# Declaration

This project is our own and has not been presented for a degree in any other university and all the sources of materials used for the project have been duly acknowledged. (Name and Signature up to the number of the project group members)

**Name Signature**

1. Abderhman Jissa -------------------
2. Asemie Alebachew -------------------
3. Ayenesh Abebe -------------------
4. Betelhem Mohabaw -------------------
5. Tewabe Mihrete -------------------

I certify that this project satisfies all the requirements as a project for degree of Bachelor of Science.

------------------------------------------ ------------------------ -----------

**Name of program coordinator Signature Date**

This is to certify that I have read this project and that in my opinion it is fully adequate, in scope and quality, as a project for the degree of Bachelor of Science

----------------------------------------- -------------------------------

**Name of Advisor Signature**

**Examining committee members Signature Date**

1. Examiner 1\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_
2. Examiner 2\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_
3. Examiner 3\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_

It is approved that this project has been written in compliance with the formatting rules laid down by the school of the university.

# Abstract

This project aims to develop online grading system for Registrar office of Atse-Tewodros campus. The online grading system is web-based system that can be accessed through out the internet and can be accessed by authorized users. The system will automate the grading system of the registrar office. The system also includes GPA and CGPA computation of students. The existing system uses MS excel and maintains their records, however it’s difficult to share data from multiple systems in multiuser environment. There is a lot of duplicate work and chance of mistake where the records are changed they need to update each and every excel file. There is no security; anybody can access any report and sensitive data. This online grading system is used to overcame the entire problem which they are facing currently and making complete atomization of manual system to computerized system

# Acknowledgment

In performing our proposal, we had to take the help and guideline of some respected persons, who deserve our greatest gratitude. The completion of this paper gives us much pleasure. We would like to show our gratitude **Mr. Yebekal** for giving us a good guideline for proposal throughout numerous consultations. We would also like to expand our deepest gratitude to all those who have directly and indirectly guided us in writing this paper.

Many people, especially our classmate and team members itself, have made a valuable comment suggestions on this proposal which gave us an inspiration to improve our proposal. We thank all people for help directly and indirectly to complete this paper.

# Tables of contents

[Declaration i](#_Toc453568296)

[Abstract ii](#_Toc453568297)

[Acknowledgment iii](#_Toc453568298)

[Tables of contents iv](#_Toc453568299)

[List of Tables vii](#_Toc453568300)

[List of Figures viii](#_Toc453568301)

[Acronyms ix](#_Toc453568302)

[CHAPTER ONE - 1 -](#_Toc453568303)

[1.1 Introduction - 1 -](#_Toc453568304)

[1.2 Background of the study - 2 -](#_Toc453568305)

[1.3 Statement of the Problem - 2 -](#_Toc453568306)

[1.4 Objectives of the project - 3 -](#_Toc453568307)

[1.4.1 General objectives - 3 -](#_Toc453568308)

[1.4.2 Specific objective - 3 -](#_Toc453568309)

[1.5 Scope of the project - 3 -](#_Toc453568310)

[1.6 Constraints - 4 -](#_Toc453568311)

[1.7 Alternative solution - 4 -](#_Toc453568312)

[1.8 Feasibility study - 5 -](#_Toc453568313)

[1.8.1 Technical feasibility - 5 -](#_Toc453568314)

[1.8.2 Economic feasibility - 5 -](#_Toc453568315)

[1.8.3 Operational feasibility - 5 -](#_Toc453568316)

[1.9 Proposed Solution - 6 -](#_Toc453568317)

[1.10 Significance of the projects - 6 -](#_Toc453568318)

[1.11 Beneficiary of the project - 6 -](#_Toc453568319)

[1.12 Methodology - 7 -](#_Toc453568320)

[1.12.1 Method of data collection - 7 -](#_Toc453568321)

[1.12.1.1 Observation - 7 -](#_Toc453568322)

[1.12.1.2 Interview - 7 -](#_Toc453568323)

[1.12.1.3 Reading stored document related to system - 7 -](#_Toc453568324)

[1.12.2 System Development Approach - 7 -](#_Toc453568325)

[1.12.3 Development Tools - 7 -](#_Toc453568326)

[1.12.3.1 Hardware tools - 8 -](#_Toc453568327)

[1.12.3.2 Software tools - 8 -](#_Toc453568328)

[1.13 Specification of hardware and software - 8 -](#_Toc453568329)

[1.13.1 Hardware specification - 8 -](#_Toc453568330)

[1.13.2 Software specification - 9 -](#_Toc453568331)

[1.14 Budgets of the project - 9 -](#_Toc453568332)

[1.14.1 Tangible - 9 -](#_Toc453568333)

[1.14.2 Intangible - 10 -](#_Toc453568334)

[1.15 Overall Cost Estimations - 10 -](#_Toc453568335)

[1.16 Schedule of the project - 10 -](#_Toc453568336)

[CHAPTER TWO - 12 -](#_Toc453568337)

[SYSTEM ANALYSIS - 12 -](#_Toc453568338)

[2.1 Detailed study of the current system - 12 -](#_Toc453568339)

[2.2 The inputs and outputs of the new system /the proposed system - 12 -](#_Toc453568340)

[2.2.1 Inputs of the new system: - 12 -](#_Toc453568341)

[2.2.2 Outputs of the new system: - 13 -](#_Toc453568342)

[2.3 Overview of the new system - 13 -](#_Toc453568343)

[2.4 Requirement Analysis of the proposed system - 14 -](#_Toc453568344)

[2.4.1 FUNCTIONAL REQUIREMENT - 14 -](#_Toc453568345)

[2.4.2NON- FUNCTIONAL REQUIREMENT - 15 -](#_Toc453568346)

[2.5 Identifying actors and use case - 16 -](#_Toc453568347)

[2.6 Use case diagram for the new system - 17 -](#_Toc453568348)

[2.6.1 Use case description: - 18 -](#_Toc453568349)

[2.7 Sequence diagram - 22 -](#_Toc453568350)

[2.7.1Sequence diagram for login - 22 -](#_Toc453568351)

[2.7.2 Sequence diagram for register - 23 -](#_Toc453568352)

[2.7.3 Sequence diagram for add course - 24 -](#_Toc453568353)

[2.8 Activity diagrams - 25 -](#_Toc453568354)

[2.8.1 Activity diagram for login - 26 -](#_Toc453568355)

[2.8.2 Activity diagram for logout - 27 -](#_Toc453568356)

[2.8.3 Activity diagram for register - 28 -](#_Toc453568357)

[2.8.4 Activity diagram for view grade - 29 -](#_Toc453568358)

[2.8.5 Activity diagram for update password - 30 -](#_Toc453568359)

[2.8.6 Activity diagram for submit grade - 31 -](#_Toc453568360)

[2.9 Class Diagram - 32 -](#_Toc453568361)

[Chapter Three - 34 -](#_Toc453568362)

[3.1 system design - 34 -](#_Toc453568363)

[3.2 Collaboration Diagram - 35 -](#_Toc453568364)

[3.2.1 Collaboration diagram for login - 35 -](#_Toc453568365)

[3.2.2 Collaboration diagram for logout - 36 -](#_Toc453568366)

[3.3 - 36 -](#_Toc453568367)

[3.4 Student table - 37 -](#_Toc453568368)

[3.5 Teacher table - 37 -](#_Toc453568369)

[3.6 Admin table - 37 -](#_Toc453568370)

[3.7 Deployment diagrams - 38 -](#_Toc453568371)

[3.8 deployment diagrm - 39 -](#_Toc453568372)

[Chapter 4 - 40 -](#_Toc453568373)

[System Implementation and Testing - 40 -](#_Toc453568374)

[4.1 Introduction - 40 -](#_Toc453568375)

[4.2 Interface Design - 40 -](#_Toc453568376)

[4.3 Login page - 42 -](#_Toc453568377)

[4.3.1 Sample code for home page - 43 -](#_Toc453568378)

[4.3.2 Sample code for login - 47 -](#_Toc453568379)

[4.4 Testing - 63 -](#_Toc453568380)

[4.4.1 Unit testing - 63 -](#_Toc453568381)

[4.4.2 Integrated testing - 63 -](#_Toc453568382)

[4.4.3 System testing - 63 -](#_Toc453568383)

[4.5 Functionality of the system - 63 -](#_Toc453568384)

[4.6 Error handling - 64 -](#_Toc453568385)

[Chapter 5 - 66 -](#_Toc453568386)

[Conclusion and Recommendation - 66 -](#_Toc453568387)

[5.1 Conclusion - 66 -](#_Toc453568388)

[5.2 Recommendations - 67 -](#_Toc453568389)

# List of Tables

Table 1 Hardware Specification---------------------------------------------------------------8

Table 2 Cost of implementation tangible way-----------------------------------------------9

Table 3Overall cost Estimation---------------------------------------------------------------10

Table 4 Project Schedule-----------------------------------------------------------------------11

Table 5 Identifying actors and use case------------------------------------------------------16

Table 6 Use case description for login use case---------------------------------------------18

Table 7 Use case description for add mark use case----------------------------------------19

Table 8 Use case description for view grade use case--------------------------------------20

Table 9 Use case description for submit grade use case------------------------------------21

Table 10 student table --------------------------------------------------------------------------37

Table 11 teacher table --------------------------------------------------------------------------37

Table 12 admin table ---------------------------------------------------------------------------37

# List of Figures

Figure 1 Use case diagram for the new system-------------------------------------------------------17

Figure 2 Sequence diagram for login -----------------------------------------------------------------22

Figure 3 Sequence diagram for register ---------------------------------------------------------------23

Figure 4 Sequence diagram for add course -----------------------------------------------------------24

Figure 5 Activity diagram for login use case---------------------------------------------------------26

Figure 6 Activity diagram for logout------------------------------------------------------------------27

Figure 7 Activity diagram for rigsterar ---------------------------------------------------------------28

Figure 8 Activity diagram for view grade ------------------------------------------------------------29

