

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การพัฒนาระบบงานบัณฑิตศึกษาและวิชาการ  
Development Academic and Education System



โดย

นางสาว บุษบง ยังเจริญ

รหัส 42067161



\*H001846\*

อาจารย์ที่ปรึกษา

ดร. จันทร์บุรณ สติตวิริยวงศ์

วัน เดือน ปี.....	15 ส.ค. 2550
เลขทะเบียน.....	01846
เลขเรียกหนังสือ.....	997 บ 674 ถ 2544
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

10/08/50

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน  
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ  
ภาคเรียนที่ 2 ปีการศึกษา 2544  
คณะเทคโนโลยีสารสนเทศ  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การพัฒนาระบบบัณฑิตศึกษาและวิชาการ
นักศึกษา	นางสาว นุชบง ยังเจริญ
อาจารย์ที่ปรึกษา	ดร. จันทรบุรณ์ สถิตวิริยวงศ์
ระดับการศึกษา	วิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2544

### บทคัดย่อ

ระบบงานบัณฑิตศึกษาและวิชาการเป็นโครงการที่ประกอบด้วยระบบงานย่อยดังนี้คือ ระบบงานทะเบียนผู้สมัครสอบ ระบบงานลงทะเบียน ระบบงานหลักสูตรการเรียนการสอน ระบบงานตรวจสอบผลการเรียนของนักศึกษาและระบบงานทะเบียนผู้สำเร็จการศึกษา โดยมีวัตถุประสงค์เพื่ออำนวยความสะดวกในการดำเนินงานของเจ้าหน้าที่และนักศึกษาโครงการนี้ได้ถูกพัฒนาขึ้นโดยใช้ Developer 2000 เป็นเครื่องมือในการพัฒนาระบบและใช้ ORACLE 8i เป็นระบบจัดการฐานข้อมูล

<b>Title</b>	Development Academic and Education system
<b>Student</b>	Ms.Boodsabong Youngcharoen
<b>Advisor</b>	Dr. Chanboon Sathitwiriawong
<b>Level Of Study</b>	Master of Science in Information Technology
<b>Major</b>	Informaton Science
<b>Academic Year</b>	2001

## ABSTRACT

Graduation and Academic system is composed of the curriculum management, apply, register, class schedule administration for instructor and student in each semester and through roster of graduate management. Thereby, propose guideline to analysis and design by UML(Unified modeling Language and convenien users in management their jobs. Developer 2000 and oracle database have used to development.

## กิตติกรรมประกาศ

ในการทำโครงการวิจัยครั้งนี้ผู้วิจัยขอขอบคุณอาจารย์ที่ปรึกษาคร.จันทร์บูรณ์ สถิตวิริยวงศ์ ที่ให้คำปรึกษา แนะนำ และข้อเสนอแนะต่างๆที่ช่วยในการทำโครงการวิจัย ทำให้โครงการสำเร็จลุล่วงไปได้ด้วยดี และขอบคุณเพื่อนๆชาว IS8.2 ทุกคนที่ส่งกำลังใจคอยช่วยเหลือมาตลอด

ผู้ดำเนินโครงการ



## สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญภาพ	V
สารบัญตาราง	VI
บทที่	
1. บทนำ	
1.1 ความเป็นมา	1
1.2 ปัญหาที่พบในปัจจุบัน	2
1.3 ขอบเขตของการพัฒนาระบบงาน	2-4
1.4 วัตถุประสงค์ของการพัฒนาระบบงาน	4
1.5 ประโยชน์ที่ได้รับจากการพัฒนาระบบ	5
2. ทฤษฎีที่เกี่ยวข้อง	
2.1 วงจรการพัฒนาระบบงานและโครงการ	6-22
3 ขั้นตอนในการดำเนินงาน	
3.1 การวิเคราะห์ระบบงาน	23-29
4 การออกแบบฐานข้อมูล	
4.1 การออกแบบฐานข้อมูล	30-34
5 การพัฒนาระบบ	
5.1 การพัฒนาโปรแกรม	35-59
6 สรุปและข้อเสนอแนะ	
5.1 ผลการดำเนินงาน	60
5.2 ปัญหาและอุปสรรค	60
5.3 ข้อเสนอแนะ	61
5.4 แนวทางการศึกษา	61
บรรณานุกรม	62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญภาพ

ภาพที่	หน้า
2.1 use-case Daigram	12
2.2 แสดงสัญลักษณ์ของ Class Diagram ทะเบียนลูกค้า	13
2.3 แสดงตัวอย่างของ Class Diagram สำหรับระบบงานจัดซื้อ	14
2.4 แสดงตัวอย่างของ Object Diagram	14
2.5 แสดงตัวอย่างของ sequence Diagram	15
2.6 แสดงตัวอย่างของ Collaborate Daigram	16
2.7 แสดงตัวอย่าง State Diagram	16
2.8 แสดงตัวอย่างของ Activity Diagram	17
2.9 แสดงตัวอย่างของ Component Diagram	18
2.10 แสดงตัวอย่าง Deployment Diagram	19
2.11 แสดงระบบ Client /Server	21
2.12 แสดงสถาปัตยกรรม Client /Server แบบ 3 -TIRE	22
3.1 แสดง use-case Diagram สำหรับระบบบัณฑิตศึกษาและวิชาการ	23
3.2 แสดง class Diagam สำหรับระบบบัณฑิตศึกษาและวิชาการ	24
3.3 แสดง activity diagram สำหรับระบบทะเบียนบัณฑิตศึกษา	25
3.4 แสดง Activity Diagram สำหรับตรวจสอบสิทธิการสอบ	26
Comprehensive และ Oral	
3.5 แสดง activity Diagram สำหรับการลงทะเบียนเรียนของนักศึกษา	27
3.6 แสดง sequence Diagram สำหรับตรวจสอบสิทธิการสอบ	28
Comprehensive และ Oral	
3.7 แสดง sequence Diagram สำหรับการลงทะเบียนเรียนของนักศึกษา	29
5.1 หน้าจอแสดงเมนูการใช้งานระบบ	35
5.2 หน้าจอสำหรับแสดงข้อมูลหลักสูตรการเรียน	36
5.3 หน้าจอสำหรับแสดงข้อมูลแขนงวิชา	37

## สารบัญภาพ

ภาพที่	หน้า
5.4 หน้าจอสำหรับแสดงข้อมูลแผนวิทยานิพนธ์	37
5.5 หน้าจอสำหรับกำหนดเงื่อนไขการสอบวิชา Comprehensive	38
5.6 หน้าจอสำหรับแสดงข้อมูลวิชาเรียน	38
5.7 หน้าจอสำหรับแสดงข้อมูลโครงสร้างหลักสูตรการเรียน	40
5.8 หน้าจอสำหรับแสดงข้อมูลหลักสูตรสำหรับการลงทะเบียนเรียน	41
5.9 หน้าจอสำหรับแสดงข้อมูลการลงทะเบียนเรียนของนักศึกษา	42
5.10 หน้าจอแสดงข้อมูลผลการเรียนของนักศึกษาเป็นรายวิชา	43
5.11 หน้าจอแสดงข้อมูลผลการเรียนของนักศึกษาเป็นรายภาคการศึกษา	44
5.12 หน้าจอแสดงข้อมูลผลการเรียนของนักศึกษาเป็นแยกตามรุ่น	46
5.13 หน้าจอแสดงข้อมูลประวัติการลงทะเบียนเรียนของนักศึกษา	48
5.14 หน้าจอแสดงข้อมูลทะเบียนนักศึกษา	49
5.15 หน้าจอแสดงข้อมูลที่อยู่นักศึกษา	50
5.16 หน้าจอแสดงข้อมูลเบอร์โทรศัพท์นักศึกษา	50
5.17 หน้าจอแสดงคำแนะนำหัวข้อ	51
5.18 หน้าจอแสดงข้อมูลประเภทของนักศึกษา	51
5.19 หน้าจอสำหรับพิมพ์ Registration Form	52
5.20 หน้าจอสำหรับพิมพ์รายงานผลการเรียนของนักศึกษาแยกตามรายวิชา	53
5.21 หน้าจอสำหรับพิมพ์รายงานผลการเรียนของนักศึกษาแยกตามรหัส นักศึกษา	54
5.22 หน้าจอสำหรับพิมพ์รายงานเกรดเฉลี่ยของนักศึกษาแยกตามห้องเรียน	55
5.23 หน้าจอสำหรับพิมพ์รายชื่อนักศึกษาที่มีสิทธิสอบ Comprehensive และ Oral	56
5.24 หน้าจอสำหรับพิมพ์รายชื่อผู้สำเร็จการศึกษา	57
5.25 หน้าจอสำหรับพิมพ์รายชื่อนักศึกษา	58
5.26 หน้าจอสำหรับพิมพ์ผลการเรียนของนักศึกษา(Transcript)	59

## สารบัญตาราง

ตารางที่	หน้า
4.1 ทะเบียนหลักสูตร	30
4.2 ทะเบียนแขนงวิชา	30
4.3 โครงสร้างหลักสูตรการเรียน	30
4.4 หลักสูตรการเรียนรายวิชา	31
4.5 หลักสูตรแผนการเรียน	31
4.6 ทะเบียนวิชา	31
4.7 ทะเบียนวิชา Prerequisite	32
4.8 ทะเบียนการลงทะเบียนรายวิชา	32
4.9 ทะเบียนการลงทะเบียนวิชาพิเศษ	32
4.10 ประวัติการลงทะเบียนของนักศึกษา	32
4.11 ทะเบียนค่าน้ำหนัก	33
4.12 ทะเบียนประเภทของนักศึกษา	33
4.13 ทะเบียนนักศึกษา	33
4.14 ทะเบียนที่อยู่นักศึกษา	34
4.15 ทะเบียนเบอร์โทรศัพท์นักศึกษา	34
4.16 ทะเบียนหลักสูตรการสอบ comprehensive	34

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมา

ในปัจจุบันระบบสารสนเทศได้เข้ามามีบทบาทอย่างมากภายในชีวิตประจำวันของเรา ระบบงานบัณฑิตศึกษาและวิชาการ ( Graduation and Academic ) หรือเรียกอีกชื่อหนึ่งว่าระบบสารสนเทศนักศึกษา ( Student Information System ) ก็เป็นอีกตัวอย่างหนึ่งของระบบสารสนเทศที่ได้ถูกพัฒนาขึ้นเพื่อใช้เป็นระบบสารสนเทศสำหรับนักศึกษาคณะเทคโนโลยีสารสนเทศ โดยมีวัตถุประสงค์หลักเพื่อมุ่งให้บริการแก่นักศึกษาทั้งในด้านการบริการการลงทะเบียนเรียน การตรวจสอบผลการเรียน รวมทั้งให้บริการข้อมูลเกี่ยวกับหลักสูตรการเรียนการสอนที่เปิดทำการสอนในแต่ละภาคการศึกษา นอกจากนี้จะให้บริการแก่นักศึกษาแล้วระบบสารสนเทศนี้ยังช่วยอำนวยความสะดวกแก่เจ้าหน้าที่ ในด้านการจัดการ งานทะเบียนผู้สมัครสอบ งานทะเบียนประวัตินักศึกษา การประมวลผลการเรียนของนักศึกษาและตรวจสอบสถานะภาพของนักศึกษา งานทะเบียนผู้สำเร็จการศึกษาและงานจัดทำหลักสูตรการเรียนการสอน ซึ่งนับเป็นประโยชน์อย่างหนึ่งของการนำเอาระบบสารสนเทศมาใช้ช่วยเพิ่มประสิทธิภาพในการทำงาน และทำให้การดำเนินกิจการต่างๆเป็นไปอย่างสะดวก รวดเร็วยิ่งขึ้น

เนื้อหาของงานวิจัยฉบับนี้จึงเป็นการนำเสนอแนวทางในการออกแบบและพัฒนาระบบบัณฑิตศึกษาและวิชาการสำหรับคณะเทคโนโลยีสารสนเทศ โดยใช้หลักการออกแบบและพัฒนาระบบในรูปแบบของ Object-Oriented Analysis and design ซึ่งการออกแบบระบบจะอาศัยมุมมองของ Problem domain และ solution ในลักษณะของ Object เป็นสำคัญ โดยใช้ UML ( Unified Modeling Language ) เป็น Methodology ในการจำลองรูปแบบของระบบงานเชิงวัตถุ เนื่องจาก UML นั้นเป็นการวิเคราะห์ระบบงานที่สนับสนุนการพัฒนาโปรแกรมด้วยเทคโนโลยีแบบ Object จึงทำให้สามารถนำมาปรับเป็นการโปรแกรมแบบ Object ได้อย่างมีประสิทธิภาพ และยังสามารถแยกการพัฒนา Module ต่างๆเป็นระบบงานย่อยๆได้

ระบบงานบัณฑิตศึกษาและวิชาการแบ่งเป็นระบบงานหลักๆได้ 6 ระบบงานดังนี้คือ

- ระบบงานทะเบียนผู้สมัครสอบ
- ระบบงานทะเบียนนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ระบบงานลงทะเบียน
- ระบบงานประมวลผลการศึกษา
- ระบบงานหลักสูตรการเรียนการสอน
- ระบบงานทะเบียนผู้สำเร็จการศึกษา

## 1.2 ปัญหาที่พบในงานปัจจุบัน

ระบบงานบัณฑิตศึกษาและวิชาการสำหรับคณะเทคโนโลยีสารสนเทศเป็นระบบงานที่ได้ถูกพัฒนามาแล้วครั้งหนึ่งภายใต้ชื่อโครงการ ISDoIT ซึ่งเป็นโครงการที่จัดทำขึ้นโดยความร่วมมือของนักศึกษาและคณาจารย์ คณะเทคโนโลยีสารสนเทศ เพื่อช่วยอำนวยความสะดวกแก่เจ้าหน้าที่ในการทำงานและการจัดการกับเอกสารต่างๆ ที่มีลักษณะการจัดเก็บเป็นแฟ้มเอกสารหรือบางส่วนที่มีการจัดเก็บเป็นไฟล์ ให้อยู่ในลักษณะของฐานข้อมูลที่พร้อมใช้งานอยู่เสมอและสะดวกต่อการสืบค้นข้อมูล และใช้เทคโนโลยีของระบบอินเทอร์เน็ตมาประยุกต์ใช้กับการทำงาน เพื่อให้ผู้ใช้สามารถเรียกใช้งาน application ผ่านทางเว็บเบราว์เซอร์ โดยใช้ภาษา Java ในพัฒนาระบบงาน แต่เนื่องจากโครงการดังกล่าวมีการพัฒนาเป็น ระบบงานที่แยกย่อยมากจนเกินไปประกอบกับการประสานการทำงานได้ไม่ดีพอระหว่างกลุ่มนักศึกษาด้วยกันเอง ทำให้การส่งข้อมูลระหว่างระบบงานต่างๆ กระทำได้ไม่ดีจนเป็นเหตุ ให้ไม่สามารถใช้ระบบงานตาม โครงการดังกล่าว ได้เต็มประสิทธิภาพเท่าที่ควร ประกอบกับการใช้ technology ของ java ที่เป็นภาษาค่อนข้างใหม่และผู้พัฒนาระบบเองจะต้องใช้เวลาในการศึกษามากค่อนข้างมาก ทำให้มีเวลาให้การออกแบบค่อนข้างน้อย จึงทำให้ขาด information ที่สำคัญบางอย่างไป

## 1.3 ขอบเขตของการพัฒนาระบบงาน

ระบบงานบัณฑิตศึกษาและวิชาการประกอบด้วยระบบงาน 5 ย่อยดังนี้คือ

### ระบบงานทะเบียนนักศึกษา

ระบบงานทะเบียนนักศึกษาหรือการขึ้นทะเบียนนักศึกษา เป็นระบบงานที่มีความเกี่ยวข้องกับการเก็บทะเบียนประวัตินักศึกษาซึ่งประกอบด้วย ข้อมูลส่วนตัว ประวัติการศึกษา และสถานภาพของนักศึกษา เมื่อผู้สมัครสอบผ่านการทดสอบและได้รับคัดเลือกเข้าเป็นนักศึกษาของทางคณะแล้ว ทางคณะก็จะทำการกำหนดรหัสนักศึกษา กำหนดแขนงวิชา ภาควิชาให้กับนักศึกษา

## ระบบงานลงทะเบียน

ระบบงานลงทะเบียนเป็นกิจกรรมที่เกิดขึ้นเป็นประจำทุกภาคการศึกษา เพื่อให้ทราบว่านักศึกษาได้ลงทะเบียนวิชาอะไรบ้างและข้อมูลรายวิชาที่นักศึกษาลงทะเบียน เพื่อจะได้นำไปใช้ในการจัดห้องสอบและประมวลผลการศึกษา แต่ทั้งนี้และทั้งนั้นจำเป็นอย่างยิ่งที่จะต้องเชื่อมโยงกับระบบงานหลักสูตรการเรียนการสอน นอกจากนี้อาจจะมีการเพิ่มฟังก์ชันการตรวจสอบสถานะภาพของนักศึกษาก่อนการลงทะเบียนและการตรวจสอบการสอบ Comprehensive และ Oral ของนักศึกษาด้วย

ระบบงานลงทะเบียนจะมีกิจกรรมหลักๆ ที่เกี่ยวข้อง 2 กิจกรรมคือ

- 1) การลงทะเบียน นักศึกษาจะต้องแจ้งความจำนงโดยยื่นบัตรนักศึกษา ซึ่งจะใช้แถบแม่เหล็กหรือรหัสแท่ง (บาร์โค้ด) เพื่อสอบถามรายวิชาที่ลงทะเบียนและจำนวนเงินที่ต้องชำระ จากนั้นทางเจ้าหน้าที่ก็จะจัดพิมพ์ใบนำฝากเงินให้แก่นักศึกษา
- 2) การชำระเงิน นักศึกษาชำระเงินผ่านเครือข่ายขององค์กรที่บริการด้านการรับฝากเงิน เช่น ธนาคาร วิธีการนี้นักศึกษาจะต้องนำไปยื่นรับการลงทะเบียนซึ่งเป็นใบนำฝากเงินที่ได้ตกลงรูปแบบกับธนาคารใด ๆ ที่มีจำนวนเงินตามที่ต้องชำระพร้อมเลขประจำตัวนักศึกษายื่นต่อธนาคาร แล้วธนาคารนำฝากเข้าบัญชีของสถานศึกษาและลงนามว่าได้รับเงินเรียบร้อยแล้ว

สำหรับระบบงานบัณฑิตศึกษาและวิชาการเราจะตัด Information เกี่ยวกับการชำระเงินออกไป

## ระบบงานประมวลผลการศึกษา

กิจกรรมหลักของระบบงานประมวลผลการศึกษาคือ รับผลการศึกษารายวิชาของนักศึกษาจากอาจารย์ประจำวิชา พร้อมกับดึงข้อมูลการลงทะเบียนของนักศึกษาในแต่ละภาคการศึกษามาคำนวณเกรดเฉลี่ยรายเทอมและเกรดเฉลี่ยสะสม เก็บเป็นประวัติผลการศึกษาของนักศึกษาในแต่ละภาคการศึกษา พร้อมกับส่งข้อมูลผลการศึกษา(ใบเกรด) กลับไปให้นักศึกษาและอาจารย์ที่ปรึกษา และทำการปรับปรุงข้อมูลสถานะภาพของนักศึกษาไปพร้อมกันด้วย นอกจากนี้ยังเพิ่มฟังก์ชันการตรวจเช็คผลการเรียนให้กับนักศึกษาด้วย

## ระบบงานหลักสูตรการเรียนการสอน

ระบบงานหลักสูตรการเรียน เป็นระบบงานที่ดำเนินกิจกรรมเกี่ยวกับการจัดเก็บข้อมูลเกี่ยวกับหลักสูตรการเรียน โดยเปิดทำการสอนเพียงหลักสูตรเดียวคือ หลักสูตรวิทยาศาสตรมหาบัณฑิต เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนักศึกษาเห็นใบแจ้งเรียนต้นการค่าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สาขาเทคโนโลยีสารสนเทศ แต่แยกออกเป็น 2 แขนงวิชาด้วยกันคือ แขนงวิชาวิทยาการสารสนเทศ(IS) และแขนงวิชาการจัดการสารสนเทศ(ITM) และภายในแต่ละแขนงวิชานักศึกษาสามารถเลือกลงแผน ก. คือแผนที่มีการทำวิทยานิพนธ์ หรือแผน ข. คือแผนที่ไม่มีวิทยานิพนธ์ ระบบงานหลักสูตรการเรียนจะมีความเกี่ยวข้องกับการกำหนดรายละเอียดของรายวิชาที่เปิดการเรียนการสอน ระบบดังกล่าวนี้ ได้ถูกพัฒนาขึ้นเพื่อนำไปใช้เชื่อมโยงกับระบบลงทะเบียนในขั้นต่อไป

### ระบบงานทะเบียนผู้สำเร็จการศึกษา

ระบบงานทะเบียนผู้สำเร็จการศึกษา เป็นระบบงานที่ดำเนินกิจกรรมเกี่ยวกับการตรวจสอบสถานะภาพของนักศึกษาและผลการเรียนของนักศึกษา การตรวจสอบการผ่านวิชา LAB การตรวจสอบการผ่านวิชา Comprehensive และ Oral พร้อมทั้งทำการปรับปรุงทะเบียนประวัติของนักศึกษา การจัดทำทะเบียนประวัติผู้สำเร็จการศึกษาสำหรับผู้ผ่านหลักสูตรการเรียนการสอนครบถ้วนตามข้อตกลงของทางคณะ

### 1.4 วัตถุประสงค์ของการพัฒนาระบบงาน

- ระบบงานบัณฑิตศึกษาและวิชาการเป็นระบบงานที่ถูกพัฒนาขึ้นหลังจากโครงการ ISDoIT ทั้งนี้และทั้งนั้นก็สืบเนื่องมาจากระบบงานเดิมยังไม่สามารถตอบสนองต่อความต้องการของผู้ใช้งานได้อย่างเต็มประสิทธิภาพ และเพื่อให้เป็นระบบงานดังกล่าวสามารถเชื่อมโยงกัน ได้กับส่วนงานอื่นๆภายในองค์กร จึงจำเป็นอย่างยิ่งที่จะต้องมีการวิเคราะห์และออกแบบระบบสารสนเทศแบบใหม่ขึ้น
- เนื่องจาก Developer 2000 เป็นภาษาที่ง่ายต่อการพัฒนาเพราะเป็นภาษาที่สามารถเชื่อมโยงกับ Oracle Database ได้โดยตรง การเรียกใช้ข้อมูลก็กระทำได้ง่าย การพัฒนาโปรแกรมจึงทำได้ง่ายกว่าภาษาอื่นๆ
- ศึกษาการออกแบบระบบในลักษณะของ Object-Oriented analysis and design ซึ่งก็เป็นอีกทางเลือกหนึ่งของการออกแบบระบบนอกเหนือไปจาก SDLC(System Development Life Cycle)
- ศึกษาการพัฒนาระบบโดยใช้ Developer2000 ร่วมกับ Oracle 8i ในการพัฒนาระบบ

