

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เว็บเซอร์วิสเพื่อการบริหารจัดการบริษัททัวร์

Web Service for Smart Tour Agency



ณัฐพล เขียวประเสริฐ

ณัฐพล เอื้อพูนวิริยะ

เลขหมู่.....  
เลขทะเบียน...62544  
วัน,เดือน,ปี...1.9.ส.ค. 2549

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# เว็บเซอร์วิสเพื่อการบริหารจัดการบริษัททัวร์

Web Service for Smart Tour Agency

จัดทำโดย

ณัฐพล เขียวประเสริฐ

ณัฐพล เอื้อพูนวิริยะ

อาจารย์ที่ปรึกษา

ดร.อรัญญา วัลย์รัชต์

ดร.สมศักดิ์ วัลย์รัชต์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2548

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เว็บไซต์ervisเพื่อการบริหารจัดการบริษัททัวร์

Web Service for Smart Tour Agency

ผู้จัดทำ

1. นายณัฐพล เขียวประเสริฐ เลขประจำตัว 45010233

2. นายณัฐพล เอื้อพูนวิริยะ เลขประจำตัว 45010245



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# เว็บเซอร์วิสเพื่อการบริหารจัดการบริษัททัวร์

นายณัฐพล เขียวประเสริฐ 45010233  
นายณัฐพล เอื้อพูนวิริยะ 45010245  
ดร.อรุณญา วลัยรัชต์ อาจารย์ที่ปรึกษา  
ดร.สมศักดิ์ วลัยรัชต์ อาจารย์ที่ปรึกษาร่วม  
ปีการศึกษา 2548

## บทคัดย่อ

ปัจจุบันอุตสาหกรรมการท่องเที่ยวในประเทศไทยนั้นเป็นอุตสาหกรรมหลักที่ส่งเสริมรายได้เข้าสู่ประเทศเป็นอย่างมาก และมีธุรกิจหลายประเภทเข้ามาเกี่ยวข้องกับอันทันได้แก่ ที่พัก การขนส่ง ร้านอาหาร สถานที่ท่องเที่ยว ซึ่งมีอยู่เป็นจำนวนมากและกระจายอยู่ตามสถานที่สำคัญ แต่บริษัทส่วนใหญ่ที่จัดแพคเกจนั้นได้ถูกกำหนดไว้อย่างแน่นอน ซึ่งอาจไม่เป็นที่พอใจของนักท่องเที่ยว เนื่องจากแพคเกจที่บริษัทได้สร้างขึ้นนั้นสถานที่ท่องเที่ยวต่างๆ ได้ถูกกำหนดไว้เรียบร้อยแล้ว โดยที่สถานที่เหล่านั้นอาจไม่ตรงตามความต้องการของนักท่องเที่ยวได้ และการที่นักท่องเที่ยวจัดแพคเกจเองทำให้เกิดความยุ่งยากในการติดต่อและหาข้อมูลทุกอย่าง

โครงการนี้จึงได้ริเริ่มพัฒนาเว็บแอปพลิเคชันที่ใช้ติดต่อกับเว็บเซอร์วิส เพื่อช่วยทำการจัดแพคเกจภายในประเทศไทยขึ้น นักท่องเที่ยวสามารถค้นหาข้อมูล ประมาณค่าใช้จ่ายเริ่มต้น และทำการเลือกและจองแพคเกจตามที่ระบบได้ประมวลผลขึ้น ทำให้นักท่องเที่ยวได้รับความสะดวกในการค้นหาข้อมูล ตลอดจนทำการส่งจองที่พัก ร้านอาหาร และสถานที่ท่องเที่ยวต่าง ๆ ภายในประเทศได้

## Web Service for Smart Tour Agency

Mr. Nuttpon Khewprasert	45010233
Mr. Nuttapol Uaepoonviriya	45010245
Dr. Arunya Walairacht	Advisor
Dr. Somsak Walairacht	Co-Advisor

Academic Year 2005

### ABSTRACT

Nowadays, tourism industry is the industry which brings a lot of money to Thailand. There are businesses that intertwined with tourism; for example, tourism agency, lodging business, mass transit systems, and restaurants. These businesses distribute themselves among the main tourist sites and tourists can sometimes arrange their own programs with these businesses. However, before tourists go on a trip, some planning need to be arranged. The planning process has frustrated tourists for quite a period of time because it is a troublesome process. Tourists would have to go through the trouble of gathering data, finding tourist attractions, making hotel reservations, etc. The solution for tourists is to buy already-setup tour packages from tourism agencies. Although, tour packages contain most of the tourist attractions, the attractions cannot cover all tourists' needs. Tourists still want to customize their own packages so that they can visit all the sites that they want to go.

