CLOUD-BASED WEB APPLICATION SYSTEM FOR CAPSTONE PROJECT MONITORING, MANAGING, AND EVALUATING WITH PLAGIARISM CHECKER

A Capstone Project

Presented to the Faculty of

SORSOGON STATE UNIVERSITY

Bulan Campus

Bulan, Sorsogon

In Partial Fulfillment of the

Requirements for the Degree

Bachelor of Science in Information Technology

BY

JOHN MATHEW D. DIÑO

RAMIL G. BUEN

KASANDRA G. ALVAREZ

ROLLY L. GILOS JR.

MAY 2021

EXECUTIVE SUMMARY

Capstone Project is a partial requirement undertaken by student taking Computer Studies degree programs like BSIT and BSIS. Students and faculty members face problems in submitting, managing deliverables, and monitoring progresses when using manual or semi-computerized method. With the rise of technologies, students are expected to be knowledgeable and be able to experience the benefits of new technologies. Sorsogon State University – Bulan Campus, specifically the ICT Department, experiences these problems and does not have an online centralized system for the various activities involved in completing capstone projects. With that in mind, this project entitled "Cloud-Based Web Application for Capstone Project Monitoring, Managing, and Evaluating System with Plagiarism Checker" aimed to assist and streamline the current Capstone Process between Department Dean, Capstone Professor, Students, and their adviser and panelists. The system aimed to (1) carry out and deliver the following features and modules related to monitoring, managing, and evaluating capstone projects electronically: (1.1) Capstone Project manuscript progress monitoring, (1.2) Capstone Project topic and title proposal, (1.3) students' adviser and panelist selection, (1.4) students' task and activities management, (1.5) deliverable's submission and management, (1.6) presentation and defense scheduler, (1.7) manuscript evaluation, (1.8) user's management, (1.9)student's grades management,

(1.10) student management, (1.11) Capstone Project guide and formatting, and (1.12)discussion forum; (2) integrate an existing plagiarism checker that will check the manuscript for plagiarism, locate it, and report via the percentage of the amount of plagiarized content; and (3) Test and evaluate the proposed project, based on ISO/IEC 25010 Software Product Quality, in terms of: (3.1) functional stability, (3.2) performance efficiency, (3.3) compatibility, (3.4) usability, (3.5) reliability, (3.6) security, (3.7) maintainability; and (3.8) portability of the system. In the span of two semesters, the proponents will develop the system using Iterative approach for the development life cycle where the flow of development focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete; Object-Oriented for the analysis and design applying object-oriented programming; and Bottom-up as the development approach. The system will be verified, validated, and tested using client and users survey and interview, functional and non-functional validation, and positive and negative testing.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER I

Introduction

This chapter will discuss the context of the project, its purpose and description, objectives, and its scope and limitations, aiming to explain the concept and the importance of the current project.

1.1 Project Context

Computer Studies degree programs – Bachelor of Science in Information Technology, Information System, and Computer Science, usually includes a capstone project as a partial requirement, involving two (2) to four (4) members, and are undertaken in the final year of study in which students conduct a software development project [5, 20, 24, 26, 40]. Students will apply and demonstrate the concepts and the comprehensive knowledge they've learned throughout their study with this project. It allows students to improve their current skills and acquire new ones and gain real-world experience from this "realistic experiential learning" [13, 16, 20, 24, 26, 33].

In undertaking and managing Capstone Projects, students and faculty members may encounter problems, such as project completion delays, inferior quality products, unstructured development processes, and overall dissatisfaction

in students and advisory [24, 25, 30]. In many instances, students face a hard time deciding on the topic they want to work on and even where to start [28]. Capstone advisers, on the other hand, also face a problem with monitoring and managing capstone projects. They sometimes receive a lack of information about the students' progress and may also experience difficulties in handling the documentation or deliverables because of the variety of tools used [24]. With the advancement of technology, students are expected to be knowledgeable about new technologies. With capstone projects implemented in a manual or semi-computerized manner, even when having various technologies that can ease tasks, students cannot showcase their knowledge and learn the benefits the present technologies can provide [27, 41]. Finally, although they can use various technologies, the lack of a centralized system can make it difficult to collaborate, review, monitor, manage, and even evaluate the project development and deliverables [22].

Sorsogon State University – Bulan Campus was one of the Tertiary Schools in the province of Sorsogon that does not have or uses Capstone Project, Thesis, or other project management systems in a single venue. They used a semi-computerized manner of creating Capstone Projects and integrated different technology in managing them. Some of the technologies they used were Google Drive, Google Spreadsheets, Email, and Facebook Messenger for communication

but used no centralized system to monitor the progress of the students nor an easier way of proposing a title, selecting their preferred adviser or panelists, or evaluate a paper in an online environment.

In an attempt to alleviate the problems stated above and to improve capstone projects, this project called Cloud-Based Web Application System for Capstone Project Monitoring, Managing, and Evaluating with Plagiarism Checker is an online system that aims to assist and streamline the current Capstone Process between the students, Capstone Professor, Capstone Adviser, and their panelists in a single venue. This project focuses on the title/topic proposal, selecting a capstone adviser and panelist, managing the project deliverables, monitoring the group progress, checking for plagiarism, and evaluating the submitted manuscript.

The system provided a separate interface for its three different users – the students, the faculty, and the dean. All of the users can access the system by providing necessary information in the login module. After logging in, the dean, as the super admin of the system, can have full access to the system, granting them the ability to manage the users where they can register new faculty, grant or remove privileges, assign a faculty that will be the Capstone Professor, and limit the number of students the faculty will be able to handle as supervisor. Furthermore, the faculty is subdivided into the capstone professor, who will also

act as the admin granted by the super admin, and the other faculty, who can be the adviser and/or the panelist. The capstone professor can have additional permission to manage capstone projects such as manage students' groups, set and edit manuscript format, view and print the students' grades and other reports, schedule the date and time of the defense, or evaluate the students' manuscript. Additionally, the capstone professor, together with the other faculty, can manage and assess the deliverables sent by the students, accept or decline the proposed title/topic, and request as adviser or panelist. On the other hand, the students can manage their group, select topic/title, adviser, and panelist, and submit their deliverables.

In addition, the system has an integrated plagiarism checker existing online that will help the students check and locate plagiarism in their manuscript to avoid and edit it if they unintentionally plagiarize other works. The plagiarism checker, accessible to all the users, will provide reports on the percentage, location, and content plagiarized.

1.2 Purpose and Descriptions

The purpose of this study was to develop a cloud-based web application system for a capstone project monitoring, managing, and evaluating with plagiarism checker to help students and faculty members of Sorsogon State University –

Bulan Campus under Information and Communication Technology Department.

The system mainly allow the faculty members to monitor the students' progress, manage deliverables, and evaluate manuscripts. The web application can also allow students to create groups, create discussion forums, and select their adviser, panelists, and capstone project title.

This system also has a plagiarism checker feature using external integrated software which scans the manuscript and compares its components to other works using a search engine. This function ensures the reliability and originality of the project.

In addition to the features of the application, the system will be reliable, performance efficient, usable, secured, compatible, portable, and can be maintained. This is to add assurance that the system's quality is satisfactory and aligned to the software standard.

The system's being centralized, has no similar system that are being used by the target users, and its ability to ease the process in monitoring, managing, and evaluation capstone project manuscripts with available manuscript plagiarism checker, adds to its uniqueness. The features and advantages that the system can provide can be beneficial the following:

Students. They can quickly generate and organize their capstone project using this system. They'll be able to select the topic or title they want to work in,

their preferred adviser and panelist, submit deliverables, monitor their progress, and view their grades.

Capstone Professors. This system can act as a single venue for monitoring, management, and evaluation of capstone projects which can benefit Capstone Professor by having one system that can assist him/her from the start of the project (grouping the students) up to the end (manuscript evaluation).

Faculty. The faculty, regardless of their role (adviser/panelist), can benefit from this system by being able to monitor and manage their advisory group/s easily and/or comment and evaluate their manuscript.

Department Dean. The system can help the dean of the department easily manage the faculty eligible to advise and/or become an evaluator of capstone projects. This system also enables the dean to select the faculty who will become the Capstone Professor, granting him/her additional privileges and control in the Capstone Project Management.

Future system developers. This project can help prospective developers who will conduct studies related to this system. They can make the present system their reference in improving monitoring, managing, and evaluating capstone projects.

1.3 Objectives of the Study

This proposed capstone project aims to design and develop a cloud-based web application for monitoring, managing, and evaluating capstone projects with an external integrated plagiarism checker. Specifically, it seeks to:

1. Carry out and deliver the following features and modules related to monitoring, managing, and evaluating capstone projects electronically:

    1.1. Capstone Project manuscript progress monitoring

    1.2. Capstone Project topic or title proposal

    1.3. Students' Adviser and Panelists selection

    1.4. Students' Task and Activities management

    1.5. Deliverable's submission and management

    1.6. Presentation and Defense Scheduler

    1.7. Manuscript evaluation

    1.8. User's management

    1.9. Students' Grades management

    1.10. Student management

    1.11. Capstone Project Guide and Formatting

1.12. Discussion Forum

2. Integrate an existing plagiarism checker that will check the manuscript for plagiarism, locate it, and report via the percentage of the amount of plagiarized content.

3. Test and evaluate the proposed project, based on ISO/IEC 25010 Software Product Quality, in terms of:

   3.1. Functional Stability;

   3.2. Performance Efficiency;

   3.3. Compatibility;

   3.4. Usability;

   3.5. Reliability

   3.6. Security;

   3.7. Maintainability; and

   3.8. Portability of the system

1.4 Scope and Limitations

This study entitled "Cloud-based Web Application for Monitoring, Managing, and Evaluating Capstone Projects with Plagiarism Checker" focuses on assisting

students and faculty members on their capstone projects which concentrate on the following capstone project process: the title/topic proposal, selecting a capstone adviser and panelist, managing the project deliverables, monitoring the group progress, checking for plagiarism and evaluating the submitted manuscript.

This system is intended to be used and tested by the Faculty and Students of Sorsogon State University – Bulan Campus Information and Communication Technology Department for managing, monitoring, evaluating, checking plagiarism of Capstone Projects.

The primary users of this system are composed of students, faculty members, a capstone professor or instructor that will act as administrator, and the department dean as the super administrator. The proposed project can handle tertiary Capstone and Thesis projects; however, it may also include Secondary Level Researchers and other project-based activities that follow the project process stated above.

The system does not include other functions like a recommender system that can automatically recommend titles or topics to students, an archiving feature that stores past written projects of students, nor a chatting feature that will enable direct messaging between the students and their adviser. The complexity and the limited time frame on creating the system is the reason for the exception of the stated functions.

The plagiarism detection capabilities of the system depend on the capabilities, specifically on its speed and reliability, of the external plagiarism detector integrated into the system. Since the current system will only integrate an existing plagiarism checker service, the limitation of that service will also apply to the current system.

The project is limited to a website application running on a Windows Operating System and does not include a mobile application or desktop application. The skills of the proponents are limited on website development therefore the current system will only be limited to a web application and will not be available as a mobile nor desktop application. The system will also be limited to the specific operating system because it will be tested and deployed with the use of the same operating system. However, the system's design responsiveness will able mobile users to access it with the use of their browsers.

Moreover, the following are the scope and limitations of the features and modules of the system:

- The Capstone Project Manuscript Progress Monitoring allows the faculty to know the progress of the student's manuscript which will be dependent on the submissions of the deliverables whether they already submitted or not.

- The Capstone Project Topic and Title Proposal feature of the system will

allow the students to submit Capstone Project title or topics together with its descriptions which will able the faculty suggest or accept it.

- The Students' Adviser and Panelist Selection module will use the list of the registered faculty members on the system so that students can select their adviser and panelist. The faculty member on the other hand, will receive a list of requests and be able to accept the request or decline.

- The Students' Task and Activities Management enables the faculty members, specifically the course professor, to create tasks with submission forms, set its deadlines, and other announcement.

- The Deliverable's Submission and Management allows the student to submit their deliverables on the submission forms created by the course professor. On the other hand, the faculty can view, manage, and comment on that submitted deliverables for suggestions and recommendations.

- Presentation and Defense Scheduler module allows the Capstone Project Professor to create a list of date and time slots that the students can select for their presentation or defense.

- The Manuscript Evaluation feature gathers the students' submitted manuscript and submit it to their selected panelists. Additionally, the panelist can

submit their recommendation, comments, as well as the grade for that manuscript.

- User's Management feature enables users to login, register, logout, reset password, and change their password. Moreover, it enables user profiles to manage and access the right information depending on their profile (student, capstone professor, adviser, and panelist). Finally, the super administrator will be able to register new faculty and grant or remove privileges.

- Student's Grades Management will use the grades of the students inputted on the evaluation module. On this feature, the instructors and students can view the result of their evaluation.

- Student Management feature will allow the Capstone Professor to generate lists of students' information. The list will contain student's name, group members, the adviser, and panelists they've chosen.

- Capstone Project Guide and Formatting is a separate module where can post any information or instruction with regards to the Capstone Projects. The post can be lessons, paper formatting, or the chapters and subsections of the manuscript.

- The Discussion Forum is a feature of the system where students and faculty

members can interact each other by creating discussion and answering queries. This feature will need the name of the user for identification as well as the title, query content, and optionally, an image.

CHAPTER II

Review of Related Systems

The following papers are systems related to the present study in terms of online monitoring, management, evaluation of Capstone Projects, Thesis, or Researches. The gathered related systems were from different foreign and local sources such as books, magazines, journals, and websites.

The University of Malaysia Pahang (UMP) Thesis Management Systems of Rabman [32] is a web management system for thesis management that can store data, read, mark, and comment on the thesis. The system also includes a system page where they can be informed about the thesis through announcements and other information and guides in thesis writing. Additionally, the system has a private message feature where students can send messages or queries to their supervisor related to the development of their manuscripts.

The system of [32] is similar to the current project where both aimed at easing students' manuscript submission to their supervisors or adviser. Both systems can manage and mark or evaluate the students' submitted papers; however, in terms of their differences, the system above offers private messaging to their users, which is not included in the scope of the current project.

