**ACKNOWLEDG**

First of all, our thanks are basically dedicated to Allah for the completion of the first part of the project. We also would like to the supervisor Fatimah Algamdi for guiding, supporting, and helping us. Allah bless his life, and give him happiness.

# Abstract

The web-based system will be develop to manage the senior project course at information technology department in the Faculty of Computing and Information Technology in King Abdul-Aziz University, Rabigh branch. Provide communication between the coordinator of senior projects and supervisors, and between supervisors and students expected to graduate. By this website will do all the assigned tasks through the system, and provide a list of projects previously implemented.

.

**Keywords:** senior project, coordinator, supervisor, students

# CONTENTS

[ABSTRACT 4](#_TOC_250010)

CHPTER 1: INTRODUCTION 9

* 1. [Introduction 9](#_TOC_250009)
  2. Problem 9
  3. [Suggested Solution 10](#_TOC_250008)
  4. Objective 10
  5. [Project Scope 11](#_TOC_250007)
  6. [Significance of this work 11](#_TOC_250006)
  7. [Outlines of the report 12](#_TOC_250005)
  8. Conclusion-------------------------------------------------------------------------------12

[CHAPTER 2: LITERATURE REVIEW 13](#_TOC_250004)

* 1. [Introduction 13](#_TOC_250003)
  2. Problem [analysis on current system 13](#_TOC_250002)
  3. [Benefit software proposed 16](#_TOC_250001)
  4. [Conclusion-------------------------------------------------------------------------------17](#_TOC_250000)

CHPTER 3: INTRODUCTION 18

3.1 Introduction  [1](#_TOC_250009)8

3.2 Preliminary Investigation 18

3.3System development life cycle overview---------------------------------------------20

3.4 Project Plan------------------------------------------------------------------------------22

3.5 Analyze of Technical Requirement--------------------------------------------------23

3.6 Conclusion------------------------------------------------------------------------------24

**CHPTER 4: SYSTEM ANALYSIS AND DESIGN---------------------**25

4.1 Introduction-------------------------------------------------------------------------------25

4.2 Function Requirement-------------------------------------------------------------------25

4.3 Non-Function Requirement ------------------------------------------------------------26

4.4 Interface Requirement-------------------------------------------------------------------26

4.5 Use Case Diagram-----------------------------------------------------------------------27

4.6 Analyze Phases -------------------------------------------------------------------------35

4.7 System Architecture and Design-------------------------------------------------------44

4.8 System Database Design ---------------------------------------------------------------45

4.9 Conclusion------------------------------------------------------------------------------------------------48

Appendix (A)-----------------------------------------------------------------------------------49

Appendix (B)**------------------------------------------------------------------------**51

Reference**----------------------------------------------------------------------------**53

**List of Figures**

Figure 2.1 Web-based systems for senior project in Norah Princess University --------------------14

Figure 2.2 Web-based systems for senior project in Whitman Collage ------------------------14

Figure 2.3 Web-based systems for senior project in Cal Poly University----------------------15

Figure 2.4 Web-based systems for senior project in Allegheny Collage ----------------------15

Figure 2.5 Web-based systems for senior project in Bard College -----------------------------16

Figure 3.1 Agile Model--------------------------------------------------------------------------------21

Figure 3.2 Project Plan--------------------------------------------------------------------------------22

Figure 4.1 Coordinator use case ---------------------------------------------------------------------27

Figure 4.2 Supervisor use case ----------------------------------------------------------------------28

Figure 4.3 Student use case---------------------------------------------------------------------------29

Figure 4.4 Coordinator sequence diagram----------------------------------------------------------36

Figure 4.5 Supervisor sequence diagram-----------------------------------------------------------37

Figure 4.6 Student sequence diagram---------------------------------------------------------------38

Figure 4.7 Fill registration transaction--------------------------------------------------------------39

Figure 4.8 Idea selection transaction----------------------------------------------------------------40

Figure 4.9 Registration transaction (supervisor)---------------------------------------------------40

Figure 4.10 Evaluation\_COIT498 (supervisor) transaction--------------------------------------41

Figure 4.11 Evaluation\_COIT498 (committee) transaction--------------------------------------41

Figure 4.12 Final evaluation transaction------------------------------------------------------------42

Figure 4.13 Coordinator activity diagram-----------------------------------------------------------43

Figure 4.14 Supervisor activity diagram------------------------------------------------------------43

Figure 4.15 Student activity diagram----------------------------------------------------------------44

Figure 4.16 The architecture of our system---------------------------------------------------------44

Figure 4.17 Database diagram------------------------------------------------------------------------45

**List of Tables:**

Table 2.1 comparison between existing systems and our system------------------------------------17

Table 3.1 Software Requirement-------------------------------------------------------------------------23

Table 3.2 Hardware Requirement------------------------------------------------------------------------24

Table 4.1 Actor Description-------------------------------------------------------------------------------29

Table 4.2 Register ------------------------------------------------------------------------------------------30

Table 4.3 Login---------------------------------------------------------------------------------------------30

Table 4.4 Manage file--------------------------------------------------------------------------------------30

Table 4.5 Manage the complete senior project----------------------------------------------------------31

Table 4.6 View evaluation---------------------------------------------------------------------------------31

Table 4.7 Monitor the progress of current project------------------------------------------------------31

Table 4.8 Manage electronic schedule-------------------------------------------------------------------32

Table 4.9 Evaluate student---------------------------------------------------------------------------------32

Table 4.10 Evaluate other student senior project -------------------------------------------------------32

Table 4.11 Monitor the progress work of student-------------------------------------------------------33

Table 4.12 Submit Assignment----------------------------------------------------------------------------33

Table 4.13 View event--------------------------------------------------------------------------------------33

Table 4.14 Brows available files--------------------------------------------------------------------------34

Table 4.15 View/ Search completed senior project ----------------------------------------------------34

Table 4.16 Manage out-coming transaction-------------------------------------------------------------34

Table 4.17 Manage in-coming transaction---------------------------------------------------------------35

Table 4.18 Send/receive IM--------------------------------------------------------------------------------35

Table 4.19 Coordinator-------------------------------------------------------------------------------------46

Table 4.20 Supervisor (committee)-----------------------------------------------------------------------46

Table 4.21 Student------------------------------------------------------------------------------------------46

Table 4.22 Supervision group-----------------------------------------------------------------------------47

Table 4.23 Follow up--------------------------------------------------------------------------------------**-**47

Table 4.24 File-------------------------------------------------------------------------------**--------------**47

Table 4.25 Transaction--------------------------------------------------------------------------------------47

Table 4.26 Repository---------------------------------------------------------------------------------------48

Table 4.27 Event---------------------------------------------------------------------------------------------48

Table 4.28 Task----------------------------------------------------------------------------------------------48

**CHPTER 1 INTRODUCTION**

# Introduction

The Senior Project is a student-selected exploration of a topic, which results in a research paper, a project or a product, and a presentation [1]. The senior moves students away from departmentalized learning toward a more interdisciplinary approach. This approach is one, which allows students to use a variety of skills in the areas of writing, speaking, research, and documentation. Upon completion of the senior project course, the students have learned more about their topics and improved their experience as well as their communications and presentation skills.

The supervisors of senior projects are responsible to register, login into system, manage incoming transactions, manage out coming transactions, monitor the progress work of student, assign work for students, evaluate students (other senior project student, under supervision), view the event and send /receive instant message IM.

As the other academic courses, senior project is managed by an assigned coordinator responsible for create an account to student and supervisor, upload Files, mange completed senior project (insert, view, update, delete), mange transaction (incoming, out coming), monitor the progress work of current project, send/receive (IM) and view evaluation result.

Once the student registers the senior project course, can login (reset password), send/receive (IM), view events, brows available files, submit assignment, mange transaction (incoming, out coming) and view/search for completed senior project (repository).

In our project, we aim to enhance the communication between the supervisor and students who work on project senior and help the supervisor to keep track of the students work. More specifically, we want to develop a solution that helps supervisor, students and course coordinator to communicate with each other and manage their works through a web-based system.

# Problem Statement

Most of the senior project supervisors in the Faculty of Computing and Information Technology at King Abdul-Aziz University, Rabigh Branch run their work manually. This is a tedious process, takes a long time, and consumes more efforts leaded to a significance delay in the task accomplishing. The problems that we address in our project can be summarize as follows:

* + - No listing for senior projects previously implemented.
    - Lack of ideas for technical projects.
    - Having difficulty in writing a senior project report.
    - Lack of an electronic system that reminds supervisors and students for the due date of weekly assignments and the other scheduled events.

# Suggested Solution

To solve the mentioned problem, we propose to developing a web-based system that allows managing projects electronically and providing archives for the completed senior projects.

We can summarize the features of our proposed system as follows:

* + - Search engine tool through which the user can search for ideas that have been applies.
    - Web-based forms to be filled and submitted electronically.
    - Receive the committee evaluation result and feedback for the proposed idea through the system.
    - Provide ideas for senior projects.
    - Web-based provides an alert to remind students of the date of weekly assignments.

# Aim & Objectives

The main aim of our project is to develop web-based system for senior projects management.

The objectives:

1. Provide electronic transaction.

2. Facilitate the communication between student, supervisor and coordinator.

1. Archiving the completed project in a repository.
2. Organize the task and event through electronic schedule.



# Project Scope

This web-based system will be develop to manage the senior project course at information technology department in the Faculty of Computing and Information Technology in King Abdul-Aziz University, Rabigh branch. The course’s coordinator, supervisors, and graduate students will do all the assigned tasks through the system. Furthermore, all the completed projects will be available for anyone who needs, as they are stored in the repository connected to the system.

# Significance of this Work

Many benefits can be gain through this web-based system especially for graduate students, supervisors and coordinator of senior project.

For graduate students: register (fill Registration form and submit), login (reset password), send/receive (IM), view events, brows available files, submit assignment, mange transaction, (in coming, out coming) and view/search for completed senior project (repository).

For supervisors: register, login into system, manage incoming transactions, manage out coming transactions, monitor the progress work of student, assign work for students, evaluate students (other senior project student, under supervision), view the event and send /receive IM.

For coordinator: create an account to student and supervisor, upload Files, mange completed senior project (insert, view, update, delete), mange transaction (incoming, out coming), monitor the progress work of current project, send/receive (IM) and view evaluation results.

This web-based system simplifies the communication between actors of senior project course how are the course’s coordinator, supervisors and graduate students.

* 1. **Outline of this Report**

This report will be divide into four chapters:

* In Chapter 1, we talked about the project's problem, objectives, project scope, suggested solution, and significance of our work.
* In Chapter 2, we will describe problem analysis of the current system, benefits of the proposed software.
* In Chapter 3, we will discuss the adopted methodology and describes clearly the phases of planning, analysis of the project requirements.
* In Chapter 4, we will describe the system analysis and design parts. This chapter will include the description of functional requirements, non-functional requirements, interface requirements, Use Case Diagrams, Sequence Diagrams, Activity Diagrams, system architecture and design, as well as the database design.

# 

# 1.8 Conclusion

This chapter gives preliminary information regarding the project that will be built in the future. The Problem of the project is also present in this chapter to give a clear picture of the motive of the project.

# CHAPTER 2 LITERATURE REVIEW

# Introduction

The management of senior projects is similar to the role of managing any work, which includes follow-up and monitoring by the coordinator and supervisors. With the development of computer technology and smart devices, such management becomes more effectively if it uses a computer-based system. In this chapter, we will review some of the existing systems used to manage the senior projects in different universities, provide brief description for each one and compare them with our proposed system.

