FTU Online Attendance Management For Lecturer and Officer System

Mr. Nurdeen Cheleng 581431005

Mr. Abdulloh Labu-api 581431023

Email: nurdeen23518@gmail.com, abdulloh.labuapi@gmail.com

(Fausan Mapa)
Advisor Mr. Fausan Mapa

Faculty of Science and Technology
Fatoni University

CHAPTER 1
INTRODUCTION

1. PROJECT OVERVIEW

Fatoni University is located in Pattani province, Southern of Thailand. Inside university there are five faculties, named Faculty of Islamic Studies, Faculty of Liberal Art and Social Science, Faculty of Science and Technology, Faculty of Education and Faculty of Quran and Sunnah. In each faculty, there are many departments and offices.

The number of lecturer and officer is around 450 people so the personnel division sector have to collect the data of those people in daily to easily calculate their salary but the current system is still use manual to organize and check even for those who need to leave, they have to submit paper leave form.

So FTU Online Attendance Management For Lecturer and Officer System is web application developed for daily lecturer and officer attendance in university. It facilitates to access the attendance information of a particular lecturer and officer in a particular department and section. This system will also help in evaluating attendance eligibility criteria of a lecturer and officer. By just a click on the mouse, the system will be able to produce the lecturers and officers' attendance report thus reducing the need for manual labour which is prone to human errors and time consuming.

This web application is built for automating the processing of attendance. It also enhances the speed of performing attendance task easily. The lecturer and officer have unique user login id and password available. The lecturer and officer can submit online leave form and view the attendance record on whole year basis. The administrator can view as well as modify the attendance record and print attendance report.

2. PROBLEM STATEMENT

Nowaday, the attendance system of Fatoni University uses manually to organize and check the status of lecturers and officers to calculate salary for them until we decided to solve the current problem and develop a new attendance system.

We can divide the current problem from 2 kind of user:

- From The personnel division sector (Admin)
    - The current system still be stand alone system.
    - The current system generate report does not meet the needs.

- From lecturers and officers
    - When they want to take a leave, they still use paper to fill information and submit to admin.
    - Unable to track their status of leaving or absent.

.

3. OBLECTIVE

- To develop the online attendance management system.
- To develop attendance report for easily to calculate salary.
- To develop the online leave forms for lecturers and staff.
- To improve the performance of the system.
- To follow the FTU concept of paperless.

4. SIGNIFICANCE OF STUDY

- Apply our knowledge to develop the FTU Online Attendance Management For Lectures and Staff System.
- Easy for administrators to manage data and make report.
- Reduce the complexity in checking and tracking status.
- Users can track their status.
- Users can access the system anywhere that has internet access.

5. SCOPE AND LIMITATION

The scope of this project is developing the FTU Online Attendance Management For Lectures and Staff System. It is base on web application. This system will provide to three kind of users:

- Administrators can manage FTU lecturers and staff attendance, manage users, make report, see user's status.
- Head of department can approve online leave form requested from lecturers and staff and see their status.
- Lecturers and Staff can request online leave form and tracking status.

However, this project will be available in Thai and English.

6. SOFTWARE AND HARDWARE REQUIREMENT

Software	**Specification**
Virtual Studio Code	V12.0.3
Adobe Photoshop	CC 2017
MySQL	5.7
Microsoft office	2019
Window 10	-

Hardware	**Specification**
ASUS	● Model ASUS X401A ● CPU Intel Core i3-3110M (2.40GHz, 3MB L3 Cache) ● GPU Intel HD Graphics 4000 ● Size 14 inch WXGA (1366x768) LED ● Memory Size 4 GB DDR3 ● Window 10 Operating System
Lenovo Y700	● CPU: 2.6GHz Intel Core i7-6700HQ (quad-core, 6MB cache, up to 3.5GHz with Turbo Boost) ● Graphics: Nvidia GeForce GTX 960M (4GB DDR5 VRAM), Intel HD Graphics 530. ● RAM: 16GB DDR4 2133MHz. ● Screen: 15.6-inch, 1,920 x 1,080 LED anti-glare back-lit multi-touch display. ● Storage: 225 GB SSD, 1TB HDD (5,400 RPM)

CHAPTER 2
LITERATURE REVIEW

REVIEW OF RELATED WORKS

Web Based Student Information Management

S.R.Bharamagoudar et al.,this paper assist in automating the existing manual system. It can be monitored and controlled remotely. This paper provides accurate information always. All years together gathered information can be saved and can be accessed at any time. The purpose is to design a college website which contains up to date information of the college. That should improve efficiency of college record management.

Attendance Management System

G.Gangagowri et al., this system is used Way to SMS software. This software is used to send SMS easily to their parent's. This system can store their data about the students and those cares absent student details. It is an efficient method to store the attendance in the Web Site rather than wasting the paper. It also updates the student report

Online Student Attendance System

P. N. Garad et al, in this project, we gave access to three users i.e. Admin, Student, Others. This project is base on client-server. Here, the serve is Tomcat and client is JSP. In this project teachers or the admin will be filling attendance and sending message to the student who is absent. They will have privilege to fill attendance form, update attendance form, send message to the guardian's account whose child is absent, also those attendance is less than 75%, and they also have privilege to send message to the students whose fees are pending. he staff can also view the message whenever they want and also can modify the details of students. Parents have privilege to view attendance and to view message sent by the teacher.

Students also have their account with the privilege to view message sent by the subject teacher and to view the attendance.

Classroom Attendance Application

Pranjul Khare1 et al, the scope of the project is the system on which the software is installed, i.e. the project is developed as an ANDROID application, and it will work for a particular institute. Android is a mobile operating system (OS) based on the Linux kernel and currently developed by Google with a user interface based on direct manipulation. RAD approaches to software development have put less emphasis on planning tasks and more emphasis on development. It has revealed that an online system for recording and reporting students' attendances is indeed a needed application in order to make the process more efficient and time-saving where more than 70% of the sample group agreed to that matter.

CHAPTER 3
METHODOLOGY

This chapter describes the systems Development Life Cycle (SDLC). It is important phases that are essential for developers, such as planning, analysis, and design. The life cycle can be thought of as a circular process (as shown in figure 3.1) in which the end of useful life of one system leads to the beginning of another project that will develop a new version or replace an existing system all together. SDLC is divided into 5 phases are planning phase, analysis phase, designing phase, implementation phase and maintenance phase.

The set of System Development Life Cycle (SDLC) activity are shown in Figure below.

