

**สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง**

**เว็บพอร์ทัลสำหรับภาควิชาวิศวกรรมคอมพิวเตอร์โดยใช้เว็บเซอร์วิส**

**CE WEB PORTAL USING WEB SERVICES**



รพ.  
๒๕๑๘๖  
๒๕๕๐

เลขหมู่.....  
เลขทะเบียน.....  
วัน,เดือน,ปี.....

83026

30 ก.ค. 2551

b.....
1195968x
i.....

**ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต**

**ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**

**ปีการศึกษา 2550**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2550

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เว็บไซต์ทอลสำหรับภาควิชาวิศวกรรมคอมพิวเตอร์โดยใช้เว็บเซอร์วิส

CE WEB PORTAL USING WEB SERVICES

ผู้จัดทำ

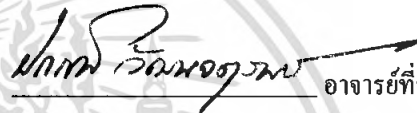
1. นางสาวนันทนา วนาธรรม รหัสนักศึกษา 47010379

2. นางสาวอรธยา เมยรินทร์ รหัสนักศึกษา 47010955



อาจารย์ที่ปรึกษา

(อาจารย์วัจจนพงศ์ เกษมศิริ)



อาจารย์ที่ปรึกษาร่วม

(ดร.ปกรณ์ วัฒนจตุรพร)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เว็บพอร์ทัลสำหรับภาควิชาวิศวกรรมคอมพิวเตอร์โดยใช้เว็บเซอร์วิส

นางสาวนันทนา	วนาธรรม	47010379
นางสาวอรรธยา	เมษินทรีย์	47010955
อาจารย์วงนพงค์	เกษมศิริ	อาจารย์ที่ปรึกษา
ดร.ปกรณ์	วัฒนจตุรพร	อาจารย์ที่ปรึกษาร่วม
ปีการศึกษา 2550		

### บทคัดย่อ

ในปัจจุบัน สารสนเทศได้กลายเป็นปัจจัยสำคัญในการทำธุรกรรมต่าง ๆ ทั้งในแต่ละบุคคลและองค์กร เช่นเดียวกับบุคคลากรในภาควิชาวิศวกรรมคอมพิวเตอร์ หากได้รับข้อมูลข่าวสารที่จำเป็นต่อการเรียน หรือการทำงานอย่างครบถ้วนแล้ว ย่อมทำให้การทำงานเกิดข้อผิดพลาดน้อยลงหรือไม่เกิดข้อผิดพลาดขึ้น

ดังนั้น ปรียนุญานิพนธ์ฉบับนี้ จึงนำเสนอการพัฒนาระบบสารสนเทศภาควิชา ฯ ในรูปแบบของเว็บพอร์ทัล (Web Portal) กล่าวคือ เป็นเสมือนเว็บที่คัดกรองเฉพาะสารสนเทศที่ต้องการในแต่ละบุคคลจากระบบทั้งภายใน และภายนอกสถาบัน เช่น ระบบของทะเบียนนักศึกษา, ระบบหอสมุดกลาง และอื่น ๆ แต่ปัญหาสำคัญอย่างหนึ่งในการเชื่อมต่อระบบ คือ แพลตฟอร์มที่ต่างกันของแต่ละแอปพลิเคชัน จึงต้องนำเทคโนโลยีเว็บเซอร์วิสมาช่วยในการแลกเปลี่ยนข้อมูลที่มีรูปแบบต่างกัน

โครงการนี้เป็นการพัฒนาเว็บแอปพลิเคชัน ด้วยโปรแกรม Microsoft Visual Studio .NET 2005 ซึ่งสร้างด้วยภาษา ASP.NET (C#) ทำงานร่วมกับโปรแกรมจัดการฐานข้อมูล Microsoft SQL Server 2005 และพัฒนาส่วนบริการเว็บเซอร์วิสของระบบทะเบียนนักศึกษาด้วยภาษา PHP ร่วมกับ NuSoap ซึ่งเว็บพอร์ทัลที่พัฒนาขึ้นมาี้ สามารถนำมาใช้งานได้จริง และสอดคล้องกับความต้องการของบุคคลากรในภาควิชาฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## The CE Web Portal using Web Services

Ms. Nuntana	Wanatham	47010379
Ms. Addhaya	Measincee	47010955
Mr. Watjanapong	Kasemsiri	Advisor
Dr. Pakorn	Watanachaturaporn	Co-Advisor

Academic Year 2550

### ABSTRACT

This report presents a software development for information system of the Department of Computer Engineering in the web portal style, or it is called "CE web portal." The information becomes an important factor for individual persons as well as persons in the department. Thus, the CE web portal is designed to filter the required information and to link information both inside and outside KMUTL; e.g., the Office of Registrar, the Central Library, or Google. Moreover, according to problems of information exchange across platforms, the CE web portal has been developed using a Web Services technology which characterizes to solve the problems.

In the phase of development, we utilized the Web Application technology. Then, we implemented CE web portal using ASP.NET (C#) including Microsoft SQL Server 2005, while the Office of Registrar's service provider is implemented by PHP plus NuSoap extension. In addition, the portal has been analyzed to be an appropriate, available and satisfactory system.

## กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้คงไม่อาจสำเร็จลุล่วงไปด้วยดี หากขาดการสนับสนุนจากอาจารย์ที่ปรึกษาทั้งสองท่าน คือ อาจารย์วัจนพงศ์ เกษมศิริ และดร.ปกรณ์ วัฒนจตุรพร ข้าพเจ้าขอกราบขอบพระคุณอาจารย์ที่คอยให้คำแนะนำ และคำปรึกษาเสมอมา รวมไปถึงอาจารย์และเจ้าหน้าที่ท่านอื่น ๆ ที่ได้อำนวยความสะดวกในการประสานงานกับระบบทะเบียนนักศึกษา และระบบหอสมุดกลาง

ขอกราบขอบพระคุณคณาจารย์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้แก่ข้าพเจ้า ให้ความรู้ตั้งแต่วันนี้ ขอขอบคุณพี่ ๆ เพื่อน ๆ และน้อง ๆ ทุกคนที่คอยให้คำปรึกษา และกำลังใจเสมอมา

สุดท้ายนี้บุคคลที่สำคัญที่สุดในชีวิตของข้าพเจ้า คือ บิดา มารดา และครอบครัวของข้าพเจ้า ข้าพเจ้าขอกราบขอบพระคุณทุกท่านที่คอยให้การสนับสนุน และช่วยเหลือในทุก ๆ สิ่งให้ข้าพเจ้ามีกำลังใจต่อสู้กับปัญหาและอุปสรรคที่ต้องเผชิญเสมอมา

คุณค่าและประโยชน์ทั้งหลายซึ่งมาจากปริญญานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้แก่ผู้มีพระคุณทุกท่าน

นันทนา วนาธรรม  
อรรธยา เมษินทรีย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VIII
สารบัญรูป.....	IX
บทที่ 1 บทนำ.....	1
1.1 ความสำคัญและที่มาของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.4 ขอบเขตของโครงการ.....	2
1.5 ขั้นตอนการดำเนินงาน.....	3
1.6 ส่วนประกอบของปฏิญญานิพนธ์.....	3
บทที่ 2 ทฤษฎีพื้นฐานที่เกี่ยวข้อง.....	5
2.1 เว็บพอร์ทัล (Web Portal).....	5
2.1.1 บทนำ.....	5
2.1.2 ความหมายของเว็บพอร์ทัล.....	5
2.1.3 คุณสมบัติของเว็บพอร์ทัล.....	5
2.1.4 ข้อดีของเว็บพอร์ทัล.....	5
2.1.5 ตัวอย่างเว็บพอร์ทัลที่เป็นที่นิยมในปัจจุบัน.....	6
2.2 เว็บแอปพลิเคชัน (Web Application).....	6
2.2.1 บทนำ.....	6
2.2.2 การทำงานของเว็บแอปพลิเคชัน.....	6
2.2.3 ภาษา HTML.....	7
2.2.4 สไตล์ชีท CSS.....	7
2.2.5 ภาษาประเภทสคริปต์.....	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)

	หน้า
2.3 เทคโนโลยีเว็บเซอร์วิส .....	8
2.3.1 บทนำ.....	8
2.3.2 นิยามของเว็บเซอร์วิส .....	9
2.3.3 ลักษณะเด่นของเว็บเซอร์วิส .....	9
2.3.4 หลักการทำงานของเว็บเซอร์วิส .....	10
2.3.5 องค์ประกอบหลักของเว็บเซอร์วิส .....	11
2.4 ภาษา XML.....	13
2.4.1 บทนำ.....	13
2.4.2 ลักษณะของภาษา XML .....	13
2.4.3 ข้อแตกต่างระหว่าง XML และ HTML.....	14
2.4.4 จุดเด่นของภาษา XML.....	15
2.4.5 XML Syntax.....	15
2.4.6 XML Schema .....	17
2.4.7 ตัวอย่างการทำงานร่วมกันระหว่างเอกสาร XML และ XML Schema .....	18
2.4.8 RSS Feed.....	18
2.5 โปรโตคอล SOAP.....	19
2.5.1 บทนำ.....	19
2.5.2 คำอธิบายของ SOAP .....	19
2.5.3 กระบวนการทำงานของ SOAP .....	19
2.5.4 โครงสร้างของ SOAP Message.....	20
2.5.5 ลักษณะเด่นของ SOAP.....	21
2.6 ภาษา WSDL.....	21
2.6.1 บทนำ.....	21
2.6.2 ความหมายของ WSDL.....	21
2.6.3 โครงสร้างของเอกสาร WSDL .....	23
2.6.4 ตัวอย่างเอกสาร WSDL .....	23
2.6.5 ตัวอย่างการ Binding กับ SOAP .....	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)

	หน้า
2.7 ภาษา PHP.....	24
2.7.1 บทนำ.....	24
2.7.2 การพัฒนาเว็บแอปพลิเคชันด้วยภาษา PHP .....	25
2.7.3 จุดเด่นของภาษา PHP .....	25
2.7.4 ภาษา PHP กับเว็บเซอร์วิส.....	25
2.8 ภาษา ASP.NET.....	26
2.8.1 คอทเน็ตเฟรมเวิร์ก (.NET Framework).....	26
2.8.2 ภาษา ASP.NET.....	26
2.8.3 จุดเด่นของ ASP.NET .....	27
2.8.4 ภาษา ASP.NET กับเว็บเซอร์วิส .....	27
บทที่ 3 การออกแบบระบบ CE Web Portal.....	28
3.1 โครงสร้างของระบบ .....	28
3.2 รายละเอียดของระบบ.....	29
3.3 Use Case Diagram.....	32
3.4 Activity Diagram .....	35
3.5 การออกแบบฐานข้อมูล (Database Design).....	48
3.5.1 ออกแบบฐานข้อมูล ด้วย ORM (Object Role Model).....	48
3.5.2 โครงตารางในฐานข้อมูล dbPortal.....	50
3.6 การออกแบบเซอร์วิส .....	52
3.6.1 เซอร์วิส โพรไวเดอร์ที่เปิดให้บริการที่สำนักงานฯ.....	52
3.6.2 เซอร์วิสรีเคิสเตอร์ของเว็บพอร์ทัล.....	52
บทที่ 4 การทดลองและผลการทดลอง .....	53
4.1 บทนำ.....	53
4.2 เครื่องมือที่ใช้ในการพัฒนา .....	53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
4.3 การพัฒนาเว็บแอปพลิเคชัน .....	54
4.3.1 โมดูลประกาศข่าว.....	55
4.3.2 โมดูลรับ/ส่งข้อความ .....	55
4.3.3 โมดูลปฏิทิน.....	56
4.3.4 โมดูลเอกสารรายวิชา.....	56
4.3.5 โมดูลสำนักทะเบียน .....	57
4.3.6 โมดูลห้องสมุด.....	58
4.3.7 โมดูลคอนเทนท์.....	58
4.3.8 โมดูลปรับแต่งอินเทอร์เฟซ .....	59
4.4 การพัฒนาเว็บเซอร์วิส .....	59
4.5 ผลการทดสอบระบบ .....	60
บทที่ 5 บทสรุปและวิจารณ์ .....	61
5.1 บทสรุป.....	61
5.2 ปัญหาอุปสรรคและแนวทางแก้ไข .....	62
5.3 แนวทางการพัฒนาต่อ.....	62
บรรณานุกรม .....	64
ภาคผนวก ก.....	67
ภาคผนวก ข.....	69

# สารบัญตาราง

ตารางที่	หน้า
3.1 แสดงสิทธิการใช้งานในแต่ละโมดูล.....	30
4.1 แสดงรายละเอียดซอร์สแวร์ที่ใช้ในการพัฒนา.....	53
4.2 แสดงรายละเอียดซอฟต์แวร์ที่ใช้ในการพัฒนา.....	53
4.3 แสดงรายละเอียดเซอร์วิสที่ให้บริการที่ <a href="http://www.reg.kmitl.ac.th/websevice/service/">www.reg.kmitl.ac.th/websevice/service/</a> .....	59
ข.1 แสดงรายละเอียดซอฟต์แวร์ที่ใช้ในการพัฒนาเซอร์วิสด้วยภาษา PHP.....	69
ข.2 แสดงรายละเอียดซอฟต์แวร์ที่ใช้ในการพัฒนาเซอร์วิสด้วยภาษา ASP.NET.....	74
ข.3 แสดงรายละเอียดซอฟต์แวร์ที่ใช้ในการพัฒนาเซอร์วิสด้วยภาษา PHP และ ASP.NET.....	79



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป

รูปที่	หน้า
2.1 หลักการทำงานของเว็บแอปพลิเคชัน.....	7
2.2 วิวัฒนาการของเว็บไซต์.....	9
2.3 รูปแบบจำลองเว็บเซอร์วิส.....	10
2.4 กระบวนการทำงานของเว็บเซอร์วิส.....	11
2.5 แนวคิดเว็บเซอร์วิสในรูปแบบเสต็กของ W3C.....	12
2.6 แนวคิดเว็บเซอร์วิสในรูปแบบเสต็กของ IBM.....	12
2.7 ตัวอย่างโค้ดภาษา HTML.....	14
2.8 ตัวอย่างโค้ดภาษา XML.....	14
2.9 โครงสร้างภาษา XML.....	16
2.10 การตรวจสอบความถูกต้องในเอกสาร XML.....	16
2.11 เอกสาร XML ที่เป็น Well-formed XML และ Valid XML.....	17
2.12 เอกสาร XML และ XML Schema.....	17
2.13 ตัวอย่างเอกสาร XML (note.xml) ซึ่งมีการอ้างถึง schema (note.xsd).....	18
2.14 ตัวอย่างเอกสาร XML Schema (note.xsd).....	18
2.15 สัญลักษณ์ของ RSS Feed.....	19
2.16 หลักการทำงานของ SOAP.....	19
2.17 ขั้นตอนการส่ง SOAP message ระหว่างแอปพลิเคชัน.....	20
2.18 โครงสร้างของ SOAP.....	21
2.19 คำอธิบายเซอร์วิสที่ใช้ในภาษา WSDL.....	22
2.20 คำนิยามอินเทอร์เฟซ และอิมพลิเมนต์เทชั่นของเซอร์วิส.....	22
2.21 รูปแบบโครงสร้างของเอกสาร WSDL.....	23
2.22 ตัวอย่างเอกสาร WSDL.....	23
2.23 ตัวอย่างเอกสาร WSDL ที่ binding กับ SOAP.....	24
2.24 สัญลักษณ์ของภาษา PHP.....	25
2.25 ตัวอย่างโค้ดภาษา PHP.....	25
2.26 โปรแกรม Visual Studio 2005.....	26
2.27 สัญลักษณ์ภาษา ASP.NET.....	26
3.1 โมเดลของระบบ CE Web Portal.....	28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.2 โมเดลของระบบ CE Web Portal .....	29
3.3 Use Case Diagram ของระบบ: โมดูล 1, 2, 4 .....	32
3.4 Use Case Diagram ของระบบ: โมดูล 3, 5 .....	33
3.5 Use Case Diagram ของระบบ: โมดูล 6, 7, 8 .....	34
3.6 Activity Diagram ของการล็อกอิน (log-in).....	35
3.7 Activity Diagram ของการสร้างประกาศข่าว .....	36
3.8 Activity Diagram ของการแก้ไขประกาศข่าว .....	37
3.9 Activity Diagram ของการแสดงประกาศข่าว .....	38
3.10 Activity Diagram ของการส่งข้อความ (private/group).....	39
3.11 Activity Diagram ของการรับข้อความ.....	40
3.12 Activity Diagram ของการดาวน์โหลดเอกสารการเรียน .....	41
3.13 Activity Diagram ของการแสดงผลเอกสารการเรียน.....	42
3.14 Activity Diagram ของการอัปโหลดเอกสารการเรียน .....	43
3.15 Activity Diagram ของการแสดงตารางเรียน และตารางสอบ.....	44
3.16 Activity Diagram ของการค้นหาหนังสือห้องสมุด .....	45
3.17 Activity Diagram ของการแสดงคอนเทนต์ .....	45
3.18 Activity Diagram ของการเพิ่มคอนเทนต์ให้ทั้งระบบ .....	46
3.19 Activity Diagram ของการเพิ่มคอนเทนต์ส่วนตัว.....	47
3.20 โมเดลของฐานข้อมูล dbPortal (1) .....	48
3.21 โมเดลของฐานข้อมูล dbPortal (2) .....	49
3.22 โครงตารางของ โมดูลสำนักทะเบียนฯ โมดูลเอกสารการเรียน และข้อมูลรายวิชา.....	50
3.23 โครงตารางเกี่ยวกับ โมดูลข่าว และ โมดูลข้อความ.....	50
3.24 โครงตารางเกี่ยวกับ โมดูลคอนเทนต์ และ โมดูลปฏิทิน .....	51
3.25 โครงตารางเกี่ยวกับ โมดูล GUI และรายละเอียดของผู้ใช้งาน .....	51
3.26 โครงตารางเกี่ยวกับ โมดูล GUI ที่ใช้ในบอกรการตั้งค่าส่วนบุคคล .....	51
4.1 เว็บเพจหน้าพอร์ทัล .....	54
4.2 โมดูลประกาศข่าวตามสิทธิ์ user และ anonymous.....	55
4.3 โมดูลประกาศข่าวตามสิทธิ์ super user .....	55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.4 โมดูลรับ/ส่งข้อความตามสิทธิ์ super user, user และ anonymous .....	55
4.5 โมดูลปฏิทินตามสิทธิ์ super user, user และ anonymous .....	56
4.6 แบบฟอร์มเพิ่มรายการปฏิทินตามสิทธิ์ super user, user และ anonymous.....	56
4.7 โมดูลเอกสารรายวิชาตามสิทธิ์ user .....	56
4.8 โมดูลเอกสารรายวิชาตามสิทธิ์ super user.....	57
4.9 แบบฟอร์มเพิ่มเอกสารในโมดูลเอกสารรายวิชาตามสิทธิ์ super user.....	57
4.10 การลบเอกสารในโมดูลเอกสารรายวิชาตามสิทธิ์ super user .....	57
4.11 โมดูลสำนักทะเบียนตามสิทธิ์ super user และ user .....	57
4.12 โมดูลห้องสมุดตามสิทธิ์ super user และ user.....	58
4.13 โมดูลคอนเทนต์ตามสิทธิ์ super user และ user.....	58
4.14 แบบฟอร์มการเพิ่มคอนเทนต์ตามสิทธิ์ super user และ user.....	58
4.15 โมดูลปรับแต่งอินเทอร์เน็ตเฟสตามสิทธิ์ super user และ user.....	59
ข.1 รอรับรหัสนักศึกษาบน Computer 2 เพื่อร้องขอเซอร์วิส.....	70
ข.2 แสดงผลข้อมูลที่ตอบกลับมาจาก Computer 1 .....	70
ข.3 ไฟล์ service_gpa.php .....	71
ข.4 ไฟล์ client.php.....	72
ข.5 เซอร์วิส gpaService ที่เปิดให้บริการ.....	73
ข.6 รายละเอียดของโอเปอเรชั่น GetGPA ที่เปิดให้บริการ .....	73
ข.7 WSDL ที่อธิบายเซอร์วิส gpaService.....	74
ข.8 เซอร์วิสแสดงคำว่า Hello World .....	75
ข.9 การสร้างเว็บไซต์ ASP.NET Web Service .....	75
ข.10 ไฟล์ service.cs.....	76
ข.11 การสร้างหน้าเว็บ default.aspx.....	76
ข.12 ไฟล์ default.aspx ที่เพิ่มปุ่มและช่องแสดงผล .....	77
ข.13 วิธี Add Web Reference ในการเซอร์วิสฝั่งไคลเอนท์.....	77
ข.14 แสดงรายละเอียดของเซอร์วิสที่ชื่อ Service .....	78
ข.15 ไฟล์ที่ได้จากการ Add Web Reference .....	78
ข.16 ไฟล์ default.aspx.cs ที่มีเมธอด B_Click .....	78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
ข.17 วิธี Add Web Reference ด้วย URL ของไฟล์ service_gpa.php .....	80
ข.18 Add Web Reference -> URL : ที่อยู่ของไฟล์ service_gpa.php -> WSDL .....	80
ข.19 ไฟล์ที่ได้จากการ Add Web Reference .....	80
ข.20 ไฟล์ default.aspx ที่เพิ่มช่องรับข้อความ, ปุ่ม และช่องแสดงผล .....	81
ข.21 ไฟล์ default.aspx.cs ที่เพิ่มการเรียกใช้เซอร์วิส และการแสดงผล.....	81
ข.22 แสดงการเรียกใช้งานเซอร์วิส PHP ด้วยเว็บ ASP.NET .....	82



# บทที่ 1

## บทนำ

### 1.1 ความสำคัญและที่มาของโครงการ

ในยุคของสารสนเทศ ข้อมูลข่าวสารถือได้ว่าเป็นสิ่งสำคัญที่แต่ละบุคคลมีความต้องการในสารสนเทศแตกต่างกัน บางครั้งข้อมูลบางอย่างอาจไม่เป็นประโยชน์กับบุคคลหนึ่ง แต่ในขณะเดียวกันกลับเป็นประโยชน์กับอีกบุคคลหนึ่ง และหากบุคคลนั้น ๆ ไม่ได้รับรู้ข่าวสารนั้นก็อาจเกิดความเสียหายหรือความผิดพลาดในการทำงานได้ ซึ่งการที่แต่ละคนจะได้รับข้อมูลที่ครบถ้วน จำเป็นต้องเปิดเว็บไซต์จำนวนมาก จึงเป็นที่มาของเว็บไซต์ในลักษณะของเว็บพอร์ทัล (**Web Portal**) กล่าวคือ เว็บพอร์ทัล (*Web Portal*) เป็นเว็บไซต์ที่สร้างขึ้นเพื่อเป็นจุดเชื่อมต่อในการเข้าถึงข้อมูลบนเว็บต่าง ๆ ไว้ในที่เดียวทั้งภายในและภายนอก (จากคำจำกัดความของ *Wikipedia.org*)

เช่นเดียวกันกับบุคลากรในสถานศึกษาก็มีความจำเป็นที่จะได้รับข้อมูลสารสนเทศที่จำเป็นในการเรียน และการทำงาน ซึ่งในปัจจุบันนักศึกษา, อาจารย์ และเจ้าหน้าที่ก็ต้องเข้าเว็บไซต์จำนวนมากเพื่อที่จะได้รับสารสนเทศอย่างครบถ้วน เช่น เว็บไซต์ของสำนักทะเบียนฯ และประมวลผล, เว็บไซต์ของภาควิชาฯ, เว็บไซต์ของสถาบันฯ และเว็บไซต์ของหอสมุดกลาง เป็นต้น ซึ่งบางทีหากหลงลืมเข้าที่ใดที่หนึ่งอาจทำให้เกิดความเสียหายต่อการเรียน และงานได้

จากความสำคัญนี้ทำให้เกิดความคิดที่จะทำระบบสารสนเทศสำหรับภาควิชาวิศวกรรมคอมพิวเตอร์ในรูปแบบของเว็บพอร์ทัลขึ้น กล่าวคือ เป็นเว็บศูนย์กลางที่เสมือนประตูเชื่อมต่อไปยังเว็บภายใน และภายนอกสถาบันเทคโนโลยีพระจอมเกล้าฯ เช่น เชื่อมต่อกับระบบสารสนเทศของสำนักทะเบียนฯ และเชื่อมต่อกับสำนักหอสมุดกลาง เพื่ออำนวยความสะดวก และคัดกรองสารสนเทศเฉพาะที่ต้องการในแต่ละบุคคลให้แก่ นักศึกษาภาควิชาฯ, อาจารย์ และเจ้าหน้าที่ รวมทั้งเป็นเสมือนเครื่องมือสื่อสารระหว่างกันภายในภาควิชาฯ

แต่การที่จะรวบรวมสารสนเทศจากหลากหลายแหล่งให้มาอยู่ในรูปแบบของเว็บพอร์ทัล ก็ย่อมมีปัญหาในเรื่องของการแลกเปลี่ยนข้อมูลที่ต่างรูปแบบกัน รวมถึงความไม่เข้ากันของแอปพลิเคชัน เนื่องจากการที่ต่างระบบต่างก็พัฒนาระบบของตนในรูปแบบที่ต่างกันทั้งด้านภาษา และแพลตฟอร์ม (Platform) ที่ใช้งาน จึงต้องนำเทคโนโลยีเว็บเซอร์วิส (**Web Services**) เข้ามาช่วยในการเชื่อมต่อออกไปยังระบบต่าง ๆ เพื่อแลกเปลี่ยนข้อมูลที่ต้องการ

## 1.2 วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาหลักการทํางาน และทฤษฎีที่เกี่ยวข้องกับเว็บเซอร์วิส
2. เพื่อศึกษาการสร้างเว็บเซอร์วิสด้วยภาษาที่แตกต่างกัน และทดลองทํางานของเว็บเซอร์วิสผ่านเครือข่าย
3. เพื่อศึกษาและพัฒนาเว็บแอปพลิเคชันด้วยเทคโนโลยีที่สนใจ
4. เพื่อออกแบบระบบเว็บพอร์ทัลสำหรับภาควิชาวิศวกรรมคอมพิวเตอร์ให้สอดคล้องกับความต้องการใช้งานกับบุคลากรในภาควิชาฯ
5. เพื่อพัฒนาเว็บพอร์ทัลสำหรับภาควิชาวิศวกรรมคอมพิวเตอร์ให้สามารถนำมาใช้ประโยชน์จริงได้

## 1.3 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้ความเข้าใจในหลักการทํางาน และ โครงสร้างพื้นฐานของเว็บเซอร์วิส
2. สามารถสร้างเว็บเซอร์วิส และเรียกใช้งานข้ามระบบที่พัฒนาต่างแพลตฟอร์มได้
3. สามารถสร้างเว็บแอปพลิเคชันที่สามารถให้บริการ และเรียกใช้งานเซอร์วิสต่าง ๆ ผ่านเครือข่ายได้ รวมทั้งสามารถติดต่อกับฐานข้อมูลได้
4. สามารถสร้างเว็บแอปพลิเคชันที่ตรงกับความต้องการกับผู้ใช้งานได้
5. สามารถนำระบบติดตั้งบนเครื่องเซิร์ฟเวอร์ได้
6. สามารถออกแบบ และพัฒนาระบบเว็บพอร์ทัลสำหรับภาควิชาวิศวกรรมคอมพิวเตอร์ให้สามารถนำมาใช้งานจริงได้

## 1.4 ขอบเขตของโครงการ

โครงการนี้จะศึกษาหลักการทํางานของเทคโนโลยีเว็บเซอร์วิส และสิ่งที่เกี่ยวข้อง เช่น ภาษา XML (eXtensible Markup Language), โพรโทคอล SOAP (Simple Object Access Protocol) และภาษา WSDL (Web Services Description Language) เป็นต้น นอกจากนี้ ยังศึกษาการพัฒนาเว็บแอปพลิเคชันด้วยเทคโนโลยี .NET, Client-side script และ Server-side script เพื่อพัฒนาเป็นรูปแบบเว็บพอร์ทัล

โครงการนี้ยังพัฒนาระบบเว็บพอร์ทัลให้ตรงตามความต้องการของบุคลากรในภาควิชาฯ ในส่วนการพัฒนาเว็บพอร์ทัลจะพัฒนาด้วยโปรแกรม Microsoft Visual Studio .NET 2005 ร่วมกับ ASP.NET และภาษา C# พร้อมทั้งใช้งานระบบการจัดการฐานข้อมูล (DBMS) คือ Microsoft SQL Server 2005 และในส่วนที่เชื่อมต่อกับระบบทะเบียนฯ จะพัฒนาเว็บเซอร์วิสด้วยภาษา PHP (Hypertext Preprocessor) ร่วมกับ NuSoap

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.5 ขั้นตอนการดำเนินงาน

1. ศึกษาตัวอย่างเว็บพอร์ทัลบนอินเทอร์เน็ต
2. จัดทำแบบสำรวจความต้องการของผู้ใช้งาน (User requirements)
3. กำหนดขอบเขตของโครงการ และศึกษาความเป็นไปได้ในการเชื่อมต่อพอร์ทัลกับระบบอื่น ๆ ในสถาบัน
4. ศึกษา และเปรียบเทียบภาษา และเฟรมเวิร์กที่ใช้ในการพัฒนาเว็บพอร์ทัล
5. ศึกษาเทคโนโลยีเว็บเซอร์วิส และทฤษฎีที่เกี่ยวข้อง
6. ศึกษาการใช้งานระบบการจัดการฐานข้อมูล (DBMS: **D**atabase **M**anagement **S**ystem) และภาษาที่ใช้ติดต่อกับฐานข้อมูล
7. วิเคราะห์และออกแบบระบบเว็บพอร์ทัล รวมทั้งออกแบบฐานข้อมูล
8. ศึกษาระบบของหน่วยงานที่ต้องการทำการเชื่อมต่อพอร์ทัล
9. พัฒนาเว็บเซอร์วิสในส่วนที่ต้องเชื่อมต่อกับระบบภายในสถาบันฯ
10. พัฒนาแอปพลิเคชันกลางของระบบเว็บพอร์ทัล และส่วนของการแสดงผลที่ติดต่อกับผู้ใช้งาน
11. ทดสอบการทำงานของระบบ และแก้ไขข้อผิดพลาด
12. เปิดให้ทดสอบระบบบนเครือข่ายจริง
13. ประเมิน และสรุปผลการทำงานของระบบ

