



**WOLKITE UNIVERSITY**  
**COLLEGE OF COMPUTING AND INFORMATICS**  
**DEPARTMENT OF SOFTWARE ENGINEERING**

**Web-based Information Management System for Yaberus  
Preparatory School**

**BY: -**

Name	ID
1. Ibsaa Mohammed	278/10
2. Abdilfeta Mohammed	338/10
3. Abbas Haji	284/10

PROJECT ADVISOR: MR. MEKURIA. G

Wolkite University, Wolkite, Ethiopia

May ,2021

**WOLKITE UNIVERSITY**  
**COLLEGE OF COMPUTING AND INFORMATICS**  
**DEPARTMENT OF SOFTWARE ENGINEERING**

**Web Based Information Management system for Yebarus**  
**Preparatory School**

SUBMITTED TO DEPARTMENT OF SOFTWARE ENGINEERING IN  
PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE  
DEGREE OF BACHELOR OF SCIENCE IN SOFTWARE  
ENGINEERING

**BY: -**

Name	ID
1. Ibsaa Mohammed	278/10
2. Abdilfeta Mohammed	338/10
3. Abbas Haji	284/10

PROJECT ADVISOR: MR. MEKURIA. G

Wolkite University, Wolkite, Ethiopia

May ,2021

# DECLARATION

This is to declare that this project work which is done under the supervision of Mr. Mekuria. G. And having the title Web Based Information Management system for Yebarus Preparatory School is the sole contribution of: -

1. Ibsaa Mohammed
2. Abas haji
3. Abdilfeta Mohammed

No part of the project work has been reproduced illegally which can be considered as 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 violation of this declaration is proven.

Date: -----

## Group Members:

Full Name

Signature

Ibsaa Mohammed

-----

Abas haji

-----

Abdilfeta Mohammed

-----

## **APPROVAL FORM**

This senior group project titled Web Based Information Management system for Yebarus Preparatory School. The documentation has been read and approved as meeting senior group project requirements of the Department of Software Engineering in partial fulfillment for the Award for the Degree of in collage of computing and informatics at Wolkite University, Wolkite, Ethiopia.

This senior Group project has been approved with the signature of advisor and Department of Software Engineering.

### **Student Team Approval Form**

Student Name	Student Signature	Date
-----	-----	-----
-----	-----	-----
-----	-----	-----

### **Advisor and department head Approval Form**

Advisor Name	Advisor Signature	Date
-----	-----	-----

Department Head Name	Department Head Signature	Date
-----	-----	-----

### **Examiner Approval Form**

Examiner Name	Examiner Signature	Date
-----	-----	-----
-----	-----	-----
-----	-----	-----

## **ACKNOWLEDGMENTS**

First and for most we would like to thank almighty God for helping us successfully complete this project. Then, we would like to express gratitude to our group advisor Mr. Mekuria Gemachu for his critical Comments, corrections, assistance and encouragement in the preparation of this project documentation.

Finally, we would like to thank you all worker of Yebarus Preparatory School for their willingness and provision of useful information to undertake this project documentation.

## Table of Contents

<b>DECLARATION</b> .....	i
<b>APPROVAL FORM</b> .....	iii
<b>LIST OF FIGURES</b> .....	ix
<b>LIST OF TABLES</b> .....	x
<b>LIST OF ABBREVIATION</b> .....	xi
<b>ABSTRACT</b> .....	xii
<b>CHAPTER ONE</b> .....	1
<b>INTRODUCTION</b> .....	1
<b>1.1. Background of the Organization</b> .....	1
<b>1.2. Statement of the Problem</b> .....	2
<b>1.3. Objective</b> .....	2
<b>1.3.1. General Objective</b> .....	2
<b>1.3.2. Specific Objective</b> .....	2
<b>1.4. Feasibility study</b> .....	3
<b>1.4.1. Technical feasibility</b> .....	3
<b>1.4.2. Operational feasibility</b> .....	3
<b>1.4.3. Economic feasibility</b> .....	3
<b>1.5. Scope and Limitation of the project</b> .....	4
<b>1.5.1. Scope of the project</b> .....	4
<b>1.5.2. Limitation of the project</b> .....	4
<b>1.5. Significance of the project</b> .....	5
<b>1.6. Beneficiary of the project</b> .....	5
<b>1.7. Methodology</b> .....	6
<b>1.7.1. Data collection method</b> .....	6
<b>1.7.2. System analysis and design</b> .....	6
<b>1.7.3. System Development Model</b> .....	7
<b>1.7.4. Testing Procedure</b> .....	7
<b>1.7.5. Development Tools</b> .....	8
<b>1.8. Budget and Time Schedule of the Project</b> .....	9
<b>1.8.1. Budget/Resource Plan</b> .....	9
<b>1.8.2. Time schedule</b> .....	10

<b>1.9. Team Composition</b> .....	10
<b>CHAPTER TWO</b> .....	12
<b>DESCRIPTION OF THE EXISTING SYSTEM</b> .....	12
<b>2.1. Introduction</b> .....	12
<b>2.2. Actors in the existing system</b> .....	12
<b>2.3. Major Functions of the Existing System</b> .....	13
<b>2.4. Forms used in current Systems</b> .....	14
<b>2.5. Drawback of the existing system</b> .....	14
<b>2.6. Business Rule</b> .....	14
<b>CHAPTER THREE</b> .....	15
<b>PROPOSED SYSTEMS</b> .....	15
<b>3.1. Introduction</b> .....	15
<b>3.2. Functional Requirement</b> .....	15
<b>3.3. Non-functional Requirements</b> .....	16
<b>3.3.1. User Interface and Human Factors</b> .....	16
<b>3.3.2. Hard ware consideration</b> .....	16
<b>3.3.3. Security Issues</b> .....	17
<b>3.3.4. Performance Characteristics</b> .....	17
<b>3.3.5. Error handling and validation</b> .....	17
<b>3.3.6. Quality Issues</b> .....	17
<b>3.3.7. Backup and Recovery</b> .....	18
<b>3.3.8. Physical Environment</b> .....	18
<b>3.3.9. Resource Issues</b> .....	18
<b>3.3.10. Documentation</b> .....	19
<b>CHAPTER FOUR</b> .....	20
<b>SYSTEM ANALYSIS</b> .....	20
<b>4.1. Introduction</b> .....	20
<b>4.2. Use Case model</b> .....	20
<b>4.3. Object Model</b> .....	34
<b>4.3.1 Class Diagram</b> .....	34
<b>4.3.2. Data Dictionary</b> .....	35
<b>4.4. Dynamic Model</b> .....	36

4.4.1. Sequence Diagram.....	36
4.4.2. Activity Diagram.....	42
4.4.3. State Chart Diagram.....	51
CHAPTER FIVE .....	56
SYSTEM DESIGN.....	56
5.1. Introduction.....	56
5.2. Design Goals .....	56
5.2.1. User Interface and Human Factors .....	56
5.2.2. Hardware Consideration.....	56
5.2.3. Security Issues .....	57
5.2.4. Performance Consideration .....	57
5.2.5. Error Handling and Validation .....	57
5.2.6. Quality Issues .....	57
5.2.7. Backup and Recovery .....	57
5.2.8. Resource Issues.....	57
5.2.9. Documentation .....	58
5.3. Current System Architecture.....	58
5.4. Proposed System Architecture.....	58
5.4.1. Subsystem Decomposition and Description .....	59
5.4.2. Hardware/software Mapping.....	62
5.4.3. Detailed Class Diagram .....	63
5.4.4. Persistence Data Management.....	64
5.4.5. Access Control and Management .....	64
5.4.6. Packages.....	65
5.5. User Interface Design .....	67
CHAPTER SIX .....	69
IMPLEMENTATION AND TESTING .....	69
6.1. Introduction.....	69
6.2. Implementation of the Database.....	69
6.3. Implementation of the Class Diagram .....	70
6.4. Configuration of the Application Server.....	72
6.5. Configuration of Application Security .....	73

<b>6.6. Implementation of User Interface</b> .....	74
<b>6.7. Testing</b> .....	77
<b>CHAPTER SEVEN</b> .....	80
<b>Conclusion and Recommendation</b> .....	80
<b>7.1. Conclusion</b> .....	80
<b>7.2. Recommendation</b> .....	80
<b>8. Reference</b> .....	81
<b>9. APPENDICES</b> .....	82

## LIST OF FIGURES

Figure4. 2 Class diagram .....	34
Figure4. 3 <i>Sequence Diagram for Login</i> .....	37
Figure4. 4 <i>Sequence Diagram for Create Account</i> .....	38
Figure4. 5 <i>Sequence Diagram for Download Course Material</i> .....	39
Figure4. 6 <i>Sequence Diagram for Upload Course material</i> .....	40
Figure4. 7 <i>Sequence Diagram for Upload Report Card</i> .....	41
Figure4. 8 <i>Sequence Diagram for View Report card</i> .....	42
Figure4. 9 Activity diagram for login.....	43
Figure4. 10 Activity diagram for Manage user account.....	44
Figure4. 11 Activity diagram for send message .....	45
Figure4. 12 Activity diagram for student registration .....	46
Figure4. 13 Activity diagram for download course material .....	47
Figure4. 14 Activity diagram for upload course material .....	49
Figure4. 15 Activity diagram for upload report card .....	49
Figure4. 16 Activity diagram for view report card .....	50
Figure4. 17 State Chart diagram for Login .....	51
Figure4. 18 State Chart diagram for Download Course Material .....	52
Figure4. 19 State Chart diagram for Upload Course material .....	53
Figure4. 20 State Chart diagram for View Report card.....	54
Figure4. 21 State Chart diagram for View Information .....	55
Figure5. 1 Proposed System Architecture.....	59
Figure5. 2 <i>Component Diagram</i> .....	61
Figure5. 3 Deployment Diagram .....	62
Figure5. 4 Detailed Class Diagram .....	63
Figure5. 5 Persistence Diagram .....	64
Figure5. 6 Package Diagram.....	66
Figure5. 7 Login Page .....	67
Figure5. 9 Student Home Page .....	68
Figure5. 10 Teacher Home Page .....	68

## LIST OF TABLES

Table1. 1 Budget/Resource plan.....	9
Table1. 2 Time table.....	10
Table1. 3 Team Composition .....	10
Table2. 1 Actors of existing System .....	12
Table4. 2 Use case description for create account.....	23
Table4. 3 Use case description for delete account .....	24
Table4. 4 use case description for Send Message .....	25
Table4. 5 use case scenario for View Message.....	26
Table4. 6 use case description for Download Course material .....	27
Table4. 7 use case description for View Report Card .....	28
Table4. 8 use case description for Event Upload.....	29
Table4. 9 use case description for Upload Course Material .....	30
Table4. 10 use case description for view information.....	31
Table4. 11 Data dictionary for Admin .....	35
Table4. 12 Data dictionary for Student.....	35
Table4. 13 Data dictionary for Teacher .....	35
Table4. 14 Data dictionary for Parent.....	35
Table4. 15 Data dictionary for Message .....	36
Table4. 16 Data dictionary for Forum Topic .....	36
Table4. 17 Data dictionary for Report Card .....	36
Table5. 1 Access Security Table .....	65

## **LIST OF ABBREVIATION**

BR.....	Business Rule
CSS.....	Cascading style sheet
HTML.....	Hypertext Mark-up Language
MYSQL.....	Structured query Language
PHP.....	Hypertext Pre-processor
OOA.....	Object Oriented Analysis
OOD.....	Object Oriented Design
OS.....	Operating System
UML.....	Unified Modeling Language
WAMP.....	Window Apache MYSQL PHP

## **ABSTRACT**

School information management system is the management of school using computerized communication. Our system is a web-based system, so the user of our system can access the system using computer browser. Some of those difficulties are students do not get updated information, students cannot get fast access of reading materials, submission of assignment is not secure it may be lost before reaching. So, the project we are going to develop will try to recommend those problems and providing prototype of the system to be developed.

Because the system to be developed is online, it includes online registration based on payment status of the students, uploading modules, posting new information. This project is not valuable only for students but also for employers.

# CHAPTER ONE

## INTRODUCTION

Education system forms the backbone of every nation. And hence it is important to provide a strong educational foundation to the young generation to ensure the development of open-minded global citizens securing the future for everyone. Advanced technology available today can play a crucial role in streamlining education-related processes to promote solidarity among students, teachers, parents and the school staff.

Education is central to development. It is one of the most powerful instruments for reducing poverty and inequality and lays a foundation for sustained economic growth. Student evaluation, submissions, teaching-learning process, education material distribution and availability, event and information communication across to students, teachers and parents, and make them easily accessible on the Internet. In order to facilitate and simplify these actions one of the major tools is to have well-structured school information management system.

In generally, well-structured school information management system is essential in the world. That is why now a day in the world teaching of new generation progress increased in order to produce skillful and educated people. As we know the technology growth further in the behave of education. Automation is the utilization of technology to replace human with a machine that can perform more quickly and more continuously by automating school information management system to replace the physical communication by computerized communication.

### **1.1. Background of the Organization**

Yaberus Preparatory School found in Southern Ethiopia, Wolkite town, Adis ketema kebele. The school was formed on 1973 E.C. It consists both high school and preparatory school, so we are going to deal with both. School information management system in the government schools of Yaberus preparatory school is manual, because of this reason we will develop a web-based School information management system for this school.

## **1.2. Statement of the Problem**

The school applies manual way of processing of tasks in some case. The manual school information management system is time taking, unqualified, costly and not satisfactory. Due to this reason the existing manual process has the following noticeable problems.

- ✓ Poor time management system.
- ✓ Academic staff and student's data are stored redundantly in multiple file.
- ✓ Loss of students and teacher's document.
- ✓ The teaching-learning process is limited to only to class room.
- ✓ The event notification system currently deployed is prone to information, resource and labor wastage.
- ✓ Accessibility of resource is limited to fixed area because the resources are hard copy.
- ✓ Information distribution is difficult
- ✓ Due to increasing the number of students takes a lot of time to register.
- ✓ Data retrieving problem due to a large number of files recorded manually.

## **1.3. Objective**

### **1.3.1. General Objective**

The main Objective of the project is to develop web-based Management system for Yebarus Preparatory School.

### **1.3.2. Specific Objective**

In order to achieve the general objectives, we have the following specific objectives.

- ✓ Exploring and studying the existing system.
- ✓ Developing assessment handling component in order to upload and submit assignments.
- ✓ Developing result processing component in order to upload students' result, and view students' result.
- ✓ Developing communication handling component in order to create communication means between teachers, students and parents.
- ✓ Developing course material sharing component in order to share course materials.

- ✓ Developing event handling component in order to enable event creation.
- ✓ Developing announcement handling component in order to board cast announcements among users.
- ✓ Developing discussion forum handling component in order to create, and view discussion forum.
- ✓ Developing employees' information recording component in order to add, and delete employees.

## **1.4. Feasibility study**

### **1.4.1. Technical feasibility**

The system is going to be developed by technological development technique such as PHP language with XAMP Server and notepad, sublime or notepad++editor, MySQL database to store files, Microsoft word for writing document, java script CSS. The system that we going to develop is familiar with new technologies and easy to understand by the system users. files, Microsoft word for writing document, java script CSS. The system that we going to develop is familiar with new technologies and easy to understand by the system users.

### **1.4.2. Operational feasibility**

It can solve the problems in registration of new and existing students and purchasing school materials. We expect that, the system can be operated in simple way and all users can access easily. The proposed system is very easy for the end user to operate it. The system performs all operations to achieve the specified objective, user friendly and interactive with the user. The user can use the system without any difficulty or complex procedures to perform their operation, since the proposed system performs all operations with window platform. So, the project is operational feasible.

### **1.4.3. Economic feasibility**

It is possible to develop the proposed system in minimum cost and also can be hosted in lower price. School information management systems can be used after completion free of cost. We are using a free hosting server and it doesn't need a lot of resources. The development of the system requires a minimum amount of cost.

**I) Tangible Benefits:** - Benefits that are easily quantified from the conducted system are:

- ✓ Fastest processing time and reduced processing error.
- ✓ Small response time and many services
- ✓ Easy and fast file management.
- ✓ Reduce cost for manual data management (Reduced expenses)
- ✓ Easy update & retrieval on stored records

**II) Intangible Benefits:** -Benefits from the system that areas unquantifiable are;

- ✓ Better decision making
- ✓ Better service to the office
- ✓ Little job burden to employees of office
- ✓ Quantity of information
- ✓ Minimization of data redundancy

## **1.5. Scope and Limitation of the project**

### **1.5.1. Scope of the project**

This project develops management system for Yaberus Preparatory School using a web-based application. The main activities of the proposed system are: -

- ✓ Assessment handling (Particularly for Assignment)
- ✓ Result processing
- ✓ Enable Communication
- ✓ Course material sharing
- ✓ Event, discussion forum, and announcement handling
- ✓ Employees' information recording

### **1.5.2. Limitation of the project**

- ✓ The system does not have attendance.
- ✓ The system does not have scheduling.
- ✓ The system does not support online payment.

## **1.5. Significance of the project**

This project benefit to the owner in the following ways:

- ✓ Schools able to reach out to parents and students in a more flexible and simple way.
- ✓ School information management system also give the owner an upper hand over those other schools that are using the traditional system bringing in more interested clients.
- ✓ Schools provide appropriate materials to their students, resulting in an improved National Exam Scores which in turn helps the school get a better ranking over the other competitors.
- ✓ Control the student's information from risk.
- ✓ Reduce time, cost and redundancy of information or data.
- ✓ Improve utilization of manpower and material resource
- ✓ Manages all administrative records with zero redundancy
- ✓ Achieve best possible resource optimization

## **1.6. Beneficiary of the project**

### **Administrator:**

Admin is benefited from the system by easily posting information about the status of the school and discuss in simple way with students, teachers and student's parents.

### **Student:**

The student is benefited from our systems in which they simply access learning materials from the system.

### **Parent:**

Parent is benefited from the system by accessing school information and communicating with teachers and Administrators.

### **Teacher:**

Teachers are benefited from our system by easily distributing course materials for the students

## **1.7. Methodology**

### **1.7.1. Data collection method**

#### **1.7.1.1. Interview**

This is one of data collection method that enables to gather information from the questions. So, we have used this method to gather information by asking the director of Yaberus preparatory School some basic questions.

Question that we asked are: -

Raised Question that we asked: -

- ✓ How the teaching and learning process work?
- ✓ What requirement are needs for your school?
- ✓ When and why meeting is taken place between parents and teachers?
- ✓ How teachers proved information to students?
- ✓ During the teaching and learning process, are there any problems? If there, what are they?

#### **1.7.1.2 Observations**

By visit, the school staffs while they work. For our project we observe the existing system, the environment of classrooms and the procedures of teacher student relationships throughout teaching learning processes that provide primary information for our project.

#### **1.7.1.3. Document analysis**

Existing document analysis to collect more information about the existing system relevant to develop the new system.

## **1.7.2. System analysis and design**

In this project, we use Object Oriented System Analysis and Design methodology (OOSAD).

This has two phases: -

**Object Oriented Analysis (OOA):** - During this phase we use to model the functions 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 object.

**Object Oriented Design (OOD):** - During this phase we use to refine the use case model to reflect the implementation environment, model object interactions and behaviors that support the use case scenario, and finally update object model.

We select it for the following advantages: -

- ✓ *Managed complexity:* -The object-oriented methods solve our system complexity in the following way, by design our system software expectation that it will need to be modified and being able to respond quickly when our system environment changed
  - ✓ To inherit class defined in super class
  - ✓ To make it easier for teams of designers and programmers to work in a single software project.
  - ✓ *Extensibility:* -when you to need to add new feature to the school management system you only need to make changes in one part of the applicable class.
  - ✓ Improved communication among users, analysis, design and programming.
  - ✓ To reuse method to decrease redundancy

### **1.7.3. System Development Model**

Software development model: we use iterative model because in iterative model you can iterate back if error is occurring in one phase and we can return back to other phase to fix errors at any phase of the project life cycle. In Iterative model, iterative process starts with a simple implementation of a small set of the software requirements and iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed.

The main reason to select this model is due to the following

- ✓ Less costly to change the scope and requirement
- ✓ Risks are identified and resolved during iteration and each iteration is managed easily
- ✓ Fixing errors return back easily.

### **1.7.4. Testing Procedure**

Testing Procedure Testing Procedure is the way to know the system functionality weather it's working according to the specification, behavioral and performance requirement. To find out type of error which occur during implementation time we will conduct this type of testing procedure:

**Unit testing:** Used for testing of individual program components. We will use this type of testing to test functions in a class and how they behave. In our system we use to Test the login part of our project and any single component in our project.

**Integration Testing:** In this section we will test the dependency between two or more components and see how components work with each other. In our system we will use to Test two or more component means home and login part of our project and any other components that exist two or more.

**System Testing:** we will use this for testing the complete project. In our system we will use to test our project completely.

### **1.7.5. Development Tools**

The tools we will use to accomplish the project are

#### **1.7.5.1. Software tools**

- MySQL: - to design data base and the used as data base server
- Apache: - used to implement the code
- Microsoft word: - for documentation
- Enterprise architect and other: - for designing UML diagram associated with the project
- Web browsers Mozilla Firefox, Google Chrome, Safari and UC Browser; to run our implementation
- Sublime Text: - to write the code.
- Microsoft PowerPoint: - for presentation.
- OS Windows 7 and 10.

### **Server-side Script**

We have selected PHP for server-side scripting because of: -

- Fast – PHP was created to develop dynamic Web Pages so it is fast on websites. The PHP code is embedded in HTML and the time it takes to process and load the browser with HTML and create a full web page is very quick.
- Free – PHP is released under the PHP License. This means that anybody can download it and use it 100% for free.

- Easy - The syntax of PHP is very easy to use and learn. PHP is usually mixed in with HTML, CSS, and java script and can be easily included in HTML files.

## Client-side script

HTML, CSS and JAVASCRIPT.

**HTML** is a client-side programming language: -for developing the client side or the user interface. The reason why we use it is editing and running code is fast to allow the addition of dynamic and interactive activities to web pages, minimum programming knowledge or experience required.

**CSS** (cascading style sheet): Styles how to display HTML elements

### 1.7.5.2. Hardware Tools

- Laptop and desktop for documentation and implementation
- Samsung Smart phone used to Captor the photo
- USB Flash disk used to copy data from one laptop to another
- Compact disk used to store data for permanently

## 1.8. Budget and Time Schedule of the Project

### 1.8.1. Budget/Resource Plan

Table1. 1 Budget/Resource plan

Requirements	Specification	Amount	Price
Paper	A4	2	250
Pen	BIC	3	30
Mobile card		4	40
Transportation		2	60

Flash memory	Sony	1	350
Print paper		1	35
Total price			765

### 1.8.2. Time schedule

Time evaluation is the most important consideration in development of the project. So, the concerned project should be completed within fixed scheduled time as far as system is concerned.

Table1. 2 Time table

Phases	Weeks			
	1-2	3-4	5-7	8-12
Requirement				
Analysis				
Design				
Implementation				
Testing				

### 1.9. Team Composition

Table1. 3 Team Composition

Project Title	"School Information management system for Yaberus preparatory school"			
Prepared by	No	Student Name	Id Number	Task
	1	Ibsaa Mohammed	278/10	Requirement gather, analysis, and implementer.