Figure 9 Activity diagram for update password ----------------------------------------------------30

Figure 10 Activity diagram for submit grade---------------------------------------------------------31

Figure 11 Class diagram --------------------------------------------------------------------------------33

Figure 12 Collaburation diagram for login-----------------------------------------------------------35

Figure 13Collaburation diagram for logout----------------------------------------------------------36

Figure 14 Deployment diagram------------------------------------------------------------------------39

Figure 15 User intterface diagram for home page---------------------------------------------------41

Figure 16 User interface diagram for login page-----------------------------------------------------42

Figure 12 Error handling diagram ---------------------------------------------------------------------65

# Acronyms

MYSQL: My Structure Query Language

HTML: Hyper Text Markup Language

UOG: University of Gondar

PHP: Hyper Text Preprocessor

CD-R: Compact Disk-Readable

RAM: Random Access Memory

GPA: Grade Point Average

UML: Unified Modeling Language

IT: Information Technology

CGPA: Cumulative Grade Point Average

PC: Personal Computer

CSS: Cascading Style Sheet

OS: Operating System

MS: Microsoft

Admin: Administrator

HD: Hard Disk

MB: Megabyte

GB: Gigabyte

GHZ: Gigahertz

# CHAPTER ONE

# 1.1 Introduction

Some registrar right now, especially those with huge numbers of students who are not yet using online-based grading system has difficulty/problems concerning the records of grades of the students. The grades are the important record to keep even for the longest time for the referral and credentials of the student to enter their next level of attaining their goals. Manual computation is very prompt to risk for any circumstances. It is time consuming in terms of recording grades, computation using of calculator. If some records are lost, they never retrieve it in case of unexpected calamity. Accuracy and security is not been so defined. Grading System is the most commonly used in computing and analyzing the performance, talent and skills of students. The online Grading System will help the registrar, teacher and the students to have easy access on the records and past records, the easier way. The staff in charge will do updating of grades. The online Grading System will also be a convenience for the registrar as well, because by this, delayed passing of grade will be prevented and class cards, certificate of grades, and transcript of records will be processed in a speedy way. There would be no unauthorized alternation of grades once the records are inputted and finalized, so the records are secure. Develop knowledge of the Internet and computers skills that will help students throughout their lives and careers. Generally, in the registrar office of UOG Online Grading System will be used for submission of all final course grades. The system also provides instructors with a flexible and user-friendly interface for posting final course grades to student records. Online Grading System is a web-based application that can be used to create account for students, used to post the grades of the students for students online.

# 1.2 Background of the study

*The registrar* office has upgraded its capabilities and research facilities. Registrar office of Atse-Tewodros is a comprehensive registrar engaged in the provision of all-round education, research, training, & community service through its diversified areas of academic units’ registrar office of Atse-Tewodros campus is actively engaged in expansion activities to increase its intake capacity.

Registrar office is the major part of the campus, computing and showing grades report for students is one of the major tasks in the registrar office of the campus because the registrar office has to a responsibility for helping student such as solving grade and course related problems during their stay in the campus. To carry out their tasks they set a rules and regulation.

The registrar office highly recommended that students should know these rule and regulation carefully and familiarize themselves with them. Atse-Tewodros campus has registrar office that submits the grade report of the students after printing it. Also the registrar assigns the courses for students which they take and for department they give the course. Registrar office of Atse-Tewodros campus also needs to make the system to be online and provide solution for some of the problem that was happening

# 1.3 Statement of the Problem

At the current technology, Registrar office of Atse-Tewedros campus has no online grading system. They are trying to cope the existing system, which is normally release the students grade list and grade report in a delayed time. Because they compute it using Microsoft Excel electronic spreadsheet, This causes the students to waste time and effort in always going back to department and registrar just to view their grades and to get their grade report. The registrar office has no online grading system that is available for all courses.

The problems encountered in grading system which are the following:

* Some records are lost.
* Incorrect computing of grades.
* As to security, easy alteration of grades happens.
* Once they received their class cards, they only see their final grade.

# 1.4 Objectives of the project

## 1.4.1 General objectives

The general objective of this project is to develop Online Student grading system for registrar office of Atse-Tewodros campus.

## 1.4.2 Specific objective

The specific objective of this project is:

* To develop web-based system which provides online grading service that will be used to reduce wastage of time and energy.
* To create a system of reliable and accurate to compute grades in all different courses.
* To provide fast access of information of grades and generate reports.
* To develops knowledge of the Internet and computers skills that will help students throughout their lives and careers
* To eliminate the problems such as wastage of time, wastage of tons of papers and others that faces the existing system and deliver the best alternative proposed solution.
* To enable students to view their grades online.
* To enable students to view their GPA and CGPA.
* To increase the use of internet and the technology of the day.

# 1.5 Scope of the project

Our project is going to implement in Registrar office of Atse-Tewedros campus and the system boundary includes regular student and it is applied to student from first year up to graduate and excluded summer and extension students.

Generally, our project scope include

The system enables students to view their grade.

Provide proposed grade submission by instructors.

Admin of the registrar update grades for students online.

Show grades for students online without going to their instructors.

* The administrator has an authority to edit the grades with the permission of the instructor.
* The administrator can filter the instructor, students and course.
* The administrator can add, edit, view and delete course, students and instructors.

# 1.6 Constraints

* Available with internet connection.
* Some students may not have a devise to access the website.
* Students may feel isolated from the instructor.

# 1.7 Alternative solution

The existing grading system in the registrar office is paper based, time consuming, less flexible. The chance of loss of records is high and also record searching is difficult.

Maintenance of the system is also very difficult and takes lot of time. Result Processing is slow due to paper work and requirement of staff. To solve these problems they required a computerized system, which is used as a core alternative solution to handle all the works. This alternative solution is the system we are going to develop that will provide a working environment that will be flexible and will provide ease of work and will reduce the time for report generation and other paper works.

The following are the alternative solution for the existing problem:

* Developing desktop application
* Developing online grading system.
* Developing mobile application

# 1.8 Feasibility study

Feasibility study is used to investigate the proposed system in multiple dimensions. It used to indicate whether the system feasible or not. The proposed system can be seen according to the following literals.

## 1.8.1 Technical feasibility

Technical feasibility is the measure of practicality of the specific technical solution and the availability of technical resources and expertise. The proposed system can be easily maintained and repaired without requiring high experts or technical assistants, because the system was developed by familiar programming language (environment).The project team members have learned programming languages that required for the successful completion of the project such as java script, CSS, HTML, PHP, MYSQL. Team members have the required skill to develop the system, So that the project can be said technically feasible.

## 1.8.2 Economic feasibility

Is a measure of cost effectiveness of a project and also used to identify the benefits and costs concerning the project. Economic feasibility is cost benefit analysis. The new system minimizes cost of using paper for manual recording. The existing system need more human effort also more budget. For example for fulfilling once work it needs more paper, pen, time and others. Our new system minimize the time, material needed also human effort that is the reason why our new system is economically feasible.

## 1.8.3 Operational feasibility

The new system that will develop is not this much complex and doesn’t need extra expertness other than simple computer operational skill and it has easy to use GUI so it is operationally feasible.

# 1.9 Proposed Solution

The new system will be website that will be accessed via the Internet easily and it will give essential ground for online grading system. The system will reduce the problem that exists in the current system that is lack of fast and efficient grading system process.

# 1.10 Significance of the projects

The significance of our system is to solve the current issues encountered and improve the quality of service of registrar office grading system. After this system developed will give the following benefit:

* Ease access of information.
* Easy update of student and instructor information.
* Reduce data processing error.
* The system provides better data management facilities.
* Greatly reduce the use of paper at the registrar office
* It creates an opportunity for the registrar to well-known registrar in the country level.
* Minimize waste of time for grading system of the registrar.
* Provide efficient and effective grading system.

# 1.11 Beneficiary of the project

**Students**: the students will get information easily. The system will provide fast access of information so that the time that has been wasted also minimized.

**Registrar Office**: the registrar can manage data in easy and efficient manner.

**Instructors:** to avoid waste of time and easily solve posting and submitting grade

# 1.12 Methodology

## 1.12.1 Method of data collection

The data collection process to conduct this project includes both the qualitative and quantitative data. The collection of information is the main core of the system analysis the information we used to describe the end to end process of the existing system ofonline grading system. This will be done through the use of instruments such as observations and the Internet. From these two data gathering tools, Observation will also be used to oversee the required things in the registrar. And the internet used to collect data from the website of the registrar

### 1.12.1.1 Observation

Observation is the other instrument that will use to collect data which will be necessary for our web-based system project for the registrar. In this process we will try to investigate the information by making our selves participates in the process. And observations will also helping us to look to the reality of the registrar.

### 1.12.1.2 Interview

Conduct an interview with the registrar staff members for each activity related with grading system.

### 1.12.1.3 Reading stored document related to system

To have detailed awareness about our project we used documents such as books, e-books and some related previously done projects which are very important to develop our project. During the analysis of documents, we give a special consideration to those documents which can bring more features to our system.

## 1.12.2 System Development Approach

This project follows or uses software development methodology which is object-oriented paradigm with greatest values.