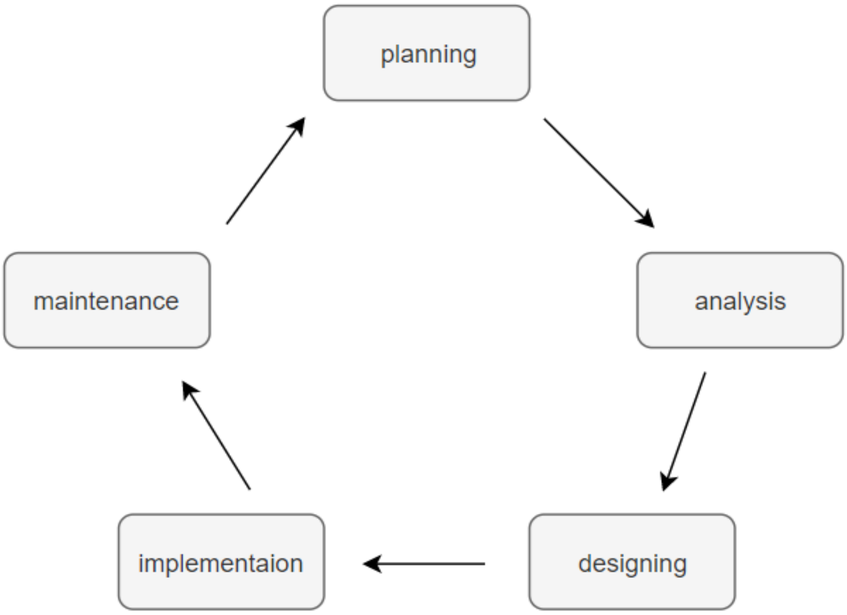


Figure 3.1: System Development Life Cycle (SDLC)

3.I PLANNING PHASE

Planning phase is the first phase of the System Development Life Cycle (SDLC) is a techniques or project management. Planning is preparing a sequence of action steps to achieve some specific goal. The purpose of this phase is to identify the nature and scope or problem of project.

The purpose of this phase, developers try to make understanding the topic to the project and what are the problems and the needs of FTU Online Attendance Management For Lectures and Officer System, and developers also try to identify the opportunities in which situation that developers can improve the project. After that developers make summarization of the information which had been obtained to estimate the scope of this project. Finally, developers document the result to have a feasibility report which state the problems of FTU Online Attendance Management For Lectures and Officer System and summarization of the objective.

3.2 ANALYSIS PHASE

In this phase, the software development process, the software's overall structure and it's defined, A software development model is thus created. Analysis and design are very crucial in the whole development cycle. Any step in the design phase could be very expensive to solve in the later stage of the software development. The logical system of the product is developed in this phase.

The goal of system analysis is to determine where the problem is in attempt to fix the system. In this phase we are going to collect requirement and then analyze these requirements then design

UML model for the system. We use Unified Modeling Language (UML) to analysis requirement because UML is a standardized general-purpose modeling language in the field of software engineering. UML includes a set of graphical notation techniques to create abstract models of specific systems.

(A) LIST OF REQUIREMENTS

M - mandatory requirements (what the system must do)
D - desirable requirements (what the system preferably should do)
0 - optional requirements (what the system may do)

No.	Requirement ID	Requirement Description	Priority
Req_01	Login		
	Req_01.1	Administrator can login to the system by ID and password	M
	Req_01.2	Head of department can login to the system by ID and password	M
	Req_01.3	Lecturer can login to the system by ID and password	M
	Req_01.4	Officer can login to the system by ID and password	M
Req_02	Manage department		
	Req_02.1	Administrator can add new department	M
	Req_02.2	Administrator can update department	O
	Req_02.3	Administrator can delete department	O
	Req_02.4	Administrator can view department	D
Req_03	Manage head of department		
	Req_03.1	Administrator can add new head	M
	Req_03.2	Administrator can update head	O

	Req_03.3	Administrator can delete head	O
	Req_03.4	Administrator can view head	D
Req_04	Manage lecturer and officer		
	Req_04.1	Administrator can add new lecturer and officer	M
	Req_04.2	Administrator can update lecturer and officer	O
	Req_04.3	Administrator can delete lecturer and officer	O
	Req_04.4	Administrator can view lecturer and officer	D
Req_05	Manage attendance and leave		
	Req_05.1	Administrator can view attendance and leave	M
	Req_05.2	Administrator can make report	M
	Req_05.3	Administrator can print report	M
	Req_05.4	Administrator can search user	D
	Req_05.5	Administrator can store leave require	M
Req_06	Manage leave require		
	Req_06.1	Head can approve require	M
	Req_06.2	Head can view status	O
	Req_06.3	Head can search user	O
	Req_06.4	Head can get notify for new require	M

	Req_06.5	Head can reply require	O
Req_07	User performance		
	Req_07.1	Head, lecturer and officer can submit leave	M
	Req_07.2	Head, lecturer and officer can view status	M
	Req_07.3	Head, lecturer and officer can reply	O

Table 3.2: List of requirement

(B) USE CASE DIAGRAM

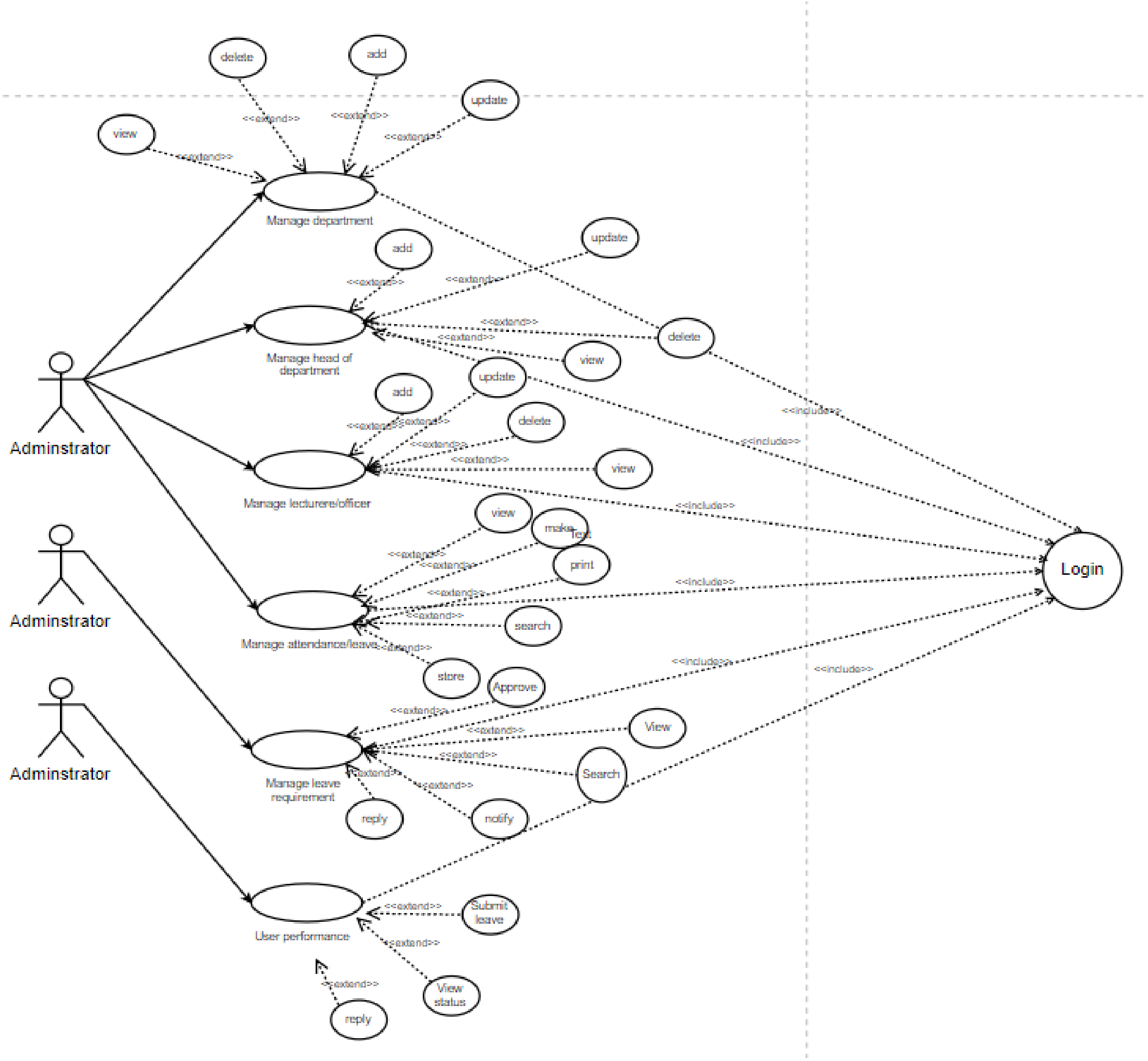


Figure 3.3: Admin and user's management whole task and login

(C) USE CASE SPECIFICATION

Use case specification is to capture the specific detail of the use case. Use case specification provides away to capture the functional requirement of the system.