	<b>2</b>	Abdulfeta Mohammednur	338/10	Requirement gather, analysis, and implementer.
	<b>3</b>	Abas Haji	284/10	Requirement gather, analysis, and implementer.
Date				
Adviser				

# CHAPTER TWO

## DESCRIPTION OF THE EXISTING SYSTEM

### 2.1. Introduction

Yaberus Wolkite secondary and preparatory school management system focuses on registering of students, registering new comer teachers, managing student and teacher's data, exchanging of report between offices of the school. The offices of the school include Director Office, Record office and etc. Yaberus Wolkite secondary and preparatory school manage their tasks manually in some cases not at all. Due to this the management system of the school needs many materials like paper, pen, binders (To organize the manual files) and other materials, waste time and budget of the school, loosing of teachers and student document, preparing reports manually is so boring, taking Time and has mistaken to prepare report card of students, updating employee's data etc. The problems make the school to lead a poor management system.

### 2.2. Actors in the existing system

Actor are a person who is external agent to the system. Those persons have directly interacted to the system. This person is performing some action like registering students, registering teachers, registering student that has discipline case preparing mark, delete student, teacher information view report.

Table2. 1 Actors of existing System

Actors	Responsibility
Director	Viewing reports updating, deleting registered teachers
Teacher	Organize student grade updating, deleting and searching Students marks
Student	View mark

The actors of the system perform all activities by work division. As described above the record officer perform tasks related with human resource management, record student's data prepare

transcript and reports. And also, the Director or Vice Director of the school access all information's of the school, i.e. view search data of academic staff and student of the school.

### **2.3. Major Functions of the Existing System**

Presently institutions that facilitate secondary school education in Ethiopia are stuck in the old, rather classic form of establishment. In this system there are different functions that are performed such as:

- ✓ **Report card delivery:** this given out to parents by home room teachers of the respective class in the school compound on a date set by the school's administration. This happen only if the parent of the student is present in person to claim the assessment card, if not no delivery made.
- ✓ **Teaching-learning process:** this exchange of academics occurs in a class room for a specified length of time for each corresponding course. In this specified period of time the teacher should finish its lecture content, provide class works and assignments and provide students with answers to questions that might arise from discussions. If this allocated time is over the teacher then leave the class room for other teachers to proceed with their lectures.
- ✓ **Event and Information communication:** communications at school compounds are often made possible by notice boards, broadcasts in devotions and meetings and by printed out paper slips.
- ✓ **Academic materials:** Schools have libraries that assist students in acquiring books available in the school ground. This library has references and text books present in a hard copy that provided to the students in person by the librarian. Books in these libraries are of limited quantity so whoever comes first is first served, thus if one student in need of a book arrives late than others, he/she have to wait a specified amount of time to attain it. Furthermore, these libraries are equipped with small amounts of reading.
- ✓ **Materials thus limiting student's need of exploration. Payment notification:** Schools notify parent and students about payments with notice. slips that written and handed to them.

## 2.4. Forms used in current Systems

There are a number of forms which are used by the existing system. The forms are used during the period of the Registration process. Some of the forms are illustrated in Appendix.

## 2.5. Drawback of the existing system

- ✓ Slow and difficult registration new coming students.
- ✓ Difficult way in searching, viewing, updates new students' information.
- ✓ Hard for teacher when prepare students marks.
- ✓ Giving slow service to the students and academic staff of the school.
- ✓ Difficult creating account for teachers, record office and vie director.
- ✓ Consume more time and budget.
- ✓ Not effective functioning method in managing the school.

## 2.6. Business Rule

- ✓ **BR1:** Student bill for registration
- ✓ **BR2:** Students fill their registration form.
- ✓ **BR3:** Students registered with their family profile.
- ✓ **BR4:** admin Record registered student's data.
- ✓ **BR5:** If students lost the school educational books will pay.
- ✓ **BR6:** Students who have serious disciples' case will be fired from the school.
- ✓ **BR7:** Working time of the organization starts from 2.30am.
- ✓ **BR8:** Students are not allowed to see other student data.
- ✓ **BR9:** Student must understand the school regulation.
- ✓ **BR10:** The student must pass every test before continuing the next level.
- ✓ **BR11:** Students are not allowed to see other student data.
- ✓ **BR12:** The administrator has the right to control the attendance of the teacher.

## **CHAPTER THREE**

### **PROPOSED SYSTEMS**

#### **3.1. Introduction**

This system as we described it on our proposal document is a broad, diverse and easily applicable system that assists our clients gets rid of problems that are clearly visible or that may be hidden. In this document we will see how this diverse system will function, its users and the interaction they will have with the system to give our clients a better understanding of the entire system and us an understanding of the constraints and needs of our clients. The intention of the system is to enable our clients and Schools to get better the teaching–learning process, also parent-teacher interaction and also give their students an education away from school. The proposed system improves things like: -

- Creating effective functioning in managing the school.
- Minimize time and budget.
- To make easy and fast in searching user’s information and students report card.
- It helps the students and staff of the school in giving fast response.

#### **3.2. Functional Requirement**

The functional requirement for the system describes the functionally or services that the system is expected to provide. The proposed systems provide the following functionalities:

- The system shall manage user accounts.
- The system shall handle Event notifications.
- The system shall handle Announcements.
- The system shall handle send and receive messages or chat among users.
- The system shall handle report card or Roster form assessment upload.

- The system shall enable teacher upload course materials.
- The system shall enable students download course materials.
- The system enables Admin register student of the school.
- The system shall enable users create and discuss on Forum topics.
- The system shall enable the Admin post a vacancy.
- The system shall allow admin add school learning material.
- The system shall allow admin add library books.

### **3.3. Non-functional Requirements**

Non-functional requirement defines how the system is supposed to be and non-functional requirement often called quality attribute of the system. It specifies criteria that can be used to judge the operation of a system rather than specific behavior. Nonfunctional requirement specifies over all characteristics such as cost and reliability. Many non-functional requirements relate to the system as a whole rather than individual system features. This are:

#### **3.3.1. User Interface and Human Factors**

The system should provide: -

- ✓ An interface that is easy to use.
- ✓ Interaction of users with system should be through GUI
- ✓ The form should be user friendly, it should not be implicated and does not need any extensive training
- ✓ This application uses English to interact with the users.
- ✓ User shall use web browser to access the system
- ✓ User shall be able to issue commands using mouse by clicking on menu

#### **3.3.2. Hard ware consideration**

User shall be able to provide a machine either a mobile phone or a computer that is able to connect to the internet.

### 3.3.3. Security Issues

The system shall allow only to authenticated users which are specified by the administrator of the System who has authority to give access to the system.

- ✓ The system does not allow unauthorized user to access it.
- ✓ The user must have permission to login into the system.

### 3.3.4. Performance Characteristics

- Performance of the system shall vary depending on the current capability of the Internet service provided by the Internet Service Provider and current capability the machine used to access the system.
- This system gives service 24 hours per day with maximum response time so, it is easy to access data from the stored document.
  - ✓ Easy to use.
  - ✓ Fast and reliable
  - ✓ Easy to search information
  - ✓ Easy to update, delete, and add information.

### 3.3.5. Error handling and validation

The system shall display error message if the user enters invalid input like password and user name.

### 3.3.6. Quality Issues

Information in database should be accurate, updated, delete, and retrieve. The quality issue of the proposed system should have expressed in terms of availability, reliability and robustness of the system.

**Reliability** The system should be highly reliable. To be reliable, hardware and software of the system should have integrated properly and should be connected to database by network thenIf any mistake happened by the user, the system displays error message before stop working in

order to minimize system failures.so the system should have high potential to recovery from loose of data.

**Availability** the system is available at any time when user wants to get service.

**Robustness** the system should be robust and adapt any kind of change that occur in the system.

### **3.3.7. Backup and Recovery**

**Backups:** - Backups are useful primarily for two purposes. The first is to restore a state following a disaster (called disaster recovery). The second is to restore small numbers of files after they have been accidentally deleted or corrupted. Data loss is also very common. 66% of internet users have suffered from serious data loss. The system should be holding a backup of the data by using different storage devices. Some of them are Flash Memory, DVD Backup, Tape Backup, Hard Drives and A drive.

#### **Back up**

- We use back up the complete file with flash disk, if suddenly the computer that contain the file deleted
- Save the folder that contain our project in different computer
- We duplicate the same folder that contain our project

**Recovery:** - Data recovery is the process of restoring data that has been lost, accidentally deleted, corrupted or made inaccessible for some reason. Data recover by using system recovery and any software like restoration, wise data recovery, recover software.

### **3.3.8. Physical Environment**

The school information management system used or deploy for yaberus preparatory school. We should also select suitable place for restoring or for setting the system to protect from danger and some external factor.

### **3.3.9. Resource Issues**

In our system we consider time, hardware and software as resource issue.

**Server:** minimum hardware requirements for Apache server are:

- CPU:32bit or 64-bit cores: single (single core 3Hz or higher dual

- Core 2GHz or higher is recommended.
- Display resolution:1360X768(or higher)

**Client:**

- Browser
- CPU:32 or 64 bit
- RAM:2GB or higher

**Editor:**

- Notepad++ or Notepad
- Adobe Photoshop (for editing an image)

We will use different hardware tools to develop our project

Laptop Computer: -used to develop the system.

Flash Disk: - used to store data.

CD/DVD: used to store and secure data as back up.

Network cable: since our system is web based, it is very necessary requirement. It also help us to extract relevant information about our project from internet

**3.3.10. Documentation**

The new system provides required full documentation, help contents and tips to allow further maintainability and to support and guide users how to use the new system. The system will provide information about how to use it and all the essential information about the system. This system will provide three types of system documentations. They are Internal Documentation, user Documentation and External Documentation.

# **CHAPTER FOUR**

## **SYSTEM ANALYSIS**

### **4.1. Introduction**

System model is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system. System modeling has now come to mean representing a system using some kind of diagram notation, which is now almost always based on notations in the Unified Modeling Language (UML)

### **4.2. Use Case model**

To model a system the most important aspect is to capture the dynamic behavior. To clarify in details, dynamic behavior means the behavior of the system when it is running or operating. So only static behavior is not sufficient to model a system rather dynamic behavior is more important than static behavior. In UML there are four diagrams available to model dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction.

- These internal and external agents are known as actors. So, use-case diagrams are consisting of actors, use cases and their relationships. The diagram is used to model the system or subsystem of an application. A single use case diagram captures a particular functionality of a system. So, to model the system we used the following use case diagrams.

### **Actor identification**

The purpose of actor analysis is to identify all of the actors that interact with the system. An actor has a role in that interacting with the system.

The actors that interact with the system are: -

- Administrator
- Student
- Parent
- Teacher

## Use-case identification

Identifying the activities that are mainly performed on the proposed system is the basic thing in analyzing a new system. The following use cases have been identified from the system specification.

- Login
- Upload course material
- Download course material
- Upload event
- Upload report card
- Create and delete event
- Manage user account
- View report card
- View message
- Send message
- User registrations
- View event
- Create forum topic
- Submit assignment
- Logout

Use case diagrams graphically describe system behavior (use cases). These diagrams present a high-level view of how the system is used as viewed from an outsider's (actor's) perspective.

### 4.2.1. Use Case Diagram

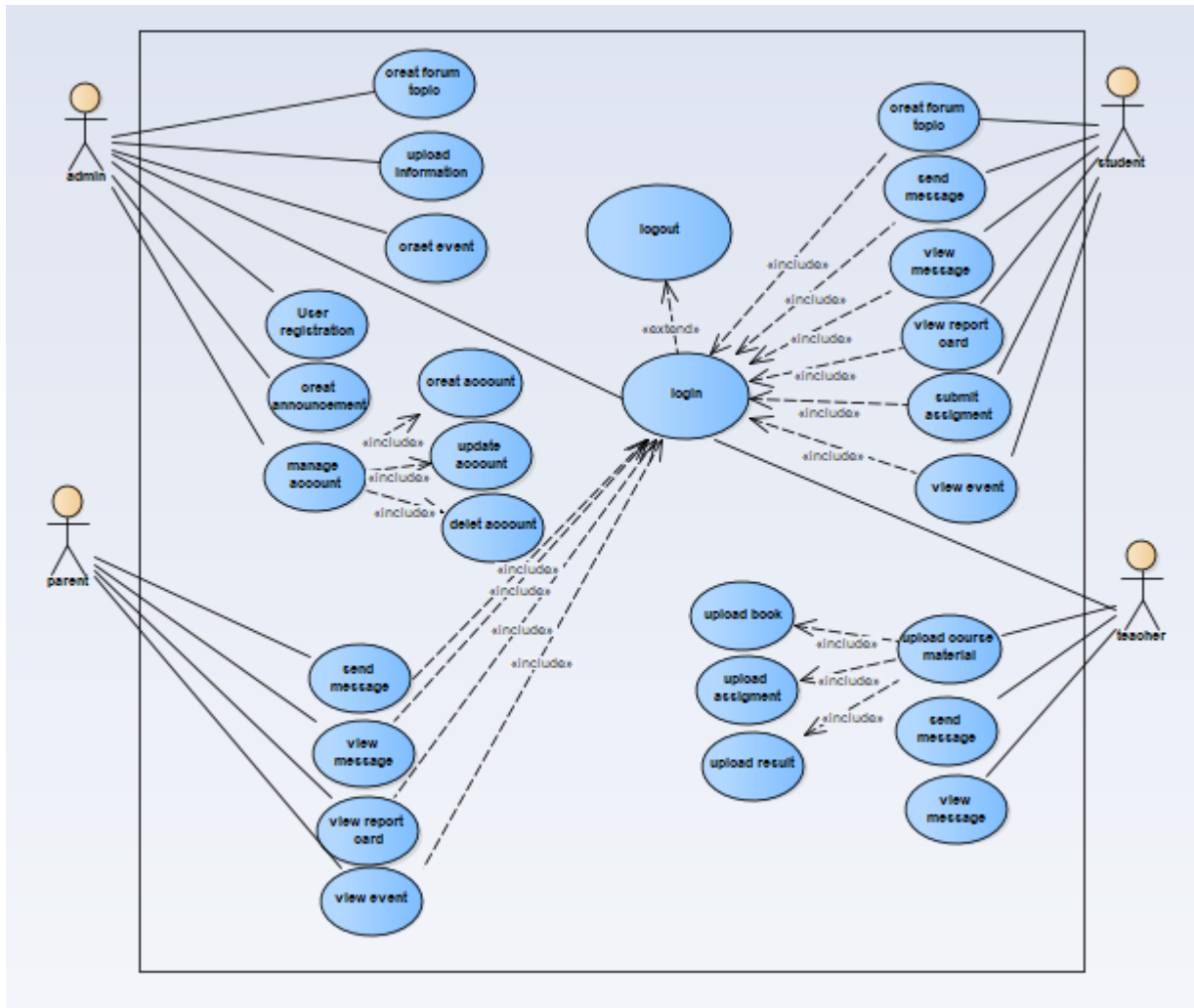


Figure4. 1 Use case diagram

### 4.2.2. Use Case Description

Table4. 1 Use case description for login

Use case name	Login
Actor	User
Pre-condition	The users must be an authorized user who have user name and password

Description	When user enters id and password, it checks the input from the database. If it is valid, it allows the user to access and if not, it displays unauthorized message.
Basic course of action:	<ol style="list-style-type: none"> <li>1)the user enters to the main menu</li> <li>2) The user clicks to the login link.</li> <li>3) The system displays the log in form.</li> <li>4) the user fills user name and password</li> <li>5) The user clicks to the submit button.</li> <li>6) The system validates user name and password.</li> <li>7) The system gives response.</li> </ol>
Alternate course of action:	A4) if the user does not fill the appropriate username and password then the systems display error message and return to step 3.
Post-condition	The user accesses the system.

Table4. 2 Use case description for create account

Use case name	Create Account
Actor	Administrator
Pre-condition	The Administrator should have user name and password
Description	Administrator creates account for student, teacher and parent.

Basic course of action:	<ol style="list-style-type: none"> <li>1) Administrator login into the system.</li> <li>2) System checks the username and password.</li> <li>3) System opens the main page.</li> <li>4) User click on the create account link.</li> <li>5) System opens the page.</li> <li>6) User fills information and create account.</li> <li>7) System checks value is valid or not.</li> <li>8) If it is valid account created.</li> <li>9) User see successful message of account created.</li> </ol>
Alternate course of action:	A4) if the user does not fill the appropriate information the systems display error message and return to step 6.
Post-condition	Administrator creates account.

Table4. 3 Use case description for delete account

Use case name	Delete Account
Actor	Administrator
Pre-condition	The Administrator should have user name and password
Description	Administrator deletes account of student, teacher and parent.

Basic course of action:	<ol style="list-style-type: none"> <li>1) Administrator login into the system.</li> <li>2) System checks the username and password.</li> <li>3) System opens the main page.</li> <li>4) User clicks on the delete account link.</li> <li>5) Then select status and specific user.</li> <li>6) User click delete button.</li> <li>7) Account deleted of successful message will be displayed.</li> <li>8) User see the message.</li> </ol>
Alternate course of action:	A4) if the user does not select the appropriate status the systems display error message and return to step 5.
Post-condition	Administrator deletes account.

Table4. 4 use case description for Send Message

Use case name	<b>send message</b>
Actor	Users
Pre-condition	The users must be an authorized user who have user name and password
Description	User enters their own id and password within the system and to send and resave authorized message.

Basic course of action:	<ol style="list-style-type: none"> <li>1. . The user wants to send message</li> <li>2. The user clicks login button.</li> <li>3. The system displays login form.</li> <li>4. The user fills username and password.</li> <li>5. The user clicks login button.</li> <li>6. The system displays a page containing the <i>New Message</i> button.</li> <li>7. The user clicks on the <i>New Message</i> button.</li> <li>8. The system displays a page containing the Message form.</li> <li>9. The user fills in the Message form and clicks on the <i>send</i> button.</li> <li>10. The system displays success message.</li> <li>11. The use case ends.</li> </ol>
Alternate course of action:	<p>A9) if the user fills the form incorrectly. the system displays error message.</p> <p>Return step 8.</p>
Post-condition	Message is sent successfully.

Table4. 5 use case scenario for View Message

Use case name	<b>View message</b>
Actor	Users
Pre-condition	The user has Username and password and is logged in to Systems
Description	User enters their own password and ID with system and to see different type of message with the system.

Basic course of action:	<ol style="list-style-type: none"> <li>1. The user wants to view message.</li> <li>2. The users clicks login button.</li> <li>3. The system display login form.</li> <li>4. The user fills user name and password.</li> <li>5. The user clicks login button.</li> <li>6. The system displays a page containing various message buttons.</li> <li>7. The user clicks on the message he/she wants to view.</li> <li>8. The system displays a page containing the clicked message.</li> <li>9 the user views the message.</li> <li>10. The use case ends.</li> </ol>
Alternate course of action:	<p>A4) if the user fills incorrect user name password. The system displays error message.</p> <p>Return to step 3.</p>
Post-condition	Message is seen successfully by the user.

Table4. 6 use case description for Download Course material

Use case name	<b>Download Course material</b>
Actor	Student
Pre-condition	The student must be an authorized user who have user name and password
Description	Student enters the system their own password and id; and they can download course material that available the web site.

Basic course of action:	<ol style="list-style-type: none"> <li>1. . The student wants to down load course material.</li> <li>2. The student enters to the main menu.</li> <li>3. The system display login form.</li> <li>4. The student fills user name and password.</li> <li>5. The student clicks the login button.</li> <li>6. The student sends data to the system.</li> <li>7. The system checks the data.</li> <li>8. The system displays the contents.</li> <li>9. The student views the content.</li> <li>10. The student double clicks the content which he/she wants to download.</li> <li>11. The student downloads successfully.</li> <li>12. The use case ends.</li> </ol>
Alternative course of action	<p>A 4) If the student does not inter the correct user name and password The use case backs to step 3.</p> <p>A 7) if the content does not exist the data base. The use case returns to step 6.</p>
Post-condition	Course Material is successfully downloaded by the student.

Table4. 7 use case description for View Report Card

Use case name	<b>upload Report Card</b>
Actor	Admin
Pre-condition	The admin must be an authorized who have user name and password

Description	When admin enters his own id and password, and he can upload student's report card.
Basic course of action:	<ol style="list-style-type: none"> <li>1. The admin wants to upload report card.</li> <li>2. The admin clicks the login button.</li> <li>3. The system displays the login form.</li> <li>4. The admin fills user name and password.</li> <li>5. The admin clicks to the login button.</li> <li>6. The system display s report card form.</li> <li>7. The admin fills the form.</li> <li>8. The admin clicks upload button.</li> <li>9. The system checks the information fill by the admin.</li> <li>10. The report card is uploaded correctly.</li> <li>11. The use case ends.</li> </ol>
Alternate course of action:	<p>A7) if the admin dose not fills the form correctly.</p> <p>Return to step 6.</p>
Post-condition	Report card of the student is seen.

Table4. 8 use case description for Event Upload

Use case name	<b>Event Upload</b>
Actor	Administrator
Pre-condition	The administrator is logged in to the systems
Description	Administrators upload events with the system.

	<ol style="list-style-type: none"> <li>1. The administrator wants to upload event on the system.</li> <li>2. The administrator clicks login button.</li> <li>3. The system displays login form.</li> <li>4. The administrator fills login form.</li> <li>5. The administrator clicks the login button.</li> <li>6. The system displays a page containing Upload Events button.</li> <li>7. The administrator clicks on the Upload Events button.</li> <li>8. The system displays a page containing Event Up loader Form.</li> <li>9. The administrator then fills in the event upload form and clicks on the upload button.</li> <li>10. Event is uploaded to the system database.</li> <li>11. The use case ends.</li> </ol>
Alternate course of action:	<p>A9) if the administrator fills the form incorrectly. The system displays error message.</p> <p>Return to step 9.</p>
Post-condition	A new Event is created.

Table4. 9 use case description for Upload Course Material

Use case name	<b>Upload Course Material</b>
Actor	Teacher
Pre-condition	The teacher is logged in system.
Description	Teacher uploads course material with the system.

Basic course of action:	<ol style="list-style-type: none"> <li>1. The teacher browses the website.</li> <li>2. The system displays the login form.</li> <li>3. The teacher enters user name and password.</li> <li>4. The teacher clicks login button.</li> <li>5. The teacher sends data to the system.</li> <li>6. The system verifies the requested data.</li> <li>7. The system displays the upload form.</li> <li>8. the teacher fills the form</li> <li>9. The teacher clicks the upload button.</li> <li>10. The data is uploaded to website.</li> <li>11. the use case ended</li> </ol>
Alternate course of action:	<p>A 3). if the teacher inters incorrect user name and password. Return to step 2.</p> <p>A 8) if the teacher fills incorrectly the upload form. Return to step 7.</p>
Post-condition	A new Course material is uploaded.

Table4. 10 use case description for view information.

Use case name	view information
Actor	Administrator
Pre-condition	The Administrator should have user name and password
Description	Administrator view information stored on the data base.

Basic course of action:	<ol style="list-style-type: none"> <li>1) Administrator login into the system.</li> <li>2) System checks the username and password.</li> <li>3) System opens the main page.</li> <li>4) User clicks on the view information link.</li> <li>5) System display view information page.</li> <li>6) User selects status.</li> <li>7) System display selected information.</li> <li>8) User see the displayed information.</li> </ol>
Post-condition	Registering and viewing report to and from database.

#### 4.2.3. Use case Scenario

A use case is an abstraction that describes all possible scenarios involving the described functionality. A scenario is an instance of a use case describing a concrete set of actions, when and where they are working, why they are using the system, and what they need the system to do for them. Scenarios used as example for illustrating Common use cases.