The goal of this project is to help tourists set up their own packages to travel in Thailand only. This project develops web applications, which will be used to connect to web services, to help set up tour packages. Tourists can search for data, estimate the cost, and book one of the tour packages that the application prepared for them. These web applications ease the traveling preparation process.

## กิตติกรรมประกาศ

โครงการและปฏิญยานี้นี้มีอาจเกิดขึ้นได้หากขาดการสนับสนุนจากอาจารย์ที่ปรึกษา  
โครงการ อาจารย์ ดร.อรุณญา วลัยรัชต์ รวมถึงอาจารย์ ดร.สมศักดิ์ วลัยรัชต์ ที่ได้ให้คำปรึกษา  
คำแนะนำ และช่วยเหลือในเรื่องต่างๆ ในการทำโครงการนี้ และบุคคลอื่นที่ไม่ได้กล่าวถึง ณ ที่นี้ ที่  
ได้มีส่วนร่วมในการส่งเสริมและสนับสนุนการทำโครงการนี้ไม่ว่าในรูปใดๆ ก็ตาม จนทำให้  
สามารถดำเนินโครงการจนสำเร็จเสร็จสิ้นไปด้วยดี ดังนั้นข้าพเจ้าจึงขอขอบคุณทุกท่านมาไว้ ณ ที่นี้

นอกเหนือจากบุคคลข้างต้นแล้ว ข้าพเจ้าขอขอบคุณบุพการีที่ทำให้ข้าพเจ้าได้มาถึง ณ จุดนี้  
ซึ่งท่านมีส่วนสำคัญต่อข้าพเจ้าในการอุปถัมภ์เลี้ยงจนเติบโต และให้ข้าพเจ้าได้รับการศึกษา รวมถึงการ  
สนับสนุนในด้านต่างๆ ดังนั้นข้าพเจ้าจึงขอระลึกถึงพระคุณ และขอกราบขอบพระคุณท่านมาไว้ ณ  
ที่นี้

ท้ายสุดนี้ข้าพเจ้าขอขอบคุณเพื่อนร่วมงาน ที่ได้ร่วมมือกันทำงาน และแก้ไขปัญหาค่างๆ  
จนทำให้โครงการสำเร็จลุล่วงด้วยดี

