BTIS3102

FINAL YEAR PROJECT I

University Content Management System (Final Report)

Lam Wei Chun(B180279C)

Supervisor: Mrs. Sim Hiew Moi

8th September 2020

Department of Computer Science

Faculty of Engineering and Information Technology

ABSTRACT

The University Content Management System is a web-based system that allows Dean and Lecturers to track, manage and store university documents and reduce paper usage. Admin can manage university data such as staff, student, program, etc. Lecturer can put teaching materials on the system so that students can watch and download them at any time. Lecturer can also develop teaching plans and assessments so that students can browse it. Dean can use this system to check whether Lecturer's teaching methods meet official government requirements. Lecturer also needs to be a moderator for review other Lecturer courses, such as whether the assessment level of the course meets and give some other opinions. Lecturer can also take student attendance about the course through this system. Students also need to store the examination papers reviewed by Lecturer by taking photos, so that the university does not need to store student examination papers in paper documents.

Keyword: University Data, teaching materials, teaching plans, assessments, check teaching methods, moderator, attendance, taking photos.

Acknowledgements

I would like to express my deepest gratitude to everyone who provided me with advice on completing this report. I would like to thank my lecturer Mrs. Sim Hiew Moi for giving me the opportunity to make this system, which is mainly to help universities and colleges to solve the current paper document problems. Mrs. Sim also provided me with numerous suggestions to help improve this project. Mrs. Sim always provides the best solution, and advises and helps me to have a better understanding of today's systems. In addition, I want to thank my classmates and friends for helping me when they encounter problems. Although they are busy completing tasks and projects.

Contents

ABSTRACT i

Acknowledgements ii

1.0 Introduction 1

2.0 Literature review 4

3.0 Methodology 12

4.0 Design 14

4.1 DFD diagram 14

4.1.1 Context level 14

4.1.2 DFD Level 0 15

4.1.3 DFD Level 1 16

4.2 ER-diagram 21

4.3 Class diagram 23

4.4 Use case diagram 25

4.4.1 Information Modules 25

4.4.2 Moderator Modules 26

4.4.3 Course Modules 27

4.4.4 Course Portfolio Modules 28

4.4.5 Faculty Portfolio Modules 29

4.5 Use case description 30

4.5.1 Information Modules 30

4.5.2 Moderator Modules 35

4.5.3 Course Modules 41

4.5.4 Course Portfolio Modules 50

4.5.5 Faculty Portfolio Modules 54

4.6 Activity diagram 57

4.6.1 Information Modules 57

4.6.2 Moderator Modules 58

4.6.3 Course Modules 59

4.6.4 Course Portfolio Modules 60

4.6.5 Faculty Portfolio Modules 61

4.7 Sequence diagram 62

4.7.1 Information Modules 62

4.7.2 Moderator Modules 63

4.7.3 Course Modules 64

4.7.4 Course Portfolio Modules 65
4.7.5 Faculty Portfolio Modules.. 66
5.0 Implementation 67
6.0 Testing 69
7.0 Evaluation... 79
8.0 Recommendations... 80
9.0 Conclusion 81
References 82
APPENDIX A – Curriculum Vitae.. 83
APPENDIX B – User Guide 85
APPENDIX C – Project Diaries 109

1.0 Introduction

Nowadays, with the rapid development of technology to the high-tech era, we have also begun to gradually accept the paper documents change to use electronic documents for storage. In the past, there are still many colleges that use paper to store information. Of course, there will be no problems at the beginning, but as time slowly builds up, the problems will become very apparent such as waste space, difficult searching, and other. Therefore, to effectively manage the growing paper documents, a paper-based document management system is needed. Since we are hooked on smartphones, tablets, etc., these devices now play an important role in electronic documents. It turns out that books are gradually being replaced by electronic teaching materials such as CD or USB. Therefore, University Content management system is essential for university colleges.

The University Content management system is a storing document system which is developed for Dean, head of department (HOD), Lecturers and Students. The University Content management system is a web-based system and it is accessed by the internet connection to receive, track, manage and store documents and reduce paper. So, the University Content management system will focus on Five target user such as the Admin, Dean, HOD, Lecturer and Student.

There is no doubt that paper document cannot be replaced in completely. However, the University Content management system can effectively, properly create, share, organize and store information. This can help paper-based documents solve problems facing them. The most important thing is to reduce paper documents. Most paper documents are Assignments, Test, Quiz and Final exam questions of some courses, which can be replaced by system archives. The system can also let lecturer put the teaching materials on the system, so that students can download and quote for free anytime, anywhere. Lecturers can also put teaching plans on the system so that students can follow or track. Dean and HOD can easily monitor lecturer and student performance.

1.1 Project Objective

After developing this project, I can achieve the objectives as shown as below:

- To provide a platform to let the student download the course materials in anywhere and anytime.
- To provide a platform to let the lecturer store the course materials and track himself to improve performance.
- To provide a platform to let the Dean and HOD more easily to track the lecturer and student performance.

1.2 Personal Objectives

After developing this project, I can achieve the personal objectives as shown as below:

- To increase my web programming skills.
- To improve the design and analysis skills of the project.
- To learn how to allocate the time and resources to the project and produce a well-planned schedule.
- To learn how to develop a formal report

1.3 Rationale for project choice

The reason I chose this project is that take Southern University College as an example. After the students handed over the assignment, they could not get it back. Because lecturers need use them to store paper-feeding documents. This causes students from reviewing back the assignment again. Plus, this led to the outbreak of paper documents in the college. University Content management system is a complex and detailed system, so the most important skills I need to improve are analysis and design skills. This can improve my programming skills and project management skills. Therefore, in this project, I want to learn how to design an efficient database and improve database design skills.

1.4 Background Research

With the explosive growth of internet, the information is no longer stored in paper archive. No longer can information be published in a manual process and left unattended. Nowadays, many universities still use paper-based documents to store materials. University Content Management System (UCMS) can help them very well. A Content Management System (CMS) is a computer application that allows content to be published, edited, modified, organized, deleted, and maintained from a central interface. A University Content Management System (UCMS) is a system for managing university content. This content management system provides the process of managing workflow in a collaborative environment.

The primary role of UCMS is to manage digital assets used for developing learning products. These systems provide a database called a course content that will save the lecturer of course materials, which can be accessed by moderator or Dean to audit it. Students can also download related course materials from this system.

- Problem Statement

  I developed this (UCMS) based on our Southern University College. My University College does not yet have a relatively complete (CMS) to store data, and most of them use paper documents to store. This leads to:

  1. A lot of paper documents, which takes up a lot of space.
  2. Finding information is very troublesome.
  3. Easy to lose.
  4. Most of these documents are student assignments, test questions and other, which results in students having no way to review back it.

- Challenge
  1. Too much information

     The University College has many information need to manage and store. I need to assign it very clearly, otherwise it is easy to cause confusion and loss.
  2. Class issue

     In university colleges, every employee has his own position, and each position corresponds to a different function. I need to know very well about every employee of function appears in my system and make corresponding functions.

2.0 Literature review

In this part, I am going to explore the products related to my project that have been conducted by other academic previously. University Content Management System an online system or software which is used to manage content of university. It helps in administration and tracking. In this regard, I explored two system of research paper related to my project, which are Content Management System(CMS) and Document management system (DMS).

2.1 CMS VS DMS

Many people think that there is no difference between this two system, which may cause confusion when it comes to the number of solutions, especially when the boundaries between solutions are not well defined. Document Management Systems (DMS) and Content Management Systems (CMS) offer similar features, but they also have some key differences. Generally, a DMS works to help create, track, and store digitized documents. These systems retain, classify, and protect electronic data. In addition to this, a DMS supports collaboration, versioning, and workflows. However, a CMS creates and manages different kinds of digital content, besides more traditional documents. Rather than solely managing PDFs, Excel files, or Word files, a CMS can manage web pages, images, flash files, and records.

Document management systems and Content management systems both do the following:

- Function of CMS and DMS in Similar

    1. **Both Are SaaS-Based Technology**

        a. SAAS-based technology, also known as software as a service technology, is a method of software delivery that is hosted on third-party servers and makes them available to users over the internet.

        b. Both DMS and CMS can be hosted on a third-party server whereby all the data are safely stored in the cloud. The data can be accessed from any device having a secure internet connection.

2. **Acts as a Centralized Data Storage**

    a. In terms of similarities, both offer a centralized database whereby all the information about your staff, business, and other confidential information are stored. These documents are easily accessible, and the plus point is that you can restrict the access of the database to individual roles and actions.

3. **Enables you to Easily Manage, Store, and Retrieve Information**

    a. Both DMS and CMS are capable of systematically collecting and storing data. You can index the documents so that you can effortlessly search the files you need. The information doesn't get lost in the piles of files stored in the system. One core benefit is that both the system allows you to search and retrieve the information needed without any hassle.

4. **High-Level Security**

    a. Another similarity between the two is that both offer high security to preserve and safeguard your confidential business data. It prevents the system from any unauthorized access. Moreover, in case you delete your files accidentally, you can also retrieve them back. The cloud-based technology creates an automatic back up of your data.

Despite the similarities, DMS and CMS tools differ in a number of ways. Each has some key features that the other doesn’t have, and they deal with different types of data.

- Function of CMS and DMS in difference
    1. A DMS manages structured data and is focused on documents in the traditional sense in such formats as Word, PDF, PowerPoint, Excel, etc. A CMS, on the other hand, can handle both structured data and unstructured data, such as web content (HTML and PDF files) and digital assets (images and audio and video files).
    2. The key purposes of a DMS are regulatory compliance and workflow management, while the key purposes of a CMS are storage, retrieval and publishing of content.

3. DMS applications have advanced imaging and scanning capabilities, such as optical character recognition (OCR), handprint character recognition (HCR), optical mark recognition (OMR) and more. CMS tools usually don’t support those functions.
4. Integration with enterprise systems (such as enterprise resource planning and customer relationship management tools) is essential for a DMS but secondary for a CMS.

In this regard, our university colleges do not have a more complete Document Management System. So, I will use some of the functions of these two management systems to build a more complete University Content Management System. Many photos and documentation will be stored in this system, I will use scanning capabilities, such as optical character recognition (OCR) to help the lecturers and students to find some relevant information. This system will also store some course materials in PDFs, Excel files, or Word files. Hence, the University content management system is more difficult to handle than I thought.

2.2 Research

2.2.1 Laravel

Figure 1.2.1: Logo of Laravel

Laravel is a free and open source PHP Web framework created by Taylor Otwell , designed to implement the MVC architecture of Web software , and as an alternative to CodeIgniter . Since Laravel has been developed for a long time, it has many discussions and solutions, so it makes it easier for me to develop University Content management system. Moreover, my system is a web-based system that will require PHP for server-side scripting.

2.2.2 *MySQL*

Figure 2.2.2: Logo of MySQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language.

MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. Since it is free and open source, I will use MySQL as my database because its development technology is relatively mature and more familiar to me.

2.2.3 Web Service

A **Web Service** is a client server application or application component for communication. It is also a method of communication between two devices over network. Moreover, web service is a collection of standards or protocols for exchanging information between two devices or application. For example, java application can interact with Java, .Net and PHP applications. So, web service is a language independent way of communication.

2.3 Outline of scope of the system

Project Title: University Content Management System
Project Justification: Provide management system for dean and lecturers to store the university content.
Product Characteristics and Requirements: 1. Information Modules 2. Moderator Modules 3. Course Portfolio Modules 4. Faculty Portfolio Modules 5. Report Modules
Detail **1. Information Modules** • Add, view, edit and delete the Faculty details. • Add, view, edit and delete the Department details. • Add, view, edit and delete the Programme details and subject details. • Add, view, edit and delete the MPU subject details. • Add, view, edit and delete the Semester details. • Add, view, edit and delete the Student details. • Add, view, edit and delete the Staff details. • Student and Staff can edit own personal data in own portal. **2. Moderator Modules** • Review specified of courses. • Fill in the Continuous Assessment Moderation Form about the course. • Fill in the Internal / External Moderation of Final Examination paper. **3. Course Modules** • The new course is generated by the course portfolio modules. • Assign students in own course. • Add the teaching plan about the course. • Upload the assessment question in image to system. • Search keyword for question using by OCR. • Store the course materials such as Lecture Note. • Download the course materials in Zip Files or Single File.

- Student can submit the assessment in image that marked by lecturer.
- Generate the QR code for student take attendance himself.
- Scan the QR code for take attendance.

4. **Course Portfolio Modules**
   - Add, view, edit and delete the course portfolio.
   - Generate the E-Portfolio Form.
   - Generate all the report about the course such as moderator report.
5. **Faculty Portfolio Modules**
   - Store the faculty materials such as activities and other.
   - View the Lecturer CV and Syllabus about that faculty.

My project focuses on the basic University Content Management system, which consists of five modules, which are Information modules, Moderator Modules, Course Portfolio modules, Faculty Portfolio modules and Report modules. In Information modules allow admin to add new staff and student. Staff is dean, Head of department and Lecturer. Before create the staff need to select the position, faculty and department. That also need to upload the Lecturer CV for each staff. After the current semester completed, the admin need to create a new semester for all new course. The Information modules allow admin to add new faculty and department, who also can add the new programme and assign the subjects in each programme. The MPU subject will assign to the programme based on the programme level. The student and staff can update the own personal data in own position portal.

In Moderator module is display the course that need to review by specify Lecturers. The moderator is assigned by dean in create new course portfolio function. Specify Lecturer need to fill in the Continuous Assessment Moderation Form about the course and fill in the Internal / External Moderation of Final Examination paper.

In Course module, the course is generated by the course portfolio modules. Each Course the lecturer need to write down the teaching plan, let the student can follow it and Dean or HOD can view it. The Lecturer can enter students into that course. In this way, students do not need to enroll the subject themselves. Lecturers can upload the course materials such as lecture note to system. Lecturers also need to upload the assessment question in image to system. Lecturers and students can search question by keyword from assessment of past year. Students can download the course materials in zip file or single file on this modules. Lecturers can generate the QR code for student, let they can take attendance by themself or take attendance in manually.

In Course Portfolio module allow the dean to add the new course. In this module, Dean or HOD can generate the e-portfolio form report to view whether the course portfolio is complete or not. Dean or HOD also can view the result of the moderator report such as Continuous Assessment Moderation Form or Internal / External Moderation of Final Examination paper.

The Faculty Modules allow Dean to store the new portfolio about own faculty to system. Dean can view the faculty portfolio such as Lecturer CV and Syllabus.

2.4 Financial Justification

	Item	**Qty**	**Total(RM)**
Hardware	**ASUS Gaming computer**	**1**	**RM 4200.00**
Software	**Microsoft Office**	**1**	**RM 150.00**
	Microsoft Visio 2010	**1**	**RM 30.00**
	Sublime Text	**1**	**RM 0.00**
	MySQL	**1**	**RM 0.00**
	Laravel	**1**	**RM 0.00**
Other	**Transportation Fee**	**-**	**RM 300.00**
	Internet Subscription Fee	**-**	**RM 300.00**
	Printing Cost	**-**	**RM 500.00**
		Total:	**RM 5480.00**

2.5 Project Schedule

ID	Task Name	Start	Finish	Duration	Jun 2020				Jul 2020				Aug 2020	
					7/6	14/6	21/6	28/6	5/7	12/7	19/7	26/7	2/8	9/8
1	Stage 1 Project Initializing	6/8/2020	6/22/2020	11d										
2	1.1 Discussion topic	6/8/2020	6/8/2020	1d										
3	1.2 Define project topic	6/10/2020	6/11/2020	2d										
4	1.3 Define project objective	6/12/2020	6/12/2020	1d										
5	1.4 Background Research	6/12/2020	6/15/2020	2d										
6	1.5 Literature review	6/16/2020	6/16/2020	1d										
7	1.6 Define project scope	6/17/2020	6/17/2020	1d										
8	1.7 Curriculum Vitae	6/18/2020	6/18/2020	1d										
9	1.8 Gantt Chart	6/19/2020	6/19/2020	1d										
10	1,9 Submit Stage 1 report	6/22/2020	6/22/2020	1d										
11	2. Stage 2 Investigation & Analysis	6/23/2020	7/21/2020	21d										
12	2.1 Create DFD	6/24/2020	6/25/2020	2d										
13	2.2 Create Use Case	6/25/2020	6/29/2020	3d										
14	2.3 Create Use Case Description	6/29/2020	7/2/2020	4d										
15	2.4 Create ERM	7/2/2020	7/10/2020	7d										
16	2.5 Create Activity Diagram	7/10/2020	7/16/2020	5d										
17	2.6 Create Sequence Diagram	7/16/2020	7/21/2020	4d										
18	2.7 Submit Stage 2 Report	7/21/2020	7/21/2020	1d										
19	3. Requirement Prototype and System	7/22/2020	8/12/2020	16d										
20	3.1 Design User Interface	7/22/2020	7/23/2020	2d										
21	3.2 Develop User Management System	7/23/2020	7/28/2020	4d										
22	3.3 Develop Course Management System	7/28/2020	8/4/2020	6d										
23	3.4 Develop Online Learning System	8/4/2020	8/11/2020	6d										
24	3.5 Finalizing the report	8/11/2020	8/11/2020	1d										
25	3.6 Submit Stage 3 Report	8/12/2020	8/12/2020	1d										

3.0 Methodology

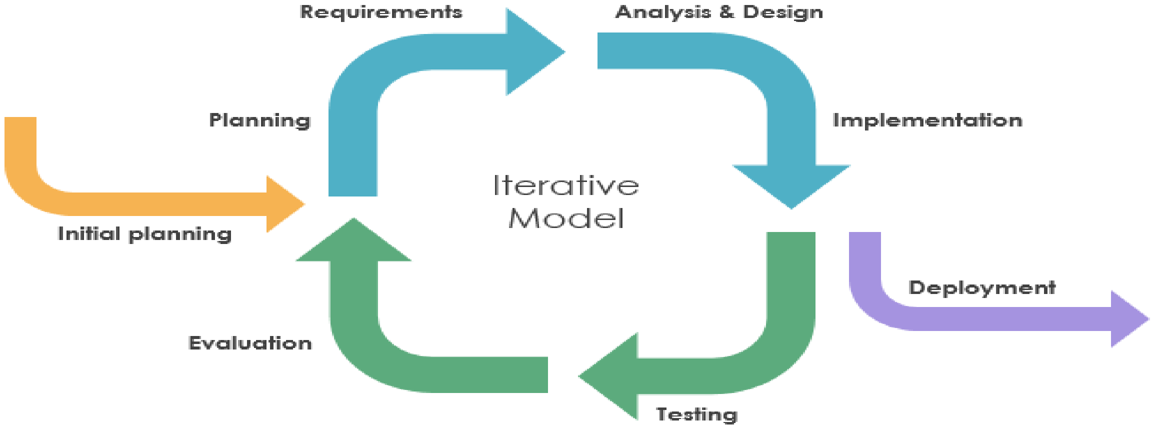


Figure 2 Iterative model

Since I am not familiar with the requirements of University Content Management System, so I will use the iterative model to develop the system. According to the iterative model, an iterative life cycle model does not start with a full specification of requirements. In this model, the development begins by specifying and implementing just part of the software, which is then reviewed in order to identify further requirements. Moreover, in iterative model, the iterative process starts with a simple implementation of a small set of the software requirements, which iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed. Each release of Iterative Model is developed in a specific and fixed time period, which is called iteration.

The process of Iterative Model is cyclic, unlike the more traditional models that focus on a rigorous step-by-step process of development. So, the iterative model is suitable to make changes in requirements when I have misunderstood the requirements of the University Content Management System. In this process, once the initial planning is complete, a handful of phases are repeated again and again, with the completion of each cycle incrementally improving and iterating on the software. Other phases of the iterative model are described below:

1. **Planning Phase:** This is the first stage of the iterative model, where proper planning is done by the team, which helps them in mapping out the specifications documents, establish software or hardware requirements and generally prepare for the upcoming stages of the cycle.

2. **Analysis and Design Phase:** Once the planning is complete for the cycle, an analysis is performed to point out the appropriate business logic, database models and to know any other requirements of this particular stage. Moreover, the design stage also occurs in this phase of iterative model, where the technical requirements are established that will be utilized in order to meet the need of analysis stage.

3. **Implementation Phase:** This is the third and the most important phase of the iterative model. Here, the actual implementation and coding process is executed. All planning, specification, and design documents up to this point are coded and implemented into this initial iteration of the project.

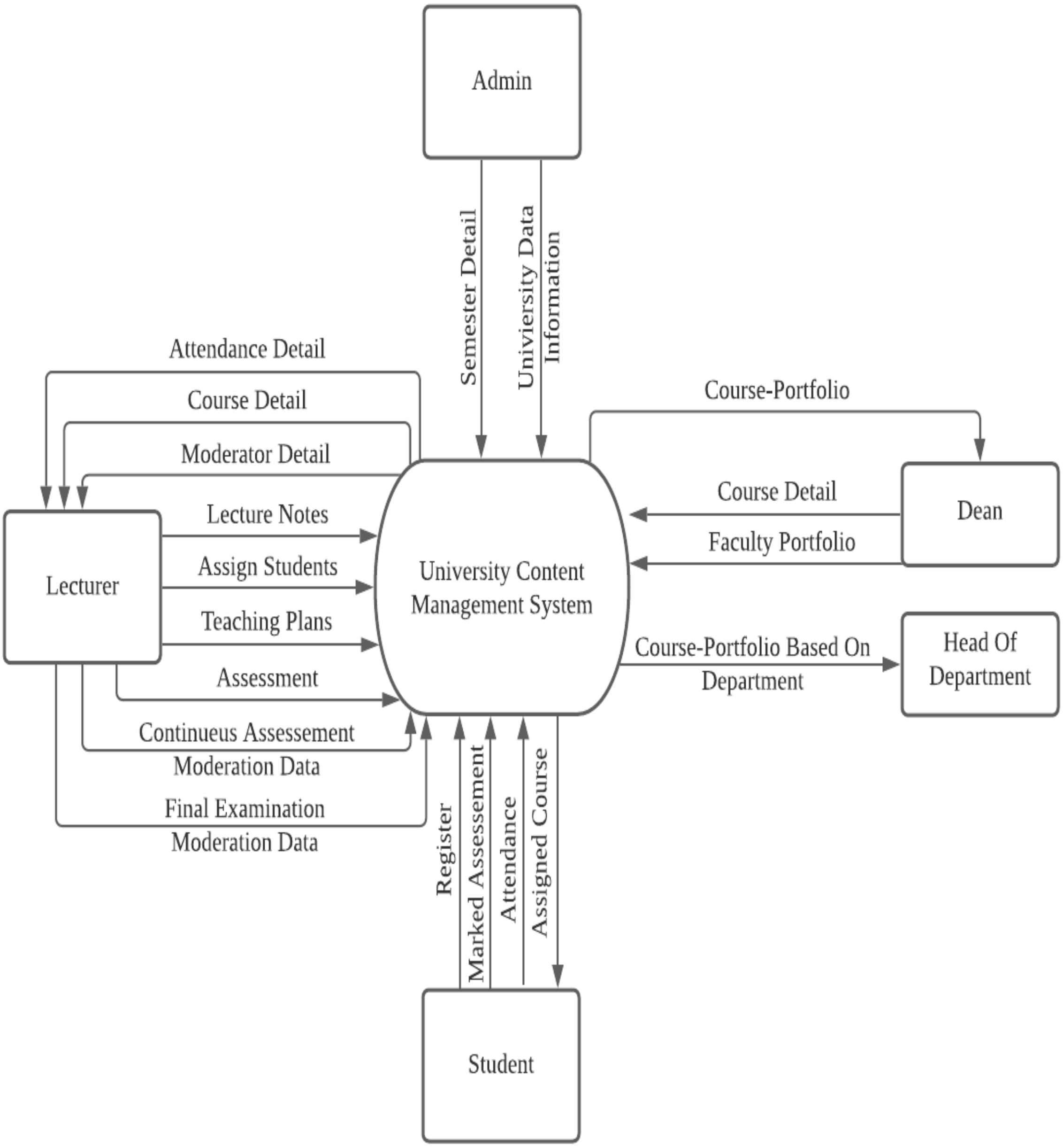
4. **Testing Phase:** After the current build iteration is coded and implemented, testing is initiated in the cycle to identify and locate any potential bugs or issues that may have been in the software.

5. **Evaluation Phase:** The final phase of the Iterative life cycle is the evaluation phase, where the entire team along with the client, examine the status of the project and validate whether it is as per the suggested requirements.

