

การพัฒนาระบบสารสนเทศศึกษาระดับบัณฑิตศึกษาโดยวิธีการเชิงวัตถุ II
OBJECT-ORIENTED SOFTWARE DEVELOPMENT FOR
GRADUATE STUDENT INFORMATION SYSTEM II



นาย ชาศริต ภูษิต
นาย อาทิตย์ ธนพงศ์วิเศษ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

b.....
i.....

เลขหมู่.....
เลขทะเบียน 49944
วัน,เดือน,ปี. 2 ๒๕๔ 2547

การพัฒนาระบบสารสนเทศศึกษาระดับบัณฑิตศึกษาโดยวิธีการเชิงวัตถุ II

OBJECT-ORIENTED SOFTWARE DEVELOPMENT FOR

GRADUATE STUDENT INFORMATION SYSTEM II



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2545

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การพัฒนาระบบสารสนเทศศึกษาระดับบัณฑิตศึกษาโดยวิธีเชิงวัตถุ II

OBJECT-ORIENTED SOFTWARE DEVELOPMENT FOR GRADUATE STUDENT
INFORMATION SYSTEM II

ผู้จัดทำ

- | | | |
|---------------|-------------|-----------------------|
| 1. นายชาคริต | ภุชิต | รหัสนักศึกษา 43015354 |
| 2. นายอาทิตย์ | ชนพงษ์วิเศษ | รหัสนักศึกษา 43015401 |



(ดร.วิศิษฐ์ หิรัญกิตติ)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การพัฒนาระบบสารสนเทศศึกษาระดับบัณฑิตศึกษาโดยวิธีการเชิงวัตถุ II

นาย ชاکริต ภูษิต 43015354

นาย อาทิตย์ ธนพงศ์วิเศษ 43015401

ดร.วิศิษฎ์ หิรัญกิตติ อาจารย์ที่ปรึกษา

ปีการศึกษา 2545

บทคัดย่อ

ระบบสารสนเทศศึกษาระดับบัณฑิตศึกษาของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่มีอยู่เดิมนั้นได้ประสบปัญหาในการใช้งานอยู่หลายด้าน เนื่องจากขั้นตอนการลงทะเบียนที่มีหลายขั้นตอน อีกทั้งในระบบข้อมูลทะเบียนนักศึกษาระดับบัณฑิตศึกษานั้นมีความซับซ้อนและมีเงื่อนไขต่างๆอยู่มาก และเปลี่ยนแปลงอยู่ตลอดเวลา ระดับความถูกต้องและการตรวจสอบกระบวนการต่างๆ ยังทำโดยใช้พนักงานดำเนินการเป็นหลัก ทำให้เกิดความผิดพลาดขึ้น ดังนั้นควรมีการปรับปรุงระบบสารสนเทศศึกษาเดิมเพื่อแก้ไขปัญหาล่าช้า

ปริญญาานิพนธ์นี้ได้ศึกษา วิเคราะห์ ออกแบบและพัฒนาระบบสารสนเทศศึกษาระดับบัณฑิตศึกษาขึ้นมาใหม่ โดยใช้วิธีการใหม่ คือใช้วิธีการพัฒนาซอฟต์แวร์เชิงวัตถุ (Object-Oriented Software Development) ในการวิเคราะห์และออกแบบ ได้ใช้ภาษาสร้างแบบจำลอง UML และทำการพัฒนาระบบโดยใช้ภาษาจาวา รวมทั้งยังได้ใช้ Java Servlet และ Java Server Page (JSP) ในการพัฒนาเว็บแอปพลิเคชัน และใช้ Java Bean ในการเข้าถึงและจัดการข้อมูลในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**OBJECT-ORIENTED SOFTWARE DEVELOPMENT FOR
GRADUATE STUDENT INFORMATION SYSTEM II**

Chakrit	Pusit
Athit	Tanaphonwiset
Dr.Visit	Hirankitti Advisor

ABSTRACT

The information system for graduate students currently used in KMITL suffers many problems due to the complication of the registration procedures, a vast amount of data, and many continuous changes with the regulations. To operate the system correctly, the registrar has to validate some procedures manually, however, this reduces the system's efficiency. To solve such problems, a new information system has been developed.

The aim of this project is to study, analyze, design, and implement a new information system for the graduate students. Using an object-oriented software development approach. That is, UML is used for modeling purpose both in system analysis and design, Java is used for implementation; Java Servlet and Java Server Page (JSP) are used to develop web application; Java Beans are used to implement database access for the system. Finally relational database are adopted to implement the system database.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จได้ด้วยดี เนื่องจากทางคณะผู้จัดทำได้รับความร่วมมือ และความอนุเคราะห์จากบุคคลต่าง ๆ หลายฝ่าย ทั้งทางด้านการทำงาน กำลังใจ ทางคณะผู้จัดทำขอขอบพระขอบคุณท่านเหล่านั้นมา ณ โอกาสนี้

ดร.วิศิษฎ์ หิรัญกิตติ อาจารย์ที่ปรึกษาที่ได้ให้ความกรุณาช่วยเหลือ ให้คำปรึกษาที่เป็นประโยชน์ ชี้แจงข้อผิดพลาด และเสียสละเวลาดูแลเอาใจใส่ในการทำงานของพวกเรามาโดยตลอด

คณาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ทุกท่านที่ได้ให้ความรู้ทั้งทางด้านวิชาการและด้านการใช้ชีวิตในสังคมแก่พวกเรา

และต้องขอขอบพระคุณบุคคลที่มีความสำคัญที่สุดที่ทำให้มีวันนี้ได้ ก็คือ บิดา และ มารดาอันเป็นที่เคารพและรักยิ่งซึ่งได้อบรมเลี้ยงดูมาเป็นอย่างดี พร้อมทั้งได้ให้โอกาสในการศึกษาที่ดีที่สุด และให้กำลังใจในการทำงานเสมอมา คณะผู้จัดทำขอระลึกในพระคุณ และขอกราบขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูปรตาราง	VI
สารบัญรูปภาพ	VII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของงานวิจัย	2
1.4 วิธีการดำเนินงาน	3
บทที่ 2 การพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุและยูเอ็มแอล	4
2.1 บทนำ	4
2.2 การพัฒนาระบบด้วยวิธีเชิงวัตถุ	5
2.3 ข้อดีของการพัฒนาระบบด้วยวิธีเชิงวัตถุ	5
2.4 ความสามารถของคลาส	7
2.5 Object Oriented Methodologies	8
2.6 กระบวนการพัฒนาซอฟต์แวร์ (Software Process) ที่ดี	11
2.7 ส่วนประกอบของ UML	13
บทที่ 3 Servlet, JSP และ Java Bean	20
3.1 เซิร์ฟเล็ต (Servlet)	20
3.2 เจเอสพี (Java Server Page : JSP)	22
3.3 จาวาบีน (JavaBean)	25
บทที่ 4 เครื่องมือที่ใช้ในการพัฒนาโปรแกรมและฐานข้อมูล	28
4.1 บทนำ	28
4.2 ลักษณะของ DB2ที่ใช้ในโครงการ	28
4.3 การเชื่อมต่อเข้ากับฐานข้อมูลโดยใช้ JDBC	28

5.1 ฝ่ายทะเบียนนักศึกษาของบัณฑิตวิทยาลัย	33
5.2 ปัญหาที่พบในระบบทะเบียนนักศึกษาเดิม	34
5.3 การวิเคราะห์ระบบ	34
5.4 การออกแบบระบบ โดยใช้ยูเอ็มแอล	35
5.5 การออกแบบการทำงานของฟังก์ชันต่าง ๆ	39
5.6 การออกแบบการรายละเอียดของคลาสต่าง ๆ	42
5.7 การออกแบบโครงสร้างของระบบ	43
5.8 การออกแบบระบบ เพื่อสร้างระบบสารสนเทศ	44
บทที่ 6 การพัฒนาและทดสอบระบบ	45
6.1 การสร้างฐานข้อมูล	45
6.2 การเขียนโปรแกรมในส่วนของแอปพลิเคชัน	50
6.3 การทดสอบโปรแกรมในส่วนของแอปพลิเคชัน	53
6.4 การเขียนโปรแกรมในส่วนของเว็บแอปพลิเคชัน	59
6.5 การทดสอบโปรแกรมในส่วนของเว็บแอปพลิเคชัน	60
บทที่ 7 บทสรุปและบทวิจารณ์	65
7.1 บทสรุป	65
7.2 บทวิจารณ์	66
7.3 ปัญหาที่พบในการดำเนินงาน	66
7.4 ข้อเสนอแนะสำหรับระบบทะเบียนนักศึกษา ของบัณฑิตวิทยาลัย	66
7.5 ข้อเสนอแนะในการนำไปพัฒนาต่อ	67
ภาคผนวก ก	68
ภาคผนวก ข	107
ภาคผนวก ค	115
บรรณานุกรม	138

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

เรื่อง

หน้า

บทที่ 3 Servlet, JSP และ Java Bean

3-1 เปรียบเทียบเทคโนโลยี JSP กับเทคโนโลยีอื่น

23



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

เรื่อง	หน้า
บทที่ 2 การพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุและยูเอ็มแอล	
2-1 การเชื่อมโยงของ Use Case Model กับ Model อื่น ๆ	10
2-2 Best Practices of Software Engineering	11
2-3 Deployment Iteratively Process	11
2-4 สถาปัตยกรรมของ View	14
2-5 แสดงไคอะแกรมต่างๆ ของ UML	15
2-6 ตัวอย่าง Use Case ในระบบลงทะเบียน	15
2-7 แสดงตัวอย่าง ซีเคิร์นซ์ไคอะแกรม	17
2-8 แสดงตัวอย่างแอคตีวิตีไคอะแกรม	18
2-9 แสดงตัวอย่างคอมโพเน้นไคอะแกรม	19
2-10 แสดงตัวอย่างดีพลอยเม้นไคอะแกรม	19
บทที่ 3 Servlet, JSP และ Java Bean	
3-1 รูปอธิบายการทำงานของ JSP	24
บทที่ 4 เครื่องมือที่ใช้ในการพัฒนาโปรแกรมและฐานข้อมูล	
4-1 แสดงลักษณะการทำงานของ JDBC	29
4-2 แสดงรูปแบบ 2-Tier	30
4-3 แสดงรูปแบบ 3-Tier	30
4-4 ระดับการเชื่อมต่อของ JDBC API	30
4-5 ลำดับการเชื่อมต่อเข้าฐานข้อมูลโดยใช้ JDBC	32
บทที่ 5 การวิเคราะห์และออกแบบระบบทะเบียนนักศึกษาในระดับบัณฑิตศึกษา	
5-1 Usecase ในส่วนของนักศึกษา	36
5-2 Usecase ในส่วนของเจ้าหน้าที่	37
5-3 Activity Diagram ของ ยูสเคส ลงทะเบียนล่วงหน้า	40
5-4 Sequence Diagram ของ ยูสเคส ลงทะเบียนปกติ	41
5-5 ตัวอย่าง Class Diagram ของหลักสูตร	42
5-6 Deployment Diagram ของระบบ	43
5-7 การแบ่งการทำงานของระบบสารสนเทศ	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6 การพัฒนาและทดสอบระบบ	
6-1 การสร้างฐานข้อมูลบน DB2 โดยใช้โปรแกรม Control Center	49
6-2 การใช้ Command Center เพื่อการสร้างตาราง	50
6-3 รูปส่วนอินเทอร์เฟซของ MainPanel	52
6-4 รูปส่วนอินเทอร์เฟซของ รายละเอียดหลักสูตร	52
6-5 แสดงหน้าจอการล็อกอิน	53
6-6 รูปแสดงข้อมูลที่อยู่ในฐานข้อมูล	53
6-7 แสดงการเลือกเมนู ข้อมูลหน่วยงานศึกษา	54
6-8 แสดงข้อมูลของคณะและภาควิชา	54
6-9 แสดงข้อมูลในตาราง Faculty	55
6-10 แสดงการเพิ่มภาควิชา	55
6-11 แสดงผลการเพิ่มภาควิชา	56
6-12 แสดงการเพิ่มสาขาวิชา	56
6-13 แสดงผลการเพิ่มสาขาวิชา	57
6-14 แสดงข้อมูลหลังจากการเพิ่มสาขาวิชา	57
6-15 แสดงข้อมูลในตาราง Department	58
6-16 การทดลองใช้งานผ่านเว็บเบราว์เซอร์	59
6-17 แสดงช่วงเวลาปัจจุบัน	60
6-18 แสดงเมนูที่เปลี่ยนแปลงตามช่วงเวลา	60
6-19 แสดงให้เลือกลงทะเบียนตามรายวิชาที่เปิดสอนตามหลักสูตร	61
6-20 แสดงรายวิชาที่เปิดสอนตามหลักสูตร	61
6-21 แสดงรายละเอียดรายวิชาที่เปิดสอนตามหลักสูตร	62
6-22 แสดงระบบตรวจสอบวิชาที่ต้องเรียนวิชาบังคับก่อน	62
6-23 แสดงรายวิชาที่ลงทะเบียน	63
6-24 แสดงค่าใช้จ่ายตามหลักสูตร	63
6-25 แสดงข้อความการลงทะเบียนสมบูรณ์	64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มา

การพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุ(Object-Oriented Software Engineering) นับเป็นกระบวนการในการพัฒนาโปรแกรมที่ประสบความสำเร็จอย่างยิ่ง อันเนื่องมาจากเป็นวิธีการที่ได้อธิบายกระบวนการ วิธีการไว้อย่างละเอียดและเป็นลำดับขั้นตอนชัดเจน ทำให้สามารถนำไปปฏิบัติตามได้ง่าย และยังมีรูปแบบที่เป็นมาตรฐาน ทำให้สามารถนำมาใช้เป็นตัวในการสื่อสารให้เกิดความเข้าใจในแนวทางเดียวกันในการทำงานเป็นทีม หรือการสื่อสารกับบุคคลอื่นที่เกี่ยวข้องกับระบบที่จะพัฒนาได้อีกด้วย

ในการพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุ นั้น ได้นำเอาภาษาสัญลักษณ์เข้ามาใช้ในการสื่อความหมายว่าระบบทำงานอย่างไร แต่ละวัตถุทำงานและเชื่อมต่อกันอย่างไร ภาษาสัญลักษณ์ดังกล่าวมีชื่อว่า ยูเอ็มเอล (Unified Modeling Language: UML) ซึ่งอยู่เราใช้ยูเอ็มเอลแสดงถึงมุมมองต่าง ๆ ของระบบออกมาในรูปของไดอะแกรม(Diagram) ซึ่งมีหลายไดอะแกรม แต่ละไดอะแกรมก็จะแสดงมุมมองแต่ละมุมมองของระบบออกมา อีกทั้งการพัฒนาโปรแกรมเชิงวัตถุยังมีโปรแกรมที่ใช้เป็น เครื่องมือช่วยในการพัฒนาโปรแกรม(CASE Tools) อยู่มาก จึงทำให้เพิ่มความสะดวกและมาตรฐานในการพัฒนาโปรแกรมขึ้นอย่างมาก

จากเหตุผลข้างต้นการพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุ นั้นจึงได้รับการนิยมนำมาใช้ในการพัฒนาโปรแกรมในระดับต่าง ๆ มากขึ้น แต่สำหรับในประเทศไทยนั้น การพัฒนาโปรแกรมคอมพิวเตอร์ในภาคอุตสาหกรรมนั้นยังมีไม่มากนัก จึงส่งผลให้วิธีการพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุถูกนำไปใช้น้อยเมื่อเทียบเชิงปริมาณ บุคลากรที่มีความรู้ความเข้าใจในด้านดังกล่าวก็มีน้อยด้วยเช่นกัน ดังนั้นในอนาคตความต้องการผู้ที่มีความรู้ความเข้าใจในด้านนี้ก็จะเพิ่มมากขึ้นเรื่อย ๆ ตามความเติบโตของอุตสาหกรรมซอฟต์แวร์

แต่ถึงแม้ว่าการพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุ นั้นจะมีหลักการที่เป็นมาตรฐาน แต่การที่จะเรียนรู้ และทำความเข้าใจจนสามารถนำวิธีดังกล่าวไปใช้อย่างถูกต้องนั้น ไม่ใช่เป็นเรื่องที่ง่ายเลย เพราะเรามักจะคุ้นเคยกับการเขียนโปรแกรมขนาดเล็ก คุ้นเคยกับการทำงานที่ลงมือทำเลย เมื่อพบปัญหาแล้วจึงค่อย ๆ แก้ปัญหานั้น ๆ แต่ในการพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุ นั้นจะต้องมองภาพของระบบทั้งหมดให้ได้ แล้วมองแยกย่อยระบบออกเป็นวัตถุเป็นชิ้น ๆ ต้องหาความสัมพันธ์ระหว่างวัตถุ การสื่อสารระหว่างวัตถุ และวิเคราะห์ไกลไปจนถึงปัญหาที่อาจเกิดขึ้นได้และทางแก้ปัญหานั้น ซึ่งนั่นก็หมายความว่าผู้ออกแบบระบบจะต้องมีการวิเคราะห์และวางแผนอย่างดียิ่งเสียก่อนที่จะดำเนินการเขียนโปรแกรมนั้นจริง ๆ

และที่อยากอีกอย่างหนึ่งก็คือความเข้าใจอย่างถูกต้องในการใช้สัญลักษณ์ต่าง ๆ ในยูเอ็มเอลเพราะเราอาจใช้อย่างผิด ๆ ทำให้การสื่อความหมายไม่ถูกต้อง และอาจไม่ก่อให้เกิดประโยชน์ใด ๆ เลย เพราะในการใช้ยูเอ็มเอลนั้นเพียงการเรียนรู้แต่ไม่ได้ลงมือทำอย่างจริงจังแล้วนั้น ไม่อาจทำให้เข้าใจในยูเอ็มเอลได้อย่างแน่นอน ถึงแม้จะเข้าใจก็อาจเข้าใจเพียงบางแง่มุมเท่านั้นเอง ดังนั้นการที่ได้ลงมือปฏิบัติจริงในการพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุ นั้น จึงก่อให้เกิดประโยชน์อย่างมากกับผู้ที่ศึกษา ทั้งก่อให้เกิดความเข้าใจและมองเห็นข้อเด่น ข้อด้อย และข้อควรระวังในการใช้วิธีนี้ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกประการหนึ่งก็คือ ระบบจัดการข้อมูลทะเบียนนักศึกษาในระดับบัณฑิตศึกษาของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่มีอยู่เดิมนั้น ได้ประสบปัญหาในดำเนินงานอยู่หลายด้าน ไม่ว่าจะเป็นเรื่องของความเร็วและความสะดวกที่มีไม่มากเท่าที่ควร อีกทั้งในระบบข้อมูลทะเบียนนักศึกษาระดับบัณฑิตศึกษานั้นมีความซับซ้อน และมีเงื่อนไขอยู่มาก ระดับความถูกต้องและการตรวจสอบกระบวนการต่าง ๆ ยังทำโดยใช้พนักงานดำเนินการเป็นหลักจึงอาจมีข้อผิดพลาดและข้อสับสนเกิดขึ้นได้ง่าย

การเข้ามาวิเคราะห์และพัฒนาระบบดังกล่าวให้มีความสามารถขึ้นและเพิ่มความสะดวกให้กับผู้ใช้ระบบซึ่งมีทั้งนักศึกษา เจ้าหน้าที่ทะเบียน อาจารย์จึงนับว่ามีประโยชน์อย่างยิ่ง

1.2 วัตถุประสงค์

1.2.1 เพื่อนำวิธีการทางวิศวกรรมซอฟต์แวร์เชิงวัตถุไปพัฒนาระบบทะเบียนนักศึกษาในระดับบัณฑิตศึกษาของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังให้มีความสามารถยิ่งขึ้น สร้างความสะดวกและความถูกต้องในการทำงานมากขึ้น

1.2.2 เพื่อศึกษาและทำความเข้าใจระบบทะเบียนนักศึกษาในระดับบัณฑิตศึกษาของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังเดิมว่าเป็นอย่างไร และสามารถปรับปรุงให้ดีขึ้นโดยไม่ทำให้เกิดผลกระทบต่อผู้ปฏิบัติงานและระบบอื่นที่เกี่ยวข้องมากนัก

1.2.3 เพื่อศึกษาและทำความเข้าใจวิธีการพัฒนา โปรแกรมด้วยวิธีเชิงวัตถุว่ามีกระบวนการอย่างไร ต้องวิเคราะห์และดำเนินการอย่างไรจึงจะถูกต้อง และทำให้ได้โปรแกรมที่มีความถูกต้องและมีประโยชน์มากที่สุด

1.2.4 เพื่อศึกษาและทำความเข้าใจในการใช้ประโยชน์จากเครื่องมือช่วยในพัฒนาโปรแกรม(CASE Tools) ว่ามีตัวใดบ้าง ใช้งานในกระบวนการใดบ้าง ใช้อย่างไร มีความสามารถใดบ้าง และ ใช้เครื่องมือเหล่านี้ทำการพัฒนาระบบให้สามารถนำมาใช้งานได้

1.3 ขอบเขตของงานวิจัย

งานวิจัยนี้จะเป็นการดำเนินการศึกษาและทำความเข้าใจการพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุ และการใช้ประโยชน์จากเครื่องมือช่วยในการพัฒนา โปรแกรม และนำความรู้ดังกล่าวเข้ามาช่วยในการพัฒนาระบบทะเบียนนักศึกษาในระดับบัณฑิตศึกษาของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ให้มีความสามารถและมีความสะดวกมากขึ้นและทำการแก้ปัญหาที่มีในระบบเดิมเช่น การตรวจสอบรายวิชาที่จำเป็นต้องเรียนมาก่อน(Prerequisite) ซึ่งในระบบเดิมยังไม่มีการตรวจสอบ โดยให้นักศึกษาสามารถดำเนินการเกี่ยวกับระบบทะเบียนและเข้าถึงข้อมูลของตนโดยผ่าน โปรแกรมประยุกต์บนอินเทอร์เน็ต(Web Application) และเจ้าหน้าที่ทะเบียนสามารถดำเนินงานโดยผ่านโปรแกรมประยุกต์ผ่านระบบเครือข่ายภายในสถาบันได้ รวมถึงได้ทำการวิเคราะห์และปรับขั้นตอนการทำงานของงานทะเบียนนักศึกษาให้มีลำดับขั้นตอนอย่างชัดเจน ให้มีความคล่องตัวมากขึ้น และในสามารถกำหนดระยะเวลาการดำเนินการต่าง ๆ ให้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่อย่างไรก็ตามขอบเขตที่จะดำเนินการนั้นจะครอบคลุมงานที่เกี่ยวข้องกับงานทะเบียนของนักศึกษาเป็นหลักไม่ว่าจะเป็นการลงทะเบียน การเปลี่ยนแปลงใข้รายการลงทะเบียน ข้อมูลประวัตินักศึกษา เกรดนักศึกษา ตารางเรียน ตารางสอบ การดำเนินการตามคำร้องต่าง ๆ การจัดการใบทรานสคริปต์และอื่น ๆ ที่เกี่ยวข้อง โดยจัดทำ Prototype ในส่วนที่ได้พัฒนาให้กับเจ้าหน้าที่ได้ทำการทดลองใช้งาน

อีกประการหนึ่งก็คือในการพัฒนาระบบดังกล่าวนี้จะพยายามไม่ทำการเปลี่ยนแปลงลักษณะสำคัญของระบบเดิม เพื่อไม่ให้กระทบกับตัวระบบเดิม และระบบอื่นที่เกี่ยวข้องมากจนเกินไป ดังนั้นจึงมีงานบางอย่างที่อาจมีความยุ่งยากมากกว่าที่ควรจะเป็น แต่ถึงอย่างนั้นก็ยังเป็นการเพิ่มความสะดวกและถูกต้องให้มากขึ้นในระดับที่น่าพอใจ และเอื้อต่อการนำไปพัฒนาต่อในอนาคต

1.4 วิธีการดำเนินงาน

งานวิจัยในโครงการนี้จะเริ่มด้วยการศึกษาทฤษฎีพื้นฐานต่าง ๆ ที่เกี่ยวข้องกับงานวิจัย ซึ่งแบ่งเป็น 2 ส่วนหลัก ๆ นั่นก็คือ เรื่องของกระบวนการพัฒนาโปรแกรมเชิงวัตถุ ยูเอ็มเอล(Unified Modeling Language: UML) และระบบทะเบียนศึกษาทั้งในระดับบัณฑิตศึกษา และระดับต่ำกว่าบัณฑิตศึกษา โดยการเข้าไปทำการสัมภาษณ์เจ้าหน้าที่ที่ปฏิบัติงานอยู่ในระบบเดิม รวมทั้งอาจารย์และนักศึกษาในระดับบัณฑิตศึกษาอีกด้วย และยังเข้าไปศึกษาถึงรูปแบบฟอร์มต่าง ๆ ที่ใช้อยู่เดิมด้วย แล้วนำความรู้ดังกล่าวมาวิเคราะห์ และนำมาเขียนเป็นยูเอ็มเอลเพื่ออธิบายมุมมองต่าง ๆ ของระบบ

ทฤษฎีอีกส่วนหนึ่งที่ต้องทำการศึกษาก็คือเรื่องของเครื่องมือที่ใช้ในการพัฒนาโปรแกรม ไม่ว่าจะเป็น IBM DB2 ซึ่งเป็นโปรแกรมจัดการฐานข้อมูล และโปรแกรมที่ใช้ในการเขียนโปรแกรมนั่นคือ Forte For Java รวมถึงส่วนของภาษาจาวาด้วย ซึ่งจะมีอยู่ในรายละเอียดในบทที่ 3, และ 4 ตามลำดับ

หลังจากที่เราได้โมเดลของระบบแล้วสิ่งที่เราต้องทำการออกแบบต่อไปก็คือเรื่องของฐานข้อมูล ซึ่งเราจะนำข้อมูลจากคลาสต่าง ๆ ที่ได้เราออกแบบไว้แล้วนั้น นำมาแปลงให้เป็นโครงสร้างตารางข้อมูลต่อไป ขั้นตอนต่อจากนี้ไปก็จะเป็นการเขียนโปรแกรมทั้งในส่วนของแอปพลิเคชัน และเว็บแอปพลิเคชัน ทั้งสองส่วนนี้ใช้ภาษาจาวาในการพัฒนาซึ่งจะกล่าวถึงในบทที่ 6

บทที่ 2

การพัฒนาโปรแกรมด้วยวิธีเชิงวัตถุและยูเอ็มแอล

2.1 บทนำ

ในราวๆปี 1990 มีวิธีที่ใช้ Model ระบบมากมาย แต่มีอยู่ 3 วิธีที่เป็นที่นิยมใช้ในการ model ระบบคือ OMT (Rumbaugh), Booch และ OOSE (Jacobson) โดยในแต่ละวิธี จะมีข้อดีข้อเสียต่างกัน วิธีแบบจำลองรูปภาพเชิงวัตถุ (Object Modeling Technology: OMT) นั้นจะมีจุดแข็งในการวิเคราะห์ระบบได้ง่าย แต่พื้นที่ออกแบบนั้นจะใช้ยาก ส่วนวิธี Booch นั้นจะมีจุดแข็งในการออกแบบ แต่การวิเคราะห์จะยาก และวิธีสุดท้ายวิธีแบบจำลองรูปภาพเชิงวัตถุทางวิศวกรรมซอฟต์แวร์ (Object Oriented Software Engineering: OOSE) จะมีจุดแข็งในคุณสมบัติของการวิเคราะห์แต่มีจุดอ่อนในจุดต่างๆอีก เมื่อเวลาผ่านไป การใช้วิธีนำเสนอรายละเอียดงานที่ต่างกัน ทำให้เกิดความสับสน ดังนั้นจึงจำเป็นต้องมีวิธีที่มาตรฐานที่สามารถเข้าใจได้ง่าย และยังคงอยู่บนพื้นฐานของวิธีเดิมที่มีอยู่ UML จึงได้ถือกำเนิดขึ้นมาจากการร่วมมือกันของทั้ง Booch ,Jacobson ,Rumbaugh รวมถึง คนอื่น โดยได้นำข้อดีของแต่ละวิธีมารวมกัน

Unified Model Language หรือ UML เป็นภาษาโมเดลที่นำเอาสัญลักษณ์มาช่วยในการเพิ่มประสิทธิภาพในส่วนของ การวิเคราะห์(Analysis) และออกแบบ(Design) ซึ่งมีประโยชน์ในหลายด้าน โดยจะเป็นรูปแบบของออปเจกต์โอเรียนเท็ด ซึ่งการออกแบบจะทำให้เราสามารถวิเคราะห์ส่วนประกอบโดยรวมของระบบและยังสามารถที่จะรองรับภาษาที่จะใช้ในการพัฒนาได้หลากหลายภาษา และยังมีหลีกเลี่ยงความซ้ำซ้อนและเฉพาะเจาะจงเกินไป ซึ่งการออกแบบโดยวิธี UML นี้สามารถที่จะแสดงออกมาในรูปแบบของไดอะแกรม(Diagram)ต่างๆ

แอปพลิเคชันขั้นที่พัฒนาขึ้นนั้นต่างก็มีขนาด ความซับซ้อน และการกระจายที่มากขึ้น การพัฒนาซอฟต์แวร์ให้สามารถทำงานได้ทั้งเร็ว และถูกต้องนั้น จึงจำเป็นต้องมีการพัฒนาร่วมกันเป็นทีม ซึ่งความยากอย่างหนึ่งของทีมพัฒนาซอฟต์แวร์ก็คือการสื่อสารระหว่างนักพัฒนาในทีม เนื่องจากเมื่อซอฟต์แวร์มีขนาดใหญ่ นักพัฒนาในทีมก็มีจำนวนมากขึ้นตามขนาดของซอฟต์แวร์ และแต่ละท่านก็มีความเชี่ยวชาญในด้านต่างๆกัน นอกจากนั้นเทคโนโลยีในการพัฒนามีการเปลี่ยนแปลงที่รวดเร็วมาก

นอกจากนั้น ปัญหาของการพัฒนาซอฟต์แวร์ ส่วนหนึ่งเกิดจากการที่นักพัฒนาไม่สามารถจัดการกับความต้องการที่เปลี่ยนแปลงของแต่ละโมดูล รวมถึงซอฟต์แวร์ที่พัฒนาขึ้นนั้นไม่สามารถทำงานร่วมกันได้ และยังคงรักษาและเปลี่ยนแปลงได้ยาก นักพัฒนาในทีมไม่สามารถติดตามได้ว่าใครแก้ไขอะไร เมื่อใด ที่ไหน และทำไม

สาเหตุสำคัญที่ทำให้เกิดปัญหาในการพัฒนาซอฟต์แวร์ก็คือ ไม่มีระบบจัดการความต้องการที่ดีเพียงพอ มีการสื่อสารที่ไม่รัดกุมและกำกวม สถาปัตยกรรมของซอฟต์แวร์ไม่มั่นคง

ดังนั้น การจัดการกระบวนการในการพัฒนาจึงจำเป็นต้องการพัฒนาซอฟต์แวร์ให้ถูกต้อง และมีคุณภาพตามที่ต้องการ จึงจำเป็นต้องอย่างยิ่งที่จะต้องศึกษาการพัฒนากระบวนกรด้วยวิธีเชิงวัตถุ, กระบวนการพัฒนาซอฟต์แวร์การค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ดี, ส่วนประกอบของยูเอ็มแอล และเครื่องมือที่ใช้วิเคราะห์และออกแบบซึ่งใช้ Rational Rose เพื่อการพัฒนา ระบบซอฟต์แวร์ที่มีคุณภาพตามความต้องการมากที่สุด

2.2 การพัฒนาระบบด้วยวิธีเชิงวัตถุ

การพัฒนาระบบด้วยวิธีเชิงวัตถุนั้นเป็นการพัฒนาที่ช่วยลดความยุ่งยากซับซ้อนของระบบใหญ่ๆได้ โดยผู้พัฒนาจะมองระบบและส่วนประกอบหรือระบบย่อย (Sub System) เป็นวัตถุ (Object) เนื่องจากสิ่งต่างๆ ที่มีอยู่ในโลกความเป็นจริงนั้นก็ถือได้ว่าเป็นวัตถุได้อยู่แล้ว และด้วยการมองระบบเป็นวัตถุทำให้ผู้พัฒนา ระบบสามารถมองระบบได้ชัดเจนมากยิ่งขึ้น ส่งผลให้การวิเคราะห์และออกแบบระบบทำได้ง่ายและมีความ ถูกต้องมากขึ้น

การพัฒนาระบบด้วยวิธีเชิงวัตถุนั้น ยังมีส่วนเกี่ยวข้องกับทุกขั้นตอนของการพัฒนาระบบ ทั้งการ รวบรวมความต้องการ (Requirement gathering), การวิเคราะห์ (analysis), การออกแบบ (design), การเขียน โปรแกรม (coding), และการทดสอบ (testing) ซึ่งเป็นการแก้ปัญหาของการพัฒนาระบบสมัยก่อนแบบที่ เรียกว่า การพัฒนาแบบโครงสร้าง (Structural) ซึ่งเป็นการบอกเพียงว่าระบบจะมีลักษณะการทำงานอย่างไรแต่ ไม่ได้บอกถึงวิธีการพัฒนาระบบหรือรูปแบบการเขียน โปรแกรมเลย จุดนี้เองทำให้หลายๆครั้ง ระบบที่ พัฒนาขึ้นมีปัญหาเนื่องจากการตรวจสอบความถูกต้องของระบบ (Validate) ระบบในเชิงพฤติกรรม (Behavior) นั้นทำได้ยาก

2.3 ข้อดีของการพัฒนาระบบด้วยวิธีเชิงวัตถุ

2.3.1 Higher level of abstraction วิธีการวิเคราะห์ระบบแบบโครงสร้างนั้นมีข้อจำกัดในการแยกแยะ เอกลักษณะ และการซ่อนรายละเอียด อย่างไรก็ตามการวิเคราะห์และพัฒนาระบบแบบโครงสร้างนี้ จะแยกแยะ โครงสร้างและการทำงานของระบบออกจากกัน เพื่อให้เกิดประสิทธิภาพในการทำงาน ตัวอย่างเช่น เซนเซอร์ จะได้รับการ โมเดลให้เป็นส่วนหนึ่งของข้อมูลการอ่านหรือการกระทำอื่นๆ กับเซนเซอร์จะถือว่าเป็น โอเปอเรชั่น (Operation) แต่การ โมเดลนี้ไม่ได้แสดงความสัมพันธ์ระหว่างข้อมูลและโอเปอเรชั่น

ส่วน โมเดลของการพัฒนาเชิงวัตถุนั้นจะแสดงความสัมพันธ์ระหว่างข้อมูลกับโอเปอเรชั่นที่กระทำ กับข้อมูลอย่างใกล้ชิด เนื่องจาก โมเดลนี้จำลองมาจากโลกจริงๆ การแยกแยะเอกลักษณะของการพัฒนาเชิงวัตถุ นั้น ให้ความเข้าใจและความสามารถที่ดีกว่า รวมทั้งมีการตั้งชื่อสำหรับวัตถุนั้นมาจากชื่อที่มีอยู่จริงใน โดเมน ของปัญหา มุมมองของวัตถุจะเป็นการมองในระดับสูงและใกล้ชิดกับโดเมนของปัญหามากกว่า

2.3.2 Seamless transition among different phases of software development การพัฒนาซอฟต์แวร์นั้น ต้องการวิธีการ (Methodology) ที่แตกต่างกันในแต่ละช่วงของการพัฒนา การ เปลี่ยนแปลงจากช่วงการทำงาน หนึ่งไปช่วงอื่นๆในแต่ละ Process จะมีความซับซ้อน และยังไม่มีความสอดคล้องกัน และยังซ้ำด้วย และยังทำ ให้ขนาดของงานใหญ่ขึ้น และมีโอกาสที่จะเกิดความผิดพลาดขึ้นด้วย แต่วิธีการของเชิงวัตถุนั้นจะใช้ภาษา เดียวกันในการติดต่อระหว่างการวิเคราะห์ การออกแบบ การเขียน โปรแกรม และการออกแบบฐานข้อมูลด้วย ทำให้ลดระดับของความซับซ้อนลง และช่วยให้การเข้าใจระบบก็ชัดเจนด้วย

2.3.3 Encouragement of good programming techniques เป็นการสนับสนุนเทคนิคการเขียนโปรแกรม ที่ดี เพราะการเขียนโปรแกรมเชิงวัตถุ นั้นจะเป็นการเขียนในลักษณะของคลาส ซึ่งบอกว่าคลาสนี้ทำอะไร และ

ทำอย่างไร และคุณสมบัติของระบบนั้นสามารถที่จะรวมคลาสเข้าด้วยกันเป็นระบบย่อยได้ และการเปลี่ยนแปลงคลาสหนึ่งก็จะไม่มีผลกระทบต่อคลาสอื่นๆด้วย อย่างไรก็ตามภาษาต่างๆเช่น C++, Smalltalk หรือ Java ก็เพิ่มส่วนที่สามารถรองรับการออกแบบเชิงวัตถุด้วยเพื่อง่ายต่อการสร้าง และการนำ Code มาใช้ใหม่กับแนวคิดของคลาสและการสืบทอด (Inheritance)

2.3.4 Promotion of Reusability ระบบแบบโครงสร้างมีข้อจำกัดในการนำสิ่งที่พัฒนาไปแล้วกลับมาใช้ใหม่ แต่ในความเป็นจริงแล้ว ถ้าผู้พัฒนาระบบมีคอมไพเลอร์ อยู่แล้วก็ควรจะใช้ได้เลย แต่ระบบแบบโครงสร้างนี้ผู้ที่นำมาใช้ต้องทำการเปลี่ยนแปลงซอร์สโค้ด (Source Code) เพื่อให้เหมาะกับความต้องการของระบบใหม่เพื่อที่จะใช้งานได้ แต่การพัฒนาเชิงวัตถุนี้มีคุณสมบัติที่สนับสนุนการนำกลับมาใช้ใหม่โดยตรงคือ Generalization และ Refinement โดย Generalization คือการสืบทอดคุณสมบัติโดยจะได้คุณสมบัติเดิมทุกประการและยังสามารถเพิ่มและขยายคุณสมบัติได้ด้วย โดยไม่ต้องมีการเปลี่ยนแปลงซอร์สโค้ดเท่าเลย ลักษณะการนำมาใช้ใหม่นี้เรียกว่า การเขียนโปรแกรมแบบแตกต่าง ส่วน Refinement นั้นก็คล้ายกับ Generalization แต่จะยอมให้นักพัฒนาใช้วัตถุที่ไม่สมบูรณ์โดยการเพิ่มส่วนที่หายไป ตามที่ตนเองต้องการได้

2.3.5 Promotion of Scalability ระบบโครงสร้างมีข้อจำกัดในการปรับเปลี่ยนขนาดของระบบ ทั้งนี้เนื่องจากระบบแบบโครงสร้างมี การแยกแยะเอกลักษณ์และการซ่อนรายละเอียดที่ไม่ดี ทำให้ระบบแบบโครงสร้างมีความซับซ้อนมากขึ้นเมื่อมีปัญหาที่เกิดขึ้นกับระบบมากขึ้น แต่ระบบที่พัฒนาด้วยวิธีเชิงวัตถุ นั้นจะแยกแยะเอกลักษณ์และการซ่อนรายละเอียดที่ดีส่งผลให้คอมไพเลอร์แต่ละส่วนแยกออกจากกันอย่างชัดเจน และการใช้เครื่องหมาย (Notation) เดียวกันตลอดการพัฒนา ทำให้สามารถเปลี่ยนจากการวิเคราะห์เป็นการออกแบบได้ง่าย การปรับเปลี่ยนระบบให้มีขนาดใหญ่ขึ้นก็สามารถทำได้ง่ายตามด้วย

2.3.6 Promotion of Reliability and Safety เนื่องจากการพัฒนาเชิงวัตถุมีการแยกแยะเอกลักษณ์และการซ่อนรายละเอียดที่ดีกว่าทำให้การสื่อสารระหว่างวัตถุที่ไม่เกี่ยวข้องกันจำกัดอยู่ด้วยอินเทอร์เฟส ที่ผู้พัฒนาได้กำหนดไว้เท่านั้น ส่งผลให้ระบบที่พัฒนาด้วยวิธีเชิงวัตถุมีความน่าเชื่อถือได้มากกว่าเพราะว่าผู้พัฒนาสามารถควบคุมการติดต่อระหว่างคอมไพเลอร์ที่ได้ และยังสามารถเพิ่มเงื่อนไขก่อนหน้าและท้าย (Pre and Post Condition) เพื่อให้ระบบสามารถทำงานได้อย่างถูกต้อง

2.3.7 Promotion of Concurrency การทำงานไปพร้อมๆกัน (Concurrency) เป็นรูปแบบการทำงานของสิ่งที่มีตัวตนในโลกแห่งความเป็นจริง และยังเป็นคุณสมบัติสำคัญของคอมพิวเตอร์ยุคใหม่ด้วย การพัฒนาแบบโครงสร้างไม่มีสัญลักษณ์สำหรับการทำงานพร้อมกัน การจัดการงาน (Task) หรือการซิงโครไนเซชัน (Synchronization) ระหว่างงานดังนั้น โครงสร้างที่สำคัญๆของระบบอาจจะไม่สามารถพัฒนาขึ้นได้โดยการใช้รูปแบบพัฒนาแบบ โครงสร้างมาตรฐานได้ แต่การพัฒนาระบบด้วยวิธีเชิงวัตถุสนับสนุนการทำงานพร้อมกัน ในตัวเอง และรายละเอียดของงาน การทำ Synchronization ระหว่างงานก็สามารถแสดงออกมาในแผนภาพ (Statecharts) ของวัตถุที่ทำงานแบบแอคทีฟ (Active Object) และการส่งแมสเสจ (Message) ระหว่างวัตถุ แผนภาพเหล่านี้เป็นเครื่องมือที่มีความเหมาะสมสำหรับการสร้างระบบที่มีเงื่อนไขทางด้านประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ความสามารถของคลาส

2.4.1 ลำดับชั้นของคลาส Class Hierarchy

2.4.1.1 Inheritance คือการสืบทอดคุณสมบัติเป็นการสร้างคลาสใหม่ขึ้นมา โดยมีพื้นฐานมาจากคลาสเดิม แต่จะมีข้อมูล (Data) หรือเมธอด ที่พิเศษเป็นของตัวเองเพิ่มขึ้นมาจากคลาสพื้นฐานเดิม ซึ่งก่อให้เกิดการแตกสายพันธุ์ของวัตถุ ขึ้นเป็นวัตถุใหม่ที่มีคุณสมบัติของวัตถุเดิมซ่อนอยู่ไม่มากก็น้อยนั่นเอง

ปัญหาสำคัญของการสืบทอดคุณสมบัติ อยู่ที่ที่ว่า ในการสร้างคลาสขึ้นมาใหม่แต่ละครั้งนั้น คลาสเหล่านั้นมีอะไรที่เป็นพื้นฐานเหมือนกันและมีคุณสมบัติอะไรที่แตกต่างกัน นักพัฒนาจะจำแนกคลาสเหล่านั้นได้อย่างไร จึงจะเหมาะสมกับพฤติกรรมการทำงานของมัน คำตอบก็คือ จะต้องเริ่มต้นมองที่ยอดบนของลำดับชั้นของคลาสดีก่อน คลาสที่อยู่สูงที่สุดจะต้องเป็นคลาสที่มีคุณสมบัติพื้นฐาน (Simplest) ของคลาสต่างๆ ให้มากที่สุด แล้วค่อยๆ แบ่งลำดับชั้นออกไปตามคุณสมบัติอื่นๆ ที่รองๆ ลงไป โดยเมื่อคุณสมบัติอย่างหนึ่งที่กำหนดลงในคลาสพื้นฐานแล้ว ทุกๆ คลาสที่สืบทอดคุณสมบัติภายใต้คลาสนั้นจะ ได้รับคุณสมบัติติดตัวไปด้วยอัตโนมัติ

2.4.1.2 Multiple Inheritance หรือ MI เป็นการสร้างคลาสขึ้นมา โดยการสืบทอดคุณสมบัติมาจากคลาสพื้นฐานที่มีมากกว่า 1 ตัว ทำให้ภาษาในลักษณะของการเขียน โปรแกรมเชิงวัตถุมีความยืดหยุ่นในการทำงานเพิ่มมากขึ้นอีกด้วย

2.4.2 แพ็กเกจ (Packages) เป็นการรวมคลาสที่มีความสัมพันธ์ใกล้ชิดกันเข้าไว้ในแพ็กเกจ เพื่อประโยชน์สำหรับระบบที่มีคลาสเป็นจำนวนมาก ในระบบที่มีความซับซ้อนเราอาจจะสร้างแพ็กเกจขึ้นมาก่อน แล้วจึงบรรจุคลาสและความสัมพันธ์เข้าไว้ในแพ็กเกจ

2.4.3 Encapsulation and Information Hiding เอ็นแคปซูลชัน คือ การรวมกันของโครงสร้างข้อมูล (Data Structure) กับฟังก์ชันที่เกี่ยวข้องหรือเรียกใช้ข้อมูลนั้น (เรียกว่าเมธอด หรือ Action) เกิดเป็นวัตถุตัวใหม่ที่สามารถทำการซ่อนข้อมูลจากภายนอกได้ ซึ่งจะเรียกวัดถุนั้นว่า คลาส นั้นเอง

ในการพัฒนาโดยวิธีแบบ โครงสร้างในสมัยก่อนนั้น ส่วนที่เป็นโครงสร้างข้อมูลและฟังก์ชันที่เรียกใช้ข้อมูลนั้นจะถูกสร้างขึ้นแยกจากกัน นั่นคือจะมองข้อมูลและฟังก์ชันเป็นโมดูลเดี่ยวๆ ไม่มีความสัมพันธ์กันแต่อย่างใด ข้อมูลชุดหนึ่งๆ อาจมีฟังก์ชันหลายๆ ตัวมาเรียกใช้งาน โดยตรง ซึ่งแต่ละฟังก์ชันอาจจัดการกับโครงสร้างของข้อมูลไม่เหมือนกัน การเข้าถึงข้อมูลโดยตรงของฟังก์ชันเหล่านี้จะนำไปสู่ปัญหาต่อไป

เอ็นแคปซูลชันก้าวมาในจุดนี้ ผลจากการที่นำโครงสร้างข้อมูลและฟังก์ชันที่ใช้งานข้อมูลนั้นมารวมกันสร้างความสัมพันธ์ให้กันและกัน จะทำให้ข้อมูลมีความมั่นคงขึ้น โดยการจะเข้ามาใช้ข้อมูลจะต้องเรียกผ่าน เมธอดเท่านั้น ผลก็คือ ไม่มีการเข้าถึงข้อมูลโดยตรงอีกต่อไป ซึ่งนั่นก็คือ เป็นการทำ Data Hiding นั้นเอง

ข้อดีอีกประการหนึ่งของเอ็นแคปซูลชัน ก็คือ เราสามารถแบ่งระดับของการเข้าถึงข้อมูลได้โดยตัวของเมธอดของวัตถุเอง

2.4.4 Polymorphism โพลิมอร์ฟิซึม หมายถึงการที่คลาสหนึ่งๆ สามารถเปลี่ยนไปได้หลายรูปร่างขึ้นอยู่กับสภาพแวดล้อมหรือสถานการณ์ในขณะนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจะเป็นการยอมให้ผู้พัฒนาใช้เมธอดชื่อเดียวกัน แต่การทำงานไม่เหมือนกัน จำแนกไปตามชนิดของข้อมูลหรือแอสเซจ(Message) ที่เมธอดจะได้รับขณะทำงาน เมื่อได้รับข้อมูลที่ต้องการแล้วก็จะไปเลือกเมธอดที่เหมาะสมที่สุดกับข้อมูลนั้นๆ ไปทำงานเองต่อไป

ความสามารถอีกอย่างหนึ่งของโพลิมอร์ฟิซึ่มก็คือ สามารถกำหนดการดำเนินการใหม่ให้กับตัวดำเนินการหรือเมธอดได้ เรียกว่าการทำ "Overloading" ยังผลให้เราสามารถเรียกชื่อเมธอดหรือตัวดำเนินการนั้น มาใช้งานกับวัตถุต่างๆ ในกรณีที่แตกต่างกัน ขึ้นอยู่กับว่าในขณะนั้น โปรแกรมกำลังทำงานอยู่กับวัตถุอะไร

2.4.5 ความสัมพันธ์

2.4.5.1 Generalization เป็นความสัมพันธ์แบบการให้กำเนิดคลาสย่อย (Subclass) จากคลาสแม่ (Superclass) โดยคลาสย่อยจะสืบทอดแอตทริบิวต์, โอเปอเรชันและความสัมพันธ์ทั้งหมดจากคลาสแม่ ทั้งของตัวเองและจากคลาสแม่ของคลาสแม่ขึ้นไปเรื่อยๆ คลาสย่อยอาจมีการเพิ่มแอตทริบิวต์และโอเปอเรชันในคลาสของมันเอง ความสัมพันธ์แบบนี้อาจถูกเรียกว่า is-a หรือ kind-of

2.4.5.2 Aggregation เป็นความสัมพันธ์แบบเป็นส่วนประกอบหรือที่เรียกว่า Part-of คำว่าเป็นส่วนประกอบบางครั้งอาจมีความหมายที่ต่างกัน เช่น เครื่องยนต์และล้อเป็นส่วนประกอบของรถยนต์ในขณะที่เรากำลังขับรถอยู่ แต่ถ้ารถยนต์คันนั้นอยู่ในระหว่างการซ่อมที่อู่ซ่อมรถ นั่นคือเครื่องยนต์และล้อก็ไม่ได้เป็นส่วนประกอบของรถยนต์แล้ว

2.4.5.3 Association เป็นความสัมพันธ์แบบสองทิศทาง ซึ่งใช้ในการติดต่อระหว่างคลาส ความสัมพันธ์แบบ แอสโซซิเอชันระหว่างคลาสหมายถึง การเชื่อมต่อระหว่างออบเจกต์ในคลาสทั้งสอง ที่ปลายของเส้นความสัมพันธ์จะมีการระบุชื่อบทบาทเพื่อบอกว่าคลาสนั้นถูกมองว่าเป็นอย่างไรจากคลาสอื่น

2.4.5.4 Constraints เป็นการกำหนดข้อบังคับของความสัมพันธ์ระหว่างเอนติตี้ โดยเอนติตี้จะหมายถึงออบเจกต์, คลาส, บทบาท, แอตทริบิวต์, การเชื่อมต่อ (Link) และความสัมพันธ์ ข้อบังคับนี้จะเป็นการจำกัดค่าที่เอนติตี้สามารถเป็นไปได้

2.5 Object Oriented Methodologies

วิธีการพัฒนาระบบเชิงวัตถุนี้มีมากมายหลายวิธี โดยแต่ละวิธีก็มีจุดเด่นและจุดด้อย แตกต่างกันไป โดยผู้พัฒนาควรจะใช้วิธีการที่มีประสิทธิภาพที่ดีที่สุด เพื่อระบบที่ดีที่สุด โดยจะใช้วิธีผสมนำข้อดีของแต่ละวิธีมาใช้ก็ได้ดังเช่น Unified Modeling Language (UML) ที่นำข้อดีของ Rumbaugh et al., Booch และ Jacobson มาใช้ และยังสามารถใช้เทคนิคต่างๆ ที่เกิดขึ้นมาพัฒนาด้วยเช่น Patterns และ Frameworks เป็นต้น

2.5.1 Rumbaugh et al.'s Object Modeling Technique (OMT) ถูกนำเสนอโดย Jim Rumbaugh และผู้ร่วมงาน โดยจะบอกถึงวิธีการวิเคราะห์ ออกแบบ และ ลงมือทำ ของระบบโดยวิธีเชิงวัตถุ OMT จะมีรายละเอียดต่างๆ เช่น คลาส, แอตทริบิวต์, เมธอด, การสืบทอดคุณสมบัติและแอสโซซิเอชันสามารถแสดงได้อย่างง่าย โดย OMT จะประกอบไปด้วย 4 ขั้นตอนต่อไปนี้ และในแต่ละขั้นตอนสามารถกลับมาทำซ้ำอีกหลายครั้งได้ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การวิเคราะห์ (Analysis)
- ออกแบบระบบ (System design)
- ออกแบบวัตถุ (Object design)
- การสร้าง (Implementation)

OMT แบ่งแยก model ต่างๆออกเป็น 3 ส่วนดังต่อไปนี้คือ

- Object model สามารถนำเสนอได้โดยใช้โมเดลวัตถุและการอธิบายข้อมูล(Data dictionary)
- Dynamic model สามารถนำเสนอได้โดยใช้สเตทไดอะแกรมและอีเวนทไฟลไดอะแกรม
- Functional model สามารถนำเสนอได้โดยใช้ไดอะแกรมโพลไดอะแกรมและข้อบังคับต่างๆ

2.5.2 The Booch Methodology เป็นที่นิยมใช้กันอย่างกว้างขวางโดยวิธีเชิงวัตถุซึ่งช่วยให้ออกแบบระบบที่ใช้วัตถุ เป็นแบบอย่างได้ โดยครอบคลุมไปถึงการวิเคราะห์และการออกแบบ วิธีการของ Booch ประกอบไปด้วย ไดอะแกรมต่างๆต่อไปนี้

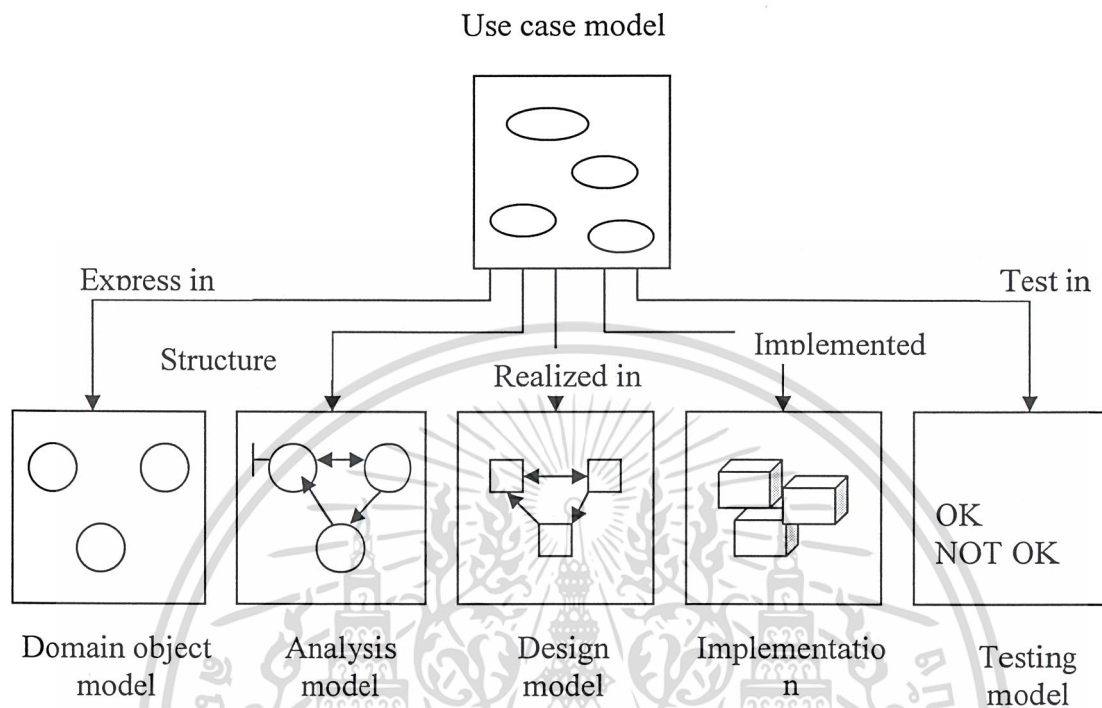
- คลาสไดอะแกรม
- ออบเจกต์ไดอะแกรม
- สเตททรานซิชันไดอะแกรม
- โมดูลไดอะแกรม
- โพรเซสไดอะแกรม
- อินเตอร์แอกชันไดอะแกรม