1. Ues case: Login (Req_01)

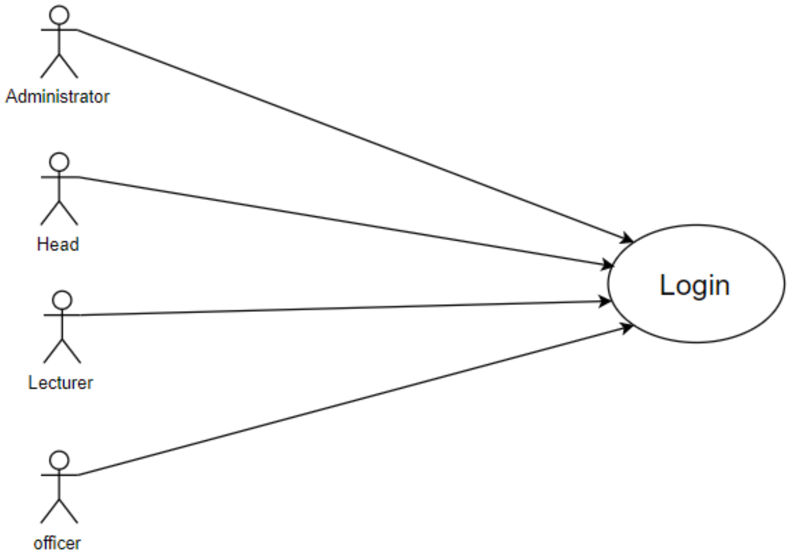


Figure ◌3.4 : The user’s login

Login

1) **Brief description**
This use case will be used to allow users (administrator, head, lecturer and officer) to enter the system.

2) **Pre condition**
The users must have user ID and password to perform this task.

3) **Characteristic of activation**
Execution dependent on user’s command.

4) **Flow of control**

4.1) Basic flow

- This use case begins when user key in user ID and password on the user login page.
- Users press login button. [A-1: Cancel]
- The system shall validate the user ID and password. [E-1: Invalid ID and password]
- This use case end when the system displays the control panel.

4.2) Alternative flow

A-1: Cancel

This system shall cancel the login process.

4.3) Exeption flow

E-1: Invalid ID and password

This system shall display error message to the user.

5) Post condition

The users can enter the system

2. Use case : Manage department (Req_02)

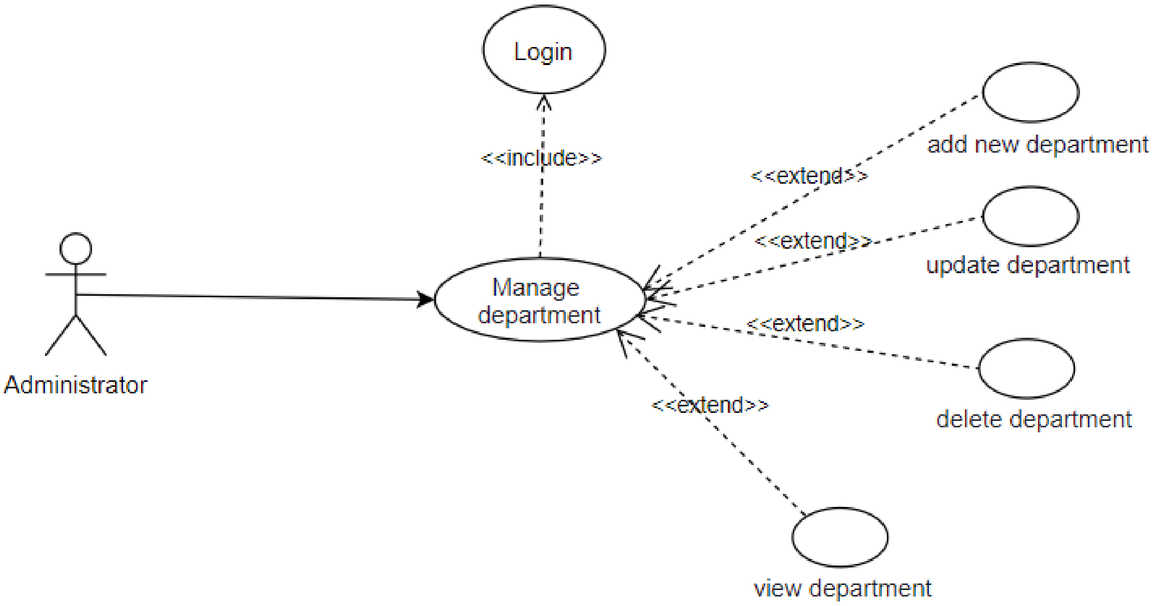


Figure 3.5 : Administrator manage department

Add department

1) **Brief description**

   This use case will be used to allow users (administrator) to manage department such as add new department, update department, delete department and view department.

2) **Pre condition**

   The users must login at the user's login page.

3) **Characteristic of activation**

   Execution depent on user's command.

4) **Flow of control**

   4.1) Basic flow

   - Users select manage department on the control panel.

- Users select delete department. [A-1: Add new department], [A-2: Update department], [A-3: View department]
- The system displays the delete department to allow the admin to delete department.
- Users select particular department.
- Users press OK button. [A-4: Cancel]
- The system shall validate the entering data. [E-1: Invalid entering data]
- This use case end when the system displays acknowledge message that has been deleted.

4.2) Alternative flow

A-1: Add new department

This system shall add new department.

A-2: Update department

This system shall update department.

A-3: View department

This system shall view department.

A-4: Cancel

This system shall cancel the delete department process.

4.3) Exeption flow

E-1: Invalid entering data

The system will display error message and admin have to reenter data.