The following are describing scenario of how the user use the systems

##### **For login**

1. User want to login.
2. Enter user name and password.
3. Check the value whether valid or not.
4. If it's not valid display error message to re-enter the correct value.
5. If it's valid send to the data base and success.

##### **For create an account**

1. User wants to login.

2. Enter user name and password.
3. Check the value whether its valid or not.
4. If it's not valid display error message to re-enter the correct value.
5. If it's valid display the main page.
6. Fill the information to create account form.
7. Check the information whether its valid or not.
8. If it's not valid display error message to re-enter the information correctly.
9. If it's valid create an account and send it to the data base.
10. Display successful message to the user.

### **For upload course material**

1. The teacher browses the website.
2. The system displays the login form.
3. The teacher enters user name and password.
4. The teacher clicks login button.
5. The teacher sends data to the system.
6. The system verifies the requested data.
7. The system displays the upload form.
8. the teacher fills the form
9. The teacher clicks the upload button.
10. The data is uploaded to website.
11. the use case ended

### **For Download course material**

1. . The student wants to down load course material.
2. The student enters to the main menu.
3. The system display login form.
4. The student fills user name and password.
5. The student clicks the login button.
6. The student sends data to the system.
7. The system checks the data.

8. The system displays the contents.
9. The student views the content.
10. The student double clicks the content which he/she wants to download.
11. The student downloads successfully.
12. The use case ends.

### 4.3. Object Model

Describe the system in terms of object classes and their associations, with common attributes and the services (operations) provided by each object.

#### 4.3.1 Class Diagram

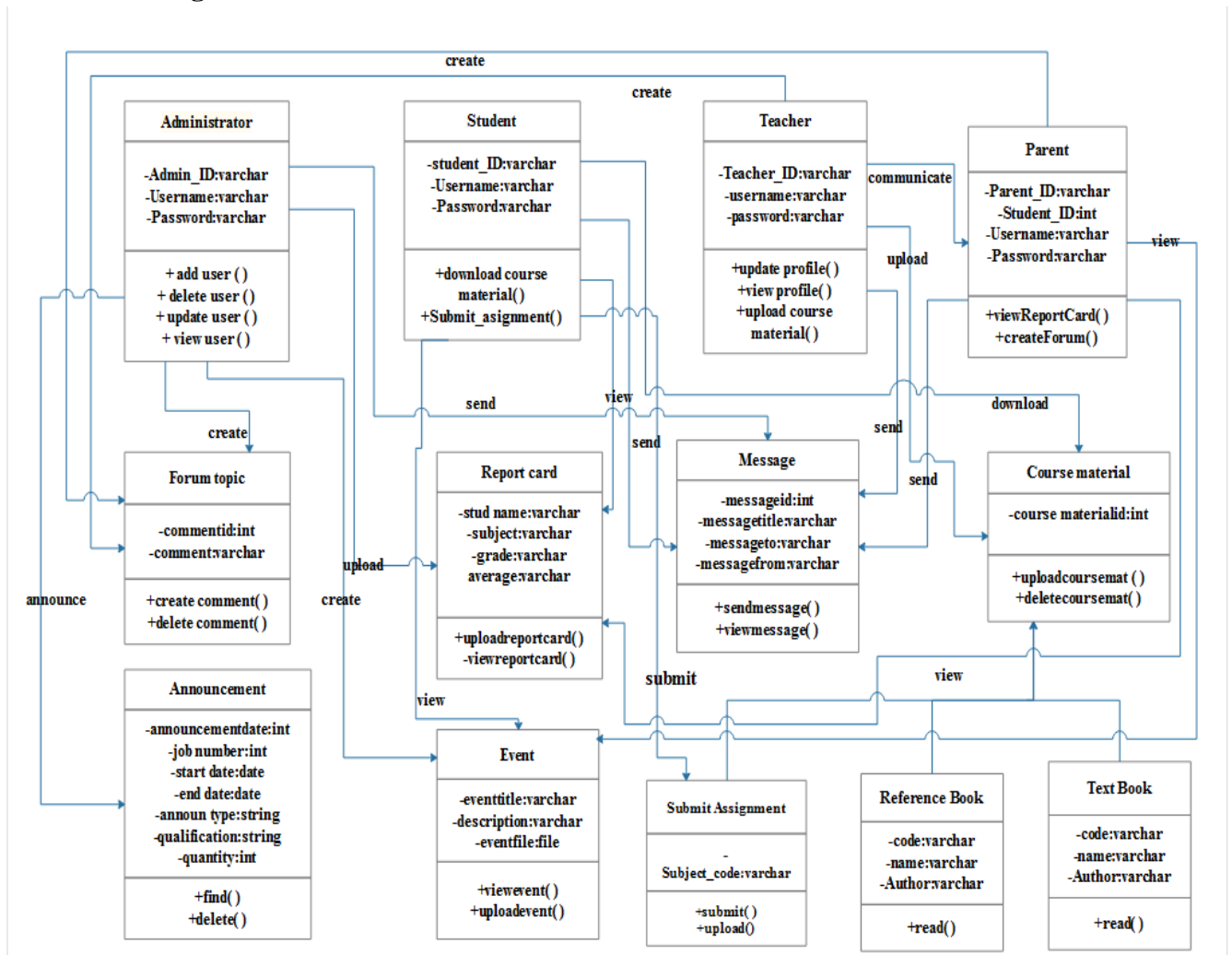


Figure4. 2 Class diagram

### 4.3.2. Data Dictionary

A set of information describing attributes Data type, data size, and key constraints of the identified entities.

Table4. 11 Data dictionary for Admin

Attribute	Data type	Size	Domain value	Format	Constraint
Admin_ID	Varchar	10	Letter, number and character	-----	Primary key
First Name	Varchar	20	Letter	-----	Not NULL
Last Name	Varchar	20	Letter	-----	Not NULL
User Name	Varchar	20	Letter	-----	Not NULL
Password	Varchar	20	Letter, number and character	-----	Not NULL
Sex	Varchar	6	Letter	Male/Female	Not NULL

Table4. 12 Data dictionary for Student

Attribute	Data type	Size	Domain value	Format	Constraint
Student ID	Varchar	10	Letter, number and character	-----	Primary key
First Name	Varchar	20	Letter	-----	Not NULL
Last Name	Varchar	20	Letter	-----	Not NULL
User Name	Varchar	20	Letter	-----	Not NULL
Password	Varchar	20	Letter, number and character	-----	Not NULL
Sex	Varchar	6	Letter	Male/Female	Not NULL

Table4. 13 Data dictionary for Teacher

Attribute	Data type	Size	Domain value	Format	Constraint
Teacher ID	Varchar	10	Letter, number and character	-----	Primary key
First Name	Varchar	20	Letter	-----	Not NULL
Last Name	Varchar	20	Letter	-----	Not NULL
User Name	Varchar	20	Letter	-----	Not NULL
Password	Varchar	20	Letter, number and character	-----	Not NULL
Sex	Varchar	6	Letter	Male/Female	Not NULL

Table4. 14 Data dictionary for Parent

Attribute	Data type	Size	Domain value	Format	Constraint
-----------	-----------	------	--------------	--------	------------

Parent ID	Varchar	10	Letter, number and character	-----	Primary key
Student ID	Varchar	10	Letter, number and character	-----	Foreign key
First Name	Varchar	20	Letter	-----	Not NULL
Last Name	Varchar	20	Letter	-----	Not NULL
User Name	Varchar	20	Letter	-----	Not NULL
Password	Varchar	20	Letter, number and character	-----	Not NULL
Sex	Varchar	6	Letter	Male/Female	Not NULL

Table4. 15 Data dictionary for Message

Attribute	Data type	Size	Domain value	Format	Constraint
Message_ID	Varchar	10	Number	-----	Primary key
Message Title	Varchar	20	Letter	-----	Not NULL
Message_From	Varchar	20	Letter	-----	Not NULL
Message_To	Varchar	20	Letter	-----	Not NULL
Message	Varchar	20	Letter	-----	Not NULL

Table4. 16 Data dictionary for Forum Topic

Attribute	Data type	Size	Domain value	Format	Constraint
Comment_ID	Varchar	10	Number	-----	Primary key
Comment	Varchar	20	Letter	-----	Not NULL

Table4. 17 Data dictionary for Report Card

Attribute	Data type	Size	Domain value	Format	Constraint
Student Name	Varchar	10	Number	-----	Primary key
Subject	Varchar	20	Letter	-----	Not NULL
Grade	Char	1	Letter	-----	Not NULL
Average	Double	20	Letter	-----	Not NULL