2.5.3 The Jacobson et al. Methodologies หัวใจหลักของพัฒนาระบบ วิธีนี้คือ หลักการซึ่งทำให้เกิด Objectory ขึ้นมา

2.5.3.1 Object Oriented Software Engineering: เรียกว่า Objectory ซึ่งเป็นวิธีการพัฒนาระบบด้วยวิธีเชิงวัตถุ ซึ่งสร้างโมเดลที่แตกต่างกันดังนี้

- ยูสเคสโมเดล (Use case Model) ซึ่งกำหนดสิ่งที่มีกระทำกับระบบ (Actors) และมาใช้ระบบอย่างไร
- โดเมนออบเจกต์โมเดล (Domain object Model)เป็นการแมปออบเจกต์ที่มีอยู่จริงให้ เป็นไปตามโดเมนของระบบ
- โมเดลการวิเคราะห์ (Analysis object Model) เป็นการนำเสนอว่าควรสร้างระบบอย่างไร มีการส่งข้อมูลอย่างไรบ้าง
- โมเดลของการสร้าง (Implementation model) เป็นการแสดงการสร้างระบบ
- โมเดลการทดสอบ (Test model) เป็นการวางแผนการทดลองระบบที่สร้างขึ้น และทำรายงานต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



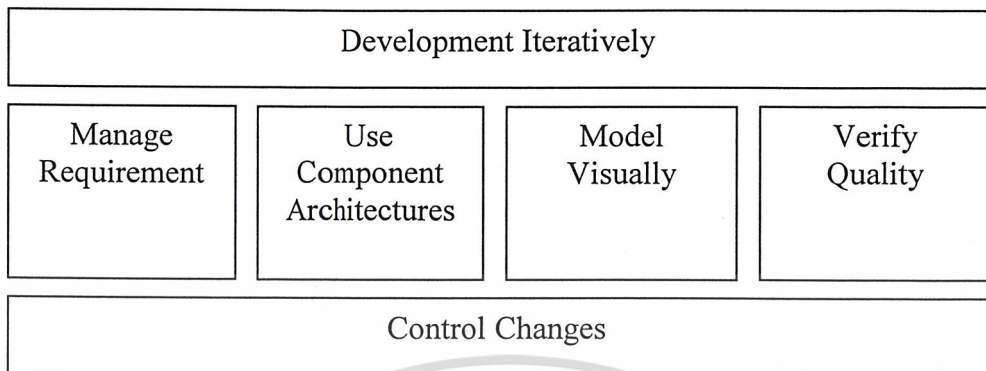
รูปที่ 2-1 การเชื่อมโยงของ Use Case Model กับ Model อื่น ๆ

2.5.3.2 Object Oriented Business Engineering (OOBE) เป็นวิธีการเชิงวัตถุที่มีระดับความเสี่ยง โดยใช้ยูสเคสเป็นตัวหลักในการทำโมเดลต่างๆ ตาม Software engineering process โดยแบ่งเป็นขั้นตอนต่างๆ ดังนี้

- Analysis phase เป็นการวิเคราะห์และกำหนดว่าระบบสร้างขึ้นมาเพื่อทำอะไร ตามโมเดลของโดเมนปัญหาของระบบ (Domain object model), โมเดลของความต้องการ (Requirement model) และ โมเดลการวิเคราะห์ (analysis model)
- Design and implementation phase เป็นการกำหนดสิ่งที่จำเป็นต่างๆ ในโมเดลของการออกแบบ (Design model) ซึ่งจะรวมแพ็คเกจต่างๆ ไว้ เช่น ระบบการจัดการฐานข้อมูล (DBMS), การกระจายโปรเซสการเขียนโปรแกรมการใช้เครื่องมือ (Tools) ต่างๆ เป็นต้น
- Testing phase สุดท้ายเป็นการทดสอบระบบโดย Jacobson ได้แสดงไว้หลายวิธีในระดับต่างๆ คือการทดสอบในระดับยูนิต (unit testing), การทดสอบในระดับการทำงานร่วมกัน (Integrate testing) และ การทดสอบระบบ(system testing)

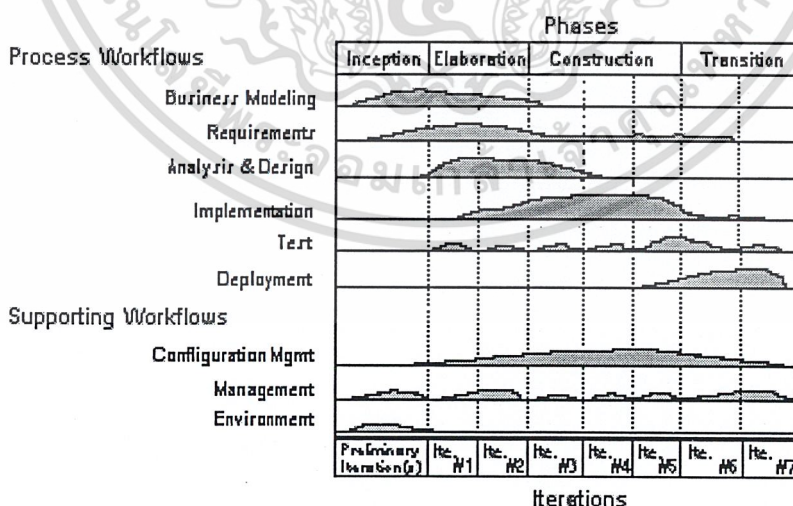
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 กระบวนการพัฒนาซอฟต์แวร์(Software Process) ที่ดี



รูปที่ 2-2 Best Practices of Software Engineering

2.6.1 การพัฒนาแบบวนรอบ (Development Iteratively) จากรูปจะเห็นได้ว่า ส่วนนี้จะนำไปใช้กับ ส่วนของ Manage Requirement ,Use Component Architecture ,Model Visually และ Verify Quality ดังนั้นการพัฒนาแบบวนรอบนี้ ถือว่าเป็นกระบวนการที่สำคัญกระบวนการหนึ่งทีเดียว คุณลักษณะของการพัฒนา วนรอบคือ นักพัฒนาจะสามารถวิเคราะห์ และแก้ปัญหาที่มีความเสี่ยงสูง ก่อนที่จะมีการลงทุนพัฒนาระบบ การพัฒนาแบบวนรอบสามารถได้รับความต้องการอย่างต่อเนื่องจากผู้ใช้ได้ นอกจากนี้ การทดสอบและการรวมระบบจะค่อยเป็นค่อยไปอย่างต่อเนื่อง การมี milestone ทำให้นักพัฒนาทราบว่าตอนนี้การพัฒนากำลังเน้นไปที่จุดใด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ควรเผยแพร่ภายนอกองค์กรโดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป แสดงให้เห็นถึงการประยุกต์ใช้การพัฒนาแบบวนรอบ สำหรับในการพัฒนารอบแรกนั้นจะใช้เวลาไปกับกระบวนการ Business Modeling มากที่สุด ในรอบต่อไปก็จะใช้เวลาการศึกษา Requirement มากที่สุด แต่ในทุกรอบการทำงานก็จะทำในกระบวนการทุกกระบวนการ แต่จะมากน้อยแตกต่างกันไปตามรอบของการพัฒนา

2.6.2 การจัดการความต้องการของระบบ (Requirement Management) เป็นกระบวนการที่เป็นระบบในการค้นหา จัดโครงสร้าง และทำเอกสารความต้องการของระบบ และเป็นกระบวนการในการจัดตั้ง ดูแล ข้อตกลงระหว่างลูกค้า หรือผู้ใช้ กับทีมพัฒนาของโครงการในการเปลี่ยนแปลงความต้องการของระบบ

2.6.3 การใช้สถาปัตยกรรมคอมโพเน้น(Use Component Architectures) สถาปัตยกรรมซอฟต์แวร์เป็นโครงสร้างหลักของระบบซอฟต์แวร์และเป็นส่วนหนึ่งที่ได้จากการออกแบบระบบซึ่งเกี่ยวข้องกับโครงสร้างและพฤติกรรมของระบบในด้านของการใช้งาน หน้าที่การทำงาน ประสิทธิภาพ และเงื่อนไขของเทคโนโลยี กระบวนการนี้ จะทำให้สถาปัตยกรรมซอฟต์แวร์มีพื้นฐานอยู่บน หลักการของ Component เป็นหลัก ซึ่งจะทำให้สถาปัตยกรรมที่ได้นั้นมีความยืดหยุ่นและทำให้สามารถดูแลรักษา เพิ่มเติมระบบได้ง่าย และยังสามารถนำกลับมาใช้ได้

2.6.4 แบบจำลองรูปภาพ (Model Visually) - เป็นการนำเอาโมเดลที่เป็นรูปภาพมาช่วย ซึ่งทำให้สามารถเก็บโครงสร้างและพฤติกรรมของสถาปัตยกรรม แสดงส่วนประกอบในระบบ หรือแสดงรายละเอียดของงาน โดยมีการดูแลรักษาความตรงกันของการออกแบบและการพัฒนา และยังช่วยลดความกำกวมในการสื่อสาร ส่วนนี้จะเป็นส่วนที่เราจะพูดถึง UML กัน ซึ่งรายละเอียดจะกล่าวต่อไป

2.6.5 การทดสอบคุณสมบัติ (Verify Quality) การพัฒนาแบบวนรอบนั้นทำให้นักพัฒนาสามารถทดสอบระบบได้ดียิ่งขึ้น ทั้งนี้แทนที่จะทำการทดสอบระบบครั้งใหญ่ครั้งเดียวก่อนที่จะส่งให้กับผู้ใช้ ด้วยการพัฒนาแบบวนรอบจะทำให้มีการทยอยทดสอบในแต่ละรอบ ส่งผลให้ระบบในรอบต่อไปมีคุณภาพมากยิ่งขึ้น

2.6.6 การควบคุมความเปลี่ยนแปลง (Control Changes) หลักการในการจัดการความเปลี่ยนแปลงก็คือจะต้องมีการแยกสถาปัตยกรรมของระบบออกเป็นระบบย่อยๆ และมีการกำหนดความรับผิดชอบของระบบย่อยให้กับทีมพัฒนาหนึ่ง มีการจัดตั้งพื้นที่ทำงานสำหรับนักพัฒนาแต่ละคน มีการกำหนดการรวมกันของแต่ละพื้นที่ทำงาน มีการกำหนดวิธีการควบคุมความเปลี่ยนแปลง มีการบันทึกความเปลี่ยนแปลงในแต่ละรุ่น มีการทดสอบในแต่ละรอบของการพัฒนาระบบ

ระบบที่ถูกดองนั้น จะมีการจัดการความต้องการความต้องการพื้นฐาน จากนั้นจึงเปรียบเทียบ ความเปลี่ยนแปลงที่เกิดขึ้นโดยใช้สถาปัตยกรรมคอมโพเน้น นักพัฒนาสามารถที่จะตรวจสอบรุ่นของ คอมโพเน้นที่จะใช้โมเดลด้วยรูปภาพ ทำให้สามารถตรวจสอบความเปลี่ยนแปลงได้ง่าย ผลที่ได้จากการตรวจสอบจะต้องสามารถติดตามได้ในแต่ละรุ่นของระบบ จากนั้นจะพัฒนายวนรอบต่อไป

จากรายละเอียดดังที่ได้กล่าวข้างต้น การสื่อสารระหว่างนักพัฒนา นับเป็นปัญหาหนึ่งในการพัฒนาระบบซอฟต์แวร์ใหญ่ๆ ดังนั้นในกระบวนการการพัฒนาซอฟต์แวร์ที่ดีนั้นจึงนำเอา แบบจำลองรูปภาพ(Visual Modeling) มาใช้ในการแสดงรายละเอียดของงาน แทนที่จะใช้ข้อความในการอธิบายเหมือนระบบในอดีต ซึ่งการใช้ข้อความนั้นอาจจะทำให้ความหมายนั้นกำกวม ต่างกับรูปภาพที่แทนคอมโพเน้น ในระบบจริงๆ ที่เอกสารเป็นเอกสารที่ส่งงานไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้เข้าไปใช้ประโยชน์ด้านการค้า สามารถสื่อความหมายได้ชัดเจน

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.7 แบบจำลองรูปภาพ (Visual Modeling Technology: VMT) VMT คือแนวทางหนึ่งของการพัฒนาระบบ โดยการออกแบบระบบด้วยโมเดลตามแนวความคิดของโลกแห่งความเป็นจริง โมเดลที่ถูกออกแบบมานั้น จะมีประโยชน์ช่วยในการเข้าใจถึงปัญหาของระบบ, การติดต่อกันของผู้ที่จัดทำโครงการ (เช่น ลูกค้า, ผู้ดูแลระบบ, นักวิเคราะห์ และนักออกแบบ เป็นต้น) , การเตรียมการทำเอกสาร ,การออกแบบโปรแกรมและดาตาเบส

โมเดลเป็นนามธรรมที่แสดงถึงลักษณะที่สำคัญของปัญหาหรือโครงสร้างของระบบ โดยการกลั่นกรองรายละเอียดที่ไม่สำคัญออก ดังนั้นจึงเป็นการทำให้ปัญหานั้นง่ายต่อการเข้าใจ เราจึงควรสร้างโมเดลสำหรับระบบที่ซับซ้อน เพราะความสามารถของคนที่จะเข้าใจระบบที่ซับซ้อนนั้นมีจำกัด การสร้างโมเดลจะทำให้นักออกแบบสามารถเห็นภาพกว้างของโครงการได้ โมเดลที่ดีควรจะใช้สัญลักษณ์ที่ถูกต้องและมีการตรวจสอบโมเดลให้ตรงความต้องการของระบบ

Visual Modeling Technique จะมีส่วนเกี่ยวข้องกับทุกขั้นตอนของการพัฒนาโปรแกรมเชิงวัตถุทั้งการรวบรวมความต้องการ(requirement gathering), การวิเคราะห์ (analysis), การออกแบบ (design), การเขียนโปรแกรม (coding) และการทดสอบ (testing)

2.7 ส่วนประกอบของ UML

UML ประกอบด้วยหลายๆส่วนดังนี้

View : แสดงมุมมองต่างๆ ของระบบที่ออกแบบขึ้นมา โดยการอธิบายรายละเอียดในแต่ละส่วนใน View จะใช้ ไดอะแกรม ที่มีอยู่ใน UML อธิบาย

Diagram : เป็นไดอะแกรมที่ใช้ในการอธิบายส่วนต่างๆของ View

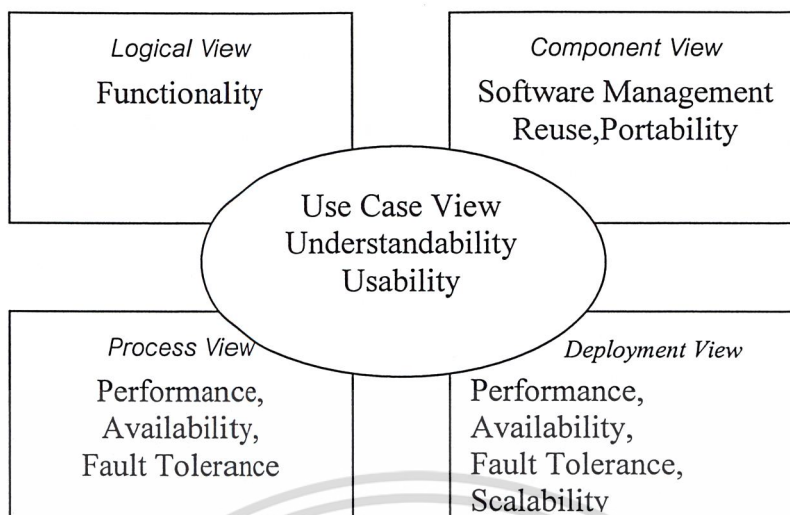
Model element : เป็นสัญลักษณ์ที่ใช้ในไดอะแกรมเพื่อแสดงหรือเป็นตัวแทนของสิ่งต่างๆ เช่น คลาส (class) ,ออปเจ็กต์ ,เมสเสจ(message) และ ความสัมพันธ์(relationship) เป็นต้น

General mechanism : เป็นส่วนที่แสดงคอมเม้นเพิ่มเติม (extra comment) , ข้อมูลอื่นๆ ที่จำเป็นหรือความหมายของโมเดลเอลิเมนต์ (model element)

2.7.1 View ในการออกแบบระบบที่มีขนาดใหญ่และมีความซับซ้อนมากขนาดนั้นจะทำให้ผู้ออกแบบระบบไม่สามารถที่จะออกแบบระบบได้ครบถ้วน ดังนั้นจึงต้องมีการมองระบบเป็นมุมมองต่างๆ เพื่อทำให้ง่ายในการออกแบบ เช่น มุมมองด้านฟังก์ชันนอล , นอนฟังก์ชันนอล(Nonfunctional), มุมมองขององค์กรเป็นต้น ดังนั้นระบบจึงมี View ที่ต่างๆ กันซึ่งแต่ละ view จะแสดงมุมมองเฉพาะของระบบซึ่งอธิบายรวมกันเป็นระบบที่สมบูรณ์ ซึ่งจะประกอบด้วย View ต่างๆ ดังนี้

2.7.1.1 Use-case View อธิบายการทำงานต่างๆ ของระบบที่ถูกมองจากภายนอกหรือผู้ใช้ระบบ Use-case view ซึ่งอธิบายโดยยูสเคสไดอะแกรม(use-case diagram) เป็นมุมมองสำหรับลูกค้า , ผู้ออกแบบ ,ผู้พัฒนาระบบและผู้ทดสอบระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-4 สถาปัตยกรรมของ View

2.7.1.2 Logical view อธิบายการทำงานต่างๆ ที่ถูกออกแบบไว้ภายในระบบ ว่าระบบจะมีบริการอะไรให้กับผู้ใช้งาน โดยจะแสดง โครงสร้างแบบสถิต (Static) เช่น คลาส, ออบเจกต์, ความสัมพันธ์และการทำงานร่วมกันแบบไดนามิก (dynamic collaboration) ซึ่งจะเกิดขึ้นเมื่อออบเจกต์ ส่งแอสเซส ระหว่างกันในการทำงาน

โครงสร้างแบบสถิตจะอธิบายโดยใช้ คลาสไดอะแกรม (Class diagram) และออบเจกต์ไดอะแกรม (object diagram) ส่วนการทำงานร่วมกันแบบไดนามิกจะอธิบายโดยใช้ สเตทไดอะแกรม (state diagram) และแอกติวิตีไดอะแกรม (activity diagram)

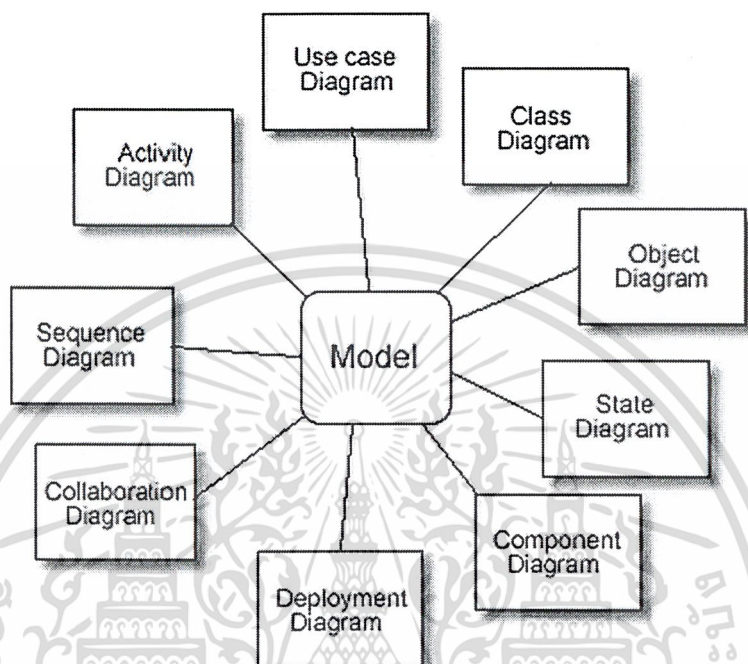
2.7.1.3 Component View อธิบายการสร้างและสร้างความขึ้นต่อกันของ โมดูล (Module) ที่ใช้ในการพัฒนาระบบ โดยใช้คอมโพเนนต์ไดอะแกรม (Component diagram) ในการอธิบาย

2.7.1.4 Deployment View อธิบายการจัดวางระบบให้เหมาะในด้านกายภาพ (Physical) แสดงด้วยคอมพิวเตอร์และโหนด (nodes) ต่างๆ เพื่อให้ระบบมีเสถียรภาพมากขึ้น โดยใช้ดีพลอยเมนต์ไดอะแกรม (deployment diagram) ในการอธิบาย

2.7.1.5 Process View แสดงการทำงานร่วมกันและการติดต่อกันของส่วนต่างๆ ในระบบ

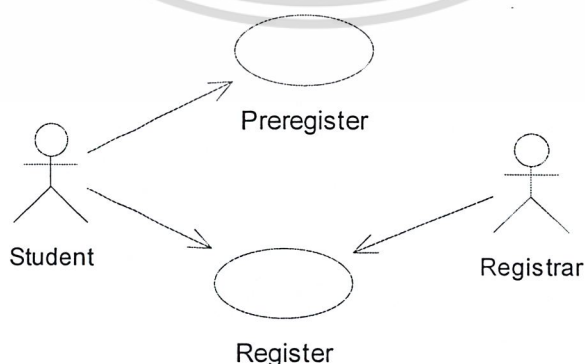
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2 Diagram เป็นไดอะแกรมซึ่งแสดงสัญลักษณ์ที่ถูกจัดเรียงเพื่อ อธิบายระบบในมุมมองต่างๆ ซึ่งในระบบหนึ่งๆ จะประกอบด้วยหลายๆ ไดอะแกรม และในแต่ละไดอะแกรมยังสามารถมองในหลายๆ มุมมองได้ UML มีไดอะแกรมที่ต่างกันอยู่ดังนี้



รูปที่ 2-5 แสดงไดอะแกรมต่างๆ ของ UML

2.7.2.1 Use Case Diagram ยูสเคสไดอะแกรมจะแสดงความสัมพันธ์ระหว่างผู้ใช้งาน (User) กับระบบ โดยใช้แอ็กเตอร์ (actors) แทนผู้ใช้งานและแอ็กเตอร์จะต้องติดต่อกับระบบโดยผ่าน ยูสเคส(Use case) ต่างๆ ซึ่งยูสเคสใน Use case diagram ก็คือการทำงานต่างๆ ของระบบที่ผู้ใช้ต้องการ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ทุกรูปที่ 2-6 ตัวอย่าง Use Case ในระบบลงทะเบียน
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แเอ็กเตอร์ไม่ใช่ส่วนประกอบของระบบ แต่จะเป็นส่วนที่ใช้ติดต่อกับระบบ ซึ่งอาจจะเป็นเพียงการป้อนข้อมูลเข้าไปในระบบหรือเป็นเพียงการรับข้อมูลจากระบบ หรืออาจจะเป็นทั้งสองอย่างก็ได้ ในระบบหนึ่งๆ จะต้องมีแเอ็กเตอร์ซึ่งจะต้องทำการกำหนดคุณลักษณะของแเอ็กเตอร์ (Identify actor) ให้ได้ก่อนดังนี้

- ใครเป็นผู้ใช้งานระบบ?
- ใครที่มีความเหมาะสมที่จะใช้งานระบบ?
- ใครที่มีส่วนสนับสนุนข้อมูลในระบบ ใช้ข้อมูลในระบบและแก้ไขข้อมูลในระบบ?
- ใครที่สนับสนุนการบำรุงรักษาระบบ (Maintenance)?
- ระบบมีการใช้งานกับภายนอกหรือเปล่า?

2.7.2.2 Class Diagram คลาสไดอะแกรมเป็นไดอะแกรมที่มีความสำคัญมากที่สุดในการวิเคราะห์และออกแบบโดยใช้วิธีเชิงวัตถุ เนื่องจากคลาสไดอะแกรมจะแสดงโครงสร้างของวัตถุและคลาสที่มีในระบบรวมทั้งแสดงความสัมพันธ์ด้วย คลาสไดอะแกรมจะเป็นโครงสร้างหลักของระบบและยังใช้ในการแยกย่อยรายละเอียด (Decomposite) ด้วยวิธีเชิงวัตถุ

สัญลักษณ์ความสัมพันธ์ต่างๆ ของคลาสไดอะแกรม มีดังนี้

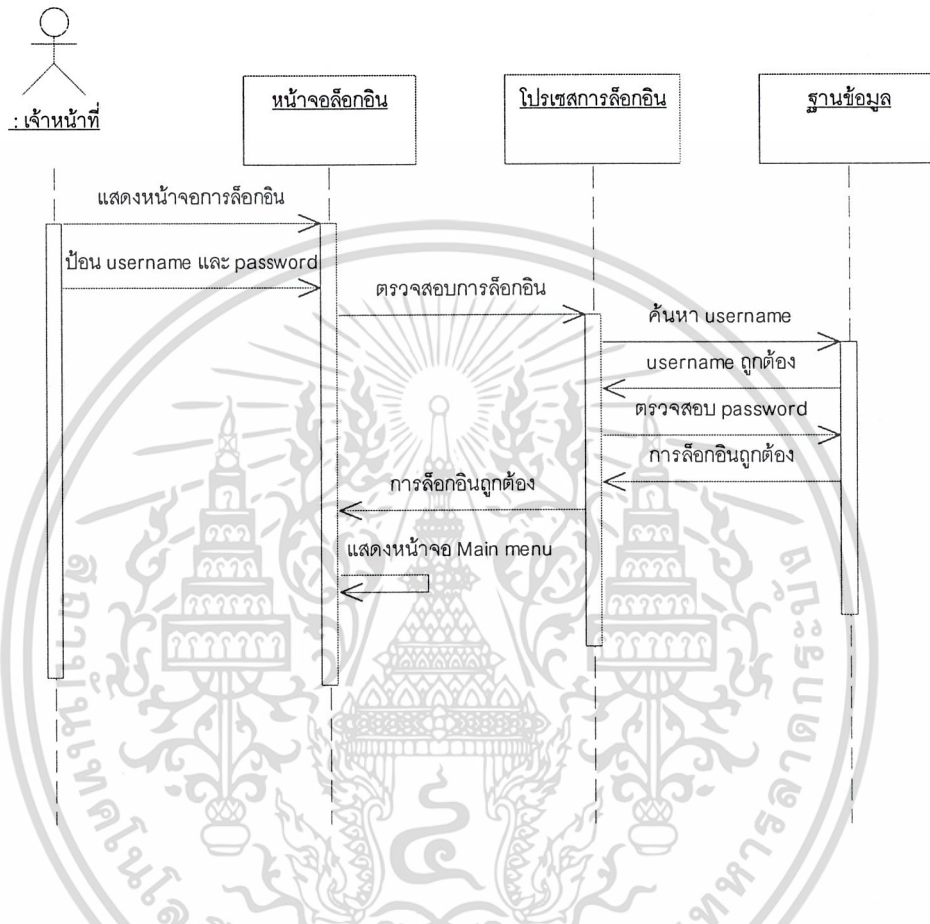
- | | |
|------|--|
| 1 | หมายถึงจะมีออบเจกต์ในคลาสไดอะแกรมได้หนึ่งออบเจกต์เท่านั้น |
| 0..1 | หมายถึงจะมีออบเจกต์ในคลาสไดอะแกรมได้แค่หนึ่งอาจจะไม่มีก็ได้ |
| M..N | หมายถึงจะมีออบเจกต์ในคลาสไดอะแกรมได้ตั้งแต่ M ถึง N (เมื่อ M และ N เป็นจำนวนเต็มบวก) |
| * | หมายถึงจะมีออบเจกต์ในคลาสไดอะแกรมได้ตั้งแต่ศูนย์ขึ้นไป |
| 0..* | หมายถึงจะมีออบเจกต์ในคลาสไดอะแกรมได้ตั้งแต่ศูนย์ขึ้นไป |
| 1..* | หมายถึงจะมีออบเจกต์ในคลาสไดอะแกรมได้ตั้งแต่หนึ่งขึ้นไป |

2.7.2.3 Object Diagram ออบเจกต์ไดอะแกรมจะได้มาจากคลาสไดอะแกรม ซึ่งจะเป็นการแสดงการเชื่อมต่อระหว่างอินชแทนซ์ (Instances) ต่างๆ ที่อยู่ภายในคลาสไดอะแกรมอย่างละเอียด เพื่อใช้เป็นตัวอย่างสำหรับคลาสไดอะแกรมที่มีความซับซ้อน โดยจะนำมาแทนเป็น ออบเจกต์จริงๆ และแสดงความสัมพันธ์ต่างๆ ให้เห็น โดยสัญลักษณ์ของอินชแทนซ์ในออบเจกต์ไดอะแกรม จะใช้การขีดเส้นใต้เป็นสัญลักษณ์ ภายในกรอบสี่เหลี่ยม

2.7.2.4 Collaboration Diagram คีอแลโบลเรนซ์ไดอะแกรมใช้อธิบายการทำงานร่วมกันของออบเจกต์ที่มีการส่งแมสเสจระหว่างกัน เพื่อบอกถึงลำดับขั้นตอนการทำงานของระบบ โดยใช้สัญลักษณ์ลูกศรเป็นตัวกำหนดทิศทางซึ่ง คีอแลโบลเรนซ์ไดอะแกรมจะเป็นส่วนอธิบายการทำงานของออบเจกต์ไดอะแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2.5 Sequence Diagram ซีควีนซ์ไดอะแกรมแสดงการติดต่อโต้ตอบระหว่างออบเจกต์ที่เน้นไปที่ลำดับก่อนหลังและเวลามีความสำคัญในการเปลี่ยนแปลง จะมีการส่งเมสเสจ เพื่อให้ออบเจกต์ต่างๆ ที่อยู่ในระบบทำงานโดยใช้สัญลักษณ์ที่เรียกว่า ออบเจกต์เมสเสจ ซีควีนซ์ชาท (Object message sequence chart) ในการแสดงผังรูป



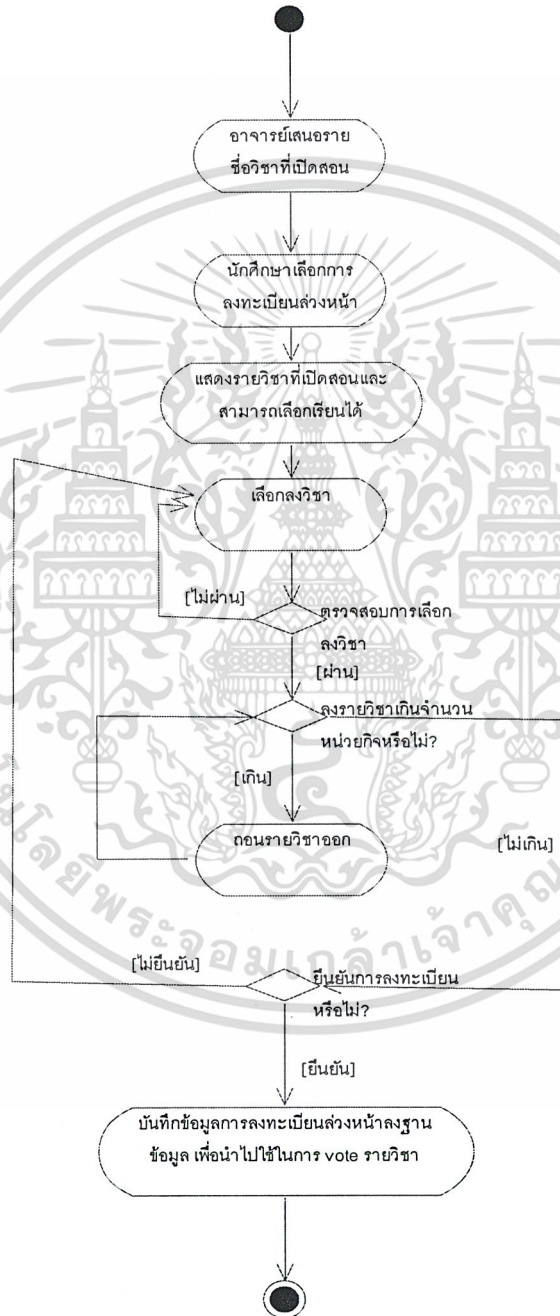
รูปที่ 2-7 แสดงตัวอย่าง ซีควีนซ์ไดอะแกรม

2.7.2.6 State Diagram สตทไดอะแกรมใช้อธิบายคลาสต่างๆ ในระบบโดยจะแสดงทุกๆ สถานะที่เป็นไปได้และเหตุการณ์ทำให้ออบเจกต์เหล่านั้นเกิดการเปลี่ยนแปลง โดยเหตุการณ์ที่ทำให้เกิดการเปลี่ยนแปลงอาจเกิดจากออบเจกต์อื่นส่งเมสเสจมา การเปลี่ยนแปลงสถานะเรียกว่า ทรานซิชัน (Transitions)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2.7 Activity diagram แอคติวิตี้ไดอะแกรมแสดงลำดับการไหลของกิจกรรม(activity)

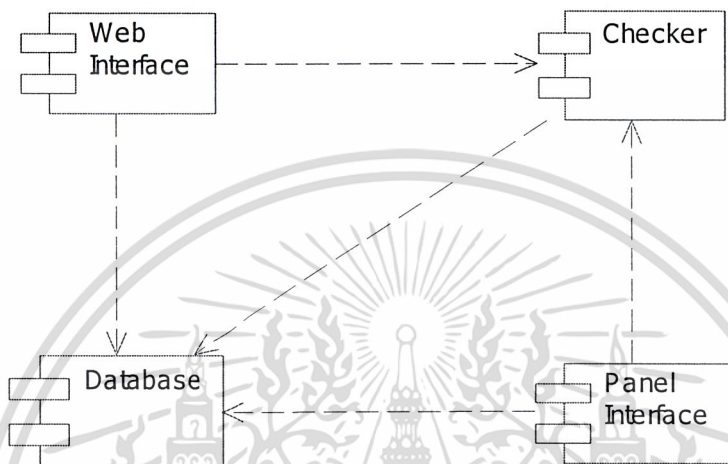
ต่างๆ โดยจะอธิบายกิจกรรมต่างๆ ในลักษณะของการกระทำในไดอะแกรม จะแสดงเป็นสถานะการกระทำ (Action State) ซึ่งสถานะจะเปลี่ยนไปเมื่อเกิดการกระทำอย่างใดอย่างหนึ่งเกิดขึ้นตามเงื่อนไขหรือการตัดสินใจที่กำหนดไว้เพื่อควบคุมการไหลของกิจกรรมรวมถึงสามารถมีแมสเชสที่รับ – ส่งระหว่างแต่ละกิจกรรม



รูปที่ 2-8 แสดงตัวอย่างแอคติวิตี้ไดอะแกรม

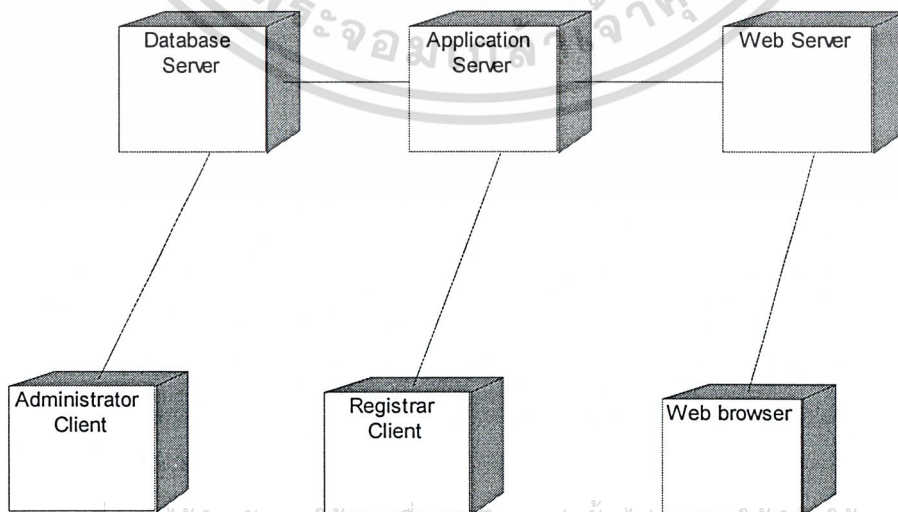
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2.8 Component diagram คอมโพเนนต์ในไดอะแกรมแสดงโครงสร้างทางกายภาพของโค้ดในรูปค็อมโพเนนต์ของโค้ด (Code) อาจเป็นส่วนประกอบของ ซอสเมื่อเกิดการเปลี่ยนแปลงของ คอมโพเนนต์หนึ่งจะมีผลต่อคอมโพเนนต์อื่นๆ อย่างไร ซึ่งช่วยในการ โปรแกรม โค้ด (source code) คอมโพเนนต์จะประกอบด้วยข้อมูลเกี่ยวกับ ลอจิกคัล คลาส (logical class) ของคอมโพเนนต์ในไดอะแกรม มีการแสดงความสัมพันธ์หรือความพึ่งพากันของคอมโพเนนต์ให้ช่วยในการวิเคราะห์หว่า



รูปที่ 2-9 แสดงตัวอย่างคอมโพเนนต์ในไดอะแกรม

2.7.2.9 Deployment diagram ดีพลอยเม้น ไดอะแกรมแสดงสถาปัตยกรรมทางกายภาพของส่วนประกอบต่างๆ ของฮาร์ดแวร์ (hardware) ซึ่งถูกเรียกว่า โหนด(Node) ในระบบซึ่งสามารถแสดงเป็นโหนดและการเชื่อมต่อระหว่างชนิดของการเชื่อมต่อ นอกจากนี้ภายในโหนด ยังสามารถมีคอมโพเนนต์ หรือออบเจกต์ที่สามารถปฏิบัติกับโหนดเพื่อแสดงว่าโปรแกรมส่วนใดถูกปฏิบัติบนโหนดใดและความสัมพันธ์ระหว่างคอมโพเนนต์ที่อยู่บนโหนด ซึ่ง โหนดจะถูกแทนด้วยสัญลักษณ์ของสี่เหลี่ยมลูกบาศก์ดังรูป



รูปที่ 2-10 แสดงตัวอย่างดีพลอยเม้นไดอะแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

Servlet, JSP และ Java Bean

3.1 เซิร์ฟเล็ต (Servlet)

Servlet คือเทคโนโลยีที่เกิดขึ้นมาจาก Java เพื่อใช้งานทางด้านการสร้าง โปรแกรมในรูปแบบ CGI สำหรับทำงานบนเว็บเซิร์ฟเวอร์ เพื่อให้ผลการแสดงเว็บผู้ใช้งาน มีการเปลี่ยนแปลงตามการใช้งาน (On The Fly) ซึ่งเป็นเทคโนโลยีที่สืบเนื่องมาจากเหตุผลดังนี้

- ความต้องการให้หน้าเว็บเปลี่ยนตามข้อมูลที่ผู้อ่านส่งให้ เช่น เว็บที่ให้บริการระบบค้นหา (Search Engine)ทั้งหลาย ที่ได้รับข้อมูลจากผู้ใช้ คือข้อความที่ต้องการค้นหา และแสดงผลลัพธ์หน้าเว็บ ตามข้อมูลที่ค้นหาเจอแก่ผู้ใช้
- ความต้องการให้หน้าเว็บเปลี่ยนตามข้อมูลที่มีการเปลี่ยนแปลงบ่อยครั้ง เช่น เว็บที่เกี่ยวข้องกับการรายงานข่าว หรือสภาพอากาศ จะเห็นว่าข้อมูลที่รายงานสู่ผู้ใช้ ดังกล่าว มีการเปลี่ยนแปลงอยู่ตลอดเวลา อาจเป็นรายชั่วโมง หรือต่อนาทีก็ได้
- ความต้องการให้หน้าเว็บเปลี่ยนตามระบบฐานข้อมูล เช่น ระบบควบคุมคลังสินค้า ที่อาศัยการเก็บข้อมูลในระบบฐานข้อมูล สามารถแสดงผลจำนวนสินค้า ตามข้อมูลที่มีอยู่จริงในฐานข้อมูล ออกสู่ผู้ใช้บนระบบเว็บ

3.1.1 ข้อแตกต่างของ Servlet เมื่อเทียบกับ CGI Servlet เป็นเทคโนโลยี ที่มีรูปแบบการทำงาน เหมือนกับ เทคโนโลยี CGI ที่มีการใช้งานอยู่ แต่เนื่องจาก Servlet ถูกพัฒนามาจากเทคโนโลยีจาวาอีกที ดังนั้นต่อไปนี้จะขอเปรียบเทียบซึ่งแสดงให้เห็น ถึงความแตกต่างระหว่าง Servlet และ CGI

3.1.1.1 ประสิทธิภาพ การทำงานของ CGI จะมีการสร้างโปรเซส (Process) ใหม่ทุกครั้ง ที่มีการเรียกใช้งานจากผู้ใช้งาน ดังนั้นหากมีการเรียกใช้งาน จำนวนหลายครั้งในช่วงเวลาเดียวกัน ก็จะมีจำนวนโปรเซสเกิดขึ้นเท่ากับจำนวนครั้งที่เรียกใช้งาน โปรเซสจะมีขนาดและรูปแบบการทำงาน ที่เหมือนกัน ทำให้กินเวลาในการทำงานตามจำนวนโปรเซส แต่สำหรับ Servlet แล้ว ระบบเวอร์ชวลแมชีน (Java Virtual Machine) จะถูกดึงขึ้นมาทำงานก่อน เพื่อทำให้เกิดโปรเซสตามลักษณะการเรียกใช้ของผู้ใช้เช่นกัน แต่การทำงานของ Servlet จะเป็นรูปแบบเธรด(Thread) แทนที่จะเรียกว่าโปรเซส ซึ่งแต่ละเธรดขนาดการทำงานเล็กกว่าเดิม และกินระยะเวลาการทำงานน้อยกว่า จึงทำให้รูปแบบของ Servlet มีประสิทธิภาพกว่านั่นเอง

3.1.1.2 ความสะดวก หากคุณสามารถเขียนโปรแกรม ด้วยภาษา C, C++ หรือ Java อยู่แล้ว จะเกิดความสะดวกและรวดเร็วในการสร้าง Servlet มากกว่า แทนที่จะเริ่มศึกษาภาษาคำสั่งใหม่ๆ และ Servlet ก็มีความสามารถเพียงพอสำหรับการใช้งานบนเว็บอยู่แล้ว เช่น การรับข้อมูลจากฟอร์ม,

เอกสารนี้ก็ารอ่านและตั้งค่า HTTP Header, การทำงานกับ Cookie, การทำงานแบบ Session และอื่นๆอีกมากการคำ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.3 ความสามารถ Servlet มีข้อดีที่เห็นได้ชัดคือ สามารถติดต่อกับเว็บเซิร์ฟเวอร์ได้โดยตรง ในขณะที่ CGI ปกติทำไม่ได้ (ยกเว้นรุ่นใหม่ๆ) และ Servlet ยังสามารถแชร์ข้อมูลการทำงานร่วมกับโปรแกรมอื่นๆ ได้ด้วย

3.1.1.4 การโยกย้าย เนื่องจาก Servlet ถูกสร้างมาจาก Java ซึ่งเป็นเทคโนโลยี ที่สามารถถูกใช้งาน ณ. ที่ใดก็ได้ โดยไม่จำกัดรูปแบบ ดังนั้น หากพัฒนา Servlet เพื่อใช้งานแล้ว สามารถที่จะโยกย้ายผลผลิตไปยัง ระบบอื่นๆ ได้ โดยไม่ต้องปรับปรุงหรือเปลี่ยนแปลง

3.1.1.5 ไม่แพง ในปัจจุบันเว็บเซิร์ฟเวอร์ที่สนับสนุนการทำงาน Servlet มีอยู่มากมาย ทั้งที่เป็นแบบฟรี(ดาวน์โหลด) และซื้อ-ขายกัน นอกจากนั้น ในอนาคตแผนการผลิตเว็บเซิร์ฟเวอร์อื่นๆ ยังมีการสนับสนุนการใช้งาน Servlet อีกด้วย

3.1.2 โครงสร้างของ Servlet การรับส่งข้อมูลโดยใช้เว็บ ผ่านทางแท็กฟอร์ม <FORM ...> มีรูปแบบการส่งอยู่ 2 ลักษณะคือ GET และ POST โดยที่การส่งแบบ GET เป็นรูปแบบการส่งข้อมูลผ่านทาง URL (ช่องพิมพ์ชื่อเว็บหรือเว็บแอดเดรส) ซึ่งผู้ใช้อาจจะพิมพ์ได้เองโดยตรง เช่น <http://www.javacentrix.com?user=1245&pwd=4444> หรือฟอร์มที่แท็ก กำหนดแอมพริบิวต์ METHOD เป็น GET เช่น <FORM METHOD="GET" ... เป็นต้น (ตัวอย่างที่เป็นตัวหนาคือการส่งข้อมูลผ่านฟอร์มในแบบ GET)

สำหรับการส่งข้อมูลผ่านฟอร์มในแบบ POST คือการกำหนดแท็กฟอร์มให้มีแอมพริบิวต์ METHOD เป็น POST เช่น <FORM METHOD="POST" ... เป็นต้น

3.1.3 การใช้ Servlet ส่งข้อมูลรูปแบบ HTML แน่่อนว่าการสร้าง Servlet ขึ้นมาใช้งาน ไม่ได้มุ่งให้ส่งตัวอักษรเหมือนกับตัวอย่างที่ยกมา แต่ความต้องการคือ ให้สามารถส่งข้อมูลเว็บ ซึ่งก็คือข้อมูลในรูปแบบ HTML นั่นเอง ดังนั้นการส่งข้อมูล HTML ที่สำคัญ จำเป็นต้องส่งข้อมูลบอกลักษณะของรูปแบบ ให้กับผู้ใช้ (ซึ่งเป็นข้อมูลที่บราวเซอร์ได้รับ) ว่าเป็นข้อมูลรูปแบบ HTML ซึ่งเรียกกันว่า Content-Type แบบ text/html การส่งข้อมูลบอกราวเซอร์นี้ ทำได้โดยการเรียกใช้เมธอด setContentType ของคลาส HttpServletResponse แล้วส่งข้อความ text/html เป็นอาร์กิวเมนต์ไป

3.1.4 การอ่านข้อมูลจากเว็บ สำหรับการใช้งานเว็บผ่านบราวเซอร์ โดยทั่วไปแล้ว ปกติในช่องที่ชี้กำหนดข้อมูล URL จะประกอบด้วยชื่อเซิร์ฟเวอร์ ชื่อไดเรกทอรี และชื่อไฟล์เว็บเพจที่ต้องการดู เช่น หากข้อมูล URL คือ <http://www.javacentrix.com/tutorials/api/index.htm> นั้นหมายถึงต้องการอ่านเว็บเพจที่ชื่อ index.htm จากไดเรกทอรี tutorials/api จากเว็บเซิร์ฟเวอร์ที่ชื่อ www.javacentrix.com

สำหรับ URL นั้นนอกจาก จะใช้ชี้ตำแหน่งไฟล์ที่ต้องการเรียกดูแล้ว ยังสามารถใช้สำหรับส่งข้อมูลให้กับระบบเซิร์ฟเวอร์ เพื่อใช้ในการทำงาน หรือดำเนินการอย่างใดอย่างหนึ่ง ลักษณะ URL ประเภทนี้ หลังจากชื่อไฟล์แล้ว จะตามด้วยเครื่องหมายคำถาม ? และข้อมูลที่ต้องการส่งไปให้เซิร์ฟเวอร์ ซึ่งเรียกโดยทั่วไปว่าการส่งข้อมูลแบบ GET โดยมีรูปแบบคือ URL?Parameter1=Data1&Parameter2=Data2

จากรูปแบบการส่งข้อมูลผ่าน URL โดย Parameter1 และ Parameter2 คือชื่อพารามิเตอร์ที่ถูกเซิร์ฟเวอร์อ้างอิงได้ ในขณะที่ Data1 และ Data2 คือข้อมูลที่อยู่ในพารามิเตอร์แต่ละตัว หากต้องการส่งข้อมูลมากกว่า 2 ตัว ก็จะเพิ่มต่อท้ายไปได้เรื่อยๆ เช่น <http://www.javacentrix.com/tutorials/api/index.htm?Parameter1=Data1&Parameter2=Data2&Parameter3=Data3> -

URL?Parameter1=Data1&Parameter2=Data2&Parameter3=Data3&Parameter4=Data4

3.2 เจเอสพี (Java Server Page : JSP)

Java Server Page หรือที่เรียกกันง่ายๆว่า JSP เป็นเทคโนโลยีที่เป็นผลรวม ของการนำเอาหลักการ สร้างเว็บแบบสแตติกด้วย HTML มารวมกับการสร้างเว็บแบบไดนามิก ก่อให้เกิดการแสดงผลเว็บ ที่มีการ เปลี่ยนแปลงข้อมูลได้ตามการใช้งานของผู้ใช้ เห็นได้ว่าหากพูดถึงหลักการนำเสนอเว็บแล้ว เป็นหลักการที่ทำให้เว็บมีการเปลี่ยนแปลงข้อมูล เช่นเดียวกับ ที่ใช้โปรแกรมจำพวก CGI ทั่วไป แต่ในความหมายของ JSP แล้ว เป็นขั้นตอนและวิธีการสร้างที่ทำร่วมกับคำสั่ง HTML และเก็บบันทึกไว้ในไฟล์เดียวกัน แต่ในขณะที่ CGI เป็นโปรแกรมที่ผลิต HTML ให้เมื่อถูกสั่งให้ทำงาน

```
<HTML><HEAD><TITLE>Show JSP Page</TITLE></HEAD>
<BODY>
<% out.println("Hello Java Server Page"); %>
</BODY>
</HTML>
```

จากตัวอย่างด้านบนเป็นเพียงส่วนหนึ่งของ JSP ที่ถูกใส่ไว้ร่วมกับคำสั่ง HTML สังเกตจากตัวหนาที่อยู่ในแท็กพิเศษ <% ... %> ซึ่งเป็นคำสั่งในรูปแบบ JSP ทำให้ส่วนดังกล่าวสร้างข้อมูลในแบบไดนามิก ไว้ในเว็บเพจตามคำสั่ง ไฟล์ดังกล่าวจะบันทึกไว้ในนามสกุล .jsp จาก HTML เดิม

รูปแบบดังกล่าวไม่สามารถทำได้สำเร็จตามเทคโนโลยี JSP หากไฟล์ตัวอย่างด้านบน ไม่ได้ถูกเก็บไว้ในเว็บเซิร์ฟเวอร์ที่สนับสนุนการทำงานของ JSP ข้อความที่เป็นไดนามิกนั้น ถูกสำรวจเพื่อประมวลผล และใส่ลงในตำแหน่งดังกล่าว ด้วยความสามารถของ เซิร์ฟเวอร์ที่สนับสนุน JSP ก่อนส่งกลับไปให้ผู้ใช้ โดยผู้ใช้งานจะได้รับข้อความ Hello Java Server Page เท่านั้น โดยไม่เห็นคำสั่ง JSP เลย

3.2.1 ข้อแตกต่างของ JSP เมื่อเทียบกับเทคโนโลยีอื่นๆ JSP ไม่ได้เป็นเทคโนโลยีเดียว ในปัจจุบันที่สามารถทำให้เว็บ แสดงข้อมูลในแบบไดนามิกได้ มีเทคโนโลยีอื่นๆที่ทำงานในลักษณะนี้หลายแบบ เนื้อหาส่วนนี้ เป็นการแนะนำให้ผู้อ่าน ได้เห็นข้อแตกต่างบางประการ เมื่อเทียบกับ JSP

- Active Server Page(ASP) ASP เป็นเทคโนโลยีที่เหมือนกับ JSP แต่เป็นเทคโนโลยีที่พัฒนามาจาก บริษัทไมโครซอฟต์ ผู้ผลิตระบบปฏิบัติการวินโดวส์ที่รู้จักกัน โดยทั่วไป ลักษณะ JSP มีรูปแบบที่แตกต่างกันเด่นชัด 2 ประการคือ ประการแรก JSP สามารถสร้างได้จาก ภาษาจาวา ต่างจากภาษา VB หรือภาษาโคจ ซึ่งเป็นการกำหนดเฉพาะของ ไมโครซอฟต์ ประการที่สอง JSP ซึ่งเป็นผลพวงมาจากประการแรกคือ สามารถโยกย้ายการทำงาน ไปใช้งานระบบปฏิบัติการใดๆก็ได้ ที่มีการใช้งานกันอยู่ในปัจจุบัน โดยไม่ต้องแก้ไขหรือเปลี่ยนแปลงใดๆ ในขณะที่ ASP ก็โยกย้ายได้เช่นเดียวกัน แต่ต้องอยู่ในระบบปฏิบัติการที่ ไมโครซอฟต์กำหนดขึ้นเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Servlet JSP สามารถทำงานได้เช่นเดียวกับที่ Servlet ทำได้ แต่มีจุดเด่นมากกว่าที่ JSP มีความสะดวกในการสร้าง และเปลี่ยนแปลง มากกว่า เพราะทำโดยตรงที่ไฟล์ HTML มากกว่าการที่ต้องลงมือเขียนคำสั่งภาษาจาวา โดยตรงเหมือนกันสร้าง Servlet ซึ่งต้องมีการนำไปคอมไพล์ก่อนการนำไปใช้งาน
- Server-Side Includes(SSI) SSI คือรูปแบบการสร้างไดนามิกเว็บเช่นเดียวกัน แต่รูปแบบการทำงานคือ การนำเอาข้อมูลที่มีอยู่แล้ว บนเซิร์ฟเวอร์นำมาประกอบใส่ ในเว็บเท่านั้น ต่างจาก JSP ซึ่งสามารถนำข้อมูลมาได้ แต่มีรูปแบบในการทำงาน มีรูปแบบในการประมวลผล หรือการเรียกโปรแกรมภายนอก เช่น Servlet มาช่วยในการทำให้ข้อมูล มีรูปแบบต่างๆ เช่น ดึงข้อมูลมาจากระบบฐานข้อมูล (Database) นอกจากข้อมูลที่มีอยู่บนเซิร์ฟเวอร์เช่นที่ใช้ใน SSI เท่านั้น
- JavaScript เป็นการทำเนื้อหาไดนามิกให้กับเว็บ แต่รูปแบบการทำงานเกิดขึ้นของ JavaScript เกิดจากการประมวลผลและดึงข้อมูล ที่มีอยู่บนเครื่องผู้ใช้ หรือไคลเอนต์เท่านั้น มารวบรวมเนื้อหาในเว็บ ต่างกับ JSP ที่เป็นการทำไดนามิก แต่ข้อมูลถูกสร้าง และดึงมาจากระบบเซิร์ฟเวอร์ ที่มีหลากหลายมากกว่า และทำให้เนื้อข้อมูลเป็นเนื้อเดียวกัน เมื่อผู้ใช้เรียกดู แต่ JavaScript เนื้อข้อมูลเป็นของเครื่องผู้ใช้เอง ซึ่งผู้ใช้ต่างคน(ต่างเครื่อง) ก็จะให้ข้อมูลที่คล้ายกัน คือไม่เหมือนกันทั้งหมด
- Static HTML หรือสแตติก HTML แน่นนอนข้อนี้ คงเห็นได้ชัดเจนที่ว่า สแตติก HTML ให้เนื้อข้อมูลที่คงที่ตลอดเวลา ไม่ว่าจะถูกเรียกดูในเวลาใดๆ (ยกเว้นมีคนมาแก้ไขคำสั่ง HTML) ในขณะที่ JSP ทำให้เนื้อข้อมูลมีการเปลี่ยนแปลงตามการใช้งาน หรือเครื่องผู้ใช้ โดยไม่ต้องเปลี่ยนแปลงแก้ไขคำสั่ง JSP