Torrechiva et al. [41] project development program for Capstone Projects

monitoring and management is an online system that aims to improve the current process of capstone management. The project focuses on project deliverables management, group progress monitoring, student-teacher interactions, and security of the submitted documents. The system was deployed at the University of Cebu – College of Computer Studies .

The system mentioned relates to the present system on its objectives of improving the current capstone process. They both focus on the deliverable's management and group progress monitoring. However, the present system is unique compared to the project above in terms of having an evaluation system and a tool where students can check their manuscripts for plagiarism.

The paper of Lo [22] presented an online collaboration system for capstone projects and other project-oriented courses. The features included in the system were project grading, social media integration, and review of documents. It is developed based on the open-sourced project management system called Redmine.

Compared to the present proposed project and the system of [22], they aim to create a system that will handle capstone projects, especially its deliverables, and monitor students' progress in a single venue. Furthermore, the system above integrates social media in their system, which is not available in the present proposed system. On the other hand, the current system includes manuscript evaluation and a plagiarism checker tool that is not included in the system stated

above.

A study conducted by Olarte et al. [24] presented a tool for students and professors or advisors to assist in the management and development of capstone projects for Computer Science Engineering. It includes planning and monitoring tasks, document management, learning community using a social network built in the system, informing students of their peers' progress, and facilitating contact between students and faculty developing projects.

The study of [24] is related to our proposed project in terms of aiming to create a tool or system that can assist students and faculty in managing capstone projects. They are similar in their features aside from the above system's integration of social networks and the present system's inclusion of evaluation module and plagiarism checker.

Grooms [15] developed a program that aimed to provide the University of North Carolina Wilmington (UNCW) Masters of Science in Computer Science and Information Systems (MSCSIS) a program for Capstone Management System (CMS) for the students, faculty, administrators, and program director to automate the capstone thesis or project process. The CMS enables the said users to complete the capstone process using an integrated web application. The CMS simplifies the current manual approval process saving time for all stakeholders. The CMS also provides high-level reporting capabilities for high-level decision-making .

The system of [15] and the proposed system both aimed to ease the capstone process workflow. Additionally, both systems have monitoring of students' progress and managing of capstone deliverables capabilities. The difference between the two is that the system above can only record and announce the capstone proposal and defense, where the present system can be used for the title proposal and final defense of the manuscripts.

Another paper of Grooms [14] aimed to replace a paper form-based process that was confusing, time-consuming, and error-prone describe with a new system and implementation of a web application for the approval workflow of a master's program in information systems. Student statuses are stored in a relational database, and program-level reports are provided for administrative decision-making. The main goals were to help guide students through the Theses or Project processes, ease the burden of obtaining approvals, and handle the needed paperwork .

The paper of [14] is similar to the proposed project in some respect. They aim to streamline the process and centralize all paperwork/process steps in one venue to eliminate the loss of papers. Both systems provide a progress tracking or status report, which allows faculty to monitor the improvement of the students. In terms of their differences, the project above focused on the committee's approval to the proposed Theses or Projects of the students, hence Approval System. In

contrast, the current project focuses on manuscript management and evaluation with a plagiarism checker.

The manuscript created by Orozco [27] aimed to develop a web-based Evaluation System which can allow panelist and advisers of College of Computer Studies of University of Perpetual Help Biñan to evaluate the graduating students and their Thesis/Capstone Projects through an innovative and automated way of grading.

The mentioned system presented an automated way of evaluating the Thesis or Capstone Projects of graduating students in their school, which is an available feature in the present proposed system. In addition, the present project has a management and progress monitoring module and plagiarism checker, which is not present in the system of the mentioned manuscript above.

The Learning Management System of Baeva [3] is a software system for managing and supporting the interaction between graduate students and their supervisors in the course of thesis writing. The system allows to plan, manage and track the progress of students' thesis or capstone projects. It includes management of activities, working out an individual development plan for graduate student's thesis work, work stages management, regulation of the interaction between a graduate student and the supervisor, resource management, and maintenance of primary documents .

The system of [3] has similarities with the current project when it comes to the management of capstone deliverables and its users. It is also similar when it comes to tracking progress or monitoring student's activities, and both aim to reduce administrative and academic burden in terms of capstone or thesis management. However, the system above has a direct messaging feature for student-supervisor interaction, which is not included in the present system. On the other hand, the system above does not include an evaluation system and a plagiarism checker feature, which are available in the present proposed project.

The system of Hamid [17] of the Department of Computer Science, Cihan University-Erbil in Iraq, was successfully developed and implemented as a web-based application for the scientific affairs department and was used and tested by one of the Iraqi governmental universities. The main goal of this application is to collect all submitted publication information in a central database, generating different types of publication reports, generating various real-time statistics, saving all researcher's lead in the Higher Education Institutions (HEIs), and facilitating checking and tracking the submitted researches. The Researches Management System aimed to manage research information in HEIs to improve the efficiency and effectiveness of the research information management processes. It is designed for three types of users: the System Admin, College Admin, and the Researchers .

Both systems, the project of [17] and the proposed system, tackled the management of scholarly articles and aimed to manage and improve research information processes. However, the system above contained finished or published manuscripts and only focused on handling those manuscripts. On the other hand, the present project addresses unfinished manuscripts until they can be issued.

The Thesis Evaluation System of Zong, Y. [45] aimed to develop an online evaluation system that can substitute the paper-based that is currently used in the Department of Computer Graphics Technology Purdue University in Indiana, United States. It is expected that this new evaluation system will increase the efficiency of the thesis evaluation process and reduce the workload of the advisors and committee members. It also aimed to ease storage, organization, and visualization of evaluation results.

The mentioned project is related to the present system in terms of aiming to replace paper-based Thesis or Capstone Project processes. Both systems include an automated evaluation system that can be used to grade or mark the manuscripts submitted by the students. The above project, however, only focuses on the said evaluation system while the proposed present system includes other features like title proposal, adviser and panelist selection, manuscript management, and plagiarism checker.

An applied project of Owusu-Afriyie [28] aimed to develop a management

system that can primarily monitor and accompany all activities involved in the capstone process. It is designed for three types of users – students, faculty, and computer science coordinator, and was deployed in the Ashesi University campus in Ghana, Africa. The system solved two pressing needs in the said university: it simplifies the entire capstone process and presents students with the opportunity to work on exciting projects that have been proposed by faculty.

The stated system of [28] is related to the present system in its aim of simplifying the capstone process by efficiently managing manuscripts, faculty selection, and progress tracking. The difference between the two is that the system above includes a meeting scheduler between the faculty and students, which is only present in the currently proposed system in terms of scheduling capstone defense. In addition, the present system includes an evaluation module for the manuscript as well as a plagiarism checker, which is not available to the above system.

CHAPTER III

Technical Background

This chapter introduces the technical background of the system. The hardware and software specifications used in development, the specifications requirements for users, and the definition of the technical terms used are discussed in this section.

3.1 System's Development Specification

This section presents the hardware, software, and service specifications of the system that will be used by the proponents in developing the project.

3.1.1 Hardware Specification

Table 3.1. Developers' Hardware Specification

Hardware	Description
Processor	AMD Ryzen 3 2200G 3.50 GHz
Memory	8.00 GB RAM
Graphics Card	AMD Radeon™ Vega 8 Graphics (Built-in)

Table 3.1 shows the specifications of the hardware used by the proponents in developing the project. The proponents used AMD Ryzen 3 2200G with 3.50 Gigahertz clock speed with built-in AMD Radeon Vega 8 graphics card and an

8.00 GB RAM. The hardware that will be used in developing the system should be sufficient enough to handle different tasks and can be installed with different software. It should be able to run efficiently, with high performance, and good quality so that the proponents can experience smooth development of the system. The specifications above was sufficient enough to be able offer the best and efficient performance needed in developing the cloud-based web application.

3.1.2 Software Specification

Table 3.2. Developers' Software Specification

Software	Description
Operating System	Windows 10 Pro v. 20H2
Browser	Google Chrome
Integrated Development Environment	PyCharm 2021.1.1
Database	MySQL
Image Editor	Adobe Photoshop
Wireframe Design	Adobe XD
Webpage Design	Adobe Dreamweaver

Table 3.2 presents the software and their specifications used in developing the proposed project. The system will be developed in Windows 10 operating system using Google Chrome as its browser. In the back-end aspect of the system, the Integrated Development Environment (IDE) and database that will be used is PyCharm and SQLite, respectively. The proponents used different Adobe products which are Adobe Photoshop for editing images and icons; Adobe XD for creating wireframe; and Adobe Dreamweaver as a tool in creating a template for webpage

design. The proponents used different applications that they are comfortable and can hone their skills. It provided easier, efficient, and fast development of the project. The proponents also made sure to use the updated versions of the software to experience to be able to experience their updated features and to make sure that there are enough community that can support them when using the software.

3.1.3 Service Specification

Table 3.3. Services Specification

Service	**Description**
Internet Service	PLDT
Plagiarism Checker	PrepostSEO
Cloud Computing/Hosting	Microsoft Azure

The services used in this project, and their description is displayed in Table 3.3. The proponents used PLDT and its plan as their Internet Service Provider, Prepostseo and its API as the integrated tool for checking plagiarisms, and Microsoft Azure for Cloud computing deployment of the system. It is necessary to have the right service in developing the system. The proponents highly believed that the services specified are the right services to be used in developing the system because of its fast and reliable performance, large community, and suits the budget of the proponents without sacrificing the quality of the project.

3.2 User's System Specification Requirements

In this section, the approximate minimum and recommended hardware and software specification of the users' system are presented. It is imperative to specify their specification so that the system can be used in the right environment.

3.2.1 Hardware and Software Specification Requirements

Table 3.4. Users' Hardware and Software Requirements

Component	Minimum	Recommended
Processor	Dual core	Intel core i3 or higher AMD a4 or higher
Memory	2.00 GB RAM	4.00 GB RAM or higher
Hard disk	120 GB	120 GB or higher
Internet Connection	1 mbps	2 mbps or higher
Peripherals	Monitor, Mouse, Keyboard	Monitor, Mouse, Keyboard, AVR/UPS
Operating Systems	Windows 7	Windows 7 or newer
Browser	Internet Explorer	Google Chrome

Table 3.4 presents the minimum and recommended user's hardware and software specifications. At least a dual core processor, 2.00 gigabyte RAM, 120gb hard disk, 1mps internet connection, and necessary peripherals to access computer are needed in order to use the system. It is necessary to have at least stated specifications to be able to use the system with good quality and performance. On the hand, the users only need two software – Operating system and a browser. The proponents require that the users should have at least

Windows 7 Operating system and Internet Explorer browser to be able to use the system.

3.3 Technical Terms

The following are terms used throughout this study and this section aims to conceptually and operationally define these words to clarify the concepts behind its use in this study.

Cloud-Based System – a system using the Cloud technology which has the ability to host software or services from a remote location that can be freely accessed and used anywhere with Internet access [29]. Microsoft Azure, which will be used in this study, is platform that can be used in deploying cloud-based systems or services.

Monitoring, Management, and Evaluation System – a monitoring system oversees the activities done by a certain people like an employee in a company or a member in a group project; Management system, handles and controls the activities of a user in a system and; Evaluation System, is a module in a system that enables user to assess a certain project. The current project has these system which enables the users to supervise or monitor the progress done by the students, manage their activities, and evaluate their submitted manuscripts.

Plagiarism Checker – a system that checks for instances of plagiarism in an article. In this project, PrepostSEO plagiarism checker platform will be used to detect the said instances.

Web Application - A web application is a computer program that utilizes web browsers and web technology to perform tasks over the Internet [11].

CHAPTER IV

Methodology

This chapter aims to discuss the procedure, techniques, tools, and documentation aids that will be used in this study. It includes various diagrams, figures and tables that can further help visually discuss the said methodology used.

4.1 Project Concept

The project aims to create a cloud-based web application system for monitoring, managing, and evaluating capstone project deliverables. It also includes an integrated plagiarism checker software that scans the received manuscript for plagiarism. It will be primarily used by students, capstone advisers, panelists, the capstone professor, and the department dean.

The project will follow the Object-Oriented Analysis and Design, use the Iterative type approach for its development life cycle, and Bottom-Up as the development approach. Additionally, it will integrate different technology tools like HTML5, Python, SQLite, and Microsoft Azure. The stated approaches and tools will be further discussed in the subsequent sections. On the other hand, the following activity diagram aims to visually present and discuss the flow of activity of the system.

4.1.1 System Architecture Diagram

System Architecture Diagram is a graphic representation that is used to abstract the overall outline of the software system and the relationships, constraints, and boundaries between components [6]. The following are the two Systems Architecture Diagram meant to show the deployment architecture of the system in both general and specified or cloud view.

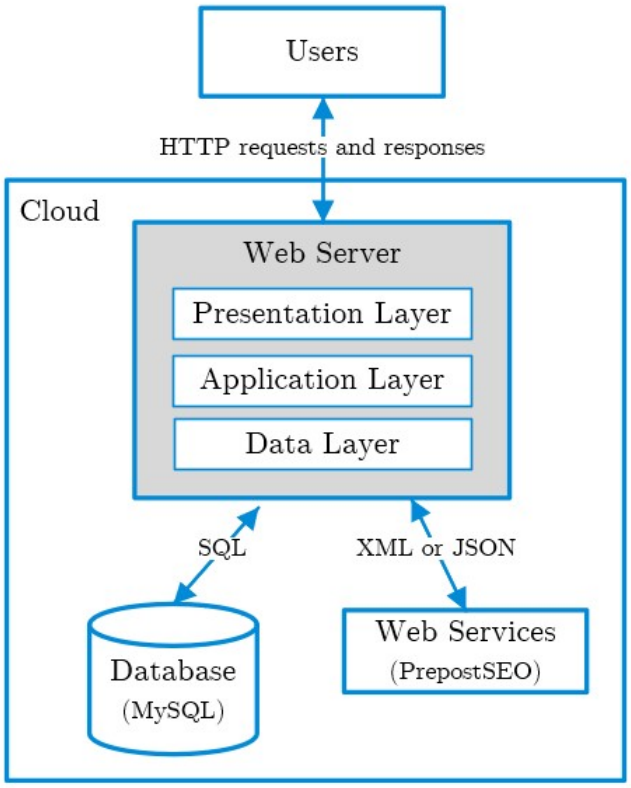

Figure 4.1. System Architecture Diagram (General View)

Figure 4.1 shows the general view of the architectural diagram that defines

the structure of the system. The users will interact to the web server by requesting and receiving through Hypertext Transfer Protocol (HTTP). The Web Server consists of the of the various layers of the application that conforms with Model-View-Controller (MVC) design pattern. The Presentation Layer is where the interaction between the users and the system will first take place. The Application Layer manages the flow of the application, implements the systems logic and liaises with the data layer to process requests from users and their responses. The Data Layer, handles the domain data and provides persistence and retrieval services for the database. The Database using MySQL is where the data is persisted and retrieved. Finally, the Web Services, is the one that handles interactions with other applications, in this case, the PrepostSEO for the plagiarism checker.