5) Post condition

The users can manage department information

3. Ues case: Manage head of department (Req_03)

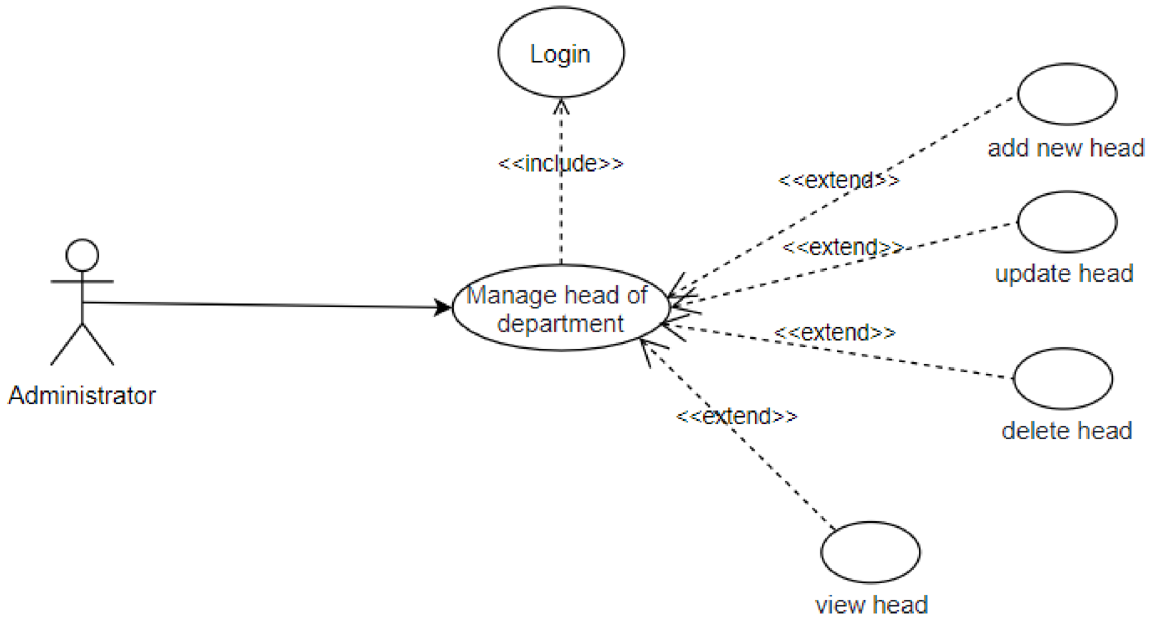


Figure 3.6 : Administrator manage head of depaartment

Add head of department

1) **Brief description**

   This use case will be used to allow users (administrator) to manage head of department such as add new head of department, update head of department, delete head of department and view head of department.

2) **Precondition**

   The users must login at the user's login page.

3) **Characteristic of activation**

   Execution depent on user's command.

4) **Flow of control**

   4.1) Basic flow

   - Users select manage head of department on the control panel.

- Users select add head of department. [A-1: Update department], [A-2: Delete department], [A-3: View department]
- The system displays the add head of department to allow the admin to add head of department information.
- Users key in the head of department information.
- Users press OK button. [A-4: Cancel]
- The system shall validate the entering data. [E-1: Invalid entering data]
- This use case end when the system displays acknowledge message that has been added.

4.2) Alternative flow

A-1: Update head of department

This system shall update head of department.

A-2: Delete head of department

This system shall delete head of department.

A-3: View head of department

This system shall view head of department.

A-4: Cancel

This system shall cancel add head of department process.

4.3) Exeption flow

E-1: Invalid entering data

The system will display error message and admin have to reenter data.

5) Post condition

The users can manage head of department information.

4. Ues case: Manage lecturer and officer (Req_04)

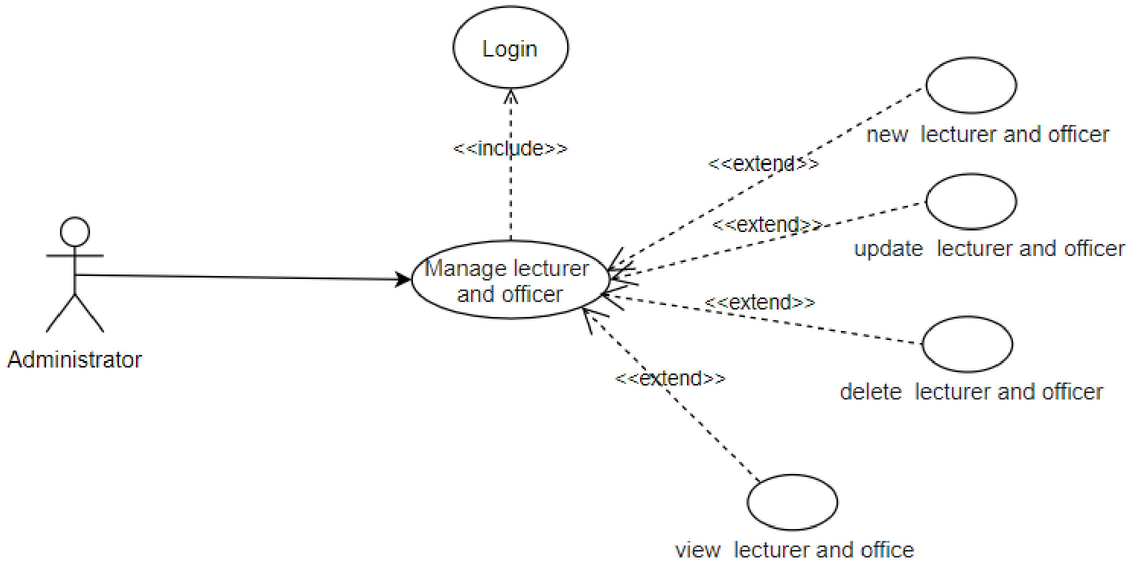


Figure ͦ3.7 : Administrator manage Lecturer and officer

Add lecturer and officer

1) **Brief description**

   This use case will be used to allow users (administrator) to manage lecturer and officer such as add new lecturer and officer, update lecturer and officer, delete lecturer and officer and view lecturer and officer.

2) **Precondition**

   The users must login at the user's login page.

3) **Characteristic of activation**

   Execution depent on user's command.

4) **Flow of control**

   4.1) Basic flow

   - Users select manage lecturer and officer on the control panel.

- Users select add lecturer and officer. [A-1: Update lecturer and officer], [A-2: Delete lecturer and officer], [A-3: View lecturer and officer]
- The system displays the add lecturer and officer to allow the admin to add lecturer and officer information.
- Users key in the lecturer and officer information.
- Users press OK button. [A-4: Cancel]
- The system shall validate the entering data. [E-1: Invalid entering data]
- This use case end when the system displays acknowledge message that has been added.

4.2) Alternative flow

A-1: Update lecturer and officer.

This system shall update lecturer and officer.

A-2: Delete lecturer and officer.

This system shall delete lecturer and officer.

A-3: View lecturer and officer.

This system shall view lecturer and officer.

A-4: Cancel

This system shall cancel add lecturer and officer process.

4.3) Exeption flow

E-1: Invalid entering data

The system will display error message and admin have to reenter data.

5) Post condition

The users can manage lecturer and officer information.