### 1.5 ประโยชน์ที่ได้รับจากการพัฒนาระบบงาน

- นักศึกษาคณะเทคโนโลยีสารสนเทศจะได้รับข้อมูลข่าวสารที่รวดเร็วขึ้น อาทิเช่น การตรวจสอบผลการเรียนของนักศึกษาที่สามารถทำได้เร็วขึ้น อีกทั้งเจ้าหน้าที่สามารถสืบค้นข้อมูลต่างๆ ได้เร็วขึ้น ซึ่งจะช่วยเพิ่มประสิทธิภาพในการทำงานให้ดีขึ้นได้
- การจัดเก็บข้อมูลของเจ้าหน้าที่ทำได้เป็นระบบมากขึ้น



## บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

### 2.1 วงจรการพัฒนากระบวนการและโครงการ

Methodology ที่ใช้ในการจำลองรูปแบบการทำงานเชิงวัตถุ สำหรับการวิเคราะห์ระบบงานบัณฑิตศึกษาและวิชาการเราใช้ UML ซึ่งเป็นภาษาสำหรับบรรยายแบบจำลองเพื่อใช้สื่อสารกับผู้ใช้ โดยภาษาที่ใช้จะใช้กับคนเฉพาะบางกลุ่ม เช่น นักออกแบบ นักพัฒนาระบบคอมพิวเตอร์

UML ย่อมาจาก Unified Modeling Language เป็นภาษาที่ใช้สำหรับอธิบายโมเดลต่างๆ และเป็นภาษาที่มีลักษณะของ Map Language คือใช้กราฟฟิกเป็นสัญลักษณ์แต่ไม่ได้ใช้สัญลักษณ์เพียงเพื่ออธิบายระบบงานและการสร้างโคแอดแกรมเท่านั้น แต่ UML ยังมีคุณสมบัติของเมต้าโมเดล (metamodel) คือเป็น โมเดลที่เอาไว้เพื่ออธิบาย โมเดลอื่นๆอีกที

#### เป้าหมายของการสร้าง model ในการพัฒนาซอฟต์แวร์

1. โมเดลช่วยให้เราสามารถมองเห็นภาพการทำงานของระบบงาน ได้ชัดเจนขึ้นว่า ระบบงานจะออกมาในลักษณะไหนหรือมองเห็นภาพว่าเราต้องการให้ระบบงานของเราออกมาในทิศทางใด อาทิเช่น ใช้อธิบายลักษณะงานบางอย่างที่ไม่สามารถเข้าใจได้โดยดูแค่โค้ดของโปรแกรมเท่านั้น

2. โมเดลจะทำให้เราสามารถระบุถึงโครงสร้างและพฤติกรรมของระบบงานที่จะพัฒนา เช่น ช่วยให้การสร้างโมเดลมีความเที่ยงตรง ไม่กำกวม และมีความครบถ้วนตามที่ต้องการ ช่วยสร้างภาพของรายละเอียดของระบบงานทั้งในขั้นตอนของการวิเคราะห์และออกแบบระบบ รวมไปถึงขั้นตอนในการนำวิธีการแก้ไขปัญหาที่ได้ไปใช้กับระบบงานจริงด้วย

3. โมเดลเป็นเสมือนเทมเพลตให้แก่การสร้างระบบงานจริงๆ นั่นคือเราสามารถนำ UML มาอธิบายระบบซอฟต์แวร์ได้ไม่ว่าเราจะพัฒนาด้วยภาษาใดก็ตาม เช่น ภาษาจาวา, C++, Visual Basic เป็นต้น เราสามารถนำ UML มาอธิบายได้แม้กระทั่งฐานข้อมูลเชิงสัมพันธ์ (Relational Database) หรือฐานข้อมูลเชิงวัตถุ (Object-Oriented Database) การใช้ UML จะทำให้เราสามารถสร้างโค้ดของโปรแกรมได้ง่ายขึ้น กล่าวคือ เราสามารถนำโมเดลมาแปลงเป็นตัวโค้ดได้เลย โดยอาจจะเพิ่มเติมรายละเอียดบางอย่างของโค้ดเข้าไป ลักษณะการแปลงโมเดลไปเป็นตัวโค้ดโปรแกรมเช่นนี้เราเรียกว่า " Forward Engineering "

4. โมเดลช่วยให้เราสามารถทำให้การตัดสินใจในเรื่องต่างๆ อยู่ในรูปของเอกสารอ้างอิงได้ง่ายขึ้น นั่นคือองค์กรที่ดีเมื่อมีการพัฒนาซอฟต์แวร์แล้ว ควรที่จะมีการทำเอกสารอธิบายประกอบในขั้นตอนต่างๆ ของการพัฒนาซอฟต์แวร์ ด้วยตัวอย่างของเอกสารประกอบ อาทิ

ความต้องการของซอฟต์แวร์

- สถาปัตยกรรมของซอฟต์แวร์
- การออกแบบซอฟต์แวร์
- ซอร์สโค้ดโปรแกรม
- รายละเอียดการวางแผนโครงการ
- การทดสอบระบบงาน
- การทำโปรโตไทป์ (prototype) ของระบบงาน
- การออกแบบเวอร์ชันต่างๆของ ซอฟต์แวร์

ทั้งนี้องค์กรจะทำเอกสารอะไรบ้างนั้นขึ้นอยู่กับวัฒนธรรมการทำเอกสารขององค์กรนั้น บางเอกสารต้องทำอย่างเป็นทางการ บางเอกสารอาจทำไว้เพื่ออ้างอิงเท่านั้น ซึ่งเอกสารประกอบเหล่านี้ นับว่ามีความสำคัญอย่างยิ่งต่อกระบวนการพัฒนาซอฟต์แวร์ทั้งในระหว่างขั้นตอนของการพัฒนา หรือหลังจากนาระบบงานนั้นไปใช้งานแล้วก็ตาม เช่น ช่วยในการควบคุมระบบงาน, ประเมินผลระบบงาน, เชื่อมโยงความสัมพันธ์ของระบบงานต่างๆ เป็นต้น

การใช้งานโมเดลไม่ได้จำกัดเฉพาะกับระบบงานที่มีความซับซ้อนเท่านั้น ระบบงานเล็กๆก็สามารถใช้โมเดลเข้ามาช่วยได้ แต่โดยมากแล้วเรามักไม่ค่อยเป็นการใช้โมเดลมาพัฒนาระบบงานเล็กๆ เพราะระบบงานเล็กมักพัฒนาได้ง่ายและต้องการให้เสร็จสิ้นในระยะเวลาอันสั้น ซึ่งแน่นอนว่าการสร้างโมเดลนั้นเราจะต้องเสียเวลาเพิ่มขึ้นในส่วนของการวิเคราะห์และสร้างโมเดลขึ้นมา

ดังนั้นในระบบงานขนาดเล็กที่ต้องการพัฒนาให้เสร็จสิ้นในระยะเวลาอันสั้นจึงอาจไม่ต้องการโมเดลเข้ามาช่วย เราจึงมักพบเห็นการใช้โมเดลกับระบบงานขนาดใหญ่หรือระบบงานที่มีความซับซ้อนมากๆทั้งนี้ในระบบงานขนาดใหญ่หรือระบบงานที่ซับซ้อนสูงผู้พัฒนาไม่สามารถเข้าใจระบบงานทั้งหมดได้ อีกทั้งความสามารถให้การเข้าใจในระบบงานที่มีความซับซ้อนมากๆ ของคนๆหนึ่งมีขีดจำกัด แต่ถ้าเรามีการนำโมเดลเข้ามาช่วยอย่างเหมาะสมแล้วจะทำให้เรามองระบบได้ง่ายขึ้น เราจะมองเห็นภาพได้อย่างชัดเจนเฉพาะในส่วนที่เรากำลังสนใจอยู่ ณ ขณะนั้น ทำให้การพัฒนาทำได้ง่ายกว่าการมองทั้งระบบด้วยตัวของเราเอง

## หลักของการเลือกใช้ model

หลักการในการใช้โมเดลนั้น มีข้อควรพิจารณาอยู่หลายอย่าง แต่โดยรวมแล้วผู้ที่ใช้โมเดลในการพัฒนาระบบงานจะจัดหลัก 4 ข้อดังนี้คือ

1. โมเดลที่จะนำมาใช้แก้ปัญหาจะต้องเป็น โมเดลที่แก้ปัญหาดังกล่าวได้ดีที่สุด นั่นคือไม่มีโมเดลอื่นทำได้ดีกว่านี้แล้ว
2. โมเดลที่นำมาใช้จะต้องสามารถให้มุมมองที่ตรงของผู้ที่จะใช้โมเดลนั้น เช่นถ้าเป็นนักวิเคราะห์ระบบหรือผู้ใช้ระบบก็จะให้ความสนใจกับมุมมองว่าระบบจะทำอะไรบ้าง หรือถ้าเป็นผู้พัฒนาระบบก็จะต้องการมุมมองว่าระบบจะทำงานต่างๆ ได้อย่างไร เป็นต้น
3. โมเดลที่นำมาใช้จะต้องสามารถนำมาพัฒนาให้เป็นระบบงานที่สามารถใช้งานได้จริง
4. ไม่มีโมเดลใดโมเดลหนึ่งที่สามารถอธิบายระบบได้ทุกมุมมอง การที่จะอธิบายระบบงานให้ได้มุมมองต่างๆ อย่างครบถ้วนจำเป็นจะต้องใช้หลายๆ โมเดลเข้ามาอธิบายร่วมกัน แต่สิ่งหนึ่งที่สำคัญที่สุดสำหรับการพัฒนาระบบงานคือ ความเข้าใจที่ตรงกันของผู้ที่เกี่ยวข้องกับระบบงานนั้นๆ ดังนั้น โมเดลที่เราเลือกมาใช้ควรมีคุณสมบัติที่จะช่วยให้ทุกคนมีความเข้าใจต่อระบบงานได้ตรงกัน เพื่อให้สามารถพัฒนาระบบงานไปได้อย่างรวดเร็วและถูกต้อง

## รูปแบบความสัมพันธ์ในภาษา UML

ความสัมพันธ์ในภาษา UML จะมองภาพความสัมพันธ์ใน 3 ประเด็นคือ ในแง่ของความหมาย , ในแง่ของโครงสร้าง และในแง่ของความสัมพันธ์ทั่วไปหรือความสัมพันธ์แบบพิเศษ

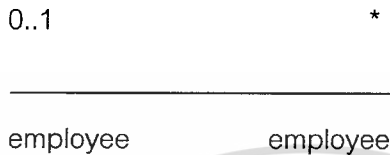
ความสัมพันธ์ในภาษา UML แบ่งเป็น 4 ชนิดคือ

- 1.dependency
- 2.association
- 3.generallization
- 4.Realization

dependency คือความสัมพันธ์กันในเชิงความหมาย กล่าวคือเป็นความสัมพันธ์ระหว่าง 2 สิ่ง โดยสนใจที่ว่าเมื่อสิ่งหนึ่งเกิดการเปลี่ยนแปลงไปแล้วจะส่งผลอย่างไรกับอีกสิ่งหนึ่ง สัญลักษณ์ของ dependency คือลูกศรที่เป็นเส้นประและในบางครั้งอาจมีเลเบลกำกับอยู่ด้วยก็ได้

----->

association คือความสัมพันธ์ในเชิงโครงสร้าง ซึ่งจะอธิบายถึงการเชื่อมโยง ระหว่าง
ออบเจกต์ต่างๆ association จะมีความสัมพันธ์หนึ่งที่เรียกว่า aggregation เอาไว้สำหรับอธิบาย
ความสัมพันธ์ของออบเจกต์ทั้งหมดกับออบเจกต์เพียงบางส่วน การวาดรูปแทน association เราจะ
ใช้เส้นตรงและมีเลเบลกำกับเพื่อบอกลักษณะความสัมพันธ์ดังรูป



generallization คือความสัมพันธ์ในเชิงทั่วไปหรือเป็นความสัมพันธ์แบบพิเศษ กล่าวคือ
เป็นความสัมพันธ์ที่อีลิเมนต์ลูก (child element) มีการรับ โครงสร้างและพฤติกรรมมาจากอีลิเมนต์
แม่(parent element) ในขณะที่เดียวกันอีลิเมนต์ลูกจะมีโครงสร้างและพฤติกรรมบางอย่างเฉพาะตัว
ที่ต่างจากอีลิเมนต์แม่ ซึ่งเราจะเรียกอีลิเมนต์แม่ว่าเป็นอีลิเมนต์ทั่วไป(generalized element) และ
เรียกอีลิเมนต์ลูกว่าเป็นอีลิเมนต์พิเศษ(special element) การวาดรูปแสดง generallization เราจะ
ใช้เครื่องหมายลูกศรที่มีหัวลูกศรเป็นสามเหลี่ยมโปร่ง โดยหัวลูกศรจะชี้ไปยังอีลิเมนต์แม่ดังรูป



Realization เป็นความสัมพันธ์ในเชิงความหมายระหว่าง Interface กับ classes หรือ
ระหว่าง use cases กับ collaboreations ในการวาดรูปเพื่อแสดง realization เราจะใช้สัญลักษณ์
เป็นลูกศร โปร่งคล้ายกับ generallization แต่ใช้ปลายลูกศรเป็นเส้นประ ดังรูป



**การแบ่งประเภทของ UML**

ภาษา UML แบ่งไดอะแกรมออกเป็นทั้งหมด 6 แบบ โดยจะมีบางไดอะแกรมที่มีการแตกย่อย
ลงไปอีก จึงทำให้จำนวนไดอะแกรมทั้งหมดของภาษา UML มีอยู่ 9 ไดอะแกรมดังต่อไปนี้

1. Use Case Diagram<Function Model>
2. Static Structure Diagram
  - Class Diagram
  - Object Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Interaction Diagram
  - Sequence Diagram
  - Collaboration Diagram
4. Statechart Diagram
5. Activity Diagram
6. Implementation Diagram
  - Component Diagram
  - Deployment Diagram

- **Use-Case Diagram**

ตามแนวคิดของไอวาร์ จากอบสัน (Ivar Jacobson) ยูสเคสไดอะแกรมนั้นใช้แสดงการทำงานเด่นๆของซอฟต์แวร์ที่สร้างขึ้นและเป็นจุดเริ่มต้นในการรวบรวมความต้องการของผู้ใช้ในขั้นแรก ซึ่งข้อดีของวิธีการนี้ก็คือสามารถกำหนดขอบเขตของฟังก์ชันงานได้อย่างชัดเจนและสามารถเข้าถึงเป้าหมายที่แตกต่างกันของผู้ใช้ได้

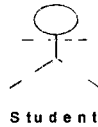
เป็น ไดอะแกรมที่ใช้แสดงความต้องการของระบบทั้งหมดและช่วยในการอธิบายรายละเอียดหลักๆภายในออบเจกต์และแสดงให้เห็นถึงการไหลของข้อมูลในแต่ละการทำงาน โดยจะสนใจเพียงแต่ว่ามีงานอะไรบ้างที่ต้องทำแต่จะไม่สนใจว่างานนั้นทำอย่างไร

ดังนั้น Use-Case Diagram ที่ดีจะต้องสามารถแสดงความต้องการของระบบทั้งหมดในลักษณะที่ผู้ใช้งานเข้าใจได้ง่ายและเป็น Diagram ที่จะช่วยให้นักวิเคราะห์ระบบกับผู้ใช้งานระบบสามารถสื่อสารเข้าใจได้ตรงกันว่าผู้ใช้งานระบบจะนำระบบงานที่เสร็จแล้วไปใช้งานอะไร โดย use-Case จะให้ภาพของการใช้งานอย่างคร่าวๆว่าระบบนั้นผู้ใช้งานจะสามารถนำไปใช้ทำอะไรได้บ้าง

### สัญลักษณ์ที่ใช้ในยูสเคสไดอะแกรมมีดังนี้คือ

**แอกเตอร์ ( Actor )** คือสิ่งที่อยู่ภายนอกระบบที่จะพัฒนาขึ้นทั้งหมด ที่ต้องการแลกเปลี่ยนหรือส่งข้อมูลให้กับระบบ(สร้าง Role)

โดยที่แอกเตอร์อาจจะเป็นคน ระบบหรือ โปรแกรมอื่นก็ได้(ส่วนใหญ่เป็นผู้เริ่มทำงานกับยูสเคส)



**ยูสเคส(Use Case)** เป็นตัวแทนกิจกรรมที่เกิดขึ้นในขั้นตอนต่างๆ โดยจะใช้สัญลักษณ์เป็นรูปวงรี หรือวงกลม



**ซิสเต็ม(System)** เป็นตัวแทนของระบบที่ถูกกระทำโดยแอ็กเตอร์ ซึ่งใช้สัญลักษณ์เป็นรูปสี่เหลี่ยมแทน ภายในประกอบด้วย ยูสเคสต่างๆ และมีเส้นที่แสดงความสัมพันธ์กันระหว่าง Use-case กับ Actor



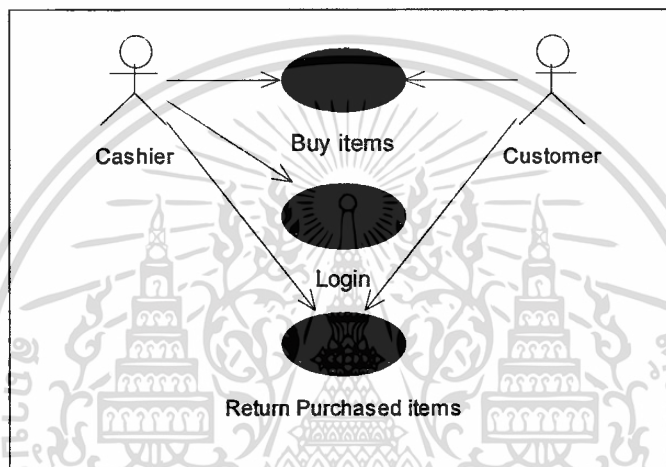
**คอมมิวนิเคชัน(Communication)** เป็นการแสดงความสัมพันธ์หรือการติดต่อสื่อสารกัน (การรับและการให้ข้อมูลข่าวสารแก่กันและกัน) ระหว่างแอ็กเตอร์และยูสเคส ซึ่งอาจจะเป็นการสื่อสารแบบทางเดียวหรือสองทางก็ได้

**รีเลชันชิพ(Relationship)** เป็นการแสดงความสัมพันธ์ระหว่างยูสเคสด้วยกัน โดยใช้เส้นที่มีหัวลูกศรสีขาวความสัมพันธ์ระหว่างยูสเคสซึ่งมีได้ 3 แบบคือ Extends, Include และ generalization

**Extension** ใช้สำหรับการเพิ่มการทำงานให้ยูสเคสใดอะแกรม ด้วยการเรียกใช้จากอีกยูสเคสใดอะแกรมหนึ่ง ลูกศรจะชี้ออกจากยูสเคสใดอะแกรมที่ถูกเรียกใช้ไปยังยูสเคสใดอะแกรมที่ต้องการเพิ่มเติมการทำงานและมีข้อความ << extends >> ระบุอยู่ข้างเส้น

**Inclusion** ใช้สำหรับการนำเอาขั้นตอนการทำงานที่ซ้ำซ้อนกันมาสร้างเป็น ยูสเคส แยกต่างหากเพื่อให้ ยูสเคส อื่น ได้เรียกใช้ เป็นลักษณะเดียวกับเวลาเราเขียน โปรแกรมเป็น โมดูลย่อยๆ เพื่อให้โมดูลอื่น ได้เรียกใช้ << Include >>

generalization เป็นการแสดงให้เห็นถึงการสืบทอดพฤติกรรมและคุณสมบัติบางอย่าง จากยูสเคสใดเคสแกรมหนึ่ง ไปสู่อีกยูสเคสใดเคสแกรมหนึ่ง ปลายลูกศรจะอยู่ที่ยูสเคสใดเคสแกรมหลักที่ถูกถ่ายทอดความสามารถออกไป



รูปที่ 2.1 แสดง Use-Case Diagram สำหรับระบบซื้อขายของผ่าน Internet

### • Class Diagram

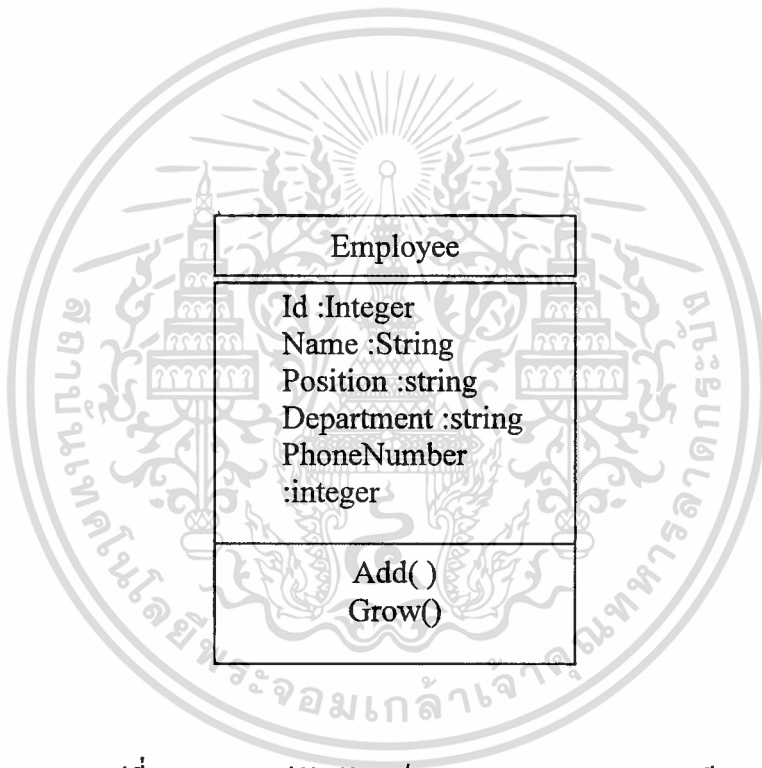
คลาสใดเคสแกรมเป็นสแตติก โมเดลชนิดหนึ่งที่ใช้อธิบายถึงคลาสและความสัมพันธ์ระหว่างคลาส สำหรับออบเจกต์ที่จัดให้อยู่ในคลาสเดียวกันจะมี แอตทริบิวต์, โอเปอเรชัน, ความสัมพันธ์ ความหมายและพฤติกรรมของข้อมูลเหมือนกัน