## 4.4. Dynamic Model

### 4.4.1. Sequence Diagram

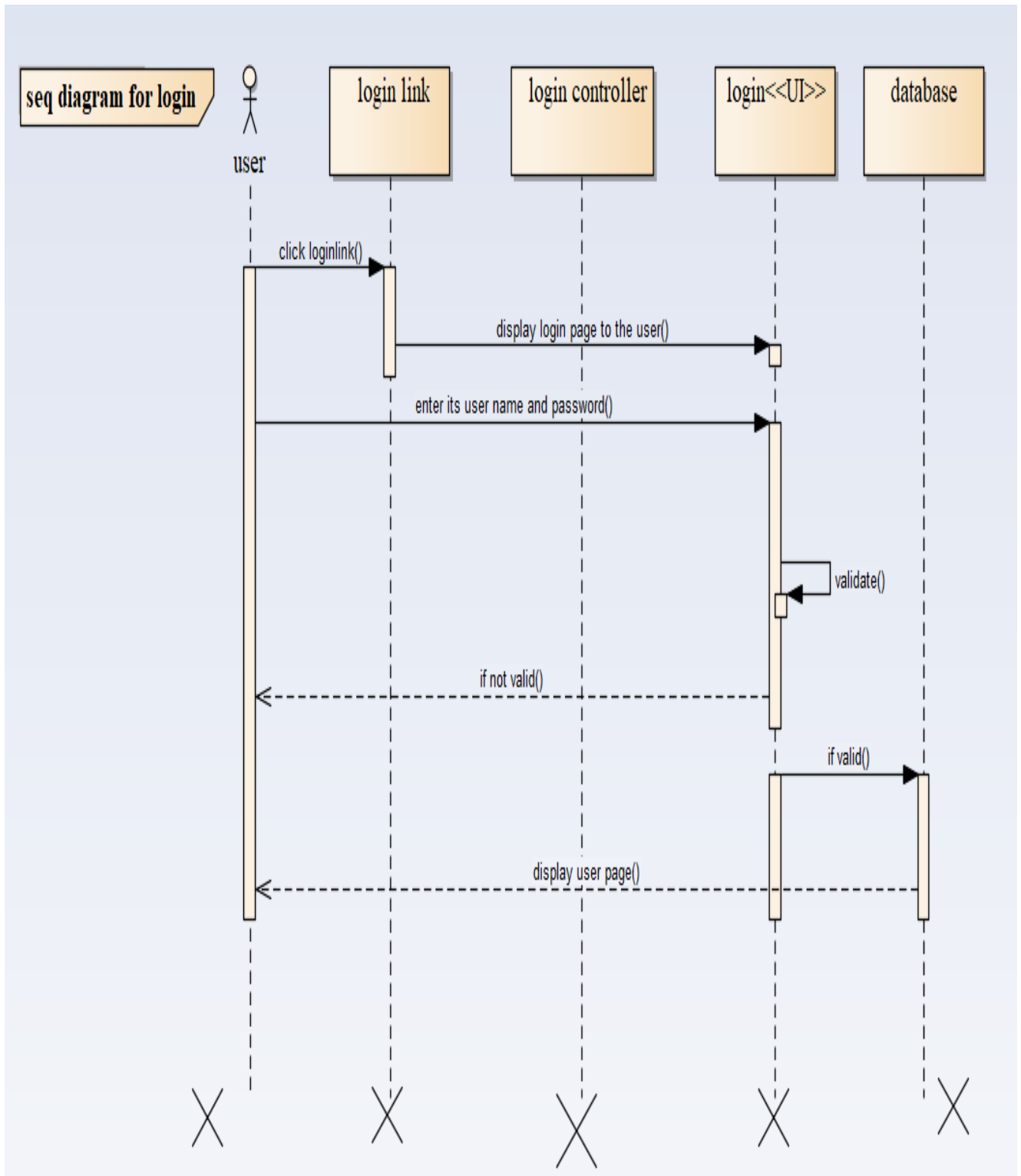


Figure4. 3 Sequence Diagram for Login

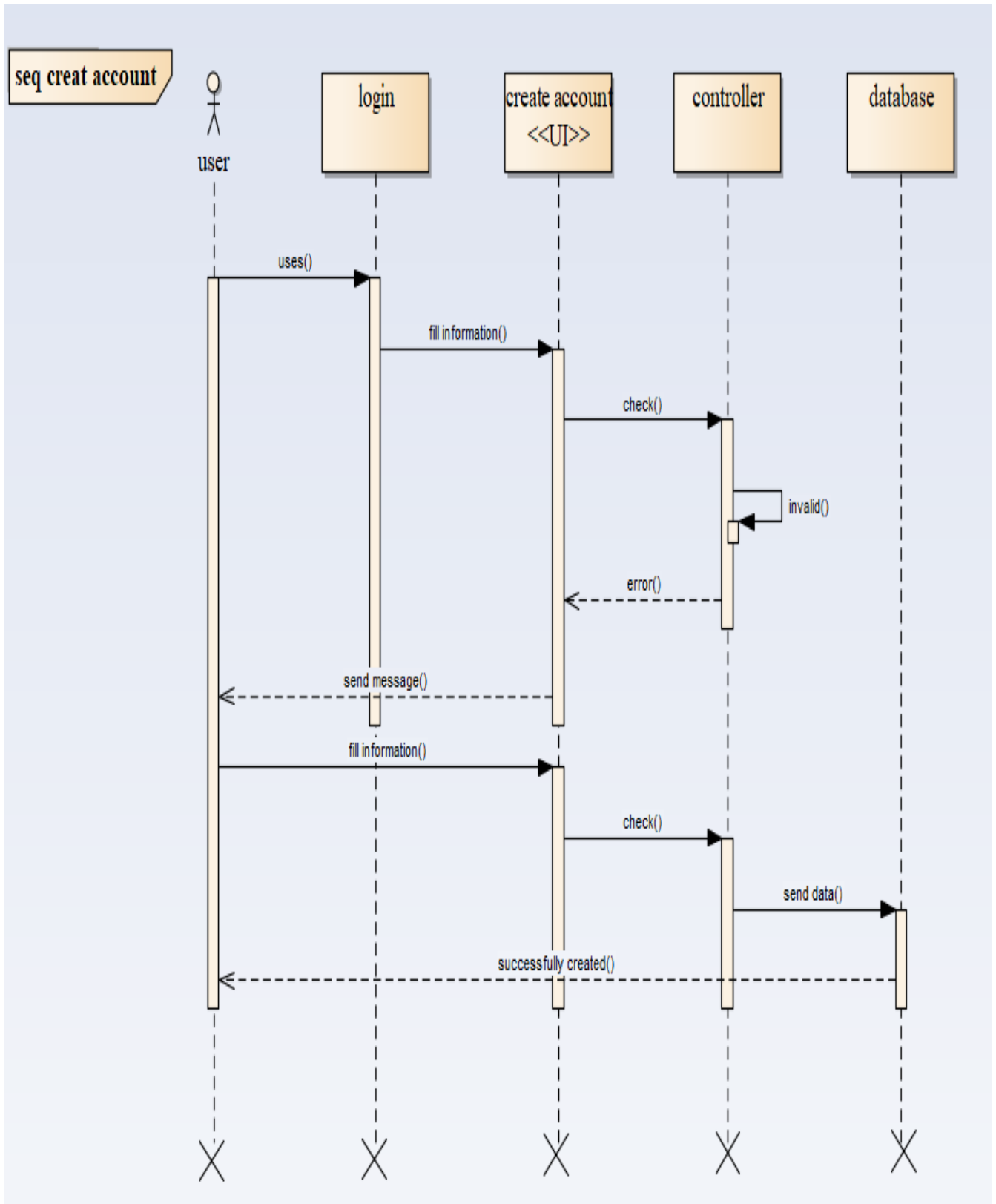


Figure4. 4 Sequence Diagram for Create Account

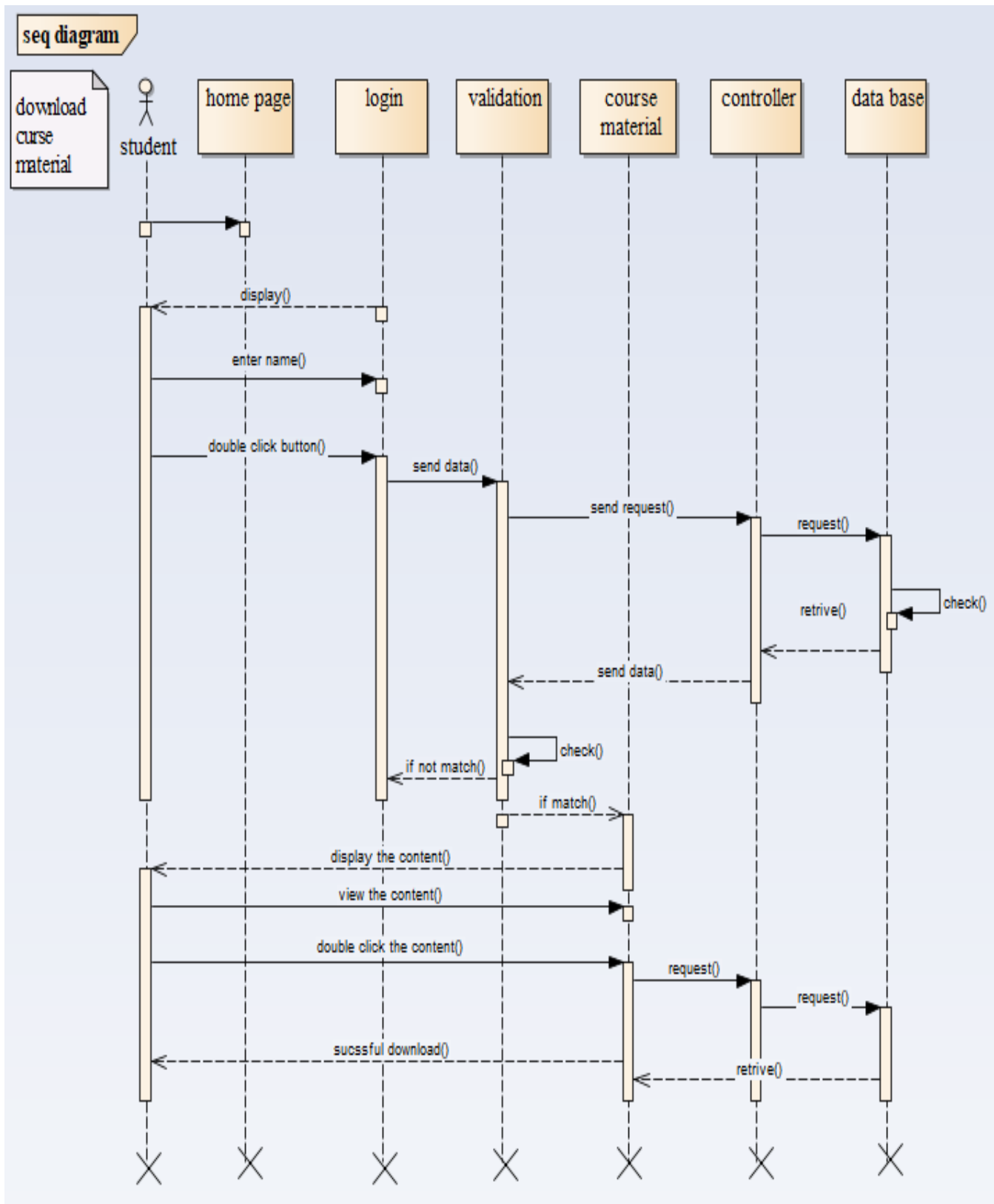


Figure4. 5 Sequence Diagram for Download Course Material

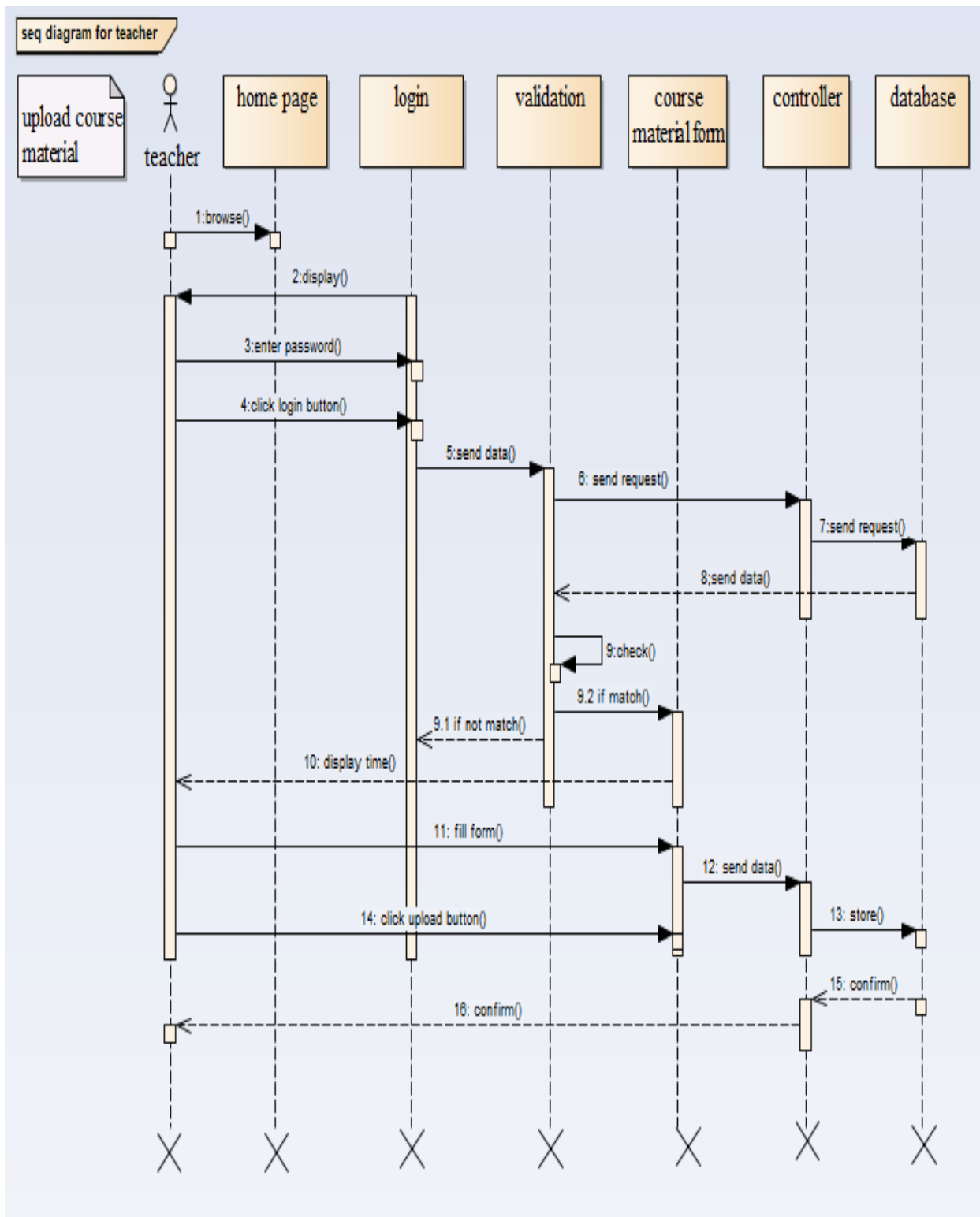


Figure4. 6 Sequence Diagram for Upload Course material

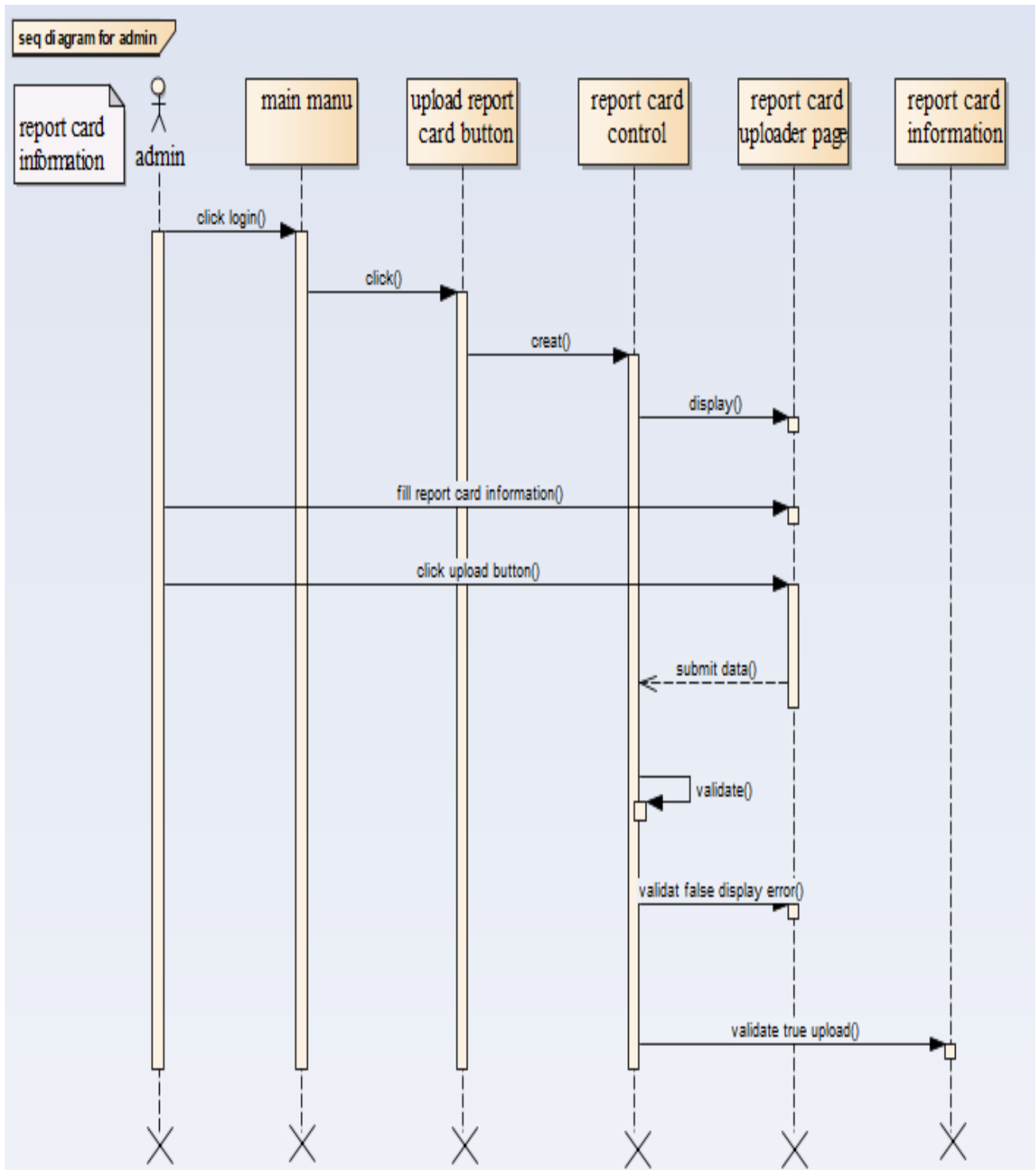


Figure4. 7 Sequence Diagram for Upload Report Card

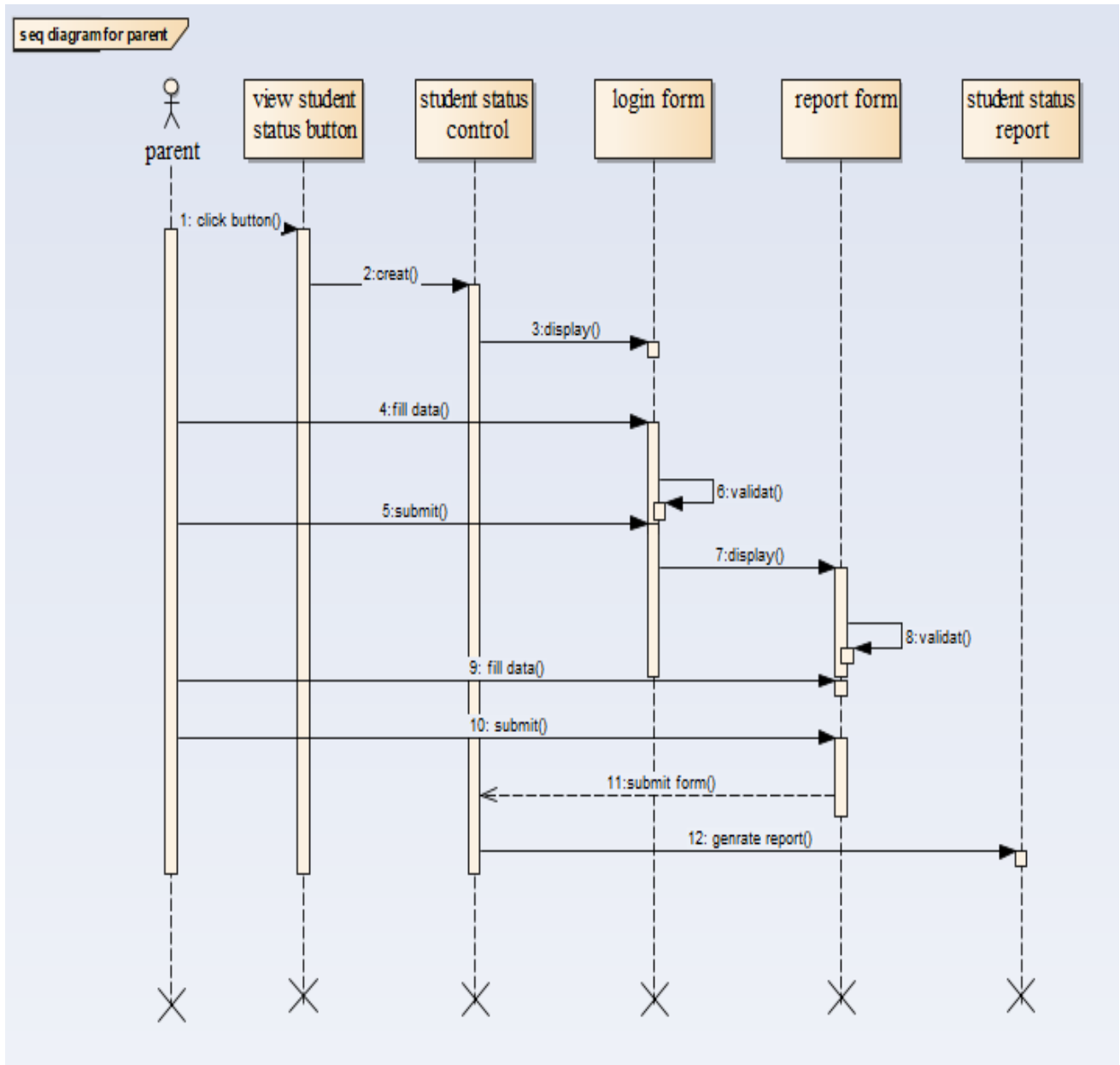


Figure4. 8 Sequence Diagram for View Report card

#### 4.4.2. Activity Diagram

An Activity diagram is UML diagram which shows the interaction/flow of procedures involved in each process. Illustrate graphical representations of workflows of stepwise activities and actions (htt4)