	CGI/Perl	Mod_Perl	ASP	JSP
เว็บเซิร์ฟเวอร์	Any Web server	Apache Web Server	Microsoft IIS or Personal Web Server	Any Web server, including Apache, Netscape, IIS today
โยกย้ายการทำงาน	No	No	No	Yes
ลักษณะนำกลับมาใช้	No	No	No	Yes
สคริปต์หรือภาษา	C, Perl	Perl	VBScript, JScript	Java
ปกป้องการใช้งานหน่วยความจำ	Yes	No	No	Yes
สนับสนุนการทำงานร่วมในหลายโปรเซส	No	Yes	Yes	Yes

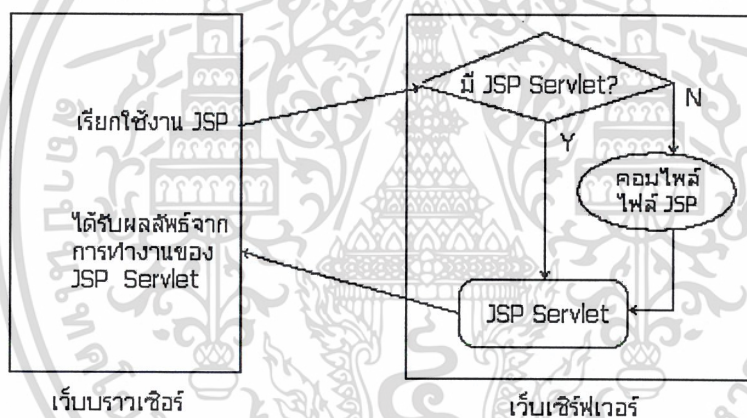
ตารางที่ 3-1 เปรียบเทียบเทคโนโลยี JSP กับเทคโนโลยีอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 หลักการทำงานของ JSP และ ไคเร็กทีฟ JSP (Java Server Page) เป็นผลพวงมาจาก เทคโนโลยีของ Java Servlet ซึ่งเป็นหลักการสร้างระบบ การประมวลผลเว็บ ที่ทำงาน โดยอาศัยทรัพยากร (เมโมรี, ซีพียู, ดิสก์) ของเว็บเซิร์ฟเวอร์ ก่อนจะส่งผลลัพธ์สู่ผู้ใช้ ผ่านระบบเน็ตเวิร์คหรืออินเทอร์เน็ต โดยผู้สร้าง JSP สามารถสร้างโดยการ แทรกคำสั่ง JSP ไว้ในชุดคำสั่ง HTML ที่ใช้สร้างเว็บ และบันทึกไว้ในไฟล์นามสกุล JSP จากนั้นเก็บไว้ที่เซิร์ฟเวอร์ ที่สนับสนุนการทำงานของ JSP

3.2.3 JSP Servlet คือ Servlet ที่เกิดขึ้นจากชุดคำสั่ง JSP ที่ถูกคอมไพล์ไปเป็น JSP Servlet โดยขบวนการนี้เกิดโดยอัตโนมัติภายในเว็บเซิร์ฟเวอร์ที่สนับสนุน การทำงานของ JSP และ JSP Servlet นี้เองที่เป็นส่วนที่เกิดการประมวลผลบนเซิร์ฟเวอร์ ดังนั้นไฟล์นามสกุล JSP ที่ถูกเรียกใช้งาน จะต้องถูกคอมไพล์ไปเป็น JSP Servlet ก่อนในครั้งแรกของการเรียกใช้งาน หรือเมื่อมีการเปลี่ยนแปลงชุดคำสั่ง

3.2.4 การทำงานของ JSP



รูปที่ 3-1 รูปอธิบายการทำงานของ JSP

เมื่อเว็บเบราว์เซอร์ มีการร้องขอ โดยผู้ใช้เพื่อเรียกข้อมูล JSP ที่อยู่บนเซิร์ฟเวอร์ ซึ่งก็หมายถึงไฟล์ที่มีนามสกุล JSP เว็บเซิร์ฟเวอร์จะทำการตรวจสอบว่า ไฟล์นั้นมีการคอมไพล์ไปเป็น JSP Servlet แล้วหรือยัง ถ้ายังไฟล์ JSP จะถูกคอมไพล์ โดยตัวเว็บเซิร์ฟเวอร์ ไปเป็น JSP Servlet แต่ถ้าไฟล์ถูกคอมไพล์แล้ว (ซึ่งเกิดขึ้นจากการใช้งานในครั้งแรก) เว็บเซิร์ฟเวอร์จะนำ JSP Servlet มาใช้งาน โดยการเรียกเพื่อประมวลผล ได้ผลลัพธ์อย่างไร้ก็จะเป็นข้อมูล ซึ่งเว็บเซิร์ฟเวอร์ จะส่งกลับไปให้ผู้เรียกใช้งาน และการประมวลผลของ JSP Servlet นี้เอง ที่ก่อให้เกิด ข้อมูลเว็บในแบบไดนามิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 จาวาบี๋น (JavaBean)

ในการเขียน JSP สำหรับระบบใหญ่ ๆ นักพัฒนาจะไม่นิยมใส่ java code ลงไปในไฟล์ JSP มากนัก สิ่งที่อยู่ในไฟล์มักจะเป็นเพียง HTML code และค่าของ variables ต่าง ๆ ที่ได้มาจาก JavaBeans เท่านั้น. ข้อดีของการใช้ JavaBean คือการง่ายต่อการเปลี่ยนแปลงลักษณะของ look-and-feel ของไฟล์ JSP ที่เป็นเช่นนั้น เพราะ ส่วนที่เป็น data (variables) กับส่วนที่เป็น presentation (HTML) จะอยู่ด้วยกันอย่างหลวม ๆ ดังนั้นเราจึงสามารถเปลี่ยนตัว presentation เมื่อไหร่ก็ได้ โดยตัว JavaBean ที่ใช้เก็บค่าของ variables ต่าง ๆ ยังคงเหมือนเดิม

สำหรับคนที่ไม่คุ้นกับ JavaBean ให้นึกถึง JavaBean เป็นลักษณะของกล่อง โดยกล่องนี้มีส่วนที่ใช้ติดต่อกับโลกภายนอกอยู่สองส่วนหลัก ๆ คือ

- ส่วนที่ใช้สำหรับรับข้อมูลต่าง ๆ โดยข้อมูลเหล่านี้จะถูกใช้เป็นตัวแปรในการควบคุมหรือเปลี่ยนแปลงคุณสมบัติและการทำงานของกล่อง (Setter)
- ส่วนที่ใช้สำหรับอ่านคุณสมบัติของกล่อง (Getter) ซึ่งโดยส่วนมาก(แต่ไม่ทั้งหมด) ก็คือค่าของตัวแปรต่าง ๆ ที่บรรจุอยู่ในกล่อง

ในการใช้งานกล่อง (JavaBean) ดังกล่าวร่วมกับ JSP เราสามารถที่จะสร้างกล่องขึ้นมาโดยใช้ `<jsp:useBean .../>`, ส่งค่าเข้าไปในกล่องโดยใช้ `<jsp:setProperty .../>` และอ่านค่าต่าง ๆ ที่อยู่ในกล่อง โดยใช้ `<jsp:getProperty .../>` tag แต่ก่อนที่เราจะเรียนรู้ถึงการ ใช้ tag เหล่านี้ เรามารู้จักถึง JavaBean ในรายละเอียดอีกซักเล็กน้อยเสียก่อน

3.3.1 Bean Conventions สำหรับคนที่เคยเขียน application ที่เกี่ยวกับ JavaServlet หรือ RMI จะพบว่าแต่ละคลาสที่เราเขียนขึ้น อาจจะต้องมีการ extend คลาสใดคลาสหนึ่งก่อน เช่น HttpServlet (ในกรณีของการเขียน JavaServlet) หรืออาจต้องทำการ implement interface ต่าง ๆ เช่น Serializable (ในกรณีของการเขียน RMI) แต่สำหรับการเขียนคลาสให้กลายเป็น JavaBean แล้ว เราไม่จำเป็นต้องทำสิ่งต่าง ๆ ที่กล่าวมาข้างต้นเลย. JavaBean จริง ๆ แล้วก็คือ จาวาคลาสธรรมดา ๆ แต่ถูกเขียนขึ้นโดยใช้ naming และ design conventions ที่อยู่ใน JavaBean Specification เท่านั้น. Conventions หลัก ๆ ที่เราจะพูดถึงสำหรับการเขียน JavaBean เพื่อใช้ใน JSP มีดังต่อไปนี้คือ

3.3.2 The Bean Constructor ในการเขียน Bean ให้สามารถใช้ได้กับ `<jsp:useBean .../>` สิ่งหนึ่งที่ Bean ของเราจะต้องมีคือ ตัว constructor ที่ไม่ต้องการ argument สำหรับการ initialize ของ Bean * เราสามารถนำคลาสเหล่านี้ไปใช้กับ `<jsp:useBean .../>` ได้ อย่างไรก็ตามคลาสนี้ยังไม่มีประโยชน์อะไรสำหรับไฟล์ JSP จริง ๆ ประโยชน์ที่เราจะได้จาก Bean ก็คือการส่งค่าต่าง ๆ เพื่อทำการควบคุมและเปลี่ยนแปลงคุณสมบัติของ Bean รวมไปถึงการอ่านค่าคุณสมบัติต่าง ๆ ของ Bean ซึ่งสิ่งเหล่านี้จะเกิดขึ้นได้โดยอาศัยสิ่งหนึ่งที่เรียกว่า Bean's Properties

3.3.3 Bean's Properties Bean's Properties คือตัวที่บอกว่าเราสามารถทำอะไรกับ JavaBean นั้นได้บ้าง [โดยส่วนมาก(แต่ไม่ทั้งหมด) ก็คือการเปลี่ยนและอ่านค่าต่าง ๆ ของ instance variables ที่บรรจุอยู่ข้างใน Bean นั้น ๆ] ซึ่งผลที่ได้ก็คือ การควบคุมและเปลี่ยนแปลงคุณสมบัติต่าง ๆ ของ Bean อย่างไรก็ตามบางที่ properties ที่เรากำหนดขึ้น อาจจะไม่ได้อะไรที่เกี่ยวข้องกับ instance variables ที่อยู่ข้างใน Bean นั้นเลย

...body...

```
</jsp:useBean>
```

3.2.6 Initializing a Bean - `<jsp:setProperty>` ใน Bean Conventions เรากล่าวถึง Bean Constructor ที่ไม่ต้องการ arguments ซึ่งถูกใช้สำหรับ `<jsp:useBean>` อย่างไรก็ตามบางครั้ง Bean ที่เราใช้อาจต้องการค่าเริ่มต้นในการ initialize ซึ่งเราสามารถทำการเซตค่าต่าง ๆ ให้ Bean ได้โดยใช้ `<jsp:setProperty .../>`

ค่าที่เราจะเซตให้ Bean บางครั้งอาจมาจากค่าที่เรากำหนดขึ้น หรืออาจจะเป็นค่าที่ได้จาก request ของ client ซึ่งมาจาก element ต่าง ๆ ใน `<FORM>` tag ของ html ดังนั้นการเซตค่า Bean จึงถูกแบ่งออกเป็นสองแบบคือ

3.2.6.1 การเซตค่าให้ Bean โดยใช้ค่าที่เรากำหนดขึ้น เช่น

```
<jsp:useBean id="myBeanInstance" class="com.myPackage.BeanClass" scope="request">
<jsp:setProperty name="myBeanInstance" property="myProperty"
value="definedValue"/>
</jsp:useBean>
```

name คือ ชื่อของ Bean instance ที่เราต้องการอ้างถึง
property คือ ชื่อของ property ที่เราต้องการเซตค่า
value คือ ค่าที่เราจะใส่เข้าไปให้ property นั้น

3.2.6.2 การเซตค่าให้ Bean โดยใช้ค่าที่ได้จาก element ต่าง ๆ ของ `<FORM>` tag เช่น

```
<jsp:useBean id="myBeanInstance" class="com.myPackage.BeanClass" scope="request">
<jsp:setProperty name="myBeanInstance" property="myProperty"
param="myFormElementName"/>
</jsp:useBean>
```

param คือ ชื่อของ element ที่อยู่ใน `<FORM>` tag ที่ส่ง request มาที่ไฟล์ JSP ที่มี
`<jsp:useBean>` นี้

หลังจากที่เราทำการโหลด และเซตค่าของ property ต่าง ๆ ใน Bean แล้ว เราสามารถดูผลที่เกิดขึ้นได้โดยใช้ `<jsp:getProperty name="myBeanInstance" property="myProperty"/>`

ค่าที่ได้จาก `<jsp:getProperty>` อาจจะเป็นค่าของ instance variables ของ Bean ที่เปลี่ยนไปหลังจากที่เราใช้ `<jsp:setProperty>` หรือบางครั้ง อาจจะเป็นลิสต์ของตารางรายชื่อ ในกรณีที่ค่า property ที่เราเซตผ่าน `<jsp:setProperty>` เป็นตัว trigger ให้ Bean ส่ง query ไปที่ดาตาเบส เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

เครื่องมือที่ใช้ในการพัฒนาโปรแกรมและฐานข้อมูล

4.1 บทนำ

ในบทนี้จะขอกกล่าวถึงเครื่องมือที่ใช้ในการพัฒนาโปรแกรมและฐานข้อมูลซึ่งมี 2 โปรแกรมด้วยกัน นั่นก็คือ IBM DB2 ที่ใช้ในการพัฒนาฐานข้อมูลและ Forte For Java ในการพัฒนาโปรแกรม

IBM DB2 เป็น ระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management) ระบบหนึ่ง ซึ่งได้รับการอนุเคราะห์จากบริษัทไอบีเอ็ม ประเทศไทย โดยทางบริษัทได้ให้ชุดผลิตภัณฑ์ของ DB2 ประกอบไปด้วย DB2 Client Tool, DB2 UDB และ DB2 Personal Edition สำหรับโครงการนี้นำเอา DB2 มาใช้ในการพัฒนา และ ทดสอบโครงการ โดย DB server จะใช้ DB2 Universal Database (UDB) for Unix ในการจัดการฐานข้อมูลสารสนเทศนักศึกษา และใช้ DB2 Client Tool for Window ในการทดสอบโปรแกรม

4.2 ลักษณะของ DB2ที่ใช้ในโครงการ

DB2 Universal Database (UDB) เป็น RDBMS ที่สนับสนุนความสามารถในการจัดการ RDBMS ผ่านเว็บอย่างเต็มรูปแบบ โดยโครงการนี้ใช้ version 7.2 ซึ่งมีความสามารถใหม่ๆ เช่น

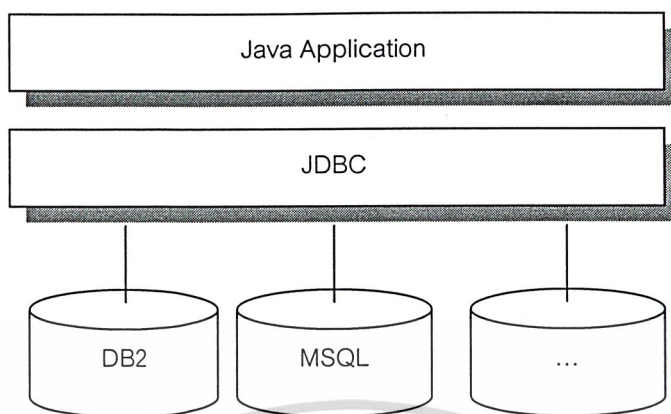
- สนับสนุน XML โดย DB2 ยอมให้ XML เป็นชนิดข้อมูลในคอลัมน์ได้
- สามารถทำงานร่วมกับโปรแกรมต่างๆ เช่น The DB2 Software Developer's Kit (SDK), XML, Forte for Java ฯลฯ

4.3 การเชื่อมต่อเข้ากับฐานข้อมูลโดยใช้ JDBC

JDBC (Java Database Connectivity) ถูกพัฒนาโดย JavaSoft Department ของบริษัท Sun Microsystems ซึ่งก็คือฟังก์ชันมาตรฐานหรือ Java Application Programming Interface (API) สำหรับการเชื่อมต่อกับระบบฐานข้อมูลสัมพันธ์ (Relation Database) ใน Library ของ JDBC ได้มีส่วนที่ติดต่อกับฐานข้อมูลและได้นำคำสั่ง SQL ไปประมวลผลและควบคุมการทำงานของฐานข้อมูลเชิงสัมพันธ์ ดังรูป 4.1

JDBC ถูกสร้างขึ้นมาในระดับของการเชื่อมต่อกับฐานข้อมูลในรูปแบบที่คล้ายกับ ODBC (ของทางบริษัทไมโครซอฟท์) หลักการของทั้งสองมาตรฐานตั้งอยู่บนฐานเดียวกันคือ X/Open SQL Call-Level Interface ของระบบ X-windows และ JDBC driver ต้องเข้ากับระดับมาตรฐานในการเข้าถึง SQL (ANSI SQL Entry Level Standards) และต้องผ่าน Conformance test ซึ่ง JavaSoft เป็นผู้กำหนดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-1 แสดงลักษณะการทำงานของ JDBC

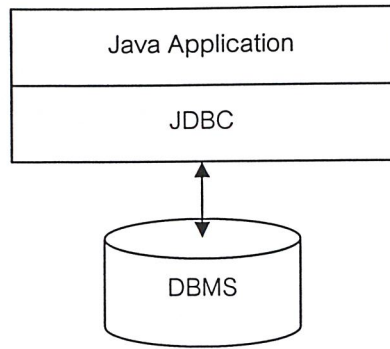
4.3.1 รูปแบบการเชื่อมต่อฐานข้อมูลของ JDBC โดย JDBC API นั้น สนับสนุนการเชื่อมต่อฐานข้อมูลแบบ 2-Tier และ 3-Tier

4.3.1.1 รูปแบบ 2-Tier จาวาแอปพลิเคชันหรือจาวาแอปพลิเคชันจะทำการติดต่อกับฐานข้อมูลโดยตรงจึงมีความจำเป็นที่โปรแกรมจาวาต้องการ JDBC Driver พิเศษที่สามารถสื่อสารกับระบบจัดการฐานข้อมูลชนิดนั้นได้ รูปแบบการเชื่อมต่อแสดงรูปที่ 4-2

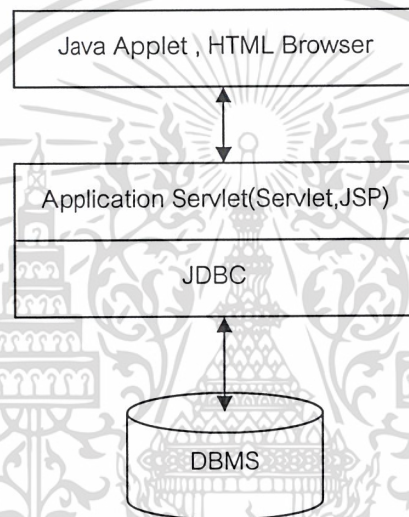
คำสั่งในการเรียกค้นข้อมูลในการเรียกค้นข้อมูลในรูปแบบของภาษา SQL จะถูกส่งจากผู้ใช้ไปสู่ฐานข้อมูลหลังจากนั้นการประมวลผลของระบบจัดการฐานข้อมูลก็จะถูกส่งกลับมาที่ผู้ใช้ ฐานข้อมูลดังกล่าวนี้ส่วนมากจะถูกติดตั้งอยู่บนคอมพิวเตอร์อีกเครื่อง โดยผู้ใช้สามารถเชื่อมต่อผ่านระบบเน็ตเวิร์ก รูปแบบ 2-tier นี้ใช้หลักการทำงานเช่นเดียวกับรูปแบบไคลเอนต์/เซิร์ฟเวอร์ที่เราจักกันดี โดยที่เครื่องคอมพิวเตอร์ของผู้ใช้คือไคลเอนต์และเครื่องคอมพิวเตอร์ที่ให้บริการฐานข้อมูลคือเซิร์ฟเวอร์ เครื่องข่ายคอมพิวเตอร์ที่นิยมใช้รูปแบบ 2-Tier มักจะเป็นเครือข่ายอินทราเน็ต (Intranet) สำหรับดำเนินธุรกรรมภายในองค์กร

4.3.1.2 รูปแบบ 3-Tier คำสั่งต่างๆจากผู้ใช้จะถูกส่งไปให้กับ Tier กลางหรือตัวกลาง หลังจากนั้น Tier กลางจะแปลงคำสั่งเหล่านั้นให้เป็นภาษา SQL เพื่อส่งให้กับระบบฐานข้อมูลเพื่อทำการประมวลผลและทำการส่งผลลัพธ์คืนให้กับ Tier กลางและส่งให้ผู้ใช้ในที่สุด หลักการทำงานเช่นนี้มักจะพบในเครือข่ายอินเทอร์เน็ตซึ่งประกอบด้วยเครื่องคอมพิวเตอร์หลากหลายชนิดและเว็บเซิร์ฟเวอร์ (Web Server) ซึ่งทำหน้าที่เป็น Tier กลางในการจัดการให้ระบบคอมพิวเตอร์ทั้งไคลเอนต์และเซิร์ฟเวอร์ฐานข้อมูลสามารถติดต่อกันได้ การปรับเปลี่ยนระบบไคลเอนต์และเซิร์ฟเวอร์ หรือแม้กระทั่งการเปลี่ยนตัวฐานข้อมูลตัวใหม่จะไม่มีผลกระทบซึ่งกันและกันเกิดขึ้น หรือแม้กระทั่งการเปลี่ยนตัวฐานข้อมูลตัวใหม่จะไม่มีผลกระทบซึ่งกันและกันเกิดขึ้น แสดงดังรูป 4.3

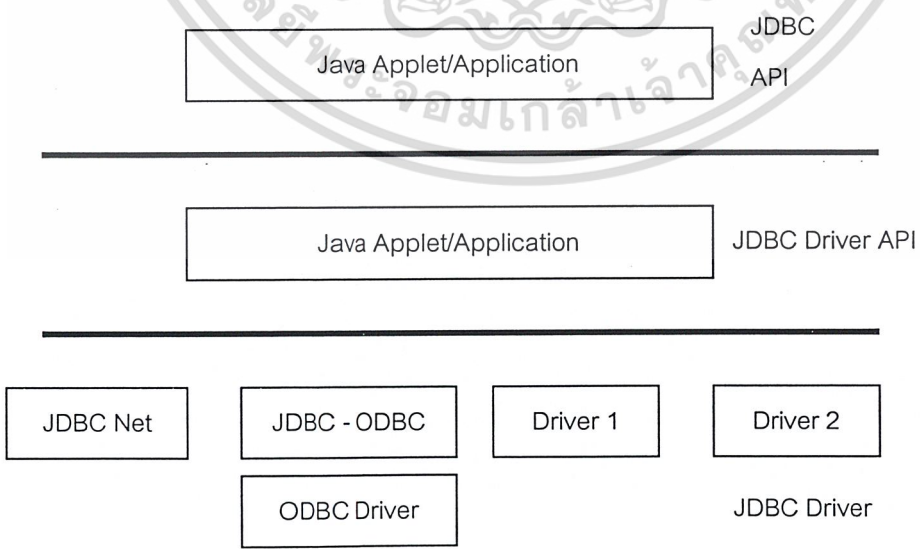
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-2 แสดงรูปแบบ 2-Tier



รูปที่ 4-3 แสดงรูปแบบ 3-Tier



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 รูปที่ 4-4 ระดับการเชื่อมต่อของ JDBC API
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอยู่ใต้อำนาจเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 โครงสร้างของ JDBC โครงสร้างการเชื่อมต่อภายใน JDBC ประกอบด้วย 3 ระดับ คือ JDBC API, JDBC Driver API และ JDBC Driver ดังรูปที่ 4.4 ระดับบนสุด เป็น JDBC API เป็นระดับฟังก์ชัน API ที่อำนวยความสะดวกให้แก่โปรแกรมประยุกต์ ระดับกลาง JDBC Driver (มี Driver ที่ต่างกันอยู่ 4 ชนิด) ที่เหมาะสม

4.3.3 รูปแบบของ JDBC Driver หรือส่วนที่อยู่เบื้องหลังการทำงานของฟังก์ชัน API ต่างๆ ของ JDBC ถูกแบ่งออกเป็นทั้งหมด 4 ชนิดคือ

4.3.3.1 JDBC-ODBC Bridge

4.3.3.2 Native-API(Party-Java) Driver

4.3.3.3 Net-Protocol(All-Java) Driver

4.3.3.4 Nativer-Protocol(All-Java)

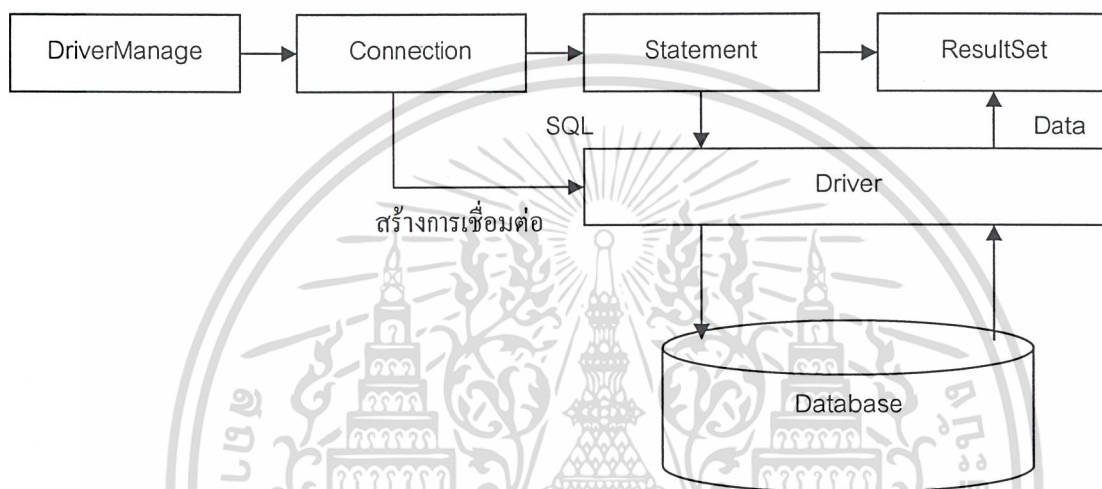
ในที่นี้จะกล่าวถึงแค่ JDBC/ODBC Bridge ถูกพัฒนาโดย JavaSoft และ InterSolv ทำหน้าที่เป็นตัวกลางในการเข้าถึงฐานข้อมูลได้โดยผ่านทางการทำงานของ ODBC โดยนำข้อดีของ ODBC-enabled data source ที่มีใช้อยู่โดยทั่วไปอย่างมากมาย ผังโคลอินต์ที่เป็นจาวาแอปเพล็ตหรือจาวาแอปพลิเคชันจะถูกเขียนโดยใช้ JDBC API Bridge จะทำการแปลงการเรียกใช้ JDBC ไปยัง ODBC Driver ที่เหมาะสมสำหรับฐานข้อมูล ข้อดีของ Bridge ทำให้แอปพลิเคชันสามารถติดต่อกับฐานข้อมูลได้ง่ายจากผู้ผลิตที่มาก โดยสามารถเลือกได้จาก ODBC Driver ที่เหมาะสม

4.3.4 ขั้นตอนการติดต่อกับระบบจัดการฐานข้อมูล (DBMS:ในที่นี้คือ IBM DB2) ด้วยภาษาจาวา สำหรับการเชื่อมต่อกับระบบจัดการฐานข้อมูลด้วยภาษาจาวานั้นจะต้องประกอบด้วยขั้นตอนที่มากพอสมควร โดยแสดงเป็นขั้นตอนดังนี้

- เรียกใช้คลาสที่จำเป็น เช่น
`import java.sql.*;`
- เลือกใช้ Driver ของ DB2
`Class.forName("COM.ibm.db2.jdbc.net.DB2Driver").newInstance();`
- สร้างออบเจกต์ Connection
`Connection dbconn;`
- กำหนดแหล่งของข้อมูลภายในระบบจัดการฐานข้อมูล
`dbconn = DriverManager.getConnection(url, userid, passwd);`
- สร้างออบเจกต์ Statement
`Statement statement = dbconn.createStatement();`
- ประมวลผลคำสั่ง SQL โดยใช้คำสั่งออบเจกต์ Statement
`statement.executeQuery("SELECT * FROM STUDENT");`
- รับผลจากการประมวลผลคำสั่งด้วยออบเจกต์ Resultset

เอกสารนี้เป็นเอกสาร Resultset res = statement.executeQuery(query); นั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ปิดออบเจกต์ ResultSet
res.close()
- ปิดออบเจกต์ Statement
statement.close();
- ปิดออบเจกต์ Connection
dbcon.close();



รูปที่ 4-5 ลำดับการเชื่อมต่อเข้าฐานข้อมูลโดยใช้ JDBC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การวิเคราะห์และออกแบบระบบทะเบียน นักศึกษาระดับบัณฑิตศึกษา

ในบทนี้จะได้กล่าวถึงขั้นตอนการวิเคราะห์และออกแบบระบบทะเบียนนักศึกษาระดับบัณฑิตศึกษา โดยเริ่มด้วยการอธิบายถึงหน้าที่ความรับผิดชอบ การดูแลงานด้านทะเบียนนักศึกษาของฝ่ายทะเบียนนักศึกษา บัณฑิตวิทยาลัย และวิเคราะห์ถึงข้อเสีย และจุดบกพร่องต่าง ๆ ของระบบทะเบียนนักศึกษาที่ใช้อยู่ในปัจจุบัน หลังจากนั้นก็จะกล่าวถึงขั้นตอนการวิเคราะห์และการออกแบบ โดยการนำเอายูเอ็มแอลเข้ามาอธิบายลักษณะของระบบ ดังนี้

5.1 ฝ่ายทะเบียนนักศึกษาของบัณฑิตวิทยาลัย

ฝ่ายทะเบียนนักศึกษาของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เป็นหน่วยงานที่รับผิดชอบงานในเรื่องของทะเบียนนักศึกษา ในระดับบัณฑิตศึกษาทุกชั้นตอนนับตั้งแต่ การรับนักศึกษาใหม่ การขึ้นทะเบียนนักศึกษาใหม่ การลงทะเบียน การแก้ไขข้อมูลการลงทะเบียน การเปิด-ปิด รายวิชา การกรอกข้อมูลเกรด การเพิ่มแก้ไขข้อมูลหลักสูตร การดำเนินการตามคำร้องต่าง ๆ ของนักศึกษา จนถึงเรื่องของการจบการศึกษา

การทำงานของฝ่ายทะเบียนในปัจจุบันนั้น มีระบบสารสนเทศของสำนักวิจัยและบริการคอมพิวเตอร์ รองรับการทำงานอยู่ ซึ่งนับว่าเป็นระบบที่ได้อำนวยความสะดวกให้มากพอสมควร แต่ก็ยังคงพบปัญหาในการทำงานอยู่เช่นกัน ในที่นี้ขอยกตัวอย่างหนึ่งในงานที่ฝ่ายทะเบียนต้องรับผิดชอบ นั่นก็คือการลงทะเบียนเรียน ของนักศึกษา แต่เดิมนั้นใช้รหัสแท่งเข้าช่วย โดยนักศึกษาจะต้องตัดแถบรหัสแท่งของรายวิชาที่ต้องการจะลงทะเบียน มาแนบลงบนเอกสารการลงทะเบียนแล้วนำเอกสารดังกล่าวไปให้อาจารย์ที่ปรึกษาทำการ ตรวจสอบ และเห็นชอบ หลังจากที่อาจารย์ที่ปรึกษาได้ลงนามเห็นชอบแล้วถึงจะนำเอกสารดังกล่าวมาให้ เจ้าหน้าที่ทะเบียน เพื่อเอาข้อมูลลงสู่ระบบ แล้วออกจะออกเอกสารการชำระเงินเพื่อให้นักศึกษานำเงินไปชำระ ที่ธนาคารที่ได้กำหนดไว้ ก็จะสิ้นสุดขั้นตอนการลงทะเบียน ต่อมาในปีการศึกษา 2544 ได้ทำการปรับระบบให้ นักศึกษาสามารถทำการลงทะเบียนผ่านระบบอินเทอร์เน็ตได้ โดยมีกรรณการตรวจสอบของอาจารย์ที่ปรึกษาไป หลังจากที้นำระบบดังกล่าวเข้ามาใช้ก็ยังคงพบปัญหาหลายอย่างอันเนื่องมาจากเกิดความสับสนใน ขั้นตอนการปฏิบัติ รวมถึงระบบที่นำมาใช้นี้ไม่ได้มีการตรวจสอบข้อกำหนดในการลงทะเบียนอย่างครบถ้วน อย่างเช่นการตรวจสอบรายวิชาที่จำเป็นต้องเรียนมาก่อน (Prerequisite) เป็นต้น จากตัวอย่างการลงทะเบียนเรียนของนักศึกษานับเป็นงานหนึ่งในหลายงานที่ฝ่ายทะเบียนต้องรับผิดชอบ ซึ่งคงไม่สามารถกล่าวถึง รายละเอียดของงานทุกงานได้ในที่นี้

ในการดำเนินงานของฝ่ายทะเบียน และการตรวจสอบเงื่อนไขที่ระบบไม่ได้รองรับไว้นั้น จะต้องใช้ เอกสารนี้เป็นเอกสารที่ส่งงานไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า บุคคลเป็นผู้ดำเนินการเป็นส่วนใหญ่ และยังมีขั้นตอนการปฏิบัติงานบางอย่างที่ยังไม่ชัดเจน รวมถึงกฎ ระเบียบต่าง ๆ ทั้งสิ้น อีกทั้งยังมีเหตุเปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อบังคับต่าง ๆ ที่อาจล้าสมัยอยู่บ้าง ทำให้เกิดความสับสนในการปฏิบัติ ซึ่งเมื่อเกิดปัญหาขึ้นก็จะมีกร แก้ปัญหาเป็นกรณี ๆ ไปซึ่งอาจก่อให้เกิดปัญหานั้นซ้ำอีกได้ในภายหลัง

5.2 ปัญหาที่พบในระบบทะเบียนนักศึกษาเดิม

ก่อนที่จะดำเนินการพัฒนาโครงการนี้ ผู้พัฒนาโครงการได้เข้าไปศึกษาทำความเข้าใจถึงการทำงาน เดิมของระบบทะเบียนนักศึกษาระดับบัณฑิตศึกษา ซึ่งทำให้ได้ทราบถึงจุดอ่อนของระบบเดิมอยู่หลายประการ เช่น

5.2.1 ในขั้นตอนการลงทะเบียนของนักศึกษาระดับบัณฑิตศึกษานั้นมีรายละเอียดหลายขั้นตอน ตั้งแต่ การเข้าดูรายวิชาที่เปิดสอนร่วมกับการปรึกษาอาจารย์ที่ปรึกษาหลังจากที่ได้ข้อตกลงในการลงทะเบียน แล้ว นักศึกษาต้องทำการตัดกระดาษหัดแท่ง รายวิชาที่ต้องการพร้อมมาดำเนินการลงทะเบียนที่ฝ่ายทะเบียนซึ่ง นับว่าเป็นขั้นตอนที่ซับซ้อน

5.2.2 รายวิชาที่เปิดสอนในระดับบัณฑิตศึกษานั้น เปิดให้นักศึกษาสามารถเลือกลงทะเบียน ดังนั้นอาจมีบางรายวิชาที่นักศึกษาให้ความสนใจลงทะเบียนน้อย ทำให้รายวิชานั้นต้องปิดลงไปสร้างความ เดือดร้อนให้กับนักศึกษาที่ลงทะเบียนวิชานั้น ๆ อยู่

5.2.3 ระบบเดิมยังไม่ได้มีการตรวจสอบเงื่อนไขการลงทะเบียนอย่างจริงจัง อย่างเช่นการตรวจสอบ รายวิชาที่จำเป็นต้องเรียนมาก่อน(Prerequisite) จำนวนหน่วยกิตรวม และในบางวิชาที่จำกัดจำนวนผู้ที่ ลงทะเบียนยังไม่สามารถจำกัดได้ ทำให้เกิดการลงทะเบียนที่ผิดเงื่อนไขได้ง่าย

5.2.4 ในระบบเดิมยังไม่สามารถตรวจสอบรายวิชาที่นักศึกษาได้ลงทะเบียนไปแล้วได้ จึงทำให้เกิด ปัญหาในการลงทะเบียนซ้ำซ้อนในรายวิชาเดียวกัน และนักศึกษาที่ถูกตัดชั้นไปแล้ว ยังสามารถทำการ ลงทะเบียนได้เหมือนนักศึกษาทั่วไป

5.2.5 ประเภทนักศึกษาที่มีหลายประเภทเช่นนักศึกษาสามัญ นักศึกษาทดลองเรียน ซึ่งมีระเบียบ ปฏิบัติบางอย่างที่แตกต่างกัน

5.2.6 นักศึกษาต้องไปติดต่อกับฝ่ายทะเบียนบ่อยครั้ง ไม่ว่าจะเป็นการยื่นคำร้องต่าง ๆ การเพิ่มแก้ไข การลงทะเบียน การตรวจสอบข้อมูลเกี่ยวกับการลงทะเบียนและเกรด

นั่นคือตัวอย่างของปัญหาที่เกิดขึ้นในระบบเดิม ปัญหาเหล่านี้ก่อให้เกิดความสับสนในการทำงาน จึง เป็นการเหมาะสมที่จะพัฒนาระบบให้มีประสิทธิภาพมากขึ้น

5.3 การวิเคราะห์ระบบ

ในการวิเคราะห์ระบบนั้นนอกจากการเข้าไปศึกษาการทำงานจากระบบเดิมแล้ว การหาถึงความ ต้องการของผู้ใช้ระบบนั้นนับว่าเป็นสิ่งสำคัญอีกประการหนึ่งในการพัฒนาระบบใด ๆ ขึ้นมา ในที่นี้ผู้พัฒนา โครงการได้ทำการสกัดความต้องการของผู้ใช้ระบบอย่างละเอียด โดยขั้นตอนในการสกัดความต้องการของ ผู้ใช้ดำเนินการหลายวิธีด้วยกันเช่น ทำการสัมภาษณ์กับผู้ใช้งานโดยตรง ไม่ว่าจะป็นเจ้าหน้าที่ทะเบียน อาจารย์และนักศึกษา การเข้าไปศึกษาเอกสาร แบบฟอร์มต่าง ๆ แล้วนำข้อมูลที่ได้นำมาทำความเข้าใจและ เล็กส่วนที่เป็นเอกสารที่ส่งงานไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่ลบลาตเนื้อหาไปใช้ประโยชน์ด้านการค้า วิเคราะห์ให้ได้ทั้งกระบวนการทำงานเดิมและแนวทางที่ผู้ใช้ต้องการ ซึ่งทำให้ได้ความต้องการหลากหลายจาก ไม่วากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุขัดแย้งและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้เข้ามากำหนดฟังก์ชันการทำงานให้กับระบบ นอกจากนั้นผู้พัฒนาโครงการยังได้พยายามเลือกใช้เทคโนโลยีที่เหมาะสม และไม่ต้องลงทุนเพิ่มเติมมากนักอีกด้วย

หลังจากที่ได้ทำการวิเคราะห์ระบบอย่างละเอียด เราได้ทั้งปัญหาของระบบและความต้องการของผู้ใช้ ซึ่งพอที่จะสรุปเป็นแนวทางในการออกแบบและพัฒนาระบบต่อไป ดังนี้

5.3.1 ให้นักศึกษาสามารถทำกิจกรรมและเข้าถึงข้อมูลต่าง ๆ ที่เกี่ยวกับงานทะเบียนได้ โดยที่ไม่ต้องมาติดต่อฝ่ายทะเบียน หรือมาติดต่อให้น้อยที่สุด เพื่อเพิ่มความสะดวกให้นักศึกษาเอง รวมถึงลดภาระงานของเจ้าหน้าที่ นั่นก็คือให้นักศึกษาสามารถทำกิจกรรมเหล่านั้นผ่านทางอินเทอร์เน็ต

5.3.2 ลดภาระการตรวจสอบเงื่อนไขต่าง ๆ ด้วยการสร้างให้โปรแกรมสามารถตรวจสอบเงื่อนไขเหล่านั้นให้ครอบคลุมและถูกต้องมากที่สุด เช่น การตรวจสอบตรวจสอบรายวิชาที่จำเป็นต้องเรียนมาก่อน (Prerequisite) จึงสามารถทำการลงทะเบียนรายวิชาได้ ซึ่งเราต้องทำการศึกษาเงื่อนไขเหล่านั้นให้ชัดเจนและออกแบบเรื่องของโปรแกรมและฐานข้อมูลให้สามารถสนับสนุนการตรวจสอบดังกล่าว

5.3.3 การเพิ่ม แก้ไขข้อมูลต่าง ๆ นั้น เจ้าหน้าที่จะต้องสามารถทำได้อย่างครบถ้วนและสะดวก เพราะข้อมูลที่เกี่ยวข้องกับงานทะเบียนนั้นค่อนข้างซับซ้อน ยกตัวอย่างเช่นข้อมูลของหลักสูตรซึ่งมีข้อมูลย่อย ๆ ประกอบอยู่หลายอย่าง ระบบจึงควรเก็บข้อมูลให้ครอบคลุมให้ได้มากที่สุด

5.3.4 ในส่วนของการลงทะเบียนควรจะให้มีการลงทะเบียนล่วงหน้าก่อน เพื่อจะเป็นการโหวตจำนวนผู้ที่ลงทะเบียนในรายวิชานั้นๆ เพื่อที่จะได้ทำการเปิดสอนและให้ลงทะเบียนจริงเฉพาะวิชาที่มีผู้ลงทะเบียนมากพอที่จะเปิดสอนได้

5.3.5 ระบบควรมีความเป็นอัตโนมัติในการแสดงข้อมูลต่าง ๆ ให้กับผู้ใช้ยกตัวอย่างเช่นในการตรวจสอบช่วงเวลาการลงทะเบียน ระบบควรสามารถตรวจสอบได้เอง ไม่ต้องให้เจ้าหน้าที่ทำการตั้งค่าเอง

5.4 การออกแบบระบบโดยใช้ยูเอ็มแอล

หลังจากที่เราได้หาความต้องการของผู้ใช้และศึกษาระบบเดิมพอสมควรแล้วเราต้องนำข้อมูลที่ได้มาทำการวิเคราะห์และออกแบบโดยใช้ยูเอ็มแอล แต่ถึงอย่างไรก็ตามในการออกแบบระบบนี้ เราใช้กระบวนการทางวิศวกรรมซอฟต์แวร์แบบวนรอบ(Iteration Software Process) ซึ่งเป็นกระบวนการที่ทำงานเป็นวงรอบ ในแต่ละรอบก็จะมีปรับปรุงในแต่ละขั้นตอนเสมอ ดังนั้นขั้นตอนที่จะอธิบายต่อไปนี้จึงมีการปรับปรุงอยู่ตลอดเวลา และในส่วนนี้จะกล่าวถึงไดอะแกรมต่าง ๆ จึงควรศึกษาในบทที่ 2 ประกอบด้วย

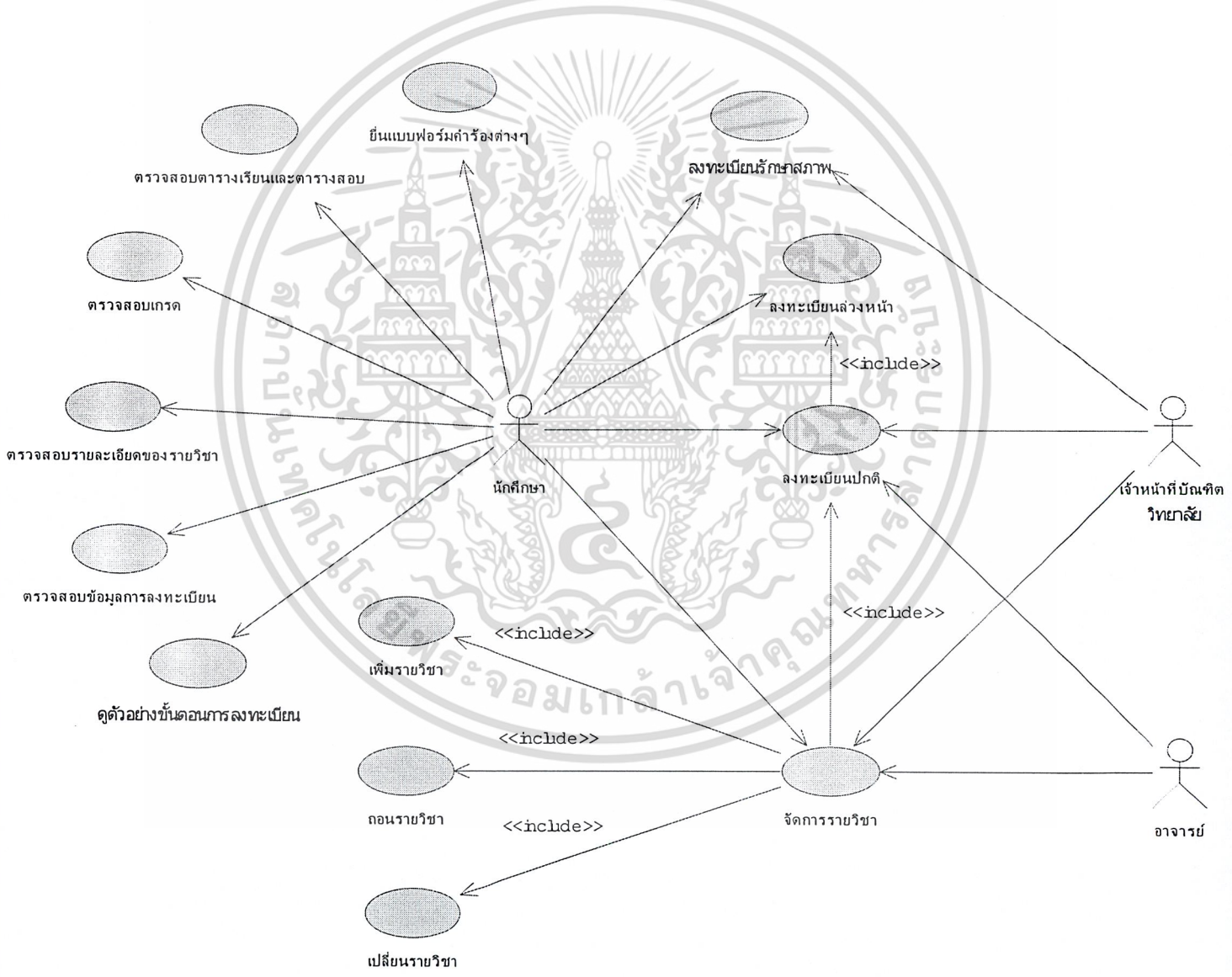
การหาว่ามีใครบ้างที่ต้องใช้ระบบนี้ซึ่งเราเรียกว่าแอ็คเตอร์(Actor) และหาว่าแต่ละคนทำอะไรกับระบบบ้างหรือใช้ฟังก์ชันใดของระบบบ้าง เมื่อได้แอ็คเตอร์และฟังก์ชันการทำงานแล้วนำมาเขียนเป็นยูสเคสไดอะแกรม(Use Case Diagram) ซึ่งเราถือว่าแต่ละฟังก์ชันการทำงานของระบบคือหนึ่งยูสเคส(Use case)

จากรูปที่ 5-1 และ 5-2 นั่นคือ ยูสเคสไดอะแกรมในส่วนของนักศึกษา และเจ้าหน้าที่ ซึ่งจะเห็นได้ว่าประกอบด้วย 5 แอ็คเตอร์ซึ่งมีรายละเอียดของแต่ละแอ็คเตอร์ดังนี้

- นักศึกษา ก็คือ ผู้ใช้ระบบ ซึ่งได้แก่นักศึกษาในระดับบัณฑิตศึกษา ซึ่งสามารถที่จะใช้งานระบบผ่านทางอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เจ้าหน้าที่บัณฑิตวิทยาลัย ก็คือ เจ้าหน้าที่ทะเบียนของบัณฑิตวิทยาลัย ซึ่งจะเป็นผู้ที่ดูแล เพิ่มแก้ไขข้อมูลของระบบ การทำงานกับระบบนั้นจะทำผ่านทางโปรแกรมที่จะพัฒนาขึ้น
- เจ้าหน้าที่บัณฑิตศึกษา ก็คือ เจ้าหน้าที่งานบัณฑิตศึกษาในส่วนคณะต่าง ๆ ก็จะสามารถในการเพิ่ม แก้ไข ข้อมูลบางอย่างของระบบ เพื่อเป็นการแบ่งภาระงานของเจ้าหน้าที่ทะเบียนของบัณฑิตวิทยาลัย รวมถึงการตัดตอนกระบวนการทำงาน ทำให้มีโอกาสเกิดข้อผิดพลาดน้อยลง
- อาจารย์ ก็คือ อาจารย์ซึ่ง อาจจะไม่ได้ใช้งานระบบโดยตรง แต่ก็มีส่วนเกี่ยวข้องกับระบบในงานบางงานเช่นการกรอกเกรดของนักศึกษา การลงทะเบียน เป็นต้น
- ผู้ดูแลระบบ คือผู้ดูแลระบบซึ่งจะเป็นผู้ที่จัดการเกี่ยวกับระบบ โดยตรงกับฐานข้อมูลทำให้สามารถจัดการกับข้อมูลได้ทั้งหมด รวมถึงเป็นผู้ที่ดูแลเกี่ยวกับความปลอดภัยของระบบด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตีพิมพ์ลงเนื้อหา และต้องอยู่ในวงจำกัดของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5-1 Usecase ในส่วนของนักศึกษา

จากคำอธิบายข้างต้น ฟังก์ชันการทำงานต่าง ๆ ของระบบที่แอดเดสสามารถใช้ได้นั้นก็ประกอบไปด้วย 3 ส่วนนั้นคือ

- ส่วนที่เป็นแอปพลิเคชันซึ่งเป็นโปรแกรมที่พัฒนาด้วยภาษาจาวาเพื่อให้เจ้าหน้าที่ทะเบียนทั้งส่วนกลางและส่วนคณะได้ใช้
- ส่วนที่เป็นเว็บแอปพลิเคชัน ซึ่งเป็นส่วนที่ให้นักศึกษาใช้ในการติดต่อกับระบบ
- ส่วนที่เป็นเอกสาร หรือโปรแกรมพื้นฐานที่มีอยู่แล้ว ซึ่งไม่ต้องทำการพัฒนาอย่างเช่นระบบงานในส่วนของผู้ดูแลระบบที่ใช้โปรแกรมจัดการฐานข้อมูล เป็นต้น

นอกจากในส่วนของแอดเดสแล้ว จะขออธิบายถึงฟังก์ชันการทำงานต่าง ๆ ของระบบนั้นก็คือยูสเคสแต่ละยูสเคสนั้นเอง มีรายละเอียดดังนี้

- ตรวจสอบตารางเรียนและตารางสอบ : เป็นส่วนที่ให้นักศึกษาจะเข้าไปดูตารางเรียนและตารางสอบ
- ตรวจสอบรายละเอียดของรายวิชา : เป็นส่วนที่ให้นักศึกษาดูรายละเอียดของวิชาที่เปิดสอน และข้อมูลต่าง ๆ ของรายวิชานั้น ๆ
- ลงทะเบียนล่วงหน้า : เป็นส่วนที่ให้นักศึกษาทำการลงทะเบียนล่วงหน้า
- ลงทะเบียนปกติ : เป็นส่วนที่ให้นักศึกษาลงทะเบียนจริง
- จัดการรายวิชา : เป็นส่วนที่ให้นักศึกษาเพิ่ม เปลี่ยน ถอน รายวิชา
- ตรวจสอบข้อมูลการลงทะเบียน : เป็นส่วนที่ให้นักศึกษาดูผลการลงทะเบียน
- ตรวจสอบเกรด : เป็นส่วนที่ให้นักศึกษาดูผลการศึกษา
- ยื่นแบบฟอร์มคำร้องต่างๆ : เป็นส่วนที่ให้นักศึกษาสามารถกรอกคำร้องและพิมพ์ออกมา เพื่อนำไปดำเนินการต่อไป ซึ่งนักศึกษาไม่จำเป็นต้องกรอกข้อมูลทั้งหมด เพราะข้อมูลบางอย่างจะถูกอ่านขึ้นมาจากฐานข้อมูลให้
- ใส่งレード : เป็นส่วนที่ให้อาจารย์ และเจ้าหน้าที่กรอกเกรดเข้าสู่ระบบ
- เปลี่ยนแปลงข้อมูลทั่วไปของนักศึกษา : เป็นส่วนที่ให้เจ้าหน้าที่เพิ่มแก้ไขข้อมูลนักศึกษา
- จัดการรายละเอียดหลักสูตร : เป็นส่วนที่ให้เจ้าหน้าที่เพิ่มแก้ไขข้อมูลหลักสูตร
- จัดการรายละเอียดองค์กร : เป็นส่วนที่ให้เจ้าหน้าที่เพิ่มแก้ไขข้อมูลหน่วยงานการศึกษา
- จัดการตารางเรียนและตารางสอบ : เป็นส่วนที่ให้เจ้าหน้าที่เพิ่มแก้ไขข้อมูลตารางเรียนและตารางสอบ
- เปิดรายวิชา : เป็นส่วนที่ให้เจ้าหน้าที่ เพิ่ม แก้ไข ข้อมูลการเปิดปิดรายวิชา
- กำหนดตารางเวลา : เป็นส่วนที่ให้เจ้าหน้าที่ เพิ่ม แก้ไข ข้อมูลกำหนดระยะเวลาต่าง ๆ เช่น ระยะเวลาการลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- จัดการรายวิชาหลังการลงทะเบียนล่วงหน้า : เป็นส่วนที่ให้เจ้าหน้าที่สั่งให้ระบบประเมินว่ามีรายวิชาใดที่มีผู้ลงทะเบียนน้อยกว่าการลงทะเบียนล่วงหน้า และให้สิทธิเจ้าหน้าที่ในการปิดรายวิชานั้น
- จัดการค่าใช้จ่ายต่างๆ : เป็นส่วนให้เจ้าหน้าที่ จัดการค่าใช้จ่ายต่างๆ เช่น ค่าธรรมเนียมการศึกษา
- จัดการใบทรานสคริป : เป็นส่วนให้เจ้าหน้าที่ ทำการ พิมพ์ใบทรานสคริปให้นักศึกษา
- คู่มือและพิมพ์รายงาน : เป็นส่วนที่ให้เจ้าหน้าที่ดู และพิมพ์รายงานต่าง ๆ จากระบบ
- แก้ไขข้อมูลการลงทะเบียน : เป็นส่วนที่ให้เจ้าหน้าที่และผู้ดูแลระบบแก้ไขข้อมูลการลงทะเบียนที่อาจเกิดความผิดพลาดขึ้นได้
- จัดการแอดเคาท์ของนักศึกษา : ให้ผู้ดูแลระบบสร้างหรือลบ รหัสผ่านให้กับนักศึกษา
- จัดการแอดเคาท์ของเจ้าหน้าที่ : ให้ผู้ดูแลระบบสร้างหรือลบ รหัสผ่านให้กับเจ้าหน้าที่ทะเบียน

5.5 การออกแบบการทำงานของฟังก์ชันต่าง ๆ

จะเห็นได้ว่าหลังจากขั้นตอนที่ผ่านมาเราได้ฟังก์ชันการทำงานต่าง ๆ ของระบบทั้งหมด ซึ่งเราสามารถมาทำการแยกออกแบบทีละฟังก์ชัน ซึ่งทำให้เราสามารถออกแบบได้ง่ายขึ้น สิ่งแรกที่เราจะต้องทำก็คือการอธิบายว่าในแต่ละฟังก์ชันมีกิจกรรมใดเกิดขึ้นบ้าง ซึ่งเราใช้แอกทิวิตี้ไดอะแกรมในการอธิบายสิ่งดังกล่าว