5. Use case: Manage attendance and leave (Req_05)

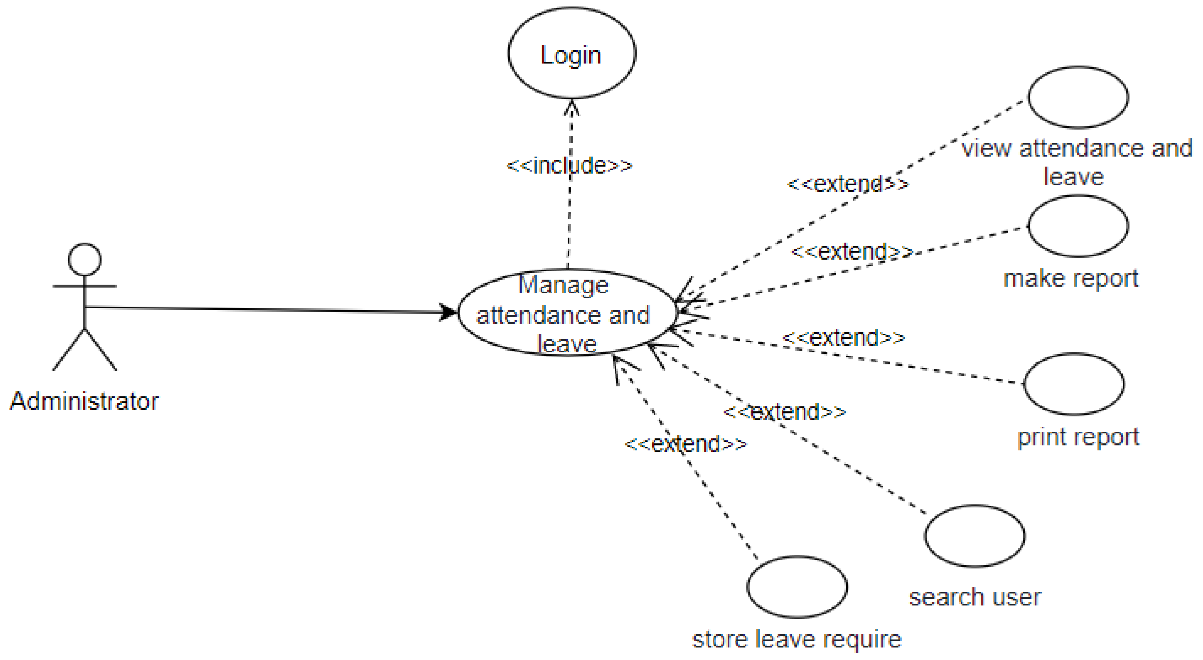


Figure 3.8 : Administrator manage attendance and leave

Manage attendance and leave

1) **Brief description**

   This use case will be used to allow users (administrator) to view attendance and leave, make and print report, search user and store leave required.

2) **Precondition**

   The users must login at the user's login page.

3) **Characteristic of activation**

   Execution dependent on user's command.

4) **Flow of control**

   4.1) Basic flow

   - Users select manage attendance and leave on the control panel.
   - The system displays the dashboard of attendance and leave information.

- Users can make report of attendance and leave information.
- Users can print report of attendance and leave information.
- Users can search attendance and leave of lecturer and officer information.

4.2) Alternative flow

A-1: Submit leave form

This system shall have online form for lecturer and officer.

4.3) Exeption flow

E-1: Invalid entering data

The system will display error message and admin have to reenter data.

5) Post condition

The users can manage attendance and leave information.

6. Use case: Manage leave require (Req_06)

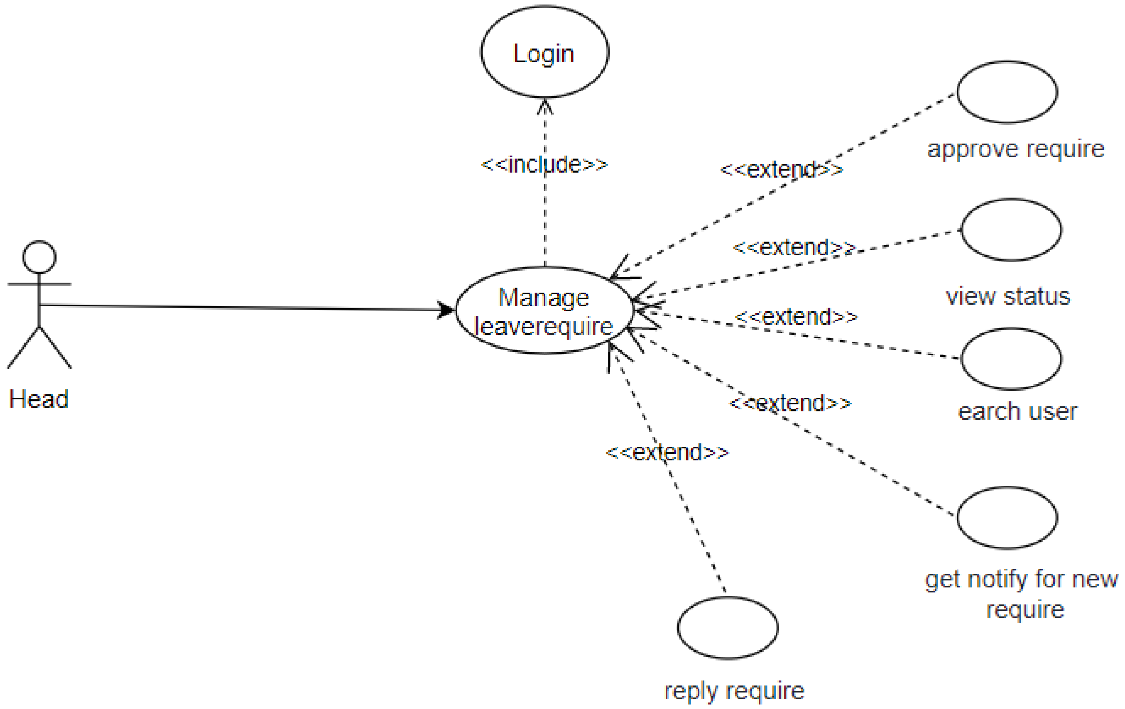


Figure 3.9 : Head of department manage leave require

Manage leave require

1) **Brief description**

   This use case will be used to allow users (Head of department) to manage leave requirement such as view lecturer and officer status, approve leave require, search, get notification of new require and reply to submit.

2) **Precondition**

   The users must login at the user's login page.

3) **Characteristic of activation**

   Execution depent on user's command.

4) **Flow of control**

4.1) Basic flow

- Users select manage leave require.
- The system displays the status of lecturer and officer information.
- Users can approve leave require.
- Users will get notification for new leave require.

4.2) Alternative flow

A-1: Reply to lecturer and officer

This system shall reply back to lecturer and officer to complete and re-submit leave form.

4.3) Exeption flow

E-1: Invalid entering data

The system will display error message and user have to reenter data.

5) **Post condition**

The users can manage leave require information.