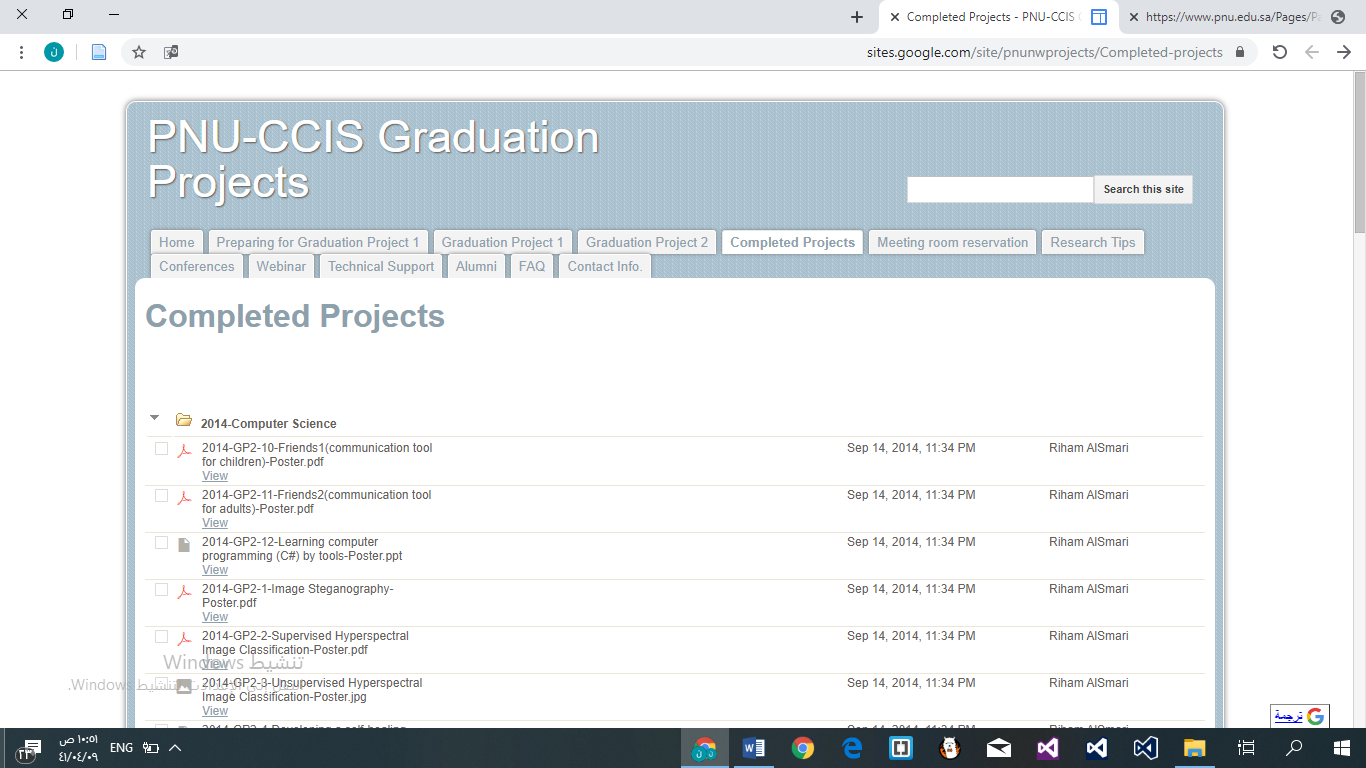
# Problem Analysis on Current System

Many different web-based systems have been develop to help students in finding the senior projects that were executed in their faculty, guideline for writing project, managing group that work in senior project and give to student suggesting idea for senior project. These systems differ in the size and services provided. Some are computer program and others website or mobile-based applications running on smart device. Our project focus on the web-based system, and in this section. We review some of similar web-based systems from different university they are list based on university/faculty name.

# Norah bint Abdul Rahman Princess University

Norah bint Abdul Rahman Princess University is located in Riyadh and was open during the reign of King Abdullah bin Abdul Aziz, the first university in Saudi Arabia especially for girls [2].

Norah bint Abdul Rahman Princess University has the website of senior project it provides an archive service for senior projects classified into three categories (information technology, computer science, information systems), but it does not have the projects management feature online.



**Figure 2.1** Web-based systems for senior project in Norah bint Abdul Rahman Princess University

# Whitman Collage

Whitman Collage has a website for senior projects archiving. It provide archives service for senior project and arrange by year. Similar to Norah bint Abdul Rahman princess university website it does not have project management feature online through the website [3].



**Figure 2.2** Web-based systems for senior project in Whitman Collage

# Cal Poly

Cal Poly University has a website provide archive service for student. It allow the students and faculty members to sign in by their university email address and upload their senior projects. Forth more the students from outside this university to create free account and publish their projects. Similar to systems Norah bint Abdul Rahman princess university website and Whitman Collage it does not have project management feature online through the website [4].

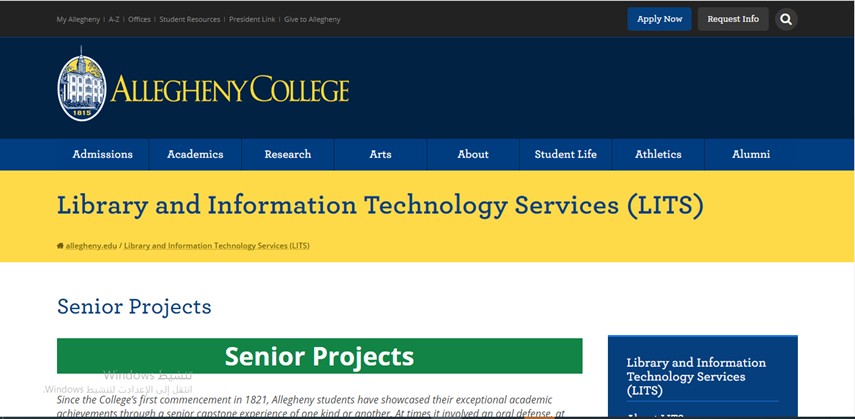


# Figure 2.3 Web-based systems for senior project in Cal Poly University

# 2.2.4 Allegheny Collage

Allegheny Collage has a website for senior project include the following functions:

* + - * Provide icons to access to all senior project.
      * Submit service for your project.
      * Guideline for uploading your projects [5].

 Still no similar systems manage senior project online.

**Figure 2.4** Web-based systems for senior project in Allegheny Collage

# 2.2.5 Bard Collage

Bard College has a website for senior project with the following functions:

* Provide archive service for senior project.
* Availability of a search engine that searches for previous projects but shows all archived projects [6].

Still no similar systems manage senior project online.

# Figure 2.5 Web-based systems for senior project in Bard College

# Benefit software proposed

Our proposed system focuses in simplifying the management of the senior project and assists the course’s coordinator, supervisors, and students to do their works online. Some of the benefits of our system includes:

* Store the completed projects in a repository to help others in finding ideas or understanding the requirements of senior project.
* Guideline for submitting your projects.
* Provides a calendar for scheduling meetings, and other services.

We have compared the existing systems and our proposed one according to different features as shown in table 2.1

Table 2.1: Comparison between existing systems and our system

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Norah bint Abdul Rahman princess university | Whitman Collage | Allegheny Collage | Cal Poly University | Bard College | Our project |
| View list of senior project | √ | √ | √ | √ | √ | √ |
| Projects management feature online | × | × | × | × | × | √ |
| Guideline for submitting  your projects | × | × | √ | × | × | √ |
| Provides a calendar for scheduling meetings | × | × | × | × | × | √ |
| Provides a  List of (students or team ) | × | × | × | × | × | √ |
| Require create account and login | × | × | × | × | × | √ |

# Conclusion

In this chapter, we reviewed different senior projects management systems and showed the differences between them and our system in terms of the provided services. Our system will include maximum number of services so it can provide a great benefit for the students, supervisors and coordinator of senior projects at FCITR.

**CHAPTER 3**

**METHODOLOGY**

**3.1** **Introduction**

This chapter explains the appropriate software development methodology used for to develop our proposed system. Every methodology has their own requirements, advantages, disadvantages and characteristics as summarized throughout the sections of this chapter at the end; we discuss the technical requirements needed to complete our system.

**3.2** **Preliminary Investigation**

To investigate about the needs for web-based system that we are proposing, we do an interview and electronic questionnaire:

**3.2.1** **Interview**

We interviewed with the coordinator of the senior project (Dr.Rathiah) in the IT department. The idea of ​​our project was explain, and one of the services in our project was to provide virtual classrooms. She explained to us that virtual classrooms exist in Blackboard and that our project does not need a virtual classroom service.

Therefore, it was replace by the instant messaging service between the supervisor and the students and between the supervisor and the coordinator.

**3.2.2 Questionnaire**

We do web based questionnaire and distributed to undergraduate students and supervisors of IT department at FCITR king Abdul-Aziz University.

The questions in the questionnaire for undergraduate students at the following:

* Difficulty finding an idea for a senior project.
* The extent to which students want to submit the project idea electronically.
* The problems that students often encounter after introducing the idea of ​​a graduation project.
* The extent students want to search for previous senior projects that have been implemented and similar to their ideas.
* The extent to which students want to search for previous senior projects on our project using words of inference.
* Services that students want in our project.

The question form with the result is append in appendix (B) as a result we found that:

100% of students find it difficult to find an idea for senior project.

77.8% of students prefer to submit a template for a major project idea electronically, while 22.2% prefer not to do so.

94.4% of students need a web-based to help them search for previously implemented ideas, while 5.6% of students do not need this.

83.3% of students prefer to search for previous projects on the web-based using words of inference, while 16.7% of students do not prefer this.

The questions in the questionnaire for supervisors at the following:

* The extent to which supervisors want to submit evaluation forms electronically.
* The extent to which supervisors want to monitor weekly progress through our project.
* The extent to which supervisors want to evaluate the students' weekly assignments using a common electronic form between them and their students.
* Services that supervisors want in our project.

The question form with the result is append in appendix (A) as a result we found that:

100% of supervisors want to submit evaluation forms electronically.

100% supervisors want to monitor weekly progress across the web-based.

100% of supervisors prefer to evaluate students ’weekly assignments using an electronic evaluation form between them and their students across the web-based.

**3.3** **System Development Life Cycle (SDLC) Overview**

The Software Development Lifecycle is a systematic process for building software that ensures the quality and correctness of the software built.  
SDLC consists of a detailed plan, which explains how to plan, build, and maintain specific software. Increases visibility of project planning to all involved stakeholders of the development process**.**

The prime reasons why SDLC is important for developing a software system.

* It offers a basis for project planning, scheduling, and estimating
* Provides a framework for a standard set of activities and deliverables it is a mechanism for project tracking and control.
* Increases visibility of project planning to all involved stakeholders of the development process
* Increased and enhance development speed
* Improved client relations
* Helps you to decrease project risk and project management plan overhead SDLC phases [10].

**3.3.1** **Popular System Development Life Cycle models**

* **Waterfall model**

This SDLC model is documentation-intensive, with earlier phases documenting what need be perform in the subsequent phases [11].

* **Incremental Approach**

Modification to the incremental model allows development cycles to overlap [12].

* **V-Model**

After that, subsequent cycle may begin before the previous cycle is complete.

Therefore, there are verification phases on the side and the validation phase on the other side [13].

* **Agile Model:**

The entire project is divide into small incremental builds. All of these builds are provide in iterations, and each iteration lasts from one to three weeks [14].

* **Spiral Model**

The spiral model is a risk-driven process model. This SDLC model helps the team to adopt elements of one or more process models like a waterfall, incremental, waterfall, etc [15].