คลาสหนึ่งๆนั้นสามารถกำหนดทิศทางของการนำไปสร้างโปรแกรมและการสร้างคาสตอื่นๆใน OOP ได้ ตัวคลาสใช้สัญลักษณ์เป็นรูปสี่เหลี่ยมซึ่งอธิบายถึงกลุ่มของออบเจกต์ต่างๆ แอตทริบิวต์(Attribute) หน้าที่การทำงาน(Operation) และความสัมพันธ์(Relationship) โดยพื้นฐานแล้วคลาสประกอบด้วยรายละเอียด 3 ส่วนคือ ชื่อคลาส แอตทริบิวต์ และ โอเปอเรชัน

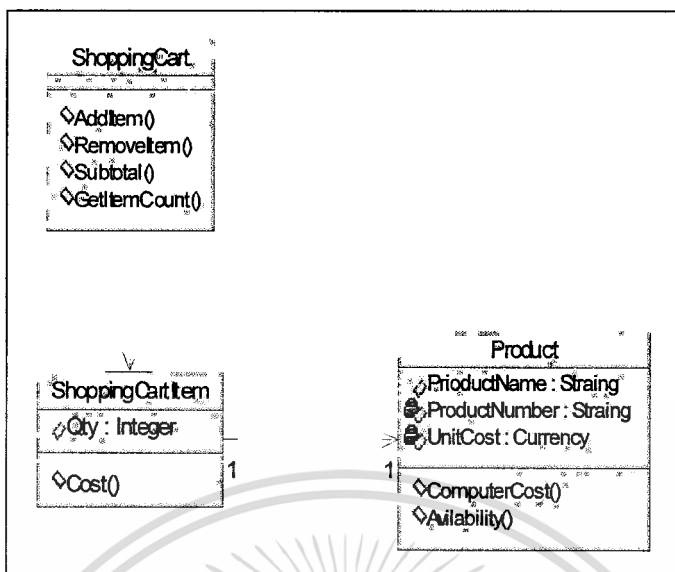
ชื่อคลาส (name) ทุกๆคลาสจะต้องมีการกำหนดชื่อเอาไว้เพื่อแยกจากคลาสอื่นๆ โดยการตั้งชื่อจะได้ตัวอักษรเป็นสตริงธรรมดาทั่วไป

แอตทริบิวต์(attributes) เป็นการบอกถึงคุณสมบัติของคลาส คลาสหนึ่งๆอาจจะมีแอตทริบิวต์ได้ตั้งแต่ 1 ตัวขึ้นไปหรืออาจจะเป็นคลาสที่ไม่มีแอตทริบิวต์เลยก็ได้ นอกจากนี้แอตทริบิวต์ที่อยู่ภายในคลาสเดียวกันอาจจะมีชื่อซ้ำกันกับคลาสอื่นก็ได้

โอเปอเรชัน(Operation) คือพฤติกรรมที่เราสามารถกระทำกับออบเจกต์ได้ โดยที่ออบเจกต์ทั้งหมดที่อยู่ในคลาสเดียวกันจะมีการใช้โอเปอเรชันของคลาสตนเองร่วมกัน คลาสหนึ่งๆ สามารถมีโอเปอเรชันได้มากกว่า 1 โอเปอเรชันหรืออาจจะมีโอเปอเรชันเลยก็ได้ นอกจากนี้เรายังสามารถระบุถึง พารามิเตอร์และประเภทของข้อมูลที่โอเปอเรชันจะให้กลับมา(return type) ได้ด้วย



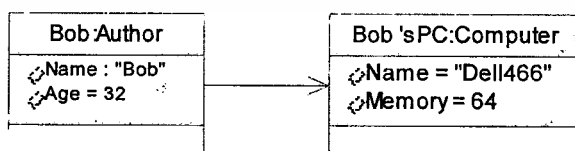
รูปที่ 2.2 แสดงรูปสัญลักษณ์ของ Class Diagram ทะเบียนลูกค้า



รูปที่ 2.3 แสดงตัวอย่างของ Class Diagram สำหรับระบบงานจัดซื้อ

#### • Object diagram

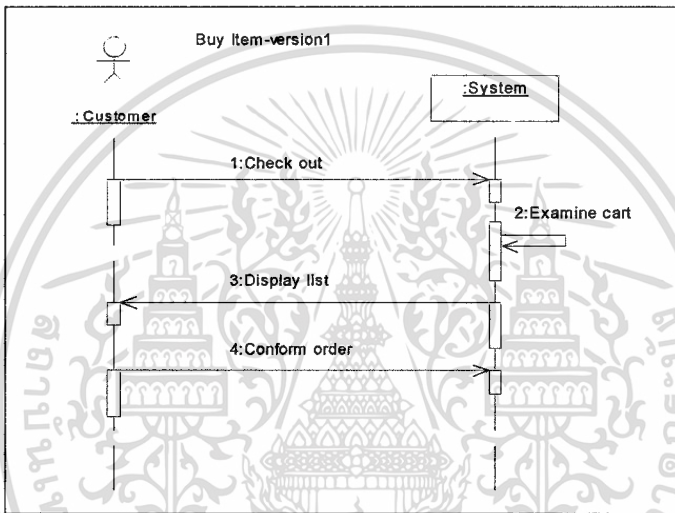
เป็นออบเจกต์โคออร์เดตเป็นสเตตโคออร์เดตที่แสดงความสัมพันธ์ระหว่าง instance ที่เชื่อมโยงกันในช่วงเวลาใดเวลาหนึ่ง ออบเจกต์โคออร์เดตสามารถทำให้เรามองเห็นภาพโดยรวมอย่างกว้างๆ ของคลาสโคออร์เดตได้และเป็นการแสดงถึงความซ้ำซ้อนในโคออร์เดตที่สามารถแสดงได้ด้วยรูปภาพ กล่าวคือ เป็นการแสดงถึงออบเจกต์ต่างๆ ที่เกี่ยวข้องกันในแต่ละช่วงเวลา ซึ่งออบเจกต์สามารถแสดงได้ด้วยคลาสหนึ่งคลาส และจะมีชื่อขีดเส้นใต้ไว้ ตัวอย่างเช่น ออบเจกต์ company เชื่อมต่อกับออบเจกต์ของ 2 Department โดย Department Object ที่มีชื่อเป็น Sales ได้เชื่อมต่อกับออบเจกต์ Department ที่มีชื่อ US sales โดย Department นี้มีผู้จัดการที่มีชื่อและรายละเอียดต่างๆ อยู่ในออบเจกต์ person และ ContractInformation



รูปที่ 2.4 แสดงตัวอย่างของ Object Diagram เพื่อแสดงสำหรับความสัมพันธ์ระหว่าง Bob กับเครื่อง Computer

## • Sequence diagram

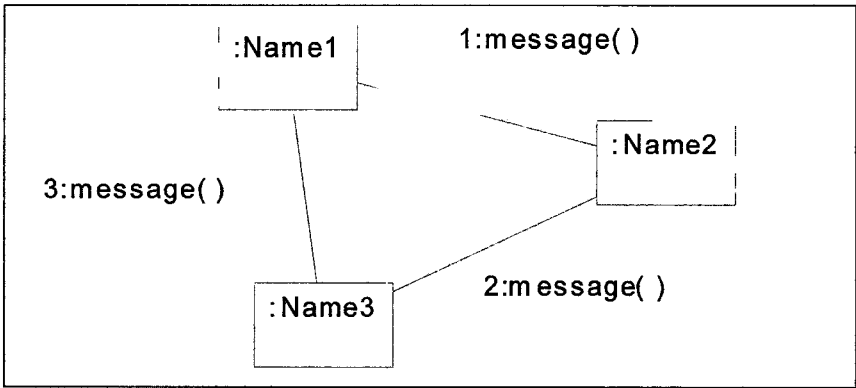
เป็นการอธิบายคลาสที่สร้างขึ้น โดยอธิบายถึงความสัมพันธ์ของออบเจกต์ในมิติของเวลา กล่าวคือการเรียงลำดับออบเจกต์ตามแนวนอน ส่วนแนวตั้งจะแสดงลำดับการส่งผ่านmessage จากออบเจกต์หนึ่งไปยังอีกออบเจกต์หนึ่งที่เกิดขึ้นเมื่อเวลาผ่านไป โดยใช้สัญลักษณ์รูปสี่เหลี่ยมผืนผ้าแทนออบเจกต์คลาส กรณีที่เป็นคลาสชื่อต้องขีดเส้นใต้ สัญลักษณ์เส้นประในแนวตั้งเรียกว่าเส้นชีวิตของออบเจกต์ เป็นการกำหนดขอบเขตการทำงานของออบเจกต์นั้นและสัญลักษณ์ลูกศรแต่ละแบบก็มีความหมายเหมือนกับชนิดของการส่งผ่านข้อความตามที่กล่าวมาข้างต้น ซึ่งซีควเอนซ์ไดอะแกรมเหมาะสำหรับส่วนของการวิเคราะห์ระบบ



รูปที่ 2.5 แสดงตัวอย่างของ Sequence Diagram สำหรับระบบตรวจสอบบัตรเครดิต

## • Collaboration diagram

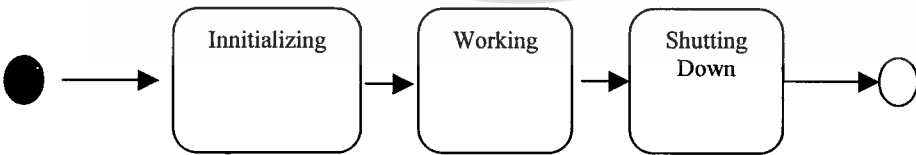
อธิบายถึงออบเจกต์และความสัมพันธ์ระหว่าง ออบเจกต์ ในลักษณะการส่งข้อความไปมาระหว่างผู้ส่งและผู้รับโดยไม่มีลำดับ สัญลักษณ์ที่ใช้เหมือนกับในซีควเอนซ์ไดอะแกรม ทั้งนี้และทั้งนั้นผู้ใช้อาจจะเลือกนำเสนอด้วยวิธีใดวิธีหนึ่งก็ได้ กล่าวคือทั้งสองแผนสามารถเปลี่ยนแปลงรูปแบบไปมาระหว่างกันได้ ถึงแม้ว่า ซีควเอนซ์ไดอะแกรมและคอลลาบอเรชันไดอะแกรมจะเป็นไดอะแกรมที่แสดงที่แสดงปฏิสัมพันธ์กันระหว่างออบเจกต์เหมือนกัน แต่ทั้งสองไดอะแกรมก็ยังมีส่วนที่แตกต่างกันคือ คอลลาบอเรชันไดอะแกรมจะเน้นไปที่เมสเสจหรือข่าวสารที่ออบเจกต์ส่งถึงกัน ส่วนซีควเอนซ์ไดอะแกรมจะเน้นไปที่ช่วงเวลาที่เกิดปฏิสัมพันธ์กันระหว่างออบเจกต์



รูปที่ 2.6 แสดงตัวอย่างของ Collaboration Diagram

• State Diagram

เป็นตัวบ่งบอกถึงเหตุการณ์ต่างๆที่จะมีผลกระทบอะไรเกิดขึ้นบ้าง สเตทไดอะแกรม(State Diagram)จะทำหน้าที่เชื่อมต่อทุกคลาสที่มีพฤติกรรมอันซับซ้อนให้ชัดเจนขึ้น และเป็น การอธิบายถึงพฤติกรรมของระบบซึ่งแต่ละสเตทจะมีความแตกต่างกันขึ้นอยู่กับสถานะในปัจจุบัน รวมทั้งบ่งบอกว่ามีเหตุการณ์ใดบ้างที่มีผลกับการเปลี่ยนแปลงทั้งภายในและภายนอก สเตทไดอะแกรมจะมีส่วนประกอบอยู่ 3 ส่วนคือ ชื่อสเตท โอเปอเรชันนอลสเตท โอเปอเรชันนอลแอกติวิตี สำหรับสเตทไดอะแกรมมี 3 สถานะคือ Initiator การเริ่มต้นกิจกรรม, Terminator สิ้นสุดการทำงานหรือการออกจากสเตท, Do การปฏิบัติงานต่างๆในสเตทเช่น การส่งแมสเสจ การรอ หรือการคำนวณ เราสามารถกล่าวได้อีกอย่างหนึ่งว่า State Diagram เป็นไดอะแกรมที่ใช้สำหรับแสดงการเปลี่ยนแปลงของออบเจกต์ตั้งเริ่มต้นจนสิ้นสุดการเปลี่ยนแปลงในรอบๆ หนึ่ง



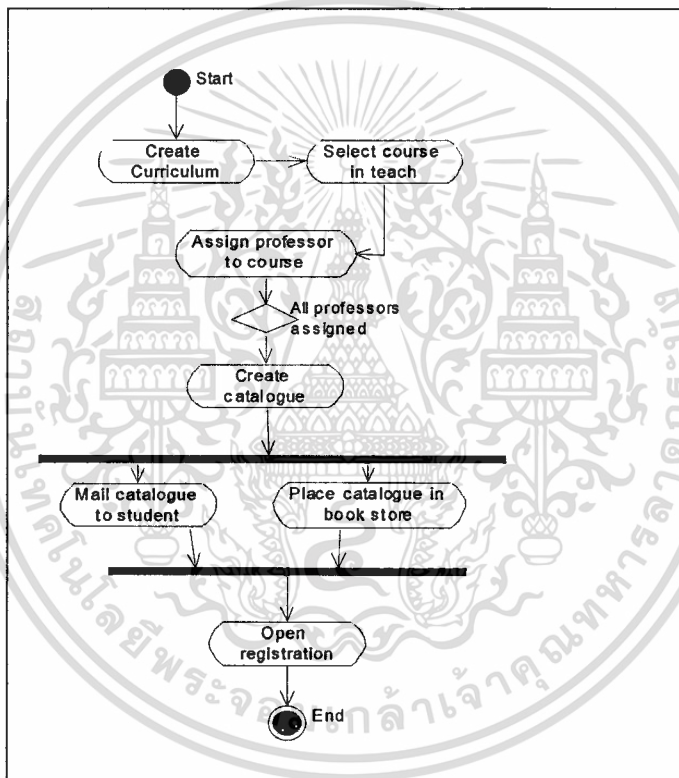
รูปที่ 2.7 แสดงตัวอย่างของ State Diagram สำหรับระบบการทำงานของเครื่องคอมพิวเตอร์

• Activity diagram

อธิบายเป็นการแสดงลำดับขั้นตอนการทำงานของกิจกรรมต่างๆ ในระบบ ซึ่งจะสามารถใช้เป็นแบบในการปฏิบัติงานในหน้าที่ต่างๆ รวมทั้งยังอธิบายถึงขั้นตอนการทำงานในขั้นตอนต่อไปของกิจกรรมอื่นๆด้วย activity diagram เป็นการพิจารณาถึงสถานะ การทำงานนั้นๆ เมื่อเริ่มเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้นแล้วต้องทำให้เสร็จก่อนที่จะเริ่มต้นทำงานในสถานะต่อไปได้ ดังนั้นสิ่งที่ใช้ในการควบคุมการทำงานในขั้นตอนต่างๆก็คือ การติดต่อกับสื่อสารกันหรือการเชื่อมโยงกันของสเตตต่างๆ ซึ่งสามารถกระทำได้ทั้งการส่งและการรับข้อมูลข่าวสารได้ไปพร้อมๆกัน โดยมีการนำแนวคิดมาจากหลายๆเทคนิคเช่น Event diagram และ SDL State Modeling เหล่านี้นำมาใช้ประโยชน์ในการเชื่อมต่อกับ Flow และรายละเอียดของพฤติกรรมที่มีการประมวลผลแบบขนาน ก็สามารถทำงานได้หลายอย่างในเวลาเดียวกันเป็นจำนวนมาก

Activity Diagram มีลักษณะเช่นเดียวกับโฟลชาร์ต คือใช้สำหรับแสดงขั้นตอนการทำงานของระบบ โดยที่แต่ละขั้นตอนในการทำงานแต่ละขั้นตอนเราจะเรียกว่า Activity Diagram



รูปที่ 2.8 แสดงตัวอย่างของ Activity Diagram สำหรับระบบงานจัดซื้อหนังสือประกอบการเรียน

• **Component diagram**

แสดงโครงสร้างทางกายภาพของการเขียนโปรแกรมในส่วนของส่วนประกอบต่างๆที่ต้องนำมาใช้ในการรันแอปพลิเคชันต่างๆ จะประกอบด้วย source code ,binary,executable ส่วนประกอบเหล่านี้จะเป็น logical class หรืออาจจะเป็นคลาสของการ Implement ดังนั้นในการสร้างคลาสที่เกิดจาก Logical ไปเป็น Component จะมีผลทำให้เกิดการทำการกิจกรรมร่วมกันภายใน

Component โดยต้องทำการวิเคราะห์ถึงส่วนประกอบต่างๆด้วย Component สามารถแสดงถึงการติดต่อสื่อสารกันของคลาสที่เปิดกว้างขึ้น

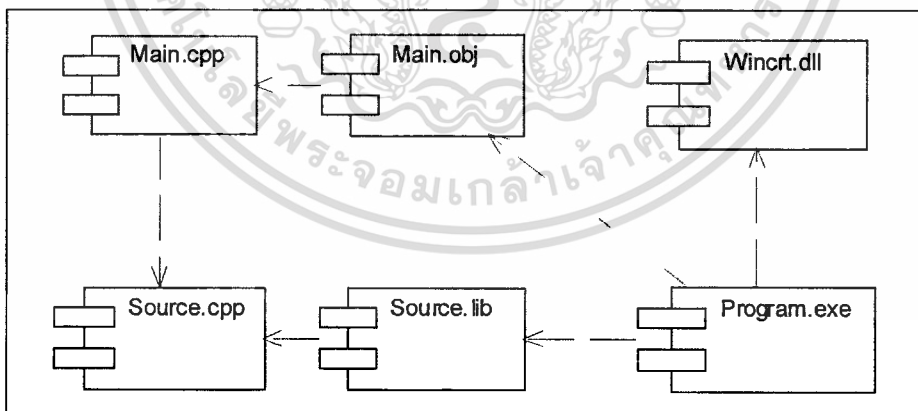
คอมโพเนนต์ของซอฟต์แวร์คือองค์ประกอบของระบบงานที่มีตัวตนจริงๆ สามารถจับต้องได้ เช่น ไฟล์ข้อมูล , โลบรารีต่างๆ, เอกสารการใช้งานของระบบ เป็นต้น

**ประโยชน์ของการใช้คอมโพเนนต์ต่อที่ผู้พัฒนาระบบ**

1. ไคลเอนต์ได้มองเห็น โครงสร้างของระบบที่สร้างเสร็จแล้ว
2. โปรแกรมเมอร์มีโครงสร้างที่ชัดเจนเพื่อเป็นแนวทางในการพัฒนาให้ ได้ระบบที่ตั้งเป้าหมายเอาไว้
3. ฝ่ายจัดทำเอกสารเชิงเทคนิคสามารถเข้าใจระบบได้อย่างถ่องแท้ และนำมาจัดทำเอกสารประกอบได้อย่างถูกต้อง
4. สนับสนุนหลักการของการนำระบบกลับมาให้ใหม่ได้

**องค์ประกอบหลักๆ ของ Component Diagram ได้แก่**

- 1.คอมโพเนนต์ Component
- 2.อินเทอร์เฟซ Interfaces
- 3.ความสัมพันธ์ Relationships



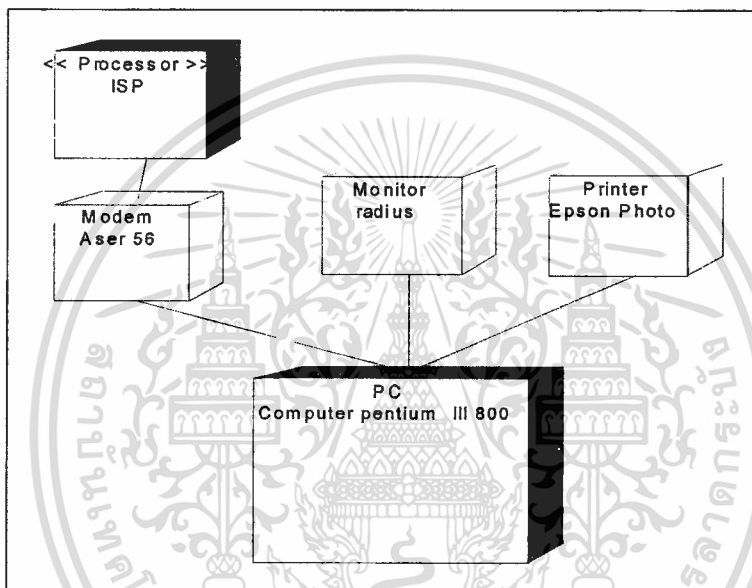
**รูปที่ 2.9 แสดงตัวอย่างของ Component Diagram สำหรับโครงการพัฒนาระบบงาน**

**• Deployment diagram**

แสดงความสัมพันธ์ทางกายภาพระหว่างส่วนประกอบของฮาร์ดแวร์และซอฟต์แวร์ในการส่ง message เข้าไปในระบบ

Deployment Diagram จะให้ภาพของการเซตอัพฮาร์ดแวร์ในขั้นตอนสุดท้ายว่าควรมีลักษณะอย่างไร Deployment Diagram จะเกิดจากสิ่งที่เรียกว่า โหนด (node) มาประกอบกัน ซึ่ง โหนดจะเป็นฮาร์ดแวร์หลักๆของระบบหรือรหัสของการคำนวณ โดยโหนดจะแบ่งออกเป็น 2 ประเภทได้แก่ processor และ device