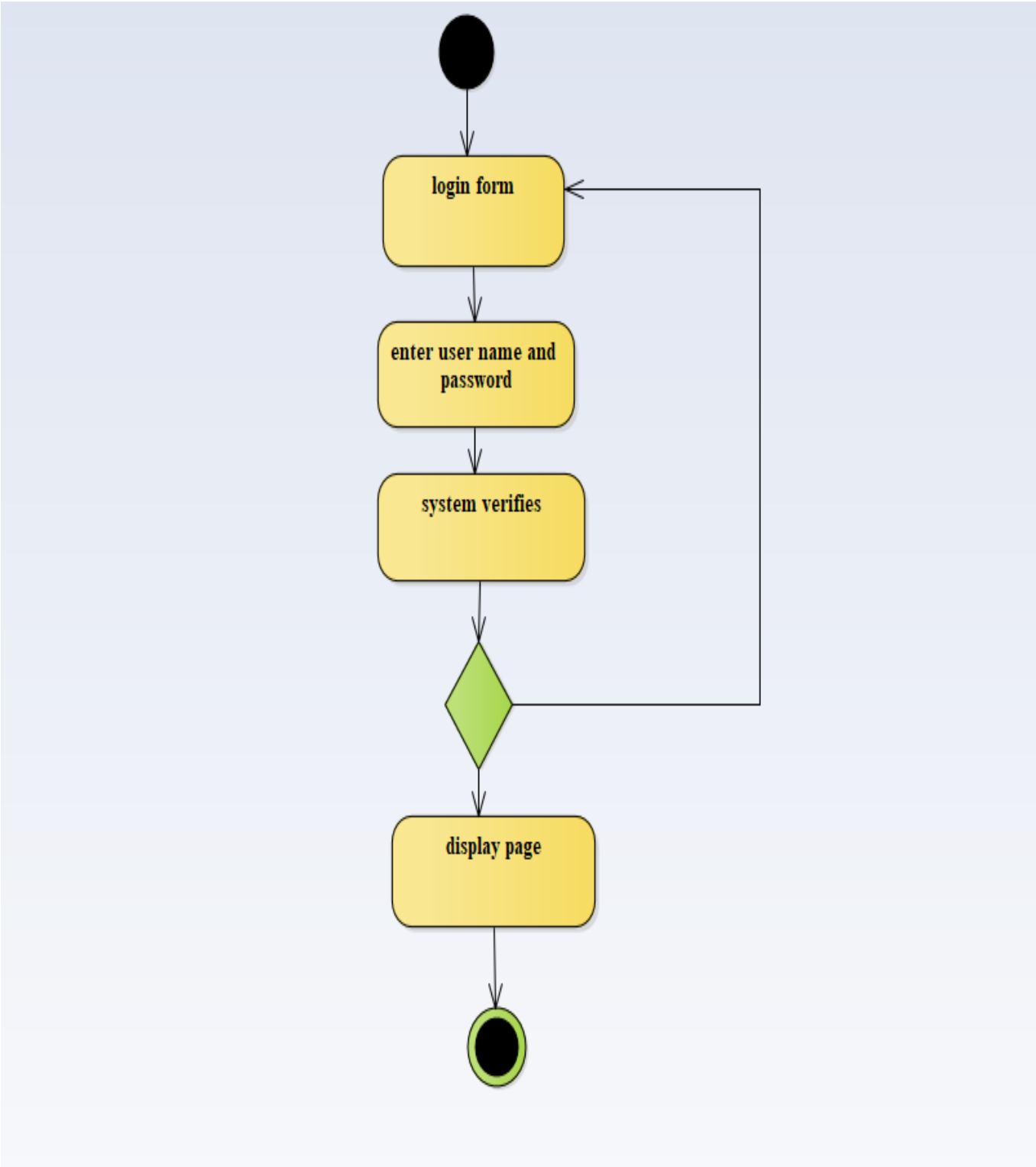


Figure4. 9 Activity diagram for login

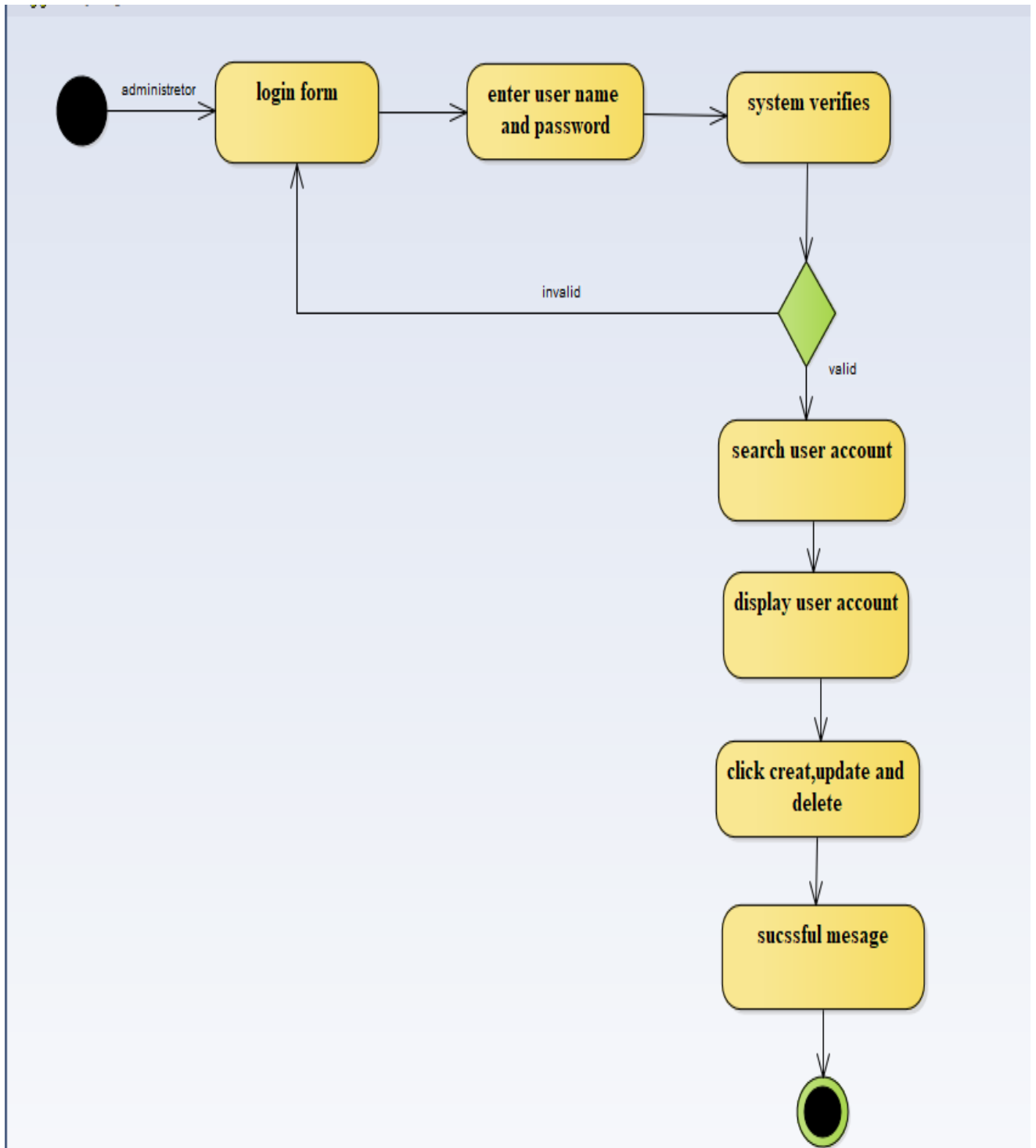


Figure4. 10 Activity diagram for Manage user account

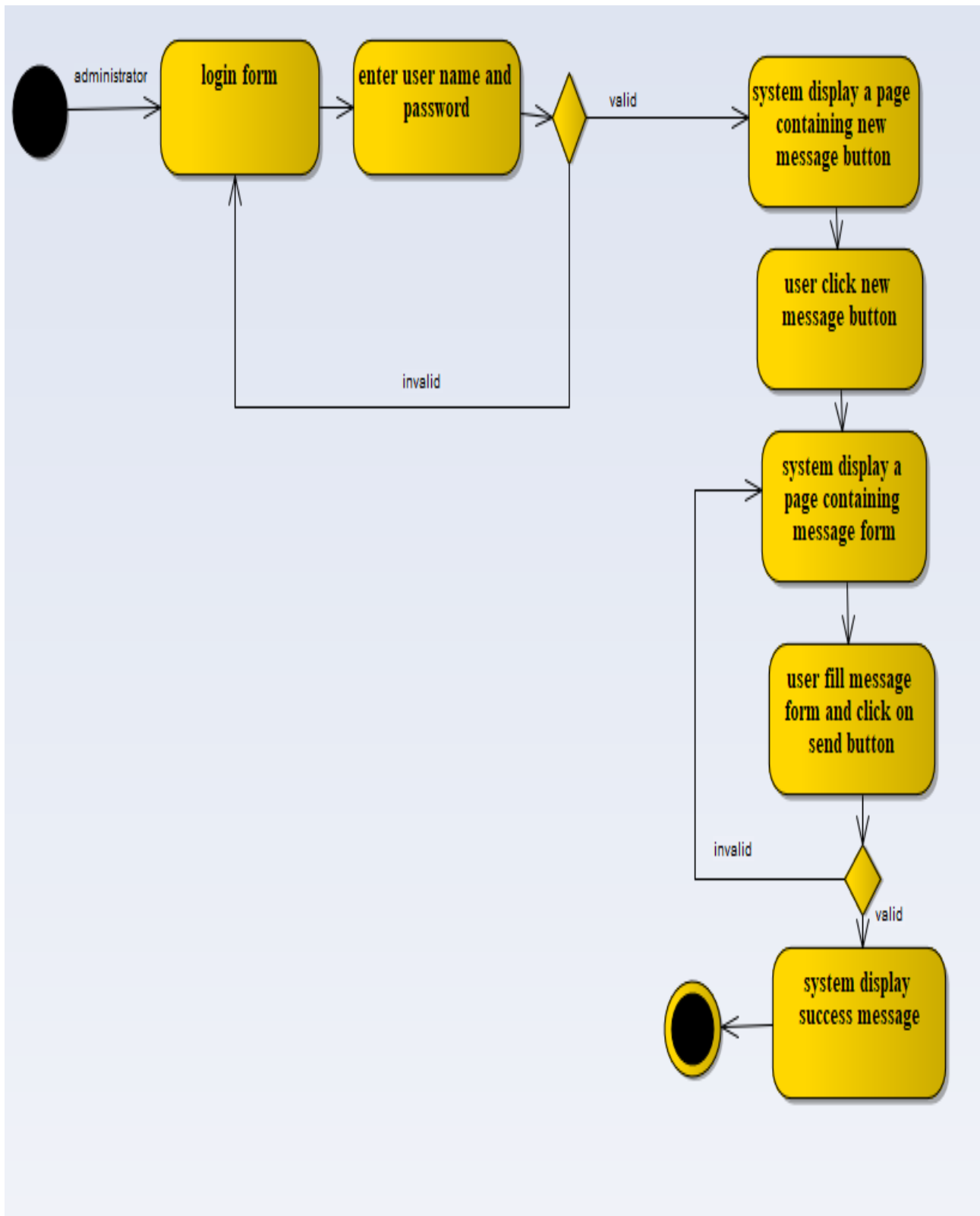


Figure4. 11 Activity diagram for send message

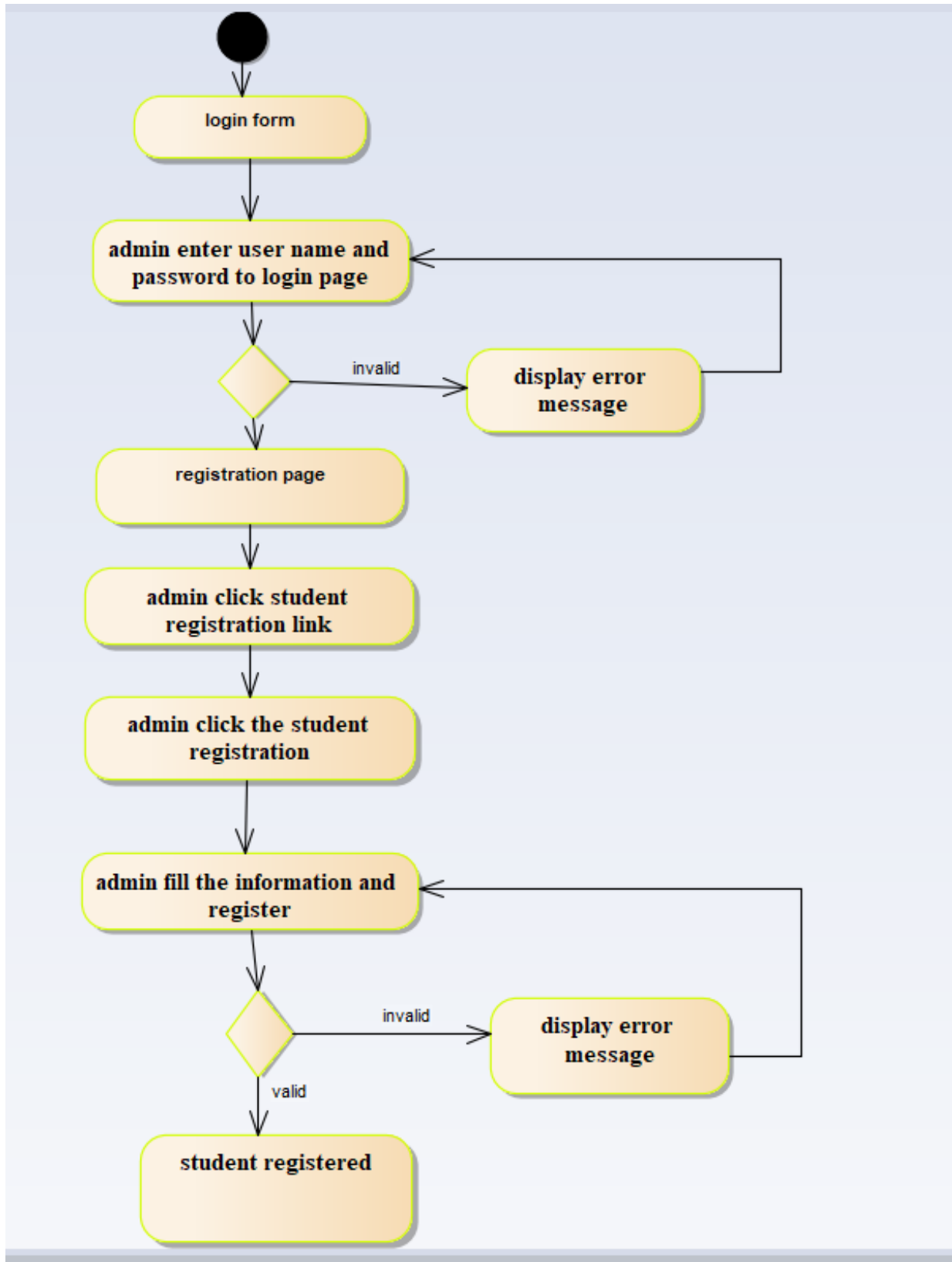


Figure4. 12 Activity diagram for student registration

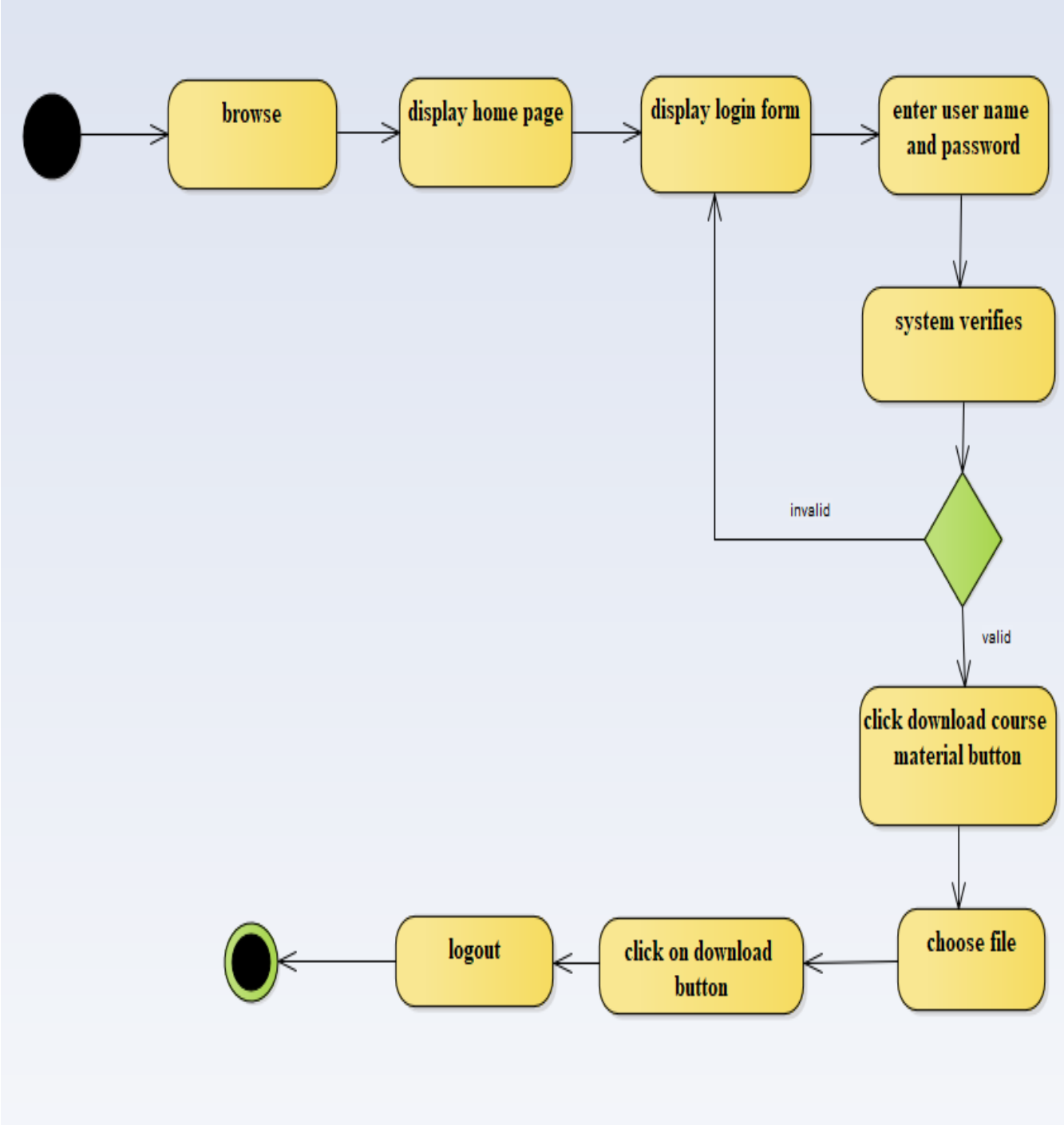


Figure4. 13 Activity diagram for download course material

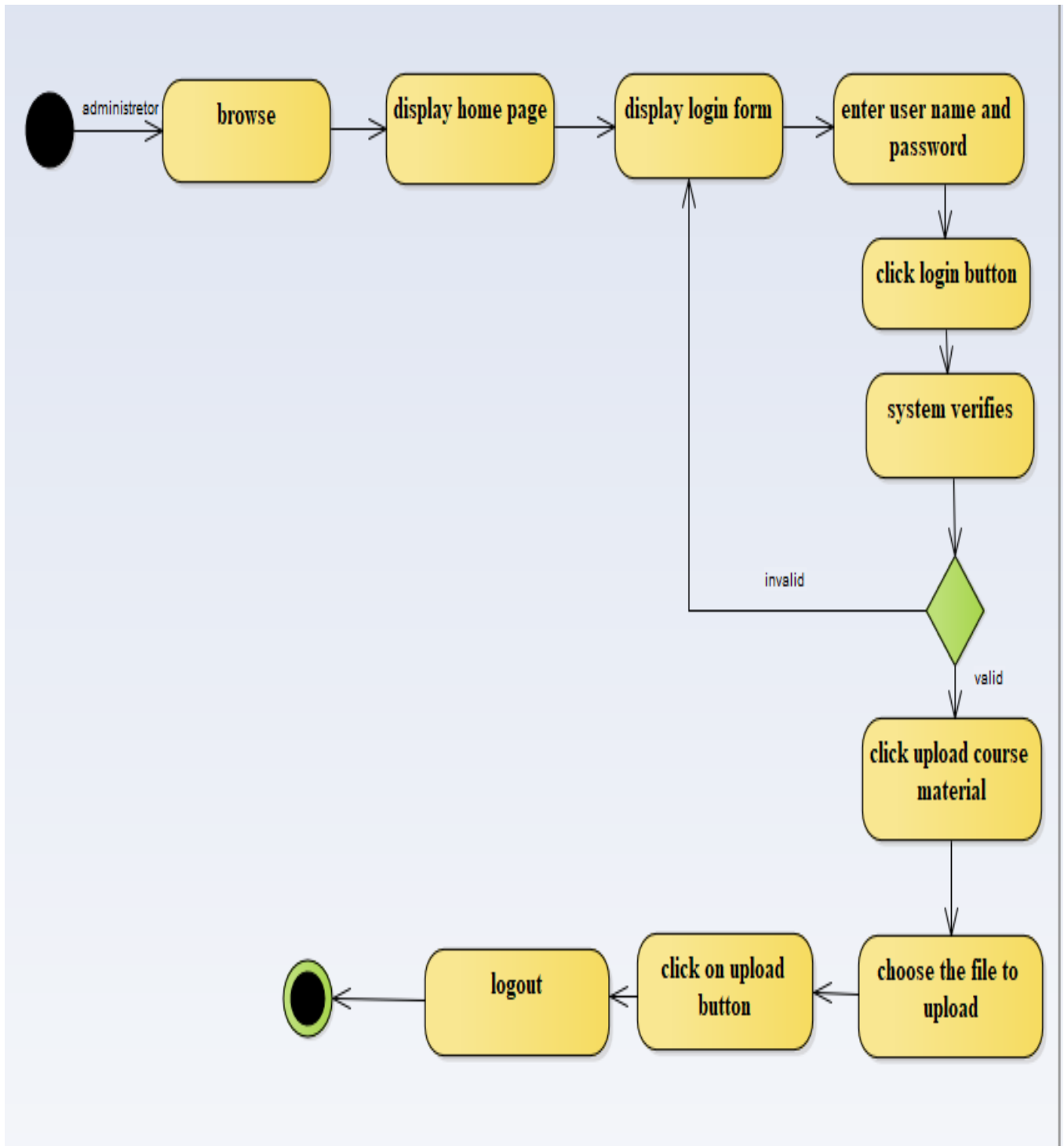


Figure4. 14 Activity diagram for upload course material

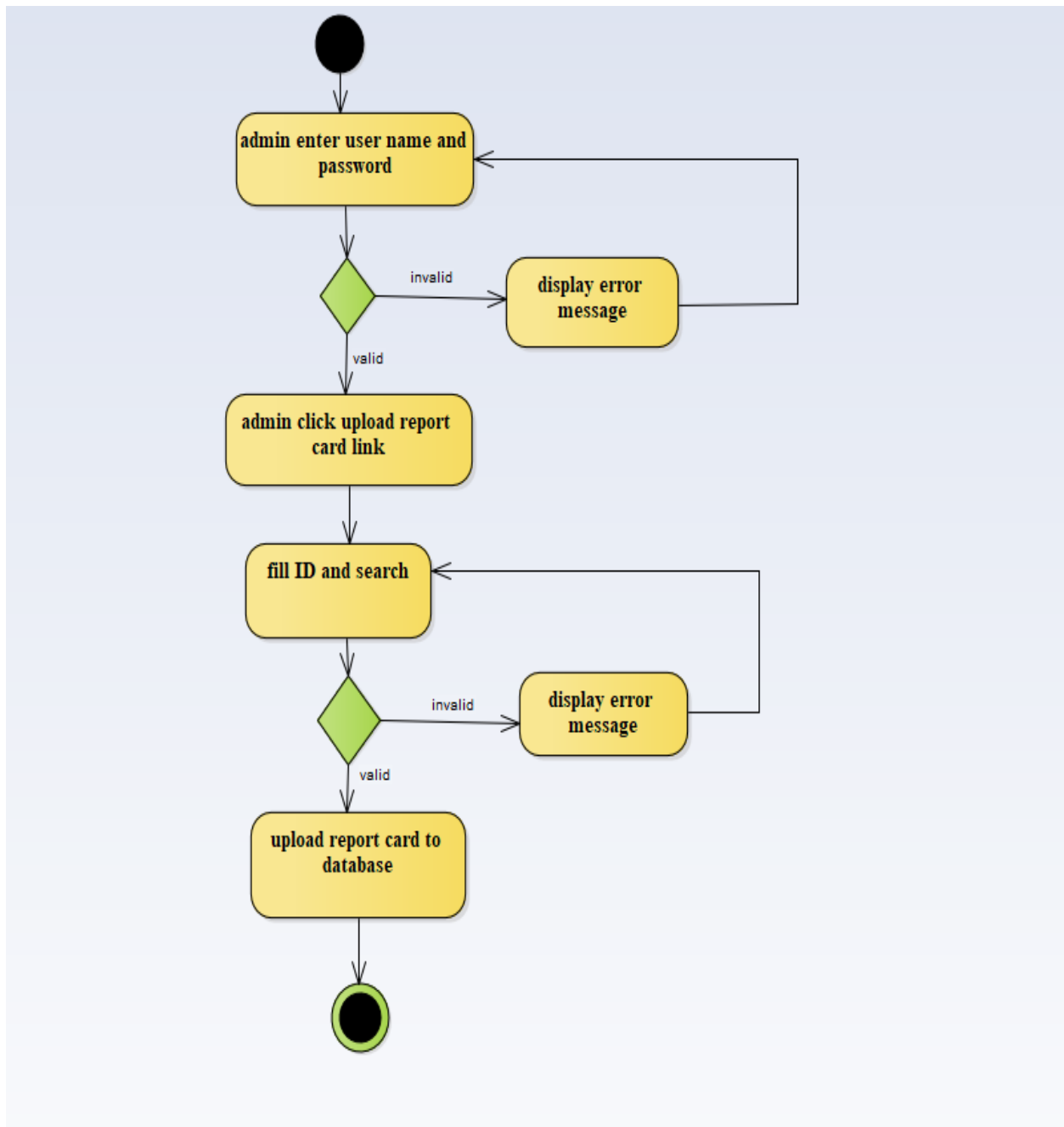


Figure4. 15 Activity diagram for upload report card

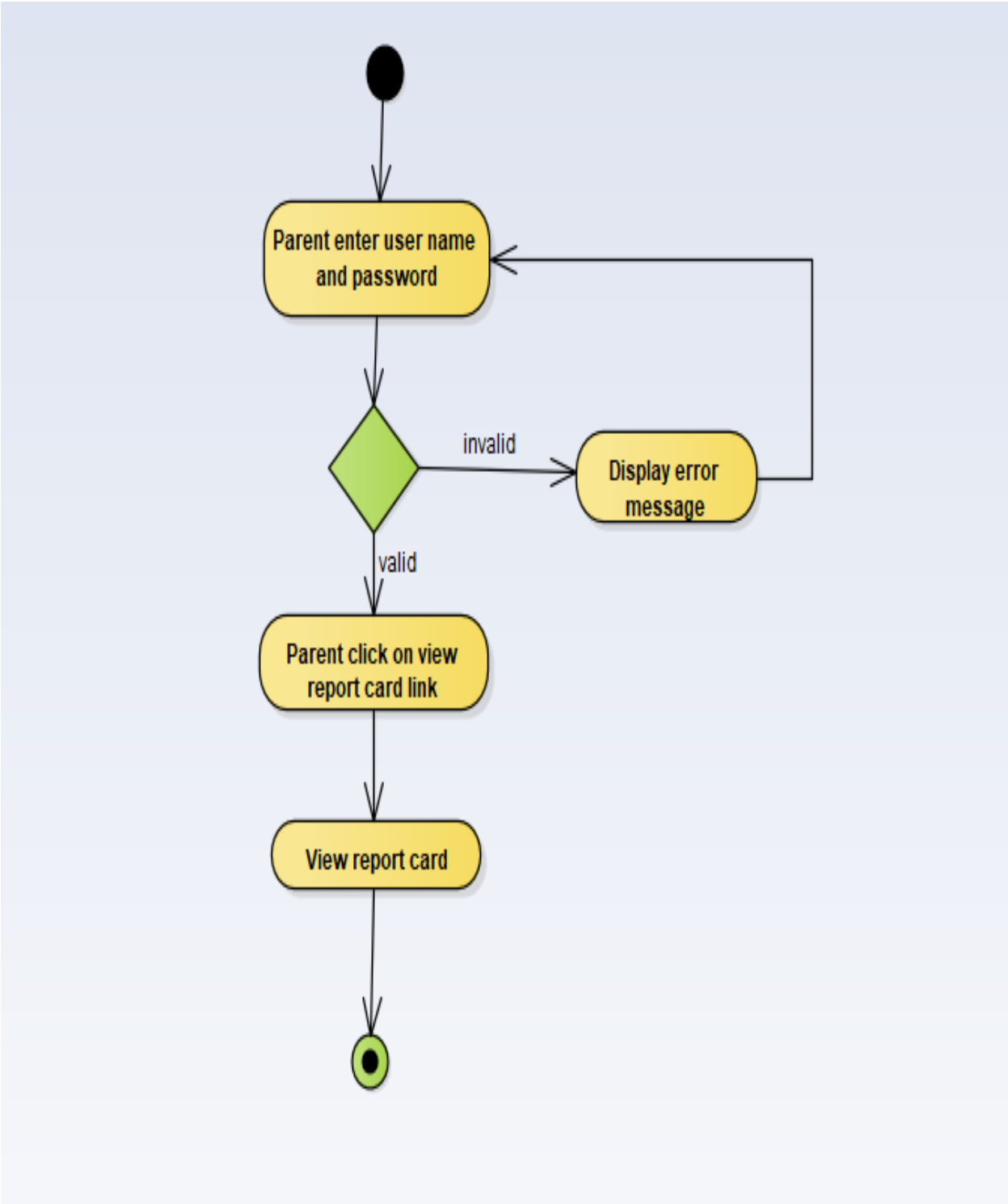


Figure4. 16 Activity diagram for view report card

4.4.3. State Chart Diagram

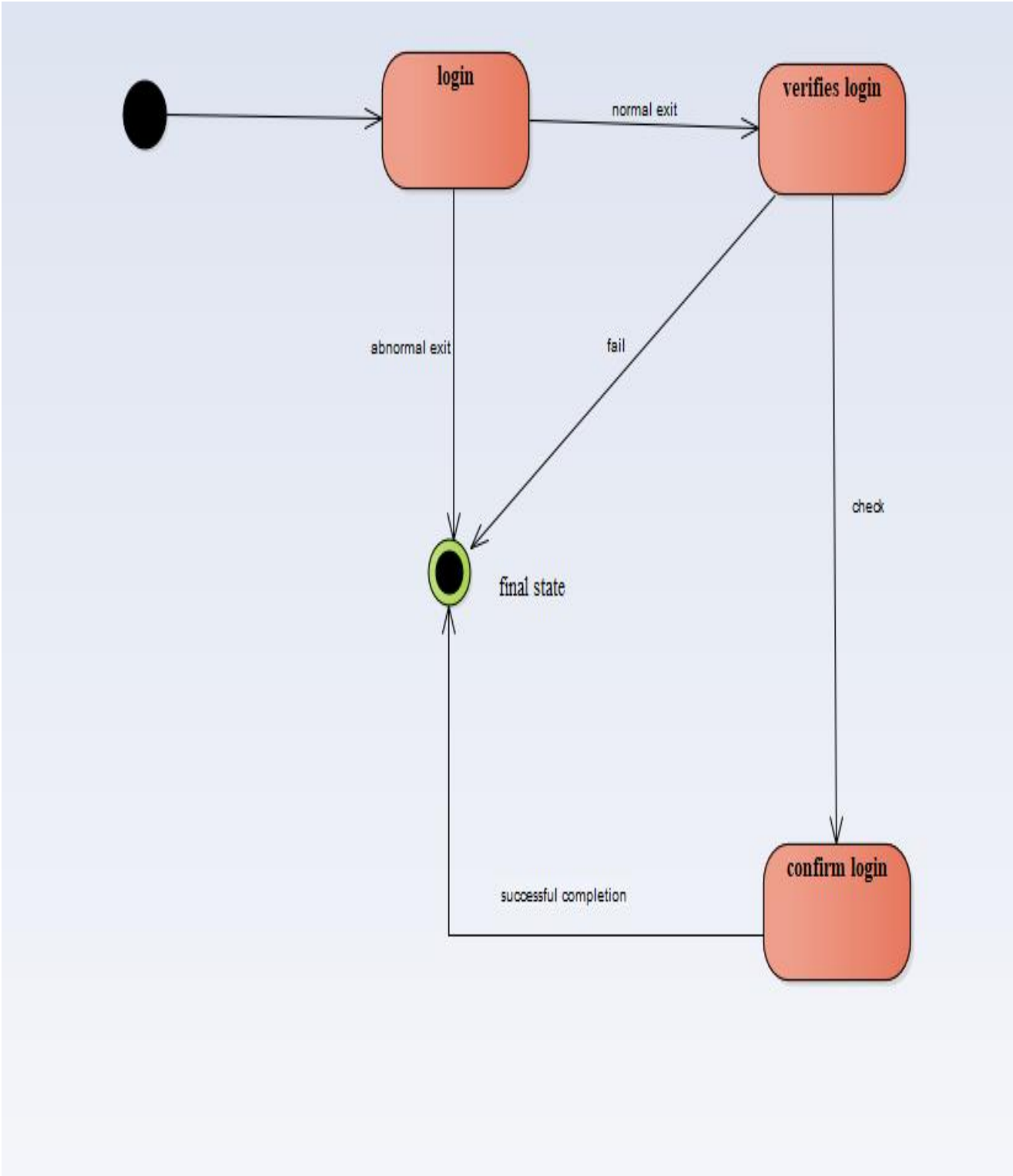


Figure4. 17 State Chart diagram for Login

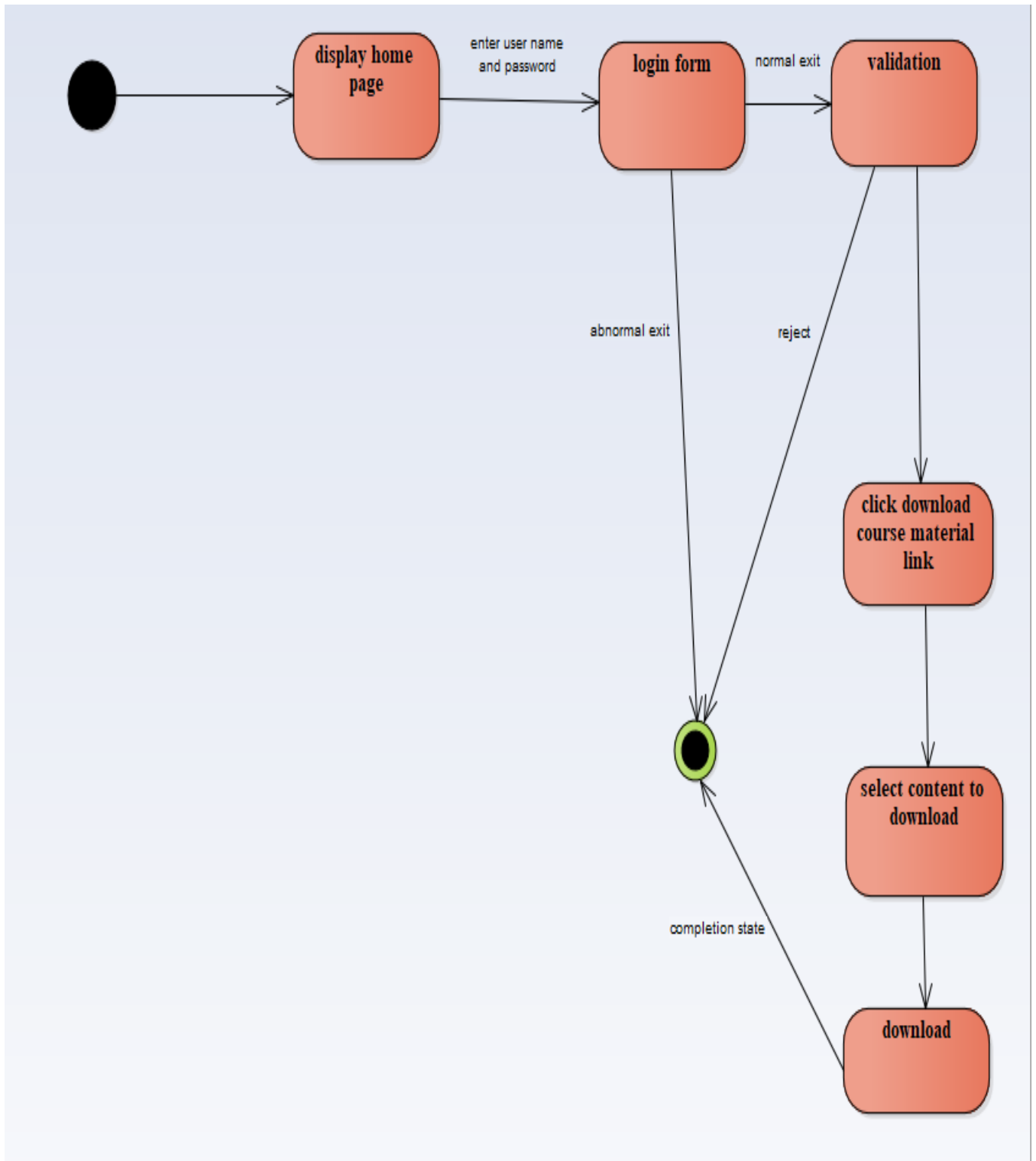


Figure4. 18 State Chart diagram for Download Course Material

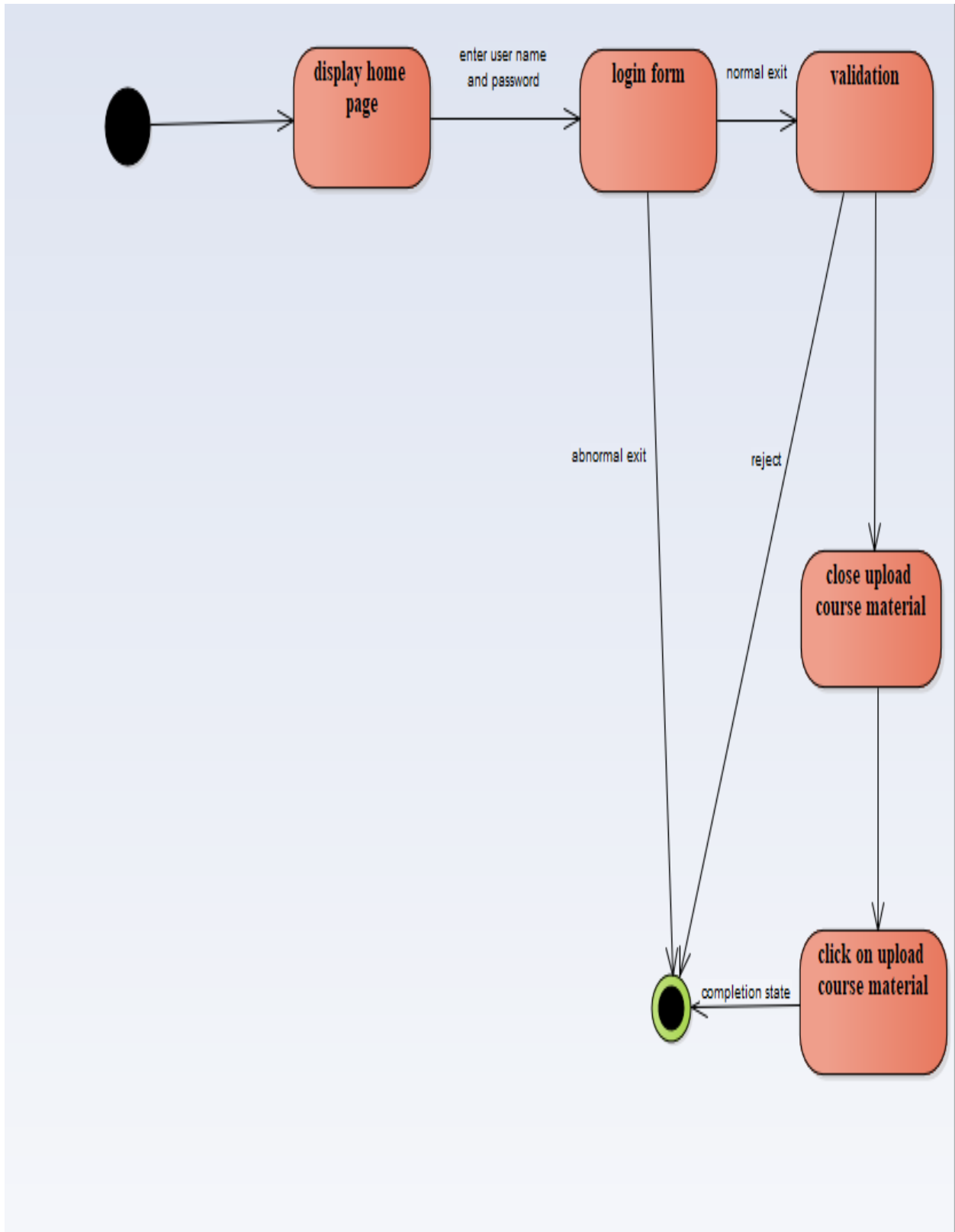


Figure4. 19 State Chart diagram for Upload Course material

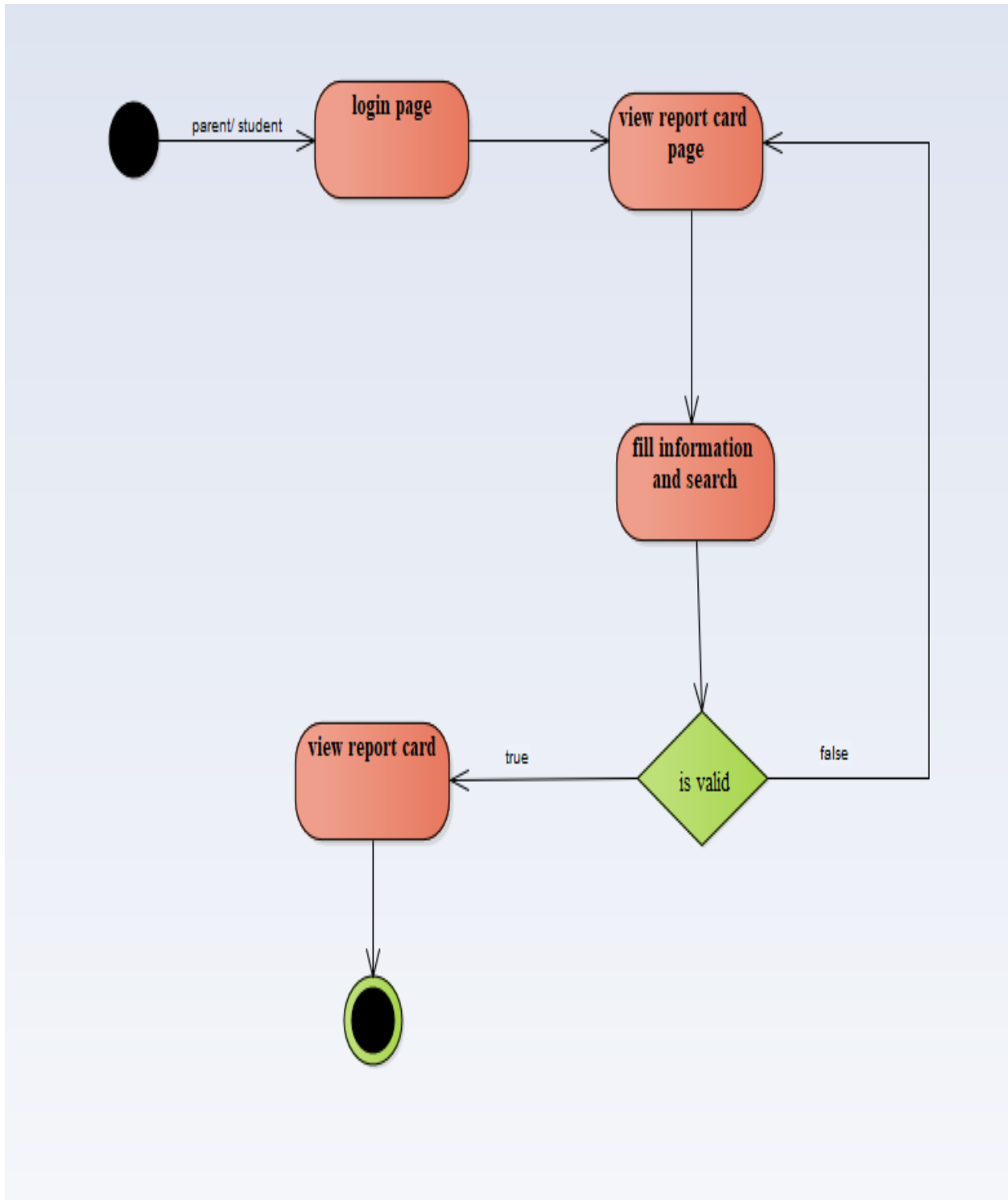


Figure4. 20 State Chart diagram for View Report card

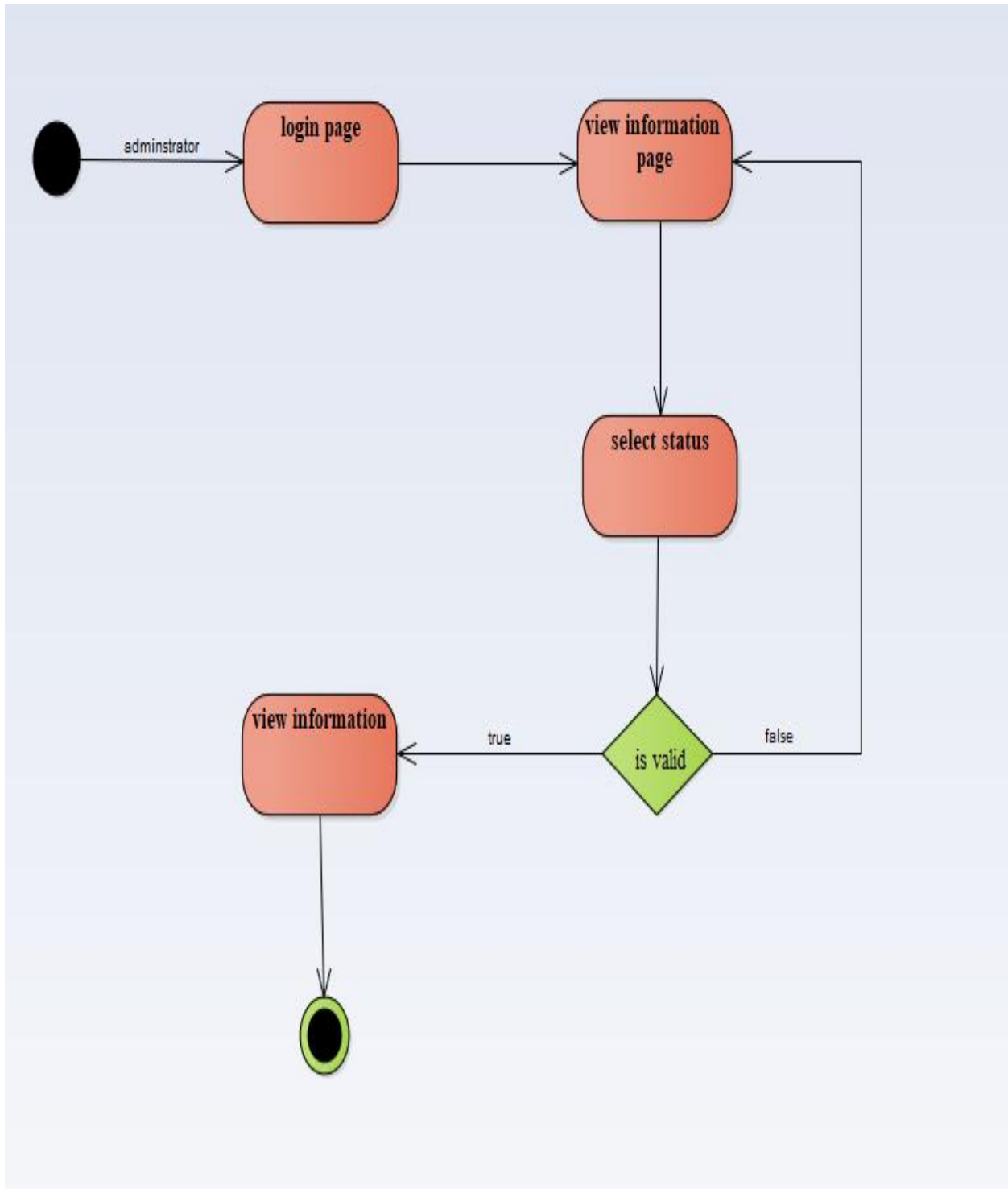


Figure4. 21 State Chart diagram for View Information

# CHAPTER FIVE

## SYSTEM DESIGN

### 5.1. Introduction

In this chapter provide a brief overview of the design goals, current and proposed software architecture, Hardware/software mapping, detailed class design and database design, Persistent data management and control and Security. Systems design is the process of defining the architecture, components, modules, and interfaces for a system to satisfy specified requirements. Design of software involves conceiving, planning out and specifying the externally observable characteristics of the software product. System design is one of the activities that are required to build and verify software. Design is the only way that we can accurately translate customer's view into finished software product or system.

### 5.2. Design Goals

The objectives of design are to model the system with high quality and efficiency. The design goals are derived from non-functional requirements. Design goals describe the qualities of the system that the user gets.

#### 5.2.1. User Interface and Human Factors

The user interface of our system should have a user-familiar and attractive interface so the system should easy and understandable for every user. To increase understandability of the system, we will use bootstrap (icon), button, text field, and preferable and attractive color user interface (CSS).

#### 5.2.2. Hardware Consideration

The system should run in any desktop and personal computer with any flat form because we will use PHP server-side scripting language to implement the system that support platform independent. And needs server to start the system and contain database.

Hardware requirement include

- Computer with 4GB RAM and above
- Server at list one

### **5.2.3. Security Issues**

Our system keeps security of school information management system related by using effectively different mechanisms like: -

- Database security: this is done to protect unauthorized user from access data and information by giving appropriate privilege to each user according their role.
- We secure the system by using cryptograph like md5 or md7 encryption.

### **5.2.4. Performance Consideration**

The system will have good performance and very fast since to increase the performance of the system by using indexes search from database and front-end validation and Best (less time complexity) algorithm. And the system shall support use of multiple users at the same time.