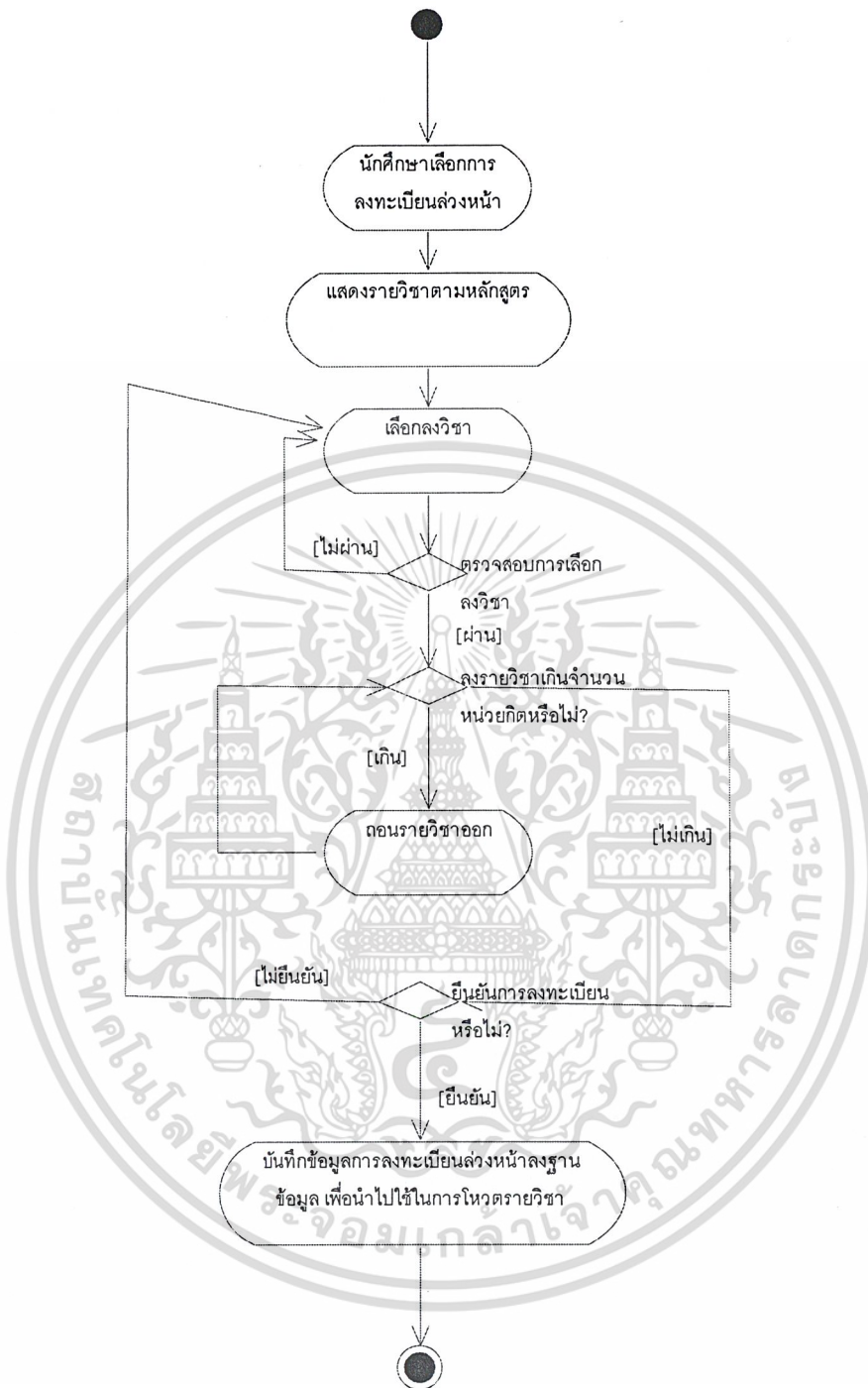
จากรูปที่ 5-3 จะเป็นแอกทิวิตี้ไดอะแกรมของ ยูสเคส ลงทะเบียนล่วงหน้า ซึ่งก็คือขั้นตอนการลงทะเบียนล่วงหน้านั่นเอง จากไดอะแกรมจะเห็นว่าเริ่มต้นเมื่อนักศึกษาเลือกที่จะทำการลงทะเบียนระบบจะตรวจสอบว่านักศึกษาค้นนี้ได้มีการลงทะเบียนล่วงหน้าหรือยัง หรือมีรายวิชาใดที่เขาลงทะเบียนล่วงหน้าไว้แล้วรายวิชานั้นถูกปิดไปหรือไม่ ซึ่งถ้าไม่มีปัญหาอะไร ระบบก็จะแสดงรายละเอียดค่าใช้จ่าย และการลงทะเบียนเพื่อให้นักศึกษาทำการยืนยันเมื่อ เสร็จขั้นตอนนี้ก็บันทึกข้อมูลลงสู่ระบบต่อไป ซึ่งแอกทิวิตี้ไดอะแกรมนี้เป็นหนึ่งในตัวอย่างของยูสเคสต่าง ๆ ซึ่งต้องทำเช่นนี้เหมือนกันจนครบ

สิ่งที่ต้องทำต่อไปก็คือการอธิบายว่าในแต่ละฟังก์ชันการทำงานนั้น มี คลาส ไตบ้าง และมีการทำงานสื่อสารกันอย่างไร ซึ่งเราใช้ซีควเนต์ไดอะแกรม(Sequence Diagram) เป็นตัวอธิบายสิ่งดังกล่าว ซึ่งแต่ละยูสเคสอาจมีมากกว่าหนึ่งซีควเนต์ไดอะแกรม หรืออาจ ไม่มีก็ได้

จากรูป 5-4 ก็เป็นตัวอย่างของซีควเนต์ไดอะแกรมของ ยูสเคส ลงทะเบียนปกติ ซึ่งจะเห็นได้ว่ามีการส่งข้อมูลกันระหว่างคลาสต่าง ๆ ทำให้เราสามารถเข้าใจและง่ายในการนำไปเขียนเป็นโปรแกรมต่อไป

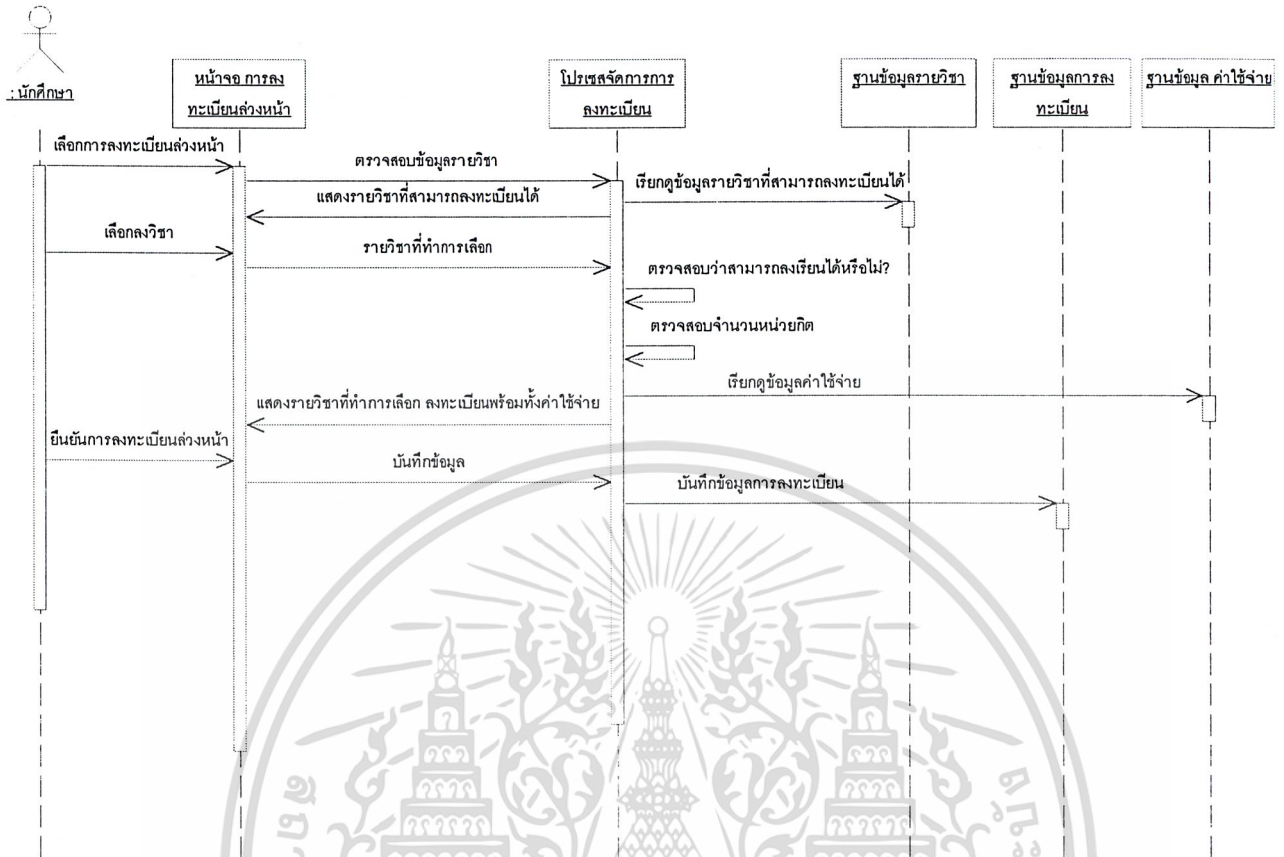
ในการการออกแบบระบบในส่วนของการเขียนไดอะแกรมต่าง ๆ เหล่านี้ค้ำึงถึงการนำไปเขียนเป็นโปรแกรมได้ นั่นหมายความว่าแต่ละคลาสที่ได้จากการเขียนไดอะแกรมก็จะนำไปเขียนเป็นคลาสจริง ๆ ดังนั้นการสกัดเพื่อหาคลาสเหล่านี้จึงต้องทำอย่างรอบคอบและค้ำึงถึงขนาดและหน้าที่ของแต่ละคลาสด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5-3 Activity Diagram ของ ยูสเคส ลงทะเบียนล่วงหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

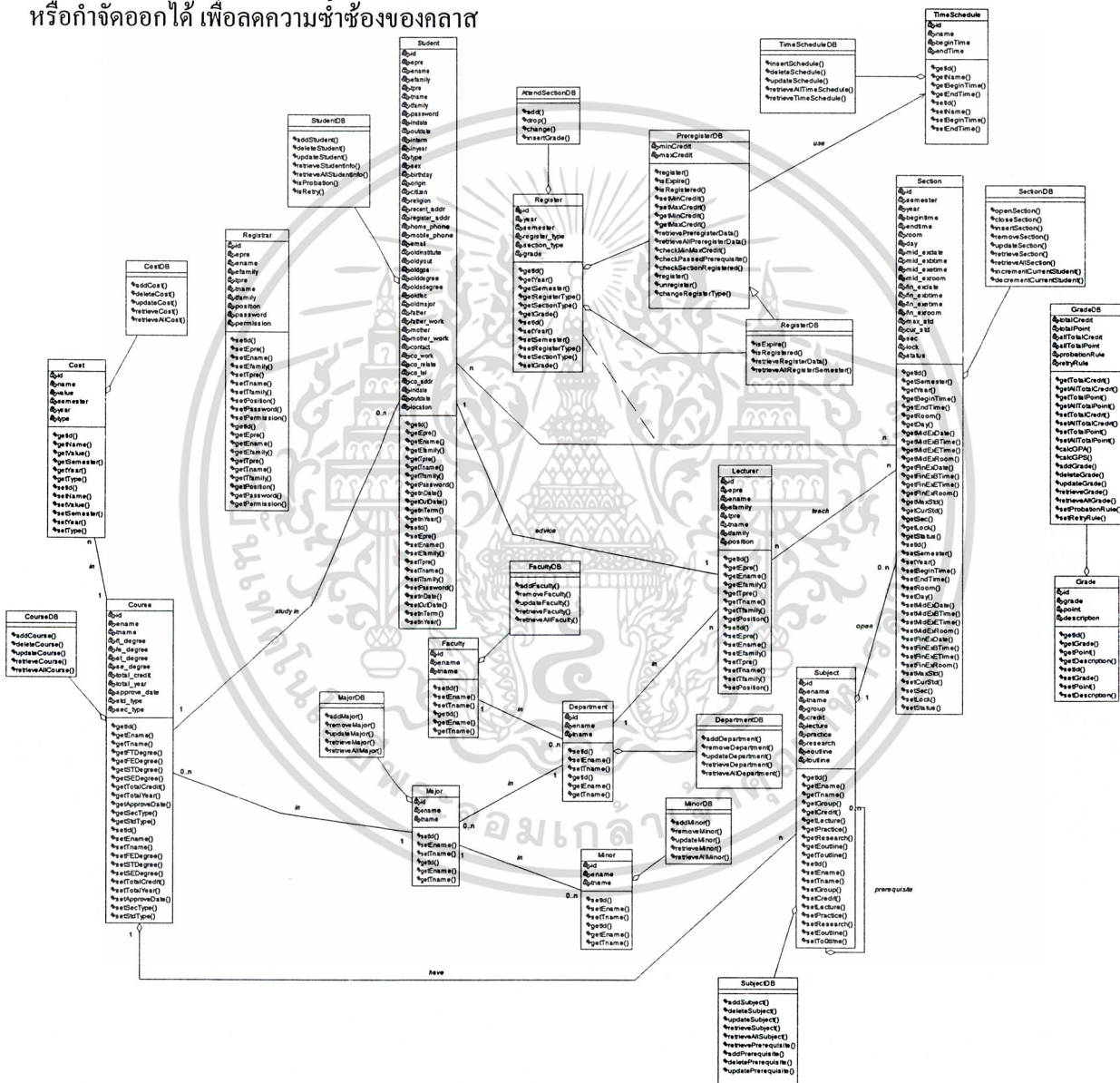


รูปที่ 5-4 Sequence Diagram ของ ยูสเคส ลงทะเบียนล่วงหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6 การออกแบบการรายละเอียดของคลาสต่าง ๆ

มาถึงขั้นตอนนี้เราจะได้คลาสที่ใช้ในระบบอย่างครบถ้วน เราก็จะนำคลาสเหล่านี้มากำหนดความสัมพันธ์ระหว่างกัน รวมถึงอาจจัดคลาสให้เป็นกลุ่มแล้วเก็บไว้ในแพ็คเกจ(Package) แล้วกำหนดความสัมพันธ์ระหว่างแพ็คเกจด้วยก็ได้ แล้วปรับแต่งซีควเนต์ไดอะแกรมเพื่อให้การส่งข้อมูลระหว่างคลาสอยู่ในรูปของการเรียกใช้เมธอด(Method) ให้ได้มากที่สุด กำหนดคุณสมบัติของคลาสแต่ละคลาส กำหนดแอตทริบิวต์(Attribute) และเมธอดของคลาสให้ครบถ้วน แล้วนำไปเขียนเป็นคลาสไดอะแกรม เขียนความสัมพันธ์ในรูปของการสืบทอดคุณสมบัติของคลาส คุรรายละเอียดที่คลาสของระบบว่ามีคลาสใดที่แยกกันได้ รวมกันได้ หรือกำจัดออกได้ เพื่อลดความซ้ำซ้อนของคลาส

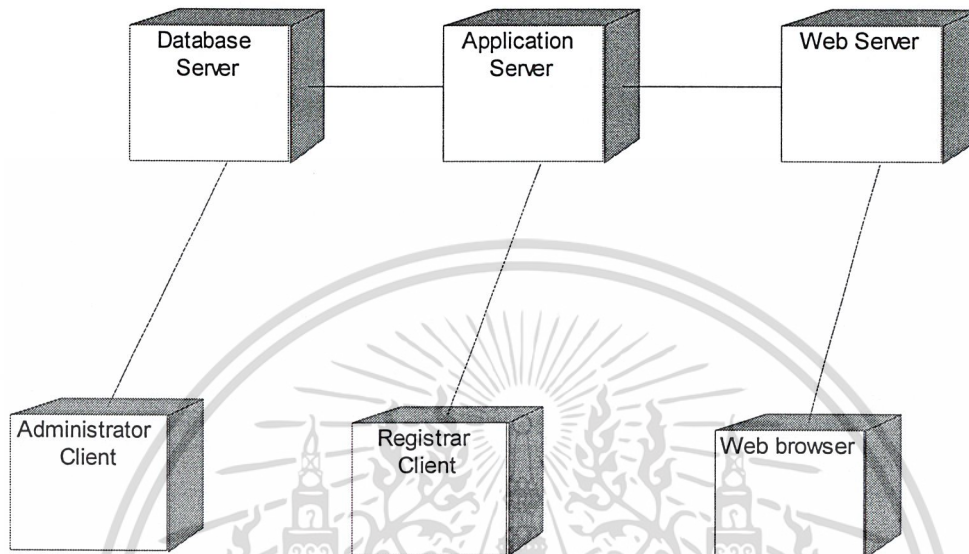


รูปที่ 5-5 Class Diagram ของระบบ

จากรูป 5-5 เป็นคลาสไดอะแกรมของระบบ ซึ่งจะอธิบายว่าคลาสแต่ละคลาสมีรายละเอียดอย่างไร และสัมพันธ์กับคลาสอื่น ๆ อย่างไร ใ้บริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.7 การออกแบบโครงสร้างของระบบ

เขียนดีพรอยเมนต์ไดอะแกรม(Deployment Diagram) ซึ่งจะต้องดูว่าระบบของเราประกอบด้วยโครงสร้างของระบบทางกายภาพอย่างไร



รูปที่ 5-6 Deployment Diagram ของระบบ

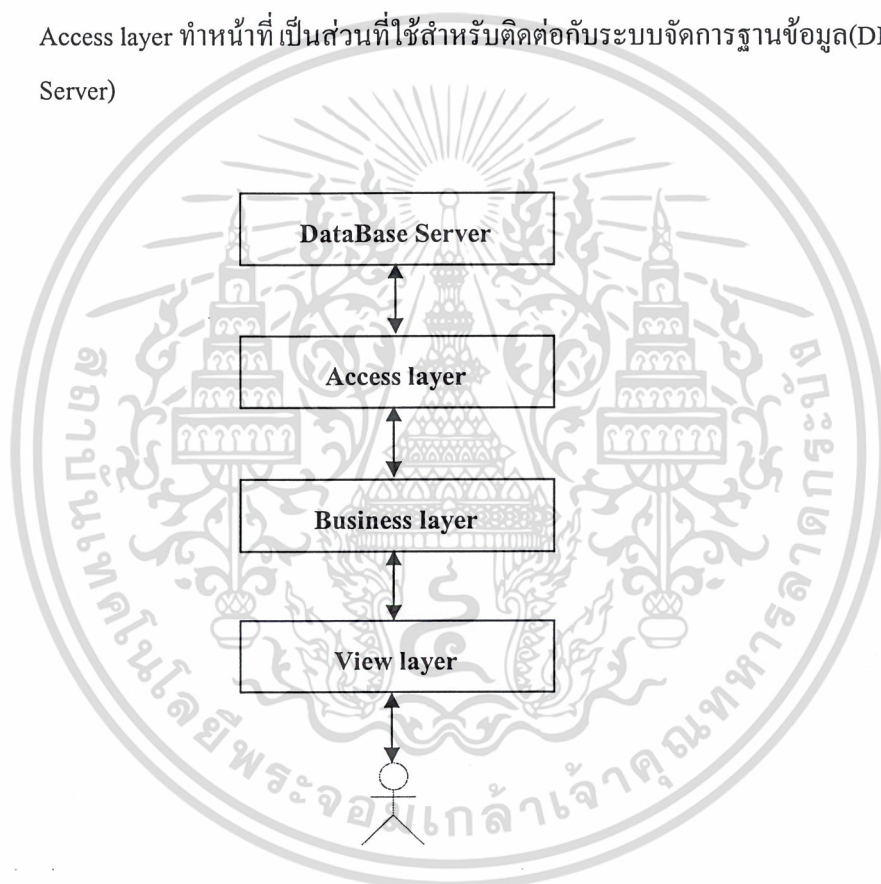
ที่ได้กล่าวมาทั้งหมดนี้คือขั้นตอนการวิเคราะห์และออกแบบระบบ ซึ่งได้ยกเอาไดอะแกรมบางไดอะแกรมมาเพื่ออธิบายขั้นตอนการทำงาน ส่วนไดอะแกรมทั้งหมดในโครงการนี้จะอยู่ในส่วนของภาคผนวก ก ท้ายเล่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.8 การออกแบบระบบ เพื่อสร้างระบบสารสนเทศ

มีการแบ่ง การทำงาน โดยภาพรวม เป็น 3 เลเยอร์ ด้วยกัน คือ

- 5.8.1 View layer เป็นส่วนของไคลเอนต์(Client) สำหรับผู้ใช้(User) โดยจะแบ่งเป็น 2 ส่วน ก็คือ ส่วนที่เป็นแอปพลิเคชัน (Application)ให้สำหรับเจ้าหน้าที่ทะเบียน และส่วนที่เป็นเว็บแอปพลิเคชัน (Web Application) สำหรับนักศึกษา
- 5.8.2 Business layer ทำหน้าที่กำหนดหน้าที่การทำงานหลัก ของระบบ เป็นการทำงาน เกี่ยวกับระบบทั้งหมด ซึ่งทำงานในลักษณะ เว็บเซิร์ฟเวอร์(Web Server)และ โปรแกรมประยุกต์
- 5.8.3 Access layer ทำหน้าที่ เป็นส่วนที่ใช้สำหรับติดต่อกับระบบจัดการฐานข้อมูล(DBMS Server)



รูปที่ 5-7 การแบ่งการทำงานของระบบสารสนเทศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การพัฒนาและทดสอบระบบ

ในส่วนของบทนี้จะได้กล่าวถึงขั้นตอนการพัฒนาและทดสอบระบบ ซึ่งเราจะเริ่มตั้งแต่การสร้างในส่วน of ฐานข้อมูลขึ้นมา ก่อน แล้วจึงทำการเขียนโปรแกรมซึ่งก็แบ่งออกเป็น 2 ส่วนก็คือส่วนที่เป็นแอฟพลิเคชัน ให้สำหรับเจ้าหน้าที่ทะเบียน และส่วนที่เป็นเว็บแอฟพลิเคชันสำหรับนักศึกษา หลังจากที่เรเขียนโปรแกรมเสร็จก็จะถึงขั้นตอนการทดสอบระบบ ซึ่งมีรายละเอียดดังนี้

6.1 การสร้างฐานข้อมูล

ในการสร้างฐานข้อมูลนั้นเราอ้างอิงจากการนำเอาโครงสร้างของคลาสต่าง ๆ ในระบบมาสร้างเป็นโครงสร้างตารางฐานข้อมูล และสร้างเป็นตัวฐานข้อมูลใน DB2 ซึ่งมีขั้นตอนดังนี้

6.1.1 นำเอาโครงสร้างของคลาสมาแปลงเป็นโครงสร้างตารางฐานข้อมูล(Database Schema)

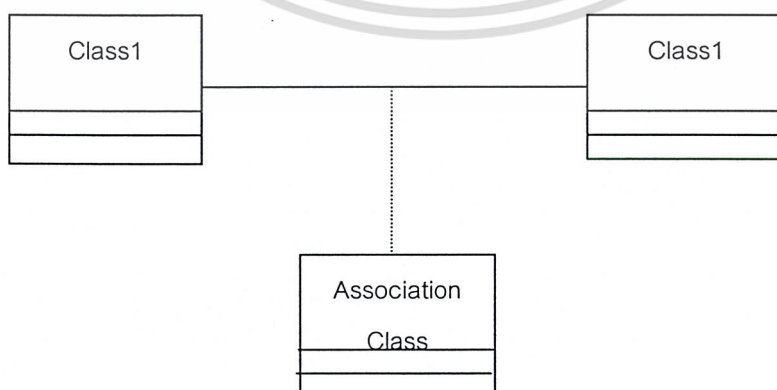
ซึ่งมีวิธีการดังนี้

การแปลงจาก คลาสไปเป็นตาราง

มีอยู่ 4 ทางที่จะแปลงจาก คลาสเป็น ตารางคือ one-to-one ,one-to-many,many-to-one many-to-many ซึ่งอาจจะแปลงได้หลากหลายขึ้นอยู่กับเหตุผลทางด้านประสิทธิภาพ,ความปลอดภัย,ความสะดวกในการค้นหา

การแปลงนั้นจะยึดหลักของฐานข้อมูลเชิงสัมพันธ์ (relational database) คือมีความสัมพันธ์แบบ many-to-many ,subtype,supertype

ความสัมพันธ์แบบ many-to-many จะต้องทำให้กลายเป็นความสัมพันธ์แบบ one-to-many โดยการสร้างตารางความสัมพันธ์ใหม่



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องยกย่องถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

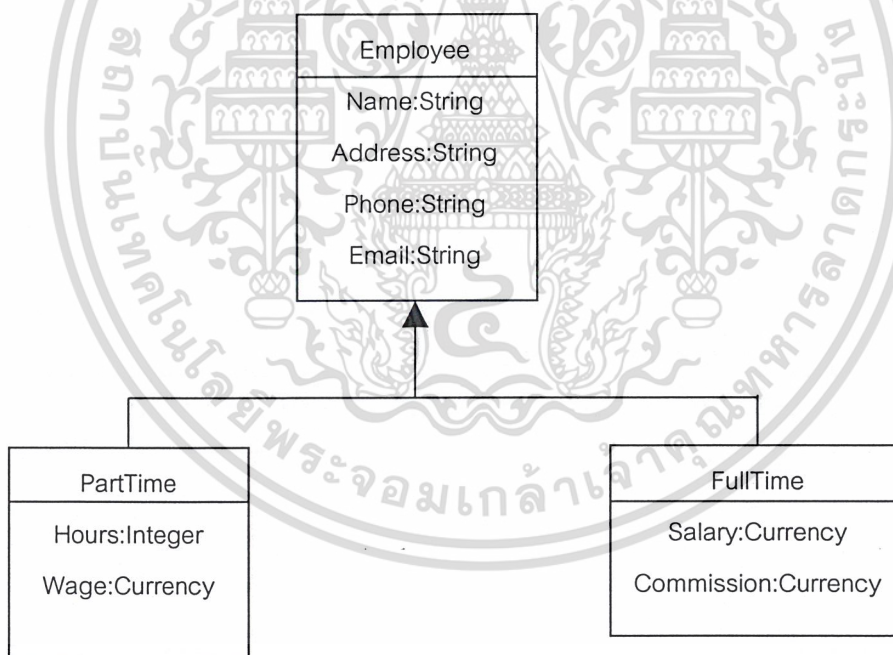
รูป ความสัมพันธ์ของคลาส

เราจะใช้หลักการของ entity-relationship(ER) ช่วยสำหรับความสัมพันธ์ระหว่างคลาส
เมื่อเราจะแปลงคลาสไปเป็นตารางเรามีอยู่ 3 ตัวเลือกคือ

1. หนึ่งตารางต่อหนึ่งคลาส
2. หนึ่งตารางต่อ คอนกรีตคลาส(concrete class)
3. หนึ่งตารางต่อไฮลาซี(hierarchy)

หนึ่งตารางต่อหนึ่งคลาสนั้นไม่ยุ่งยากแต่ละคลาสแปลงมาเป็นตารางที่เหมือนกันได้โดยตรง

หนึ่งตารางต่อ คอนกรีตคลาส(concrete class) เรียกว่า “rolling down” คือจาก ตาราง supertype ไปยัง ตาราง subtype ของมัน หนึ่งตารางต่อไฮลาซี(hierarchy) เรียกว่า “rolling up” คือจาก subtype ไปยัง supertype เราจะนำ แอททริบิวท์(attribute) ใน subtype และ supertype คลาสมาแปลงเป็นคอลัมน์ในตารางเดียวและจะมีคอลัมน์ใหม่เพิ่มขึ้นมาในตารางเพื่อเป็นตัวระบุถึง subtype เดิม



Logical design

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table Employee
PK emp_id:INTEGER
emp_name:VARCHAR(25)
emp_address: VARCHAR(45)
emp_phone: VARCHAR(10)
emp_email: VARCHAR(45)
emp_salary:DOUBLE PRECISION
emp_commission:DOUBLE PRECISION
emp_hours:INTEGER
emp_wage: DOUBLE PRECISION
emp_type:CHAR(1)

Database Design

จากตัวอย่างในรูปมีคลาส Employee เป็นคลาสแม่ และมี Partime ,Fulltime เป็นลูก โดยแปลงมาเป็นตาราง Employee 1ตารางแต่จะมีการสร้างคอลัมน์ใหม่ขึ้นมาในตารางที่มีชื่อว่า Emp_type เป็นตัวบ่งบอกว่า เป็น Employee ประเภทใดซึ่งจะทำให้ไม่ต้องเสียเวลาในการค้นหา 3 ตารางในการดูข้อมูลของ Employee

การแปลง แอททริบิวท์ ไปเป็น คอลัมน์

มีหลายทางที่จะแปลงจากแอททริบิวท์เป็นคอลัมน์แต่นั้น ไม่มีผลในการแปลงมาเป็นคอลัมน์มันอาจจะ มีผลในการแปลงจาก คลาสมาเป็นตาราง เราอาจจะมี แอททริบิวท์ในคลาสแต่อาจจะไม่มีอยู่ในฐานข้อมูลหรือ คอลัมน์ก็ได้

โปรแกรมประยุกต์อาจจะมีการคำนวณของยอดขายทั้งหมดแต่นั้นไม่จำเป็นว่ายอดขายทั้งหมดจะอยู่ในฐานข้อมูล โดย แอททริบิวท์ ที่ได้มาจากการคำนวณของข้อมูลในฐานข้อมูลจะถูกเรียกว่า “derived attribute” และจะไม่อยู่ในฐานข้อมูล

6.1.2 นำโครงสร้างตารางฐานข้อมูลที่ได้ไปทำการนอร์มัลไลซ์(Normalization)เราจะทำการนอร์มัลไลซ์จนโครงสร้างตารางฐานข้อมูลอยู่ในโนนอร์มัลฟอร์มที่ห้า(Fifth Normal Form) ซึ่งสามารถดูรายละเอียดของตารางที่ได้ในภาคผนวก ข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

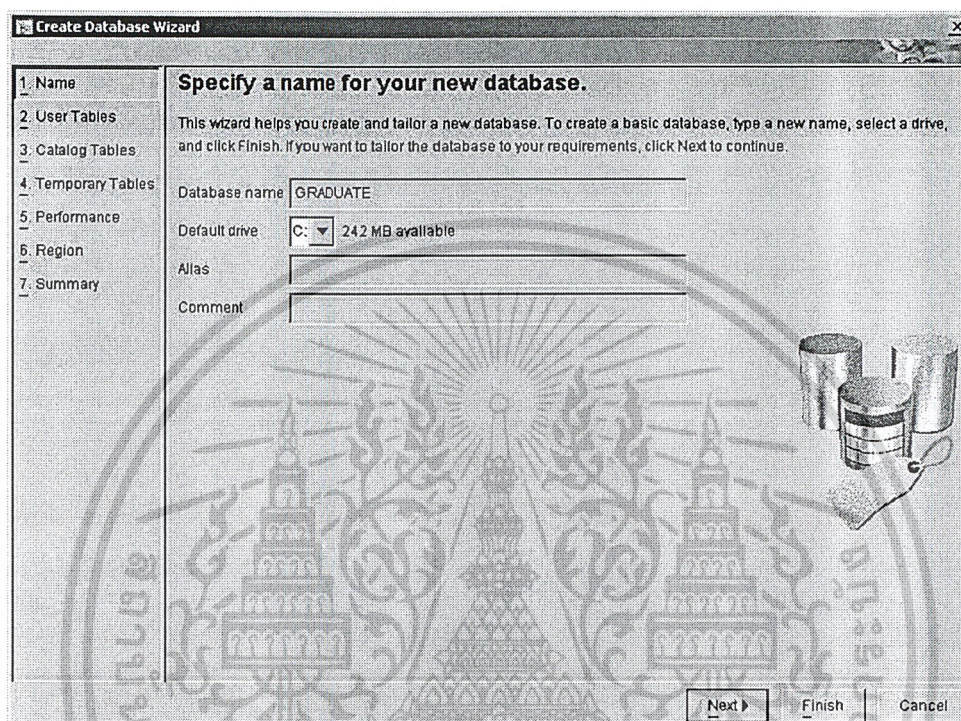
COURSE		
CO#	SMALLINT	รหัสหลักสูตร (PK) (Generate)
ENAME	VARCHAR(100)	ชื่อหลักสูตรภาษาอังกฤษ
TNAME	VARCHAR(100)	ชื่อหลักสูตรภาษาไทย
FT_DEGREE	VARCHAR(100)	ชื่อเต็มปริญญาภาษาไทย
FE_DEGREE	VARCHAR(100)	ชื่อเต็มปริญญาภาษาอังกฤษ
ST_DEGREE	VARCHAR(100)	ชื่อย่อปริญญาภาษาไทย
SE_DEGREE	VARCHAR(100)	ชื่อย่อปริญญาภาษาอังกฤษ
DEGREE	VARCHAR(1)	ปริญญาโท / เอก (M:Master,D:Doctor)
MAJOR#	VARCHAR(2)	รหัสสาขาวิชา (FK)
FACT#	VARCHAR(2)	รหัสคณะ (FK)
DEPT#	VARCHAR(2)	รหัสภาควิชา (FK)
MINOR#	VARCHAR(2)	รหัสแขนงวิชา (FK)

ตารางที่ 6-1 ตัวอย่างตารางที่ได้มาจากการแปลงคลาสไดอะแกรมมาเป็นโครงสร้างตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1.3 กำหนดรายละเอียดของชนิดข้อมูล(Data Type) ของแต่ละแอตทริบิวต์โดยอ้างอิงตามชนิดข้อมูลของ DB2

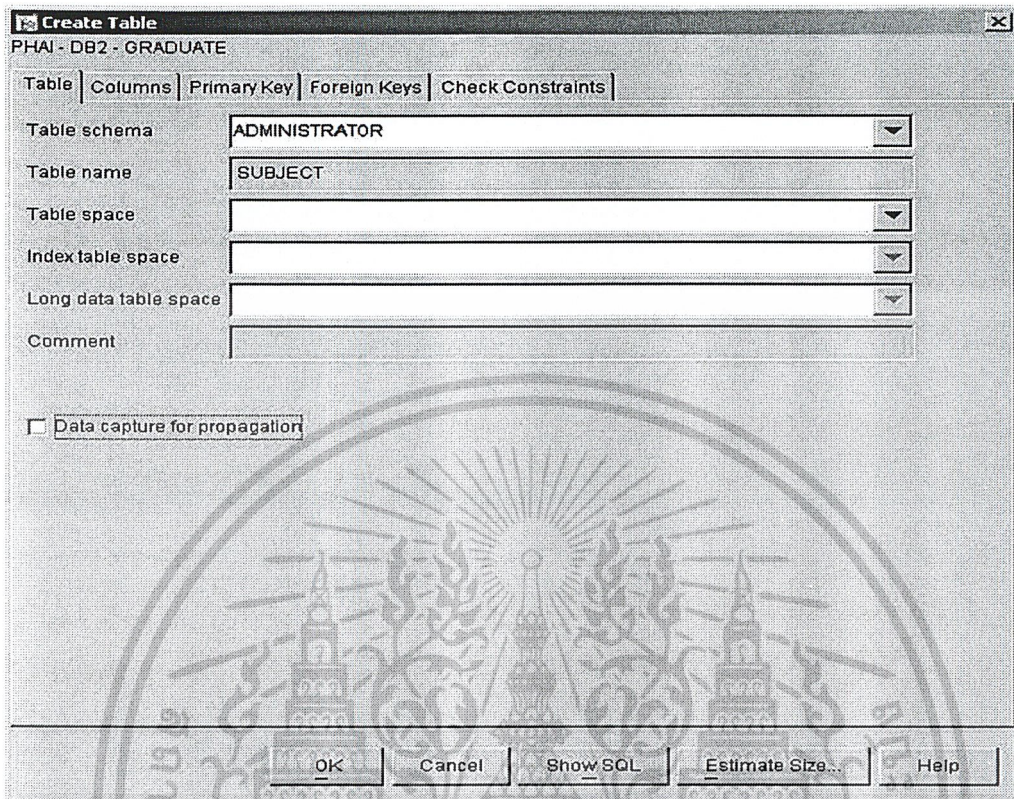
6.1.4 ทำการสร้างฐานข้อมูลบน DB2 ขึ้น โดยใช้ โปรแกรม Control Center



รูปที่ 6-1 การสร้างฐานข้อมูลบน DB2 โดยใช้โปรแกรม Control Center

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1.5 ทำการสร้างตารางแต่ละตาราง บนฐานข้อมูล ที่เราสร้างขึ้น โดยใช้โปรแกรม Control Center



รูปที่ 6-2 การใช้ Control Center เพื่อการสร้างตาราง

6.1.6 ทำการกรอกข้อมูลบางอย่างเพื่อเป็นตัวอย่างในการทดลองเขียน โปรแกรมโดยใช้โปรแกรม Command Center

6.1.7 ในการติดต่อกับฐานข้อมูลต่าง ๆ เราก็จะใช้โปรแกรม Command Center ซึ่งสามารถทำงานด้วยภาษาเอสคิวแอล(SQL)ได้ และโปรแกรม Control Center ซึ่งจะเป็นตัวจัดการในการบริหารฐานข้อมูล

หลังจากทั้ง 7 ขั้นตอนนี้เราก็จะมีฐานข้อมูลอยู่บนเครื่องเซิร์ฟเวอร์ที่เราสามารถติดต่อเข้ามาใช้งานได้จากเครื่องคอมพิวเตอร์อื่นได้โดยผ่านทางโพรโทคอลที่ซีพีทีพีไอพี(TCP/IP Protocol)

6.2 การเขียนโปรแกรมในส่วนของแอปพลิเคชัน

แอปพลิเคชันที่สร้างขึ้นนี้จากข้างต้นที่เคยกล่าวแล้วว่าใช้ภาษาจาวา ซึ่งเป็นภาษาที่มีข้อดีหลายประการและรองรับการพัฒนาโปรแกรมแบบ Object Oriented ได้เป็นอย่างดี รวมถึงใช้โปรแกรม Forte for java เป็นเครื่องมือในการพัฒนา ในการเขียนโปรแกรมนั้นค่อนข้างมีรายละเอียดมากมาย และสามารถหาความรู้เอกสารเป็นเอกสารที่ส่งวันเวสสำหรับการใช้งานเพื่อการศึกษาก็มีแน่นอน เหมือนญาติเหมาไปเซประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหลดได้ฟรีจากเว็บไซต์ของชั้น ผู้พัฒนาจะพยายามทำให้เนื้อหาง่ายที่สุดในการทำความเข้าใจโดยการแบ่งเป็นข้อ ๆ ตามลำดับขั้นตอนดังนี้

6.2.1 ออกแบบยูสเซอร์อินเทอร์เฟซ(User Interface) โดยแบ่งโปรแกรมออกเป็นส่วน ๆ ตามยูสเคสในแต่ละยูสเคสก็นำมาแบ่งเป็นส่วนย่อย ๆ อีกตามซีเควนซ์ไดอะแกรม โดยให้แต่ละส่วนเป็นพาแนล(Panel)

6.2.2 หลังจากที่ทำการแบ่งเป็นพาแนลได้แล้วนั้น แต่ละพาแนลก็หมายถึงหนึ่งคลาสนั่นเอง เราจึงสามารถแยกแต่ละพาแนลไปเขียนโปรแกรม อย่างอิสระ แล้วจึงค่อยนำมารวมกันในภายหลัง

6.2.3 ในการเขียนโปรแกรมติดต่อกับฐานข้อมูลนั้นเราใช้ JDBC เป็นตัวเชื่อมต่อให้โดยมีการระบุถึงเซิร์ฟเวอร์ฐานข้อมูลด้วยหมายเลขไอพี(IP Address)

6.2.4 หลังจากขั้นตอนนี้เราสามารถแยกแต่ละส่วนมาเขียนโปรแกรมได้ ซึ่งแยกไปตามคลาสต่าง ๆ

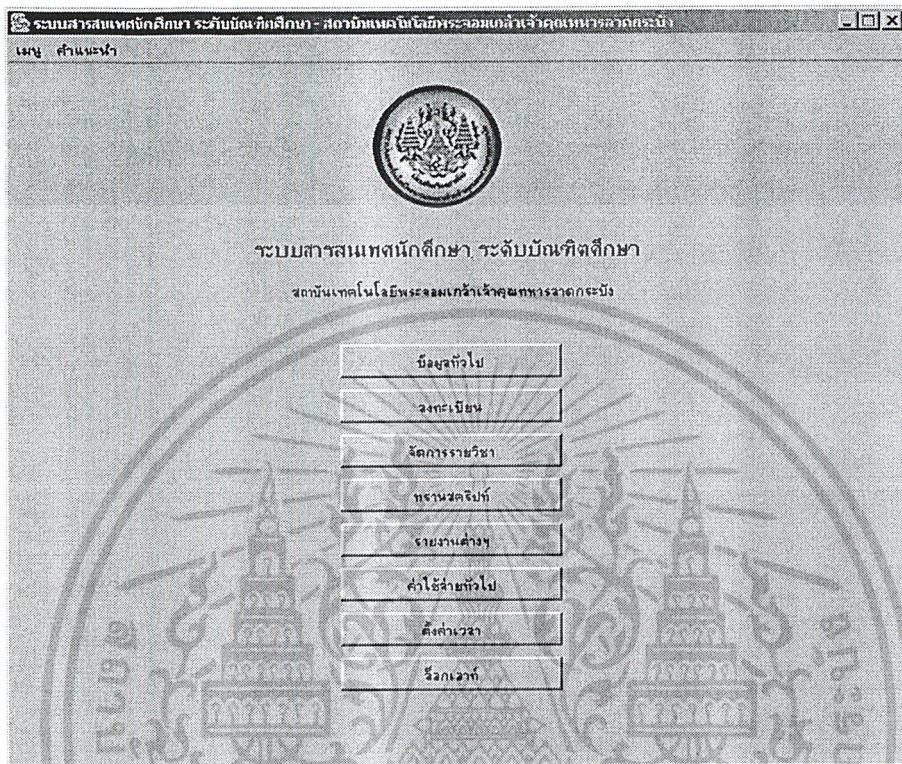
```
try
{
    Class.forName("COM.ibm.db2.jdbc.net.DB2Driver").newInstance();
    con = DriverManager.getConnection(url, uname, passwd);
}
catch(ClassNotFoundException cnfex)
{
    System.err.println("Failed to load JDBC/ODBC driver.");
    cnfex.printStackTrace();
}
catch(SQLException sqlx)
{
    System.err.println("Unable to connect");
    sqlx.printStackTrace();
}
catch(Exception ex)
{
    ex.printStackTrace();
}
```

ตัวอย่างโค้ดการติดต่อกับฐานข้อมูลด้วย JDBC

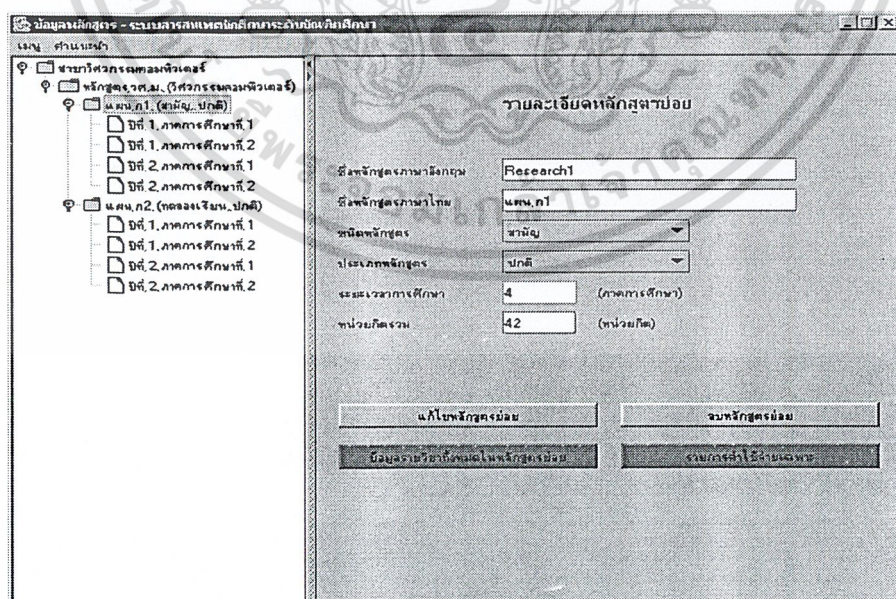
6.2.5 สิ่งหนึ่งที่ต้องคำนึงในการเขียนโปรแกรมก็คือข้อกำหนดต่าง ๆ ที่ได้ทำการศึกษาจากระบบจริง ซึ่งเราต้องเขียนโปรแกรมให้มีการตรวจสอบตรงตามข้อกำหนดเหล่านั้นด้วย ยกตัวอย่างเช่นนักศึกษาในระดับเอกสาธิตเป็นเอกสารทส่งงานไว้สำหรับการศึกษาเท่านั้น เมื่อญาติเดินทางไปต่างประเทศด้านการค้าบัณฑิตศึกษานั้นสามารถเข้ามานักศึกษาใหม่ได้ทั้งในภาคเรียนที่ 1 และภาคเรียนที่ 2 ดังนั้นการจัดแบ่งกลุ่มไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุตบแต่งสิ่งเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นักศึกษาจึงต้องแบ่งตามภาคและปีที่เข้า ไม่สามารถแบ่งตามรหัสนักศึกษาแต่เพียงอย่างเดียวได้เหมือนในระดับปริญญาตรี

6.2.6 เมื่อเราเขียน โปรแกรมแต่ละส่วนเสร็จก็จะนำมาประกอบในส่วนของพาเนลหลัก



รูปที่ 6-3 รูปส่วนอินเทอร์เฟซของ MainPanel



เอกสารนี้เป็นเอกสารที่สงวน **รูปที่ 6-4 รูปส่วนอินเทอร์เฟซของ รายละเอียดหลักสูตร** หน้าไปใช้ประโยชน์ด้านการค้า ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากขั้นตอนข้างต้นอาจไม่สามารถอธิบายลงในรายละเอียดทั้งหมดได้ซึ่งสามารถดูในรายละเอียดจากซอสโค้ด(Source Code)

6.3 การทดสอบโปรแกรมในส่วนของแอปพลิเคชัน

การทดสอบนี้จะทดสอบในกรณีของโครงสร้างของหน่วยงานการศึกษา ดังขั้นตอนต่อไปนี้

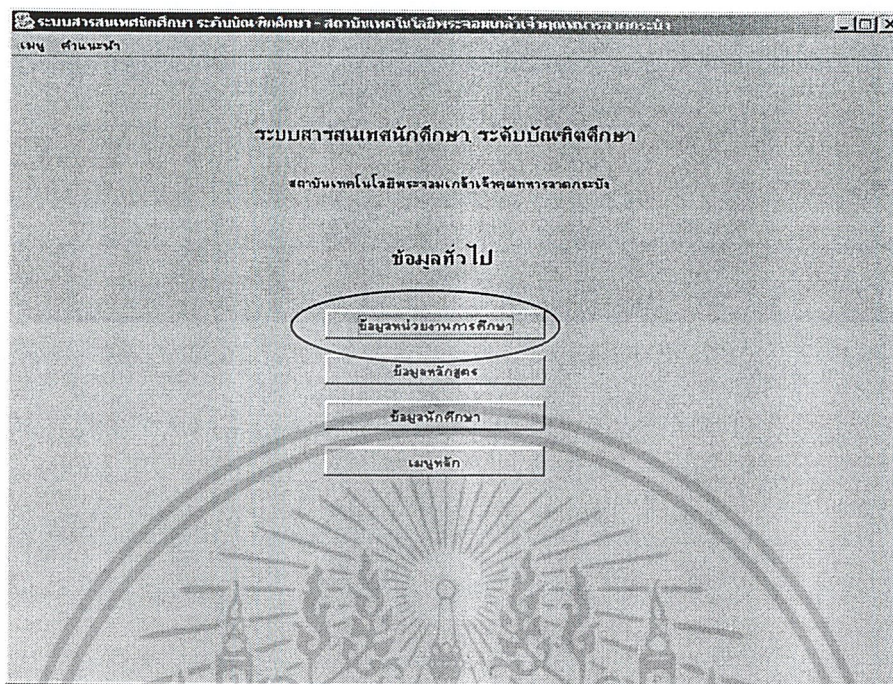
1. ล็อกอินเข้าสู่ระบบ ดังรูป โดย รหัสเจ้าหน้าที่ และ รหัสผ่านได้จากตาราง Registrar ในฐานข้อมูลของ DB2

รูปที่ 6-5 แสดงหน้าจอการล็อกอิน

REG#	EP...	ENAME	EFAMILY	TP...	TNAME	TFAMILY	POSITION	PASSWORD	PERMISSION
217	Miss	Supanee	Sompran	อ.บ.	คุณาณี	สมปราช	อธิการบดี	123	

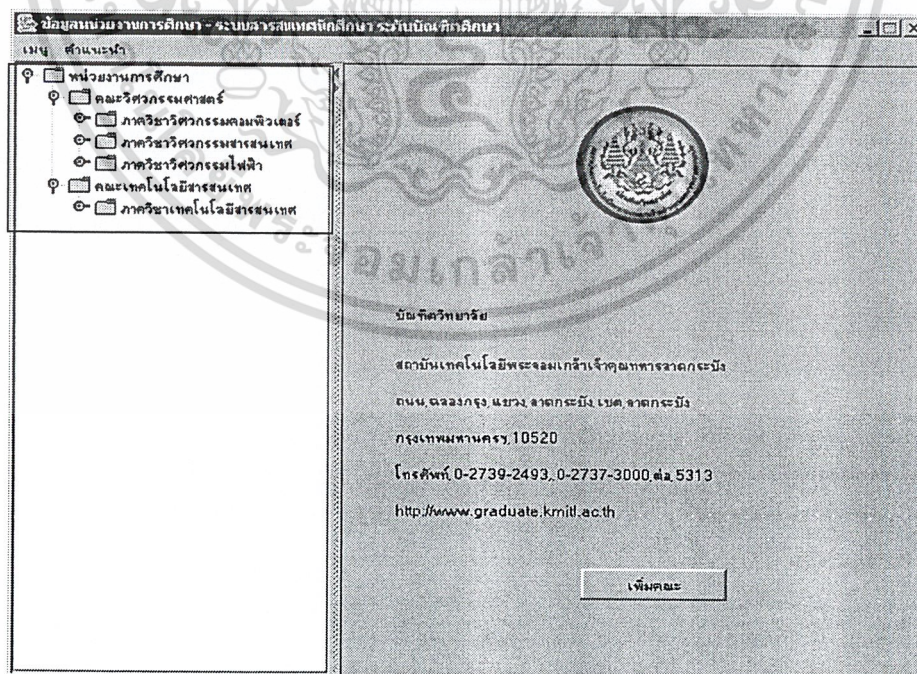
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 6-6 รูปแสดงข้อมูลที่อยู่ในฐานข้อมูล อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ในหมวดของข้อมูลทั่วไปเลือกเมนูข้อมูลหน่วยงานการศึกษา



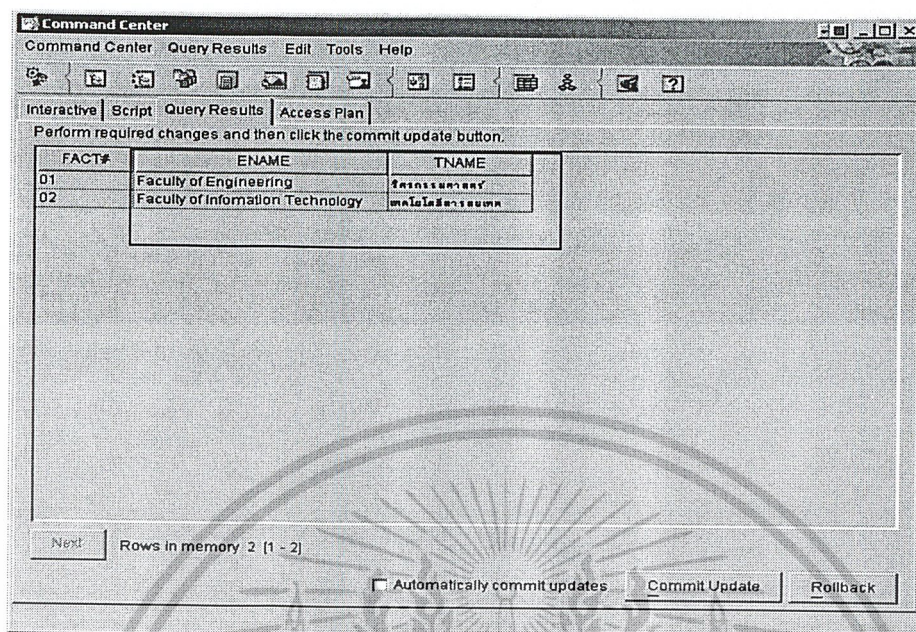
รูปที่ 6-7 แสดงการเลือกเมนู ข้อมูลหน่วยงานศึกษา

3. จะได้น้ำตาของโครงสร้างของข้อมูลหน่วยงานการศึกษา ลักษณะของทรีโมเดลดังรูป



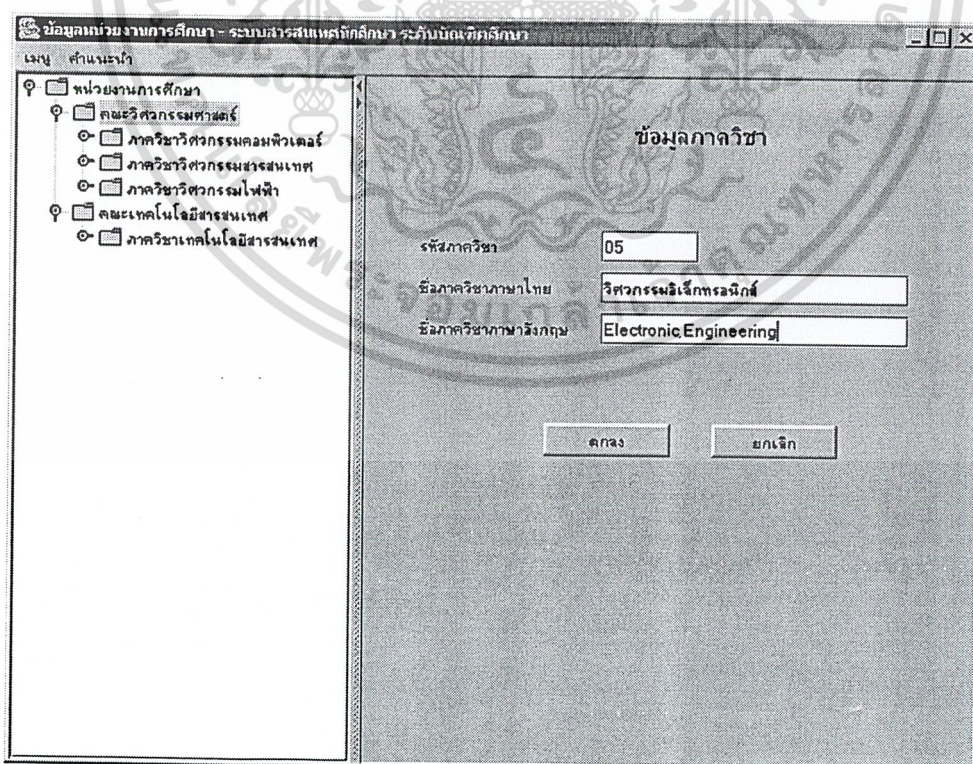
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในของคณะและภาควิชา อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าการณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ลองเทียบกับข้อมูลในตาราง Faculty ใน Command Center ของ db2 จะพบว่า มี 2 คณะเท่านั้นดังรูป