7. Use case: User performance (Req_07)

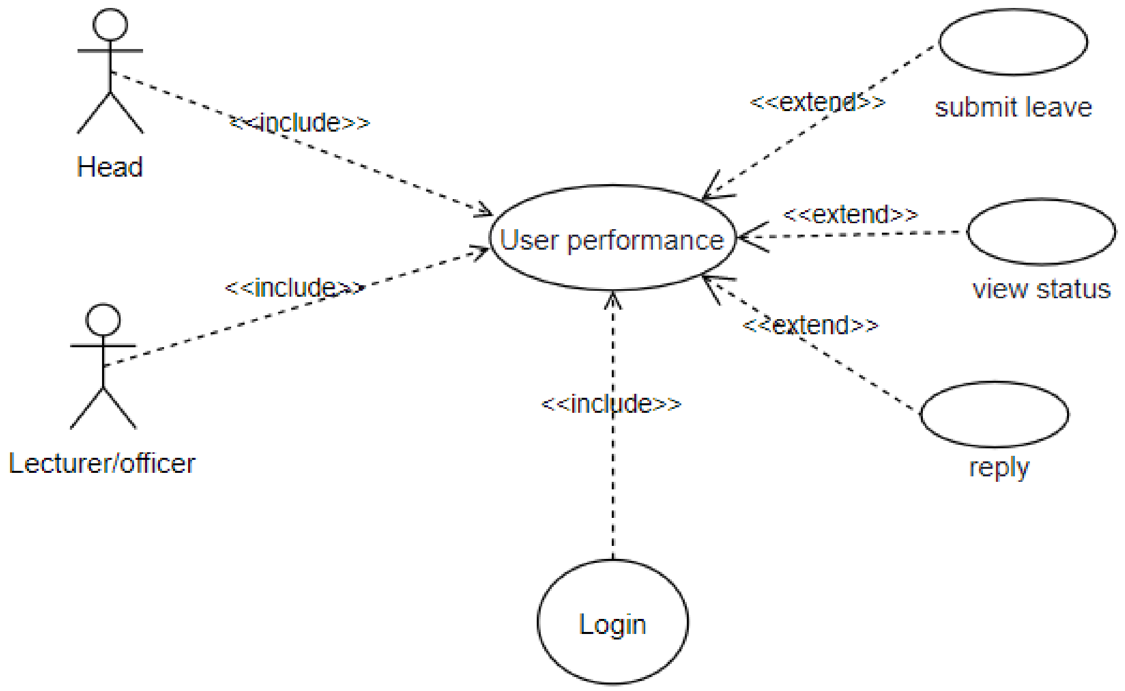


Figure 3.10 : User performance

User performance

1) **Brief description**
   This use case will be used to allow users (head, lecturer and officer) to submit online leave, view status and reply to complete form.

2) **Pre condition**
   The users must login at the user's login page.

3) **Characteristic of activation**
   Execution depent on user's command.

4) **Flow of control**
   4.1) Basic flow

- head, lecturer and officer select leave form.
- head, lecturer and officer fill the form.
- head, lecturer and officer submit the form.
- head, lecturer and officer view attendace and leave status.

4.2) Alternative flow

A-1: Users will get reply notification to complete the form.

4.3) Exeption flow

E-1: Invalid entering data

The system will display error message and admin have to reenter data.

5) **Post condition**

Head, lecturer and officer can manage user performance

3.3 DESIGN

A) DATABASE

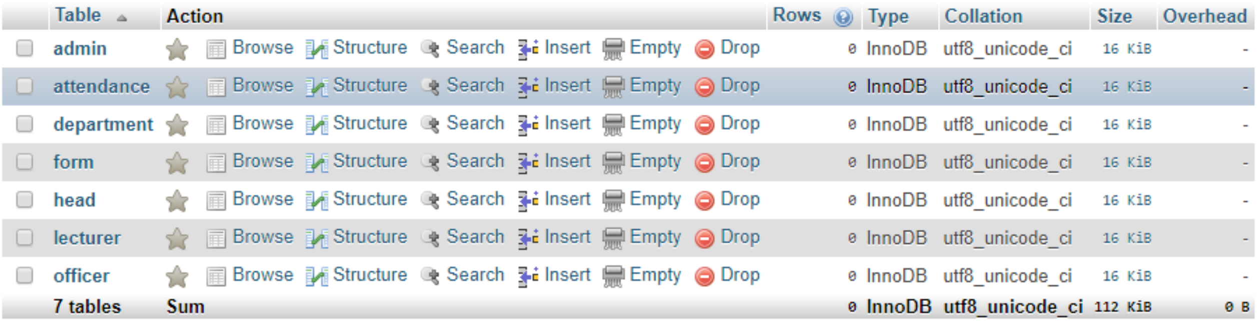
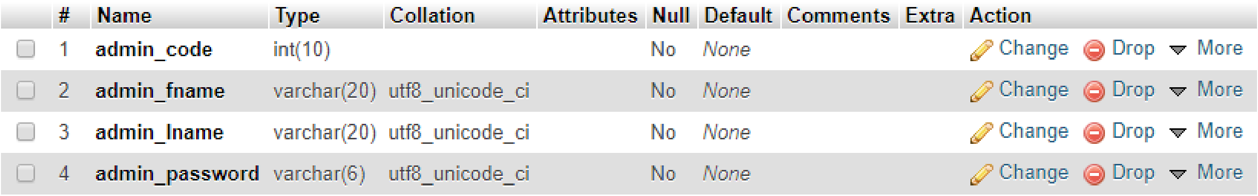


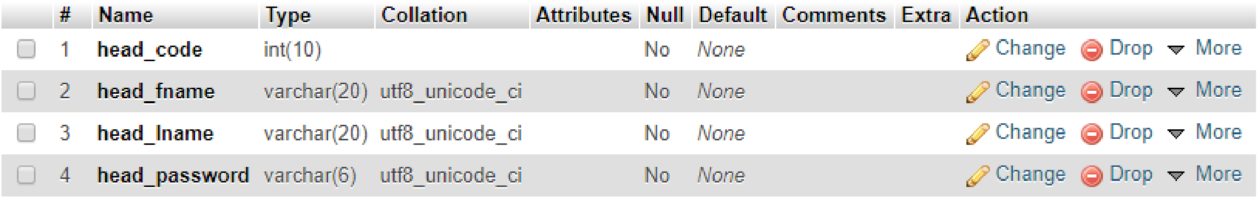
	Table ▲	Action	Rows	Type	Collation	Size	Overhead
□	admin	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci	16 KiB	-
□	attendance	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci	16 KiB	-
□	department	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci	16 KiB	-
□	form	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci	16 KiB	-
□	head	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci	16 KiB	-
□	lecturer	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci	16 KiB	-
□	officer	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci	16 KiB	-
	7 tables	Sum	0	InnoDB	utf8_unicode_ci	112 KiB	0 B

Figure 3.11 : The table of FTU Online Attendance for Lecturer and Officer



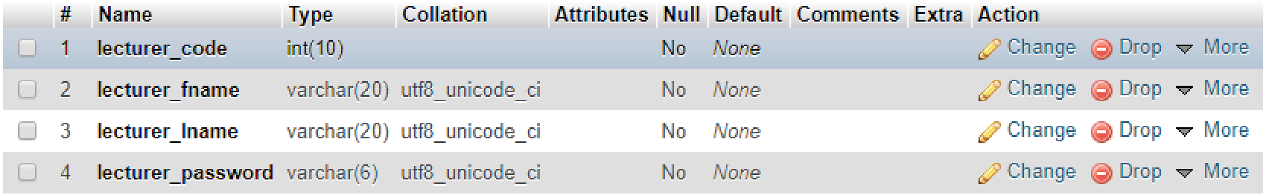
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	admin_code	int(10)			No	*None*			Change Drop ▼ More
□	2	admin_fname	varchar(20)	utf8_unicode_ci		No	*None*			Change Drop ▼ More
□	3	admin_lname	varchar(20)	utf8_unicode_ci		No	*None*			Change Drop ▼ More
□	4	admin_password	varchar(6)	utf8_unicode_ci		No	*None*			Change Drop ▼ More