### **5.2.5. Error Handling and Validation**

Our system has some rules (validation issue) that means server with PHP and client side with JavaScript validation and error handling methods to reduce input fault, the system will allow the user interface generate Error message should contain necessary information about the error. The error message is be concise and a short description that is meaningful for user and gives a clear idea of the problem and how to resolve it. Enable the user to confirm before creation, deletion or modification occurs. Respond to error inputs by asking the user to re-enter data in the correct format. So, our system has high levels of error correction and input validation by using database constraint and data types that entered to the system as input.

### **5.2.6. Quality Issues**

#### **5.2.6.1. Availability**

Our system will be available for 24 hours, if intranet network available.

### **5.2.7. Backup and Recovery**

The system should provide a backup and restore mechanism for solving accidental failure.

- The user should take back up. Back up will be saved in backup server. For best results, Backup copies are made on a consistent, regular basis to minimize the amount data lost.
- Take back up from database weekly using email, One Drive and hard copy.

### **5.2.8. Resource Issues**

We develop the system minimize wastage of resource like reducing human power. Tomorrow the System use computer, flash, server, etc.

### **5.2.9. Documentation**

We prepare user manual document for system users to simplify access the system.

### **5.3. Current System Architecture**

We know that at the current system there is no any software or system is deployed.

### **5.4. Proposed System Architecture**

Our proposed system is consisting of 3-tier architecture namely presentation layer, business logic layer, and data layer. The presentation layer is client layer and topmost layer of the application. This is the layer we see when we use this system. It is the interface to our system which takes information from the user. The main functionality of this layer is to communicate with the application layer. This layer passes the information which given in terms of keyboard action and mouse click to the application layer. Example when user want to login into the system first see two text box and login button to enter user name and password and click on login. Business logic layer it is application layer which interacts with data layer and sends information retrieved from database toward to presentation layers.it act as the mediator between presentation layer and data layer. From above example once, the user clicks on login button application layer interact with the database and send information towards to presentation layer. The third one is data layer which used to store data entered by the user. In general client of our system use browser to access the system using the internet.in this case when the user enters input and takes certain action application server process client request to interacting with the database server.

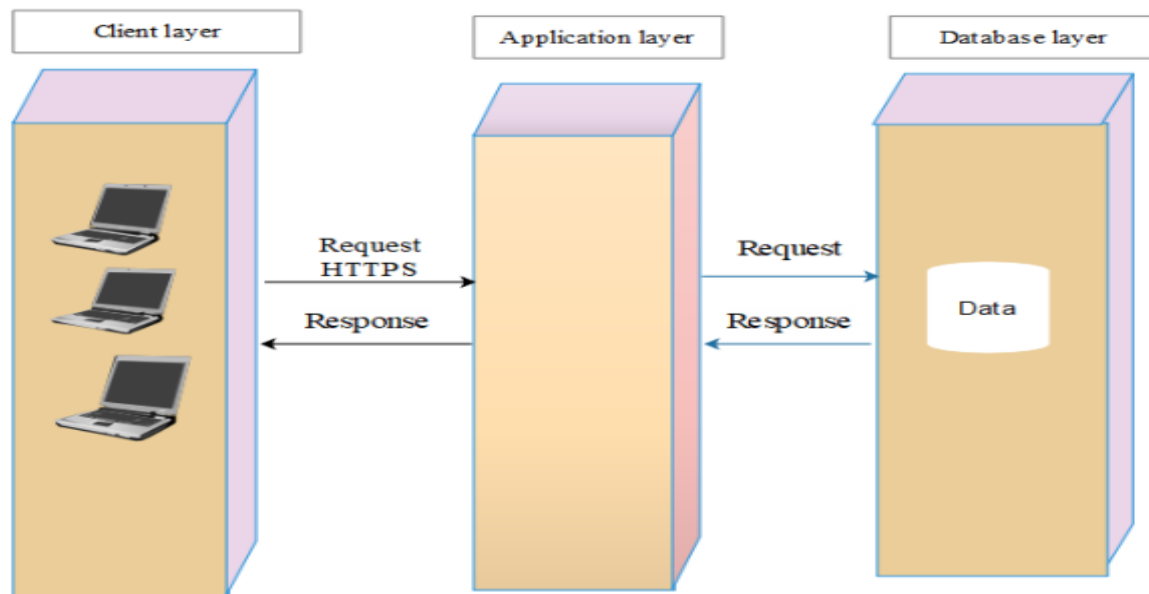


Figure5. 1 Proposed System Architecture

#### 5.4.1. Subsystem Decomposition and Description

- ✓ Administrator Subsystem: - Responsible for action that administrator performs.
  - Create Forum
  - Create Accounts
  - User Registration
  - Upload information
  - Create Announcement
  - Create event
- ✓ Student Subsystem: - Responsible for action that Student performs.
  - View Report card
  - Download Course material
  - Submit Assignment
  - Send Message
  - View an event
  - Make a payment
  - View Event
- ✓ Teacher Subsystem: - Responsible for action that Teacher performs.
  - Upload Course material
  - Send Message
  - Create forum Topic
  - View Event
  - Login
  - logout

- ✓ Parent Subsystem: - Responsible for action that Parent performs.
  - View Report Card
  - View message
  - Send message
  - View Event
  - Create forum topic
  - Login
  - logout