## 1.6 ส่วนประกอบของปฏิญานิพนธ์

ปฏิญานิพนธ์ฉบับนี้มีทั้งสิ้น 5 บท โดยแต่ละบทมีเนื้อหาดังต่อไปนี้

บทที่ 1 บทนำ กล่าวถึงความสำคัญและที่มาของโครงการ, วัตถุประสงค์ของโครงการ, ประโยชน์ที่คาดว่าจะได้รับ, ขอบเขตของโครงการ, ขั้นตอนการดำเนินงาน และส่วนประกอบของปฏิญานิพนธ์ฉบับนี้

บทที่ 2 ทฤษฎีพื้นฐานที่เกี่ยวข้อง กล่าวถึงเทคโนโลยีที่ใช้ในการพัฒนาโครงการ ได้แก่ แนวคิดเว็บพอร์ทัล, เทคโนโลยีที่ใช้พัฒนาเว็บแอปพลิเคชัน, เทคโนโลยีเว็บเซอร์วิส, ภาษา XML, โปรโตคอล SOAP, ภาษา WSDL, ภาษา PHP, และภาษา ASP.NET

บทที่ 3 การออกแบบระบบ CE Web Portal กล่าวถึง โครงสร้างของระบบโดยรวม, รายละเอียดของของเว็บพอร์ทัล, การแบ่งระดับสิทธิการใช้งานในระบบ, Use Case Diagram, Activity Diagram, การออกแบบฐานข้อมูลด้วยโมเดล ORM, โครงตารางทั้งหมดในฐานข้อมูล dbPortal, การออกแบบเซอร์วิสของสำนักทะเบียนฯ, และฟังก์ชันของเว็บพอร์ทัลที่เรียกใช้งาน เซอร์วิส

บทที่ 4 การทดลองและผลการทดลอง กล่าวถึง เครื่องมือที่ใช้ในการพัฒนา, การพัฒนาเว็บ  
แอปพลิเคชันในแต่ละโมดูล, การพัฒนาเว็บเซอร์วิส, และผลการทดสอบระบบ

บทที่ 5 บทสรุปและวิจารณ์ กล่าวถึง บทสรุปของการทำโครงการ, ปัญหาอุปสรรคและ  
แนวทางแก้ไข, และแนวทางการพัฒนาต่อ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# ทฤษฎีพื้นฐานที่เกี่ยวข้อง

### 2.1 เว็บพอร์ทัล (Web Portal)

#### 2.1.1 บทนำ

ในปัจจุบันข้อมูลเป็นสิ่งสำคัญ โดยข้อมูลต่าง ๆ จะกระจายตัวอยู่ในเว็บไซต์ต่าง ๆ บนอินเทอร์เน็ต โดยบุคคลบางกลุ่มที่มีความสนใจร่วมกัน หรืออยู่ในสังคมเดียวกันย่อมมีกลุ่มข้อมูลที่สนใจกลุ่มเดียวกัน แต่ลำดับความสำคัญของข้อมูลเหล่านั้นอาจแตกต่างกันไปตามความต้องการส่วนบุคคล จึงเกิดเว็บไซต์ในรูปแบบที่รวบรวมแหล่งข้อมูลที่ต้องการไว้ด้วยกันก็คือ เว็บพอร์ทัล (Web Portal) นั่นเอง

#### 2.1.2 ความหมายของเว็บพอร์ทัล

คำว่า พอร์ทัล (Portal) แปลว่า ประตูทางเข้า ดังนั้นเว็บพอร์ทัลจึงเป็นเสมือน “ห้อง” ที่มีประตูอยู่มากมาย ซึ่งอีกฟากของประตูเป็นข้อมูลที่เราสนใจ โดยเราสามารถเปิดประตุนั้นทิ้งไว้ตลอดเวลา หรือเปิดเฉพาะเมื่อต้องการ หรือไม่ต้องเปิดเลยก็ได้

ดังนั้น เว็บพอร์ทัล คือ เว็บไซต์ที่รวบรวมจุดเชื่อมต่อไปยังข้อมูลต่าง ๆ บนอินเทอร์เน็ตไว้ด้วยกัน โดยแต่ละแอปพลิเคชันที่เชื่อมต่อถึงมีวิธีการเรียกใช้เซอร์วิสต่างกันไป

#### 2.1.3 คุณสมบัติของเว็บพอร์ทัล

- มีรูปแบบสวยงาม, สบายตา และใช้งานง่าย (Look and feel)
- มีหัวข้อสอดคล้องกัน มีรูปแบบ, ไอคอน, สี สัน หรือสัญลักษณ์ ที่สื่อความหมาย และอยู่ในรูปแบบเดียวกัน
- รวบรวม พอร์ทัลเล็ก (Portlet) ซึ่งเป็นเว็บแอปพลิเคชันในตัวไว้ด้วยกัน
- เป็นจุดเชื่อมต่อในการการถึงข้อมูลที่ต้องการอย่างมีประสิทธิภาพ และล่าสุดเสมอ

#### 2.1.4 ข้อดีของเว็บพอร์ทัล

- สามารถเชื่อมโยงและเข้าถึงข้อมูล, แอปพลิเคชัน และกระบวนการขององค์กรได้
- เพิ่มประสิทธิภาพในการติดต่อสื่อสารกันระหว่างบุคคลในองค์กร
- สามารถเข้าถึงระบบที่ต่างกันได้ในหน้าเดียว
- มีการโต้ตอบกับผู้ใช้งานต่างกันไปตามบุคคล
- สามารถแก้ไข และปรับปรุงหน้าเว็บได้ง่าย และรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.5 ตัวอย่างเว็บพอร์ทัลที่เป็นที่นิยมในปัจจุบัน

- My MSN (<http://my.msn.com/>)
- iGoogle (<http://www.google.com/ig>)
- My Yahoo! (<http://my.yahoo.com>)

การพัฒนาเว็บพอร์ทัลขึ้นมา นั้น จะต้องใช้เทคโนโลยีด้านเว็บแอปพลิเคชันหลายอย่าง ประกอบกันดังที่จะกล่าวถึงในหัวข้อที่ 2.2 นอกจากนี้เว็บพอร์ทัลยังสามารถเชื่อมโยงไปหาข้อมูล ได้หลากหลายวิธี แต่เทคโนโลยีที่ได้เลือกมาใช้ในโครงการนี้คือ เทคโนโลยีเว็บเซอร์วิส ดังจะ กล่าวในหัวข้อที่ 2.3

## 2.2 เว็บแอปพลิเคชัน (Web Application)

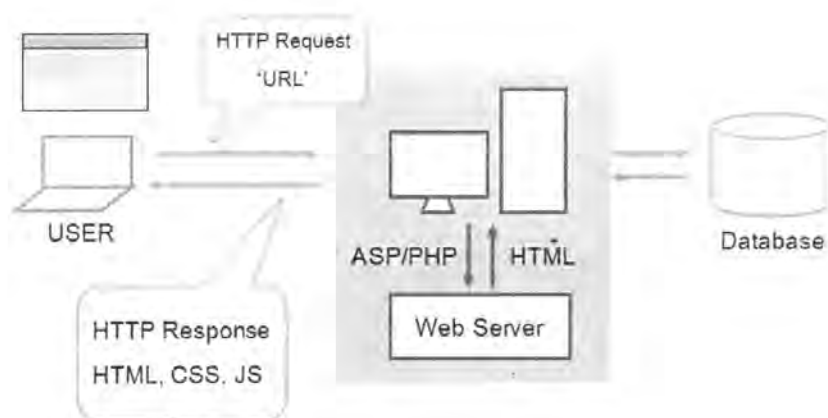
### 2.2.1 บทนำ

ในช่วงสิบกว่าปีที่ผ่านมา จะพบว่าอัตราผู้เล่นอินเทอร์เน็ตเพิ่มขึ้นเป็นจำนวนมากจนแทบ จะกล่าวได้ว่าอินเทอร์เน็ตกลายเป็นส่วนหนึ่งในชีวิตประจำวันของคนทั่วไป ซึ่งสิ่งหนึ่งที่ทำให้ อินเทอร์เน็ตได้รับความนิยม และถูกใช้อย่างแพร่หลายก็คือ การที่ผู้เล่นอินเทอร์เน็ตสามารถท่อง เว็บไซท์ (Website) ต่าง ๆ ได้อย่างง่ายดายผ่านเว็บเบราว์เซอร์ (Web Browser) นั้นเอง

ในช่วงแรก ๆ เว็บไซท์จะอยู่ในรูปแบบของ Static Website กล่าวคือ เว็บไซท์นั้น ๆ จะมีความสามารถเพียงแค่แสดงข้อมูลต่าง ๆ ออกมาในรูปแบบของข้อความ และรูปภาพเท่านั้น ไม่ สามารถตอบโต้กับผู้ใช้งาน ได้ ซึ่งเป็นการใช้งานที่ค่อนข้างจำกัด ยุคถัดมาจึงมีการพัฒนาเป็น รูปแบบของ Dynamic Website ซึ่งมีความสามารถโต้ตอบกับผู้ใช้งาน และติดต่อกับ Database ได้ เว็บแอปพลิเคชันจึงเกิดขึ้นมาเพื่อตอบสนองการทำธุรกรรมบนอินเทอร์เน็ตและความต้องการที่ เพิ่มขึ้นของผู้ใช้งาน

### 2.2.2 การทำงานของเว็บแอปพลิเคชัน

เว็บแอปพลิเคชัน (Web Application) คือ แอปพลิเคชันที่เข้าถึงโดยผ่านทางเว็บไซท์ และ ทำงานบนเครือข่าย เช่น อินเทอร์เน็ต (internet) บน โพรโทคอล HTTP (**H**ypertext **T**ransfer **P**rotocol) ซึ่งมีหลักการทำงาน ดังรูป 2.1



รูปที่ 2.1 หลักการทำงานของเว็บแอปพลิเคชัน

จากรูปจะเห็นว่าเว็บแอปพลิเคชันนั้นจะใช้ภาษาและเทคโนโลยีหลายประเภทประกอบกันในการพัฒนา ซึ่งโครงการนี้ใช้เทคโนโลยีดังที่จะกล่าวถึงต่อไป

### 2.2.3 ภาษา HTML

ภาษา HTML (**H**yper **T**ext **M**arkup **L**anguage) เป็นภาษาประเภท Mark up ซึ่งมีลักษณะเป็นแท็ก (Tag) ซึ่งเป็นตัวบอกเบราว์เซอร์ให้แสดงผลออกมาตามรูปแบบที่ต้องการ ภาษานี้ได้รับความนิยมอย่างมากในการใช้แสดงข้อมูลบนเว็บเพจ ซึ่งจะมีนามสกุลเป็น .htm หรือ .html โดยใช้ภาษา HTML เป็นหลักประกอบกับภาษาสคริปต์ เช่น JavaScript, VBScript, PHP และอื่น ๆ ในการพัฒนาเว็บไซต์ ตัวอย่างภาษา HTML ดังรูปที่ 2.7 ปัจจุบันเวอร์ชันล่าสุดของภาษานี้คือ HTML 5.0 องค์กรที่ทำหน้าที่กำหนดมาตรฐานคือ W3C (World Wide Web Consortium)

แต่ภาษานี้ยังคงมีข้อจำกัดหลายอย่างในตัวเอง กล่าวคือ เป็นภาษาที่ใช้แสดงผลเท่านั้น ไม่สามารถโต้ตอบกับผู้ใช้งานได้ และไม่สามารถอธิบายข้อมูลที่ตัวเองแสดงได้ จึงต้องใช้เทคโนโลยีอื่น ๆ เข้ามาใช้งานร่วมกัน

### 2.2.4 สไตล์ชีต CSS

CSS (**C**ascading **S**tyle **S**heet) เป็นภาษาประเภทสไตล์ชีต ใช้ในการกำหนดรูปแบบการแสดงผลของเอกสารที่เป็นภาษาประเภท Mark up ซึ่งส่วนใหญ่ก็คือ ภาษา HTML และ XHTML แต่ที่จริงแล้วสามารถใช้งานกับภาษาอื่น ๆ ได้อีก เช่น XML, SVG เป็นต้น โดยจะมีนามสกุลเป็น .CSS

คุณสมบัติเด่นที่ทำให้สไตล์ชีตนี้ได้รับความนิยม กล่าวคือ เป็นตัวควบคุมการแสดงผลของเว็บไซต์ไว้ในไฟล์ ๆ เดียว ซึ่งจะสะดวกต่อการปรับแต่งรูปแบบ โดยเฉพาะในแอปพลิเคชันขนาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใหญ่ และสามารถจัดการให้เป็นรูปแบบที่ต้องการได้เหมือนกันทั้งเว็บไซต์ เพียงแค่แก้ไขไฟล์ในที่เดียว ทั้งเว็บไซต์ก็จะเปลี่ยนเป็นรูปแบบที่ต้องการโดยอัตโนมัติ

### 2.2.5 ภาษาประเภทสคริปต์

ภาษาประเภทสคริปต์ (Scripting Languages) จัดเป็นภาษาสำหรับเขียนโปรแกรมแบบหนึ่ง แต่สิ่งที่ต่าง คือ เป็นภาษาที่มีวัตถุประสงค์สำหรับการพัฒนาเว็บแอปพลิเคชัน โดยส่วนใหญ่จะฝังตัวเองในโค้ด HTML เพื่อสร้างเว็บไซต์ที่มีลูกเล่น, สามารถตอบโต้กับผู้ใช้งานได้ สามารถแบ่งออกเป็น 2 ประเภท ดังนี้

#### 2.2.5.1 Client-side scripting

เป็นภาษาที่รันโดยเว็บเบราว์เซอร์ ซึ่งติดตั้งอยู่บนฝั่งไคลเอนต์ มีความสามารถตอบโต้กับผู้ใช้งาน, ทำการคำนวณต่าง ๆ และบริการคอนเทนต์แบบไดนามิก ภาษาประเภทนี้ ได้แก่ JavaScript และ VBScript

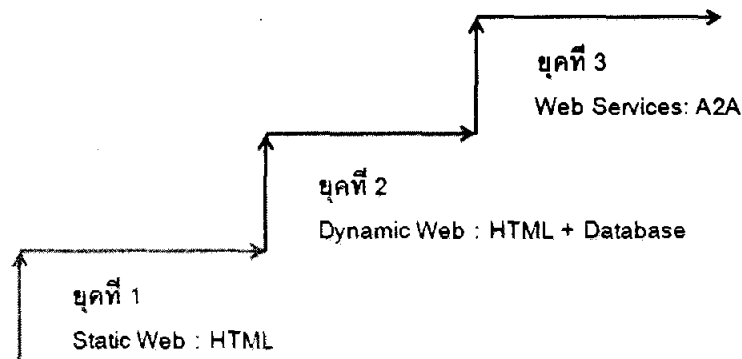
#### 2.2.5.2 Server-side scripting

เป็นภาษาที่รันบนฝั่งเซิร์ฟเวอร์ เมื่อมีการร้องขอจากไคลเอนต์ เว็บเซิร์ฟเวอร์ (Web Server) ส่งเพจกลับไปจะอยู่ในรูปแบบของ HTML ความสามารถของภาษาประเภทนี้ คือ สามารถดึงข้อมูลจากแหล่งข้อมูลต่าง ๆ โดยเฉพาะฐานข้อมูล ภาษาประเภทนี้ ได้แก่ ASP, ASP.NET, JSP, PHP และอื่น ๆ

## 2.3 เทคโนโลยีเว็บเซอร์วิส

### 2.3.1 บทนำ

จากหัวข้อที่ผ่านมากล่าวถึงการสร้างเว็บแอปพลิเคชันนั้น ยังคงมีขีดจำกัดในเรื่องการติดต่อกันระหว่างแอปพลิเคชันข้ามแพลตฟอร์ม จึงทำให้เกิดแนวคิด “เว็บเซอร์วิส (Web Services)” ขึ้นมา



รูปที่ 2.2 วิวัฒนาการของเว็บไซต์

### 2.3.2 นิยามของเว็บเซอร์วิส

คำว่า เว็บเซอร์วิส หรือ **Web Services** มีผู้ให้คำจำกัดความต่างกัน ไปมากมาย ซึ่งสามารถสรุปได้ว่าเว็บเซอร์วิสคือ แนวคิดที่สามารถทำให้เว็บแอปพลิเคชัน และ/หรือแอปพลิเคชันสามารถติดต่อ และแลกเปลี่ยนข้อมูล หรือบริการต่าง ๆ ได้โดยอัตโนมัติระหว่างเครื่องคอมพิวเตอร์เครื่องหนึ่งไปยังอีกเครื่องหนึ่งผ่านเครือข่ายด้วยโปรโตคอลพื้นฐานของเว็บเพจคือ HTTP (**H**yper**t**ext **T**ransfer **P**rotocol) และสามารถทำงานได้โดยไม่ขึ้นอยู่กับภาษาหรือแพลตฟอร์มใด ๆ (Platform Independent) ซึ่งมีภาษา XML (**e**X**t**ensible **M**arkup **L**anguage) เป็นภาษากลางในการสื่อสาร และมีโปรโตคอล SOAP (**S**imple **O**bject **A**ccess **P**rotocol) เป็นโปรโตคอลสำคัญในการสื่อสาร โดยจะทำงานร่วมกับเทคโนโลยีอื่น ๆ ที่เกี่ยวข้อง เช่น WSDL (**W**eb **S**ervices **D**escription **L**anguage) และ UDDI (**U**niversal **D**escription **D**iscovery and **I**ntegration) เป็นต้น

### 2.3.3 ลักษณะเด่นของเว็บเซอร์วิส

ลักษณะเด่นที่ทำให้เว็บเซอร์วิสเป็นเทคโนโลยีที่น่าสนใจ มีดังต่อไปนี้

- 1) Interoperability คือ ทำให้เว็บเซอร์วิสสามารถได้ต่อกันได้แม้ว่าจะเขียนด้วยภาษาต่างกัน หรืออยู่บนต่างแพลตฟอร์มกัน
- 2) Ubiquity คือ มีอยู่ทั่วไป และเข้าถึงได้ง่าย เนื่องจากเว็บเซอร์วิสได้ใช้ HTTP และ XML ในการสื่อสาร ซึ่งเทคโนโลยีสองชนิดนี้เป็นเทคโนโลยีพื้นฐานสำหรับเว็บไซต์ทั่วไป
- 3) Reusable Components คือ สามารถนำองค์ประกอบย่อยมาใช้ซ้ำได้ เช่น อัตรการแลกเปลี่ยนเงินตรา หรือรายงานพยากรณ์อากาศ ซึ่งช่วยลดการทำงานที่ซ้ำซ้อน
- 4) Loose Coupling คือ โครงสร้างการทำงานไม่ตายตัว พัฒนาได้ง่าย ไม่ว่าจะนำไปพัฒนาข้ามภาษา หรือข้ามเวอร์ชันก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) Modular Design คือ มีลักษณะเป็นส่วน ๆ นำมาประกอบกัน เห็นได้จากโครงสร้างของเว็บเซอร์วิสที่ประกอบขึ้นมาจากมาตรฐาน และภาษาที่หลากหลายหลาย ทำให้ง่ายต่อการแยกพัฒนาในแต่ละส่วน เช่น ด้าน Security, ด้าน Reliability และด้าน Transaction เป็นต้น

### 2.3.4 หลักการทำงานของเว็บเซอร์วิส

การทำงานของเว็บเซอร์วิสได้ถูกจำลองอยู่ในรูปแบบเว็บเซอร์วิสโมเดล (Web Service Model) โดยจะมีการโต้ตอบกันระหว่างสามส่วนหลัก ๆ ซึ่งประกอบไปด้วย เซอร์วิสโพรไวเดอร์ (Service Provider), เซอร์วิสรีควีสเตอร์ (Service Requestor) และเซอร์วิสรีจิสทรี (Service Registry) โดยมีกระบวนการ (Operation) ที่เกี่ยวข้องคือ การประกาศ (Publish), การค้นหา (Find) และการเรียกใช้ (Bind) ซึ่งจะกระทำบนเซอร์วิส และเซอร์วิสเดสคริปชัน (Service Description) ซึ่งเป็นตัวอธิบายคุณลักษณะของเซอร์วิสนั้น ดังรูปที่ 2.3 และ 2.4



รูปที่ 2.3 รูปแบบจำลองเว็บเซอร์วิส

ในหลักการการทำงานของเว็บเซอร์วิส นั้น เซอร์วิสโพรไวเดอร์จะให้บริการตัวโมดูลซอฟต์แวร์ของเว็บเซอร์วิสผ่านเครือข่าย มีการจำกัดความเซอร์วิสบนเซอร์วิสเดสคริปชัน พร้อมทั้งประกาศตัวเองไปยังเซอร์วิสรีควีสเตอร์ หรือเซอร์วิสรีจิสทรี จากนั้นเซอร์วิสรีควีสเตอร์จะทำการค้นหาเซอร์วิสเดสคริปชันจากเซอร์วิสรีจิสทรี แล้วใช้เซอร์วิสเดสคริปชันนั้นเรียกใช้เซอร์วิส ร้องขอ และโต้ตอบกับเว็บเซอร์วิสนั้น ๆ

#### 2.3.4.1 ส่วนประกอบหลักในโครงสร้างของเว็บเซอร์วิส

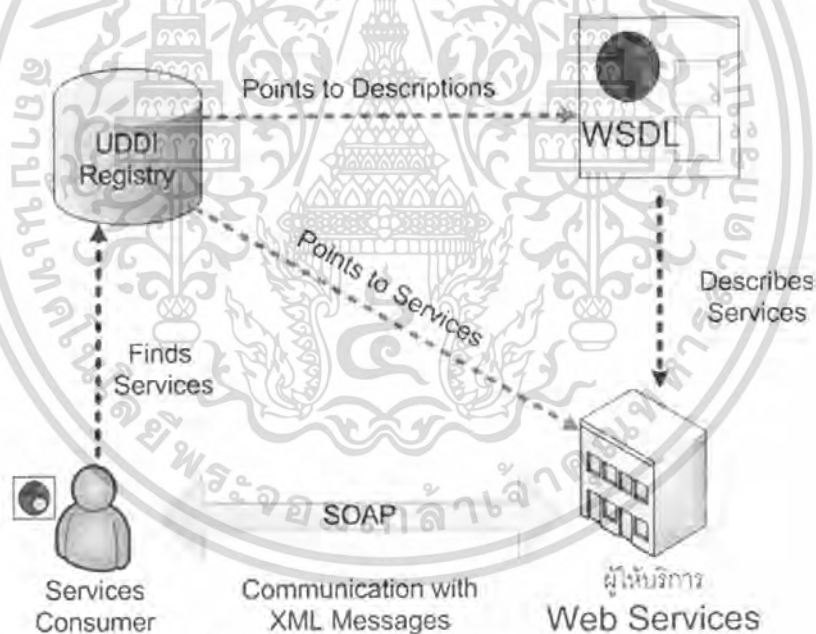
- 1) Service Provider คือ ผู้ให้บริการ ในที่นี้คือเจ้าของเซอร์วิส นั้น ๆ หรือ แพลตฟอร์มที่จะเข้าถึงเซอร์วิส
- 2) Service Requestor คือ ผู้ร้องขอบริการจากเซอร์วิส นั้น ๆ ในที่นี้คือการที่แอปพลิเคชันทำการค้นหา และเรียกใช้เซอร์วิส

เอกสารนี้เป็นเอกสารที่สงวนไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่สามารถนำข้อมูลไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) Service Registry คือ ผู้ที่ทำหน้าที่เปรียบเสมือนตัวแทนผู้ให้บริการ ซึ่งสามารถทำการค้นหาเซอร์วิสเดสคริปชันที่เซอร์วิส โพรไวเดอร์ได้ทำการประกาศไว้ได้ และเซอร์วิสรีเคสเตอร์ที่ค้นหาเซอร์วิสอยู่ก็จะได้รับข้อมูลที่อยู่ในเซอร์วิสเดสคริปชันนั้นเพื่อนำไปใช้งานต่อไป

### 2.3.4.2 กระบวนการที่เกิดขึ้นในโครงสร้างของเว็บเซอร์วิส

- 1) การประกาศ (Publish) เป็นกระบวนการที่ทำให้เซอร์วิสรีเคสเตอร์สามารถหาเซอร์วิสเจอ โดยการหาจากเซอร์วิสเดสคริปชันที่ถูกประกาศไว้
- 2) การค้นหา (Find) เป็นกระบวนการที่เซอร์วิสรีเคสเตอร์ดึงหรือค้นหาเซอร์วิสเดสคริปชันด้วยตัวเอง หรือทำการจัดเรียงข้อมูล (query) จากเซอร์วิสรีจิสทรีเพื่อหาเซอร์วิสที่ต้องการ
- 3) การเรียกใช้ (Bind) เกิดขึ้นเมื่อเซอร์วิสถูกร้องขอ โดยเซอร์วิสรีเคสเตอร์จะทำการร้องขอและเริ่มการโต้ตอบกับเซอร์วิสนั้น



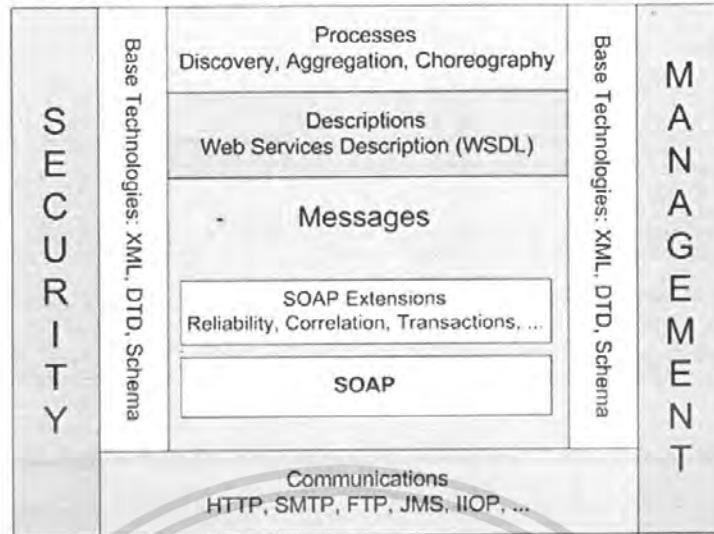
รูปที่ 2.4 กระบวนการทำงานของเว็บเซอร์วิส

### 2.3.5 องค์ประกอบหลักของเว็บเซอร์วิส

เทคโนโลยีเว็บเซอร์วิสมีองค์ประกอบหลักอยู่ในรูปแบบของสแต็ก ที่แบ่งออกเป็นเลเยอร์ ดังรูปที่ 2.5 และ 2.6 เพื่อทำกระบวนการประกาศ, เรียกใช้ และค้นหา ให้เป็นไปในลักษณะของ interoperable จึงจำเป็นจะต้องมีมาตรฐานมาเป็นตัวควบคุมในแต่ละเลเยอร์ โดยเลเยอร์บนจะถูกสร้างขึ้นตามความสามารถที่เลเยอร์ล่างจัดเอาไว้ให้ ส่วนแท่งในแนวตั้งเป็นเทคโนโลยีที่จำเป็นต่อเอกสารที่เป็นเอกสารที่ส่งวนไว้สำหรับการทำงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า มีในเลเยอร์นั้น ๆ

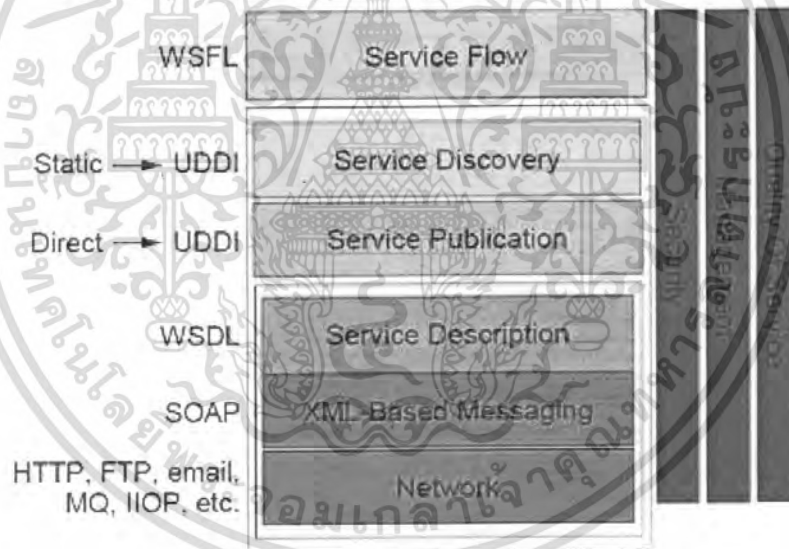
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### W3C Web Services Stack



รูปที่ 2.5 แนวคิดเว็บเซอร์วิสในรูปแบบเสต็กของ W3C

### IBM Conceptual Web Services



รูปที่ 2.6 แนวคิดเว็บเซอร์วิสในรูปแบบเสต็กของ IBM

#### 2.3.5.1 Communications

เป็นระดับขั้นพื้นฐานของการทำงานของเว็บเซอร์วิส เนื่องจากเว็บเซอร์วิสจะถูกเซอร์วิสรีเคสเตอร์เรียกใช้ผ่านเครือข่าย และตัวเว็บเซอร์วิสเองจะถูกประกาศบนอินเทอร์เน็ต

#### 2.3.5.2 XML-based messaging

เป็นเลขอร์ที่ใช้ประโยชน์ของ XML เป็นพื้นฐานในการส่งผ่านเมสเซจ โดยโปรโตคอล XML-based messaging ที่เลือกใช้ คือโปรโตคอล SOAP ซึ่งเป็นโปรโตคอลสื่อสารที่เอกสารจัดว่าเป็นมาตรฐานของ W3C การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.5.3 Descriptions