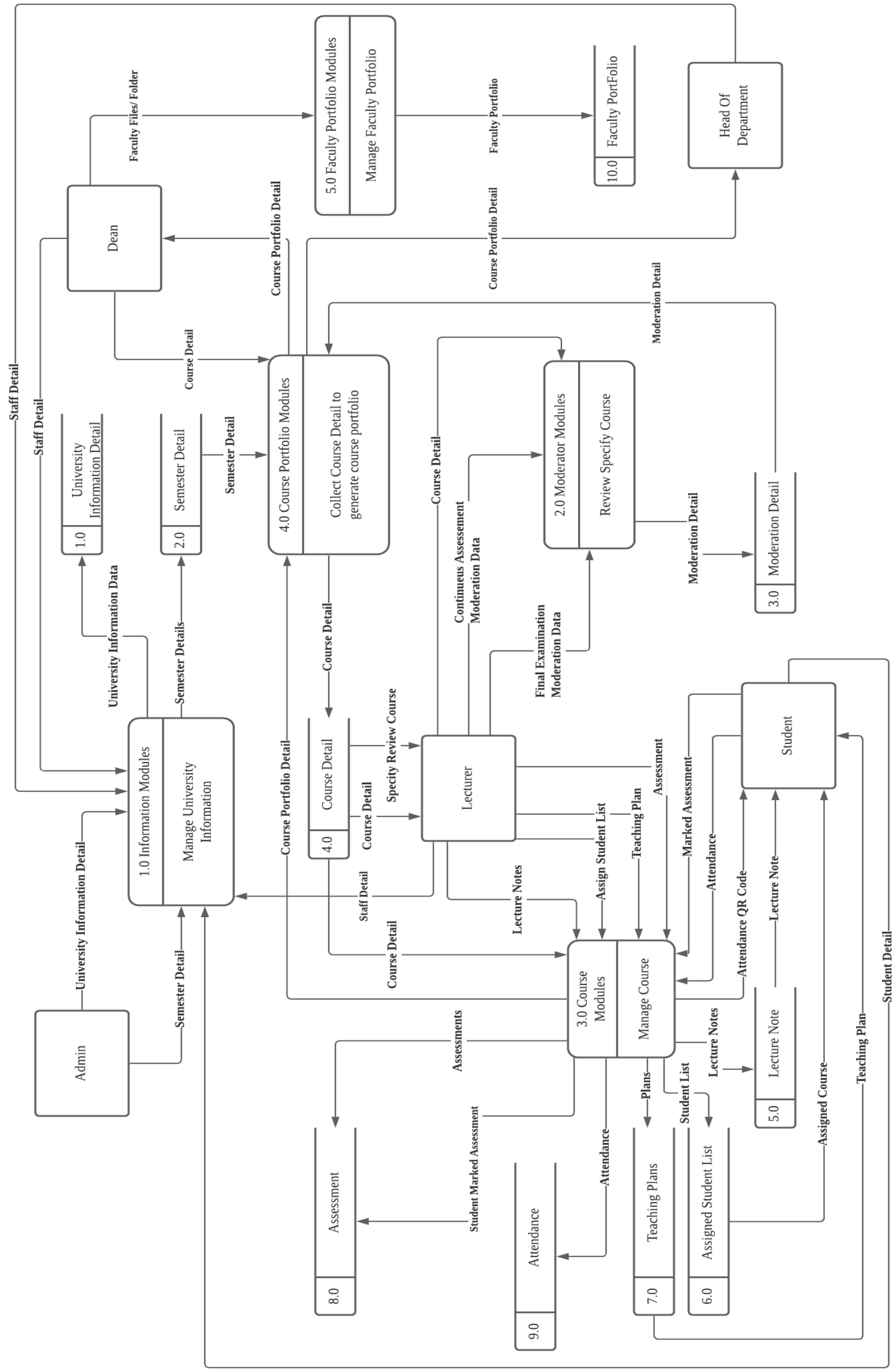
4.0 Design

4.1 DFD diagram

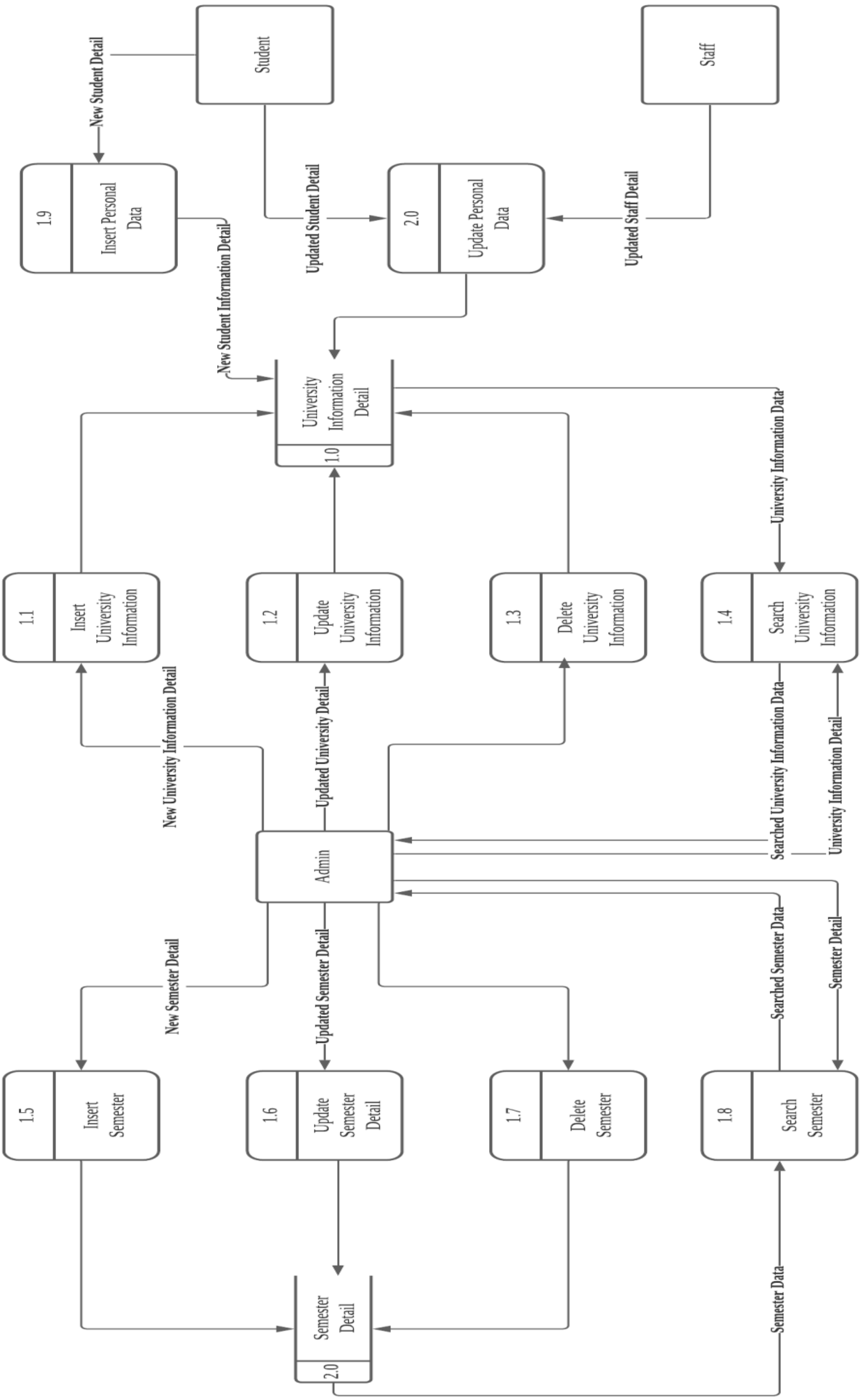
4.1.1 Context level



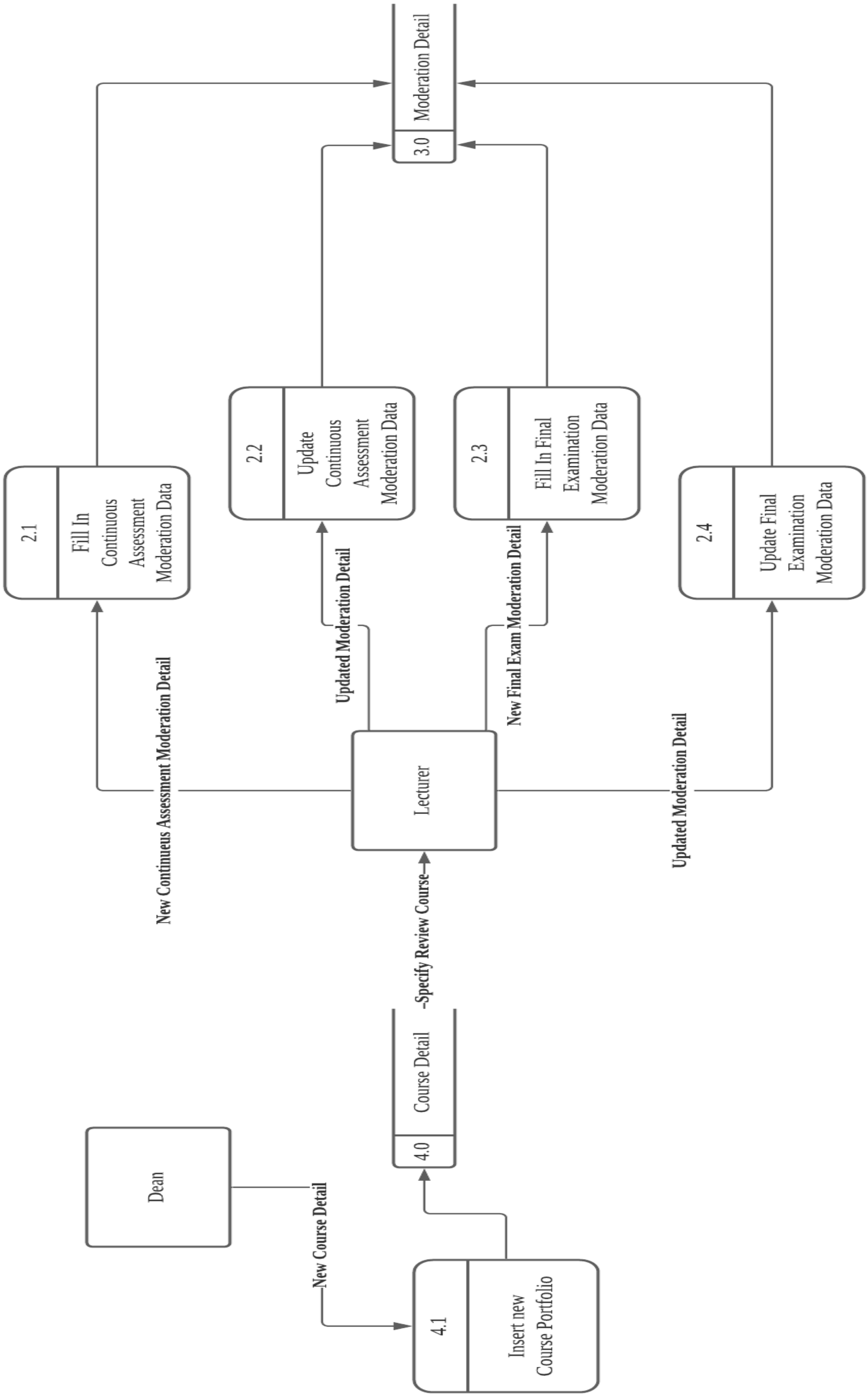
4.1.2 DFD Level 0



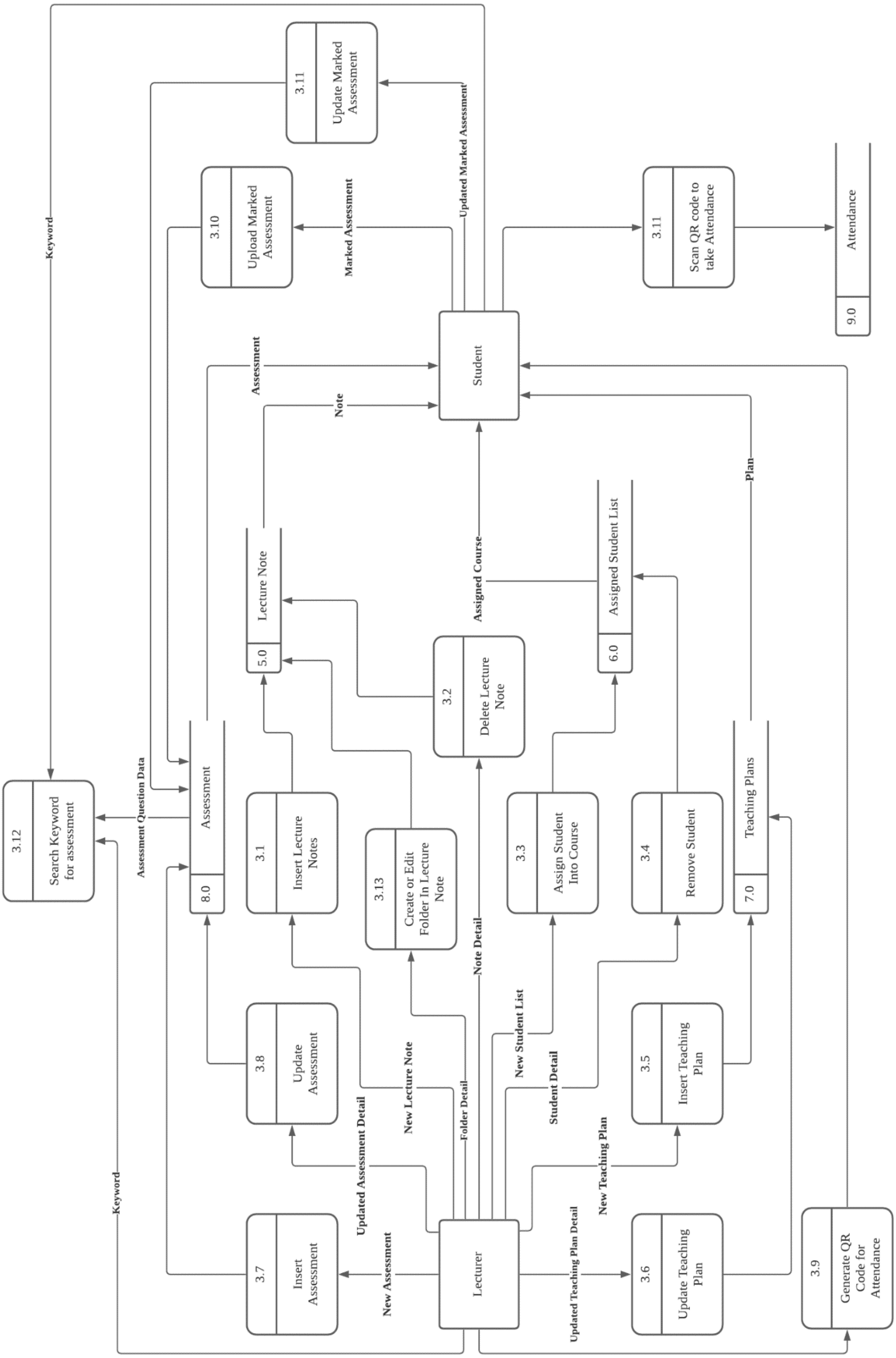
4.1.3 DFD Level 1
4.1.3.1 Information Modules



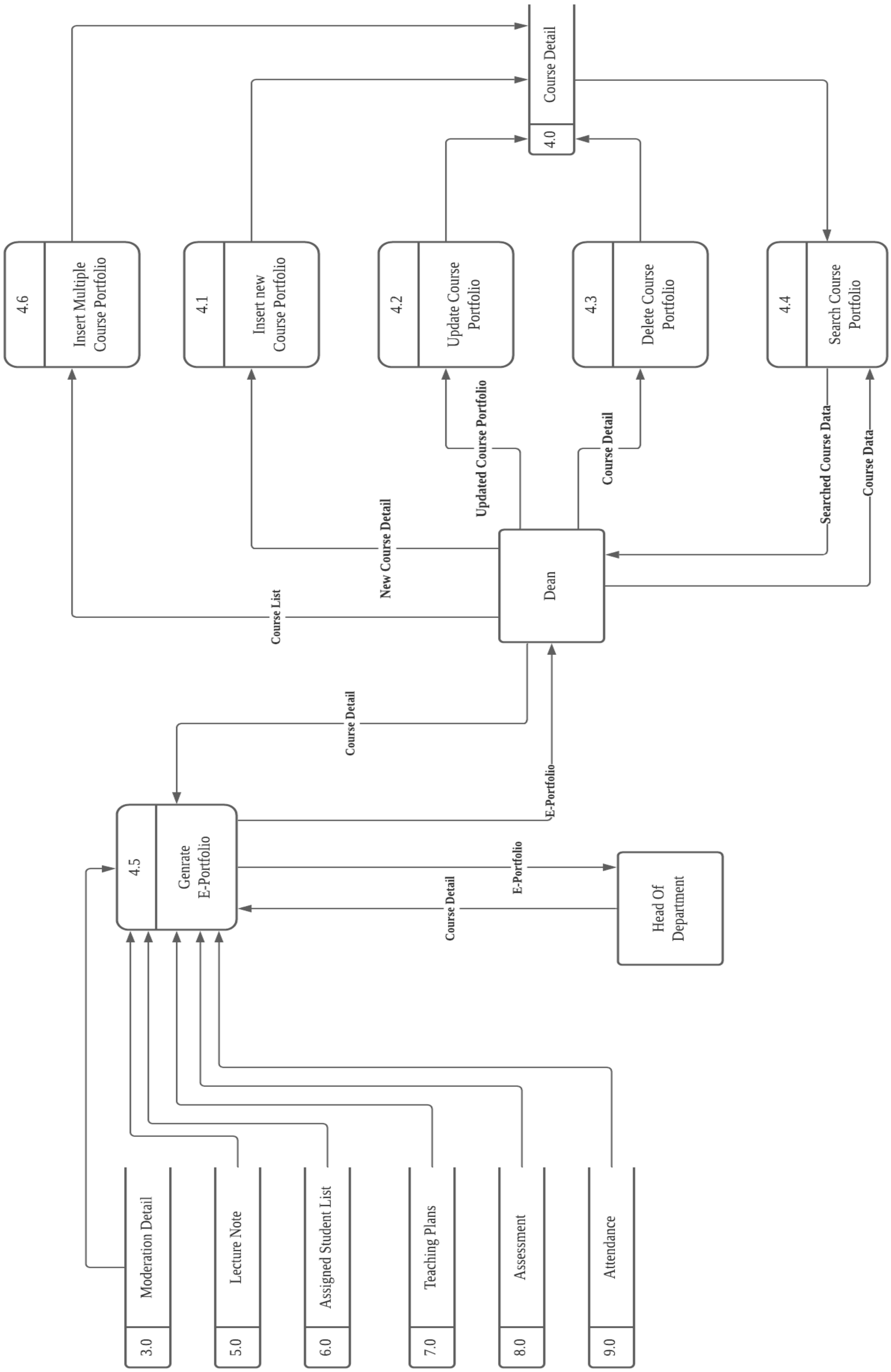
4.1.3.2 Moderator Modules



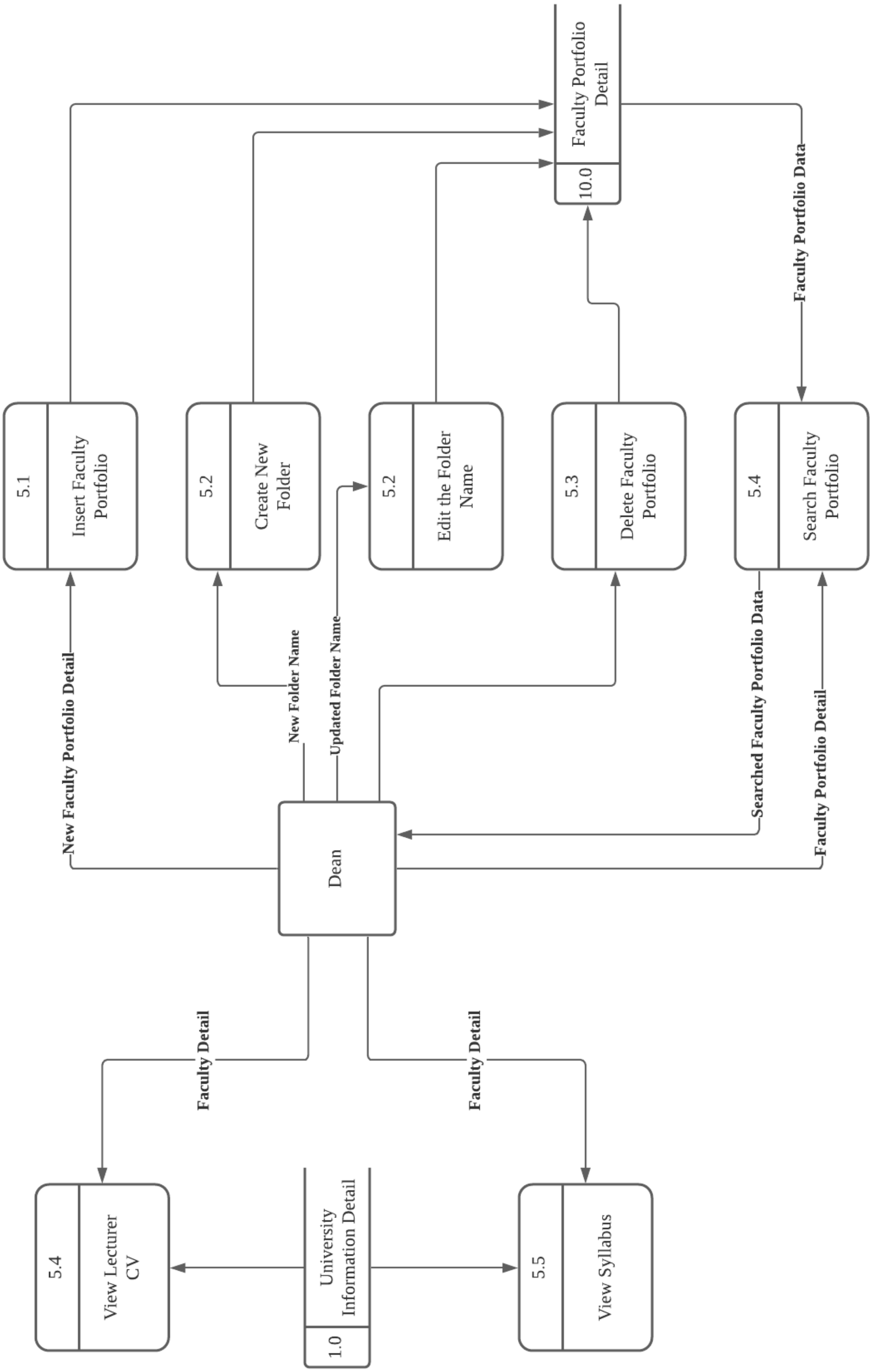
4.1.3.3 Course Modules



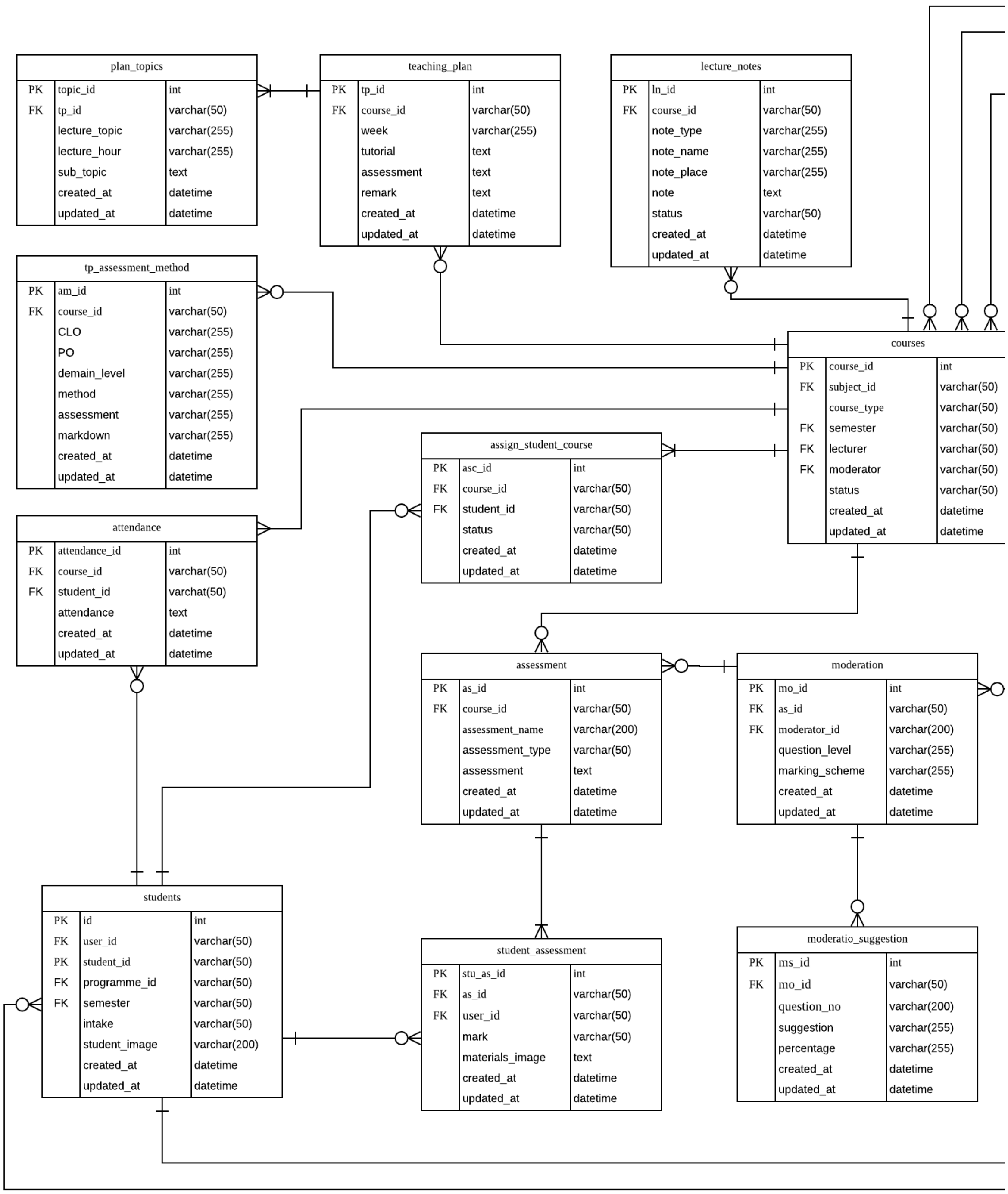
4.1.3.4 Course Portfolio Modules

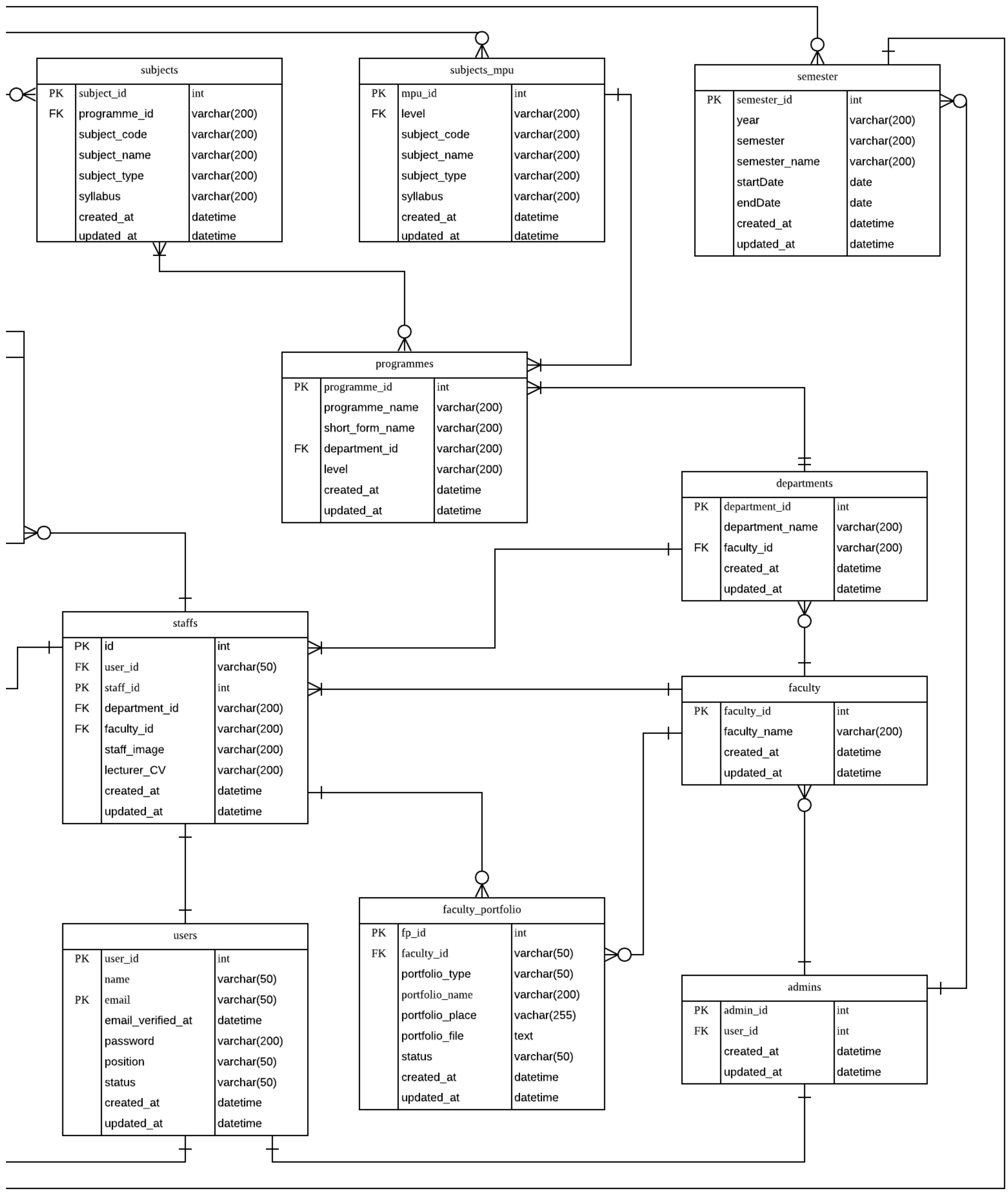


4.1.3.5 Faculty Portfolio Modules

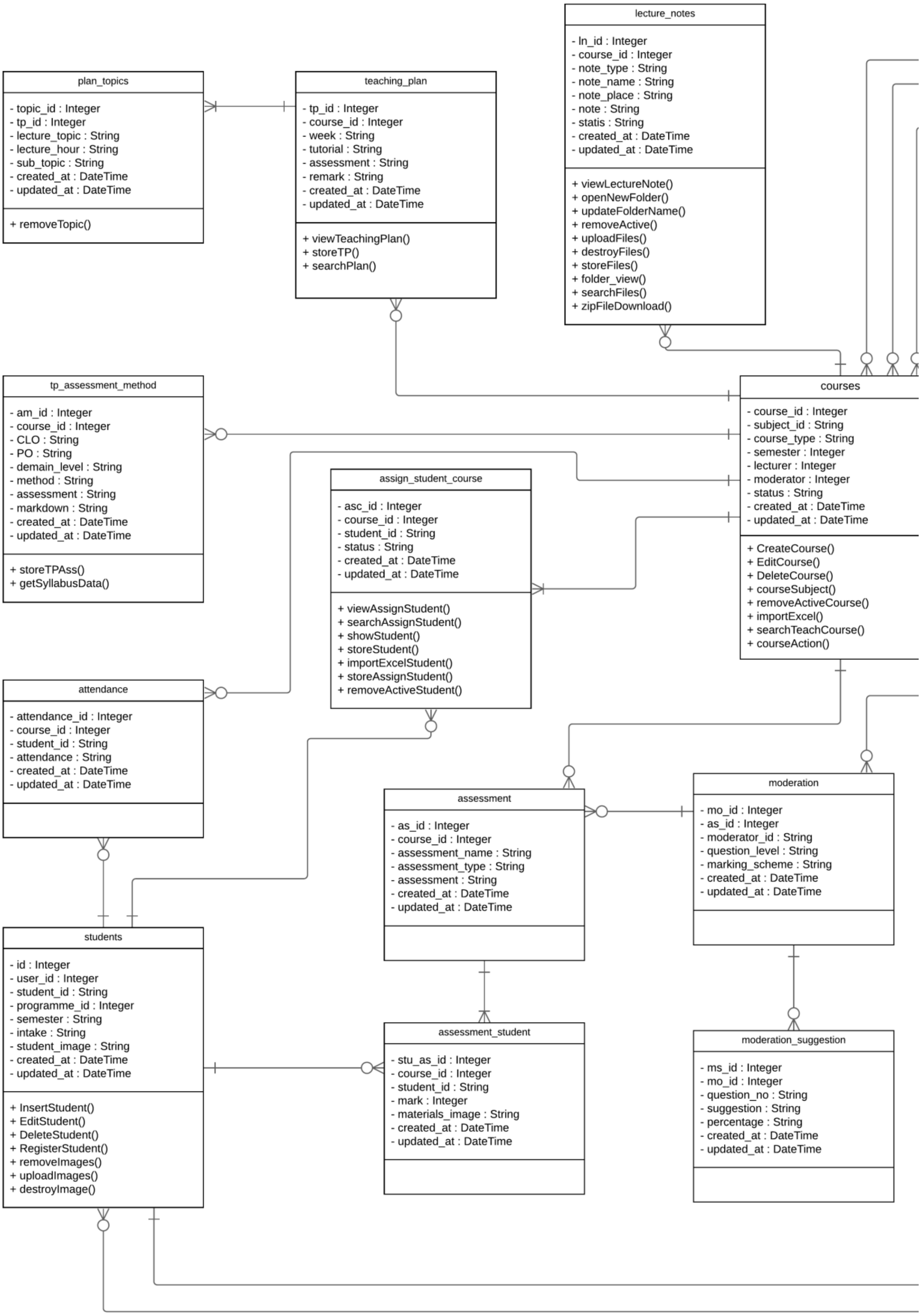


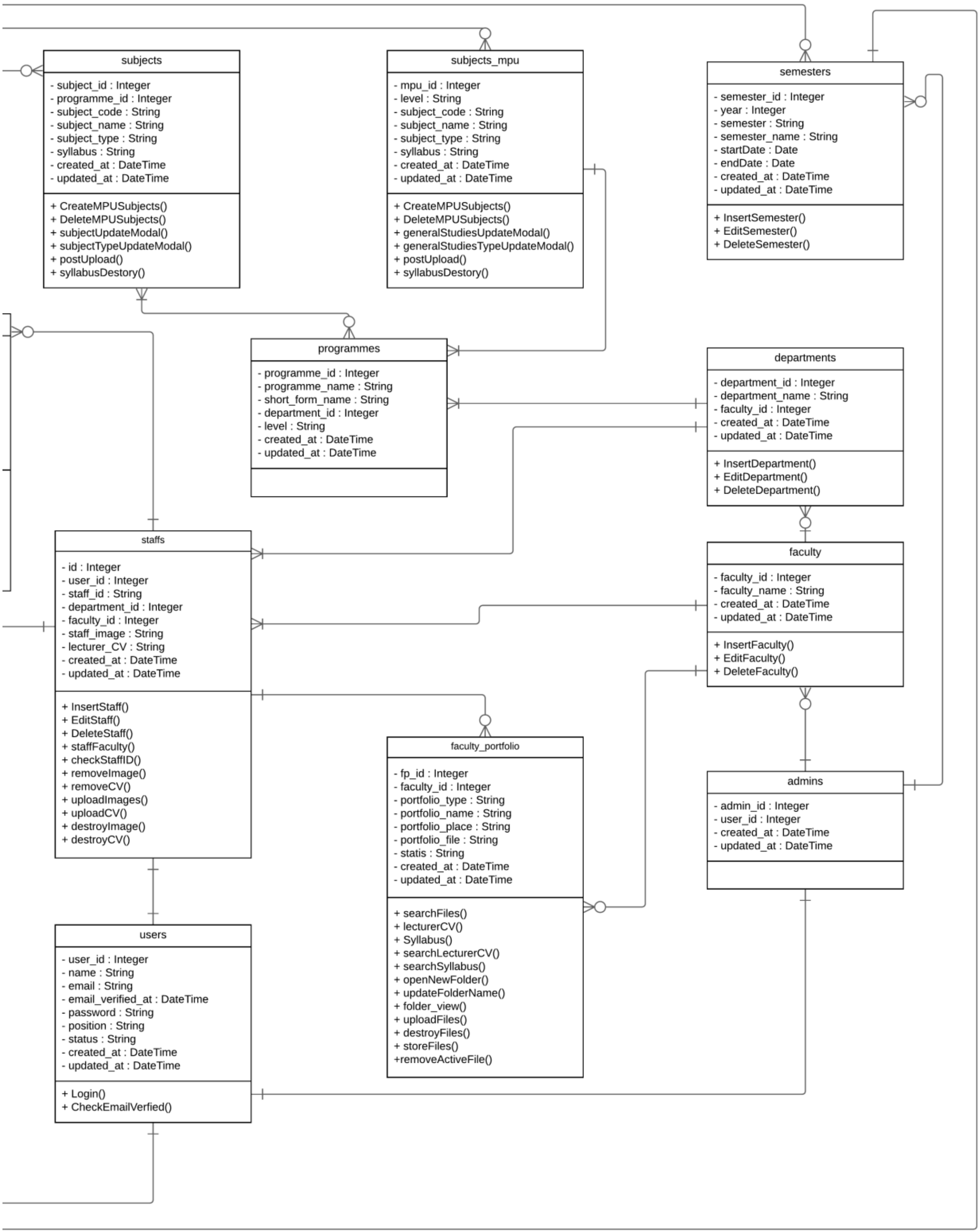
4.2 ER-diagram




subjects
PK subject_id int
FK programme_id varchar(200)
subject_code varchar(200)
subject_name varchar(200)
subject_type varchar(200)
syllabus varchar(200)
created_at datetime
updated_at datetime
subjects_mpu
PK mpu_id int
FK level varchar(200)
subject_code varchar(200)
subject_name varchar(200)
subject_type varchar(200)
syllabus varchar(200)
created_at datetime
updated_at datetime
semester
PK semester_id int
year varchar(200)
semester varchar(200)
semester_name varchar(200)
startDate date
endDate date
created_at datetime
updated_at datetime
programmes
PK programme_id int
programme_name varchar(200)
short_form_name varchar(200)
FK department_id varchar(200)
level varchar(200)
created_at datetime
updated_at datetime
departments
PK department_id int
department_name varchar(200)
FK faculty_id varchar(200)
created_at datetime
updated_at datetime
staffs
PK id int
FK user_id varchar(50)
PK staff_id int
FK department_id varchar(200)
FK faculty_id varchar(200)
staff_image varchar(200)
lecturer_CV varchar(200)
created_at datetime
updated_at datetime
faculty
PK faculty_id int
faculty_name varchar(200)
created_at datetime
updated_at datetime
users
PK user_id int
name varchar(50)
PK email varchar(50)
email_verified_at datetime
password varchar(200)
position varchar(50)
status varchar(50)
created_at datetime
updated_at datetime
faculty_portfolio
PK fp_id int
FK faculty_id varchar(50)
portfolio_type varchar(50)
portfolio_name varchar(200)
portfolio_place vachar(255)
portfolio_file text
status varchar(50)
created_at datetime
updated_at datetime
admins
PK admin_id int
FK user_id int
created_at datetime
updated_at datetime