นายณัฐพล เจียวประเสริฐ  
นายณัฐพล เอื้อพูนวิริยะ

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูป	VIII
บทที่ 1 บทนำ	1
1.1. บทนำ	1
1.2. วัตถุประสงค์ของโครงการ	1
1.3. ประโยชน์ที่คาดว่าจะได้รับ	1
1.4. ขอบเขตของโครงการ	1
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	2
2.1. ความเป็นมาของการพัฒนาเทคโนโลยีเว็บเซอร์วิส	2
2.1.1. Static Web	2
2.1.2. Dynamic Web	2
2.2. หลักพื้นฐานของ Web Services	3
2.2.1. ความหมายของ Web Services	3
2.2.2. เว็บเซอร์วิสกับเว็บแอปพลิเคชัน	4
2.2.3. ประโยชน์ของการทำงานร่วมกันระหว่างแอปพลิเคชันกับเว็บเซอร์วิส	5
2.2.4. การประยุกต์ใช้งาน	5
2.2.5. Web Services Model	6
2.2.6. XML (The Extensible Markup Language 1.0)	6
2.2.7. SOAP(Simple Object Access Protocol)	7
2.2.8. SOAP Intermediaries	12
2.2.9. WSDL(Web Services Description Language)	14
2.2.10. UDDI(Universal Description, Discovery and Integration protocol)	17
2.3. .NET เทคโนโลยี	18
2.3.1. บทนำ	18
2.3.2. .NET Framework	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงแก้ไข และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.3.3. ข้อดีของ .NET Framework	19
2.4. ASP.NET	20
2.4.1. ความเปลี่ยนแปลงที่เกิดขึ้นของ ASP.net	20
2.4.2. นามสกุลของไฟล์ และการแยกส่วนออกแบบกับส่วนเขียนโค้ด	20
2.4.3. การคอมไพล์โปรแกรม	21
2.4.4. คอนโทรลที่ใช้ออกแบบในแอปพลิเคชัน ASP.net	22
2.4.5. คอนโทรล HTML	22
2.4.6. คอนโทรล Web Forms	23
2.4.7. คอนโทรล Customer	23
2.4.8. คอนโทรลที่ทำงานฝั่งเซิร์ฟเวอร์	24
2.4.9. การเขียนโปรแกรมใน ASP.net	25
2.4.10. สรุปตารางเปรียบเทียบระหว่าง ASP กับ ASP.net	25
2.5. ภาษา C#	26
2.5.1. โครงสร้างพื้นฐาน	26
2.5.2. Control Statement	32
2.5.3. Loop Statement	33
2.6. MS SQL Server 2000	35
2.6.1. SQL Server 2000	35
2.6.2. TSQL	36
2.6.3. Stored Procedure	41
บทที่ 3 การออกแบบและพัฒนา	46
3.1. บทนำ	46
3.2. สถาปัตยกรรมทรีเทียร์	46
3.3. รายละเอียดของแอปพลิเคชัน	47
3.4. ขั้นตอนการจัดทำแพคเกจ	49
3.5. โครงสร้างระบบ	49
3.6. Use Case Diagram	50
3.7. Database Model Diagram	52
3.8. Flowchart	56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
บทที่ 4 การทดลองและผลการทดลอง	57
4.1. การติดต่อแสดงข้อมูลการจองผ่านทางเว็บไซต์ของ Smart Tour Agency	57
4.1.1. การติดต่อแสดงข้อมูลการจองรถโดยสาร	58
4.1.2. การติดต่อแสดงข้อมูลการจองห้องพัก	59
4.1.3. การติดต่อแสดงข้อมูลการจองร้านอาหาร	60
4.2. การติดต่อการจองผ่านทางเว็บไซต์ของ Smart Tour Agency	61
4.3. เว็บไซต์ที่เปิดให้ใช้บริการกับเว็บแอปพลิเคชัน	62
4.3.1. เว็บไซต์ที่เซิร์ฟเวอร์ได้เปิดให้ใช้บริการกับเว็บเซิร์ฟเวอร์ตัวกลาง	62
4.3.2. เว็บไซต์ที่เซิร์ฟเวอร์ได้เปิดให้ใช้บริการกับ Smart Tour Agency	64
4.4. ขั้นตอนการทำงานของระบบ	66
4.4.1. กรอกข้อมูลของผู้ใช้เข้าสู่ระบบ	66
4.4.2. เลือกประเภทของรถที่จะใช้ในการเดินทาง	67
4.4.3. เลือกโรงแรมที่จะใช้เป็นที่พักในการเดินทาง	68
4.4.4. เลือกสถานที่ท่องเที่ยวที่ต้องการในแต่ละวัน	69
4.4.5. เลือกร้านอาหารที่ต้องการในแต่ละวัน	70
4.4.6. แสดงข้อมูลทั้งหมดของผู้ใช้ระบบ	71
4.4.7. เลือกแพ็คเกจทัวร์ที่ระบบได้สร้างขึ้น	72
4.4.8. ทำการเข้าสู่ระบบ	73
4.4.9. ทำการลงทะเบียนลงสู่ระบบ	74
4.4.10. กรอกข้อมูลที่ต้องใช้ชำระเงิน	75
4.4.11. แสดงผลสรุปทั้งหมดของการจอง	75
4.4.12. แสดงข้อมูลและทำการยกเลิกการจองแพ็คเกจที่ได้จองเอาไว้ของสมาชิก	77
4.5. การแสดงสถิติด้วย Crystal Report	78
บทที่ 5 บทวิจารณ์และสรุป	80
5.1. บทสรุป	80
5.2. ปัญหาอุปสรรคและแนวทางการแก้ไข	80
5.3. แนวทางในการพัฒนาต่อ	81
บรรณานุกรม	82
ภาคผนวก ก	84