The Web Server, Database, and Web Services are all located in the Cloud, specifically with the service of Microsoft Azure. In this way, the proponents will not be limited to the specifications of their hardware for developing the system, will be available in the internet, and with fast and reliable performance, security, and scalability.

The figure 4.2 is the closer look of the Web Application Architecture deployed in a Cloud-based environment.

To access the system, the users and the developers will need an Internet

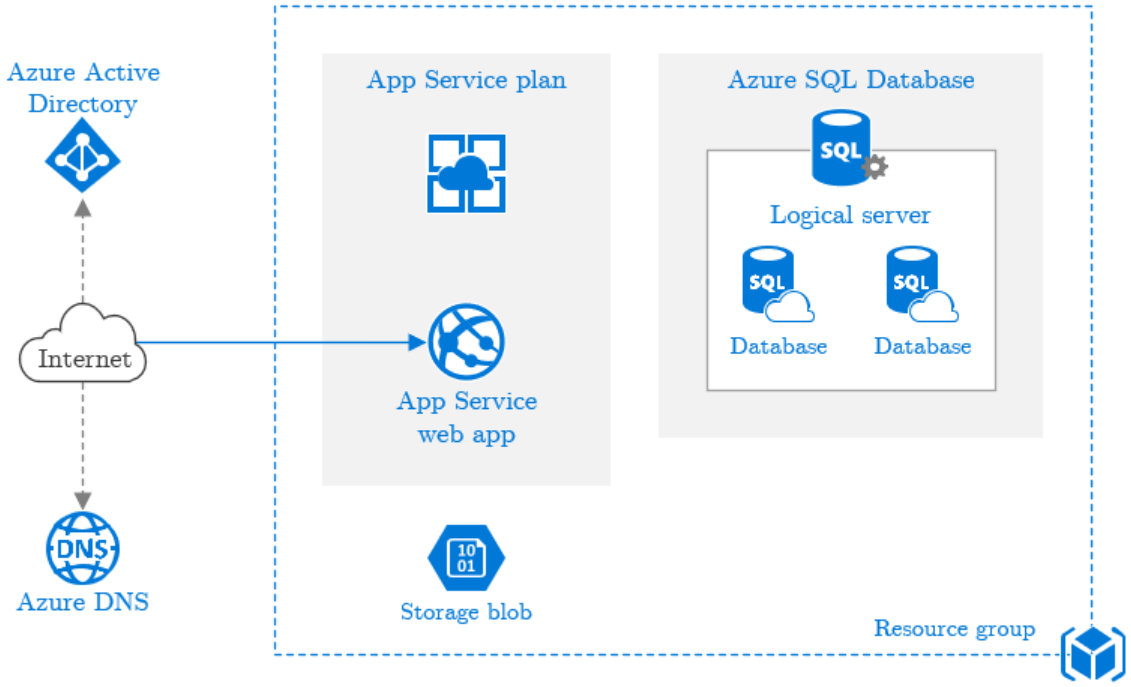

Figure 4.2. System Architecture Diagram (Cloud View)

connection to type the Domain Azure DNS provided. They will be authenticated with the use of Azure Active Directory and its firewall for security purposes. The main Cloud Server has different components in a single resource group, mainly the App Service Plan, App Service Web App, Storage blob, and the databases.

The App Service Plan provides the managed virtual machines (VMs) that will host the application in the cloud. The App Service Web App is the one that will fully manage the application and where the app will be deployed. Storage Blob stores unstructured data and serves them to users over the HTTP and HTTPS. The Azure SQL Database enables the developers to have logical server where the database is hosted and multiple databases for storing data. They will

all be under the same Resource Group which is a container that holds related resources for an Azure solution

4.1.2 Data Flow Diagram

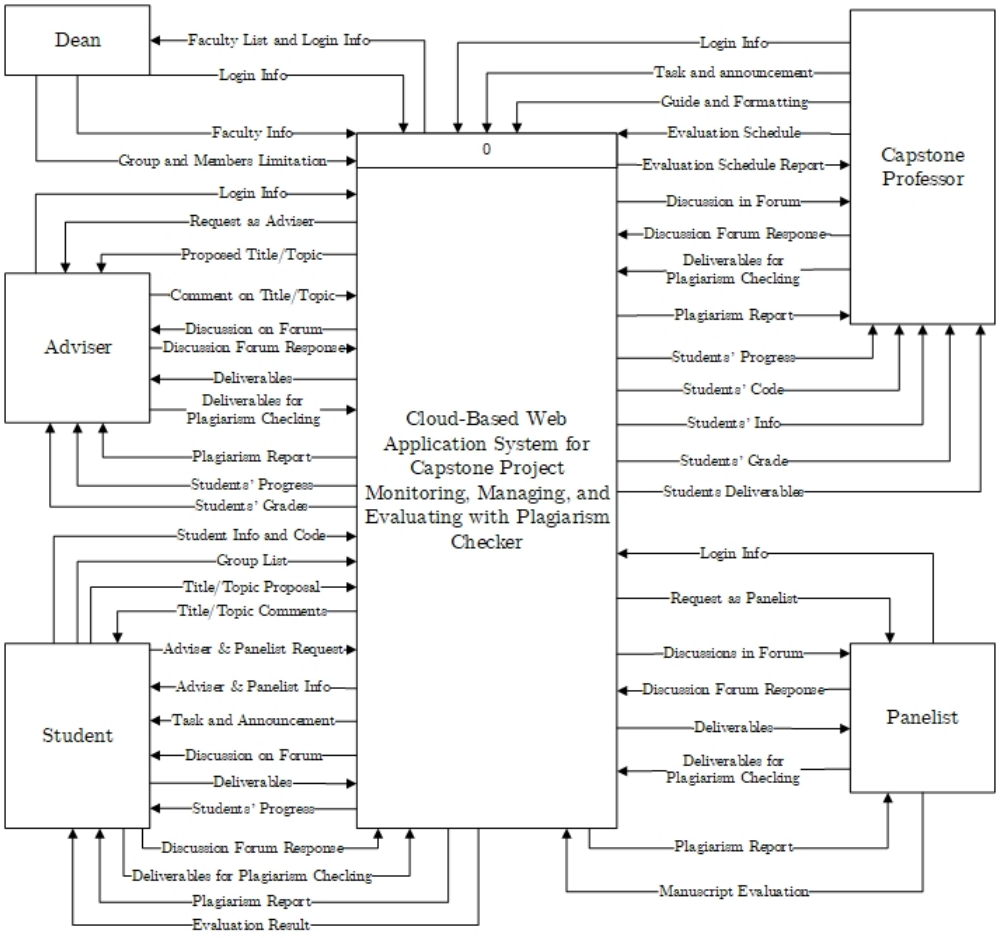

Figure 4.3. Data Flow Diagram Level 0 (Context Diagram)

According to [44], Data Flow Diagram (DFD) is a graphical representation of the "flow" of data through a system that models the systems process aspects. It is the preliminary step in creating overview of the system which will be later

elaborated using different diagrams. The context-level data flow diagram is the first step in creating DFDs. It shows the interaction between the system and external agents that act as data sources and data sinks.

Figure 4.3 presents the level 0 DFD or the context-level data flow of the system. The system is represented as rectangle located in the middle of the figure with number zero at the top, indicating the 0th level of the diagram. Users Dean, Adviser, Student, Capstone Professor, and Panelists, are also represented as rectangle and its interaction to and from the system are represented as arrows. As shown in the figure, the system accepts different inputs from the user and returns different outputs in accordance to their privileges.

Figure 4.4. is the level 1 DFD. While the context-level data flow showed the whole system as a single process, Level 1 DFD notates each of the main sub-process that comprises the system. It includes external entities (Admin, Capstone Professor, Faculty Members, and Students), the processes they do in the system (Manage Faculty and Task Management among others), and the data store where the processes will store or retrieve data to be processed.

4.1.3 Use-Case Diagram

A Use-Case Diagram is a visual representation of the functionality and the possible interactions between the system and the users form the latter's point of

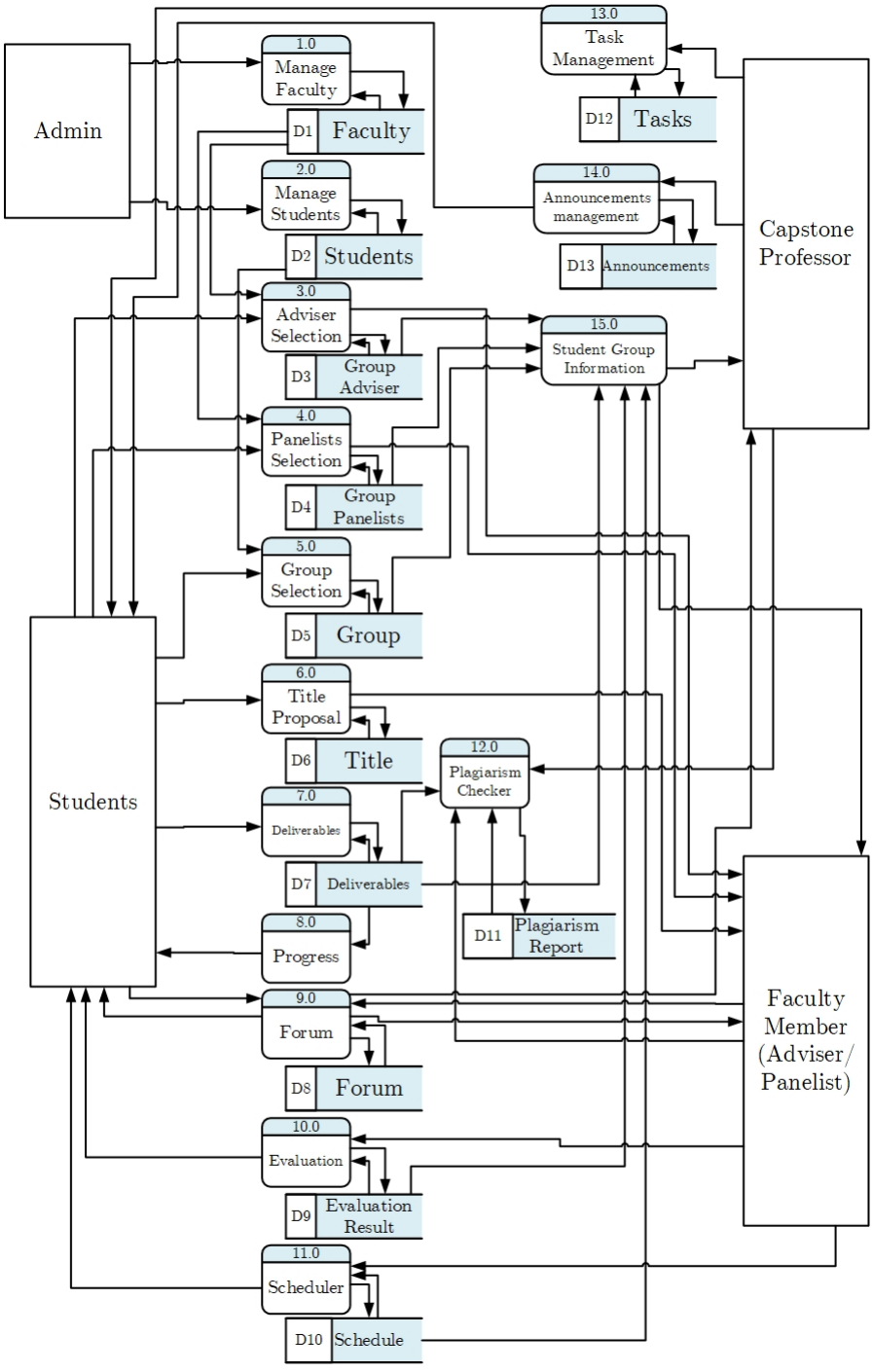

Figure 4.4. Data Flow Diagram Level 1

view [4, 23]. Figure 4.5 shows the system and its boundary as a rectangle, the functionalities in oval shape, the actors and its interaction to the functionalities within the system. The Table Descriptions of the use cases utilized in the diagram can be found in at Appendix B: Use Case Table Description. The following actors and their scope of interaction are as follows:

Department Dean

The Department Dean as the Super Admin, after signing their credentials, will be able to manage the main users of the system, specifically the Faculty members in their department. Managing users includes registering or creating an account for new faculty members, granting them privileges, and assigning the Capstone Professor from the list of the registered faculty members.