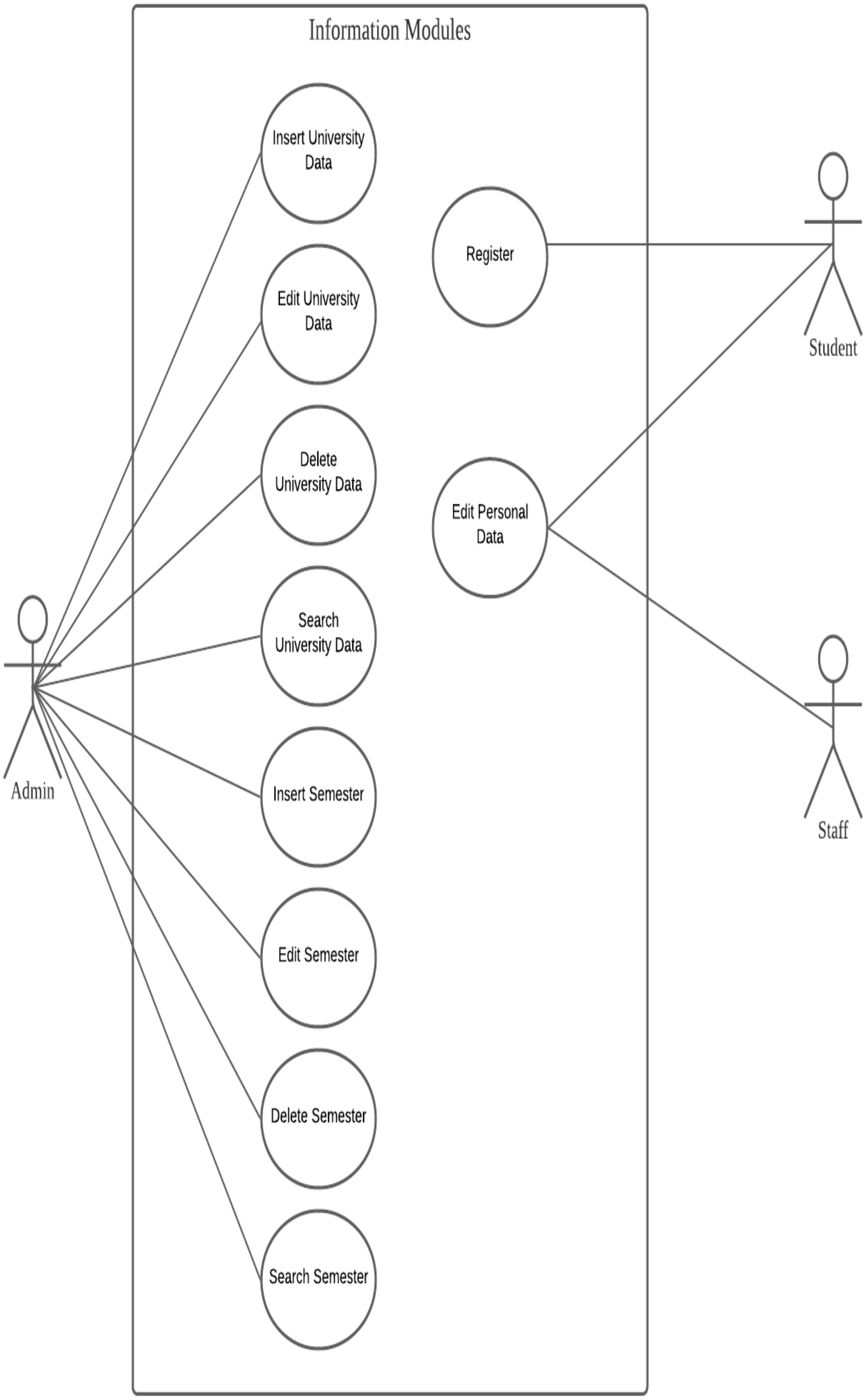
4.3 Class diagram



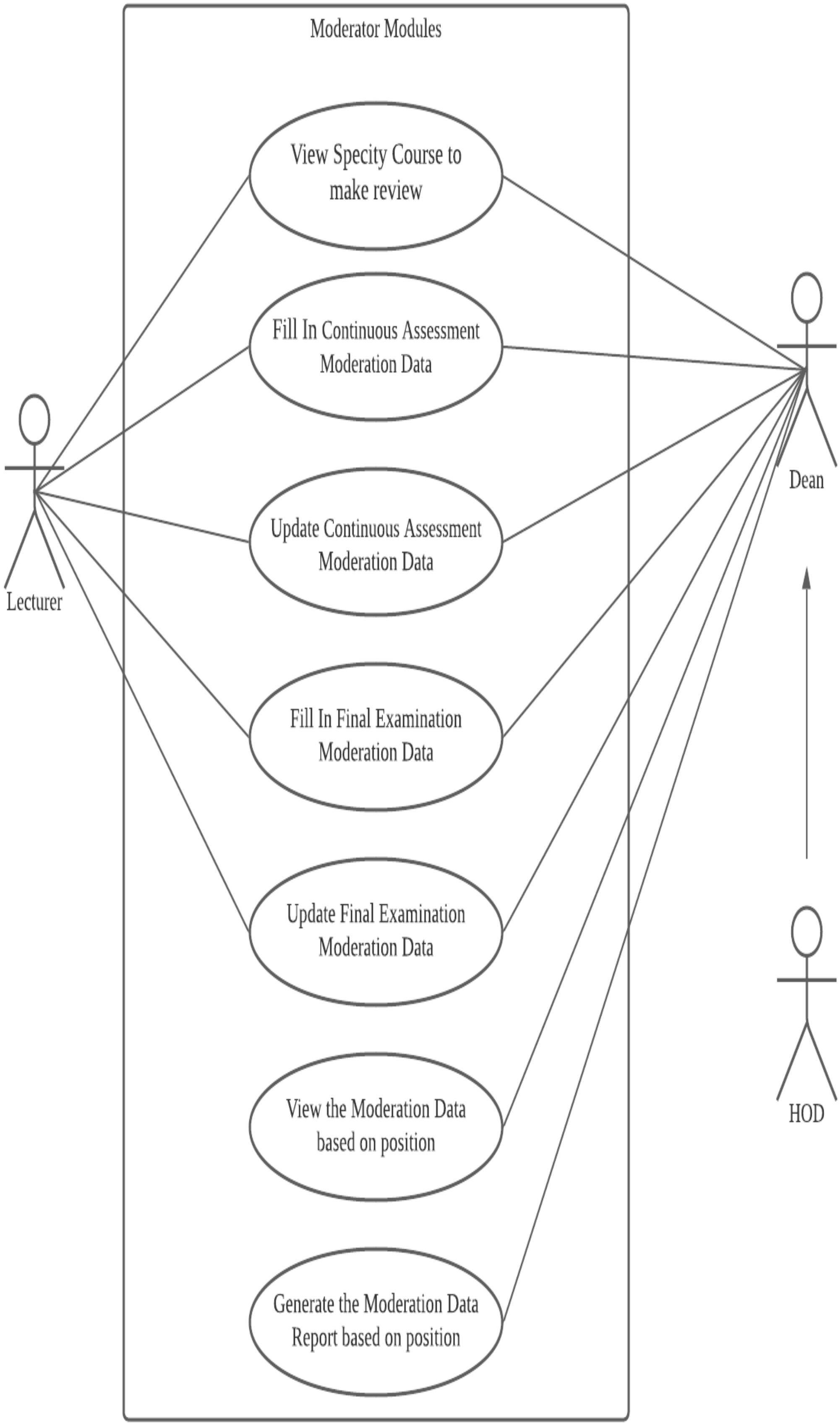

subjects
- subject_id : Integer
- programme_id : Integer
- subject_code : String
- subject_name : String
- subject_type : String
- syllabus : String
- created_at : DateTime
- updated_at : DateTime
+ CreateMPUSubjects()
+ DeleteMPUSubjects()
+ subjectUpdateModal()
+ subjectTypeUpdateModal()
+ postUpload()
+ syllabusDestory()
subjects_mpu
- mpu_id : Integer
- level : String
- subject_code : String
- subject_name : String
- subject_type : String
- syllabus : String
- created_at : DateTime
- updated_at : DateTime
+ CreateMPUSubjects()
+ DeleteMPUSubjects()
+ generalStudiesUpdateModal()
+ generalStudiesTypeUpdateModal()
+ postUpload()
+ syllabusDestory()
semesters
- semester_id : Integer
- year : Integer
- semester : String
- semester_name : String
- startDate : Date
- endDate : Date
- created_at : DateTime
- updated_at : DateTime
+ InsertSemester()
+ EditSemester()
+ DeleteSemester()
programmes
- programme_id : Integer
- programme_name : String
- short_form_name : String
- department_id : Integer
- level : String
- created_at : DateTime
- updated_at : DateTime
departments
- department_id : Integer
- department_name : String
- faculty_id : Integer
- created_at : DateTime
- updated_at : DateTime
+ InsertDepartment()
+ EditDepartment()
+ DeleteDepartment()
staffs
- id : Integer
- user_id : Integer
- staff_id : String
- department_id : Integer
- faculty_id : Integer
- staff_image : String
- lecturer_CV : String
- created_at : DateTime
- updated_at : DateTime
+ InsertStaff()
+ EditStaff()
+ DeleteStaff()
+ staffFaculty()
+ checkStaffID()
+ removeImage()
+ removeCV()
+ uploadImages()
+ uploadCV()
+ destroyImage()
+ destroyCV()
faculty
- faculty_id : Integer
- faculty_name : String
- created_at : DateTime
- updated_at : DateTime
+ InsertFaculty()
+ EditFaculty()
+ DeleteFaculty()
faculty_portfolio
- fp_id : Integer
- faculty_id : Integer
- portfolio_type : String
- portfolio_name : String
- portfolio_place : String
- portfolio_file : String
- statis : String
- created_at : DateTime
- updated_at : DateTime
+ searchFiles()
+ lecturerCV()
+ Syllabus()
+ searchLecturerCV()
+ searchSyllabus()
+ openNewFolder()
+ updateFolderName()
+ folder_view()
+ uploadFiles()
+ destroyFiles()
+ storeFiles()
+removeActiveFile()
admins
- admin_id : Integer
- user_id : Integer
- created_at : DateTime
- updated_at : DateTime
users
- user_id : Integer
- name : String
- email : String
- email_verified_at : DateTime
- password : String
- position : String
- status : String
- created_at : DateTime
- updated_at : DateTime
+ Login()
+ CheckEmailVerfied()

4.4 Use case diagram

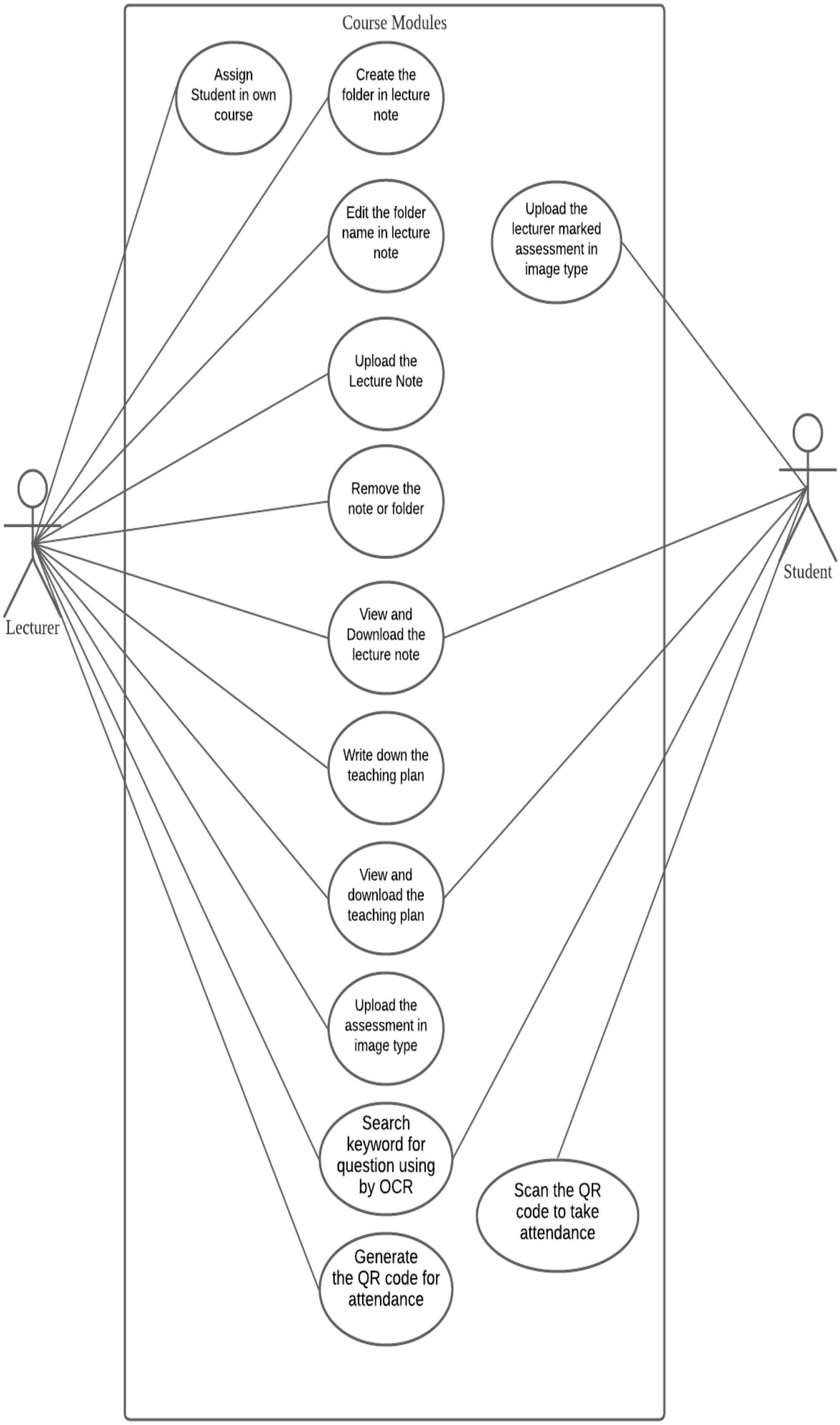
4.4.1 Information Modules



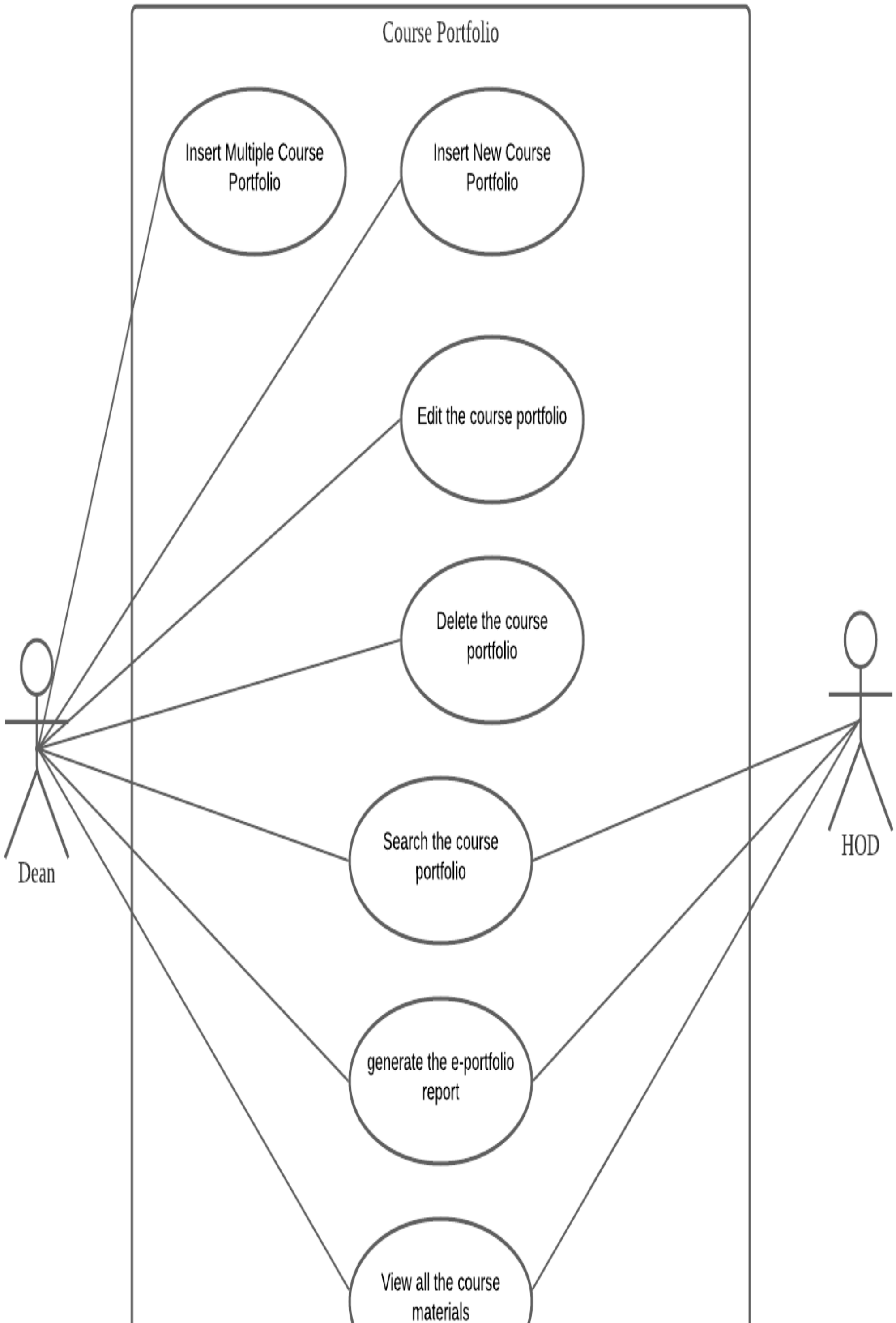
4.4.2 Moderator Modules



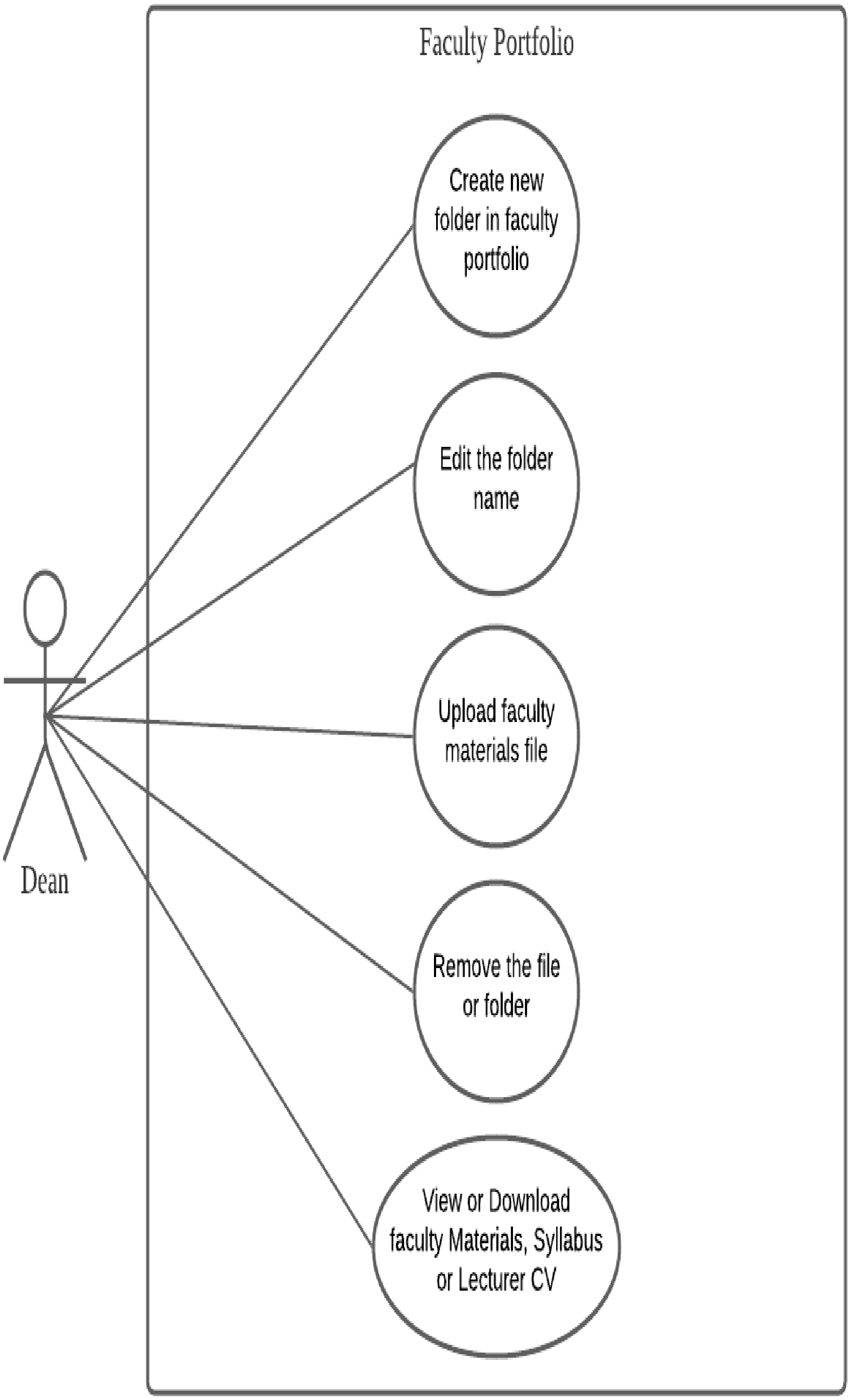
4.4.3 Course Modules



4.4.4 Course Portfolio Modules



4.4.5 Faculty Portfolio Modules



4.5 Use case description

4.5.1 Information Modules

Use case: Insert University Data (Information Modules)		
Overview: This case describes how the admin inserts university data.		
Pre – conditions: 1. The admin must login		
Post – conditions: 1. Successful condition • admin has inserted university data.		
Event		Response
1	Admin login to admin position portal.	• Perform admin account validation.
2	Admin selects which data want to insert.	• Enter to the selected data list webpage.
3	Select "…" and click the Add New Function	• Enter to the data form webpage.
4	Enter all the required information	• Perform information validation.
5	Admin has made the university data successfully	• University data is inserted successfully.

Use case: Edit University Data (Information Modules)		
Overview: This case describes how the admin edits the university data.		
Pre – conditions: 1. The admin must login		
Post – conditions: 1. Successful condition • Admin has updated the university data successfully.		
Event		Response
1	Admin login to admin position portal	• Perform admin account validation.
2	Admin selects which data want to edit.	• Enter to the selected data list webpage.
3	Select which data want to edit and click the "Edit" button	• Enter to the data form webpage.
4	Enter the updated information	• Perform information validation.
5	Admin has updated the university data successfully	• University Data is updated successfully.

Use case: Delete University Data (Information Modules)		
Overview: This case describes how the admin deletes University Data.		
Pre – conditions: 1. The admin must login		
Post – conditions: 1. Successful condition • admin has deleted university data.		
Event		Response
1	Admin login to admin position portal	• Perform receptionist account validation.
2	Admin selects which data want to delete.	• Enter to the selected data list webpage.
3	Select which data want to delete and click the "Delete" button	• Prompt out the confirm message to delete the university data.
4	Admin has deleted the university data successfully	• University data is deleted successfully.

Use case: Search University Data (Information Modules)		
Overview: This case describes how the admin searches University Data		
Pre – conditions: 1. The admin must login.		
Post – conditions: 1. Successful condition • Admin has searched the university data successfully.		
Event		Response
1	Admin logins to the system	• Perform admin account validation.
2	Admin selects which data want to view.	• Enter to the selected data list webpage.
3	Search the data information	• Show the searched data result.
4	Admin has searched the data successfully.	• Admin can continues to do action like edit or delete.

<table>
<tr><td colspan="3">Use case: Insert Semester (Information Modules)</td></tr>
<tr><td colspan="3">Overview: This case describes how the admin inserts new Semester.</td></tr>
<tr><td colspan="3">Pre – conditions:
1. The admin must login</td></tr>
<tr><td colspan="3">Post – conditions:
1. Successful condition
• Admin has inserted new semester.</td></tr>
<tr><td colspan="2">Event</td><td>Response</td></tr>
<tr><td>1</td><td>Admin login to admin position portal.</td><td>• Perform admin account validation.</td></tr>
<tr><td>2</td><td>Admin selects “Semester”.</td><td>• Enter to the semester list webpage.</td></tr>
<tr><td>3</td><td>Select “…” and click the Add New Semester Function</td><td>• Enter to the semester form webpage.</td></tr>
<tr><td>4</td><td>Enter all the required information</td><td>• Perform information validation.</td></tr>
<tr><td>5</td><td>Admin has made new semester successfully</td><td>• New Semester is inserted successfully.</td></tr>
</table>

<table>
<tr><td colspan="3">Use case: Edit Semester (Information Modules)</td></tr>
<tr><td colspan="3">Overview: This case describes how the admin edits the semester.</td></tr>
<tr><td colspan="3">Pre – conditions:
1. The admin must login</td></tr>
<tr><td colspan="3">Post – conditions:
1. Successful condition
• Admin has updated the semester successfully.</td></tr>
<tr><td colspan="2">Event</td><td>Response</td></tr>
<tr><td>1</td><td>Admin login to admin position portal</td><td>• Perform admin account validation.</td></tr>
<tr><td>2</td><td>Admin selects “ Semester”.</td><td>• Enter to the selected data list webpage.</td></tr>
<tr><td>3</td><td>Select which semester detail need to edit and click the “Edit” button</td><td>• Enter to the semester form webpage.</td></tr>
<tr><td>4</td><td>Enter the updated information</td><td>• Perform information validation.</td></tr>
<tr><td>5</td><td>Admin has updated the semester successfully</td><td>• Semester Data is updated successfully.</td></tr>
</table>

Use case: Delete Semester (Information Modules)		
Overview: This case describes how the admin deletes Semester.		
Pre – conditions: 1. The admin must login		
Post – conditions: 1. Successful condition • admin has deleted semester.		
Event		Response
1	Admin login to admin position portal	• Perform receptionist account validation.
2	Admin selects "Semester"	• Enter to the selected data list webpage.
3	Select which semester want to delete and click the "Delete" button	• Prompt out the confirm message to delete the semester.
4	Admin has deleted the semester successfully	• Semester is deleted successfully.

Use case: Search Semester (Information Modules)		
Overview: This case describes how the admin searches Semester		
Pre – conditions: 1. The admin must login.		
Post – conditions: 1. Successful condition • Admin has searched the semester successfully.		
Event		Response
1	Admin logins to the system	• Perform admin account validation.
2	Admin selects "Semester"	• Enter to the selected data list webpage.
3	Search the semester information	• Show the searched semester result.
4	Admin has searched the semester successfully.	• Admin can continues to do action like edit or delete.

<table>
<tr><td colspan="3">Use case: Register (Information Modules)</td></tr>
<tr><td colspan="3">Overview: This case describes how the student register new account by own. (The admin also can insert new student that function in university data.)</td></tr>
<tr><td colspan="3">Pre – conditions:
1. The student must study in Southern University College</td></tr>
<tr><td colspan="3">Post – conditions:
1. Successful condition
• Student register a new account successfully.</td></tr>
<tr><td colspan="2">Event</td><td>Response</td></tr>
<tr><td>1</td><td>Student selects “Register”</td><td>• Enter to the register form webpage.</td></tr>
<tr><td>2</td><td>Enter all the required information</td><td>• Perform information validation.</td></tr>
<tr><td>3</td><td>If the student ID is already exists</td><td>• Prompt out the error message “The Email has been existed”</td></tr>
<tr><td>4</td><td>Student has made new account successfully. (If want to login, need to go the SUC email to verify the account first.)</td><td>• New Student is inserted successfully.</td></tr>
</table>

<table>
<tr><td colspan="3">Use case: Edit Personal Data (Information Modules)</td></tr>
<tr><td colspan="3">Overview: This case describes how the student and staff can edit their own profile.</td></tr>
<tr><td colspan="3">Pre – conditions:
1. The student and staff must have an account in this system.
2. They must be login.</td></tr>
<tr><td colspan="3">Post – conditions:
1. Successful condition
• Student and staff edit own profile successfully.</td></tr>
<tr><td colspan="2">Event</td><td>Response</td></tr>
<tr><td>1</td><td>Student or staff selects the menu bar at the right-top side and click the “profile”</td><td>• Enter to the profile webpage.</td></tr>
<tr><td>2</td><td>Enter all the required information</td><td>• Perform information validation.</td></tr>
<tr><td>3</td><td>If got any mistake.</td><td>• Prompt out the error message “Something got wrong. Please try again”</td></tr>
<tr><td>4</td><td>Student or staff has edit their profile successfully.</td><td>• Student and staff of profile edit successfully.</td></tr>
</table>

4.5.2 Moderator Modules

Use case: View specify course to make review (Moderator Modules)		
Overview: This case describes how the lecturer views the specify course that need him to review it.		
Pre – conditions: 1. The lecturer must login		
Post – conditions: 1. Successful condition • Lecturer is able to view the specify course that assigned by Dean.		
Event		Response
1	Lecturer login to own position portal.	• Perform lecturer account validation.
2	Lecturer selects “Moderator”	• Enter to the Moderator list webpage.
3	Lecturer can searches which course is needed to review	• Show the searched course data results.
4	Click or selects the course wanted.	• Enter to the webpage that display the assessment list of the selected course.

Use case: Fill In Continuous Assessment Moderation Data (Moderator Modules)		
Overview: This case describes how the lecturer fill in the continuous assessment moderation data form.		
Pre – conditions: 1. The lecturer must login		
Post – conditions: 1. Successful condition • Lecturer has inserted new moderation data.		
Event		Response
1	Lecturer login to own position portal.	• Perform lecturer account validation.
2	Lecturer selects “Moderator”	• Enter to the Moderator list webpage.
3	Lecturer can searches which course is needed to review.	• Show the searched course data results.
4	Click or selects the course wanted.	• Enter to the webpage that display the assessment list of the selected course.
5	Select which assessment wanted to review.	• Enter to the moderation form webpage.
6	Enter all the required information	• Perform information validation.
7	Lecturer has made the moderation form successfully	• Moderation form is inserted successfully.

Use case: Update Continuous Assessment Moderation Data (Moderator Modules)		
Overview: This case describes how the lecturer update continuous assessment moderation data form.		
Pre – conditions: 1. The lecturer must login		
Post – conditions: 1. Successful condition • Lecturer has updated moderation data successfully.		
Event		Response
1	Lecturer login to own position portal.	• Perform lecturer account validation.
2	Lecturer selects “Moderator”	• Enter to the Moderator list webpage.
3	Lecturer can searches which course is needed to review.	• Show the searched course data results.
4	Click or selects the course wanted.	• Enter to the webpage that display the assessment list of the selected course.
5	Select which assessment moderation data wanted to edit.	• Enter to the moderation form webpage.
6	Enter all the required information	• Perform information validation.
7	Lecture has updated the moderation data successfully	• Moderation Data is updated successfully.

Use case: Fill In Final Examination Moderation Data (Moderator Modules)		
Overview: This case describes how the lecturer fill in the final examination moderation data form.		
Pre – conditions: 1. The lecturer must login.		
Post – conditions: 1. Successful condition • Lecturer has inserted new moderation data.		
Event		Response
1	Lecturer login to own position portal.	• Perform lecturer account validation.
2	Lecturer selects "Moderator"	• Enter to the Moderator list webpage.
3	Lecturer can searches which course is needed to review.	• Show the searched course data results.
4	Click or selects the course wanted.	• Enter to the webpage that display the assessment list of the selected course.
5	Select "Final Examination".	• Enter to the final examination moderation form webpage.
6	Enter all the required information	• Perform information validation.
7	Lecturer has made the final examination moderation form successfully	• Final Examination Moderation form is inserted successfully.

Use case: Update Final Examination Moderation Data (Moderator Modules)		
Overview: This case describes how the lecturer update final examination moderation data form.		
Pre – conditions: 1. The lecturer must login		
Post – conditions: 1. Successful condition • Lecturer has updated moderation data successfully.		
Event		Response
1	Lecturer login to own position portal.	• Perform lecturer account validation.
2	Lecturer selects "Moderator"	• Enter to the Moderator list webpage.
3	Lecturer can searches which course is needed to review.	• Show the searched course data results.
4	Click or selects the course wanted.	• Enter to the webpage that display the assessment list of the selected course.
5	Select "Final Examination"	• Enter to the final examination moderation form webpage.
6	Enter all the required information	• Perform information validation.
7	Lecture has updated the final examination moderation data successfully	• Final Examination Moderation Data is updated successfully.

Use case: View the Moderation Data based on position (Moderator Modules)		
Overview: This case describes how the Dean or HOD views the moderator review course result.		
Pre – conditions: 1. The Dean or HOD must login		
Post – conditions: 1. Successful condition • Dean or HOD is able to view the moderator of review result in each course.		
Event		Response
1	Dean or HOD login to own position portal.	• Perform lecturer account validation.
2	Dean or HOD selects "Moderator"	• Enter to the Moderator list webpage, but after run in that will show their needed review course. (Dean and HOD also need to review some course.)
3	Dean or HOD can searches which course is needed to view	• Show the searched course data results.
4	Click or selects the course wanted.	• Enter to the webpage that display the assessment list of the selected course.
5	Dean or HOD has searched the course of moderator successfully.	• Dean or HOD can continues to do action like view or download the result.

<table>
<tr><td colspan="3">Use case: Generate the Moderation Data Report based on position (Moderator Modules)</td></tr>
<tr><td colspan="3">Overview: This case describes how the Dean or HOD generation the moderation report.</td></tr>
<tr><td colspan="3">Pre – conditions:
1. The lecturer must login</td></tr>
<tr><td colspan="3">Post – conditions:
1. Successful condition
• Dean or HOD is able to view the moderator of review result in each course and then to generate a report of moderation data.</td></tr>
<tr><td colspan="2">Event</td><td>Response</td></tr>
<tr><td>1</td><td>Dean or HOD login to own position portal.</td><td>• Perform lecturer account validation.</td></tr>
<tr><td>2</td><td>Dean or HOD selects “Moderator”</td><td>• Enter to the Moderator list webpage, but after run in that will show their needed review course.</td></tr>
<tr><td>3</td><td>Dean or HOD can searches which course is needed to view</td><td>• Show the searched course data results.</td></tr>
<tr><td>4</td><td>Click or selects the course wanted.</td><td>• Enter to the webpage that display the assessment list of the selected course.</td></tr>
<tr><td>5</td><td>Click the “Download” button on needed assessment.</td><td>• That will get the result of moderation data and generate it in pdf file report.</td></tr>
<tr><td>6</td><td>Dean or HOD has generated the report successfully.</td><td>• Moderation form report is created successfully.</td></tr>
</table>

4.5.3 Course Modules

Use case: Assign Student in own course (Course Modules)		
Overview: This case describes how the lecturer assign student in own course.		
Pre – conditions: 1. The lecturer must login.		
Post – conditions: 1. Successful condition • Lecturer has assigned new student in course successfully.		
Event		Response
1	Lecturer login to own position portal.	• Perform lecturer account validation.
2	Lecturer selects "Course"	• Enter to the Course list webpage.
3	Lecturer can searches which course and select it.	• Show the searched course data results and enter into course action page.
4	Click the "Assign Student" button.	• Enter to the webpage that display the assigned student list of the selected course.
5	Select "…" and click the Upload a File.	• Prompt a model that can be upload a excel file that record the student list.
6	Or lecturer can click the link "Template" to download the sample of the excel file.	• Download a file that is sample of record the student list.
7	Lecturer can drop or select the file and then click the "Save Changes" button.	• If the data is available then it will inserted successfully. • If the data got error then it will prompt the error student id and inserted the correct student list.
8	Lecturer has assigned the student list successfully	• Students is assigned in course successfully.

Use case: Create the folder in lecture note (Course Modules)		
Overview: This case describes how the lecturer create a new folder in lecture note.		
Pre – conditions: 1. The lecturer must login.		
Post – conditions: 1. Successful condition • Lecturer has created a new folder successfully.		
Event		Response
1	Lecturer login to own position portal.	• Perform lecturer account validation.
2	Lecturer selects "Course"	• Enter to the Course list webpage.
3	Lecturer can searches which course and select it.	• Show the searched course data results and enter into course action page.
4	Click the "Lecture Note" button.	• Enter to the webpage that display the lecture note of the selected course.
5	Select "…" and click the Make a new folder.	• Prompt a model that is a form to insert the folder name.
6	Select the inserted folder can link into the folder.	• Enter the inserted folder and then can upload the files in this folder.
7	Lecturer has inserted a new folder successfully	• New folder is inserted successfully.

Use case: Edit the folder name in lecture note (Course Modules)		
Overview: This case describes how the lecturer edit the folder name in lecture note.		
Pre – conditions: e lecturer must login.		
Post – conditions: 1. Successful condition • Lecturer has updated folder name successfully.		
Event		Response
1	Lecturer login to own position portal.	• Perform lecturer account validation.
2	Lecturer selects "Course"	• Enter to the Course list webpage.
3	Lecturer can searches which course and select it.	• Show the searched course data results and enter into course action page.
4	Click the "Lecture Note" button.	• Enter to the webpage that display the lecture note of the selected course.
5	Select the folder and click the "wrench" button.	• Enter the inserted folder and then can upload the files in this folder.
6	Enter new folder name and save it.	• The folder name is updated successfully.