* **Big bang model**

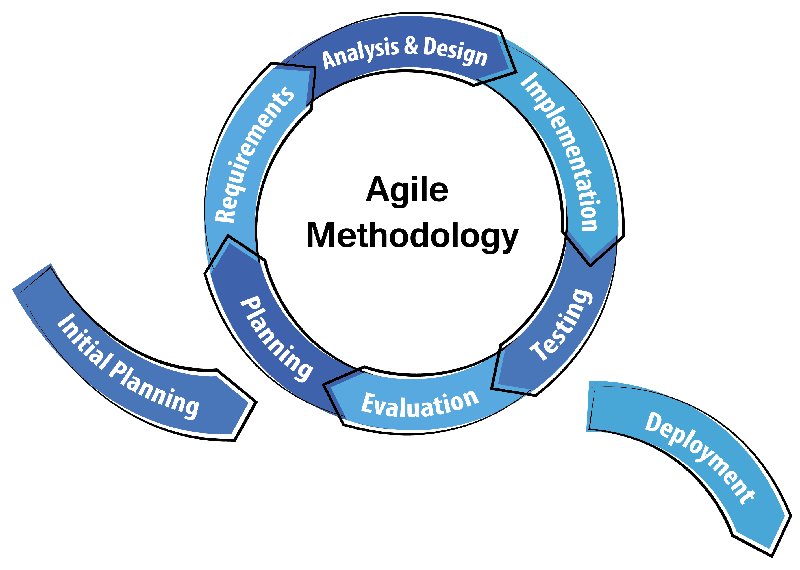
Big bang model is focusing on all types of resources in software development and coding, with no or very little planning. The requirements are understood and implemented when they come [16].

**3.3.2** **Selected System Development Life Cycle for our project**

The agile model is select as SDLC for our project because it has many advantages includes [17]:

* Rapid deployment of solutions, more efficient use of resources
* Greater flexibility and adaptability to changing needs.
* More rapid detection of problems and thus quicker fixes.
* Increased collaboration with user’s products that had better meet user needs.

There are also potential drawbacks, however, including a tendency for projects to go off track, a lack of documentation and less predictable outcomes [18].



**Figure3.1** Agile Model

As shown in figure 3.1, agile model has different phases [19]:

**Planning:** Putting plan of works that must be done and set deadline for each process.

**Requirements:** Gathering information through interview and questionnaires can determine the end user requirements.

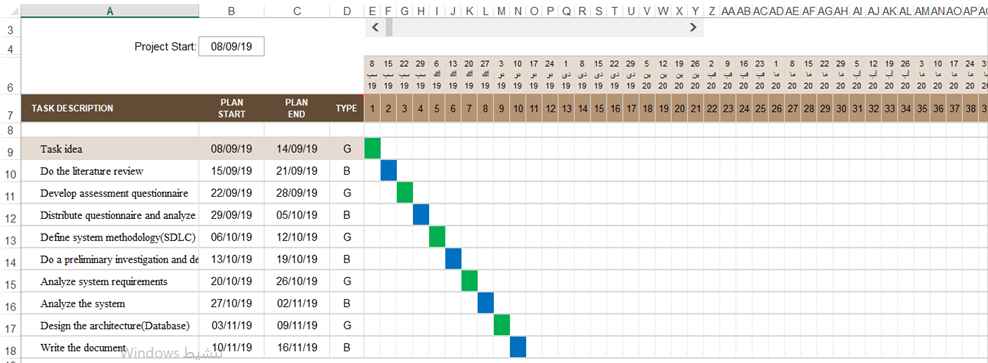
**Analysis & Design:** Determine the requirements of functional and non-functional for each model of the system and create UML charts, ERD diagram based on analysis.

**Implementation:** Start coding and implementing the activities described in the design phase.

**Testing & Evaluation:** Test and evaluation of the coding according to functions of the system into the final version of prototype that being developed.

**3.4** **Project Plan**

We summarized the steps to implement our project in a Gantt chart as shown in figure 3.2. For drawing Gantt chart, we used Microsoft Excel software.



**Figure 3.2** Project Plan

**3.5** **Analyze of Technical Requirement**

The technical requirements for our projects are analyze as software and hardware requirements.

**3.5.1** **Software Requirement**

Table 3.1: Software Requirement

|  |  |
| --- | --- |
| IDE(Integrated Development Environment) | Visual studio |
| Programming language | Bank-end: c#  Front-end: HTML, CSS, Java Script |
| Framework/Library | ASP.NET |
| Database | MySQL |

As shown in table 3.1, Integrated Development Environment we choose is the visual studio because visual studio is integrated development environment from Microsoft; it is use to develop computer programs, as well as websites, web apps, web services and mobile apps [20].

Programming language we choose for front-end is java script because java script is the programming language of HTML. HTML: Hypertext markup language that understood by browser to display content is use to render webpage in browser [21]. C# is back-end language in ASP.NET web framework (client site) [22].

Framework/library we choose is ASP.NET. ASP stands for Active Server Pages and it is a web application framework and it extends the .NET platform with tools and libraries specifically for building web apps marketed by Microsoft to allow programmers to build dynamic web sites [23, 24, 25, 26].

The reasons to consider using ASP.NET:

* A rich toolbox and designer in the Visual Studio integrated development environment complement the ASP.NET framework
* ASP.NET reduces the amount of code required to build large applications the source code and HTML are together therefore ASP.NET pages are easy to maintain and write.
* It is server-side technology so, ASP.NET code executes on the server before it is sent to the browser.
* ASP.NET makes it easier to build a site while using less code than other programming options. With less code to process, websites and apps load faster and more efficiently .ASP.NET packages also uses compiled code rather than interpreted code.

* ASP.NET it is low cost so we can write ASP.NET code in any simple text editor, including free options like Microsoft’s Visual Studio application.
* It is offers support for multiple languages ASP.NET.
* ASP.NET includes built-in features that protect against common hacker.

Web server we choose is Apache HTTP server because it is the worlds most used and popular web server software, its job basically to accept requests from clients and send responses to those requests and its interpreter of the PHP codes.

Database we choose is MySQL because it is the most popular open source SQL, database management system, is developed, distributed, and supported by Oracle Corporation [27].

**3.5.2** **Hardware Requirement**

We will develop our system in a personal computer with the following specification:

Table 3.2: Hardware Requirement

|  |  |
| --- | --- |
| Operating system | windows 10 |
| Hard disk drive | 300 gigabytes |
| RAM | 8 gigabytes |
| Architecture | 64 bits |

In addition, we need a modem device to connect to internet with data transfer rate in download 4.2 Mbps, and upload 1.8 Mbps.

**3.6** **Conclusion**

In this chapter, identified system requirements by interviewing the coordinator and distributing the questionnaire among the students and supervisors. We reviewed the software development methodology. In addition, we set up the plan to implement our project. Finally, we analyzed the software/hardware requirements.

**Chapter 4**

**SYSTEM ANALYSIS AND DESIGN**

**4.1 Introduction**

This chapter implements the analysis and design phase for this system by  
identifying the functional and non-functional requirement also identifying the requirement interface of system. The UML diagram will be use to illustrate the basic function of the system and how the user interacts with the system such as sequence, activity, use case and activity Diagram. In the last section, the architecture design phase of the system will be discuses.

**4.2 Functional Requirement**

There will be three users for this project, each user has own functional requirements as shown in the following subsections.

**4.2.1 Coordinator**

Coordinator is one of cadmic staff how is responsible for managing the course of senior project. Some of the function done by coordinator are list following:

* Register.
* Log in.
* Create an account to student and supervisor.
* Manage Files.
* Mange Completed Senior project (Insert, View, Update, delete).
* Mange incoming transaction.
* Monitor the progress work of current project.
* Send/Receive instant message (IM).
* View evaluation result.
* Log out.

**4.2.2 Supervisor**

Supervisor is one of cadmic staff how is assign for group of student to guide them and follow their progress to complete the senior project. Some of supervisor's function is as following:

* Register.
* Login into system.
* Manage Incoming transactions.
* Manage out coming transactions (evaluate COIT498, final evaluate).
* Monitor the progress work of student.
* Assign work for students.
* View the event.
* Send /receive IM.
* Logout.

**4.2.3 Student**

Student included in our system are the undergraduate once how register the senior project course. Some of student function is as following:

* Register (fill Registration form and submit).
* Login (reset password).
* Send/receive (IM).
* View events.
* Brows available files.
* Submit assignment.
* Mange transaction (in coming, out coming).
* View/search for completed senior project (repository).
* Logout.

**4.3 Non-Functional Requirements**

* Usability: ease of use the web site.
* Fast: the web site must be fast in response for user request.
* Availability: the web site will be available at every time for user.
* Reliability: the information that will appear to the user must be correct [28].

**4.4 Interface Requirement**

The interface must be easy for the user to understand, so interact between the user and the system will become more efficient. In the web site focus on some characteristic to get better Interface. The most thing keep the interface is a simple by using clear language in label and massage in the interface and avoid unnecessary elements to make it easier to user understanding [29].

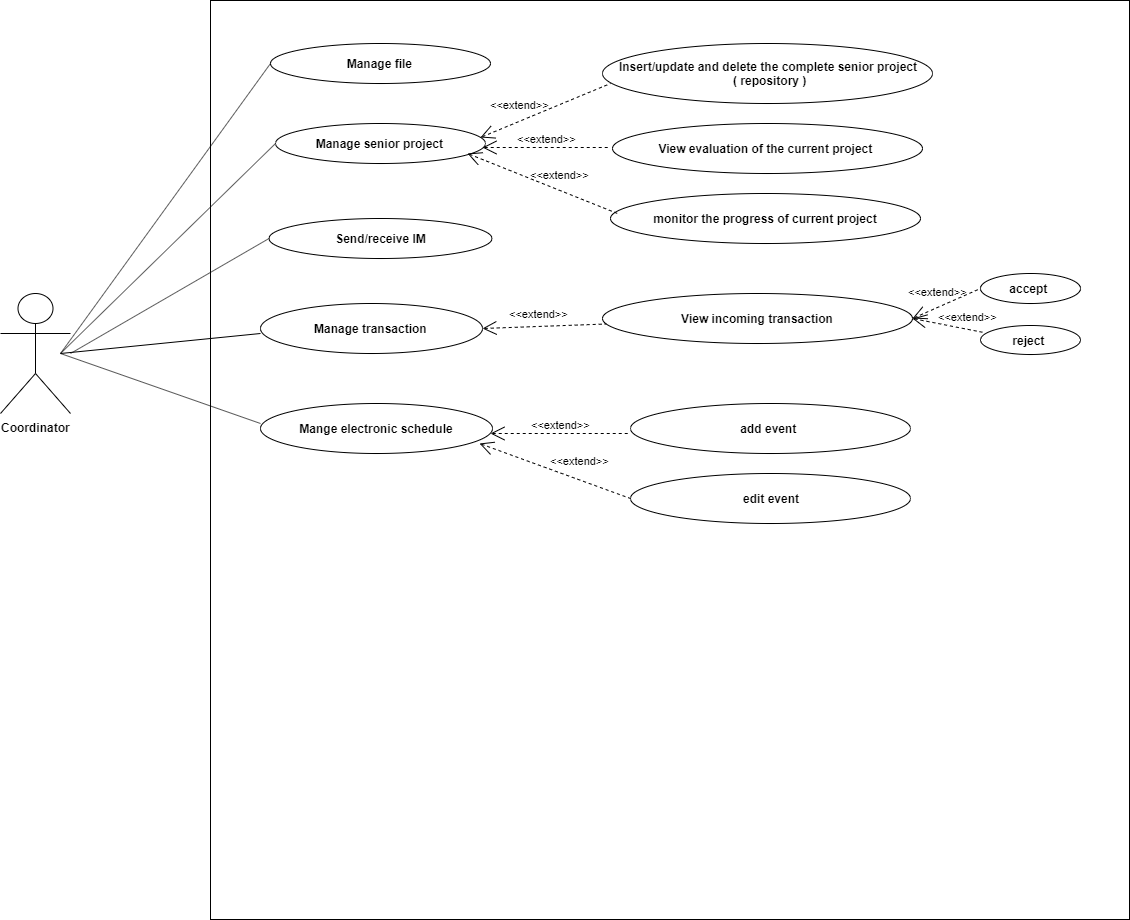
* User interface design should have clear colors.
* The transition movement between pages and access to each should be easy and sooth.

**4.5 Use Case Diagram**

Use case diagram is use to present a graphical overview of functionality provided in a system in terms of actor, functions and dependencies between use and actors.