Capstone Professor

The capstone professor is an actor that can manage the activities of the students and the faculty in the system. After signing the credentials given by the Dean, they can post a guide or formatting to be followed by the students in making their manuscript. They can create tasks, set its deadlines and make announcements. They can also participate in forums, manage the deliverables and check for plagiarism, monitor the progress of the students, schedule the evaluation, and manage the grades of the students. This actor will be registered by the super admin and henceforth will be known as the administrator of the

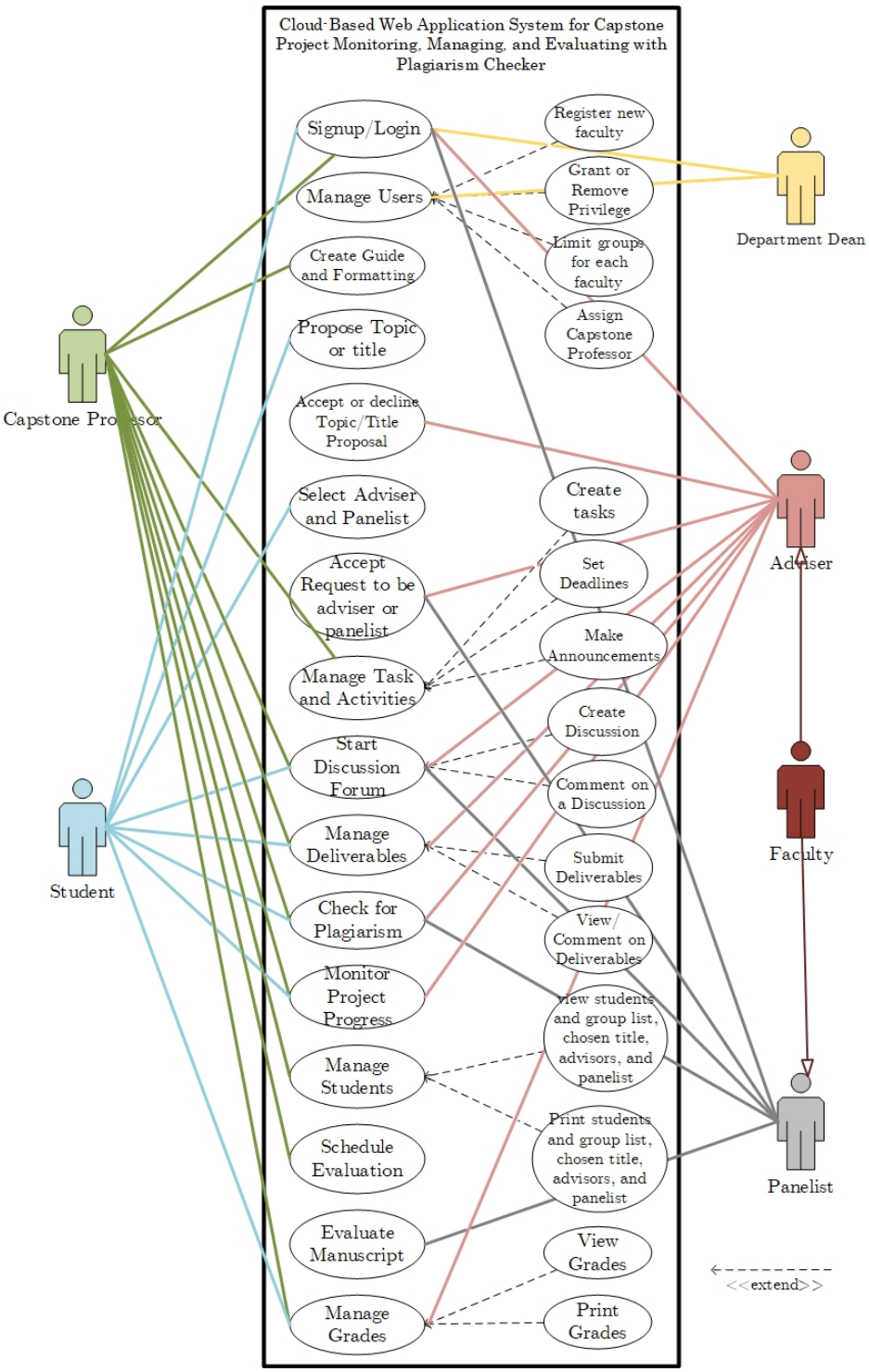

Figure 4.5. Use Case Diagram

system.

Adviser

An adviser is a faculty member that accepted the request of the students to guide them in their project. They can accept or decline the students' proposed project or title, manage and monitor the deliverables submitted by the student, and check for plagiarism. They can also participate in discussion forums and view the evaluation result of the students.

Panelist

The panelist is a faculty member chosen by the students who will evaluate their capstone project. They can participate in forums, use the plagiarism checker, and evaluate the manuscript submitted by the students.

Student

The student is the actor that perform the activities within the system given by the capstone professor/faculty and manage the deliverables that they made. Upon registering, they can propose capstone project or title and can select their preferred adviser and panelist. They can also take part in the discussion forum, submit and check their deliverables for plagiarism, monitor their progress and view their grades.

4.1.4 Activity Diagram

Activity Diagram is a flow diagram used to represent the behavior of a system in terms of activities which can represent control flow and data flow [4]. Figure 4.6 to Figure 4.10 presents the Activity Diagram or the control flow that occurs in the system from login or registering and to the various activities of the different users.

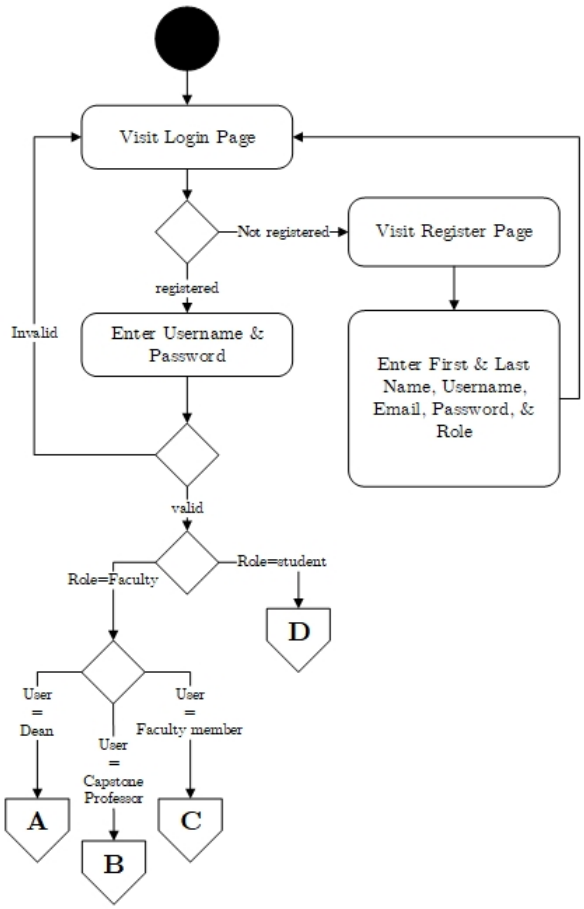

Figure 4.6. Login and Register Activity Diagram

Figure 4.6 showed the login and registering activity of the users. If users were already registered, they can login to the system by inputting their username and password. If not yet registered, they can visit the Registration Page and enter their necessary information.

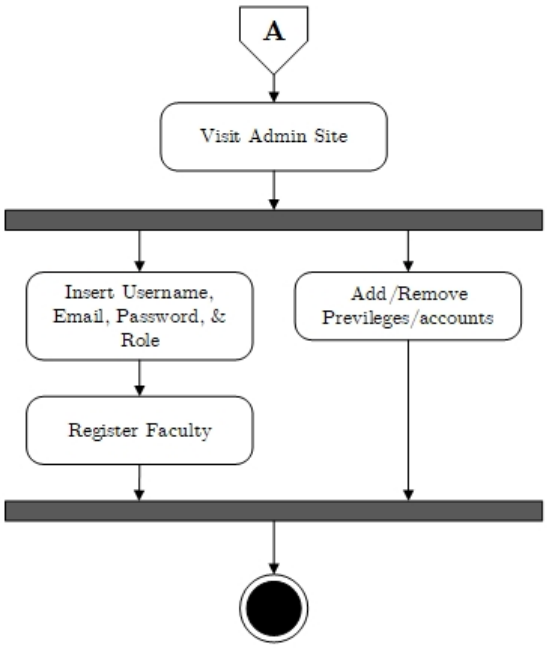

Figure 4.7. Department Dean User Activity Diagram

Figure 4.7 showed the activity flow of the Department Dean as a super admin of the system. They have the privilege to manage the users wherein they can register new faculty and add or remove said privileges or accounts.

Figure 4.8 showed the flow of the activities that can be done by the Capstone Professor. In the figure, they can monitor the group progress and manage students

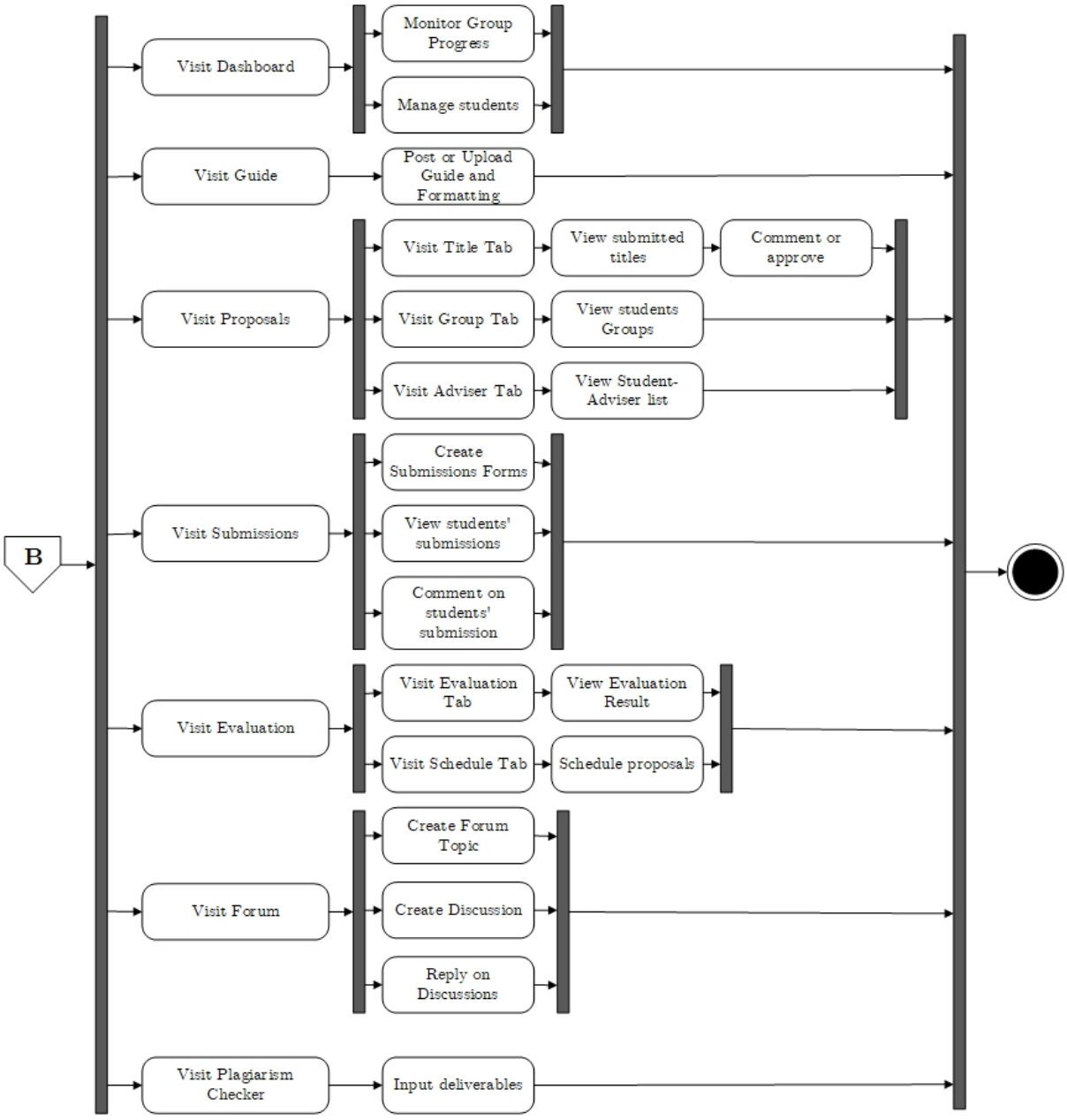

Figure 4.8. Capstone Professor User Activity Diagram

in their dashboard. They can perform other activities such as creating or posting documents containing guide and formatting, view submitted titles, create and view submission forms, view evaluation result, create a forum topic, participate on the discussion forums, and check manuscript for plagiarism.

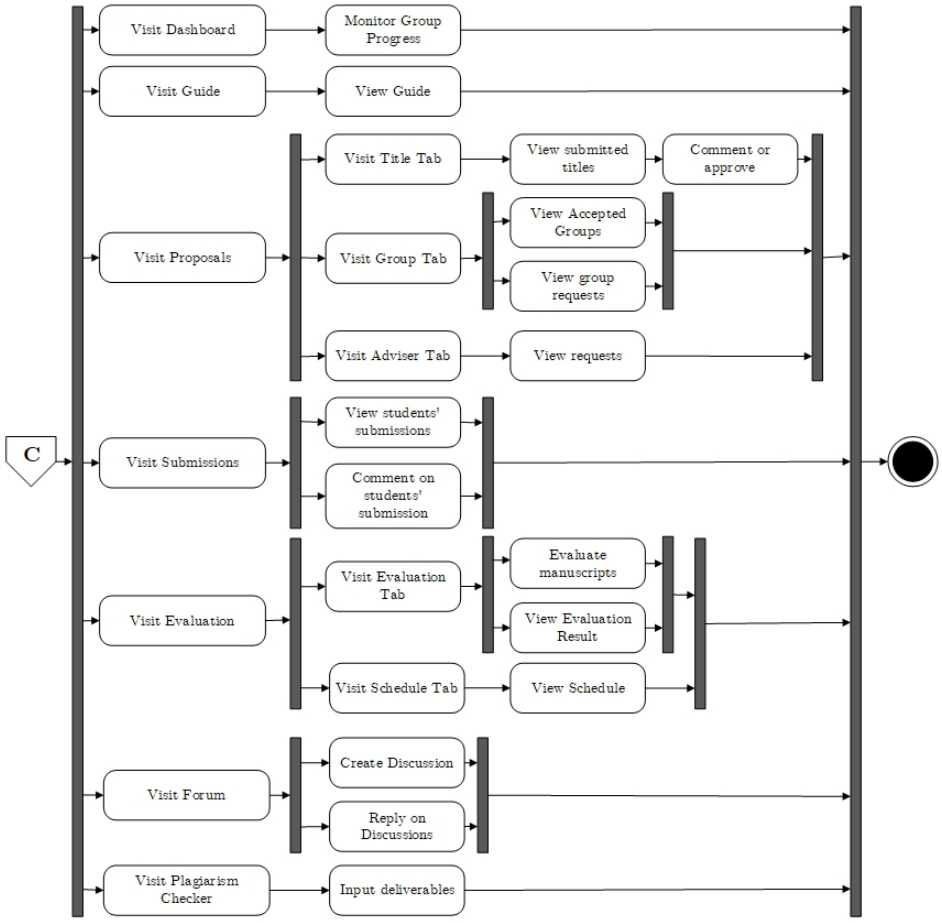

Figure 4.9. Faculty Users Activity Diagram

Figure 4.9 illustrated the activities perform by the faculty members as an adviser and/or panelist. Depending on their role, they can accept or decline

title proposals and requests to be the adviser or panelist, evaluate the students' manuscripts, and monitor the group project progress among others.