<table>
<tr><td colspan="3">Use case: Upload the Lecture Note (Course Modules)</td></tr>
<tr><td colspan="3">Overview: This case describes how the lecturer how to upload the lecture note.</td></tr>
<tr><td colspan="3">Pre – conditions:
1. The lecturer must login.</td></tr>
<tr><td colspan="3">Post – conditions:
1. Successful condition
• Lecturer has upload the file successfully.</td></tr>
<tr><td colspan="2">Event</td><td>Response</td></tr>
<tr><td>1</td><td>Lecturer login to own position portal.</td><td>• Perform lecturer account validation.</td></tr>
<tr><td>2</td><td>Lecturer selects “Course”</td><td>• Enter to the Course list webpage.</td></tr>
<tr><td>3</td><td>Lecturer can searches which course and select it.</td><td>• Show the searched course data results and enter into course action page.</td></tr>
<tr><td>4</td><td>Click the “Lecture Note” button.</td><td>• Enter to the webpage that display the lecture note of the selected course.</td></tr>
<tr><td>5</td><td>Select the folder and then click the “…” and select the Upload Files.</td><td>• Prompt a model that can be upload files and save it in that folder.</td></tr>
<tr><td>6</td><td>Lecturer can drop or select the file and rename it.</td><td>• The file name will be follow it.</td></tr>
<tr><td>7</td><td>Lecturer has inserted the lecture note successfully</td><td>• Lecture note is inserted in course successfully.</td></tr>
</table>

<table>
<tr><td colspan="3">Use case: Remove the note or folder (Course Modules)</td></tr>
<tr><td colspan="3">Overview: This case describes how the lecturer deletes the note or folder.</td></tr>
<tr><td colspan="3">Pre – conditions:
1. The lecturer must login</td></tr>
<tr><td colspan="3">Post – conditions:
1. Successful condition
• Lecturer has deletes the note or folder.</td></tr>
<tr><td colspan="2">Event</td><td>Response</td></tr>
<tr><td>1</td><td>Lecturer login to own position portal.</td><td>• Perform lecturer account validation.</td></tr>
<tr><td>2</td><td>Lecturer selects “Course”</td><td>• Enter to the Course list webpage.</td></tr>
<tr><td>3</td><td>Lecturer can searches which course and select it.</td><td>• Show the searched course data results and enter into course action page.</td></tr>
<tr><td>4</td><td>Click the “Lecture Note” button.</td><td>• Enter to the webpage that display the lecture note of the selected course.</td></tr>
<tr><td>5</td><td>Select which file or folder wanted to delete and click “X” button.</td><td>• Prompt out the confirm message to delete it.</td></tr>
<tr><td>6</td><td>Lecturer has deleted the note successfully</td><td>• Note is deleted successfully.</td></tr>
</table>

Use case: View and Download the lecture note (Course Modules)		
Overview: This case describes how the student view and download the lecture note.		
Pre – conditions: 1. The lecturer must login.		
Post – conditions: 1. Successful condition • Student is able to download the lecture note successfully.		
Event		Response
1	Student login to own position portal.	• Perform student account validation.
2	Student selects "Course"	• Enter to the Course list webpage.
3	Student can searches which course and select it.	• Show the searched course data results and enter into course action page.
4	Click the "Lecture Note" button.	• Enter to the webpage that display the lecture note of the selected course.
5	Select "…" and click the Download all Notes.	• Generate a zip files and put all the note and folder in it. (If already link in any folder then will be download the files only.)
6	Just Click the note	• It will also can download it.
7	Student has download the lecture note successfully	• Lecture note is downloaded successfully.

Use case: Write down the teaching plan (Course Modules)		
Overview: This case describes how the lecturer write down the teaching plan.		
Pre – conditions: 1. The lecturer must login.		
Post – conditions: 1. Successful condition • Lecturer has write down the teaching plan successfully.		
Event		Response
1	Lecturer login to own position portal.	• Perform lecturer account validation.
2	Lecturer selects “Course”	• Enter to the Course list webpage.
3	Lecturer can searches which course and select it.	• Show the searched course data results and enter into course action page.
4	Click the “Teaching Plan” button.	• Enter to the webpage that display the teaching plan of the selected course.
5	Select “…” and click the Manage Assessment Method.	• Enter to the Assessment Method form webpage and the assessment method data that is from subject syllabus.
6	Lecturer can fill in the teaching and assessment method.	• Perform information validation.
7	Or Select “…” and click the Manage Weekly Plan.	• Enter to the Weekly Plan Form webpage and the lecture topic that also is from subject syllabus.
8	Student has download the teaching plan successfully	• The Teaching plan is downloaded successfully.

<table>
<tr><td colspan="3">Use case: View and download the teaching plan (Course Modules)</td></tr>
<tr><td colspan="3">Overview: This case describes how the student view and download the teaching plan.</td></tr>
<tr><td colspan="3">Pre – conditions:
1. The student must login.</td></tr>
<tr><td colspan="3">Post – conditions:
1. Successful condition
• Student is able to view the teaching plan and download it about the course.</td></tr>
<tr><td colspan="2">Event</td><td>Response</td></tr>
<tr><td>1</td><td>Student login to own position portal.</td><td>• Perform student account validation.</td></tr>
<tr><td>2</td><td>Student selects “Course”</td><td>• Enter to the Course list webpage.</td></tr>
<tr><td>3</td><td>Student can searches which course and select it.</td><td>• Show the searched course data results and enter into course action page.</td></tr>
<tr><td>4</td><td>Click the “Teaching Plan” button.</td><td>• Enter to the webpage that display the teaching plan of the selected course.</td></tr>
<tr><td>5</td><td>Student can view the teaching plan successfully.</td><td>• Enter to the webpage that display the result of assessment method and weekly plan. (If Lecturer haven’t uploaded teaching plan, student view it is null.)</td></tr>
<tr><td>6</td><td>Student can select “…” and click the Download button.</td><td>• Generate a teaching plan report that record the assessment method and weekly plan about the course.</td></tr>
<tr><td>7</td><td>Or Select “…” and click the Manage Weekly Plan.</td><td>• Enter to the Weekly Plan Form webpage and the lecture topic that also is from subject syllabus.</td></tr>
<tr><td>8</td><td>Lecturer can fill in all week of teaching planning</td><td>• Perform information validation.</td></tr>
<tr><td>9</td><td>Lecturer has inserted the teaching plan successfully.</td><td>• The teaching plan is inserted successfully</td></tr>
</table>

Use case: Upload the assessment in image type (Course Modules)		
Overview: This case describes how the lecturer upload the assessment in image type.		
Pre – conditions: 1. The lecturer must login.		
Post – conditions: 1. Successful condition • Lecturer has upload the assessment successfully.		
Event		Response
1	Lecturer login to own position portal.	• Perform lecturer account validation.
2	Lecturer selects “Course”	• Enter to the Course list webpage.
3	Lecturer can searches which course and select it.	• Show the searched course data results and enter into course action page.
4	Click the “Assessment” button.	• Enter to the webpage that display the assessment list of the selected course.
5	Select which assessment is needed upload or edit.	• Enter to the Assessment form webpage.
6	Lecturer can drop or select the question file in image type and upload it.	• Perform information validation.
7	Lecturer has upload the assessment successfully	• The Assessment is inserted successfully.

<table>
<tr><td colspan="3">Use case: Upload the lecturer marked assessment in image type (Course Modules)</td></tr>
<tr><td colspan="3">Overview: This case describes how the student upload the lecturer marked assessment in image type.</td></tr>
<tr><td colspan="3">Pre – conditions:
1. The student must login.</td></tr>
<tr><td colspan="3">Post – conditions:
1. Successful condition
• student has upload the assessment successfully.</td></tr>
<tr><td colspan="2">Event</td><td>Response</td></tr>
<tr><td>1</td><td>Student login to own position portal.</td><td>• Perform student account validation.</td></tr>
<tr><td>2</td><td>Student selects “Course”</td><td>• Enter to the Course list webpage.</td></tr>
<tr><td>3</td><td>Student can searches which course and select it.</td><td>• Show the searched course data results and enter into course action page.</td></tr>
<tr><td>4</td><td>Click the “Assessment” button.</td><td>• Enter to the webpage that display the assessment list of the selected course.</td></tr>
<tr><td>5</td><td>Select which assessment is needed upload or edit.</td><td>• Enter to the Assessment form webpage.</td></tr>
<tr><td>6</td><td>Student can drop or select the marked assessment file in image type and upload it.</td><td>• Perform information validation.</td></tr>
<tr><td>7</td><td>Student has upload the assessment successfully</td><td>• The Assessment is inserted successfully.</td></tr>
</table>

Use case: Search keyword for question using by OCR (Course Modules)		
Overview: This case describes how the lecturer search keyword for question using by OCR.		
Pre – conditions: 1. The Lecturer or Student must login.		
Post – conditions: 1. Successful condition • Lecturer or student has searched the result for question successfully.		
Event		Response
1	Lecturer or student login to own position portal.	• Perform lecturer or student account validation.
2	Lecturer or student selects “Course”	• Enter to the Course list webpage.
3	Lecturer or student can searches which course and select it.	• Show the searched course data results and enter into course action page.
4	Click the “Assessment” button.	• Enter to the webpage that display the assessment list of the selected course.
5	Fill the keyword about the question in search input form.	• Perform OCR to search which image in merge the keyword and display it.
6	Lecturer or student can view the question has in which semester and year.	• The keyword searched for question successfully.

Use case: Generate the QR code for attendance (Course Modules)		
Overview: This case describes how the lecturer generate the QR code for attendance.		
Pre – conditions: 1. The Lecturer must login.		
Post – conditions: 1. Successful condition • Lecturer or student has searched the result for question successfully.		
Event		Response
1	Lecturer or student login to own position portal.	• Perform lecturer or student account validation.
2	Lecturer or student selects “Course”	• Enter to the Course list webpage.
3	Lecturer or student can searches which course and select the right hand side of the small QR code.	• Generate the QR code about that one times of attendance.
4	Or Select the course and click the “Attendance”	• Enter to the attendance list webpage.
5	Lecturer has view the attendance list successfully.	• The QR code for attendance is created successfully.

Use case: Scan the QR code to take attendance (Course Modules)		
Overview: This case describes how the student scan the QR code to take attendance.		
Pre – conditions: -		
Post – conditions: 1. Successful condition • Student is able to take attendance successfully.		
Event		Response
1	Student can use any scan QR app or tools to scan it.	• Enter to the student detail form webpage.
2	Student can fill in own student detail and submit it.	• Perform the information verification.
3	The student take attendance successfully.	• The attendance is inserted successfully.

4.5.4 Course Portfolio Modules

Use case: Insert Multiple Course Portfolio (Course Portfolio Modules)		
Overview: This case describes how the dean insert multiple course portfolio.		
Pre – conditions: 1. The dean must login.		
Post – conditions: 1. Successful condition • Dean has inserted multiple course portfolio successfully.		
Event		Response
1	Dean login to own position portal.	• Perform dean account validation.
2	Dean selects “Course Portfolio”	• Enter to the Course Portfolio list webpage.
3	Select “…” and click the Add Multiple Courses.	• Prompt a model that can be upload a excel file that record new course list.
4	Or Dean can click the link “Template” to download the sample of the excel file.	• Download a file that is sample of record the course list.
5	Dean can drop or select the file and then click the “Save Changes” button.	• If the data is available then it will inserted successfully. • If the data got error then it will prompt the error student id and inserted the correct course list.
6	Dean has insert the course portfolio successfully	• Course Portfolio is inserted successfully. • That also will has the new course that is assigned to each lecturer.

Use case: Insert new course portfolio (Course Portfolio Modules)

Overview: This case describes how the dean inserts new course portfolio.		
Pre – conditions: 1. The dean must login		
Post – conditions: 1. Successful condition • Dean has inserted the course portfolio successfully.		
Event		Response
1	Dean login to own position portal.	• Perform dean account validation.
2	Dean selects “Course Portfolio”	• Enter to the Course Portfolio list webpage.
3	Select “…” and click the Add New Course.	• Enter to the course form webpage.
4	Enter the required information	• Perform information validation.
5	Dean has insert the new course portfolio data successfully	• Course portfolio is inserted successfully.

Use case: Edit the course portfolio (Course Portfolio Modules)		
Overview: This case describes how the dean edits the course portfolio.		
Pre – conditions: 1. The dean must login		
Post – conditions: 1. Successful condition • Dean has updated the course portfolio detail successfully.		
Event		Response
1	Dean login to own position portal.	• Perform dean account validation.
2	Dean selects “Course Portfolio”	• Enter to the Course Portfolio list webpage.
3	Select which course want to edit and click the “Wrench” button	• Enter to the course portfolio form webpage.
4	Enter the updated information	• Perform information validation.
5	Dean has updated the course portfolio data successfully	• Course portfolio is updated successfully.

Use case: Delete the course portfolio (Course Portfolio Modules)		
Overview: This case describes how the dean deletes course portfolio.		
Pre – conditions: 1. The dean must login		
Post – conditions: 1. Successful condition • Dean has deleted university data.		
Event		Response
1	Dean login to own position portal.	• Perform dean account validation.
2	Dean selects "Course Portfolio"	• Enter to the Course Portfolio list webpage.
3	Select which course portfolio want to delete and click the "X" button	• Prompt out the confirm message to delete the course portfolio.
4	Dean has deleted the course portfolio successfully	• Course portfolio is deleted successfully.

Use case: Search the course portfolio (Course Portfolio Modules)		
Overview: This case describes how the dean searches the course portfolio		
Pre – conditions: 1. The dean must login.		
Post – conditions: 1. Successful condition • Dean has searched the course portfolio successfully.		
Event		Response
1	Dean login to own position portal.	• Perform dean account validation.
2	Dean selects "Course Portfolio"	• Enter to the Course Portfolio list webpage.
3	Search the course portfolio information (Faculty, Department, Semester, Subject Name…)	• Show the searched course portfolio result.
4	Dean has searched the course portfolio successfully.	• Dean can continues to do action like edit or delete.

Use case: Generate the e-portfolio report (Course Portfolio Modules)		
Overview: This case describes how the dean or HOD generate the e-portfolio report.		
Pre – conditions: 1. The dean or HOD must login.		
Post – conditions: 1. Successful condition • Dean or HOD is able to generate the e-portfolio report successfully.		
Event		Response
1	Dean or HOD login to own position portal.	• Perform dean or HOD account validation.
2	Dean or HOD selects "Course Portfolio"	• Enter to the Course Portfolio list webpage.
3	Dean or HOD can searches which course and select it. (HOD just can searches own department of course only.)	• Show the searched course data results and enter into course action page.
4	Click "E – Portfolio report"	• Enter to the result of course portfolio webpage.
5	Dean or HOD can click the "Download" button to download it.	• E- Portfolio is download successfully.

Use case: View all the course materials (Course Portfolio Modules)		
Overview: This case describes how the dean or HOD view all the course materials.		
Pre – conditions: 1. The dean or HOD must login.		
Post – conditions: 1. Successful condition • Dean or HOD is able to view the course materials successfully.		
Event		Response
1	Dean or HOD login to own position portal.	• Perform dean or HOD account validation.
2	Dean or HOD selects "Course Portfolio"	• Enter to the Course Portfolio list webpage.
3	Dean or HOD can searches which course and select it. (HOD just can searches own department of course only.)	• Show the searched course data results and enter into course action page.
4	Click which action is needed to view.	• Enter to the action of result webpage.
5	Dean or HOD has view the course materials successfully.	• Course portfolio is reviewed successfully.

4.5.5 Faculty Portfolio Modules

Use case: Create new folder in faculty portfolio (Faculty Portfolio Modules)		
Overview: This case describes how the lecturer create a new folder in faculty portfolio.		
Pre – conditions: 1. The dean must login.		
Post – conditions: 1. Successful condition • Dean has created a new folder successfully.		
Event		Response
1	Dean login to own position portal.	• Perform dean account validation.
2	Dean selects "Faculty Portfolio"	• Enter to the faculty portfolio webpage.
3	Select "…" and click the Make a new folder.	• Prompt a model that is a form to insert the folder name.
4	Select the inserted folder can link into the folder.	• Enter the inserted folder and then can upload the files in this folder.
5	Dean has inserted a new folder successfully	• New folder is inserted successfully.

Use case: Edit the folder name (Faculty Portfolio Modules)		
Overview: This case describes how the dean edit the folder name in faculty portfolio.		
Pre – conditions: 1. The dean must login.		
Post – conditions: 1. Successful condition • Dean has updated folder name successfully.		
Event		Response
1	Dean login to own position portal.	• Perform dean account validation.
2	Dean selects "Faculty Portfolio"	• Enter to the faculty portfolio webpage.
3	Select the folder and click the "wrench" button.	• Enter the inserted folder and then can upload the files in this folder.
4	Enter new folder name and save it.	• The folder name is updated successfully.

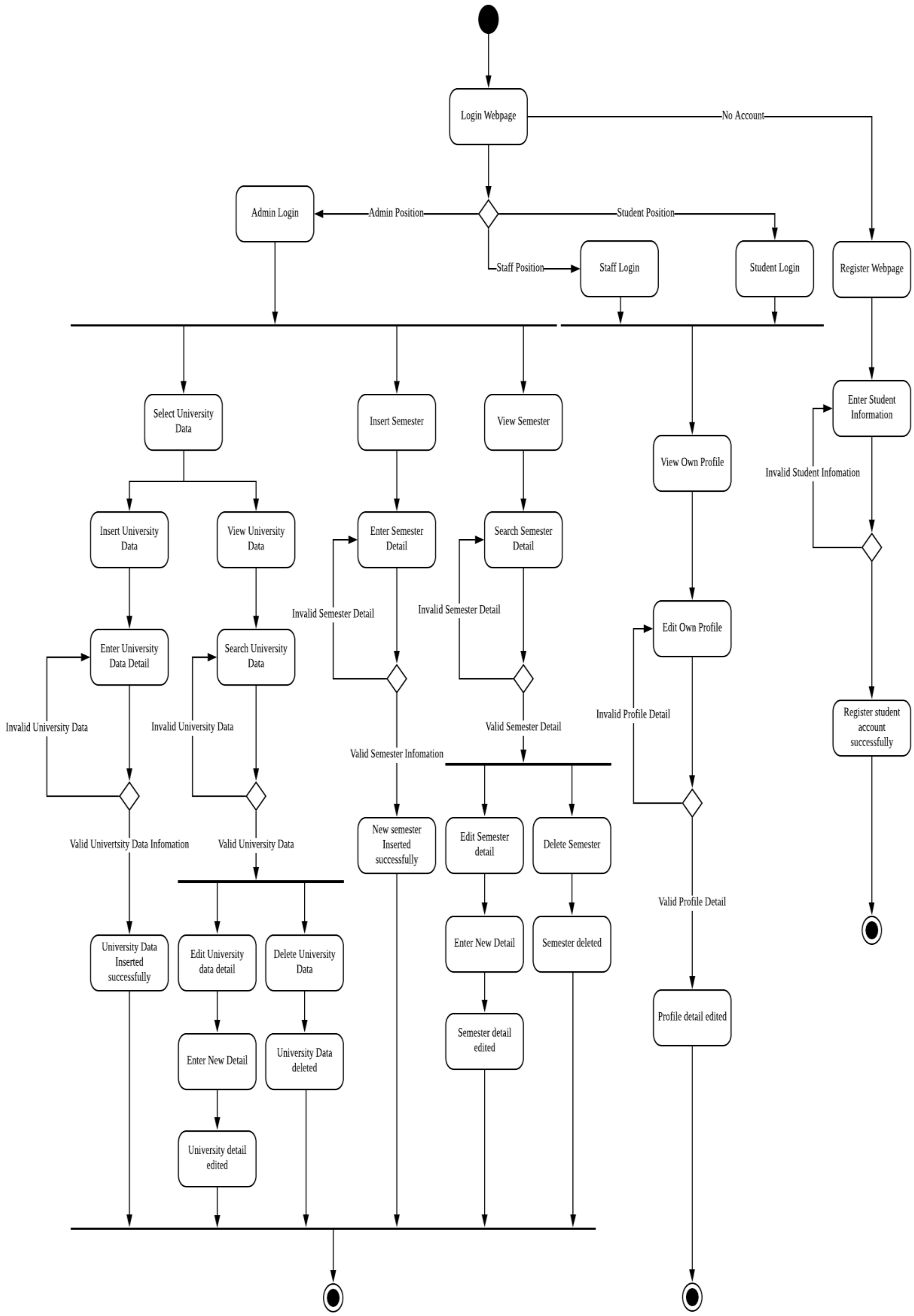
Use case: Upload faculty materials file (Faculty Portfolio Modules)		
Overview: This case describes how the dean how to upload the faculty materials file.		
Pre – conditions: 1. The dean must login.		
Post – conditions: 1. Successful condition • Dean has upload the file successfully.		
Event		Response
1	Dean login to own position portal.	• Perform dean account validation.
2	Dean selects “Faculty Portfolio”	• Enter to the faculty portfolio webpage.
3	Select the folder and then click the “…” and select the Upload Files.	• Prompt a model that can be upload files and save it in that folder.
4	Lecturer can drop or select the file and rename it.	• The file name will be follow it.
5	Dean has inserted the faculty portfolio successfully	• Faculty portfolio is inserted in course successfully.

Use case: Remove the note or folder (Faculty Portfolio Modules)		
Overview: This case describes how the dean deletes the note or folder.		
Pre – conditions: 1. The dean must login		
Post – conditions: 1. Successful condition • Dean has deletes the note or folder.		
Event		Response
1	Dean login to own position portal.	• Perform dean account validation.
2	Dean selects “Faculty Portfolio”	• Enter to the faculty portfolio webpage.
3	Select which file or folder wanted to delete and click “X” button.	• Prompt out the confirm message to delete it.
4	Dean has deleted the faculty portfolio successfully	• Faculty portfolio is deleted successfully.

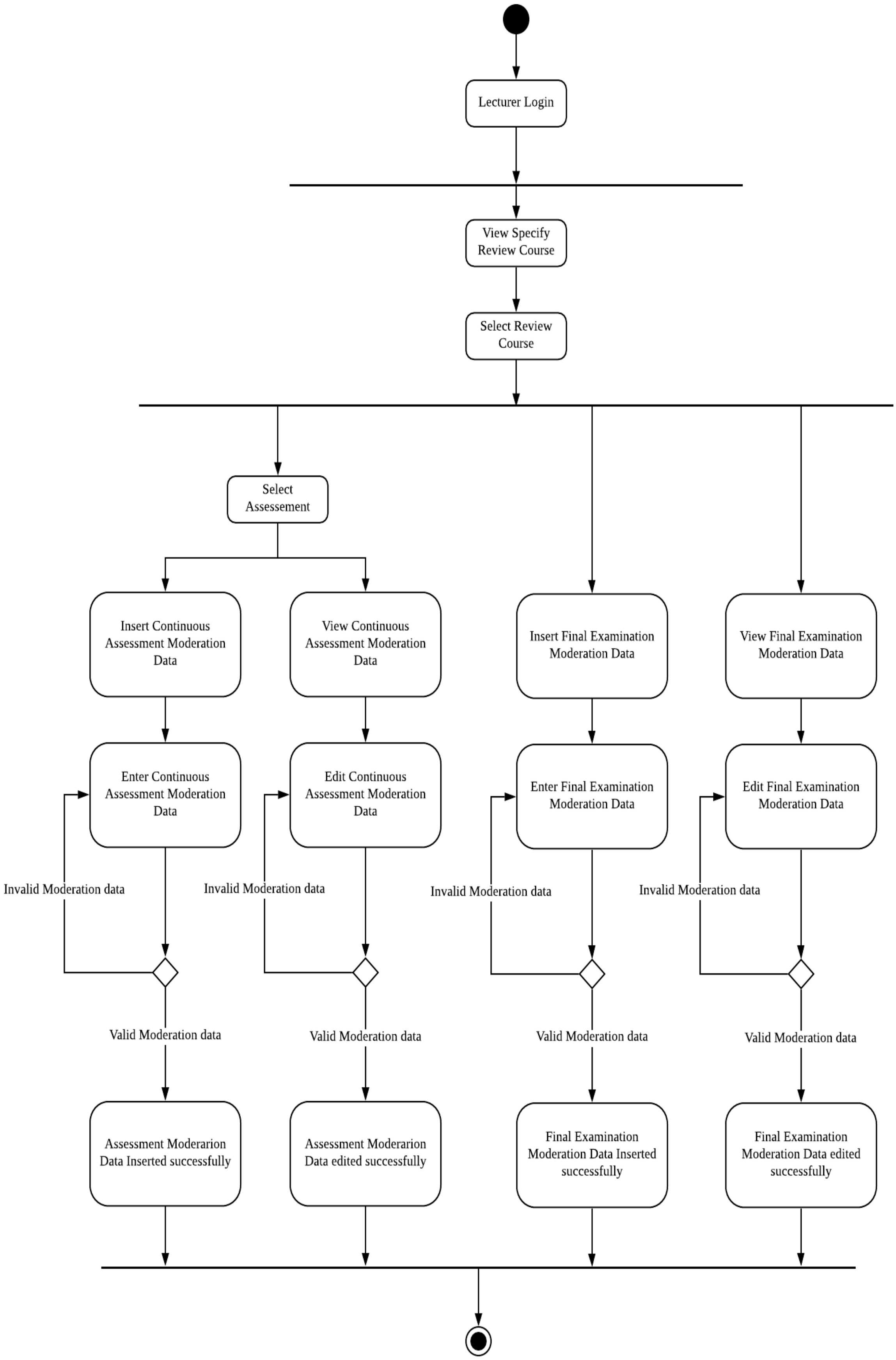
<table>
<tr><td colspan="3">Use case: View or Download the faculty materials, Syllabus or Lecturer CV (Faculty Portfolio Modules)</td></tr>
<tr><td colspan="3">Overview: This case describes how the dean view and download the faculty portfolio.</td></tr>
<tr><td colspan="3">Pre – conditions:
1. The dean must login.</td></tr>
<tr><td colspan="3">Post – conditions:
1. Successful condition
• Dean is able to download the faculty portfolio successfully.</td></tr>
<tr><td colspan="2">Event</td><td>Response</td></tr>
<tr><td>1</td><td>Dean login to own position portal.</td><td>• Perform dean account validation.</td></tr>
<tr><td>2</td><td>Dean selects “Faculty Portfolio”</td><td>• Enter to the faculty portfolio webpage.</td></tr>
<tr><td>3</td><td>Just Click the materials</td><td>• It will download it.</td></tr>
<tr><td>4</td><td>Dean has download the faculty portfolio successfully</td><td>• Faculty Portfolio is downloaded successfully.</td></tr>
</table>