เป็นเลขอร์ที่ใช้ในการอธิบายคุณลักษณะของเว็บเซอร์วิส และวิธีการเข้าถึงเว็บเซอร์วิส นั้น ๆ ซึ่งก็คือเซอร์วิสเดสคริปชันที่เป็นภาษา XML-based ภาษาที่ใช้ได้แก่ ภาษา WSDL จัดเป็นมาตรฐานประเภท de facto แต่ยังไม่เป็นมาตรฐานของ W3C

### 2.3.5.4 Discovery

เป็นการรวบรวมไดเรกทอรี (Directory) ของเซอร์วิสเดสคริปชันไว้เพื่อให้ง่ายต่อการค้นพบ โดยบริการที่นิยมมาใช้คือ UDDI เนื่องจากจะไม่พบเว็บเซอร์วิสที่ไม่มีการประกาศเอาไว้ UDDI จึงถูกนำมาเพิ่มเข้าไปในส่วนของภาษา WSDL กล่าวได้ว่า UDDI คือ ที่รวบรวมไดเรกทอรีของเซอร์วิสที่ธุรกิจ และองค์กรต่าง ๆ สามารถค้นหาข้อมูลเกี่ยวกับเว็บเซอร์วิสได้ผ่านอินเทอร์เน็ต โดยใช้โปรโตคอล SOAP

### 2.3.5.5 XML

เป็นภาษา Markup ที่ใช้อธิบายข้อมูล และเน้นที่ตัวข้อมูลว่าเป็นอะไร ซึ่งกลายมาเป็นภาษามาตรฐานที่ใช้เป็นสื่อกลางในการแลกเปลี่ยนข้อมูลบนอินเทอร์เน็ต โดยลักษณะของ XML จะขอล่าถึงในหัวข้อถัดไป

## 2.4 ภาษา XML

### 2.4.1 บทนำ

เมื่อกล่าวถึงเว็บเซอร์วิส คงจะไม่กล่าวถึงภาษา XML ไม่ได้ เพราะภาษานี้เป็นตัวหลักที่ทำให้แนวคิดเว็บเซอร์วิสเป็นจริงขึ้นมาได้ หรืออาจกล่าวได้อีกนัยหนึ่งว่า เว็บเซอร์วิสทำงานบนภาษา XML ร่วมกับโปรโตคอล HTTP นั่นเอง ทั้งนี้เนื่องจากลักษณะเด่นของ XML คือ ไม่ขึ้นอยู่กับแพลตฟอร์มใด ๆ และสามารถทำงานร่วมกับภาษาอื่นได้ทุกภาษา ดังนั้น ภาษา XML จึงกลายเป็นภาษามาตรฐานที่ใช้เป็นสื่อกลางในการแลกเปลี่ยนข้อมูล และเรียกใช้งานเว็บเซอร์วิสผ่านเครือข่ายอินเทอร์เน็ต

### 2.4.2 ลักษณะของภาษา XML

ภาษา XML หรือ eXtensible Markup Language เป็นภาษา markup ชนิดหนึ่ง ซึ่งเป็นประเภทหนึ่งของภาษา SGML (Standard Generalized Markup Language) เช่นเดียวกับภาษา HTML และเป็น W3C Recommendation

ภาษา XML ได้ถูกกล่าวถึงว่า “XML does not **DO** anything” ([www.w3schools.com/xml](http://www.w3schools.com/xml)) ซึ่งหมายความว่า ภาษา XML เป็นภาษาที่ใช้สำหรับนิยามหรืออธิบายข้อมูล โดยที่ตัวมันเองไม่ทำอะไรเลย ไม่สนใจลักษณะการแสดงผลของข้อมูล รวมไปถึงการสื่อสารผ่านเครือข่ายก็ต้องอาศัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาตรฐานอื่นเข้ามาช่วย ในการเขียนภาษา XML เราต้องสร้างแท็ก (Tag) ขึ้นมาเอง ไม่มีแท็กกำหนด มาให้ตายตัว ทำให้มีความเป็นอิสระ และมีความยืดหยุ่นสูงในการใช้งาน โดยใช้ XML Schema หรือ DTD (**D**ocument **T**ype **D**efinition) ในการอธิบายตัวข้อมูลเอง (Self-descriptive)

### 2.4.3 ข้อแตกต่างระหว่าง XML และ HTML

ภาษา HTML ได้รับความนิยมนอย่างมากในการใช้แสดงข้อมูลบนเว็บเพจ แต่เนื่องจาก ข้อจำกัดของภาษา HTML ที่ไม่สามารถบอกได้ว่าข้อมูลที่แสดงอยู่เป็นข้อมูลอะไร ดังตัวอย่าง โค้ด ในรูปที่ 2.7 และ 2.8 มนุษย์เราเมื่อเห็น โค้ดด้านล่างจะสามารถบอกได้ทันทีว่าเป็นข้อมูลเกี่ยวกับชื่อ คน แต่เครื่องคอมพิวเตอร์จะไม่สามารถบอกได้ว่าข้อมูลนั้นคืออะไร รู้เพียงแต่ว่าต้องแสดงออกมา แบบไหน ดังนั้น ภาษา XML จึงได้เข้ามาแก้ปัญหาจุดนี้

```
<html>
<head> </head>
<body>
<b>Member List</b><br>
My name is Chip Brown.<br>
</body>
</html>
```

รูปที่ 2.7 ตัวอย่างโค้ดภาษา HTML

```
<?xml version="1.0" encoding="UTF-8"?>
<member>
  <first-name>Chip</first-name>
  <last-name>Brown</last-name>
</member>
```

รูปที่ 2.8 ตัวอย่างโค้ดภาษา XML

จากที่ได้กล่าวมาข้างต้น ข้อแตกต่างระหว่างสองภาษา คือ XML ถูกออกแบบมาเพื่ออธิบาย ข้อมูลและสนใจว่าข้อมูลคืออะไร ในขณะที่ HTML ถูกออกแบบมาเพื่อแสดงผลข้อมูลเท่านั้น เป็นการนำมาทำงานร่วมกันเพื่อเพิ่มประสิทธิภาพในการทำงาน

#### 2.4.4 จุดเด่นของภาษา XML

- มีความยืดหยุ่นสูง เนื่องจากถูกออกแบบมาให้สามารถนิยามความหมายของข้อมูล และสามารถสร้างแท็กขึ้นมาเองได้ มีการจัดโครงสร้างข้อมูลเป็นหมวดหมู่ และส่วนประกอบย่อย
- สามารถทำงานโดยไม่ขึ้นกับแพลตฟอร์มใด ๆ ทำให้สามารถทำงานข้ามแพลตฟอร์ม และทำงานร่วมกับภาษาอื่น ๆ ได้
- ภาษา XML แยกส่วนของข้อมูลออกจาก HTML เพื่อนำไปเก็บในรูปแบบของไฟล์ xml เมื่อข้อมูลมีการเปลี่ยนแปลงจะสามารถจัดการกับข้อมูลได้โดยไม่ต้องเข้าไปยุ่งเกี่ยวกับ HTML ที่ใช้ในการแสดงข้อมูล แต่ก็สามารถเก็บข้อมูล xml ไว้ร่วมกับ html ได้เช่นกัน ซึ่งจะเรียกว่า “Data Islands”
- ทำให้การแลกเปลี่ยนข้อมูลระหว่างแอปพลิเคชัน หรือระบบต่าง ๆ ง่ายขึ้น ลดเวลาการทำงานของผู้พัฒนาระบบที่ต้องการพัฒนาให้ระบบสามารถแลกเปลี่ยนข้อมูลระหว่างกันได้โดยไม่ต้องส่งผลกระทบต่อรูปแบบของข้อมูลภายในตัวแอปพลิเคชัน หรือระบบเลย
- ทำให้การสื่อสารระหว่างธุรกิจ และ/หรือองค์กร หรือที่เรียกว่า B2B (Business-to-Business communication) ง่ายขึ้น และมีการแลกเปลี่ยนข้อมูลระหว่างกันได้ เช่น ข้อมูลทางการเงิน ซึ่งควรเป็นมาตรฐานเดียวกันทั่วโลก
- XML ทำให้เข้าถึงข้อมูลได้ง่ายผู้ใช้งาน และแอปพลิเคชันต่าง ๆ สามารถเข้าถึงไฟล์ xml ได้เสมือนเป็นแหล่งข้อมูล หรือระบบฐานข้อมูล
- XML ทำให้ข้อมูลมีความถูกต้องสูง ตัวอย่างที่เห็นได้ชัด คือ ผลลัพธ์ที่ได้จากการค้นหา (search) จะได้ข้อมูลที่ตรงตามความต้องการ เนื่องจากมีการระบุแท็กที่ชัดเจน

#### 2.4.5 XML Syntax

##### 2.4.5.1 โครงสร้างภาษา

ประกอบด้วย ส่วนหลัก ๆ 2 ส่วน ดังแสดงในรูปที่ 2.9

1) ส่วน **Prolog** เป็นส่วน XML Declaration คือ `<?xml version="1.0" ... ?>`

2) ส่วนของ **Elements** จะต้องประกอบด้วย

- Root element ที่ครอบคลุมอีลิเมนต์อื่นๆ ในที่นี้คือ `<member>...</member>`
- Child element ที่เป็นส่วนประกาศข้อมูล ในที่นี้คือ `<first-name>...</first-name>`, `<last-name>...</last-name>`

```
<?xml version="1.0" encoding="UTF-8"?>
<member>
  <first-name>Chip</first-name>
  <last-name>Brown</last-name>
</member>
```

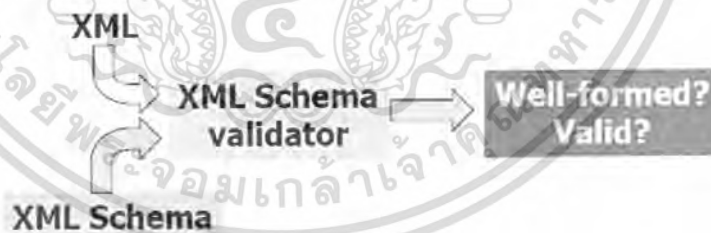
รูปที่ 2.9 โครงสร้างภาษา XML

#### 2.4.5.2 Correctness in XML Document

เนื่องจากภาษา XML ขอมให้สร้างแท็กขึ้นมาเองได้ ดังนั้นจึงจำเป็นต้องมีมาตรฐานที่ใช้วัดความถูกต้องของเอกสาร XML ขึ้นมา โดยมี 2 ประเภท ได้แก่

- 1) **Well-formed XML** คือ เอกสาร xml ที่มี syntax ถูกต้อง ดังตัวอย่างในรูปที่ 2.10 และ 2.11 ซึ่งไม่ได้บอกว่าข้อมูลนี้มีความถูกต้องและตรงตามความต้องการที่จะนำไปใช้หรือไม่ จึงต้องมีการ validation ดังข้อ 2
- 2) **Valid XML** คือเอกสาร xml ที่มีโครงสร้างข้อมูลถูกต้อง และครบถ้วน ซึ่งใช้ XML Schema หรือ DTD ในการอธิบายตัวข้อมูลดังตัวอย่างในรูปที่ 2.10 และ 2.11

```
<location>
  <latitude>8.9</latitude>
  <longitude>60.5</longitude>
  <altitude>2000</altitude>
</location>
```



```
<xs:schema>
  ...
  ...
  ...
</xs:schema>
```

รูปที่ 2.10 การตรวจสอบความถูกต้องในเอกสาร XML

ในรูปที่ 2.11 จะเห็นได้ว่า ข้อมูลใน XML#1 ไม่ครบถ้วน แต่ข้อมูลใน XML#2 อธิบาย <location> ได้ครบถ้วนและถูกต้อง คือ มี <latitude>, <longitude> และ <altitude> ข้อมูล XML ทั้งสองจึงเป็น well-formed xml แต่มีเพียง XML # 2 เท่านั้นที่เป็น valid xml data เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
1 <location>
  <latitude>8.9</latitude>
  <longitude>60.5</longitude>
</location>
```

```
2 <location>
  <latitude>8.9</latitude>
  <longitude>60.5</longitude>
  <altitude>2000</altitude>
</location>
```

ถ้าข้อบ่งชี้ของข้อมูล <location> คือ "ต้อง" ประกอบด้วย latitude longitude และ altitude

รูปที่ 2.11 เอกสาร XML ที่เป็น Well-formed XML และ Valid XML

### 2.4.6 XML Schema

XML Schema มีจุดประสงค์เพื่อการอธิบายและกำกับโครงสร้างของเอกสาร xml ในตัวของมันเอง เขียนอยู่ในรูปของภาษา XML โดยเป็นตัวกำหนดกฎที่เอกสาร xml จะต้องเป็นไปเพื่อให้แน่ใจว่าโครงสร้างในเอกสาร xml นั้นเป็นไปอย่างถูกต้องตามที่ต้องการ เป็นเหมือนกับรูปแบบในการเขียน elements และ attributes ดังรูปที่ 2.12 XML Schema

<pre>&lt;location&gt;   &lt;latitude&gt;8.9&lt;/latitude&gt;   &lt;longitude&gt;60.5&lt;/longitude&gt;   &lt;altitude&gt;2000&lt;/altitude&gt; &lt;/location&gt;</pre>	<pre>&lt;xs:schema&gt;   ...   ...   ... &lt;/xs:schema&gt;</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------

XML

XML Schema

โครงสร้างของข้อมูล (data structure)

Schema ทำหน้าที่บอกว่า ...

"element <location> ต้องประกอบด้วย 3 elements - <latitude>, <longitude>, <altitude>"

ชนิดของข้อมูล (datatype)

Schema ทำหน้าที่บอกว่า ...

"<latitude> ต้องเป็นตัวเลข (มีทศนิยม 1 ตำแหน่ง)  
<altitude> ต้องเป็นจำนวนเต็ม ฯลฯ"

รูปที่ 2.12 เอกสาร XML และ XML Schema

## 2.4.7 ตัวอย่างการทำงานร่วมกันระหว่างเอกสาร XML และ XML Schema

```
<?xml version="1.0"?>

<note xmlns="http://www.w3schools.com"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="note.xsd">

  <to>George</to>
  <from>Laura</from>
  <heading>Reminder</heading>
  <body>Don't forget ... !</body>
</note>
```

รูปที่ 2.13 ตัวอย่างเอกสาร XML (note.xml) ซึ่งมีการอ้างอิงถึง schema (note.xsd)

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="note">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="to" type="xs:string"/>
        <xs:element name="from" type="xs:string"/>
        <xs:element name="heading" type="xs:string"/>
        <xs:element name="body" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

รูปที่ 2.14 ตัวอย่างเอกสาร XML Schema (note.xsd)

## 2.4.8 RSS Feed

RSS (**R**eally **S**imple **S**yndication) เป็นรูปแบบหนึ่งของเว็บฟีด (Web Feed) กล่าวคือ เป็นวิธีการที่ใช้ XML ในการกระจายคอนเทนต์จากเว็บไซต์หนึ่งไปยังเว็บอื่น ๆ ซึ่งมีการเปลี่ยนคอนเทนต์ให้ใหม่อยู่เสมอโดยอัตโนมัติ นามสกุลของไฟล์ คือ .rss หรือ .xml ประโยชน์ในการใช้งานคือ สะดวกในการกระจายข่าวสารให้ใหม่อยู่เสมอ และสะดวกต่อผู้อ่านที่จะได้รับข่าวสารที่ตนสนใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 สัญลักษณ์ของ RSS Feed

## 2.5 โพรโทคอล SOAP

### 2.5.1 บทนำ

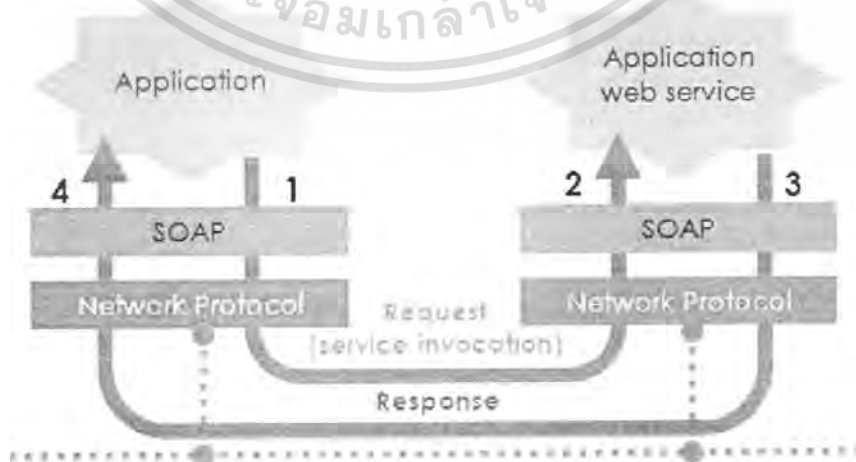
แม้เราจะทำเว็บเซอร์วิสของเราออกมาได้สมบูรณ์แบบแค่ไหน หากเว็บเซอร์วิสไม่สามารถติดต่อกับใครได้ก็ไม่มีประโยชน์ เราจึงต้องมีโพรโทคอลเข้ามาช่วยให้เว็บเซอร์วิสสามารถติดต่อสื่อสารกันผ่านเครือข่ายได้ ซึ่งโดยส่วนมากจะเป็น HTTP ซึ่งโพรโทคอลตัวหนึ่งที่ขึ้นมาเป็นมาตรฐานได้อย่างรวดเร็วก็คือ โพรโทคอล SOAP นั่นเอง

### 2.5.2 คำอธิบายของ SOAP

SOAP หรือ **S**imple **O**bject **A**ccess **P**rotocol เป็นโพรโทคอลสำหรับการแลกเปลี่ยนข้อมูลในสภาพแวดล้อมแบบกระจาย (decentralized, distributed environment) หรือการทำงานที่ไม่ต้องมีศูนย์กลาง โดยจะมีการกำหนดเมสเสจจิงโพรโทคอล (Messaging Protocol) เพื่อการสื่อสารระหว่างผู้ขอบริการ กับผู้ให้บริการ

SOAP เป็น XML-based Message ดังนั้น SOAP จึงถูกออกแบบขึ้นมาบนลักษณะเด่น 2 ข้อ คือ เข้าใจง่าย และมีความยืดหยุ่นสูง

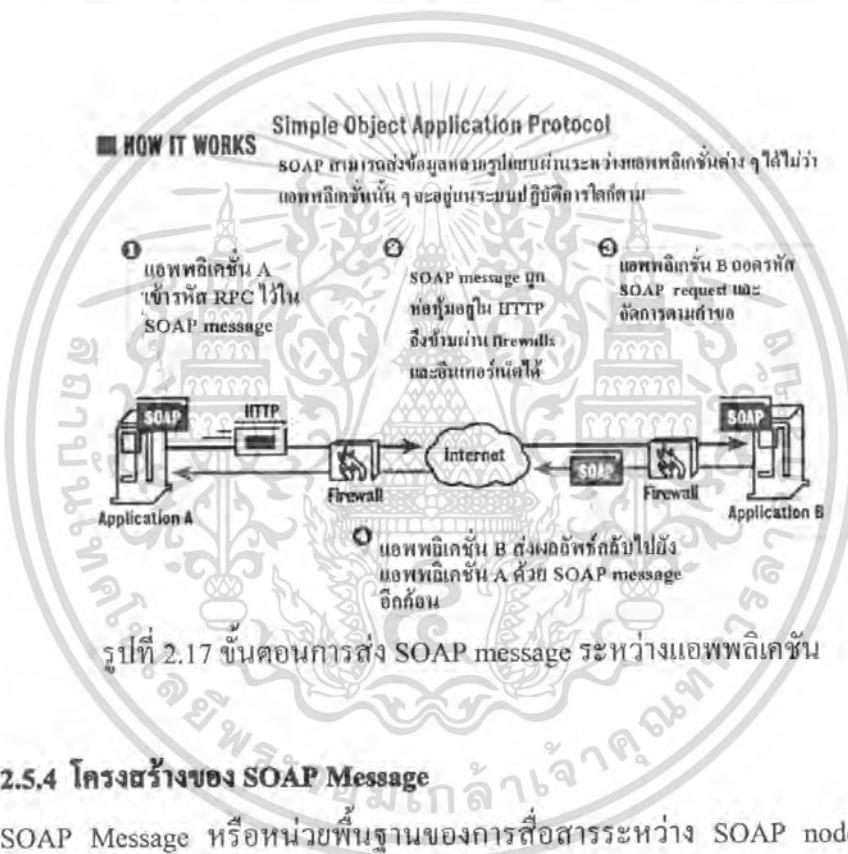
### 2.5.3 กระบวนการทำงานของ SOAP



รูปที่ 2.16 หลักการทำงานของ SOAP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) แอปพลิเคชันร้องขอบริการโดยการสร้าง SOAP message ส่งไปยังผู้ให้บริการเว็บเซอร์วิส
- 2) SOAP ส่งเมสเสจที่ได้รับไปให้ผู้ให้บริการเว็บเซอร์วิส โดย SOAP จะรับผิดชอบการแปลงเมสเสจที่ได้รับไปเป็นภาษาโปรแกรมที่แอปพลิเคชันเข้าใจ
- 3) เว็บเซอร์วิสทำตามคำร้องขอและสร้าง Response เป็น SOAP message ส่งกลับไปยังผู้ร้องขอ
- 4) ฟังก์ชันได้รับเมสเสจตอบกลับโดยผ่าน SOAP ที่แปลง SOAP message จากภาษา XML ไปเป็นภาษาโปรแกรมก่อน แล้วจึงส่งต่อให้แก่ตัวแอปพลิเคชัน

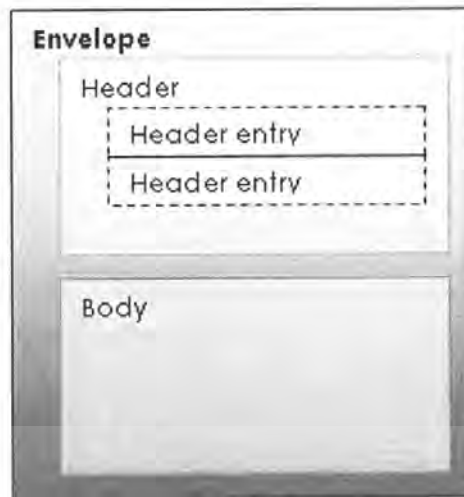


#### 2.5.4 โครงสร้างของ SOAP Message

SOAP Message หรือหน่วยพื้นฐานของการสื่อสารระหว่าง SOAP nodes ดังรูปที่ 2.18 ประกอบด้วย

- 1) **SOAP envelope** คือ ข้อมูลส่วนที่อยู่นอกสุดของ SOAP message ภายในประกอบด้วย SOAP header และ SOAP body
- 2) **SOAP header** คือ กลุ่มของ SOAP header block ซึ่งสามารถมีได้ตั้งแต่ 0 ขึ้นไป โดยอาจเป็นข้อมูลเกี่ยวกับผู้รับ (SOAP receiver) หรือเส้นทางของ SOAP message ก็ได้
- 3) **SOAP body** คือ กลุ่มของข้อมูลที่มีเป้าหมายอยู่ที่ SOAP receiver ตัวสุดท้ายของเส้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 โครงสร้างของ SOAP

### 2.5.5 ลักษณะเด่นของ SOAP

- มีขนาดเล็ก ใช้งานง่าย
- ไม่ขึ้นกับโปรโตคอล
- ไม่ขึ้นกับรูปแบบ และระบบปฏิบัติการ
- ไม่ขึ้นกับภาษาที่เขียน และเป็นกลางไม่มีใครเป็นเจ้าของ
- มีความปลอดภัยโดยอนุญาตให้เฉพาะ โปรแกรมหที่กำหนดไว้เท่านั้นสามารถเรียกใช้ SOAP ได้

## 2.6 ภาษา WSDL

### 2.6.1 บทนำ

หากถามว่าผู้ใช้บริการจะทราบได้อย่างไรว่า เว็บเซอร์วิสที่ต้องการนั้นมีลักษณะเป็นอย่างไร, ต้องส่งค่าอะไรไป, ได้อะไรกลับมา หรือจะเรียกใช้ได้อย่างไรนั้น คำตอบคือ WSDL ที่ช่วยให้การเรียกใช้บริการเป็นไปอย่างง่ายคยทั้งสำหรับผู้ให้บริการที่ต้องการอธิบาย และผู้ใช้บริการที่ต้องการทราบข้อมูล

### 2.6.2 ความหมายของ WSDL

WSDL หรือ **Web Services Description Language** เป็นเอกสาร XML ที่ทำงานตามชื่อของมัน คือ อธิบายลักษณะของตัวเว็บเซอร์วิส, ระเบียบวิธีการทำงาน, การให้บริการ รวมไปถึงแหล่งที่มา และวิธีการติดต่อกับเว็บเซอร์วิสนั้น ๆ อีกด้วย โดย WSDL มีการระบุค่าต่าง ๆ ดังรูปที่

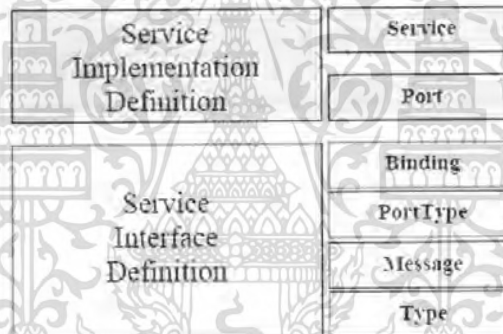
2.19 และ 2.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Types	Message	Operation
Port Type	Binding	Services

รูปที่ 2.19 คำอธิบายเซอร์วิสที่ใช้ในภาษา WSDL

- **Types** – ใช้ในการประกาศชนิดของตัวแปรที่ไม่ใช่ตัวแปรทั่วไป
- **Message** – ระบุรายละเอียดของเมสเสจที่ใช้สื่อสารระหว่างผู้ใช้บริการกับเซอร์วิส
- **Operation** – รายละเอียดของฟังก์ชันที่ให้บริการ
- **Port Type** – กำหนดโอเปอร์เรชัน และประกาศเมสเสจที่ใช้เป็นอินพุต และเอาท์พุต
- **Binding** – กำหนดลักษณะการส่งเมสเสจ
- **Port** – ระบุการเชื่อมต่อที่ชี้ไปยัง web services
- **Services** – กำหนด URL ของเว็บเซอร์วิส



รูปที่ 2.20 คำนิยามอินเทอร์เฟซ และอิมพลีเมนต์เทชันของเซอร์วิส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.6.3 โครงสร้างของเอกสาร WSDL

```

<definitions>
  <types>
    definition of types.....
  </types>

  <message>
    definition of a message....
  </message>

  <portType>
    definition of a port.....
  </portType>

  <binding>
    definition of a binding....
  </binding>
</definitions>

```

รูปที่ 2.21 รูปแบบโครงสร้างของเอกสาร WSDL

### 2.6.4 ตัวอย่างเอกสาร WSDL

```

<message name="getTermRequest">
  <part name="term" type="xs:string"/>
</message>

<message name="getTermResponse">
  <part name="value" type="xs:string"/>
</message>

<portType name="glossaryTerms">
  <operation name="getTerm">
    <input message="getTermRequest"/>
    <output message="getTermResponse"/>
  </operation>
</portType>

```

รูปที่ 2.22 ตัวอย่างเอกสาร WSDL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6.5 ตัวอย่างการ Binding กับ SOAP

```

<message name="getTermRequest">
  <part name="term" type="xs:string"/>
</message>

<message name="getTermResponse">
  <part name="value" type="xs:string"/>
</message>
<portType name="glossaryTerms">
  <operation name="getTerm">
    <input message="getTermRequest"/>
    <output message="getTermResponse"/>
  </operation>
</portType>
<binding type="glossaryTerms" name="b1">
<soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
  <operation>
    <soap:operation
      soapAction="http://example.com/getTerm"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
</binding>

```

รูปที่ 2.23 ตัวอย่างเอกสาร WSDL ที่ binding กับ SOAP

## 2.7 ภาษา PHP

### 2.7.1 บทนำ

PHP หรือ Hypertext Preprocessor เป็นภาษาประเภท Server-side script ที่ใช้ในการพัฒนาเว็บไซต์แบบไดนามิก ซึ่งนิยมใช้งานกันอย่างแพร่หลาย เนื่องจากดาวน์โหลดใช้งานได้ฟรีและเป็นซอฟต์แวร์โอเพ่นซอร์ส (open source software) จึงเป็นอีกทางเลือกหนึ่งนอกจากภาษา ASP และ JSP

ภาษานี้ถือกำเนิดขึ้นโดย Rasmus Lerdorf ในปี 1994 โดยปัจจุบันเป็น PHP เวอร์ชัน 5 ซึ่งมีการสนับสนุนการเขียนโปรแกรมเชิงวัตถุ (OOP) และเว็บไซต์อย่างเป็นทางการคือ <http://php.net/> นอกจากนี้ยังถูกนำไปพัฒนาแอปพลิเคชันที่ได้รับความนิยม เช่น phpBB, WordPress และ MediaWiki

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 สัญลักษณ์ของภาษา PHP

### 2.7.2 การพัฒนาเว็บแอปพลิเคชันด้วยภาษา PHP

การพัฒนาสามารถทำได้โดยการฝังโค้ด PHP เข้าไปในโค้ดภาษา HTML ดังโค้ดรูปที่ 2.25 ซึ่งสามารถนำไปแทรกส่วนไหนก็ได้ในหน้าเพจ HTML แต่ต้องบันทึกนามสกุลเป็น .php ลักษณะไวยากรณ์ที่ใช้จะใกล้เคียงกับภาษาเพิร์ล (Perl) และภาษาซี (C) ส่วนใหญ่นิยมรันบนเว็บเซิร์ฟเวอร์ Apache ซึ่งทำงานได้ในหลายระบบปฏิบัติการ แต่ที่จริงสามารถรันบนเว็บเซิร์ฟเวอร์อื่น ๆ ได้

ในส่วนของการทำงานติดต่อกับฐานข้อมูลนั้น ภาษานี้สนับสนุนการทำงานกับโปรแกรมจัดการฐานข้อมูลหลายยี่ห้อ แต่ที่นิยมใช้งานมากที่สุดคือ โปรแกรม MySQL ซึ่งสนับสนุนภาษา SQL มาตรฐาน, สามารถดาวน์โหลดใช้งานฟรี, และเหมาะสำหรับการใช้งานของแอปพลิเคชันขนาดกลางและเล็ก

