Wolkite University

We Strive for Wisdom!

WOLKITE UNIVERSITY

College of Computing and Informatics

Department of Information System

Project Title:Web Based Distance Education Management System For Great Vision College In Wolkite City

Team member

	Name	ID.No.
1	Yeshineh Abebaw……..	NSR2201/13
2	Endalew Shumet……	NSR/0753/13
3	Yesuf Asefa ……..……..	NSR/2204/13

Advisor Name:Mr Kabtamu.D

October, 2023

Wolkite, Ethiopia

Contents

1.Introduction

This proposal presents the concept of an Web Based Distance Education Management System, which aims to address the challenges faced by educational institutions in managing and delivering distance learning programs. With the increasing demand for flexible and accessible education, especially in recent times, there is a growing need for an efficient and comprehensive system to support Web Based distance education.

The proposed system will serve as a centralized platform for educational institutions to manage various aspects of distance learning, including course administration, student enrollment, content delivery, assessments, and communication. By leveraging technology, this system aims to streamline processes, enhance collaboration, and improve the overall learning experience for both students and educators.

Implementing an Web Based Distance Education Management System will enable educational institutions to overcome geographical barriers and reach a wider audience. It will provide students with the flexibility to access educational resources at their convenience, regardless of their location. Through virtual classrooms, discussion forums, and multimedia content, students will engage in interactive learning experiences.

2.Background of Great Vision College

Great Vision College, founded in 2013 E.C in the city of Wolkite, the college introduced diploma programs in key departments such as accounting, management, and HRM. These programs were designed to prepare students with practical skills and knowledge, preparing them for the workforce.

As time progressed and the college gained valuable insights into the evolving needs of its students, it decided to take a significant step forward. To further empower its students and provide them with more extensive academic opportunities, the college expanded its department offerings.

In this expansion, Great Vision College introduced new departments like accounting, management, and economics, along with an enhanced HRM program at the degree level. This decision was rooted in the college commitment to nurturing well-rounded, highly skilled graduates who are not only equipped with practical knowledge but are also capable of attend advanced studies and contributing to their fields at a higher academic level.

3. Statement of the Problem

Great Vision College in Wolkite town currently relies on a manual distance education

management system, which poses several challenges and limitations. The absence of

an efficient and automated system create obstacle the college's ability to effectively manage and deliver distance learning programs. This statement of problem aims to highlight the key issues associated with the manual distance education management system at Great Vision College.

The manual system results in time consuming and labor intensive administrative tasks. The process of enrolling students, managing course registrations, and maintaining accurate records is prone to errors and inefficiencies.This not only consumes valuable staff resources but also leads to delays and inaccuracies in student data management. Communication between students, instructors, and administrative staff is hindered by the lack of a centralized platform. Important announcements, course materials, and assignment submissions are often exchanged through leading to confusion and miscommunication.

- Traditional education methods have limitations, restricting students to physical classrooms and fixed schedules.
- Managing student data and academic information manually is time consuming and prone to errors.
- The need for a better system to track and analyze student progress.
- These problems are uses manual system to process data.

4 . Objectives of the Project

4.1 General Objective

The main objective of this project is to design and implement web based distance education management system for Great Vision College in Wolkite.

4.2 Specific Objectives

To achieve the general objective mentioned above the following are specific objective:

- Study the existing manual system

- Gathering required information for proposed system
- Analyse the requirements of the proposed system
- Design and build a particular model of this proposed system
- Implementing the model with the flexible user interface
- Test and evaluate the proposed system

5. Scope and limitations

5.1 Scope of the Project

The following scopes are including in the project.

- Upload modules.
- Download module.
- View courses present from department.
- Viewing result online.
- Assign instructor.
- Posting announcement and information.
- Adding, updating and searching student information.
- Online registration
- Online payment
- Online Examination
- Grading system

5.2 Limitations

- Security Concerns.
- User Training and Support.
- Cost and Budget Constraints.

6.Feasibility Study

To determine the feasibility of the project, we have considered economic, technical, operational, and political factors:

6.1 Economic Feasibility

The project is economically feasible, requiring a reasonable investment to provide significant

benefits. It offers cost effective solutions and efficiency gains for students and administrators.

6.2 Technical Feasibility

The system to be developed by using technologically system development techniques such as HTML,CSS,JS ,PHP and also database without any problems and the group members have enough capability to develop the project. So the system will be technically feasible.

6.3 Operational feasibility

This Web Based system for distance education in Great Vision College will attain its desired objectives. It can solve the problems in distributing module and tutorials; therefore it will minimize the amount of effort to do all through manually.

6.4 Political Feasibility

The project aligns with the goal of providing accessible, efficient, and quality education, and it does not conflict with any government directives. It is politically feasible.

7 .Significance and beneficiary of the project

7.1 Significance of the Project

This system has the following benefits:

- Access of student detail information through internet.
- Reduce resource wastage.
- Support to distribute information.
- Easy to search and manages the student information.
- Reduce employees work load.
- The proper and ordered files of employees which has stability means which is not easily lost.
- Enabling students to learn at their own pace and convenience.

7.2 Beneficiaries of the Project

The project will benefit various stakeholders, including:

- **Educational Institutions**: Streamlined operations, improved course quality, and reduced administrative overhead.
- **Students**: Access to user-friendly, interactive online learning platforms.

- **Teachers**: Tools for efficient course management and effective communication with students.
- **Administrators**: Data-driven insights for curriculum development and quality enhancement.
- **IT Teams**: Opportunities to contribute to a cutting edge educational infrastructure.

8 .Methodology of the Project

8.1 Data Gathering Method

There are different data gathering technique which helps us to understand the present systems general activities and its procedures. We used three methodologies to gather the needed information. The methods we use for data collection are:

8.1.1 Interviewing

As a method for the collection of data about the activities in College of distance education we use interviewing method to understand peoples who belongs to the current system also we raised questions that helps us to develop the new system.

- ✓ Ask the background of the organization?
- ✓ Ask how to work the current system?
- ✓ Ask the problem of current system?

8.1.2 Document Analysis:

we analyze some documents and forms found in the office to get information about the current system.

- ✓ **Assessment Review:** Ensuring that assessments, such as quizzes, exams, and assignments, are aligned with learning outcomes and are fair and unbiased.
- ✓ **Financial Document Review**: Assessing financial documents, such as budgets and expenditure reports, to ensure efficient resource allocation.

8.2 System Analysis and Design

In the system analysis and design phase of a project we will use the object oriented approach that examines requirements from the perspective of the class and objects found in the problem domain. The reasons that we will use the object oriented approaches are:

- We can reuse methods for avoiding redundancy.
- To make it easier for teams of designers and programmers to work in a single software project

- The data and functions are encapsulated in the objects that help us for easily debugging purpose.
- It will increase consistency among analysis, design and programming activities.
- It will improve communication among users, analysis, design and programming
- Modification of the object implementation is easy because objects are loosely coupled.
- Understanding of the structure is easy because object oriented modelling represents real world entities.

8.3 System Development Model

In the system development model to develop good software we will use agile model because we have different reason such:

- We will iterate and increment with in each phase.
- We can easily control it, and it is flexible for developers.
- It primarily concentrates on the deliverable and fewer about paperwork.

8.4 Development Tools

8.4.1 Hardware tools

- Computer with internet connection
- Secondary storage device
- Flash disk (8 GB)
- Memory :8GB RAM
- Hard disk:300GB

8.4.2 Software tools

The different kind of software we used for developing the project is:

- MySQL server – to develop database application
- Microsoft office word 2013 for documentation.
- Edraw Max—to design Use case Diagram, design Sequence Diagram, design

- Activity Diagram and. Class Diagram, Deploy Diagram and other diagram
- Programming Language: - PHP, CSS, JavaScript and HTML

9 .Testing Procedure

Developing software for an Web Based Distance Education Management System is a complex process. No matter how diligently we progress through the phases of requirements, analysis, design, specification, and implementation, it's essential to ensure that significant faults are prevented from arising in the first place. To achieve this, a dedicated testing phase is required to identify and rectify any remaining issues before the system is ready for use. The testing process involves different levels, including unit testing, integration testing, and system testing.

9.1 Unit Testing

During unit testing, developers of the Web Based Distance Education Management

System will assess individual components, sub-procedures, and functions. Both

black-box and white-box testing methodologies will be applied to ensure the

reliability of the system.

❖ **Test Scenario:**

Scenario 1: Testing User Authentication and Examination Submission

Test whether the user authentication function correctly authenticates valid users.

Test whether the system allows students to submit examinations and records their submissions accurately.

❖ **Test Plan:**

- Identify the components and functions related to user authentication and examination submission.

- Create test cases for valid and invalid user credentials.
- Create test cases for submitting examinations.
- Execute the tests by providing different inputs.
- Verify that authentication and examination submission processes work as expected.

❖ Test Cases:

Test Case 1:

- Verify a valid user can log in successfully.
- Input: Valid username and password.
- Expected Output: User is logged in.

Test Case 2:

- Verify an invalid user cannot log in.
- Input: Invalid username and password.
- Expected Output: Authentication failure.

Test Case 3:

- Verify that a student can submit an examination.
- Input: Student selects an examination and submits answers.
- Expected Output: Examination submission is recorded.

9.2 Integration Testing

In integration testing, we assess how different system components work together to achieve the intended goals of the subsystems in the Web Based Distance Education Management System.

Test Scenario:

Scenario 2: Testing Course Registration, Payment, Examination Integration, and Grading

Test how the course registration module integrates with the payment system.

Ensure that students can register for a course, make a payment, take examinations, and receive grades accurately.

Test Plan:

- Identify the components related to course registration, payment, examination submission, and grading.
- Create test cases for different scenarios of registration, payment, examination submission, and grading.
- Execute the tests by simulating the entire process.
- Verify that the integration works as intended.

Test Cases:

Test Case 1:

- Verify that the system handles payment failures gracefully.
- Input: Student's payment fails.
- Expected Output: Registration is not confirmed, and an error message is displayed.

Test Case 2:

- Verify that a student can register for a course, pay for it, take an examination, and receive a grade.
- Input: Student registers, pays, takes an examination, and receives a grade.
- Expected Output: Registration, payment, examination submission, and grading are successful.

9.3 System Testing

System testing evaluates how well the various subsystems of the complete Web Based Distance Education Management System function cohesively to achieve the desired educational outcomes. This phase ensures that the system operates seamlessly and efficiently when used by students, instructors, and administrators.

Test Scenario:

Scenario 1:

- Testing End-to-End Educational Outcome, Examination, and Grading
- Evaluate how the entire system functions to support the educational process, including examinations and grading.
- Ensure that students, instructors, and administrators can use the system efficiently to achieve their goals.

Test Plan:

- Identify key user roles (students, instructors, administrators) and their goals, including examination and grading processes.
- Create test cases to cover common user journeys for each role, including examination and grading tasks.
- Execute the tests by simulating the actions of users in different roles.
- Verify that the system operates seamlessly and efficiently for all users in the context of examinations and grading.

Test Cases:

Test Case 1:

- Verify that an instructor can create and manage courses, grade examinations, and communicate with students.
- Input: Instructor logs in, creates a course, grades examinations, and communicates with students.
- Expected Output: All actions are completed successfully, including examination grading.

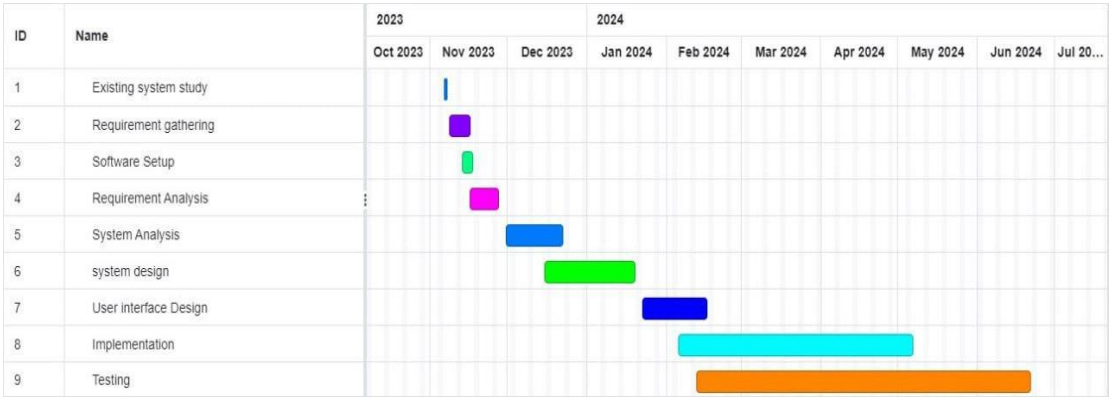
Test Case 2:

- Verify that an administrator can manage user accounts, oversee course enrollment, generate reports, and manage the grading system.

- Input: Administrator logs in, manages user accounts, checks course enrollment, generates reports, and manages grading.
- Expected Output: All actions are completed successfully, including grading tasks.

10. Project Plan and Budget

10.1 Project Plan



10.2 Project budget

No.	Material	Unit	Price per unit	Total price
1	Flash Disk	3(16 GB)	3*400	1200
2	Paper	1(‘desta’)	500	500
3	Pen	3	3*20	60
4	Pc(hp)	1	40000	40,000

5	Total			41,760

10.3 Project Team Organization

No.	Name	ID NO.	E-mail Address	Responsibilities
1	Yeshineh Abebaw	NSR/2201/13	yeshinehabebaw@gmail.com	Project Manager
2	Endalew shumet	NSR/0753/13	Endalews92@gmail.com	Programmer, system design
3	Yesuf Assefa	NSR/2204/13	Yusufassefa@9gmail.com	Systems Analyst
4	Mr.Kabtamu.D			Advisor

Chapter Two:

2. The Existing System

2.1 Overview of Existing System

The existing system Great Vision College Distance Education is manually operated.

Different activities are performed as follow:

The department gives education service for the students at these sectors. First the students apply to register by coming personally to one of the specified sectors. Then the department has its own criteria to approve the student's application. The center employee/supervisor checks the registration form filled by the student weather it satisfies the criteria or not. If the student satisfies these criteria he/she will be approved to be registered. Then after the approval, the student will pay the payment for the courses that he/she will take. Then the student will be registered and start the class by the schedule posted by the department. The department handles this information manually. And the file management system is paper based. So, the information recording or filling and retrieval system is too tedious, error prone, and also time taking. Distributing modules, allocating assignments, announcing registration and examination dates and some other activities are performed by distance education office.

2.2 Users of the Existing System

The user of existing system is students, registrar officers, Instructors, Department Head, Coordinator and Administrators. Their activities in the existing systems are:-

- **System Administrator:**

The system will allow the system administrator to perform the following tasks.

- create student and instructor account
- update student and instructor account
- delete student and instructor account
- view user's information
- generate user's password

- **Instructor:**

The system will allow the instructor to perform the following tasks.

- post progress quiz

- upload assignment
- upload module
- view assign course
- view student's information

❖ **Student:**

The system will allow the student to perform the following tasks.

- Applying to be registered.
- update her/his profile.
- download module
- submit assignment
- view grade report
- view his/her profile
- Take progress quiz

❖ **Registrar Officer:**

- enrollment management
- Add calendar
- prepare grade report
- update grade
- generate ID

❖ **Academic Dean:**

- Allows to add department
- Allows to assign instructor
- Allows to delete department

❖ **Finance staff**:

- financial reporting
- control payment
- change password

❖ **Vice President:**

- view academic schedule
- view module

- view student

❖ **College Dean:**

- course scheduling
- view notification
- change password

❖ **Department Head:**

- Add course
- Post exam
- Approve grade report
- update notice
- Approve course result

2.3 Major Functions of Existing System

Module Distribution:

Instructors play a pivotal role in distributing learning modules to students, a task that is currently part of the manual operations within the system.

Tutorial Sessions:

Instructors are responsible for conducting tutorial sessions, adding to the manual workload and potentially contributing to inefficiencies in the teaching process.

Course Result View:

Instructors are also tasked with view course results, requiring students to interact directly with them to access their grades, quizzes, and test outcomes.

Registration Process:

The registrar's office manages the registration process, encompassing responsibilities such as preparing grade reports, transcripts, and registering students for courses.

Academic Oversight:

The academic dean has a central role in controlling the overall teaching-learning process, including supervising various aspects of the academic schedule.

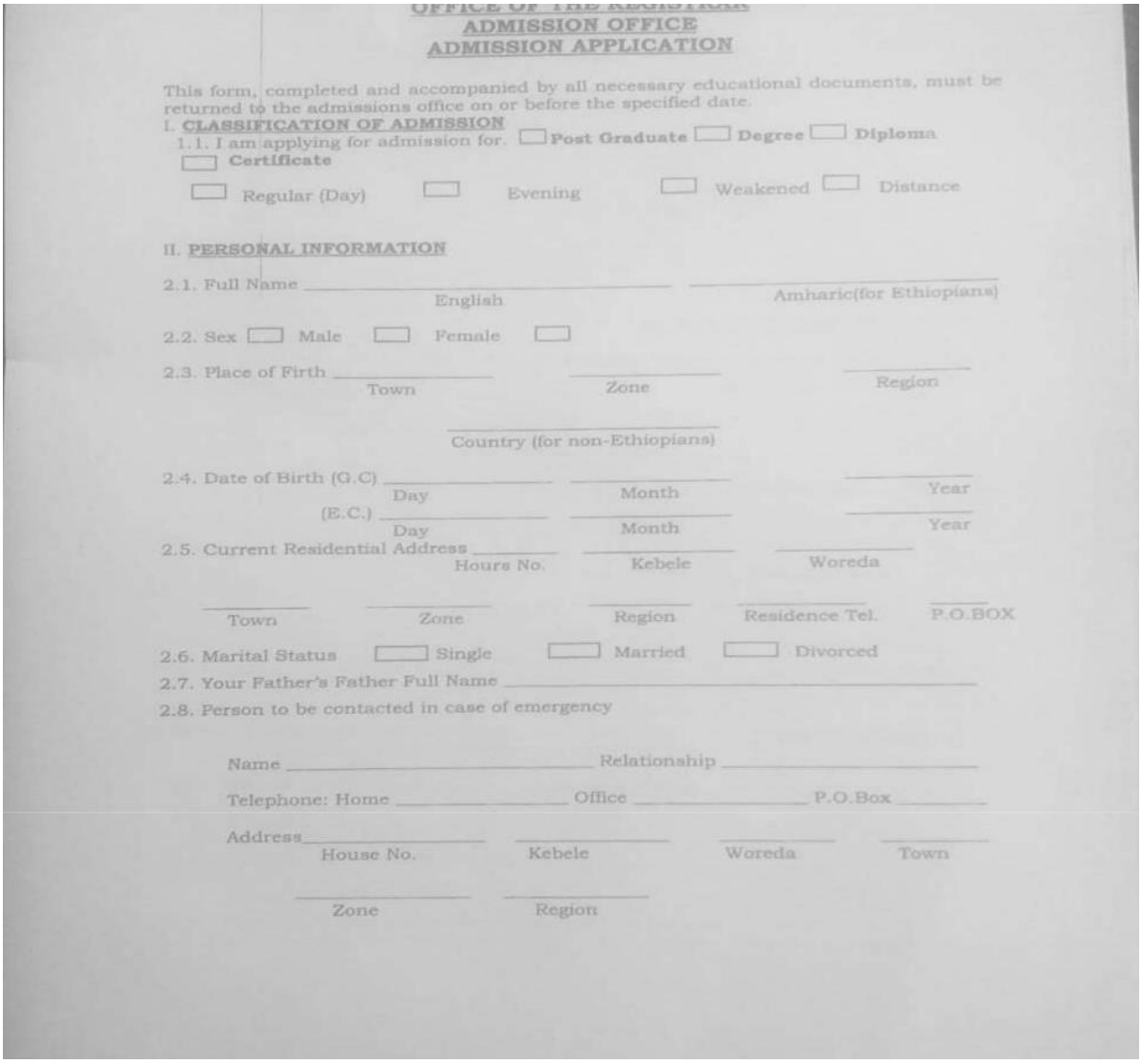
Departmental Functions:

Departments are involved in assigning instructors, addressing student queries, and approving grades, contributing to the overall academic management.

Verification of Academic Schedule:

The academic vice president verifies the academic schedule prepared by the registrar, ensuring coherence and compliance with the college's academic objectives.

2.4 Forms and Other Documents of the Existing System



OFFICE OF THE REGISTRAR

ADMISSION OFFICE

ADMISSION APPLICATION

This form, completed and accompanied by all necessary educational documents, must be returned to the admissions office on or before the specified date.