## 1.12.3 Development Tools

Developing online grading system needs a number of tools that makes the process easy and fast. These development tools are hardware tools and software tools both collaboratively work to achieve specific goals. Hardware tools are all tools that we touch and feel and help to work with the project. Software tools are programs or instructions that help us to simplify work. Here are some development tools:-

## 1.12.3.1 Hardware tools

* Personal computer(Pc)
* Web server machine
* Pen and paper
* Flash
* Printer: - for printing documentation

## 1.12.3.2 Software tools

* Web browser (Mozilla Firefox, Google chrome, opera, IE8): So we can check the system across a variety of platforms.
* Windows operating system to develop the system
* Adobe Photoshop: for editing images and icons for the interface of the system.
* WAMP with PHP and MY-SQL, apache server
* Windows notepad: HTML, CSS, Java script

# 1.13 Specification of hardware and software

## 1.13.1 Hardware specification

**Table1**

|  |  |  |
| --- | --- | --- |
| Hardware | TYPE | USE |
| Flash disk | 8GB | For storing files  For transferring those files from one computer to the other |
| CD-R | 700 MB | For permanent storing data  Use as a backup files |
| Personal computer | Any brand with minimum 2 GB RAM and above  HD capacity minimum 250 and above with 2.0 GHZ and above processor | The whole project will be done through using computer.  Used when the lab class is closed |

*Table 1.1 Hardware specification*

## 1.13.2 Software specification

* + Ms-word:for project proposal documents.
  + Notepad++: to write HTML and PHP code.
  + WAMP server: to develop the PHP program and to connect with database
  + Web Browser: any internet browser used to access web pages
  + Edrawmax: for drawing UML diagram

# 1.14 Budgets of the project

The budget allocated by group members to complete the project is explained in two ways. This are:-

* Tangible
* Intangible

## 1.14.1 Tangible

**Table 2**

|  |  |  |  |
| --- | --- | --- | --- |
| Item | Quantity | Unit cost in birr | Total cost in birr |
| **Paper** | 3 packet | 100 | 300 |
| **Pen** | 6 | 4 | 24 |
| **Laminating** | 4 | 10 | 40 |
| **Flash** | 2 | 200 | 400 |
| **Laptop** | **1** | 12000 | 12000 |
| **Total cost** | | | 12,764 |

*Table 1.2Table for cost of implementation tangible way*

## 1.14.2 Intangible

Costs consequent from the design of an automated system that cannot be easily considered as:

* Time requires for adapting new system
* Requiting or train staffs who operates the new system.

Generally, it is possible to get positive result by subtracting the cost break down of the current system from the previous system. Getting positive result informs the economic feasibility. Therefore, the system is economically feasible.

# 1.15 Overall Cost Estimations

**Table 3**

|  |  |
| --- | --- |
| **Hardware and software** | **Estimated cost** |
| Hardware cost | 12,764 |
| Software cost | Free (trial) |
| Total cost | 12,764birr |

*Table 1.3 Cost Estimation*

# 1.16 Schedule of the project

This project is expected to be completed in two semesters of 2015/2016 academic year. Some parts of the project component to be completed in the first semester are up to analysis and the rest of activities such as implementation, testing and maintenance would complete in the next semester including other minor activities.

Generally, the schedule of the project is provided for the purpose of doing the tasks on time. So we represent the schedule of the project using Gantt chart as follows.

**Table 4**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Project phase | Nove2015 | Dec201 5 | Jan2016 | Feb2016 | Mar2016 | Apr2016 | May 2016 | Jun2016 |
| **1** | Proposal |  |  |  |  |  |  |  |  |
| **2** | Requirement analysis |  |  |  |  |  |  |  |  |
| **3** | Design |  |  |  |  |  |  |  |  |
| **4** | Implementation(coding) |  |  |  |  |  |  |  |  |
| **5** | Testing |  |  |  |  |  |  |  |  |

*Table 1.4 project schedule*

# CHAPTER TWO

# SYSTEM ANALYSIS

# 2.1 Detailed study of the current system

The function of existing system is to announce the student to grading using different methods and Medias. Therefore there is a report to be generated.

The major practices of grading process in the registrar office are the following:

* Then the instructor shows the result to the students.
* The existing system is a manual one, requiring much time with paper and ink. It is a very tedious task for teachers to calculate grades of students and write them down in a mark sheet. This method proves to be uneconomical; redundancy and errors in data (grade) entry may occur as well.
* The case is not so with the proposed online grading system. This system makes it very easy for teachers to calculate grades and publish it online in the project website. Students can then view their respective grades by visiting the site.
* Finally the student can view their grade.

# 2.2 The inputs and outputs of the new system /the proposed system

2.2.1 Inputs of the new system: the inputs to the new system is derived from student‘s submit mark, registration and grading report form.

The inputs are processed to obtain the output.

* Students mark.
* Student information such as full name and photo.
* Instructor information such as full name, contact number address and photo.
* System should have a form to accept the student details.

2.2.2 Outputs of the new system: the output design was based on the inputs

The Outputs of the new system includes the following:

* Grade
* GPA
* CGPA
* Full student information
* Full instructor information

# 2.3 Overview of the new system

The Online Student Grading System in registrar office of Atse-Tewodros campus automates each and every activity of the manual system and increases its throughput. It is created for computing, storing and showing online grade of the student.

Some of the features are the following:

* In comparison to the existing system, the response time of the system is very less and it works very fast.
* Result will be very precise and accurate and will be declared in very short span of time because grade, GPA and CGPA calculations are done by the system itself.
* The proposed system is very secure since all users enter to the system by their own user accounts.
* The logs of appeared students and their marks are stored and can be for backup for future use.
* The proposed system will reduce cost of grading system.
* It saves the instructors from time wasting while they can get the results of their students from the system immediately.
* It can generate grade reports when and where required.
* The system uses computer to handle data and perform complex tasks.
* Grading systems such as GPA and CGPA are followed by many educational institutions, colleges and universities in the world. This online grading calculator calculates the average marks the student has received based on CGPA grading system. It can be used by any college or university to calculate the grade obtained by a student in any subject.
* While many offline GPA and CGPA calculator are available to calculate grades, all the grading procedure in the proposed system is done through online. So, project is much computerized and easier compared to existing grading system.

# 2.4 Requirement Analysis of the proposed system

Once the aims of the project have been established, the work of eliciting, analyzing and validating the system requirements can commence. This is crucial to gaining a clear understanding of the problem for which the system is to provide a solution and its likely cost. The requirement phase translates the ideas in the minds of the clients into a formal document.

To help the analysis of the system requirements, conceptual models of the system are constructed. These aid understanding of the logical partitioning of the system, its context in the operational environment and the data and control communications between the logical entities. This section includes functional and non-functional requirements of the system described below.

## 2.4.1 FUNCTIONAL REQUIREMENT

Online grading system in registrar office of Atse-Tewodros campus has the following functionalities:

* **Registration:** Enables admin to register both the instructors and students to the system.
* **Viewing grades:** Enables the students to login to the system and view their grades, GPA, CGPA.
* **Uploading**: Enables the admin to upload a photo for students and instructors.
* **Posting grade:** Enables the instructors to login to the system and calculate the grade and submit grades for students.
* **Record Information:** System will record information about students.
* **Updating:** The admin will update information and users can update their password.
* **Authentication**: The system will be verified by denying unauthorized user from using the system.
* **Authorization***:* The system provides enough privilege to access and update the recourses.

**Notice:** The system will display information needed for the students on time.

## 2.4.2NON- FUNCTIONAL REQUIREMENT

Registrar office Atse-Tewodros has the following non-functional requirements to achieve its functionality.

* **Usability:-**The system is easy to operate. The User interface for this system will be simple and clear. The student information will be easy to read, understand and access
* **Availability (ubiquity)**: – This system is available in everywhere (where internet service reach) and at all time for those who have access to use the system.
* **Performance**-The system will have good performance i.e. fast response time and optimal workload.
* The web application server used should provide good performance and ability to manage performance with techniques such as support for caching.
* **Security**: – The system will be secured as much as possible so that there is permissible information flow regarding to who can do what.

So it is designed to be very secure by providing a login feature which authenticates the user by means of a username/Id and password with which user will be able to login to his/her respective pages and use the system as required.

* **Portability:**-The system is machine independent and software system independent so it can be moved to different target platforms.
* **Reliability: –** The system is operational and consistent in that integrity of information is maintained and supplied to the system.
* **Robustness**: The system shall validate data entry and prompt the user when the invalid data is entered
* **Scalability**: The system can scale to different department and different campus.

A use case is a list of actions or event steps, typically defining the interactions between a role (known in the Unified Modeling Language as an actor) and a system, to achieve a goal. The actor can be a human or other external system’s.

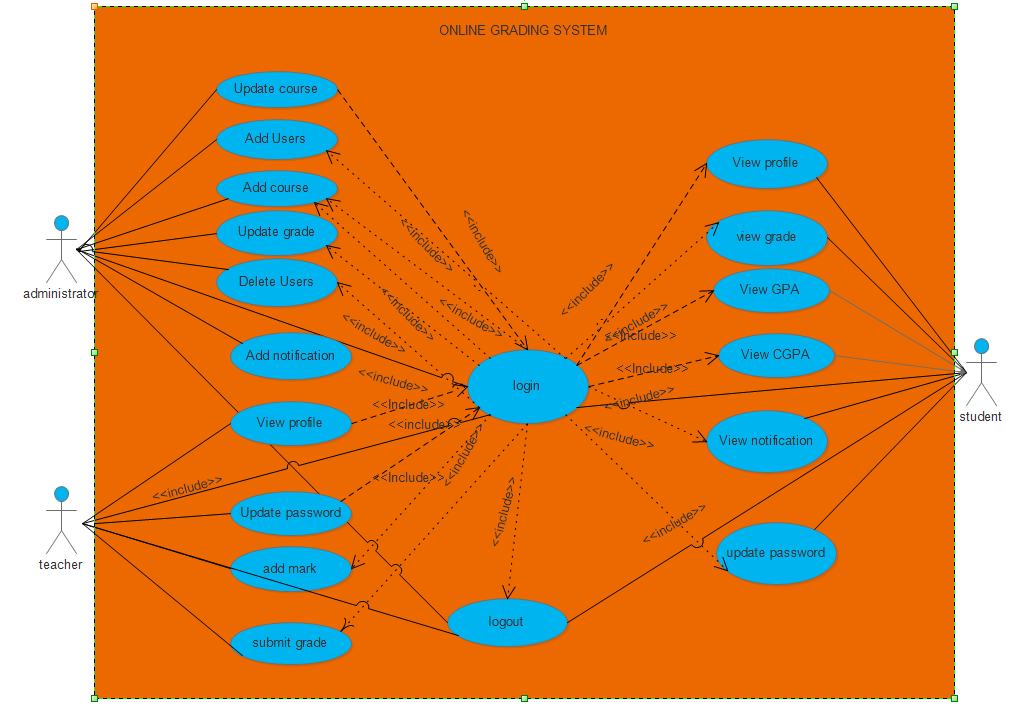
# 2.5 Identifying actors and use case

**Table 5**

|  |  |
| --- | --- |
| Actors | Use case |
| * Student * Administrator * teacher | * Register * *Login* * *Logout* * View CGPA * View GPA * Submit Grade * Modify grade * Add Notification * View notification * Add course * Update course * Add student * View profile * Update password |