```
<?php
function hello
{
    echo "Hello World!\n";
}

hello();
?>
```

รูปที่ 2.25 ตัวอย่างโค้ดภาษา PHP

### 2.7.3 จุดเด่นของภาษา PHP

- 1) สามารถทำงานบนแพลตฟอร์มใด ๆ ก็ได้ เช่น Windows, Unix และ Linux
- 2) สามารถทำงานบนเว็บเซิร์ฟเวอร์ยี่ห้อใดก็ได้ เช่น Apache, IIS และอื่น ๆ
- 3) สามารถดาวน์โหลดใช้งานได้ฟรีและเป็นโอเพ่นซอร์ส
- 4) เรียนรู้ได้ง่ายมาก โดยเฉพาะผู้ที่เคยเขียนภาษา C มาก่อน

### 2.7.4 ภาษา PHP กับเว็บเซอร์วิส

การพัฒนาเว็บเซอร์วิสด้วยภาษา PHP นั้นสามารถสร้างได้ทั้งเป็นเซอร์วิสรีเคสเตอร์และเซอร์วิสโพรไวเดอร์ ตามมาตรฐานของ W3C ซึ่งปัจจุบัน PHP 5 มีฟังก์ชันสนับสนุนการพัฒนาเว็บเซอร์วิสในตัวเอง หรือ ใช้ไลบรารี NuSoap ซึ่งเป็นตัวสนับสนุนการสร้างเว็บเซอร์วิสในการสร้างเว็บเซอร์วิสก็ได้ (สามารถหาดาวน์โหลดได้ฟรี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในปริญญานิพนธ์ฉบับนี้ได้มีการกล่าวถึงการสร้างและทดลองเว็บเซอร์วิสด้วยภาษา PHP ร่วมกับไลบรารี NuSoap ซึ่งรายละเอียดได้กล่าวไว้ในภาคผนวก ข

## 2.8 ภาษา ASP.NET

### 2.8.1 คอทเน็ตเฟรมเวิร์ก (.NET Framework)

คอทเน็ตเฟรมเวิร์กเป็นโครงสร้างหรือกรอบการทำงานของไมโครซอฟท์ ซึ่งเตรียมสภาวะแวดล้อมสำหรับสร้าง, ดีพลอย (Deploy), และรัน เว็บเซอร์วิส, เว็บแอปพลิเคชัน รวมไปถึงแอปพลิเคชันบนมือถือ เป็นต้น ภายในเฟรมเวิร์กจะประกอบด้วยคลาสไลบรารีพื้นฐาน เช่น ADO.NET, ASP.NET และอื่น ๆ

นอกจากนี้คอทเน็ตเฟรมเวิร์กยังมีลักษณะเป็นกลางทางภาษา (Language Neutral) กล่าวคือ รองรับการทำงานของภาษาใดก็ได้ เช่น C++, C#, Visual Basic และ COBOL ซึ่งมีโปรแกรม Visual Studio .NET ใช้พัฒนาคอทเน็ตเฟรมเวิร์กได้อย่างง่ายดายมากขึ้น ซึ่งทุกวันนี้จะสนับสนุนการเขียนโปรแกรมภาษา XML และ OOP



รูปที่ 2.26 โปรแกรม Visual Studio 2005

### 2.8.2 ภาษา ASP.NET

ASP.NET (Active Server Pages) เป็นเจเนอเรชันใหม่ของภาษา ASP แต่ไม่ใช้การเปลี่ยนเวอร์ชันของภาษา ASP และเป็นส่วนหลักของคอทเน็ตเฟรมเวิร์ก กล่าวคือ เป็นภาษาประเภท Server-side script ที่อยู่ในรูปแบบของเว็บแอปพลิเคชันเฟรมเวิร์กที่ใช้ในการสร้างเว็บไซต์, เว็บแอปพลิเคชัน และเว็บเซอร์วิสของบริษัทไมโครซอฟท์ ซึ่งสามารถเลือกใช้ภาษาในการพัฒนาเป็นภาษาใดก็ได้ที่คอทเน็ตเฟรมเวิร์กรองรับ



รูปที่ 2.27 สัญลักษณ์ภาษา ASP.NET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการงานจะคล้ายกับการทำงานของเว็บแอปพลิเคชัน คือ ไฟล์ .aspx ของ ASP.NET เป็น Server side script ที่ประมวลผลบน IIS (Internet Information Services) เมื่อผู้ใช้งานเรียกใช้สคริปต์ ASP.NET แล้ว สคริปต์นั้นจะถูกส่งกลับมายังเครื่องเซิร์ฟเวอร์เพื่อประมวลผล แล้วจึงส่งกลับไปแสดงผลยังเว็บเบราว์เซอร์ของผู้เรียกใช้ในรูปแบบของ HTML

ปัจจุบันที่ใช้งานกันโดยทั่วไป คือ ASP.NET 2.0 ซึ่งพัฒนาขึ้นมาจาก ASP.NET และล่าสุดมีการออก ASP.NET 3.0 ซึ่งไม่ใช่เวอร์ชันใหม่ของ ASP.NET 2.0 แต่เป็นการเพิ่มไลบรารีสนับสนุนการทำงาน เช่น Windows Presentation Foundation, Windows Communication Foundation เป็นต้น

### 2.8.3 จุดเด่นของ ASP.NET

- 1) สามารถพัฒนาได้ด้วยหลายภาษา เช่น C#, VB.NET หรือ JScript
- 2) มีความยืดหยุ่นในการเขียนโปรแกรมมากขึ้น เพราะในหนึ่งหน้าสามารถใช้ภาษามากกว่าหนึ่งภาษาในการพัฒนาได้
- 3) มีไลบรารีเสริมให้เลือกใช้มากมาย
- 4) สามารถตรวจจับเหตุการณ์ต่าง ๆ ที่เกิดขึ้นบนเว็บได้ เช่น การคลิก, การเลือก เป็นต้น
- 5) แยกส่วนของ ASP.NET ออกจาก HTML อย่างชัดเจนทำให้พัฒนาได้ง่าย
- 6) เหมาะกับการพัฒนาแอปพลิเคชันขนาดใหญ่

### 2.8.4 ภาษา ASP.NET กับเว็บเซอวิศ

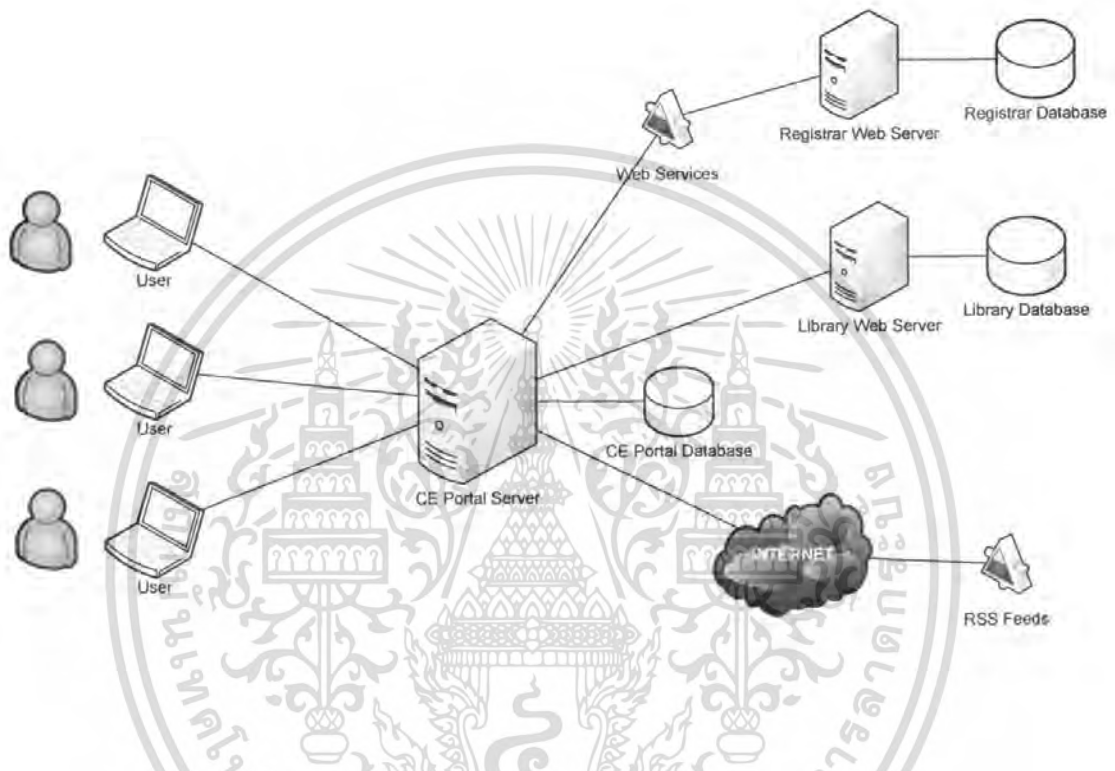
การพัฒนาเว็บเซอวิศด้วยภาษา ASP.NET นั้นสามารถสร้างได้ง่ายมาก เนื่องจากมีโปรแกรม Visual Studio .NET คอยอำนวยความสะดวกในการสร้างเซอวิศทั้งฝั่งรับฝั่งส่ง

ในปฏิญญาพันธบัตรฉบับนี้ได้มีการกล่าวถึงการสร้างและทดลองเว็บเซอวิศด้วยภาษา ASP.NET ซึ่งรายละเอียดได้กล่าวไว้ในภาคผนวก ข

## บทที่ 3

### การออกแบบระบบ CE Web Portal

#### 3.1 โครงสร้างของระบบ



รูปที่ 3.1 โมเดลของระบบ CE Web Portal

ระบบเว็บพอร์ทัลสำหรับภาควิชาวิศวกรรมคอมพิวเตอร์ (CE Web Portal) เป็นระบบที่รวบรวมเนื้อหา และข้อมูลสารสนเทศที่จำเป็นในการเรียนหรือการทำงานของนักศึกษา, อาจารย์ และเจ้าหน้าที่ รวมทั้งเป็นสื่อกลางในการสื่อสารระหว่างนักศึกษา, อาจารย์, เจ้าหน้าที่ ตลอดจนภาควิชาฯ ด้วยการใช้หลักการของเว็บเซอร์วิส

การออกแบบระบบพอร์ทัลนั้น ออกแบบโดยสรุปตามผลจากการสำรวจความต้องการของผู้ใช้งาน (User Requirements) ซึ่งก็คือบุคลากรในภาควิชาคอมพิวเตอร์จำนวนทั้งสิ้น 47 คน แบ่งเป็นนักศึกษา 40 คน และอาจารย์ 7 คน โดยใช้แบบสอบถามตามภาคผนวก ก.

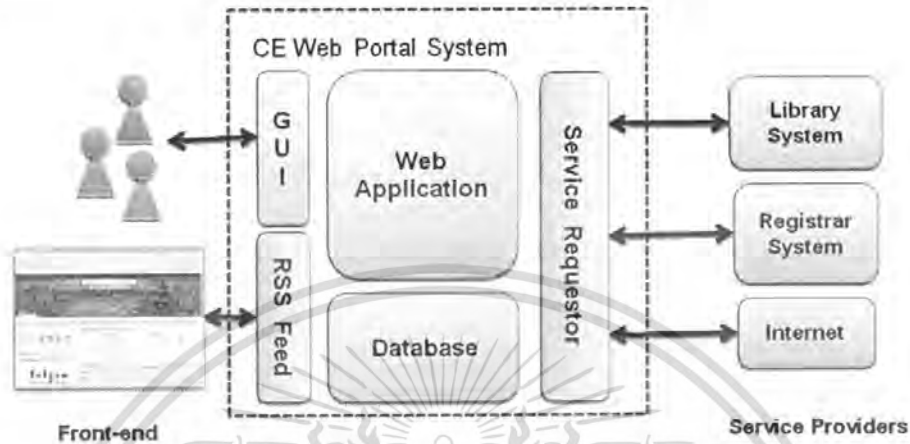
การทำงานของระบบเริ่มจากยูสเซอร์ใช้งานผ่านเว็บแอปพลิเคชันกลางของระบบ เมื่อยูสเซอร์ทำการร้องขอข้อมูลและ/หรือบริการใด ๆ ระบบจะทำการตรวจสอบว่าเป็นข้อมูลและ/หรือ

บริการที่มีอยู่ในระบบพอร์ทัลและฐานข้อมูลหรือไม่ หากไม่พบหรือพบข้อมูลที่ไม่มีอัปเดต ระบบ

เอกสารณเป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาดูเท่านั้น เมื่ออนุญาตให้เข้าใช้จะกระเป็ช่นด้านกาารควา

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะทำการร้องขอไปยังระบบอื่นที่มีข้อมูลและ/หรือบริการที่ผู้สเซอร์ต้องการใช้งาน เพื่อให้ระบบนั้น ๆ ส่งข้อมูลที่ผู้สเซอร์ต้องการกลับมา แล้วนำไปอัปเดตลงฐานข้อมูลและแสดงผลให้แก่ผู้สเซอร์ต่อไป



รูปที่ 3.2 โมเดลของระบบ CE Web Portal

**โครงสร้างของระบบ** ประกอบด้วยส่วนต่าง ๆ ดังนี้

**3.1.1 ระบบซีอีเว็บพอร์ทัล** เป็นระบบกลางในการเชื่อมต่อระบบอื่น ๆ ซึ่งประกอบด้วย

- 1) Web Application แบ่งการพัฒนาเป็นสองส่วนหลัก ๆ คือ
  - ส่วนของ Front-end อันได้แก่ ส่วนของ GUI (Graphic User Interface) และส่วน RSS Feed ที่ฟีดคอนเทนต์ของพอร์ทัลไปยังเว็บอื่น ๆ
  - ส่วนของ Back-end อันได้แก่ ส่วนที่ทำหน้าที่ติดต่อกับฐานข้อมูล และส่วนกลไกการทำงานอื่น ๆ
- 2) Service Requestor เป็นส่วนของการเรียกใช้งานเว็บเซอร์วิสและคอนเทนต์

**3.1.2 คอนเทนต์ และระบบอื่น ๆ ที่อยู่บนเครือข่าย** ซึ่งประกอบด้วย

- 1) ระบบทะเบียนนักศึกษา ที่มีเว็บเซอร์วิสเปิดให้บริการอยู่บนเครื่องเซิร์ฟเวอร์
- 2) ระบบห้องสมุด ที่เปิดให้บริการ WebOPAC
- 3) คอนเทนต์บนอินเทอร์เน็ต เช่น RSS Feeds เป็นต้น

## 3.2 รายละเอียดของระบบ

**3.2.1 รายละเอียดของเว็บพอร์ทัล** ประกอบด้วยโมดูลหลักทั้งสิ้น 8 โมดูล ดังนี้

- 1) โมดูลประกาศข่าว (News) คือ โมดูลที่แสดงข่าวประกาศจากอาจารย์ เจ้าหน้าที่ และภาควิชาฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) โมดูลรับ/ส่งข้อความ (Internal Message) คือ โมดูลที่ทำการรับและส่งข้อความหากันภายในระบบ โดยผู้ส่งสามารถเลือกส่งเป็นรายบุคคล, รายวิชา, ชั้นปี หรือ ทั้งภาควิชาฯ ได้
- 3) โมดูลปฏิทิน (Calendar) คือ โมดูลที่เกี่ยวข้องกับปฏิทินต่าง ๆ ได้แก่
- ปฏิทินจัดการส่วนบุคคล
  - การแจ้งเตือน
  - ตารางนัดหมาย
- 4) โมดูลเอกสารรายวิชา (Sheets) คือ โมดูลที่แสดงเอกสารรายวิชา โดยแสดงเอกสารทั้งหมดหรือตามหมวดหมู่ได้ รวมทั้งอัปโหลด และดาวน์โหลดได้
- 5) โมดูลสำนักทะเบียน (Registrar) คือ โมดูลที่เกี่ยวข้องกับงานทะเบียน ดังนี้
- แสดงตารางเรียน
  - แสดงตารางสอบ
  - แสดงรายชื่อนักศึกษาในรายวิชาหรือชั้นปี
- 6) โมดูลห้องสมุด (Library) คือ โมดูลที่สามารถทำการค้นหาหนังสือ และทำการเข้าสู่ระบบอัตโนมัติได้
- 7) โมดูลคอนเทนต์ (Contents) เช่น RSS Feeds โดยที่ผู้ใช้งานแต่ละคนเลือกคอนเทนต์ที่ต้องการได้ และสามารถเพิ่มคอนเทนต์ส่วนบุคคล และเพิ่มของส่วนรวมได้
- 8) โมดูลปรับแต่งอินเทอร์เฟซ (GUI) คือ สามารถปรับแต่ง look and feel ได้

### 3.2.2 การแบ่งระดับสิทธิการใช้งานในระบบ

- 1) ระดับ Anonymous ได้แก่ บุคคลทั่วไป ซึ่งไม่ต้องทำการล็อกอิน
- 2) ระดับ User ได้แก่ นักศึกษาทุกระดับปริญญาตรี
- 3) ระดับ Super user ได้แก่ อาจารย์และเจ้าหน้าที่
- 4) ระดับ Administrator ได้แก่ ผู้ดูแลระบบ

ตารางที่ 3.1 แสดงสิทธิการใช้งานในแต่ละ โมดูล

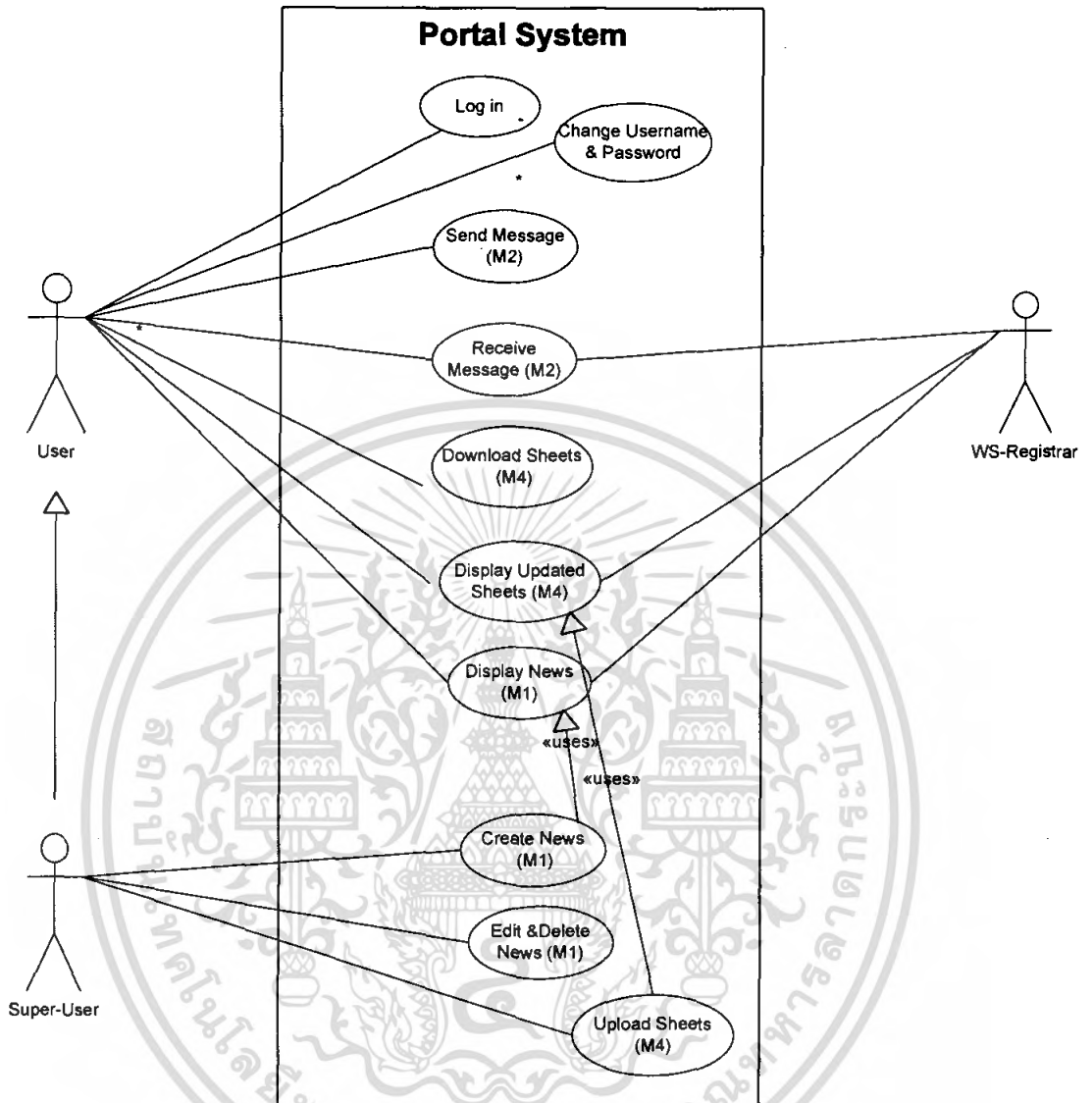
โมดูล	Anony mous	User	Super User	หมายเหตุ
<b>1.โมดูลประกาศข่าว (News)</b>				
แสดงประกาศข่าว	o	o	o	
สร้างประกาศใหม่	x	x	o	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมดูล	Anony mous	User	Super User	หมายเหตุ
<b>2.โมดูลรับ/ส่งข้อความ (Internal message)</b>				
รับข้อความ	x	o	o	
ตั้งค่าสัญญาณเตือน	x	o	o	เมื่อได้รับข้อความใหม่
ส่งข้อความ (Announcement / PM)	x	o	o	เลือกส่งเป็นรายบุคคล หรือรายกลุ่ม
<b>3.โมดูลปฏิทิน (calendar)</b>				
แสดงและสร้างปฏิทินจัดการส่วนบุคคล	x	o	o	
แสดงกำหนดส่งงานทั้งหมด	x	o	o	
แสดงและสร้างตารางนัดหมาย	x	o	o	
ตั้งค่าสัญญาณเตือน	x	o	o	เช่น ใกล้เคียงเวลาส่งงาน
<b>4) โมดูลเอกสารรายวิชา (Sheets)</b>				
แสดงเอกสารล่าสุด	x	o	o	
อัปโหลดเอกสาร	x	x	o	
ดาวน์โหลดเอกสาร	x	o	o	
<b>5) โมดูลสำนักทะเบียน (Registrar)</b>				
แสดงรายชื่อนักศึกษา	x	x	o	เป็นรายวิชา, ชั้นปี หรือ กลุ่มเรียน
แสดงตารางเรียน/สอบ	x	o	o	
<b>6) โมดูลห้องสมุด (Library)</b>				
ค้นหาหนังสือในห้องสมุด	o	o	o	
ล็อกอินเข้าระบบ WebOPAC อัตโนมัติ	x	o	o	
<b>7) โมดูลคอนเทนต์ (Contents) เช่น ข่าว</b>				
แสดงคอนเทนต์	x	o	o	
เพิ่มคอนเทนต์ ส่วนรวม	x	x	o	
เพิ่มคอนเทนต์ ส่วนตัว	x	o	o	
<b>8) โมดูลปรับแต่งอินเทอร์เฟซ (GUI)</b>				
ปรับแต่งอินเทอร์เฟซ	x	o	o	เปลี่ยน logo
กำหนดค่าการแสดงผลหน้าพอร์ทัล	x	o	o	เลือก และย้ายตำแหน่ง โมดูล

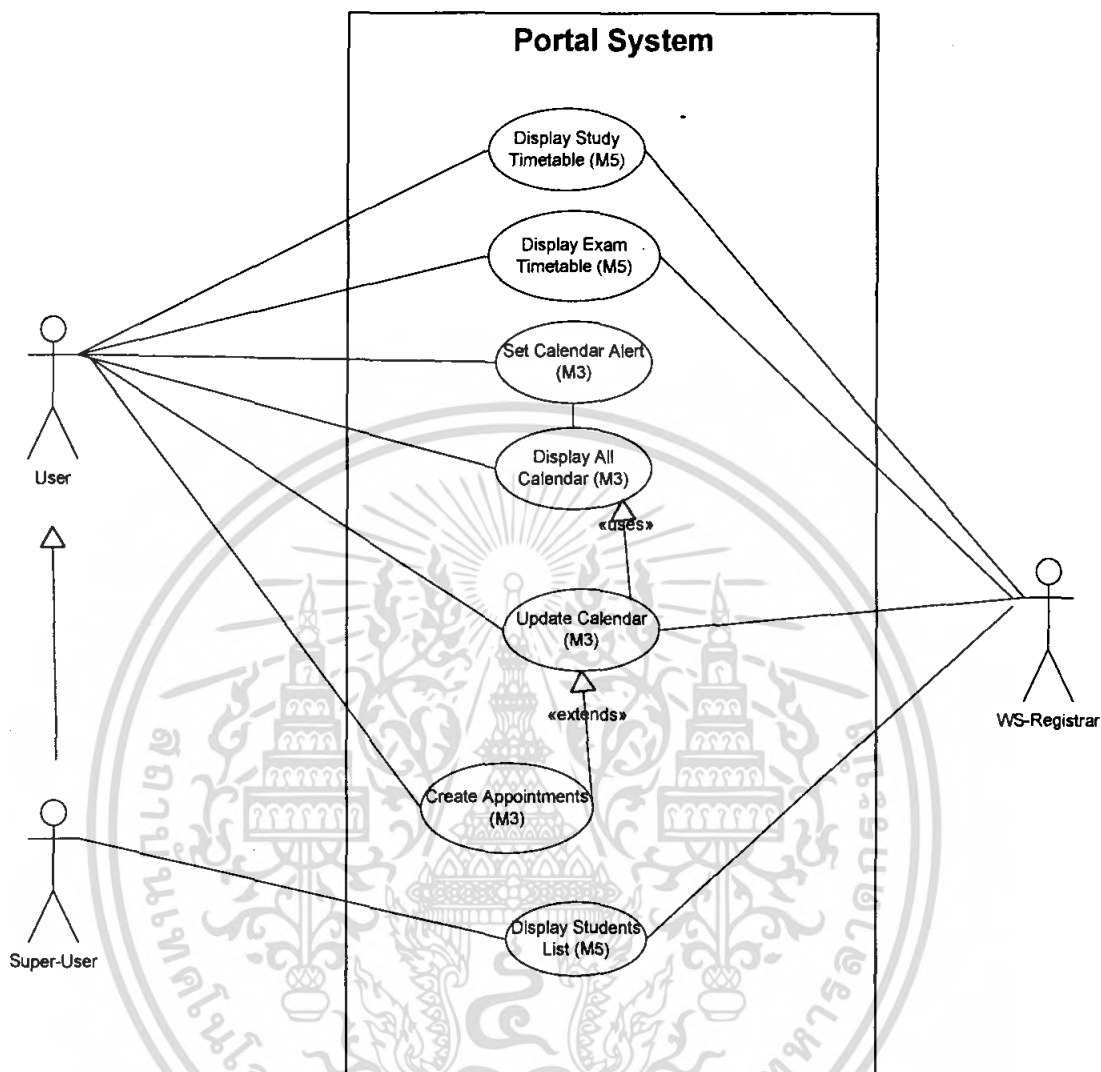