Figure 3.12 : Administrator table



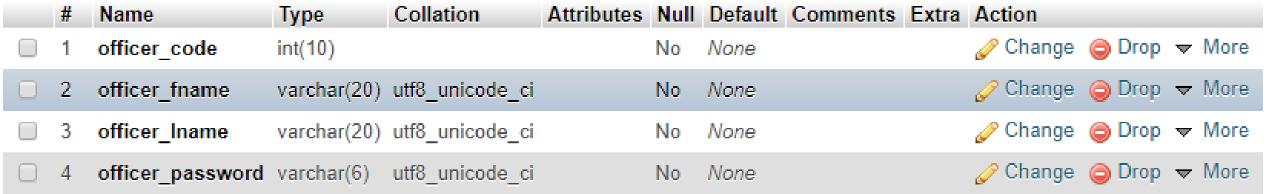
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	head_code	int(10)			No	*None*			Change Drop ▼ More
□	2	head_fname	varchar(20)	utf8_unicode_ci		No	*None*			Change Drop ▼ More
□	3	head_lname	varchar(20)	utf8_unicode_ci		No	*None*			Change Drop ▼ More
□	4	head_password	varchar(6)	utf8_unicode_ci		No	*None*			Change Drop ▼ More

Figure 3.13 : Head of department table



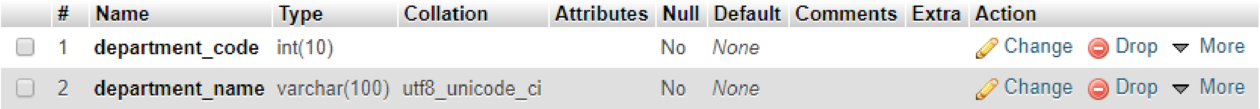
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	lecturer_code	int(10)			No	*None*			Change Drop ▼ More
□	2	lecturer_fname	varchar(20)	utf8_unicode_ci		No	*None*			Change Drop ▼ More
□	3	lecturer_lname	varchar(20)	utf8_unicode_ci		No	*None*			Change Drop ▼ More
□	4	lecturer_password	varchar(6)	utf8_unicode_ci		No	*None*			Change Drop ▼ More

Figure 3.14 : Lecturer table



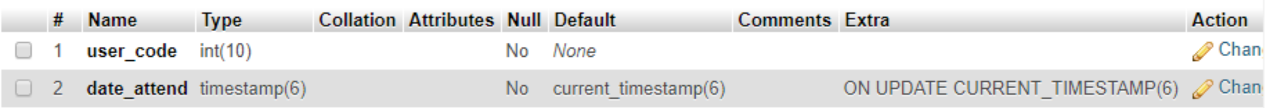
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**officer_code**	int(10)			No	*None*			Change Drop ▾ More
☐	2	**officer_fname**	varchar(20)	utf8_unicode_ci		No	*None*			Change Drop ▾ More
☐	3	**officer_lname**	varchar(20)	utf8_unicode_ci		No	*None*			Change Drop ▾ More
☐	4	**officer_password**	varchar(6)	utf8_unicode_ci		No	*None*			Change Drop ▾ More

Figure 3.15 : Officer table



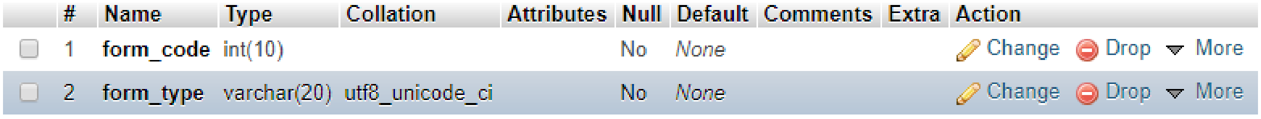
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**department_code**	int(10)			No	*None*			Change Drop ▾ More
☐	2	**department_name**	varchar(100)	utf8_unicode_ci		No	*None*			Change Drop ▾ More

Figure 3.16 : Department table



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**user_code**	int(10)			No	*None*			Chan
☐	2	**date_attend**	timestamp(6)			No	current_timestamp(6)		ON UPDATE CURRENT_TIMESTAMP(6)	Chan

Figure 3.17 : Attendance table



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	**form_code**	int(10)			No	*None*			Change Drop ▾ More
☐	2	**form_type**	varchar(20)	utf8_unicode_ci		No	*None*			Change Drop ▾ More