*Table 5 Identifying actors and use case*

# 2.6 Use case diagram for the new system



*Figure 1 use case diagram*

## 2.6.1 Use case description:

**Table 6**Use case description for login use case

|  |  |
| --- | --- |
| **Use case name** | Login |
| **Description** | Enables all users of the system to login. |
| **participating Actor** | Instructor, student, administrator. |
| **Precondition** | The instructor and student must have username and password. |
| **Flow of event** | * The system displays the Login page. * The users enter their user name and password to enter to the system. * The user of the system clicks Login button on the login page. * The system displays access page for the respective user. |
| **Alternative Flow of event** | **A1: Information Not Filled Message**   * The system displays “Please enter your user name and password!” message. * The system resumes at step 2.   **A2: Invalid Entry Message**   * The system displays “Incorrect User Name or Password!” massage. * The system resumes at step 2. |
| **Post condition** | The user entered to the system and can access the system. |

**Table 7** Use case description for add mark use case

|  |  |
| --- | --- |
| **Use case name** | Add mark |
| **Participating Actors** | Teacher |
| **Description** | The instructor Insert the mark of students |
| **Precondition** | Username and password to login |
| **Flow of events** | * Instructor selects the add mark option. * System displays add grade page. * Instructor selects which course grade to be posted. * The system posts the grade Use case ends. |
| **Alternative Flow of events** | * System check the data entered. * System inform to the instructor, you enter incorrect data or empty data and reenter. * The system returns to step 2 of normal course. * Use case ends. |
| **Post condition** | The grade posted into student profile. |

**Table 8** Use case description for view grade use case

|  |  |
| --- | --- |
| **Use case name** | View Grade |
| **Participating Actors** | **Student** |
| **Description:** | Students need to view grades |
| **Precondition** | Username and Password to login |
| **Flow of events** | * A user selects the view grade option. * System request what the user needs. * The user selects the course grade to be displayed. * The system displays the request grade automatically. * The user views the grade. * Use case ends. |
| **Alternative Flow of events:** | * The requested course grade is not complete to be viewed. * The system prompts the student the request grade is not complete. * The system returns to step 2 of normal course. * Use case ends. |
| **Post condition** | Grades are available for students. |

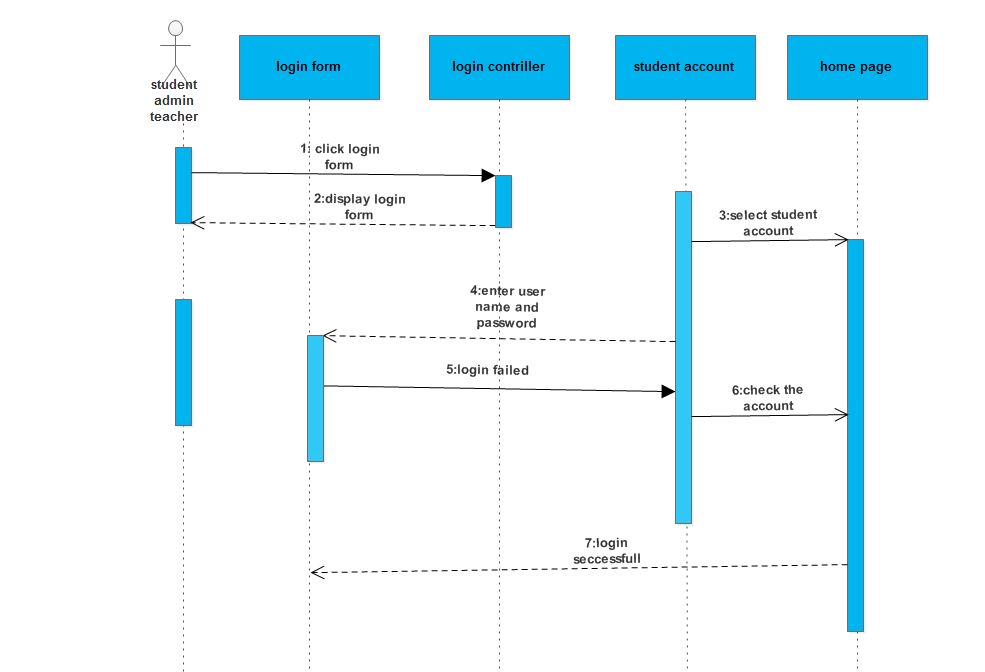
**Table 9** Use case description for submit grade use case

|  |  |
| --- | --- |
| **Use case name** | Submit Grade |
| **Participating Actors** | Instructor |
| **Description** | Enables student to submit his/her answer |
| **Precondition** | Username and Password |
| **Flow of events** | * The user selects the submit grade option * System request what the user need * The user select the course grade for submitting * System submit the grade * Use case end |
| **Alternative Flow of events** | * User enters invalid grade * System inform ,you enter wrong data * The system return to step 2 of normal course * Use case ends |

# 2.7 Sequence diagram

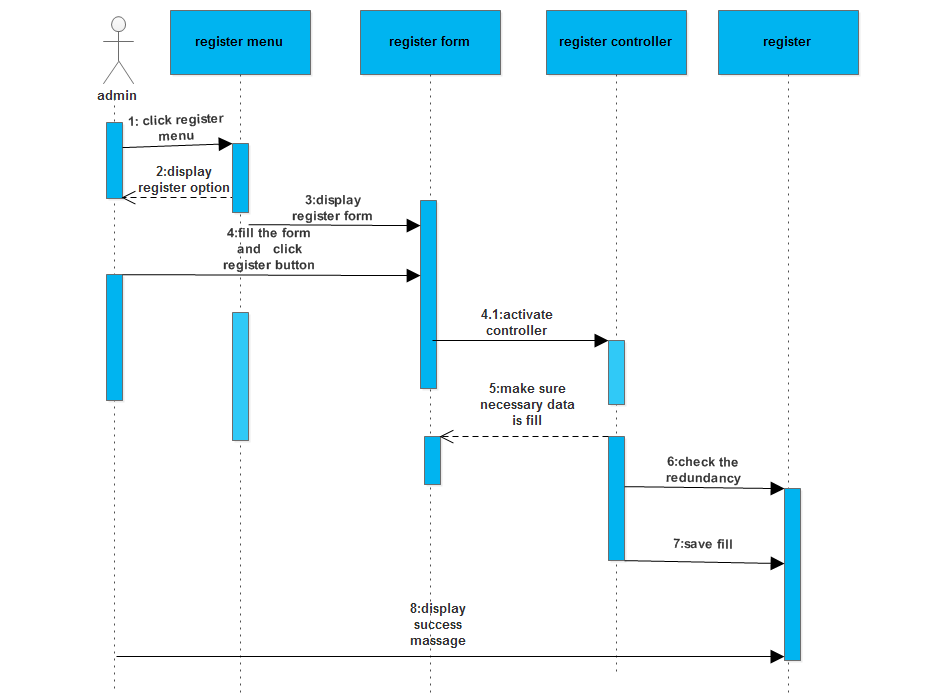
Sequence diagram is the most common kind of interaction diagram, which focuses on the message interchange between a number of lifelines.Sequence diagram describes an interaction by focusing on the sequence of messages that are exchanged, along with their corresponding occurrence specifications on the lifelines.

## 2.7.1Sequence diagram for login



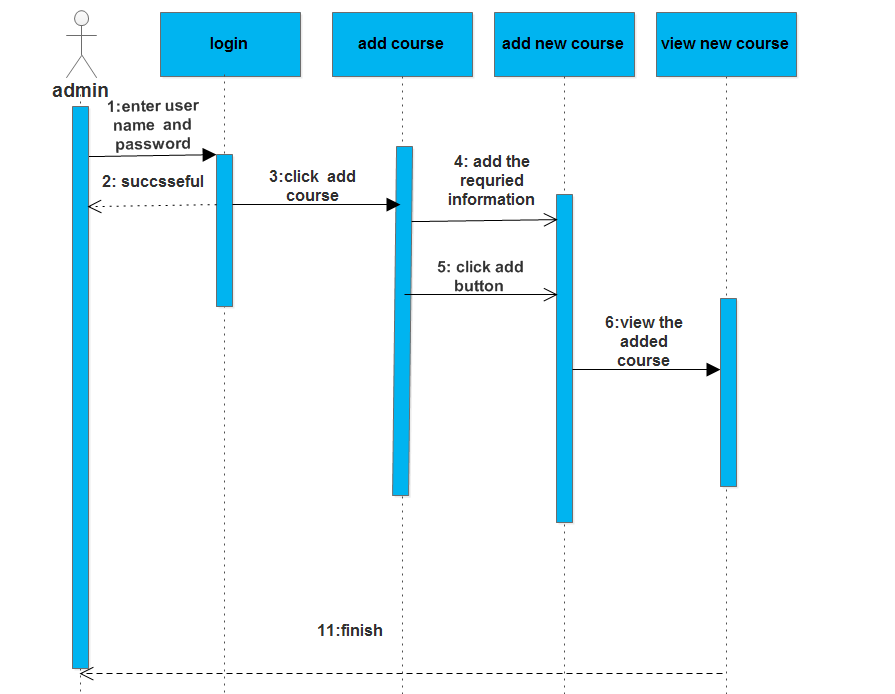
*Figure2 sequence diagram for login use case*