I. CLASSIFICATION OF ADMISSION

1.1. I am applying for admission for. ☐ Post Graduate ☐ Degree ☐ Diploma ☐ Certificate

☐ Regular (Day) ☐ Evening ☐ Weakened ☐ Distance

II. PERSONAL INFORMATION

2.1. Full Name _____ (English) _____ (Amharic(for Ethiopians))

2.2. Sex ☐ Male ☐ Female ☐

2.3. Place of Firth _____ (Town) _____ (Zone) _____ (Region)

_____ (Country (for non-Ethiopians))

2.4. Date of Birth (G.C) _____ (Day) _____ (Month) _____ (Year)

(E.C.) _____ (Day) _____ (Month) _____ (Year)

2.5. Current Residential Address _____ (Hours No.) _____ (Kebele) _____ (Woreda)

_____ (Town) _____ (Zone) _____ (Region) _____ (Residence Tel.) _____ (P.O.BOX)

2.6. Marital Status ☐ Single ☐ Married ☐ Divorced

2.7. Your Father's Father Full Name _____

2.8. Person to be contacted in case of emergency

Name _____ Relationship _____

Telephone: Home _____ Office _____ P.O.Box _____

Address _____ (House No.) _____ (Kebele) _____ (Woreda) _____ (Town)

_____ (Zone) _____ (Region)

2.5 Drawbacks of the Existing System:

- **Manual Module Distribution**:The reliance on instructors for the manual distribution of learning modules contributes to inefficiencies, potential errors, and increased workload on teaching staff.

- **Manual Tutorial Sessions**:The manual handling of tutorial sessions by instructors adds to the overall workload and may lead to variations in the quality and consistency of the teaching process.

- **Student Result Retrieval**:Students having to go back to instructors to view their course results introduces a time-consuming and cumbersome process, adversely affecting the overall student experience.

- **Registrar Office Workload**:The registration process, managed by the registrar's office, involves various tasks such as preparing grade reports, transcripts, and managing academic schedules. The manual nature of these tasks may result in delays and errors.

- **Academic Dean Control:**While the academic dean oversees the teaching-learning process, the centralized control may lead to bottlenecks and potential delays in decision-making.

- **Departmental Inefficiencies:**Departments assigning instructors, responding to student queries, and approving grades may face challenges related to manual processes, potentially impacting the efficiency of academic management.

- **Verification Process Complexit**y:The academic vice president's role in verifying the academic schedule prepared by the registrar may introduce complexities and potential errors, especially if the verification process is not streamlined.

- **Resource and Time Wastage:**The overall system exhibits a significant wastage of college resources and time, particularly during the preparation, duplication, and distribution of study modules. This not only impacts operational efficiency but also contributes to unnecessary costs.

- **Work Overload on College Employees**:The duplication and distribution of modules for each course, coupled with manual tasks, create an overburden on college employees, potentially affecting their productivity and job satisfaction.

- **Student Dissatisfaction:**The manual processes, coupled with the identified drawbacks, contribute to overall student dissatisfaction. The need for students to invest additional time, effort, and money in navigating these processes detracts from their overall educational experience.

2.6 Business Rule of the Existing System

BR1: The students must full fill minimal criteria of minister of Education to be registered or possess a competency certification of Level IV from COC Agency.

BR2: Registration date is starting from the announced day to 15 days continued and after the last days of registration the students will punish 50 birr in addition to normal registration fee to register.

BR3: The student monthly fee should be based on credit hour of the course, for a credit it is only 90 birr.

BR4: The total credit hour for the courses in a semester must be between 11 and 14.

BR5: The Assessment method for students is 50% Assignment and 50% final Exam.

BR6: The students to take the next course they must complete the pre-request course of the previous semester.

BR7: The students should reasonable and have evidence for their reason that why they cannot full fill their requirement.

BR8: All students must have their own identification card.

CHAPTER THREE

3. PROPOSED SYSTEM

The proposed system that we analyze can solve some portion of the existing system. When we see the solution, making the College automate system, it will solve most of the problems in the teaching-learning process. This project has much significance

- Reduce the extravagance of the college resources.
- Reduce the time and task required to perform the operation within the College.
- For students, better satisfaction of the speed provided by the instructor in course material distributing, seeing course result.
- And it improves the moral (motivation) of the users.

3.1 Functional Requirement

The functional requirements are features that needed to satisfy the users, the proposed system will be designed according to the roles of the users.

- **System Administrator:**

The system will allow the system administrator to perform the following tasks.

- Allow to create student and instructor account
- Allow to update student and instructor account
- Allow to delete student and instructor account
- Allow to view user's information
- Allow to generate user's password

- **Instructor:**

The system will allow the instructor to perform the following tasks.

- Allow to post progress quiz
- Allow to upload assignment
- Allow to upload module
- Allow to view assign course
- Allow to view student's information

- **Student:**

The system will allow the student to perform the following tasks.

- Allows to update her/his profile.
- Allow to download module
- Allow to submit assignment
- Allow to view grade report
- Allow to view his/her profile
- Allow to take progress quiz

❖ **Registrar Officer:**

- Allows to user management
- Allows to enrollment management
- Allows to academic records
- Allow to prepare grade report
- Allows to update grade
- Allows to generate ID
- Allows to communication

❖ **Academic Dean:**

- Allows to add department
- Allows to assign instructor
- Allows to delete department

❖ **Finance staff**:

- Allows to financial reporting
- Allows to control payment
- Allows to change password

❖ **Vice President:**

- Allows to view academic schedule
- Allows to view module
- Allows to view student

❖ **College Dean:**

- Allows to course scheduling
- Allows to view notification
- Allows to change password

❖ Department Head:

- Add course
- Approve grade report
- update notice
- Approve course result

3.2 Non-Functional Requirement

Non-functional requirements describe aspects of the system that are not directly related to the functional behavior of the system, only related to how the system will do the functional requirement. Non-functional requirements include a broad variety of requirements that apply to many different aspects of the system, from usability to performance.

3.2.1 Performance:

- The system should respond to user requests within 2 seconds to ensure a seamless and efficient user experience.
- It should be able to handle a concurrent user load of at least 1000 users without significant performance degradation.

3.2.2 User interface and human factors:

- Our proposed system will have an interactive interface and will provide a simple, attractive, flexible interface for every user of our system.
- Users of the system must also have legal practice experience and basic digital literacy.

3.2.3 Reliability:

- The system should have a 80.9% uptime, allowing for scheduled maintenance.
- It should be capable of recovering from failures within 5 minutes without data loss.

3.2.4 Scalability:

- The system should be scalable to accommodate a 20% growth in the number of users and courses over the next year.
- It should support an increasing amount of concurrent users as the user base expands.

3.2.5 Security:

- User data should be encrypted during transmission and storage.
- The system should have measures in place to prevent unauthorized access, including robust user authentication and access controls.

3.2.6 Usability:

- The user interface should be intuitive and user-friendly, requiring minimal training for users to navigate the system.
- It should be compatible with common web browsers and accessible to users with disabilities.

3.2.7 Compatibility:

- The system should be compatible with the latest versions of major web browsers (Chrome, Firefox, Safari, Edge).
- It should support various devices, including desktops, laptops, tablets, and smartphones.

3.2.8Availability:

- The system should be available 24/7, with planned downtime communicated to users in advance.
- In the event of unexpected downtime, a user-friendly maintenance page should be displayed.

3.2.9 Data Backup and Recovery:

- Regular automated backups of the system data should be performed daily, and backups should be stored securely offsite.
- The system should have a data recovery plan in place to restore data in case of data loss or corruption.

Documentation:

Our system will have well-defined documents which help to easily maintain the system. We will also prepare a short and understandable file for users on how to use the system. And the development process will be provided for the user to read to know about the process and what type of model used to develop the system

CHAPTER FOUR

4. SYSTEM ANALYSIS

In this chapter we will deal with the proposed system by using use case diagrams, use case descriptions, object model, dynamic model (sequence diagrams and activity diagrams). After identifying the actors and the use cases of our new system, the use cases are developed and textual descriptions are depicted based on the use case. Next the sequence diagram will be depicted based on the use cases which are developed for the newly proposed system. Activities will be represented by activity diagram. Precondition, post condition and flow of event will be covered under this chapter.

4.1. System Model

This section consists of the modeling of the proposed system using object oriented methodologies such as unified modeling language (UML). Here represent the proposed system by using different system models such as use case models, object models, dynamic models, that describe the problem to be solved and as system models represented by graphically they are more understandable than more detailed natural language description of the system requirement.

4.1.2 Use Case Model

A use-case model is a model of how different types of users interact with the system to solve a problem. A use-case model consists of a number of model elements. The most important model elements are: use cases, actors and the relationships between them. A use case diagram is a summary of who uses the system and what they can do with it. Use case diagram shows the relationships between users (actors) and use cases with in a system.

The following use cases have been identified from the system specification:

Table 4-1: Use case of the System

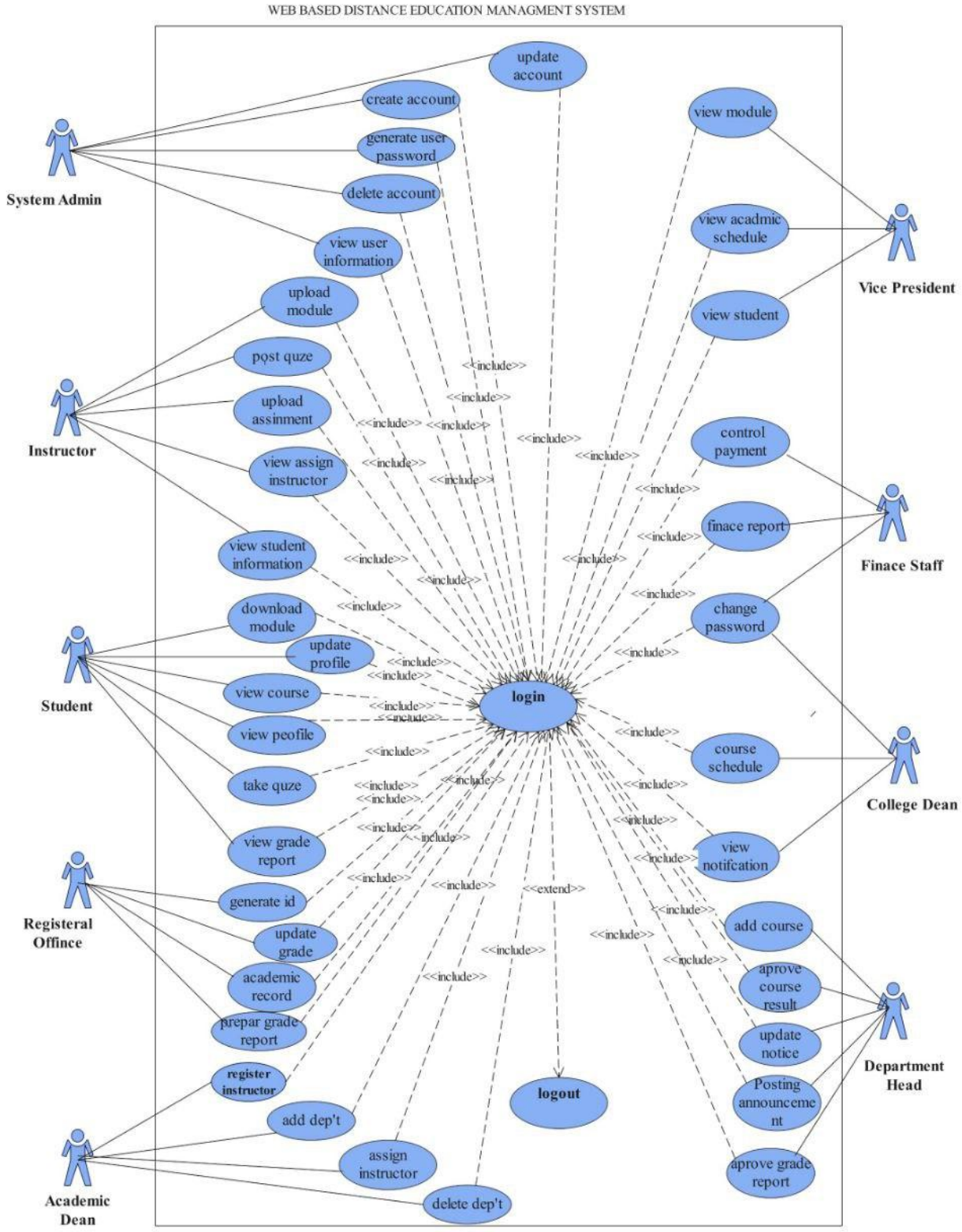
Download Files	View Academic sechedule
View Assign Instructor	Control payment

Post quiz	Assign instructor
Create and Delete Account	Create Account
Approve result	Post announcement
View user information	View report
Approve Grade Report	Assign course
View grade and announcement	View Module
Delete Account	Finance Report
Change Password	Generate User password
View,Update Profile	Add,Delete Dep't
Add Course	Login,Logout

Actors

In this system the identified actors that will be participating in the system are:

- Student
- Registrar
- Department head
- Instructor
- vice president
- System admin
- College dean
- Finance stuff
- Academic dean


WEB BASED DISTANCE EDUCATION MANAGMENT SYSTEM
System Admin
Instructor
Student
Registeral Offince
Academic Dean
Vice President
Finace Staff
College Dean
Department Head
update account
create account
generate user password
delete account
view user information
upload module
post quze
upload assinment
view assign instructor
view student information
download module
update profile
view course
view peofile
take quze
view grade report
generate id
update grade
academic record
prepar grade report
register instructor
add dep't
assign instructor
delete dep't
login
logout
view module
view acadmic schedule
view student
control payment
finace report
change password
course schedule
view notifcation
add course
aprove course result
update notice
Posting announcement
aprove grade report
<<include>>
<<extend>>

4.1.2. Use Case Description

Table : Use case Description for Login

Use case number	UC 01	
Use case name	Login	
Actor	Academic dean, instructor, students , registrar officer , system Administrator,College dean,Finance stuff,Department head And vice president	
Description	Checking the intended user is authorized or not	
Precondition	The user must have username and password	
Post condition	The users successfully login.	
Basic course of action	User action	System response
	1.The user opens the system 3. The user enters user name and password then click login button. 6. End use case.	 2. The system displays the login page. 4. The system checks the username and password. 5. The system opens the users' home page.
Alternative course of action	If the username and password is invalid, the system displays an error message, then go back to step 3 of basic course of action.	

Table : Use case documentation of logout

Use case number	UC 02	
Use case name	Logout	
Actor	Academic dean, instructor, students , registrar officer , system Administrator,College dean,Finance stuff,Department head And vice president	
Description	After doing any private activity in the system the user log out from the system.	
Precondition	The user should be in private page.	
Post condition	The user is in public page.	
Basic course of action	User action	System response
	1. The user clicks the	

	logout button. 3. End use case.	2. The system displays the login page.
Alternative course of action	If connection is fail, try again.	

Table : Use case documentation of download module

Use case number	UC 03	
Use case name	Down load module/learning material	
Actor	student	
Description	Downloading learning materials	
Precondition	The student must be login and the module must be uploaded before	
Post condition	The student successfully downloading learning materials	
Basic course of action	User action	System response
	1. The student clicks module link. 3. The student select and click “download” link. 5. End use case.	2. The system displays the module option. 4. The system down loads module
Alternative course of action	If there is no module uploaded before, the system display “there is no module in the system” message.	

Table : Use case documentation of view Result

Use case number	UC 05	
Use case name	View result	
Actor	Students.	
Description	The user looks result added by the instructor in the system.	
Precondition	The user must have user name and password and the result must be added before.	
Post condition	The user access and know the result.	
Basic course of action	User action	System response

	1. The user login to the system. 3. The user click on view result links. 5. the user view exam result 6. End use case.	2. The system displays the student home page. 4. The system displays available result that is added by the instructor.
Alternative course of action	If result is not added, the system displays the message “there is no result available”.	

Table :Use case documentation of view course

Use case number	UC 06	
Use case name	View course	
Actor	Students.	
Description	The user looks course added by the academic dean in the system.	
Precondition	The user must be login to the system and the course must be added before.	
Post condition	The user access and know the course.	
Basic course of action	User action	System response
	1. The user click on view course link. 3.The user view course 4. End use case.	2. The system displays available course that is added by the academic dean.
Alternative course of action	If course is not added, the system displays the message “there is no course available”.	

Table :Use case documentation of Register Instructor

Use case number	UC 07	
Use case name	Register instructors	
Actor	Academic dean	
Description	Registering instructor in the system	
Precondition	Academic dean must be login to the system.	
Post condition	The instructor successfully register in the system by academic dean	
Basic course of action	User action	System response
	1. The user clicks register link. 3.The user fill the form and click register button 5. End use case.	2. The system displays registration form. 4. The system registers instructor
Alternative course of action	If the user fill incorrect info, the system displays an error message, then go back to step 3 of basic course of action.	

Table :Use case documentation of Assign Instructor

Use case number	UC 08	
Use case name	Assign instructor	
Actor	Academic dean	
Description	assigning instructor to class in the system	
Precondition	Academic dean must have user name and password and instructor must register already	
Post condition	The instructor successfully assign in the system	
Basic course of action	User action	System response
	1.The Academic dean login to the system 3. The user clicks assign link. 5.The user select the instructor and click assign button 7. End use case.	2. The system displays the academic dean home page. 4. The system displays instructor list. 6. The system assign instructor
Alternative course of action	If user click assign link without select, the system display “no selection” message then return back to step 5 basic course of	

	action.

Table : Use case documentation of Add course

Use case number	UC 10	
Use case name	Add course	
Actor	Department Head	
Description	Adding course in the system	
Precondition	Academic dean must have user name and password	
Post condition	The course successfully added in the system by academic dean	
Basic course of action	User action	System response
	1.The academic dean login to the system 3. The user clicks add course link. 5. The user fills the form and click add button. 7. End use case.	2. The system displays the academic dean home page. 4. The system displays course registration form 6. The system adds course
Alternative course of action	If the user miss some information to fill, error message display and turn back to step 5 basic course of action	

Table :Use case documentation of Add student

Use case number	UC 12	
Use case name	Add student	
Actor	Registrar officer	
Description	Adding of student to the system they can pass entrance exam	
Precondition	Registrar officer must be login to the system and the student must pass the entrance exam	
Post condition	The student is added in to the system by registrar officer	
Basic course of action	User action	System response

	1. The user clicks add link. 3. The user fill student and click add button. 5. End use case.	2. The system displays the registration form. 4. The system adds the student
Alternative course of action	If the user misses some information to fill, error message display and turn back to step 3 basic course of action	

Table :Use case documentation of create account

Use case number	UC 13	
Use case name	Create account	
Actor	System Administrator	
Description	The system administrator creates user account to students, instructors, registrar officer and academic dean to give authorization.	
Precondition	Users registered to the system	
Post condition	Users account Created.	
	User Action	System Response
Basic course of action	1. The system Administrator login to the system 3.The system administrator click on create account link 5. System Administrator Fill create account form. 6. click on create button 8. End of use case.	2. The system display system administrator home page. 4. The system display creates account form. 7. The system displays create successful message.

Alternative course of action	If the system administrator fills incorrect information, the system displays error message, and go back to step 5 of basic course of action.

Table :Use case documentation of deactivate user account

Use case number	UC 14	
Use case name	Deactivate user account	
Actor	System administrator	
Description	The system administrator block accounts of the academic dean, students, registrar officer and instructors if there is any inconvenience.	
Precondition	The administrator must login to the system.	
Post condition	Protect user account from unauthorized access	
Basic course of action	User action	System response
	1.The system admin clicks manage account link 2. The admin clicks on view users account link 4. The system admin enters keywords on search box and clicks on search button 6. The system admin check on block account. 8. End use case.	 3. The system displays search box 5. The system displays the search result. 7. The system displays message.
Alternative course of action	If the search result is empty or if the user enters incorrect keyword in the search box the systems lets the user to try again and back to step 4 in basic course of action.	

Table :Use case documentation of activate user account

Use case number	UC 15	
Use case name	Activate user account	
Actor	System administrator	
Description	The system administrator unblocks or activate user accounts of the academic dean, students, registrar officer and instructors	
Precondition	The administrator must login to the system and the account of user deactivate before.	
Post condition	Successfully activate user account	
	User action	System response

Basic course of action	1.The system admin clicks manage account link 3. The system admin select deactivate user account and click activate button 5. End use case.	2.The system displays user account 4. The system activates user account
Alternative course of action	If connection is fail, try again.	

Table :Use case documentation of Add student result

Use case number	UC 16	
Use case name	Add student result	
Actor	Instructor	
Description	Add course result of the student in the system	
Precondition	The instructor must have user name and password	
Post condition	The instructor successfully add course result of the student in the system	
Basic course of action	User action	System response
	1.The instructor login to the system 3. The instructor clicks add result link. 5. The instructor fills the course information and fills each student result, then click “add” button. 7. End use case.	2. The system displays the instructor home page. 4. The system displays the result form. 6. The system add result to student
Alternative course of action	If user misses some information, system generates error message and back to step 5 to try again.	