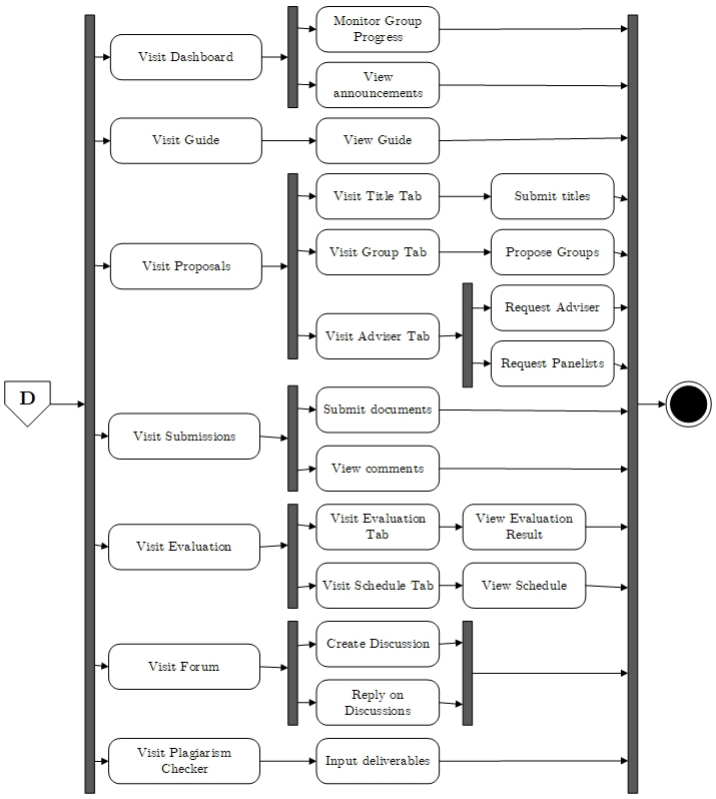

Figure 4.10. Student Users Activity Diagram

Figure 4.10 illustrated the flow of the different activities the students can do within the system. They can monitor their progress, present proposals that includes group selection, title proposal, and requests for the adviser and panelists, submit deliverables, and view the evaluation results among others.

4.1.5 Entity-Relationship Diagram

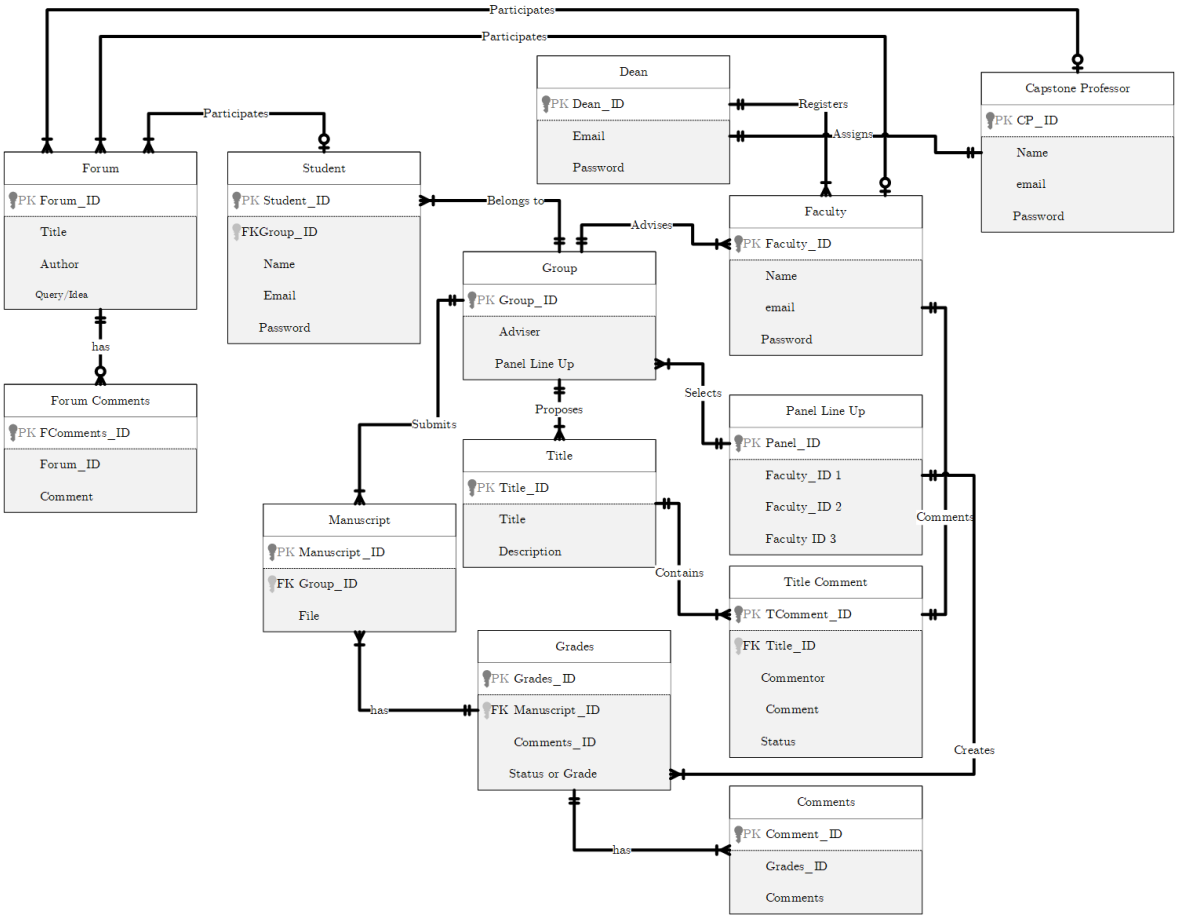

Figure 4.11. Entity Relationship Diagram

Entity-relationship model (ER model) is blueprint that describes the structure of a database with the use of a diagram which can then be implemented as the actual database [36].

The ER diagram that the project will use can be seen in Figure 4.11. It shows the cardinality, attributes, and relationship of one entity to another,

showcasing the logical structure of the database

4.2 Analysis and Design

The analysis and design approach that this study will use is the Object-Oriented Analysis and Design approach (OOAD). It is an iterative and incremental process which are used for analyzing and designing a system by applying object-oriented programming, and visual modeling throughout the development process [12].

According to [23], OOAD is an approach in software creation that focuses on making sure that the code does what it is supposed to and that it is well designed. OOAD also highly focuses on both customers' and programmers' satisfaction with the system for its flexibility, ease of making changes, and maintenance and reusable.

4.2.1 Requirement Analysis

Requirement Analysis is an essential aspect in developing a web application used in determining the needs and expectations of the application [35]. Following the Object-Oriented Approach in Analysis and design, the following tables are the functional and non-functional requirements that the system should comply with. Diagrams are also presented to further visualize the functional requirements of the system.

4.2.1.1 Functional Requirements

Functional Requirements presents the features that must be implemented in the system and to be used by the users [2]. Table 4.1 presents the functional requirements of the system specifically its task description and reference. The task requirements reference is directly linked to the objectives of this study.

Table 4.1. Functional Requirements

Task Requirements	Task Reference
The users of the system should be able to monitor the progress of the students	Project Progress Monitoring
The students should be able to propose a title or topic they want to work with to be viewed and commented on and accepted or declined by the faculty	Topic or Title Proposal
The students should be able to request faculty members to be their adviser or panelist and to be accepted or decline by the chosen faculty member	Adviser and Panelists Selection
The system should be able to display announcements created by the Capstone Professor, task's deadlines and enable the said professor to create tasks	Task and Activities Management
The system should enable the students to submit deliverables and be viewed by their capstone professor, adviser, and panelists.	Deliverable's submission, monitoring, and management
The Capstone Professor should be able to schedule the evaluation of the students	Schedule Defense
The system should have a module where panelists can input the grade of the students as well as their comments or recommendations	Manuscript Evaluation

The users should only view the appropriate content depending on their role and for the super administrator and administrator to add or remove users	User's Management
The system should enable all users to view the student's grades and the evaluation result	Grades Management
Allow administrator to generate report of students' information	Student management
Enable Capstone Professor to post standard formatting and guides in creating capstone projects	Capstone Project Guide and Formatting
The system should have a page for forums where all users can participate	Discussion Forum
The system should be able to integrate a plagiarism checker in the system and check for cases of plagiarism	Integrate Plagiarism Checker

4.2.1.2 Non-Functional Requirements

Non-Functional Requirements, also known as the system's quality attribute, in Table 4.2, describes how the system should behave, quality-wise, and establishes the constraints of its functionality to ensure Software Quality Assurance [2]. Table 4.2 presents the non-functional requirements of the systems. The description of each requirements and its task reference are also presented. The stated references were adapted from Software Quality Standards of International Organization for Standardization, specifically in ISO/IEC 25010.

Table 4.2. Non-Functional Requirements

Task Requirement	Task Reference
The system should be able to perform all the functional requirements and provide the appropriate results	Functional Stability
The response and processing time of the system and the amount and types of resources used should be reasonable	Performance Efficiency
The system should be able to share data with other or specified products, systems, or components	Compatibility
The system should be accessible to its different users, has aesthetic user interfaces appropriate for its users, and be easily recognized by the users the use and appropriateness of the system	Usability
The system should be stable and available to the users with fault tolerance and recoverability	Reliability
The system should be accessible to authorized users, prevent unauthorized users' access to the system, and collect appropriate and reasonable information only when necessary.	Security
The system should be able to be modified or improved in the future to adapt to changes in the environment	Maintainability
The system should be able to be used in various and developing technology or environment	Portability
The system should enable all users to view the student's grades and the evaluation result	Grades Management
Allow administrator to generate report of students' information	Student management
Enable Capstone Professor to post standard formatting and guides in creating capstone projects	Capstone Project Guide and Formatting
The system should have a page for forums where all users can participate	Discussion Forum
The system should be able to integrate a plagiarism checker in the system and check for cases of plagiarism	Integrate Plagiarism Checker

4.3 Development Model

As shown in Figure 4.12, the Software Development Life Cycle (SDLC) model that will be used in this study is the Iterative Model. In this model, development can begin with some of the software specifications and develop the first version of the software so that it will be in a deliverable state. If improvement is required or satisfactions met, iterative enhancement will be done until the complete system is implemented and ready to be deployed. Every release of the Iterative Model finishes in an exact and fixed period is called an iteration. The final output of the project renewed at the end of the Software Development Life Cycle (SDLC) process [18, 19].

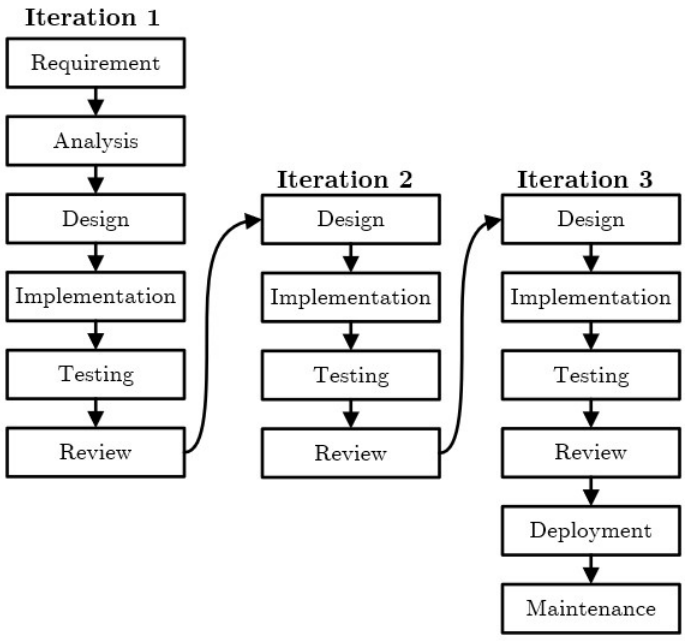

Figure 4.12. Iterative Development Model

According to [10, 18], the following are the various phases of the Iterative Model:

1. Requirement gathering and analysis: In this phase, requirements are gathered from customers and checked whether the requirements can be fulfilled or not. These requirements are features or the target output of the clients when using the system. The client of this study is the ICT Department of Sorsogon State University – Bulan Campus (SSCBC). The proponent, as also taking the Capstone Project under the same Department as the client, observed the current process in creating the said projects. They initially planned the features that the system should have in order to assist the client in streamlining the process in creating the students' project. After that, they asked the client if the planned features are sufficient enough to achieve the goal of the system and the client also suggested other features that the system should have. The developers also studied existing projects to get an idea for the enhancement of the project and analyze the features to make the present project unique.