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิได้อนุญาตให้เผยแพร่หรือใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 Use Case Diagram



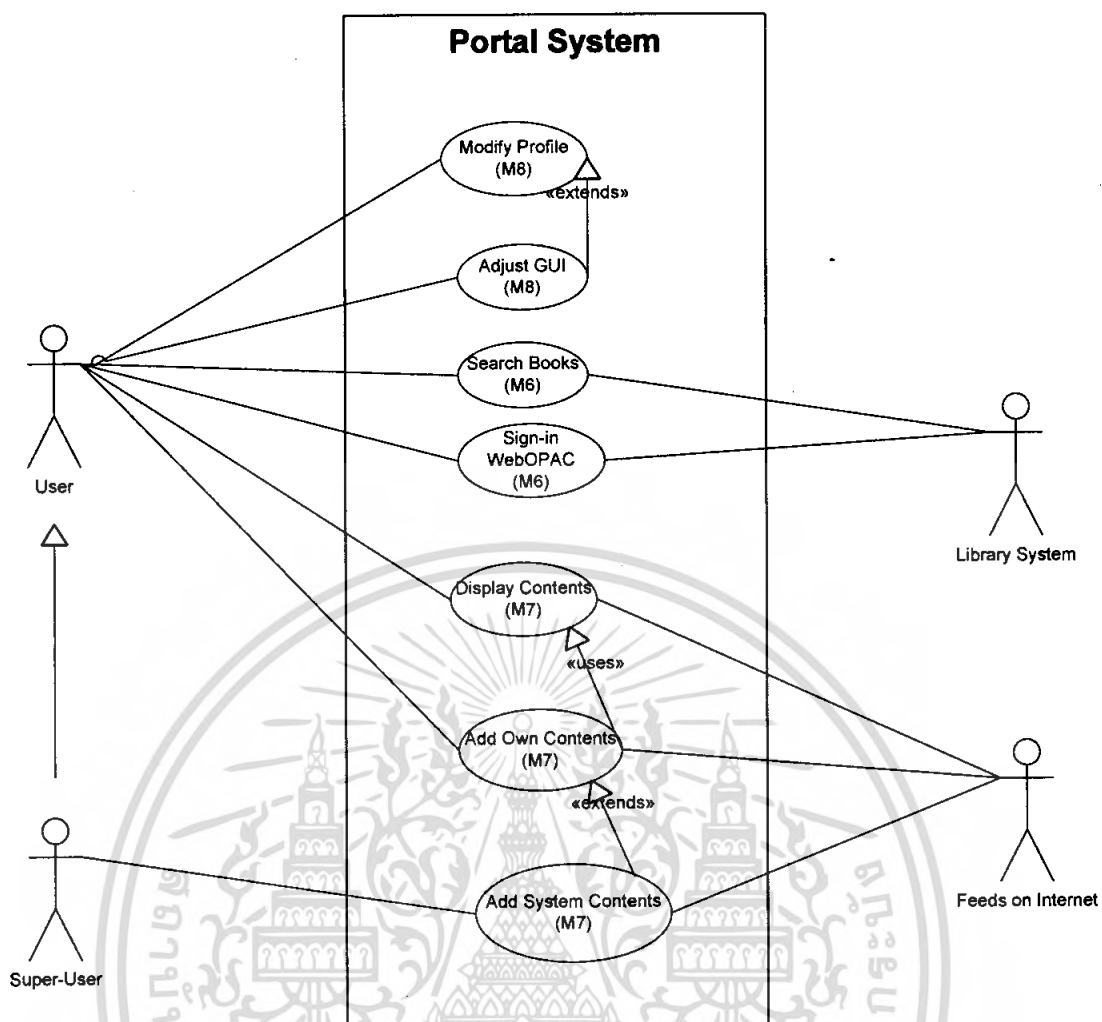
รูปที่ 3.3 Use Case Diagram ของระบบ: โมดูล 1, 2, 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 Use Case Diagram ของระบบ: โมดูล 3, 5

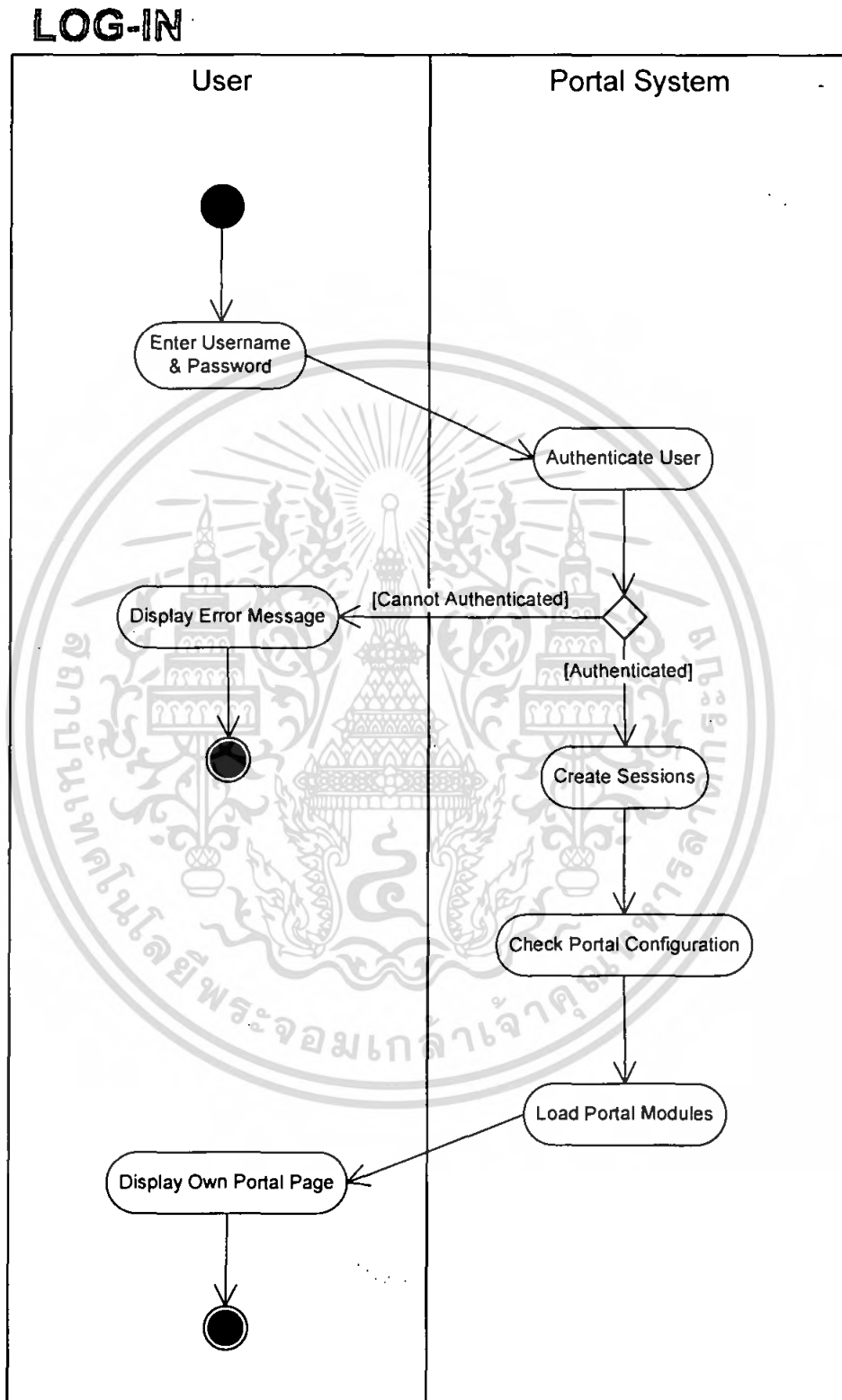
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 Use Case Diagram ของระบบ: โมดูล 6, 7, 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

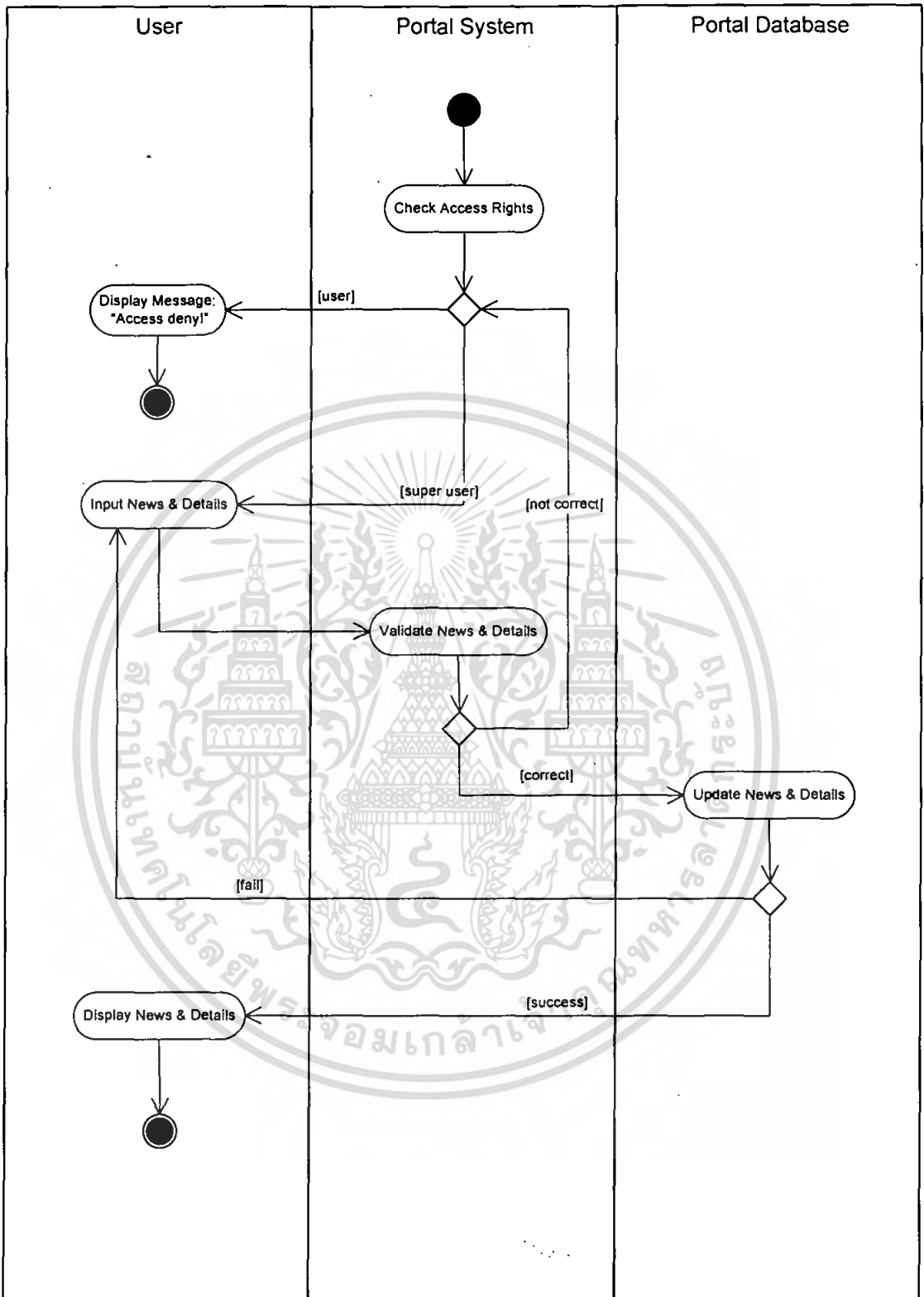
### 3.4 Activity Diagram



รูปที่ 3.6 Activity Diagram ของการล็อกอิน (log-in)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

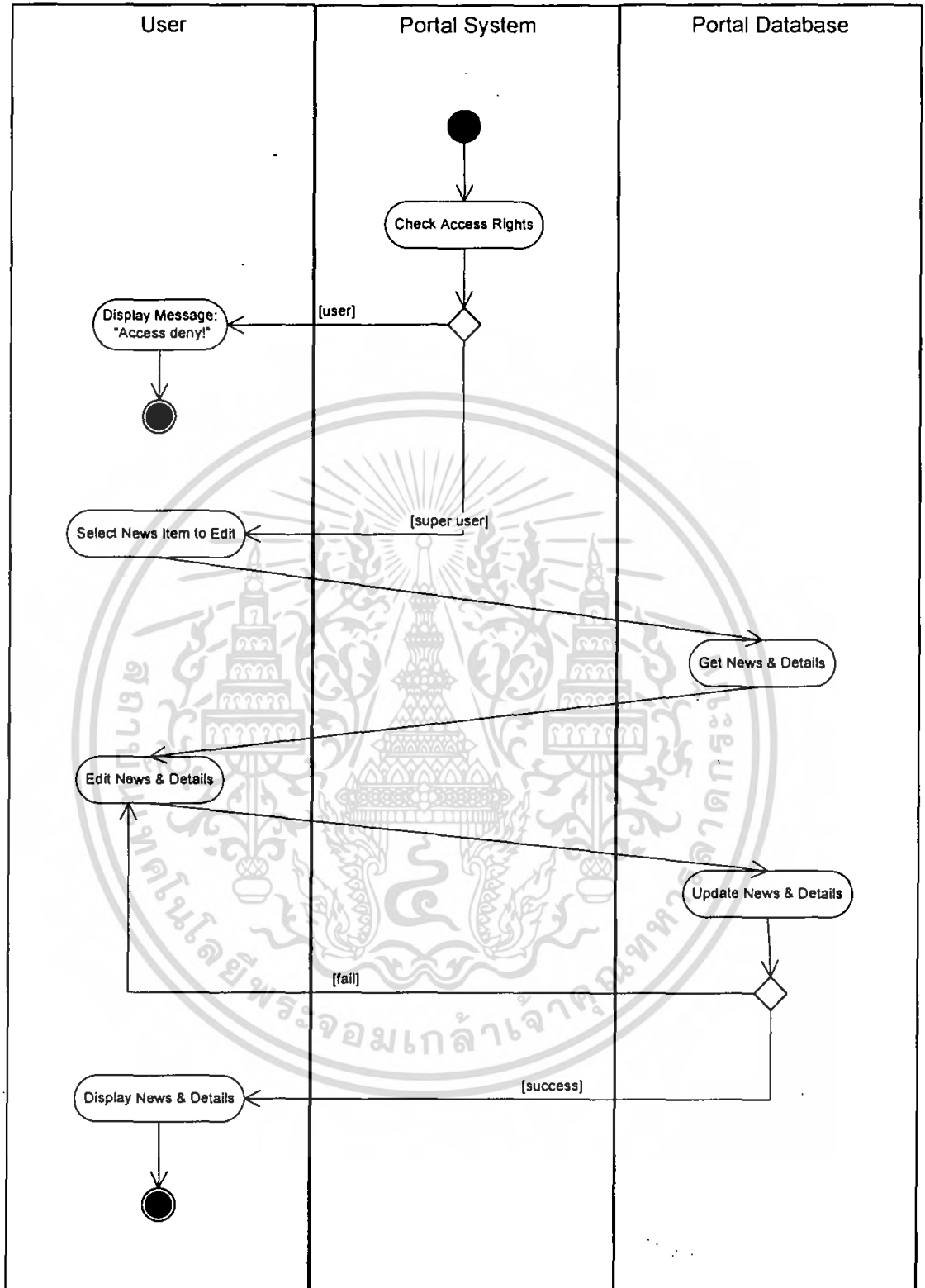
# M1:CREATE NEWS



รูปที่ 3.7 Activity Diagram ของการสร้างประกาศข่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

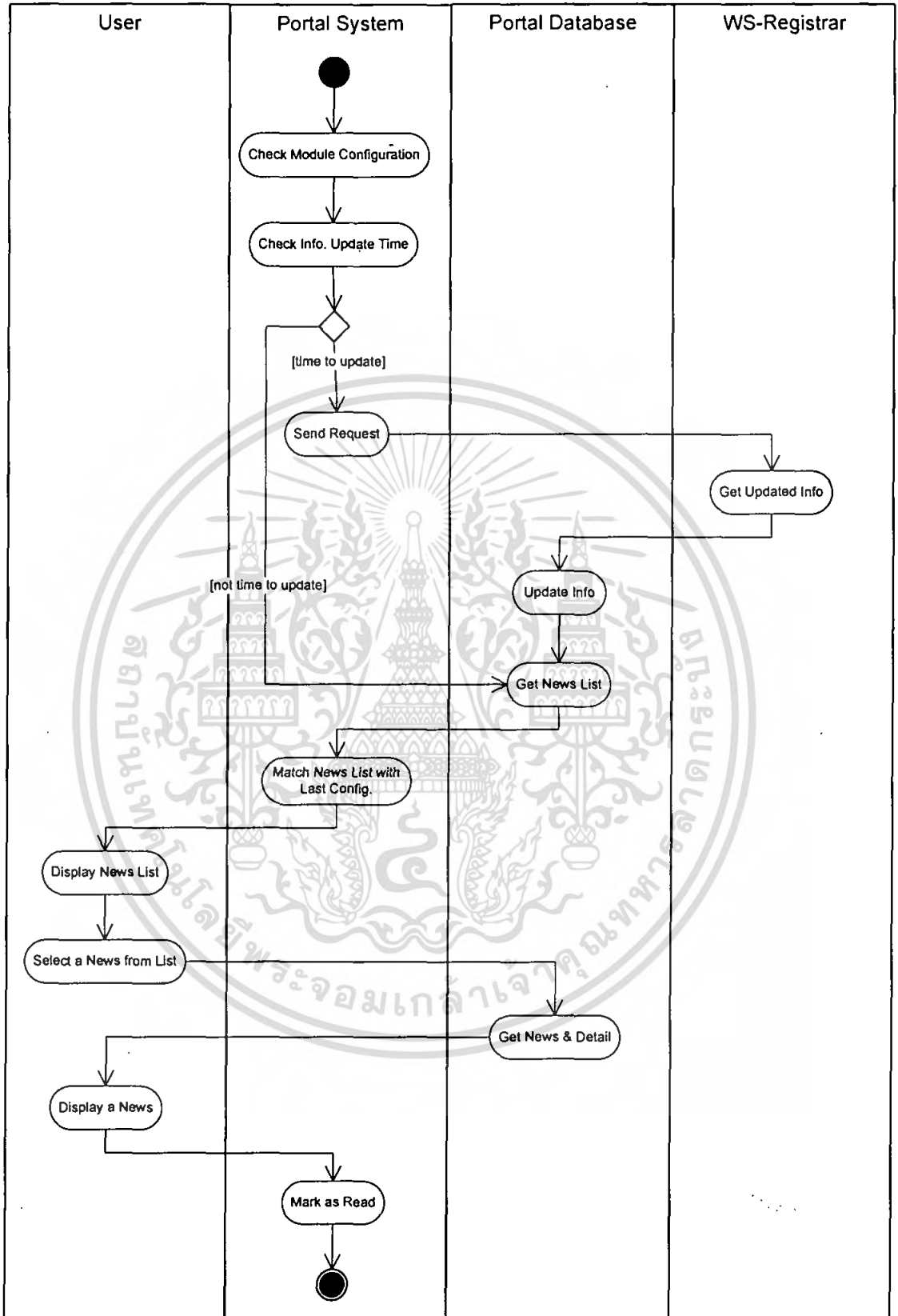
## M1:EDIT NEWS



รูปที่ 3.8 Activity Diagram ของการแก้ไขประกาศข่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

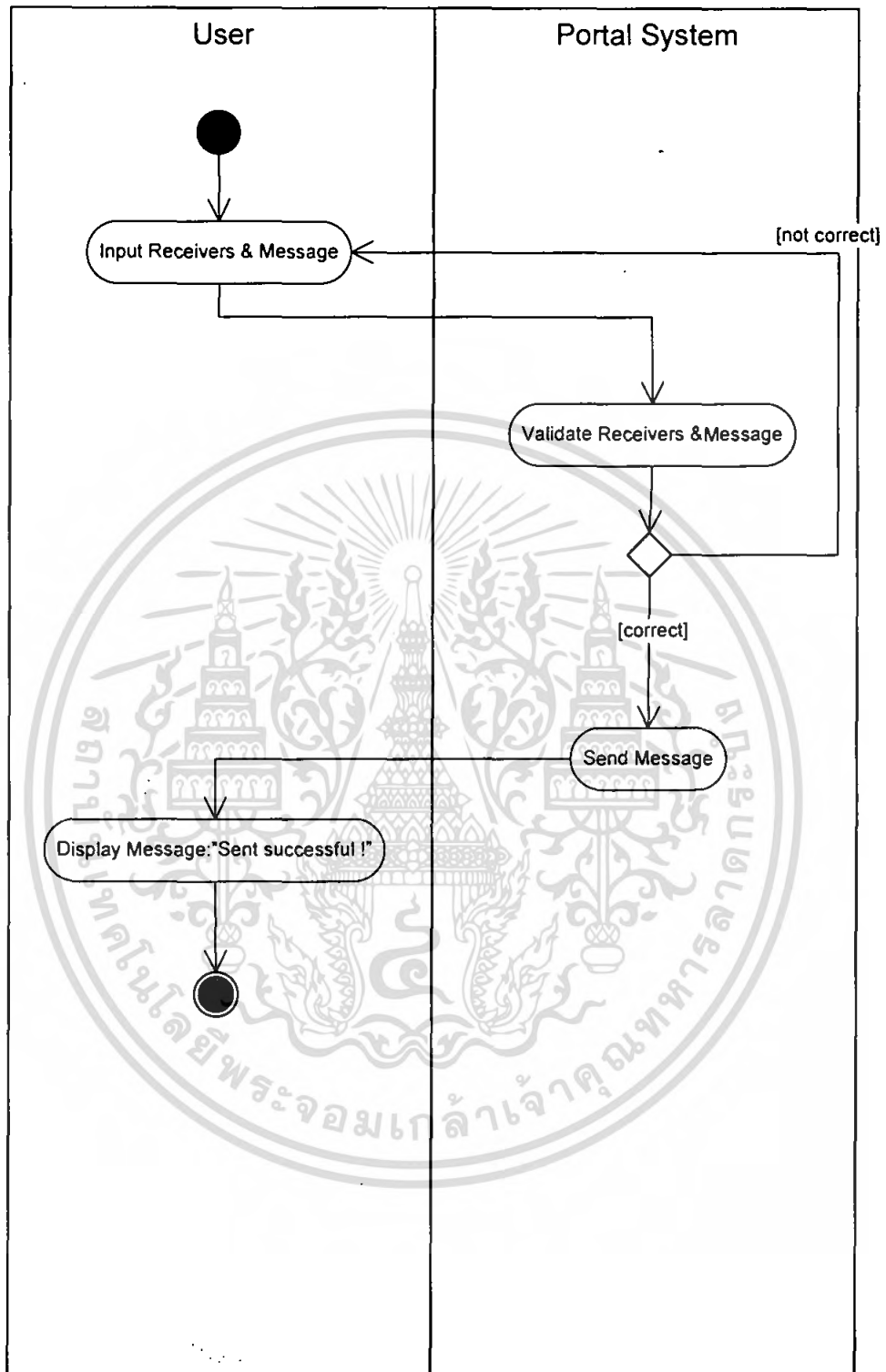
## M1: DISPLAY NEWS



รูปที่ 3.9 Activity Diagram ของการแสดงผลข่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

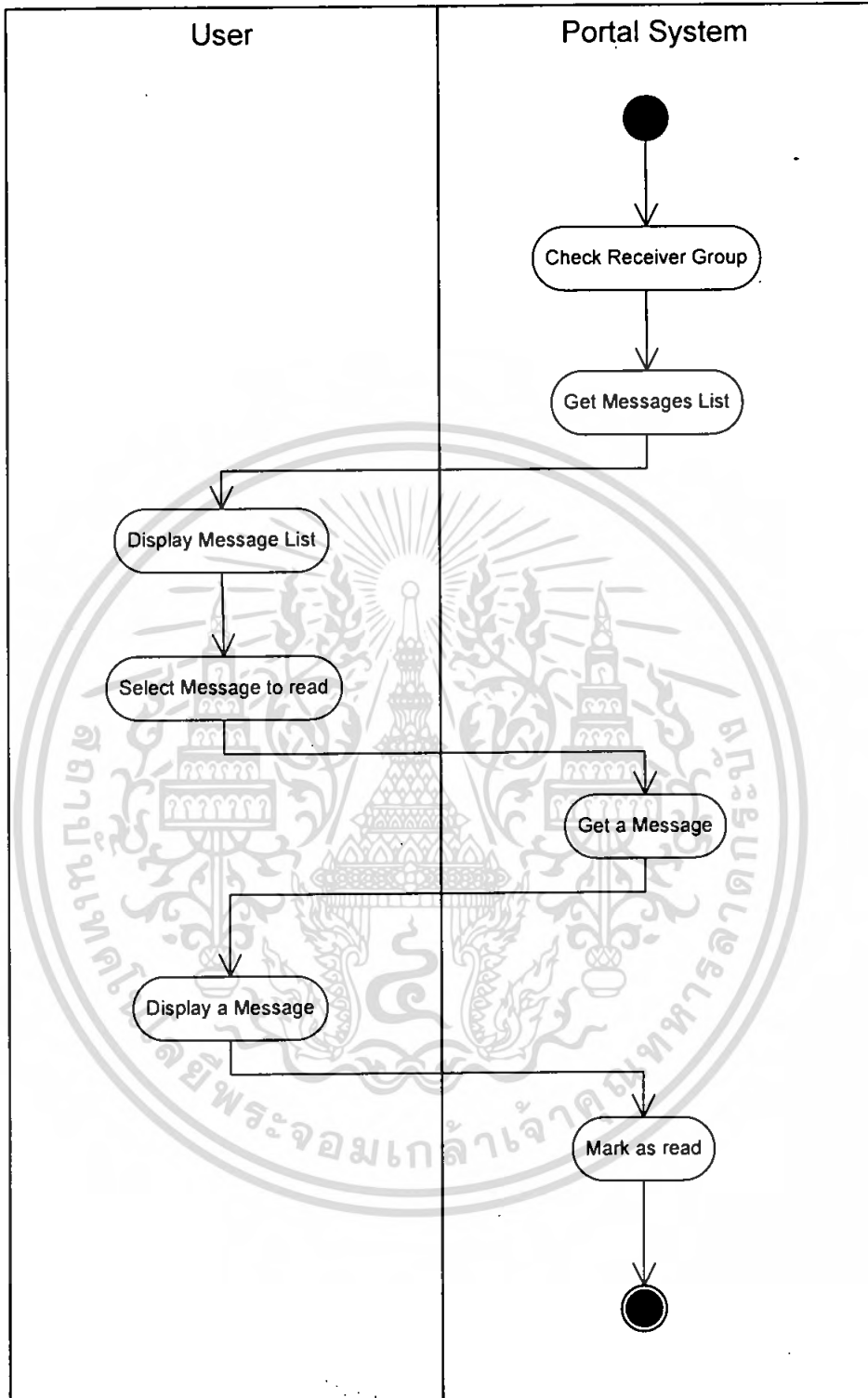
## M2:SEND MESSAGE



รูปที่ 3.10 Activity Diagram ของการส่งข้อความ (private/group)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

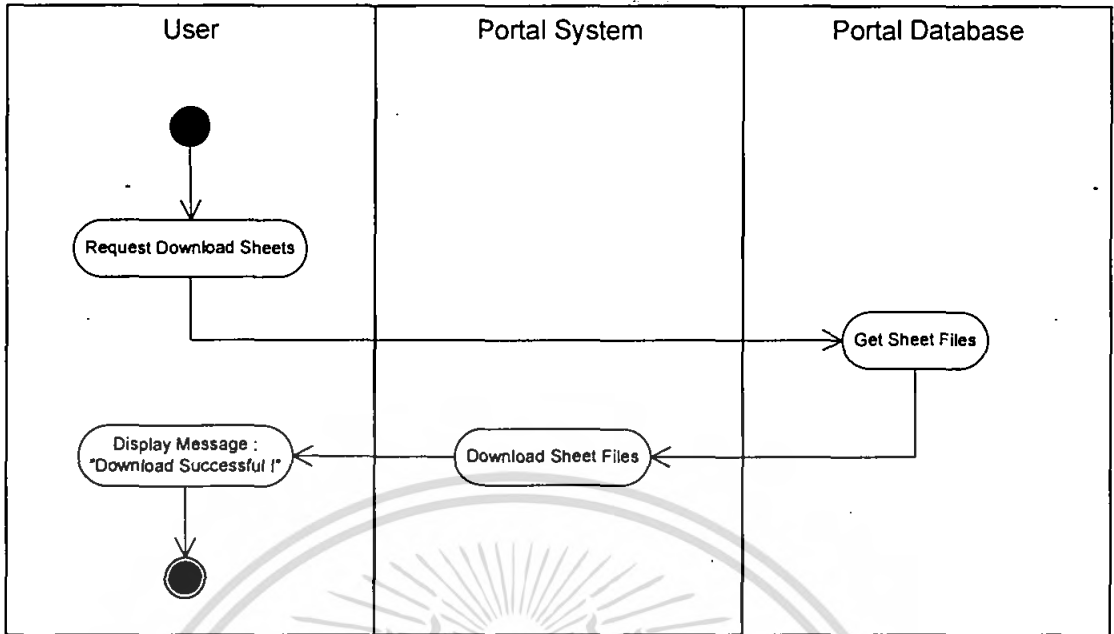
## M2:RECEIVE MESSAGE



รูปที่ 3.11 Activity Diagram ของการรับข้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

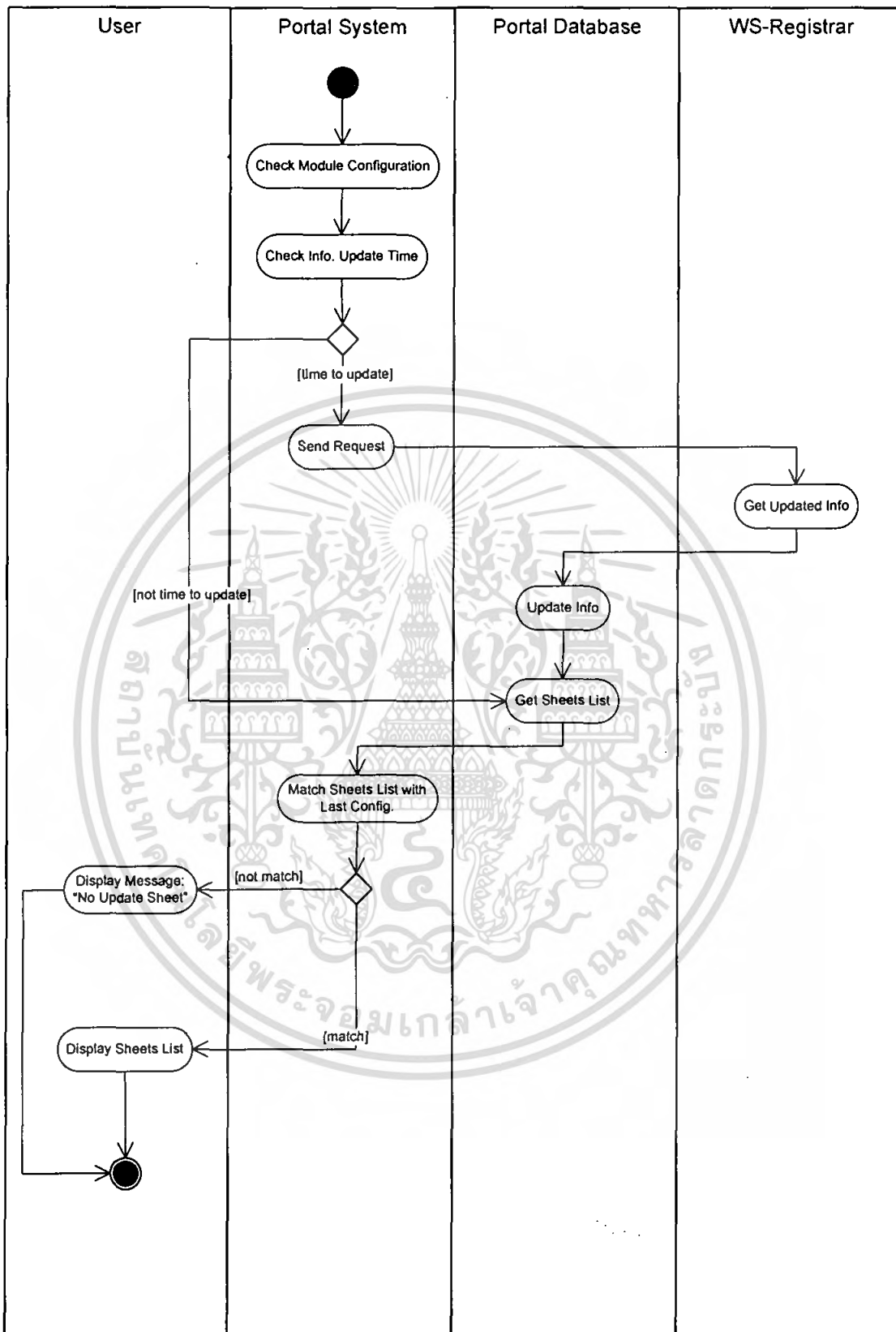
## M4: DOWNLOAD SHEETS



รูปที่ 3.12 Activity Diagram ของการดาวน์โหลดเอกสารการเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

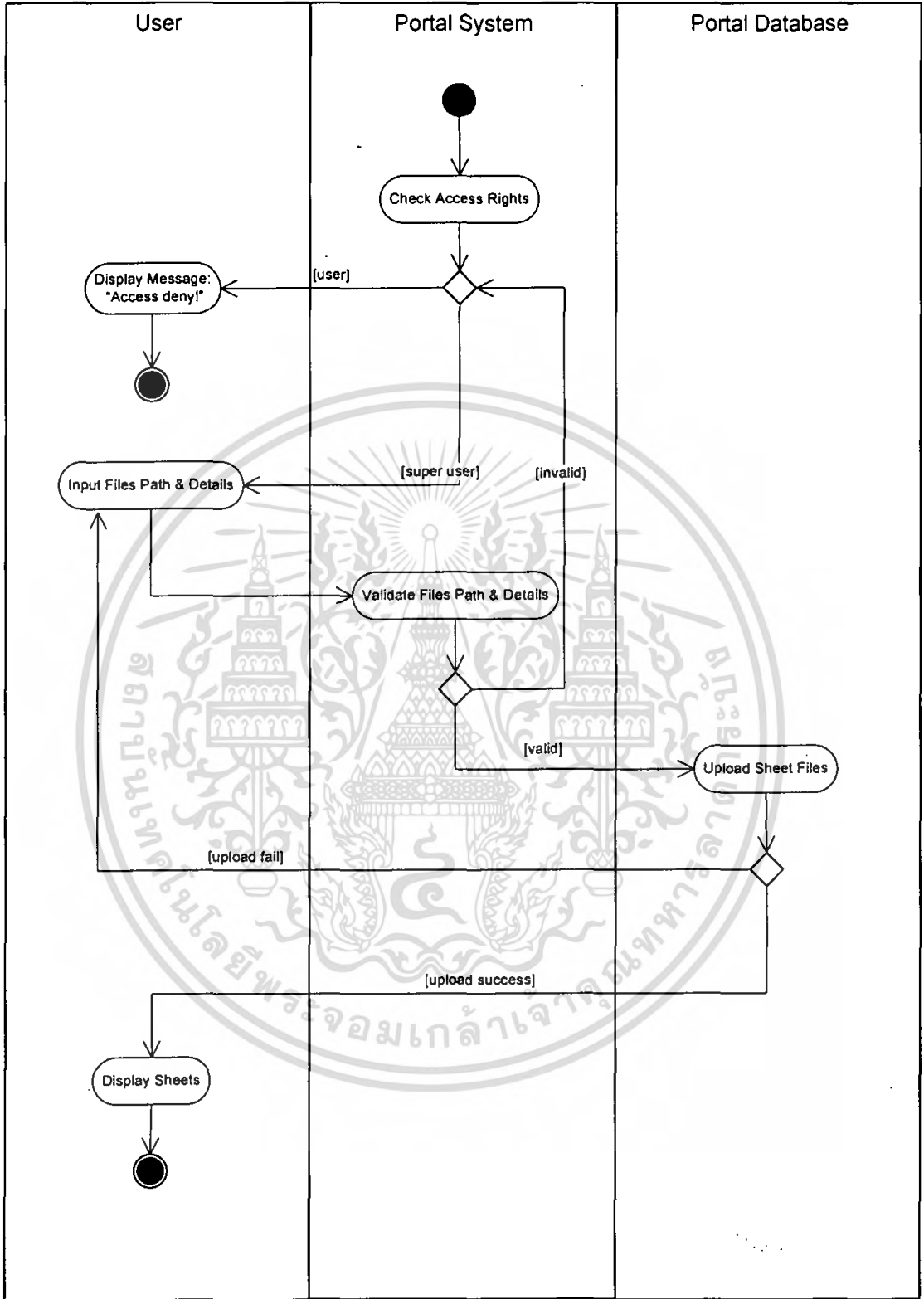
### M4: DISPLAY SHEETS



รูปที่ 3.13 Activity Diagram ของการแสดงผลเอกสารการเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

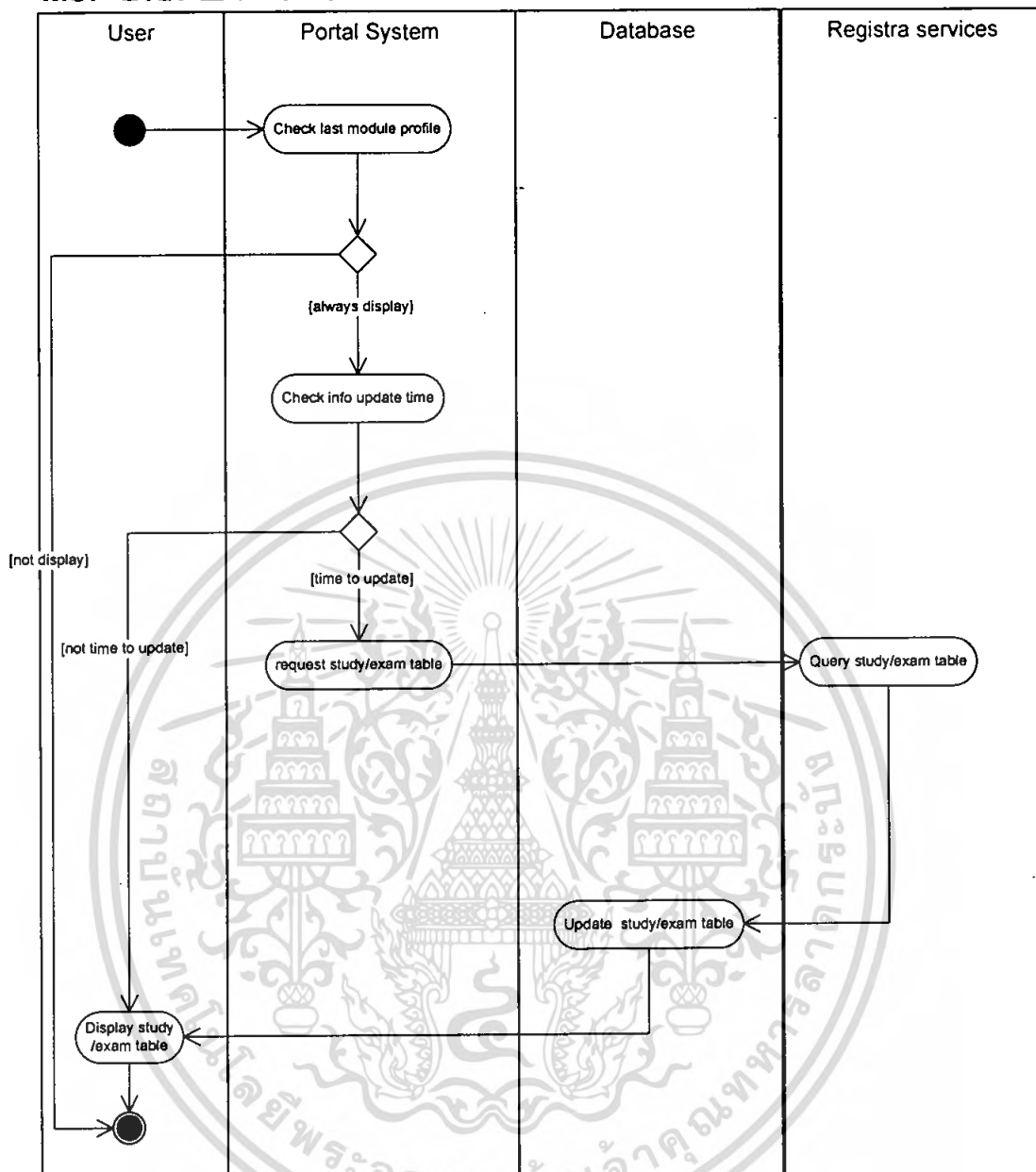
### M4: UPLOAD SHEETS



รูปที่ 3.14 Activity Diagram ของการอัปโหลดเอกสารเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

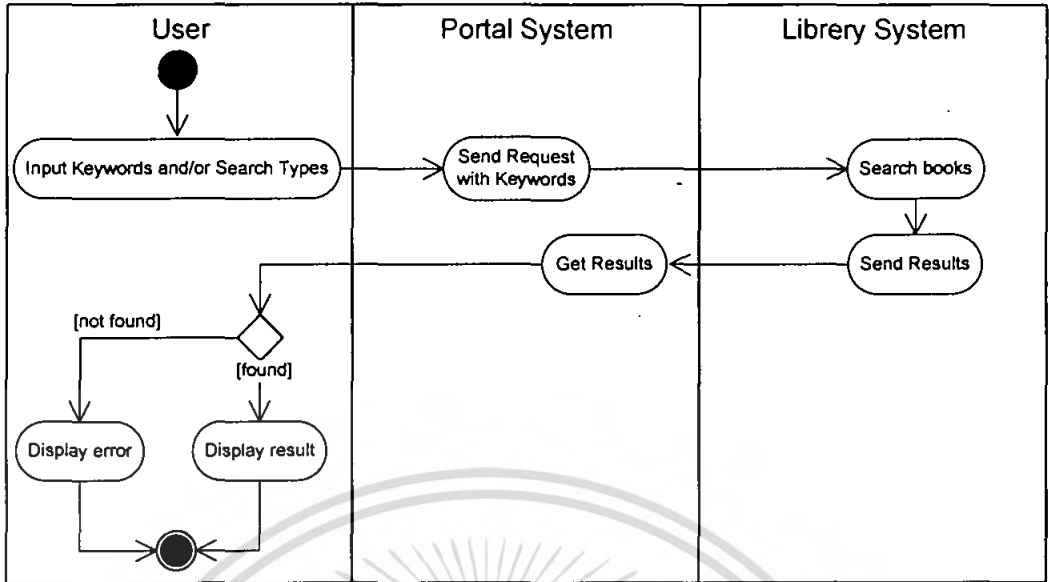
### M5: DISPLAY STUDY/EXAM TABLE



รูปที่ 3.15 Activity Diagram ของการแสดงตารางเรียน และตารางสอบ

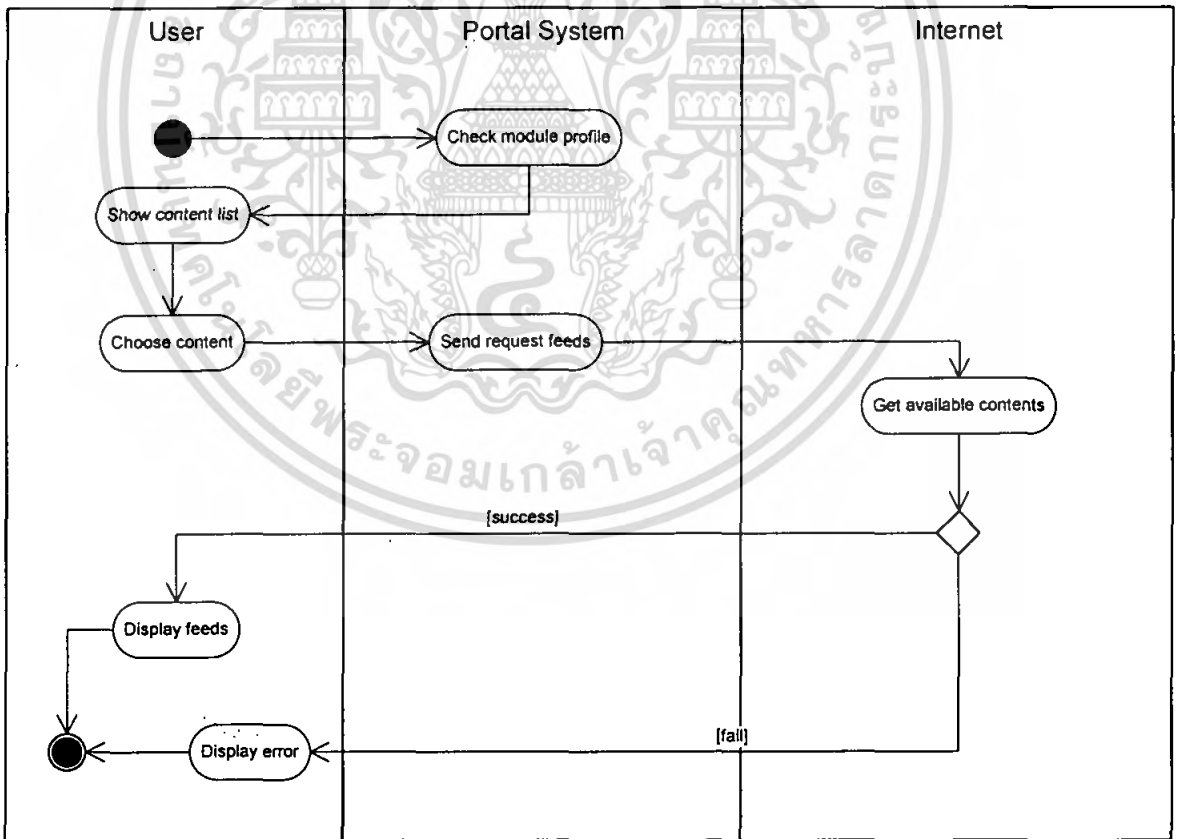
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### M6: SEARCH BOOKS



รูปที่ 3.16 Activity Diagram ของการค้นหาหนังสือห้องสมุด

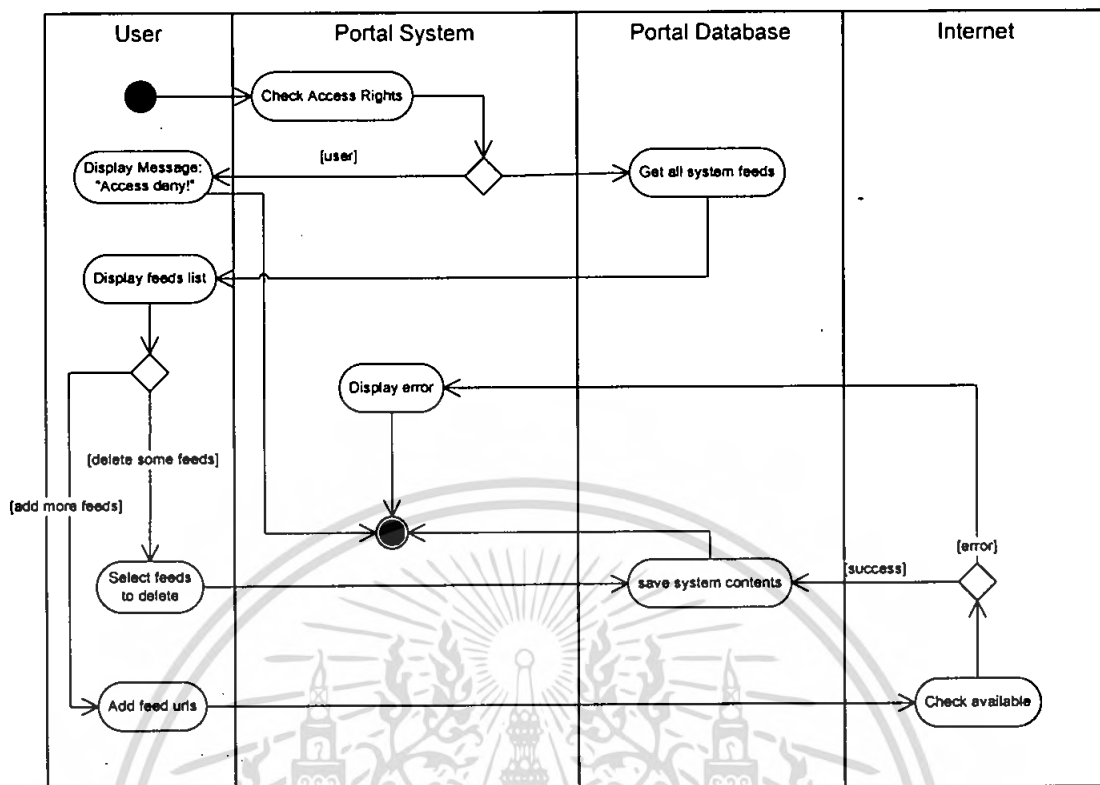
### M7: DISPLAY CONTENTS



รูปที่ 3.17 Activity Diagram ของการแสดงผลคอนเทนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

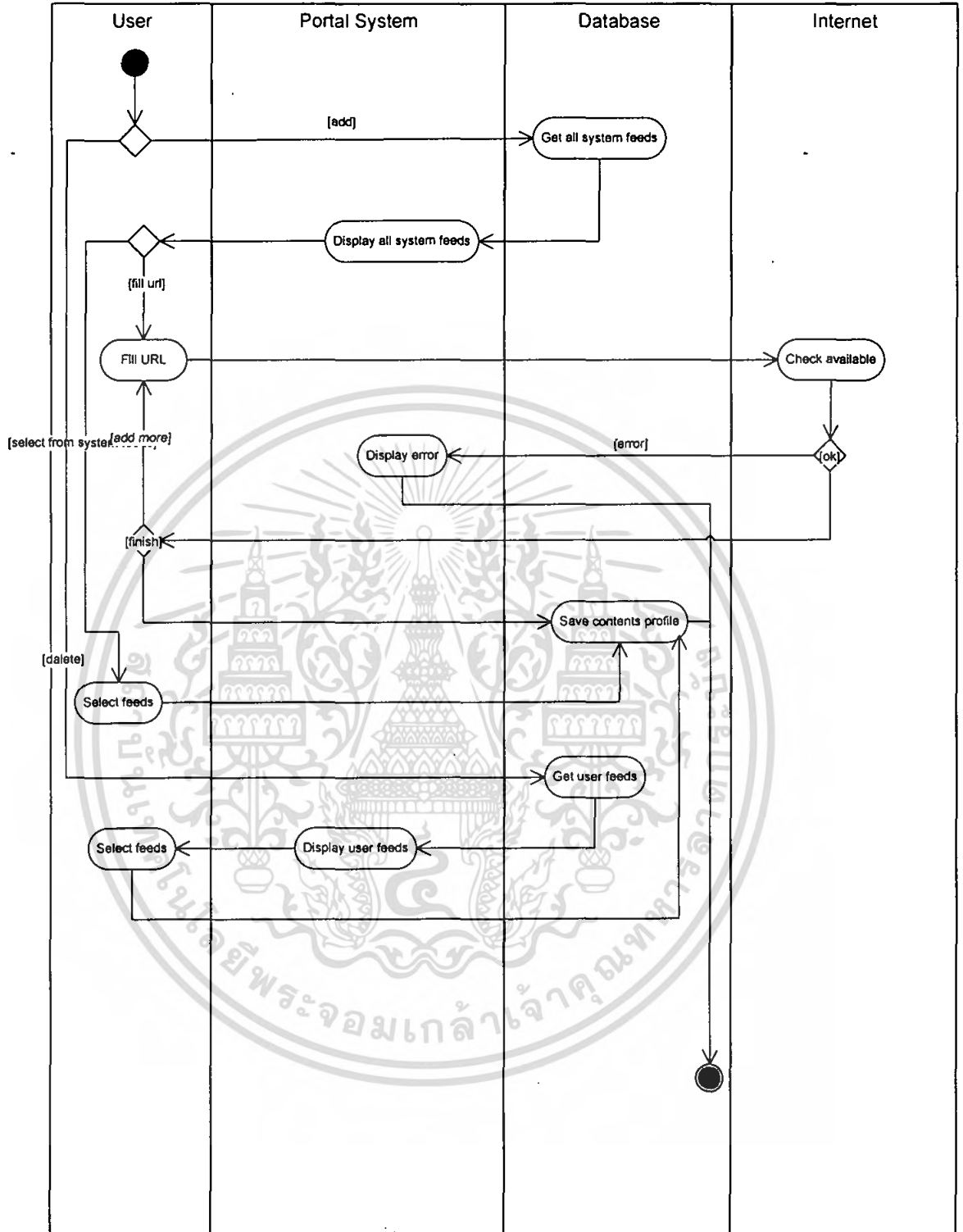
### M8: ADD SYSTEM CONTENTS



รูปที่ 3.18 Activity Diagram ของการเพิ่มคอนเทนต์ให้ทั้งระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### M8: ADD OWN CONTENTS

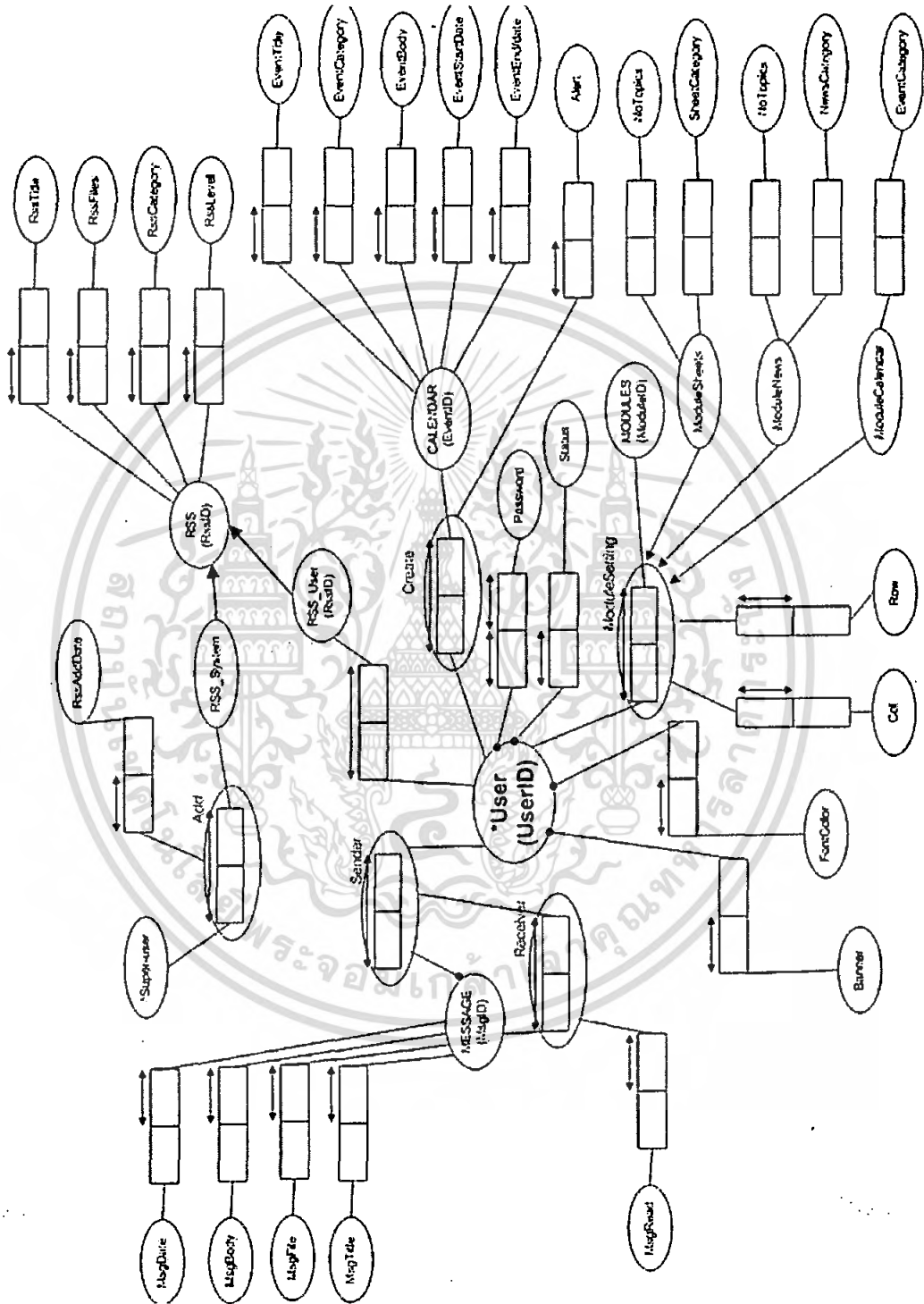


รูปที่ 3.19 Activity Diagram ของการเพิ่มคอนเทนต์ส่วนตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

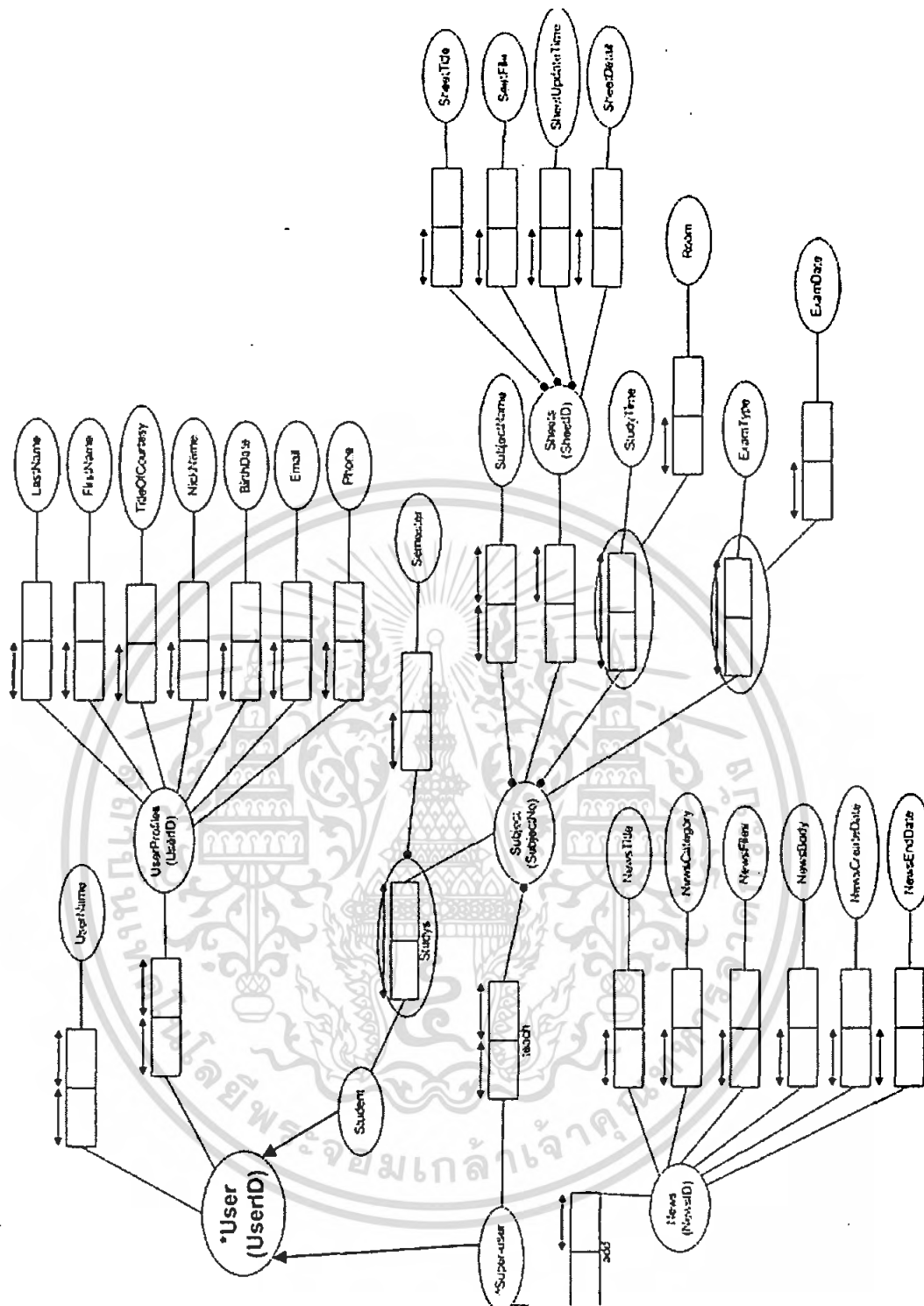
### 3.5 การออกแบบฐานข้อมูล (Database Design)

#### 3.5.1 ออกแบบฐานข้อมูล dbPortal ด้วย ORM (Object Role Model)



รูปที่ 3.20 โมเดลของฐานข้อมูล dbPortal (1)

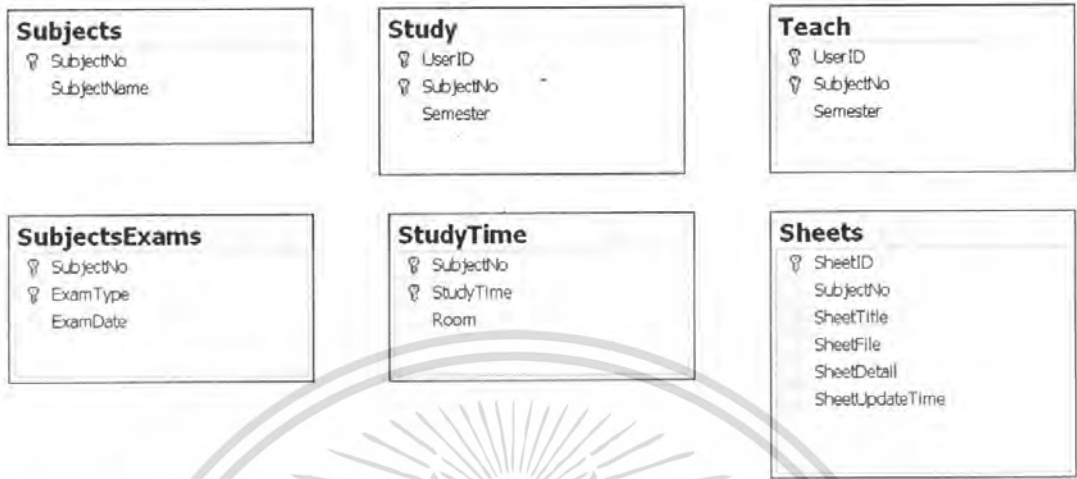
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



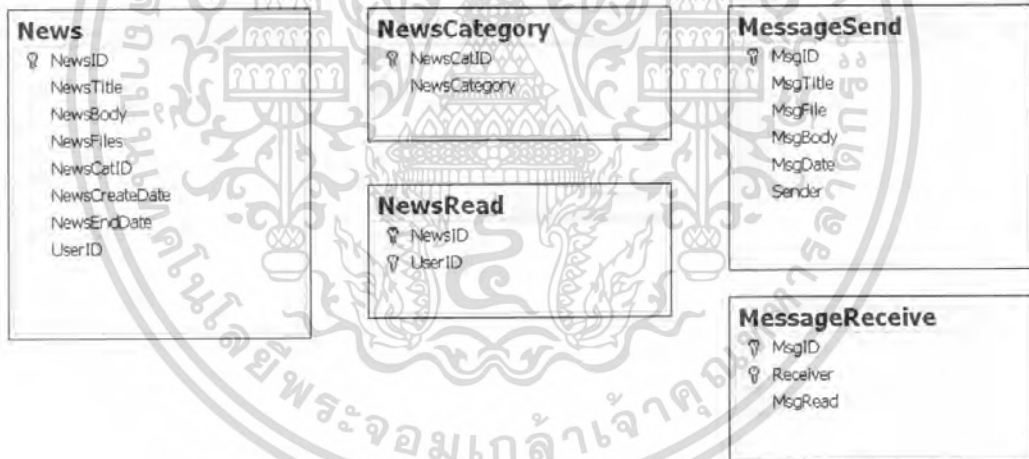
รูปที่ 3.21 โมเดลของฐานข้อมูล dbPortal (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.2 โครงตาราง (Schema) ในฐานข้อมูล dbPortal



รูปที่ 3.22 โครงตารางของโมดูลสำนักทะเบียนฯ โมดูลเอกสารการเรียน และข้อมูลรายวิชา

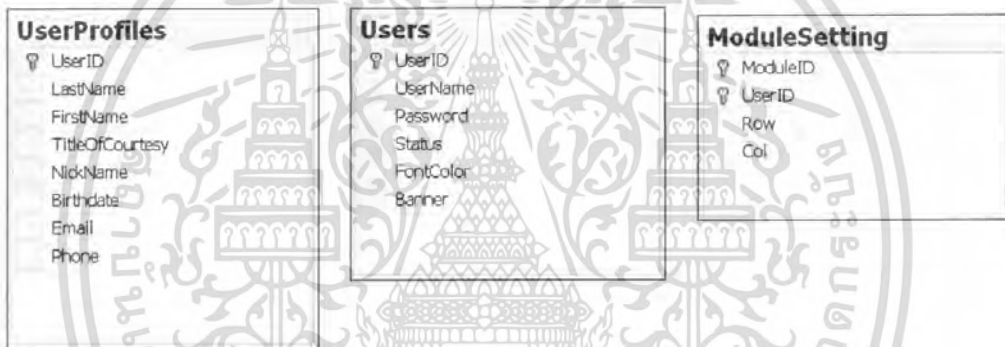


รูปที่ 3.23 โครงตารางเกี่ยวกับ โมดูลข่าว และ โมดูลข้อความ

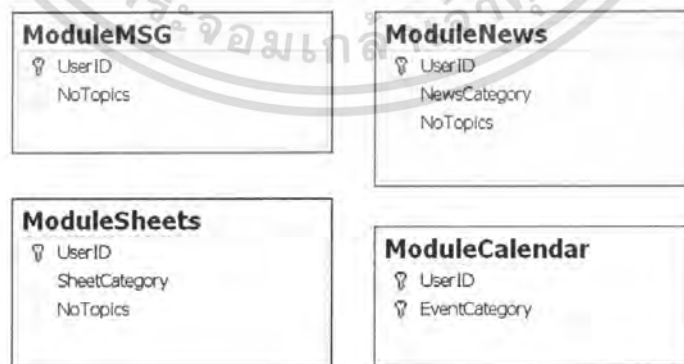
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.24 โครงตารางเกี่ยวกับ โมดูลคอนเทนท์ และ โมดูลปฏิทิน



รูปที่ 3.25 โครงตารางเกี่ยวกับ โมดูล GUI และรายละเอียดของผู้ใช้งาน



รูปที่ 3.26 โครงตารางเกี่ยวกับ โมดูล GUI ที่ใช้ในบอกรการตั้งค่าส่วนบุคคล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.6 การออกแบบเซอร์วิส

#### 3.6.1 เซอร์วิสโทรวาเคอร์ที่เปิดให้บริการที่สำนักทะเบียนฯ

เซอร์วิสที่ออกแบบเพื่อเปิดให้เรียกใช้งานผ่านอินเทอร์เน็ต ซึ่งอยู่บนเครื่องเซิร์ฟเวอร์ของสำนักทะเบียนฯ มีเซอร์วิส ดังต่อไปนี้ -

- 1) เซอร์วิส **Student\_info\_eng** ให้บริการข้อมูลส่วนตัวของนักศึกษา
- 2) เซอร์วิส **Student\_in\_dep** ให้บริการแสดงรายการรหัสนักศึกษาที่มีอยู่ในแต่ละภาควิชาและแต่ละคณะ
- 3) เซอร์วิส **Subject\_each\_student** ให้บริการข้อมูลวิชาที่นักศึกษาลงทะเบียนเรียนในแต่ละเทอม
- 4) เซอร์วิส **Subject\_info** ให้บริการรายละเอียดของวิชาเรียนแต่ละวิชา
- 5) เซอร์วิส **Subjects\_in\_dept** ให้บริการข้อมูลวิชาที่เปิดสอนทั้งหมดในแต่ละภาควิชาฯ
- 6) เซอร์วิส **Teacher\_info\_eng** ที่ให้บริการข้อมูลส่วนตัวของอาจารย์
- 7) เซอร์วิส **Teacher\_teach** ให้บริการข้อมูลวิชาที่อาจารย์แต่ละท่านสอนในแต่ละเทอม
- 8) เซอร์วิส **Teacher\_in\_dep** ให้บริการข้อมูลรหัสอาจารย์ที่มีอยู่ในแต่ละภาควิชาและแต่ละคณะ

#### 3.6.2 เซอร์วิสรีเคสต์เคอร์ของเว็บพอร์ทัล

ตัวเรียกใช้งานเซอร์วิส หรือ เซอร์วิสรีเคสต์เคอร์เป็นตัวเรียกการทำงานของเซอร์วิสที่สำนักทะเบียนฯ เพื่อรับข้อมูลที่จำเป็นในการพัฒนาเว็บพอร์ทัลเก็บลงฐานข้อมูล มีทั้งหมด 3 เซอร์วิส โดยจะทำการอัปเดตประมาณปีละสองครั้ง ซึ่งพัฒนาเป็นโปรแกรมที่แยกส่วนจากระบบพอร์ทัลส่วนกลาง ดังนี้

- 1) ฟังก์ชัน **GetStudent** เรียกใช้งานเซอร์วิส **Student\_in\_dep** และนำค่าที่ได้ไปเรียกใช้เซอร์วิส **Student\_info\_eng** และเซอร์วิส **Subject\_each\_student** เพื่อนำรายละเอียดของนักศึกษาทั้งหมดในภาควิชาฯเก็บลงฐานข้อมูล
- 2) ฟังก์ชัน **GetTeacher** เรียกใช้งานเซอร์วิส **Teacher\_in\_dep** และนำค่าที่ได้ไปเรียกใช้งานเซอร์วิส **Teacher\_info\_eng** และเซอร์วิส **Teacher\_teach** เพื่อนำรายละเอียดของอาจารย์ทั้งหมดในภาควิชาฯเก็บลงฐานข้อมูล
- 3) ฟังก์ชัน **GetSubjects** เรียกใช้งานเซอร์วิส **Subjects\_in\_dept** จากนั้นนำค่าที่ได้ไปเรียกใช้งานเซอร์วิส **Subject\_info** เพื่อนำรายละเอียดของวิชาเรียนในภาควิชาฯ เก็บลงฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### การทดลองและผลการทดลอง

#### 4.1 บทนำ

ในบทนี้เป็นรายละเอียดเกี่ยวกับเว็บแอปพลิเคชัน และเว็บเซอร์วิสที่ได้พัฒนาขึ้นตามจุดประสงค์ และการออกแบบดังที่ได้กล่าวไว้ในบทที่ 1 และบทที่ 3 โดยมีการนำเอาเทคโนโลยีที่กล่าวถึงในบทที่ 2 มาประยุกต์ใช้ จนสามารถพัฒนาขึ้นมาเป็นเว็บพอร์ทัลสำหรับภาควิชาวิศวกรรมคอมพิวเตอร์ โดยใช้เว็บเซอร์วิสได้

#### 4.2 เครื่องมือที่ใช้ในการพัฒนา

ตารางที่ 4.1 แสดงรายละเอียดฮาร์ดแวร์ที่ใช้ในการพัฒนา

	Portal Web Server
หน่วยประมวลผล	AMD Athlon 64 Processor 3000 + 1.8 GHz
หน่วยความจำ	512 MB
ฮาร์ดดิสก์	160 GB

ตารางที่ 4.2 แสดงรายละเอียดซอฟต์แวร์ที่ใช้ในการพัฒนา

	Portal Web Server	Registrar Web Server
Status	Service Requestor	Service Provider
Operating System	Windows Server 2003	Linux
Web Server	IIS 5.1	Apache 2.2.4
DBMS	Microsoft SQL Server 2005	MySQL
ภาษา	C# (ASP.NET)	PHP 5
Development Tool	Microsoft Visual Studio .NET 2005	Macromedia Dreamweaver 8
Extenslon	-	Nusoap

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 การพัฒนาเว็บแอปพลิเคชัน

เว็บแอปพลิเคชันที่พัฒนาขึ้นมาจะถูกรวบรวมเข้ามาเป็นโมดูลย่อย (หรือ พอร์ทัล) อยู่ในหน้าพอร์ทัลหลักหน้าเดียว โดยแต่ละโมดูลจะมีการจำกัดสิทธิการใช้งานตามสิทธิ์ของผู้ใช้งาน ซึ่งพัฒนาเป็นโมดูลดังต่อไปนี้



รูปที่ 4.1 เว็บเพจหน้าพอร์ทัล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3.1 โมดูลประกาศข่าว



รูปที่ 4.2 โมดูลประกาศข่าวตามสิทธิ์ user และ anonymous



รูปที่ 4.3 โมดูลประกาศข่าวตามสิทธิ์ super user

### 4.3.2 โมดูลรับ/ส่งข้อความ



รูปที่ 4.4 โมดูลรับ/ส่งข้อความตามสิทธิ์ super user, user

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 โมดูลเอกสารรายวิชาตามสิทธิ์ super user

รูปที่ 4.9 แบบฟอร์มเพิ่มเอกสารใน โมดูลเอกสารรายวิชาตามสิทธิ์ super user

รูปที่ 4.10 การลบเอกสารใน โมดูลเอกสารรายวิชาตามสิทธิ์ super user

#### 4.3.5 โมดูลสำนักทะเบียน

รูปที่ 4.11 โมดูลสำนักทะเบียนตามสิทธิ์ super user และ user

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3.6 โมดูลห้องสมุด



รูปที่ 4.12 โมดูลห้องสมุดตามสิทธิ์ super user, user และ anonymous

### 4.3.7 โมดูลคอนเทนต์

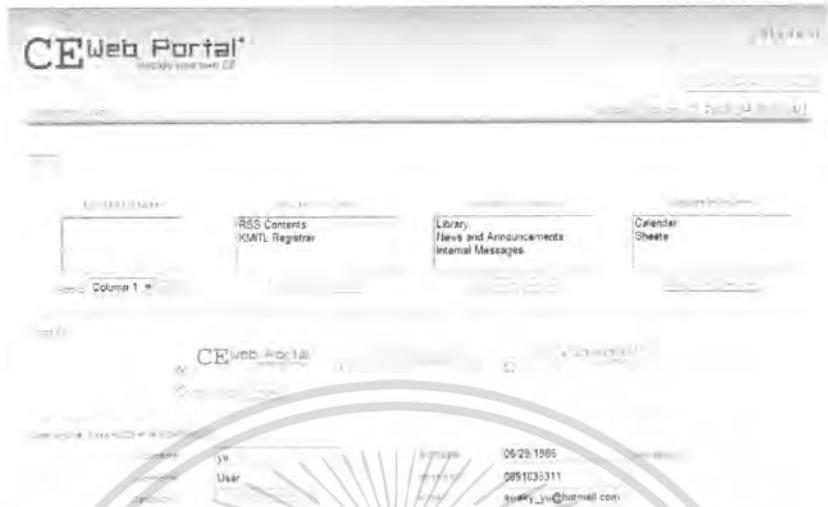


รูปที่ 4.13 โมดูลคอนเทนต์ตามสิทธิ์ super user และ user

รูปที่ 4.14 แบบฟอร์มการเพิ่มคอนเทนต์ตามสิทธิ์ super user และ user

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3.8 โมดูลปรับแต่งอินเทอร์เน็ต



รูปที่ 4.15 โมดูลปรับแต่งอินเทอร์เน็ตตามสิทธิ์ super user และ user

### 4.4 การพัฒนาเว็บเซอร์วิส

ตารางที่ 4.3 แสดงรายละเอียดเซอร์วิสที่ให้บริการที่ [www.reg.kmutt.ac.th/webservice/service/](http://www.reg.kmutt.ac.th/webservice/service/)