Table :Use case documentation of upload learning material

Use case number	UC 17	
Use case name	upload learning material	
Actor	Instructor	
Description	uploading learning materials for the student	
Precondition	The instructor must have user name and password.	
Post condition	The instructor successfully uploading learning materials.	
Basic course of action	User action	System response

	1.The instructor login to the system 3. The user clicks module upload link. 5. The user fills the information of learning material and click “upload” button. 7. End use case.	2. The system displays the instructor home page. 4. The system displays the form. 6. The system up loads modules and other learning materials
Alternative course of action	If user miss some information about learning materials, the system displays error messages and back to step 5 to try again	

Table : Use case documentation of add department

Use case number	UC 18	
Use case name	Add department	
Actor	Academic dean	
Description	Adding department in the system	
Precondition	Academic dean must have user name and password	
Post condition	The department successfully added in the system by academic dean	
Basic course of action	User action	System response
	1.The academic dean login to the system 3. The user clicks add department link. 5. The user fills the form and click add button. 7. End use case.	2. The system displays the academic dean home page. 4. The system displays department registration form 6. The system adds department
Alternative course of action	If the user miss some information to fill, error message display and turn back to step 5 basic course of action	

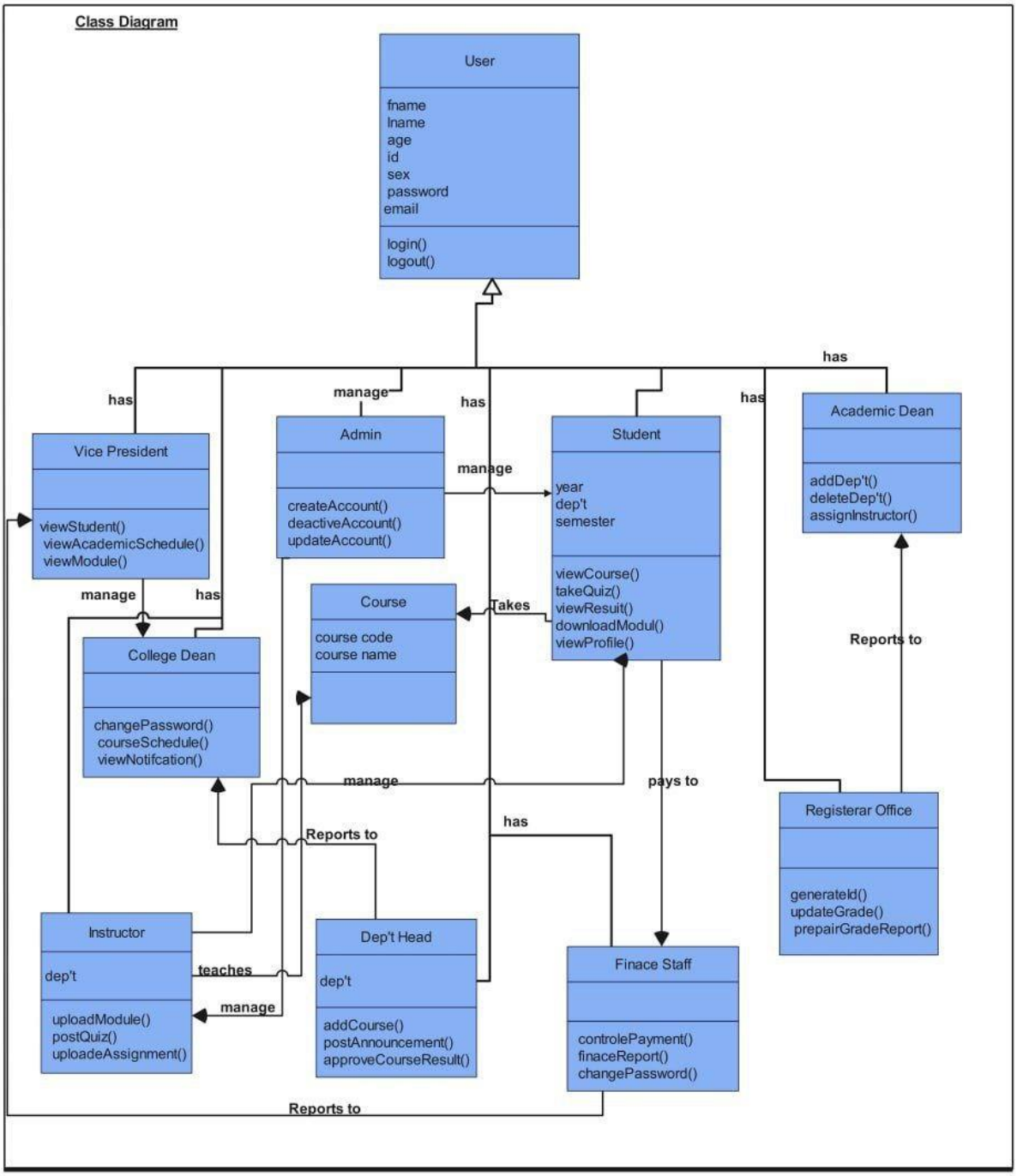
4.2. Object Model

Object Model is a collection of objects or classes through which a program can examine and manipulate some specific parts of its world. In other words, the object-oriented interface to some service or system. Such an interface is said to be the object model of the represented service or system.

4.2.1. Class Diagram

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 constructing executable code of the software application. This class diagram also describes the attributes and operations of a class and also the constraints imposed in the system. The class diagram is 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. We are developing object oriented base system, so we are going to use class diagram to model the static view of our system

Figure :Class Diagram



4.2.2 Data Dictionary

Data Dictionary is a collection of names, definitions, and attributes about data elements that are being used or captured in a database, information system, or part of a research project. It

describes the meanings and purposes of data elements within the context of a project, and provides guidance on

interpretation, accepted meanings and representation.

Attributes	Caption	Example	Data type	Constraints
user_ID	User Identifier	NSR/0753/13	varchar(100)	Primary key
firstName	First Name	Endalew	varchar(100)	Not null
lastName	Last Name	Shumet	varchar(100)	Not null
gender	Gender	Male	varchar(100)	Not null
email	Email	Endalews92@gmail.con	varchar(100)	Not null
address	Address	Wolkite	varchar(100)	Not null
password	Password	Endex@123	varchar(100)	Not null

4.3 Dynamic Model

Dynamic models are generally models that contain or depend upon an element of time, especially allowing for interactions between variables over time. A separate idea with the same name is models that are updated over time with new data.

4.3.1. Sequence Diagram

Sequence diagrams are used to depict graphically how objects interact with each other via messages in the execution of a use case or operation. They illustrate how the operations are performed between objects and in what sequence. A virtual course delivery system sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence chart, this sequence diagram shows object inter actions arranged in time sequence

Figure :Sequence diagram login

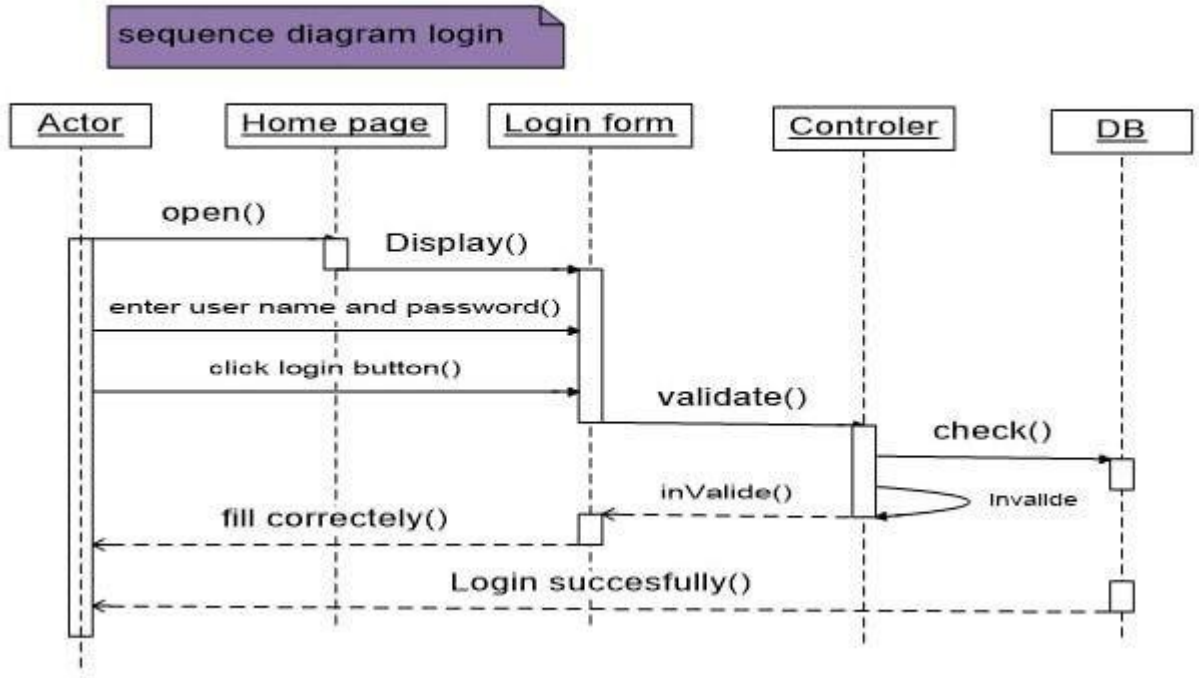


Figure : Sequence diagram logout

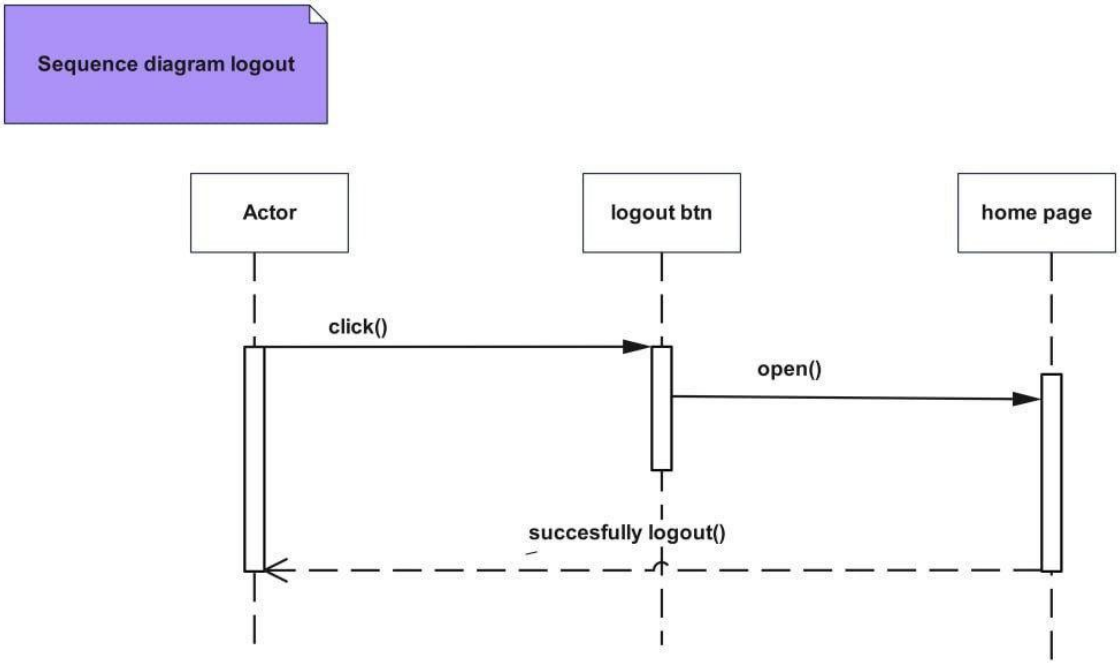


Figure : Sequence diagram download

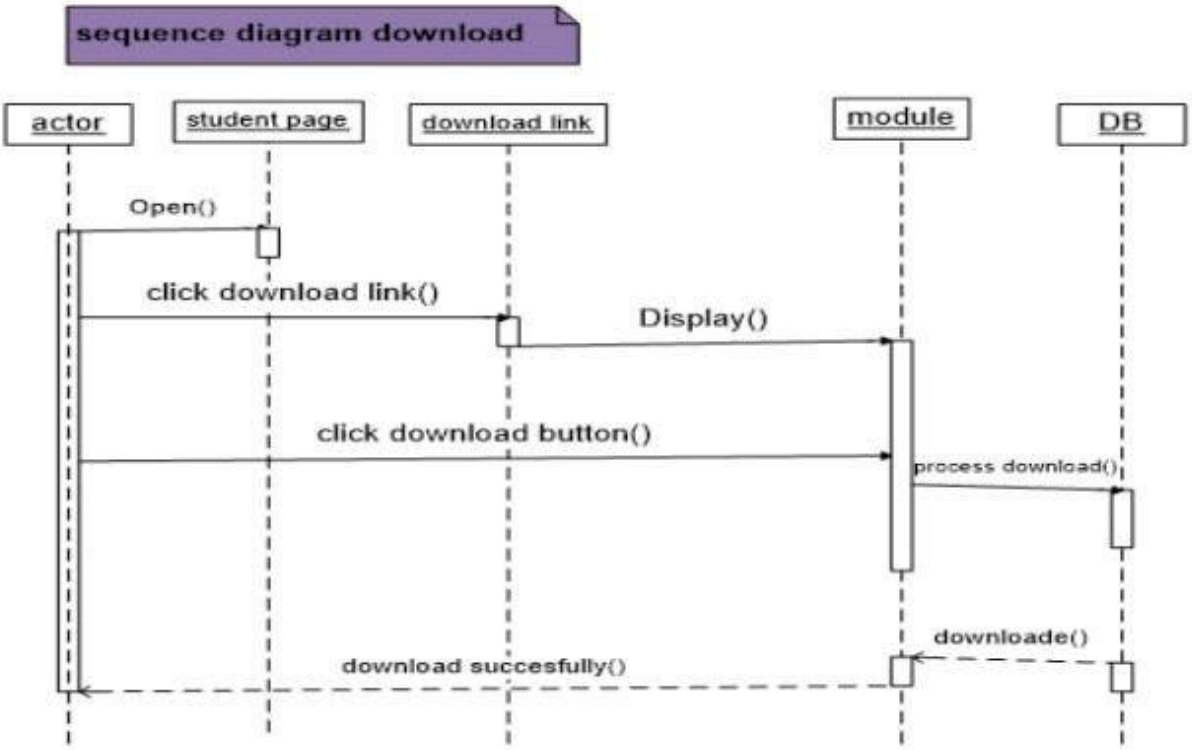


Figure :Sequence diagram create account

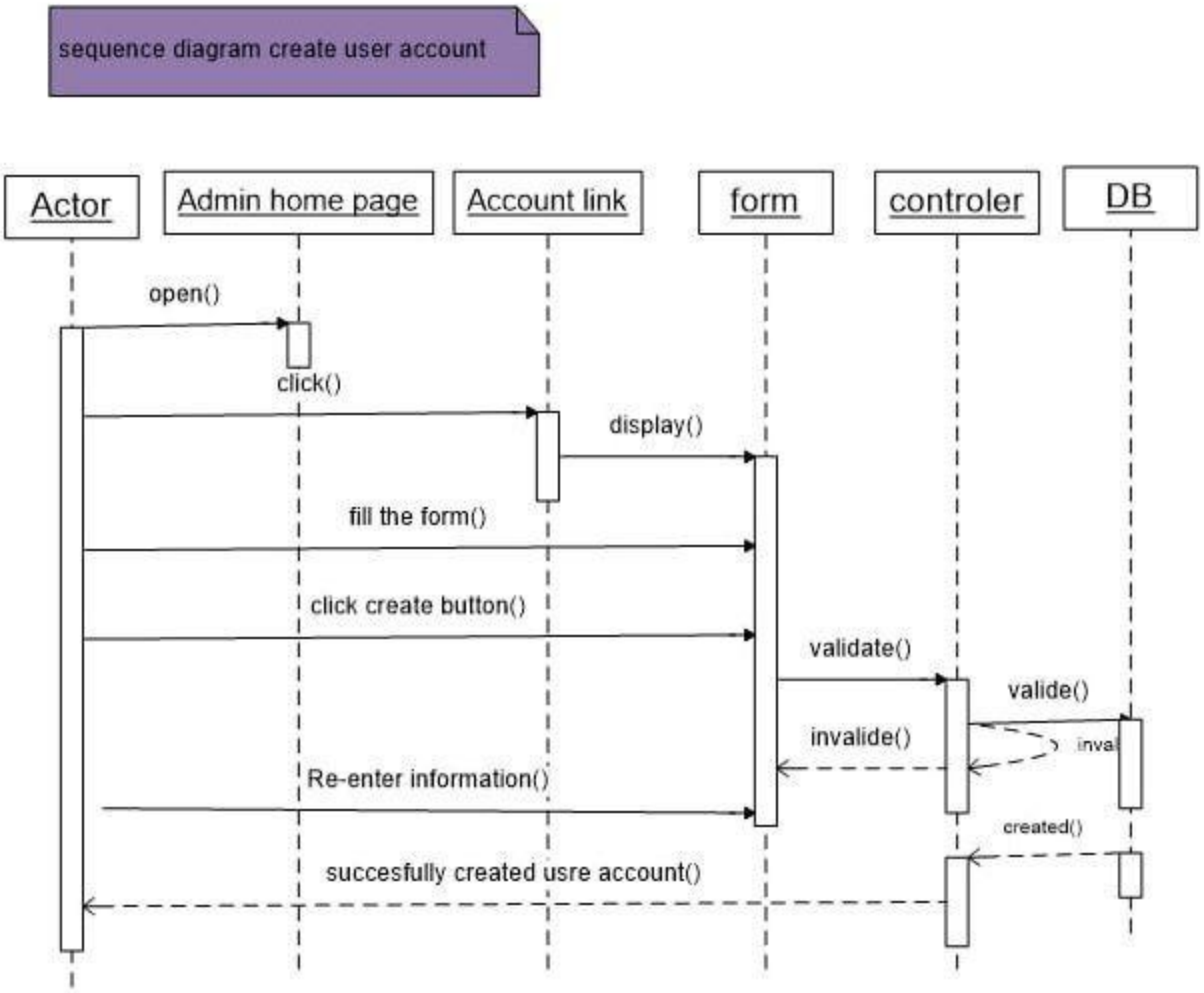

sequence diagram create user account
Actor
Admin home page
Account link
form
controler
DB
open()
click()
display()
fill the form()
click create button()
validate()
valide()
invalide()
Re-enter information()
created()
succesfully created usre account()

Figure :Sequence diagram upload learning material

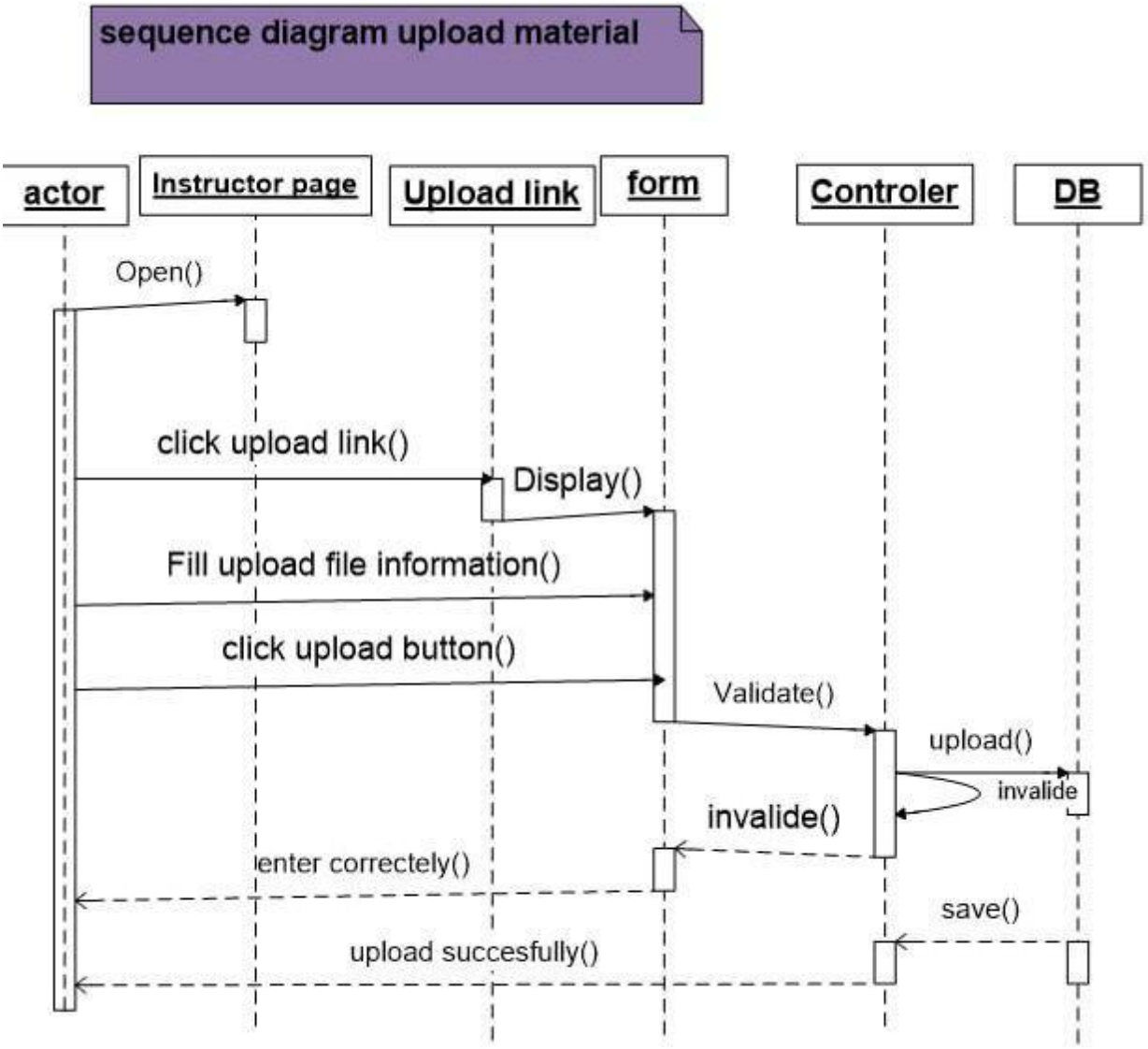


Figure :Sequence diagram add dept.