Processor จะเป็นโหนดที่สามารถทำให้คอมพิวเตอร์ทำงานได้แต่ device จะเป็นโหนดที่ไม่ได้ทำให้คอมพิวเตอร์เกิดการทำงานและโดยปกติ device จะเป็นอินเทอร์เฟซที่ใช้ติดต่อกับสิ่งที่อยู่ภายนอกระบบ เช่น เครื่องพิมพ์ โมเด็ม เป็นต้น



รูปที่ 2.10 แสดงตัวอย่างของ Deployment Diagram เพื่อแสดงส่วนประกอบของ Computer

Component Diagram และ Deployment Diagram เป็นไคอะแกรมที่อยู่ในขั้นตอนของการอิมพลิเมนต์ระบบงานนั้นคือเป็นขั้นตอนที่จะต้องมีการออกแบบระบบงานโดยพิจารณาถึงสถาปัตยกรรมโดยรวมของระบบงาน และเป็นขั้นตอนของการแปลงออบเจกต์ต่างๆในกระบวนการออกแบบออบเจกต์ให้กลายเป็นโค้ดโปรแกรม

### ประโยชน์ของ UML

1.UMLเป็นการสร้างแนวคิดในการ โปรแกรมแบบใหม่คือ การ โปรแกรมเชิงวัตถุหรือ OOP(Object Oriented Programming) ซึ่งมีพื้นฐานอยู่บนคุณสมบัติในการทำงานร่วมกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ออบเจกต์ (Object) เช่น Property, Method, Encapsulation ทำให้การเปลี่ยนแปลงแก้ไขทำได้ง่ายและสะดวกต่อการจัดการ

2.UMLเป็นภาษาที่เป็นมาตรฐานเปิด(Open Standard) ของทุกภาษาในปัจจุบันไม่ว่าจะเป็น java, VisualAge และผลิตภัณฑ์ใหม่ ๆ ของ Microsoft ล้วนแล้วแต่สนับสนุน UML

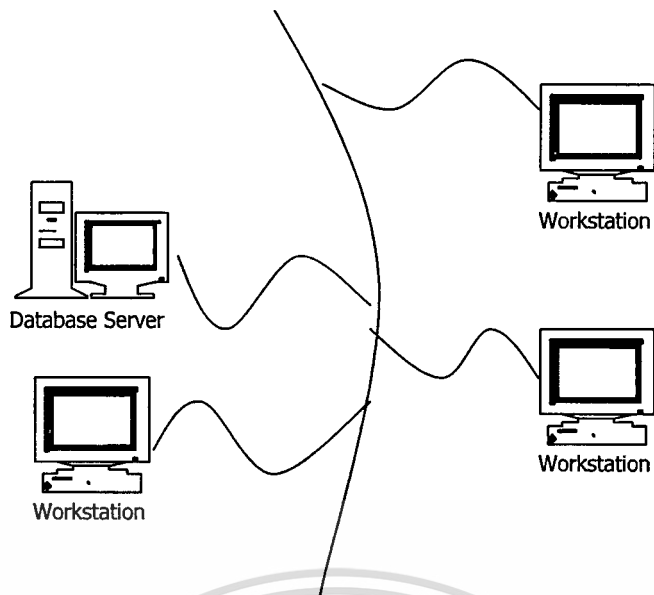
3.ภาษา UML ครอบคลุมทุกส่วนในวงจรชีวิต (Life cycle) ของการพัฒนาระบบตั้งแต่ขั้นตอนของการหาความต้องการของระบบ requirement , การออกแบบระบบ design,การนำไปใช้งานจริง Implement ,การติดตั้งระบบ Installation ไปจนถึงขั้นตอนการจัดทำเอกสาร documentation และถึงแม้ว่าระบบงานนั้นจะมีการใช้เทคโนโลยีหลายๆอย่างร่วมกันก็ยังคงสามารถนำภาษา UML ไปประยุกต์ใช้งานได้

4.UML เป็นภาษาที่มีความสมดุลในแง่ของความเรียบง่ายและความซับซ้อน กล่าวคือไม่ยากเกินไปที่จะเรียนรู้และนำมาใช้งานจริง และก็สามารถนำไปใช้กับงานที่มีความซับซ้อนมากๆ ได้ด้วย

5.มีบริษัทชั้นนำและอุตสาหกรรมต่างๆให้การยอมรับและให้การสนับสนุน ไม่ว่าจะเป็น HP รวมทั้งโปรดักส์ต่างๆ เช่น Websphere, VisualAge ของบริษัท IBM เป็นต้น

## 2.2ระบบฐานข้อมูลแบบไคลน์/เซิร์ฟเวอร์

ฐานข้อมูลแบบไคลเอนต์/เซิร์ฟเวอร์(C/S) แบ่งการจัดการฐานข้อมูลออกเป็นสองระบบคือ ไคลเอนต์พีซีจะรันแอปพลิเคชันฐานข้อมูลและดาต้าเบส เซิร์ฟเวอร์(database server) ซึ่งรัน DBMS ทั้งหมดหรือบางส่วน ไฟล์เซิร์ฟเวอร์บนแลนยังเตรียมรีซอร์สที่ใช้ร่วมกันเช่นเนื้อที่ดิสก์สำหรับแอปพลิเคชันและเครื่องพิมพ์ ดาตาเบสเซิร์ฟเวอร์สามารถทำงานบนเครื่องเดียวกับไฟล์เซิร์ฟเวอร์หรือ(โดยทั่วไป)บนเครื่องคอมพิวเตอร์ของตนเอง แอปพลิเคชันฐานข้อมูลบนเครื่องไคลเอนต์พีซีถูกเรียกว่า ระบบฟรอนเอนต์(Front End) ทำหน้าที่จัดการกับหน้าจอและจัดการอินพุต/เอาต์พุตของผู้ใช้ ระบบส่วนแบ็กเอนต์(Back End) บนดาตาเบสเซิร์ฟเวอร์ทำหน้าที่บนข้อมูลและการเข้าถึงดิสก์ ตัวอย่างเช่น ผู้ใช้ที่อยู่บนฟรอนเอนต์สร้างคำร้องขอ (คิวรี) เพื่อขอข้อมูลจากดาตาเบสเซิร์ฟเวอร์ ดาตาเบสเซิร์ฟเวอร์จะดำเนินการค้นหาอย่างแท้จริงและส่งเฉพาะข้อมูลที่เป็นคำตอบของคิวรีของผู้ใช้กลับไปที่เท่านั้น



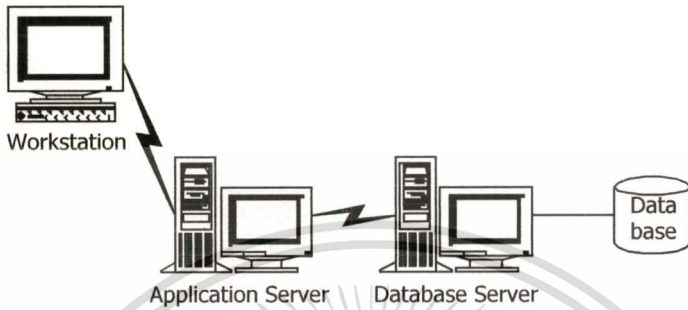
รูปที่ 2.11 แสดงระบบ Client/Server

ประโยชน์โดยตรงของระบบ C/S เป็นเรื่องที่ชัดเจนก็คือ การแบ่งการจัดการออกเป็นสองระบบซึ่งเป็นการช่วยลดจำนวนของการจราจรของข้อมูลบนสายเคเบิลเน็ตเวิร์ค ในบางครั้งจะพบความสับสนในความหมายที่แตกต่างกันของคำศัพท์บางคำในด้านคอมพิวเตอร์ คำจำกัดความของไคลเอนต์/เซิร์ฟเวอร์ที่ปรากฏอยู่บนระบบ UNIX ที่ทำงานอยู่บนระบบติดต่อกับผู้ใช้กราฟฟิก X-WINDOW นั้นการแบ่งกระบวนการจัดการจะเป็นเช่นเดียวกับระบบ C/S บนพีซี แต่อย่างไรก็ตามส่วนพร้อมเอนต์จะถูกเรียกว่า เซิร์ฟเวอร์ภาพโต้ X-WINDOW และมันทำหน้าที่เตรียมบริการเพื่อแสดงและติดต่อกับผู้ใช้ระบบแบ็กเอนต์ซึ่ง DBMS ทำงาน เป็นการอ้างอิงถึงไคลเอนต์ของบริการที่ถูกเตรียมโดยระบบพร้อมเอนต์ จำนวนของระบบ C/S ได้เติบโตขึ้นอย่างรวดเร็วระบบใหม่ๆหนึ่งระบบได้ถูกออกแบบและเปิดตัวเกือบทุกเดือน โดยทั่วไประบบไคลน์จะทำงานอยู่บนเครื่องพีซี ส่วนมากดาตาเบสเซิร์ฟเวอร์สามารถทำงานได้ตั้งแต่เครื่องพีซีตัวหนึ่งจนถึงเครื่องเมนเฟรม

ข้อเสียที่มากที่สุดของระบบฐานข้อมูล แบบไคลน์/เซิร์ฟเวอร์คือ ต้องการให้ข้อมูลนั้นถูกจัดเก็บบนระบบเดียว เรื่องนี้ก่อให้เกิดปัญหาได้ในบริษัทขนาดใหญ่ ซึ่งต้องรองรับผู้ใช้งานข้อมูลที่อยู่กระจายไปทั่วพื้นที่หรือที่จำเป็นต้องใช้ส่วนของฐานข้อมูลแผนกบางส่วนร่วมกับฐานข้อมูลแผนกอื่นๆ หรือโฮสต์ศูนย์กลาง วิธีการบางอย่างเพื่อที่จะกระจายข้อมูลไปยังโฮสต์ต่างๆ เป็นสิ่งที่จำเป็น ซึ่งเป็นการนำเข้าสู่การพัฒนาแบบกระจาย

ไคลน์/เซิร์ฟเวอร์ ชนิด 3-Tire ระบบไคลน์/เซิร์ฟเวอร์ ชนิดนี้จะมีการแยกส่วนแสดงผลออกจากตัวโปรแกรมประยุกต์ โดยให้คอมพิวเตอร์ที่เป็นลูกข่ายจะทำหน้าที่จัดการส่วนติดต่อผู้ใช้

เพียงอย่างเดียว ส่วนตัวโปรแกรมประยุกต์นั้นจัดเก็บไว้เพียงที่เดียวบนเครื่องคอมพิวเตอร์ที่ทำหน้าที่ประมวลผล ส่วนของโปรแกรมประยุกต์เรียกว่า Application ดังแสดงในรูปที่ 12.12



รูปที่ 2.12 แสดงสถาปัตยกรรม Client/server แบบ 3-Tire

Server ซึ่งทำหน้าที่ติดกับตัว DBMS ซึ่งอยู่บน Database Server ซึ่งเป็นคอมพิวเตอร์อีกเครื่องหนึ่งซึ่งจะจัดเก็บเฉพาะ DBMS และ File Server ระบบลักษณะนี้สามารถหลีกเลี่ยงปัญหาความต้องการขยายขีดความสามารถของ Database Server โดยนำโปรแกรมประยุกต์ไว้บน Application Server ซึ่งถ้าจะเพิ่มจำนวนผู้ใช้ก็เพียงแต่เพิ่มคอมพิวเตอร์ชั้นที่สองนี้ขึ้นได้โดยไม่ต้องเปลี่ยนรุ่นหรือเปลี่ยนเครื่อง Database Server ให้มีขนาดใหญ่ขึ้น ลักษณะนี้ทำให้การขยายขีดความสามารถระบบเมื่อจำนวนผู้ใช้เพิ่มขึ้นทำได้ง่าย นอกจากนี้ถ้ามีการแก้ไขโปรแกรมก็จะแก้ไขเพียงในส่วนของ Application Server เท่านั้น

# บทที่ 3

## ขั้นตอนในการดำเนินงาน

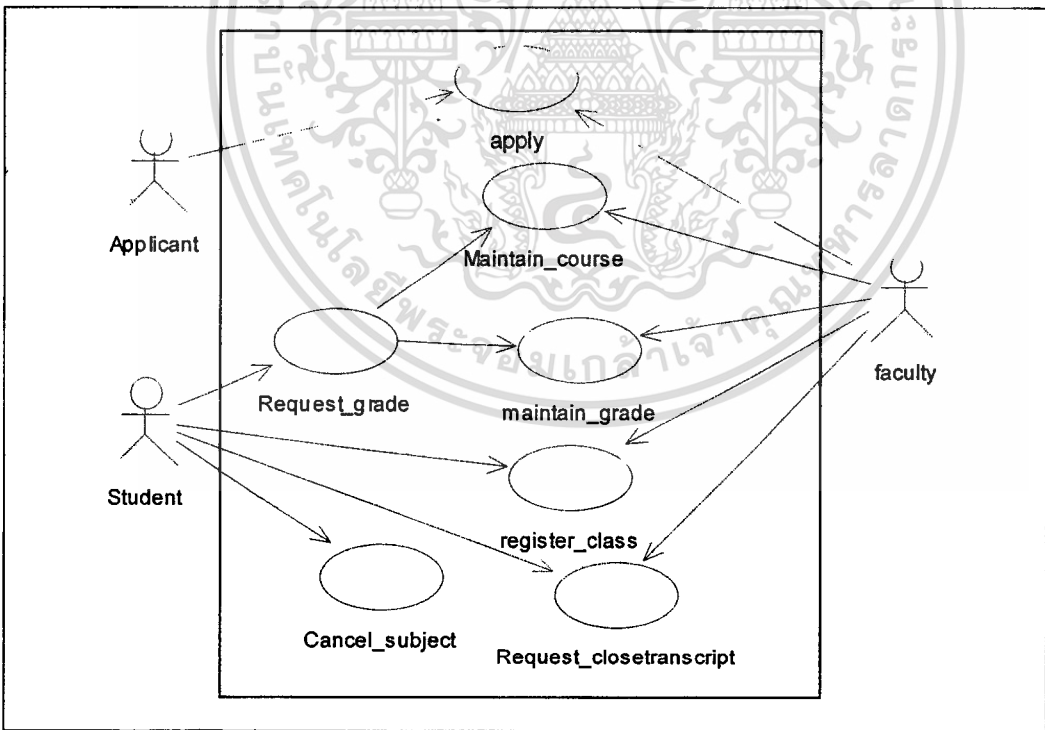
ขั้นตอนในการพัฒนาระบบงานบัณฑิตศึกษาและวิชาการสำหรับคณะเทคโนโลยีสารสนเทศ ประกอบด้วย 2 ขั้นตอนดังนี้คือ

- 1.วิเคราะห์ระบบงาน
- 2.การออกแบบระบบงาน

### 3.1 การวิเคราะห์ระบบงาน

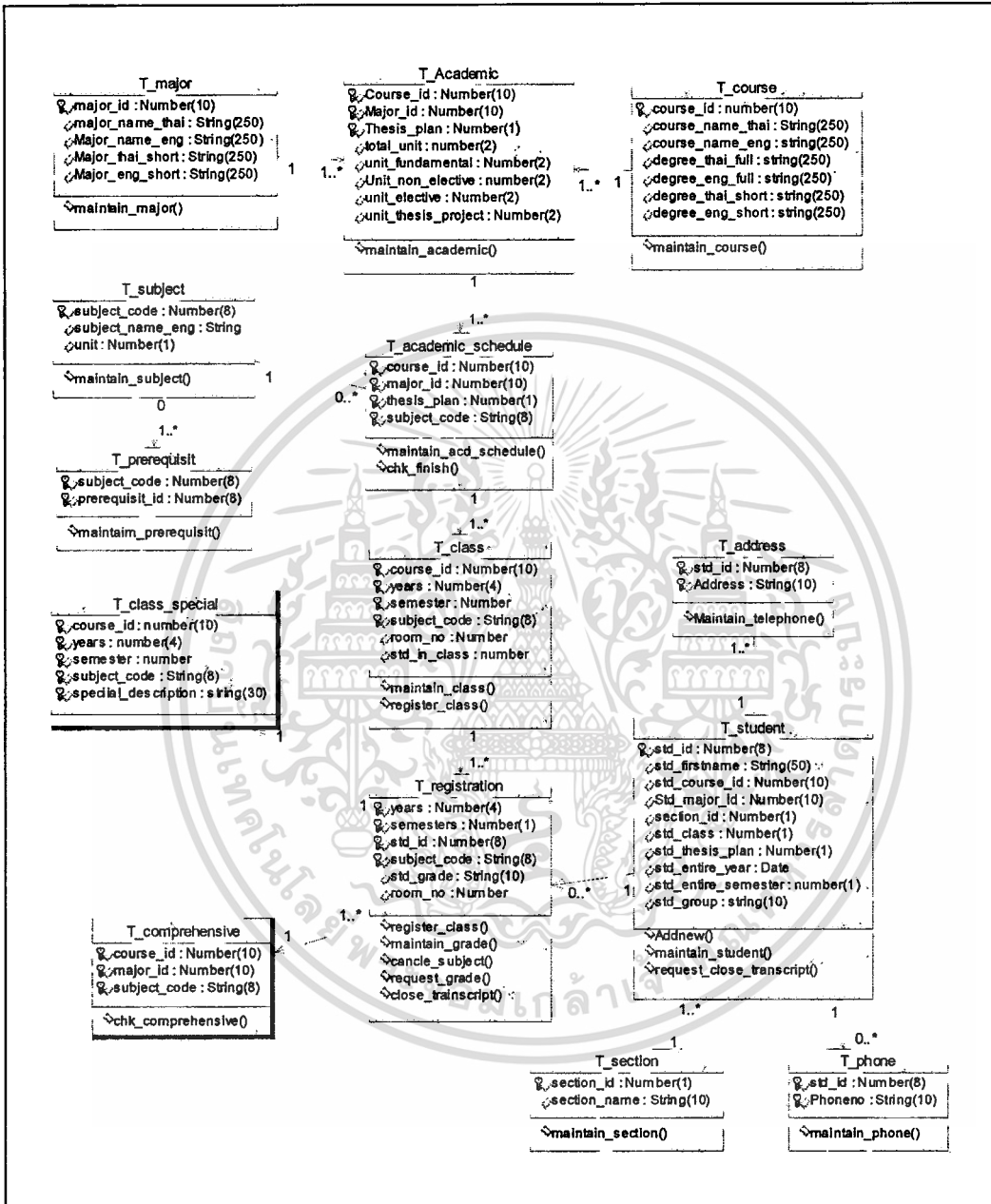
การออกแบบระบบงานเราใช้ UML เป็นเครื่องมือในการวิเคราะห์ โดยในที่นี้เราจะนำเสนอเพียงแค่ 2 Model คือ Uses case daigram และ Class Diagram เท่านั้น

User case diagram



รูปที่ 3.1 Uses case diagram สำหรับระบบบัณฑิตศึกษาและวิชาการ

Class diagram

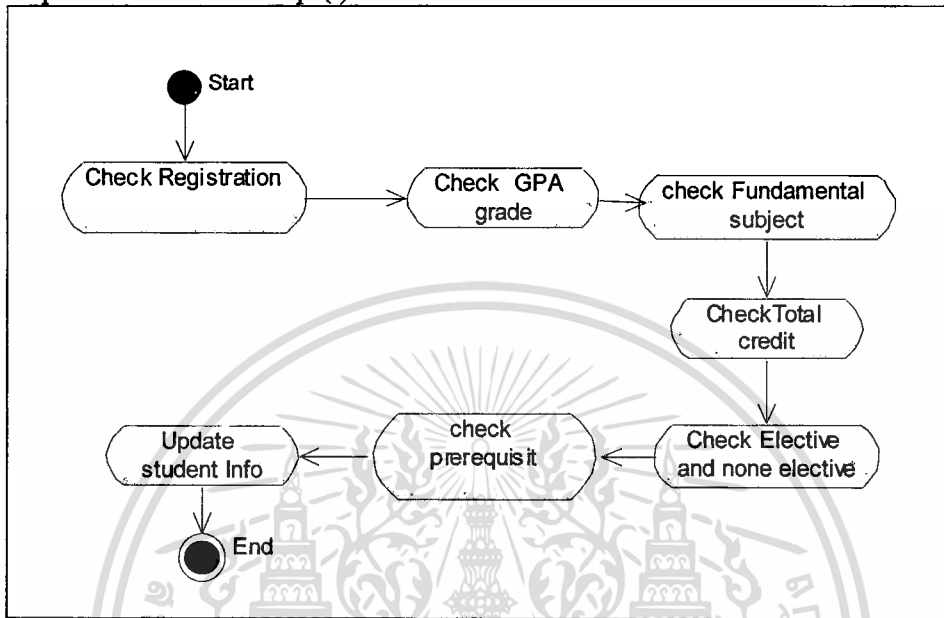


รูปที่ 3.2 Class diagram สำหรับระบบบัณฑิตศึกษาและวิชาการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

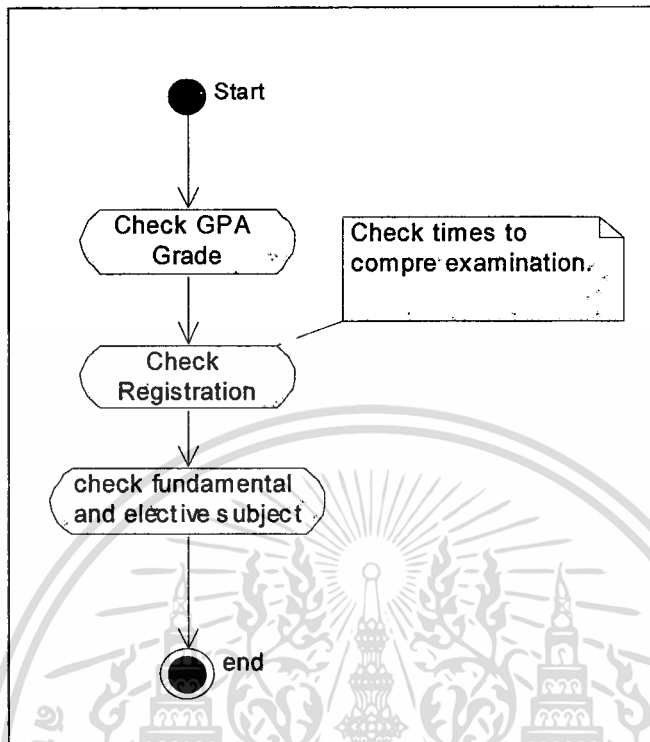
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### Request Close Transcript()