ชื่อเซอร์วิส	ชื่อฟังก์ชัน	Input	Output
<b>Student_in_dep</b>	GetStudents	String : faculty String : department	String : student id
<b>Student_info_eng</b>	GetInfo	Decimal : student id	Decimal : student id String : prename String:name String : surname
<b>Teacher_in_dep</b>	GetTeachers	String : faculty String : department	String : teacher id
<b>Teacher_info_eng</b>	GetInfo	Decimal : teacher id	Decimal : teacher id String : prename String : name
<b>Subject_in_dep</b>	GetSubjects	String : faculty String : department	String : subject id

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้วยวิธีการ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อเซอร์วิส	ชื่อฟังก์ชัน	Input	Output
<b>Subject_info</b>	GetInfo	String : subject id String : semester String : year	String : subject id String : name Decimal : credit Decimal : section Decimal : semester Decimal : year String : room String : building String : day String : time String : mxdate String : mxtime String : xdate String : xtime
<b>Teacher_Teach</b>	GetTeacherTeach	Decimal : teacher id String : semester String : year	String : subject id, section
<b>Subject_each_student</b>	GetSubjectEach	Decimal : student id String : semester String : year	String : subject id, section

#### 4.5 ผลการทดสอบระบบ

เมื่อพัฒนาระบบพอร์ทัลสำเร็จแล้ว ได้มีการนำไปติดตั้งบนเครื่องเซิร์ฟเวอร์ Bronze01 เพื่อเปิดให้ทดลองใช้งานบนเครือข่ายจริง โดยเรียกผ่าน URL “http:// bronze01.ce.kmit.ac.th” ได้จริง สามารถใช้งานทุกฟังก์ชันของทุกโมดูลได้ตามที่พัฒนาไว้ ไม่ว่าจะใช้เว็บเบราว์เซอร์ Internet Explorer หรือ Firefox ก็ตาม และมีการให้บริการฟีด CeNews ซึ่งเป็น RSS แสดงข่าวล่าสุดจากโมดูลประกาศข่าว โดยเปิดให้บริการเรียกผ่าน URL “http://bronze01.ce.kmitl.ac.th/cePortal/CeNews.aspx” และในส่วนของ Web Services ก็สามารถเรียกใช้บริการได้โดยไม่เกิดปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

# บทสรุปและวิจารณ์

### 5.1 บทสรุป

ในการพัฒนาระบบซีอีเว็บพอร์ทัลนี้ ในช่วงเริ่มต้นการพัฒนาจะต้องทำการศึกษาเทคโนโลยีต่าง ๆ ที่จำเป็น ได้แก่ ศึกษาแนวคิดเรื่องเว็บพอร์ทัล ซึ่งได้นำมาเป็นรูปแบบหลักในการเข้าถึงข้อมูลของผู้ใช้งานแต่ละคน และศึกษาเทคโนโลยีด้านเว็บแอปพลิเคชัน โดยเฉพาะภาษา ASP.NET (C#) ที่จะนำมาเป็นภาษาหลักในการพัฒนา รวมถึงการติดต่อฐานข้อมูลด้วย ADO.NET ซึ่งสองเทคโนโลยีนี้เป็นส่วนหนึ่งของคอทเน็ตเฟรมเวิร์ก

การพัฒนาในรูปแบบเว็บพอร์ทัลเพื่อให้เป็นเสมือนจุดเชื่อมต่อของข้อมูลที่มาจากระบบที่ถูกพัฒนาด้วยแพลตฟอร์มที่ต่างกัน จำเป็นต้องศึกษาเทคโนโลยีเว็บเซอร์วิส อันเกิดขึ้นจากเทคโนโลยีหลายอย่าง เช่น ภาษา XML, โปรโตคอล SOAP, และภาษา WSDL เพื่อแก้ปัญหาเรื่องความต่างของแพลตฟอร์ม และมีการทดลองพัฒนาเว็บเซอร์วิสข้ามแพลตฟอร์มระหว่างภาษา PHP และภาษา ASP.NET ซึ่งพบว่าสามารถรับ – ส่งข้อมูลข้ามแพลตฟอร์มได้จริง แต่ยังคงมีปัญหาอยู่บ้าง สืบเนื่องจากมาตรฐานเว็บเซอร์วิสที่ไม่ได้ถูกกำหนดมาอย่างเคร่งครัด

ก่อนการลงมือพัฒนาเว็บแอปพลิเคชัน ต้องมีการออกแบบระบบซีอีเว็บพอร์ทัลให้สอดคล้องกับความต้องการของผู้ใช้งานมากที่สุดตามที่ได้ทำการสำรวจ จึงออกแบบระบบในภาพรวมและในรายละเอียดของแต่ละโมดูล รวมถึงนำ UML (Unified Model Language) 2 แบบ คือ Use case Diagram และ Activity Diagram มาช่วยในการออกแบบให้การพัฒนาเป็นไปตามขั้นตอน และไม่สับสน ในส่วนของการออกแบบฐานข้อมูลได้ออกแบบโดยใช้ ORM (Object Role Model) เพื่อสร้างโครงสร้างที่ไม่มี ความซ้ำซ้อน (Redundancy) นอกจากนี้ ยังออกแบบเว็บเซอร์วิสที่จะพัฒนาเพื่อเปิดให้ใช้บริการที่สำนักทะเบียนฯ และเพื่อดึงข้อมูลมาใช้งานในระบบพอร์ทัล โดยได้ออกแบบโดยคำนึงถึงการใช้งานของผู้อื่นที่อาจเกิดขึ้นในอนาคต

ช่วงลงมือพัฒนาระบบเว็บพอร์ทัลก็ได้นำเทคโนโลยีและแนวคิดที่ได้ศึกษามาประยุกต์ใช้ในการสร้างเว็บพอร์ทัลให้ตรงตามที่ได้ออกแบบไว้ให้มากที่สุด และเกิดปัญหาในการใช้งานน้อยที่สุด

เมื่อพัฒนาเสร็จแล้วพบว่าระบบซีอีเว็บพอร์ทัลนี้สามารถเรียกใช้งานเว็บเซอร์วิสจากสำนักทะเบียนฯ ได้ และเปิดบริการระบบเว็บพอร์ทัล รวมทั้งให้บริการเว็บพีคของโมดูลข่าวได้ โดยเฉพาะในส่วนของ GUI จะพบว่าค่อนข้างใช้งานง่าย สวยงาม และตรงแนวคิดของเว็บพอร์ทัล

กล่าวได้ว่า ระบบซีอีเว็บพอร์ทัลนี้ได้ถูกออกแบบและพัฒนาให้สามารถนำมาใช้งานได้จริงภายในภาควิชาฯ และสอดคล้องกับความต้องการของผู้ใช้งานในระดับหนึ่ง แต่อาจยังไม่

ครอบคลุมความต้องการทั้งหมด ซึ่งจัดได้ว่าเป็นข้อจำกัดอย่างหนึ่งของระบบที่ต้องทำการพัฒนาต่อไป

## 5.2 ปัญหาอุปสรรคและแนวทางแก้ไข

- 1) ปัญหาการพัฒนาในรูปแบบของพอร์ตเลทในช่วงเริ่มต้นพัฒนา  
เนื่องจาก ในช่วงเริ่มแรกพัฒนาส่วน GUI ด้วยภาษา JavaScript ทำให้ไม่สามารถสร้างเป็นรูปแบบพอร์ตเลทได้  
**แก้ไข** โดยใช้การพัฒนาด้วย C # ในลักษณะของ Code Behind
- 2) ปัญหาเรื่อง Overhead ของรูปแบบการรับส่งข้อมูลที่ใช้เว็บเซอร์วิสค่อนข้างสูง และข้อมูลมีจำนวนค่อนข้างมาก ทำให้การทำงานในการเรียกใช้แต่ละครั้งค่อนข้างช้า  
**แก้ไข** โดยพัฒนาให้ไม่ต้องมีการดึงข้อมูลใหม่ทุกครั้งทีลือกอิน และทำการตั้งค่าเป็นช่วงเวลาแทน
- 3) ปัญหาในการขออนุญาตเชื่อมต่อกับระบบห้องสมุด ซึ่งไม่ได้รับอนุญาตให้ติดต่อเครื่องเซิร์ฟเวอร์โดยตรง  
**แก้ไข** โดยต้องเชื่อมต่อในลักษณะของ HTML ธรรมดาแทน
- 4) ปัญหาเรื่องการพัฒนาในสภาวะแวดล้อมที่ต่างกัน ทำให้โปรแกรมที่ใช้งานได้ตอนพัฒนา กลับใช้ไม่ได้ตอนอยู่บนเครื่องเซิร์ฟเวอร์  
**แก้ไข** โดยพยายามพัฒนาบนสภาวะแวดล้อมที่ใกล้เคียงกับระบบที่ใช้งานจริงมากที่สุด
- 5) ปัญหาการออกแบบระบบเว็บพอร์ทัลและฐานข้อมูลไม่ตรงกับความต้องการใช้งานจริง ทำให้ต้องมีการปรับแก้เพิ่มเติม  
**แก้ไข** โดยพยายามแก้ไขส่วนบกพร่องโดยไม่ให้กระทบส่วนอื่น
- 6) ปัญหาเรื่อง encoding ภาษาในการใช้งานเว็บเซอร์วิสระหว่างภาษา PHP และ ASP.NET ทำให้ไม่สามารถรับ-ส่งข้อมูลที่เป็นภาษาไทยได้
- 7) ปัญหาการประสานงานในการพัฒนาเว็บเซอร์วิสร่วมกับสำนักทะเบียนฯ ที่ไม่สามารถทำการทดสอบและดีบั๊กได้ทันที

## 5.3 แนวทางการพัฒนาต่อ

- 1) พัฒนาเว็บพอร์ทัลให้ถูกต้องตามมาตรฐานของ W3C ในแต่ละภาษาที่ใช้งาน เช่น ภาษา HTML
- 2) พัฒนาระบบพอร์ทัลให้รองรับการใช้งานในปริมาณมาก และมีการจัดการเรื่องทรานแซคชัน (transaction) ให้เหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 3) ออกแบบและพัฒนาในเรื่องของการรักษาความปลอดภัย ทั้งในส่วนจากระบบกลางและเว็บไซต์
- 4) นำมาตรฐานเว็บไซต์อื่น ๆ ที่เกี่ยวข้องกันมาพัฒนาร่วมกับระบบพอร์ทัล เช่น WS-Transaction
- 5) พัฒนาเว็บไซต์และเว็บฟีดให้บริการกับระบบภายนอกมากขึ้น
- 6) หากหน่วยงานอื่นในสถาบันได้เปิดให้บริการเว็บไซต์แล้ว ก็น่าจะพัฒนาโมดูลเพิ่มเติมให้เป็นเว็บพอร์ทัลที่สมบูรณ์
- 7) พัฒนาการใช้งานให้มากขึ้นสำหรับอาจารย์ และเจ้าหน้าที่ เช่น เรื่องการออกรายงานคะแนนหรือเกรดของนักศึกษา
- 8) พัฒนาโมดูลเพิ่มเติมให้ครอบคลุมมากขึ้น เช่น เชื่อมต่อกับอีเมล, โมดูลเอกสารวิชาโครงการ และ โมดูลสนทนา เป็นต้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

GDI Camp. 2549. **ความรู้เบื้องต้นเกี่ยวกับเว็บเซอร์วิส**. [Online].

Available : <http://gdi.mict.go.th/gdi/press.html>.

IBM. 2000. **The Web services (r)evolution: Part 1**. [Online].

Available : <http://www.ibm.com/developerworks/webservices/library/ws-peer1.html>.

Wikipedia. 2001. **Wikipedia, the Free Encyclopedia**. [Online].

Available : <http://wikipedia.org>.

W3Schools. **W3Schools Online Web Tutorials**. [Online].

Available : <http://www.w3schools.com>.

Cover Pages. 2002. **Cover Pages Simple Object Access Protocol (SOAP)**. [Online].

Available : <http://xml.coverpages.org/techReports.html>.

**SOAP Basics - Chapter Contents**. [Online].

Available : [http://www.vbip.com/xml/soap\\_syd.asp](http://www.vbip.com/xml/soap_syd.asp).

**alphaWorks SOA and Web services New to SOA and Web services**. [Online].

Available : <http://www.alphaworks.ibm.com/webservices>.

W3C Note. 2001. **Web Service Definition Language (WSDL)**. [Online].

Available : <http://www.w3.org/TR/wsdl>.

**Using WSDL in SOAP applications**. [Online].

Available : <http://www.ibm.com/developerworks/webservices/>.

W3Schools. **WSDL Documents**. [Online].

Available : <http://www.w3schools.com/wsdl/default.asp>.

W3C Note. 2001. **Simple Object Access Protocol (SOAP) 1\_1**. [Online].

Available : <http://www.w3.org/TR/SOAP>.

**ASP.NET Playground for Thai folks**. [Online].

Available : <http://www.aspnetthai.com/Home/tabid/36/Default.aspx>.

**GreatFriends.Biz .NET Community**. [Online].

Available : <http://www.greatfriends.biz/index2.asp>.

**ASP.NET FAQ**. [Online].

Available : [http://www.syncfusion.com/FAQ/aspnet/WEB\\_c23c.aspx](http://www.syncfusion.com/FAQ/aspnet/WEB_c23c.aspx).

**Microsoft ASP.NET QuickStarts Tutorial**. [Online].

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ศุภชัย สมพานิช. 2549. **คู่มือ ASP.NET 2.0 ฉบับสมบูรณ์**. นนทบุรี : ไอดีซีฯ .

นเรศ นันทบุรุษย์ และจิรวัดน์ ผดุงกิจจานนท์. 2550. **พัฒนาเว็บแอปพลิเคชัน ด้วย ASP.NET 2.0 ฉบับสมบูรณ์**. กรุงเทพฯ : ซัคเซส มีเดีย.