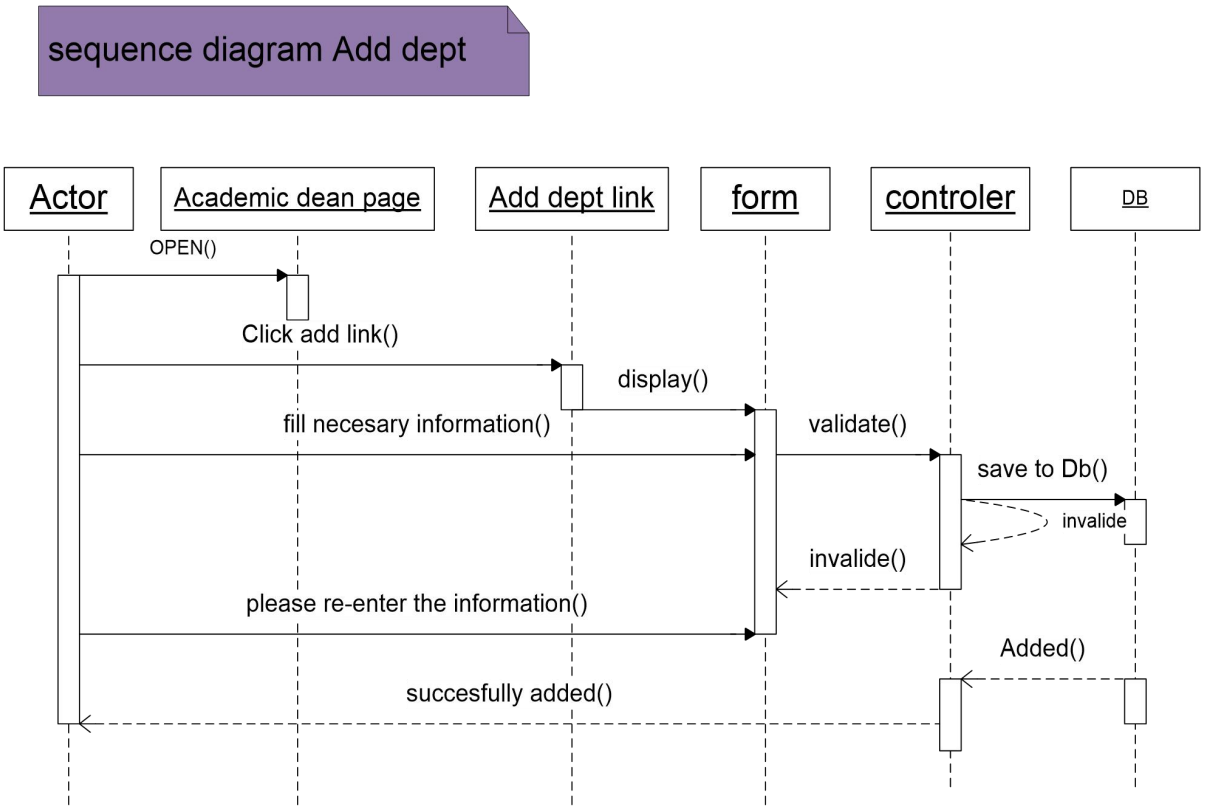


Figure :Sequence diagram add course

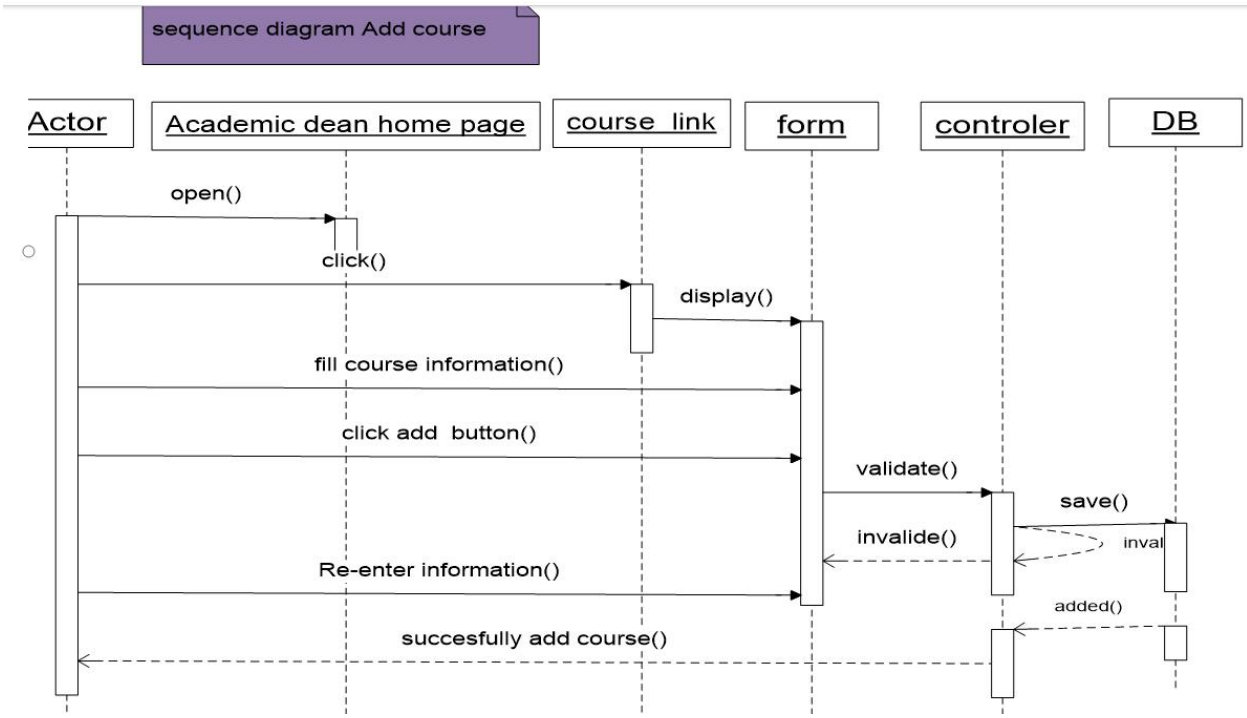


Figure :Sequence diagram register Instructor

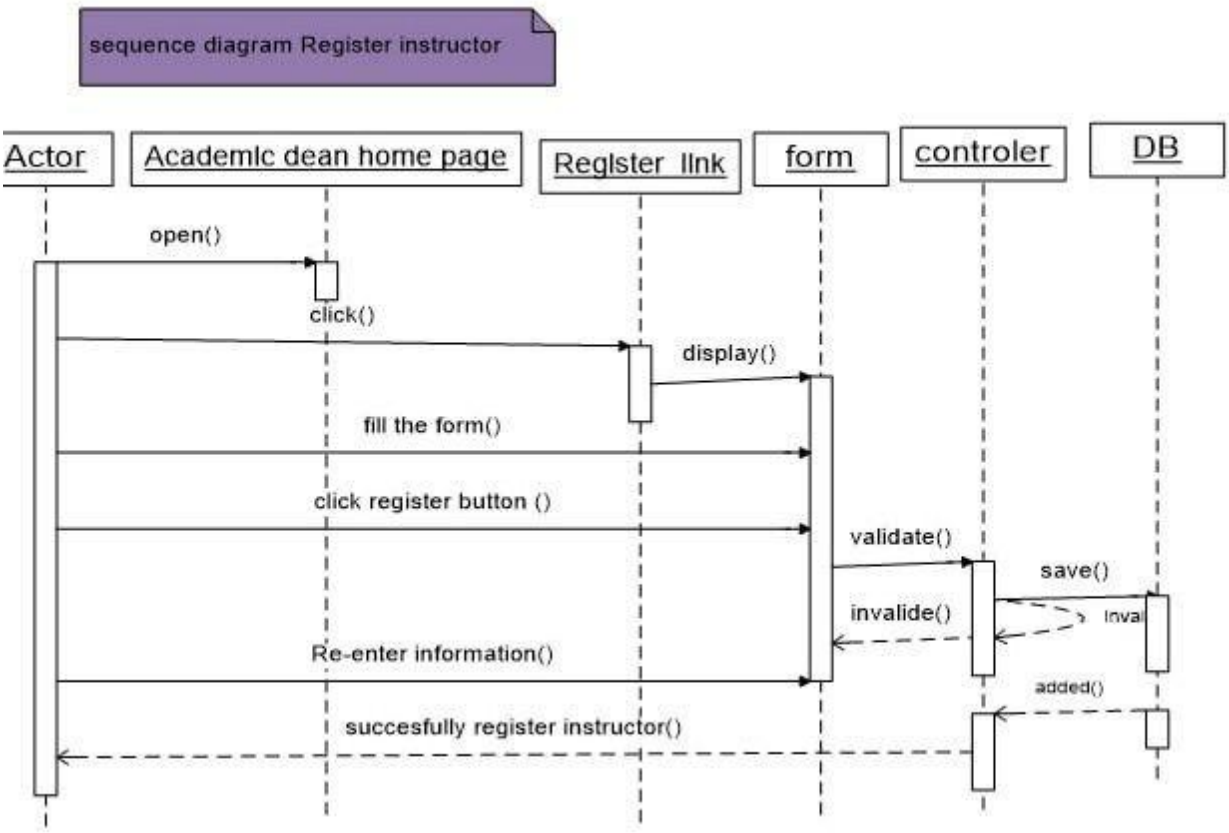


Figure : Sequence diagram Assign Instructor

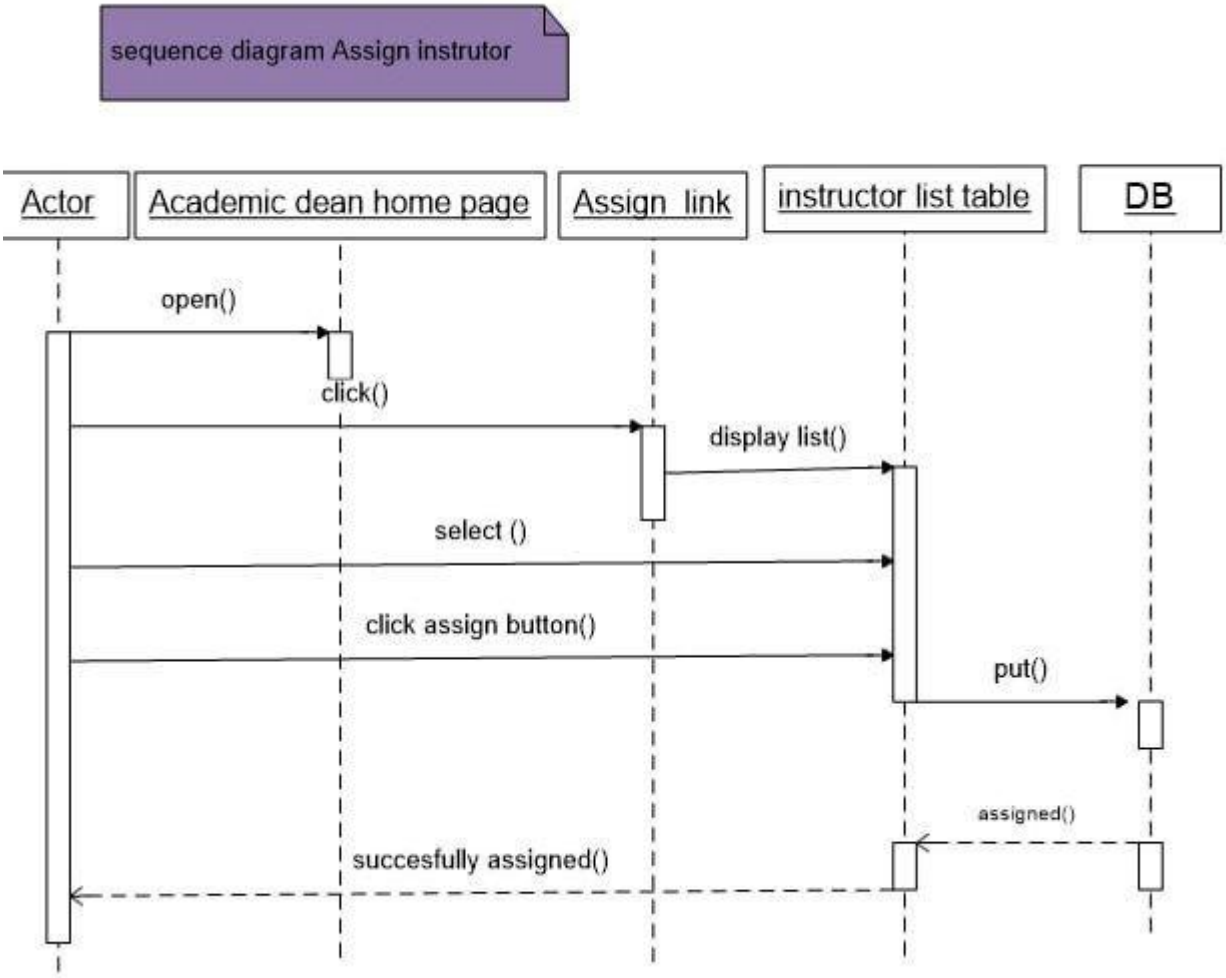


Figure :Sequence diagram view notice

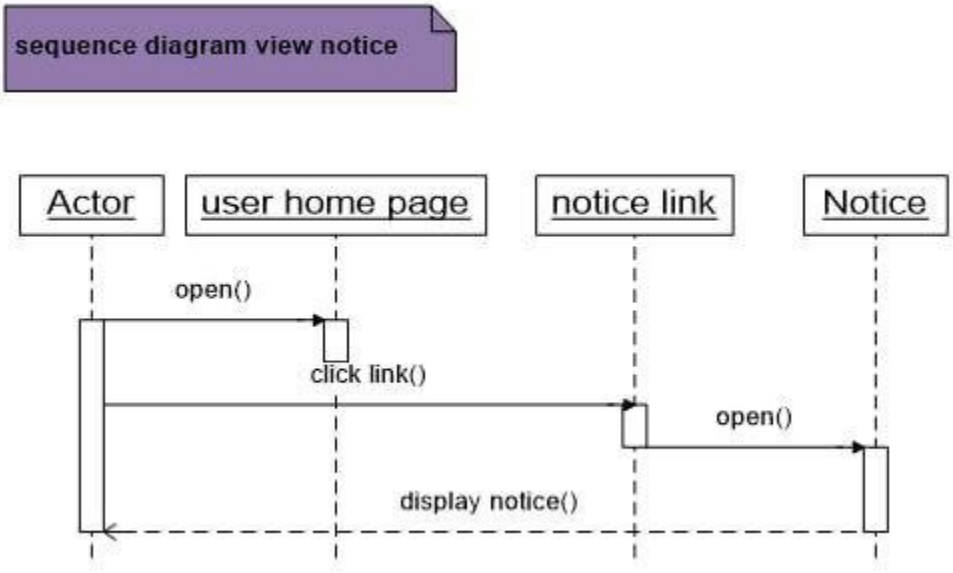
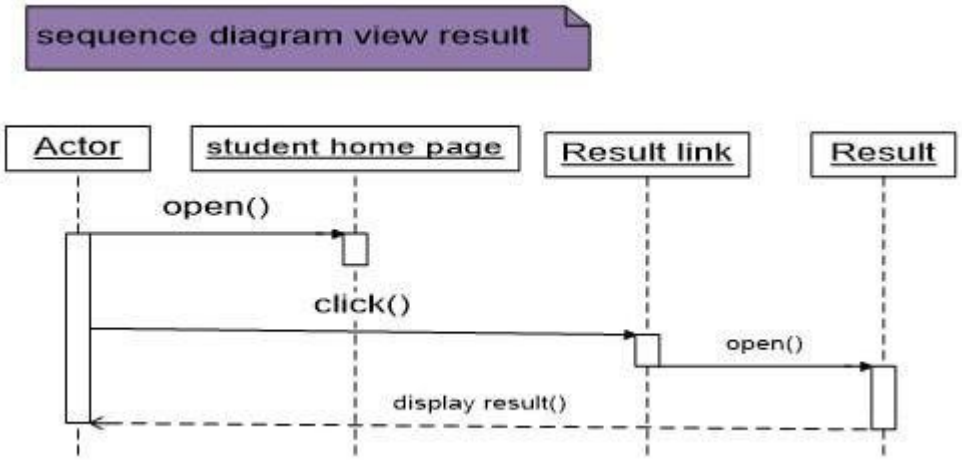


Figure : Sequence diagram view result



4.3.2. Activity Diagrams

An activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all types of flow control by using different elements such as forks, joins, etc.

The basic purposes of activity diagrams are to capture the dynamic behavior of the system.

Activity diagrams are used to show message flow from one activity to another.