## 2.7.2 Sequence diagram for register



*Figure3 sequence diagram for register*

## 2.7.3 Sequence diagram for add course

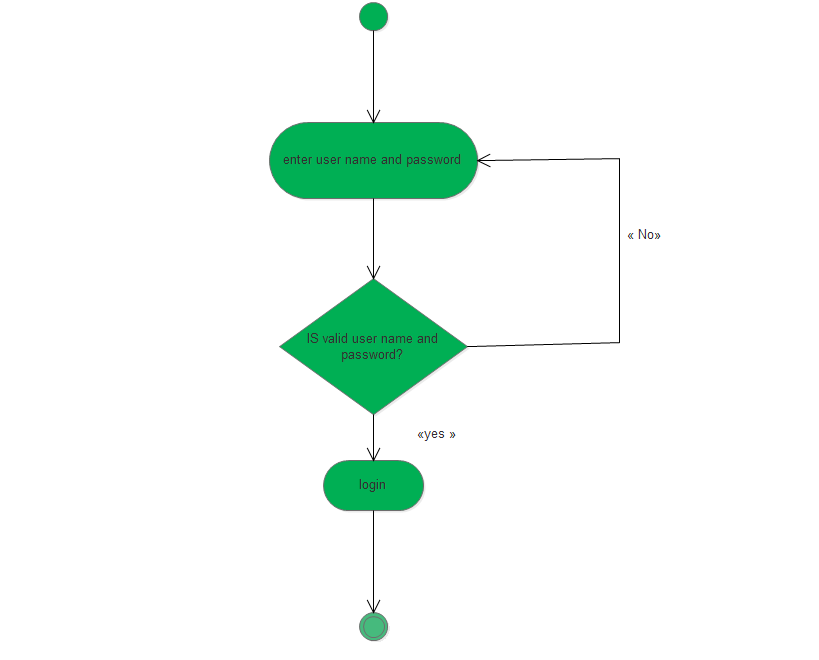


*Figure 4 sequence diagram* for add course

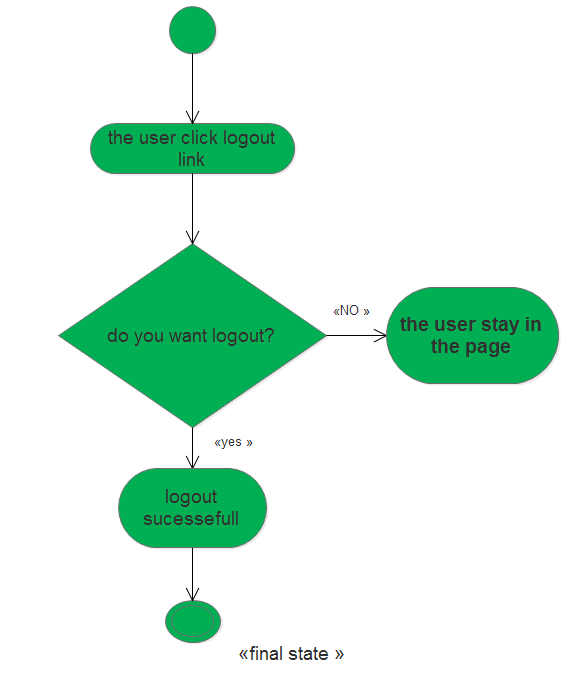
# 2.8 Activity diagrams

Activity diagrams, which are related to program flow plans (flowcharts), are used to illustrate activities. In the external view, we use activity diagrams for the description of those business processes that describe the functionality of the business system. Contrary to use case diagrams, in activity diagrams it is obvious whether actors can perform business use cases together or independently from one another. Activity diagrams allow you to think functionally. Purists of the object-oriented approach probably dislike this fact. We, on the other hand, regard this fact as a great advantage, since users of object-oriented methods, as well as users of functional thinking patterns, find a common and familiar display format, which is a significant aid for business-process modeling. Because it is possible to explicitly describe parallel events, the activity diagram is well suited for the illustration of business processes, since business processes rarely occur in a linear manner and often exhibit parallelisms. Activity diagrams can be developed in various degrees of detail. They can be refined step by step. In the external view, activity diagrams, just like use case diagrams, exclusively represent business processes and activities from the outside perspective.

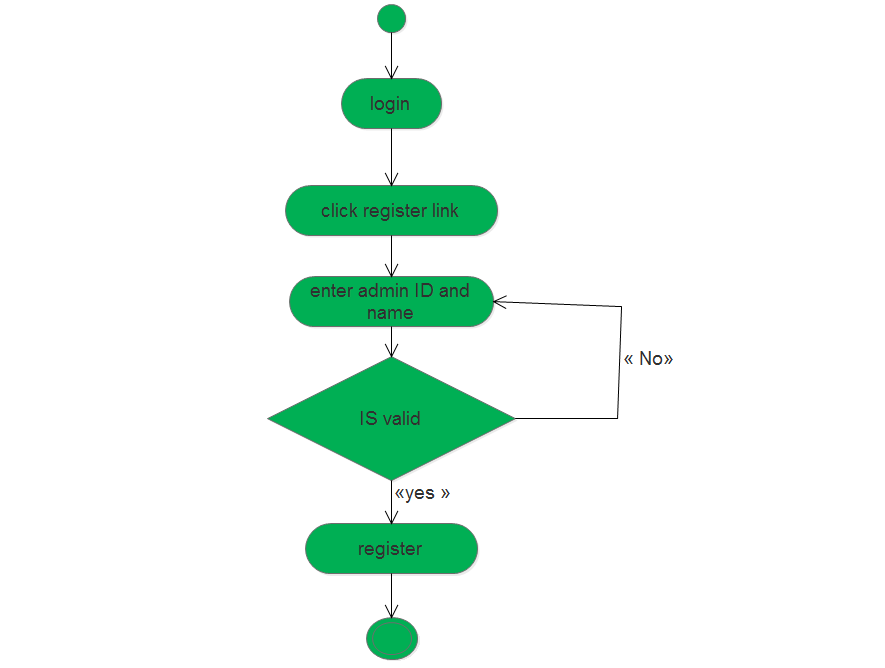
## 2.8.1 Activity diagram for login

*****Figure 5 Activity diagram for login*

## 2.8.2 Activity diagram for logout

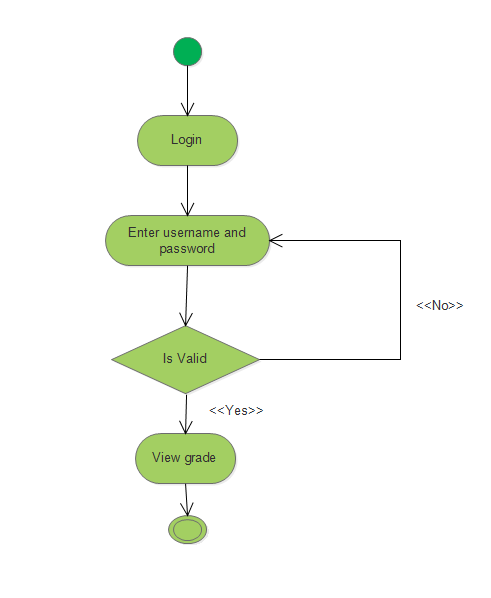
*****Figure 6 Activity diagram for logout*

## 2.8.3 Activity diagram for register



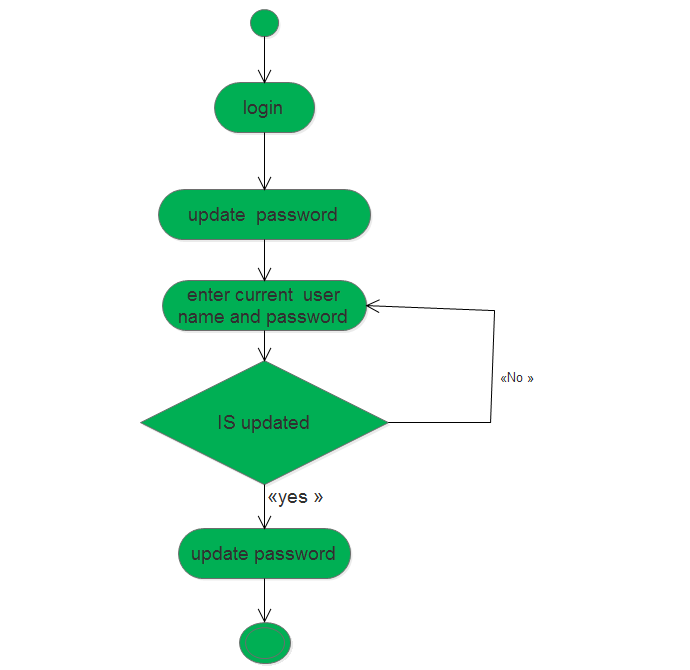
*Figure 7 Activity diagram for register*