รูปที่ 6-9 แสดงข้อมูลในตาราง Faculty

5. ทำการทดสอบการเพิ่มภาควิชาในคณะวิศวกรรมศาสตร์ โดยคลิกปุ่มเพิ่มภาควิชา จะได้ช่องใส่ข้อมูลดังรูปที่



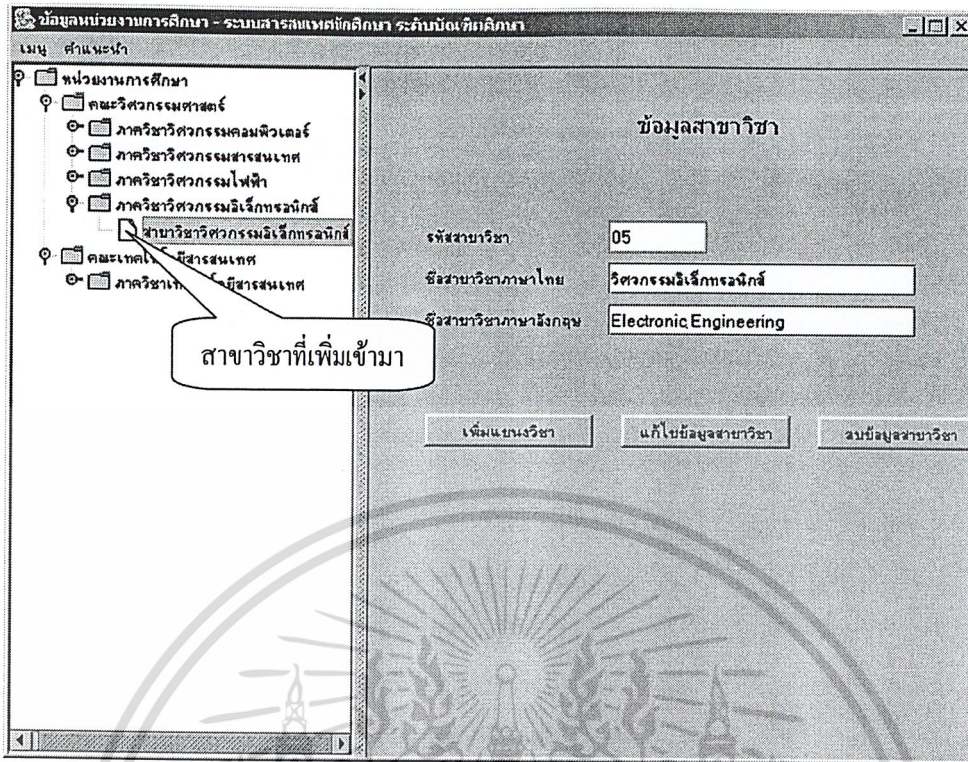
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนเว็บไซต์หรือสื่อออนไลน์ใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ผู้รับอนุญาตนำข้อมูลไปใช้
รูปที่ 6-10 แสดงการเพิ่มภาควิชา

6. ทำการใส่ข้อมูลลงไป แล้วกดปุ่มตกลง
7. ทดสอบดูกับในฐานข้อมูลของตาราง Department จะได้ดังรูป

รูปที่ 6-11 แสดงผลการเพิ่มภาควิชา

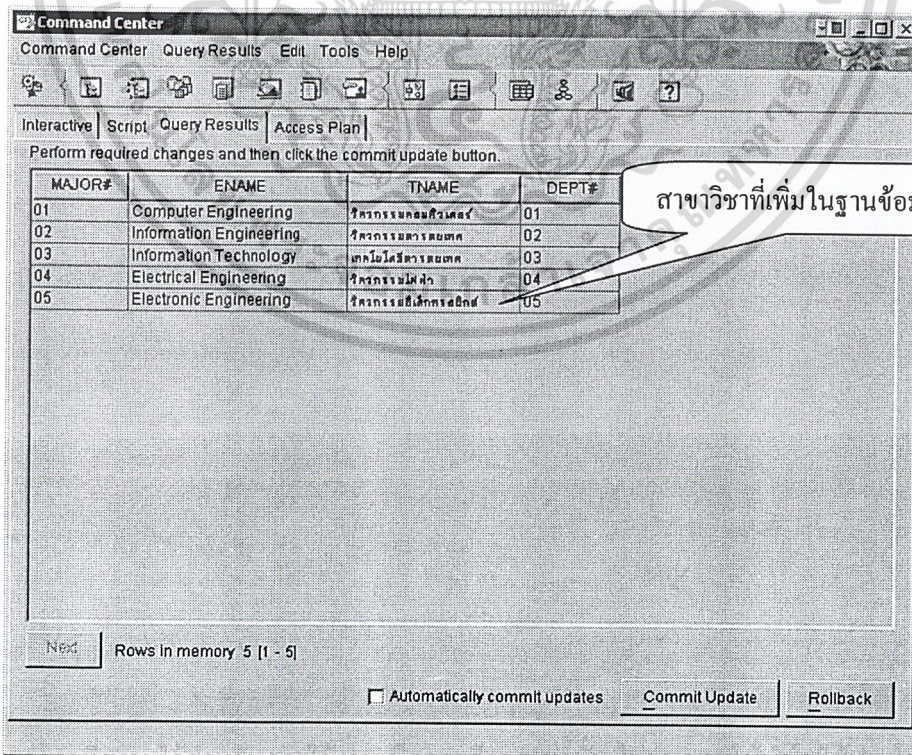
8. ทดสอบการเพิ่มสาขาวิชาในภาควิชาวิศวกรรมอิเล็กทรอนิกส์ ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 6-12 แสดงการเพิ่มสาขาวิชาไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



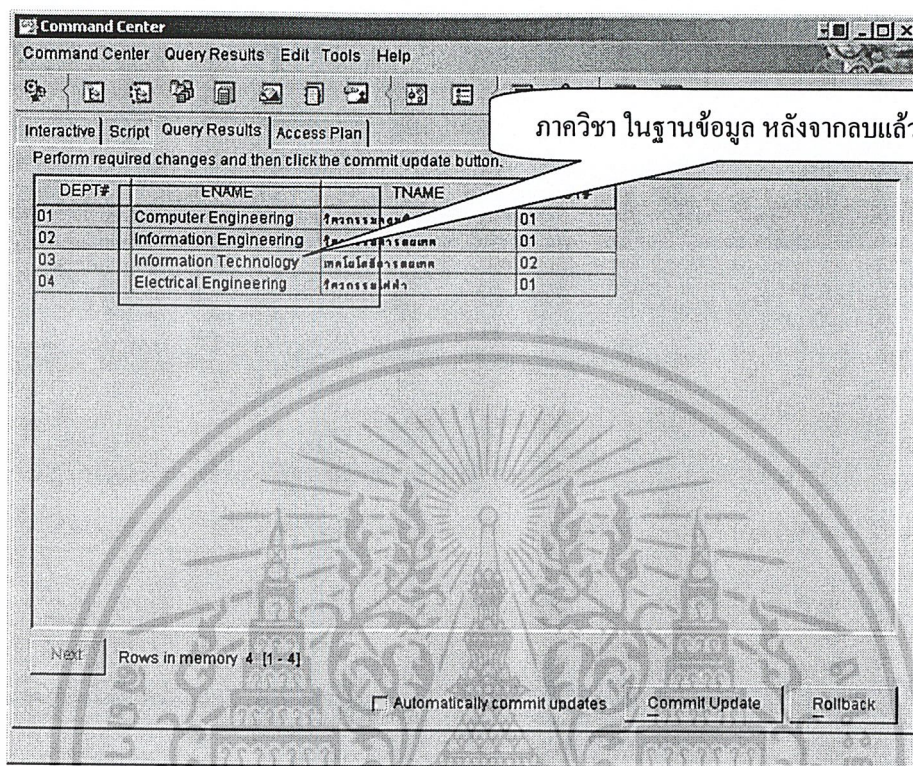
รูปที่ 6-13 แสดงผลการเพิ่มสาขาวิชา

9. หลังจากเพิ่มข้อมูลสาขาวิชา แล้วลองเทียบกับตาราง Major ในฐานข้อมูล



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การเชิงพาณิชย์หากมีข้อผิดพลาด กรุณาแจ้งผู้พิมพ์เพื่อปรับปรุงข้อมูลด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกที่ รูปที่ 6-14 แสดงข้อมูลหลังจากการเพิ่มสาขาวิชาของเอกสารทุกครั้งที่มีการนำไปใช้

10. ลองทดสอบการลบสาขาวิชา วิศวกรรมอิเล็กทรอนิกส์โดยการกดปุ่ม ลบข้อมูลสาขาวิชา
11. หลังจากนั้นจึงจะลบภาควิชา วิศวกรรมอิเล็กทรอนิกส์ได้โดยกดปุ่มลบภาควิชา



รูปที่ 6-15 แสดงข้อมูลในตาราง Department

จะเห็นว่ากรทดสอบ Application Program ในส่วนของข้อมูลหน่วยงานการศึกษา เป็นไปได้่างถูกต้องตามกระบวนการที่ได้ออกแบบไว้ ในส่วนอื่นๆก็สามารถทดสอบได้ในกรณีเดียวกัน ซึ่งยกตัวอย่างไว้เพียงส่วนของข้อมูลการศึกษาเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า . ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

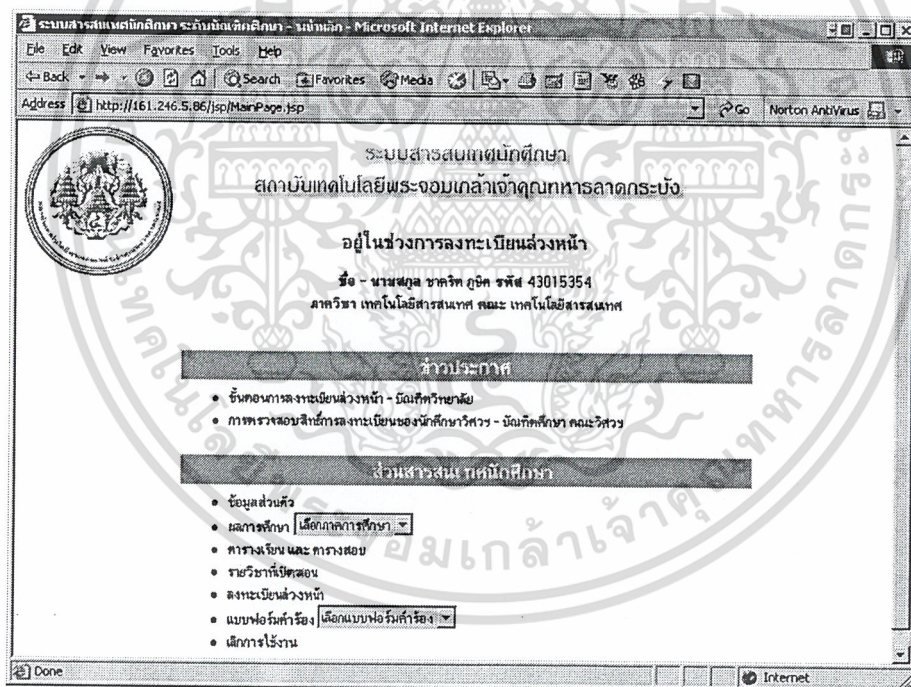
6.4 การเขียนโปรแกรมในส่วนของเว็บแอปพลิเคชัน

การเขียนโปรแกรมในส่วนนี้จะแตกต่างจากการเขียนโปรแกรมในส่วนแอปพลิเคชันเพราะว่า การแสดงผลในส่วนนี้จะอยู่บนเว็บเบราว์เซอร์(Web Browser) ดังนั้นการเขียนโปรแกรมแบบนี้จึงอยู่บนสิ่งแวดล้อมอินเทอร์เน็ตโดยใช้เทคโนโลยีของ Servlet, JSP และ JavaBean ซึ่งได้กล่าวถึงทฤษฎีทั้งสองเรื่องนี้แล้วในบทที่ 3 แล้ว

6.4.1 เนื่องจากโปรแกรมในส่วนนี้จะอยู่บนเว็บเบราว์เซอร์ เราจึงต้องมี เว็บเซิร์ฟเวอร์(WebServer) ที่รองรับเทคโนโลยีของ Servlet และ JSP ซึ่งในโครงการนี้เราใช้ Apache TOMCAT

6.4.2 เราสามารถเขียนโปรแกรมของเราทั้งในส่วนคลาสที่เป็น Servlet, JSP และ JavaBean ไว้ในโฟลเดอร์(Folder) ที่ Apache TOMCAT สร้างไว้ให้ เราก็จะสามารถทดลองโปรแกรมที่เราเขียนได้โดยผ่านทางเว็บเบราว์เซอร์บนเครื่องเราเอง

6.4.3 ขั้นตอนต่อไปก็คือการออกแบบยูสเซอร์อินเทอร์เฟซในแต่ละส่วน แล้วก็สามารถนำไปเขียนโปรแกรม โดยแยกจากกันเพื่อทดลองทีละส่วนได้



รูปที่ 6-16 การทดลองใช้งานผ่านเว็บเบราว์เซอร์

6.4.4 ในขั้นตอนนี้เราสามารถเขียนคลาสต่าง ๆ ขึ้นมาใช้งานได้ตามปกติ การติดต่อกับฐานข้อมูลก็ทำคล้ายกับการติดต่อในส่วนแอปพลิเคชัน แต่คลาสที่เราสร้างขึ้นเพื่อนำไปใช้ควรสร้างให้เป็น JavaBean เพื่อจะมีความสามารถสูงสุดในการนำไปใช้ใน Servlet และ JSP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

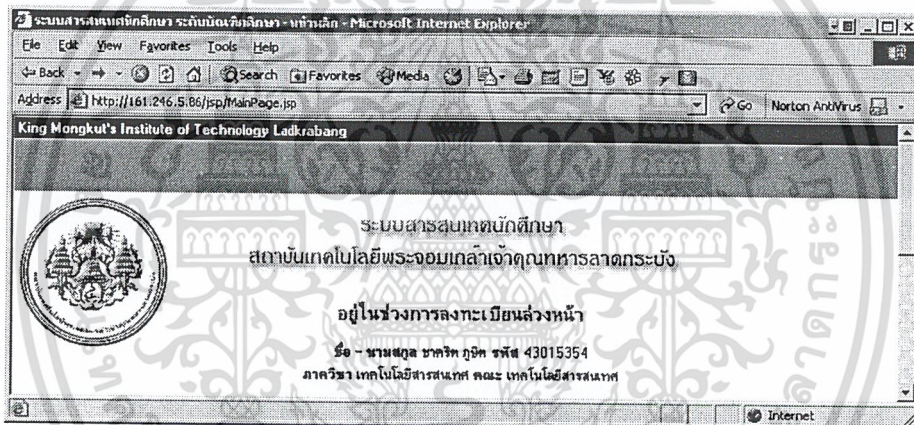
6.4.5 นอกจากใช้ Forte for java แล้วในการเขียน JSP ซึ่งเป็นการทำงานแบบ Server Side Included อย่างหนึ่งจำเป็นต้องสร้างโครงร่างของหน้า html ขึ้นมาก่อนซึ่งในที่นี้เราใช้ Macromedia Dreamweaver เข้ามาช่วยเสริมการทำงานในส่วนนี้ด้วย

6.4.6 นำส่วนต่าง ๆ มาประกอบกันให้สามารถทำงานเชื่อมต่อกันได้ด้วยการกำหนดลิงก์(Link) ให้เชื่อมโยงถึงกัน

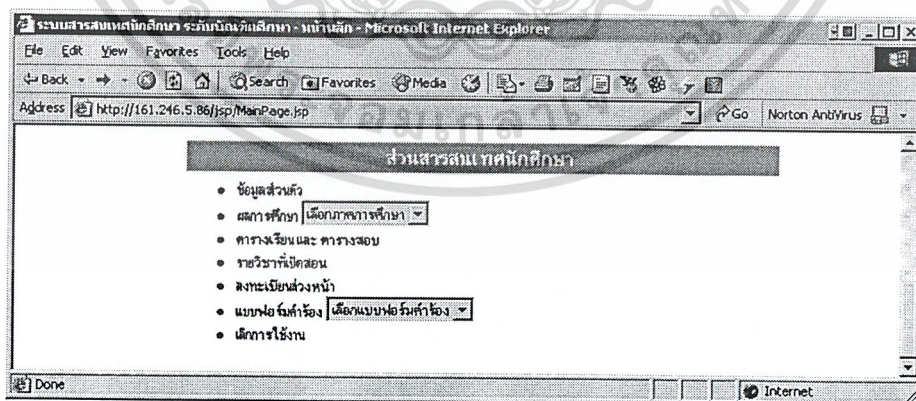
6.5 การทดสอบโปรแกรมในส่วนของเว็บแอปพลิเคชัน

การทดสอบในส่วนของเว็บแอปพลิเคชันนั้น เพื่อให้เข้าใจในการทำงานของระบบ ในที่นี้จะยกตัวอย่างการลงทะเบียนเว็บของนักศึกษา

6.5.1 การลงทะเบียนเว็บนั้น เป็น 1 ในเมนูที่มีการเปลี่ยนแปลงตามเวลาปัจจุบัน กล่าวคือ หากวันที่นักศึกษามีการล็อกอินเข้ามาอยู่ในช่วงของการลงทะเบียนเว็บ เมนูการลงทะเบียนเว็บก็จะสามารถทำงานได้(Active) ซึ่งจะสามารถตรวจสอบช่วงเวลาได้จาก แถบแสดงเวลา

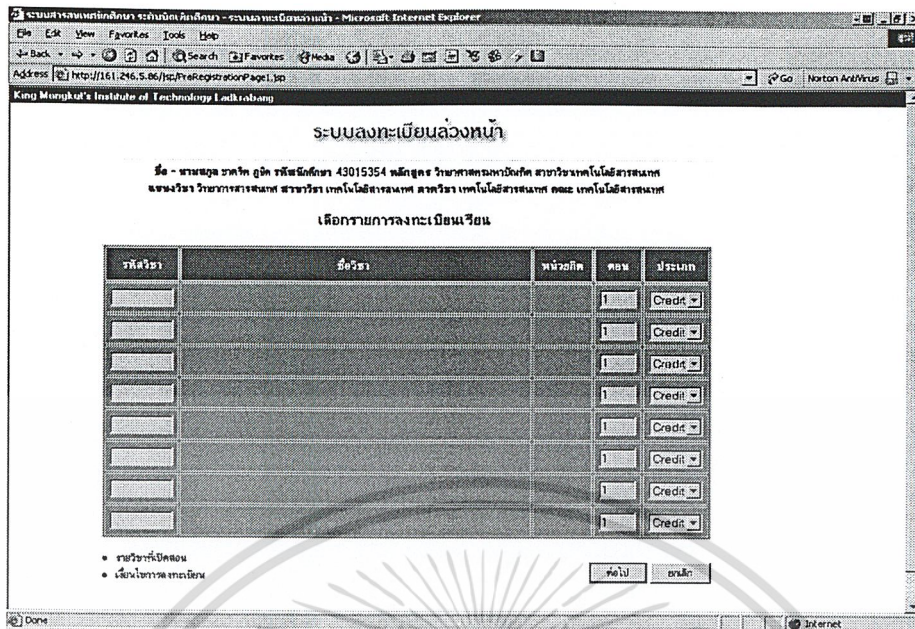


รูปที่ 6-17 แสดงช่วงเวลาปัจจุบัน



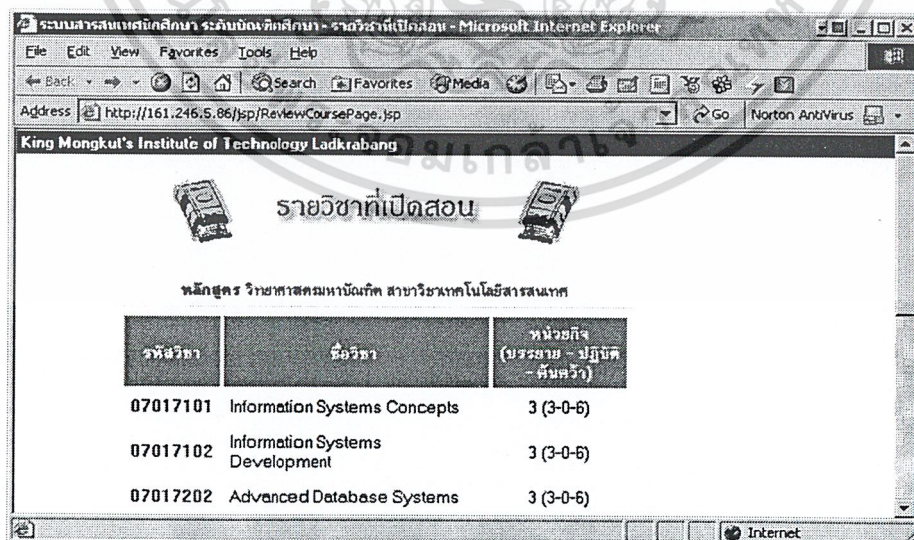
รูปที่ 6-18 แสดงเมนูที่เปลี่ยนแปลงตามช่วงเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

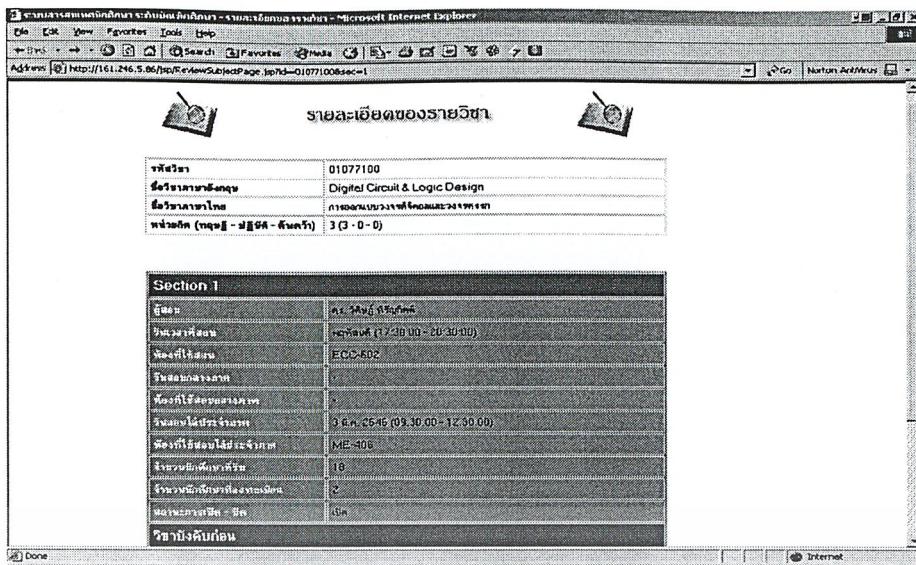


รูปที่ 6-19 แสดงให้เลือกลงทะเบียนตามรายวิชาที่เปิดสอนตามหลักสูตร

6.5.2 นักศึกษาจะทำการเลือกลงทะเบียนรายวิชาโดยป้อนรหัสวิชา ตามรายวิชาที่เปิดสอนตามหลักสูตร โดยนักศึกษาสามารถตรวจสอบรายชื่อวิชาที่เปิดสอนได้จากเมนู รายชื่อวิชาที่เปิดสอน โดยนักศึกษาสามารถเข้าไปดูรายละเอียดของแต่ละรายวิชา คำอธิบายรายวิชา และดูว่าวิชาดังกล่าวมีวิชาบังคับก่อนหรือไม่ หรือดูสถานะของรายวิชาว่าเปิด หรือปิด สามารถดูจำนวนนักศึกษาที่ลงทะเบียนไปแล้ว และจำนวนนักศึกษาที่รับสูงสุดได้ ในกรณีที่รายวิชามีการจำกัดจำนวนผู้ที่ลงทะเบียนรายวิชาดังกล่าวไว้



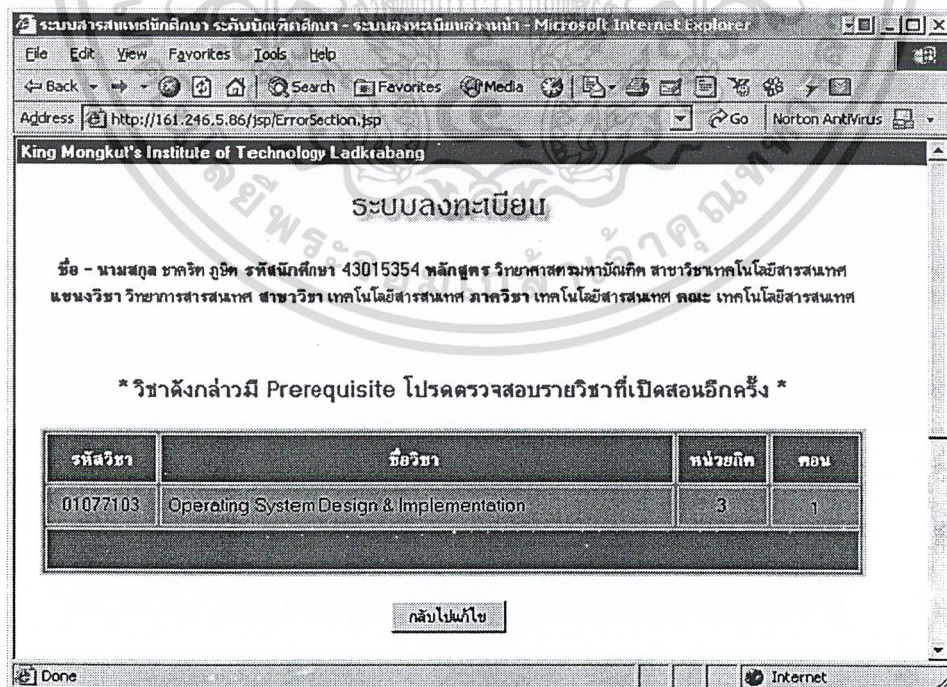
เอกสารนี้เป็นเอกสารที่สงวน **รูปที่ 6-20 แสดงรายวิชาที่เปิดสอนตามหลักสูตร** อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-21 แสดงรายละเอียดรายวิชาที่เปิดสอนตามหลักสูตร

6.5.3 ระบบจะทำการตรวจสอบเงื่อนไขการลงทะเบียนต่างๆ เช่น วิชาที่จะลงทะเบียนมีวิชาบังคับก่อนหรือไม่ เคยลงทะเบียนเรียนวิชาดังกล่าวมาแล้วหรือยัง และจำนวนหน่วยกิตรวมที่สามารถลงทะเบียนได้ รวมถึงรายวิชาที่มีการจำกัดจำนวนผู้ที่ลงทะเบียน ถ้าไม่ผ่านการตรวจสอบก็ไม่สามารถลงทะเบียนวิชาดังกล่าวได้ ระบบก็จะทำการแสดงรายวิชาที่ลงทะเบียนอีกครั้ง พร้อมทั้งแสดงรายละเอียดค่าใช้จ่ายทั้งหมดให้นักศึกษาทราบ

6.5.4 ขั้นตอนสุดท้าย ระบบจะทำการเก็บข้อมูลการลงทะเบียนนี้ไว้เพื่อนำไปทดสอบกับการลงทะเบียนจริง และแสดงข้อความเพื่อบอกช่วงเวลาที่นักศึกษาต้องมาลงทะเบียนจริงอีกครั้ง



เอกสารนี้เป็นเอกสารที่รูปที่ 6-22 แสดงระบบตรวจสอบวิชาที่ต้องเรียนวิชาบังคับก่อน
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบสารสนเทศศึกษาระดับบัณฑิตศึกษา - ระบบลงทะเบียนล่วงหน้า - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://161.246.5.86/jsp/PreRegistrationPage2.jsp

King Mongkut's Institute of Technology Ladkrabang

ระบบลงทะเบียนล่วงหน้า

ชื่อ - นามสกุล ชวกริช อุทัย รหัสนักศึกษา 43015354 หลักสูตร วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา วิทยาการสารสนเทศ สาขาวิชา เทคโนโลยีสารสนเทศ สาขาวิชา เทคโนโลยีสารสนเทศ คณะ เทคโนโลยีสารสนเทศ

รายวิชาทั้งหมดที่ลงทะเบียน

รหัสวิชา	ชื่อวิชา	หน่วยกิต	ตอน	ประเภท
07017101	Information Systems Concepts	3	1	Credit
07017102	Information Systems Development	3	1	Credit
รวม		6		หน่วยกิต

ต่อไป กลับไปแก้ไข

Done Internet

รูปที่ 6-23 แสดงรายวิชาที่ลงทะเบียน

ระบบสารสนเทศศึกษาระดับบัณฑิตศึกษา - ระบบลงทะเบียนล่วงหน้า - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://161.246.5.86/jsp/PreRegistrationPage3.jsp

King Mongkut's Institute of Technology Ladkrabang

ระบบลงทะเบียนล่วงหน้า

ชื่อ - นามสกุล ชวกริช อุทัย รหัสนักศึกษา 43015354 หลักสูตร วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา วิทยาการสารสนเทศ สาขาวิชา เทคโนโลยีสารสนเทศ สาขาวิชา เทคโนโลยีสารสนเทศ คณะ เทคโนโลยีสารสนเทศ

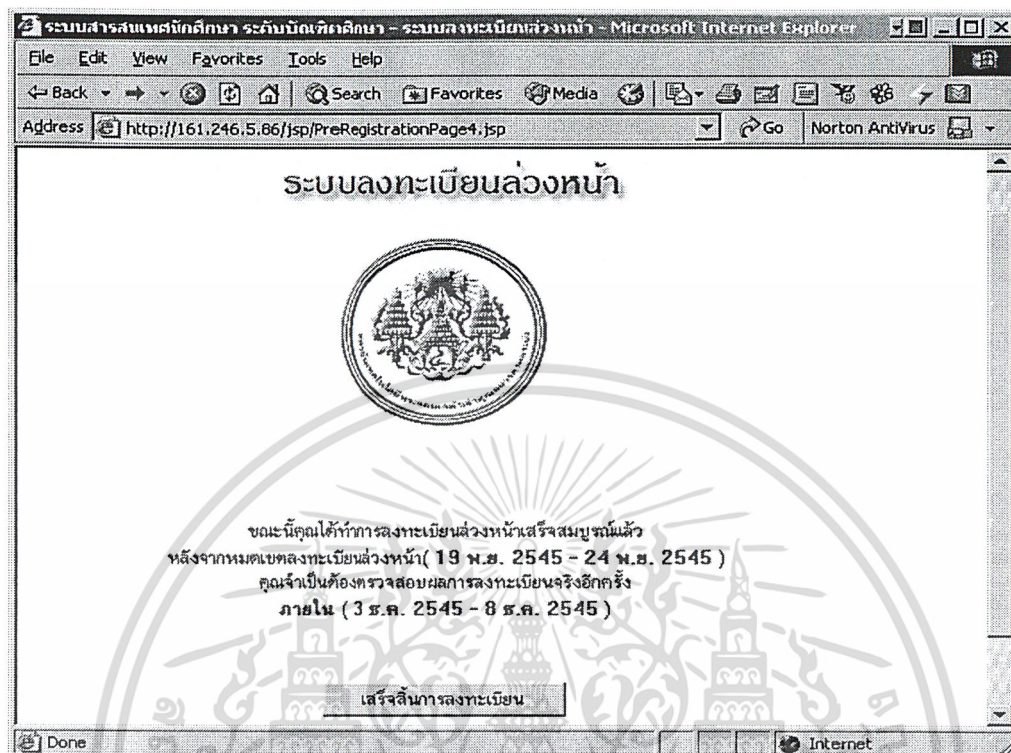
รายละเอียดค่าใช้จ่าย ภาคเรียนที่ 2 ปีการศึกษา 2545

ประเภทยอด	จำนวนเงิน (บาท)
เงินค่าหน่วยกิต	25000.00
เงินบำรุงการศึกษา	7000.00
ค่าลงทะเบียนรายวิชา 5 วิชา	1800.00
ค่าบำรุงหอพัก	500.00
ค่าบำรุงหอสมุด	200.00
ค่าบำรุงรถเข็น	50.00
รวมเงินที่ต้องชำระ	34550.00

โอนเงินผ่านบัญชีเงินฝากธนาคารของเคทีไอ พงษ์ กลับไปแก้ไข

Done Internet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูป 6-24 แสดงค่าใช้จ่ายตามหลักสูตร
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6-25 แสดงข้อความการลงทะเบียนสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

บทสรุปและบทวิจารณ์

7.1 บทสรุป

การพัฒนากระบวนการสอนเทคนิคศึกษา ระดับบัณฑิตศึกษาด้วยวิธีเชิงวัตถุนับว่ามีได้รับความสำเร็จอย่างยิ่งในแง่ของการวิเคราะห์และออกแบบ เพราะในระบบนี้การวิเคราะห์และออกแบบได้ทำอย่างรอบคอบที่สุด จึงทำให้ได้ระบบที่อำนวยความสะดวกและครอบคลุมประโยชน์การใช้สอยมากในระดับที่น่าพอใจ ส่วนในเรื่องของการสร้างหรือการเขียนโปรแกรมนั้นก็สามารถดำเนินการได้ครอบคลุมในส่วนระบบการลงทะเบียนได้เกือบทั้งหมด ทั้งนี้เรายังได้นำโปรแกรมตัวอย่าง(Prototype) ไปให้เจ้าหน้าที่บัณฑิตวิทยาลัยลองทำการทดสอบระบบ ก็ได้รับความพอใจจากทางเจ้าหน้าที่มากในระดับที่น่าพอใจ

ส่วนเครื่องมือที่เราใช้ในการทำงานนั้นคือ Case tool ต่าง ๆ ซึ่งได้แก่

- โปรแกรม Rational Rose 2002 เป็น CASE Tool ที่สามารถทำให้เราสร้างโมเดลของระบบโดยใช้ UML ได้อย่างสะดวก
- โปรแกรม Forte for java เป็นเครื่องมือที่ใช้ในการเขียนภาษาจาวาได้เป็นอย่างดีด้วยความสามารถที่มีให้มาไม่ว่าจะเป็น การเขียนโปรแกรมแบบแอปพลิเคชัน หรือใช้ Servlet ,JSP ก็ตาม
- โปรแกรมฐานข้อมูล DB2 จัดเป็น DBMS ที่มีความสามารถในด้านการสร้างและจัดการฐานข้อมูลได้ดี

ส่วนในเรื่องของการศึกษาและทำความเข้าใจระบบทะเบียนนักศึกษา ระดับบัณฑิตศึกษานั้น ได้ทำการวิเคราะห์หาจุดอ่อนต่าง ๆ ซึ่งได้กล่าวมาแล้วข้างต้น รวมทั้งยังสามารถนำไปสร้างเป็นโมเดลของระบบที่พยายามสร้างความสะดวกให้มากที่สุดอีกด้วย

ส่วนในการเขียนโปรแกรมนั้นได้มีการทำงานไปประมาณ 80 เปอร์เซ็นต์ ซึ่งจะขอชี้แจงรายละเอียดดังนี้

- ส่วนของแอปพลิเคชันได้ทำการพัฒนาในส่วนของการจัดการข้อมูลหลักสูตร นักศึกษา หน่วยงานการศึกษา ระยะเวลาการลงทะเบียน การจัดการรายวิชา จัดการค่าใช้จ่ายต่างๆ ลงทะเบียนเรียน เพิ่ม ถอน เปลี่ยนรายวิชา ป้อนเกรดในแต่ละรายวิชาและจัดการใบทรานสคริป
- ในส่วนของเว็บแอปพลิเคชันได้ทำการพัฒนาในส่วนของการลงทะเบียนล่วงหน้า ลงทะเบียนปกติ ตรวจสอบรายละเอียดรายวิชา ตรวจสอบข้อมูลการลงทะเบียน ตรวจสอบเกรด และยื่นแบบฟอร์มคำร้อง
- ระบบที่พัฒนาขึ้นสามารถทำการตรวจสอบเงื่อนไขได้ต่างๆ เช่น

- ตรวจสอบรายวิชาที่นักศึกษาได้ลงทะเบียนไปแล้ว จึงทำให้ไม่เกิดปัญหาในการลงทะเบียนซ้ำซ้อนในรายวิชาเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูในวงวนเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ตรวจสอบจำนวนผู้ลงทะเบียน เนื่องจากมีบางคณะจำกัดจำนวนผู้ลงทะเบียนแต่ละวิชา โดยถ้ามีผู้ลงทะเบียนครบแล้ว ผู้ที่ลงทะเบียนทีหลังจะไม่สามารถลงทะเบียนได้
- ตรวจสอบรายวิชาที่จำเป็นต้องเรียนมาก่อน (Prerequisite)
- ตรวจสอบรายวิชาที่ถูกปิดไป โดยถ้ารายวิชาถูกปิดไป จะไม่สามารถทำการลงทะเบียนรายวิชาดังกล่าวได้
- ตรวจสอบจำนวนรวมหน่วยกิต สูงสุด , ต่ำสุด ที่นักศึกษาสามารถทำการลงทะเบียนได้
- สามารถตรวจสอบ ว่านักศึกษาถูกทำทัณฑ์บน หรือ พ้นสภาพ หรือไม่

จากสิ่งที่ได้กล่าวมาจะเห็นได้ว่าการศึกษาวิจัยนี้ได้ทำครอบคลุมตามวัตถุประสงค์ทั้ง 4 ข้อที่ได้ตั้งไว้ข้างต้นแล้ว

7.2 บทวิจารณ์

หลังจากที่ได้ลงมือทำโครงการวิจัยนี้ สิ่งที่ผู้วิจัยได้ทำการศึกษาและค้นคว้า รวมถึงทำการวิเคราะห์ทำให้สามารถทราบได้ถึงข้อเท็จจริงต่าง ๆ ที่ส่งผลให้การทำงาน ซึ่งถ้าไม่ลงมือเข้าไปวิเคราะห์และศึกษาอย่างจริงจังอาจเป็นการยากที่จะทราบถึงข้อเท็จจริงเหล่านั้นได้ และอาจจะมีข้อจำกัดบางประการที่จะทำให้ความสามารถของระบบไม่สามารถใช้ได้ เช่นถ้าอาจารย์ทำการส่งเกรดล่าช้าก็จะทำให้ไม่สามารถทำการลงทะเบียนล่วงหน้าได้ตามเวลาที่วางไว้ ผู้วิจัยจึงจะขอกล่าวถึงข้อเท็จจริงเหล่านั้นเพื่อให้ผู้อ่านสามารถใช้ประโยชน์ในปริญญาณิพนธ์ และระบบที่พัฒนาขึ้นนี้ได้อย่างเต็มที่

7.3 ปัญหาที่พบในการดำเนินงาน

7.3.1 ด้วยกฎระเบียบข้อบังคับต่าง ๆ ที่มีมากมายทำให้เกิดความยุ่งยากและสับสนอีกทั้งยังมีความแตกต่างกันในแต่ละคณะซึ่งทำให้ยังเป็นการเพิ่มรายละเอียดในการวิเคราะห์เป็นอย่างยิ่ง ยกตัวอย่างเรื่องของ การที่อาจารย์ที่ปรึกษาต้องทำการตรวจสอบการลงทะเบียนของนักศึกษาซึ่งทำให้นักศึกษาต้องเข้าพบอาจารย์ที่ปรึกษา ทั้ง ๆ ในความเป็นจริงอาจารย์ที่ปรึกษานั้นไม่ได้ตรวจสอบอย่างจริงจัง

7.3.2 ผู้ปฏิบัติงานไม่ทราบขั้นตอนการปฏิบัติงานอย่างชัดเจน ทำให้ต้องมีการปรับแก้เงื่อนไขข้อมูลที่ได้มาบางอย่าง นั่นหมายถึงการปรับแก้เรื่องของความต้องการของผู้ใช้ ซึ่งส่งผลต่อระบบที่ได้วิเคราะห์ไว้แล้ว ต้องทำการวิเคราะห์เปลี่ยนแปลงอีกครั้ง

7.3.3 เครื่องมือที่ใช้ ในการเขียนโปรแกรมไม่ว่าจะเป็น Rational Rose ,Forte for Java หรือ IBM DB2 ล้วนต้องการทรัพยากรของเครื่องคอมพิวเตอร์ที่ใช้สูงทำให้การทำงานเป็นไปอย่างเชื่องช้า

7.3.4 เซิร์ฟเวอร์ที่จะใช้งานจริงไม่มีเสถียรภาพเพียงพอ เนื่องด้วยปัญหาทางด้านเครือข่าย และตัวเซิร์ฟเวอร์เองจึงทำให้การทำงานไม่สามารถทดลองได้จริง

7.4 ข้อเสนอแนะสำหรับระบบทะเบียนนักศึกษา ของบัณฑิตวิทยาลัย

7.4.1 งานในระบบทะเบียนนักศึกษาของบัณฑิตวิทยาลัย นั้นมีความซับซ้อนและมีรายละเอียดเฉพาะ จึงเป็นการถูกต้องแล้วที่แยกออกมาทำงานเฉพาะจากงานทะเบียนของนักศึกษาระดับปริญญาตรี แต่สิ่งที่ควรที่จะต้องทำการปรับปรุงนั้นคือ ควรมีการกระจายงานของฝ่ายทะเบียนของบัณฑิตวิทยาลัยส่วนกลางไปยังส่วนคณะ อย่างเช่นการกรอกตารางเรียนตารางสอน การกรอกเกรด การเปิดปิดรายวิชาซึ่งจะทำให้เกิดความรวดเร็ว

และคล่องตัวมากกว่า ทั้งยังสามารถจัดการกับข้อผิดพลาดได้โดยตรง เพราะในปัจจุบันมีข้อผิดพลาดบางอย่างเกิดขึ้น อย่างเช่นตารางเรียนผิด ไม่สามารถหาผู้รับผิดชอบได้ เพราะ มีการส่งต่องานจากส่วนคณะเข้ามาในส่วนกลางด้วย

7.4.2 ในเรื่องของระบบสารสนเทศนั้น โดยปกติแล้วสำนักวิจัยและบริการคอมพิวเตอร์จะเป็นผู้ให้การสนับสนุนแก่บัณฑิตวิทยาลัย แต่ด้วยความแตกต่างของระบบปริญญาตรี และระดับบัณฑิตศึกษาทำให้เกิดปัญหาหลายอย่างขึ้น ในส่วนของบัณฑิตวิทยาลัย ซึ่งการแก้ปัญหาจากสำนักวิจัยและบริการคอมทำได้ล่าช้า ทางบัณฑิตวิทยาลัยจึงคิดที่จะสร้างระบบสารสนเทศของตัวเองขึ้นมา ซึ่งในมุมมองของผู้ทำการวิจัยนี้เห็นว่าการระบบสารสนเทศนั้นทั้งในด้านของโปรแกรม และฐานข้อมูลยังควรให้สำนักวิจัยรับผิดชอบ แต่ทางสำนักวิจัยควรจัดกำลังคน หรือใช้กำลังคนของทางบัณฑิตวิทยาลัย ขึ้นมาเฉพาะเพื่อดูแลระบบในส่วนนี้ ทั้งนี้เพื่อให้สามารถใช้ทรัพยากรของสำนักวิจัยที่มีอยู่แล้วได้อย่างเต็มที่ ส่วนบุคลากรที่จัดไว้ดูแลเฉพาะนี้ก็จะสามารถจัดการกับปัญหาต่าง ๆ หรือเพิ่มแก้ไขได้ในทันที ซึ่งปัญหาเรื่องของภาษาโปรแกรมที่ใช้ไม่ได้เป็นปัญหาอยู่แล้วเพราะถึงอย่างไรก็สามารถใช้ฐานข้อมูลร่วมกันได้

7.4.3 ควรมีการประชุมหาหรือเพื่อปรับแก้ข้อบกพร่องต่าง ๆ หาข้อดีข้อเสียของเงื่อนไขแต่ละข้ออย่างชัดเจน รวมทั้งให้แต่ละคณะมีกฎระเบียบไปในแนวทางเดียวกัน หรือถ้าไม่สามารถทำให้กฎระเบียบเหมือนกันได้ก็ควรระบุถึงข้อแตกต่างนั้นอย่างชัดเจนให้เจ้าหน้าที่ผู้ปฏิบัติงานได้ทราบ นอกจากกฎระเบียบแล้วยังมีแนวทางต่าง ๆ เช่นการเขียนหลักสูตรควรเขียนให้มีความเป็นมาตรฐานมีรายละเอียดครบถ้วน อย่างเช่นทุกหลักสูตรควรให้มีการกำหนดแผนการศึกษา(Course Work) การกำหนดรหัสวิชาที่ไม่ควรซ้ำกัน ควรใช้รหัสวิชาที่เป็นตัวเลขทั้งหมด

7.4.4 นอกจากเรื่องของกฎระเบียบแล้ว การปฏิบัติตามก็ควรทำอย่างเคร่งครัดด้วย อย่างเช่นเรื่องกำหนดการส่งเกรดของอาจารย์ควรจะให้ส่งตรงตามกำหนดเวลา

7.5 ข้อเสนอแนะในการนำไปพัฒนาต่อ

7.5.1 จัดทำระบบเพิ่มเติม ในส่วนที่ได้ออกแบบไว้แล้ว แต่ยังไม่ได้มีการสร้างจริง เนื่องจากด้วยข้อจำกัดเรื่องระยะเวลาทำให้ผู้วิจัยไม่สามารถทำการสร้างโปรแกรมได้ครบทุกส่วน แต่ได้ทำการวิเคราะห์และออกแบบไว้ทั้งหมดแล้ว เช่น ในส่วนของจัดการตารางเรียนและตารางสอบซึ่งตอนนี้ในการลงทะเบียนยังไม่มี การตรวจสอบเวลาสอบของแต่ละรายวิชา ซึ่งผู้ที่ทำการพัฒนาต่อสามารถนำไปสร้างต่อได้

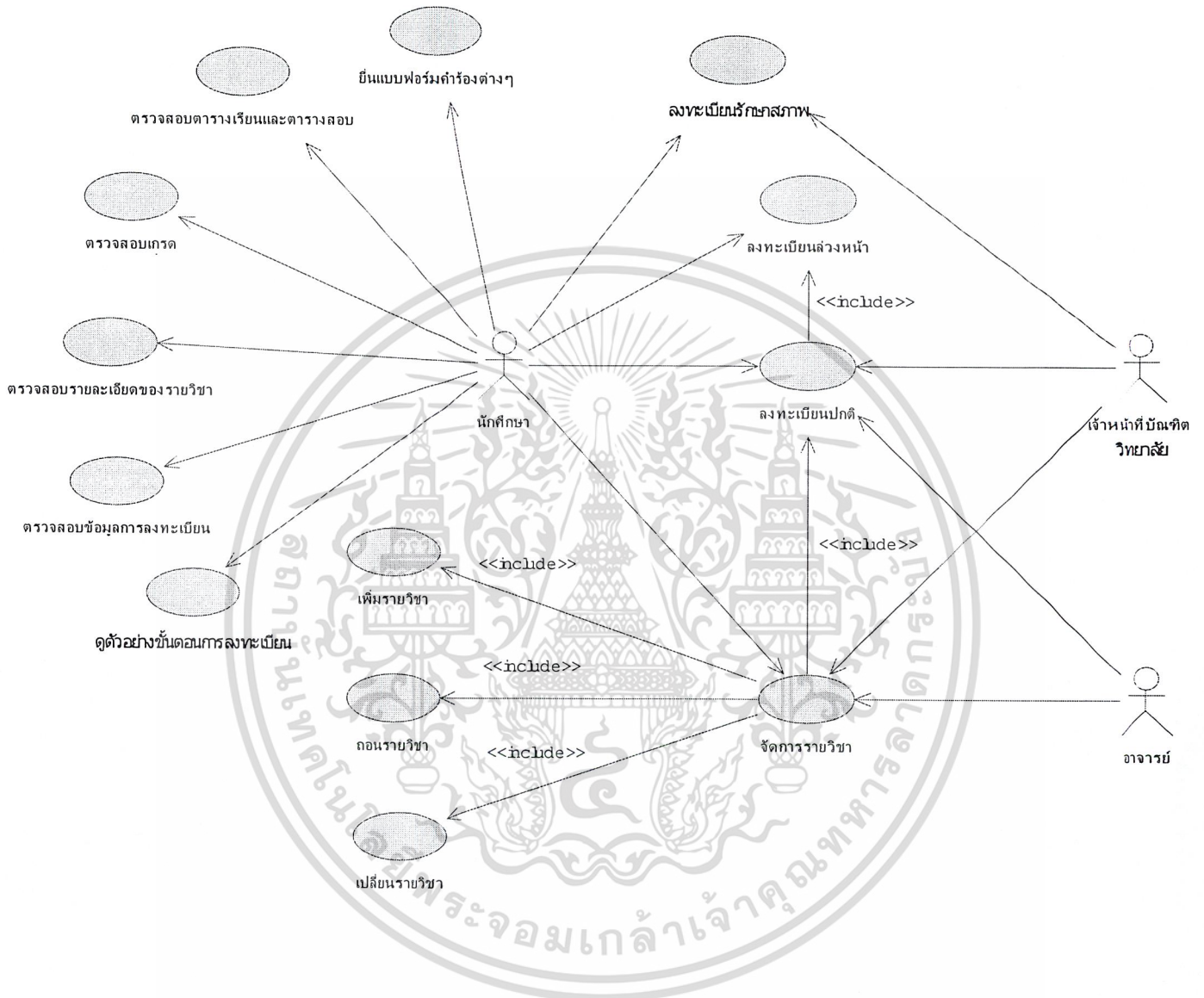
7.5.2 หากจุดบกพร่องต่างๆที่เกิดขึ้นในโปรแกรมและทำการปรับปรุงแก้ไข

7.5.3 นำเอาเทคนิคใหม่ เข้ามาใช้ เพื่อทำให้ระบบมีความสะดวกมากขึ้น อย่างเช่นในเรื่องของการกรอกเกรดอาจจะทำให้อาจารย์แต่ละท่านสามารถกรอกเกรดผ่านระบบอินเทอร์เน็ตเลยก็ได้เพื่อความสะดวก อาจพัฒนาในเรื่องการส่งข้อความ และการสื่อสารระหว่างอาจารย์และนักศึกษาให้สามารถสื่อสารข้อความกันได้ผ่านทางระบบนี้ได้เลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

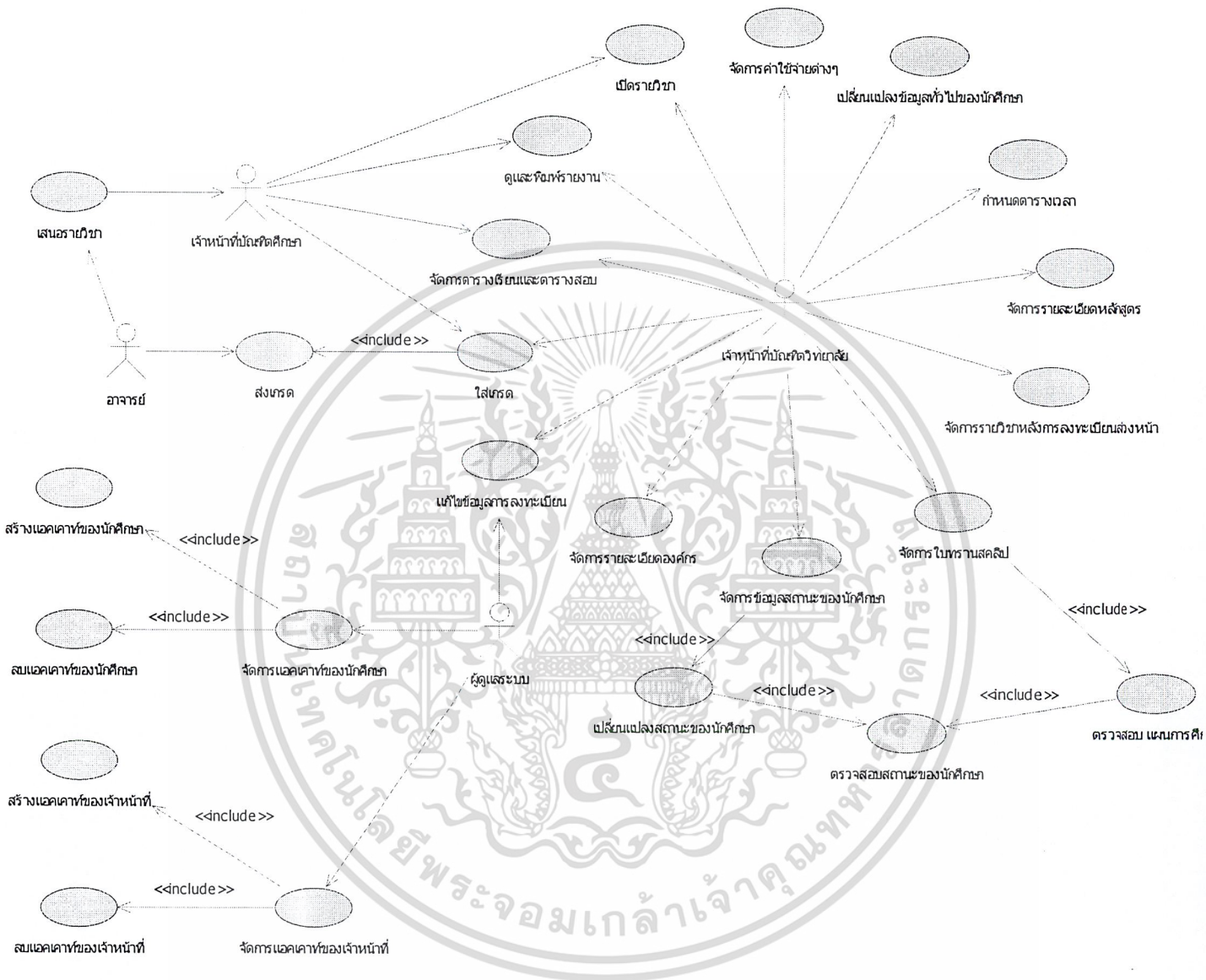


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



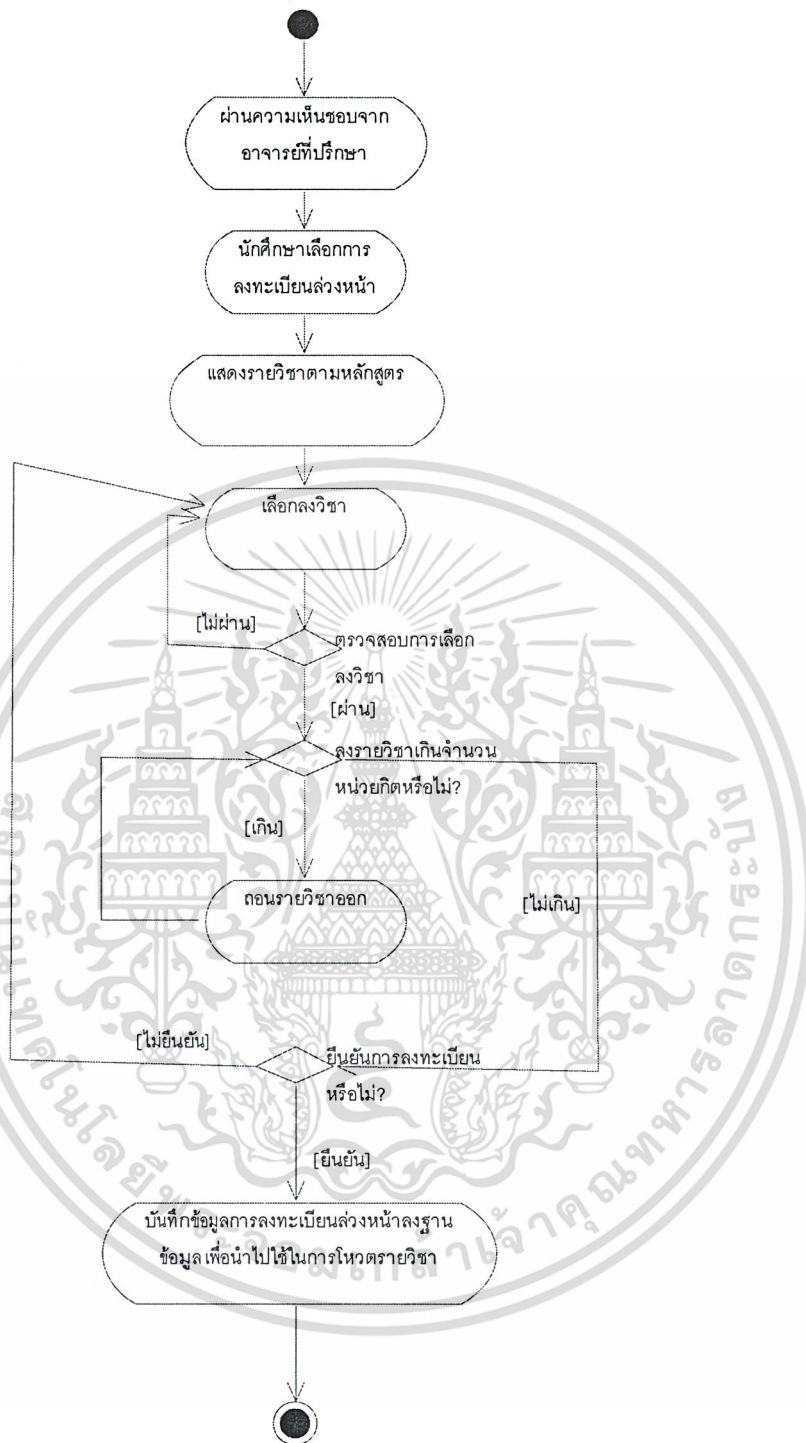
Use Case Diagram ในส่วนของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Use Case Diagram ในส่วนของเจ้าหน้าที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