Figure : Activity Diagram login

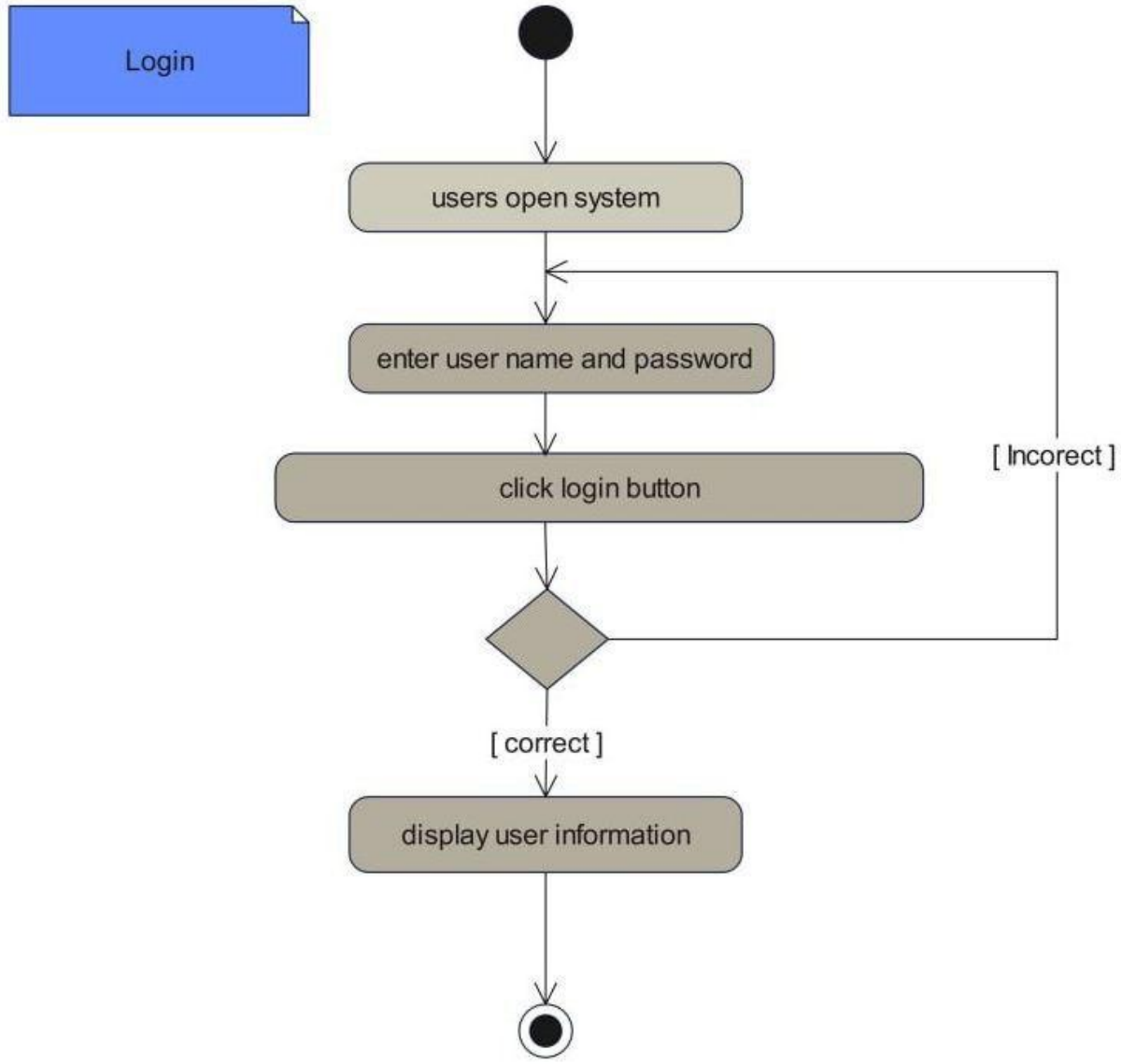


Figure :Activity Diagram logout

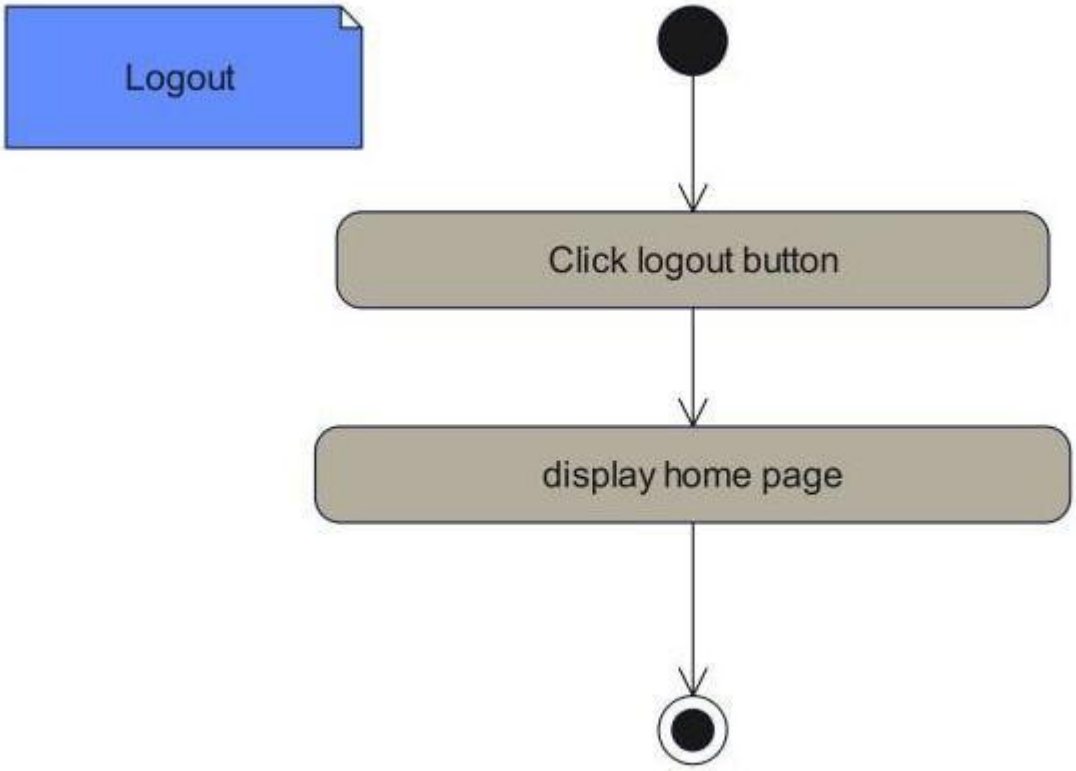


Figure : Activity Diagram add student

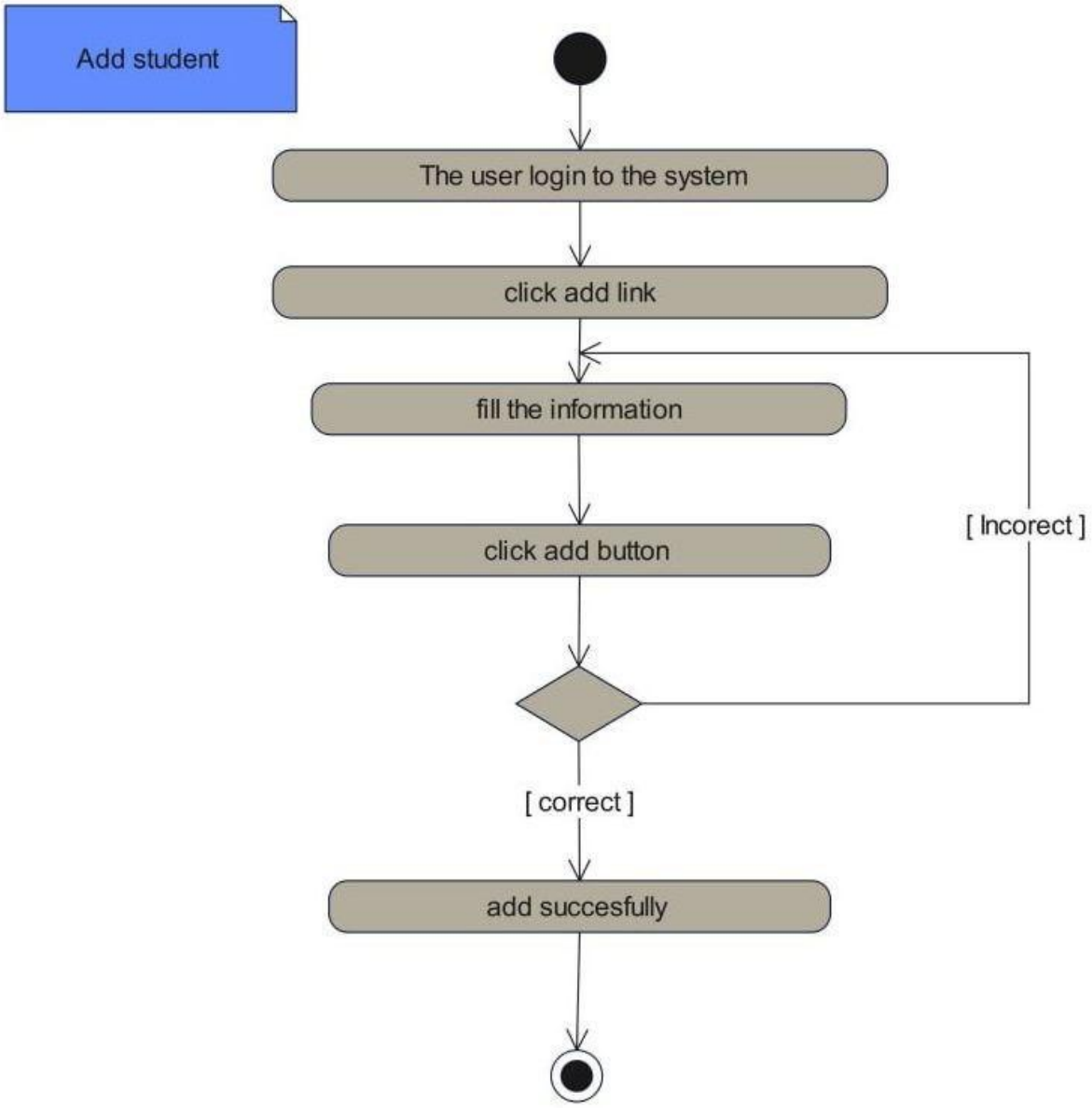


Figure: Activity Diagram upload matrial

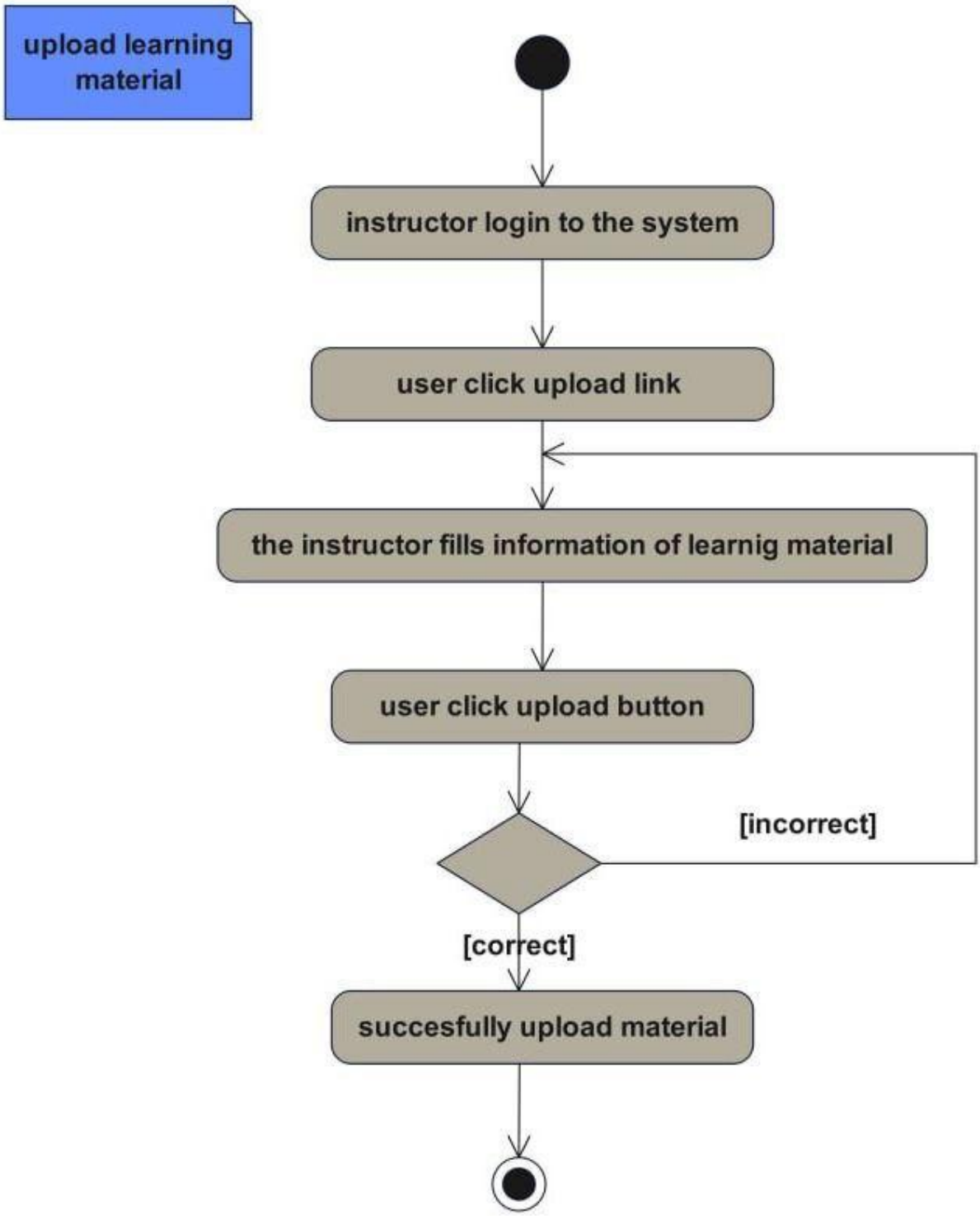

upload learning material
instructor login to the system
user click upload link
the instructor fills information of learnig material
user click upload button
[incorrect]
[correct]
succesfully upload material

Figure : Activity Diagram download module

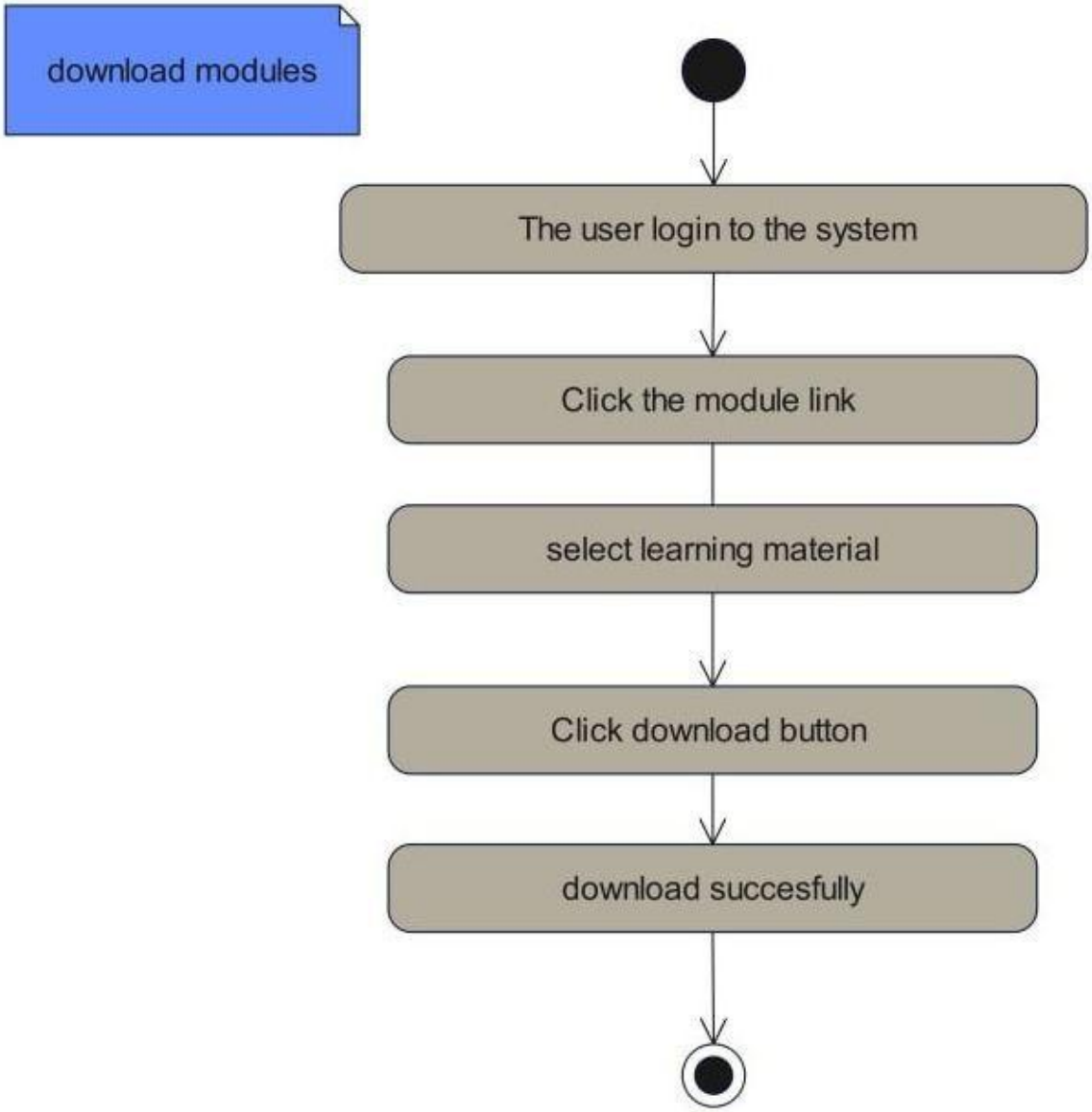


Figure : Activity Diagram view notice

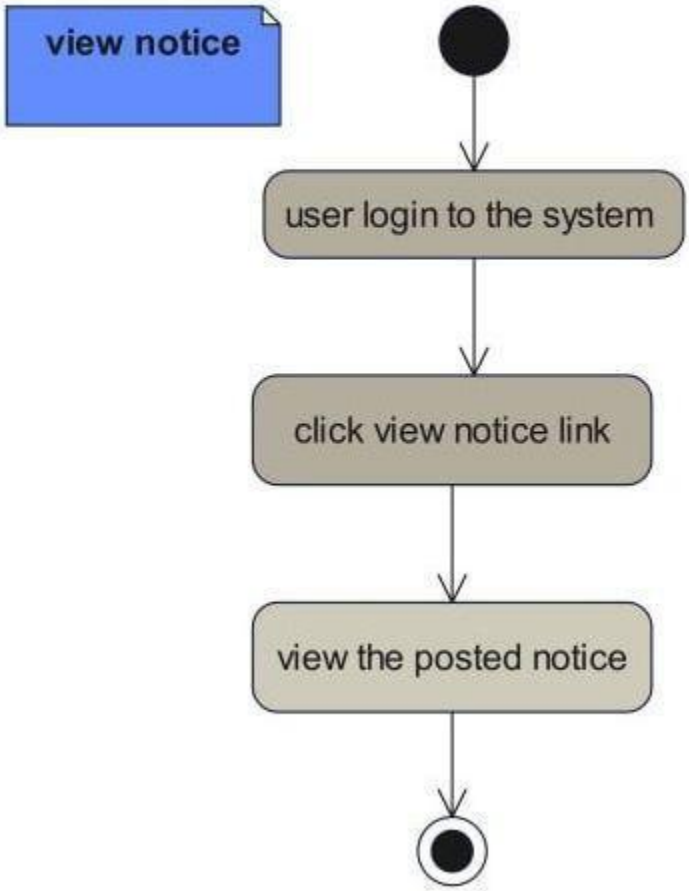


Figure : Activity Diagram view course result

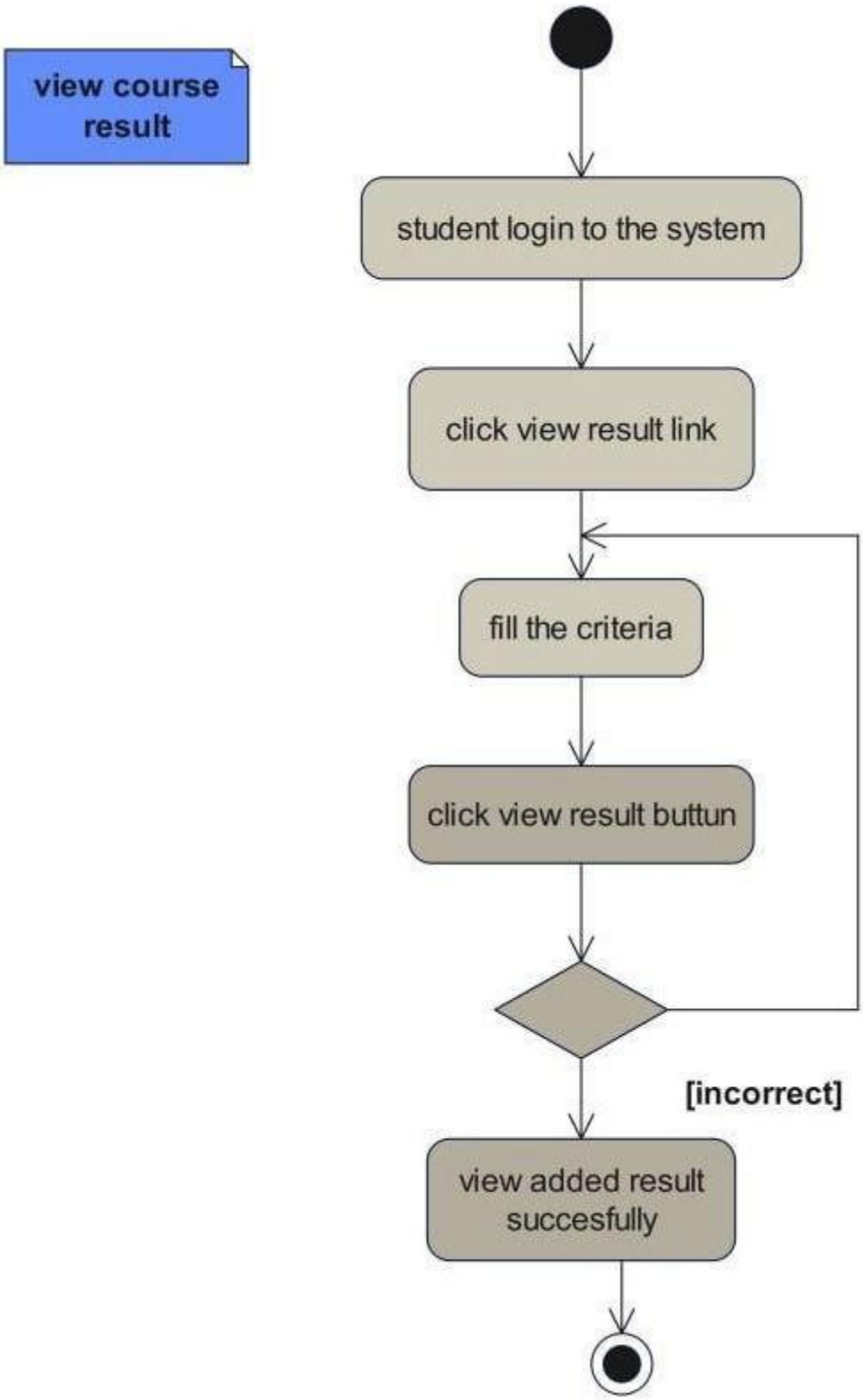

view course result
student login to the system
click view result link
fill the criteria
click view result buttun
[incorrect]
view added result succesfully