พงษ์พันธ์ ศิวิลัย. 2549. **SQL Server 2005ฉบับสมบูรณ์**. กรุงเทพฯ : ซีเอ็ดยูเคชั่น.

ลาภลอย วานิชอังกูร. 2550. **เรียนรู้ด้วยตนเอง OOP C# ASP.NET**. กรุงเทพฯ : ซีเอ็ดยูเคชั่น.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ก.

## แบบสอบถามประกอบการจัดทำเว็บส่วนบุคคลของภาควิชาวิศวกรรม คอมพิวเตอร์ (CE Web Portal \*)

(\*CE Web Portal คือเว็บศูนย์กลางที่สามารถติดต่อกับระบบต่าง ๆ ในสถาบันได้ ในหน้าเว็บเดียว และสามารถจัดการรูปแบบส่วนตัวได้)

### 1. ข้อมูลทั่วไป

#### 1.1 สถานะ

- อาจารย์  
 นักศึกษาปริญญาตรี  ชั้นปี 2  ชั้นปี 3  
 ชั้นปี 4  
 นักศึกษาปริญญาโท

### 2. ลักษณะการใช้งานเว็บภาควิชา

#### 2.1 ท่านเคยใช้งานเว็บภาควิชาหรือไม่

- เคย  ไม่เคย

#### 2.2 ท่านคิดว่าเว็บภาคของท่านทำอะไรได้บ้าง (ตอบได้มากกว่า 1 ข้อ)

- ประกาศจากอาจารย์, เจ้าหน้าที่  
 แสดงรายละเอียดรายวิชา, หลักสูตร  
 แสดงข้อมูลอาจารย์, บุคลากร  แสดงข้อมูลนักศึกษาปัจจุบัน  
 แสดงข้อมูลศิษย์เก่า  เว็บบอร์ด  
 แสดงข้อมูล Projects, Research  ดาวน์โหลดเอกสารเรียน  
 อื่นๆ \_\_\_\_\_

#### 2.3 ท่านพอใจในประสิทธิภาพของเว็บภาควิชาหรือไม่

- พอใจ  พอใจบ้าง  ไม่ค่อยพอใจ  ไม่พอใจมาก

#### 2.4 ท่านอยากให้เว็บภาคของท่านมีอะไรเพิ่มเติมจากเดิมบ้าง

---



---



---



---

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. สอบถามความต้องการ

3.1 ท่านมีความคิดเห็นอย่างไร หากมี CE web Portal ขึ้นมาให้ใช้งานจริงแทนเว็บไซต์ปัจจุบัน

- น่าสนใจมาก       น่าสนใจ       เฉยๆ       ไม่น่าสนใจ

3.2 ท่านคิดว่าการใช้งานส่วนใด **ไม่**มีความจำเป็นต่อ CE web Portal บ้าง (ตอบได้มากกว่า 1 ข้อ)

- |                                                                     |                                                                 |
|---------------------------------------------------------------------|-----------------------------------------------------------------|
| <input type="checkbox"/> ประกาศจากอาจารย์, เจ้าหน้าที่              | <input type="checkbox"/> แสดงรายละเอียดรายวิชา, หลักสูตร        |
| <input type="checkbox"/> แสดงข้อมูลอาจารย์, บุคลากร                 | <input type="checkbox"/> แสดงข้อมูลนักศึกษาปัจจุบัน             |
| <input type="checkbox"/> แสดงข้อมูลศิษย์เก่า                        | <input type="checkbox"/> เว็บไซต์                               |
| <input type="checkbox"/> แสดงข้อมูล Projects, Research              | <input type="checkbox"/> ดาวน์โหลดเอกสารเรียน                   |
| <input type="checkbox"/> แสดงผลการเรียน                             | <input type="checkbox"/> ตรวจสอบจบการศึกษา                      |
| <input type="checkbox"/> ดาวน์โหลดเอกสารราชการ                      | <input type="checkbox"/> แสดงค่าค้างชำระห้องสมุด                |
| <input type="checkbox"/> ค้นหาหนังสือจากห้องสมุด                    | <input type="checkbox"/> เตือนเมื่อใกล้เวลาคืนหนังสือ และยืมต่อ |
| <input type="checkbox"/> ส่งข้อความส่วนตัวระบุปลายทางได้            | <input type="checkbox"/> เตือนเมื่อต้องส่งงาน                   |
| <input type="checkbox"/> ปฏิทินจัดการส่วนบุคคล                      | <input type="checkbox"/> ปฏิทินการศึกษา                         |
| <input type="checkbox"/> จัดรูปแบบหน้าเว็บส่วนบุคคล(คล้าย msn live) |                                                                 |
| <input type="checkbox"/> พยากรณ์อากาศ                               | <input type="checkbox"/> คำทำนายจักรราศี                        |
| <input type="checkbox"/> ข่าวสาร RSS                                | <input type="checkbox"/> ติดต่อ web mail                        |
| <input type="checkbox"/> อื่นๆ _____                                |                                                                 |

3.3 ท่านคิดว่า CE web Portal ควรมีความสามารถอะไรที่แตกต่างจากบริการอื่น ๆ ของสถาบัน

---



---



---

3.4 ข้อคิดเห็นเพิ่มเติม

---



---



---

ขอขอบพระคุณทุกท่านที่ให้ความร่วมมือ  
นางสาวนันทนา วนารธรรม  
นางสาวอรรดยา เมธินทรีย์

## ภาคผนวก ข.

### ตัวอย่างการทดลองพัฒนาเว็บเซอร์วิสข้ามแพลตฟอร์ม

#### 1 บทนำ

ในบทนี้เป็นการทดลองพัฒนาเว็บเซอร์วิสด้วยภาษา และแพลตฟอร์มที่แตกต่างกัน คือ เซอร์วิสที่พัฒนาด้วย .NET Technology กับภาษา PHP เพื่อเป็นการทดสอบว่าเว็บเซอร์วิสมี หลักการทำงาน และคุณสมบัติตามทฤษฎีที่ได้ศึกษามาจริง และศึกษาปัญหาที่อาจเกิดขึ้นก่อนที่จะ นำไปพัฒนาระบบจริง

#### 2 การทดลองพัฒนาเว็บเซอร์วิสด้วยภาษา PHP

##### 2.1 ซอฟต์แวร์ที่ใช้ในการพัฒนา

ตารางที่ ข.1 แสดงรายละเอียดซอฟต์แวร์ที่ใช้ในพัฒนาเซอร์วิสด้วยภาษา PHP

	Computer 1	Computer 2
<b>Status</b>	Service Provider	Service Requestor
<b>Operating System</b>	Windows XP SP2	Windows XP SP2
<b>Web Server</b>	Apache 2.2.4	Apache 2.2.4
<b>DBMS</b>	MySQL 5.0.27	MySQL 5.0.27
<b>ภาษา</b>	PHP 5.2.1	PHP 5.2.1
<b>Extension</b>	Nusoap	Nusoap

##### 2.2 การทดลองสร้าง และใช้งานเว็บเซอร์วิส

เป็นการจำลองเปิดบริการเซอร์วิสที่ **Computer 1** ให้ผู้ร้องขอหรือ **Computer 2** สามารถ สอบถามเกรดเฉลี่ยของคนได้โดยใช้ “รหัสนักศึกษา” เป็นค่าที่ส่ง ไปพร้อมกับคอนร้องขอบริการ เมื่อเซอร์วิสของ **Computer 1** ได้รับการร้องขอก็จะทำการประมวลผลด้วยการดึงข้อมูลจาก ฐานข้อมูล แล้วทำการส่ง “เพศ”, “ชื่อ”, “นามสกุล” และ “เกรดเฉลี่ย” กลับไปให้ผู้ร้องขอ ดังรูปที่ ข.1 และ ข.2

ตรวจสอบเกรด	
รหัสนักศึกษา :	47010003 <input type="button" value="Submit"/>

รูปที่ ข.1 รอรับรหัสนักศึกษามาบน Computer 2 เพื่อร้องขอเซอร์วิส

ตรวจสอบเกรด	
รหัสนักศึกษา :	<input type="button" value="Submit"/>
รหัสนักศึกษา	47010003
ชื่อ-นามสกุล	Ms.CCC ccc
เกรดเฉลี่ย	2.87

รูปที่ ข.2 แสดงผลข้อมูลที่ตอบกลับมาจาก Computer 1

โดยสามารถแบ่งการทำงานเป็นฝั่งไคลเอนท์ (Service Requestor) กับฝั่งเซิร์ฟเวอร์ (Service Provider) โดยมีรายละเอียดของโค้ด ดังรูปที่ ข.3 และ ข.4

- ฝั่งเซิร์ฟเวอร์ จะมีไฟล์ `service_gpa.php` (รูปที่ ข.3) ทำงานอยู่โดยเป็นผู้ให้บริการ เซอร์วิสที่ชื่อ `gpaService` โดยมีโอเปอเรชั่น `GetGPA` ที่จะรับค่ารหัสนักศึกษามา นำไปค้นหาในฐานข้อมูลแล้วจึงส่งข้อมูลกลับไปให้แก่ผู้เรียกใช้เซอร์วิส โดยเริ่มจากการอ้างอิงถึง `nuSOAP` เพื่อเรียกใช้งานฟังก์ชันของ `nuSOAP` จากนั้นจะสร้าง ออบเจกต์ `Soap server` ขึ้นมา จากนั้นก็กำหนดค่าต่างๆ บน `WSDL` เช่น `data type` ที่กำหนดขึ้น แล้วจึงทำการลงทะเบียนเว็บเซอร์วิสที่สร้างขึ้นมา จากนั้นเป็นส่วนของการเขียนฟังก์ชันที่จะเปิดให้บริการ และส่วนของการรอรับเมสเสจที่ส่งมาเพื่อร้องขอเซอร์วิส
- ฝั่งไคลเอนท์ จะมีไฟล์ `client.php` (รูปที่ ข.4) ทำงานอยู่โดยเป็นผู้เรียกใช้เซอร์วิส และรับค่ากลับมาแสดงผล โดยเป็นการเขียนภาษา `PHP` แทรกไว้กับ `HTML` เพื่อใช้แสดงข้อมูลที่ได้รับมา เริ่มจากสร้าง `form` รับค่ารหัสนักศึกษาจากยูสเซอร์ จากนั้นก็สร้างออบเจกต์ `soap client` ส่งชื่อฟังก์ชันที่ต้องการเรียกใช้ไปยังที่อยู่ของเซอร์วิสพร้อมกับรหัสนักศึกษา ในที่นี้คือ `GetGPA` และ `47010003` แล้วจึงนำค่าที่ได้จาก `response` มาแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<?php
require('../nusoap/nusoap.php');
$namespace = "http://www.reg.kmitl.ac.th";
$server = new soap_server();
$server->debug_flag = false;
$server->configureWSDL("gpaService", $namespace);
$server->wsdl->schemaTargetNamespace = $namespace;
$server->wsdl->addComplexType(
    'gpa', 'complexType', 'struct', 'all', '',
    array(
        'id' => array('name' => 'id', 'type' => 'xsd:decimal'),
        'gender' => array('name' => 'gender', 'type' => 'xsd:string'),
        'f_name' => array('name' => 'f_name', 'type' => 'xsd:string'),
        'l_name' => array('name' => 'l_name', 'type' => 'xsd:string'),
        'grade' => array('name' => 'grade', 'type' => 'xsd:double')
    )
);
$server->register('GetGPA',
    array('id' => 'xsd:decimal'), array('return' => 'tns:gpa'),
    $namespace, $namespace . '#GetGPA',
    'rpc', 'encoded', 'Get GPA from registra'
);
function GetGPA($id)
{
    mysql_connect('localhost', 'root', '0000')or
die(mysql_error());
    mysql_select_db(student) or die( mysql_error() );
    $charset = "SET NAMES 'utf8'";
    mysql_query($charset) or die('Invalid query: ' .
mysql_error());
    $sql = "SELECT * FROM `gpa` WHERE `ID` = '$id'";
    $rst = mysql_query($sql);
    $row = mysql_fetch_array($rst);
    if($rst) mysql_free_result($rst);
    $gender = $row['gender'];
    $f_name = $row['f_name'];
    $l_name = $row['l_name'];
    $grade = $row['grade'];
    $data = array(
        "id" => $id, "gender" => $gender, "f_name" => $f_name,
        "l_name" => $l_name, "grade" => $grade
    );
    return $data;
}
$HTTP_RAW_POST_DATA = isset($GLOBALS['HTTP_RAW_POST_DATA']) ?
$GLOBALS['HTTP_RAW_POST_DATA'] : '';
$server->service($HTTP_RAW_POST_DATA);
exit();
?>