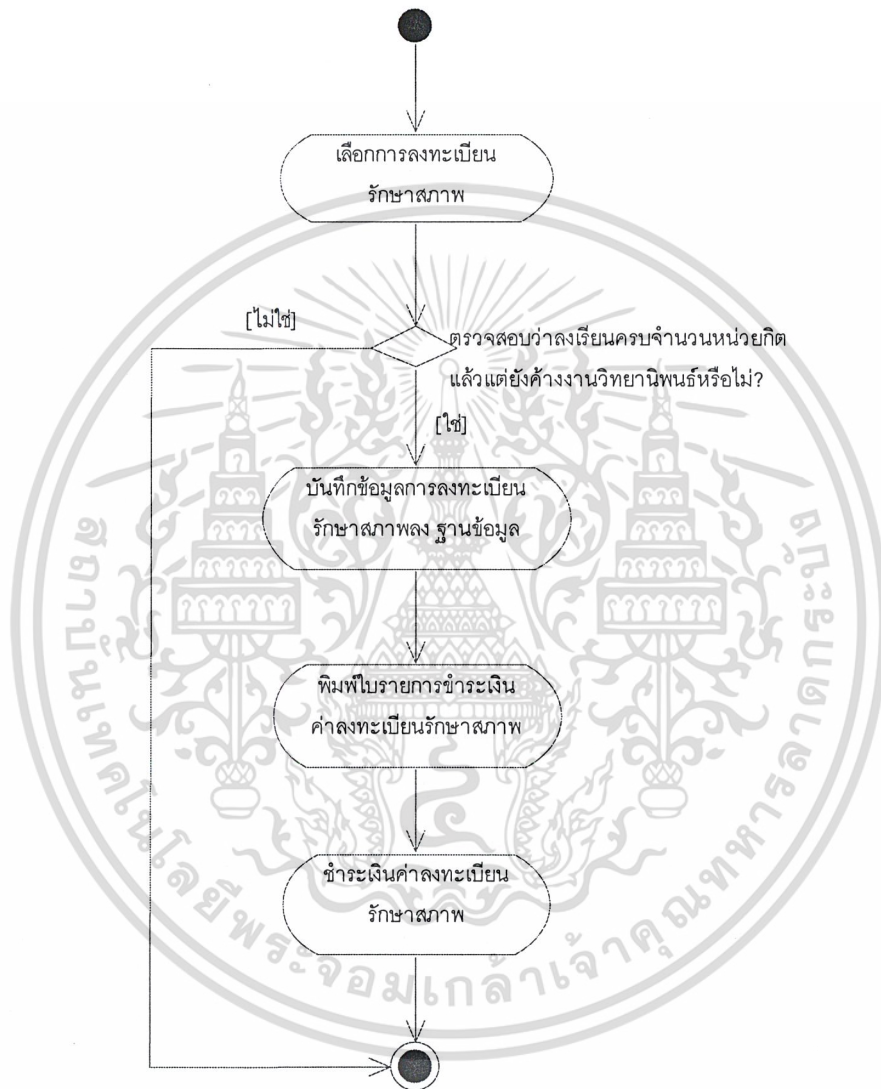


Activity Diagram ของ การลงทะเบียนล่วงหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

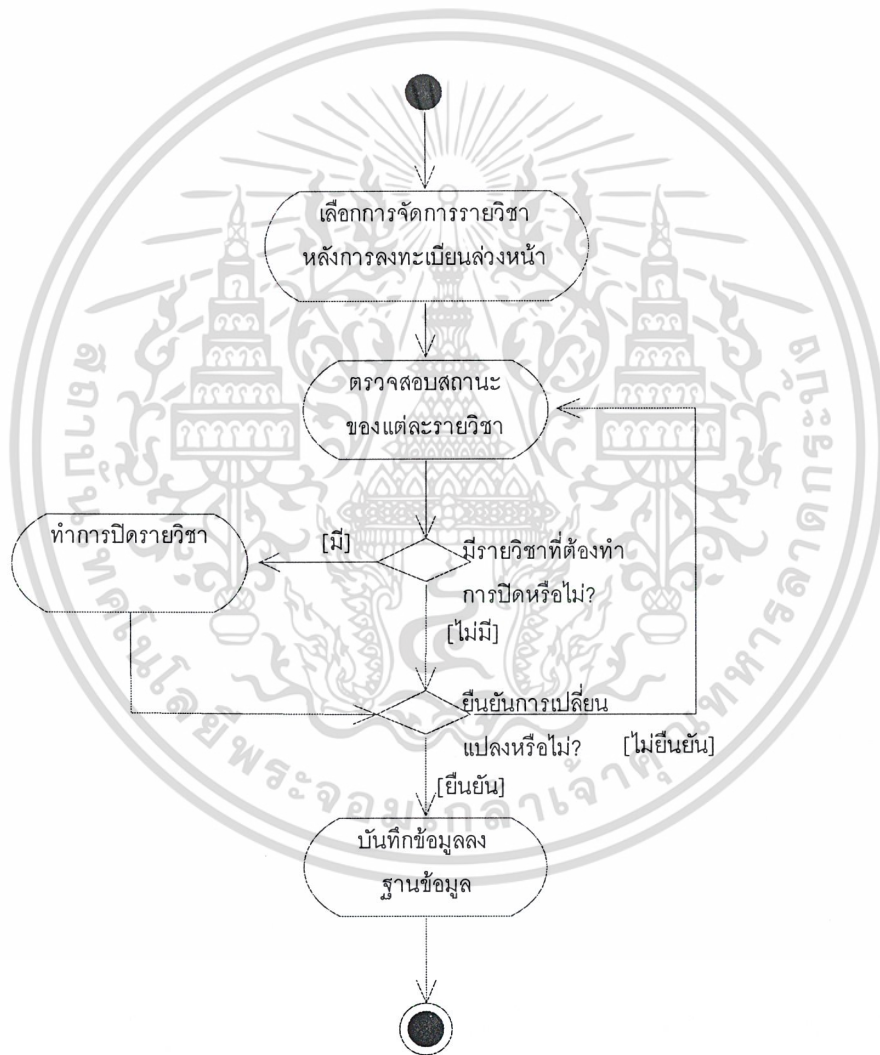


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามทำซ้ำโดยไม่ขออนุญาตจากศูนย์ทนายความประจำคณะนิติศาสตร์ มหาวิทยาลัยราชภัฏ



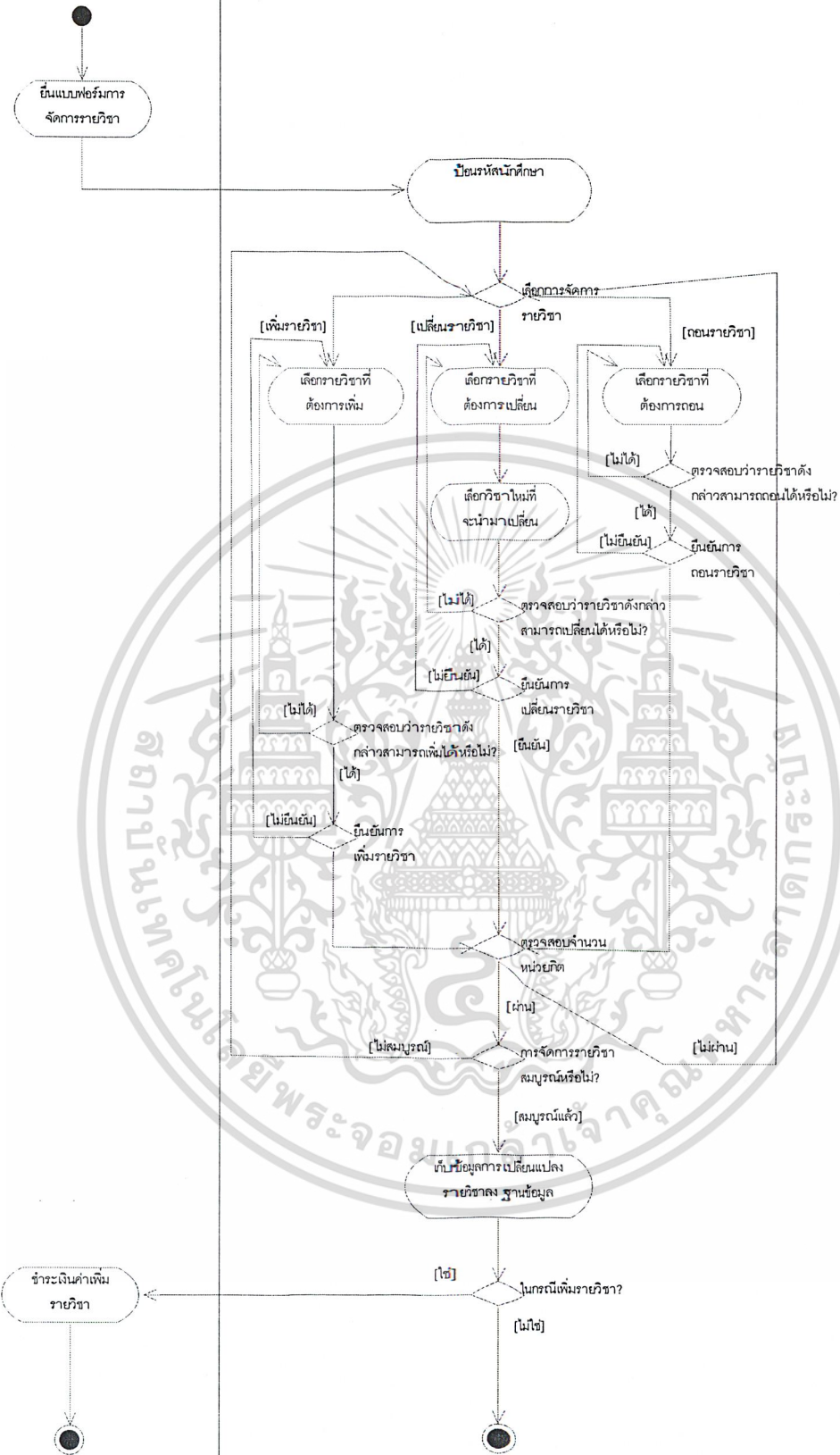
Activity Diagram ของ การลงทะเบียนรักษาสภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

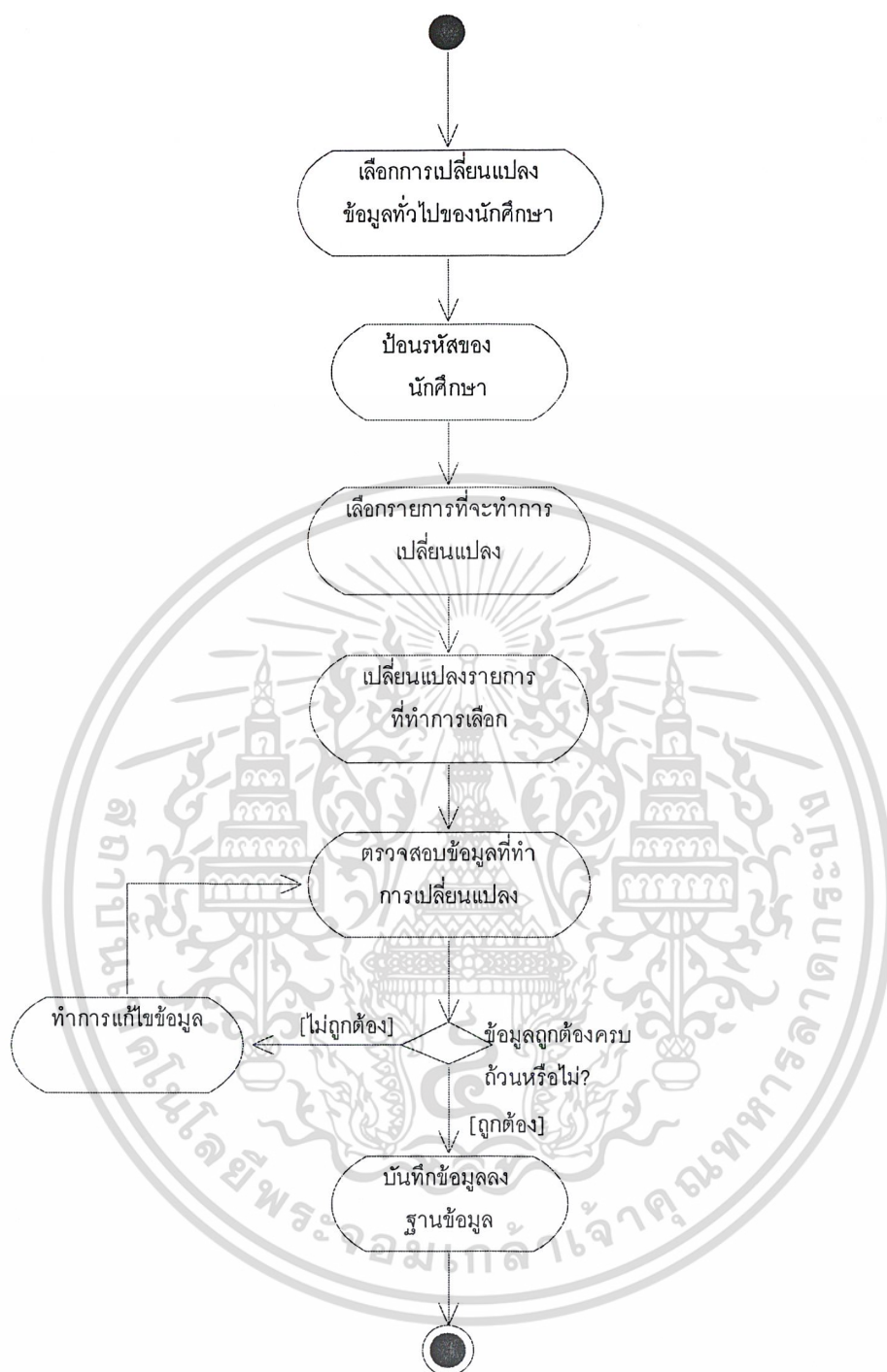


Activity Diagram ของ การจัดการรายวิชาหลังจากการลงทะเบียนล่วงหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
Activity Diagram ของ การจัดการรายวิชา
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



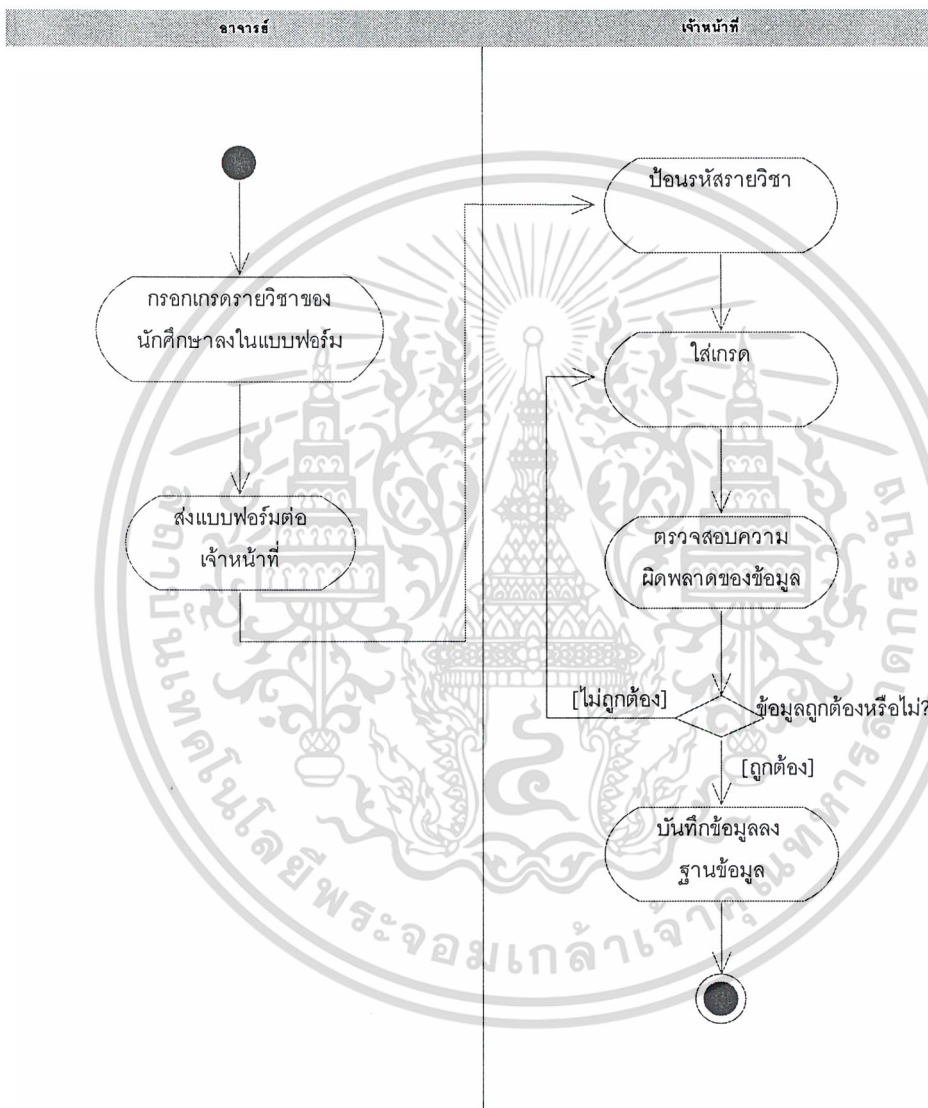
Activity Diagram ของ การเปลี่ยนแปลงข้อมูลทั่วไปของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



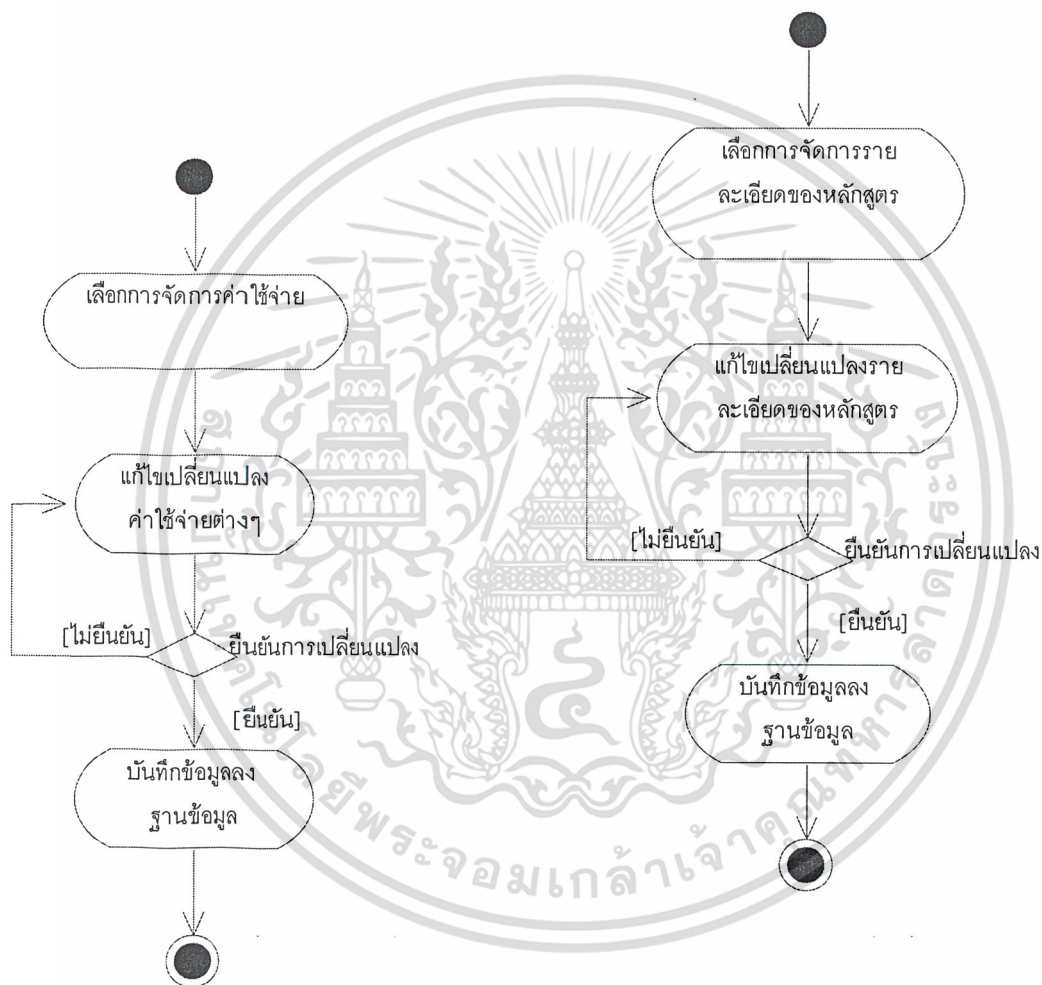
Activity Diagram ของ การแก้ไขข้อมูลการลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Activity Diagram ของ การใส่เกรด

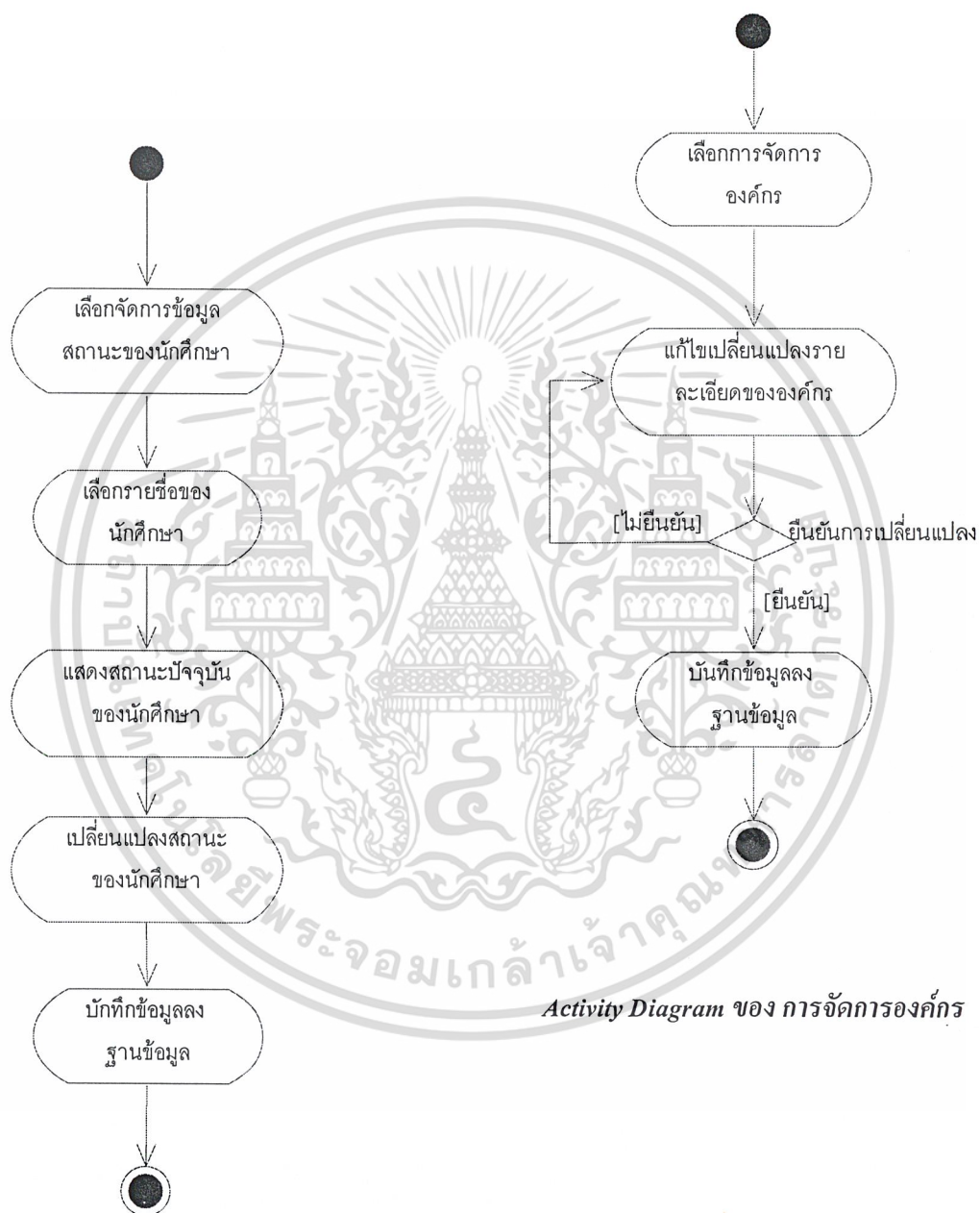
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Activity Diagram ของ การจัดการรายละเอียดของหลักสูตร

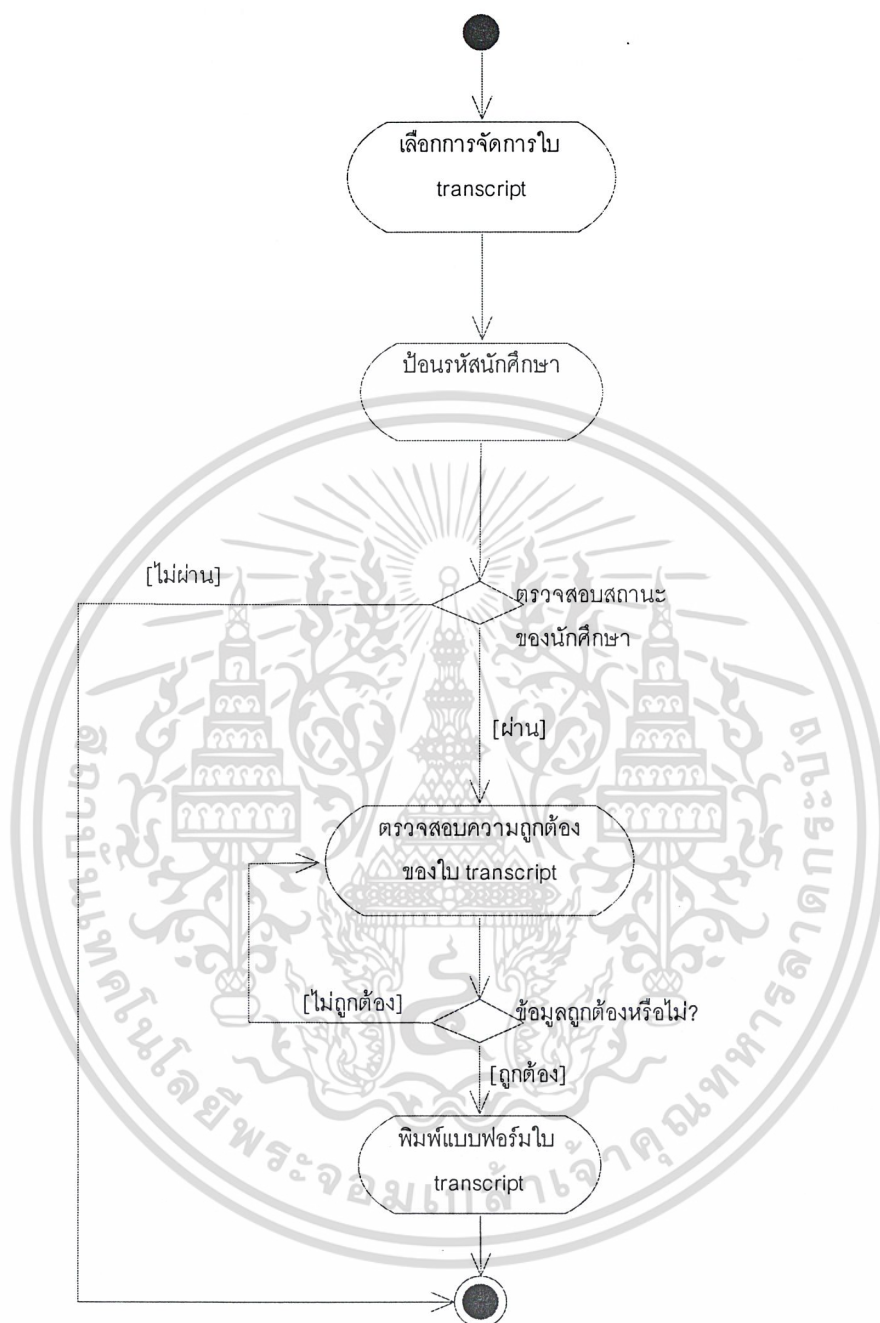
Activity Diagram ของ การจัดการค่าใช้จ่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



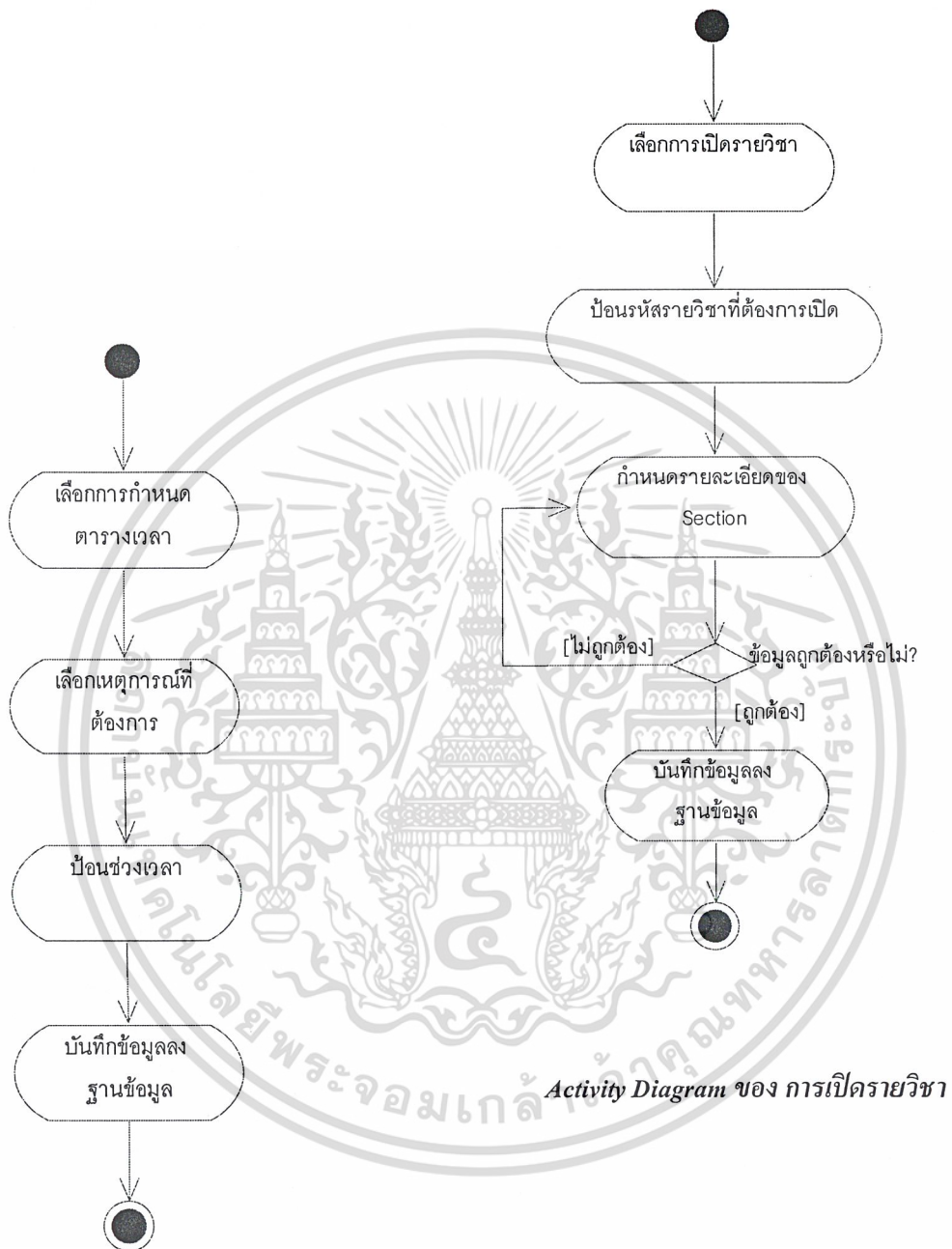
Activity Diagram ของ การจัดการข้อมูลสถานะของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



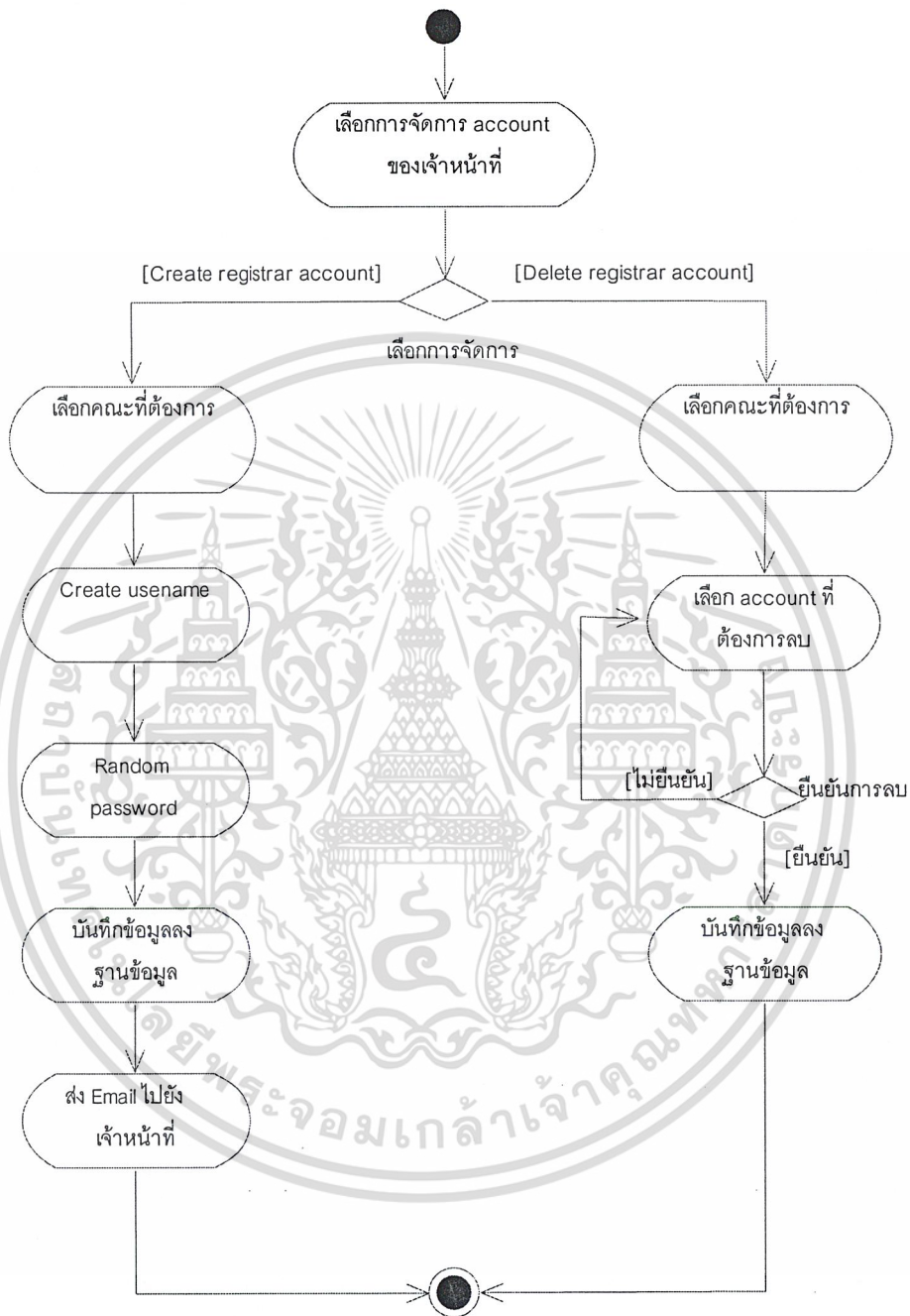
Activity Diagram ของ การจัดการใบ transcript

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



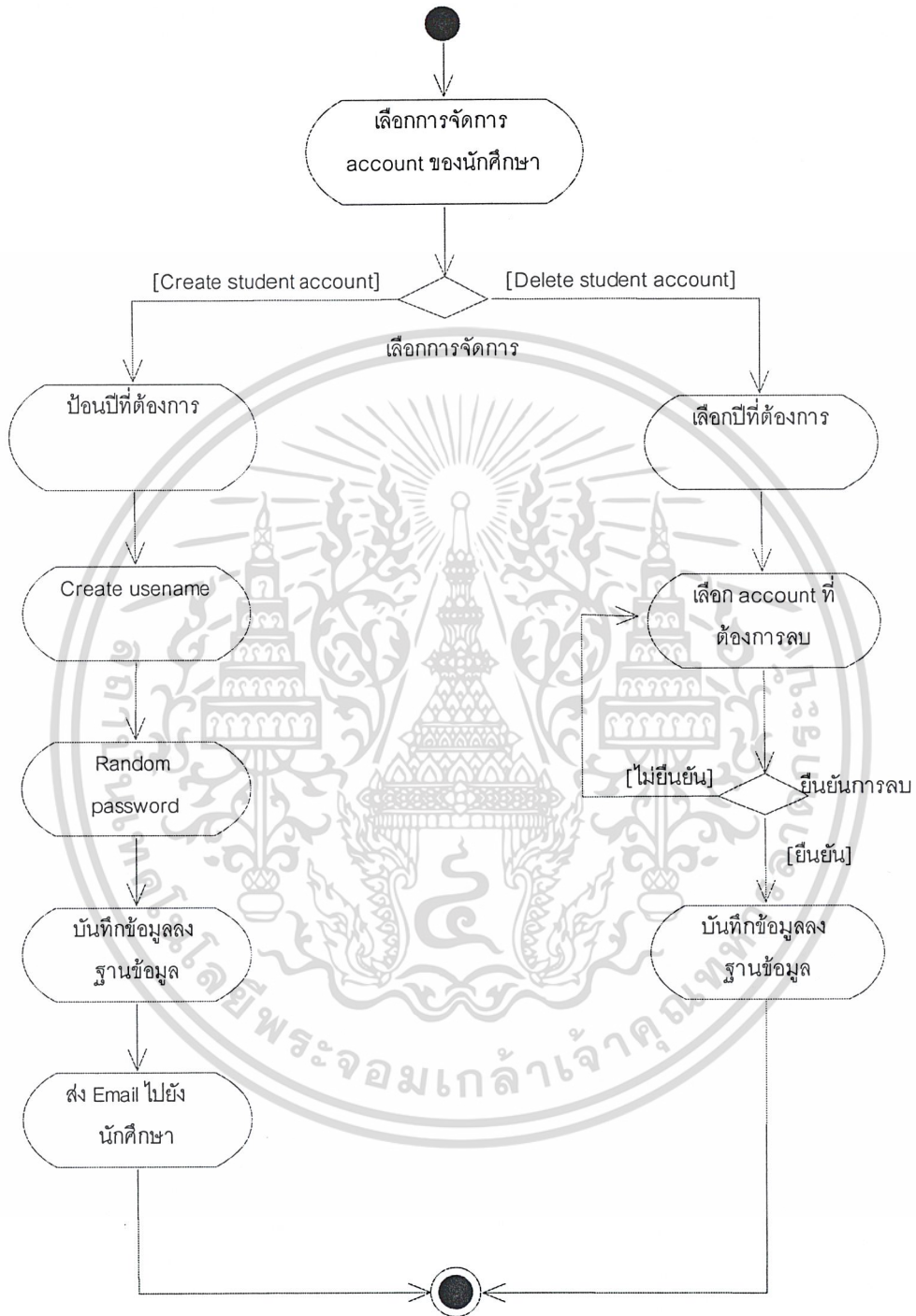
Activity Diagram ของ การกำหนดตารางเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



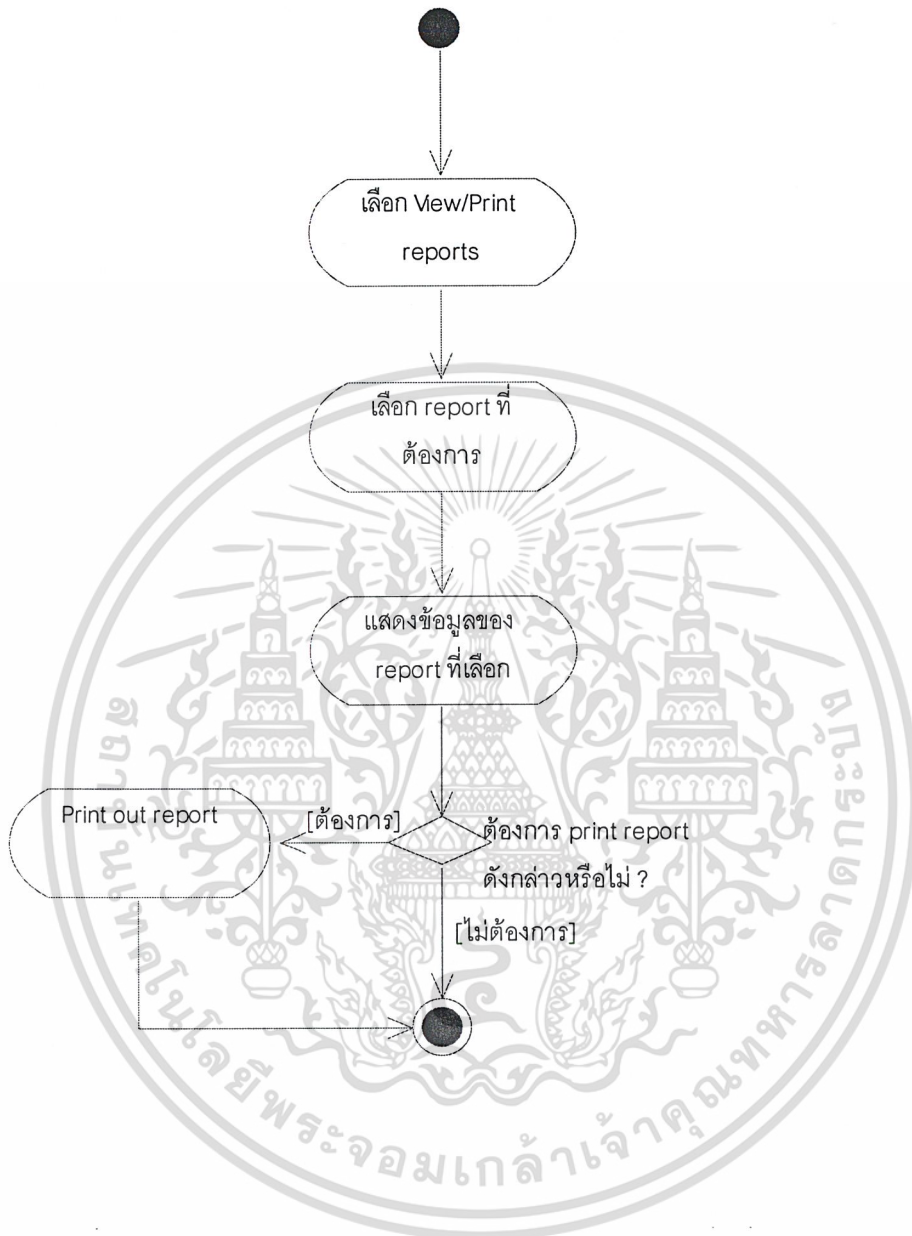
Activity Diagram ของ การจัดการ account ของเจ้าหน้าที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



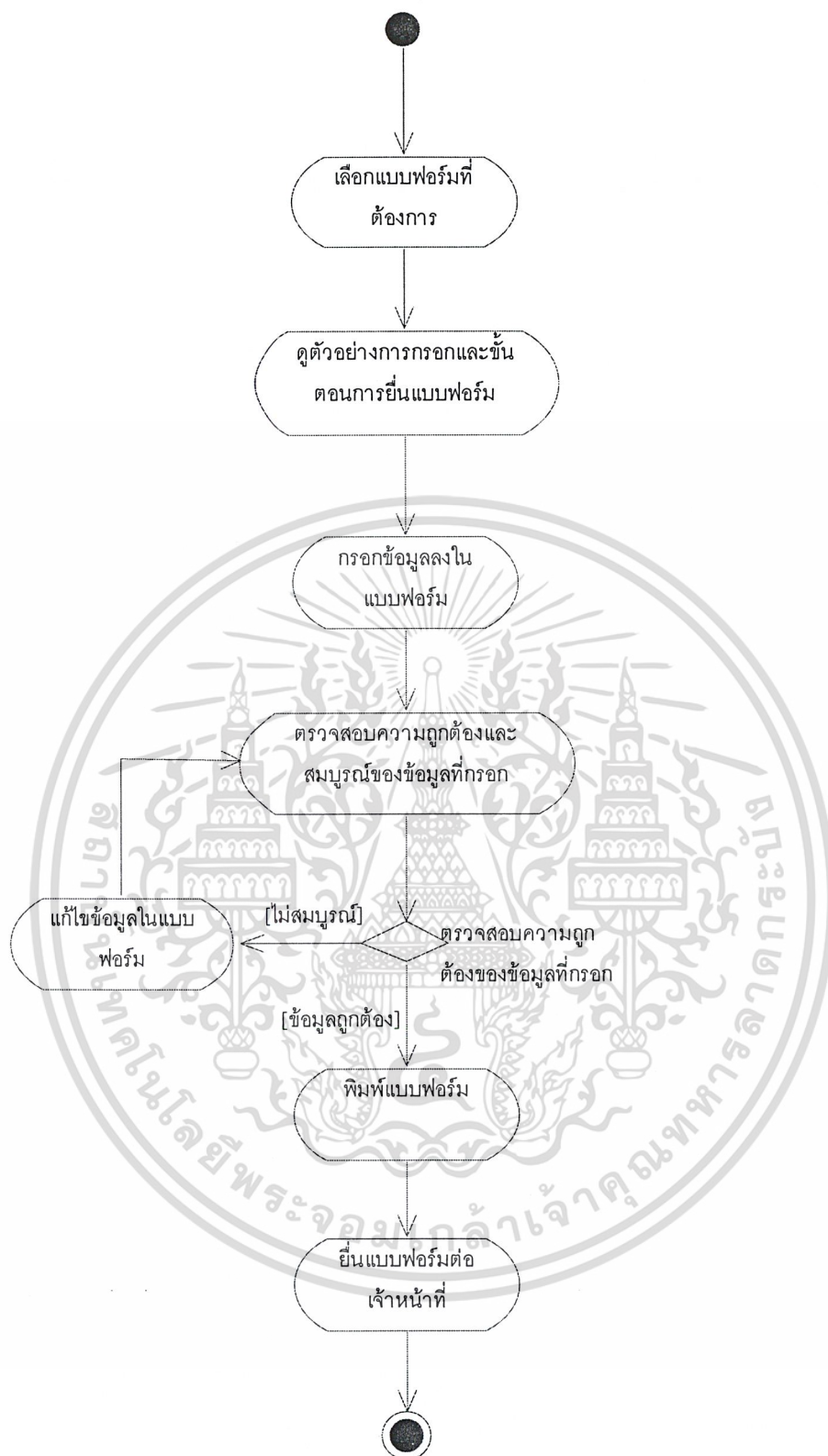
Activity Diagram ของ การจัดการ account ของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



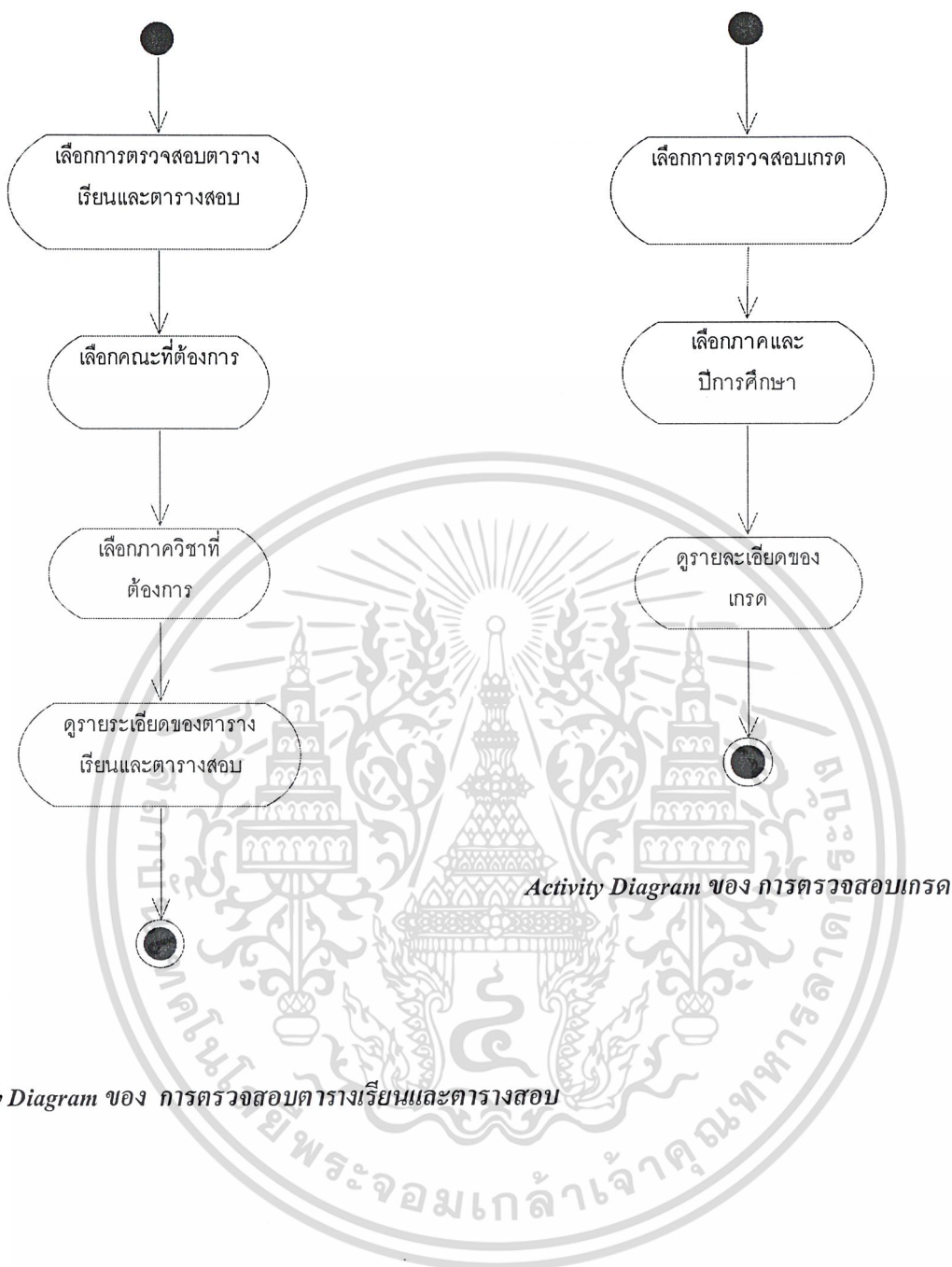
Activity Diagram ของ ดูและพิมพ์รายงานต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

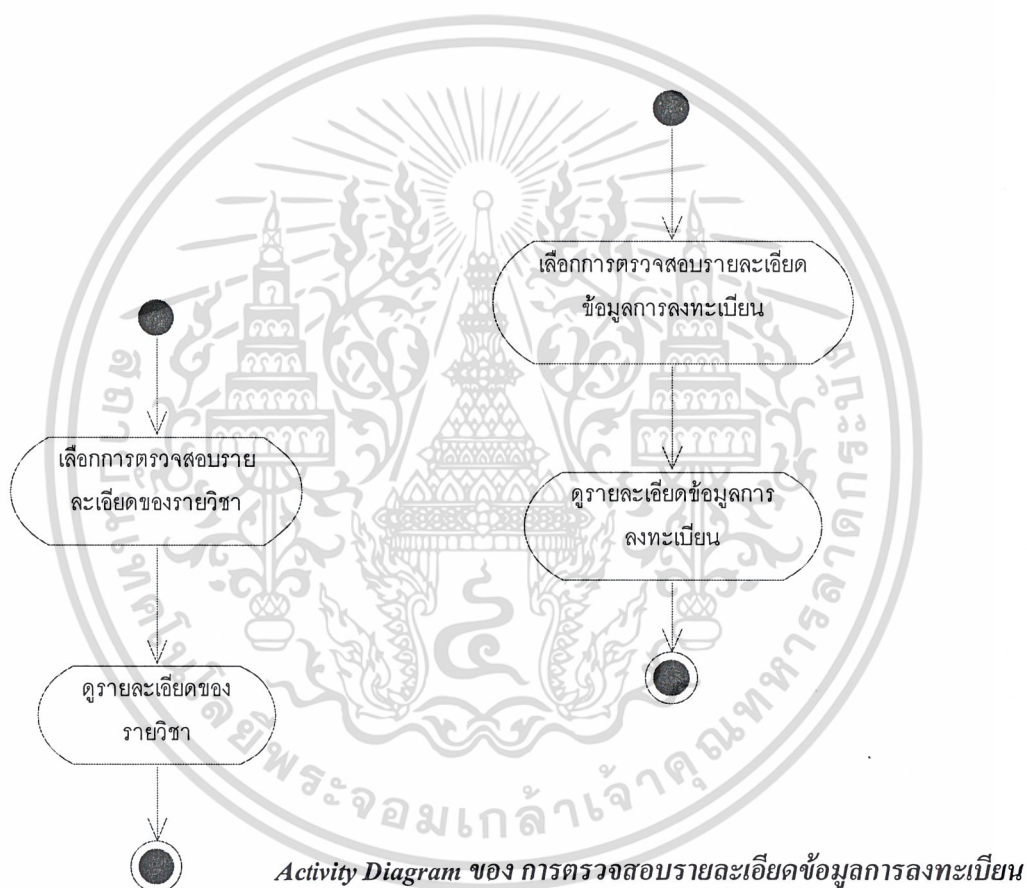


Activity Diagram ของ ยื่นแบบฟอร์มคำร้องต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