Figure :Activity Diagram view course

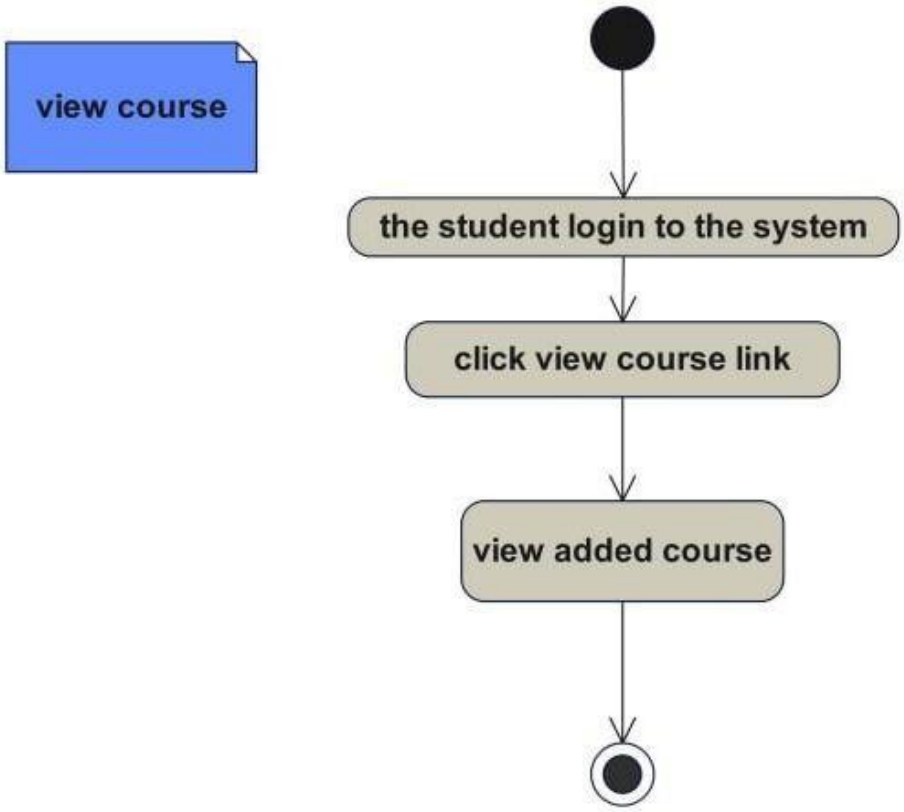


Figure : Activity Diagram Assign Instructor

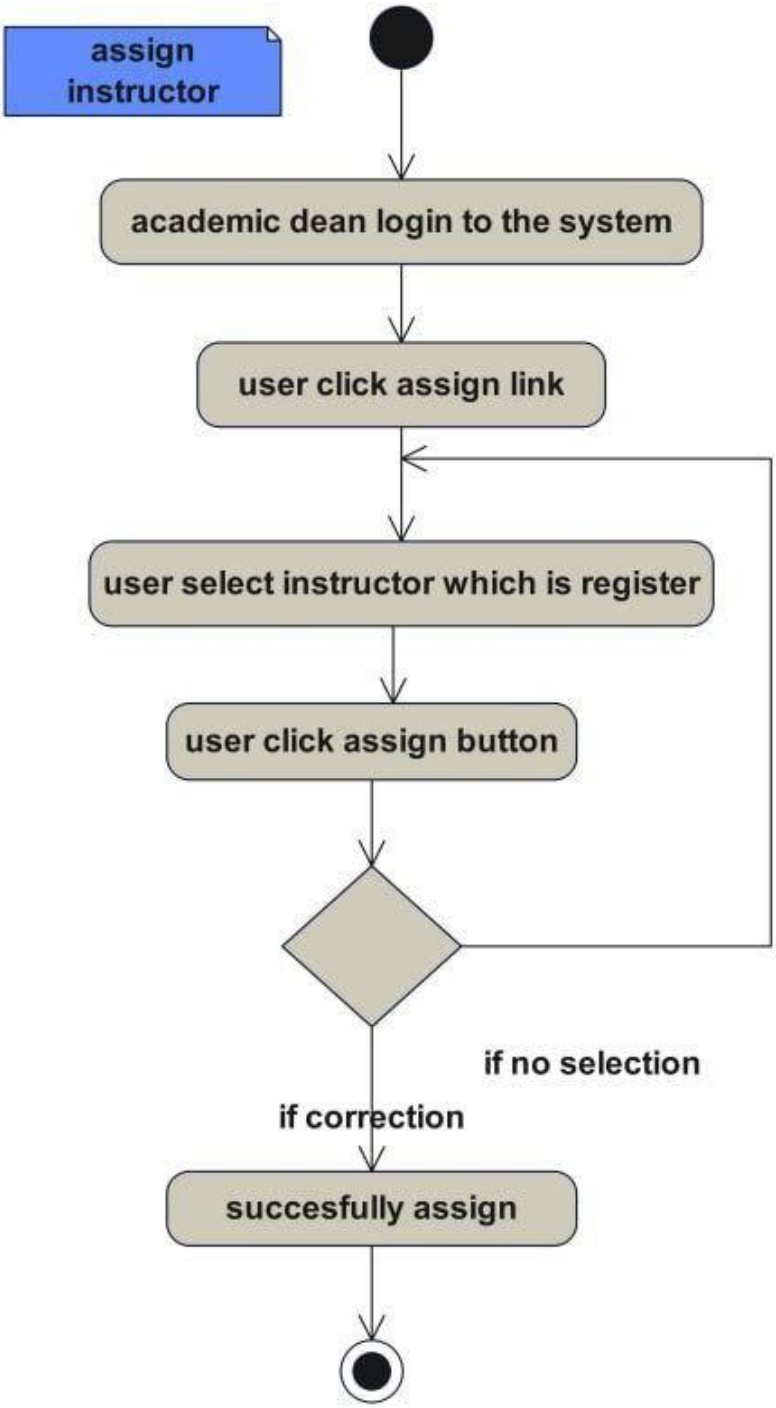


Figure : Activity Diagram deactive account

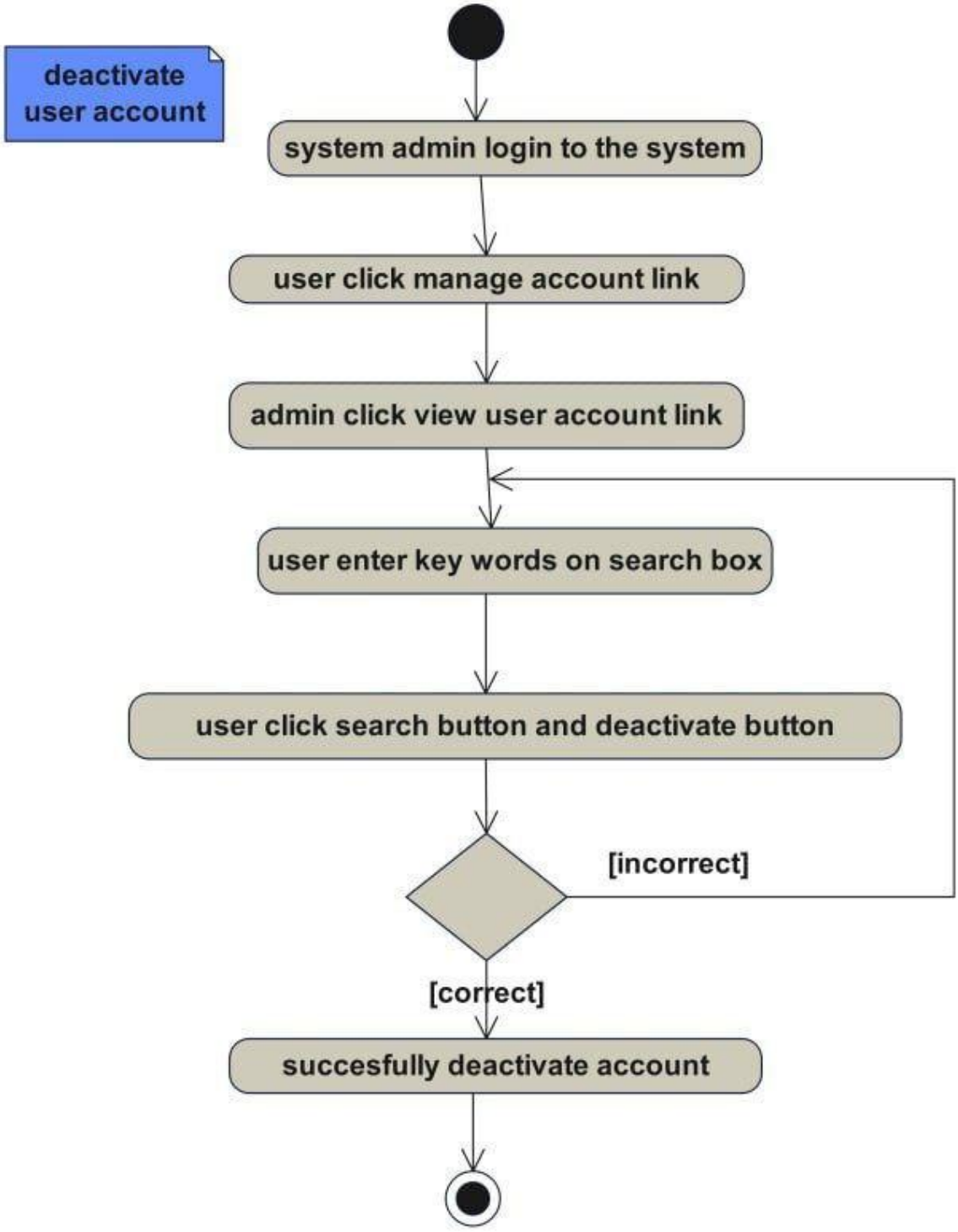


Figure : Activity Diagram create account

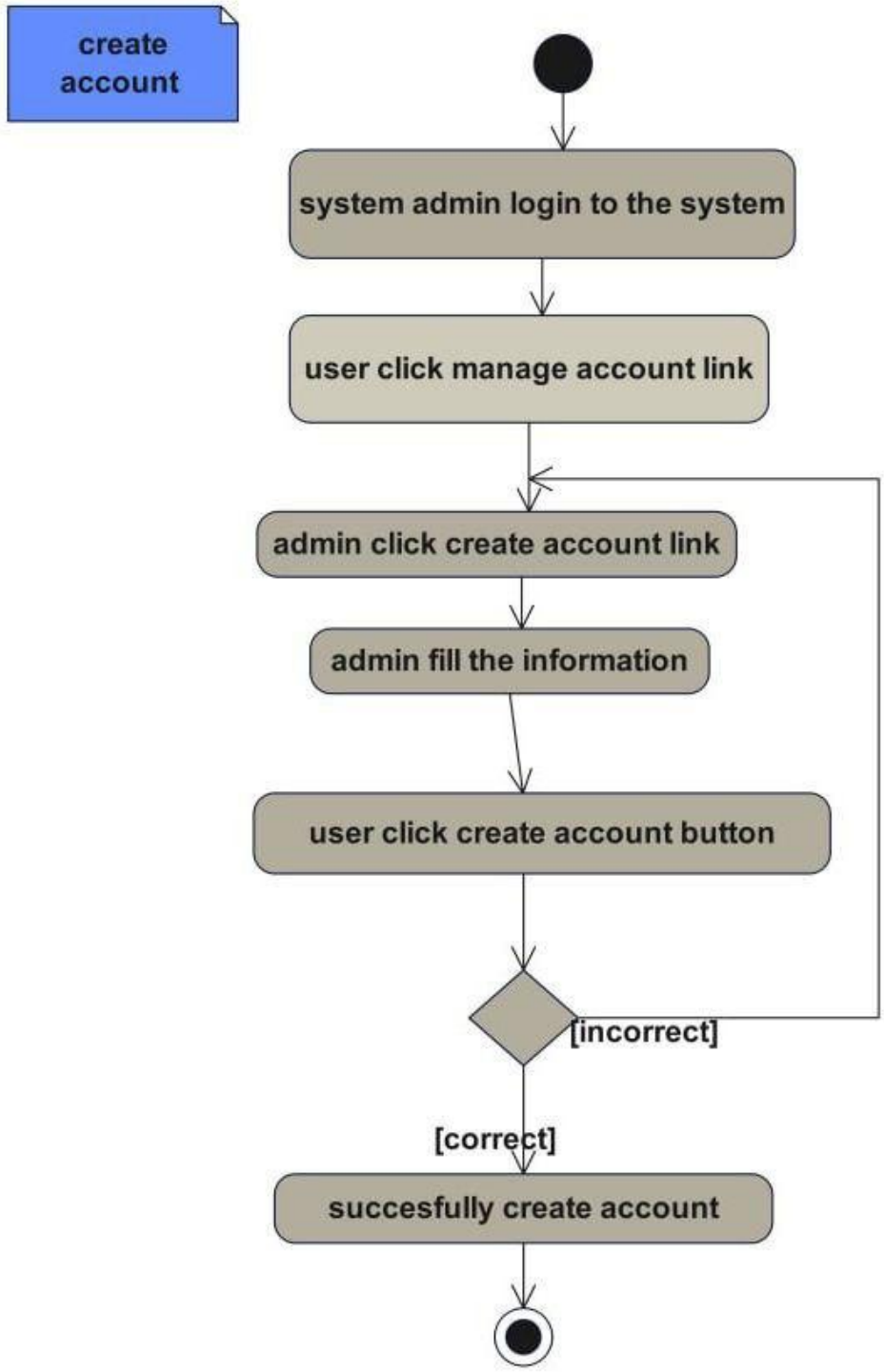


Figure : Activity Diagram add course

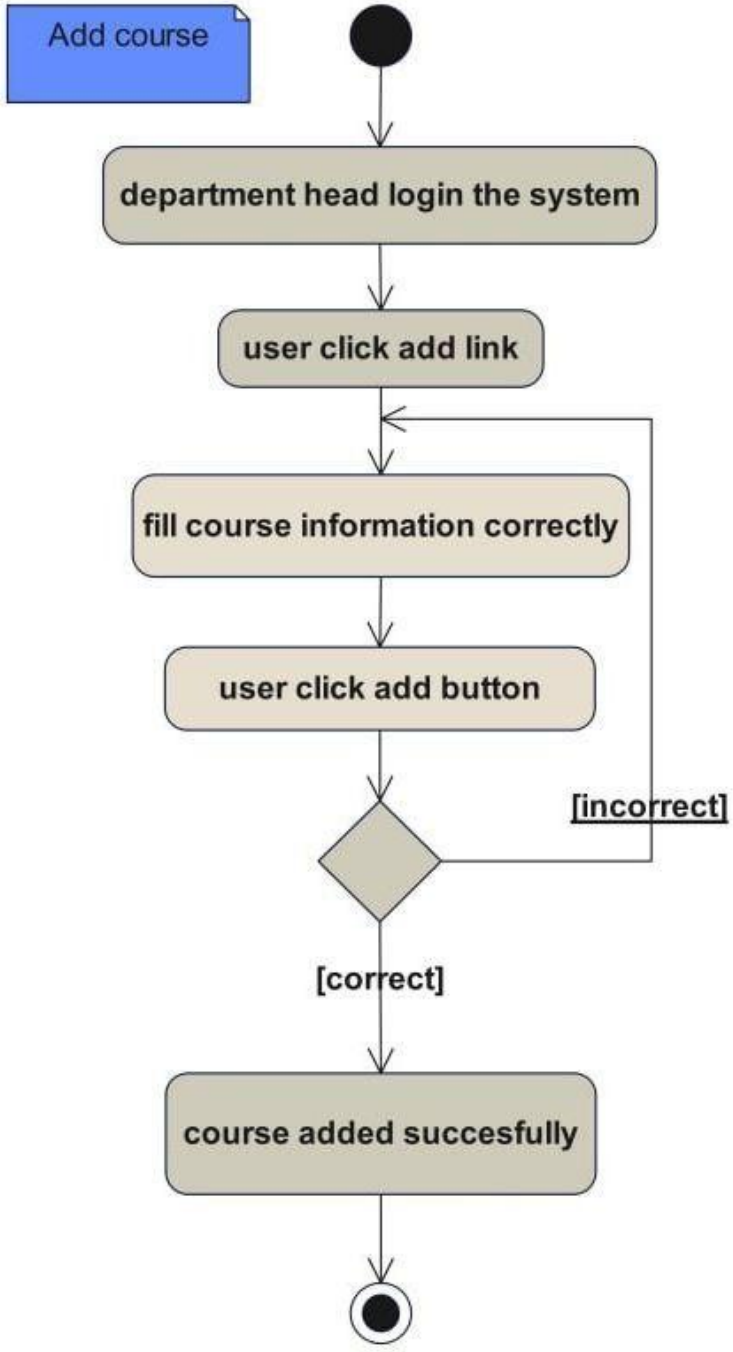
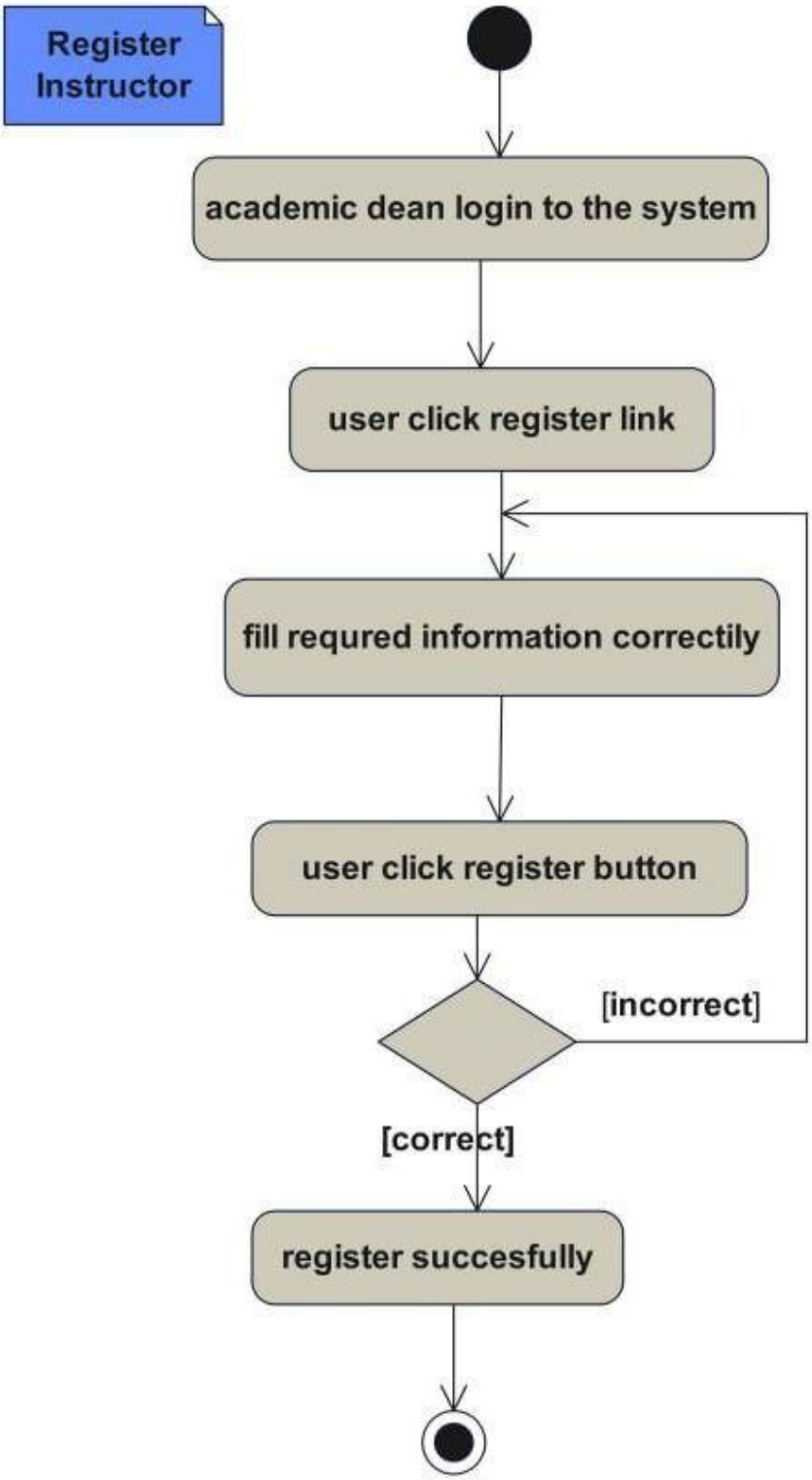


Figure : Activity Diagram Register Instructor


Register Instructor
academic dean login to the system
user click register link
fill requred information correctily
user click register button
[incorrect]
[correct]
register succesfully

4.3.3 State Chart Diagram

state chart diagram defines different states of an object during its lifetime. These states can change based on events that trigger the object. The diagram is particularly useful for modeling reactive systems, which respond to internal or external events.

state chart diagram represent the condition of an object when a specific event occurs. The main purpose of a state chart diagram is to model an object's entire lifespan, from creation to termination.

