TYPE']=='Student'){

```

```

date_default_timezone_set('Asia/Manila');

$date = date("Y-m-d");

$time = date(' g:i: a ');

$mydb->setQuery("SELECT                                TIMEALLOTTED,
DATE_FORMAT(`TIMEALLOTTED`,          '%H')          as          HR,
DATE_FORMAT(`TIMEALLOTTED`,          '%i')          as          MM,
DATE_FORMAT(`TIMEALLOTTED`,  '%S')  as  ss  FROM  `tblenrollment`
WHERE          `STUDENTID`='".  $_SESSION['EMPID']  ."          AND
`SEMESTER`='".$_SESSION['SEMESTER'].          AND
`AY`='".$_SESSION['AY'].""");

$res = $mydb->loadSingleResult();

$logtime =$res->TIMEALLOTTED;

$_SESSION['HR'] = $res->HR;

$_SESSION['MM'] = $res->MM;

$_SESSION['SS'] = $res->ss;

$mydb->setQuery("INSERT          INTO          `tbltimelogs`          (`LOGID`,
`LAST_LOGINDATE`, `LAST_LOGINTIME`, `TIMESTART`, `TIMEEND`,
`IDNUM`,ACTIVESTATUS) VALUES (NULL, '{$date}', '{$time}', '{$logtime}',
'00:00:00', '" . $_SESSION['EMPID'] . "', 'ACTIVE');");

$mydb->executeQuery();

}

redirect(WEB_ROOT."index.php");

```

```

    }

    }else{

        message("Incorrect Username or password .", "error");

        redirect(WEB_ROOT." home.php");

    }

} }

?>

```

Sample code create account

```

<?php

    if (!isset($_SESSION['ACCOUNT_ID'])){

        redirect(web_root."admin/index.php");

    }

?>

<script>

function GEEKFORGEEKS() {

    var name =

        document.forms["Addaccount"]["name"];

    var username =

        document.forms["Addaccount"]["username"];

```

```

var pass =

    document.forms["Addaccount"]["pass"];

var what =

    document.forms["Addaccount"]["type"];

if (name.value == "") {

    window.alert("Please enter name.");

    name.focus();

    return false;

}


if (username.value == "") {

    window.alert("Please enter username.");

    username.focus();

    return false;

}


if (pass.value == "") {

    window.alert(

        "Please enter a valid password.");

    pass.focus();

```

```
        return false;
    }

    if (type.value == "") {

        window.alert(

            "Please select the type of User.");

        type.focus();

        return false;

    }

    return true;

}

</script>
```