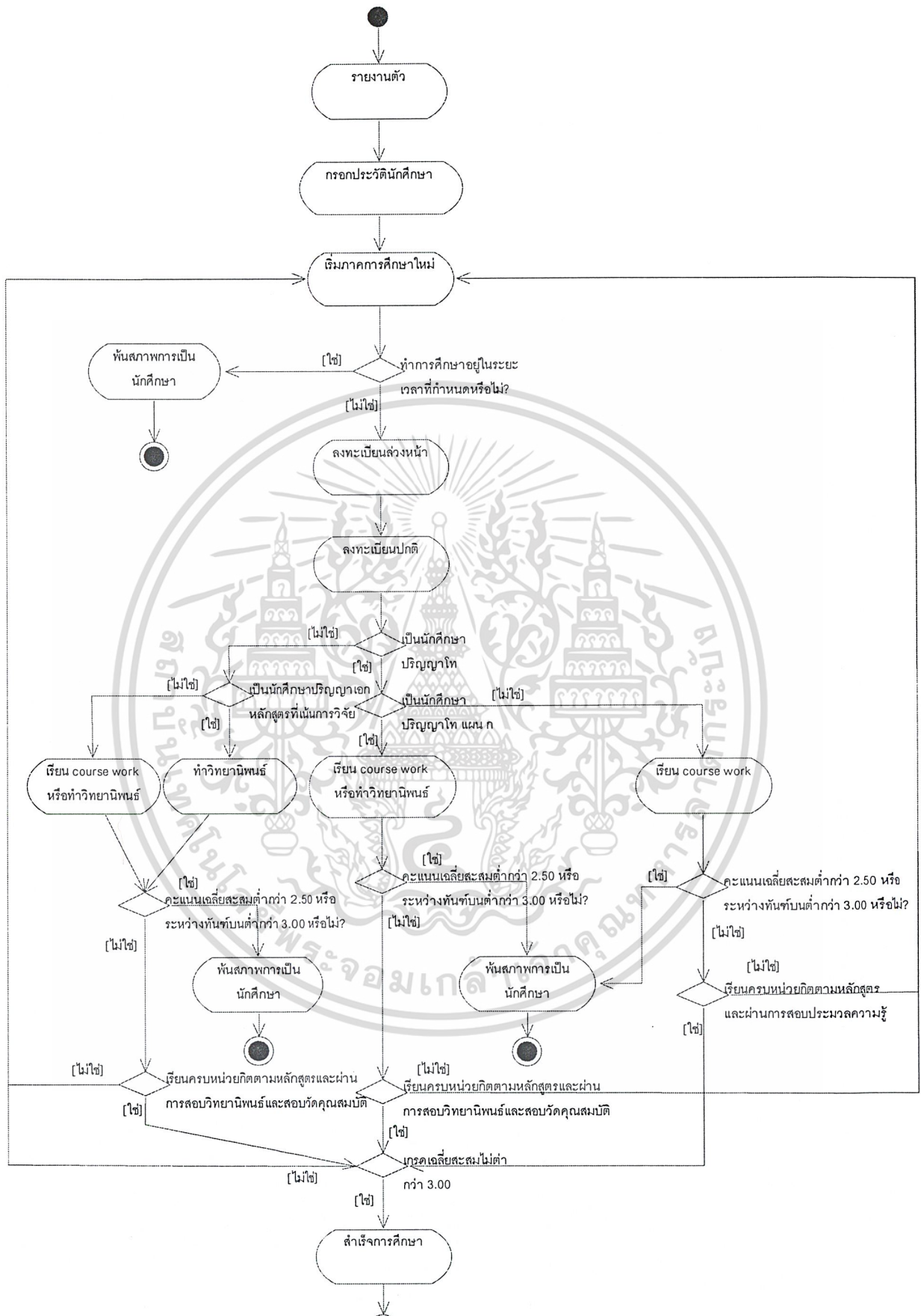


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

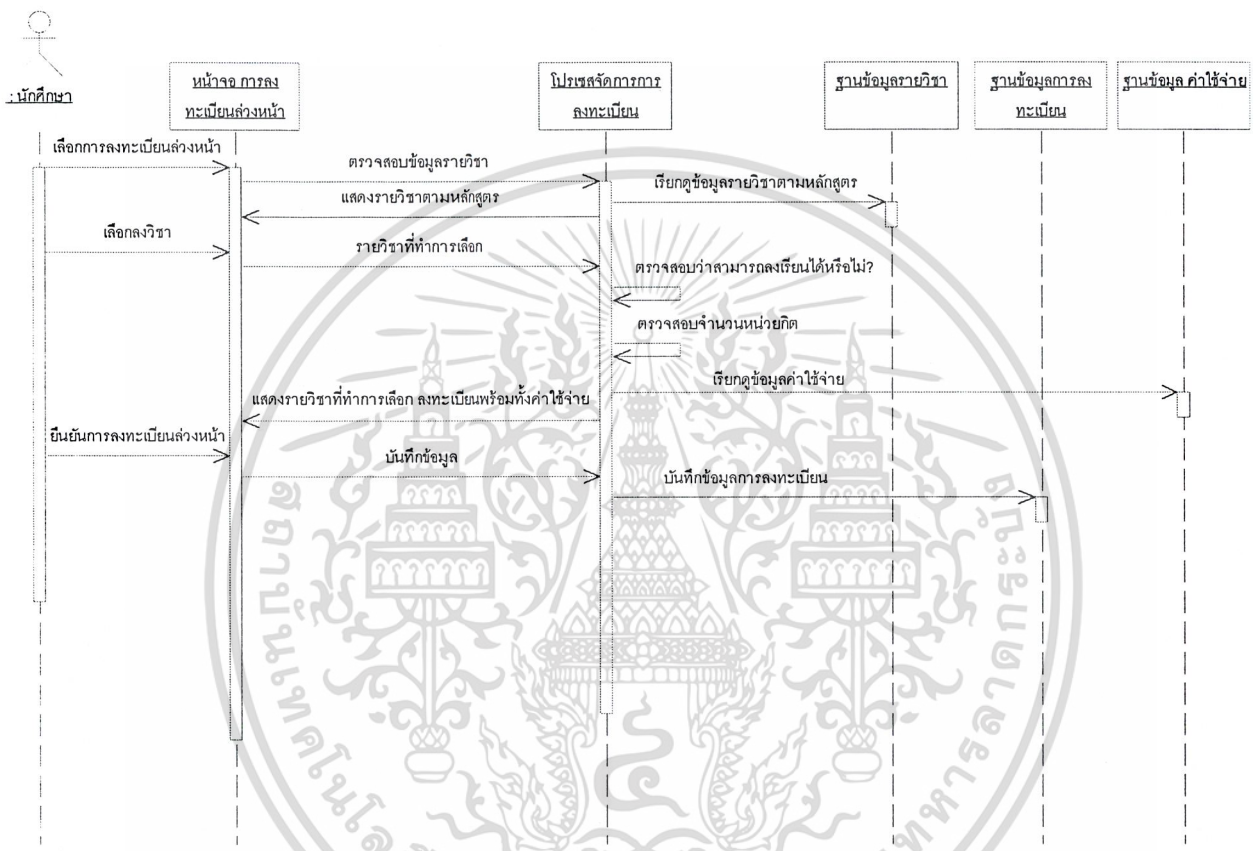


Activity Diagram ของ การตรวจสอบรายละเอียดของรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

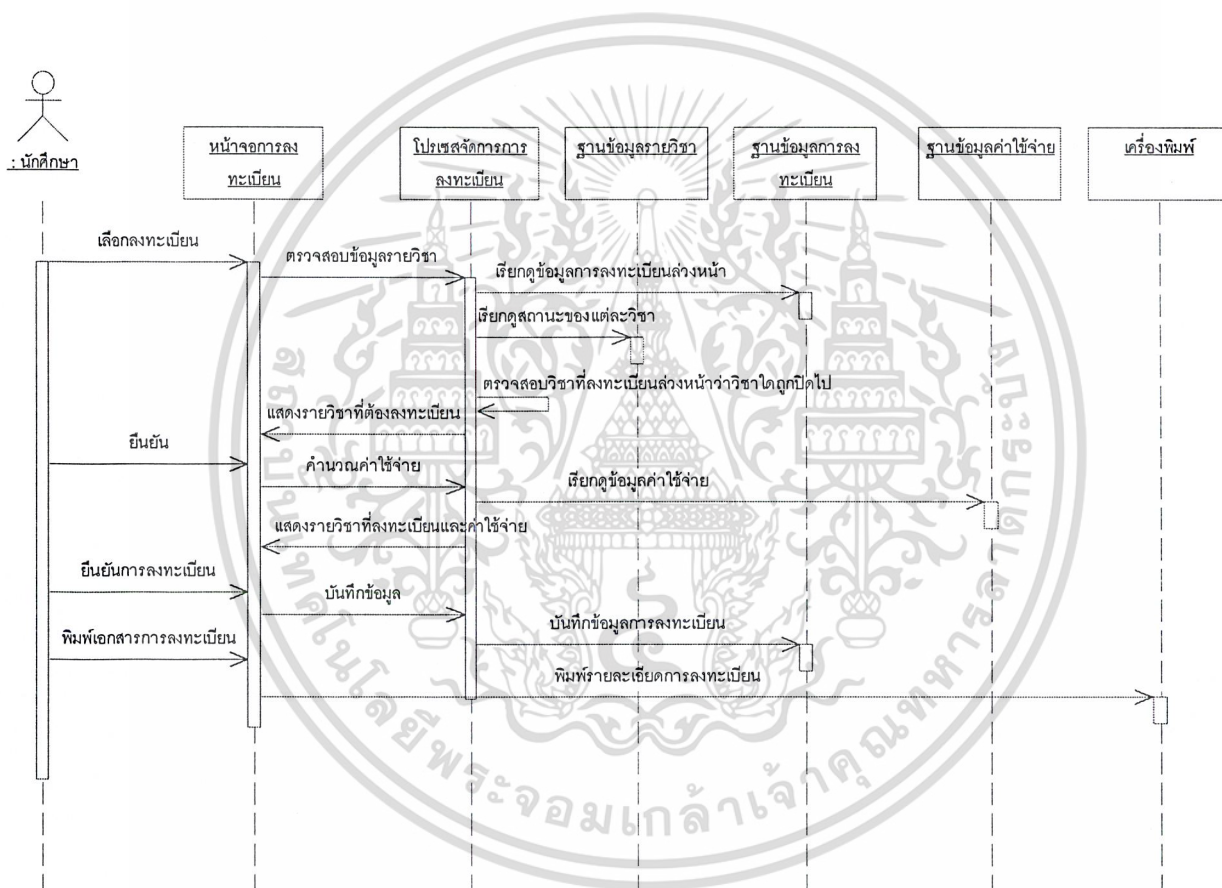


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
Activity Diagram ของ life cycle ของ การลงทะเบียน
 . ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



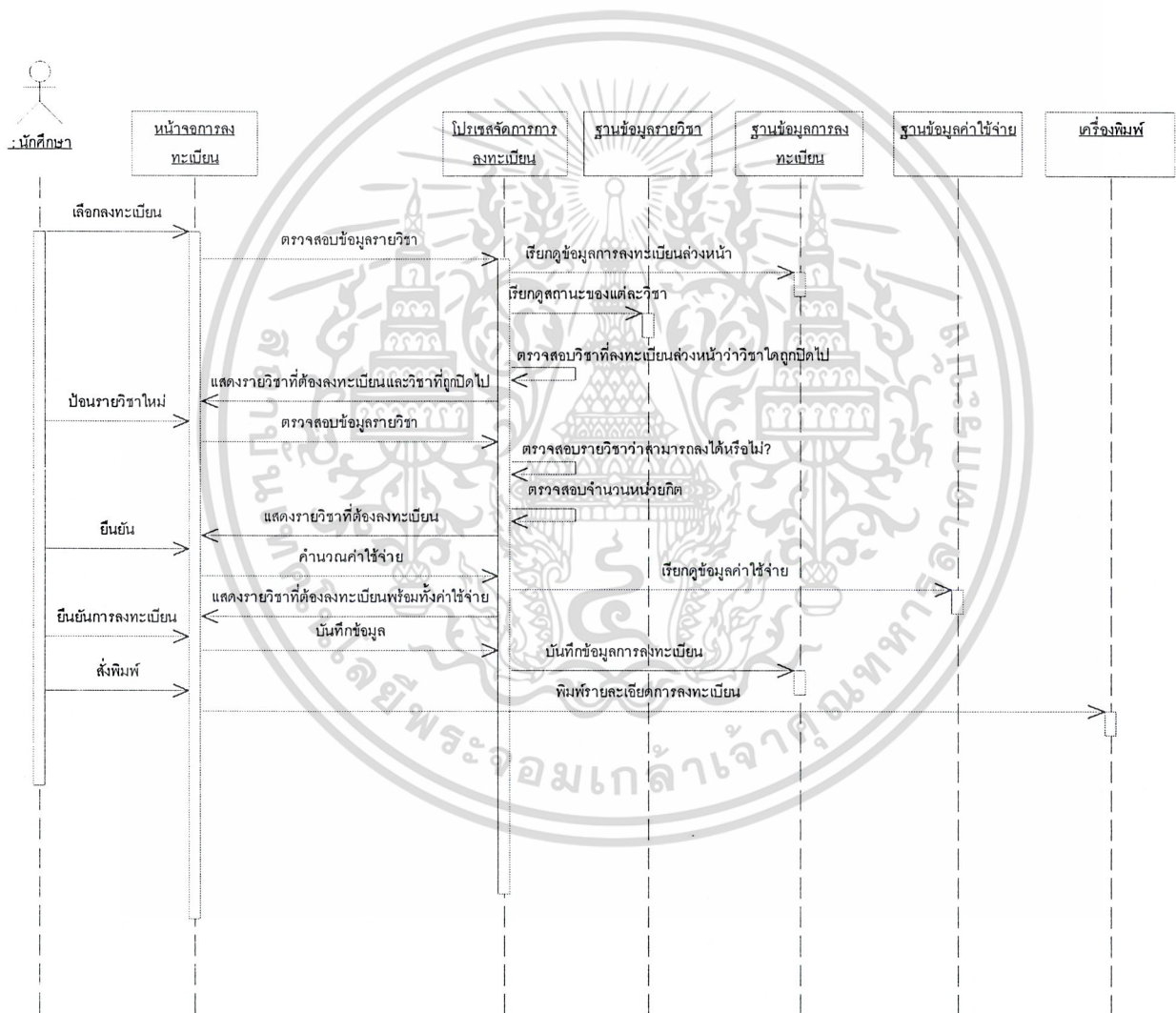
Sequence Diagram ของ การลงทะเบียนล่วงหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

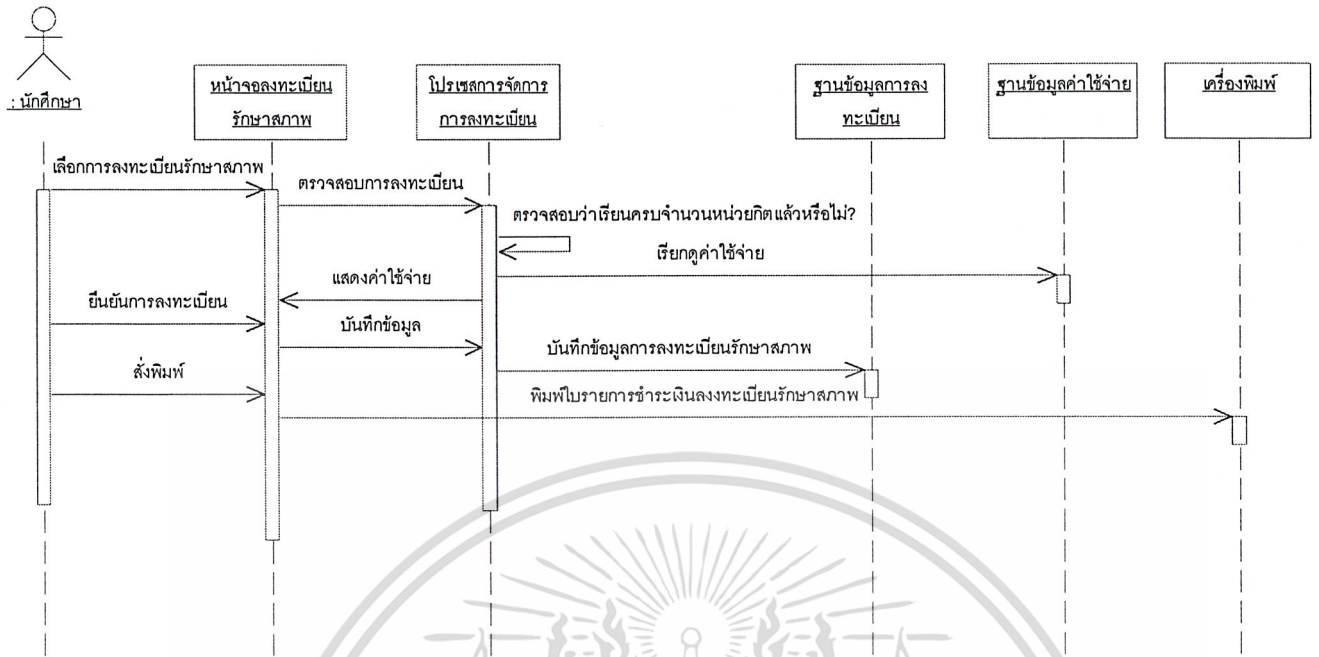


Sequence Diagram ของ การ ลงทะเบียนปกติในกรณีที่ไม่มีการปิด

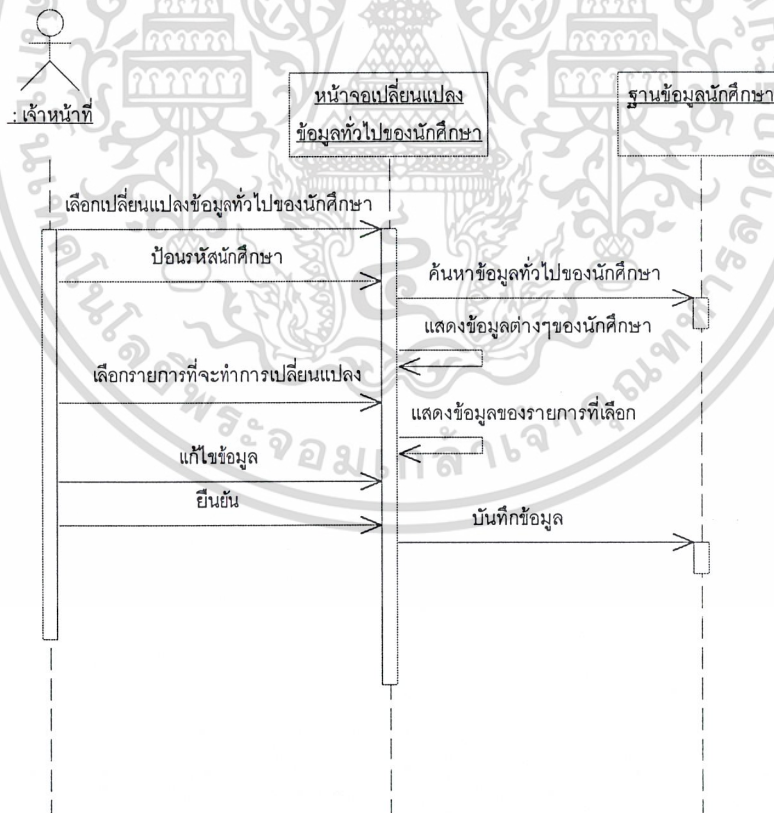
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



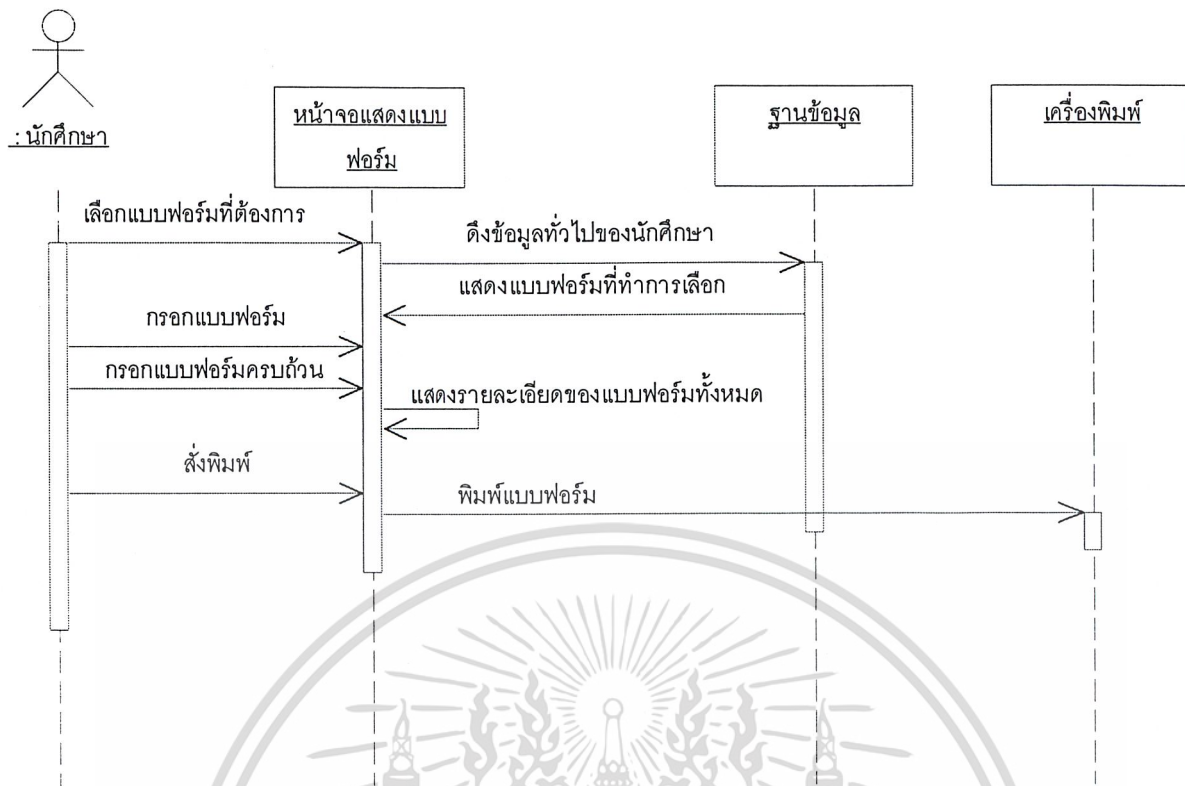
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
Sequence Diagram ของ การลงทะเบียนปกติในกรณีที่มิมีวิชาถูกปิด
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุเปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



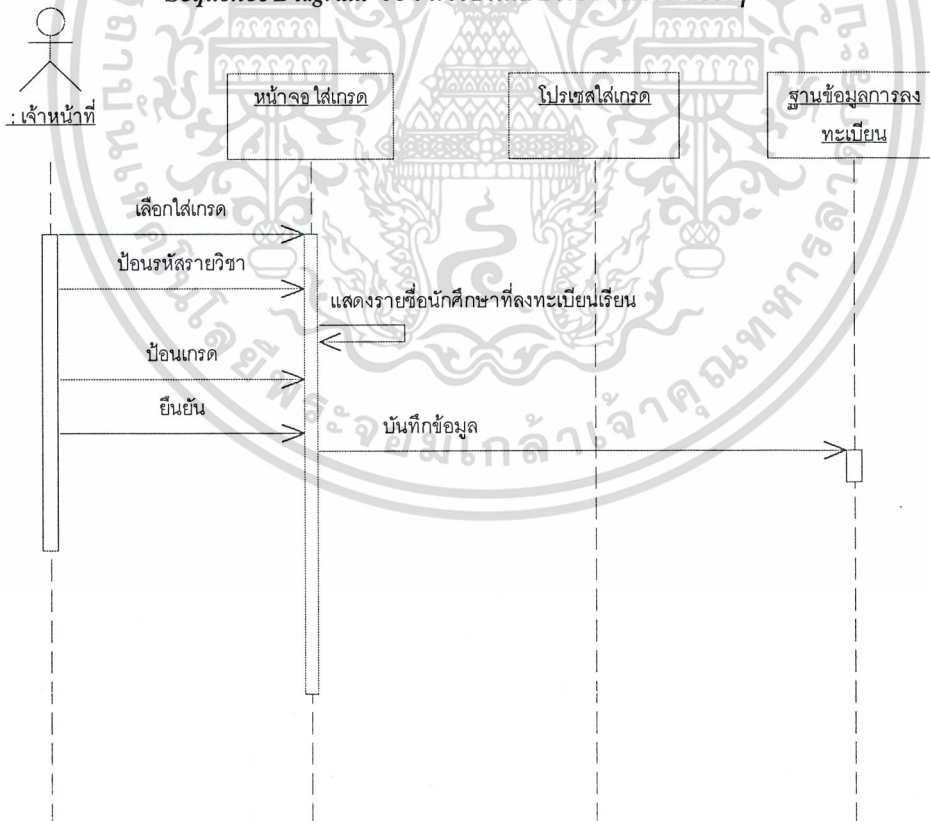
Sequence Diagram ของ การ ลงทะเบียนรักษาสภาพ



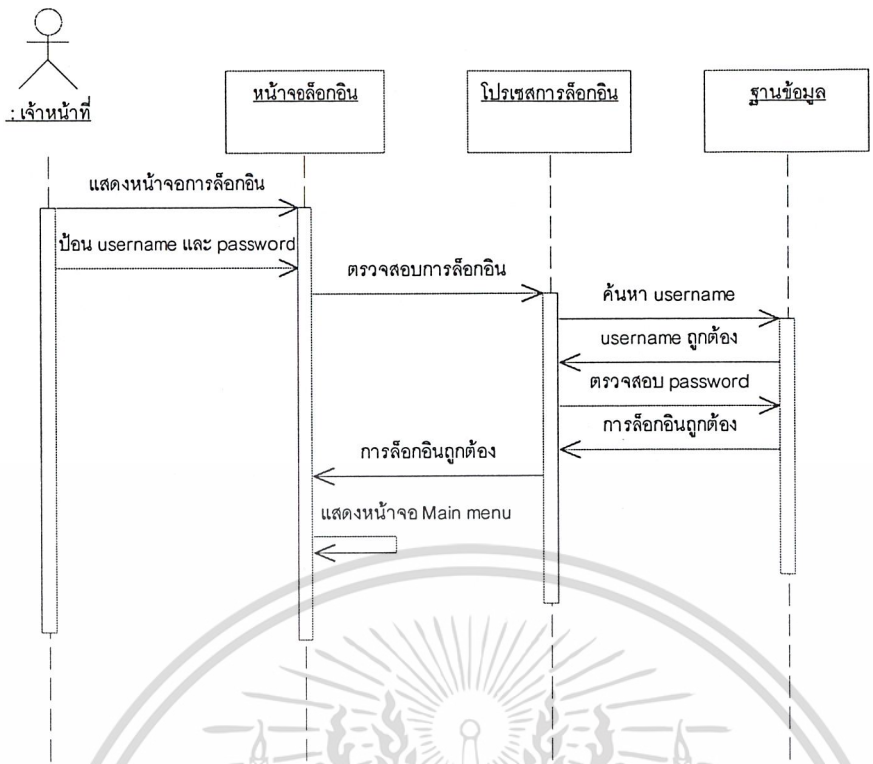
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
Sequence Diagram ของ การ เปลี่ยนแปลงข้อมูลทั่วไปของนักศึกษา
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



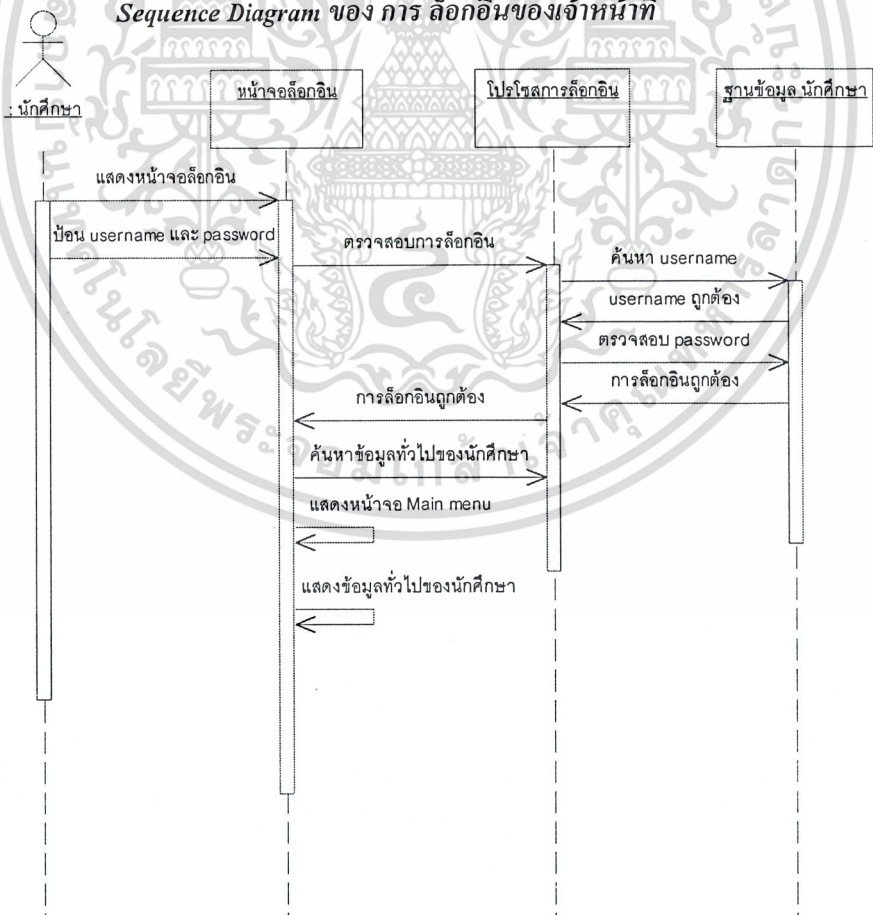
Sequence Diagram ของ การยื่นแบบฟอร์มคำร้องต่างๆ



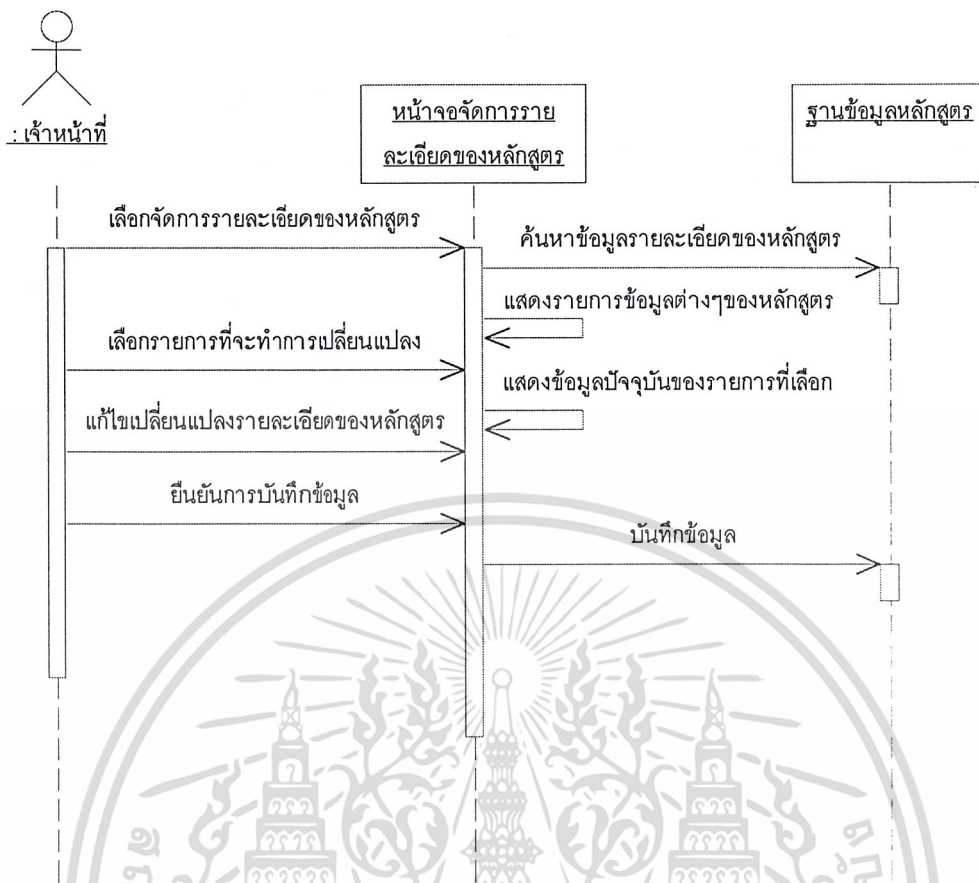
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
Sequence Diagram ของ การ ใ้เลือก
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



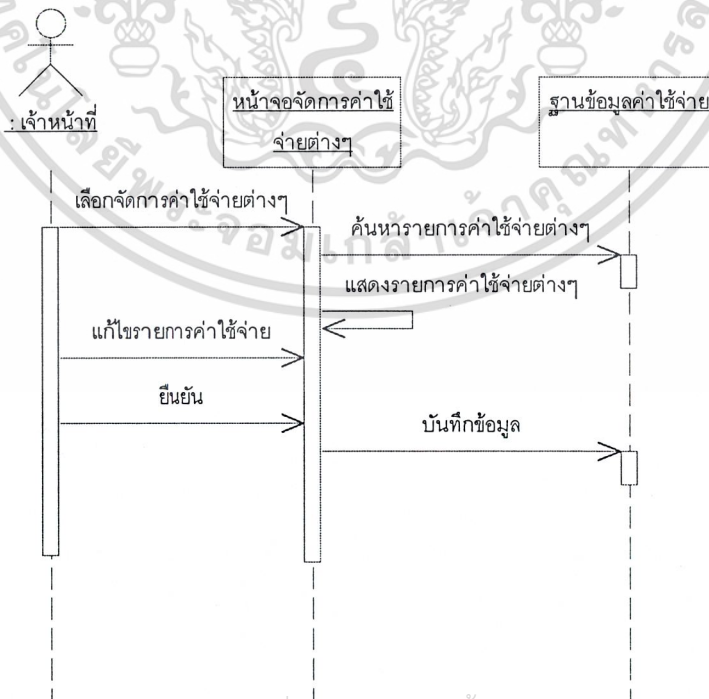
Sequence Diagram ของ การ ล็อกอินของเจ้าหน้าที่



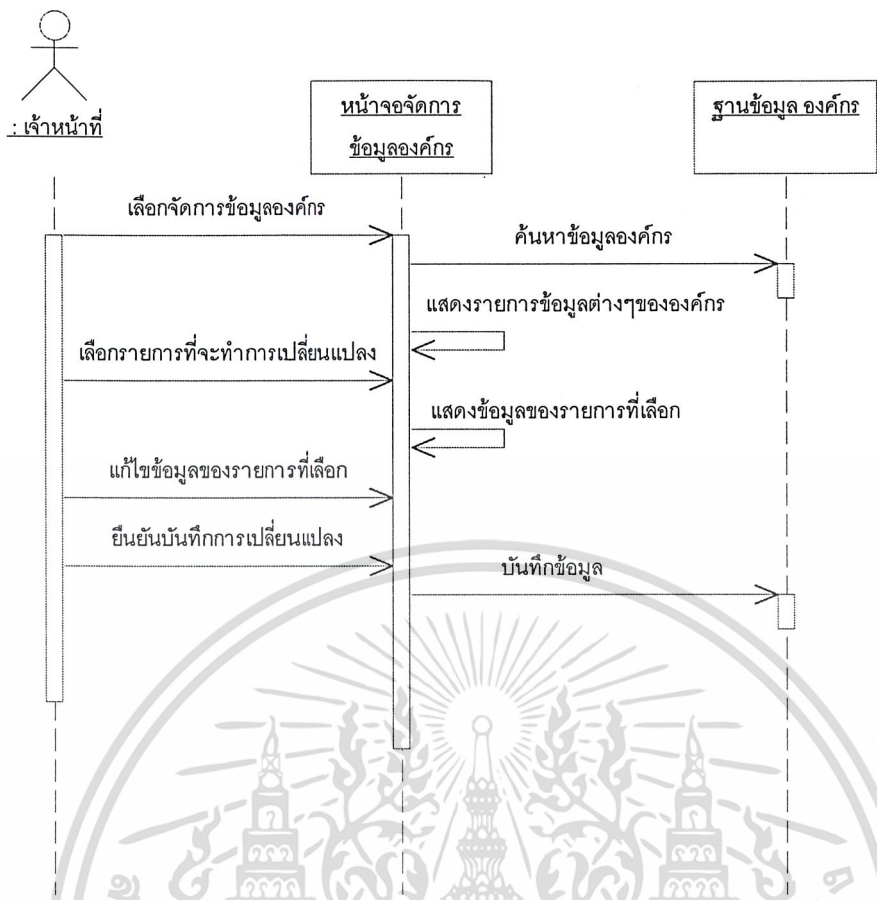
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
Sequence Diagram ของ การ ล็อกอินของนักศึกษา
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



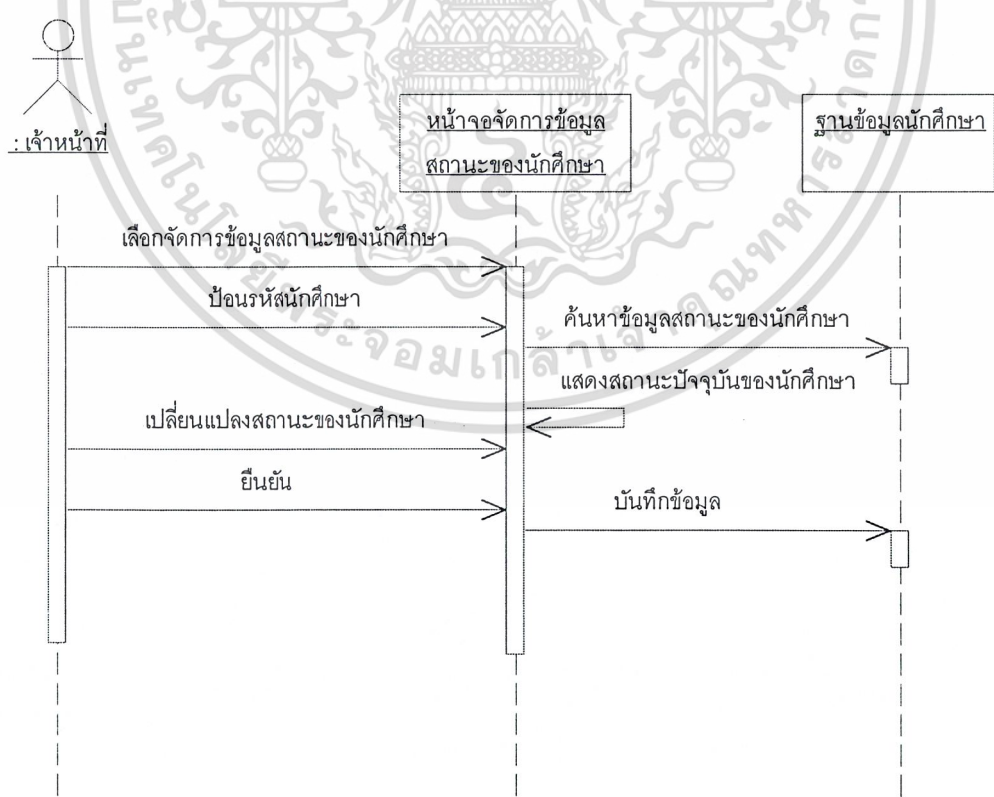
Sequence Diagram ของ การ จัดการรายละเอียดของหลักสูตร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
Sequence Diagram ของ การ จัดการค่าใช้จ่ายต่างๆ
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

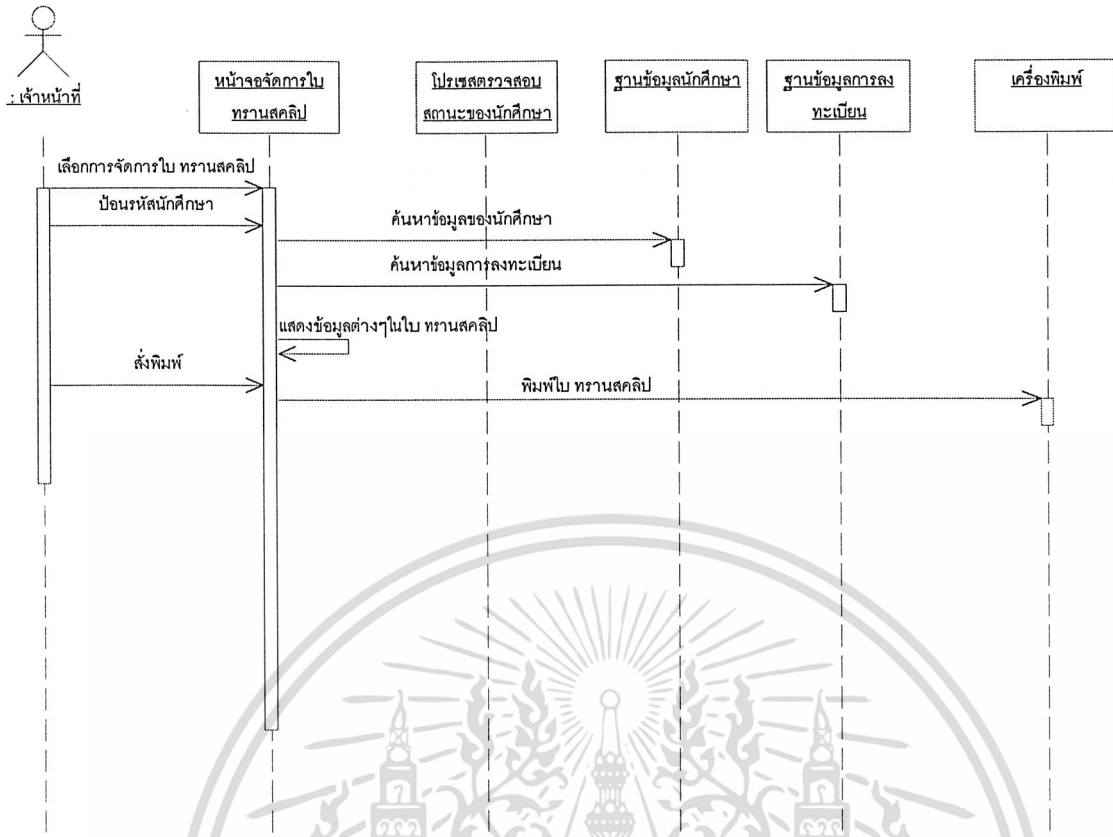


Sequence Diagram ของ การ จัดการข้อมูลองค์กร

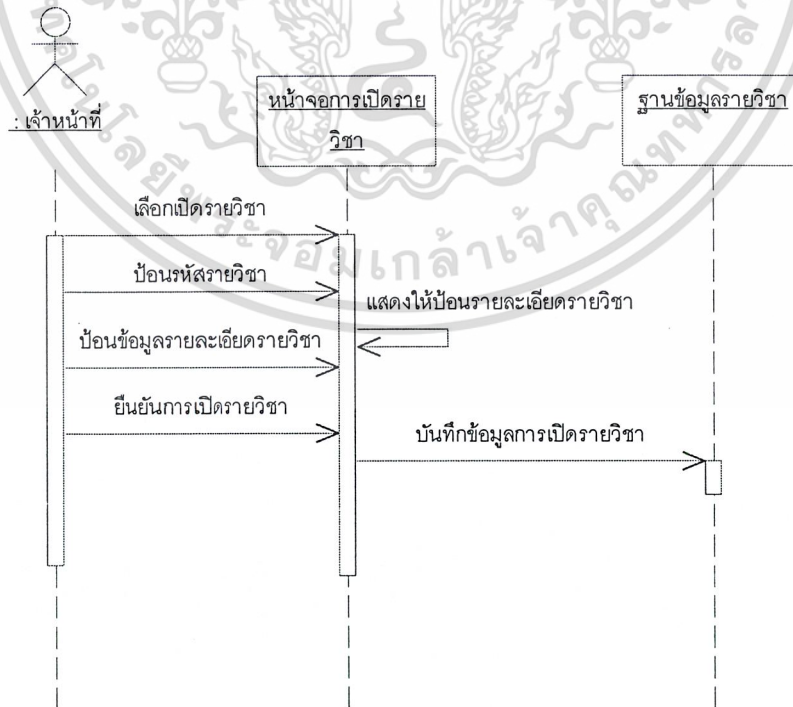


Sequence Diagram ของ การ จัดการข้อมูลสถานะของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 Sequence Diagram ของ การ จัดการข้อมูลสถานะของนักศึกษา
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

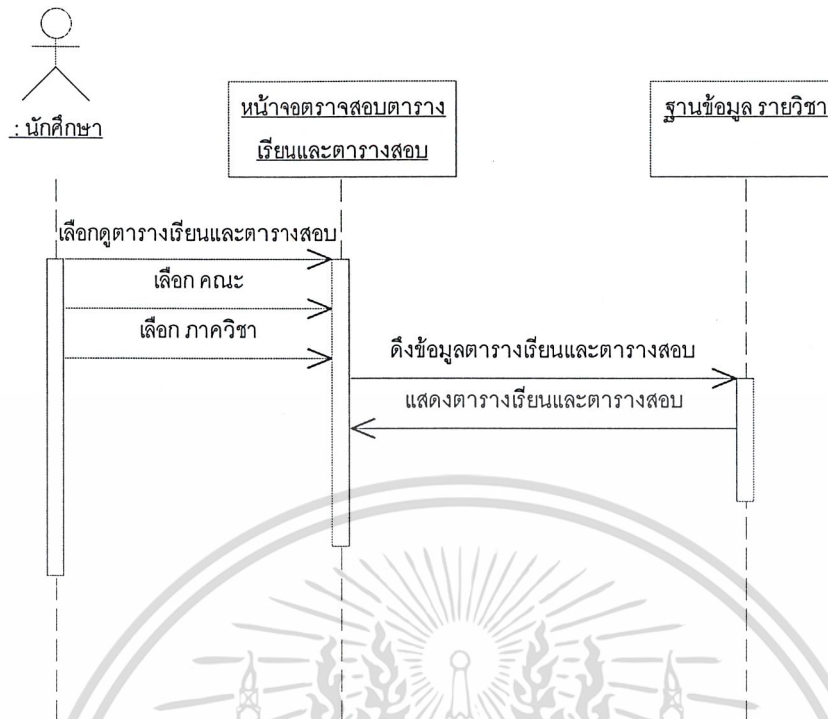


Sequence Diagram ของ การ จัดการใบ ทรานสคริป



Sequence Diagram ของ การ เปิดรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

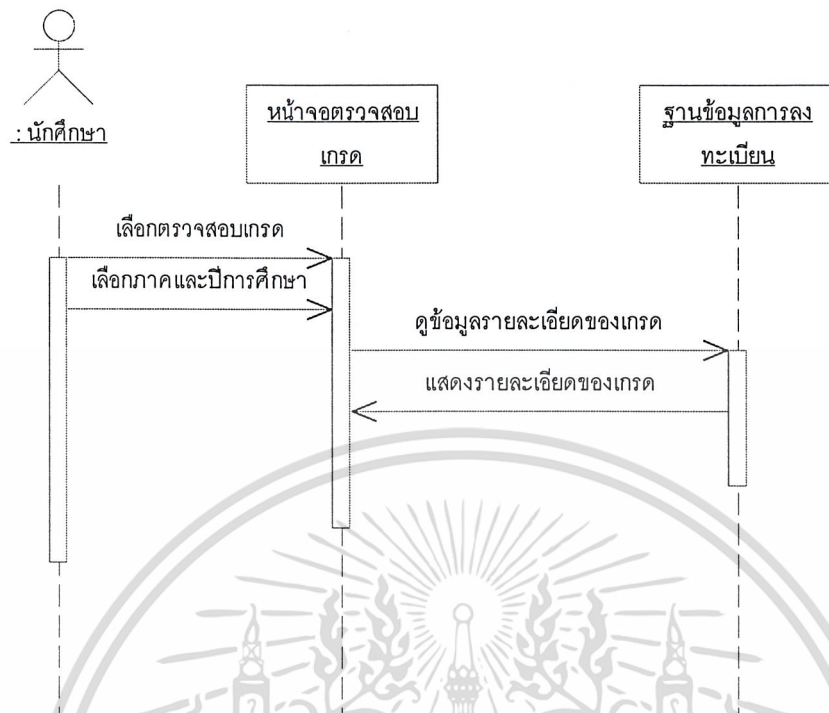


Sequence Diagram ของ การ ตรวจสอบตารางเรียนและตารางสอบ

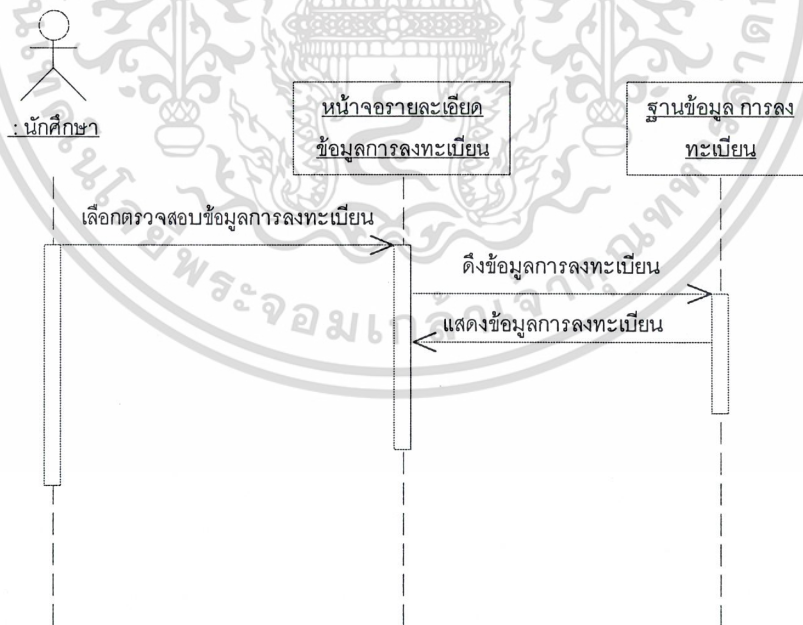


Sequence Diagram ของ การ ตรวจสอบรายละเอียดรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

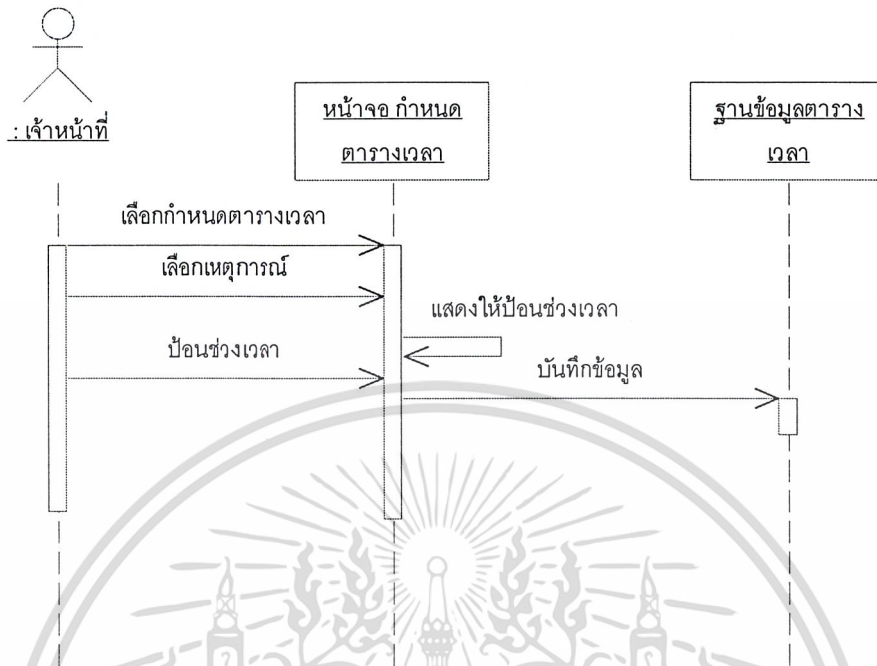


Sequence Diagram ของ การ ตรวจสอบเกรด

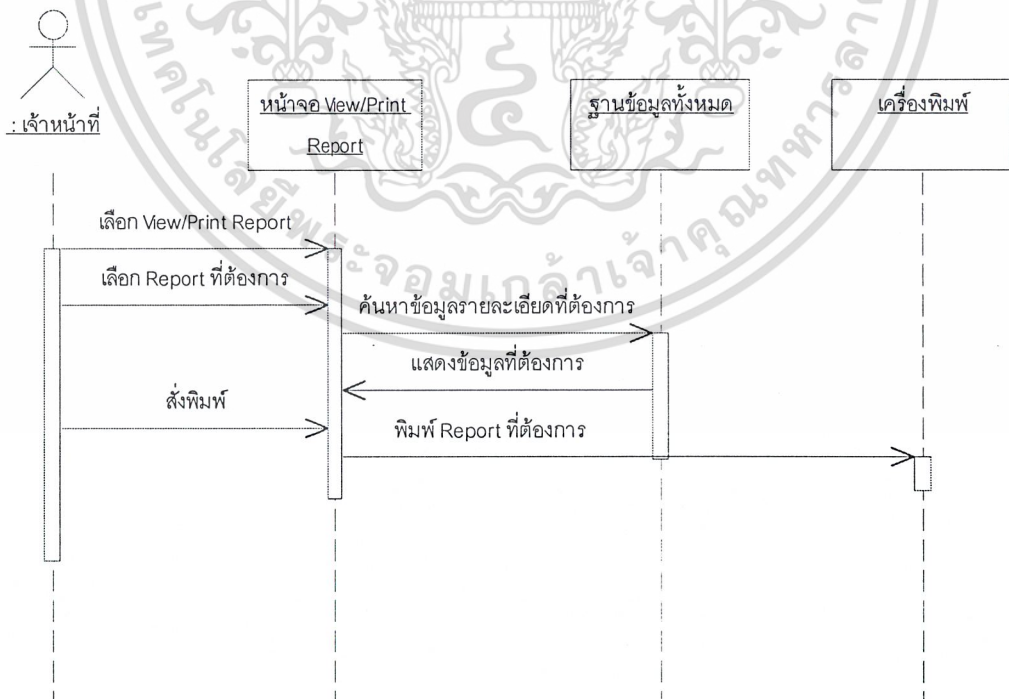


Sequence Diagram ของ การ ตรวจสอบข้อมูลการลงทะเบียน

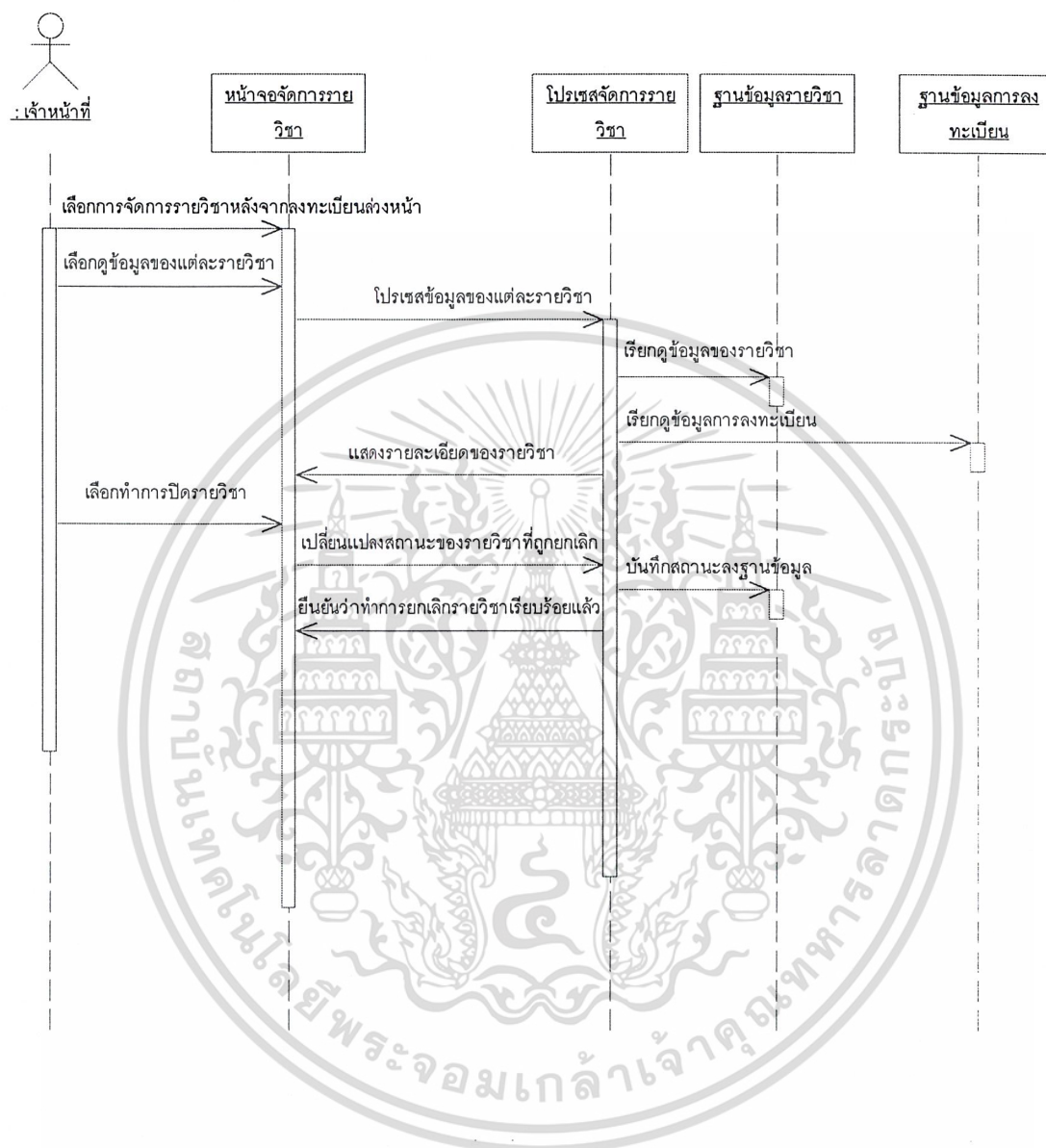
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Sequence Diagram ของ การ กำหนดตารางเวลา