The Following are the main purposes of using State chart diagrams

- To model the dynamic aspect of a system.
- To model the life time of a reactive system.
- To describe different states of an object during its life time.
- Define a state machine to model the states of an object.

Figure: State chart diagram for login

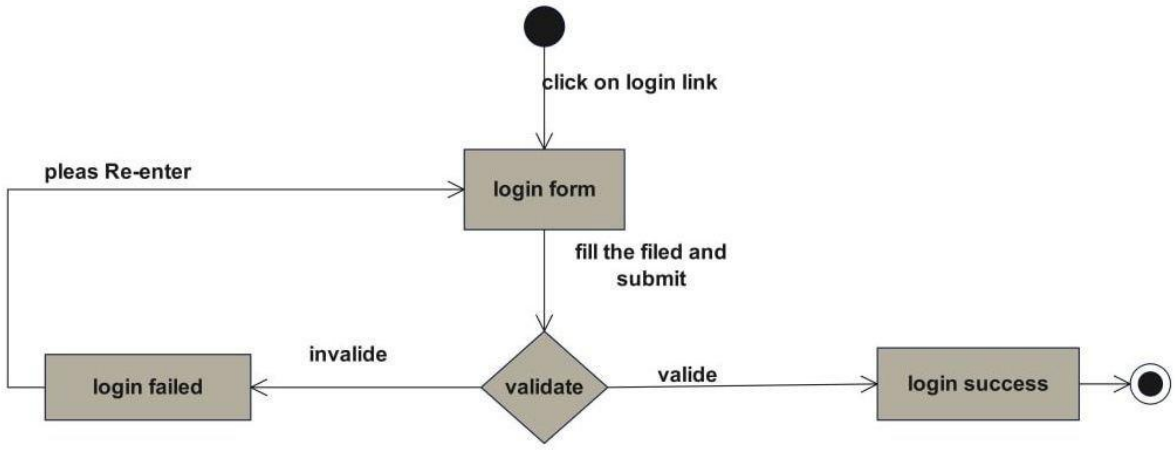
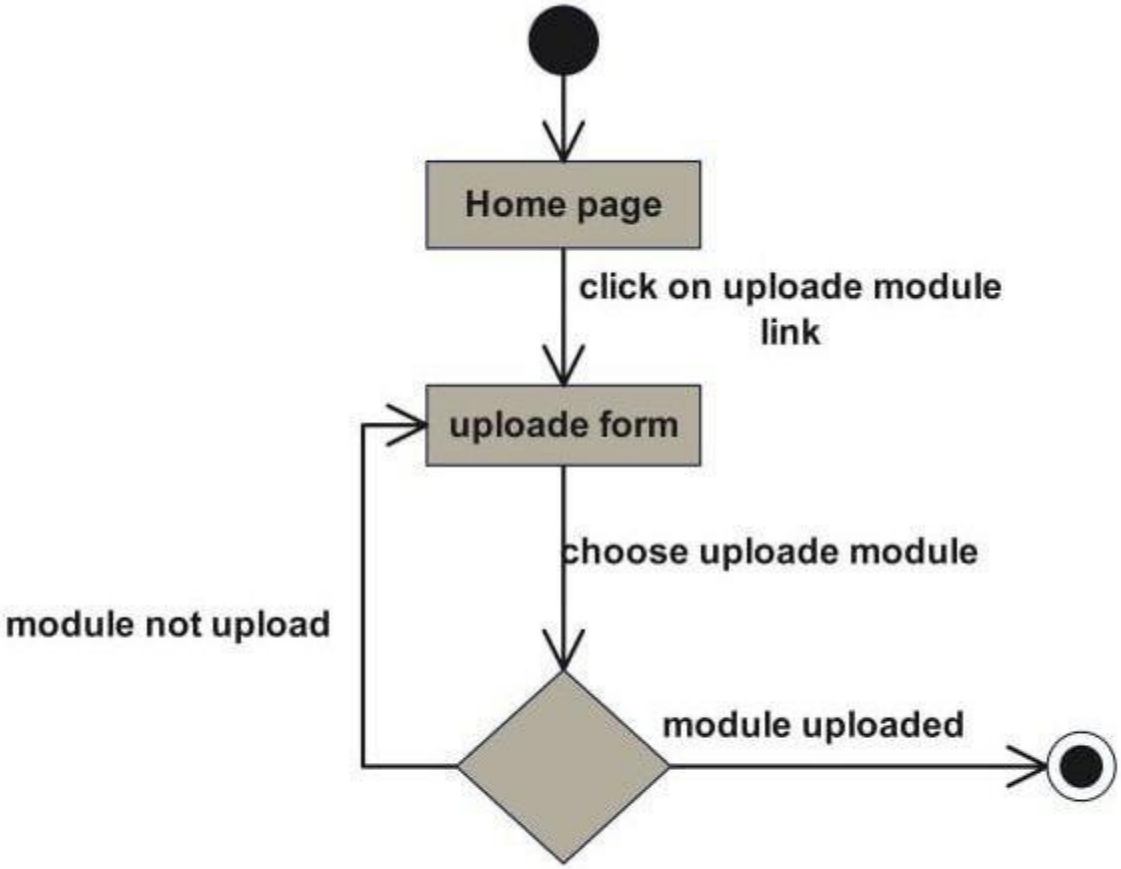


Figure: State chart diagram for upload module


Home page
click on uploade module link
uploade form
choose uploade module
module not upload
module uploaded

CHAPTER Five

5. SYSTEM DESIGN

We have identified the functional and non-functional requirements of the system and produced an analysis model in the previous chapter. Now we will discuss in this chapter: The design goal of our system, System architecture, System decomposition, deployment, etc. This project is designed in a manner that solves the problems of Atlas College by minimizing the workload that appears on the employees, especially regarding sharing files and managing users, and all the problems we identified in the system analysis phase of this project.

In this phase, our group project will try to illustrate about:

- Design goal of the system
- Proposed system architecture
- Subsystem Decomposition and Description
- Persistent Data Management
- Hardware/Software Mapping
- Detailed Class Diagram
- Algorithm design and
- User Interface design

5.1 Design Goals

The design goals are derived from non-functional requirements, which means non-functional requirements are the description of the feature characteristics and attributes of the system, as well as any constraints that may limit the boundary of the proposed solution. The objectives of design are to model the system with high quality. They describe the qualities of the system. These goals consider the following criteria.

5.2 Current System Architecture

The current system is not designed and there is no software architecture for the current software

of this project is not existed and the users are using the manual way to get the information.

5.2.1 Performance Consideration

The system will have good performance as much as possible, this will be attained via easily loadable information and fast response time in searching, updating, and viewing information because we will use good algorithms and minimize lines of code to perform specific tasks (functions).

- Load Time: The system will be loading information within a second.
- Throughput: Depending on the performance of our computers that run the system, the system performs many operations at a given time.
- Response Time: This system gives a response to the user according to the delivery of messages and notifications, reaching within a second. The system will support multiple users at a time, and it works very well with short response times.

5.2.2 Dependability

This system should achieve the following dependability characteristics in order to resist a crash and be available and reliable:

- Security: This system is secured by preventing unauthorized users from accessing the database system.
- Reliability: This system is reliable by providing the correct information and giving a response correctly and accurately for every query from authorized users.

5.2.3 Maintenance

The system should be maintainable if there is a failure in a system or when the system needs modification. That means we will develop this system by dividing the system into different modules which are loosely coupled and highly cohesive. If the system needs maintenance, it should meet the following maintenance criteria:

- Extensibility: If it is needed to add new functionality to the system, this must be achieved by

only making a separate page and integrating this page with the existing system.

- Modifiability: If the system needs some functionality modification, the modification is done to that function (the function to be modified) or page (the page to be modified) without affecting the overall system organization.

5.2.4 End User

The user interface of the system should prevent users from issuing commands in the wrong order. That means whenever users of the system want to insert unnecessary commands toward the next function, the user interface prevents them from performing such actions.

5.3 Current System Architecture

The current system is not designed and there is no software architecture for the current software of this project is not existed and the users are using the manual way to get the information.

5.4 Proposed System Architecture

This proposed system is consisting of three-tier architecture namely presentation layer, business logic layer, and data layer. The presentation layer is the client layer and the 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 the user wants to login into the system first you see two text boxes and a login button to enter a username and password and click on the login button.

Business logic layer which interacts with data layer and sends information retrieved from database towards to presentation layer. It acts as the mediator between presentation layer and data layer.From above example once, the user clicks on login button application layer interacts with the database and sends 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 aceess 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.

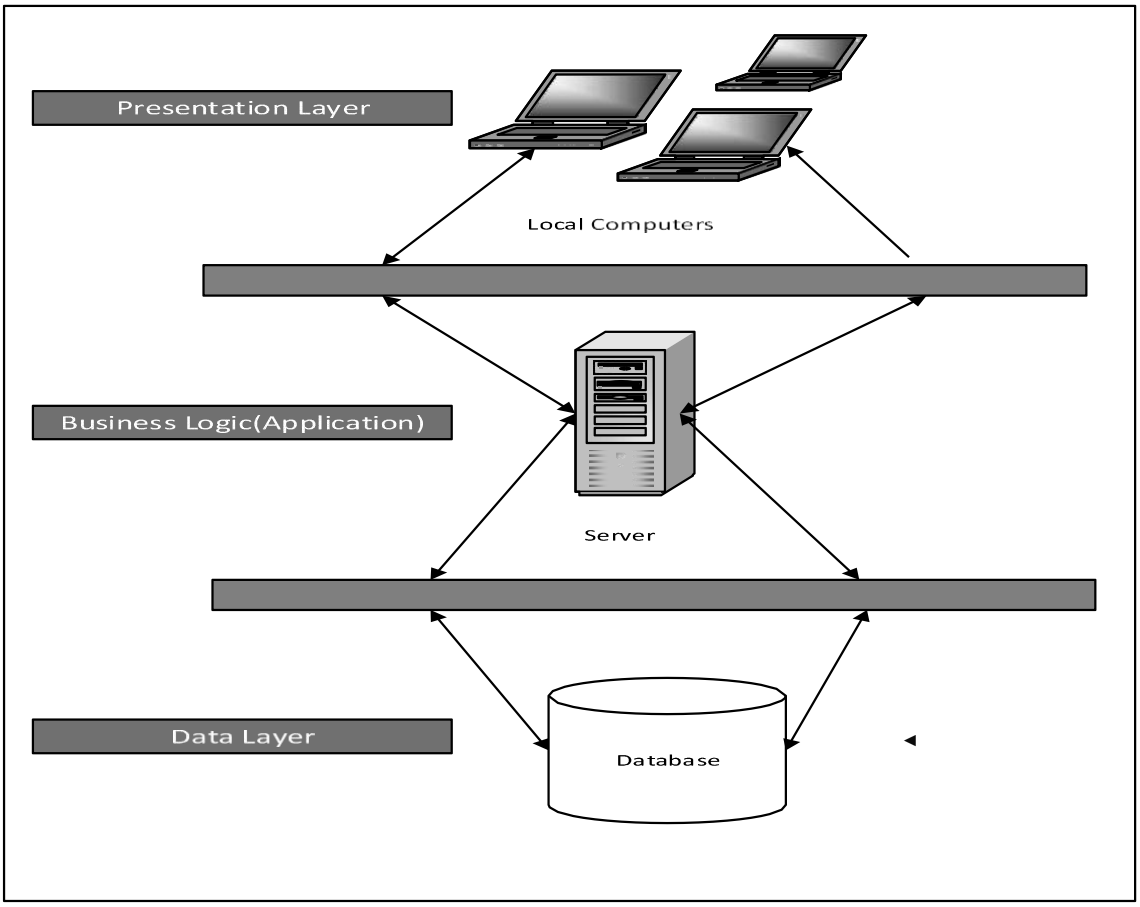


Figure : Proposed System Architecture

5.4.1. Subsystem Decomposition and Description

Subsystem decomposition helps to reduce the complexity of the system. The subsystem integrates the classes that this system contains and the operation performed in the class. The following are the subsystems of our system.

- **Manage account subsystem**

- Create account

- Update account
- Activate account
- Deactivate account
- Change password

❖ **Communication Subsystem**

- Post announcement
- View comment
- Apply to register
- Give Feedback
- Upload module

❖ **User Subsystem**

- Add user
- View user
- Delete user

❖ **Communication Subsystem**

- Give feedback
- Generate report
- View feedback
- View report
- Apply to register

❖ **Material Subsystem**

- Upload module and assignment
- Download module and assignment

❖ **Work Subsystem**

- Prepare grade report
- Approve result
- Submit result
- Generate report
- View result

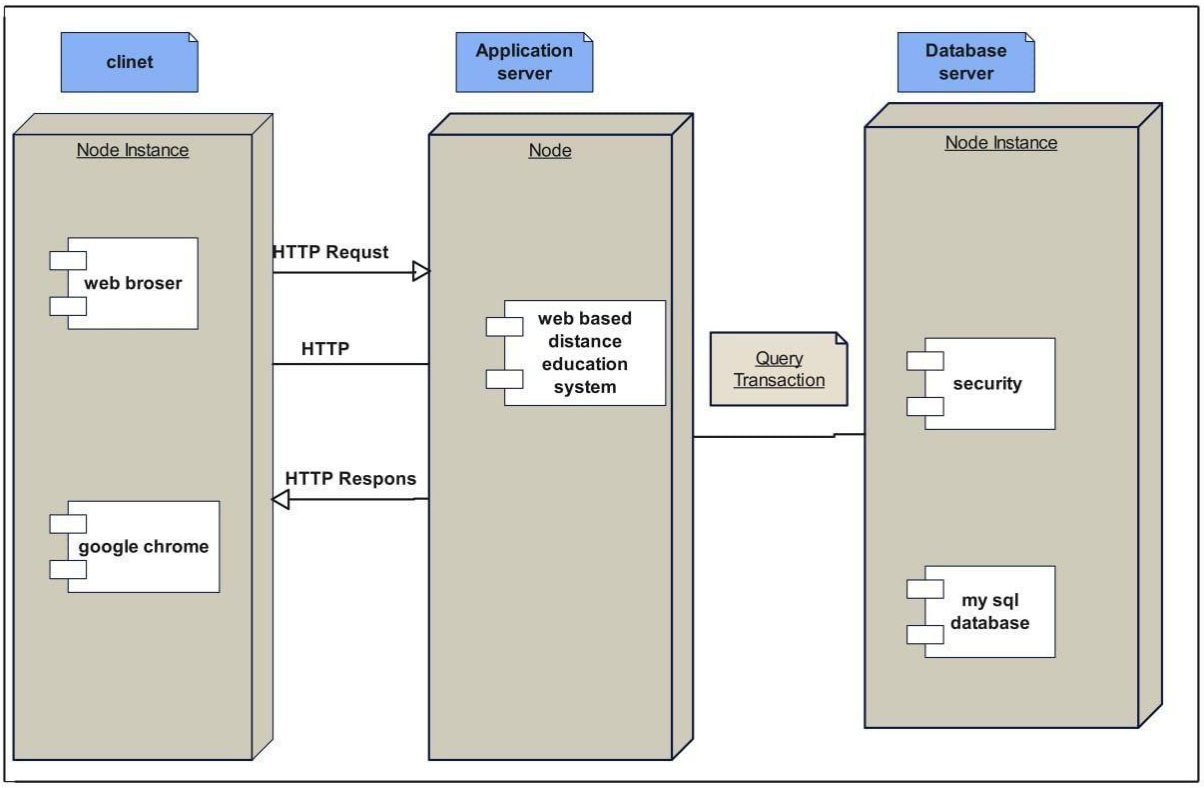
Figure : Subsystem Decomposition Diagram.

5.4.2. Hardware/Software Mapping

One of the major tasks in system design deals with hardware/software mapping which deals with which components would be part in which hardware and so on.

- User interface and processing management will be deployed on the client machine.
- Web-based distance education system will present the Graphical User Interface (GUI), which is used for display user information.
- Web-based distance education system use existing database, which deployed on the server.
- The data collection module deployed on any computer.