4.6 Activity diagram

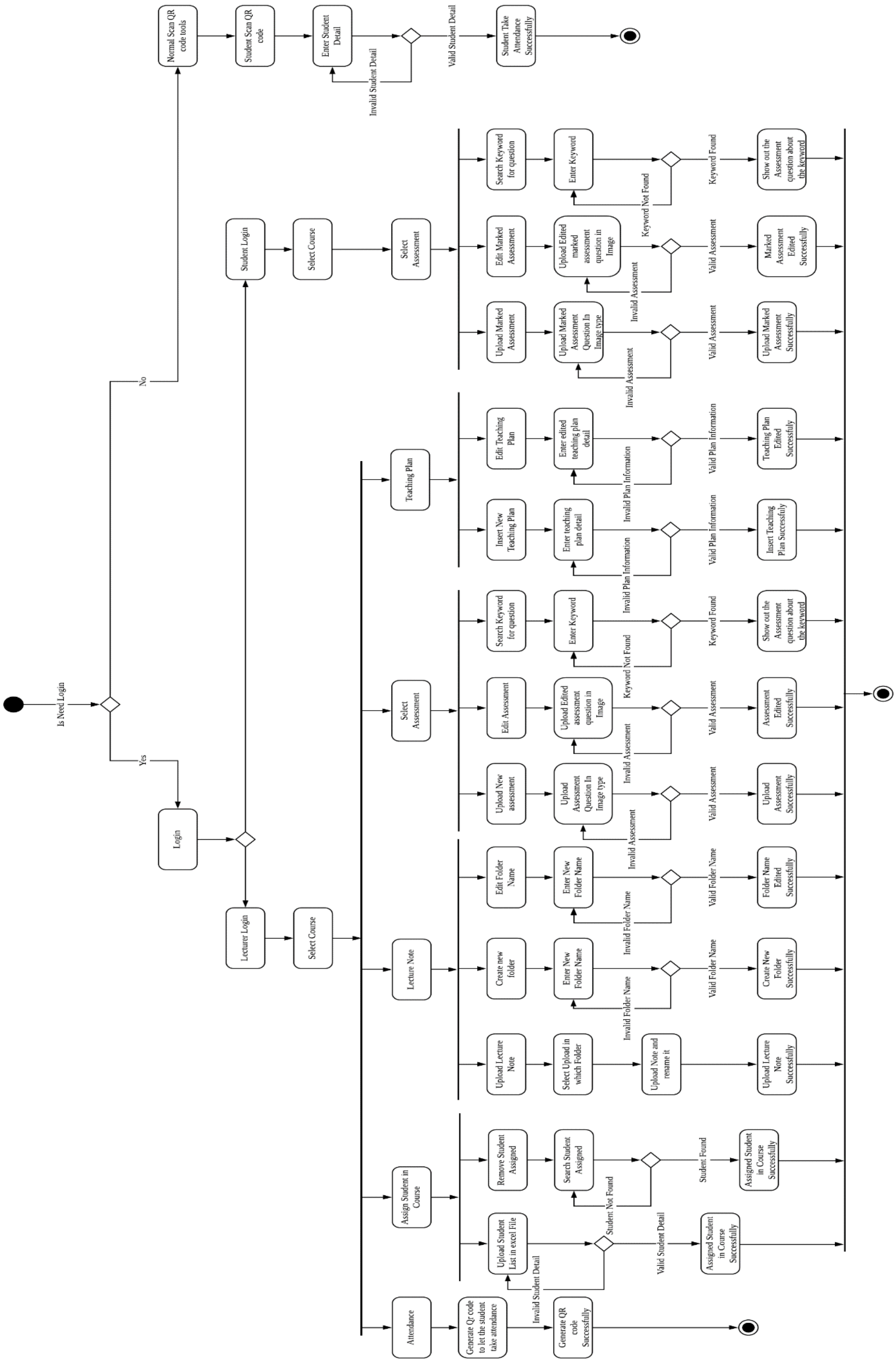
4.6.1 Information Modules



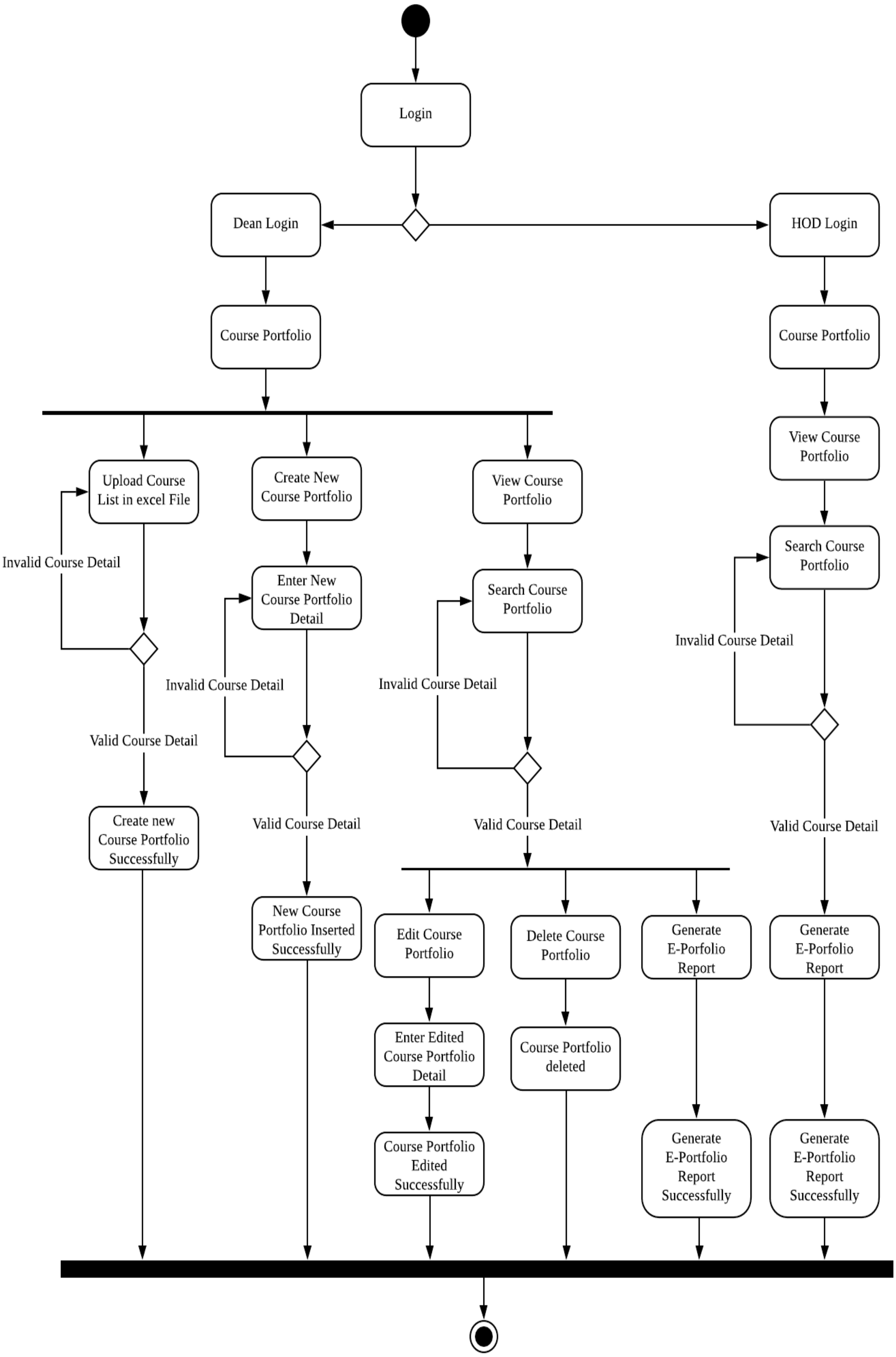
4.6.2 Moderator Modules



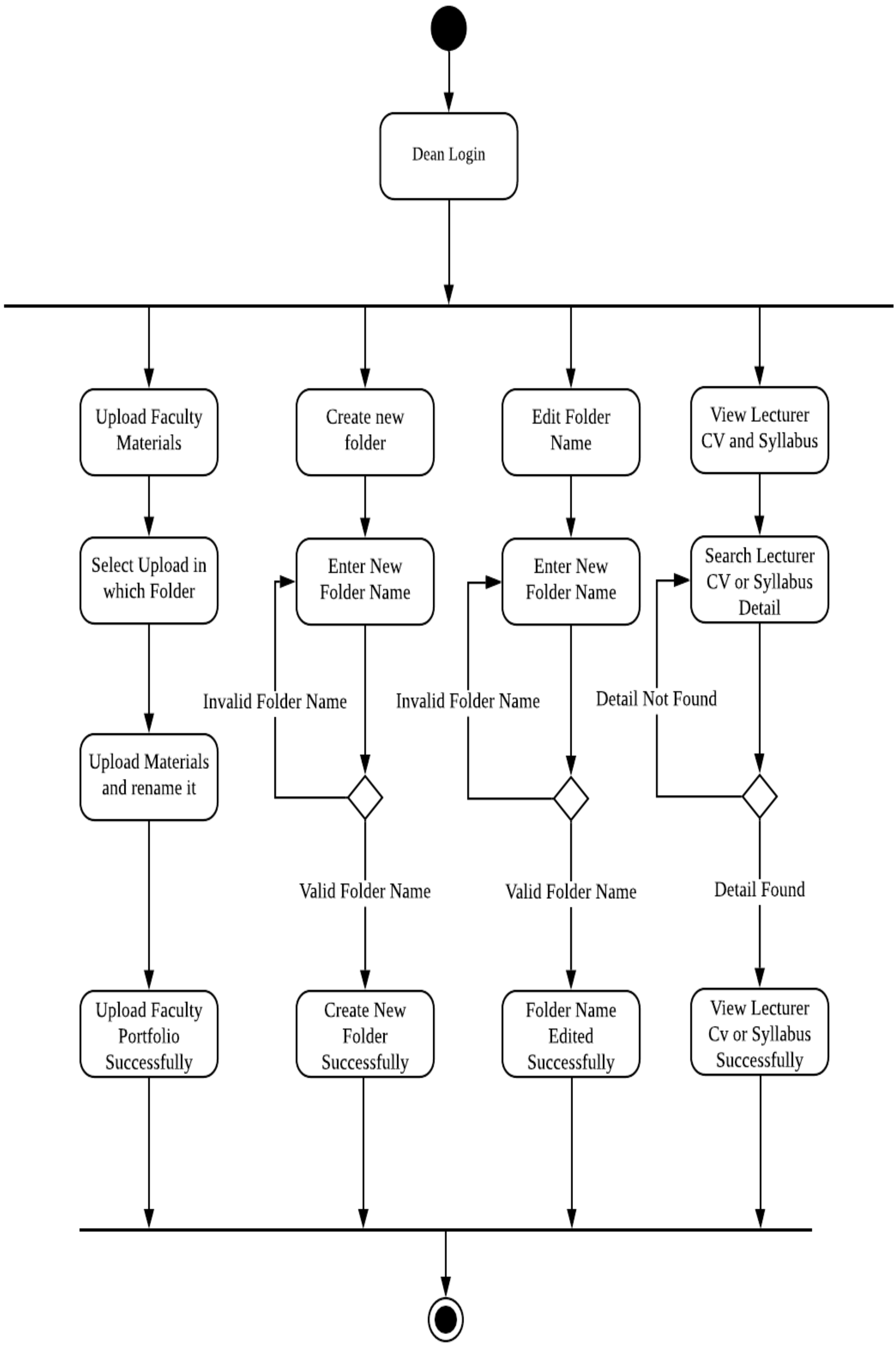
4.6.3 Course Modules



4.6.4 Course Portfolio Modules

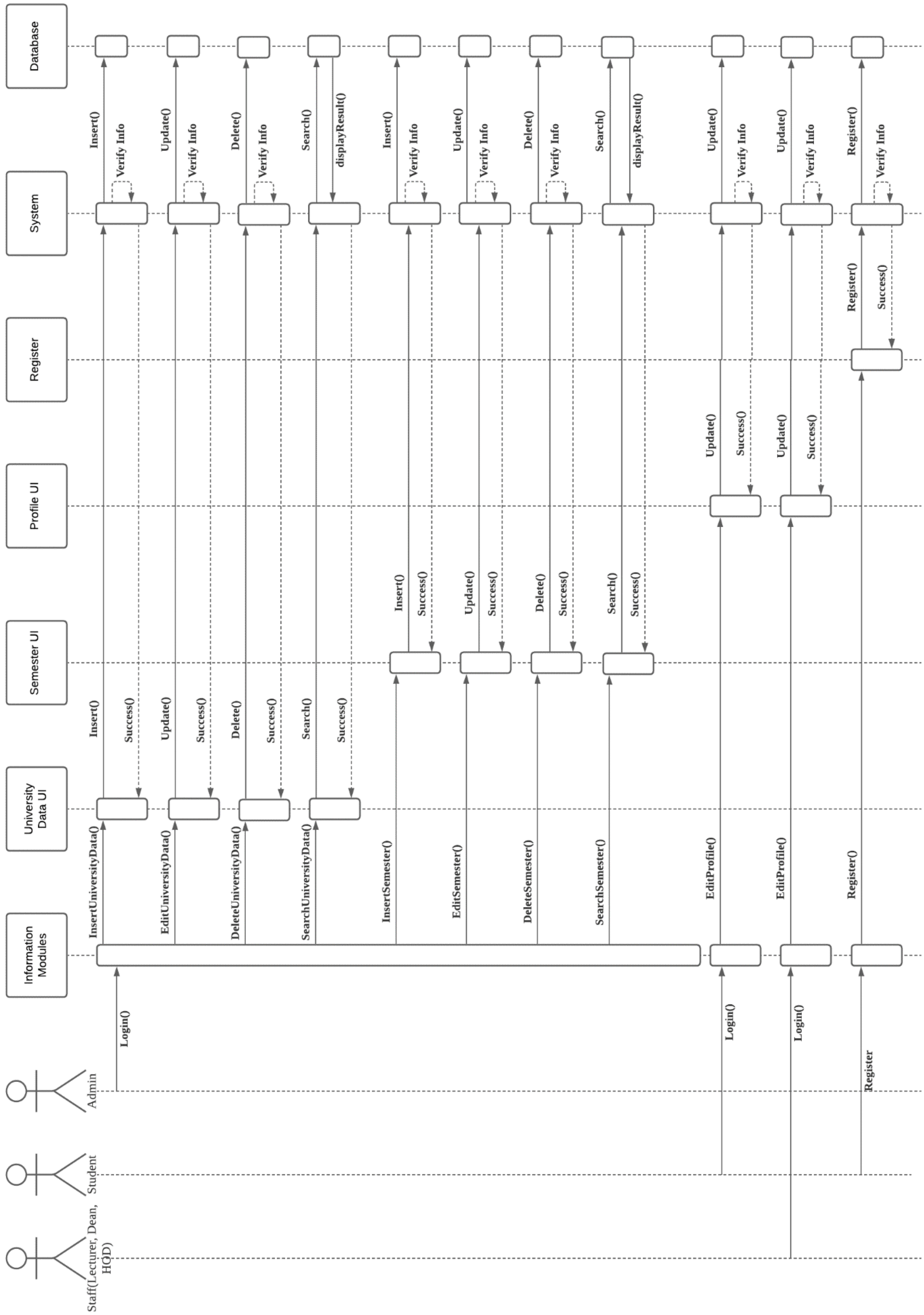


4.6.5 Faculty Portfolio Modules

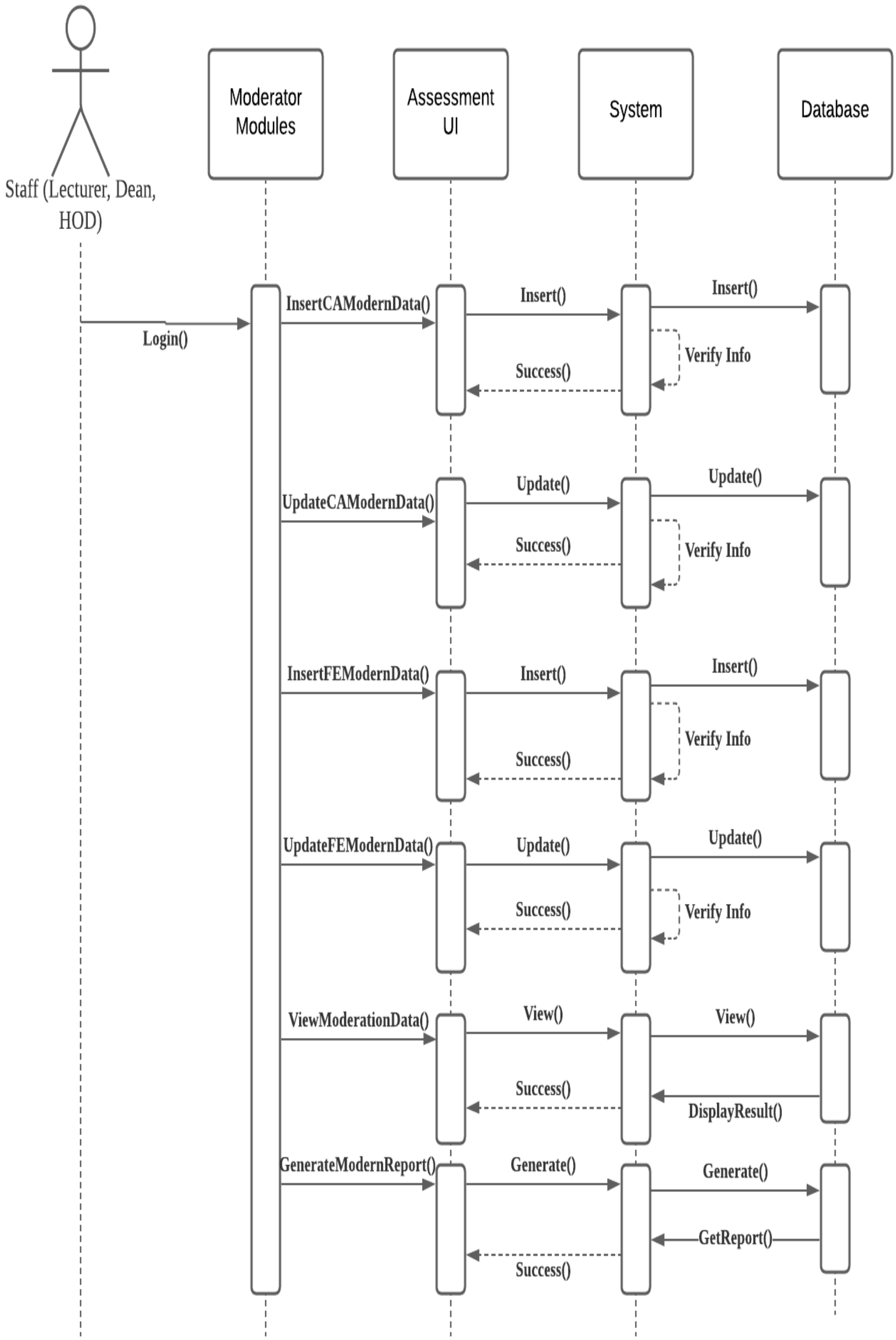


4.7 Sequence diagram

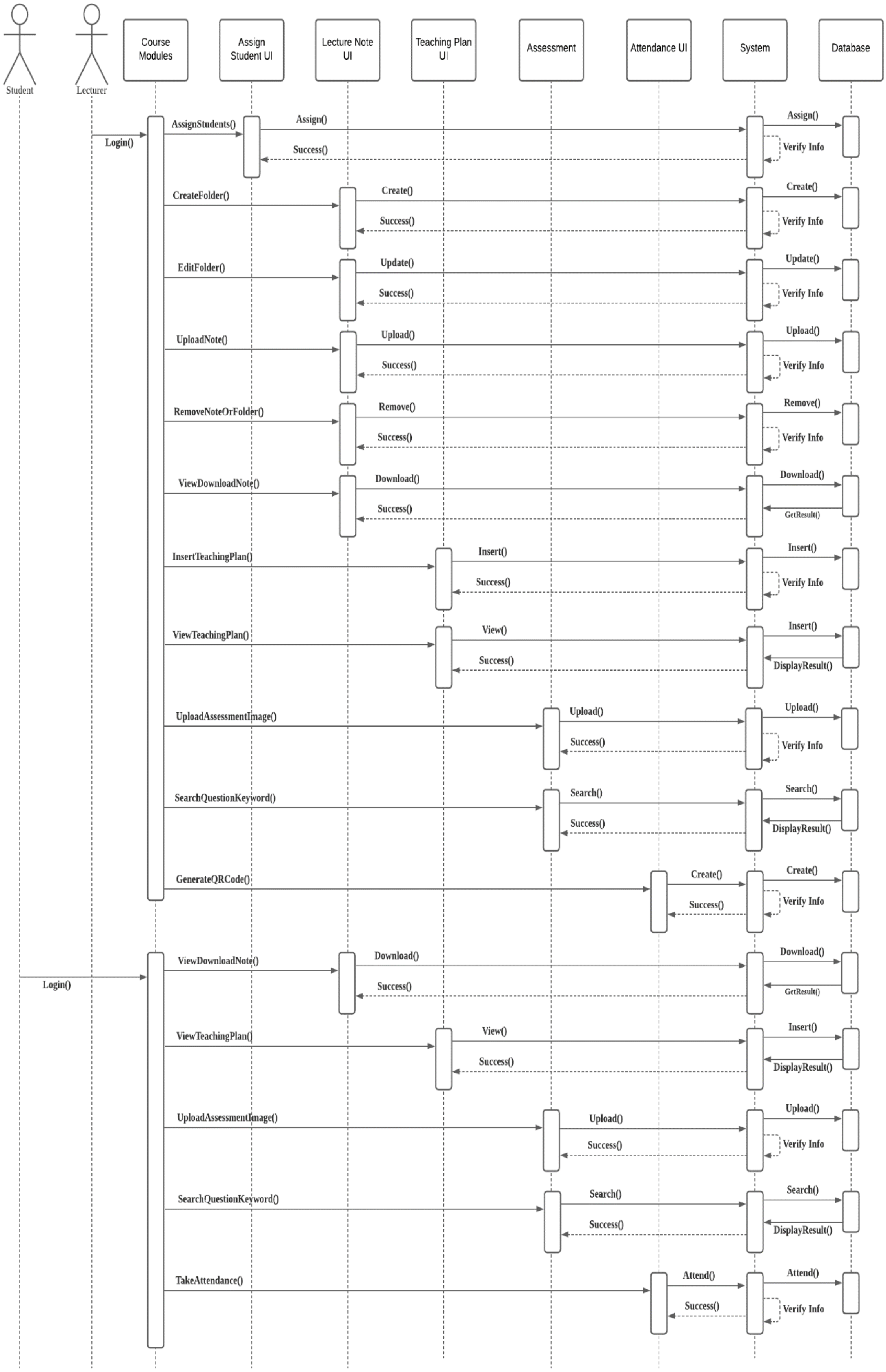
4.7.1 Information Modules



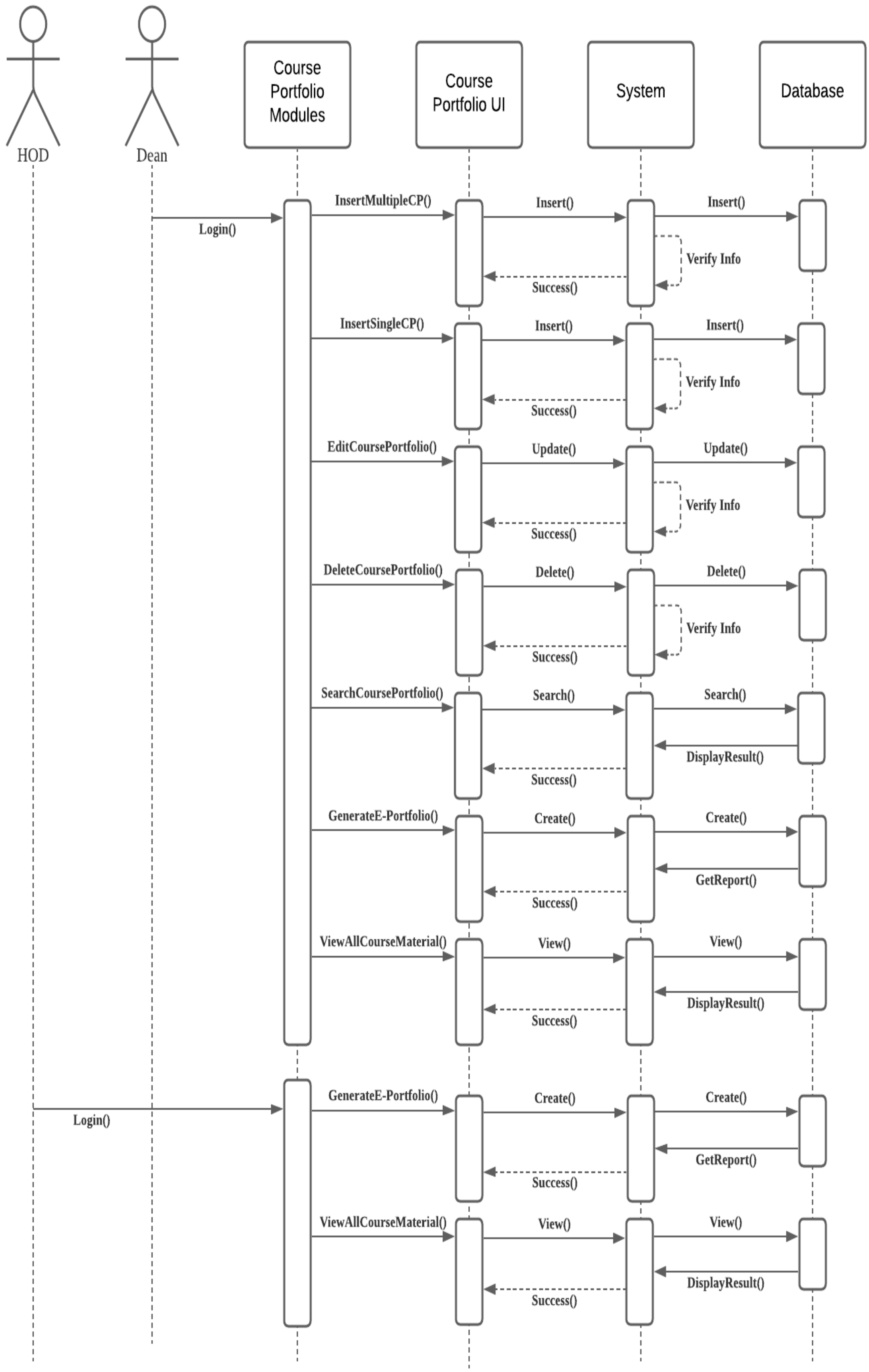
4.7.2 Moderator Modules



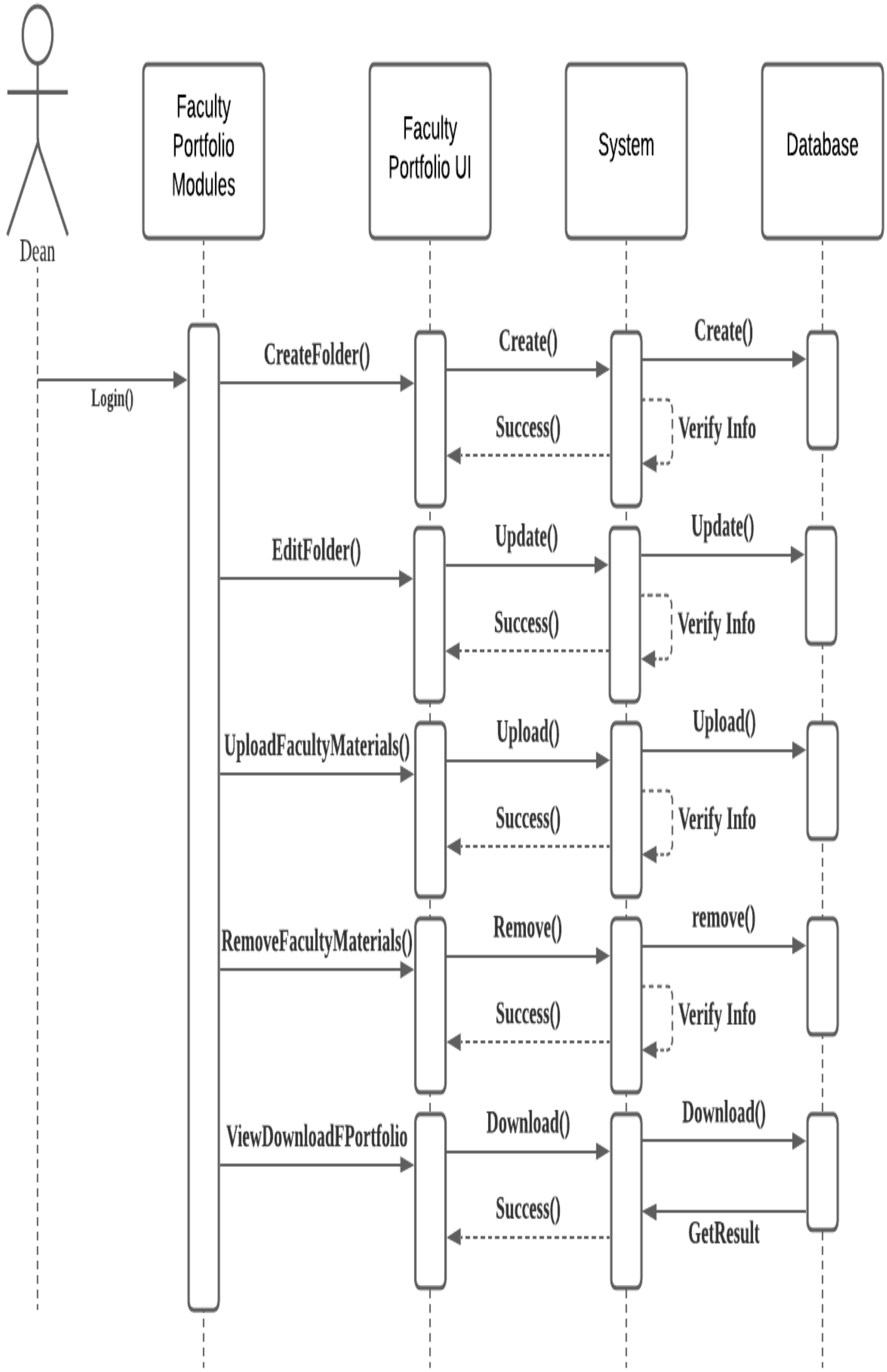
4.7.3 Course Modules



4.7.4 Course Portfolio Modules



4.7.5 Faculty Portfolio Modules



5.0 Implementation

The following section will describe the implementation of system.

5.1 System Interface Design

According to Margaret Rouse, she said that the user interface (UI) is everything designed into an information device with which a person may interact. This can include display screens, keyboards, a mouse and the appearance of a desktop. It is also the way through which a user interacts with an application or a website. The growing dependence of many companies on web applications and mobile applications has led many companies to place increased priority on UI in an effort to improve the user's overall experience. So, UI is very important for the user to interact with the system.

Below are the main characteristics that applied to University Content Management System:

1. Clarity (Clear)

- With the clarity of user interface system, user is able to interact with the system easily and does not spend excess time in interacting the system. So, the characteristic of clarity of user interface in University Content Management System is able to allow the user to perform the functions easily in the daily.

2. Consistent

- The arrangement and patterns of the interface elements must be consistent to prevent the confusion of the admin user when he/she interacts to the system. Furthermore, the consistent of the interface elements also can speed up the daily process due to the familiar of the user interface.

5.1.1 University Content Management System Interface Design

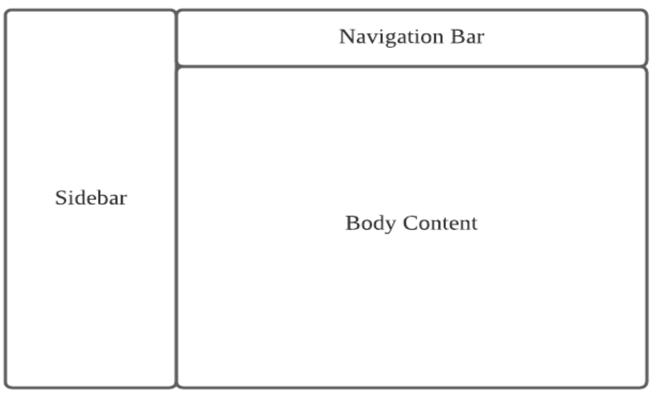


Figure 5.1.1.1: Interface Design Content

In University Content management system, the web-based system for user is designed by using HTML 5 and CSS. In this web-based system, it consists of three parts, which are sidebar, navigation bar and the body content.

For the Navigation bar, it is placed above of the body content of the system. It contains the user setting and it is saw easily.

For the sidebar, it is placed on the left hand side of the screen of the system. It consists of buttons for selections to the modules. The corresponding page will display in body content when the button is clicked.

For the body content, it displays the data. It also contains few buttons to allow user performs the functions. The layout is shown as below.

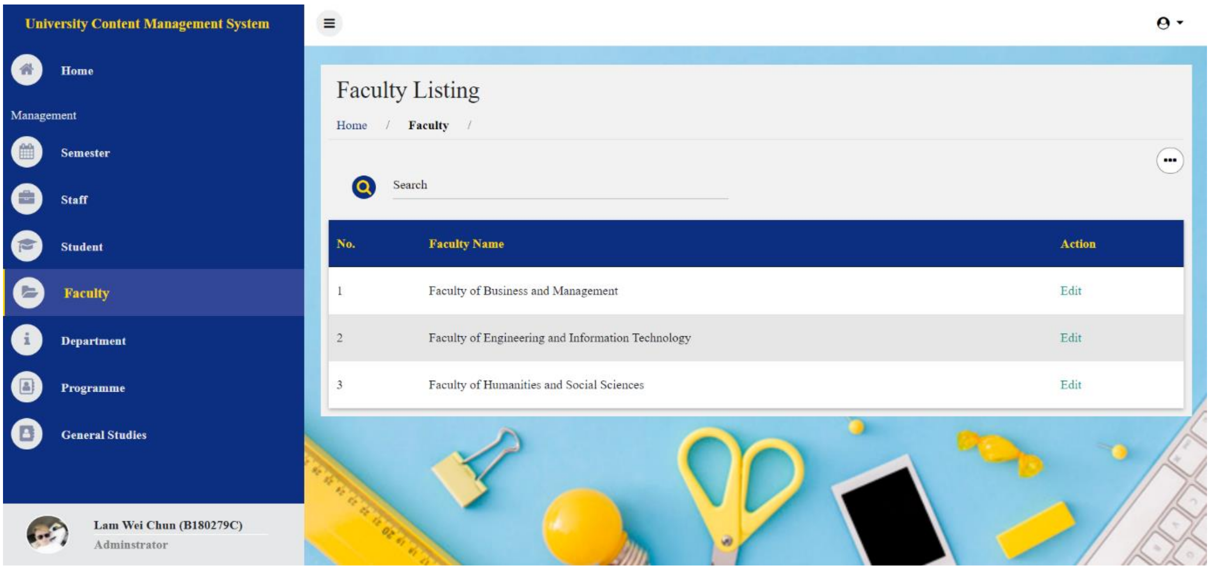


Figure 5.1.1.2: Admin Panel Interface Design

The webpage after design. Above the content page also provide the “Title” which indicates the current page's location. The corresponding page will display in content block when button is clicked. Data table or form will be displayed when the button is clicked.

6.0 Testing

Testing is an important part of software of software development. It is an activity that allow us to test and evaluate our system that we develop to meet its required results. It is crucial for us to find any remaining bugs or error in the system before being release to the general public. A test should be held when there any slight change in the source. To follow this ideology we will be testing every single function in our system.

6.1 Approach

Since this is final year project 1 and the system is not completed yet, so I decide to choose the unit testing to test each features separately. Unit testing is a level of software testing where individual units or components of the software is tested. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. Hence, I perform the unit testing by manually.

6.2 Features to be tested

University Content Management System features to be tested.

Features	Description
Login	• Login as own position portal.
Logout	• Logout from the management system.
Email Verified Confirm	• Check the account is already email verified yet.
Information Modules	• Add/edit/delete the university data detail. • Add/edit/delete the semester detail.
Course Modules	• Assign and remove student in the course. • Lecture Note Function • Teaching Plan Function
Course Portfolio Modules	• Add/edit/delete the Course Portfolio
Faculty Portfolio Modules	• Faculty Portfolio function

6.3 System testing

6.3.1 Cases 1: Login

Step	Test Cases	Test Data	Expected result	Actual result	Status (Pass/Fail)
1	Click ”Login” button		Redirect to login page	Redirect to login page	Pass

2	Fill In Incorrect email or password click Login button	Email: xxx@xxx.com Password: 12345678	Show up messages “Email-Address And Password Are Wrong.”	Show up messages “Email-Address And Password Are Wrong.”	Pass
3	Fill in admin email and password click login button	Email: weijunlam1@-gmail.com Password: password	Redirect to admin portal page	Redirect to admin portal page	Pass
4	Fill in dean email and password click login button	Email: lr1111@-sc.edu.my Password: password	Redirect to dean portal page	Redirect to dean portal page	Pass

6.3.2 Cases 2: Logout

Step	Test Cases	Test Data	Expected result	Actual result	Status (Pass/Fail)
1	Click ”Logout” button		Redirect to login page	Redirect to login page	Pass

6.3.3 Cases 3: Email Verified Confirm

Step	Test Cases	Test Data	Expected result	Actual result	Status (Pass/Fail)
1	After verify the email and password, need to check the account email verified confirm.		If Not, Redirect to Verify Your Email Address page.	If Not, Redirect to Verify Your Email Address page.	Pass

			If Yes, Redirect to the related position portal page.	If Yes, Redirect to the related position portal page.	

6.3.4 Cases 4: Information Modules Functions

Step	Test Cases	Test Data	Expected result	Actual result	Status (Pass/Fail)
1	Enter correct email and password at login page	Email: weijunlam1@-gmail.com Password: password	Redirect to admin portal page	Redirect to admin portal page	Pass
2	Press which university data is need to add and click "…" to select the add action button.		Display the form webpage	Display the form webpage	Pass
3	Fill in all required information details and submit it.	Required Information	Redirect to university data list page and show message **"**Data Added**".**	Redirect to university data list page and show message "Data Added".	Pass
4	Click 'Edit" button on university data list page	-	Display the edit form webpage	Display the edit form webpage	Pass

5	Fill in the edited required information details and submit it.	Required Information	Redirect to university data list page and show message “Data Updated”.	Redirect to university data list page and show message “Data Updated”.	Pass
6	Press “Semester” and click “…” to select the add action button.		Display the form webpage	Display the form webpage	Pass
7	Fill in all required information details and submit it.	Required Information	Redirect to semester list page and show message “Data Added**”.**	Redirect to semester list page and show message “Data Added”.	Pass
8	Click ‘Edit” button on semester list page	-	Display the edit form webpage	Display the edit form webpage	Pass
9	Fill in the edited required information details and submit it.	Required Information	Redirect to semester list page and show message “Data Updated”.	Redirect to semester list page and show message “Data Updated”.	Pass

6.3.5 Cases 5: Course Modules Functions

Step	Test Cases	Test Data	Expected result	Actual result	Status (Pass/Fail)
1	Enter correct email and password at login page	Email: lr1111@sc.edu.my Password: password	Redirect to dean portal page	Redirect to dean portal page	Pass

2	Press “Course” button and select which course you need to action.	-	Redirect to course list page and redirect to that selected course action page.	Redirect to course list page and redirect to that selected course action page.	Pass
3	Select the student list button and click “…” and press the upload a file.	-	Redirect to assign student list page and prop up a model for upload a file such as the template of student list.	Redirect to assign student list page and prop up a model for upload a file such as the template of student list.	Pass
4	Upload a file that follow the template student list and submit it.	A excel file the data is follow the template.	-Display a table that listing the file data. -Redirect to assign student list page and show message “data added” or which data got error.	-Display a table that listing the file data. -Redirect to assign student list page and show message “data added” or which data got error.	Pass
5	Upload a file that without follow the template student list and submit it.	A excel file the data without follow the template.	Show message “The Input Data are not completed. Please Check Again the excel file of data.”	Show message “The Input Data are not completed. Please Check Again the excel file of data.”	Pass

6	Select the Lecture Note button and click "…" and press the make a new folder.		Prop up a model for fill in the folder name.	Prop up a model for fill in the folder name.	Pass
7	Fill in folder name and submit it	Folder Name: Chapter 1	Show message "New Folder Added Successfully"	Show message "New Folder Added Successfully"	Pass
8	Click the wrench icon button and enter the edited folder name and submit it	Folder Name: Chapter-1	Show message "Edit Folder Name Successfully"	Show message "Edit Folder Name Successfully"	Pass
9	click "…" and press the Upload Files and submit it.	Files : Chapter 1 pptx, Chapter 2 pptx,	Show message "New Document Added Successfully"	Show message "New Document Added Successfully"	Pass
10	Click "X" button to remove the file or folder and click the prop up "Yes".	-	Show message "Remove Successfully"	Show message "Remove Successfully"	Pass
11	Click "…" and press the download all note.		Download a zip file that has all the note.	Download a zip file that has all the note.	Pass
12	Select the teaching plan button and click "…" and press the manage assessment method.	-	Redirect to assessment method form page and main data is get from syllabus.	Redirect to assessment method form page and main data is get from syllabus.	Pass

13	Fill in the teaching method and assessment markdown in each CLO and submit it.	-Teaching Method: Lecture, Tutorials -Markdown: Assignment mark, Final Exam Mark.	Redirect to teaching plan page.	Redirect to teaching plan page.	Pass
14	click "…" and press the manage weekly plan.	-	Redirect to weekly plan form page and topic data is get from syllabus.	Redirect to weekly plan form page and topic data is get from syllabus.	Pass
15	Fill in the required information and submit it	-Fill Each weekly plan -Can be empty	Redirect to teaching plan page.	Redirect to teaching plan page.	Pass

6.3.6 Cases 6: Course Portfolio Modules Functions

Step	Test Cases	Test Data	Expected result	Actual result	Status (Pass/Fail)
1	Enter correct email and password at login page	Email: weijunlam1@-gmail.com Password: password	Redirect to admin portal page	Redirect to admin portal page	Pass
2	Press Course Portfolio click "…" to select the make a new course.		Display the course form webpage	Display the course form webpage	Pass

3	Fill in all required information details and submit it.	Required Information	-Redirect to course portfolio list page and show message "Data Added". -The new course portfolio will be a new course and added in course modules that follow by the lecturer.	Redirect to course portfolio list page and show message "Data Added". - The new course portfolio will be a new course and added in course modules that follow by the lecturer.	Pass
4	Click 'wrench" icon button on course portfolio list page	-	Display the edit course form webpage	Display the edit course form webpage	Pass
5	Fill in the edited required information details and submit it.	Required Information	Redirect to course portfolio list page and show message "Data Updated".	Redirect to course portfolio list page and show message "Data Updated".	Pass
6	Click "…" and press the add multiple courses	-	Prop up a model for upload a file such as the template of course list.	Prop up a model for upload a file such as the template of course list.	Pass

7	Upload a file that follow the template course list and submit it.	A excel file the data is follow the template.	-Display a table that listing the file data. -Redirect to course portfolio list page and show message "data added" or which data got error.	- Display a table that listing the file data. -Redirect to course portfolio list page and show message "data added" or which data got error.	Pass
8	Click "X" button to remove the course portfolio and click the prop up "Yes".	-	-Show message "Remove Successfully" - The course in course modules also removed.	Show message "Remove Successfully" -The course in course modules also removed.	Pass

6.3.7 Cases 7: Faculty Portfolio Modules Functions

Step	Test Cases	Test Data	Expected result	Actual result	Status (Pass/Fail)
1	Enter correct email and password at login page	Email: lr1111@sc.edu.my Password: password	Redirect to dean portal page	Redirect to dean portal page	Pass
2	Select the Faculty Portfolio button and click "…" and press the make a new folder.	-	Prop up a model for fill in the folder name.	Prop up a model for fill in the folder name.	Pass

3	Fill in folder name and submit it	Folder Name: Activities	Show message “New Folder Added Successfully”	Show message “New Folder Added Successfully”	Pass
4	Click the wrench icon button and enter the edited folder name and submit it	Folder Name: Activities In Times	Show message “Edit Folder Name Successfully”	Show message “Edit Folder Name Successfully”	Pass
5	click “…” and press the Upload Files and submit it.	Files : Faculty … File.	Show message “New Document Added Successfully”	Show message “New Document Added Successfully”	Pass
6	Click “X” button to remove the file or folder and click the prop up “Yes”.	-	Show message “Remove Successfully”	Show message “Remove Successfully”	Pass
7	Click the Lecture Note or Syllabus and select which one for needed to view.	-	Download it successfully.	Download it successfully.	Pass

7.0 Evaluation

According to the main goal of the project's university content management system, the system can enable university teachers and students to easily perform daily actions such as take attendance and so on. Teachers can also upload teaching materials to the system so that students can download materials anytime and anywhere via the Internet.