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงแก้ไข และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตารางที่	หน้า
2.1 ตารางแสดง โครงสร้างของ Web Service Model	9
2.2 ตารางแสดง โครงสร้างของ SOAP	10
2.3 ตารางอธิบาย Element ของ SOAP	12
2.4 ตารางอธิบาย Sub Element ของ SOAP	12
2.5 ตารางแสดง โครงสร้างของ WSDL	15
2.6 แสดงตารางเปรียบเทียบระหว่าง ASP กับ ASP.net	25
2.7 ตารางแสดงชื่อคลาสในภาษา C#	27
2.8 แสดง Primitive Datatype	30
2.9 แสดง Primitive Datatype อื่น ๆ ที่ใช้บ่อย	31
2.10 แสดงผล ที่ได้จากการQuery	40
2.11 แสดงวิว ที่ได้จากการQuery	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อ **VII** และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่	หน้า
2.1 เปรียบเทียบลักษณะการทำงานระหว่างเทคโนโลยี Static Web และ Dynamic Web	3
2.2 แสดง การส่ง XML โดยใช้ SOAP	8
2.3 โครงสร้างของเอกสาร SOAP	9
2.4 Web Services with SOAP, UDDI และ WSDL	17
2.5 แสดง โครงสร้างของ .NET Framework	18
2.6 แสดงเว็บแอปพลิเคชันที่สร้างด้วย ASP.net จะแยกไฟล์สำหรับออกแบบ(file.aspx) กับไฟล์ สำหรับเขียน โค้ดโปรแกรมจากกัน(file.aspx.cs)	21
2.7 คอนโทรล HTML ที่เราสามารถเลือกสร้างได้จาก Tool Box	22
2.8 คอนโทรล Web Forms ที่เราสามารถเลือกสร้างได้จาก Tool Box	23
2.9 แสดงรูปแบบการทำงานของ If. Clause	32
2.10 แสดงรูปแบบการทำงานของ If Statement	33
2.11 แสดงรูปแบบการทำงานของ While loop	34
2.12 แสดงรูปแบบการทำงานของ Do..While loop	34
2.13 แสดงการสร้าง Stored Procedure	44
2.14 แสดงการเปิดโปรแกรม Query Analyzer	44
2.15 แสดงการเปิดเลือกเปิด SQL Server	45
2.16 แสดงการเปิดเลือกเปิด Database NorthWind ใน SQL Server	45
2.17 แสดงผลจากการ Query โดยใช้ Store Procedure	45
3.1 ภาพแสดงโครงสร้างสถาปัตยกรรม 3 เทียร์ของระบบ Smart Tour Agency	47
3.2 แสดงโครงสร้างระบบ Smart Tour Agency	50
3.3 แสดง Use Case Diagram	51
3.4 แสดง STA Database Model Diagram	52
3.5 แสดง Hotel Database Model Diagram	52
3.6 แสดง Bus Database Model Diagram	53
3.7 แสดง Restaurant Database Model Diagram	53
3.8 แสดง HotelCenter Database Model Diagram	54
3.9 แสดง BusCenter Database Model Diagram	54
3.10 แสดง RestaurantCenter Database Model Diagram	55
3.11 แสดงกระบวนการการจัดสร้างแพคเกจ	56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงแก้ไข VIII และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.1 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้ระบบทำการกรอกข้อมูลเข้ามา	57
4.2 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลในการให้บริการในการจองรถที่เป็นไปได้	58
4.3 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลของรถที่เป็นไปได้ตามที่ผู้ใช้ระบบทำการกรอกข้อมูลเข้ามา	58
4.4 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลในการให้บริการในการจองโรงแรมที่เป็นไปได้	59
4.5 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลของโรงแรมที่เป็นไปได้ตามที่ผู้ใช้ระบบทำการกรอกข้อมูลเข้ามา	59
4.6 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลในการให้บริการในการจองร้านอาหารที่เป็นไปได้	60
4.7 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลของโรงแรมที่เป็นไปได้ตามที่ผู้ใช้ระบบทำการกรอกข้อมูลเข้ามา	60