Figure: Development diagram Hardware/Software Mapping



5.4.3. Detailed Class Diagram

5.4 Persistent Data Management

Figure : Persistent Data Management Diagram for System Admin

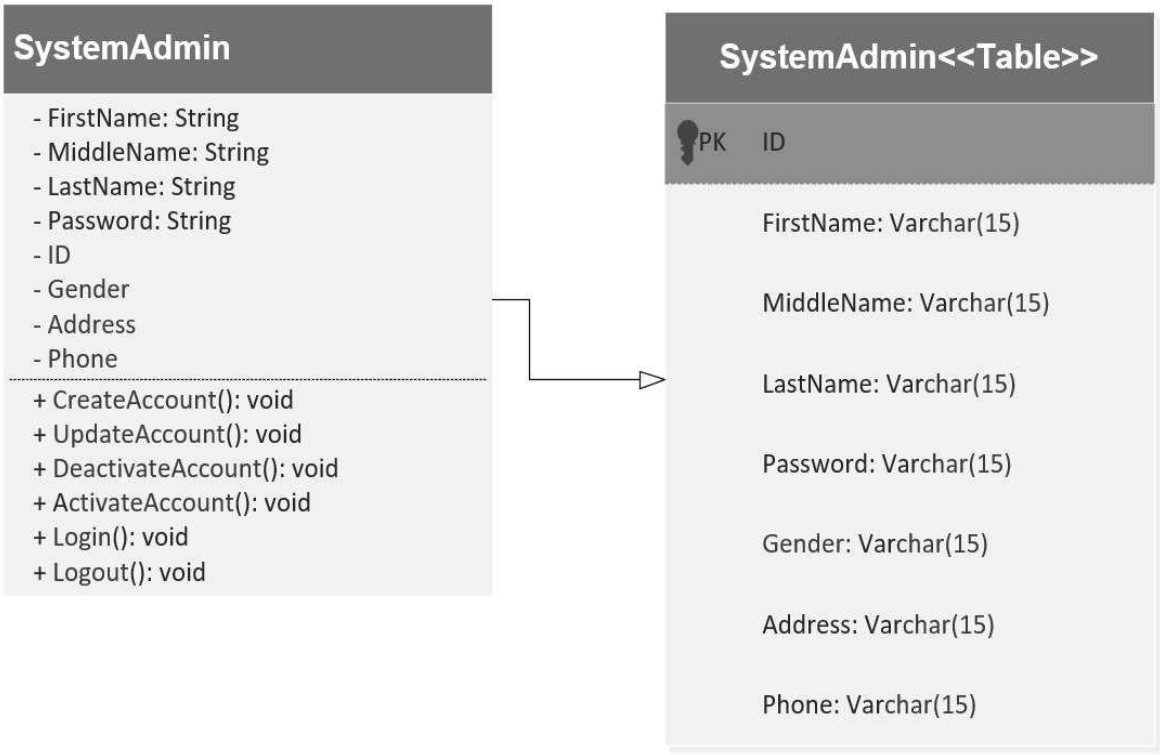


Figure: Persistent Data Management Diagram for student

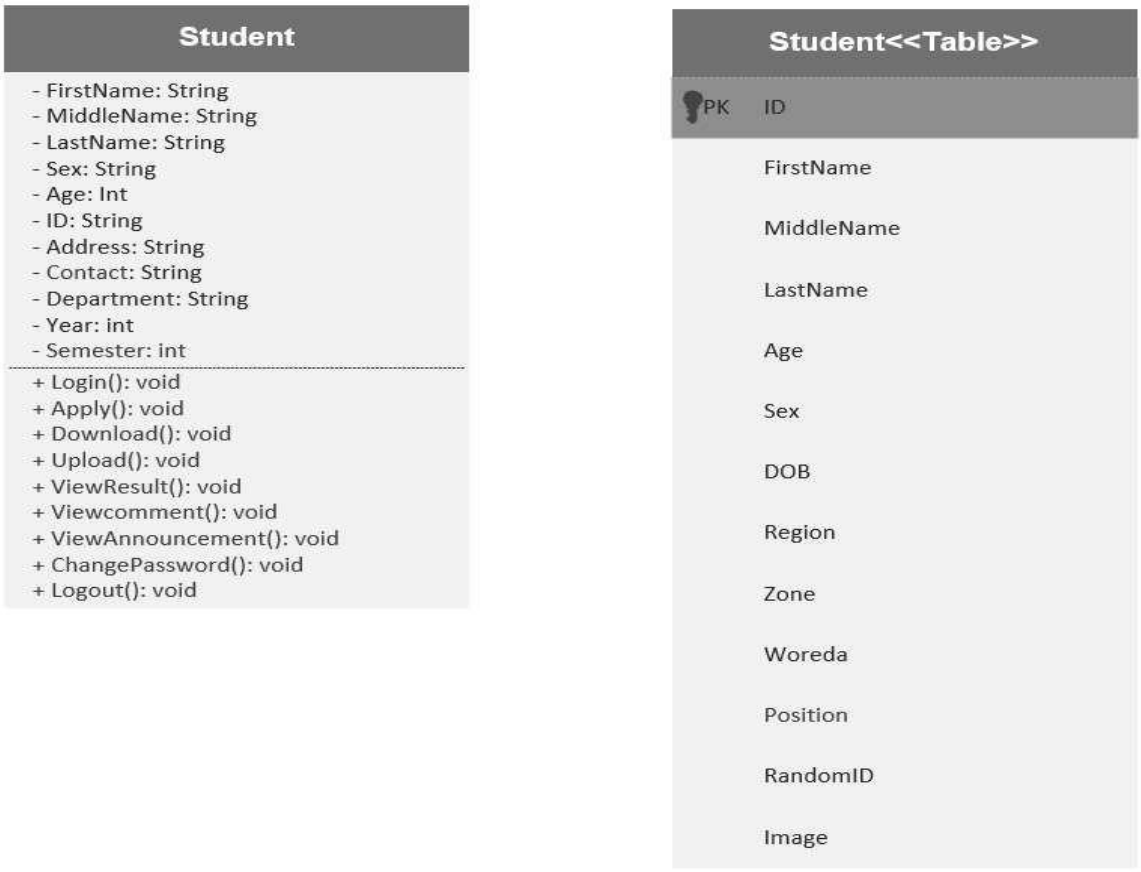


Figure: Persistent Data Management Diagram for Registrar

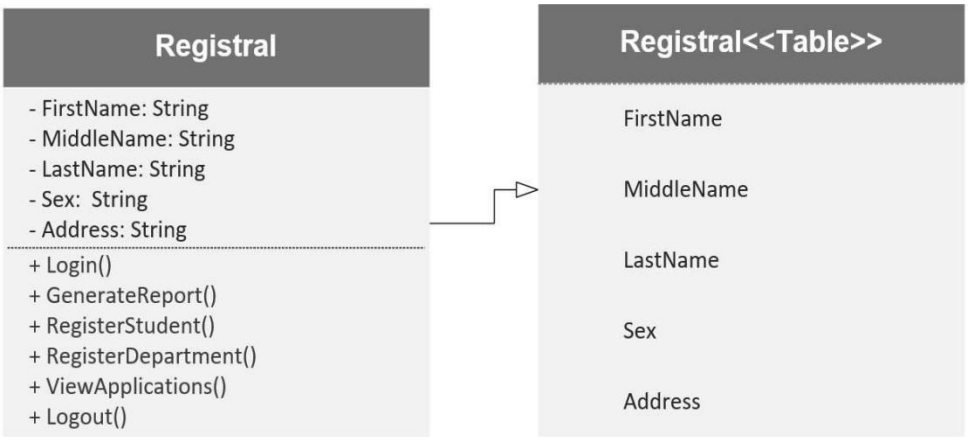
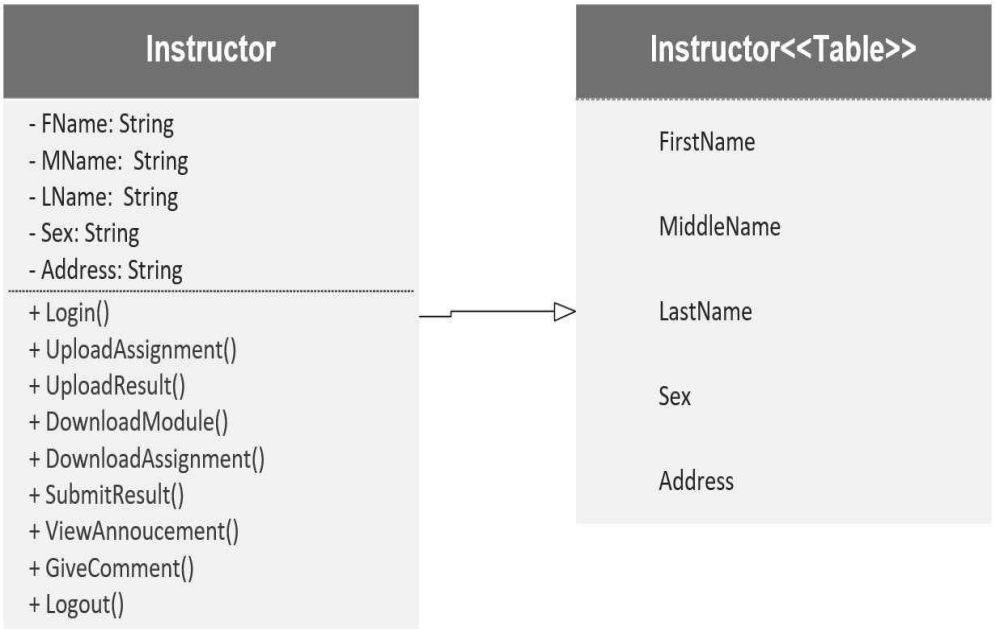


Figure : Persistent Data Management Diagram for instructor

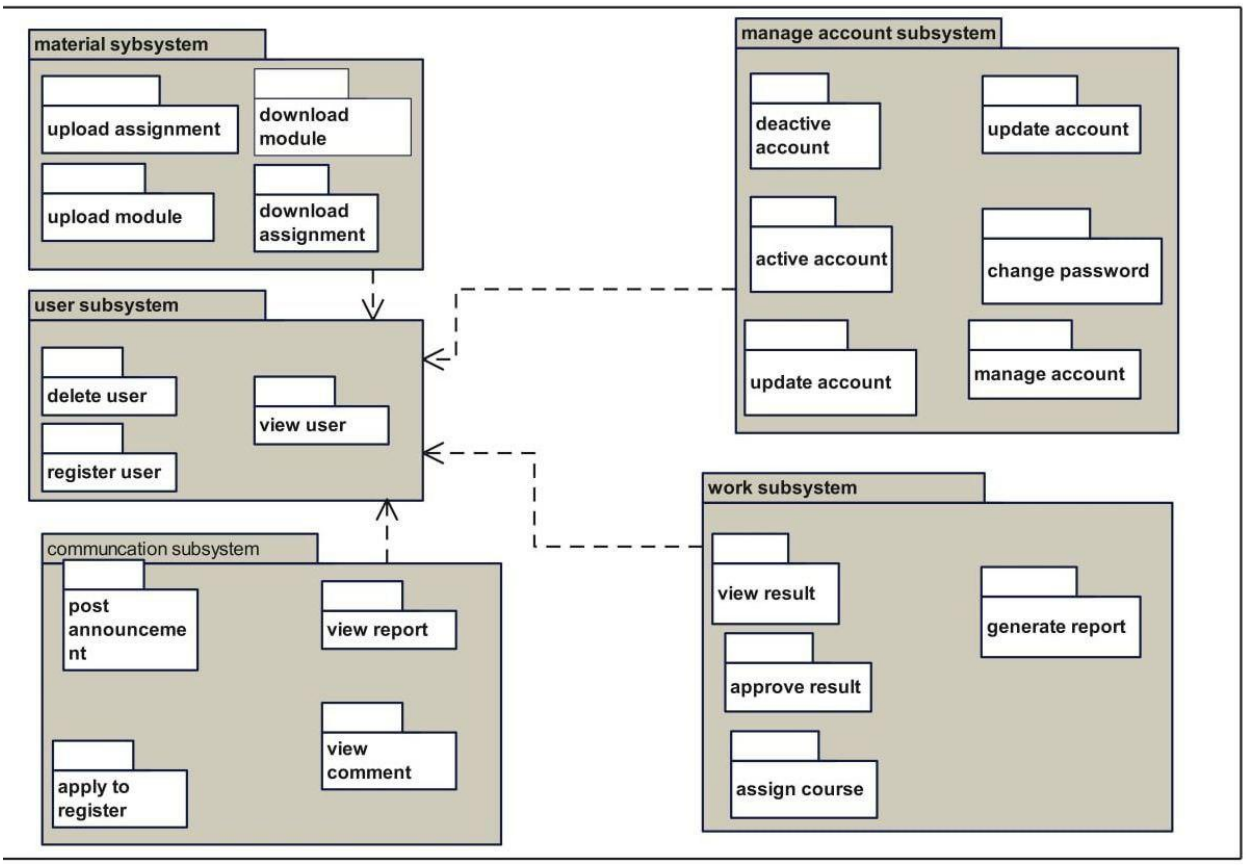


5.5.5. Access Control and Security

Table : Access Control and Security of the System

5.5. Packages

A package is a piece of reusable code that can be dropped into any application and be used without any tinkering to add functionality to that code.



5.6. Algorithm Design

❖ **Login**

- Page displayed on the client machine.
- The user clicks on the login link.
- The system displays the login page or log n page.
- The user enters a username and password in the available space.
- If the username and passwords are correct, then

✓ Login successfully

- Else the user username and password is incorrect,

✓ The system displays an error message and redisplays the login page

❖ Register

- System displays Registrar Officer home page.
- Registrar Officer clicks on register student button.
- The system displays student register form.
- The Registrar Officer enters student information and clicks on Submit button
- If the filled information is correct

✓ Student registered successfully.

- Else the system displays an error message.

❖ Create account

- System displays Admin Page.
- The user account form displayed.
- Admin clicks on create account button.
- System displays create account form.
- Admin inserts all available information and clicks on save button.
- If the filled are correct,

✓ the system displays account created successfully messages.

- Else, the system displays error message.

5.7. User Interface Design

User interface design is the overall process of designing how a user will be able to interact with a system and this is the design of the new system interface. In this system users will communicate with it through the user interface in this section we show the home page, login and create account page. The home page appears as the site on which the system is deployed is opened. This form contains some links which lead it to the concerned page, and if the user has an account he/she will directly go to concerned page by entering their username and password.

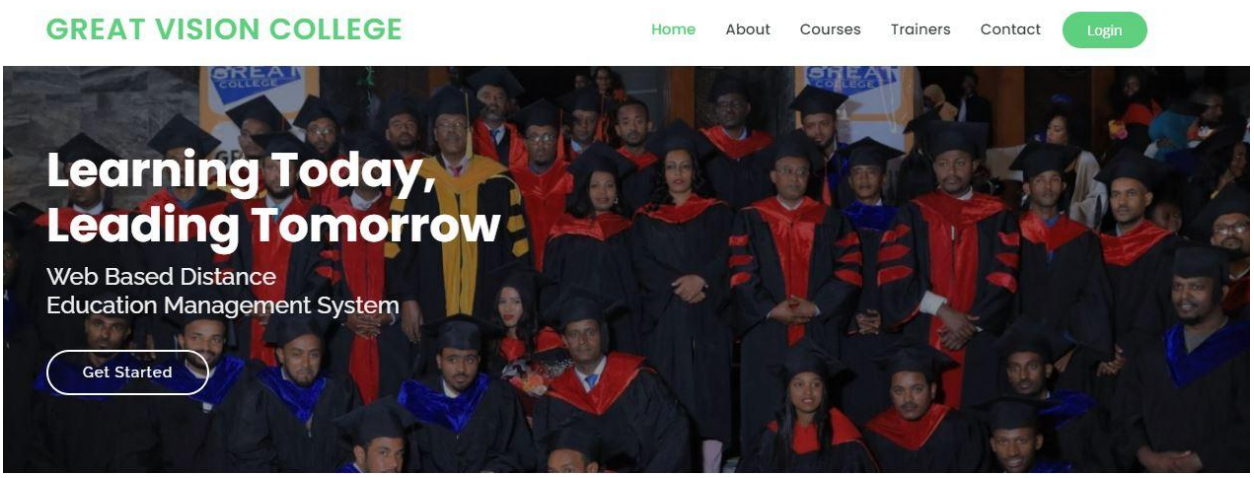


Figure : User Interface for Home Page

Login page: This page displayed, when user's clicks on login link. As one user click login there is chose of account type such as Admin, Instructor, Registrar and Student. All have their own password and user name. Those forms will not accessible by other persons except for those who have privilege.

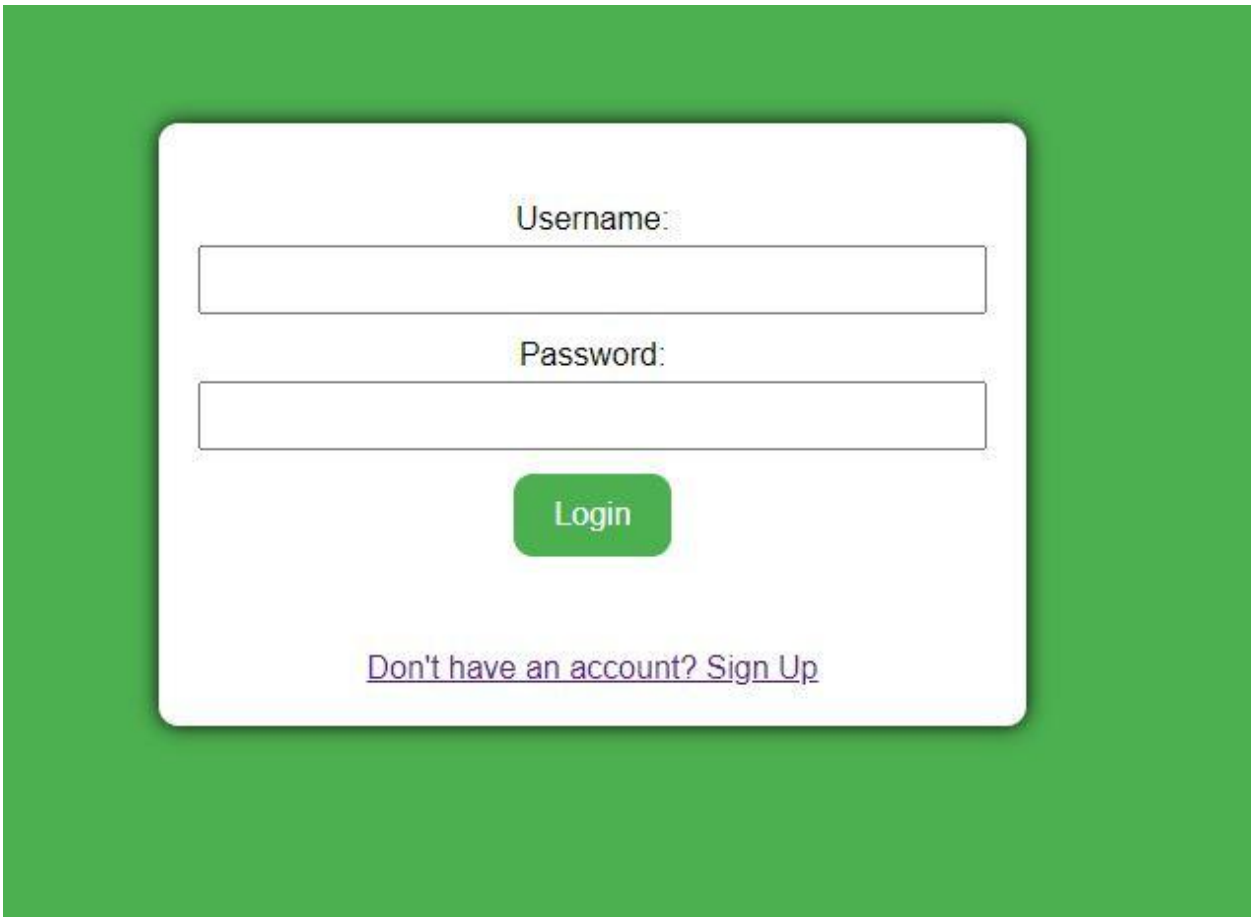
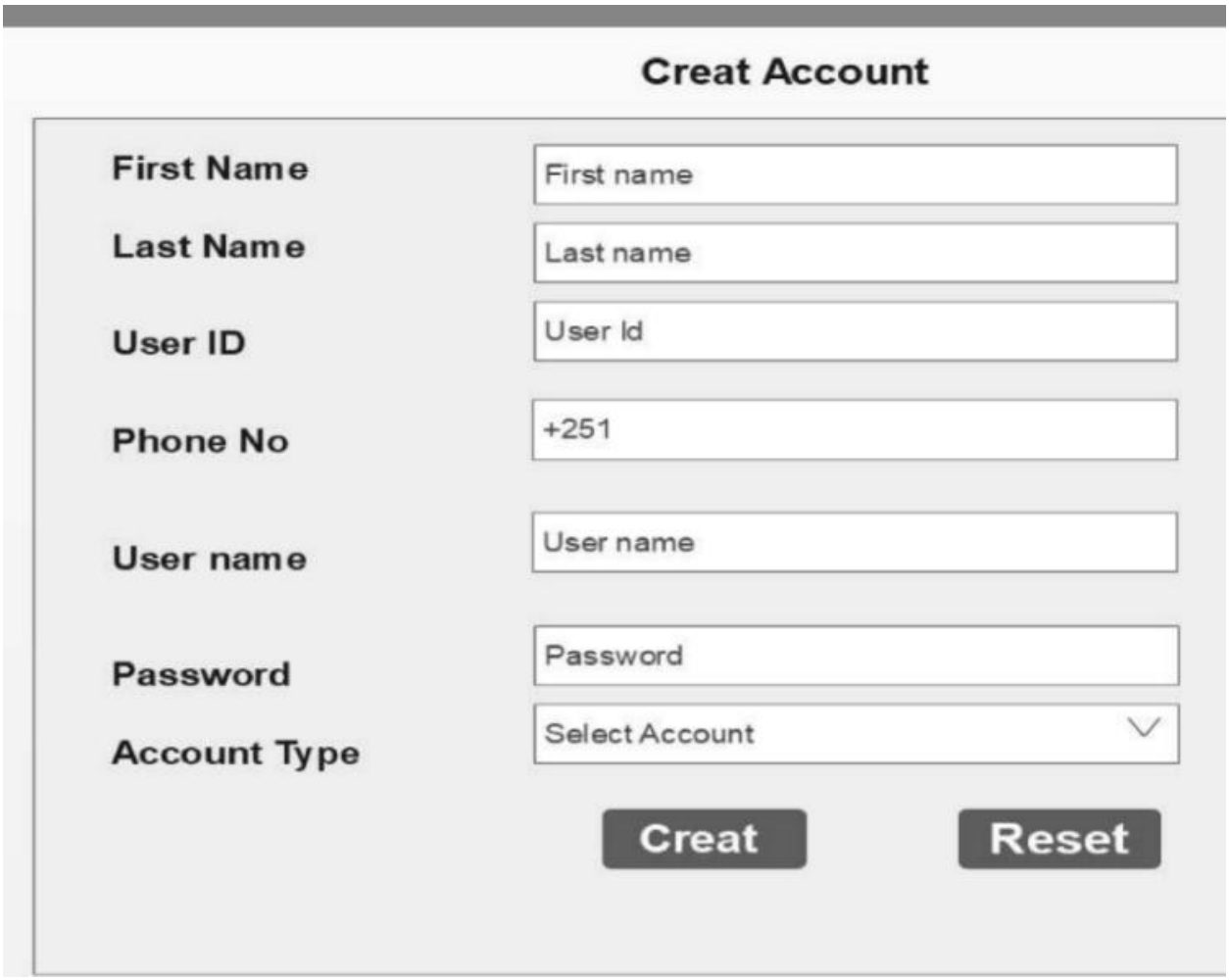


Figure 5-15: User Interface for Login Page

Create Account: This is creating account page in this page the Admin create accounts for the user (academic Dean, instructor, and Student). The account is identified by their type of responsibility.


Creat Account
First Name
First name
Last Name
Last name
User ID
User Id
Phone No
+251
User name
User name
Password
Password
Account Type
Select Account
Creat
Reset

Reference

[1]Agile modeling the Object primer 2nd edition Scott W.Ambler's book.

[2]Refer from dean of great vision college