Another goal of the system is to reduce the paper documents in the university. There are many exam questions and homework submitted by the students in the university, which takes up space and is very difficult to search. The system can solve these problems well.

Overall this final year project 1 still is not a complete system but it still has lots and lots of improvement that I can do in the future to make this an even better system. University content management system should continue to be improved so that I can stand our ground against others that have been release to the public.

8.0 Recommendations

This Section will be dedicated to list my several improvements which my University Content Management System can be explored in future development.

Make a mobile version

With the rate of smartphone being improve on and how smartphone has become part of our daily life. Because of this and the general portability of smartphone make me consider to port into the android platform. So the students can use it wherever they.

Improve security

With hackers all around the world it would be the best for me to put my focus on my system security sometime in the future. Like checking for loopholes in my system or exploit in my system to prevent hacker from getting the upper hand on us.

9.0 Conclusion

The main goal of the university content management system is to make the system easy to use so that users can solve current problems and manage daily actions without encountering many problems. The university content management system will help Dean or HOD to track the performance status of teachers and students. The university content management system is a big and complex project for me. This is the first time I have developed a complete system.

During this semester, when the teacher introduced the system to me, he supported me in developing the system. I spent about 2 weeks thinking about how to develop the system, and all the background research, investigation and analysis of how the university content management system works. After completing all the research, I spent 2 months drawing all the demand maps and completing the system.

Of course, during the development stage, I encountered many problems, especially in the design drawings or code, but I still insisted on reading the solutions online and seeking advice and solutions from my friends and lecturers. In addition, in the process of developing the system, I learned many new ways to solve problems and new programming skills.

In conclusion, I am very happy to have the opportunity to develop my university content management system. With the experience gained from this project, I hope to become more efficient in developing the system.

References

aiim. (23 Sep, 2019). *Document Management vs. Content Management: What's the Difference?* Retrieved from Document Management vs. Content Management: What's the Difference?: https://info.aiim.org/aiim-blog/document-management-vs-content-management-whats-the-difference

cmswire. (July, 2018). *How Document Management and Content Management Differ*. Retrieved from How Document Management and Content Management Differ: https://www.cmswire.com/information-management/how-document-management-and-content-management-differ/

researchgate. (2016). *researchgate*. Retrieved from WEB CONTENT MANAGEMENT SYSTEM: https://www.researchgate.net/publication/299438184_WEB_CONTENT_MANAGEMENT_SYSTEM

APPENDIX A – Curriculum Vitae

Curriculum Vitae

Name: Lam Wei Chun

Age: 22

I/C No: 980424-01-5705

Address: 7, Jalan NB2 1/5, Taman Nusa Bestari2, 81300, Skudai, Johor.

H/P Number: 011-37757993

Student ID: B180279C

Class: Bose_18_C2

E-mail: weijunlam1@gmail.com

Educational Qualifications:

Name of Insituition	From-To	Highest Qualification Attained
SJK(C)Kuo Kuang 2	2005-2010	UPSR
SMK Taman Selesa Jaya	2011-2015	SPM
Southern University College	2016-2018	Diploma in information Technology
Southern University College	2018-present	Bachelor of Software Engineering (Hons)

Communication skill:

•Spoken and write level (Best= 10 and Worst= 1)

Language	Writing level	Speaking level
English	3	5
Malay	3	5
Mandarin	3	7

APPENDIX B – User Guide

Figure 1.0 Login Page

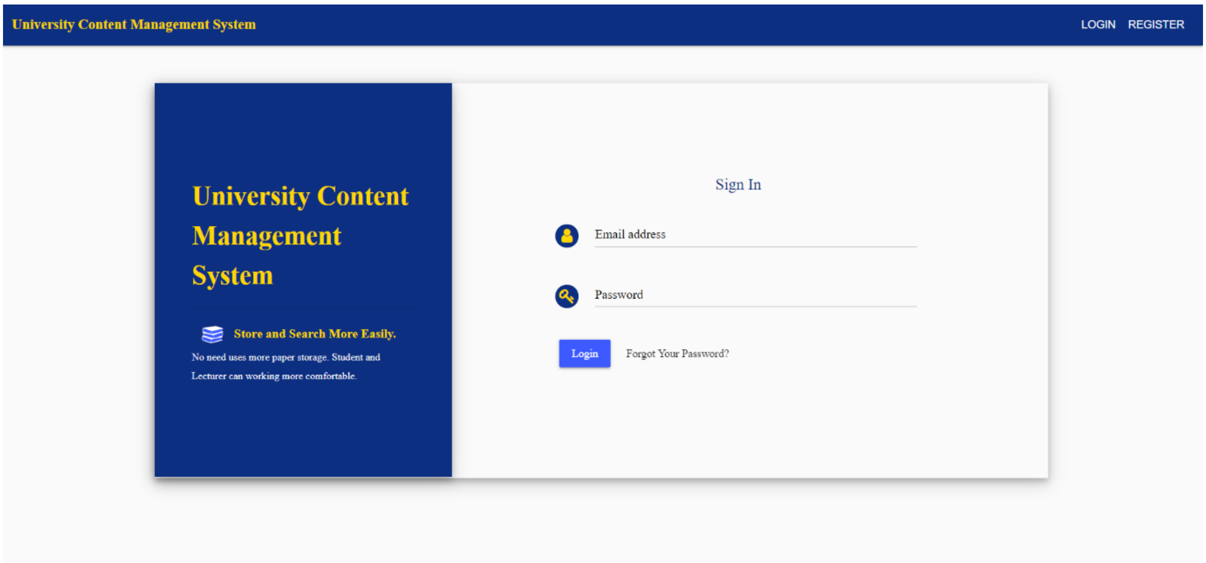


Figure 1.0 is the login page to the login all position panel of University Content Management System. If the username or password is wrong it will just refresh the page instead of entering the position panel of University Content Management System.

Figure 2.0 University Content Management System in Admin Panel

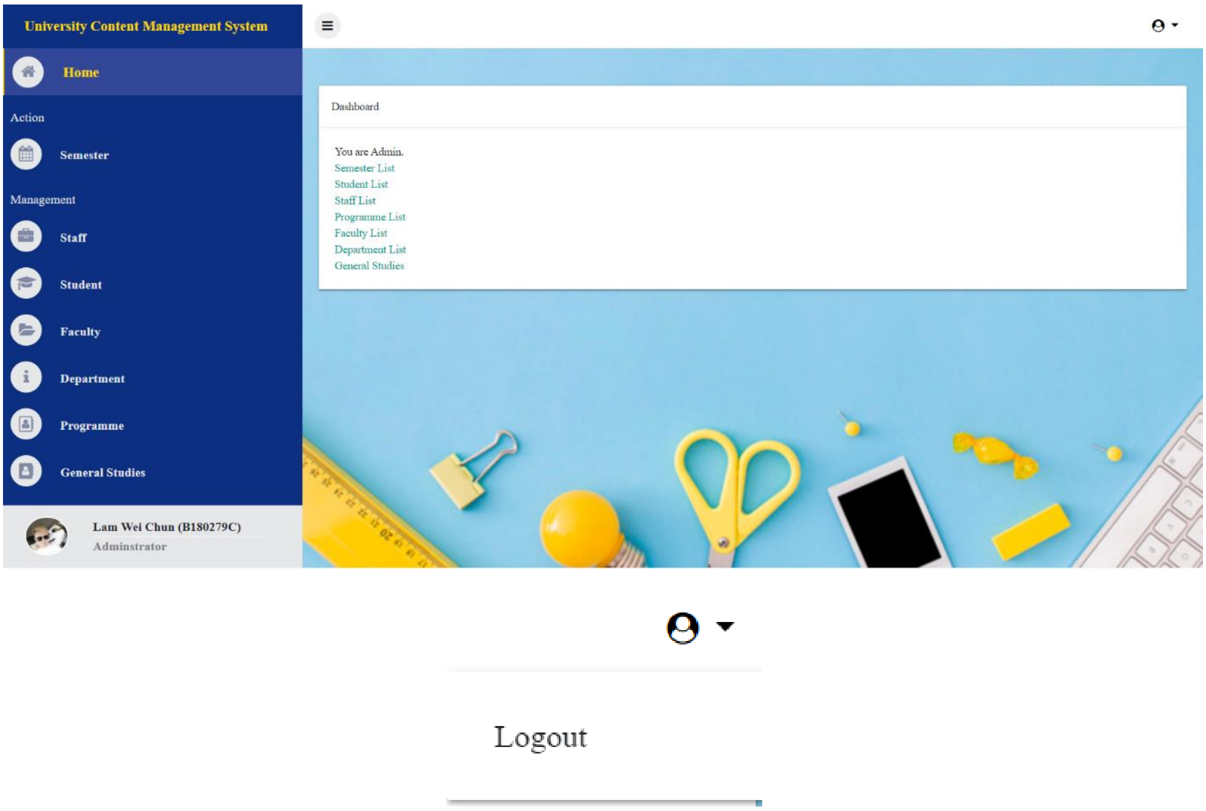
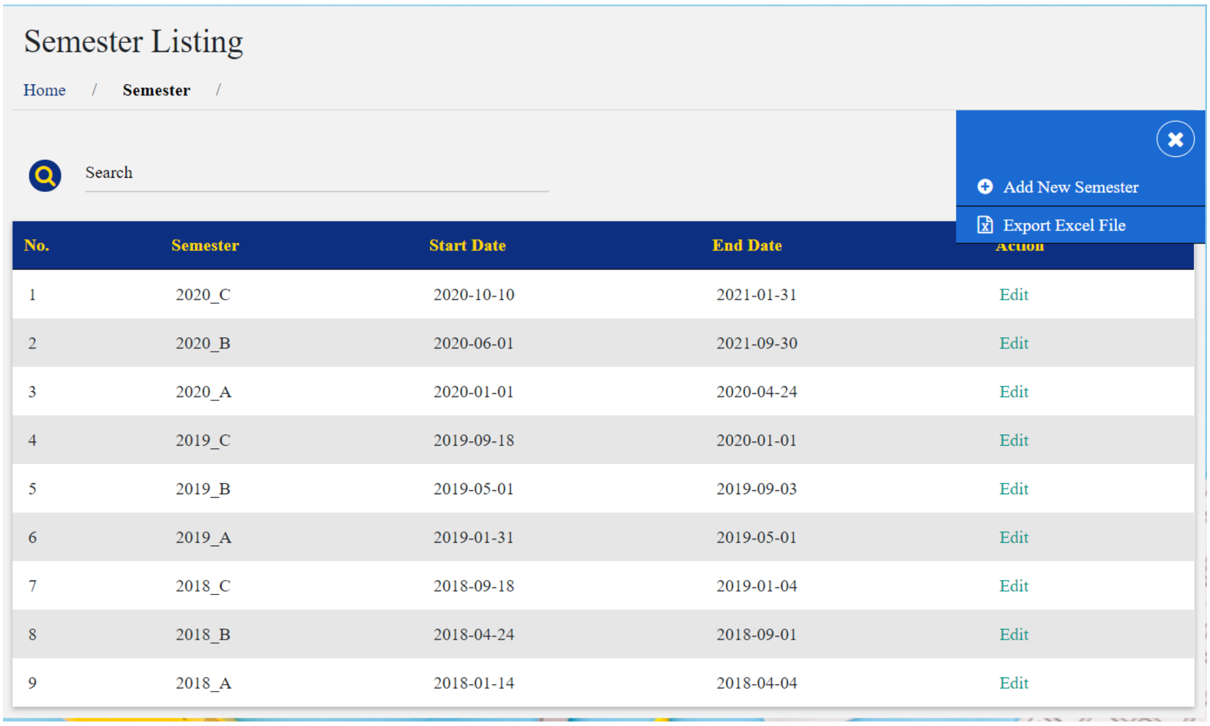


Figure 2.0 shows the home page of University content management system in admin panel. By pressing the user icon button on the top right it will list down the action of the user such as logout, after you click the logout button it will redirect back to the login page.

Figure 3.0 Semester Listing Page



No.	Semester	Start Date	End Date	Action
1	2020_C	2020-10-10	2021-01-31	Edit
2	2020_B	2020-06-01	2021-09-30	Edit
3	2020_A	2020-01-01	2020-04-24	Edit
4	2019_C	2019-09-18	2020-01-01	Edit
5	2019_B	2019-05-01	2019-09-03	Edit
6	2019_A	2019-01-31	2019-05-01	Edit
7	2018_C	2018-09-18	2019-01-04	Edit
8	2018_B	2018-04-24	2018-09-01	Edit
9	2018_A	2018-01-14	2018-04-04	Edit

Figure 3.0 show the semester listing page, it will display all the semester and our admin is able to create/edit/or delete semester detail by using the Add/edit/delete button above.

Figure 3.1 Add New Semester function

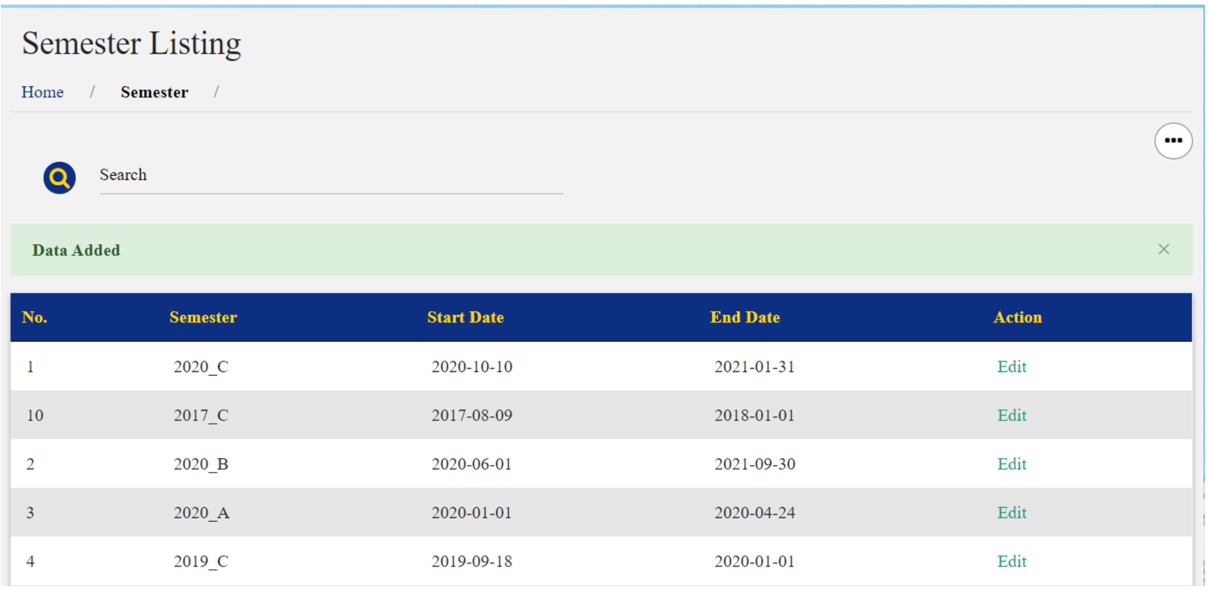


Figure 3.1 show the form for creating the new semester detail. If the all inputs are valid then it will redirect back to semester listing page and display the success message such as Data Added.

Figure 3.2 Edit semester detail function

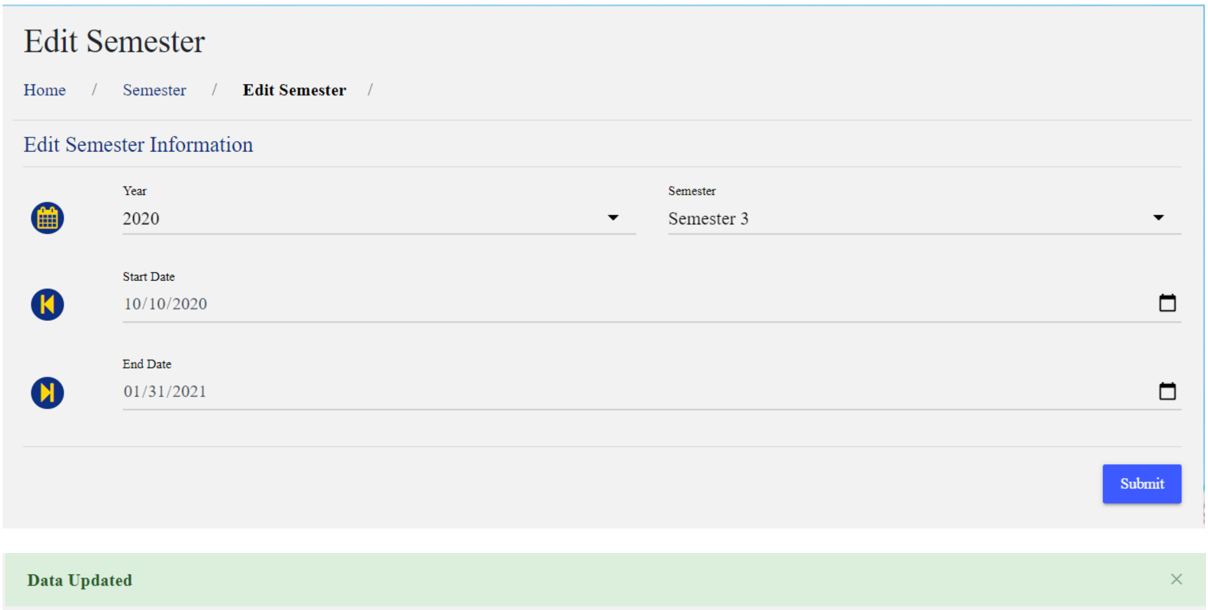


Figure 3.2 shows the edit semester detail function by pressing the edit button and it will take us to the edit semester page with all the current information. After we finish editing it will take back to the semester listing page and the semester will be successfully updated with the latest information.

Figure 3.3 Search Semester function

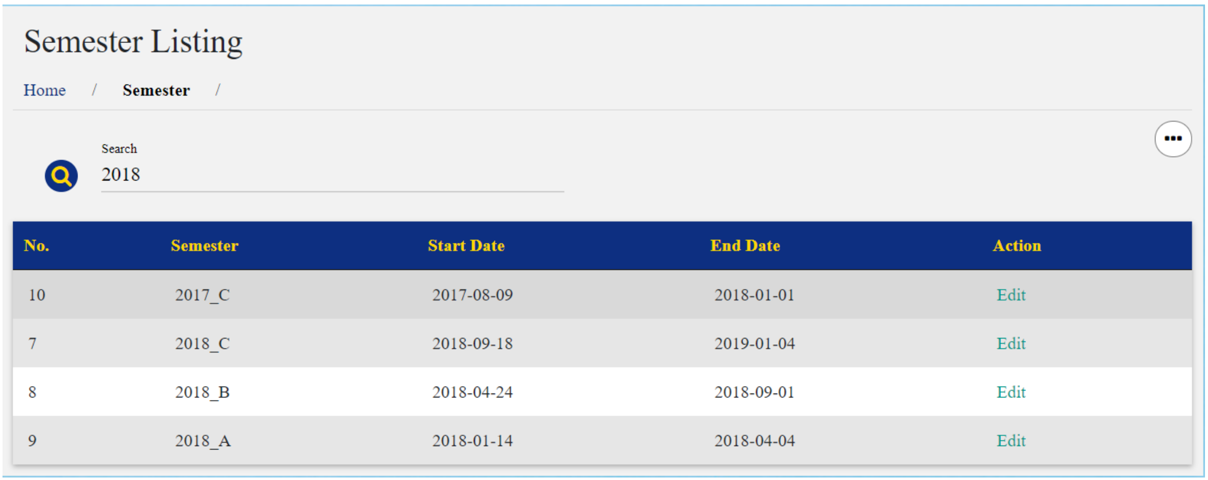


Figure 3.3 shows our search semester function by typing the related that semester detail in the provided search input. After type in, which semester details are similar it will show up for to admin to view.

Figure 4.0 University Data Management function

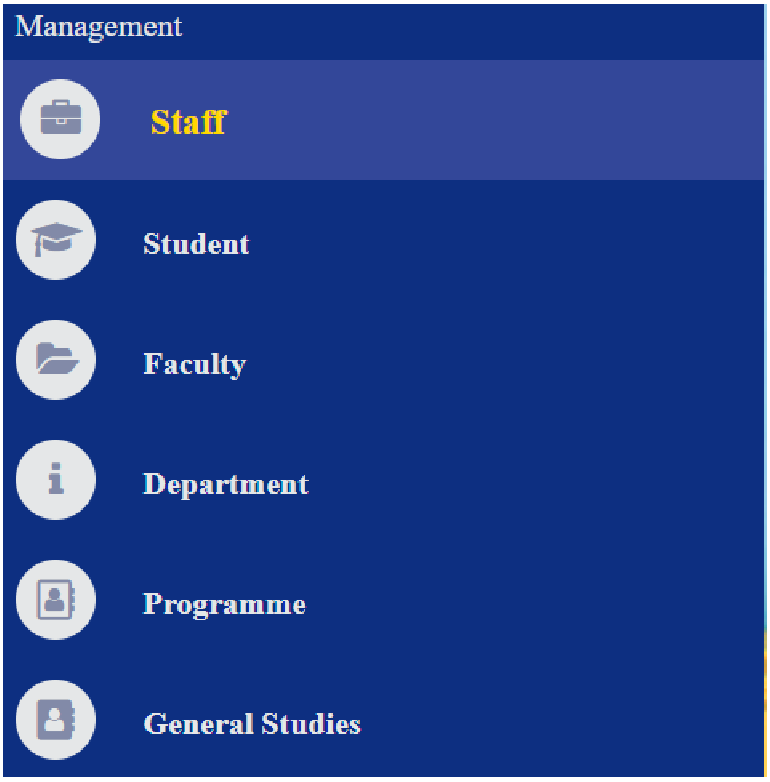


Figure 4.0 is the list of management of university data function, after admin selected which data is need to action, it will display all the selected data in listing and our admin is able to create/edit/or delete the university data detail by using the Add/edit/delete button above.

Figure 4.1 Programme Listing Page

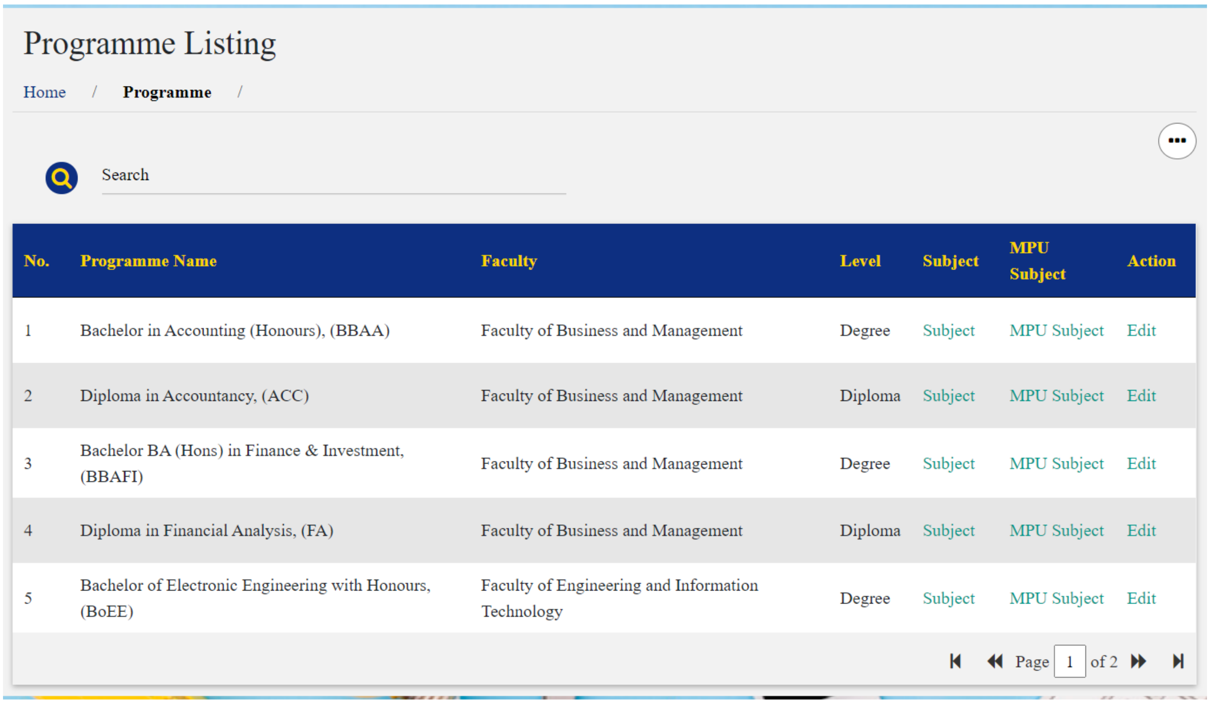


Figure 4.1 is the list of management of programme listing in university data, it will display all the programme in listing and our admin is able to create/edit/or delete the programme detail by using the Add/edit/delete button above. The normal CRUD function is similar with the semester listing.

Figure 4.2 Subject in each programme

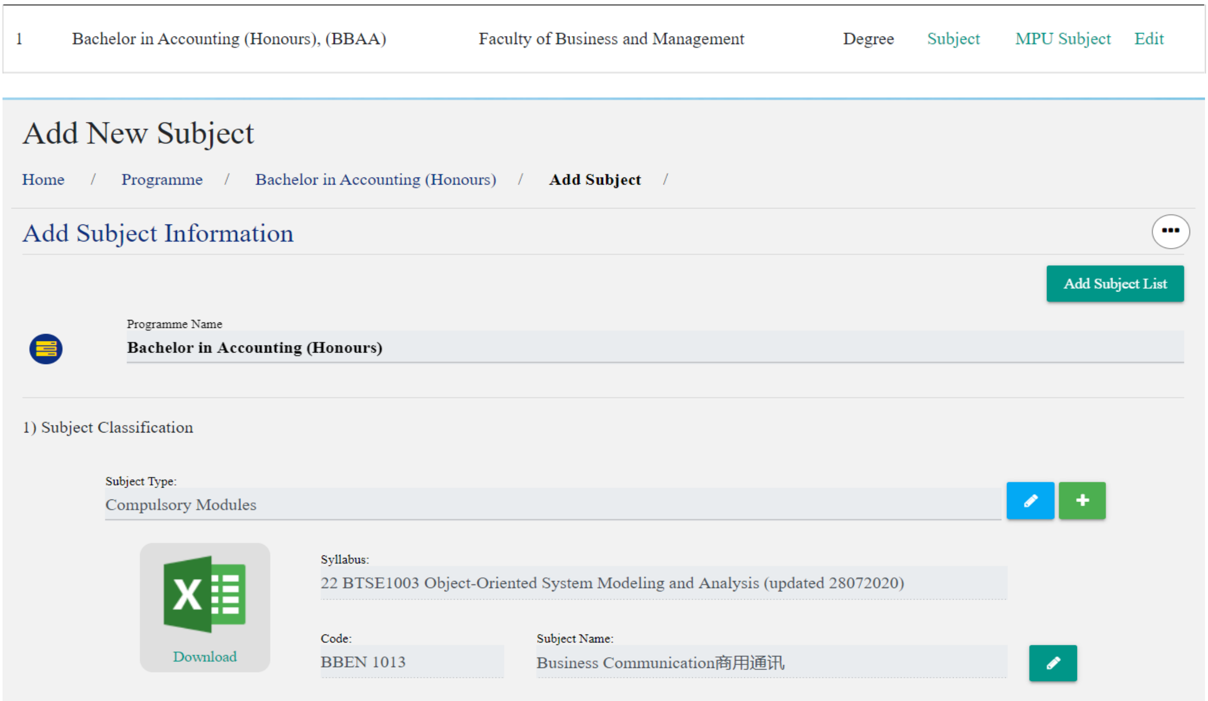
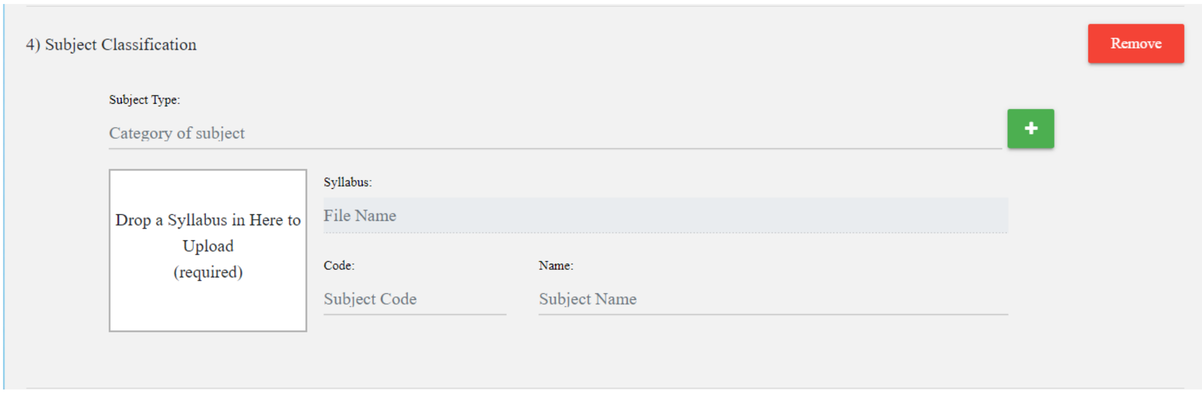


Figure 4.2 is setting the subject in each programme. After press the subject link button, it will redirect to that programme of subject list.

Figure 4.3 Add New Subject



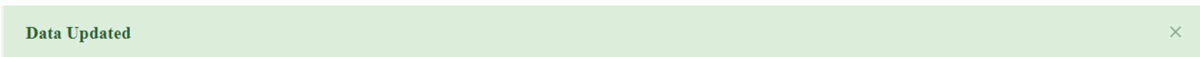


Figure 4.3 is the methods to add subject. If the new subject is in already exists of subject type, just click the + button to add the new subject. If the subject type is new one, just click the add subject list button to add a new subject type and subject. After all input are valid, it will refresh the page and display the message “Data Added”. If got error, it will display the message error.