2. Design: In the design phase, team design the software by the different diagrams like Data Flow diagram, activity diagram, class diagram, state transition diagram, etc. These diagrams are essential in visualizing the

gathered requirements from the clients and to picture out its interaction with other requirements and elements in the system. Activity Diagram presented in the Project Concept section and the System Architecture, Data Flow, Use Case, Entity Relationship, and Class Diagram in the following section, are created in this phase in relation to the requirements approved by the client

3. Implementation: In the implementation, development of the requirements or functionalities required to meet the specifications stated from the last phase, are written in the coding language. The proponents integrated different technologies in the development of the system. These technologies will be discussed in the Software Development Tools section.

   The proponents, after their learning phase, started implementing the system by creating the project and its feature. As part of the 1st iteration, they were able to the Discussion Forum, Guide and Formatting, User's Management, the deliverables submission, and task management features of the system.

4. Testing: After completing the coding phase, software testing starts using different test methods. In this phase, units or features that are not working or performing properly to the client's and proponents' expectations are to be identified, located, and fixed. The testing method that will be used in this

study will be discussed in the Testing section in this chapter.

5. Review: In this phase, review is performed to check the behavior and validity of the developed product. Moreover, the current iteration will be compared to the requirements and expectations of the users. If satisfaction met on the current iteration or reviews are gathered, the next iteration will start until the completion of the whole system.

6. Deployment: After completing all the phases, software is deployed to its work environment. Specifically, it will be deployed in SSCBC to be used by the dean, faculty members, and students of the ICT Department.

7. Maintenance: In the maintenance phase, after deployment of the software in the working environment there may be some bugs, some errors or new updates are required. Maintenance involves debugging and new addition options.

4.4 Development Approach

The system design strategy that this study will use is the Bottom-up development approach. According to [1, 42], the Bottom-up approach uses the modular components to develop the design of a system. The design starts at identifying the modules at the lowest level then grouped together based on their functions

by each module to form the next higher-level modules. This process will continue until all levels or components and subsystems are composed into a single and complete system. In this approach, the amount of abstraction grows high as the design moves to more high levels.

Bottom-up development approach will be used in this study because it works well with the Object-Oriented approach [43] which is will also be adapted in this study. Additionally, since the study will be an iterative process following the Iterative Model in SDLC, the proponent deduced that Bottom-up approach will be suitable when the current project's is in the implementation phase. The proponents used this approach by first creating the different features of the system and then joining them to create the whole system.

4.5 Software Development Tools

The following are software development tools and applications that will be used for developing the system:

- **Front-End Development**
  - **HTML5**. The latest version of the Hypertext Markup Language standard used to structure and present content on the World Wide Web [34]. Since the current project will be implemented with the use of the web

and to experience the new features of the latest version, the HTML5 will be used in this project.

- **CSS**. A markup language that describes the layout and colors of web pages, which can be used to adapt the presentation to various devices [7]. Together with HTML, they are the core technologies for building Web pages and this language is responsible for adding aesthetics to the webpage. This is why it is necessary to use this markup language in this project.
- **Bootstrap**. A framework that includes variety of tools for building web apps that is responsible for the responsiveness and compatibility of the web app to different browsers [31]. Additionally, its time-saving, easy to use, responsive grid system, customizability, cross-browser compatibility, and consistency capability, and for it being an open-source framework and huge number of resources and community support, encouraged the proponents to use this framework for designing the web app.
- **JavaScript**. A tool for developers to add interactivity to websites [9]. The proponents will use this tool because of JavaScript's capability of offering faster user experiences, user interface interactivity, good

responsive web design, easy to learn, and popularity

- **Back-End Development**

  - **SQLite**. A relational database management system coded in a C-language that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine [39].

  - **Python**. A general purpose, object-oriented, high level programming language used for developing desktop GUI applications, websites, and web applications [37]. Python's is also a simple, easy-to-learn syntax that focuses on readability, which lowers software maintenance costs and it has various modules and packages, which supports program modularity and code reuse. Additionally, its interpreter and extensive standard library are free to download and distribute in source or binary form for all major platforms. With the stated capabilities of Python, the proponents decided to use this programming language in conduct of this study.

  - **Django**. Since the developer uses Python as the programming language, it is applicable to use Django for its systems framework. Django is an open-source framework for backend web applications based on Python which aims to achieve simplicity, flexibility, reliability. and scalability.

It offers other technical features like simple syntax, its own web server, a Model-View-Controller core architecture, and unit test framework [21].

- **Text Editor and IDE**
  - **PyCharm**. A well-known cross-platform Integrated Development Environment (IDE) for Python [38]. PyCharm also enables programmers to easily write various web applications using widely used web technologies like HTML, CSS, and JavaScript. Making it ideal as a text editor for the Front-End Development and as an IDE for the Back-End for this study.
- **Version Control System**
  - **Git**. The proponents needs an excellent way to manage changes and version code in codebases. This can be done with the use of version control system. Git is the most commonly used version control system that tracks and records changes in the files, and enables the users to revert back to a specific version of the file [8]. Additionally, with its collaborative feature, it is the ideal version control system to be used in this study.

4.6 Schedule and Timeline

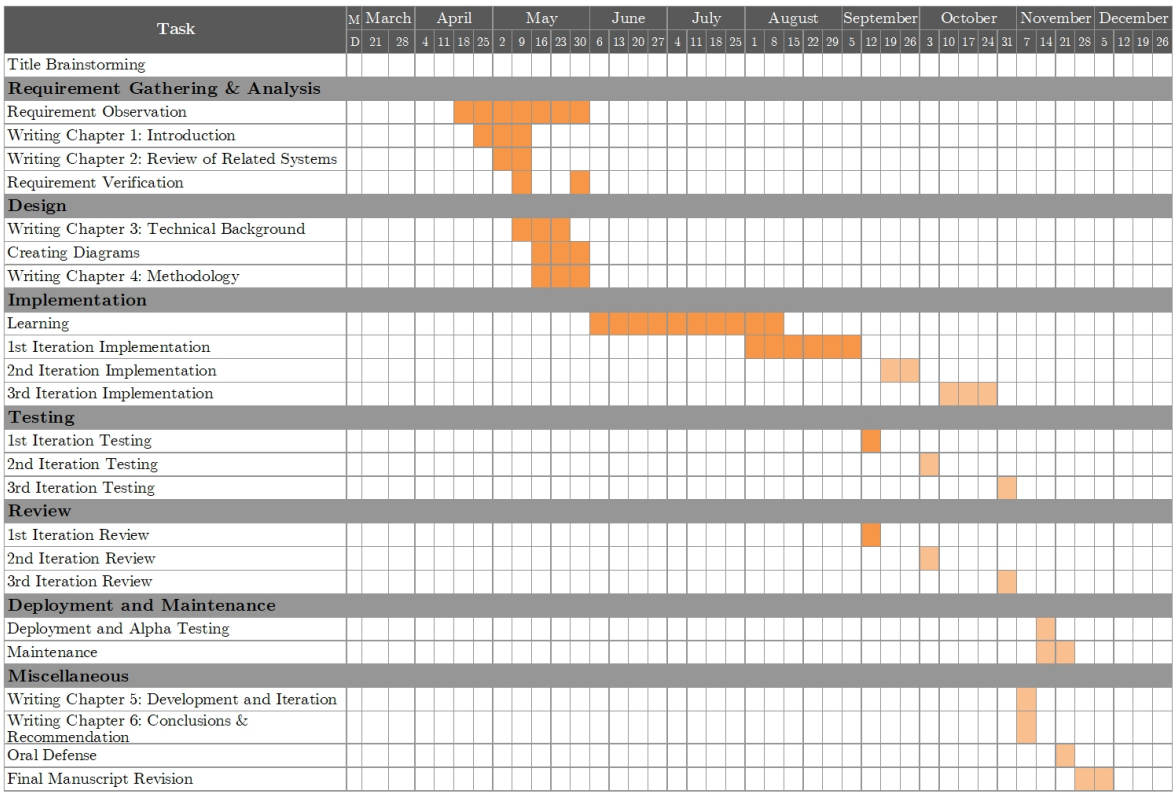

Figure 4.13. Development Timeline

This project timeline, as shown in Figure 4.13, is composed of two academic semesters – the 2^{nd} semester of Academic Year 2020-2021 and the 1^{st} semester of Academic Year 2021-2022. The project started from the idealization of the title on the 3^{rd} week of March and will end approximately in December.

The proponents started with brainstorming the title, followed by the requirements gathering and analysis. Observations, verifications, and writing the 1^{st} chapter and 2^{nd} chapter of this study are conducted during this phase.

The proponents then listed the technical background of the study as part of the design phase and created diagrams and the 4^{th} chapter of the study to visually present and explain the concepts of the study.

After the proposal of the study, the proponents went on a learning phase where they studied the different technologies that they will use especially the Django Framework and Web Development. They registered to different learning sites for this phase.

The implementation started on the month of August. In this phase, the implementation for the 1^{st} iteration started from the creation of the project and its applications or features. The proponents built the system from creating the different features of the system and then combining them together. This will be done until the completion of the whole system.

The 1st iteration testing and review will be done during and after the 1st iteration Capstone Project Checking on the 2^{nd} week of September. It will be followed by the 2^{nd} and 3^{rd} iteration implementation, testing, and review.

The deployment and maintenance phase will start after the completion of the iterations, and the system is ready for deployment to its predestined environment. At this phase, the last chapter and other papers will also be written.

After the stated phase, the final defense of the proponents will start, and

the System Development Life Cycle will end.

4.7 Responsibilities

The proponents of this study are composed of four BSIT students and an adviser. The following are their names, role, and responsibilities. However, although roles are specified per members, they are also in charged in developing different modules in the system.

- Anthony Gacis – Capstone Adviser

  The responsibilities of the capstone adviser are to track the progress of the students, serve as a guiding path in creating the capstone projects in terms of reviewing the papers and providing feedback and comments on the work.

- John Mathew Diño – Project Leader

  The responsibilities of the Project Leader are to delegate tasks to the members of the group, ensure that the team remains focused, foresee the progress of each member and the overall project, help teammates one way or another, and sets a goal for the teammates to cooperate and work as one to achieve success.

  John Mathew Diño will also oversee the development of the modules that will be done by the Capstone Professor but not including the common

features.

- Ramil Buen – Front-End Developer

  The front-end developer is responsible for implementing visual elements that users see and interact with. They design the interfaces of the system and other visual elements of the project including graphical elements in the manuscript.

  Ramil Buen will also develop the modules that involves the Department Dean.

- Rolly Gilos Jr. – Back-End Developer

  The responsibilities of the back-end is to handle the server-side development of the system. They are in charge to make sure that the features and functionalities of the system will align to the satisfaction of the client.

  Rolly Gilos Jr. will lead the development of the modules under the activities done by the students as well as the common features of the system

- Kasandra Alvarez – Technical Writer

  The Technical Writer is responsible of preparing, reviewing, revising, and maintaining technical papers and the documentation of this project.

  Kasandra Alvarez will also be responsible in developing the modules under

the faculty members that includes the adviser and panelists.

4.8 Budget and Cost Management

This project will need a budget for acquiring materials and services it needs to create the documentation and the whole system. The approximate budget and cost allocation for this project are indicated below:

The materials in Table 4.3 will be needed for printing the documentation of the study. Printed copies of the manuscript will be needed for proposal and final defense and for the safekeeping of the deliverable. The approximate total amount of the materials is Php 1,340.

Table 4.3. Materials and Supplies Budget

Item	Quantity	Price	Amount
Bond Paper (Hard Copy 70 gsm, 500 sheets, letter	2	Php. 170.00	Php. 340.00
Ink (Epson L3110 Ink 003)	1 set (4 pieces)	Php. 1,000.00	Php. 1,000.00
		Total Amount	Php. 1,340.00

In Table 4.4, Internet Service will be needed for the learning, searching information, hosting, and communication between the proponents and their adviser, professor, and the panelist. Plagiarism Checker using PrepostSEO will be needed at least a month for integrating plagiarism checker in the system. Web and Cloud Hosting will be free due to the proponents' sponsorship from the Student Developer Pack of GitHub. The amount needed for the services and

hosting expenses is Php 2,100.

Table 4.4. Services and Hosting Budget

Service	Fee (Monthly)	Amount
Internet Service	Php. 200.00	Php. 1,600.00
Plagiarism Checker API (PrepostSEO Basic Plan)	Php. 500.00	Php. 500.00
Web and Cloud Hosting	Free	Free
	Total Amount	Php. 2100.00

Overall, the approximate total expenses of this project will be Php 3,440. However, it does not include the personal expenses of the proponents like food, transportations, and rent. The source of the fund will be from the proponents' contribution.

4.9 Verification and Validation

The project will undergo verification and validation in order to confirm that the system made is in working properly and in accordance with the client's expectation. A survey, Likert scale, and an interview will also be used as a quantitative and qualitative measurement.

4.9.1 Verification

Since the project's system development life cycle involves an iterative process, the system will be verified and checked by the client per module over time. This will

include walkthroughs, inspection, and review of the design and the processes in the system. This will make sure that the application that whether the software conforms to the specification requirements.

4.9.2 Validation

The system will undergo validation where the actual system or modules are checked if it meets the requirements and expectations of the users. Validation will be made with the use of functional and non-functional validation of the system.

4.10 Testing

Testing will be done to check that the system works correctly without bugs and errors, especially in accepting inputs and the overall processes inside the system. Positive and Negative testing will be used to check if the system can accept correct data and decline inappropriate inputs.

CHAPTER V

Development and Implementation

BIBLIOGRAPHY

[1] Aman Agarwal. Software engineering | system design strategy. Available at https://www.geeksforgeeks.org/software-engineering-system-design-strategy/ (2021/05/11).

[2] Altexsoft. Functional and nonfunctional requirements: Specification and types. Available at https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/ (2019/05/29).

[3] Desislava Baeva. Management of graduate student's thesis writing through a specialized software system. *Journal of International Scientific Publications: Educational Alternatives*, 14(1000021):336–341, 2016.