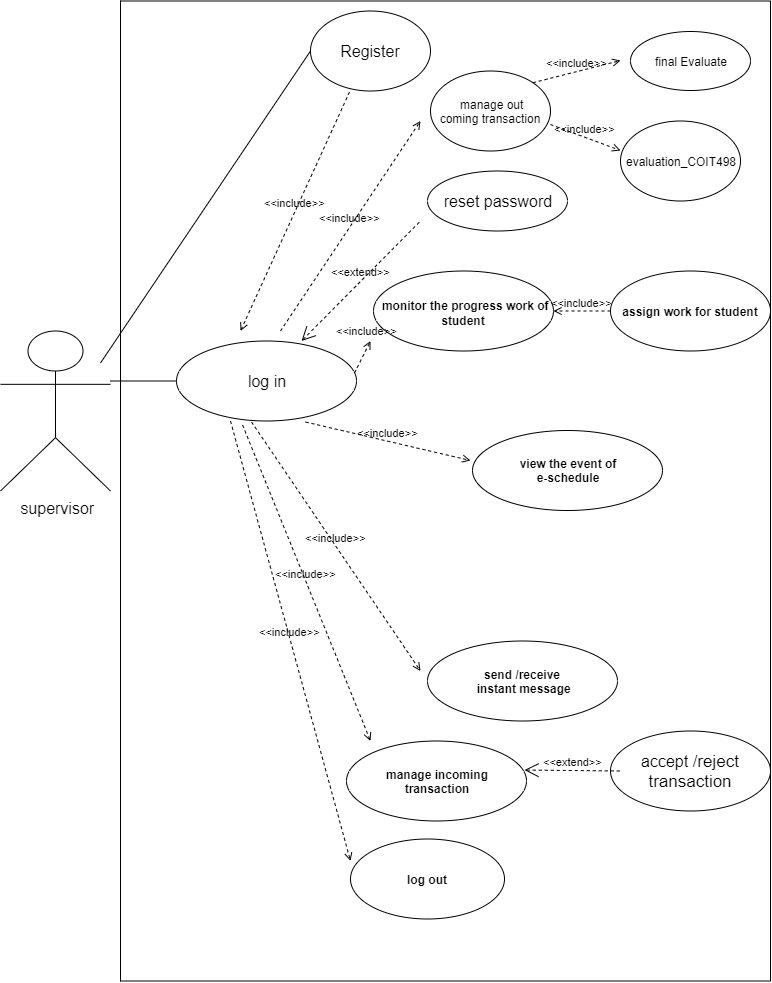
The diagram is to illustrate the interactivity between user and system or website where the actor is the user of the system or website, while use case is the function implement in system or website. An important part of the Unified Modeling Language (UML) is the facilities for drawing use case diagrams [30]. The use case diagram for some operations of our web site are illustrated in figure 4.1-4.3

Coordinator use case diagram:



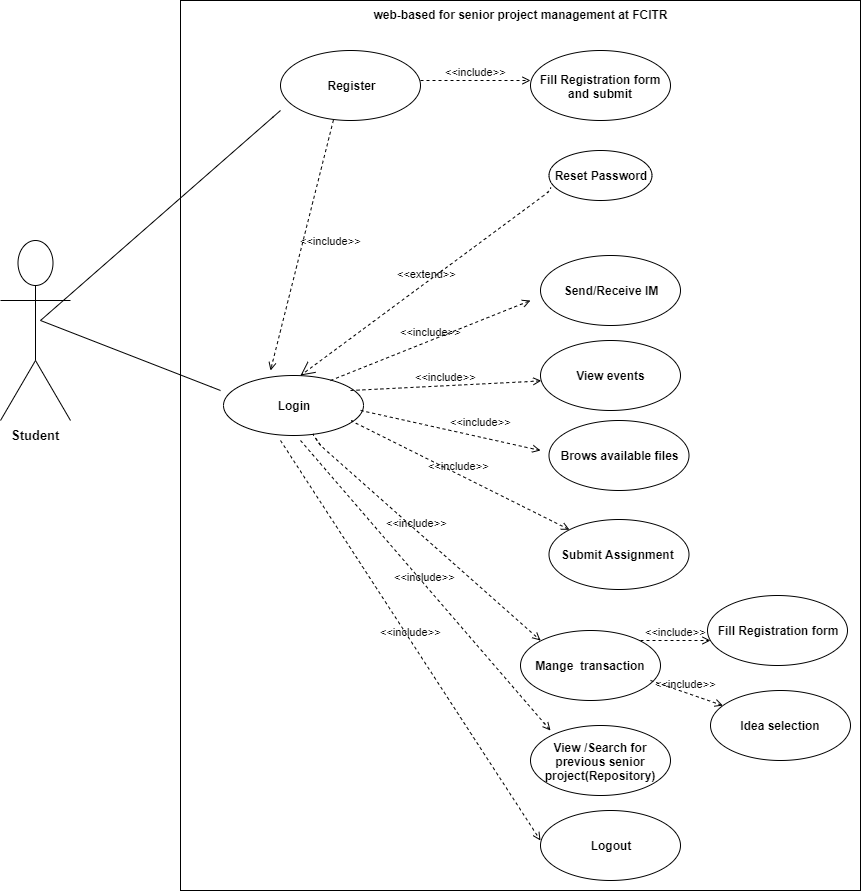
**Figure 4.1** Coordinator Use Case

Supervisor use case diagram:



**Figure 4.2** Supervisor Use Case

Student use case diagram:



**Figure 4.3** Student Use Case

**4.5.1 Actors Description**

Three actors for our proposed systems as described in table 4-1:

Table 4-1: Actor Description

|  |  |
| --- | --- |
| Actors | Description |
| Coordinator | Coordinator is one of cadmic staff how is responsible for managing the course of senior project. |
| Supervisor | Supervisor is one of cadmic staff how is assign for group of student to guide them and follow their progress to complete the senior project. |
| Student | Student included in our system are the undergraduate once how register the senior project course. |

**4.5.2 Use Case Description**

Table 4-2: Register Use Case

|  |  |
| --- | --- |
| Use Case Name | Register |
| Actor | Coordinator, Supervisor, Student |
| Description | The actors (supervisor, student) register into system then the home page are display. |
| Precondition | Correct information. |
| Post-condition | Home page is display. |

Table 4-3: Login Use Case

|  |  |
| --- | --- |
| Use Case Name | Login |
| Actor | Coordinator, Supervisor, Student |
| Description | The actors (supervisor, student) log in into system then the home page are display. |
| Precondition | Register and have account. |
| Post-condition | Home page is display if the login is validate. |

Table 4-4: Manage files Use Case

|  |  |
| --- | --- |
| Use Case Name | Manage files. |
| Actor | Coordinator |
| Description | The coordinator can upload files (proposed idea, templates and guidelines). |
| Precondition | Log in |
| Post-condition | The page is upload file is display. |

Table 4-5: Manage the complete senior project Use Case

|  |  |
| --- | --- |
| Use Case Name | Manage the complete senior project. |
| Actor | Coordinator |
| Description | The coordinator can insert and update the complete senior project. |
| Precondition | Log in |
| Post-condition | Display page after insert and update the complete senior project. |

Table 4-6: View evaluation result Use Case

|  |  |
| --- | --- |
| Use Case Name | View evaluation result. |
| Actor | Coordinator |
| Description | The coordinator can view evaluation result of the current project. |
| Precondition | Log in |
| Post-condition | Send the evaluation result. |

Table 4-7: Monitor the progress of current project Use Case

|  |  |
| --- | --- |
| Use Case Name | Monitor the progress of current project. |
| Actor | Coordinator |
| Description | The coordinator can view information about the progress work of current projects, if there is delay in one group, the coordinator can send notification. |
| Precondition | Log in |
| Post-condition | List of the senior project along with progress work information is display. |

Table 4.8: Manage electronic schedule Use Case

|  |  |
| --- | --- |
| Manage electronic schedule. | Use Case Name |
| Coordinator | Actor |
| The coordinator can add, delete event to schedule. | Description |
| Log in | Precondition |
| The added event will displayed for all actors and the system triggers notification for each event. | Post-condition |

Table 4-9: Evaluate student Use Case

|  |  |
| --- | --- |
| Evaluate student under supervision. | Use Case Name |
| Supervisor | Actor |
| The supervisor can evaluate student (under supervision) based on specific form and submit the result. | Description |
| Log in | Precondition |
| If the file was send the system display successful message otherwise display resubmit request message. | Post-condition |

Table 4-10: Evaluate other student senior project Use Case

|  |  |
| --- | --- |
| Evaluate other student senior project. | Use Case Name |
| Supervisor(Committee) | Actor |
| The supervisor can evaluate student (student is not under supervision) based on specific form and submit the result. | Description |
| Log in | Precondition |
| If the file was send the system display successful message otherwise display resubmit request message. | Post-condition |

Table 4-11: Monitor the progress work of students Use Case

|  |  |
| --- | --- |
| Monitor the progress work of students. | Use Case Name |
| Supervisor | Actor |
| The supervisor can view and evaluate it weakly. | Description |
| Assign work for students. | Precondition |
| The follow up form is display and update weakly. | Post-condition |

Table 4-12: Submit assignment Use Case

|  |  |
| --- | --- |
| Submit assignment. | Use Case Name |
| Student | Actor |
| The student can send assignment through the system to supervisor. | Description |
| The supervisor should assign task as assignment for student in the system. | Precondition |
| If the file was send the system display successful message otherwise display resubmit request message. | Post-condition |

Table 4.13: View Event Use Case

|  |  |
| --- | --- |
| Use Case Name | View Event |
| Actor | Student |
| Description | The student can view important appointments. |
| Precondition | Login |
| Post-condition | List of the important appointments is display. |

Table 4.14: Brows available files Use Case

|  |  |
| --- | --- |
| Brows available files. | Use Case Name |
| Student | Actor |
| A student can view any files uploaded by coordinator. | Description |
| Login | Precondition |
| The page of uploaded files as displayed. | Post-condition |

Table 4.15: View/ Search completed senior project (repository) Use Case

|  |  |
| --- | --- |
| Use Case Name | View/ Reach completed senior project (repository) |
| Actor | Student |
| Description | The student can view information about the completed senior project. |
| Precondition | Existence of previous senior projects. |
| Post-condition | Page of completed senior project is display. |

Table 4-16: Manage out coming transaction Use case

|  |  |
| --- | --- |
| Use Case Name | Create out coming transaction |
| Actor | Supervisor, Student |
| Description | The actors(create, view) the out coming transaction |
| Precondition | Login |
| Post-condition | The out-coming transaction is display, so the actor can view existing out-coming and create new one. |

Table 4-17: Manage in-coming transaction Use Case

|  |  |
| --- | --- |
| Use Case Name | Manage in-coming transaction |
| Actor | Coordinator, supervisor, student |
| Description | The actors(view, reply)the in-coming transaction |
| Precondition | Login |
| Post-condition | The in-coming transaction is display, so the actor can view existing incoming and reply new one. |

Table 4-18: Send/Receive IM Use Case

|  |  |
| --- | --- |
| Use Case Name | Send/Receive IM |
| Actor | Coordinator, supervisor, student |
| Description | One actor can start chatting with other actor and send/receive messages. |
| Precondition | Login |