Figure 4.4 Edit the subject details

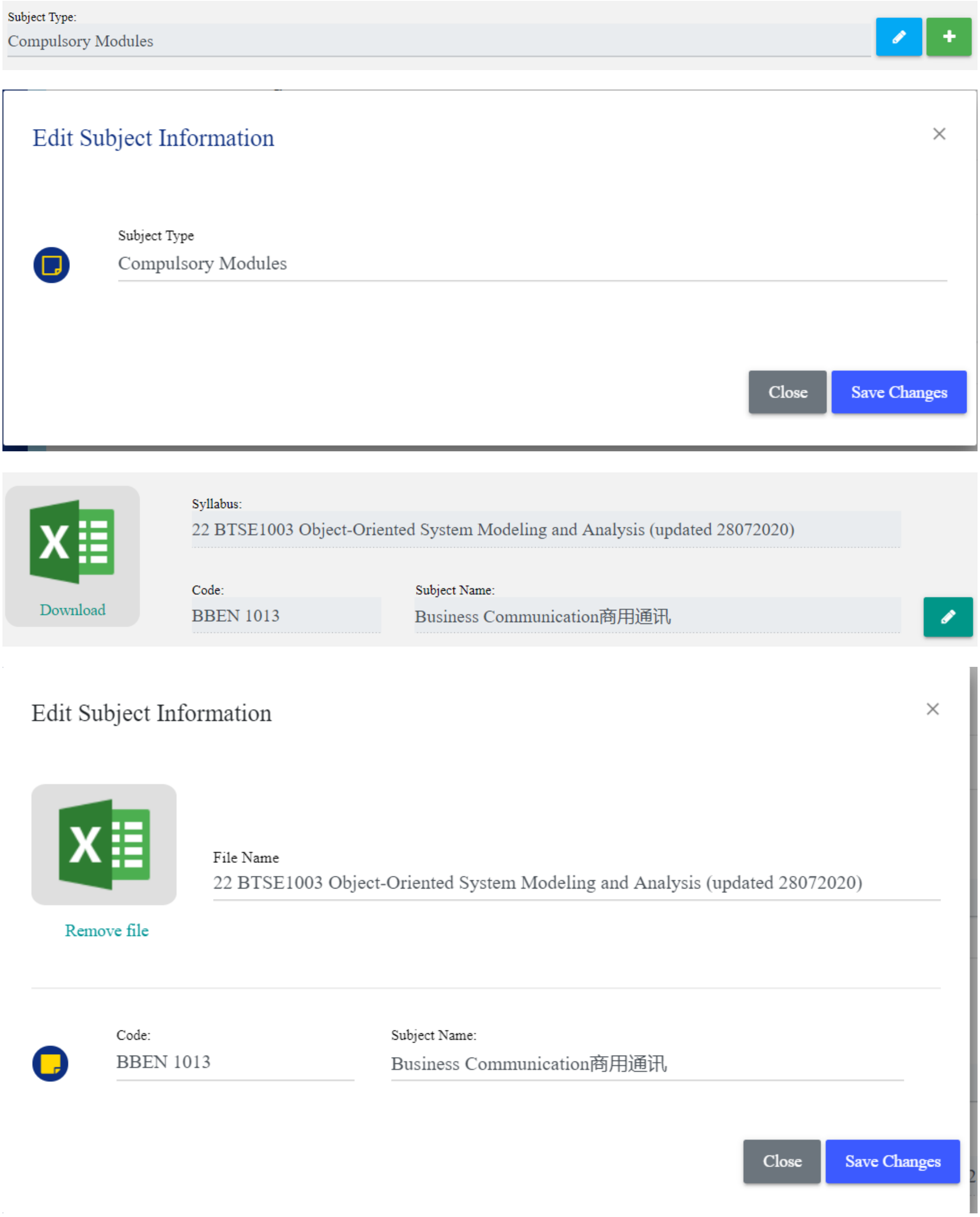


Figure 4.3 is edit the subject details of methods. Admin can click the button icon pencil at the subject type side, it will display a model that let admin to edit the subject type. Admin can click the button icon pencil at the subject list side, it will display a model that let admin to edit the subject details.

Figure 4.5 MPU subject in each programme

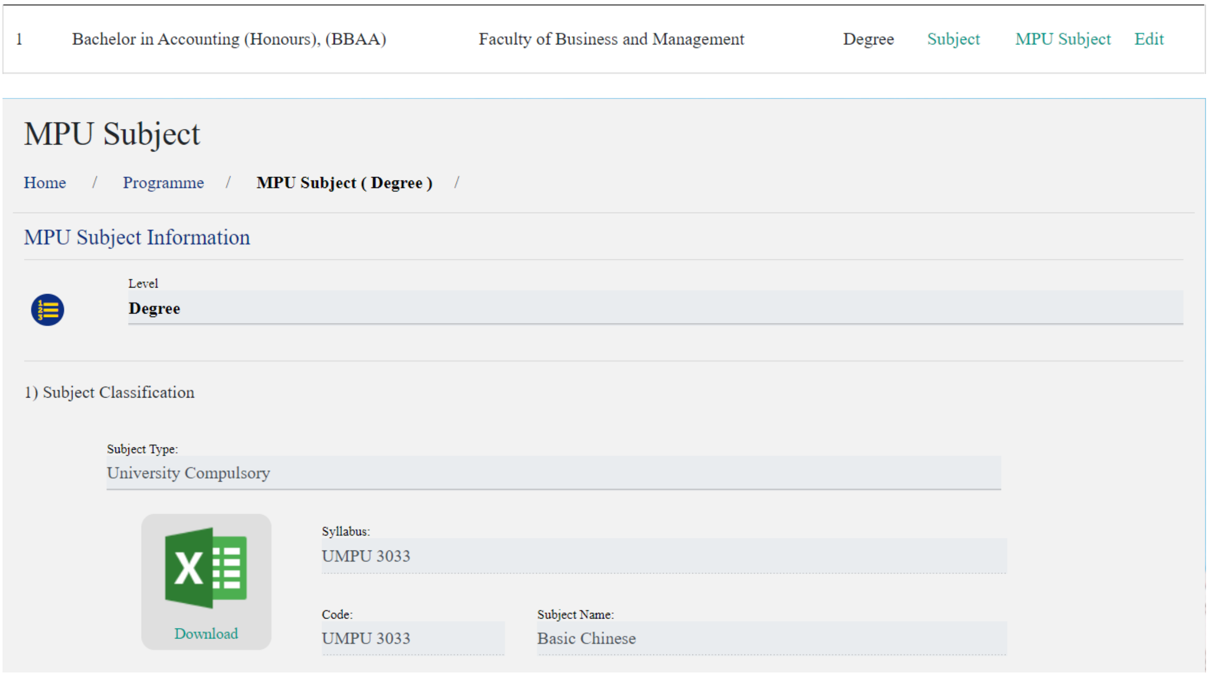


Figure 4.5 is view the MPU subject in each programme. After press the MPU subject link button, it will redirect to that programme of MPU subject list but in this page, admin just can view and download the syllabus only, cannot to add and edit the subject details. Add MPU subject function in the General Studies of university data. All the action is similar with the Add New Subjects.

Figure 5.0 Faculty Portfolio function

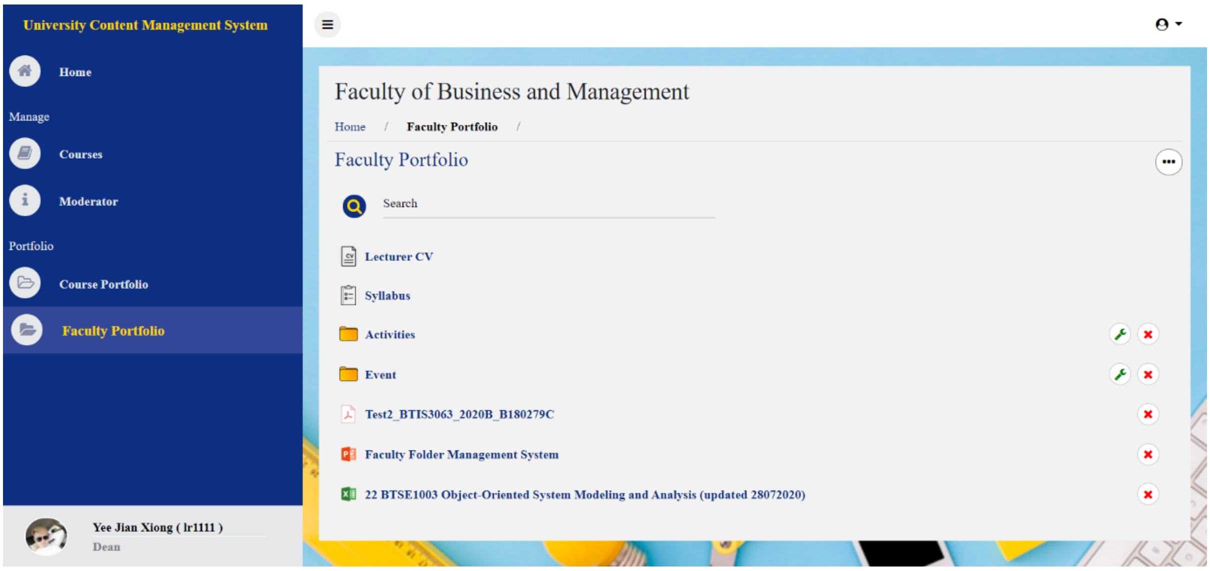


Figure 5.0 is the faculty portfolio function. In first need to login in DEAN panel and click the Faculty Portfolio link button in the sidebar. After that, it will display the faculty of materials.

Figure 5.1 Add New Folder in Faculty Portfolio

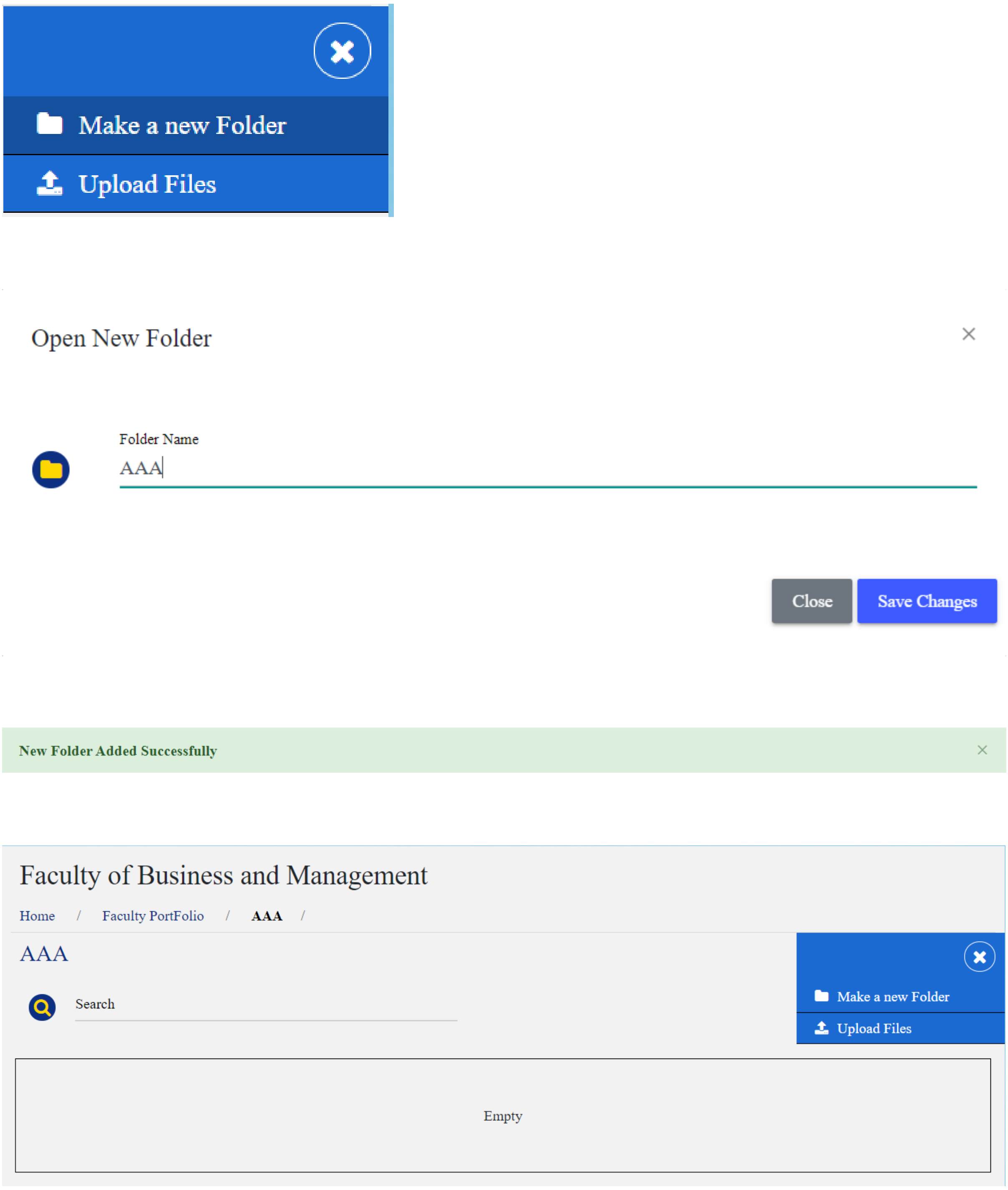


Figure 5.1 is adding the new folder in faculty portfolio. Dean can click the Make a new folder link button, it will display a model that for insert the folder name. After inserted, it will redirect back to faculty portfolio page and show message "New Folder Added Successfully.". After already inserted the folder, Dean can click the folder, it will work in the folder. In that folder, dean can create a new folder again or upload the faculty materials in this folder.

Figure 5.2 Edit Folder Name in Faculty Portfolio

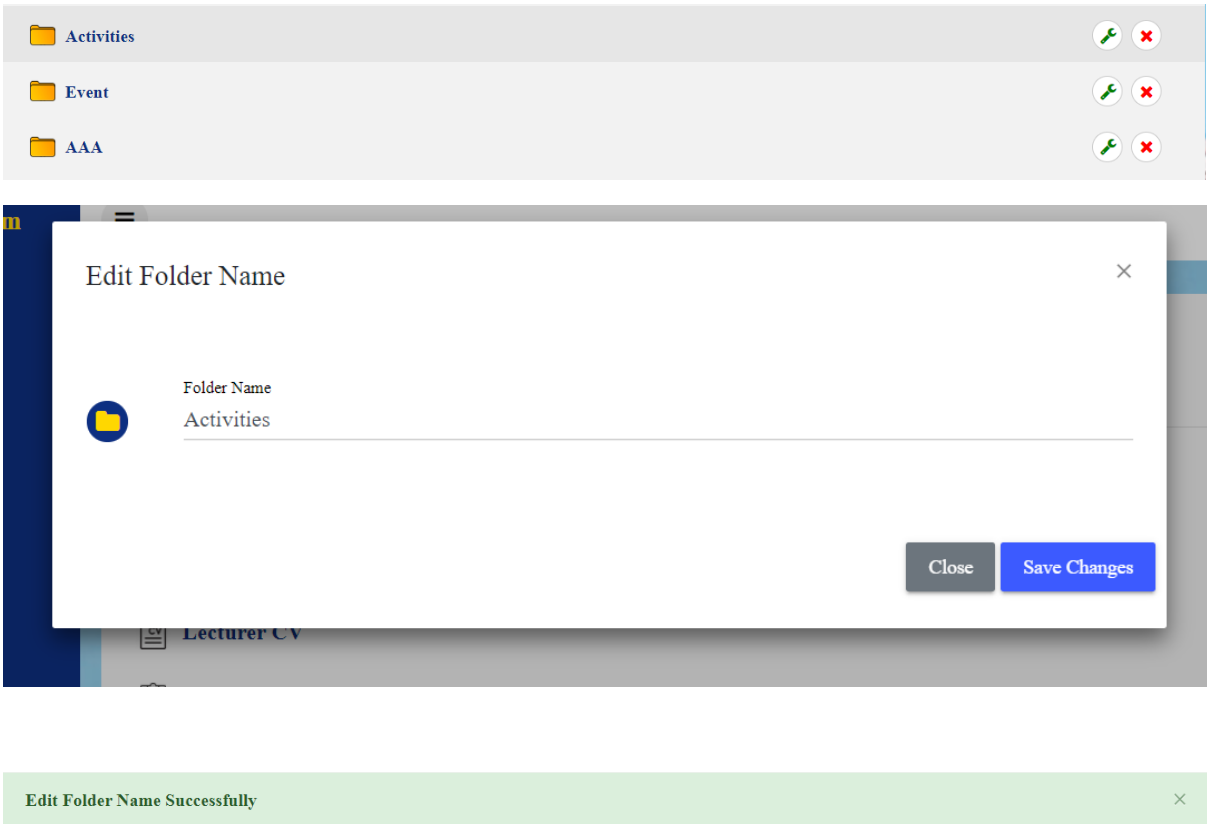
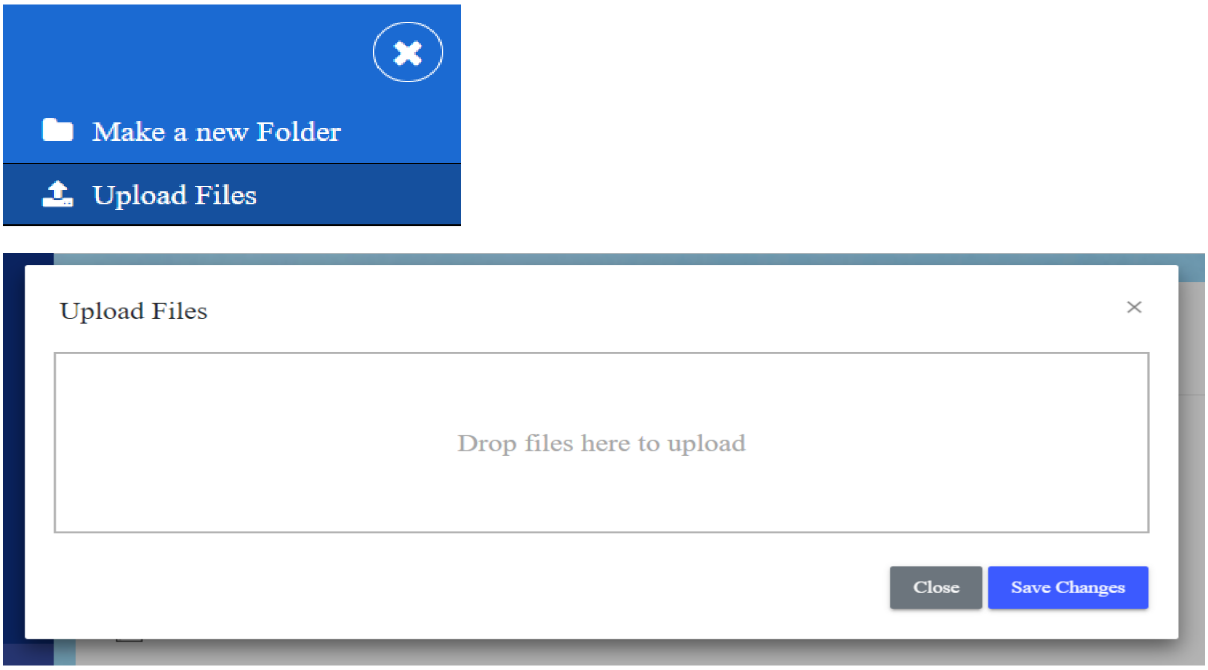


Figure 5.2 is show the edit folder name model. Dean can click the wrench icon button at the right hand side of folder. After that, it will display a model that show out the folder name and can be edit. After we finish editing it will take back to the that page and the folder will be successfully updated with the latest information.

Figure 5.3 Upload Faculty Materials in Faculty Portfolio



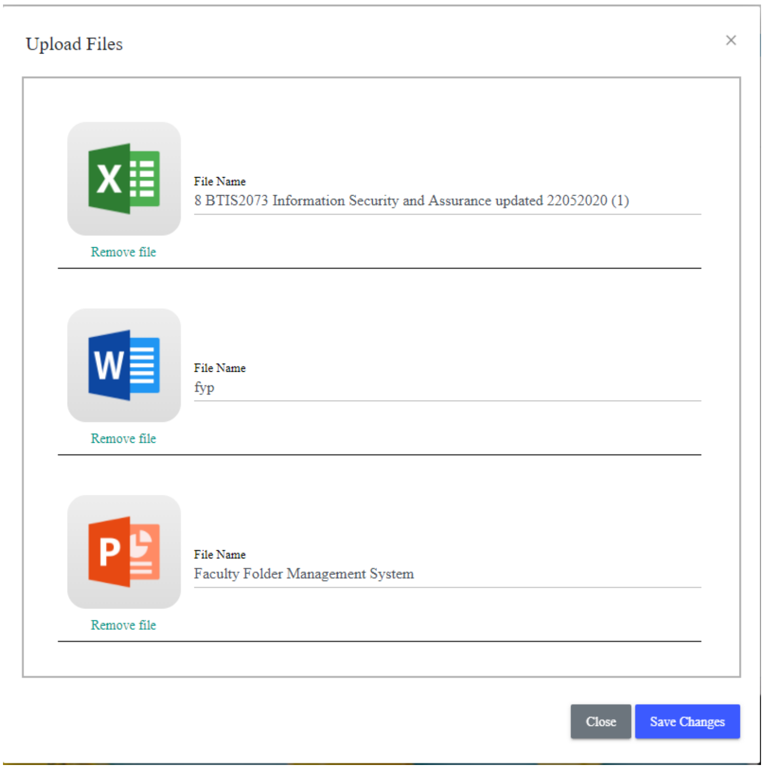
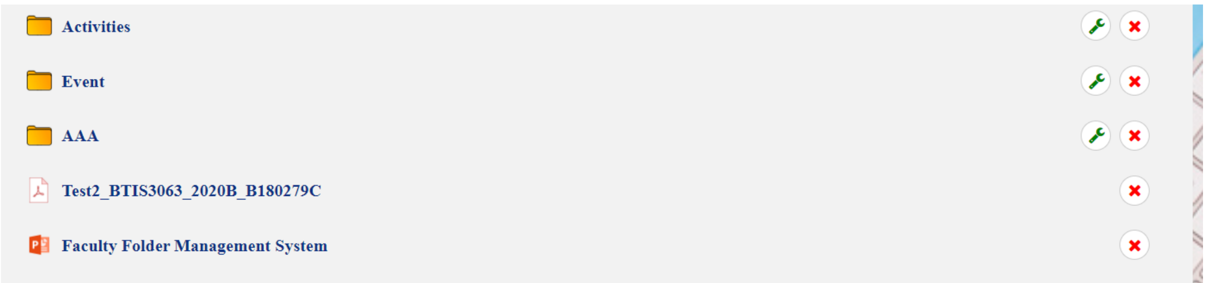


Figure 5.3 is uploading the faculty materials in faculty portfolio. Dean can click the "Upload Files" link button, it will display a model that for upload the materials. After Uploaded, it will redirect back to just now page and show message "New Document Added Successfully". After already inserted the materials, Dean can click the materials to download it.

Figure 5.4 Remove Faculty Materials in Faculty Portfolio



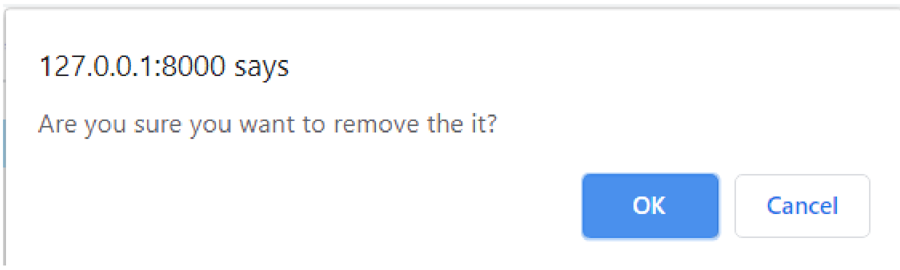


Figure 5.4 is remove the material in faculty portfolio. Dean can click “X” icon button to remove the material. After clicked, it will prop up a confirm input to make sure it. If wanted to remove it just click the OK. If Not, just click the cancel button.

Figure 5.5 View Lecturer CV and syllabus.

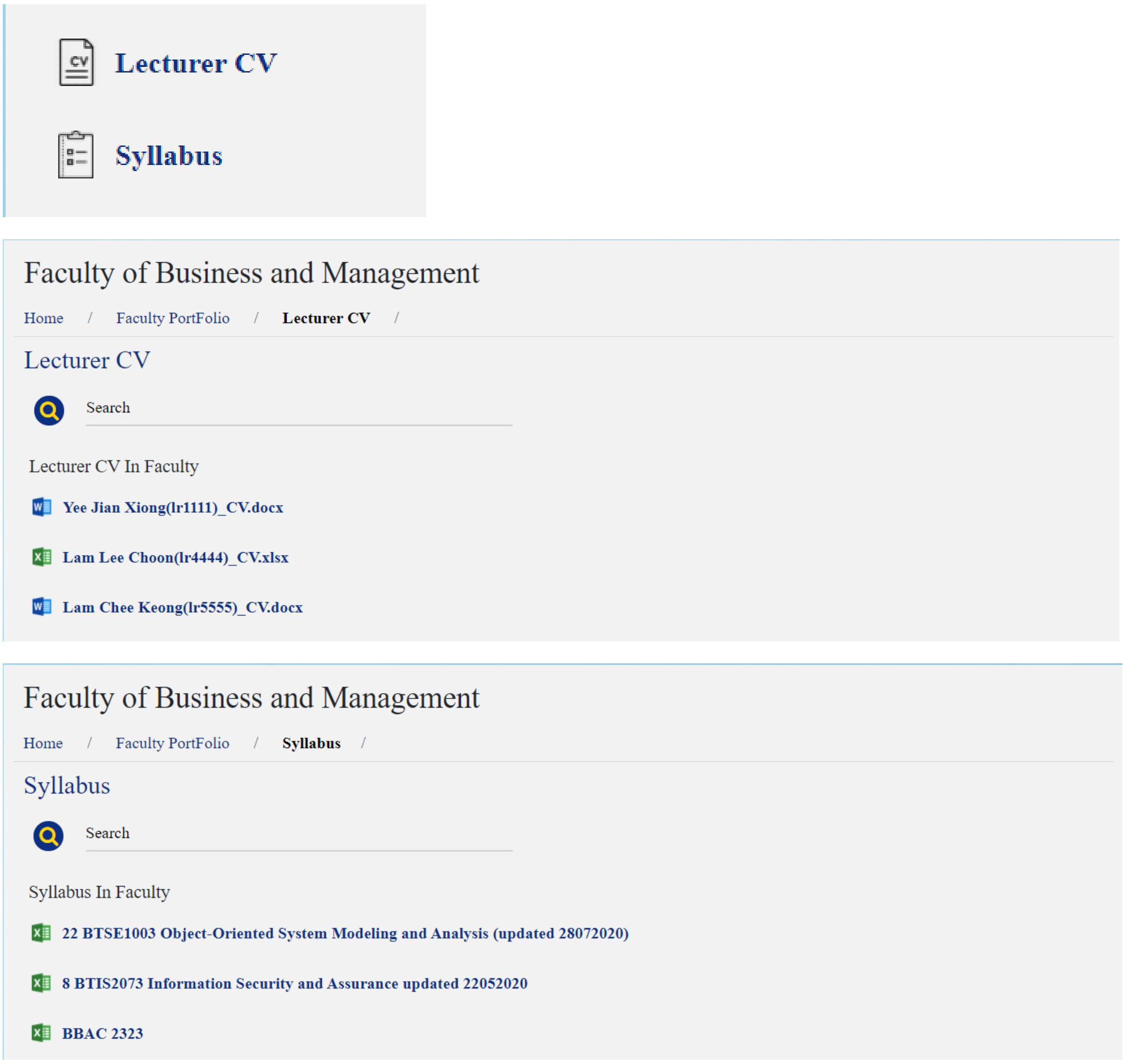
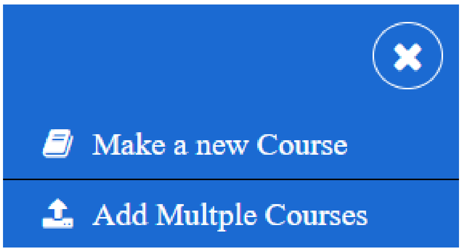


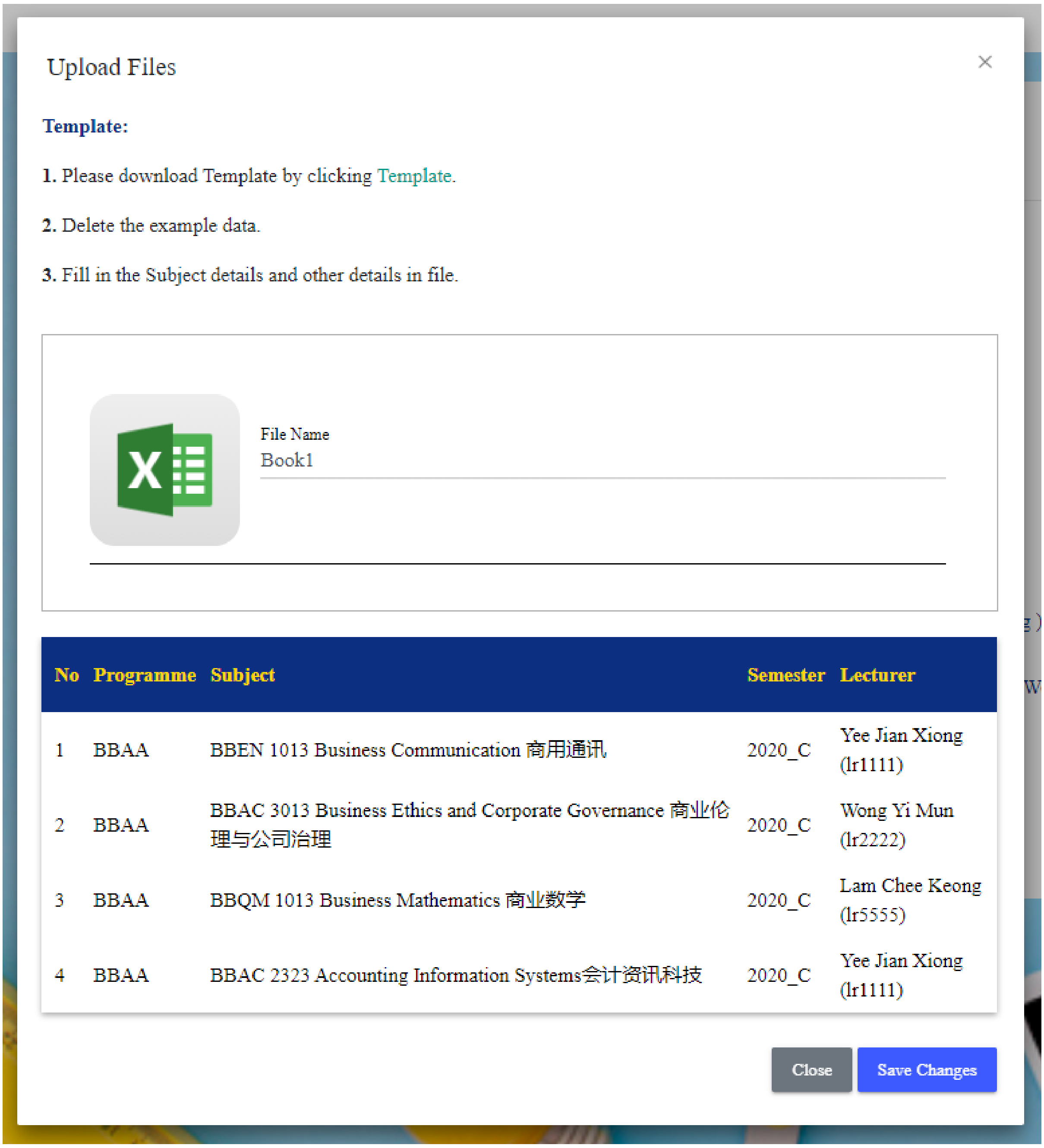
Figure 5.5 is view the lecturer CV and Syllabus in that faculty. Dean can view or download the Lecturer CV or Syllabus in own faculty.

Figure 6.0 Course Portfolio Function

Figure 6.0 is course portfolio listing page. Dean can create/edit/remove the course in this page. It is show out the newest semester of courses in this faculty.

Figure 6.1 Add New Course Portfolio Function





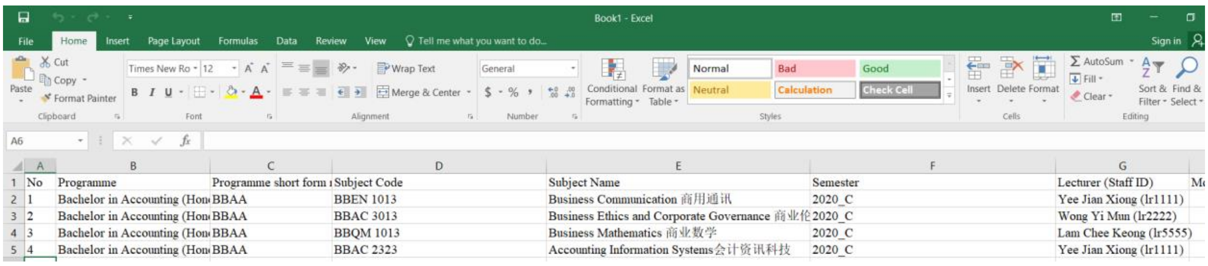


Figure 6.1 is the method of add new course. In First, if add one course the dean can click the "Make a new course", it will redirect to course form page. If add multiple courses the dean can click the "Add Multiple course", it will display a model that let dean to upload the course list. The Course list must follow the template excel file.

Figure 6.2 Edit the course portfolio details function

Figure 6.2 is edit the course details. Dean can click the "wrench" icon button, it will redirect to edit course portfolio details page. After the input are valid and submit, it will redirect back to course portfolio page and show message "Data Updated".

Figure 6.3 Remove the course portfolio function

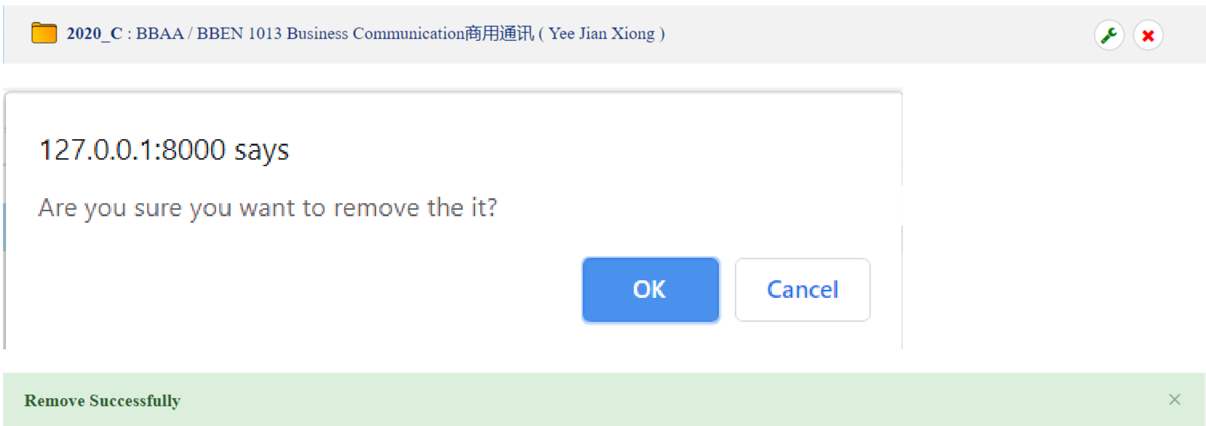


Figure 6.3 is remove the course portfolio. Dean can click "X" icon button to remove the course. After clicked, it will prop up a confirm input to make sure it. If wanted to remove it just click the OK. If Not, just click the cancel button.