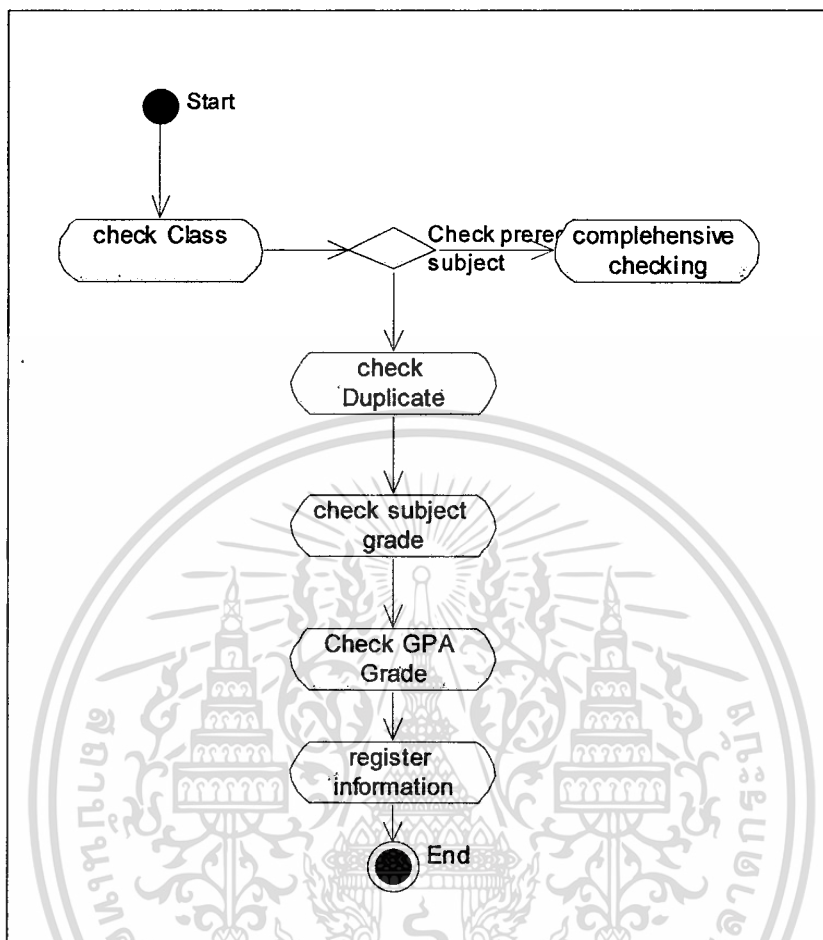
รูปที่ 3.3 แสดง Activity Diagram สำหรับระบบทะเบียนบัณฑิตศึกษา

Comprehensive Checking( )

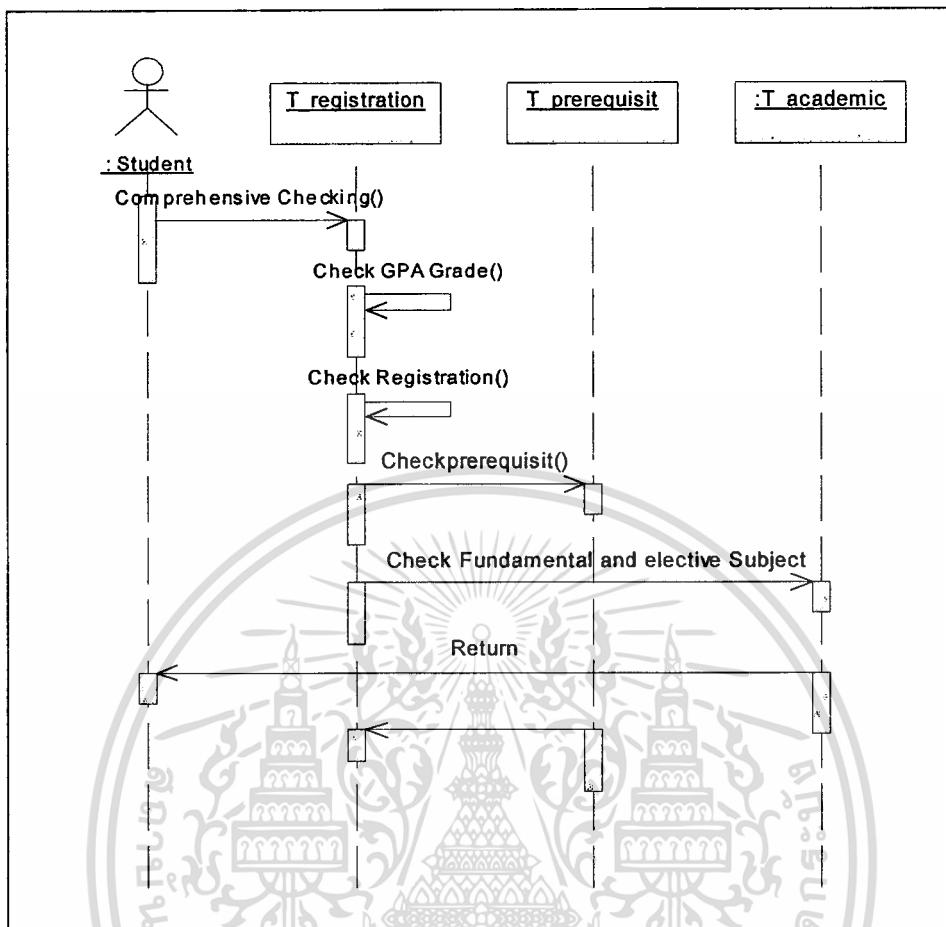


รูปที่ 3.4 แสดง Activity Diagram สำหรับตรวจสอบสิทธิ์การสอบ Comprehensive และ Oral

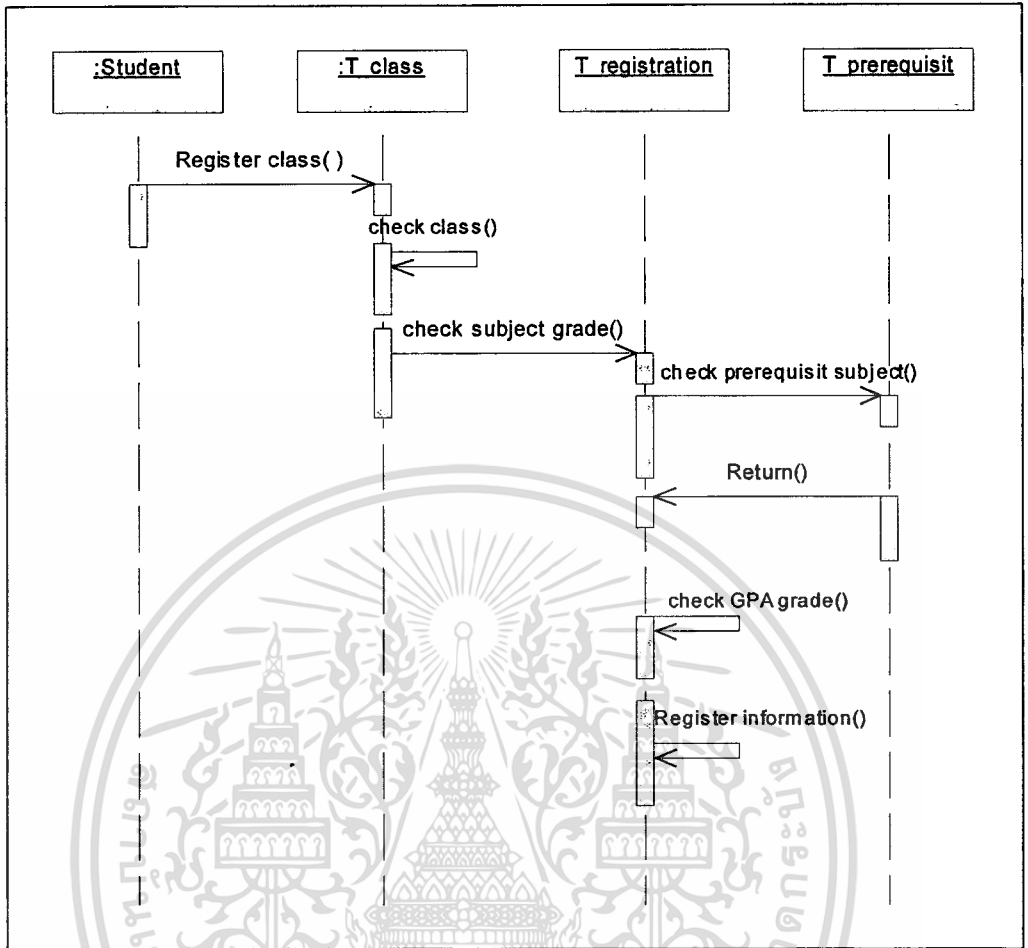
:Register Class ()



รูปที่ 3.5 แสดง Activity Diagram สำหรับการลงทะเบียนเรียนของนักศึกษา



รูปที่ 3.6 แสดง Sequence Diagram สำหรับตรวจสอบสิทธิการสอบ  
Comprehensive และ Oral



รูปที่ 3.7 แสดง Sequence Diagram สำหรับการลงทะเบียนเรียนของนักศึกษา

## บทที่ 4

### การออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลสำหรับระบบบัณฑิตศึกษาและวิชาการ โดยอาศัยรูปแบบการ Design ในลักษณะของ Object หรือ association Object-Oriented Analysis and Design ประกอบกับทฤษฎีการ Normalization สามารถนำเสนอในรูปแบบของตารางได้ 16 ตาราง(Table) ดังนี้คือ

#### ตารางที่ 4.1 ทะเบียนหลักสูตร

T COURSE

Field	Data Type	Description	Remark
course_id	number(10)	หมายเลขหลักสูตร	primary key
course_name_thai	varchar2(250)	ชื่อหลักสูตรภาษาไทย	not null
course_name_eng	Varchar2(250)	ชื่อหลักสูตรภาษาอังกฤษ	not null
degree_thai_full	varchar2(250)	ชื่อปริญญาภาษาไทย	not null
degree_eng_full	Varchar2(250)	ชื่อปริญญาภาษาอังกฤษ	not null
degree_thai_short	varchar2(250)	ชื่อปริญญาภาษาไทย(ย่อ)	
degree_eng_short	Varchar2(250)	ชื่อปริญญาภาษาอังกฤษ(ย่อ)	

#### ตารางที่ 4.2 ทะเบียนแขนงวิชา

T MAJOR

Field	Data Type	Description	Remark
major_id	number(10)	หมายเลขแขนงวิชา	primary key
major_name_thai	varchar2(250)	ชื่อแขนงวิชาภาษาไทย	not null
major_name_eng	Varchar2(250)	ชื่อแขนงวิชาภาษาอังกฤษ	not null
major_thai_short	varchar2(250)	ชื่อแขนงวิชาภาษาไทย(ย่อ)	
major_eng_short	Varchar2(250)	ชื่อแขนงวิชาภาษาอังกฤษ(ย่อ)	

#### ตารางที่ 4.3 โครงสร้างหลักสูตรการเรียน

T ACADEMIC

Field	Data Type	Description	Remark
course_id	number(10)	ระดับการศึกษา	primary key
major_id	number(10)	แขนงวิชา	primary key
thesis_plan	number(1)	แผน(วิทยานิพนธ์/โปรเจก)	primary key

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้ไปใช้ประโยชน์ด้านอื่นได้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Field	Data Type	Description	Remark
total_unit	number(2)	จำนวนหน่วยกิตทั้งหมด	
unit_fundamental	Number(2)	จำนวนหน่วยกิตวิชาบังคับร่วม	
unit_non_elective	Number(2)	จำนวนหน่วยกิตวิชาบังคับ สาขา	
unit_elective	Number(2)	จำนวนหน่วยกิตวิชาเลือก	
unit_thesis_project	Number(2)	จำนวนหน่วยกิตปัญหาพิเศษ	

#### ตารางที่ 4.4 หลักสูตรการเรียนรายวิชา

##### T ACADEMIC SCHEDULE

Field	Data Type	Description	Remark
course_id	number(10)	หมายเลขหลักสูตร	primary key
major_id	number(10)	แขนงวิชา	primary key
thesis_plan	Number(1)	แผน(วิทยานิพนธ์/โปรเจค)	primary key
subject_code	varchar2(8)	รหัสวิชา	primary key

#### ตารางที่ 4.5 หลักสูตรแผนการเรียน

##### T THESIS PLAN

Field	Data Type	Description	Remark
Thesis_plan	number(1)	หมายเลขแผนการเรียน	primary key
Thesis_plan_desc	Varchar2(100)	ชื่อแผนการเรียน	not null
Thesis_plan_eng	Varchar2(100)	ชื่อแผนการเรียน(ภาษาอังกฤษ)	

#### ตารางที่ 4.6 ทะเบียนวิชา

##### T SUBJECT

Field	Data Type	Description	Remark
subject_code	Number(8)	รหัสวิชา	primary key
subject_name_eng	varchar2(100)	ชื่อวิชาภาษาอังกฤษ	not null
subject_name_thai	varchar2(100)	ชื่อวิชาภาษาไทย	not null
unit	number(1)	จำนวนหน่วยกิต	not null
unit_desc	varchar2(10)	รายละเอียดของหน่วยกิต	
subject_type	number(2)	หมวดวิชา	

#### ตารางที่ 4.7 ทะเบียนวิชา prerequisite

##### T PREREQUISIT

Field	Data Type	Description	Remark
subject_code	Varchar2(8)	รหัสวิชา	primary key
Prerequisite_id	Varchar2(8)	รหัสวิชาบังคับเรียน	primary key

#### ตารางที่ 4.8 ทะเบียนการลงทะเบียนรายวิชา

##### T CLASS

Field	Data Type	Description	Remark
course_id	Number(10)	หมายเลขหลักสูตร	primary key
Years	Number(4)	ปีการศึกษา	primary key
semester	Number	ภาคการศึกษา	primary key
subject_code	varchar2(8)	รหัสวิชา	primary key
Room_no	Number	จำนวนห้องเรียน	
Std_in_class	Number	จำนวนนักศึกษาในแต่ละห้อง	

#### ตารางที่ 4.9 ทะเบียนการลงทะเบียนวิชาพิเศษ

##### T CLASS SPECIAL

Field	Data Type	Description	Remark
course_id	Number(10)	หมายเลขหลักสูตร	primary key
Years	Number(4)	ปีการศึกษา	primary key
semester	Number	ภาคการศึกษา	primary key
subject_code	varchar2(8)	รหัสวิชา	primary key
Special_no	varchar2(8)	หมายเลขลำดับวิชา	primary key
Special_description	Varchar2(100)	รายละเอียดของวิชา	not null
Room_no	Number	จำนวนห้องเรียน	
Std_in_class	Number	จำนวนนักศึกษาในแต่ละห้อง	

#### ตารางที่ 4.10 ประวัติการลงทะเบียนของนักศึกษา

##### T REGISTRATION

Field	Data Type	Description	Remark
years	Number(4)	ปีการศึกษา	primary key
Semester	Number(1)	ภาคการศึกษา	primary key
std_id	Number(8)	รหัสนักศึกษา	primary key
subject_code	varchar2(8)	รหัสวิชา	primary key
std_grade	varchar2(10)	เกรดของนักศึกษา	

#### ตารางที่ 4.11 ทะเบียนคำนำหน้าชื่อ

##### T PREFIX

Field	Data Type	Description	Remark
Prefix_id	Number (2)	รหัสคำนำหน้าชื่อ	primary key
Prefix_name	varchar2 (10)	คำนำหน้าชื่อ(ภาษาไทย)	not null
Prefix_name_eng	varchar2 (10)	คำนำหน้าชื่อ(ภาษาอังกฤษ)	

#### ตารางที่ 4.12 ทะเบียนประเภทของนักศึกษา

##### T SECTION

Field	Data Type	Description	Remark
Section_id	Number (1)	รหัส section	primary key
section_name	varchar2 (10)	ชื่อ section ปกติ / สมทบ	not null

#### ตารางที่ 4.13 ทะเบียนนักศึกษา

##### T STUDENT

Field	Data Type	Description	Remark
std_id	number	รหัสนักศึกษา	primary key
std_firstname	varchar2 (50)	ชื่อนักศึกษา	not null
std_lastname	varchar2 (50)	นามสกุลนักศึกษา	not null
std_firstname_eng	varchar2 (50)	ชื่อนักศึกษาภาษาอังกฤษ	not null
std_lastname_eng	varchar2 (50)	นามสกุลนักศึกษาภาษาอังกฤษ	not null
std_course_id	number (10)	หลักสูตรการเรียน	not null
std_major_id	number (10)	แขนงวิชา	not null
Section_id	number (1)	ประเภทของนักศึกษา ปกติ/สมทบ	not null
std_entire_year	date	ปีที่เข้ารับการการศึกษา	
std_entire_semester	number	ภาคการศึกษาที่เข้ารับการการศึกษา	
std_sex	number (1)	เพศ	
std_prefix	number (2)	คำนำหน้าชื่อ	
std_birthdate	date	วันเกิด	
std_email	varchar2 (100)	อินเตอร์เน็ตแอดเดรส	
std_finish_year	date	ปีที่สำเร็จการศึกษา	
std_finish_semester	Number (1)	ภาคการศึกษาที่สำเร็จการศึกษา	
Std_group	Varchar2 (10)	รุ่น	not null
std_class	Number (1)	ห้องเรียน	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Field	Data Type	Description	Remark
std_thesis_plan	Number (1)	ประเภทของแผนการเรียน วิทยานิพนธ์ / โปรเจก	

#### ตารางที่ 4.14 ทะเบียนที่อยู่นักศึกษา

##### T ADDRESS

Field	Data Type	Description	Remark
Std_id	Number (8)	รหัสนักศึกษา	primary key
Address	Varchar2 (250)	ที่อยู่นักศึกษา	primary key
province	Varchar2 (30)	ชื่อจังหวัด	not null

#### ตารางที่ 4.15 ทะเบียนเบอร์โทรศัพท์นักศึกษา

##### T PHONE

Field	Data Type	Description	Remark
Std_id	Number (8)	รหัสนักศึกษา	primary key
Phoneno	Varchar2 (30)	เบอร์โทรศัพท์	Primary key

#### ตารางที่ 4.16 ทะเบียนหลักสูตรการสอบ COMPREHENSIVE

##### T COMPREHENSIVE

Field	Data Type	Description	Remark
Course_id	Number (8)	หลักสูตรการเรียน	primary key
Major_id	Number (8)	แขนงวิชา	primary key
Subject_code	Varchar2 (8)	รหัสวิชาสำหรับตรวจสอบการ ผ่าน Comprehensive	primary key

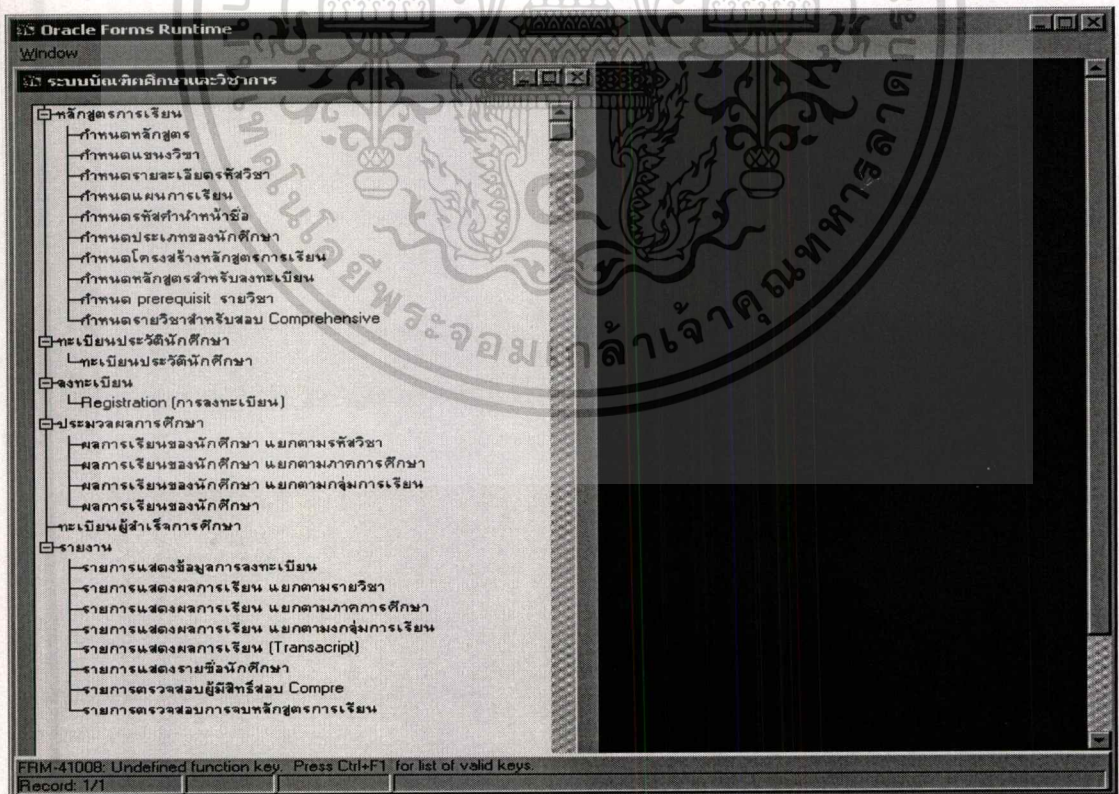
## บทที่ 5

### การพัฒนาระบบ

#### 5.1 การพัฒนาโปรแกรม

ระบบงานบัณฑิตศึกษาและวิชาเป็นระบบงานที่ได้ถูกพัฒนาขึ้น โดยแบ่งเมนูการใช้งานของระบบออกเป็น 5 ส่วนดังนี้คือ

- หลักสูตรการเรียน
- ทะเบียนประวัตินักศึกษา
- ลงทะเบียน
- ประมวลผลการศึกษา
- ทะเบียนผู้สำเร็จการศึกษา
- รายงาน



รูปที่ 5.1 หน้าจอแสดงเมนูการใช้งานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Design screen

สำหรับส่วนรับข้อมูลและแสดงผลข้อมูลทางจอภาพ เราใช้ Tool Developer 2000 สำหรับออกแบบและพัฒนาระบบงาน โดยสามารถสรุปรายละเอียดและฟังก์ชันงานหลักๆของระบบตามเมนูผู้ใช้งาน ได้ดังนี้คือ

### 5.1.1 ขั้นตอนการกำหนดหลักสูตร

การกำหนดหลักสูตรการเรียน หรือขั้นตอนการกำหนดชื่อโปรแกรมการศึกษา ในที่นี้กำหนดให้มีหลักสูตรเดียวคือ หลักสูตร มหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ ดังรูปที่ 5.2

The screenshot shows the Oracle Forms Runtime window titled 'Frm010 Setup Course'. The form has a menu bar with 'Action', 'Edit', 'Query', 'Block', 'Record', 'Field', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons. The main area of the form is titled 'หลักสูตร' and contains several text input fields:

- ชื่อหลักสูตร (ภาษาไทย): สาขาเทคโนโลยีสารสนเทศ
- ชื่อหลักสูตร (ภาษาอังกฤษ): Master of Science Program in Information Technology
- ชื่อปริญญา (เต็ม) (ภาษาไทย): วิทยาศาสตร สาขาเทคโนโลยีสารสนเทศ
- ชื่อปริญญา (เต็ม) (ภาษาอังกฤษ): Master of Science (Information Technology)
- ชื่อย่อ (ภาษาไทย): ท.ม. (ศาสตรเทคโนโลยีสารสนเทศ)
- ชื่อย่อ (ภาษาอังกฤษ): M.Sc. (Information Technology)

At the bottom of the window, there is a status bar with the text 'FRM-41008: Undefined function key. Press Ctrl+F1 for list of valid keys.' and 'Record: 1/1'.