[4] Bernd Bruegge and Allen H. Dutoit. *Object-Oriented Software Engineering Using UML, Patterns, and Java*. Prentice Hall Press, USA, 3rd edition, 2009.

[5] Mac Clark and Roger D. Boyle. A Personal Theory of Teaching Computing Through Final Year Projects. *Computer Science Education*, 9(3):200–214, 1999.

[6] Alibaba Clouder. How to create an effective technical architectural diagram? Available at https://www.alibabacloud.com/blog/how-to-create-an-effective-

technical-architectural-diagram_596100 (2020/04/10).

[7] World Wide Consortium. Html & css. Available at https://www.w3.org/standards/webdesign/htmlcss.html (2015/09/22).

[8] Noble Desktop. What is git & why should you use it? Available at https://www.nobledesktop.com/blog/what-is-git-and-why-should-you-use-it (2018/09/21).

[9] Mathieu Dionne. Reasons why javascript is omnipresent in modern development. Available at https://snipcart.com/blog/why-javascript-benefits (2019/04/11).

[10] Kate Eby. The power of iterative design and process. Available at https://www.smartsheet.com/iterative-process-guide (2019/01/02).

[11] Robert Gibb. What is a web application? Available at https://blog.stackpath.com/web-application/ (2016/05/31).

[12] Glossaria. Object oriented analysis and design - object oriented analysis and design (glossary) © http://www.glossaria.net. Available at http://www.glossaria.net/en/object-oriented-analysis-and-design/ (2019/12/07).

[13] Annegret Goold. Providing Process for Projects in Capstone Courses. *ACM Sigcse Bulletin*, 35(3):26–29, 2003.

[14] James Grooms, Douglas Kline, and Jeffrey Cummings. Streamlining the capstone process: A time-saving approval system for graduate theses/projects. *Information Systems Education Journal*, 14(4):81, 2016.

[15] James Grooms, Douglas Kline, Jeffrey Cummings, and Devon Simmonds. A web application for capstone management for the mscsis program. *Annals of the Master of Science in Computer Science and Information Systems at UNC Wilmington*, 8(1), 2014.

[16] Jatinder N D Gupta and Robert M Wachter. A capstone course in the information systems curriculum. *International Journal of Information Management*, 18(6):427–441, 1998.

[17] Omar Talal Hamid. Design and implementation a novel research management system. *Cihan University-Erbil Scientific Journal*, 3(2):21–24, 2019.

[18] Javatpoint. Iterative model. Available at https://www.javatpoint.com/software-engineering-iterative-model (2019/02/16).

[19] Javatpoint. Sdlc - iterative model. Available at https://www.tutorialspoint.com/sdlc/sdlc_iterative_model.htm (2013/05/07).

[20] Mike Joy. Undergraduate computing projects — an investigation into the student experience. *Innovation in Teaching and Learning in Information and Computer Sciences*, 8(1):60–78, 2009.

[21] Julie Korsun. Why we use django framework & what is django used for. Available at https://djangostars.com/blog/why-we-use-django-framework/ (2020/03/03).

[22] Dan Chia-Tien Lo and Orlando Karam. Enhance capstone projects with a new online collaboration system. In *2013 IEEE 13th International Conference on Advanced Learning Technologies*, pages 217–218. IEEE, 2013.

[23] Brett D McLaughlin, Gary Pollice, and Dave West. *Head First Object-Oriented Analysis and Design: A Brain Friendly Guide to OOA&D (Head First)*. O'Reilly Media, Inc., 2006.

[24] Juan José Olarte, César Domínguez, and Francisco José García-Izquierdo. A tool for capstone project management in computer science engineering. In *2014 International Symposium on Computers in Education (SIIE)*, pages 65–68. IEEE, 2014.

[25] Juan José Olarte, César Domínguez, Francisco José García-Izquierdo, and Arturo Jaime. Capstone projects in computer science: evaluated by stakeholders. In *Proceedings of the 2014 conference on Innovation & technology in computer science education*, pages 345–345, 2014.

[26] Commission on Higher Education. Revised policies, standards, and guidelines for bscs, bsis, and bsit programs. Memorandum Order Number 25, 2015.

[27] Michael Orozco. Web-based thesis/capstone project defense evaluation system of the css binan. Unpublished Manuscript, 2016.

[28] Philip Owusu-Afriyie. Capstone management system. *Ashesi Institutional Repository*, 2020.

[29] Malika Pathak. What is a cloud-based system? Available at https://www.quora.com/What-is-a-cloud-based-system (2020/02/26).

[30] César Domínguez Pérez, Arturo Jaime Elizondo, Francisco J García-Izquierdo, and Juan José Olarte Larrea. Supervision typology in computer science engineering capstone projects. *Journal of Engineering Education*, 101(4), 2012.

[31] John Prabhu. 8 reasons why you should use bootstrap. Available at https://techaffinity.com/blog/why-use-bootstrap-for-frontend-design/ (2020/05/26).

[32] Mohammad Thalhah Amir Bin Abdul Rabman. *UMP Thesis Management System*. PhD thesis, Universiti Malysia Pahang, 2012.

[33] Hamilton Scott. Management of real-world projects in university computing courses. *Proceedings of Student-Faculty Research Day*, pages A21–A27, 2008.

[34] Palak Shah. What all you need to know about html5. Available at https://www.opensourceforu.com/2017/06/introduction-to-html5/ (2017/06/17).

[35] Simplilearn. What is requirement analysis: Overview, applications, techniques and top tools used. Available at https://www.simplilearn.com/what-is-requirement-analysis-article (2020/12/02).

[36] Chaitanya Singh. Entity relationship diagram - er diagram in dbms. Available at https://beginnersbook.com/2015/04/e-r-model-in-dbms/ (2018/11/16).

[37] Mindfire Solutions. Python: 7 important reasons why you should use python. Available at https://medium.com/@mindfiresolutions.usa/python-7-important-reasons-why-you-should-use-python-5801a98a0d0b (2017/10/03).

[38] Mindfire Solutions. What is pycharm ide? Available at https://medium.com/@mindfiresolutions.usa/what-is-pycharm-ide-cc0735784f64 (2018/04/17).

[39] SQLite.Org. What is sqlite? Available at https://www.sqlite.org/index.html (2018/04/17).

[40] CORPORATE The Joint Task Force on Computing Curricula. Computing curricula 2001. *Journal on Educational Resources in Computing (JERIC)*, 1(3es):1–es, 2001.

[41] Leoven Torrechiva, Marc Christian Orais, Ken Gabaca, Ronald Ian Gabutan, Jhury Espinosa, Celso Malinao Jr, and Melvin Niñal. Online uc-cics capstone projects monitoring and management system. Unpublished Manuscript, 2012.

[42] Tutorialspoint. Design strategies. Available at https://www.tutorialspoint.com/system_analysis_and_design/system_-analysis_and_design_strategies.htm (2017/01/03).

[43] Tutorialspoint. Object oriented approach. Available at https://www.tutorialspoint.com/system_analysis_and_design/system_-analysis_and_design_object_oriented_approach.htm (2021/05/22).

[44] Henrik von Scheel, Mark von Rosing, Maria Hove, Marianne Fonseca, and Ulrik Foldager. Phase 2: Process concept evolution. In Mark von Rosing, August-Wilhelm Scheer, and Henrik von Scheel, editors, *The Complete Business Process Handbook*, pages 11–35. Morgan Kaufmann, Boston, 2015.

[45] Yanhua Zong. *A Thesis Evaluation System*. PhD thesis, Purdue University Graduate School, 2019.