Figure 7.0 Courses Function

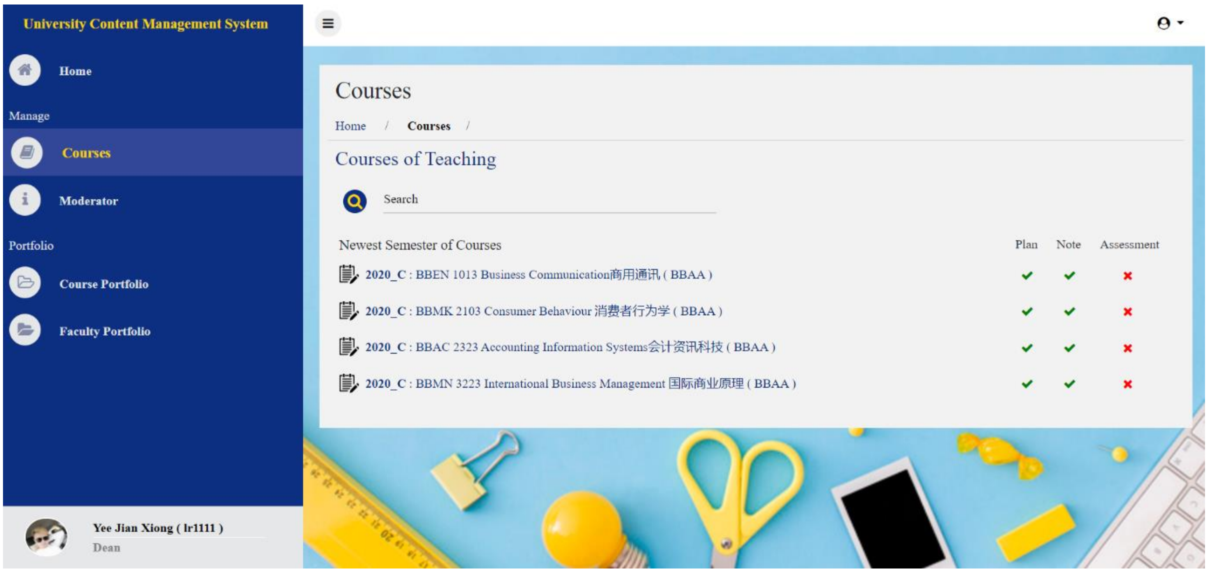


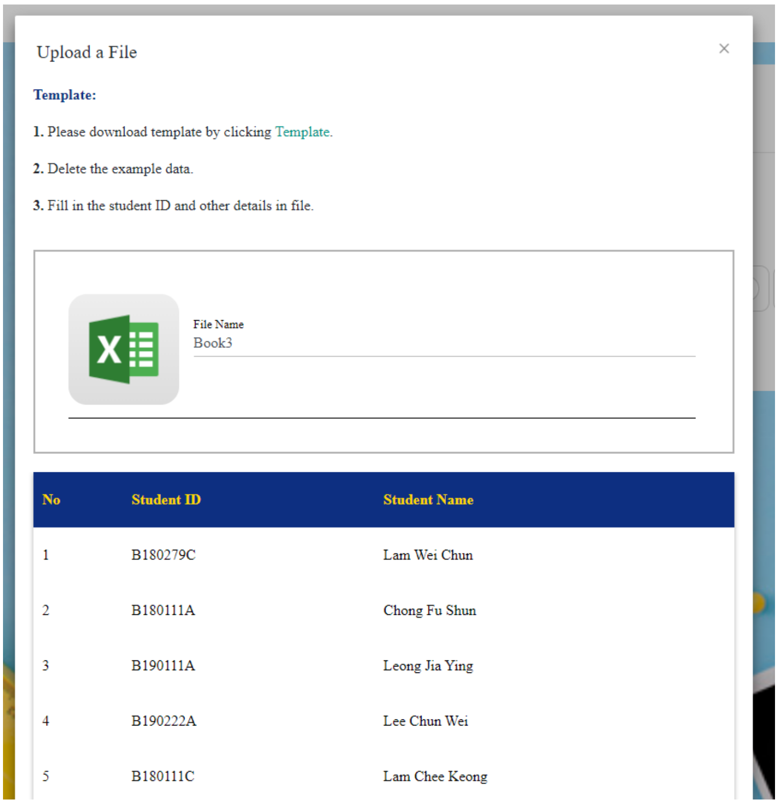
Figure 7.0 is the course list of teaching by the lecturer in newest semester.

Figure 7.1 Course Action Function

Figure 7.1 is the methods of each course. Lecturer can assign the student in course, upload the lecture note, write down the teaching plan.

Figure 7.2 Assign student Function


BBEN 1013 Business Communication商用通讯
Home / Courses / BBEN 1013 Business Communication商用通讯 / Student List /
Student List (4)
Search
Lam Wei Chun (B180279C)
Chong Fu Shun (B180111A)
Leong Jia Ying (B190111A)
Lee Chun Wei (B190222A)


Upload a File
Template:
1. Please download template by clicking Template.
2. Delete the example data.
3. Fill in the student ID and other details in file.
File Name
Book3
No Student ID Student Name
1 B180279C Lam Wei Chun
2 B180111A Chong Fu Shun
3 B190111A Leong Jia Ying
4 B190222A Lee Chun Wei
5 B180111C Lam Chee Keong

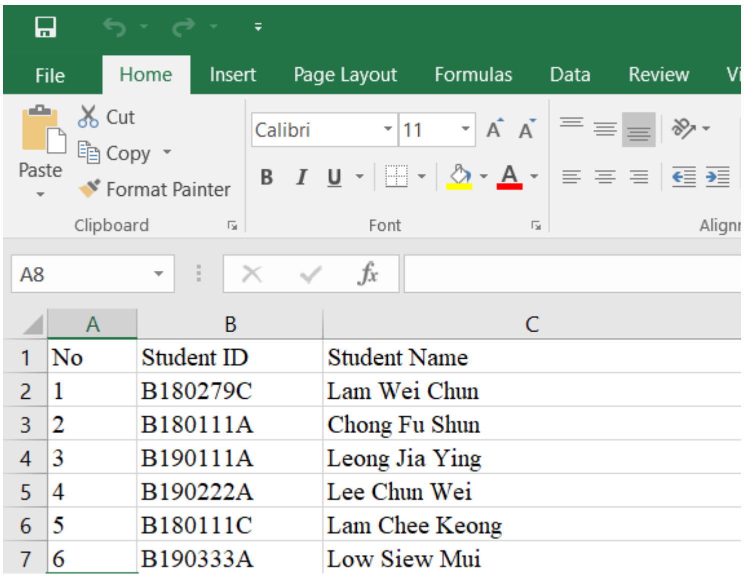

File Home Insert Page Layout Formulas Data Review
Cut
Copy
Format Painter
Paste
Clipboard
Calibri 11
Font
A8
No Student ID Student Name
1 B180279C Lam Wei Chun
2 B180111A Chong Fu Shun
3 B190111A Leong Jia Ying
4 B190222A Lee Chun Wei
5 B180111C Lam Chee Keong
6 B190333A Low Siew Mui

Figure 7.2 is assign student list in course. After click the student list button, it will display the student list in that course. If wanted to assign new student, Lecturer can click the "Upload a File" link button, it will display a model that let lecturer to upload a student list. The student list is follow the template excel file.

Figure 7.3 Remove Assigned Student.

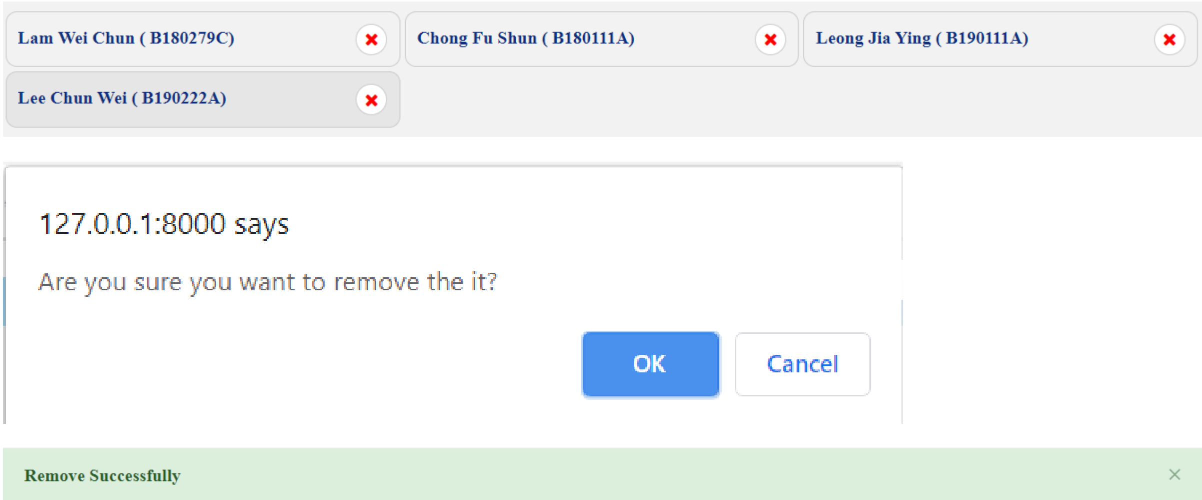


Figure 7.3 is remove the assigned student in course. Lecturer can click "X" icon button to remove the student. After clicked, it will prop up a confirm input to make sure it. If wanted to remove it just click the OK. If Not, just click the cancel button.

Figure 7.4 Lecture Note function

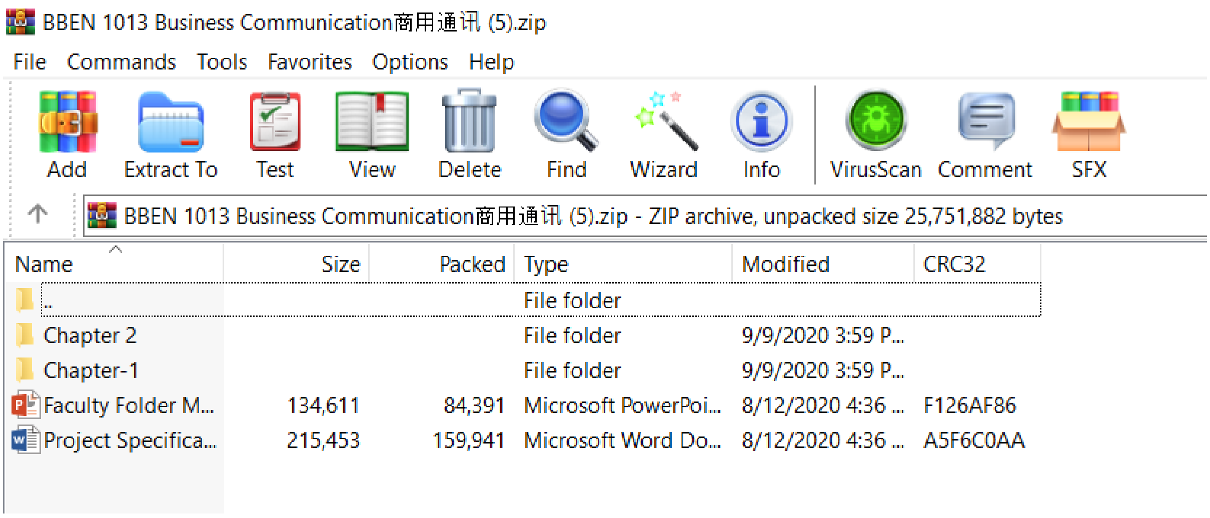


Figure 7.4 is lecturer upload the lecture note function. The function control methods are similar with the **5.0 Faculty Portfolio Function**. But, the lecture note got a more function is can be download all note. The download all note just in the started page of lecture note, if join in other folder will not has the download all note link button.

Figure 7.5 Teaching Plan Function

Figure 7.5 is teaching plan function. Lecturer can write down the teaching plan to student, let students know what can learn in this course.

Figure 7.6 Manage Assessment Method

7.	Course Learning Outcomes (CLO) : At the end of the course the students will be able to: (example) - explain the basic principles of immunisation (C2,PLO1)	
	CLO1	Demonstrate knowledge of system development life cycle and understanding of quality management techniques in the life cycle. (C2, PLO1)
	CLO2	Apply Standard Unified Modeling Language (UML) notation to create requirement artifacts. (C3, PLO6)
	CLO3	Construct an efficient and precise modeling of the software system environment.(C4, PLO5)

9.	Transferable Skills (if applicable) (Skills learned in the course of study which can be useful and utilized in other settings)	1	Knowledge
		2	Communication, leadership and team skills
		3	Scientific methods, critical thinking and problem solving skills.

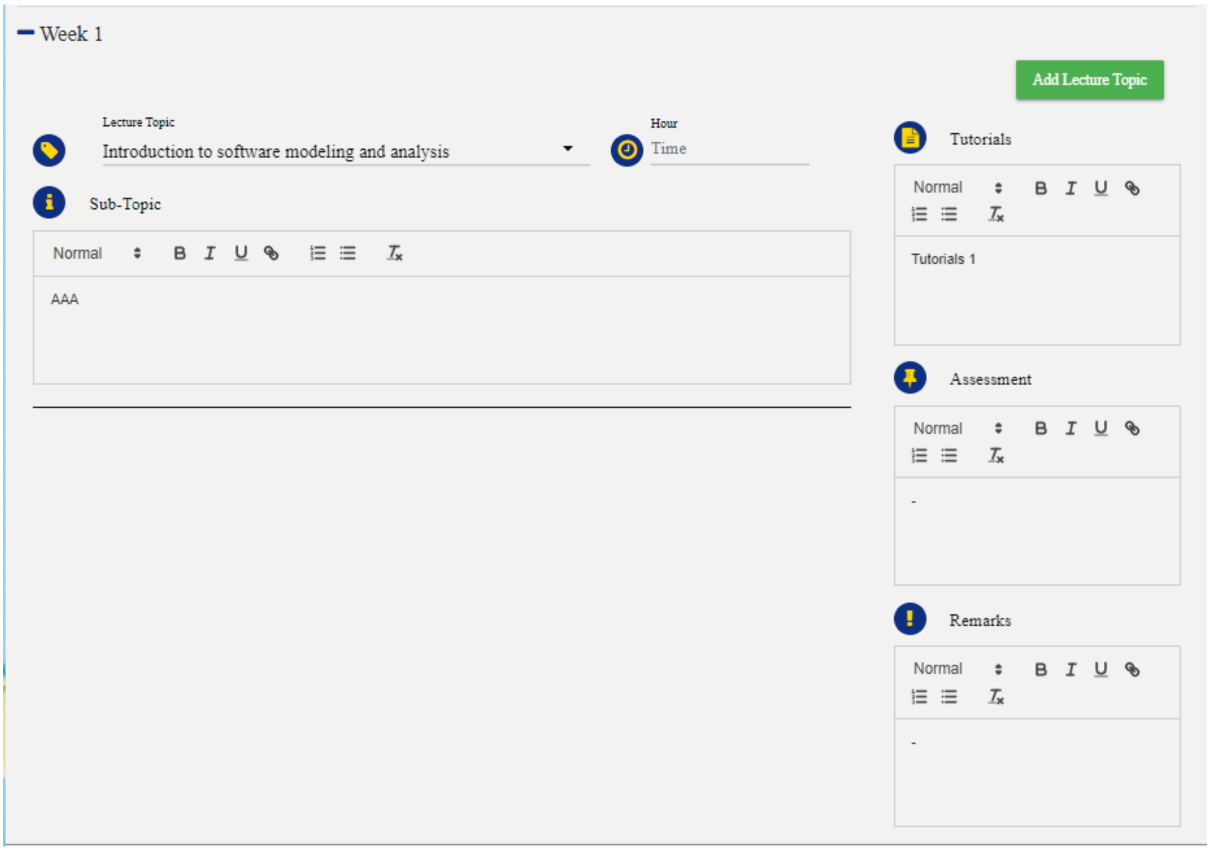
Teaching Planning

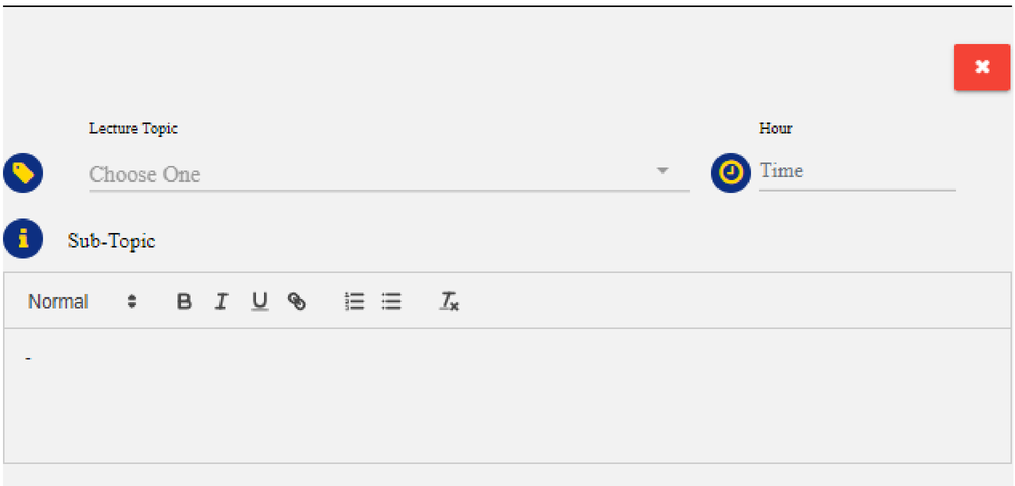
Methods of Assessment

					Assessmnt Methods & Mark Breakdown			
					Assignments	Test	Project	Final Examination
No	Course Outcomes (CO)	Programme Outcomes(PO)	Domain & Taxonomy Level (e.g A2/ C3)	Teaching Methods	15%	25%	20%	40%
1	Demonstrate knowledge of system development life cycle and understanding of quality management techniques in the life cycle.	Scientific methods, critical thinking and problem solving skills.	C2	Lecture	✔	✖	✖	✔
2	Apply Standard Unified Modeling Language (UML) notation to create requirement artifacts.	Knowledge	C3	Tutorial	✔	✖	✖	✔
3	Construct an efficient and precise modeling of the software system environment.	Communication, leadership and team skills	C4		✖	✖	✔	✔

Figure 7.6 is write down the method of assessment. Lecturer need to write down the methods of assessment in each CLO. The CLO Data is get from the syllabus of subject. Each CLO has one PO. Lecturer need to mark down the CLO are related with which assessment. After completed and submitted, it will redirect back to teaching plan page and show out the result of assessment methods.

Figure 7.7 Manage Weekly Plan


Week 1
Add Lecture Topic
Lecture Topic
Introduction to software modeling and analysis
Hour
Time
Tutorials
Normal
Tutorials 1
Sub-Topic
Normal
AAA
Assessment
Normal
-
Remarks
Normal
-


Lecture Topic
Choose One
Hour
Time
Sub-Topic
Normal
-

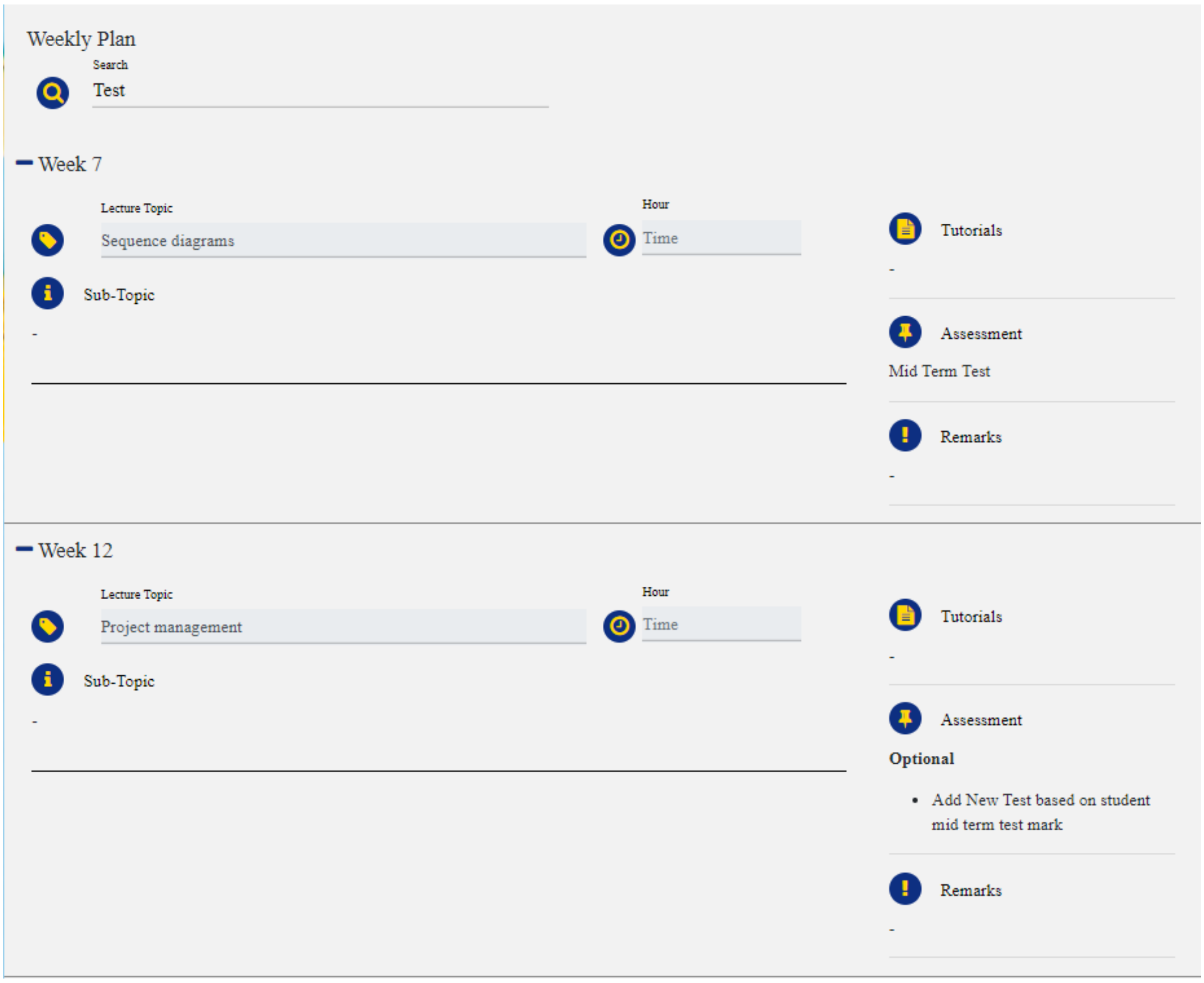


Figure 7.7 is write down the weekly plan. Lecturer need to write down the weekly plan for each week. In each week, lecturer need to select the lecture topic and write down the lecture hour, if got more lecture topic just click the "Add Lecture topic", it will add new input form for other lecture topic. Lecturer also need to write down the tutorials, assessment and remark in each week. After the input are valid and submit, it will redirect back to the teaching plan page. The lecturer can search the plan, that will show up the related plan in each week. For example, lecturer search "Test", that will show up the week 7 and week 12 because this two week have "Test".

APPENDIX C – Project Diaries

Project Diary & Supervisor Feedback-1

Student Name: Lam Wei Chun

Supervisor: Mrs. Sim Hiew Moi

Meeting Date: 1/6/2020

Issues identified in previous meeting (including personal development goals):

This is the first project meeting, so no issue has been identified.

Feedback received in previous meeting:

This is the first project meeting, so no feedback has been identified.

Action taken on feedback:

This is the first project meeting, so no action has been taken.

Matters to discuss:

1. **Project topic**
   - In fact, at the beginning, I thought about making mobile apps, which is a service-based application. I am responsible for finding and joining people who provide services. There are let the customer contact the servicer directly. What services are there, such as repairing daily necessities, decoration, etc. However, later I learned that when Final Year Project members in other groups did something related, I decided to give up the topic first.
2. **Mrs. Sim Hiew Moi of Opinion**
   - Mrs. Sim said that she hoped that we would develop a system for the university and college to reduce the paper documents in each faculty. Next, paper documents are not conducive to searching.

3. **Conclusion**
    - I decided to use the system as my project topic, and decided to discuss with the lecturer next week how to start the system.

References Consulted:

No References Consulted.

Student's Signature: ___________________________________	**Date: ______________**
Supervisor's Signature:_________________________________	**Date: ______________**

Project Diary & Supervisor Feedback-2

Student Name: Lam Wei Chun

Supervisor: Mrs. Sim Hiew Moi

Meeting Date: 8/6/2020

Issues identified in previous meeting (including personal development goals):

- To decide the project topic and collect the information about the college.

Feedback received in previous meeting:

- Get more information for the internet.

Action taken on feedback:

- Get more information for the internet.
- Write down the task for discuss with lecturer.

Matters to discuss:

1. **Discuss the overall project scope**
   - I have prepared the abstract of the system and ready to discuss with Mrs. Sim about my main functions of the system.
2. **Mrs. Sim Opinion**
   - The lecturer's requirement is that paper documents can be converted into electronic documents, which is convenient for lecturers to store and search. The lecturer also gave the BOE system to my friend. These two systems are related. So we can discuss together.
   - At this time, something unfortunate happened. My topic was the same as that of another member of my group. The reason was that when the lecturer gave the topic, we did not choose it immediately, and we did not know that everyone did the same thing. In the end, the lecturer decided that I would do this topic, and another group member would do another system for renting resources from the school.

References Consulted:

No References Consulted.

Student's Signature: ____________________________________	**Date: ______________**
Supervisor's Signature:__________________________________	**Date: ______________**

Project Diary & Supervisor Feedback-3

Student Name: Lam Wei Chun

Supervisor: Mrs. Sim Hiew Moi

Meeting Date: 15/6/2020

Issues identified in previous meeting (including personal development goals):

- To decide using which language to develop the system.
- In first, I decide to use the MEAN language.

Feedback received in previous meeting:

- Using the framework language to develop the system because the framework has the NVC method to manage the coding.

Action taken on feedback:

- I find the MEAN language of tutorials in internet but it is very difficult to understands.
- I also find the Laravel framework of tutorials in internet but it is easier than the MEAN language.

Matters to discuss:

1. **Using which language to develop the system**
    - At the beginning, I said that I might use the MEAN language. However, my friend said that he wanted to use the Laravel framework for development, and then I decided to use Laravel to develop the system together, because if I encountered any problems, I could discuss with each other.
2. **Mrs. Sim Opinion**
    - The Lecturer's suggestion is that we can use the same framework to develop the system, and the reason is the same. We can discuss and solve problems together.
3. **Conclusion**
    - My members and I decided to develop systems using the Laravel framework to develop their own systems.

References Consulted:

No References Consulted.

Student's Signature: ___________________________________	**Date: ______________**
Supervisor's Signature:_________________________________	**Date: ______________**

Project Diary & Supervisor Feedback-4

Student Name: Lam Wei Chun

Supervisor: Mrs. Sim Hiew Moi

Meeting Date: 22/6/2020

Issues identified in previous meeting (including personal development goals):

- I am not sure whether my project scope is correct. Therefore, I will submit Mrs. Sim to review my project scope.

Feedback received in previous meeting:

- No feedback has been identified.

Action taken on feedback:

- No action has been taken.

Matters to discuss:

1. **Project Scope**
    - My project scope is a bit different from the lecturer's idea. The reason is that the system requested by the lecturer is to replace paper documents with electronic documents, thereby helping the faculty to reduce the storage of paper documents. However, my idea is to develop an online teaching system, and storing electronic documents is just one of the sub systems. The reason is that the system is a bit too small if it is only the function that the teacher said.
2. **Conclusion**
    - I need to expand the teacher's idea to make it more complete.

References Consulted:

No References Consulted.

Student's Signature: ________________ **Date:** ________

Supervisor's Signature: ________________ **Date:** ________

Project Diary & Supervisor Feedback-5

Student Name: Lam Wei Chun

Supervisor: Mrs. Sim Hiew Moi

Meeting Date: 29/6/2020

Issues identified in previous meeting (including personal development goals):

- Need to expand the function of the system.
  1. Add some online learning functions, such as lecture note, attendance and assign student function
  2. Understand the needs of Mrs. Sim such as faculty portfolio and course portfolio, etc.

Feedback received in previous meeting:

- Can find about the Content Management System or Document management system.

Action taken on feedback:

- Find about the content management system and document management system for internet.

Matters to discuss:

1. **Project Scope**
   - This time the project scope meets the lecturer's requirements. But the lecturer asked for an extra moderator function, that is to allow other teachers to help review the course materials such as continuous assessment and final exam.
2. **Doing Methods**
   - The lecturer says can doing the system first after that just do the report.
3. **Conclusion**
   - I will decide all the function of the system and name it as University Content Management System.

References Consulted:

No References Consulted.

Student's Signature: ____________________________________	**Date: ______________**
Supervisor's Signature:__________________________________	**Date: ______________**

Project Diary & Supervisor Feedback-6

Student Name: Lam Wei Chun

Supervisor: Mrs. Sim Hiew Moi

Meeting Date: 8/7/2020

Issues identified in previous meeting (including personal development goals):

- No Issues has been identified

Feedback received in previous meeting:

- Doing the system first.

Action taken on feedback:

- Using Laravel framework to develop the system.
- Do the login and CRUD function.

Matters to discuss:

1. **Target user in system**
   - My system has 5 target users: admin, dean, head of department (HOD), lecturers and students. Admin can manage the university data management. Dean and HOD can track the student and lecturer of status performance. Lecturer can upload the course materials to student. Student can view or download the course materials in system.
2. **Normal Management function**
   - The management function got Staff, Student, Faculty, Department, Programme, General Studies.
3. **Conclusion**
   - Mrs. Sim view the system and does not any problem for it.

References Consulted:

No References Consulted.

Student's Signature: ____________________________________ **Date:** ______________

Supervisor's Signature:__________________________________ **Date:** ______________

Project Diary & Supervisor Feedback-7

Student Name: Lam Wei Chun

Supervisor: Mrs. Sim Hiew Moi

Meeting Date: 15/7/2020

Issues identified in previous meeting (including personal development goals):

- No Issues has been identified.
- Doing the faculty portfolio function.

Feedback received in previous meeting:

- No feedback has been identified.

Action taken on feedback:

- No action has been taken.

Matters to discuss:

1. **Faculty Portfolio function**
    - The faculty portfolio function is like the window File explorer, that can create the folder and put the faculty materials in each folder.
    - The type of materials are docs, xlsx, and pdf only.
    - Before upload the materials can rename it.
2. **Conclusion**

- Mrs. Sim view the system and does not any problem for it but the type of materials need to add the pptx and image.

References Consulted:

No References Consulted.

Student's Signature: ______________________________ **Date:** ____________

Supervisor's Signature:______________________________ **Date:** ____________

Project Diary & Supervisor Feedback-8

Student Name: Lam Wei Chun

Supervisor: Mrs. Sim Hiew Moi

Meeting Date: 22/7/2020

Issues identified in previous meeting (including personal development goals):

- The type of materials need to add the pptx and image. In add the pptx is not problem but the image got the display problem such as I no sure the user is want to display the image or download the image.
- Doing the course portfolio function
- Doing the course action function

Feedback received in previous meeting:

- No feedback has been identified.

Action taken on feedback:

- No action has been taken.

Matters to discuss:

1. **Course Portfolio Function**
   - Mrs. Sim says in this function is need to multiple insert such as using the excel import function to insert the course.
   - After the course portfolio inserted new course, that will assign to each lecturer of course modules.
2. **Course Action Function**
   - In Course Action that has the assign student function. In that function, I create a form to assign student that is follow by Batch. That mean user can select the batch of student to assign in course. But Mrs. Sim says sometimes not all student in the batch got the course in same semester. However, Mrs. Sim says no need to remove the form because it can be using in single assign student. Need to add a new multiple insert similar with course portfolio function.

References Consulted:

No References Consulted.

Student's Signature: ___________________________________ Date: _______________

Supervisor's Signature:___________________________________ Date: _______________

Project Diary & Supervisor Feedback-9

Student Name: Lam Wei Chun

Supervisor: Mrs. Sim Hiew Moi

Meeting Date: 5/8/2020

Issues identified in previous meeting (including personal development goals):

- Need to Create the multiple insert in course portfolio function and assign student function.

Feedback received in previous meeting:

- Using the excel import methods to do the multiple insert.

Action taken on feedback:

- View the tutorials of excel import about the Laravel.
- Implement the excel import into the system.

Matters to discuss:

1. **Course Action Function**
   - Mrs. Sim said that the teaching plan has two parts, one is assessment method and second is weekly plan. This two parts of simple data is getting from the course of syllabus. For example, assessment method will get the CLO, PO in syllabus. Weekly Plan will get the course content outline. Mrs. Sim also gave me some example of syllabus. But, that has a question in assessment method because the simple data is getting from syllabus, if the syllabus is changed. The assessment method will get wrong in edit part. Mrs. Sim suggest me not need to do the edit function, just make in reinsert again.

References Consulted:

No References Consulted.

Student's Signature: ______________________ **Date:** __________

Supervisor's Signature:______________________ **Date:** __________

Project Diary & Supervisor Feedback-10

Student Name: Lam Wei Chun

Supervisor: Mrs. Sim Hiew Moi

Meeting Date: 12/8/2020

Issues identified in previous meeting (including personal development goals):

- Edit function in assessment method.

Feedback received in previous meeting:

- Not need to do edit function, just reinsert again.

Action taken on feedback:

- Create a link for create an assessment method, every times user need to edit assessment methods, they need to reinsert again the method and markdown the assessment in each CLO.

Matters to discuss:

In this week, there is no discussion for my project and Mrs. Sim suggest us want to start to do our Final Report.

References Consulted:

No References Consulted.

Student's Signature: ______________________ **Date:** __________

Supervisor's Signature:______________________ **Date:** __________