รูปที่ 5.2 หน้าจอสำหรับแสดงข้อมูลหลักสูตรการเรียน

โปรแกรมการกำหนดหลักสูตร เป็นส่วนที่ถูกกำหนดโดยเจ้าหน้าที่ฝ่ายวิชาการ

### 5.1.2 ขั้นตอนการกำหนดแขนงวิชา

การกำหนดรายชื่อแขนงวิชา หรือกำหนดแขนงวิชาที่เปิดทำการเรียนการสอน สำหรับหลักสูตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ ( คณะเทคโนโลยีสารสนเทศ ) ดังรูปที่ 5.3 ซึ่งกำหนดให้มี 2 แขนงวิชาคือ

1.แขนงวิชาวิทยาการสารสนเทศ

2.แขนงวิชาการจัดการเทคโนโลยีสารสนเทศ

โปรแกรมการกำหนดแขนงวิชา เป็นส่วนที่ถูกกำหนดโดยเจ้าหน้าที่ฝ่ายวิชาการ

Oracle Forms Runtime

Action Edit Query Block Record Field Window Help

FRM011 Setup Major

แขนงวิชา

ชื่อแขนงวิชา วิทยาการสารสนเทศ วิทยาการสารสนเทศ สาขาเทคโนโลยีสารสนเทศ  
(ภาษาไทย)

ชื่อแขนงวิชา IS Information science  
(ภาษาอังกฤษ)

EXIT

Record: 1/?

รูปที่ 5.3 หน้าจอสำหรับแสดงข้อมูลแขนงวิชา

### 5.1.3. ขั้นตอนการกำหนดแผนการศึกษา

การกำหนดรายชื่อแผนการศึกษา สำหรับหลักสูตรมหาบัณฑิตนี้ดังรูปที่ 5.4 มี 2 แผนการศึกษา ดังนี้คือ

1. แผน ก มีวิทยานิพนธ์
2. แผน ข ไม่มีวิทยานิพนธ์

Oracle Forms Runtime

Action Edit Query Block Record Field Window Help

FRM013 Setup Thesis Plan

ชื่อแผนการเรียน แผน ก มีวิทยานิพนธ์  
(ภาษาอังกฤษ) Thesis

EXIT

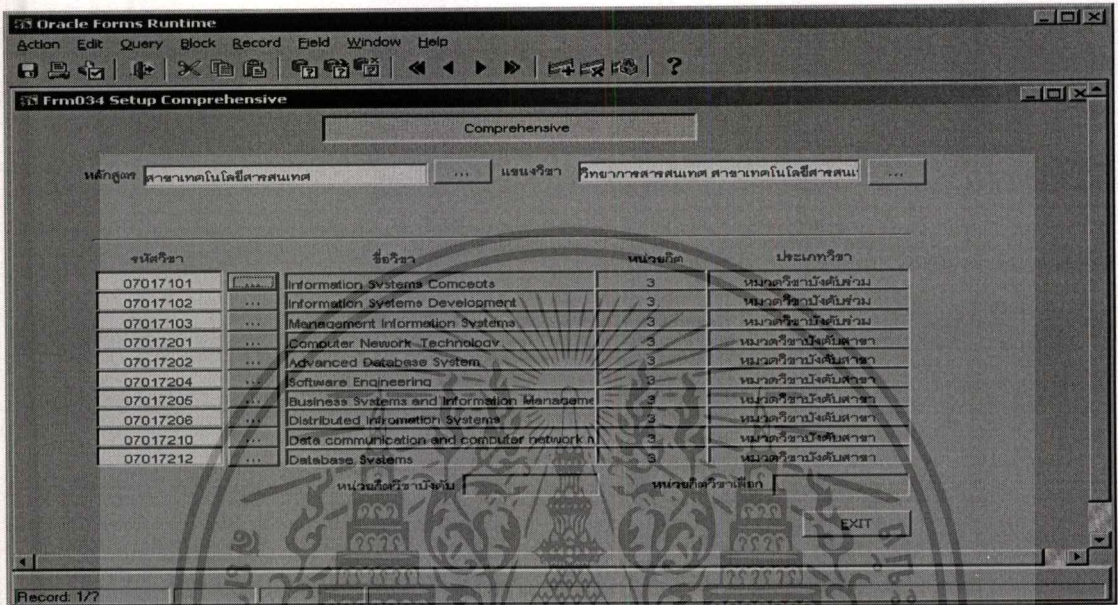
Record: 1/?

รูปที่ 5.4 หน้าจอสำหรับแสดงข้อมูลแผนวิทยานิพนธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.1.4. ขั้นตอนการกำหนดรายละเอียดสำหรับการสอบ Comprehensive

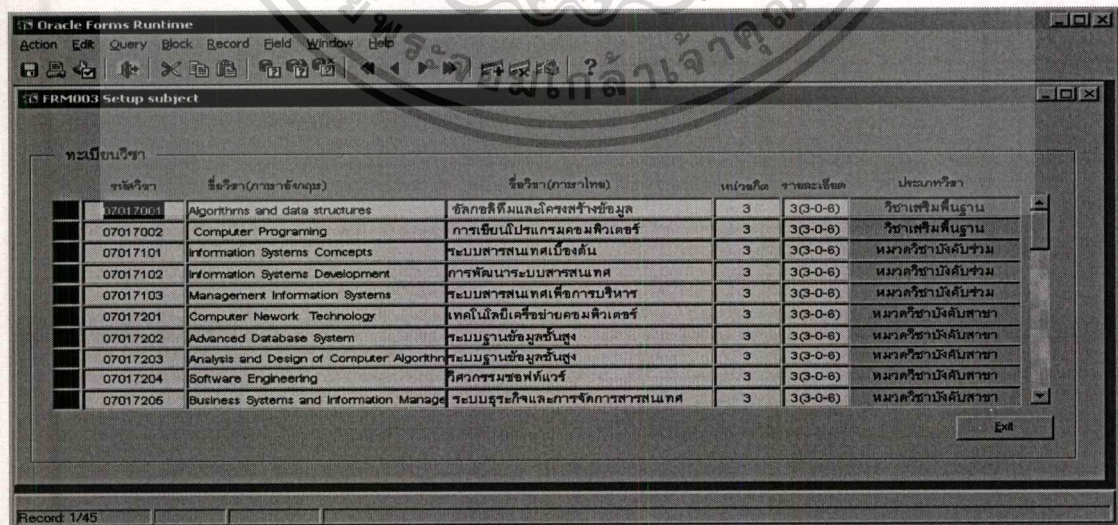
การกำหนดรายละเอียดของรายวิชาสำหรับเป็นเงื่อนไขในการตรวจสอบสิทธิผู้ที่สามารถสอบวิชา Comprehensive ได้นั้นแสดงได้ดังรูปที่ 5.5



รูปที่ 5.5 หน้าจอสำหรับกำหนดเงื่อนไขการสอบวิชา Comprehensive

### 5.1.5. ขั้นตอนการกำหนดรายชื้อวิชา

กำหนดรายชื้อวิชาที่มีกำหนดไว้ในแผนการศึกษา ดังรูปที่ 5.6



รูปที่ 5.6 หน้าจอสำหรับแสดงข้อมูลวิชาเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ความหมายของรหัสวิชา

รหัสวิชาที่ใช้ถูกกำหนดให้เป็นเลข 8 หลักคือ

รหัสตัวที่ 1 และ 2 อาทิเช่นเลข 07 หมายถึง คณะเทคโนโลยีสารสนเทศ

รหัสตัวที่ 3 และ 4 อาทิเช่นเลข 01 หมายถึง สาขาวิชาเทคโนโลยีสารสนเทศ

รหัสตัวที่ 5 อาทิเช่นเลข 7 หมายถึง ปริญญาโท

รหัสตัวที่ 6 ได้แก่

0 หมายถึง วิชาเสริมพื้นฐาน

1 หมายถึง หมวดวิชาบังคับร่วม

2 หมายถึง หมวดวิชาบังคับสาขา

3 หมายถึง หมวดวิชาเลือก

4 หมายถึง วิทยานิพนธ์

7 และ 8 หมายถึง ลำดับที่วิชา



### 5.1.6 ขั้นตอนการกำหนดโครงสร้างหลักสูตร

การกำหนดโครงสร้างหลักสูตรการเรียนดังรูปที่ 5.7 โดยการกำหนดโครงสร้างหลักสูตรการเรียนจะถูกแบ่งตาม หลักสูตร แขนงวิชา (วิทยาการสารสนเทศและ การจัดการสารสนเทศ) และ แผนการศึกษา(แผน ก แผน ข )ซึ่งการกำหนดโครงสร้างหลักสูตรการเรียนนี้จะถูกนำไปใช้เพื่อเป็นการกำหนดเงื่อนไขในการลงทะเบียนเรียนสำหรับนักศึกษาในแต่ละภาคการศึกษา และยังใช้เพื่อเป็นเงื่อนไขในการตรวจสอบการจบการศึกษาของนักศึกษา

ลำดับ	รหัสวิชา	ชื่อภาษาอังกฤษ	ชื่อภาษาไทย	ประเภทวิชา	หน่วยกิต
1	07017 101	Information Systems/ Concepts	ระบบสารสนเทศเบื้องต้น	หมวดวิชาบังคับร่วม	3
2	07017 102	Information Systems/ Development	การพัฒนาสารสนเทศ	หมวดวิชาบังคับร่วม	3
3	07017 103	Management Information Systems	ระบบสารสนเทศเพื่อการบริหาร	หมวดวิชาบังคับร่วม	3
4	07017 201	Computer Network Technology	เทคโนโลยีเครือข่ายคอมพิวเตอร์	หมวดวิชาบังคับสาขา	3
5	07017 202	Advanced Database System	ระบบฐานข้อมูลขั้นสูง	หมวดวิชาบังคับสาขา	3
6	07017 203	Analysis and Design of Computer Algorithm	ระบบฐานข้อมูลขั้นสูง	หมวดวิชาบังคับสาขา	3
7	07017 208	Seminar 1	สัมมนา 1	หมวดวิชาบังคับสาขา	1
8	07017 209	Seminar 2	สัมมนา 2	หมวดวิชาบังคับสาขา	1
9	07017 401	Thesis 1	วิทยานิพนธ์ 1	หมวดวิชาวิทยานิพนธ์	6
10	07017 402	Thesis 2	วิทยานิพนธ์ 2	หมวดวิชาวิทยานิพนธ์	6

รูปที่ 5.7 หน้าจอสำหรับแสดงข้อมูลหลักสูตรการเรียน

ลักษณะข้อมูลส่วนต้นประกอบด้วย หลักสูตรการเรียน แขนงวิชา ปีการศึกษา ภาคการศึกษา จำนวนหน่วยกิตรวมในแต่ละหลักสูตร จำนวนหน่วยกิตวิชาบังคับร่วม จำนวนหน่วยกิตวิชาบังคับสาขา จำนวนหน่วยกิตวิชาเลือก จำนวนหน่วยกิตปัญหาพิเศษ

ลักษณะข้อมูลส่วนรายละเอียดประกอบด้วย รหัสวิชา ชื่อวิชาภาษาไทย ชื่อวิชาภาษาอังกฤษ ประเภทวิชาและหน่วยกิตการเรียนในแต่ละรายวิชา

### 5.1.7. ขั้นตอนการกำหนดหลักสูตรสำหรับการลงทะเบียน

การกำหนดหลักสูตรรายวิชาที่เปิดลงทะเบียนในแต่ละภาคการศึกษาดังรูปที่ 5.8 เป็นข้อมูลที่จะต้องถูกกำหนดไว้ทุกครั้งในแต่ละภาคการศึกษาก่อนที่จะเปิดการลงทะเบียนเรียน เพื่อเป็นข้อมูลสำหรับการตรวจสอบเงื่อนไขการลงทะเบียนของนักศึกษา ซึ่งก็คือนักศึกษจะสามารถลงทะเบียนได้เฉพาะรายวิชาที่กำหนดไว้แล้วเท่านั้น

The screenshot shows an Oracle Forms Runtime window titled 'FRM004 Set Class'. It contains a form with several input fields and a table. The form fields include 'หลักสูตร' (Course), 'ปีการศึกษา' (Academic Year) set to 2543, 'ภาคการศึกษา' (Semester) set to 1, and 'หมวดวิชา' (Subject) set to 'หมวดวิชา บัญชี'. The table below lists 14 courses with their respective details.

ลำดับ	รหัสวิชา	ชื่อวิชา	จำนวน ห้องเรียน	จำนวน นักศึกษาในห้อง	หมวดวิชา	หน่วยกิต
1	07017101	Information Systems Concepts	4	40	หมวดวิชา บัญชีทั่วไป	3
2	07017102	Information Systems Development	4	40	หมวดวิชา บัญชีทั่วไป	3
3	07017103	Management Information Systems	4	40	หมวดวิชา บัญชีทั่วไป	3
4	07017201	Computer Network Technology	2	40	หมวดวิชา บัญชีสาขา	3
5	07017202	Advanced Database System	2	40	หมวดวิชา บัญชีสาขา	3
6	07017203	Analysis and Design of Computer Algorithms	2	40	หมวดวิชา บัญชีสาขา	3
7	07017204	Software Engineering	2	40	หมวดวิชา บัญชีสาขา	3
8	07017205	Business Systems and Information Management	2	40	หมวดวิชา บัญชีสาขา	3
9	07017206	Distributed Information Systems	2	40	หมวดวิชา บัญชีสาขา	3
10	07017207	Systems Development project	2	40	หมวดวิชา บัญชีสาขา	3
11	07017208	Seminar1	2	40	หมวดวิชา บัญชีสาขา	1
12	07017209	Seminar2	2	40	หมวดวิชา บัญชีสาขา	1
13	07017210	Data communication and computer network	2	40	หมวดวิชา บัญชีสาขา	3
14	07017217	Seminar1	2	40	หมวดวิชา บัญชีสาขา	1

รูปที่ 5.8 หน้าจอสำหรับแสดงข้อมูลหลักสูตรสำหรับการลงทะเบียนเรียน

ลักษณะข้อมูลส่วนต้นประกอบด้วย หลักสูตรการเรียน ปีการศึกษาที่เปิดลงทะเบียน ภาคการศึกษาที่เปิดลงทะเบียน

ลักษณะข้อมูลส่วนรายละเอียดประกอบด้วย รหัสวิชา ชื่อวิชา ห้องเรียน จำนวนนักศึกษาในแต่ละห้อง ประเภทวิชา จำนวนหน่วยกิตการเรียนสำหรับแต่ละรายวิชา

### 5.1.8 ขั้นตอนการลงทะเบียน

การลงทะเบียนเรียนสำหรับนักศึกษาในแต่ภาคการศึกษาแสดงดังรูปที่ 5.9 ถูกใช้โดยเจ้าหน้าที่ฝ่ายทะเบียนนักศึกษา

ลำดับ	รหัสวิชา	ชื่อวิชา	ประเภทวิชา	หน่วยกิต	ห้องเรียน
1	07017101	Information Systems Concepts	หมวดวิชาบังคับร่วม	3	2
2	07017102	Information Systems Development	หมวดวิชาบังคับร่วม	3	2
3	07017201	Computer Network Technology	หมวดวิชาบังคับสาขา	3	2

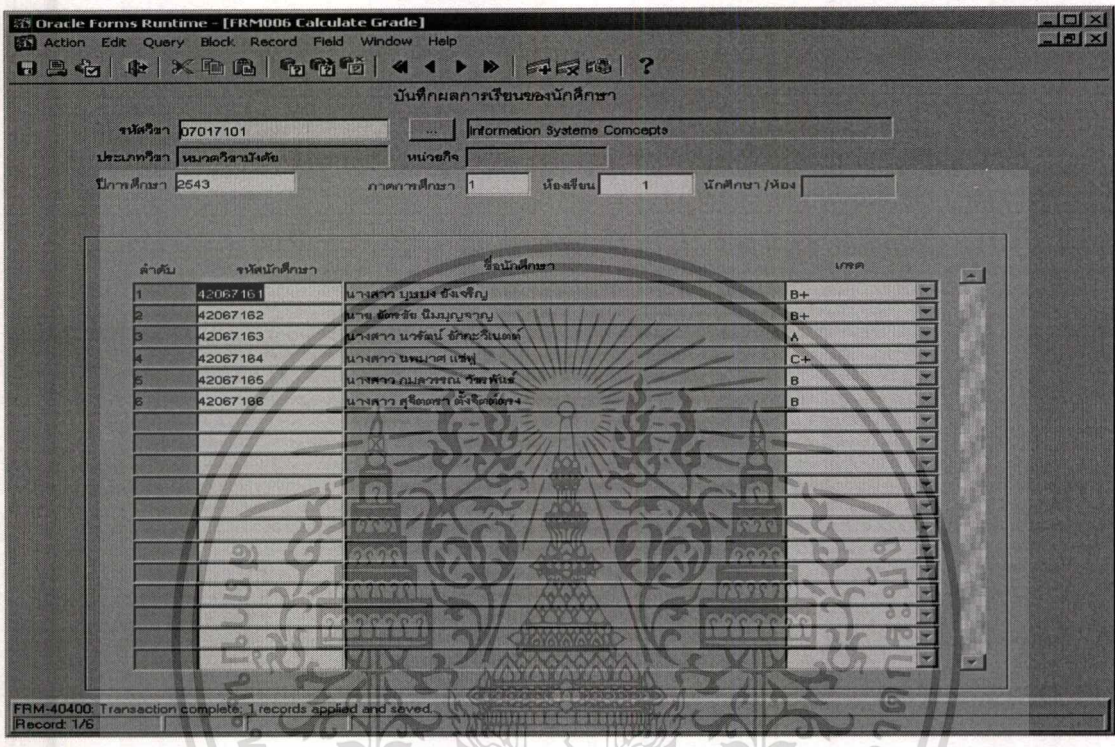
รูปที่ 5.9 หน้าจอแสดงข้อมูลการลงทะเบียนของนักศึกษา

ลักษณะข้อมูลส่วนต้นประกอบด้วย ข้อมูลรหัสนักศึกษา ปีการศึกษาและภาคการศึกษาที่ลงทะเบียนเรียน เมื่อมีการกรอกรหัสนักศึกษาโปรแกรมจะแสดงข้อมูลรายละเอียดของนักศึกษา อาทิเช่น ชื่อนักศึกษา หลักสูตร(สาขาวิชา) ประเภทของนักศึกษา(ปกติ/สมทบ) ห้องเรียน(ถ้ามี) กลุ่มของนักศึกษา แผนการเรียน(วิทยานิพนธ์/โปรเจก) จากนั้นก็จะต้องกำหนดปีการศึกษาและภาคการศึกษาที่เปิดการลงทะเบียน

ลักษณะข้อมูลส่วนรายละเอียดประกอบด้วย รหัสวิชา คือรหัสวิชาที่นักศึกษาต้องการลงทะเบียน เมื่อนักศึกษาเลือกรหัสวิชาที่ลงทะเบียน โปรแกรมจะตรวจเช็ครหัสวิชากับข้อมูลรายวิชา (หลักสูตรการเรียน)ที่เปิดการเรียนในแต่ละภาคการศึกษา พร้อมทั้งตรวจเช็คด้วยว่านักศึกษามีสิทธิลงทะเบียนในรายวิชานั้นหรือไม่ ถ้าผลการตรวจเช็คถูกต้องตามเงื่อนไขที่กำหนดไว้ในโปรแกรม โปรแกรมก็จะแสดงรายละเอียด ชื่อวิชาภาษาอังกฤษ ประเภทวิชา หน่วยกิตสำหรับแต่ละรายวิชา ห้องเรียนที่นักศึกษาลงทะเบียนเรียน(สำหรับรายวิชาที่เปิดให้มีการเรียนการสอนมากกว่า 1 ห้องเรียน)

### 5.1.9.ขั้นตอนการประมวลผลการเรียนของนักศึกษาแยกตามรหัสวิชา

การประมวลผลการเรียนของนักศึกษา(ตามรายวิชาที่เปิดลงทะเบียน) ดังรูปที่ 5.10



รูปที่ 5.10 หน้าจอแสดงข้อมูลผลการเรียนของนักศึกษาเป็นรายวิชา

โปรแกรมการประมวลผลการเรียนของนักศึกษานี้ถูกใช้โดยเจ้าหน้าที่ฝ่ายวิชาการ เพื่อประมวลผลการเรียนของนักศึกษาในแต่ละรายวิชา

ลักษณะข้อมูลส่วนต้นกำหนดให้มีการแสดงผลโดยกำหนดเงื่อนไขดังนี้คือผู้ใช้จะต้องระบุรหัสวิชา ปีการศึกษาและภาคการศึกษาที่ลงทะเบียน และห้องเรียนที่เปิดการเรียนการสอนในแต่ละภาคการศึกษา