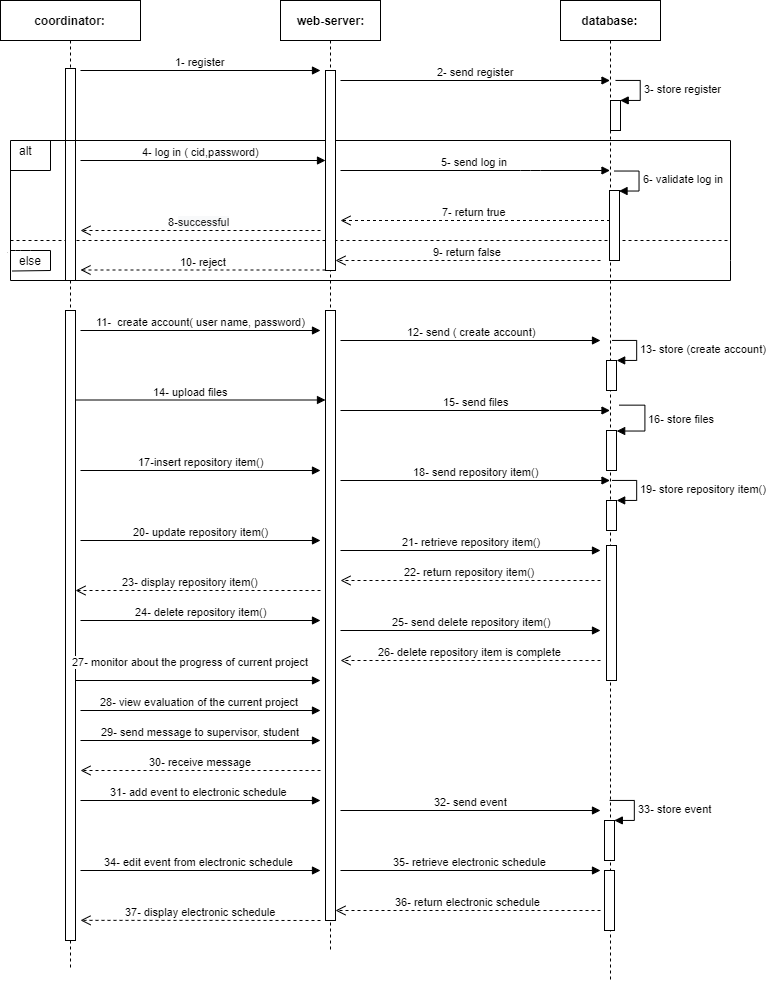
**4.6 Analysis Phases**

The analysis phase defines the requirements of the system, independent of  
how these requirements will be accomplish. The process will be describe using sequence diagram and activity diagram.

**4.6.1 Sequence Diagram**

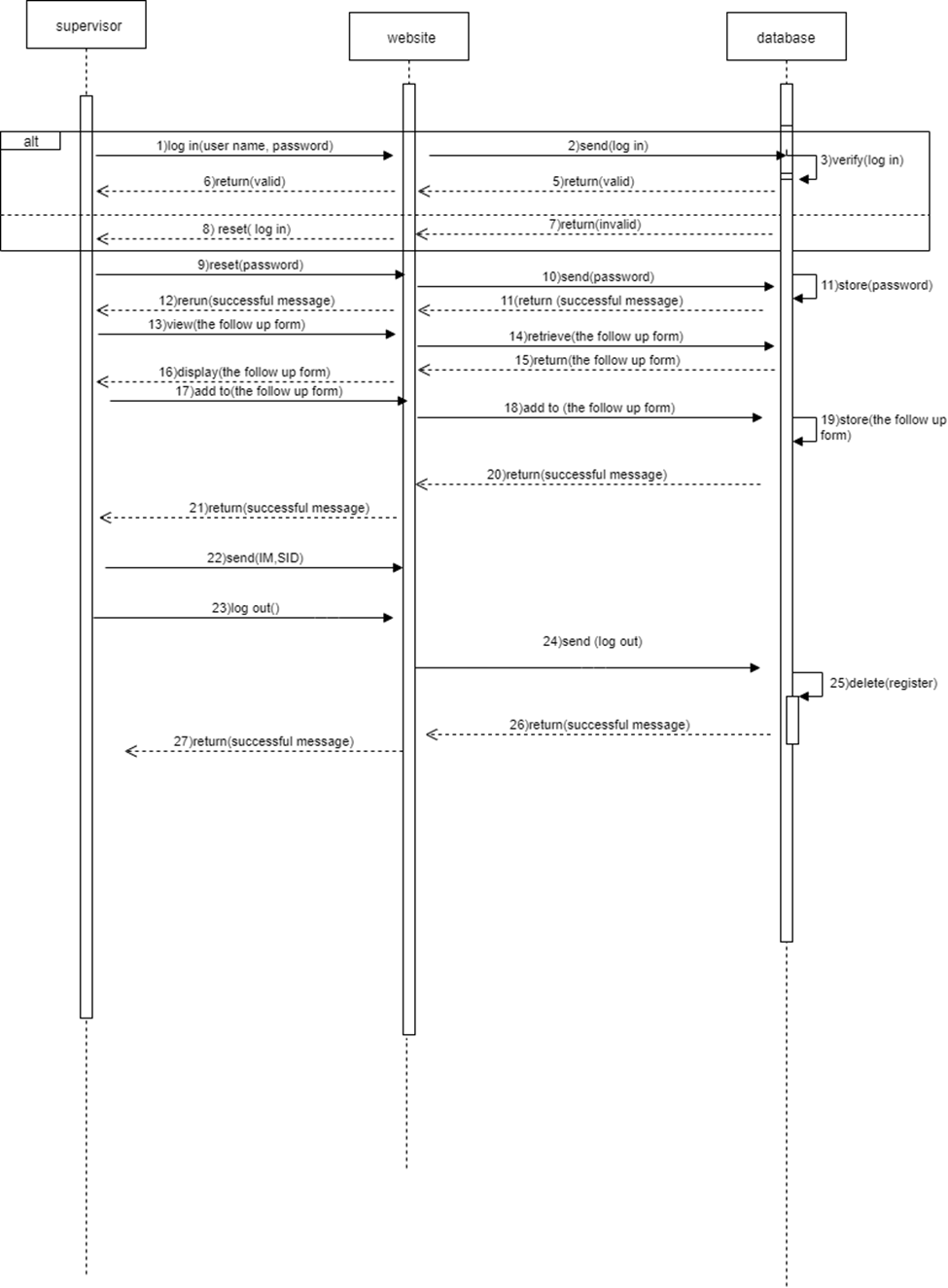
A sequence diagram shows object interactions arranged in time sequence. The sequence diagram for some operations of our web site are illustrated in figure 4.4-4.12

Coordinator sequence diagram:



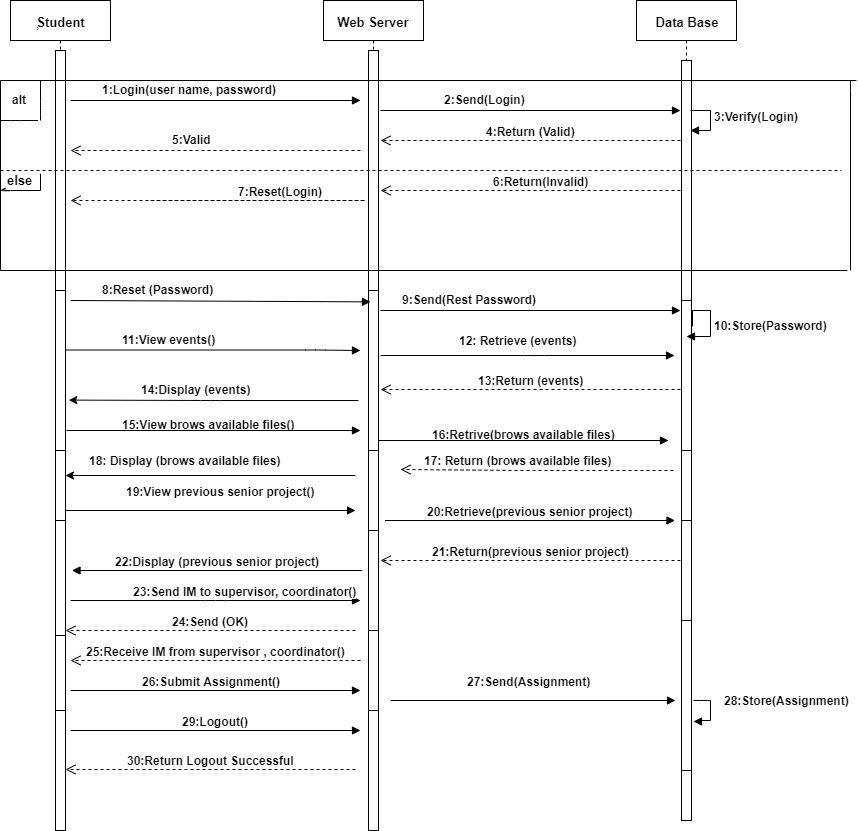
**Figure 4.4** Coordinator Sequence Diagram

Supervisor sequence diagram:



**Figure 4.5** Supervisor Sequence Diagram

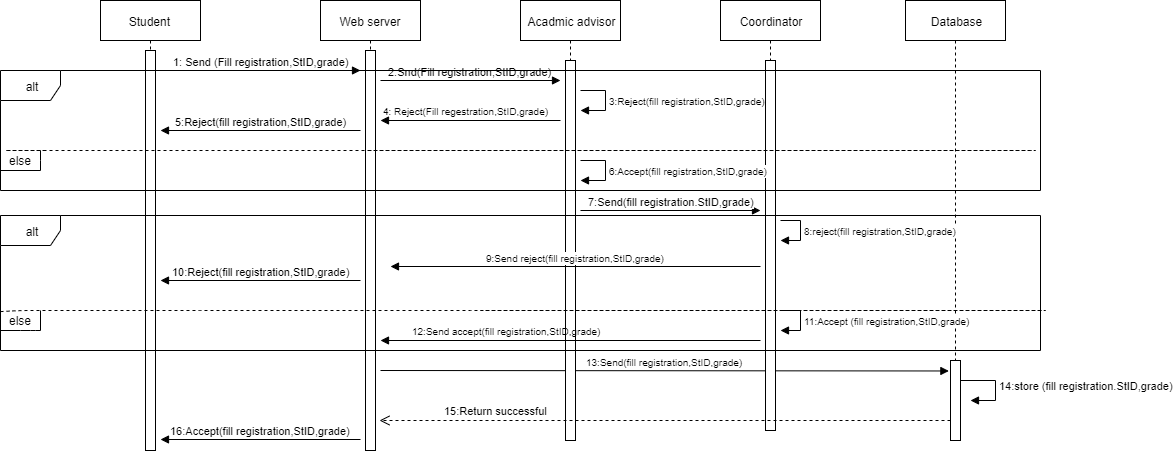
Student sequence diagram:



**Figure 4.6** Student sequence diagram

Fill registration transaction by student:

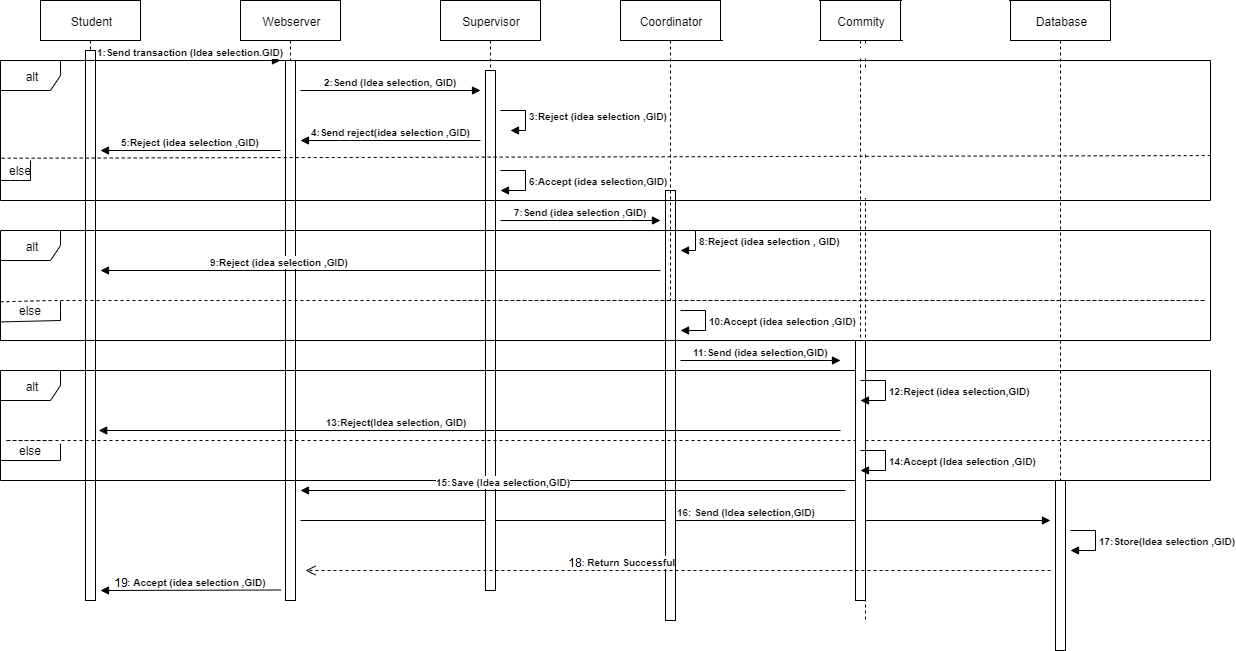
\* Academic advisor he is one of the supervisors registered in the system