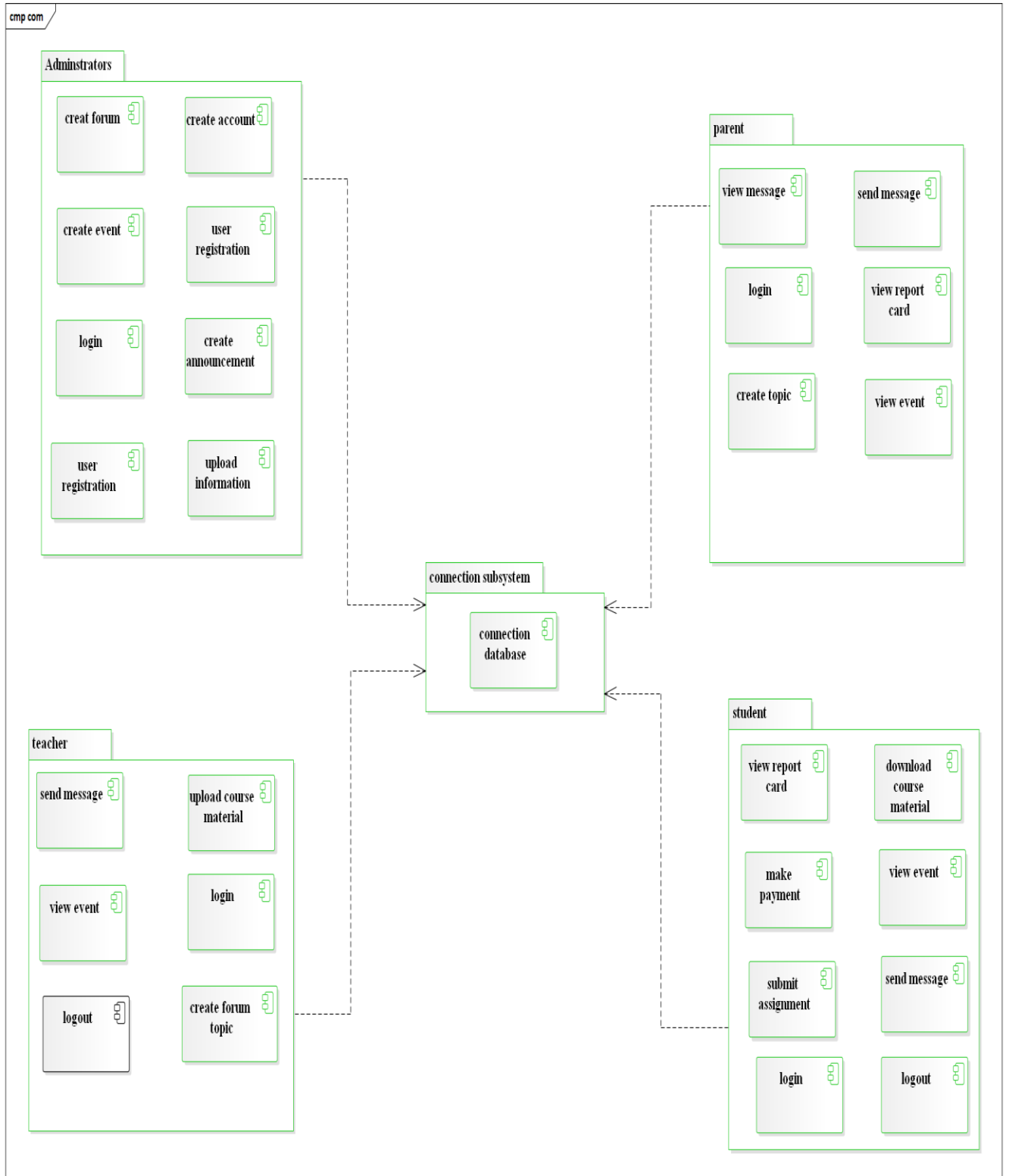


Figure5. 2 Component Diagram

### 5.4.2. Hardware/software Mapping

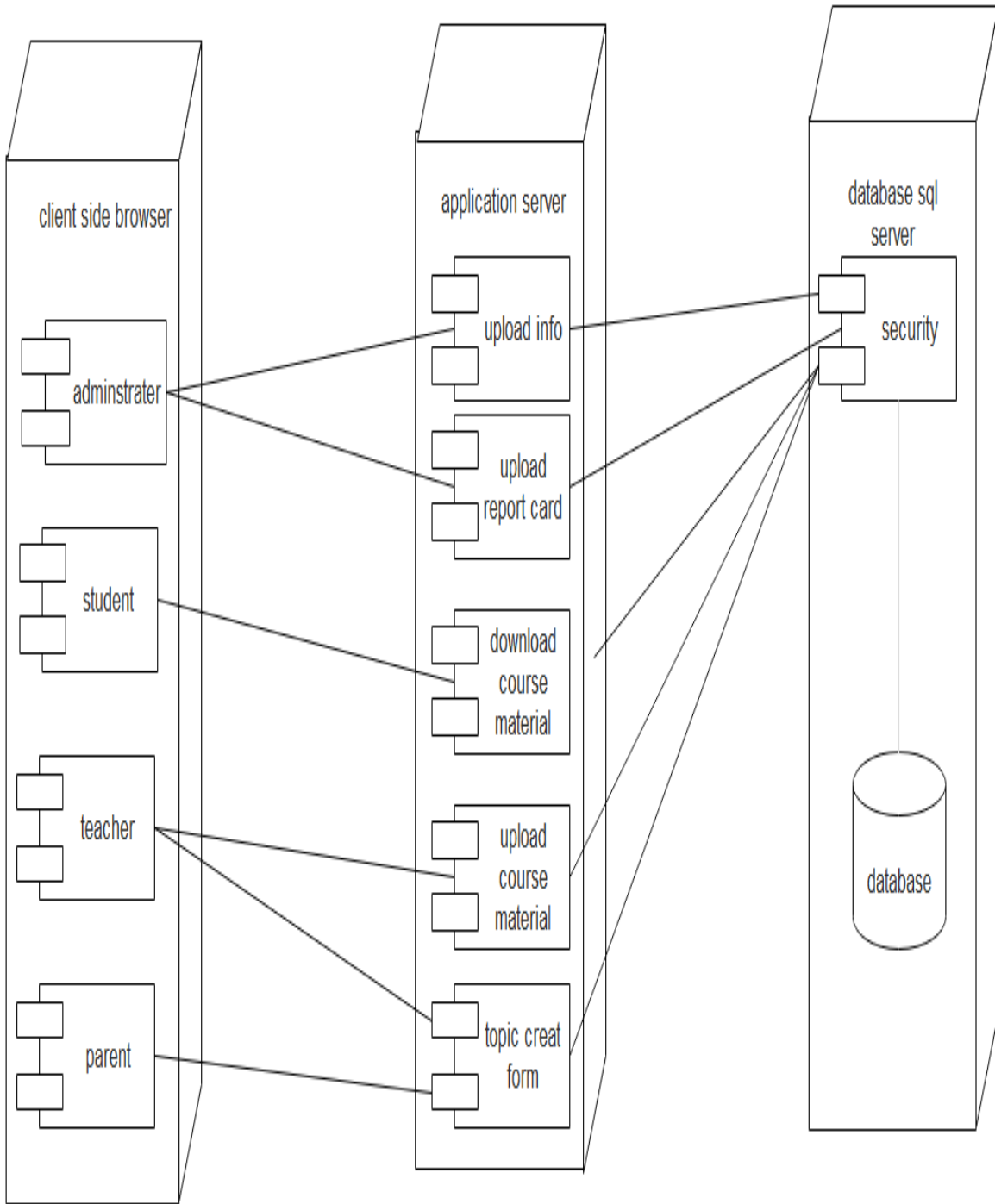


Figure5. 3 Deployment Diagram

### 5.4.3. Detailed Class Diagram

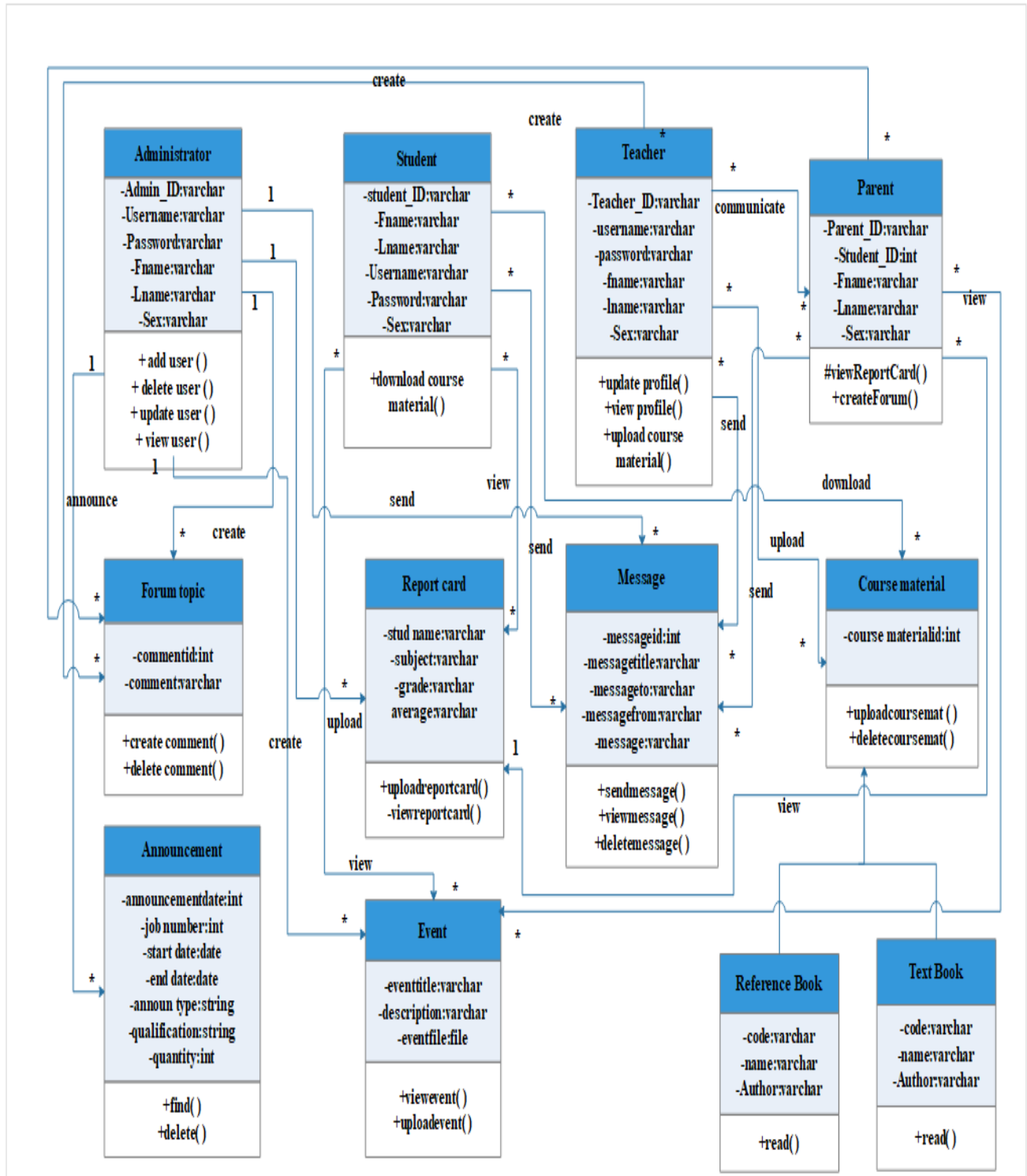


Figure5. 4 Detailed Class Diagram

### 5.4.4. Persistence Data Management

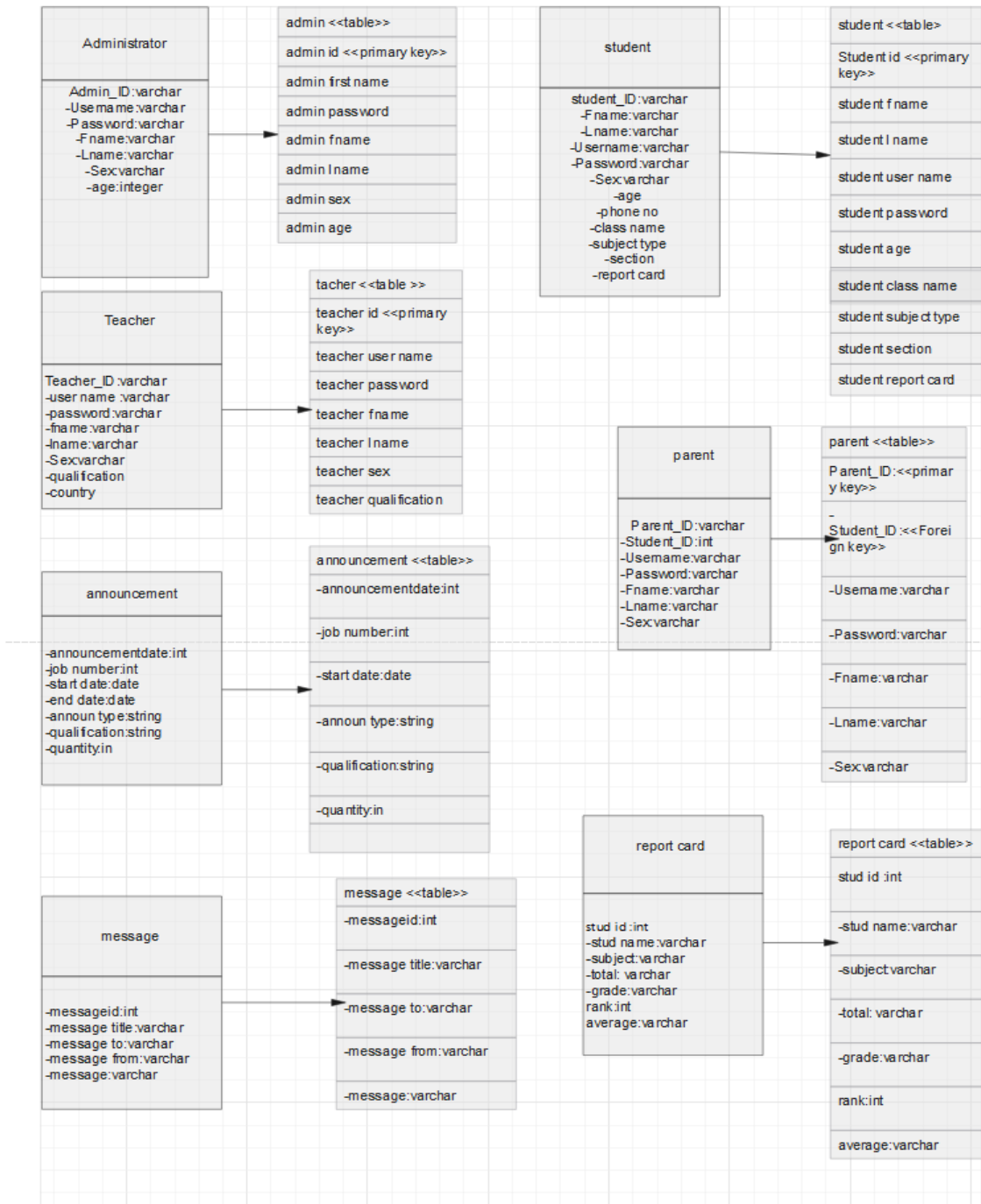


Figure5. 5 Persistence Diagram

### 5.4.5. Access Control and Management

In our system, 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 username and password. System admin represents an authenticated user. 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.

Table5. 1 Access Security Table

Function/Actors	Administrator	Student	Teacher	Parent
Manage Account	✓			
Create Forum	✓		✓	✓
View Report Card		✓		✓
User registration	✓			
Upload Course Material			✓	
Download Course material		✓		
Submit Assignment		✓		
View Event		✓		✓
Send Message			✓	✓
Create Announcement	✓			
Upload Information	✓			

#### 5.4.6. 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. In our

system we category into database management package, report management package, notification management package, account management package, resource management package.

### Database management package

The database subsystem will be implemented by relational database management system which is used to store the persistent data.

### Report management package

It is a subsystem that helps administrator to handle the overall information about the system and Users are needs to know the overall information status of the system.

### Notification Management

It is a sub system that helps communicates between each other and provides notification service.

### Resource Management

It is a sub system that helps resource sharing between each other and provides service.

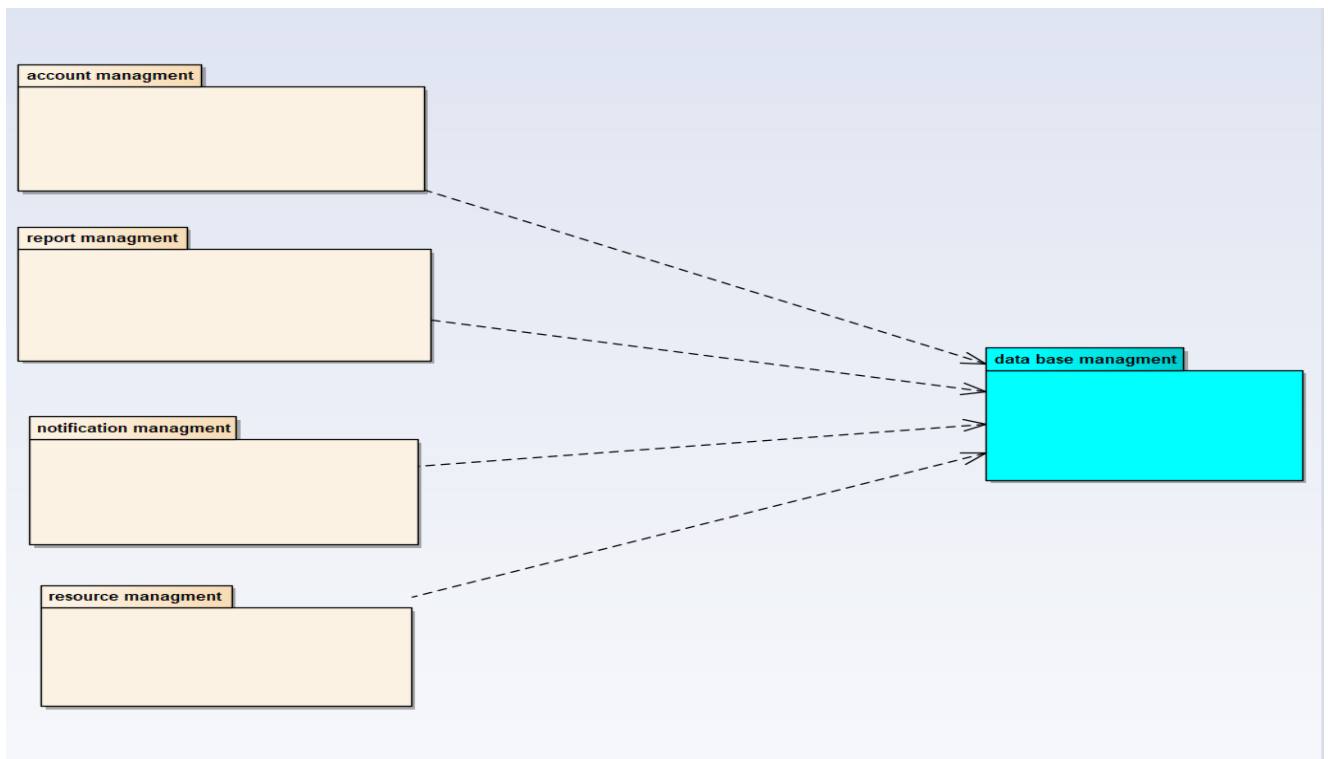


Figure5. 6 Package Diagram

## 5.5. User Interface Design

Home About Us Help Admin Login

### Login Form

**Login**

Username

Password

Login

[Forgot Password](#)

Figure5. 7 Login Page

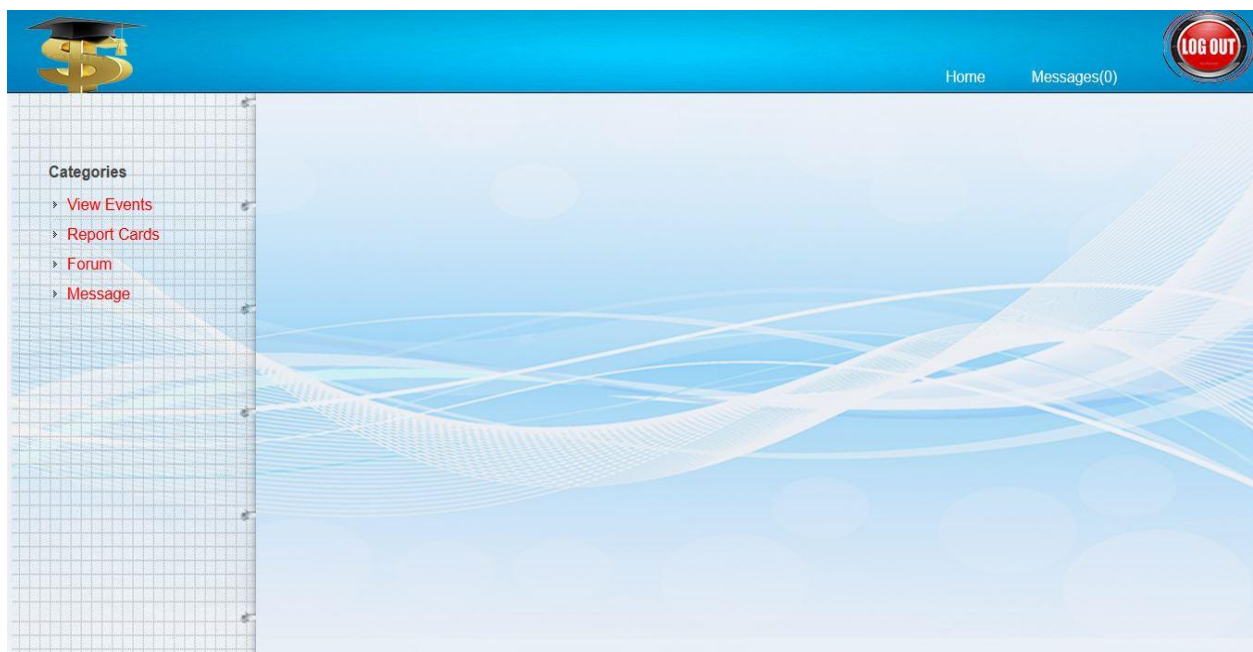


Figure5. 8 Parent Home Page



Figure5. 9 Student Home Page

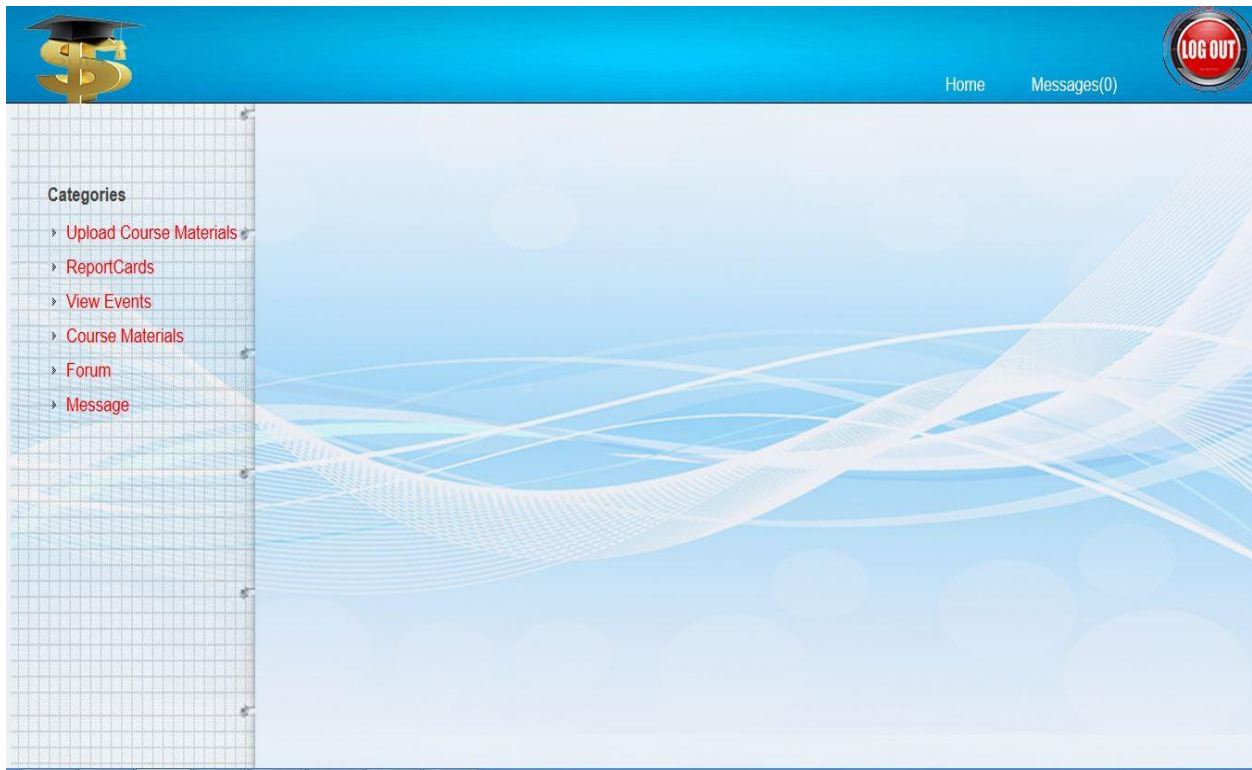


Figure5. 10 Teacher Home Page

# CHAPTER SIX

## IMPLEMENTATION AND TESTING

### 6.1. Introduction

During this phase physical design specification is turned into working computer code, and provides help for current and future users and takes care of the system. And then the code is tested until most of the errors have been detected and corrected. The purpose of this activity is to convert the final physical system specification into working model with reliable software and hardware.

### 6.2. Implementation of the Database

For implementation of system database, we choose my structured query language database (MySQL). We choose this database because of its familiarity with us and also easy for implement.

And we have created a data base which is called 111\_multi\_purpose and it have 15 different tables.