## 2.8.4 Activity diagram for view grade



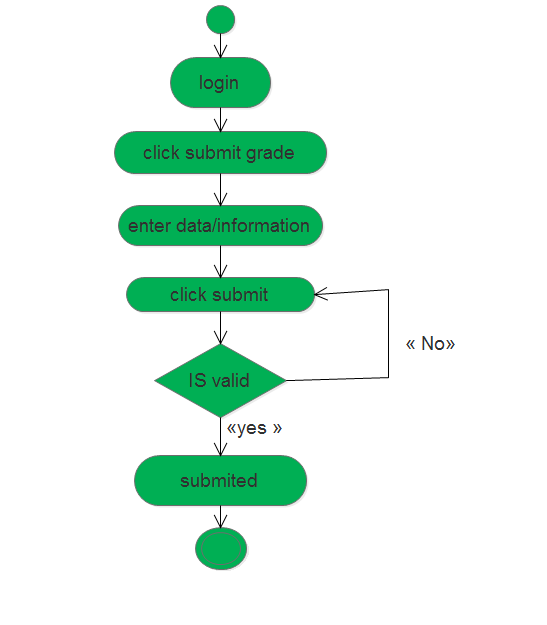
*Figure 8 Activity diagram view grade*

## 2.8.5 Activity diagram for update password



*Figure 9 Activity diagram for update password*

## 2.8.6 Activity diagram for submit grade



*Figure 10Activity diagram for submit grade*

# 2.9 Class Diagram

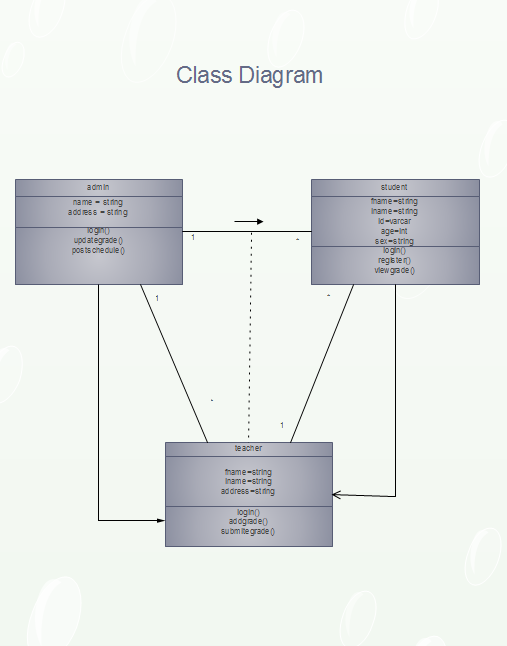
The class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application.

The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages.

The class diagram shows a collection of classes, interfaces, associations, collaborations and constraints. It is also known as a structural diagram.

So the purpose of the class diagram can be summarized as:

* Analysis and design of the static view of an application.
* Describe responsibilities of a system
* Base for component and deployment diagrams



*Figure 11class diagram*

## Chapter Three

## 3.1 system design

System design is the process and focuses on decomposing the system into manageable parts. During requirements analysis, we concentrated on the purpose and the functionality of the system design. During system design, we focus on the processes, data structures, and software and hardware components necessary to implement it. The challenge of system design is that many conflicting criteria and constraints need to be met when decomposing the system. The analysis model describes the system completely from the actors’ point of view and serves as the basis of communication between the client and the developers. The analysis model, however, does not contain information about the internal structure of the logical, its hardware configuration, or, more generally, how the system should be realized. System design results in the following products:

* List of design goals, describing the qualities of the system that developers should optimize.
* Webpage architecture, describing the subsystem decomposition in terms of subsystem responsibilities, dependencies among subsystems, subsystem mapping to hardware, and major policy decisions such as control flow, access control, and data storage.

# 3.2 Collaboration Diagram

Collaboration diagram, also called a communication diagram or interaction diagram is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML).

## 3.2.1 Collaboration diagram for login

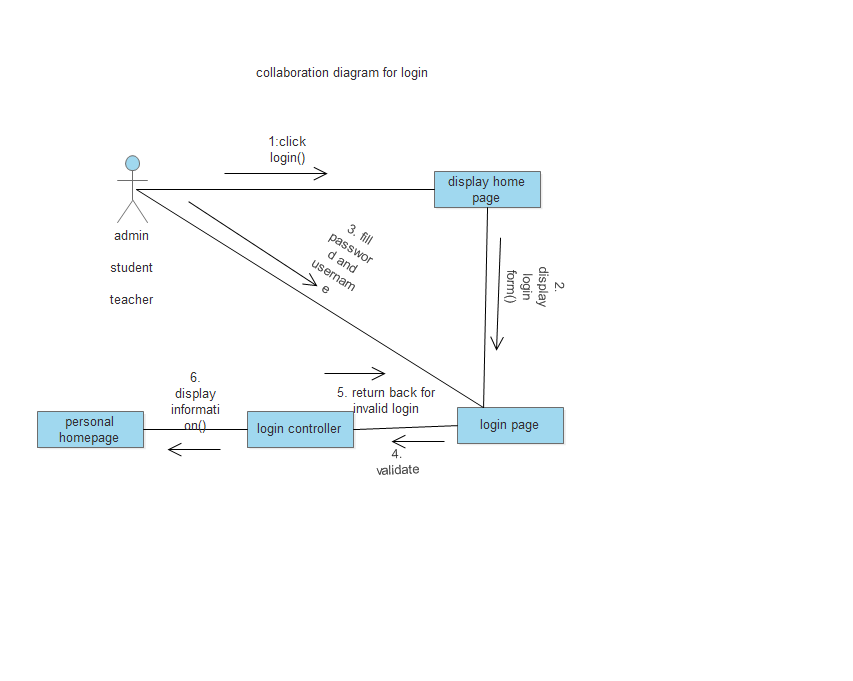


Figure 12 collaboration diagram for login

## 3.2.2 Collaboration diagram for logout

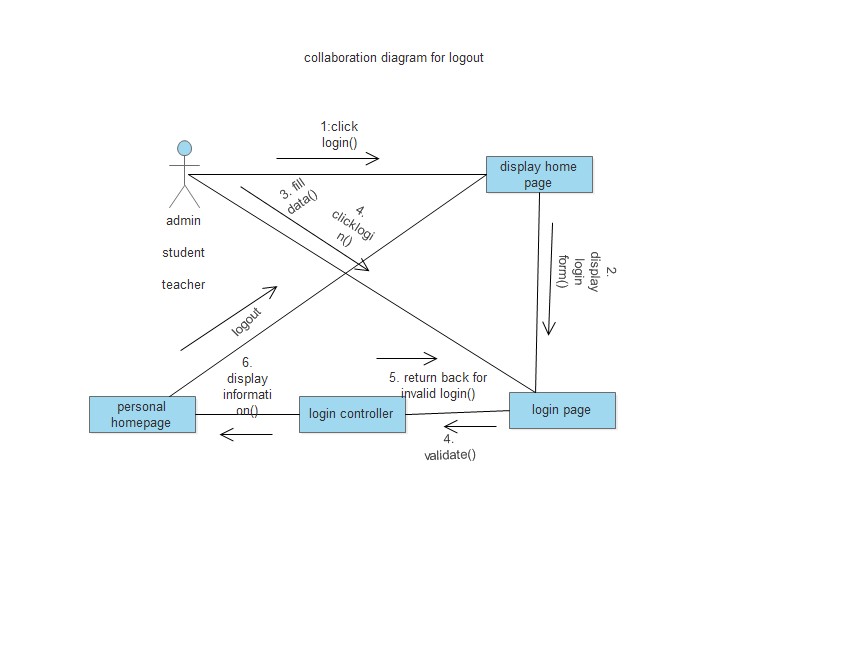


Figure 13 collaboration diagram for logout

3.3 persistence diagram

Persistence of our object can be achieved by relational database since it used as machine to make object persistent. It describes the persistent data aspect of software system. Our system includes the basic table that handles the data of system implemented using MySQL server.

**Mapping class and relational table**

Mapping refers how objects and their relationship are stored in relational database. The mapping of the data to be persisted in our system is given as follows

# 3.4 Student table

Table 10

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| student ID | Username | Password | Contact  number | Upload  Photo | Fame | Lame | middle name | address | gender | Place of birth |
| Int(10) | Varcare(30) | Varcare(30) | Varecare(30) | Varcare(30) | Varcare(30) | Varcare(30) | Varcare(30) | Varcare(30) | Varcare(30) | Varcare(30) |

# 3.5 Teacher table

Table 11

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| teacher ID | Username | Password | Contact  number | Upload  Photo | Fame | Lame | middle name | address | position | Dateadded |
| Int(103) | Varcare(34) | Varcare(34) | Varecare(34) | Varcare(34) | Varcare(34) | Varcare(34) | Varcare(34) | Varcare(34) | Varcare(34) | Varcare(34) |

# 3.6 Admin table

Table 12

|  |  |  |
| --- | --- | --- |
| admin ID | Username | Password |
| Int(12) | Varcare(32) | Varcare(32) |

# 3.7 Deployment diagrams

Deployment diagrams are used for describing the hardware components where software components are deployed.Component diagrams and deployment diagrams are closely related.

Component diagrams are used to describe the components and deployment diagrams shows how

they are deployed in hardware.UML is mainly designed to focus on software artifacts of a system. But these two diagrams are special diagrams used to focus on software components and hardware components. So most of the UML diagrams are used to handle logical components but deployment diagrams are made to focus on hardware topology of a system. Deployment diagrams are used by the System engineers.

The purpose of deployment diagrams can be described as:

* Visualize hardware topology of a system.
* Describe the hardware components used to deploy software components.
* Describe runtime processing nodes.