ลักษณะข้อมูลส่วนรายละเอียดประกอบด้วย รหัสนักศึกษา ชื่อนักศึกษา(ภาษาไทย) พร้อมกับเกรดที่ได้ซึ่งสามารถเปรียบเทียบระดับคะแนนได้ดังนี้คือ

ค่าระดับคะแนน	แต้ม
A	4.0
B+	3.5
B	3.0
C+	2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

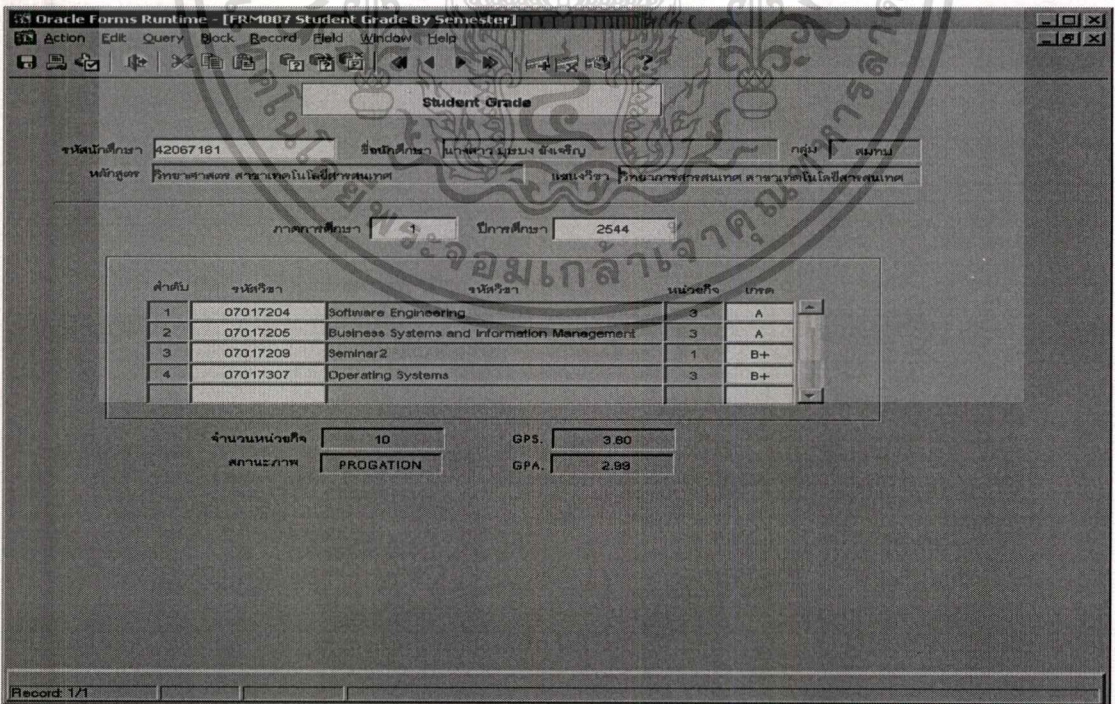
C	2.0
D+	1.5
D	1.0
F	0
W	ขอถอนวิชาเรียนภายในกำหนด
I	ไม่สมบูรณ์
S	พอใจ
U	ไม่พอใจ

สำหรับการวัดผลวิทยานิพนธ์

O	ดีเยี่ยม
G	ดี
P	ผ่าน
F	ไม่ผ่าน

5.1.10. ขั้นตอนการตรวจสอบผลการเรียนของนักศึกษา

การแสดงผลการเรียนของนักศึกษาดังรูปที่ 5.11



รูปที่ 5.11 หน้าจอสำหรับแสดงข้อมูลผลการเรียนของนักศึกษาเป็นรายภาคการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการประมวลผลการเรียนของนักศึกษาถูกใช้โดยเจ้าหน้าที่ฝ่ายวิชาการ และเจ้าหน้าที่ฝ่ายทะเบียนนักศึกษา เพื่อตรวจสอบผลการเรียนของนักศึกษาเป็นรายบุคคลตามรหัสนักศึกษา โดยส่วนแสดงผลนี้ไม่อนุญาตให้ผู้ใช้งานเข้าไปแก้ไขข้อมูลใดๆ

ลักษณะข้อมูลส่วนต้นประกอบด้วย รหัสนักศึกษา ชื่อนักศึกษา ประเภทของนักศึกษา (ปกติ/สมทบ) หลักสูตรการเรียนและแขนงวิชาของนักศึกษา ปีการศึกษาและภาคการศึกษาที่ต้องการตรวจสอบผลการเรียน

ลักษณะข้อมูลส่วนรายละเอียดประกอบด้วย รหัสวิชา ชื่อวิชาภาษาอังกฤษ หน่วยกิตรายวิชา และเกรดเฉลี่ยที่ได้รับ รวมถึงเกรดเฉลี่ยรายภาคการศึกษา GPS และ เกรดเฉลี่ยรวม GPA สถานะภาพและหน่วยกิตสะสม

ลักษณะการใช้งานผู้ใช้ต้องใส่ข้อมูลรหัสนักศึกษา ปีการศึกษาและภาคการศึกษาที่ต้องการตรวจสอบผลการเรียน จากนั้น โปรแกรมจะทำการแสดงข้อมูลรายละเอียดของแต่ละรายวิชาที่นักศึกษาลงทะเบียนในภาคการศึกษานั้นพร้อมทั้งเกรดที่ได้และแสดงข้อมูลเกรดเฉลี่ยของนักศึกษาในภาคการศึกษานั้น

เทคนิคอย่างหนึ่งที่ใช้สำหรับการออกแบบส่วนแสดงผลข้อมูล คือ การนำ View มาประยุกต์ใช้กับการแสดงผลการเรียนและเกรดเฉลี่ยของนักศึกษา ซึ่งมีรายละเอียดดังนี้คือ

```
Create view v_registration_grade as
select s.std_course_id,
       c.course_name_thai,
       s.std_major_id,
       m.major_name_thai,
       t.std_id,
       rtrim(x.prefix_name)||' '|rtrim(s.std_firstname)||' '|rtrim(s.std_lastname)
       std_name,
       t.years,
       t.semester,
       s.section_id,
       t.std_grade,
       t.subject_code,
       p.subject_name_eng,
       p.unit,
       t.room_no,
       s.std_group,
       (decode(t.std_grade, 'A',4,
              'B+',3.5,
              'B',3,
              'C+',2.5,
              'C',2,
              'D+',1.5,
              'D',1,
              'F',0,
              'W',0,
              'P',0,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

'S',0) * p.UNIT) grade
from t_registration t,
     t_student       s,
     t_subject       p,
     t_prefix        x,
     t_course        c,
     t_major         m
where t.std_id       = s.std_id
and   s.std_prefix  = x.prefix_id
and   s.std_course_id = c.course_id
and   t.subject_code = p.subject_code
and   s.std_major_id = m.major_id ;

```

### 5.1.11. ขั้นตอนการตรวจสอบผลการเรียนตามภาคการศึกษา

การแสดงผลการเรียนของนักศึกษาตามภาคการศึกษาที่ลงทะเบียนเรียนดังรูปที่ 5.12

ลำดับ	ชื่อนักศึกษา	วิชา	หน่วยกิต	เฉลี่ย	สถานะ
1	42067161	นางสาว บุญบง ฉิ่งขวัญ	9	9.87	PASS
2	42067162	นาย ชัยสิทธิ์ นิยมบุญจาม	9	9.67	PASS
3	42067163	นางสาว นวรัตน์ อักกะวีเนตต์	9	3.51	PASS
4	42067164	นางสาว นนมาศ แซ่ฟู่	8	3	PASS
5	42067165	นางสาว กมลวรรณ วัชรพิณ	9	2.83	PROGATION
6	42067168	นางสาว สุจิตตรา ตั้งจิตตพงษ์	9	9.33	PASS

รูปที่ 5.12 หน้าจอแสดงข้อมูลผลการเรียนของนักศึกษาแยกตามรุ่น

โปรแกรมการตรวจสอบผลการเรียนตามภาคการศึกษานี้ถูกใช้โดยเจ้าหน้าที่ฝ่ายทะเบียนและเจ้าหน้าที่ฝ่ายวิชาการเพื่อตรวจสอบความถูกต้องของผลการเรียน(เกรดเฉลี่ย)ของนักศึกษา ซึ่งผู้ใช้สามารถเข้าไปตรวจสอบได้เท่านั้นแต่ไม่มีสิทธิ์เข้าไปแก้ไขข้อมูลดังกล่าวได้

ลักษณะข้อมูลส่วนต้นประกอบด้วย หลักสูตร แขนงวิชา ปีการศึกษาและภาคการศึกษาที่ลงทะเบียนเรียน รุ่น (Is8.2)

ลักษณะข้อมูลส่วนรายละเอียดประกอบด้วย รหัสนักศึกษา ชื่อนักศึกษา หน่วยกิตในแต่ละภาคการศึกษา เกรดเฉลี่ยของนักศึกษา สถานะภาพการศึกษาของนักศึกษา

View ที่ใช้สำหรับการออกแบบส่วนแสดงผลการเรียนและเกรดเฉลี่ยของนักศึกษา ตามในแต่ละห้องเรียนมีรายละเอียดดังนี้คือ

```

Create view v_class_grade          as
select
  s.std_course_id,
  s.std_major_id,
  t.std_id,
  rtrim(p.prefix_name)||' '||rtrim(s.std_firstname)||' '||rtrim(s.std_lastname)
  std_name,
  t.years,
  t.semester,
  s.std_group,
  sum(x.unit) total_unit,
  sum(decode(t.std_grade,'A',4,
             'B+',3.5,
             'B',3,
             'C+',2.5,
             'C',2,
             'D+',1.5,
             'D',1,
             'F',0,
             'W',0,
             'P',0,
             'S',0) * x.UNIT) avg_grade
from t_registration
     t,
     s,
     t_prefix
     p,
     t_subject
     x
where t.std_id      = s.std_id and
t.subject_code     = x.subject_code and
s.std_prefix       = p.prefix_id and
t.std_grade        Not in ('W','I','S','U','P','')
-- nvl(t.std_grade,'F') NOT IN ('W','I','S','U','P','')
group by
  s.std_course_id,
  s.std_major_id,
  t.std_id,
  rtrim(p.prefix_name)||' '||rtrim(s.std_firstname)||' '||rtrim(s.std_lastname) ,
  t.years,
  t.semester,
  s.std_group;

```

5.1.12. ขั้นตอนการตรวจสอบประวัติการลงทะเบียนเรียนของนักศึกษา  
 การแสดงประวัติการลงทะเบียนเรียนของนักศึกษาดังรูปที่ 5.13

Oracle Forms Runtime - [FRM009 Student Grade]

Student Grade

รหัสนักศึกษา: 42067161 ชื่อนักศึกษา: นางสาว นุชมา ช่างพิมพ์ แผน: มนชน.

หลักสูตร: ศึกษาศาสตร สาขาเทคโนโลยีสารสนเทศ แขนงวิชา: วิทยาการสารสนเทศ สาขาเทคโนโลยีสารสนเทศ

สถานะภาพ: PROGATION จำนวนหน่วยกิต: 36 GPS: 2.99

ผลการสอบ

จำนวนครั้งที่สอบ Comprehensive:  Comprehensive:

จำนวนครั้งที่สอบ Oral:  Oral:

Thesis/Project:

ลำดับ	ภาคปีการศึกษา	รหัสวิชา	ชื่อวิชา	หน่วยกิต	เกรด
1	1/2543	07017101	Information Systems Concepts	3	B+
2	1/2543	07017102	Information Systems Development	3	A
3	1/2543	07017201	Computer Network Technology	3	B+
4	2/2543	07017103	Management Information Systems	3	B+
5	2/2543	07017202	Advanced Database System	3	A
6	2/2543	07017208	Seminar 1	1	B+
7	2/2543	07017312	Client /Server System Development	3	C+
8	1/2544	07017204	Software Engineering	3	A
9	1/2544	07017205		3	A
10	1/2544	07017208	Seminar 2	1	B+
11	2/2544	07017206	Distributed Information Systems	3	
12	2/2544	07017207	Systems Development project	3	W

Record: 1/7

รูปที่ 5.13 หน้าจอแสดงข้อมูลประวัติการลงทะเบียนเรียนของนักศึกษา

โปรแกรมการตรวจสอบประวัติการลงทะเบียนเรียนของนักศึกษานี้ถูกใช้โดยเจ้าหน้าที่ฝ่ายทะเบียนและเจ้าหน้าที่ฝ่ายวิชาการ เพื่อแสดงสถานะภาพของนักศึกษา แสดงข้อมูลการสอบ Comprehensive รวมถึงการตรวจสอบข้อมูลรายวิชาที่นักศึกษาลงทะเบียนเรียนเพื่อพิมพ์ Transcript

ลักษณะข้อมูลส่วนต้นประกอบด้วยรหัสนักศึกษา และรายละเอียดเกี่ยวกับนักศึกษา อาทิ ชื่อนักศึกษา แผนการศึกษา หลักสูตร แขนงวิชา สถานะภาพของนักศึกษา จำนวนหน่วยกิตสะสม และ GPA

ลักษณะข้อมูลส่วนรายละเอียดประกอบด้วย ปีการศึกษาที่ลงทะเบียนเรียน รหัสวิชาที่ลงทะเบียน ชื่อวิชา จำนวนหน่วยกิต และเกรดที่ได้ในแต่ละรายวิชา

ลักษณะการใช้งานโปรแกรม ผู้ใช้ใส่ข้อมูลรหัสนักศึกษา จากนั้นโปรแกรมที่ทำการแสดงผลการศึกษาของนักศึกษาทั้งหมดมาให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.1.13. ขั้นตอนการลงทะเบียนนักศึกษา

ข้อมูลทะเบียนนักศึกษาแสดงดังรูปที่ 5.14

The screenshot shows the Oracle Forms Runtime interface for a student registration form. The form is titled "ทะเบียนนักศึกษา" (Student Registration). The fields and their values are as follows:

- รหัสนักศึกษา (Student ID): 42067161
- ชื่อ (Name): บุษบง
- นามสกุล (Surname): ด้งจัญญ์
- ชื่อ(ภาษาอังกฤษ) (English Name): Boodsabong
- นามสกุล(ภาษาอังกฤษ) (English Surname): Youngcharoen
- เพศ (Gender): Female
- ตำแหน่ง (Position): นางสาว (Miss)
- วันเกิด (Date of Birth): 14 มี.ค. 2001 (DDMMYYYY)
- หลักสูตร (Course): วิทยาศาสตร์ สาขาเทคโนโลยีสารสนเทศ (Science, Information Technology)
- สาขาวิชา (Department): วิทยาศาสตร์ สาขาเทคโนโลยีสารสนเทศ (Science, Information Technology)
- แผนกการศึกษา (Faculty): มนน ๑.ไม่มีวิทยาพิเศษ (Faculty 1, Specialized)
- ภาค (Semester): 2
- รุ่น (Year): 1
- ปี (Year): ๒๕๕๒
- เข้ารับการศึกษ (Enrollment Number): 2543
- ภาค (Semester): 1
- ทำเรื่องการศึกษา (Application Number):
- E\_mail: Boodsabong@yahoo.com

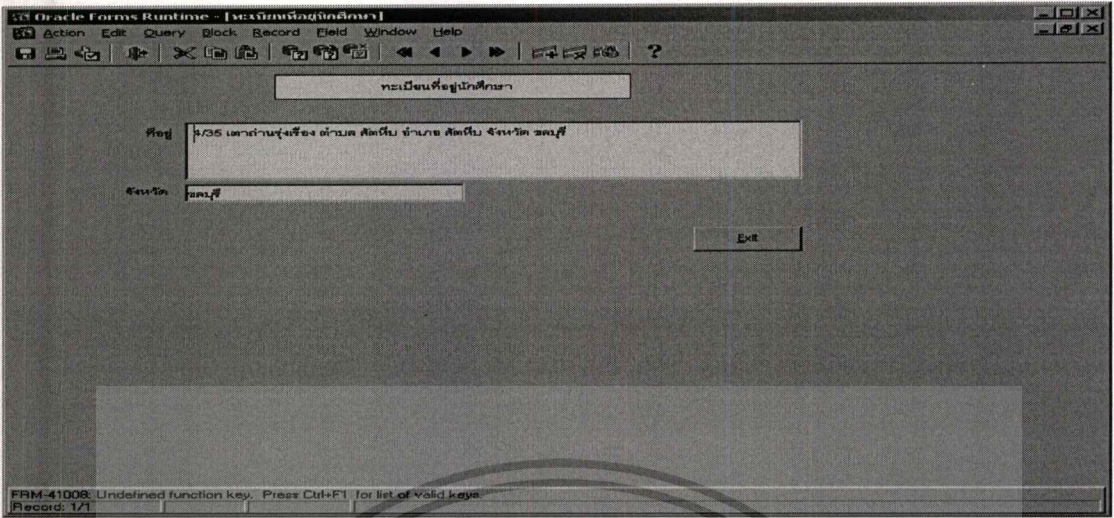
Buttons for "Address", "Phone", and "EXIT" are visible at the bottom right. The status bar at the bottom left shows "Record: 1/7".

รูปที่ 5.14 หน้าจอแสดงข้อมูลทะเบียนนักศึกษา

โปรแกรมทะเบียนนักศึกษาถูกใช้โดยเจ้าหน้าที่ฝ่ายทะเบียน เพื่อเป็นการสร้างฐานข้อมูลนักศึกษาและกำหนดรายละเอียดอื่นๆที่เกี่ยวกับนักศึกษา

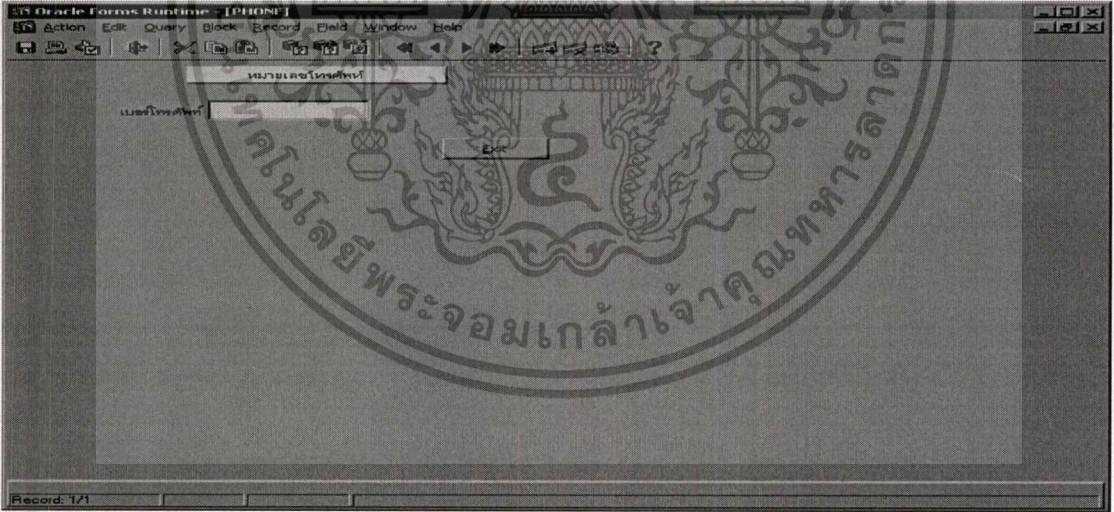
ขั้นตอนการดำเนินงาน เจ้าหน้าที่ฝ่ายทะเบียนจะทำการกรอกข้อมูลรหัสนักศึกษา พร้อมกับรายละเอียดอื่นๆ ที่ได้จากส่วนงานบัณฑิตศึกษาประจำสถาบันซึ่งเป็นหน่วยงานกลางที่เก็บข้อมูลนักศึกษาทั้งหมดและเป็นผู้กำหนดแผนการเรียนการสอนให้กับทางคณะ

โปรแกรมทะเบียนที่อยู่นักศึกษาเป็นส่วนที่เชื่อมโยงกับข้อมูลทะเบียนนักศึกษา สำหรับเป็นฐานข้อมูลทะเบียนที่อยู่นักศึกษาแสดงดังรูปที่ 5.15



รูปที่ 5.15 หน้าจอสำหรับแสดงข้อมูลที่อยู่นักศึกษา

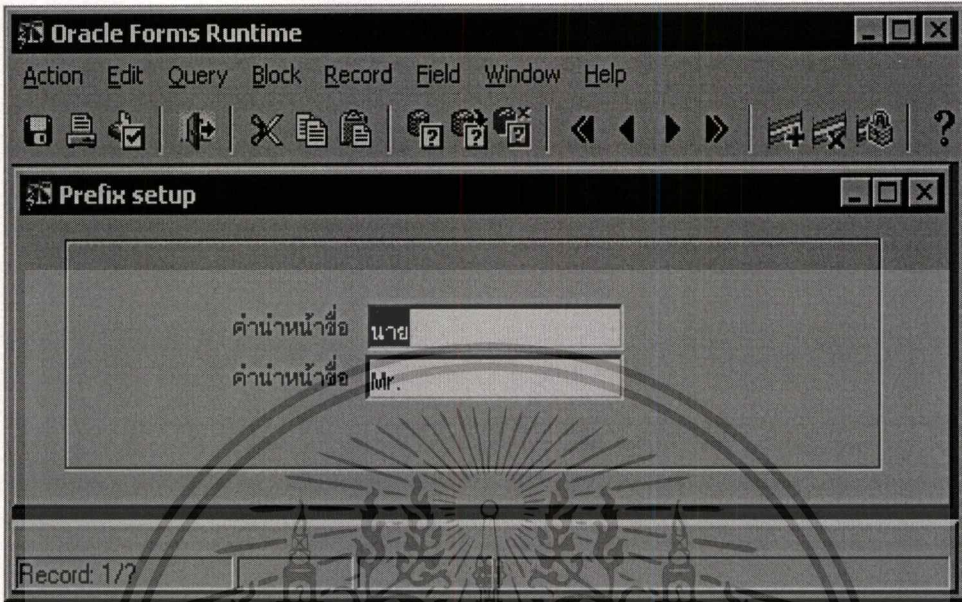
โปรแกรมทะเบียนเบอร์โทรศัพท์นักศึกษาเป็นส่วนที่เชื่อมโยงกับข้อมูลทะเบียนนักศึกษา สำหรับเป็นฐานข้อมูลทะเบียนเบอร์โทรศัพท์ ลักษณะของฐานข้อมูลทะเบียนเบอร์โทรศัพท์นักศึกษาแสดงดังรูปที่ 5.16