**Figure 4.7** Fill registration transaction

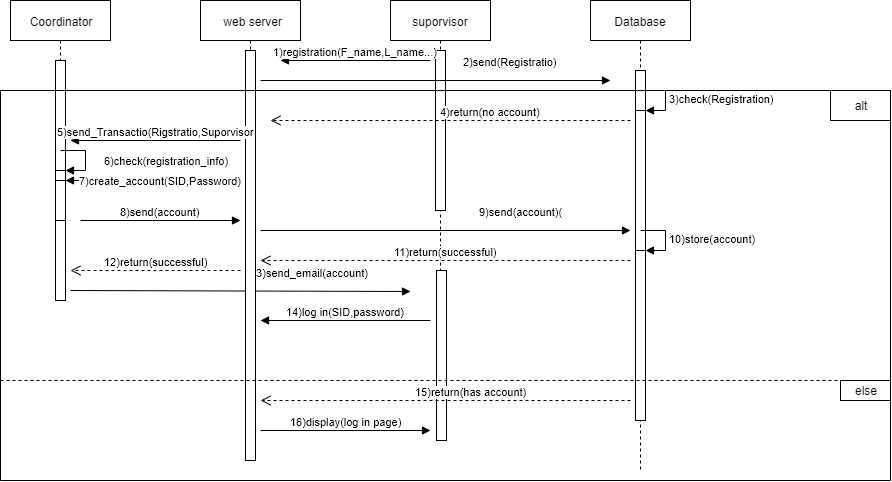
Idea selection transaction by student:

\* Committee he is one of the supervisors registered in the system



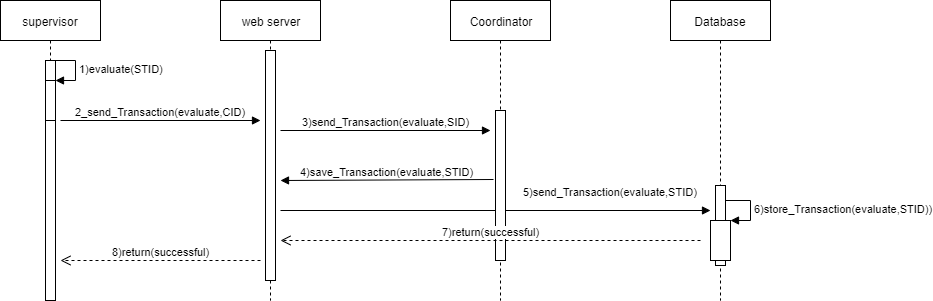
**Figure 4.8** Idea selection transaction

Registration Transaction (supervisor) sequence diagram:



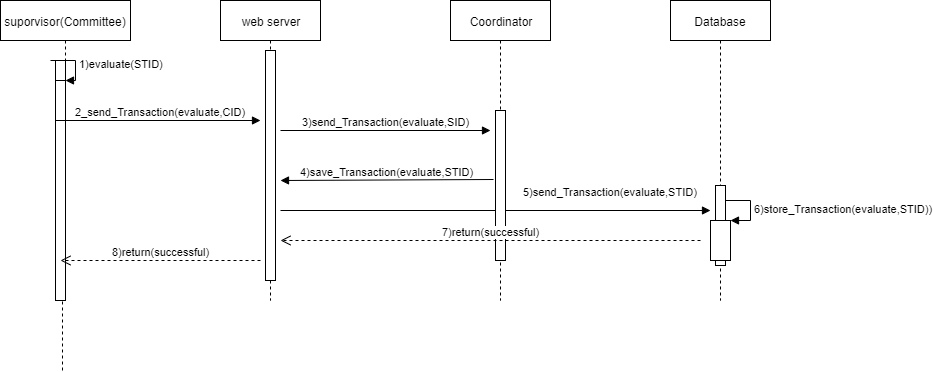
**Figure 4.9** Registration transaction (supervisor)

Evaluation\_COIT498 (supervisor) Transaction sequence diagram:



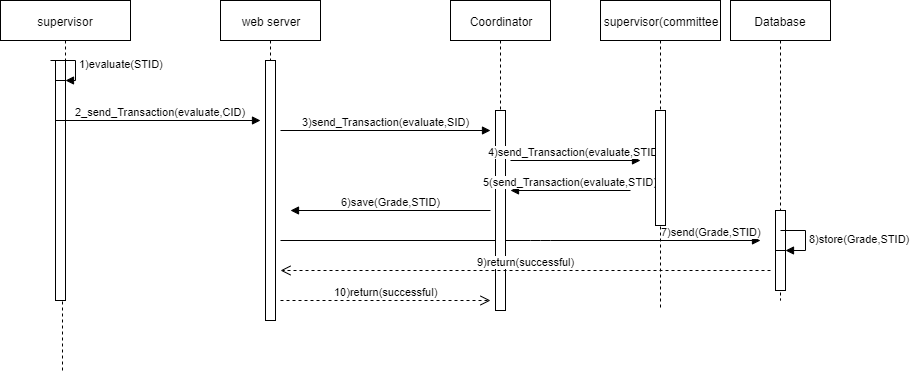
**Figure 4.10** Evaluation\_COIT498 (supervisor) transaction

Evaluation\_COIT498 (committee) Transaction sequence diagram:



**Figure 4.11** Evaluation\_COIT498 (committee) transaction

Final evaluation Transaction sequence diagram:

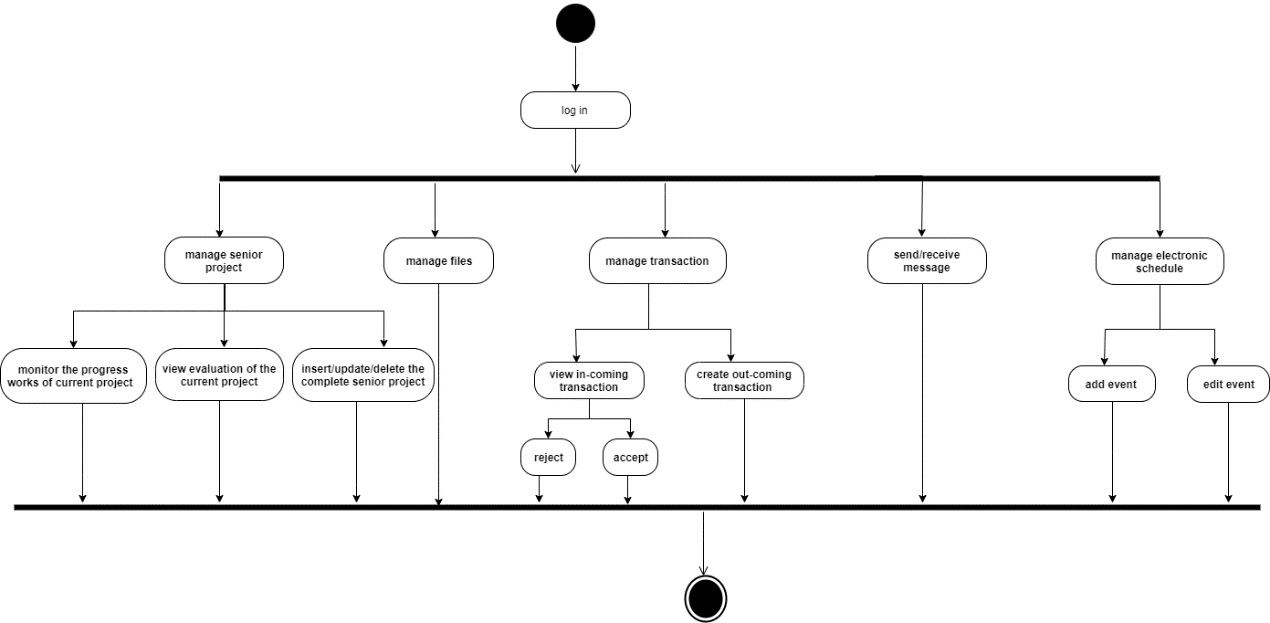


**Figure 4.12** Final evaluation transaction

**4.6.2 Activity Diagram**

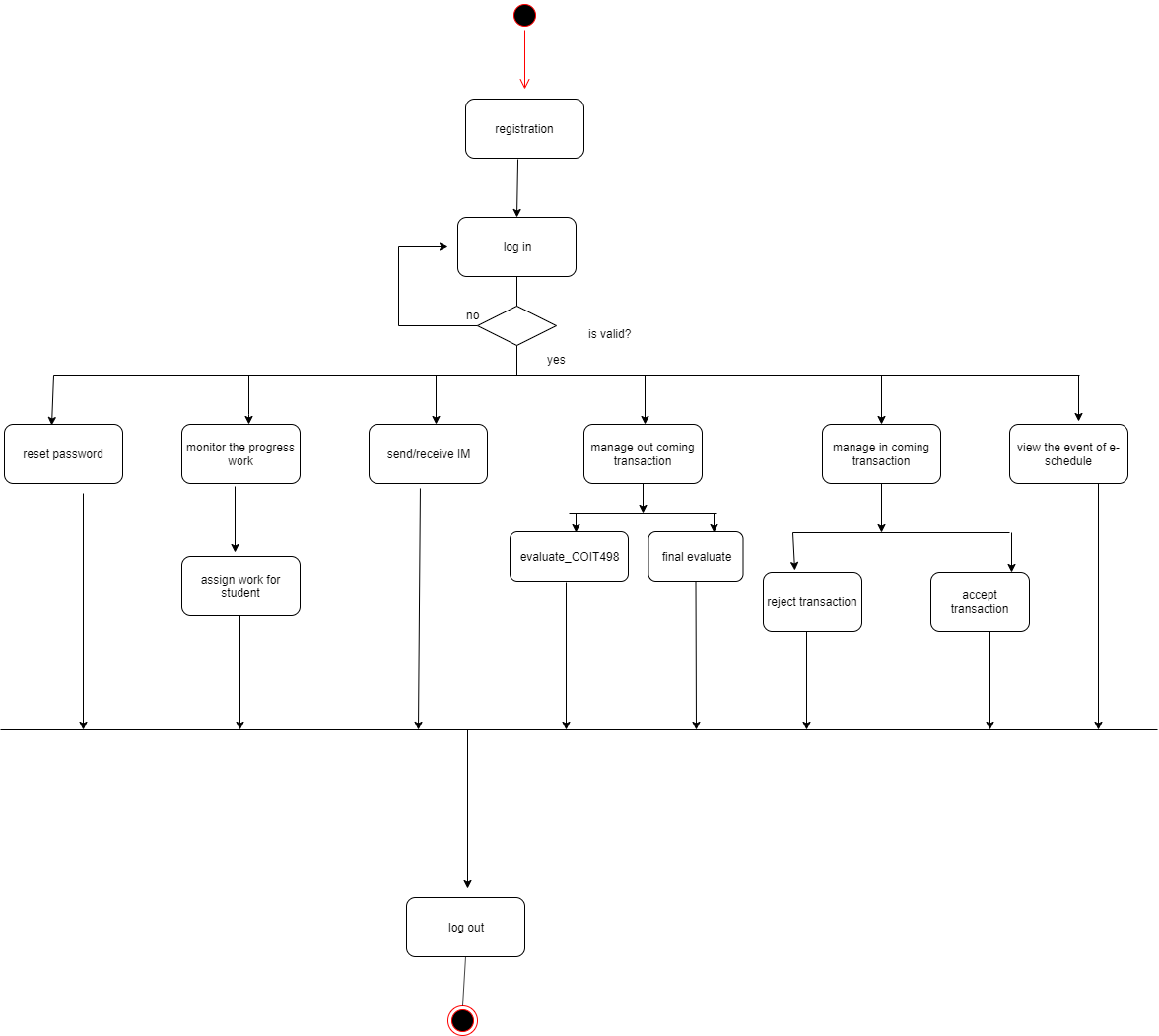
Activity diagram are graphical representation of the process and workflow for activities in system developed. The activity diagram for some operations of our web site are illustrated in figure 4.13-4.15