```

รูปที่ ข.3 ไฟล์ service\_gpa.php

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title>Check GPA</title>
<link href="style.css" rel="stylesheet" type="text/css">
</head>
<body>
<div align="center" id="bef">
  <table width="350" border="1" cellpadding="5" cellspacing="0"
bordercolor="#999999"><tr>
  <td align="center" bgcolor="#999999" class="bef_w">ตรวจสอบเกรด
  </td></tr><tr><td align="center">
    <form name="form1" method="post"
action="client.php?cmd=3">
    รหัสนักศึกษา: <input name="id" type="text" id="id" maxlength="8">
      <input type="submit" name="Submit" value="Submit">
    </form></td></tr></table></div>
<?php
  $id = $_POST['id'];
  require('../nusoap/nusoap.php');
  $serviceURL =
"http://localhost/test_student01/service_gpa.php";
  $s = new soapclient($serviceURL);
  $data = $s->call('GetGPA', "$id");
  if($cmd == 1) echo '<xml>'.$s->request.'</xml>';
  else if($cmd == 2) echo '<xml>'.$s->response.'</xml>';
  else {
    $id = $data['id']; $gender = $data['gender'];
    $f_name = $data['f_name']; $l_name =
    $data['l_name']; $grade = $data['grade'];}
  if ($id){
    echo ' <div align="center" id="aft">
      <table width="350" border="1" cellpadding="0"
        cellspacing="0" bordercolor="#999999">
        <tr><td align="center">
          <table width="340" border="0" cellspacing="0"
            cellpadding="3">
          <tr><td width="93">รหัสนักศึกษา</td>
            <td width="195">'. $id.'</td>
          </tr><tr>
            <td>ชื่อ-นามสกุล</td><td>';
          if($gender=="M")
            echo "Mr.". $f_name." ". $l_name;
          else if($gender=="F")
            echo "Ms.". $f_name." ". $l_name;
          else echo "";
          echo '</td></tr><tr>
            <td>เกรดเฉลี่ย</td><td>'. $grade.'</td></tr >
          </table></td></tr> </table></div>';}
?>
</body></html>

```

#### รูปที่ ข.4 ไฟล์ client.php

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากทำการเปิดไฟล์ `service_gpa.php` ด้วยเว็บเบราว์เซอร์ผ่าน URL `http://localhost/test_student01/service_gpa.php` จะเห็นหน้าต่างคล้าย reference manual ดังรูปที่ ข.5 แสดงให้เห็นว่าที่ URL นี้มีเว็บเซอร์วิสที่ชื่อ `gpaService` เปิดให้บริการอยู่ โดยมีโอเปอเรชั่นชื่อ `GetGPA` อยู่โอเปอเรชั่นเดียว และเมื่อเลือกที่ `GetGPA` จะแสดงรายละเอียดของโอเปอเรชั่น `GetGPA` ดังรูปที่ ข.6

gpaService

View the **WSDL** for the service. Click on an operation name to view its details.

**GetGPA**

รูปที่ ข.5 เซอร์วิส gpaService ที่เปิดให้บริการ

gpaService

View the **WSDL** for the service. Click on an operation name to view its details.

**GetGPA**

Name:		GetGPA
Binding:		gpaServiceBinding
Endpoint:		http://localhost/test_student01/service_gpa.php
SoapAction:		http://www.reg.kmitl.ac.th#GetGPA
Style:		rpc
Input:		
use:	encoded	
namespace:	http://www.reg.kmitl.ac.th	
encodingStyle:	http://schemas.xmlsoap.org/soap/encoding/	
message:	GetGPARequest	
parts:		
id:	xsd:decimal	
Output:		
use:	encoded	
namespace:	http://www.reg.kmitl.ac.th	
encodingStyle:	http://schemas.xmlsoap.org/soap/encoding/	
message:	GetGPAResponse	
parts:		
return:	tns:gpa	
Namespace:	http://www.reg.kmitl.ac.th	
Transport:	http://schemas.xmlsoap.org/soap/http	
Documentation:	Get GPA	

รูปที่ ข.6 รายละเอียดของโอเปอเรชั่น `GetGPA` ที่เปิดให้บริการ

และหากเปิดดู WSDL เบราวเซอร์ก็จะแสดงไฟล์ WSDL ของเซอร์วิสนี้ให้เห็นดังรูปที่ ข.7 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า ซึ่งในที่นี้เราใช้ `nuSOAP` ในการช่วยสร้าง WSDL และสร้าง SOAP Message ให้โดยอัตโนมัติ ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<?xml version="1.0" encoding="ISO-8859-1" ?>
- <definitions xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:tns="http://www.reg.kmitl.ac.th" xmlns:soap="http://schemas.xmlsoap.org/soap/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns="http://schemas.xmlsoap.org/wsdl/"
  targetNamespace="http://www.reg.kmitl.ac.th">
- <types>
  - <xsd:schema targetNamespace="http://www.reg.kmitl.ac.th">
    <xsd:import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
    <xsd:import namespace="http://schemas.xmlsoap.org/wsdl/" />
    - <xsd:complexType name="gpa">
      - <xsd:all>
        <xsd:element name="id" type="xsd:decimal" />
        <xsd:element name="gender" type="xsd:string" />
        <xsd:element name="f_name" type="xsd:string" />
        <xsd:element name="l_name" type="xsd:string" />
        <xsd:element name="grade" type="xsd:double" />
      </xsd:all>
    </xsd:complexType>
  </xsd:schema>
</types>
- <message name="GetGPARequest">
  <part name="id" type="xsd:decimal" />
</message>
- <message name="GetGPAResponse">
  <part name="return" type="tns:gpa" />
</message>
- <portType name="gpaServicePortType">
  - <operation name="GetGPA">
    <documentation>Get GPA from registra</documentation>
    <input message="tns:GetGPARequest" />
    <output message="tns:GetGPAResponse" />
  </operation>
</portType>
  <binding name="gpaServiceBinding" type="tns:gpaServicePortType">

```

รูปที่ ข.7 WSDL ที่อธิบายเซอร์วิส gpaService

### 3 การทดลองพัฒนาเว็บเซอร์วิสด้วยภาษา ASP.NET

#### 3.1 ขอฟต์แวร์ที่ใช้ในการพัฒนา

ตารางที่ ข.2 แสดงรายละเอียดซอฟต์แวร์ที่ใช้ในการพัฒนาเซอร์วิสด้วยภาษา ASP.NET

	Computer 3	Computer 4
<b>Status</b>	Service Provider	Service Requestor
<b>Operating System</b>	Windows XP SP2	Windows XP SP2
<b>Web Server</b>	IIS5.1	IIS5.1
<b>DBMS</b>	Microsoft SQL Server 2005	Microsoft SQL Server 2005
<b>ภาษา</b>	C#	C#
<b>Development Tool</b>	Microsoft Visual Studio .NET 2005	Microsoft Visual Studio .NET 2005

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2 การทดลองสร้างและใช้งานเว็บเซอร์วิส

เป็นการจำลองเปิดบริการเซอร์วิสที่ **Computer 3** เพื่อส่งคำว่า “Hello World” กลับมาให้ผู้ร้องขอ หรือ **Computer 4** ดังรูปที่ ข.8



รูปที่ ข.8 เซอร์วิสแสดงคำว่า Hello World

### 3.3 ขั้นตอนการสร้างเซอร์วิสฝั่งเซิร์ฟเวอร์

เมื่อสร้างเว็บไซต์ที่เป็น ASP.NET Web Service ขึ้นมาดังรูปที่ ข.9 จะมีการสร้าง service.cs ขึ้นมาให้โดยอัตโนมัติ โดยจะมีเว็บเมธอด HelloWorld สร้างขึ้นมาให้ ดังรูปที่ ข.10 ซึ่งเป็นส่วนที่สามารถเรียกใช้ผ่านเซอร์วิสได้ และผู้พัฒนาสามารถเขียนเมธอดเพิ่มได้เท่าที่ต้องการ



รูปที่ ข.9 การสร้างเว็บไซต์ ASP.NET Web Service

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

using System;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;

[WebService(Namespace = "http://tempuri.org/")]
[WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
public class Service : System.Web.Services.WebService
{
    public Service () {

        // Uncomment the following line if using designed
        // components
        // InitializeComponent();
    }

    [WebMethod]
    public string HelloWorld() {
        return "Hello World";
    }
}

```

รูปที่ ข.10 ไฟล์ service.cs

### 3.4 ขั้นตอนการสร้างเซอร์วิสฝั่งไคลเอนท์

เมื่อสร้างเว็บที่จะพัฒนาด้วยภาษา ASP ขึ้นมาดังรูปที่ ข.11 จะได้ไฟล์มา 2 ไฟล์คือ Default.aspx ที่เป็นส่วนของเว็บเพจ และ Default.aspx.cs ที่เป็นส่วนของโค้ด C# โดยเริ่มต้นด้วยการสร้างปุ่มขึ้นมารอการคลิกของยูสเซอร์เพื่อเรียกใช้งานเว็บเซอร์วิสดัง โค้ดรูปที่ ข.12



รูปที่ ข.11 การสร้างหน้าเว็บ default.aspx

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="Default.aspx.cs" Inherits="_Default" %>

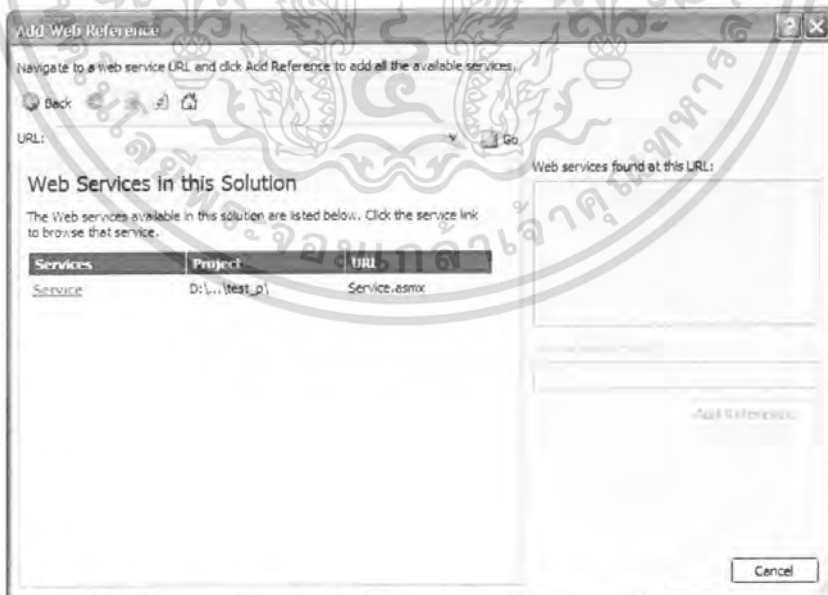
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title>Untitled Page</title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:Button ID="Button1"runat="server"Text="hello"OnClick="B_Click"/>
            <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
        </div>
    </form>
</body>
</html>

```

รูปที่ ข.12 ไฟล์ default.aspx ที่เพิ่มปุ่มและช่องแสดงผล

จากนั้นจึง Add Reference (Add Web Reference -> Web Services in this Solution) เพื่อค้นหาเซอร์วิสที่ต้องการเรียกใช้งานโดยการกรอก URL หรือค้นหาบนเครือข่ายต่าง ๆ มีการแสดงรายชื่อเซอร์วิสที่พบดังรูปที่ ข.13 และเมื่อเลือกเซอร์วิสที่ต้องการจะมีการแสดงรายละเอียดของเซอร์วิส และรายชื่อโอเปอเรชั่นที่มีดังรูปที่ ข.14



รูปที่ ข.13 วิธี Add Web Reference ในการเซอร์วิสฝั่งไคลเอนท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.14 แสดงรายละเอียดของเซอร์วิสที่ชื่อ Service

แล้วจึงตั้งชื่อ Web Reference ซึ่งในที่นี้ชื่อ localhost แล้วเลือก Add Reference จะได้ไฟล์ Service.disco, Service.discomap และ Service.wsdl มาดังรูปที่ ข.15 แล้วจึงเขียนเมธอด B\_Click เพื่อเรียกใช้โอเปอเรชัน HelloWorld() ของเซอร์วิส Service ผ่าน Web Reference ที่ชื่อ localhost แล้วนำค่าที่ได้ไปแสดงที่ TextBox1 ดังรูปที่ ข.16 และเมื่อนำไปแสดงผลจะได้ผลดังรูปที่ ข.8



รูปที่ ข.15 ไฟล์ที่ได้จากการ Add Web Reference

```

...
public partial class Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void B_Click(object sender, EventArgs e)
    {
        localhost.Service S = new localhost.Service();
        string hw = S.HelloWorld();
        TextBox1.Text = hw;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับอาจารย์ที่ออกข้อสอบเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
รูปที่ ข.16 ไฟล์ default.aspx.cs ที่มีเมธอด B\_Click  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4 การทดลองพัฒนาเว็บเซอร์วิสด้วยภาษา PHP และ ASP.NET

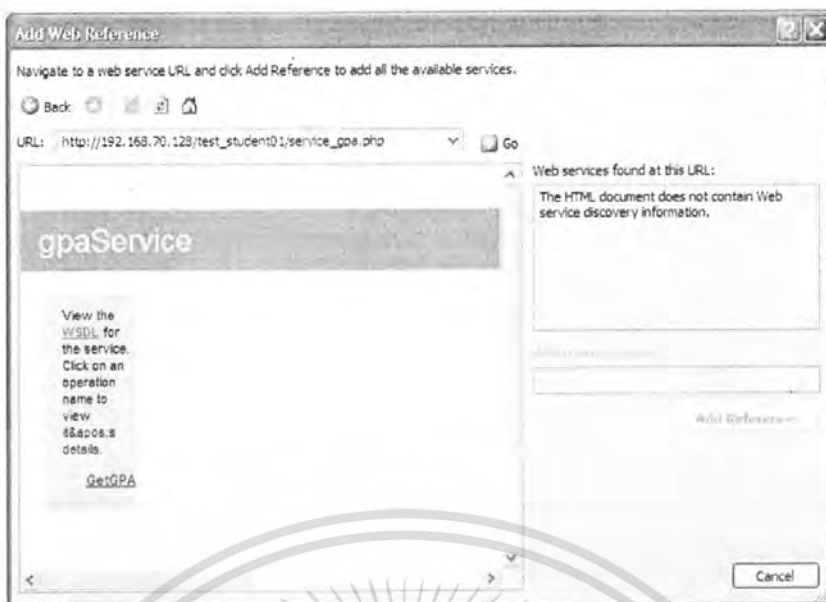
### 4.1 ซอฟต์แวร์ที่ใช้ในการพัฒนา

ตารางที่ ข.3 แสดงรายละเอียดซอฟต์แวร์ที่ใช้ในการพัฒนาเซอร์วิสด้วยภาษา PHP และ ASP.NET

	Computer 1	Computer 4
<b>Status</b>	Service Provider	Service Requestor
<b>Operating System</b>	Windows XP SP2	Windows XP SP2
<b>Web Server</b>	Apache 2.2.4	IIS5.1
<b>DBMS</b>	MySQL 5.0.27	Microsoft SQL Server 2005
<b>ภาษา</b>	PHP 5.2.1	C#
<b>Development Tool</b>	Macromedia Dreamweaver 8	Microsoft Visual Studio .NET 2005
<b>Extension</b>	Nusoap	-

### 4.2 การทดลองสร้าง และใช้งานเว็บเซอร์วิส

ในการทดลองนี้จะเรียกใช้เซอร์วิส gpaService ที่เขียนด้วยภาษา PHP ที่ให้บริการบน Computer 1 ผ่านเว็บไซต์ที่เขียนด้วยภาษา ASP.NET จาก Computer 4 โดยมีขั้นตอนการสร้างหน้าเว็บเหมือนการเรียกใช้เว็บเซอร์วิสด้วย ASP.NET แต่ลักษณะการแสดงรายละเอียดของเซอร์วิสในขั้นตอน Add Web Reference และไฟล์ที่ได้จะแตกต่างกัน ดังรูปที่ ข.17, ข.18 และ ข.19 โดยจะสังเกตได้ว่ารูปที่ ข.17 จะมีลักษณะเดียวกับรูปที่ ข.5 และไฟล์ที่ได้จากการ Add Reference ได้แก่ Service.discomap และ Service.wsdl (สังเกตว่าไม่มี Service.disco)



รูปที่ ข.17 วิธี Add Web Reference ด้วย URL ของไฟล์ service\_gpa.php



รูปที่ ข.18 Add Web Reference -> URL : ที่อยู่ของไฟล์ service\_gpa.php -> WSDL



รูปที่ ข.19 ไฟล์ที่ได้จากการ Add Web Reference

จากนั้นทำการสร้างช่องใส่ข้อความรับรหัสนักศึกษา, ปุ่มยืนยัน และช่องข้อความเพื่อแสดง  
 เอกสาร ข้อมูลที่ได้จากเรียกใช้เว็บเซอร์วิสดังรูปที่ ข.20 รัชชศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs"
Inherits="_Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title>Untitled Page</title>
</head>
<body>
<form id="form1" runat="server">
    <div>
        <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
        <asp:Button ID="Button1" runat="server" Text="Button"
OnClick="B_Click"/>
        <br /><asp:TextBox ID="TextBox2" runat="server"></asp:TextBox>
        <asp:TextBox ID="TextBox3" runat="server"></asp:TextBox>
        <asp:TextBox ID="TextBox4" runat="server"></asp:TextBox><br />
        <asp:TextBox ID="TextBox5" runat="server"></asp:TextBox></div>
</form>
</body>
</html>

```

รูปที่ ข.20 ไฟล์ default.aspx ที่เพิ่มช่องรับข้อความ, ปุ่ม และช่องแสดงผล

เรียกใช้โอเปอเรชั่น GetGPA(decimal id) ของเซอร์วิส gpaService จาก Web Reference ที่ชื่อ WebReference บน default.aspx.cs เนื่องจากค่าที่ได้จาก TextBox เป็น String จึงต้องทำการแปลงค่าเป็น decimal ก่อน และเนื่องจาก gpaService มีการใช้ complex type (data type ที่กำหนดขึ้นเอง) ที่ชื่อ gpa จึงต้องรับค่าที่ได้กลับมาในรูปแบบ WebReference.gpa ดังรูปที่ ข.21 แล้วจึงนำค่าที่ได้ไปแสดงผลดังรูปที่ ข.22

```

...
public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void B_Click(object sender, EventArgs e)
    {
        decimal id = decimal.Parse(TextBox1.Text);
        WebReference.gpaService S = new WebReference.gpaService();
        WebReference.gpa G = S.GetGPA(id);
        TextBox2.Text = G.gender;
        TextBox3.Text = G.f_name + " " + G.l_name;
        TextBox4.Text = G.grade.ToString();
    }
}

```

รูปที่ ข.21 ไฟล์ default.aspx.cs ที่เพิ่มการเรียกใช้เซอร์วิส และการแสดงผล



รูปที่ ข.22 แสดงการเรียกใช้งานเซอร์วิส PHP ด้วยเว็บ ASP.NET

## 5 สรุปผลการทดลองพัฒนาเว็บเซอร์วิส

การทดลองพัฒนาทั้งภาษา PHP และ ASP.NET ทำให้พบว่าต่างก็ทำงานบนหลักการทำงานพื้นฐานของเว็บเซอร์วิสทั้งสิ้น กล่าวคือ มีการ request และ response ข้อมูลที่เปิดให้บริการด้วย SOAP Message ที่อยู่ในลักษณะของเอกสาร XML ที่สื่อสารผ่านเครือข่ายด้วยโปรโตคอล HTTP แต่การพัฒนาทั้งสองภาษาก็มีข้อแตกต่างกันไปตามลักษณะของแต่ละภาษา โดยที่การเรียกใช้งานข้ามแพลตฟอร์มก็สามารถทำงานได้จริง (จากหัวข้อที่ 4)

แต่ในขณะเดียวกันก็พบปัญหาบางประการ เช่น ภาษา PHP ยังสามารถเรียกใช้เซอร์วิสที่มีข้อผิดพลาดในเรื่องของ data type ใน WSDL ได้ ในขณะที่ภาษา ASP.NET จะไม่สามารถเรียกใช้ได้ เนื่องจาก ASP.NET มีความเป็นมาตรฐานมากกว่า แต่ขณะเดียวกัน PHP ก็มีความยืดหยุ่นสูงกว่าเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้