รูปที่ 5.16 หน้าจอสำหรับแสดงข้อมูลเบอร์โทรศัพท์นักศึกษา

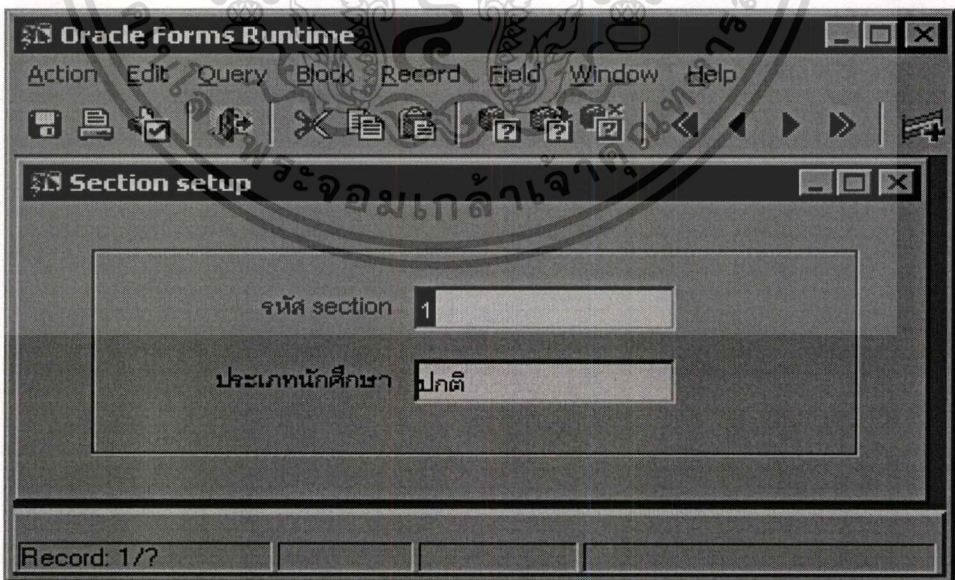
#### 5.1.14. ขั้นตอนการกำหนดพารามิเตอร์ที่สำคัญของระบบ

ข้อมูลค่านำหน้าชื่อ(ศัพท์นาม) แสดงดังรูปที่ 5.17



รูปที่ 5.17 หน้าจอสำหรับแสดงข้อมูลค่านำหน้าชื่อ

ข้อมูลกำหนดความหมายของรหัสและประเภทของนักศึกษา ดังแสดงดังภาพที่ 5.18



รูปที่ 5.18 หน้าจอสำหรับแสดงข้อมูลประเภทของนักศึกษา

## Design Output

5.1.15 หน้าจอสำหรับพิมพ์รายงานแบบฟอร์มการลงทะเบียน ซึ่งผู้ใช้งานจะต้องใส่ข้อมูล รหัส นักศึกษา ปีการศึกษาและภาคการศึกษาที่เปิดลงทะเบียน

The screenshot displays the Oracle Forms Runtime window for 'FRM026 Print Registration Form'. The form is titled 'Registrations Form' and contains three input fields: 'รหัสนักศึกษา' (Student ID) with value 42067161, 'ปีการศึกษา' (Academic Year) with value 2545, and 'ภาคการศึกษา' (Semester) with value 1. Below the input fields are two buttons: 'Print' and 'Exit'. The status bar at the bottom indicates 'Record: 1/1'.

รูปที่ 5.19 หน้าจอสำหรับพิมพ์ Registration Form

รายงานแสดงข้อมูลการลงทะเบียนของนักศึกษา(สำหรับแต่ละภาคการศึกษา)

Rpt001 Registration		Date :dd/mm/yyyy		
King Mongkut's Institute of Technology Landkrabong				
Student : xxxxx(30)xxxxx	ID : 99999999	Semester/Year 9/yyyy		
Faculty : xxxxx(30)xxxxx	Degree : xxx(15)xxx	Program : xxx(30)xxx		
Course : xxxxx(30)xxxxx	Date : dd/mm/yyyy	No.		
Registration				
Code	Description	Sec	Type	Credit
99999999	xxxxx(40)xxxxx	xx		9
Total Credits				99

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

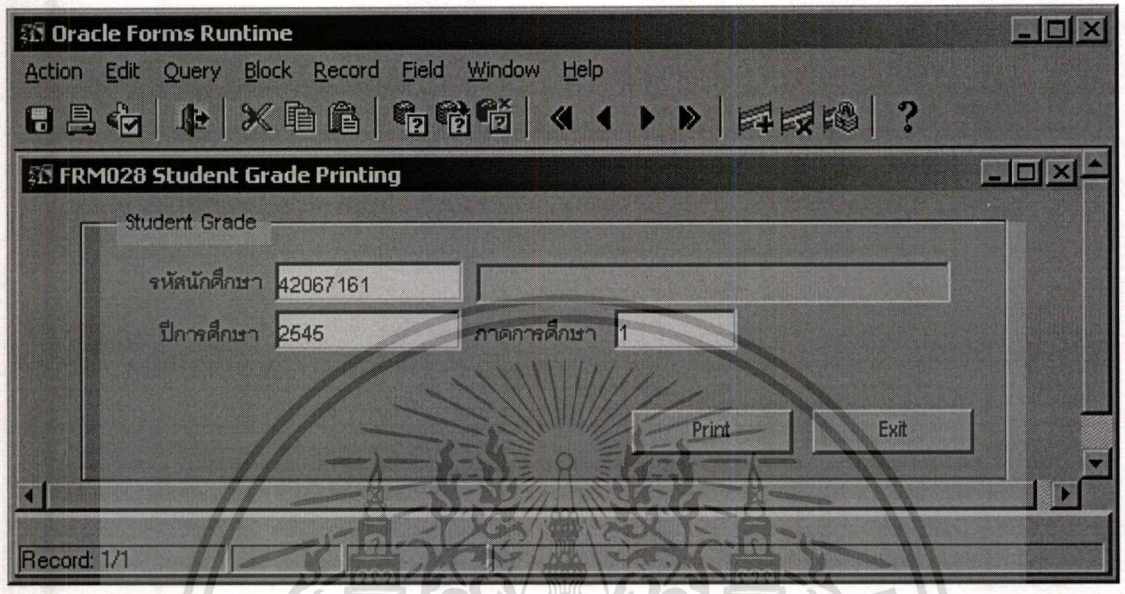
5.1.16 หน้าจอสำหรับพิมพ์รายงานเกรดของนักศึกษาเป็นรายวิชา ซึ่งผู้ใช้จะต้องใส่ข้อมูล รหัสวิชา และปีการศึกษาที่เปิดลงทะเบียน

รูปที่ 5.20 หน้าจอสำหรับพิมพ์ผลการเรียนของนักศึกษาแยกตามรายวิชา

รายงานแสดงข้อมูลผลการเรียนของนักศึกษา(ตามรายวิชา)

Rpt002 Subject Grade		Date :dd/mm/yyyy		
King Mongkut's Institute of Technology Landkrabong				
Subject : xxxxx(40)xxxxx	Subject Code : 99999999	Semester/Year 9/yyyy		
Faculty : xxxxx(30)xxxxx	Degree : xxx(15)xxx	Program : xxx(30)xxx		
Subject Grade				
ID	Student	Sec	Credit	Grade
99999999	xxxxx(40)xxxxx	xx	9	x

5.1.17 หน้าจอสำหรับพิมพ์รายการผลการเรียนของนักศึกษาในแต่ละภาคการศึกษา ซึ่งผู้ใช้จะต้องใส่ข้อมูล รหัสนักศึกษา และปีการศึกษาที่เปิดลงทะเบียน



รูปที่ 5.21 หน้าจอสำหรับพิมพ์ผลการเรียนของนักศึกษาแยกตามรหัสนักศึกษา

รายงานแสดงผลการเรียนของนักศึกษาในแต่ละภาคการศึกษา

Rpt003 Student Grade		Date :dd/mm/yyyy		
King Mongkut's Institute of Technology Landkrabong				
Student : xxxxx(30)xxxxx	ID : 99999999	Semester/Year 9/yyyy		
Faculty : xxxxx(30)xxxxx	Degree : xxx(15)xxx	Program : xxx(30)xxx		
Course : xxxxx(30)xxxxx	Date : dd/mm/yyyy	No.		
Student Grade				
Code	Description	Sec	Credit	Grade
99999999	xxxxx(40)xxxxx	xx	9	x
		GPS	99	9.99
		GPA	99	9.99

5.1.18 หน้าจอสำหรับพิมพ์รายการผลการเรียนของนักศึกษาแยกตามกลุ่มของนักศึกษา ซึ่งผู้ใช้จะต้องใส่ข้อมูล ปีการศึกษาและภาคการศึกษาที่เปิดลงทะเบียน กลุ่มของนักศึกษา หลักสูตร แขนงวิชา

รูปที่ 5.22 หน้าจอสำหรับพิมพ์รายงานเกรดเฉลี่ยของนักศึกษาตามห้องเรียน

รายงานแสดงเกรดเฉลี่ยของนักศึกษาในแต่ละภาคการศึกษา

Rpt004 Class Grade :dd/mm/yyyy		Date		
King Mongkut's Institute of Technology Landkrabong				
Faculty : xxxxx(30)xxxxx		Degree : xxx(15)xxx	Program : xxx(30)xxx	
Semester/Year : 9/yyyy		Class : xxxxxxxxxx	Section : xxxxxxxxxx	
Class Grade				
ID	Student	Credit	Grade	Status
99999999	xxxxx(40)xxxxx	9	9.99	xx(10)xx

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

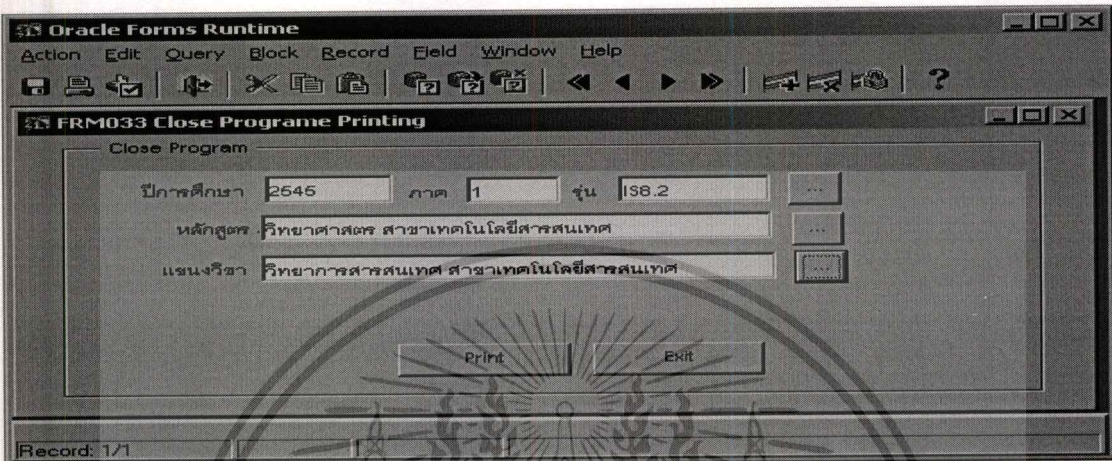
5.1.19 หน้าจอสำหรับพิมพ์รายชื่อนักศึกษาที่มีสิทธิ์สอบ Comprehensive และ Oral ในแต่ละภาคการศึกษา ซึ่งผู้ใช้จะต้องใส่ข้อมูล กลุ่มของนักศึกษา หลักสูตร และแขนงวิชา

รูปที่ 5.23 หน้าจอสำหรับพิมพ์รายชื่อนักศึกษาที่มีสิทธิ์สอบ Comprehensive และ Oral

รายงานแสดงรายชื่อผู้มีสิทธิ์สอบ Comprehensive และ Oral

Rpt005 Check Comprehensive and Oral Printing			Date
:dd/mm/yyyy			
King Mongkut's Institute of Technology Landkrabong			
Faculty : xxxxx(30)xxxxx	Degree : xxx(15)xxx	Program : xxx(30)xxx	
Semester/Year : 9/yyyy	Section : xxxxxxxxxx		
Check Comprehensive and Oral Printing			
ID	Student	Times	GPA
99999999	xxxxx(40)xxxxx	99	9.99

5.1.20 หน้าจอสำหรับพิมพ์รายชื่อนักศึกษาที่ผ่านหลักสูตรการเรียนการสอนในแต่ละภาคการศึกษา ซึ่งผู้ใช้งานจะต้องใส่ข้อมูล ปีการศึกษาและภาคการศึกษาที่เปิดลงทะเบียน กลุ่มของนักศึกษา หลักสูตร และแขนงวิชา



รูปที่ 5.24 หน้าจอสำหรับพิมพ์รายชื่อผู้สำเร็จการศึกษา

รายงานแสดงรายชื่อผู้สำเร็จการศึกษา

Rpt006 Check		King Mongkut's Institute of Technology Landkrabong				Date :dd/mm/yyyy
Faculty : xxxxx(30)xxxxx		Degree : xxx(15)xxx		Program : xxx(30)xxx		
Semester/Year : 9/yyyy		Section : xxxxxxxxxx				
Check						
ID	Student	Total Credit	Comprehensive	Project	Thesis	GPA

5.1.21 หน้าจอสำหรับพิมพ์รายชื่อนักศึกษาซึ่งผู้ใช้จะต้องใส่ข้อมูล กลุ่มของนักศึกษา หลักสูตร แขนงวิชา

รูปที่ 5.25 หน้าจอสำหรับพิมพ์รายชื่อนักศึกษา

รายงานแสดงรายชื่อนักศึกษาแยกตามกลุ่มของนักศึกษา

Rpt009 Student Info		Date :dd/mm/yyyy	
King Mongkut's Institute of Technology Landkrabong			
Faculty : xxxxx(30)xxxxx	Degree : xxx(15)xxx	Program : xxx(30)xxx	
Semester/Year : 9/yyyy	Class : xxxxxxxxxx	Section : xxxxxxxxxx	
Student Informations			
ID	Student	Tel	Email
99999999	xxxxx(40)xxxxx	(xx)xxxxxxx	xxx(30)xxx

5.1.22 หน้าจอสำหรับพิมพ์ผลการเรียนของนักศึกษาซึ่งผู้ใช้จะต้องใส่ข้อมูล รหัสนักศึกษา หลักสูตร และแขนงวิชา

รูปที่ 5.26 หน้าจอสำหรับพิมพ์ผลการเรียนของนักศึกษา

รายงานแสดงผลการเรียนของนักศึกษา

Rpt008 Student Grade		Date :dd/mm/yyyy		
King Mongkut's Institute of Technology Landkrabong				
Student : xxxxx(30)xxxxx	ID : 99999999	Semester/Year 9/yyyy		
Faculty : xxxxx(30)xxxxx	Degree : xxx(15)xxx	Program : xxx(30)xxx		
Course : xxxxx(30)xxxxx	Date : dd/mm/yyyy	No.		
Transcript				
Years	Code	Description	Credit	Grade
99/yyyy	xxxxxxxx	xxxxx(40)xxxxx	9	x
			GPS	99
			GPA	99

## บทที่ 6

### สรุปและข้อเสนอแนะ

#### 6.1 ผลการดำเนินงาน

หลังจากที่ได้ทำการพัฒนาโปรแกรมระบบสารสนเทศคณะเทคโนโลยีสารสนเทศ เพื่อช่วยอำนวยความสะดวกแก่เจ้าหน้าที่ในการจัดเก็บข้อมูลนักศึกษา ข้อมูลการลงทะเบียนเรียนและข้อมูลผลการเรียนของนักศึกษา รวมถึงการตรวจสอบสิทธิในการสอบ Comprehensive และ Oral ของนักศึกษาคณะเทคโนโลยีสารสนเทศ ซึ่งระบบดังกล่าวนี้สามารถช่วยผู้ใช้งานได้ดังต่อไปนี้คือ

- 6.1.1 สามารถจัดเก็บข้อมูล บันทึก แก้ไขและแสดงข้อมูลของข้อมูลหลักสูตรการเรียนและรายวิชาที่มีอยู่ในหลักสูตรการเรียนของนักศึกษาคณะเทคโนโลยีสารสนเทศได้
- 6.1.2 สามารถกำหนดรายวิชาที่เปิดลงทะเบียนในแต่ละภาคการศึกษาได้
- 6.1.3 สามารถตรวจสอบข้อมูลการลงทะเบียนของนักศึกษาและสถานะภาพของนักศึกษา ก่อนการลงทะเบียนได้
- 6.1.4 สามารถบันทึกผลการเรียนของนักศึกษาในแต่ละวิชาได้
- 6.1.5 สามารถตรวจสอบผลการเรียนของนักศึกษาและพิมพ์ใบ Transcript ได้
- 6.1.6 สามารถตรวจสอบเงื่อนไขการจบหลักสูตรของนักศึกษาได้

#### 6.2 ปัญหาและอุปสรรค

6.2.1 ด้านเวลา เนื่องจากผู้พัฒนาระบบมีเวลาในการพัฒนาโปรแกรมน้อยและไม่สามารถติดต่อกับผู้ใช้งานจริงได้ตลอดเวลา ทำให้การพัฒนาระบบเป็นไปได้ค่อนข้างลำบาก

6.2.2 ด้านข้อมูล เนื่องจากการพัฒนาระบบงาน ต้องใช้เวลาในการสืบค้นข้อมูลมาก และปัญหาทางด้านเวลาที่ค่อนข้างมีอยู่อย่างจำกัด ทำให้ขาดข้อมูลบางอย่างที่จำเป็นต่อการพัฒนาโปรแกรมจึงอาจทำให้โปรแกรมนี้อาจมีข้อผิดพลาดอยู่บ้าง

6.2.3 ด้านการออกแบบระบบ การใช้ UML ช่วยในการออกแบบระบบจึงประสบความสำเร็จจากความล่าช้าเนื่องจากความไม่ชำนาญของตัวผู้พัฒนาระบบเอง และเอกสารที่ใช้ประกอบการสืบค้นมีค่อนข้างน้อย จึงเป็นผลสืบให้เวลาในการพัฒนาระบบมีค่อนข้างน้อยตามไปด้วย

6.2.4 ความซับซ้อนของระบบงานหลักสูตรการเรียนของคณะเทคโนโลยีสารสนเทศ ทำให้ต้องมีการพิจารณา ในรายละเอียดค่อนข้างมากจึงทำให้การออกแบบและพัฒนาระบบเป็นไปได้ค่อนข้างล่าช้า

### 6.3 ข้อเสนอแนะ

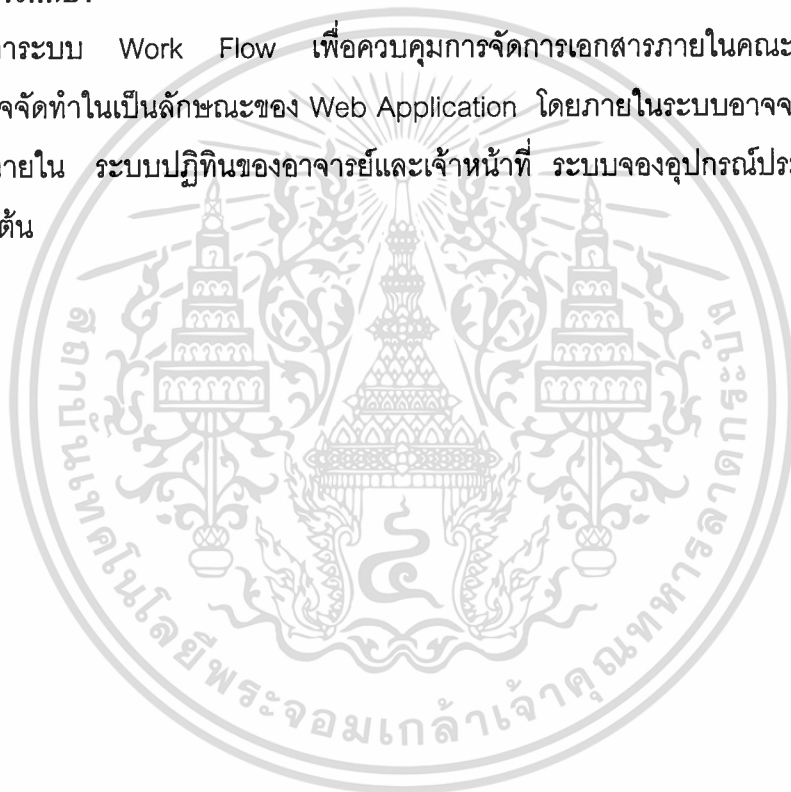
6.3.1 เพื่อสนับสนุนทางด้านเทคโนโลยีแบบออนไลน์ ควรจะมีการพัฒนาระบบงานบางอย่างผ่านทาง Internet ด้วย เช่นการตรวจสอบผลการเรียนของนักศึกษาผ่านทาง Internet เป็นต้น

6.3.2 ผู้พัฒนาโปรแกรมควรมีเวลาในการออกแบบและพัฒนาโครงการของฐานข้อมูลให้เพียงพอ

6.3.3 ผู้ที่สนใจอาจใช้ระบบสารสนเทศเป็นแนวทางในการพัฒนาโปรแกรมอื่นๆ ได้

### 6.4 แนวทางการศึกษา

จัดทำระบบ Work Flow เพื่อควบคุมการจัดการเอกสารภายในคณะเทคโนโลยีสารสนเทศ ซึ่งอาจจัดทำในเป็นลักษณะของ Web Application โดยภายในระบบอาจจะประกอบด้วยระบบ Mail ภายใน ระบบปฏิทินของอาจารย์และเจ้าหน้าที่ ระบบจองอุปกรณ์ประกอบการเรียนการสอน เป็นต้น



## บรรณานุกรม

งานบริการการศึกษา .2542 . คู่มือนักศึกษา หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ ประจำปีการศึกษา 2542.คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง : งานบริการการศึกษา.

สุนทริน วงศ์ศิริกุล.2537. พัฒนาโมเดลยูคใหม่ UML Unified Modeling Language. กรุงเทพฯ : บริษัท ชัคเชส มีเดีย จำกัด.

Hans-Erik Eriksson,Magnus Penker .1961.Business Modeling With UML :Business Patterns at Work :John Wiley &Sons, Inc.