4.8 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงแพ็คเกจทัวร์ที่เป็นไปได้ตามที่ผู้ใช้ระบบทำการกรอกข้อมูลเข้ามา	61
4.9 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงแพ็คเกจทัวร์ที่ได้ทำการจองไว้เรียบร้อยแล้ว	61
4.10 ภาพแสดงตัวอย่างเซอร์วิสของเซิร์ฟเวอร์โรงแรมที่เปิดให้เว็บเซอร์วิสเรียกใช้	62
4.11 ภาพแสดงตัวอย่างเซอร์วิสของเซิร์ฟเวอร์รถเดินทางที่เปิดให้เว็บเซอร์วิสเรียกใช้	62
4.12 ภาพแสดงตัวอย่างเซอร์วิสของเซิร์ฟเวอร์ร้านอาหารที่เปิดให้เว็บเซอร์วิสเรียกใช้	63
4.13 ภาพแสดงตัวอย่างเซอร์วิสของเว็บเซอร์วิสโรงแรมที่เปิดให้เว็บเซอร์วิส Smart Tour Agency เรียกใช้	64
4.14 ภาพแสดงตัวอย่างเซอร์วิสของเว็บเซอร์วิสรถเดินทางที่เปิดให้เว็บเซอร์วิส Smart Tour Agency เรียกใช้	64
4.15 ภาพแสดงตัวอย่างเซอร์วิสของเว็บเซอร์วิสร้านอาหารที่เปิดให้เว็บเซอร์วิส Smart Tour Agency เรียกใช้	65
4.16 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้กรอกข้อมูลเข้าสู่ระบบ	66
4.17 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้จะทำการเลือกชนิดของรถโดยสารที่ต้องการ	67
4.18 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้จะทำการเลือกโรงแรมที่ต้องการ	68
4.19 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้จะทำการเลือกสถานที่ท่องเที่ยวในแต่ละวัน	69
4.20 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้จะทำการเลือกร้านอาหารในแต่ละวัน	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงแก้ไข และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.21 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้จะทำการยืนยันข้อมูลที่ได้ออกข้อมูลเข้ามาทั้งหมด	71
4.22 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงแพ็คเกจให้ผู้ใช้งานเลือก	72
4.23 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ให้ผู้ใช้งานเข้าสู่ระบบ	73
4.24 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ให้ผู้ใช้งานทำการลงทะเบียนสู่ระบบ	74
4.25 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ให้ผู้ใช้งานชำระค่าใช้จ่ายผ่านบัตรเครดิต	75
4.26 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ระบบแสดงผลการจองให้กับผู้ใช้	76
4.27 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลการจองและยกเลิกการจองของผู้ใช้ระบบ	77
4.28 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงรายละเอียดการจองของผู้ใช้ระบบ	77
4.29 กราฟเส้นแสดงแนวโน้มการเข้ามาใช้บริการของผู้ใช้ระบบ	78
4.30 กราฟแท่งแสดงสถิติจำนวนของผู้ใช้ระบบที่เลือกไปสถานที่ท่องเที่ยวต่างๆ	78
4.31 กราฟวงกลมแสดงสถิติที่ผู้ใช้ระบบได้เลือกจอง	79
ก-1 แสดงรูป Book Online	86
ก-2 แสดงรูป Network Utility	86
ก-3 แสดงรูปการกำหนด Alias ใน Network Utility	87
ก-4 แสดงรูป DB-Library Options ใน Network Utility	87
ก-5 แสดงรูป Network Library ใน Network Utility	88
ก-6 แสดงรูปการ Config IIS ใน SQL Server	88
ก-7 แสดงรูป SQL Enterprise Manager	89
ก-8 แสดงรูปการ Import และ Export ข้อมูล	89
ก-9 แสดงรูป SQL Profiler	90
ก-10 แสดงรูป SQL Query Analyzer	90
ก-11 แสดงรูป SQL Server Network Utility	91
ก-12 แสดงรูป SQL Server Service Manager	91