## 3.8 deployment diagrm

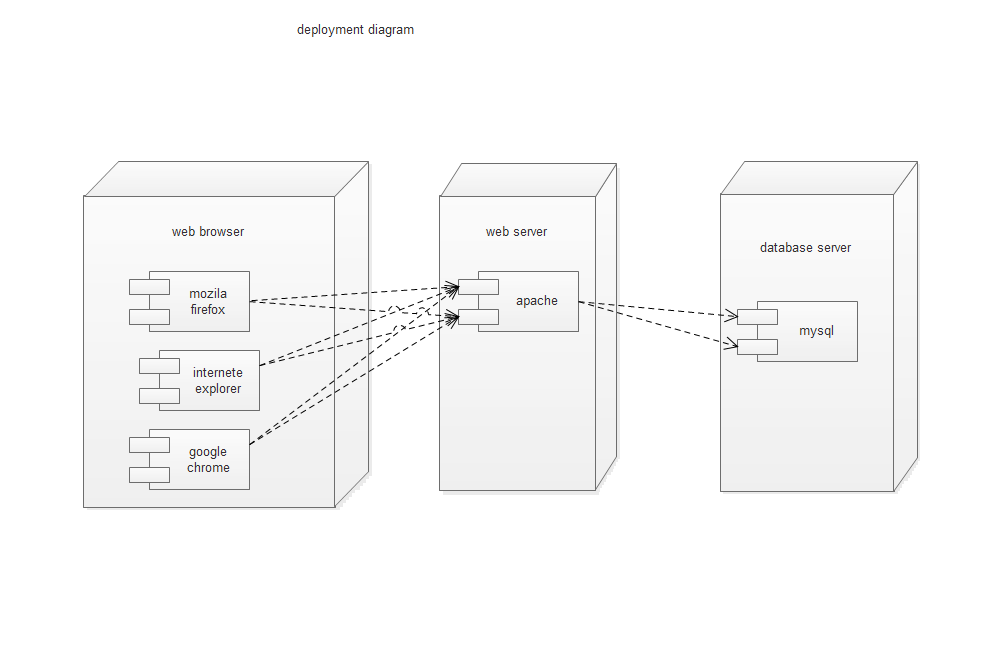


Figure 14 deployment diagram

## 

## Chapter 4

## System Implementation and Testing

## Introduction

System implementation is the construction of the new system and the delivery of that system into production (that is, the day-to-day business or organization operation)The Implementation Phase has one key activity: installing the new system in its target environment.  Supporting actions include training end-users and preparing to turn the system over to maintenance personnel.  After this phase, the system enters the Operations and Maintenance Phase for the remainder of the system’s operational life.

## Interface Design

Interface design is the process of putting clear visualization and graphical sample of the designed website. The following are some of the interfaces of the system.

Home page

The Home page contains all the links that students of the registrar can see that background, logout, add and update and also provides the student to have different information about the online grading system.

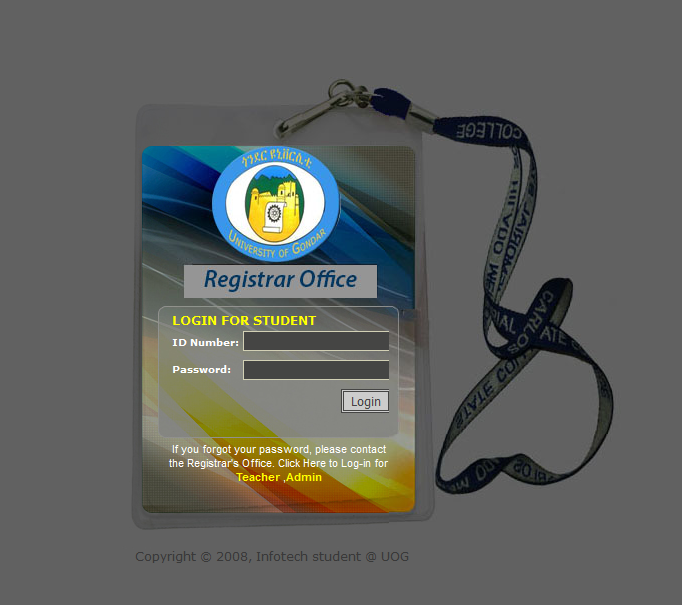


*Figure 15 User interface design diagram for home page*

# 4.3 Login page

The login page enables students, admin and teacher to enter to the system.

They can login to the system by entering the correct user name and password.



*Figure 16 User interface design diagram for login*

## 4.3.1 Sample code for home page

**<?phpsession\_start();?>**

**<?php**

**if($\_SESSION['logged'] != 'true'){**

**header('location:index.php');}**

**?>**

**<!DOCTYPE html >**

**<html>**

**<link rel="shortcut icon" href="images/logo.png">**

**<link rel="icon" href="images/logo.png" type="image/png" >**

**<style type="text/css">**

**<!--**

**.style7 {font-size: 26px}**

**.style8 {font-family: tahoma}**

**.style9 {font-size: 12px; }**

**.style10 {**

**color: #d1e050;**

**font-style: italic;**

**}**

**.style11 {color: #d1e050}**

**-->**

**</style>**

**<head>**

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

**<title>Home admin</title>**

**<meta name="keywords" />**

**<meta name="description" />**

**<link href="templatemo\_style.css" rel="stylesheet" type="text/css" />**

**<body>**

**<div id="templatemo\_container">**

**<div id="nav">**

**<?php include('include/header.php')?>**

**</div><!-- end of menu -->**

**<div id="templatemo\_header">**

**<div id="templatemo\_special\_offers" style="margin-left: 650px; margin-top: 85px; width: 254px;"><p align="center" class="style4"> Welcome </p>**

**<p align="center" class="style4">Administrator </p>**

**</div>**

**</div>**

**<!-- end of header -->**

**<div id="templatemo\_content">**

**<!-- end of content left -->**

**<div id="contentokana">**

**<div id="ndpah" align="center"><imgsrc="images/3.jpg" width="900" height="114"/></div>**

**<div class="style8" id="contento">**

**<br><p align="center" class="style7">Welcome to university of gondar</p>**

**<p class="style7">&nbsp;</p>**

**<p align="justify" class="style9 style10">college of nathural and computational science(CNCS) is a public, state-owned college, the main campus of which is in GONDAR CITY, It provides higher technological, professional and vocational instruction and training in science, agriculture and industrial fields, as well as short-term or vocational courses. It is notable for being the only state college in the province of. </p><br />**

**<p><imgsrc="images/chmsc.png" width="142" height="117" style="margin-top: 18px;" /><!-- end of content right -->**

**<table cellspacing="5" style="float: right; margin: -95px -16px 0px 0px;">**

**<tbody>**

**<tr><th scope="row"><span class="style11">Motto</span></th>**

**<td>&quot;He who hath a trade, hath an estate&quot;</td>**

**</tr>**

**<tr><th scope="row"><span class="style11">Established</span></th>**

**<td>1954</a></td>**

**</tr>**

**<tr><th scope="row"><span class="style11">Type</span></th>**

**<td>State college</td></tr>**

**<tr><th scope="row"><span class="style11">Location</span></th>**

**<td>Gondar university, atsetewdros campus </td></tr>**

**<tr><th scope="row"><span class="style11">Campus</span></th>**

**<td>Urban, 5 hectares</td></tr>**

**</tbody></table>**

**</div>**

**<div id="content" align="right">**

**<br />**

**&nbsp;&nbsp;&nbsp;&nbsp;**

**<div id="text">**

**</div>**

**<div class="cleaner\_with\_height" style="margin-top: 18px;">><p><imgsrc="images/text1.png" width="281" height="369" class="img" /><p></div>**

**</div>**

**</div>**

**</div>**

**<div id="templatemo\_footer" style="margin-top:12px;">**

**<?phpinclude('include/footer.php')?></div>**

**</div>**

**</body>**

**</html>**

## 4.3.2 Sample code for login

**<?phpsession\_start(); ?>**

**<!DOCTYPE html >**

**<html>**

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

**<script type="text/javascript">**

**$(document).ready(function(){**

**$(".menu a").hover(function() {**

**$(this).next("em").animate({opacity: "show", top: "-75"}, "slow");**

**}, function() {**

**$(this).next("em").animate({opacity: "hide", top: "-85"}, "fast");**

**});**

**});**

**</script>**

**<style type="text/css">**

**.menu {**

**margin: 0px 0 0;**

**padding: 0;**

**list-style: none;**

**}**

**.menu {**

**padding: 0;**

**margin: 0 0px;**

**float: left;**

**position: relative;**

**text-align: center;**

**}**

**.menu a {**

**padding: 0px 0px;**

**color: #000000;**

**width: 0px;**

**text-decoration: none;**

**font-weight: bold;**

**}**

**.menu em {**

**background: url(student/images/hover.png) no-repeat;**

**width: 180px;**

**height: 45px;**

**position: absolute;**

**color: #000000;**

**top: -85px;**

**left: -15px;**

**text-align: center;**

**padding: 20px 12px 10px;**

**font-style: normal;**

**z-index: 2;**

**display: none;**

**}**

**</style>**

**<link rel="shortcut icon" href="student/images/chmsc.png">**

**<link rel="icon" href="student/images/chmsc.png" type="image/png" >**

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

**<title>Log-In students</title><link href="student/stylelog.css" rel="stylesheet" type="text/css" />**

**</head>**

**<body>**

**<div id="layer01\_holder">**

**<div id="left"></div>**

**<div id="center"></div>**

**<div id="right"></div></div>**

**<div id="layer02\_holder">**

**<div id="left"></div>**

**<div id="center"></div>**

**<div id="right"></div></div>**

**<div id="layer03\_holder">**

**<div id="left"></div>**

**<div id="center">**

**<table width="100%" >**

**<tr>**

**<td colspan="2">LOGIN FOR STUDENT</td>**

**</tr>**

**<form id="form1" name="form1" method="post">**

**<tr>**

**<td><label>ID Number:</label></td>**

**<td><input name="UserName" type="text" id="textfield" /></td>**

**</tr>**

**<tr>**

**<td><label>Password:</label></td>**

**<td><input type="password" name="Password" id="textfield2" style="margin-top:5px;" /></td>**

**</tr>**

**<tr>**

**<td colspan="2" ><input type="submit" name="login" id="button" value="Login" />**

**<?php**

**require ("student/db.php");**

**$error = "";**

**if (isset($\_POST['login']))**

**{**

**$UserName = trim($\_POST['UserName']);**

**$Password = trim($\_POST['Password']);**

**// sending query**

**$result = mysql\_query("SELECT \* FROM student WHERE UserName = '$UserName' AND Password =**

**'$Password'");**

**if (!$result)**

**{**

**die("Query to show fields from table failed");**

**}**

**$numberOfRows = mysql\_num\_rows($result);**

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

**if ($numberOfRows == 0)**

**{**

**echo " <br><center><font color= 'red' size='1'>Invalid UserName and Password!</center></font>";**

**}**

**else if ($numberOfRows> 0)**

**{**

**//session\_start();**

**session\_register('is');**

**$\_SESSION['is']['login'] = TRUE;**

**$\_SESSION['is']['SUserName'] = $\_POST['UserName'];**

**$\_SESSION['Suserid']=$row['studentID'];**

**$\_SESSION['logged']="true";**

**$session = "1";**

**header("location:student/home.php");**

**}**

**}**

**?>**

**</td>**

**</tr>**

**</form>**

**</table>**

**</div>**

**<div id="right"></div>**

**</div>**

**<div id="layer04\_holder">**

**<div id="left"></div>**

**<div id="center" style="padding-top: 8px;">**

**<div class="menu">**

**If you forgot your password, please contact the Registrar's Office. Click Here to Log-in for<br /><a href="teacher/index.php">Teacher</a>**