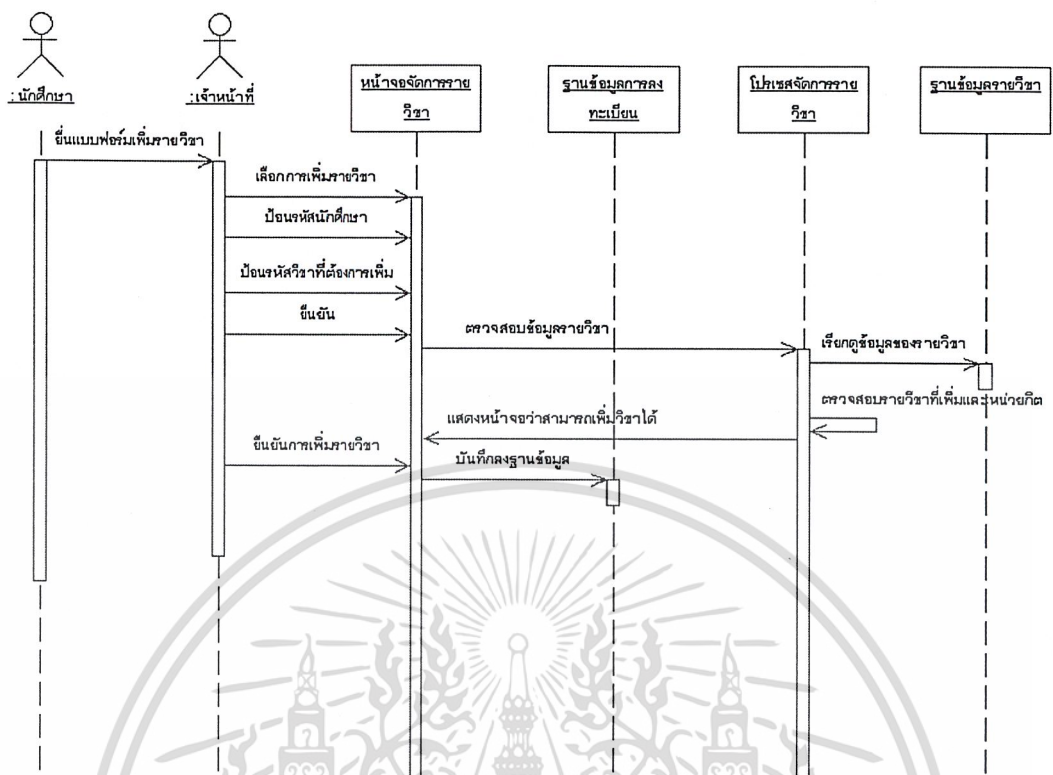


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 Sequence Diagram ของ การ ดูและพิมพ์รายงาน
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



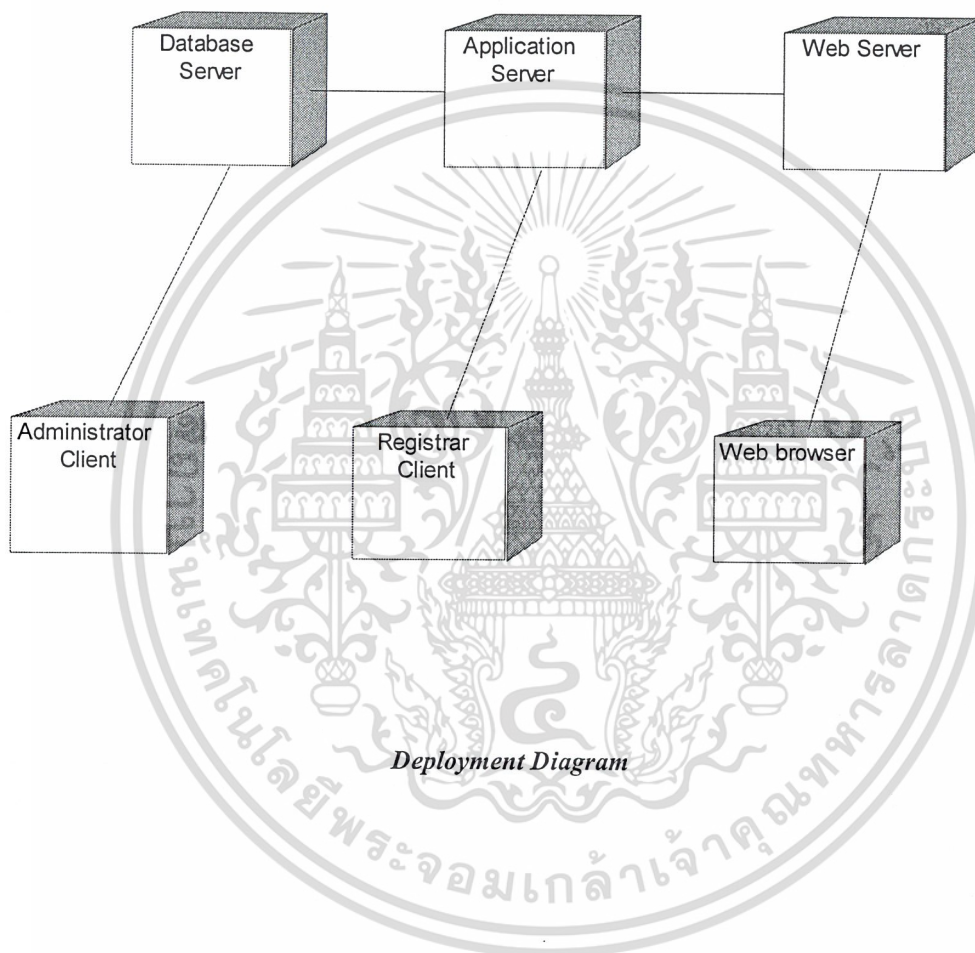
Sequence Diagram ของ การ จัดการรายวิชาหลังจากการลงทะเบียนล่วงหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Sequence Diagram ของ การเพิ่มรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
โครงสร้างตารางฐานข้อมูล (Table Schema)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างตารางฐานข้อมูล(Table Schema)

FACULTY		
FACT#	VARCHAR(2)	รหัสคณะ (PK)
ENAME	VARCHAR(50)	ชื่อคณะภาษาอังกฤษ
TNAME	VARCHAR(50)	ชื่อคณะภาษาไทย

DEPARTMENT		
DEPT#	VARCHAR(2)	รหัสภาควิชา (PK)
ENAME	VARCHAR(50)	ชื่อภาควิชาภาษาอังกฤษ
TNAME	VARCHAR(50)	ชื่อภาควิชาภาษาไทย
FACT#	VARCHAR(2)	รหัสคณะ (FK)

MAJOR		
MAJOR#	VARCHAR(2)	รหัสสาขาวิชา (PK)
ENAME	VARCHAR(50)	ชื่อสาขาวิชาภาษาอังกฤษ
TNAME	VARCHAR(50)	ชื่อสาขาวิชาภาษาไทย
DEPT#	VARCHAR(2)	รหัสภาควิชา (FK)

MINOR		
MINOR#	VARCHAR(2)	รหัสแขนงวิชา (PK)
ENAME	VARCHAR(50)	ชื่อแขนงวิชาภาษาอังกฤษ
TNAME	VARCHAR(50)	ชื่อแขนงวิชาภาษาไทย
MAJOR#	VARCHAR(2)	รหัสสาขาวิชา (FK)

LECTURER		
LECT#	VARCHAR(4)	รหัสอาจารย์ (PK)
EPRE	VARCHAR(20)	คำนำหน้าชื่อภาษาอังกฤษ
ENAME	VARCHAR(50)	ชื่อภาษาอังกฤษ
EFAMILY	VARCHAR(50)	นามสกุลภาษาอังกฤษ
TPRE	VARCHAR(20)	คำนำหน้าชื่อภาษาไทย
TNAME	VARCHAR(50)	ชื่อภาษาไทย
TFAMILY	VARCHAR(50)	นามสกุลภาษาไทย

POSITION	VARCHAR(50)	ตำแหน่ง
DEPT#	VARCHAR(2)	รหัสภาควิชาที่สังกัด (FK)

STUDENT		
STD#	VARCHAR(8)	รหัสนักศึกษา (PK)
LECT#	VARCHAR(4)	รหัสอาจารย์ที่ปรึกษา (FK)
THESIS#	SMALLINT	รหัส THESIS ที่วิจัย (FK)
SCO#	SMALLINT	รหัสหลักสูตรย่อยที่สังกัด (FK)
EPRE	VARCHAR(20)	คำนำหน้าชื่อภาษาอังกฤษ
ENAME	VARCHAR(50)	ชื่อภาษาอังกฤษ
EFAMILY	VARCHAR(50)	นามสกุลภาษาอังกฤษ
TPRE	VARCHAR(20)	คำนำหน้าชื่อภาษาไทย
TNAME	VARCHAR(50)	ชื่อภาษาไทย
TFAMILY	VARCHAR(50)	นามสกุลภาษาไทย
PASSWORD	VARCHAR(20)	รหัสผ่านของนักศึกษา
INTERM	VARCHAR(1)	ภาคเรียนที่เข้าศึกษา
INYEAR	VARCHAR(4)	ปีที่เข้าศึกษา
CO#	SMALLINT	รหัสหลักสูตร (FK)
FACT#	VARCHAR(2)	รหัสคณะ (FK)
DEPT#	VARCHAR(2)	รหัสภาควิชา (FK)
MAJOR#	VARCHAR(2)	รหัสสาขาวิชา (FK)
MINOR#	VARCHAR(2)	รหัสแขนงวิชา (FK)
TYPE	VARCHAR(1)	ประเภทนักศึกษา
SEX	VARCHAR(1)	เพศ
BIRTHDAY	DATE	วันเกิด
ORIGIN	VARCHAR(20)	เชื้อชาติ
CITIZEN	VARCHAR(20)	สัญชาติ
RELIGION	VARCHAR(10)	ศาสนา
RECENT_ADDR	VARCHAR(300)	ที่อยู่ปัจจุบัน
REGISTER_ADDR	VARCHAR(300)	ที่อยู่ตามทะเบียนบ้าน
HOME_PHONE	VARCHAR(20)	โทรศัพท์บ้าน
MOBILE_PHONE	VARCHAR(20)	โทรศัพท์เคลื่อนที่
EMAIL	VARCHAR(20)	อีเมล

OLDINSTITUTE	VARCHAR(100)	ชื่อสถานศึกษาเดิม
OLDYROUT	VARCHAR(4)	ปีที่จบการศึกษาจากสถานศึกษาเดิม
OLDGPA	VARCHAR(4)	คะแนนเฉลี่ยสะสมเดิม
OLDDEGREE	VARCHAR(100)	ปริญญาที่ได้รับ (ชื่อเต็ม)
OLDSDEGREE	VARCHAR(10)	ปริญญาที่ได้รับ (ชื่อย่อ)
OLDFAC	VARCHAR(50)	คณะที่จบจากสถานศึกษาเดิม
OLDMAJOR	VARCHAR(50)	สาขาวิชาที่จบจากสถานศึกษาเดิม
FATHER	VARCHAR(100)	ชื่อนามสกุลของบิดา
FATHER_WORK	VARCHAR(20)	อาชีพของบิดา
MOTHER	VARCHAR(100)	ชื่อนามสกุลของมารดา
MOTHER_WORK	VARCHAR(20)	อาชีพของมารดา
CONTACT	VARCHAR(100)	ชื่อนามสกุลของผู้ที่ติดต่อนักศึกษาได้ สะดวก
CO_WORK	VARCHAR(20)	อาชีพของผู้ที่ติดต่อนักศึกษาได้สะดวก
CO_RELATE	VARCHAR(15)	ผู้ที่ติดต่อได้สะดวกมีความเกี่ยวข้องกับ
CO_TEL	VARCHAR(20)	เบอร์โทรศัพท์ของผู้ที่ติดต่อได้สะดวก
CO_ADDR	LONG VARCHAR	ที่อยู่ปัจจุบันของผู้ที่ติดต่อได้สะดวก
INDATE	DATE	วันที่เข้าศึกษา
OUTDATE	DATE	วันที่จบการศึกษา
LOCATION	VARCHAR(20)	ศูนย์ที่เรียน (ลาดกระบัง, อุดร, หาดใหญ่ ...)

SUBJECT		
SUBJ#	VARCHAR(8)	รหัสวิชา (PK)
ENAME	VARCHAR(85)	ชื่อวิชาภาษาอังกฤษ
TNAME	VARCHAR(85)	ชื่อวิชาภาษาไทย
GROUP	VARCHAR(30)	กลุ่มวิชา(T:THESIS,E:วิชาเลือก,C:บังคับ)
CREDIT	VARCHAR(2)	หน่วยกิตรวม
LECTURE	VARCHAR(2)	เวลาบรรยาย
PRACTICE	VARCHAR(2)	เวลาปฏิบัติ
RESEARCH	VARCHAR(2)	เวลาดင်คั่ว
EOUTLINE	VARCHAR(500)	คำอธิบายรายวิชาภาษาอังกฤษ
TOUTLINE	VARCHAR(500)	คำอธิบายรายวิชาภาษาไทย
CO#	SMALLINT	รหัสหลักสูตร (PK)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ในการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีเหตุที่เปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DEGREE	VARCHAR(1)	ปริญญาโท / เอก (M:Master,D:Doctor)
MAJOR#	VARCHAR(2)	รหัสสาขาวิชา (FK)
FACT#	VARCHAR(2)	รหัสคณะ (FK)
DEPT#	VARCHAR(2)	รหัสภาควิชา (FK)
MINOR#	VARCHAR(2)	รหัสแขนงวิชา (FK)

SUBCOURSE		
SCO#	SMALLINT	รหัสหลักสูตรย่อย (PK) (Generate)
ENAME	VARCHAR(100)	ชื่อหลักสูตรภาษาอังกฤษ
TNAME	VARCHAR(100)	ชื่อหลักสูตรภาษาไทย
STD_TYPE	VARCHAR(1)	ชนิดหลักสูตร (สามัญ, ทดลองเรียน) (C:สามัญ, T:ทดลองเรียน)
SEC_TYPE	VARCHAR(1)	ประเภทหลักสูตร (ปกติ, สมทบ, เสาร์-อาทิตย์) (A:ปกติ, B:สมทบ, C:เสาร์-อาทิตย์)
CO#	SMALLINT	รหัสหลักสูตร (FK)
TOTAL_CREDIT	SMALLINT	โครงสร้างหลักสูตร(หน่วยกิต)
TOTAL_YEAR	SMALLINT	เวลาของแผนการศึกษา(ปี)

COST		
COST#	SMALLINT	รหัสค่าใช้จ่าย (PK) (Generate)
NAME	VARCHAR(100)	ชื่อค่าใช้จ่าย
VALUE	DECIMAL(8,2)	จำนวนเงิน
SEMESTER	VARCHAR(1)	เทอมที่ต้องชำระ
YEAR	VARCHAR(1)	ปีที่ต้องชำระ
TYPE	VARCHAR(1)	ชนิดของค่าใช้จ่าย (C:Common, S:Special)
SCO#	SMALLINT	รหัสหลักสูตรย่อย (FK)

REGISTRAR		
REG#	VARCHAR(4)	รหัสเจ้าหน้าที่ (PK)
EPRE	VARCHAR(20)	คำนำหน้าชื่อภาษาอังกฤษ
ENAME	VARCHAR(50)	ชื่อภาษาอังกฤษ
EFAMILY	VARCHAR(50)	นามสกุลภาษาอังกฤษ
TPRE	VARCHAR(20)	คำนำหน้าชื่อภาษาไทย

TNAME	VARCHAR(50)	ชื่อภาษาไทย
TFAMILY	VARCHAR(50)	นามสกุลภาษาไทย
POSITION	VARCHAR(50)	ตำแหน่ง
PASSWORD	VARCHAR(20)	รหัสผ่าน
PERMISSION	VARCHAR(1)	สิทธิ์(N:Normal,S:Special)

TIME_SCHEDULE		
TIME#	SMALLINT	รหัสกำหนดเวลา (PK) (Generate)
NAME	VARCHAR(50)	ชื่อกำหนดเวลา
BEGIN	DATE	วันที่เริ่มต้น
END	DATE	วันที่สิ้นสุด

PREREQUISITE		
SUBJ#	VARCHAR(8)	รหัสรายวิชา (PK)
PRE_SUBJ#	VARCHAR(8)	รหัสรายวิชาที่ต้องเรียนก่อน (PK)

REGISTER		
REG#	SMALLINT	รหัสลงทะเบียน (PK) (Generate)
STD#	VARCHAR(8)	รหัสนักศึกษา
SEC#	SMALLINT	รหัสสอน
YEAR	VARCHAR(4)	ปีการศึกษา
SEMESTER	VARCHAR(1)	ภาคการศึกษา
REGISTER_TYPE	VARCHAR(1) (P,C)	ประเภทการลงทะเบียน(ล่วงหน้า,ปกติ)
SECTION_TYPE	VARCHAR(1)	ประเภทของตอนที่ลงทะเบียน (Credit,Audit)
GRADE	VARCHAR(2)	เกรดที่ได้

GRADE		
GRADE#	SMALLINT	รหัสเกรด (PK) (Generate)
GRADE	VARCHAR(2)	เกรด (PK)
POINT	DECIMAL(2,1)	ค่าคะแนน
DESCRIPTION	VARCHAR(300)	คำอธิบายค่าคะแนน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TEACH		
SEC#	SMALLINT	รหัสตอน (PK)
LECT#	VARCHAR(4)	รหัสอาจารย์ผู้สอน (PK)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค
แบบฟอร์มคำร้องต่าง ๆ ของนักศึกษาระดับบัณฑิตศึกษา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มคำร้องทั่วไป

วันที่.....เดือน.....พ.ศ.....

เรื่อง

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า (นาย,นาง,นางสาว,อื่นๆ)

รหัสประจำตัว.....นักศึกษาชั้นปีที่.....สาขาวิชา.....ภาควิชา.....

แขนงวิชา.....คณะ.....สถานที่ติดต่อ.....

โทร./มือถือ.....

อีเมล.....

มีความประสงค์

จึงเรียนมาเพื่อโปรดพิจารณาดำเนินการต่อไป

ขอแสดงความนับถือ

ลงชื่อ

(.....)

ความเห็นอาจารย์ที่ปรึกษา

ลงชื่อ

(.....)

ความเห็นของเจ้าหน้าที่	คำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปเผยแพร่ภายนอก
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มคำร้องขอ TRANSCRIPT

วันที่.....เดือน.....พ.ศ.....

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า (นาย,นาง,นางสาว,อื่น ๆ.....) ชื่อ - นามสกุล.....

(Mr.,Mrs.,Miss,Other.....)(เขียนภาษาอังกฤษตัวพิมพ์ใหญ่)

วัน,เดือน,ปีเกิด.....(เขียนภาษาอังกฤษ ห้ามเขียนตัวย่อ คัด ค.ศ.โดย พ.ศ.เกิดลบด้วย 543)

รหัสประจำตัวนักศึกษา..... หลักสูตร.....แผน.....

คณะ.....สาขาวิชา.....

แขนงวิชา / วิชาเอก.....

สำเร็จการศึกษาเมื่อภาคเรียนที่.....ปีการศึกษา.....

ยังไม่สำเร็จการศึกษา

มีความประสงค์ขอ Transcript เพื่อวัตถุประสงค์.....

จำนวน.....ชุด (ชุดละ 20 บาท) พร้อมทั้งแนบรูปถ่ายจำนวน.....รูป

หลักฐานประกอบการขอ Transcript

กรณีนักศึกษาสำเร็จการศึกษา

- รูปถ่ายขนาด 1 นิ้ว ถ่ายมาแล้วไม่เกิน 6 เดือน สวมเสื้อครุย (ภาพสีหรือขาวดำ) 1 ชุด ต่อ 1 รูป
- ใบรับรองวิทยานิพนธ์ (ถ่ายเอกสารแล้วให้ฝ่ายวิจัยฯ รับรองเกรด)

กรณีนักศึกษายังไม่สำเร็จการศึกษา

- รูปถ่ายขนาด 1 นิ้ว ถ่ายมาแล้วไม่เกิน 6 เดือน (ภาพสีหรือขาวดำ) 1 ชุดต่อ 1 รูป
 - นักศึกษาชาย สวมเสื้อเชิ้ตสีขาวเกลี้ยงไม่มีลายและผูกเนคไทสีดำ หรือสีกรมท่าไม่มีลาย เท่านั้น
 - นักศึกษาหญิง สวมเสื้อเชิ้ตสีขาวเกลี้ยงไม่มีลาย แขนสั้น เท่านั้น

จึงเรียนมาเพื่อโปรดพิจารณาดำเนินการ

ลงชื่อ.....

ที่อยู่ติดต่อได้.....

โทรศัพท์(บ้าน).....โทรศัพท์(ที่ทำงาน).....มือถือ.....

อีเมล.....

หมายเหตุ : 1. ผู้ขอจะมารับ Transcript ได้หลังจากวันที่ขออย่างน้อย 10 วันทำการ

2. ผู้ขอต้องกรอกข้อความข้างต้นด้วยตัวบรรจงให้ชัดเจน หากผิดพลาดบัณฑิตวิทยาลัยจะไม่รับผิดชอบ

3. ผู้ขอต้องยื่นเอกสารประกอบการขอ Transcript ให้ครบถ้วน มิฉะนั้นบัณฑิตวิทยาลัยจะไม่พิจารณา

เอกสารนี้ เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยญาติให้นำไปเผยแพร่บนสื่อออนไลน์

ไม่ว่ากรณีใดๆ กรุณานำบัตรนักศึกษามาด้วยทุกครั้ง เพื่อติดต่อขอรับ Transcript ของเอกสารทุกครั้งที่มีการนำไปใช้



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มคำร้องขอหนังสือรับรองผลการศึกษา

วันที่.....เดือน.....พ.ศ.....

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า (นาย,นาง,นางสาว,อื่นๆ)

รหัสประจำตัว..... หลักสูตร วศ.ด. คอ.ด. วศ.ม. ผ.ม. สถ.ม. วท.ม. คอ.ม.

แผน.....นักศึกษาชั้นปีที่.....สาขาวิชา.....ภาควิชา

แขนงวิชาคณะ.....

มีความประสงค์ขอหนังสือรับผลการศึกษา ทุกภาคการศึกษา (ชุดละ 20 บาท) เฉพาะภาคเรียนที่.....
ปีการศึกษา..... จำนวน.....ชุด เพื่อวัตถุประสงค์.....

จึงเรียนมาเพื่อโปรดพิจารณาดำเนินการต่อไป

ขอแสดงความนับถือ

ลงชื่อ
(.....)

ที่อยู่ติดต่อได้.....

โทรศัพท์(บ้าน).....โทรศัพท์(ที่ทำงาน).....มือถือ.....

อีเมล.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มคำร้องขอหนังสือรับรอง

วันที่.....เดือน.....พ.ศ.....

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า (นาย,นาง,นางสาว,อื่นๆ).....

รหัสประจำตัว.....หลักสูตร วศ.ด. คอ.ด. วศ.ม. ผ.ม. สศ.ม. วท.ม. คอ.ม.

แผน.....นักศึกษาชั้นปีที่.....สาขาวิชา.....ภาควิชา.....

แขนงวิชา.....คณะ.....

มีความประสงค์ขอหนังสือรับรองดังต่อไปนี้

หนังสือรับรองส่งตัวกลับเข้ารับราชการ

หนังสือขอขยายเวลาการศึกษาต่อจากต้นสังกัด

อื่นๆ.....

(โปรดเพิ่มเติมรายละเอียดหากต้องการ).....

จึงเรียนมาเพื่อโปรดพิจารณาดำเนินการต่อไป

ขอแสดงความนับถือ

ลงชื่อ.....
(.....)

ที่อยู่ติดต่อได้.....

โทรศัพท์(บ้าน).....โทรศัพท์(ที่ทำงาน).....มือถือ.....

อีเมล.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มคำร้องขอลาพักการศึกษา

วันที่.....เดือน.....พ.ศ.....

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า (นาย,นาง,นางสาว,อื่นๆ)

รหัสประจำตัว..... เป็นนักศึกษาชั้นปีที่..... ปีการศึกษา.....หลักสูตร.....

แผน.....สาขาวิชา.....ภาควิชา

แขนงวิชา.....คณะ.....

มีความประสงค์ขอลาพักการศึกษาในภาคเรียนที่.....ปีการศึกษา.....เนื่องจาก.....

.....
.....
.....

จึงเรียนมาเพื่อโปรดพิจารณา



ขอแสดงความนับถือ

ลงชื่อ.....

(.....)

(ผู้ยื่นคำร้อง)

ที่อยู่ติดต่อได้.....

โทรศัพท์(บ้าน).....โทรศัพท์(ที่ทำงาน).....มือถือ.....

อีเมลล์.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(1) ความเห็นของอาจารย์ที่ปรึกษา

- เห็นควรให้ลาพักการศึกษาต่อไปได้
- ไม่เห็นควรให้ลาพักการศึกษา เนื่องจาก.....

.....

.....

.....

.....

.....

.....

.....

(3) คำสั่ง (คณบดีบัณฑิตวิทยาลัย)

- อนุมัติลาพักการศึกษาได้และดำเนินการต่อไป
- ไม่อนุมัติให้ลาพักการศึกษา เนื่องจาก.....

.....

.....

.....

.....

.....

.....

.....

ลงชื่อ.....
 (.....)
 / /

ลงชื่อ.....
 (.....)
 / /

(2) ความเห็นของเจ้าหน้าที่บัณฑิตวิทยาลัย

ได้ตรวจสอบเรียบร้อยแล้ว

- เห็นควรพิจารณาอนุมัติได้ตามระเบียบ
- ไม่เห็นควรพิจารณาอนุมัติ เนื่องจาก.....

.....

.....

.....

- อื่นๆ.....

ลงชื่อ.....
 (.....)
 / /



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มคำร้องขอลาออก

วันที่.....เดือน.....พ.ศ.....

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า (นาย,นาง,นางสาว,อื่นๆ).....

รหัสประจำตัว..... เป็นนักศึกษาชั้นปีที่..... ปีการศึกษา..... หลักสูตร.....

แผน.....สาขาวิชา.....ภาควิชา.....

แขนงวิชา.....คณะ.....

มีความประสงค์ขอลาออกจากการเป็นนักศึกษาของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เนื่องจาก.....

.....

.....

จึงเรียนมาเพื่อโปรดพิจารณา



ขอแสดงความนับถือ

ลงชื่อ.....

(.....)

(ผู้ยื่นคำร้อง)

ที่อยู่ติดต่อได้.....

โทรศัพท์(บ้าน).....โทรศัพท์(ที่ทำงาน).....มือถือ.....

อีเมล.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(1) ความเห็นของอาจารย์ที่ปรึกษา

- เห็นควรให้นศ. ลาออกได้
- ไม่เห็นควรให้นศ. ลาออก เนื่องจาก.....

ลงชื่อ.....
(.....)
..... / /

(2) ความเห็นของเจ้าหน้าที่ที่เกี่ยวข้อง

- บรรณารักษ์ห้องสมุดของคุณฯ

- นศ. ไม่ได้ติดค้ำหนังสือของห้องสมุด
- นศ. ติดค้ำหนังสือของสำนักหอสมุดกลาง
จำนวน.....เล่ม

ลงชื่อ.....
(.....)
..... / /

- บรรณารักษ์สำนักหอสมุดกลาง

- นศ. ไม่ได้ติดค้ำหนังสือของห้องสมุด
- นศ. ติดค้ำหนังสือของสำนักหอสมุดกลาง
จำนวน.....เล่ม

ลงชื่อ.....
(.....)
..... / /

- เจ้าหน้าที่ฝ่ายทุนของการศึกษาของบัณฑิตวิทยาลัย

- นศ. ไม่มีภาระในการรับทุนการศึกษา
- นศ. มีภาระในการรับทุนการศึกษา.....

ลงชื่อ.....
(.....)

(3) ความเห็นของเจ้าหน้าที่บัณฑิตวิทยาลัย

ได้ตรวจสอบเรียบร้อยแล้ว

- นศ. มีภาระในการรับทุนการศึกษา
- นศ. ติดค้ำหนังสือของ.....

จำนวน เล่ม

- นศ. ไม่มีภาระหนี้สินใดๆ ต่อสถาบันฯ
ดังนั้นจึง

เห็นควรพิจารณาอนุมัติให้ลาออก

ไม่เห็นควรพิจารณาอนุมัติให้ลาออก

เนื่องจาก.....

ลงชื่อ.....
(.....)
..... / /

(4) คำสั่ง (คณบดีบัณฑิตวิทยาลัย)

- อนุมัติให้ลาออกตามที่เสนอ
- ไม่อนุมัติให้ลาออก เนื่องจาก.....

ลงชื่อ.....
(.....)
..... / /

เอกสารนี้เป็นเอกสาร/ผลงาน/ลิขสิทธิ์/หรือกรรมสิทธิ์ในการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มคำร้องขอปรับสภาพเป็นนักศึกษาสามัญ

วันที่.....เดือน.....พ.ศ.....

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า (นาย,นาง,นางสาว,อื่นๆ).....

รหัสประจำตัว..... เป็นนักศึกษาชั้นปีที่..... ปีการศึกษา.....หลักสูตร.....

แผน.....สาขาวิชา.....ภาควิชา.....

แขนงวิชา.....คณะ.....

มีความประสงค์จะขอปรับสภาพเป็นนักศึกษาสามัญในภาคเรียนที่.....ปีการศึกษา.....ทั้งนี้ได้ปฏิบัติตามเงื่อนไข
ที่คณะฯ กำหนดแล้ว ดังรายละเอียดต่อไปนี้

1. ได้เรียนผ่านรายวิชาที่คณะฯ กำหนดแล้ว จำนวน วิชา
2. เทียบโอนรายวิชาได้ จำนวน วิชา
3. ได้เสนอบทความเรื่อง.....
ลงตีพิมพ์ในวารสาร.....
4. อื่น ๆ.....

จึงเรียนมาเพื่อโปรดพิจารณา

ขอแสดงความนับถือ

ลงชื่อ.....

(.....)

(ผู้ยื่นคำร้อง)

พร้อมกันนี้ข้าพเจ้าได้ส่งหลักฐาน มาประกอบการพิจารณา ดังนี้

- บทความที่ได้รับการตีพิมพ์ (ถ่ายหน้าปกของวารสารแนบมาด้วย กรณีที่ได้รับการตีพิมพ์แล้ว)
- หนังสือการรับรองผลงานวิจัยทางวิชาการ (ในกรณีที่มีผู้ร่วมทำการวิจัยมากกว่า 2 คน)
- ใบตอบรับการลงตีพิมพ์พร้อมบทความ (กรณีที่ยังไม่ได้ลงตีพิมพ์)

ที่อยู่ติดต่อได้.....

โทรศัพท์(บ้าน).....โทรศัพท์(ที่ทำงาน).....มือถือ.....

อีเมล.....ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(1) ความเห็นของอาจารย์ที่ปรึกษา

- เห็นควรให้ปรับสภาพได้
- ไม่เห็นควรให้ปรับสภาพ เนื่องจาก.....

ลงชื่อ.....
(.....)
..... / /

(2) ความเห็นของเจ้าหน้าที่บัณฑิตวิทยาลัย

ได้ตรวจสอบวิชาเรียนและผลการเรียนเรียบร้อยแล้ว

- นักศึกษาได้ผ่านเงื่อนไขครบถ้วนสมบูรณ์
- นักศึกษายังไม่ผ่านเงื่อนไขตามที่คณะฯ กำหนด เนื่องจาก.....

ลงชื่อ.....
(.....)
..... / /

(3) ความเห็นเจ้าหน้าที่บัณฑิตศึกษา

ได้ตรวจสอบบทความ / หนังสือรับรองทางวิชาการ / เงื่อนไขของหลักสูตร

- ถูกต้องครบถ้วนสมบูรณ์
- ไม่ถูกต้อง เห็นควรนำเข้าสู่ที่ประชุมคณะกรรมการบัณฑิตศึกษาพิจารณา เนื่องจาก.....
- ไม่ถูกต้อง ไม่ควรนำเข้าสู่ที่ประชุมคณะกรรมการบัณฑิตศึกษาพิจารณา เนื่องจาก.....
- อื่น ๆ.....

ลงชื่อ.....
(.....)

(4) ผลการพิจารณากรณีต้องนำเข้าสู่ที่ประชุมคณะกรรมการบัณฑิตศึกษา

- ปรับสภาพนักศึกษาได้ ตั้งแต่ภาคเรียนที่..... ปีการศึกษา.....
- ปรับสภาพนักศึกษาไม่ได้ เนื่องจาก.....

ลงชื่อ.....
(.....)

รองคณบดีฝ่ายบัณฑิตศึกษา
..... / /

(5) คำสั่ง คณบดีบัณฑิตวิทยาลัย

- อนุมัติ
- ไม่อนุมัติ เนื่องจาก.....

ลงชื่อ.....
(.....)

คณบดีบัณฑิตวิทยาลัย
..... / /

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มคำร้องขอขยายเวลาการศึกษา(นักศึกษาสามัญ รหัส 38...., 39....)

วันที่.....เดือน.....พ.ศ.....

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า (นาย,นาง,นางสาว,อื่นๆ).....

รหัสประจำตัว.....เป็นนักศึกษาสามัญตั้งแต่ภาคเรียนที่.....ปีการศึกษา.....หลักสูตร.....

แผน.....สาขาวิชา.....ภาควิชา.....

แขนงวิชา.....คณะ.....

มีความประสงค์ขอขยายเวลาการศึกษาในภาคเรียนที่.....ปีการศึกษา.....เป็นครั้งที่.....เนื่องจากอยู่ในระหว่างการทำวิทยานิพนธ์ในหัวข้อเรื่อง

.....
.....
.....
.....
.....

โดยได้รับอนุมัติหัวข้อวิทยานิพนธ์จากบัณฑิตวิทยาลัยเรียบร้อยแล้ว เมื่อวันที่.....และได้ลงทะเบียนครบหน่วยกิตและรายวิชาตามหลักสูตรแล้ว จำนวน.....หน่วยกิต ได้ระดับคะแนนเฉลี่ยสะสม.....

จึงเรียนมาเพื่อโปรดพิจารณา

ขอแสดงความนับถือ

ลงชื่อ.....

(.....)

(ผู้ยื่นคำร้อง)

ที่อยู่ติดต่อได้.....

โทรศัพท์(บ้าน).....โทรศัพท์(ที่ทำงาน).....มือถือ.....

อีเมล.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(1) ความเห็นของอาจารย์ผู้ควบคุมวิทยานิพนธ์

- เห็นควรให้ขยายเวลาการศึกษาต่อไปได้
- ไม่เห็นควรให้ขยายเวลาการศึกษา เนื่องจาก.....

ลงชื่อ.....
(.....)
..... / /

(2) ความเห็นของเจ้าหน้าที่บัณฑิตวิทยาลัย

1. การขอขยายเวลาครั้งที่ 1

- มีประกาศอนุมัติหัวข้อเรียบร้อยแล้ว
- เรียนครบตามหลักสูตร
- ระดับคะแนนเฉลี่ย.....

2. การขอขยายเวลาครั้งที่ 2, 3, 4

- รายงานความก้าวหน้าวิทยานิพนธ์เพิ่มเติม

ผลการตรวจสอบ

- ถูกต้องตามระเบียบ เห็นควรแจ้งคณะ เพื่อทราบ
 - ไม่ถูกต้องตามระเบียบ เห็นควรนำเสนอบอร์ดคณะกรรมการบัณฑิตศึกษาคณะ.....พิจารณา
- เนื่องจาก.....

ลงชื่อ.....

(3) ความเห็นของคณะ
กรณียุทธต้องตามระเบียบ

- คณะรับทราบ/เสนอคนบดีบัณฑิตวิทยาลัยอนุมัติ
กรณียุทธต้องตามระเบียบ
ผลการพิจารณาของคณะกรรมการบัณฑิตศึกษา
 - เห็นควรให้ขยายเวลาการศึกษาได้ในภาคเรียนที่.....
ปีการศึกษา.....
 - ไม่เห็นควรให้ขยายเวลาการศึกษาในภาคเรียนที่.....
ปีการศึกษา.....
- เนื่องจาก.....

ลงชื่อ.....
(.....)
..... / /

(4) คำสั่ง (คนบดีบัณฑิตวิทยาลัย)

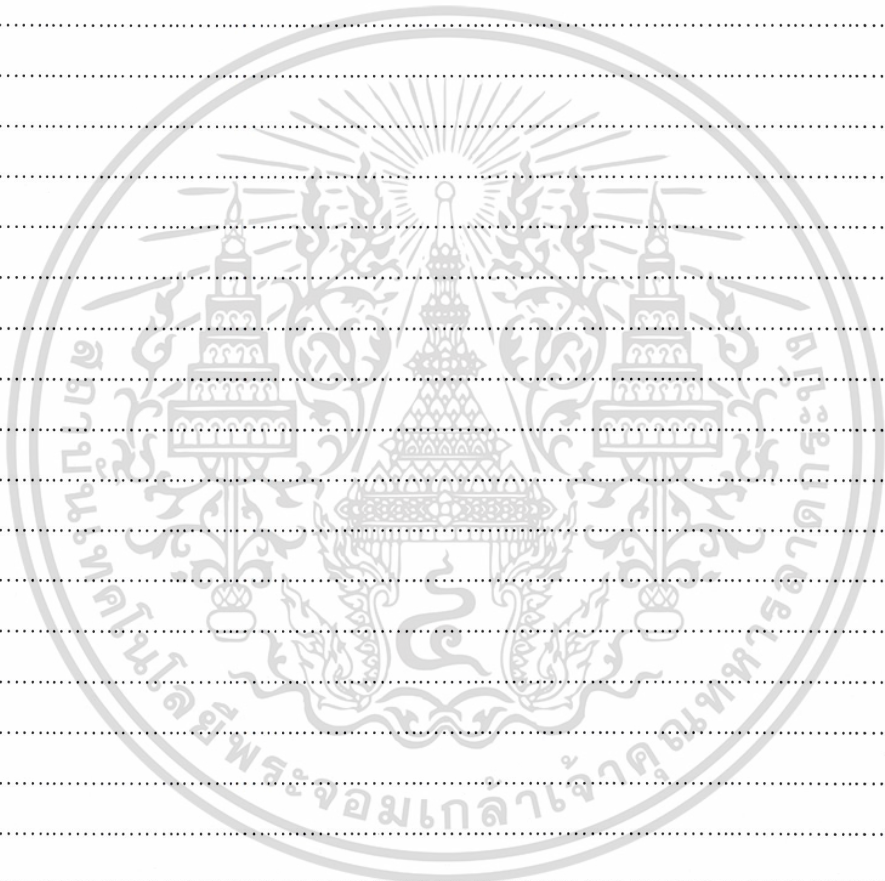
- อนุมัติ
- ไม่อนุมัติ เนื่องจาก.....

ลงชื่อ.....
(.....)
..... / /

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบรายงานความก้าวหน้าในการทำวิทยานิพนธ์

หัวข้อวิทยานิพนธ์(ภาษาไทย).....
 (ภาษาอังกฤษ)
 อาจารย์ควบคุมวิทยานิพนธ์.....
 รายงานความก้าวหน้าของวิทยานิพนธ์ มีรายละเอียดดังต่อไปนี้



ลงชื่อ.....
 (.....)
 นักศึกษา
 / /

ลงชื่อ.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น (ห้ามไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



..... / /

แบบรายงานแผนดำเนินการขยายเวลาการศึกษาเพื่อขอสอบวิทยานิพนธ์ (สำหรับนักศึกษาคณะวิศวกรรมศาสตร์)

ชื่อ - นามสกุล..... รหัสประจำตัว.....

หลักสูตร วศ.ม ไฟฟ้า วศ.ม เครื่องกล วศ.ม ปีโตรเคมี วศ.ด ไฟฟ้า

ชื่ออาจารย์ที่ปรึกษา.....

ได้ขอขยายเวลาการศึกษาเป็นครั้งที่..... ภาคการศึกษาที่..... ปีการศึกษา.....

เหตุผลที่ขอขยายเวลา

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

แผนดำเนินการเพื่อขอสอบวิทยานิพนธ์ (ให้ระบุรายละเอียดและแสดงช่วงเวลาอย่างชัดเจน)

รายละเอียดของงาน	เดือน	เดือน	เดือน	เดือน	เดือน	เดือน
1. ดำเนินการวิจัย
2. รวบรวมและวิเคราะห์ข้อมูลงานวิจัย
3. จัดทำวิทยานิพนธ์
.....
.....
การขอสอบวิทยานิพนธ์

รายละเอียดของงาน(ตัวอย่าง)	มิ.ย.	ก.ค.	ส.ค.	ก.ย.
1. ดำเนินการวิจัย	←→					
2. รวบรวมและวิเคราะห์ข้อมูลงานวิจัย	←	→				
3. จัดทำวิทยานิพนธ์		←	→			
.....						
การขอสอบวิทยานิพนธ์				←→		

ลงชื่อ..... เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
(.....) (.....)
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
นักศึกษา อาจารย์ที่ปรึกษา



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มคำร้องขอลงทะเบียนวิชา (Audits)

วันที่.....เดือน.....พ.ศ.....

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า (นาย,นาง,นางสาว,อื่นๆ).....

รหัสประจำตัว..... หลักสูตร..... แผน..... สาขาวิชา.....

ภาควิชา.....แขนงวิชา.....คณะ.....

มีความประสงค์ขอลงทะเบียนรายวิชาที่เปิดสอนในระดับบัณฑิตศึกษา เป็นวิชา Audits ดังมีรายวิชาต่อไปนี้ โดยบันทึกผล
การเรียนเป็น S หรือ U ตามประกาศของบัณฑิตวิทยาลัย

รหัสวิชา

ชื่อวิชา

.....
.....
.....

จึงเรียนมาเพื่อโปรดพิจารณา

ขอแสดงความนับถือ

ลงชื่อ.....
(.....)

(ผู้ยื่นคำร้อง)

ที่อยู่ติดต่อได้.....

โทรศัพท์(บ้าน).....โทรศัพท์(ที่ทำงาน).....มือถือ.....

อีเมล.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(1) ความเห็นของอาจารย์ที่ปรึกษา

เห็นควรให้ น.ศ. ลงทะเบียนในวิชาดังกล่าวเป็นวิชา Audits

ไม่เห็นควรให้ น.ศ. ลงทะเบียนในวิชาดังกล่าว เป็นวิชา Audits เนื่องจาก.....

(3) ความเห็นของเจ้าหน้าที่บัณฑิตวิทยาลัย

เห็นควรพิจารณาอนุมัติได้ตามระเบียบ

ไม่เห็นควรพิจารณาอนุมัติ เนื่องจาก.....

ลงชื่อ.....

(.....)

...../...../.....

ลงชื่อ.....

(.....)

...../...../.....

(2) ความเห็นของอาจารย์ผู้สอน

เห็นควรให้ น.ศ. ลงทะเบียนในวิชาดังกล่าว เป็นวิชา Audits

ไม่เห็นควรให้ น.ศ. ลงทะเบียนในวิชาดังกล่าว เป็นวิชา Audits เนื่องจาก.....

(4) คำสั่ง (คณบดีบัณฑิตวิทยาลัย)

อนุมัติตามที่เสนอ

ไม่อนุมัติ เนื่องจาก.....

ลงชื่อ.....

(.....)

...../...../.....

ลงชื่อ.....

(.....)

...../...../.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มคำร้องขอถอนวิชาเรียน (เกินกำหนด)

วันที่.....เดือน.....พ.ศ.....

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า (นาย,นาง,นางสาว,อื่นๆ).....

รหัสประจำตัว..... หลักสูตร..... แผน..... สาขาวิชา.....

ภาควิชา.....แขนงวิชา.....คณะ.....

มีความประสงค์ขอถอนวิชาเรียนเกินกำหนด เป็นกรณีพิเศษ ในภาคเรียนที่..... ปีการศึกษา.....
เนื่องจาก.....

ดังมีรายวิชาต่อไปนี้

รหัสวิชา

ชื่อวิชา

.....
.....
.....

จึงเรียนมาเพื่อโปรดพิจารณา

ขอแสดงความนับถือ

ลงชื่อ.....

(.....)

(ผู้ยื่นคำร้อง)

ที่อยู่ติดต่อได้.....

โทรศัพท์(บ้าน)..... โทรศัพท์(ที่ทำงาน)..... มือถือ.....

อีเมล.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(1) ความเห็นของอาจารย์ที่ปรึกษา

- เห็นควรให้ น.ศ. ถอนวิชาเรียนดังกล่าวได้
- ไม่เห็นควรให้ น.ศ. ถอนวิชาเรียนดังกล่าว เนื่องจาก.....

.....

.....

.....

.....

.....

(3) คำสั่ง (คณบดีบัณฑิตวิทยาลัย)

- อนุมัติตามที่เสนอ
- ไม่อนุมัติ เนื่องจาก.....

.....

.....

.....

.....

.....

ลงชื่อ.....
 (.....)
 / /

ลงชื่อ.....
 (.....)
 / /

(2) ความเห็นของเจ้าหน้าที่บัณฑิตวิทยาลัย

- เห็นควรพิจารณาอนุมัติได้ตามระเบียบ
- ไม่เห็นควรพิจารณาอนุมัติ เนื่องจาก.....

.....

.....

.....

.....

.....

ลงชื่อ.....
 (.....)
 / /



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
แบบฟอร์มขอถอนเงินประกันของเสียหาย

สถาบันจะคืนเงินค่าประกันทรัพย์สินเสียหายให้นักศึกษา เมื่อพ้นสภาพการเป็นนักศึกษาแล้ว เต็มจำนวนคงเหลือ เมื่อได้หักหนี้สินที่ต้องชำระแล้ว หากนักศึกษาไม่ขอรับคืนภายใน 6 เดือน หลังจากพ้นสภาพการเป็นนักศึกษาให้ถือว่าค่าประกันทรัพย์สินเสียหายนี้เป็นสมบัติของสถาบันฯ

หลักฐานที่ต้องใช้

ใบเสร็จรับเงินที่ระบุค่าประกันทรัพย์สินเสียหาย (ใบเสร็จที่ได้รับเมื่อเข้าศึกษาในภาคเรียนแรก)

ข้อปฏิบัติในการกรอกข้อความ

1. กรอกข้อความในแบบฟอร์มที่แนบมาให้ครบด้วยตัวบรรจง ยกเว้นในส่วนที่ได้รับเงินค่าประกันทรัพย์สินเสียหายคืนจากสถาบัน
2. ให้นักศึกษากรอกที่อยู่ตามแบบฟอร์มด้านล่างนี้ให้ชัดเจน

ชื่อ.....นามสกุล.....รหัสประจำตัว.....
 ที่อยู่ติดต่อได้.....
 โทรศัพท์(บ้าน).....โทรศัพท์(ที่ทำงาน).....มือถือ.....
 อีเมลล์.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สถาบันเทคโนโลยีพระจอมเกล้า-
เจ้าคุณทหารลาดกระบัง กรุงเทพฯ

วันที่.....เดือน.....พ.ศ. 25.....

เรื่อง ขอดอนเงินค่าประกันของเสียหาย

เรียน คณบดีบัณฑิตวิทยาลัย

ข้าพเจ้า(ชื่อ-สกุล).....รหัสประจำตัว.....นักศึกษาชั้นปีที่.....
หลักสูตร..... สาขาวิชา.....คณะ..... เข้าเรียนเมื่อปี
การศึกษา 25..... ได้ชำระเงินค่าประกันของเสียหายไว้เป็นเงิน.....บาท
(.....)

บัดนี้ข้าพเจ้าได้ ขอลาออก พ้นสภาพการเป็นนักศึกษา
 สำเร็จการศึกษา ปรับสภาพนักศึกษา

เมื่อวันที่..... เดือน.....พ.ศ. 25..... จึงใคร่ขอดอนเงินค่าประกันของเสียหาย จำนวนดังกล่าวคืนตามระเบียบ
จึงเรียนมาเพื่อโปรดพิจารณาอนุมัติ

ขอแสดงความนับถือ

ลงชื่อ.....
(.....)

ได้ตรวจสอบทะเบียนนักศึกษาแล้ว (นาย, นาง, นางสาว).....
ได้(ขอลาออก, พ้นสภาพการเป็นนักศึกษา, สำเร็จการศึกษา, ปรับสภาพนักศึกษา) แล้ว เมื่อวันที่.....เดือน.....
พ.ศ. 25.....

ลงชื่อ.....
(.....)

นายทะเบียน

...../...../.....
เสนอคณบดีคณะฯ พิจารณาอนุมัติ

ลงชื่อ.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่ควรเผยแพร่ให้ผู้อื่นได้โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึง คณบดี บัณฑิตวิทยาลัย ที่มีการนำไปใช้

...../...../.....

สถาบันเทคโนโลยีพระจอมเกล้า-
เจ้าคุณทหารลาดกระบัง กรุงเทพฯ

วันที่..... เดือน..... พ.ศ. 25....

เรื่อง ขอดอนเงินค่าประกันของเสียหายคืน

เรียน อธิการบดี

ข้าพเจ้า.....นามสกุล.....รหัสประจำตัว.....

หลักสูตร.....สาขาวิชา.....คณะ.....

มีความประสงค์จะขอเงินค่าประกันของเสียหายคืน เป็นเงินจำนวน.....บาท (.....)

บัดนี้ข้าพเจ้าได้ ขอลาออก พ้นสภาพกรเป็นนักศึกษา

สำเร็จการศึกษา ปรับสภาพนักศึกษา

จึงเรียนมาเพื่อโปรดพิจารณาอนุมัติ

ขอแสดงความนับถือ

ลงชื่อ.....

(.....)

อนุมัติให้จ่ายได้

ลงชื่อ.....

(.....)

คนบดี้คณะ.....

...../...../.....

ข้าพเจ้าได้รับเงินค่าประกันคืนจากสถาบันฯ เป็นจำนวนเงิน.....บาท (.....)

ตั้งแต่วันที่..... เดือน..... พ.ศ. 25.....

จึงได้ลงลายมือชื่อไว้เป็นหลักฐาน

ลงชื่อ.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าขอ ผู้รับเงิน ทุกครั้งที่มีการนำไปใช้

สถาบันเทคโนโลยีพระจอมเกล้า-
เจ้าคุณทหารลาดกระบัง กรุงเทพฯ

วันที่..... เดือน..... พ.ศ. 25....

เรื่อง ขอดอนเงินค่าประกันของเสียหายคืน

เรียน อธิการบดี

ข้าพเจ้า.....นามสกุล.....รหัสประจำตัว.....

หลักสูตร.....สาขาวิชา.....คณะ.....

มีความประสงค์จะขอเงินค่าประกันของเสียหายคืน เป็นเงินจำนวน.....บาท (.....)

บัดนี้ข้าพเจ้าได้ ขอลาออก พ้นสภาพการเป็นนักศึกษา

สำเร็จการศึกษา ปรับสภาพนักศึกษา

จึงเรียนมาเพื่อโปรดพิจารณาอนุมัติ

ขอแสดงความนับถือ

ลงชื่อ.....
(.....)

อนุมัติให้จ่ายได้

ลงชื่อ.....
(.....)

คนบดี้คณะ.....
...../...../.....

ข้าพเจ้าได้รับเงินค่าประกันคืนจากสถาบันฯ เป็นจำนวนเงิน.....บาท (.....)

ตั้งแต่วันที่..... เดือน..... พ.ศ. 25.....

จึงได้ลงลายมือชื่อไว้เป็นหลักฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้ (เดิมถึงแล้วขอเอกสารทุกครั้งที่มีคา) นำไปใช้

บรรณานุกรม

- [1] Martin Fowler with Kendall Scott, *UML Distilled*, Second Edition: A brief guide to the standard Object Modeling Language, Addison - Wesley Longman, 2000
- [2] Ali Bahrami, Boeing Applied Research & Technology, *Object Oriented Systems Development*, International Edition 1999, Irwin McGraw-Hill, 1976
- [3] Ivar Jacobson, Grady Booch, James Rumbaugh, *The Unified Software Development Process* Rational Software Corporation, Addison - Wesley Longman, 1998
- [4] Terry Quatrani, *Visual Modeling with Rational Rose 2000 and UML*, Addison-Wesley, Pearson Education Asia Pte Ltd., 1998
- [5] อาจารย์อภิเนตร อุนากุล, *Object-Oriented Analysis and Design*, แผนกตำรา คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหาร ลาดกระบัง, 1999
- [6] Marty Hall, *Core Servlet and Java Server Page*, Prentice Hall PTR Upper Saddle River, 2000
- [7] ดร. วีระศักดิ์ ซึ่งถาวร, *Fundamental of JAVA Programming Volume 1 and Volume 2*, SUM Publishing Department, SUM System Company Limited, 1998
- [8] Dale R. Nilsson, Peter M. Jakab, Bill Sarantakos, Russell A. Stinehour, *Enterprise Development with VisualAge for Java, Version 3*, Wiley Computer Publishing, John Wiley & Sons. Inc., 2000
- [9] Osamu Takagiwa, Frederik Haesbrouck, Veronique Quilblier, Sarah Poger, *Programming with VisualAge for Java, Version 3.5*, Third Edition, Prentice Hall PTR, Upper Saddle River, New Jersey 07458, April 2001
- [10] Eric J. Naiburg, Robert A. Maksimchuk, *UML For Database Design*, Addison - Wesley, 2001
- [11] Akmal B. Chaudhri, Roberto Zicari, *Succeeding with Object Databases*, Wiley Computer Publishing, John Wiley & Sons. Inc., 2001
- [12] กิตติ ภัคดีวัฒน์กุล, *Java ฉบับโปรแกรมเมอร์*, KTP COMP & CONSULT, 2001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้