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อ X และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 1

### บทนำ

#### 1.1. บทนำ

ปัจจุบันได้มีแนวโน้มที่มีการนำเว็บเซอร์วิสเข้ามาใช้เป็นจำนวนมาก ซึ่งเทคโนโลยีที่น่าสนใจนำมาใช้ในการพัฒนางานนั้นมีอยู่หลายอย่าง แต่ที่ได้เลือกมาใช้ก็คือ เทคโนโลยี .NET เพราะว่า ในเทคโนโลยี .NET นั้นมีอุปกรณ์หลาย ๆ อย่าง ที่เหมาะสมจะนำมาใช้ อย่างเช่น Crystal Report ที่ใช้ในการสร้างรายงาน

#### 1.2. วัตถุประสงค์ของโครงการ

ภายในโครงการนี้ได้ทำการศึกษาดังต่อไปนี้ การสร้างและนำ ASP.NET เว็บเซอร์วิสไปใช้งาน โดยมีจุดประสงค์ดังนี้

- 1) ศึกษาทำความเข้าใจใน .NET Technology
- 2) ศึกษาเรื่องเว็บเซอร์วิสและเทคโนโลยีที่เกี่ยวข้องได้แก่ SOAP, XML, WSDL UDDI เป็นต้น
- 3) ศึกษาการสร้างเว็บเซอร์วิส โดยการใช้ .NET Technology

#### 1.3. ประโยชน์ที่คาดว่าจะได้รับ

- 1) ได้รับความรู้ความเข้าใจถึงการใช้งาน .NET Technology
- 2) ได้รับความรู้ความเข้าใจเกี่ยวกับกระบวนการการติดต่อระหว่างเครื่องเซิร์ฟเวอร์ ในการทำเว็บเซอร์วิส
- 3) สามารถสร้างเว็บเซอร์วิส โดยใช้ .NET Technology ได้

#### 1.4. ขอบเขตของโครงการ

- 1) ออกแบบและสร้างเว็บแอปพลิเคชัน โดยใช้สถาปัตยกรรมแบบ 3 เทียร์ส โดยการใช้ .NET Technology
- 2) สร้างเว็บแอปพลิเคชันที่มาติดต่อให้บริการเว็บเซอร์วิส
- 3) ออกแบบระบบให้มีความสามารถในการเรียนรู้และประมวลผลข้อมูลที่ได้รับเข้ามา
- 4) สามารถสร้างรายงานที่ใช้ในการแสดงผล ซึ่งใช้ Crystal Report ที่เป็น tool ของ Visual Studio .Net
- 5) ระบบที่พัฒนาเป็นระบบที่มีการจัดแพคเกจ โดยใช้ .NET Technology ซึ่งเว็บเซอร์วิสที่ทำนั้นจะเป็นระบบส่วนกลางที่ทำการติดต่อกับระบบอื่นๆ คือ ระบบขนส่ง ร้านอาหาร ซึ่งจะนำผลลัพธ์ที่ได้มาจากเว็บเซอร์วิส มาทำการประมวลผลเป็นชุดแพคเกจให้กับนักท่องเที่ยวที่เข้ามาใช้บริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

ในบทนี้อธิบายถึงรายละเอียดความเป็นมาของเทคโนโลยีเว็บแต่เดิมก่อนที่จะพัฒนามาเป็นเทคโนโลยีเว็บเซิร์ฟวิส ความหมายของเว็บเซิร์ฟวิสและเทคโนโลยีอื่นๆ ที่เกี่ยวข้องกับเทคโนโลยีเว็บเซิร์ฟวิส