Coordinator activity diagram:



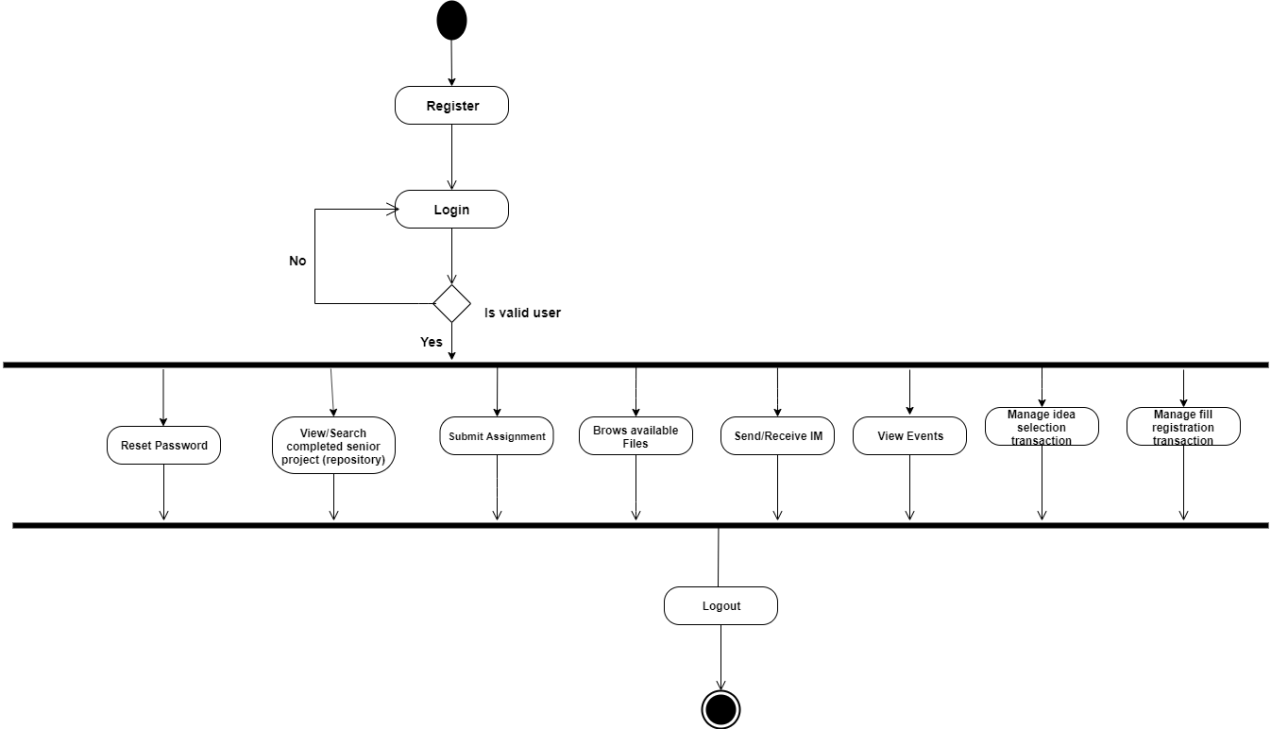
**Figure 4.13** Coordinator activity diagram

Supervisor activity diagram:



**Figure 4.14** Supervisor activity diagram

Student activity diagram:



**Figure 4.15** Student activity diagram

**4.7 System Architecture and Design**

Figure 4.16 shows the architecture of our system and it works according to the following steps:

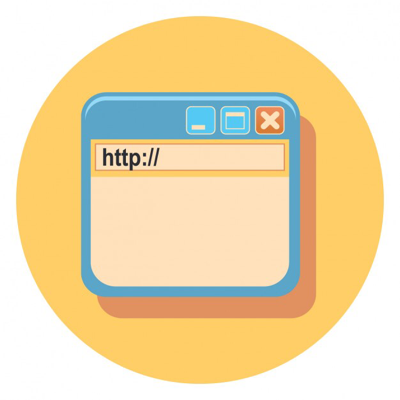
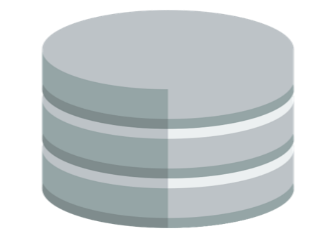
1- The client (browser) send request to ASP.NET web server (website) run at Apache server.

2- The web server process the request and connect to the MySQL database to read or write in database.

3- The MySQL server return the data to the web server.

4- The web server prepare data as HTTP document response and send it back to the browser.

5- The client (browser) reserve HTTP document and deal with it.



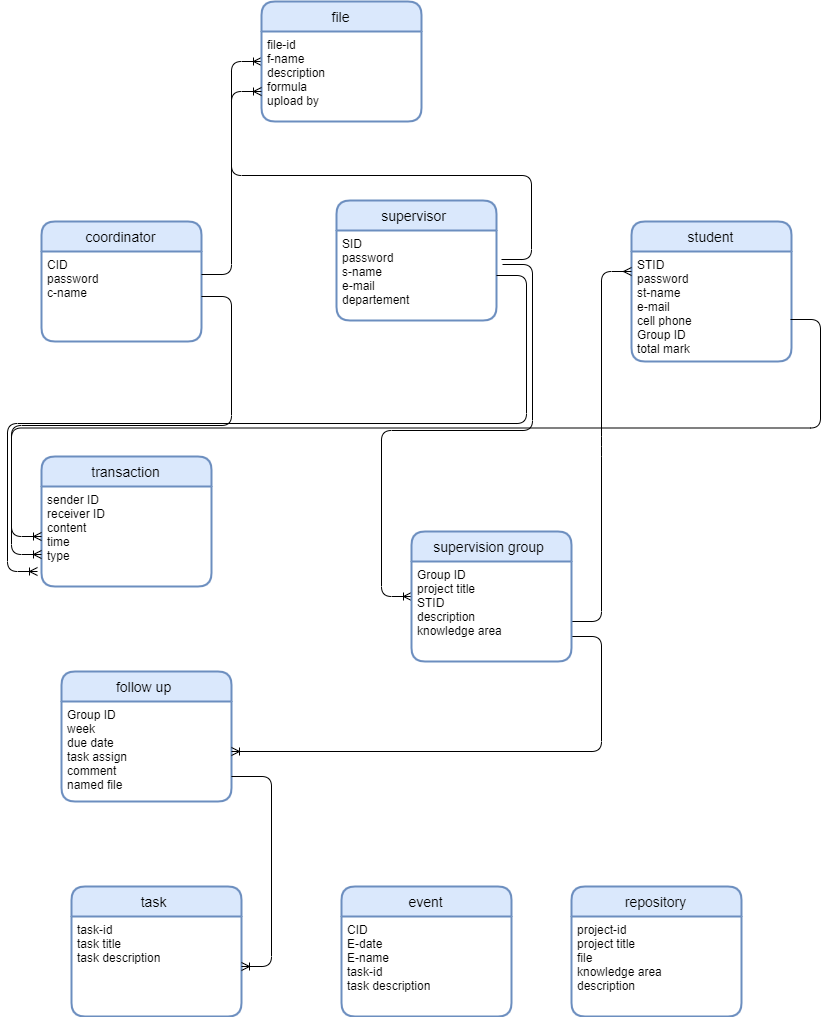
Browser ASP.NET web server (website) MySQL

**Figure 4.16** The architecture of our system

**4.8 System Database Design**

The proposed system needs database to store a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a Data Definition Language. The design is use to identify relationship between the modules or tables [31].

Figure 4.17 shows the database diagram of our proposed system.

****

**Figure 4.17** Database diagram

The details of the database tables are given in the tables 4-19 to 4-28.

Table 4-19: Coordinator

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| CID (PK) | Int (11) | NO |  |
| Password | Varchar (60) | No |  |
| C-name | Varchar (30) | NO |  |

Table 4-20: Supervisor (committee)

|  |  |  |  |
| --- | --- | --- | --- |
| **Default** | **Null** | **Type** | **Column** |
|  | No | Int(11) | SID |
|  | No | Varchar (30) | S-name |
|  | No | Varchar(30) | Password |
|  | No | Varchar(50) | E-mail |
|  | No | Varchar(30) | Department |

Table 4-21: Student

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| STID (PK) | Int (11) | NO |  |
| Password | Varchar(60) | NO |  |
| St-name | Varchar(30) | NO |  |
| E-mail | Varchar(20) | NO |  |
| Cell-phone | Varchar(15) | NO |  |
| Group ID (FK) | Int (11) | NO |  |
| Total-mark | Int(11) | No |  |

Table 4-22: Supervision group

|  |  |  |  |
| --- | --- | --- | --- |
| **Default** | **Null** | **Type** | **Column** |
|  | No | Int(11) | Group\_ ID |
|  | No | Varchar(30) | Project\_ Title |
|  | No | Int(11) | STID |
|  | No | Varchar(100) | Description |
|  | No | Varchar(100) | Knowledge Area |

Table 4-23: Follow up

|  |  |  |  |
| --- | --- | --- | --- |
| **Default** | **Null** | **Type** | **Column** |
|  | No | INT(11) | Group\_ ID |
|  | No | Date() | Weak |
|  | No | Date Time() | Due\_ Date |
|  | No | PDF | Task Assign |
|  | No | Int(11) | STID |
|  | No | Varchar(100) | Comment |

Table 4-24: File

|  |  |  |  |
| --- | --- | --- | --- |
| **Default** | **Null** | **Type** | **Column** |
|  | No | Int(11) | File-id |
|  | No | Char(50) | F\_name |
|  | No | Varchar(100) | Description |
|  | No | Int(11) | Upload by |

Table 4-25: Transaction

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| Sender ID | Int(11) |  |  |
| Receiver ID | Int(11) |  |  |
| Content | Varchar(30) |  |  |
| Time | Date Time |  |  |
| Type | Varchar(30) |  |  |

Table 4-26: Repository

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| Project title | Varchar(30) | NO |  |
| Project-id | Int(11) | NO |  |
| File | PDF | NO |  |
| Description | Varchar(30) | NO |  |
| Knowledge area | Varchar(30) | NO |  |

Table 4-27: Event

|  |  |  |  |
| --- | --- | --- | --- |
| **Default** | **Null** | **Type** | **Column** |
|  | No | int(11) | CID |
|  | No | DateTime() | E\_Date |
|  | No | Char(50) | E\_name |
|  | No | Int(11) | Task-id |
|  | No | Varchar(30) | Task description |

Table 4-28: Task

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Type** | **Null** | **Default** |
| Task-id | Int(11) | No |  |
| Task title | Varchar (30) | No |  |
| Task description | Varchar (30) | No |  |

**4.9 Conclusion**

In this chapter we determined system functional, nonfunctional and

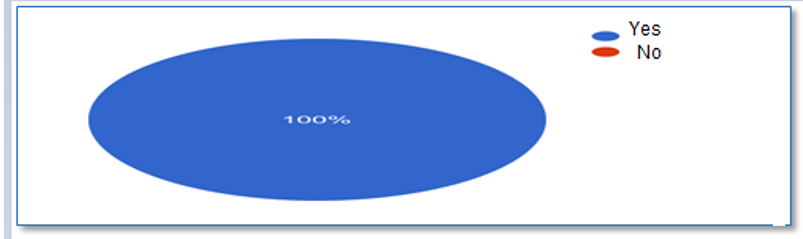
Interface requirements. In addition, we draw the important UML diagrams such as use case, sequence and activity diagrams. In addition, we design the architecture and database of our proposed system.