**<em>Click Here To Log-in For Teacher.</em>,<a href="01-for login page/index.php">Admin</a><em>Click Here To Log-in For The Admin.</em></div></div>**

**<div id="right"></div>**

**</div>**

**<div id="layer05\_holder">**

**</div></body></html>**

**Sample code for add course**

**<?phpsession\_start();?>**

**<?php**

**if($\_SESSION['logged'] != 'true'){**

**header('location:index.php');}**

**?>**

**<!DOCTYPE html >**

**<html>**

**<link rel="shortcut icon" href="images/chmsc.png">**

**<link rel="icon" href="images/chmsc.png" type="image/png" >**

**<head>**

**<meta charset="utf-8">**

**<link rel="stylesheet" href="jquery-ui/development/themes/base/jquery.ui.all.css">**

**<scriptsrc="jquery-ui/development/jquery-1.5.1.js"></script>**

**<scriptsrc="jquery-ui/development/ui/jquery.ui.core.js"></script>**

**<scriptsrc="jquery-ui/development/ui/jquery.ui.widget.js"></script>**

**<scriptsrc="jquery-ui/development/ui/jquery.ui.datepicker.js"></script>**

**<link rel="stylesheet" href="jquery-ui/development/demos/demos.css">**

**<script>**

**$(function() {**

**$( "#datepicker" ).datepicker({**

**changeMonth: true,**

**dateFormat: 'yy-mm-dd',**

**changeYear: true,**

**showButtonPanel: true**

**});**

**});**

**</script>**

**<script language="javascript" >**

**functionisNumberKey(evt)**

**{**

**varcharCode = (evt.which) ? evt.which :event.keyCode**

**if (charCode> 31 && (charCode< 48 || charCode> 57))**

**return false;**

**return true;**

**}**

**</script>**

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

**<title>Add New Course</title>**

**<link href="templatemo\_style.css" rel="stylesheet" type="text/css" />**

**<style type="text/css">**

**<!--**

**.style7 {**

**font-family: "Times NewRoman";**

**font-size: 15px;**

**font-weight: regular;**

**}**

**.style14 {font-family: "Times NewRoman"; font-size: 15px; font-weight: regular; }**

**.style17 {font-family: "Times NewRoman"; font-size: 15px; font-weight: regular; }**

**-->**

**</style>**

**</head>**

**<body>**

**<div id="templatemo\_container">**

**<div id="nav">**

**<?php include('include/header.php')?>**

**</div><!-- end of menu -->**

**<div id="templatemo\_header">**

**<div id="templatemo\_special\_offers" style="margin-left: 577px; margin-top: 85px; width: 254px;">**

**<p align="center" class="style4"> Welcome </p>**

**<p align="center" class="style4">Administrator </p>**

**</div>**

**</div>**

**<!-- end of header -->**

**<div id="templatemo\_content">**

**<!-- end of content left -->**

**<div id="templatemo\_content\_right">**

**<form id="form1" name="form1" method="post" action="">**

**<h1> Add New Course </h1>**

**<table border="0">**

**<tr>**

**<td width="115"><span class="style7">Course Code:</span></td>**

**<td><input type="text" name="coursecode"/></td>**

**</tr>**

**</table>**

**<table>**

**<tr>**

**<td width="115"><span class="style7">Course Description:</span></td>**

**<td><input type="text" name="coursedesc" size="64" /></td>**

**</tr>**

**</table>**

**<table>**

**<tr>**

**<td width="115"><span class="style7">Hours/Week:</span></td>**

**<td><input type="text" name="hrsweek" onKeyPress="return isNumberKey(event)" /></td>**

**<td>&nbsp;</td>**

**<td width="97"><span class="style17">Units:</span></td>**

**<td><input type="text" name="units" onKeyPress="return isNumberKey(event)"/></td>**

**</tr>**

**<tr>**

**<td><span class="style14">Pre-Requesite</span></td>**

**<td><input type="text" name="prereq" /></td>**

**<td>&nbsp;</td>**

**<td width="115"><span class="style7">Date Added:</span></td>**

**<td><input type="text" name="coursedateadd" id="datepicker"/></td>**

**</tr>**

**<tr>**

**<td>&nbsp;</td>**

**<td><input type="submit" name="addcourse" value="Add" class="btn"/></td>**

**</tr>**

**</table>**

**<a href="assign\_course.php">Assign existing course </a>**

**</form>**

**</div>**

**<!-- end of content right -->**

**<div class="cleaner\_with\_height">&nbsp;</div>**

**</div><!-- end of content -->**

**<div id="templatemo\_footer">**

**<?php include('include/footer.php')?>**

**</div>**

**<!-- end of footer -->**

**</div><!-- end of container -->**

**<?php**

**if (isset($\_POST['addcourse']))**

**{**

**include 'db.php';**

**/\*$prog=$\_POST['program'] ;**

**$major=$\_POST['major'] ;**

**$year=$\_POST['year'] ;**

**$sem=$\_POST['semester'] ;\*/**

**$course=$\_POST['coursecode'] ;**

**$pre=$\_POST['prereq'] ;**

**$coursedesc=$\_POST['coursedesc'] ;**

**$hrs=$\_POST['hrsweek'] ;**

**$unit= $\_POST['units'] ;**

**$date=$\_POST['coursedateadd'] ;**

**mysql\_query("INSERT INTO `course`(coursetitle,prerequesit,coursedescription,hrsperweek,unit,dateadded)**

**VALUES ('$course','$pre','$coursedesc','$hrs','$unit','$date')");**

**header("Location: view\_course.php");**

**}**

**?>**

**</body>**

**</html>**

# 

# 

# 4.4 Testing

Testing is a process of executing a program with the intent of finding an error. A successful test is one that uncovers an as yet undiscovered error. Our Objective is to design test processes that systematically uncover different classes of errors and do so with minimum amount of time and effort.

## 4.4.1 Unit testing

Verification (A set of operations that the software correctly implemented a particular function) on the smallest element of the program i.e. the modules are tested alone in order to discover any error in the code**.**

## 4.4.2 Integrated testing

The process of bringing together all the modules that a program comprises for testing purpose.

## 4.4.3 System testing

System test insures that the entire integrated software system meets requirements. It tests a configuration to insure known and predictable results. System testing is based on process description and flows, emphasizing pre-driven process links and integration points. In essence system testing is not about checking the individual parts of design, but about checking the system as a whole. In effect it is one giant component.

# 4.5 Functionality of the system

In this test we have input the correct data to system and check if the system performs correctly or not. For example in the student add registration form we checked that it work perfectly as expected if we input data correctly, and display successful message.



## 4.6 Error handling

In this testing, we have tested the system by inputting incorrect data and waiting for the result. For example the picture below show, when we try to login with invalid username and password, but the system display error message, and ask to try again with the correct data.



*Figure 17 error handling diagram*

## Chapter 5

## Conclusion and Recommendation

# 5.1 Conclusion

Considering the drawbacks of the existing system and importance of new technologies the developed system, online grade system for registrar office of Atse-Tewodros campus in UOG is very useful to simplify the grading system. The system performs works better than existing system (usability, speed, efficiency and effectiveness). Security also included in this system developed and every user can access the required services. The system is also very useful in minimizing time and other utilities wastage.

In general the system that the team developed will benefit the enterprise in by changing its business range from manual to online level.

The overall benefits of the system to the enterprise are:

* provide sufficient security
* Minimize the time required to perform task by getting order to be served.
* Facilitates the services given by the enterprise specially the customer registration.
* Reducing human power and cost that are spent on the manual system.

As a system developing team the group member have acquired more knowledge and experience about the different phases of the software development life-cycle. As software developers, the team has worked together and assessed risks, and minimizes them.

They also cooperated to enterprise staff that we give more knowledge and experience about computer technology; that they are deciding to use such system for their efficient work and reduce their work force by human power.

# 5.2 Recommendations

Current grading System is manual system or file based system and far away from advanced technology and information. Since this developed system is advanced our system have to recommend by registrar office of Atse-Tewodros campus in UOG to accept and use this system because it is easy to use and save the time and resources’ for the registrar office.

**APPENDIX**

|  |  |
| --- | --- |
| Symbol | Description |
|  | Actor |
|  | System boundary |
|  | Decision |
|  | Use case |
|  | Class |
|  | Component interface |
|  | Deployment diagram |
|  | Message line extends from the lifeline of one object to the lifeline. |
|  | Return message extend from the lifeline of one object to the lifeline |
|  | Starting point of activity/state diagram |
|  | Ending point of activity/state diagram |

Reference

[1] Albin, S. T. (2003). The Art of Software Architecture – Design

Methods and Techniques. Wiley Publishing, Inc. [Albin, 2003]

[2] IEEE Architecture Planning Group. (2006). IEEE Std 1471-2000 IEEE

Recommended Practice for Architectural Description of Software-Intensive Systems. IEEESoftware. [APG, 2006]

[3] Dijkstra, A. (2008). Course Assignments - Architecture project. [Dijkstra, 2008]

[4] http://www.cs.uu.nl/wiki/Swa/CourseAssignments [Accessed 15-02-2016].

[5] Kruchten, P. (1995). Architectural Blueprints: The 4+1 View Model of Architecture. IEEE Software [Kruchten, 1995].

[6] <https://www.uog.edu.et> [Accessed 10-5-2016].