### 2.1. ความเป็นมาของการพัฒนาเทคโนโลยีเว็บเซิร์ฟวิส

เทคโนโลยีเว็บอาจแบ่งการประยุกต์และพัฒนาออกได้เป็น 3 ยุค ดังนี้

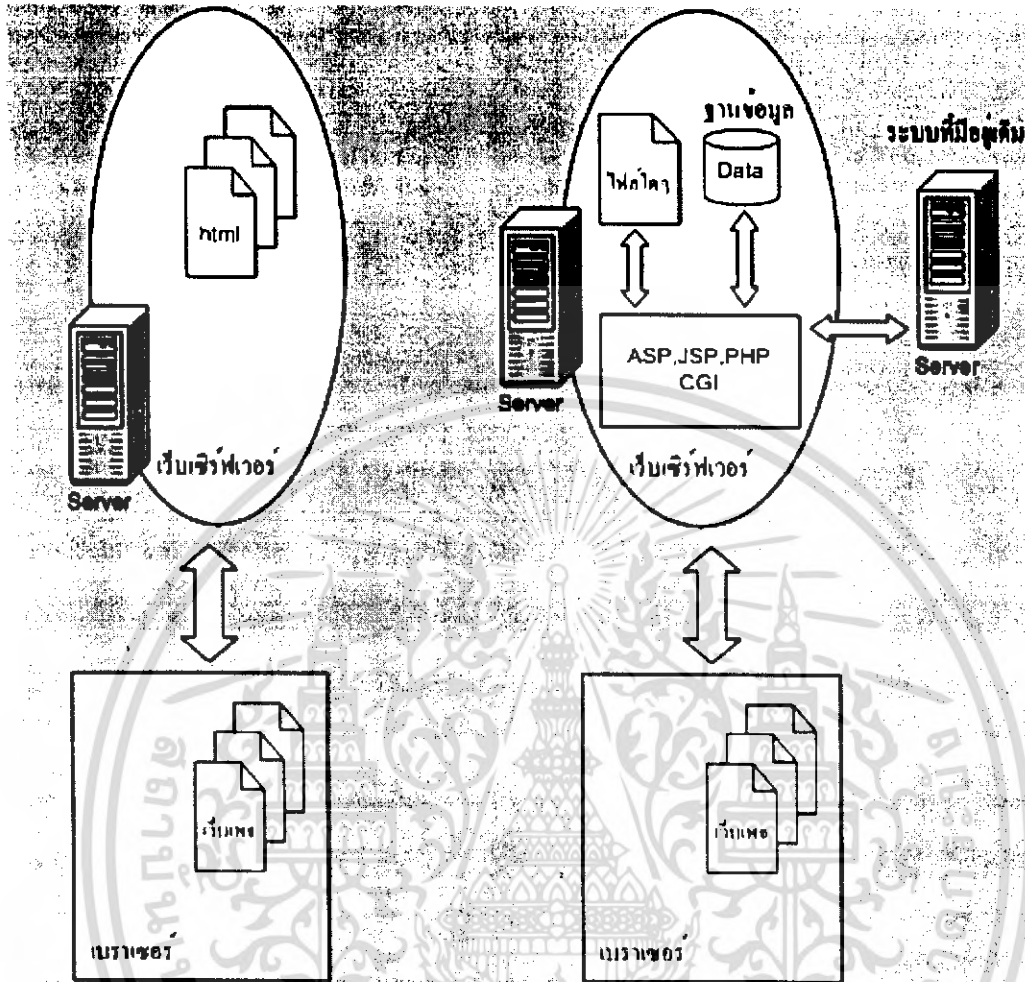
#### 2.1.1. Static Web

ยุคเริ่มต้นของเว็บนั้นเป็นยุคที่มีการใช้บราวเซอร์เรียกเว็บเพจที่สร้างขึ้นมาจากภาษา HTML ทั้งหมดหรือมีสคริปต์ทางฝั่งไคลเอนต์ ตัวอย่างเช่น JavaScript, VBScript เป็นต้น ว่าเป็นไฟล์บนเซิร์ฟเวอร์ ผู้ใช้งานเรียกดูผลการประมวลผลผ่าน HTTP โพรโตคอลในรูปแบบ HTML ซึ่งไฟล์เว็บเพจที่เป็นสคริปต์เหล่านี้ไม่สามารถติดต่อกับองค์ประกอบอื่นๆ ของทางฝั่งเซิร์ฟเวอร์ได้ เนื่องจากคุณลักษณะของเว็บเพจมีลักษณะคงที่ หรือมีการเปลี่ยนแปลงน้อยจึงเรียกว่า ยุคสแตติกเว็บ