**Appendix (A)**

This questionnaire for supervisors:

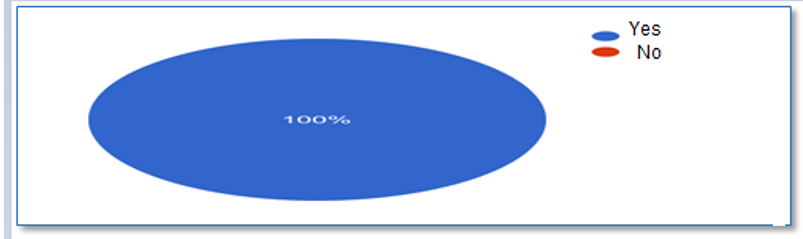
Do you want the evaluation forms to be provided electronically? (Not as printed paper)

* Yes
* No



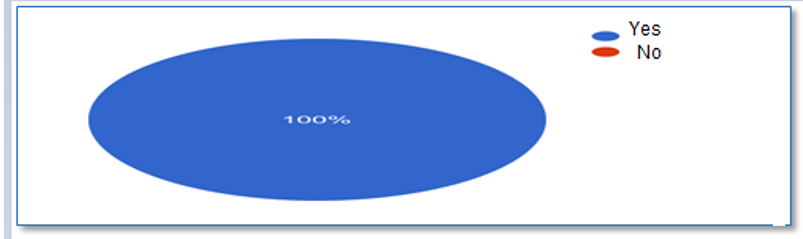
Do you need to do the weekly progress monitoring through a web-based system?

* Yes
* No



Do you prefer to evaluate the student weekly tasks using an electronic form shared between you and your students?

* Yes
* No



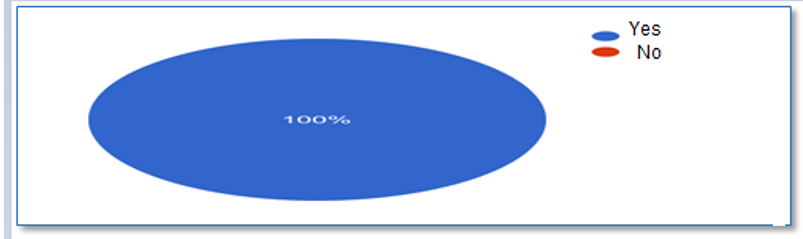
What are the other electronic services that you need as a project supervisor in our proposed system?

**Appendix (B)**

أسئلة الاستبيان الخاص بطلاب التخرج:

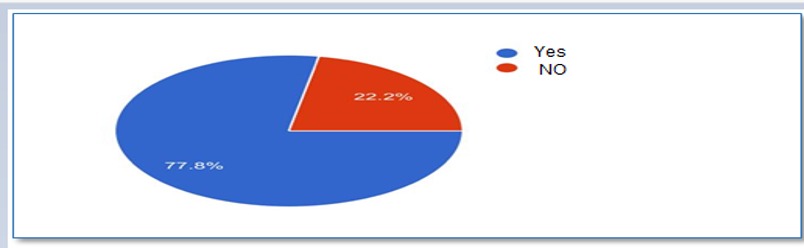
هل وجدت صعوبة في أثناء بحثك عن فكرة لمشروع تخرجك؟

* نعم
* لا



هل تفضل أن تقدم فكرة نموذجية لمشروعك إلكترونيا؟

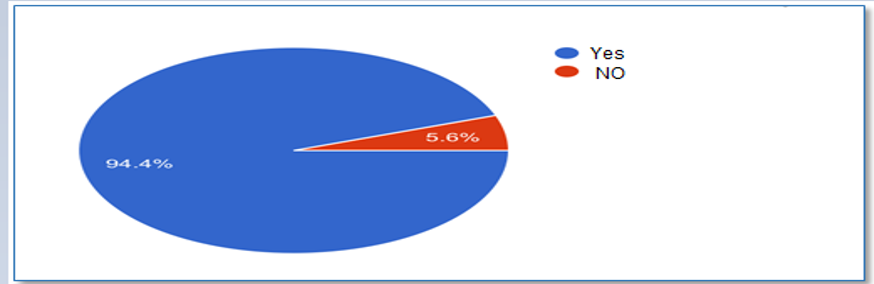
* نعم
* لا



ماهي المشاكل التي واجهتك بعد تقديم فكرة مشروعك؟

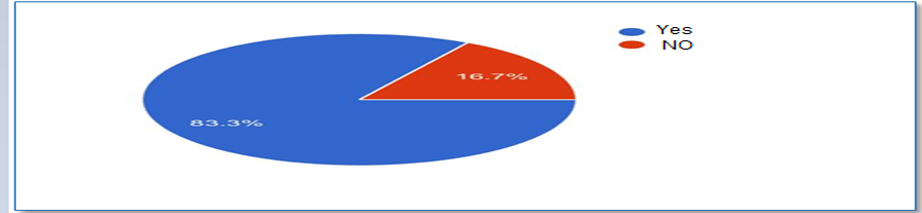
هل تعتقد أن إنشاء موقع لمشاريع التخرج السابقة يوفر عليك الكثير من الوقت والجهد في البحث عن الأفكار المشابهه لمشروعك؟

* نعم
* لا



هل تفضل أن يسمح لك الموقع بالبحث عن المشاريع السابقة بكلمات استدلالية؟

* نعم
* لا



ماهي الخدمات التي ترغب أن تكون موجودة في الموقع؟

**Reference**

[1] Senior project ,''<https://www.bcsc.k12.in.us/domain/230>", [Online]: Available: <https://www.bcsc.k12.in.us/domain/230>, [Access: Oct.8.2019].

[2] Norah Princess University (computer faculty)," www.pnu.edu.sa/arr/Faculties/ComputerScience ", [Online]: Available: <http://www.pnu.edu.sa/arr/Faculties/ComputerScience/Pages/GGP/Booklet.aspx>, [Access: Oct.8.2019].

[3]:Whitman collage ," www.whitman.edu ", [Online]: Available: <https://www.whitman.edu/academics/departments-and-programs/mathematics-and-statistics/resources-for-students/senior-project-archive>

, [Access: Oct.8.2019].

[4]: Cal poly,"digitalcommons.calpoly.edu ", [Online]: <https://digitalcommons.calpoly.edu/seniorprojects/>, [Access: Oct.8.2019].

[5]: Allegheny collage, " sites.allegheny.edu ", [Online]: Available: <https://sites.allegheny.edu/lits/digitizedcollections/repository-senior-projects/>, [Access: Oct.8.2019].

[6]: Bard undergraduate senior project, " drexel.edu/engineering/academics ", [Online]: Available: <https://drexel.edu/engineering/academics/departments/electrical-computer-engineering/resources/current-undergrad/senior-design/project-archive/>

, [Access: Oct.8.2019].

[7]: Casual, " <https://itconsultingonline.net/>" , [Online]: <https://itconsultingonline.net/>, [Access: Oct.8.2019].

[8]: Basecamp, " <https://itconsultingonline.net/>" , [Online]: <https://itconsultingonline.net/>, [Access: Oct.8.2019].

[9]: Trello, " <https://itconsultingonline.net/>" , [Online]: <https://itconsultingonline.net/>, [Access: Oct.8.2019].

[10]: Agile Software Development , "<https://ndiastorage.blob.core.usgovcloudapi.net/>", [Online]: Available: <https://ndiastorage.blob.core.usgovcloudapi.net> , [Access:Oct.8.2019].

[11]: Agile Software Development , "<https://ndiastorage.blob.core.usgovcloudapi.net/>", [Online]: Available: <https://ndiastorage.blob.core.usgovcloudapi.net> , [Access:Oct.8.2019].

[12]: Agile Software Development , "<https://ndiastorage.blob.core.usgovcloudapi.net/>", [Online]: Available: <https://ndiastorage.blob.core.usgovcloudapi.net> , [Access:Oct.8.2019].

[13]: Agile Software Development , "<https://ndiastorage.blob.core.usgovcloudapi.net/>", [Online]: Available: <https://ndiastorage.blob.core.usgovcloudapi.net> , [Access:Oct.8.2019].

[14]: Agile Software Development , "<https://ndiastorage.blob.core.usgovcloudapi.net/>", [Online]: Available: <https://ndiastorage.blob.core.usgovcloudapi.net> , [Access:Oct.8.2019].

[15]: Agile Software Development , "<https://ndiastorage.blob.core.usgovcloudapi.net/>", [Online]: Available: <https://ndiastorage.blob.core.usgovcloudapi.net> , [Access:Oct.8.2019].

[16]: Agile Software Development , "<https://ndiastorage.blob.core.usgovcloudapi.net/>", [Online]: Available: <https://ndiastorage.blob.core.usgovcloudapi.net> , [Access:Oct.8.2019].

[17]: Agile Software Development , "<https://ndiastorage.blob.core.usgovcloudapi.net/>", [Online]: Available: <https://ndiastorage.blob.core.usgovcloudapi.net> , [Access:Oct.8.2019].

[18]: Guru99 , "[https://www.guru99.com/agile-scrum-extreme-testing.html"](https://www.guru99.com/agile-scrum-extreme-testing.html%22), [Online]: Available: <https://www.guru99.com/agile-scrum-extreme-testing.html>, [Access:Oct.8.2019].

[19]: Guru99 , "[https://www.guru99.com/agile-scrum-extreme-testing.html"](https://www.guru99.com/agile-scrum-extreme-testing.html%22), [Online]: Available: <https://www.guru99.com/agile-scrum-extreme-testing.html>, [Access:Oct.8.2019].

[20]: Visual studio, <https://visualstudio.microsoft.com/vs/>, [Online]: Available: <https://visualstudio.microsoft.com/vs/> [ Access: Nov.8.2019].

[21]: Html," <https://www.quora.com/>", [Online]: Available: <https://www.quora.com/>, [ Access: Nov.8.2019].

[22]: c#, " <https://www.quora.com/>", [Online]: Available: <https://www.quora.com/> [Access: Nov.8.2019].

[23]: ASP.NET , " dotnet.microsoft.com ", [Online]: Available: <https://dotnet.microsoft.com/learn/aspnet/what-is-aspnet> , [Access: Nov.8.2019].

[24]: ASP.NET , " www.geeksforgeeks.org ", [Online]: <https://www.geeksforgeeks.org/introduction-to-asp-net/> , [Access: Nov.8.2019].

[25]: ASP.NET , " www.hostgator.com ", [Online]: https://www.hostgator.com/blog/what-is-asp-net-hosting/, [Access: Nov.8.2019].

[26]: ASP.NET , " www.itegritygroup.com ", [Online]: http://www.itegritygroup.com/asp-net-advantages, [Access: Nov.8.2019].

[27]: MySql, <https://www.mysql.com/>, [Online]: Available: <https://www.mysql.com/>, [Access: Nov.8.2019].

[28]: Requirements Experts Grady , "<https://reqexperts.com/>" , [Online]: Available: <https://reqexperts.com/> [Access:Oct.10.2019].

[29]: Requirements Experts Grady , "<https://reqexperts.com/>" , [Online]: Available: <https://reqexperts.com/> [Access:Oct.10.2019].

[30] : Use case , " <https://www.usability.gov/>" ,[Online]: Available: <https://www.usability.gov/> [Access:Nov.8.2019].

[31]: Database Design, "<https://www.guru99.com/>", ,[Online]: Available: <https://www.guru99.com>, [Access:Nov.8.2019].