Figure 3.18 : Form table

B) USER INTERFACE

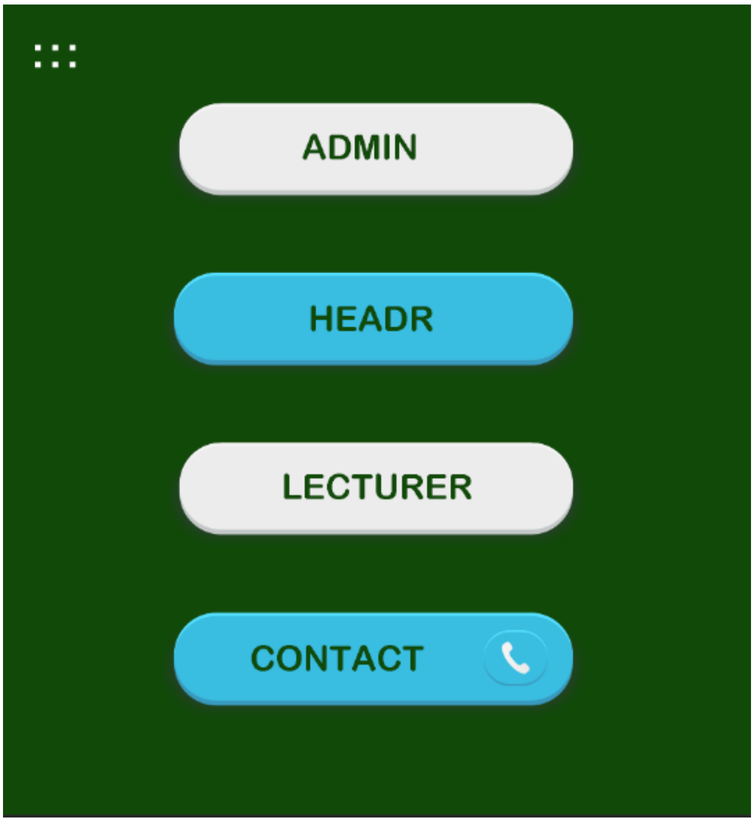


Figure 3.19 : Select position page to login page

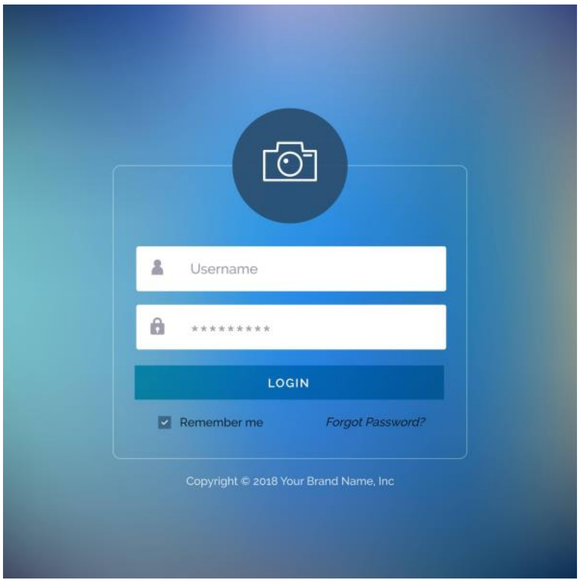


Figure 3.20 : Login page

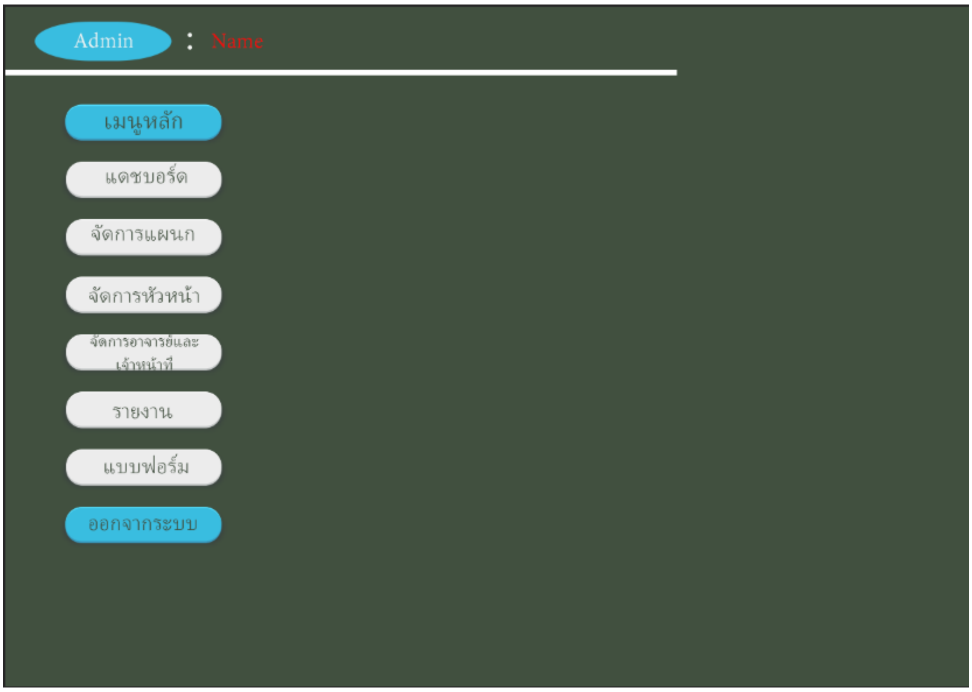


Figure 3.21 : Administrator management page

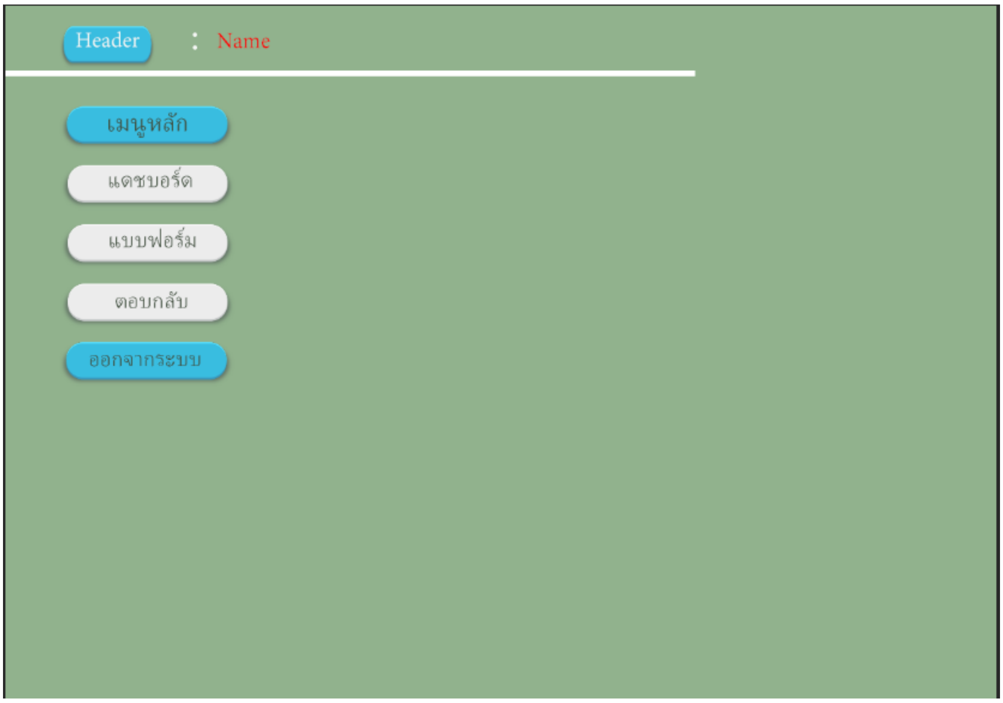


Figure 3.22 : Head of department management page

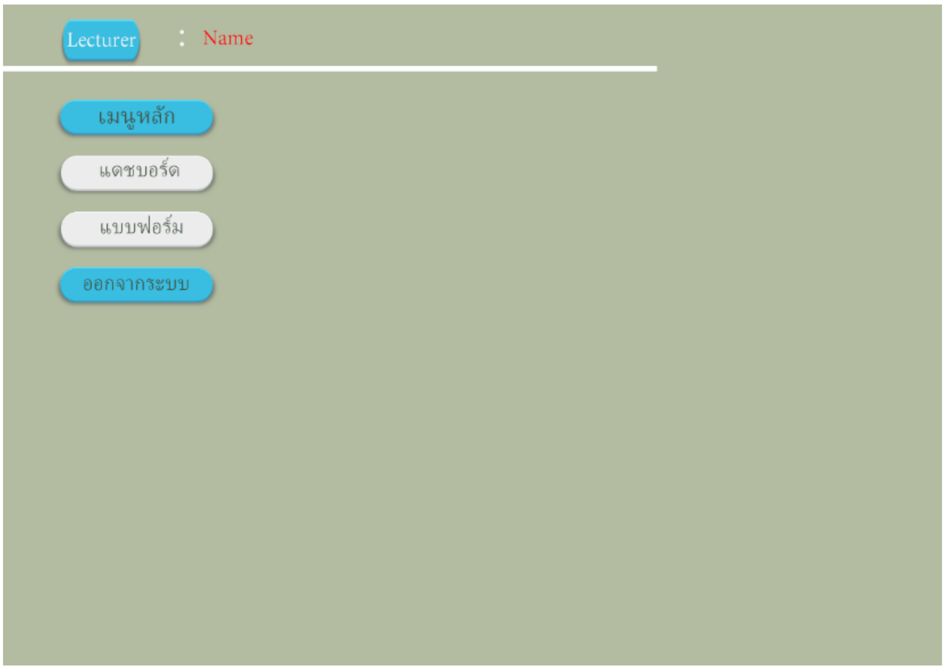


Figure 3.23 : Lecturer and officer page

3.4 IMPLAMENTATION

In the implementation phase we have a discussion about the coding and testing of the system. The implementation phase refers to the final process of moving the solution from development status to production status. Then we will explain more detail in the next chapter.

SUMMARY

Generally, this chapter describes the development process of FTU Online Attendance Management For Lecturer and Officer System. The development processes in which are including planning, analysis and design. The output of planning phase is the project feasibility or proposal and the output of analysis is the list of requirement and UML diagrams and the output of design phase is the database designing and user interface designing. After completing this chapter, the reader will know the number of subsystems and its main function of FTU Online Attendance Management For Lecturer and Officer System.