Table	Action	Rows	Type	Collation	Size	Overhead
chat		170	MyISAM	latin1_swedish_ci	21.2 KiB	-
class		2	InnoDB	latin1_swedish_ci	16.0 KiB	-
class_assignment		38	InnoDB	latin1_swedish_ci	16.0 KiB	-
class_assignment_answer		36	InnoDB	latin1_swedish_ci	16.0 KiB	-
class_exam_scedule		19	InnoDB	latin1_swedish_ci	16.0 KiB	-
class_scedule		22	InnoDB	latin1_swedish_ci	16.0 KiB	-
class_subject		32	InnoDB	latin1_swedish_ci	16.0 KiB	-
discussion_forum_collection		1	InnoDB	latin1_swedish_ci	16.0 KiB	-
discussion_forum_topic_collection		0	InnoDB	latin1_swedish_ci	16.0 KiB	-
mark_rank		18	InnoDB	latin1_swedish_ci	16.0 KiB	-
members		22	InnoDB	latin1_swedish_ci	16.0 KiB	-
post_photo_data		129	MyISAM	latin1_swedish_ci	30.1 KiB	1.6 KiB
post_photo_data_coment		51	MyISAM	latin1_swedish_ci	7.7 KiB	-
ragistratation_status		2	InnoDB	latin1_swedish_ci	16.0 KiB	-
stuent_rank		11	InnoDB	latin1_swedish_ci	16.0 KiB	-
<b>15 tables</b>	<b>Sum</b>	<b>553</b>	<b>InnoDB</b>	<b>utf8mb4_general_ci</b>	<b>251.0 KiB</b>	<b>1.6 KiB</b>

### 6.3. Implementation of the Class Diagram

✓ *Implementation of class administrator*

```
public class Administrator {  
  
    private varchar Admin ID;  
    private varchar password;  
    private varchar username;  
    private Forum Topic m_Forum Topic;  
    private Announcement m_Announcement;  
    private Report card m_Report card;  
    private Event m_Event;  
    private Message m_Message ;  
  
    public Administrator(){  
  
    }  
  
    public void finalize() throws Throwable {  
  
    }  
    public manage user(){  
  
    }  
  
    public upload information(){  
  
    }  
}//end Administrator
```

✓ *Implementation of class teacher*

```
public class Teacher {  
  
    private varchar Password;  
    private varchar Teacher ID;  
    private varchar Username;  
    private Course material m_Course material;  
    private Message m_Message ;  
    private Forum Topic m_Forum Topic;  
  
    public Teacher(){  
  
    }  
  
    public void finalize() throws Throwable {  
  
    }  
    public Send/view message(){  
  
    }  
  
    public Upload course material(){  
  
    }  
}//end Teacher
```

✓ *Implementation of class student*

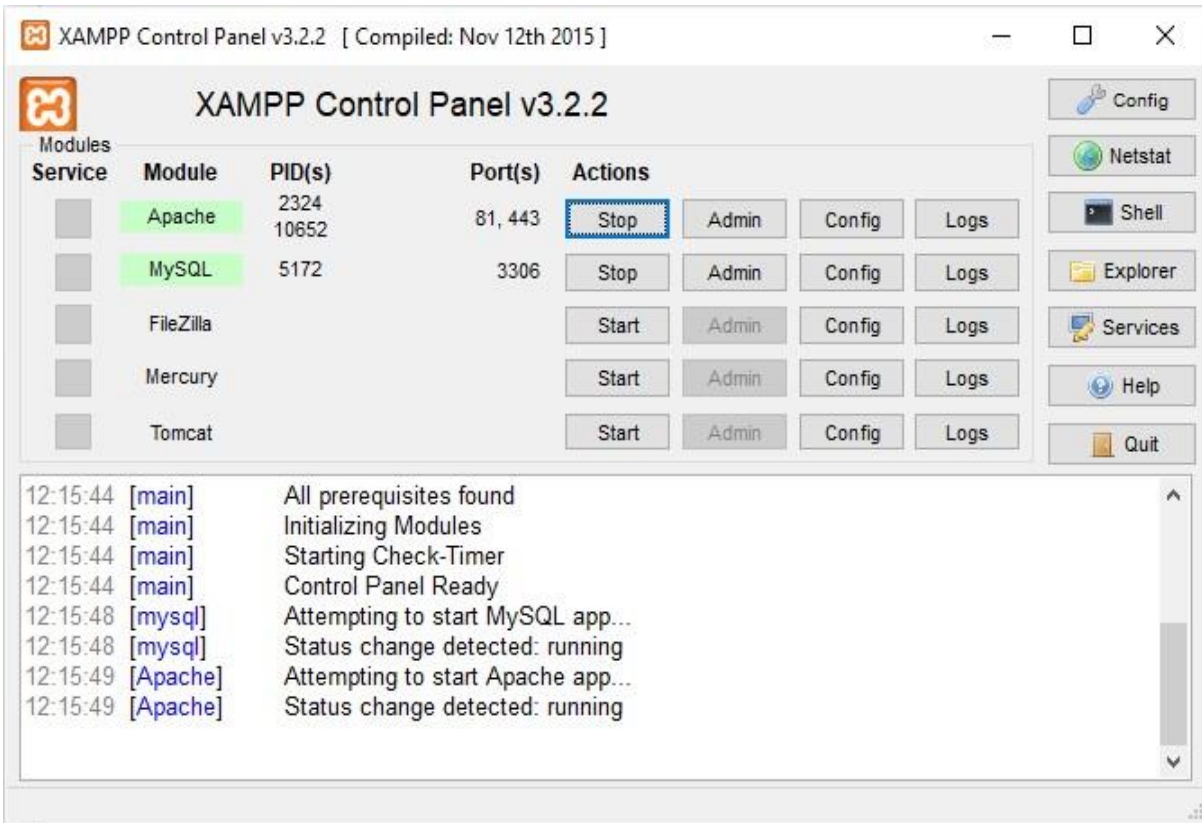
```
public class Student {  
  
    private varchar Password;  
    private varchar Student ID;  
    private varchar Username;  
    private Message m_Message ;  
    private Report card m_Report card;  
    private Course material m_Course material;  
    private Submit assignment m_Submit assignment;  
    private Event m_Event;  
  
    public Student(){  
  
    }  
  
    public void finalize() throws Throwable {  
  
    }  
    public Download course material(){  
  
    }  
  
    public Submit assignment(){  
  
    }  
} //end Student
```

✓ *Implementation of class parent*

```
public class Parent {  
  
    private varchar Parent ID;  
    private varchar Password;  
    protected varchar Student ID;  
    private varchar Username;  
    private Report card m_Report card;  
    private Event m_Event;  
    private Message m_Message ;  
    private Forum Topic m_Forum Topic;  
  
    public Parent(){  
  
    }  
  
    public void finalize() throws Throwable {  
  
    }  
    public Create Forum(){  
  
    }  
  
    public View Report card(){  
  
    }  
} //end Parent
```

## 6.4. Configuration of the Application Server

For our project we use XAMPP application server which is used for implement the server locally within the same machine. And it is very easy also widely used in order to implement server in local machine. So we download this application and install to our machine (computer) then configured it with 81 port number.



Figure\_\_\_ XAMPP application server

Then we configure it with our code-igniter PHP framework in order to use it as base URL of the system.

```
$config ['base URL'] ='http://localhost/multip/index';
```

## 6.5. Configuration of Application Security

Form validation of our system

```
<?php
include("conf.php");

if (isset($_POST['Register'])){
    $firstname=$_POST['firstname'];
    $lastname=$_POST['lastname'];

    if( preg_match('[a-zA-Z].*[0-9]|[0-9].*[a-zA-Z]', $firstname) )
    {
        die("First Name Must Not Be Number!");
    }

    if( preg_match('[a-zA-Z].*[0-9]|[0-9].*[a-zA-Z]', $lastname) )
    {
        die("First Name Must Not Be Number!");
    }

    if (preg_match('/[0-9]/', $firstname))
    {
        die("First Name Must Not Be Number!");
    }

    if (preg_match('/[0-9]/', $lastname))
    {
        die("Last Name Must Not Be Number!");
    }

    $username=$_POST['username'];
    if(strlen($username) <=7 ) {
        echo '<div class="alert alert-dismissible alert-e">';
        die( '<strong>'. "Username Must Be Grater Than Or Equal To 8 Characters !" . '</strong>' );
        echo '</div>';
    }

    $passw=$_POST['password'];
    if(strlen($passw) <=7 ) {
        echo '<div class="alert alert-dismissible alert-e">';
        die( '<strong>'. "Password Must Be Grater Than Or Equal To 8 Characters !" . '</strong>' );
        echo '</div>';
    }

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

✓ Implemented encryption

The other thing is we implement an MD5 password encryption to save the user password in unreadable format in the database. MD5 is a one way encryption algorithm which is hash function family and it is widely used in web application to save passwords in cypher text way to the database.

```
'username'=>$_POST['username'],
'password'=>md5($_POST['password']),
```

MD5 Implementation

By calling a function which is called md5 ( ) and pass the input from the field as a parameter, the input will be encrypted to cypher text and saves to database.

username	password
jerry123	e10adc3949ba59abbe56e057f20f883e
ali123	e10adc3949ba59abbe56e057f20f883e
sis123	e10adc3949ba59abbe56e057f20f883e

#### ✓ Roles in the system

In our system the system will support five different types of roles. And each roles have their own privileged pages and the access is determined by their roles.

User can login to the system as: -

- 1, Administrator
- 2, teacher
- 3, student
- 4, parent


```
<select name="member_type" id="k1" required>
  <option></option>
  <option value="teacher">Teacher</option>
  <option value="student">Student</option>
  <option value="parent">Parent</option>
  <option value="admin">Administrator</option>
</select>
```

## 6.6. Implementation of User Interface

For the actors that we are defined previously we create a user interface with different access privilege

## Login page

# Login Page



Username

Password

Login Us

[Forget Password?](#)

Developed By Wolkite University Software Engineering Graduate Students

## Admin homepage

Multipurpose School Website

[Home](#) [Message](#) [Account Management \\*](#) [School Activity \\*](#) [Resource Management \\*](#) [Logout](#)

### Welcome To School Management System Administrator Page

Successfully Deleted!

School Management System First Wall

Write :

Developed By Wolkite University Software Engineering Graduate Students

### Administrator

- Teachers
- Students
- Events
- Discussion Forum
- Class Information
- Teachers Information
- Vacancy
- News About School
- Announcement
- Students Class Management

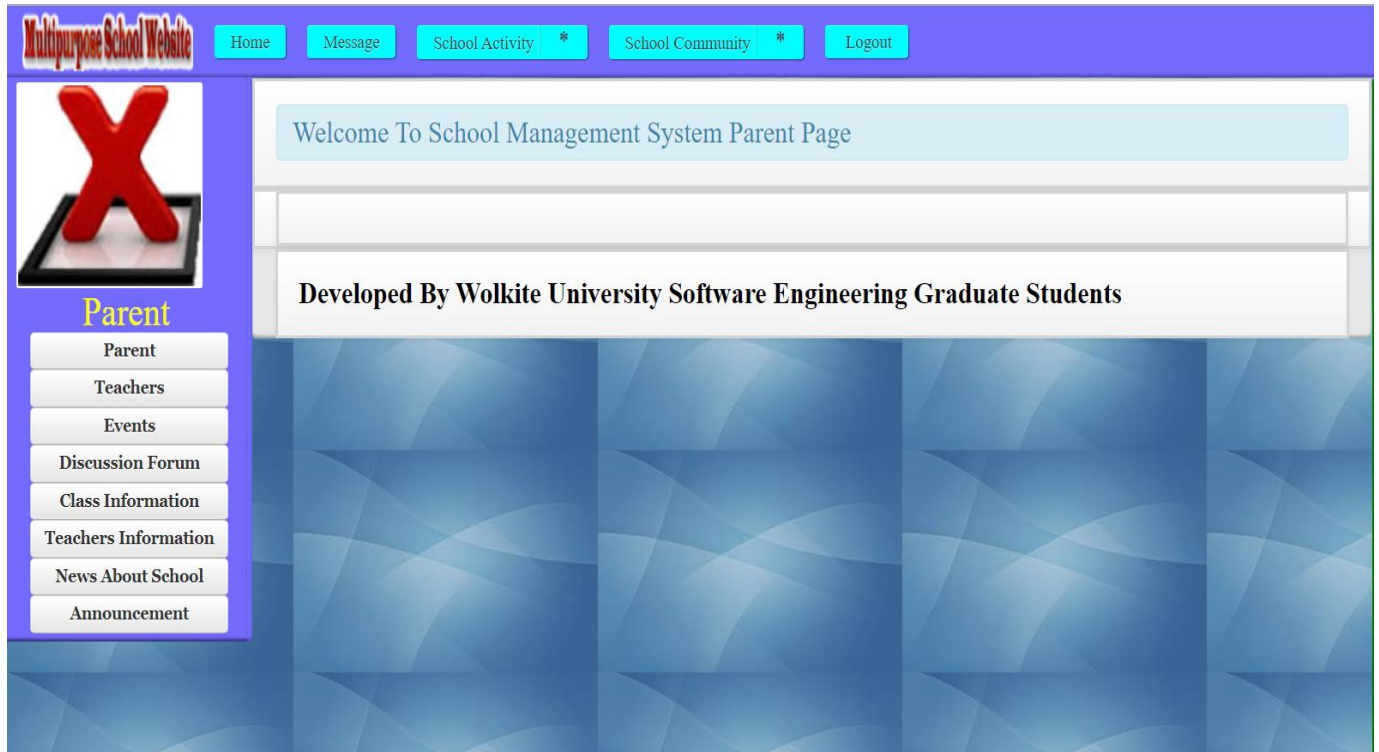
## Teacher page

The screenshot shows the 'Teacher page' of a 'Multipurpose School Website'. The top navigation bar includes links for Home, Message, Class, School Community, School Activity, and Logout. On the left, a sidebar features a green checkmark icon and a 'Teacher' section with buttons for Parent, All Students, Events, Discussion Forum, Class Information, Teachers Information, Vacancy, News About School, and Announcement. The main content area has a 'Profile Information' tab and a green message box stating 'Profile Picture Informatio Successfully Updated!'. Below this is a 'Change Profile Picture' section with a 'Choose File' button (showing 'No file chosen') and a 'Change-prof' button. At the bottom, a footer reads 'Developed By Wolkite University Software Engineering Graduate Students'.

## Student page

The screenshot shows the 'Student page' of the same 'Multipurpose School Website'. The top navigation bar is identical to the teacher page. The left sidebar features a student profile picture and a 'Student' section with buttons for Parent, All Students, Events, Discussion Forum, Class Information, Teachers Information, News About School, and Announcement. The main content area has a light blue header with the text 'Welcome To School Management System Students Page'. Below this is a 'School Management System First Wall' section with a 'Write : ' text area containing the placeholder 'Whats on Your Mind...' and a red 'Post' button. At the bottom, a footer reads 'Developed By Wolkite University Software Engineering Graduate Students'.

## Parent page



## 6.7. Testing

**System testing:** - System testing is a testing process in which we aim to ensure that our overall system works as defined by our requirements. In this testing, the team performs over all functional testing by checking whether it meets the required target.

### Authentication of login

Test = Authentication of login attempt		
Assumptions = Redirects to home.php		
Test Data ✓ User name (valid user name, invalid user name, empty) ✓ Password (invalid password, valid password, empty)		
Steps to be Executed	Data	Result
Empty User name and Password filled and Click login	Empty data	“Please fill out the field.” “Please fill out the field.”

button		
Enter invalid User name and Password	user name = ibsaamohammed password = ibsaa123 Doesn't exist in the user table	<b>“Incorrect password /user name combination”</b>
Trying to access page by directly writing the path	<a href="http://localhost/multip/home.php?MPSW_Event=">http://localhost/multip/home.php?MPSW_Event=</a>	<b>“Warning! Please login first to view this page”</b>
Enter valid User name and Password and Click Login button	User name=ibsaamohammed Password =ibsaa123	Redirects to home.php

### validation of user input

Test = validation of user input		
Assumptions = Register Users and display Success message		
<b>Test Data</b>		
<ul style="list-style-type: none"> <li>• Fname (valid Fname, empty)      <input type="checkbox"/>      Password (valid password , empty)</li> <li>• Lname (valid lname, empty)      <input type="checkbox"/>      Gender (valid gender, empty)</li> <li>• User name (valid username, empty)</li> </ul>		
<b>Steps to be Executed</b>	<b>Data</b>	<b>Result</b>
Empty Name and all others filled and click register button	Empty	All fields say <b>“Please fill out the field.”</b>

Enter invalid Name value and click register button	name=1963445 and valid data for the other fields	“Name must not be a number.”
All fields with valid data and click register button	All fields with valid data	Displays message “Successfully added as customer”

When the form is not valid the system will show this type of error message to the user in order to check his/her input to the form.

## **CHAPTER SEVEN**

### **Conclusion and Recommendation**

#### **7.1. Conclusion**

It is known that developing a system for an organization is not easy. But the team have tried its best and developed interesting web-based clinic management system for Wolkite University. This system is a web-based application to serve the whole students who wants to get treatments from the university's clinic and also the employees that are working in the clinic.

The system is flexible, accurate and attractive with easy GUI approach and the team confidently can say that the system accomplishes its objective and completed successfully. Generally, passing through the development of this system gives us a lot of knowledge in software developing life cycle not only theoretically but also practically it shows us the challenge in the real world to develop software and benefits from it. It also shows us the need for the software in the society to over control of data management.

Finally, the team expects the system will change the general information management of the clinic and makes it more reliable and effective than the previous manual system.

#### **7.2. Recommendation**

Yaberus preparatory is the earliest school that found in Wolkite town. And in our assumption this project can't handle the work load if it installed for the school. So as a recommendation the system needs a great enhancement to fit the school needs. So the upcoming students can enhance the system by adding functionality's like payment and by installing the system in the town and in the school can have communication like send message, managing user and employee's information and so on.

So if a person who strongly motivated to make an enhancement on the system should try to implement with other powerful proگرامing language like python and java, this type of programming languages will make the system so powerful and can provide an easy way to add additional functionality.

## 8. Reference

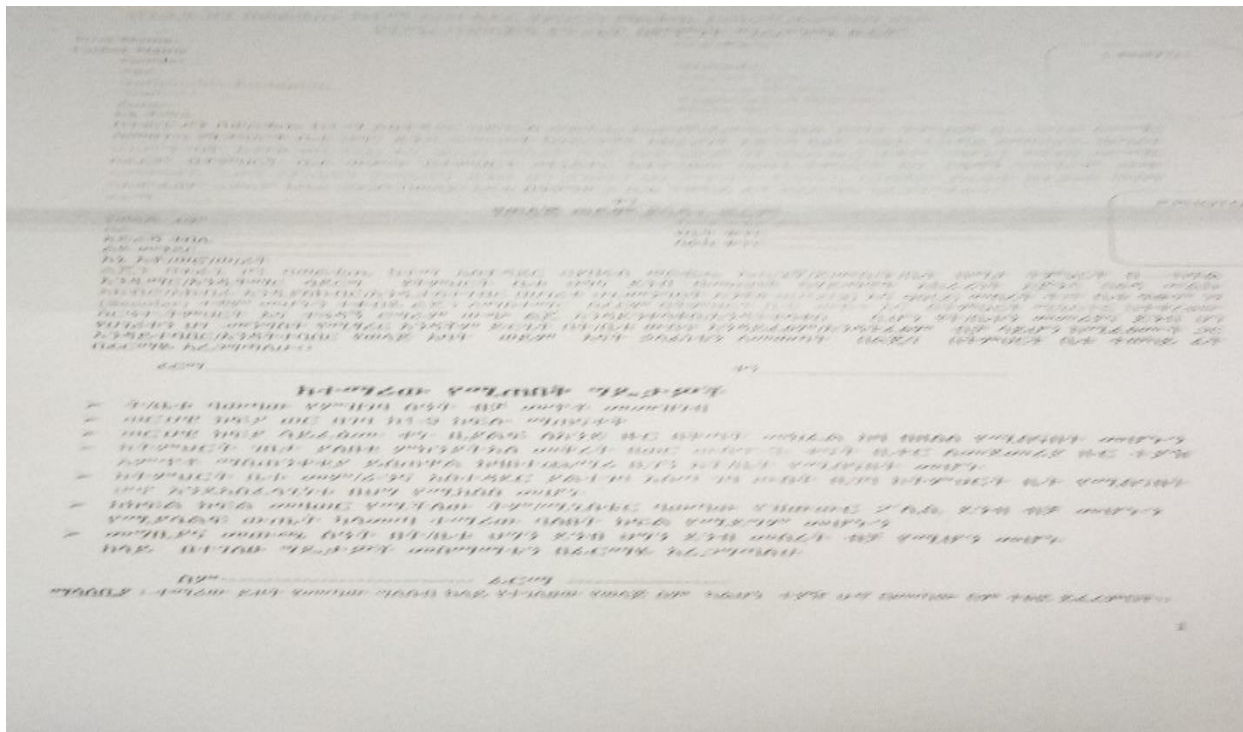
- (GradyBooch, in Object-Oriented Analysis and Design.
- Rumbaugh, J., Jacobson, I., & Booch, G. (2003). The Unified Modeling Language.
- (O'Reilly.Lane, D.&Williams,H.E, H.E., (2004)WebDatabaseApplicationswithPHP MySQL. (n.d.). Retrieved from [https://www.tutorialspoint.com/uml/uml\\_deployment\\_diagram.htm](https://www.tutorialspoint.com/uml/uml_deployment_diagram.htm)
- Duckett, J. W. (n.d.).

## 9. APPENDICES

➤ Teachers' registration form

No	Name	Age	Sex	Address	Phone	Signature	Date
1	...	...	...	...	...	...	...
2	...	...	...	...	...	...	...
3	...	...	...	...	...	...	...
4	...	...	...	...	...	...	...
5	...	...	...	...	...	...	...
6	...	...	...	...	...	...	...
7	...	...	...	...	...	...	...
8	...	...	...	...	...	...	...
9	...	...	...	...	...	...	...
10	...	...	...	...	...	...	...
11	...	...	...	...	...	...	...
12	...	...	...	...	...	...	...
13	...	...	...	...	...	...	...
14	...	...	...	...	...	...	...
15	...	...	...	...	...	...	...
16	...	...	...	...	...	...	...
17	...	...	...	...	...	...	...
18	...	...	...	...	...	...	...
19	...	...	...	...	...	...	...
20	...	...	...	...	...	...	...
21	...	...	...	...	...	...	...
22	...	...	...	...	...	...	...
23	...	...	...	...	...	...	...
24	...	...	...	...	...	...	...
25	...	...	...	...	...	...	...
26	...	...	...	...	...	...	...
27	...	...	...	...	...	...	...
28	...	...	...	...	...	...	...
29	...	...	...	...	...	...	...
30	...	...	...	...	...	...	...
31	...	...	...	...	...	...	...
32	...	...	...	...	...	...	...
33	...	...	...	...	...	...	...
34	...	...	...	...	...	...	...
35	...	...	...	...	...	...	...
36	...	...	...	...	...	...	...
37	...	...	...	...	...	...	...

➤ Students with Family registration form



➤ Report card/Roster form

**YABERUS HIGH SCHOOL**  
**Roster Report 2013 Semester: One**  
**Grade 10 Section: 100**

S.No.	Student Name	Gr.	Math	Eng	Science	SS	Art	PE	Music	Relg	Health	Elect	Lab	Total	Avg	Rank
1	ABDUL SAMIR ELKHOUBI	M	89	48	44.5	68	81	51	63	48	51	68	78	67.50	52.50	13
2	ABDUL WAFIQ JENAL KEDER	M	97	93	28	78	20	21	97	90	51	78	100	1400.00	100.00	2
3	ADAM ABDEL	M	81											65.00		38
4	ADAM ABDEL	M	89	82	24	81	37	63	53	80	24	66	83	68.40	80.40	38
5	ADAM ABDEL	M	82	48	0	19	31	52	66	46	68	66	51	55.00	24.75	110
6	ADAM ABDEL	M	79	87	87	97	80	28	69	68	69	69	69	69.00	69.00	69
7	ADAM ABDEL	M	83	78	28	28.5	68	81	61	65	20	78	68	65.00	50.50	29
8	ADAM ABDEL	M	80	81	38.8	28	37	69	66	68	78	74	66	68.00		
9	ADAM ABDEL	M	82	79		24								65.00		38
10	ADAM ABDEL	M	77	72	13	63	49	69	68	77	64	67	67	67.00	38.75	110
11	ADAM ABDEL	M	80	85	83	69	69	83	72	64	83	80	72	69.00	53.10	54
12	ADAM ABDEL	M	76	80	36	85	33	33	64	59	73	80	78	60.00	44.10	54
13	ADAM ABDEL	M	89	87	87	89	38	40	49	66	69	68	68	68.00	68.00	68
14	ADAM ABDEL	M	75	50	13.5	29	50	45	84	88	67	55	62	58.00	60.00	62
15	ADAM ABDEL	M	89	85	48	48	68	68	68	68	68	68	68	68.00	68.00	68
16	ADAM ABDEL	M	80	36	49	36	49	40	23	47	48	58	51	48.00	33.00	110
17	ADAM ABDEL	M	45	33	33	0	0	0	0	0	0	0	0	0.00	0.00	110
18	ADAM ABDEL	M	76	40	30	51	37	40	22	46	21	59	51	48.00	40.00	110
19	ADAM ABDEL	M	89	89	89	89	89	89	89	89	89	89	89	89.00	89.00	89
20	ADAM ABDEL	M	72	77	48.5	60	50	58	53	53	53	53	53	53.00	53.00	53
21	ADAM ABDEL	M	79	79	33.5	65	75	52	60	60	60	60	60	60.00	60.00	60
22	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
23	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
24	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
25	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
26	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
27	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
28	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
29	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
30	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
31	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
32	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
33	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
34	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
35	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
36	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
37	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
38	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
39	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
40	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
41	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
42	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
43	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
44	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
45	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84
46	ADAM ABDEL	M	84	84	84	84	84	84	84	84	84	84	84	84.00	84.00	84