APPENDIX A

Wireframe

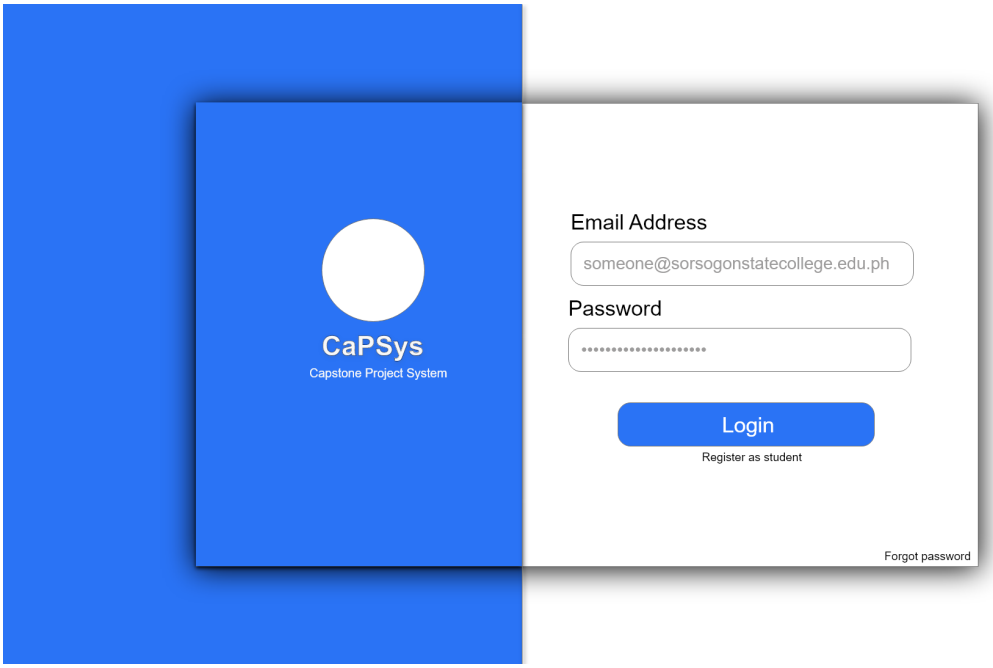

Figure A.1. Login Wireframe Design

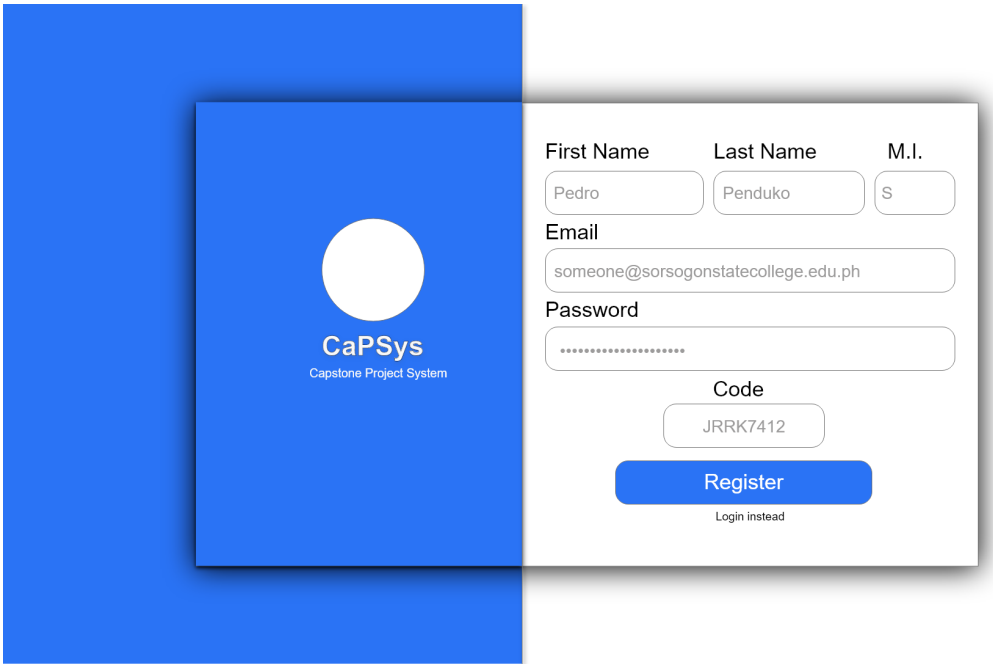

Figure A.2. Register Wireframe Design

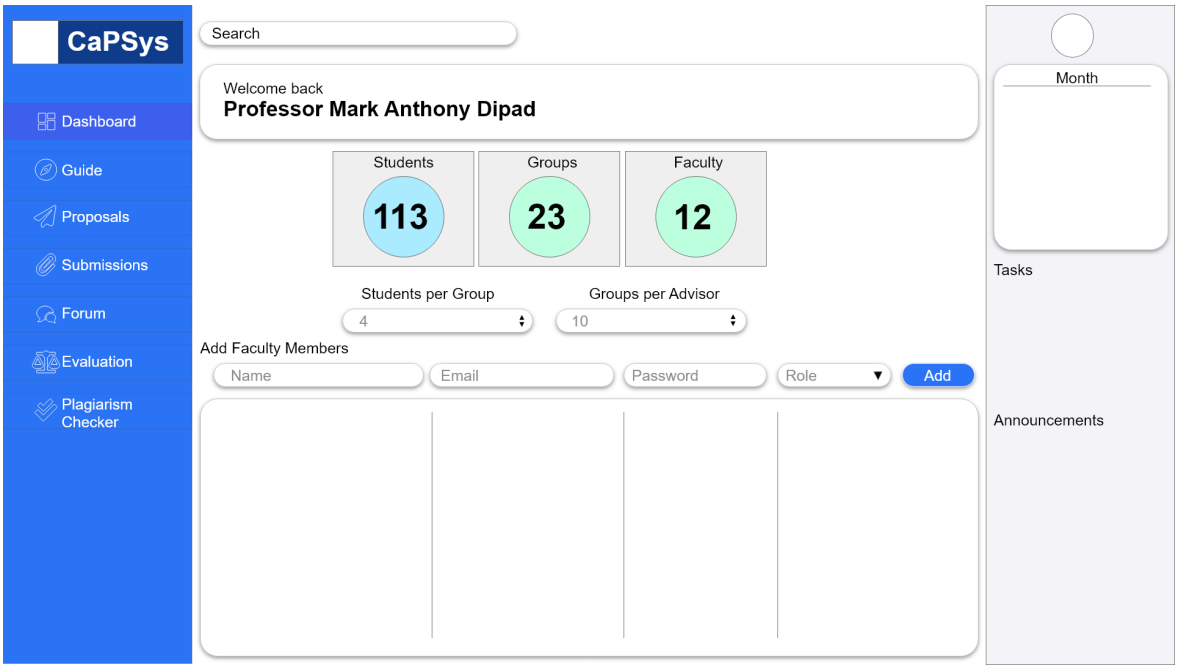

Figure A.3. Dean Dashboard Wireframe Design

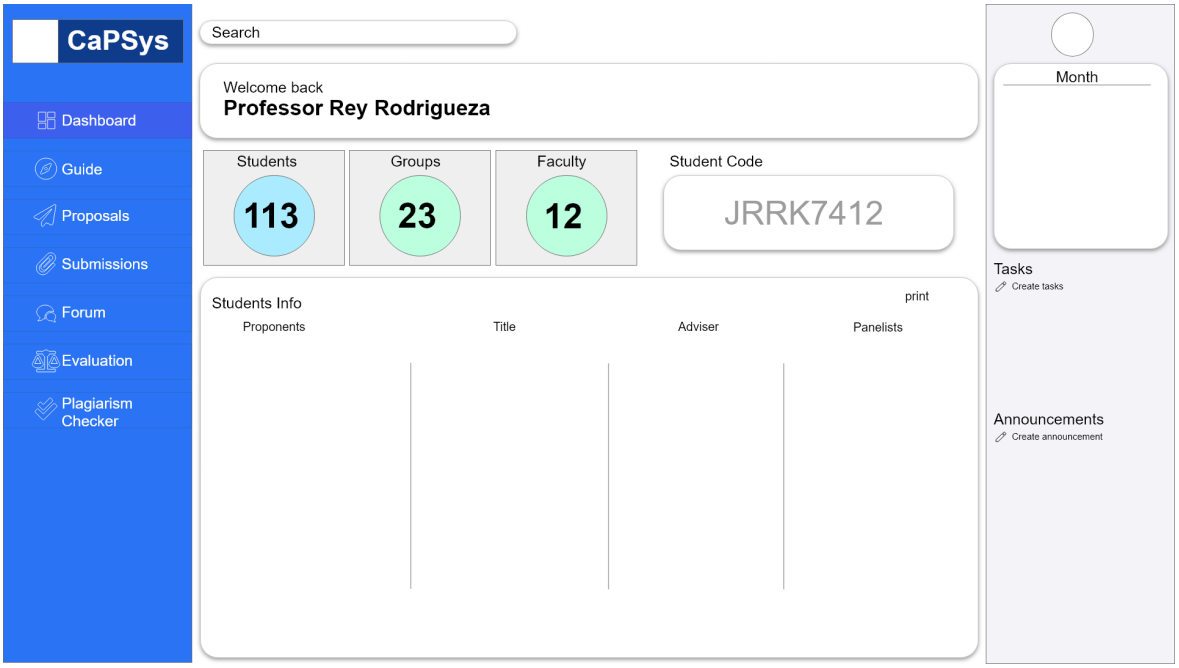

Figure A.4. Capstone Professor Dashboard Wireframe Design

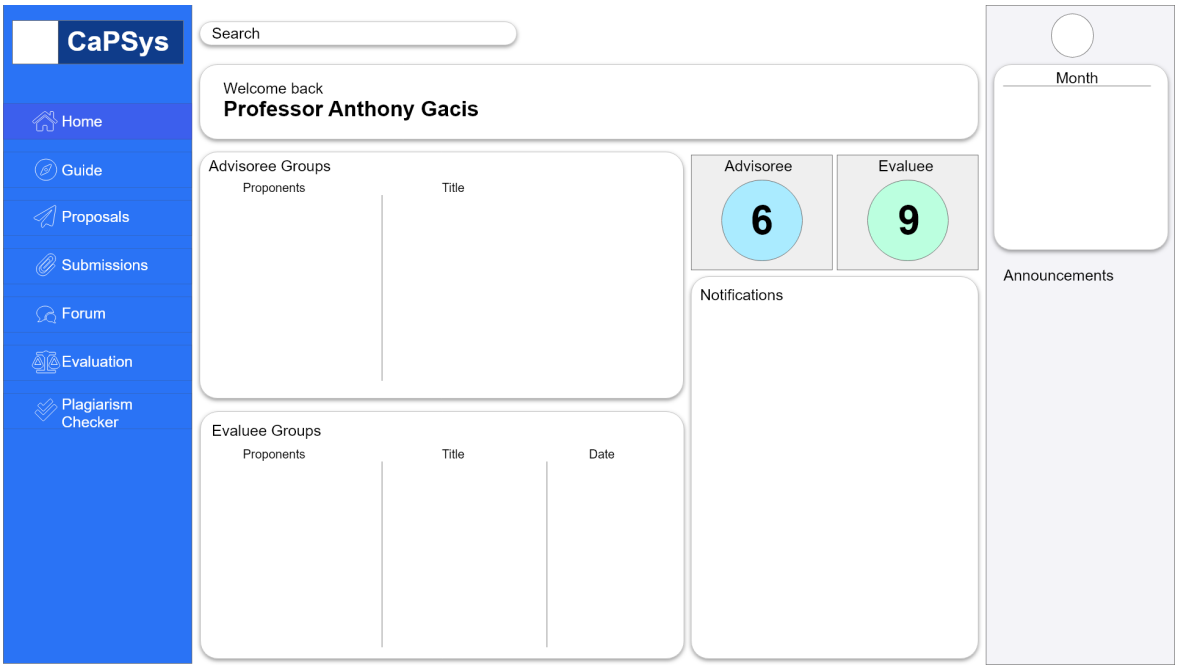

Figure A.5. Faculty Home Wireframe Design

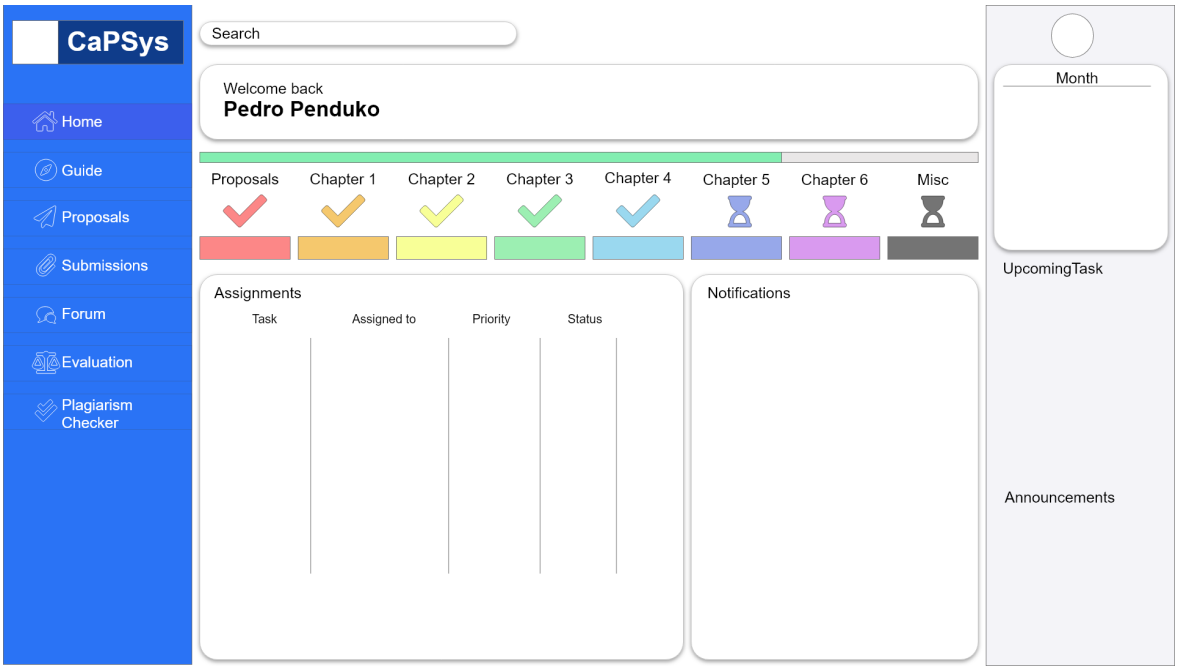

Figure A.6. Student Home Wireframe Design

APPENDIX B

Use Case Table Description

Use Case Name	Login
Use Case Overview	A user login to System to access the functionality of the system
Actor(s)	Students, Faculty Members, Capstone Professor, Department Dean
Precondition(s)	1. System must be connected to the network 2. Users' credentials are registered to the database
Trigger(s)	1. First time accessing the site 2. Redirect after registering 3. Logging out from the system
Step(s)	1. Enter Username 2. Enter Password 3. Click Login Button
Post Condition(s)	User is logged into appropriate homepage

Figure B.1. Login Use Case Description

Use Case Name	Register
Use Case Overview	Users registers to the application and provide a logon username and password
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network
Trigger(s)	Clicked Register Instead link
Step(s)	Enter Appropriate user information
Post Condition(s)	1. User information stored and registered to the database 2. Redirect to login page

Figure B.2. Register Use Case Description

Use Case Name	Discussion Forum
Use Case Overview	Users can raise queries or discussions and comment on posts
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked Discussion Forum Navigation Bar
Step(s)	Capstone Professor: 1. Create a Discussion Forum Topic Others: 1. Select a Discussion Forum Topic 2. Create A Discussion under the topic 3. Reply to the existing discussions
Post Condition(s)	Users will be able to start a discussion, post and comment in the forum

Figure B.3. Discussion Forum Use Case Description

Use Case Name	Guide and Formatting
Use Case Overview	Users view the guide and formatting standard in writing capstone project manuscript
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked Guide and Formatting Navigation Bar
Step(s)	Capstone Professor: 1. Upload file/Input guides and formatting Others: 2. View or download the uploaded file or text
Post Condition(s)	Users will be able to view the guide and formatting

Figure B.4. Guide and Formatting Use Case Description

Use Case Name	Task, Activities, and Deliverables Management
Use Case Overview	Users can create task or activities where students will submit their documents
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked Submission Navigation Bar
Step(s)	Capstone Professor: 1. Create a submission form and input necessary details Students: 2. Click on the created submission form 3. Submit necessary documents on the created submission forms Faculty Members, Capstone Professor: 1. View or download students' submitted documents
Post Condition(s)	1. Capstone Professor created a submission form 2. Student successfully submitted their documents 3. Faculty members able to view the submitted documents

Figure B.5. Task, Activities, and Deliverables Management Use Case Description

Use Case Name	Topic/Title Proposal
Use Case Overview	Users will be able to propose topic or title as well as view the said proposal and suggest.
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked the Title Proposal Tab in the Proposal Navigation Bar
Step(s)	Students: 1. Input title and description of the study Faculty: 1. View, comment and approve submitted title
Post Condition(s)	Students have an approved title

Figure B.6. Topic/Title Proposal Use Case Description

Use Case Name	Adviser and Panelists Selection
Use Case Overview	Students can select their adviser and panelist
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked the Adviser/Panelist Proposal Tab in the Proposal Navigation Bar
Step(s)	Students pick their adviser and panelist from the list of faculty members registered in the system
Post Condition(s)	Students now have an adviser and panelist

Figure B.7. Adviser and Panelists Use Case Description

Use Case Name	Plagiarism Checker
Use Case Overview	Be able to detect manuscript for plagiarism
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked Submission Navigation Bar
Step(s)	Users paste their manuscript or upload their document
Post Condition(s)	Users will know the percentage and location of the plagiarized content and

Figure B.8. Plagiarism Checker Use Case Description

Use Case Name	User Management
Use Case Overview	Manage all
Actor(s)	Capstone Professor, Department Dean
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Logged in the Administrator Site
Step(s)	1. Grant or remove privileges 2. View or delete registered users
Post Condition(s)	Users will be able to register, login and access the right information in accordance to their role or privileges

Figure B.9. User Management Use Case Description

Use Case Name	Capstone Project Manuscript Monitoring
Use Case Overview	View the students' progress based on the submitted documents
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked the dashboard navigation bar
Step(s)	View the students' progress by going to the dashboard
Post Condition(s)	Thru percentage, the users will be able to view the students' progress

Figure B.10. Capstone Project Manuscript Monitoring Use Case Description

Use Case Name	Student Management
Use Case Overview	Generate list of students' information
Actor(s)	Capstone Professor, Department Dean
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked the Students, Groups, Adviser Information in the dashboard
Step(s)	1. View the information from the lists 2. Edit the list 3. Print or download the document
Post Condition(s)	Users will be able to view, edit, and print students' information

Figure B.11. Student Management Use Case Description

Use Case Name	Presentation and Defense Scheduler
Use Case Overview	Users should be able to input their desired time and date for presentation
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked Presentation tab in the Evaluation Navigation Bar
Step(s)	Capstone Professor: 1. Inputs the available time and date for presentation Capstone Adviser: 1. Selects a time and date to schedule the student group for presentation
Post Condition(s)	Students has their schedule of presentation

Figure B.12. Presentation and Defense Sceduler Use Case Description

Use Case Name	Manuscript Evaluation
Use Case Overview	Panelists can view, comment, grade the submitted manuscript of the students
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked Manuscript tab in the Evaluation Navigation Bar
Step(s)	1. Students submits their final paper for presentation 2. Panelists view or download the submitted document 3. Panelists comments for suggestions 4. Panelists inputs rating for the manuscript
Post Condition(s)	Manuscript has comments as well as ratings

Figure B.13. Manuscript Evaluation Use Case Description

Use Case Name	Grades Management
Use Case Overview	User can view the result or their grades from their evaluation
Actor(s)	Students, Faculty Members, Capstone Professor
Precondition(s)	System must be connected to the network User must be logged in to the system
Trigger(s)	Clicked Grade Tab in the Evaluation Navigation Bar
Step(s)	Clicked Grade Tab in the Evaluation Navigation Bar
Post Condition(s)	Users can view their grade as well as the suggestion and recommendations of their panelists

Figure B.14. Grades Management Use Case Description