#### 2.1.2. Dynamic Web

เนื่องด้วยรูปแบบการแสดงผลและประมวลผลแบบสแตติกเว็บนั้นตายตัว จึงมีการพัฒนาให้เว็บมีความสามารถในการติดต่อกับองค์ประกอบอื่นๆ ทางฝั่งเซิร์ฟเวอร์ โดยทำการพัฒนาโปรแกรมทางฝั่งเซิร์ฟเวอร์ทำงานตามการเรียกขอบริการของไคลเอนต์ผ่านทางโปรโตคอล HTTP และให้มีการเชื่อมโยงกับโปรแกรมเฉพาะตามที่ทำการเขียนสคริปต์ขึ้นมา หรือเขียนสคริปต์ขึ้นเพื่อใช้ความสามารถในการประมวลผลของเซิร์ฟเวอร์เพื่อทำงานบางอย่าง เช่น สร้างห้องสนทนา(chat room) กระดานตอบ-ถาม(webboard) เป็นต้น โดยมีการใช้เทคโนโลยี CGI (Common Gateway Interface) ในการทำไดนามิกเว็บ CGI คือ โปรแกรมที่ทำงานอยู่บนฝั่งเซิร์ฟเวอร์ เมื่อผู้ใช้งานเรียกใช้ CGI เมื่อใด CGI ก็จะทำงานตามหน้าที่ที่ถูกเขียนสคริปต์ขึ้นมา ซึ่งอาจมีการส่งผลลัพธ์ตอบกลับไปยังผู้ใช้ หรือไม่มีการส่งผลลัพธ์ไปยังผู้ใช้โดยขึ้นอยู่กับว่ามีการระบุหน้าที่ไว้ในสคริปต์ CGI ว่าเป็นอย่างไร ต่อมาจึงมีการพัฒนาเทคโนโลยีหลายอย่างที่มีหลักการคล้ายกับ CGI เพื่อทำงานทางฝั่งเซิร์ฟเวอร์ เช่น ASP(Active server Pages), JSP(JavaServer Pages), PHP การพัฒนาโดยใช้เทคโนโลยี CGI เรียกว่ายุคไดนามิกเว็บ

ดังนั้นความแตกต่างระหว่างสแตติกเว็บกับไดนามิกเว็บจะแตกต่างกันทั้งรูปแบบการทำงานและการประมวลผล



รูป 2.1 เปรียบเทียบลักษณะการทำงานระหว่างเทคโนโลยี Static Web และ Dynamic Web

## 2.2. หลักพื้นฐานของเว็บเซอร์วิส

### 2.2.1. ความหมายของเว็บเซอร์วิส

เว็บเซอร์วิสเป็นระบบซอฟต์แวร์ที่ออกแบบมาเพื่อสนับสนุนการทำงานระหว่างคอมพิวเตอร์กับคอมพิวเตอร์ผ่านระบบเครือข่าย โดยที่ภาษาที่ใช้ในการติดต่อสื่อสารระหว่างคอมพิวเตอร์คือภาษา XML)

การอธิบายวิธีการใช้งานของเว็บเซอร์วิสนั้นจะอธิบายโดยใช้ภาษาวิสเคิล WSDL (Web Services Description Language) ซึ่งเป็นภาษา XML ประเภทหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบอื่นๆ จะสามารถติดต่อและทำงานกับเว็บเซอร์วิส โดยใช้ โปรโตคอลที่ชื่อว่า SOAP (Simple Object Protocol) ซึ่งใช้ภาษา XML เป็นมาตรฐานในการติดต่อระหว่างระบบโดยผ่านทางโปรโตคอลอื่นที่ใช้ในการส่งข้อมูลบนเว็บ อาทิเช่น โปรโตคอล HTTP

## 2.2.2. เว็บเซอร์วิสกับเว็บแอปพลิเคชัน

เทคโนโลยีในการกระจายข่าวสารข้อมูลทางอินเทอร์เน็ตในปัจจุบันก็คือ เว็บเพจ แต่จากการที่มันมีความสามารถที่จะทำงานได้ด้วยการรวมภาษาทั้งไคลเอนท์และเซิร์ฟเวอร์ไชนด์สคริปต์ ไว้ในตัวเองเช่นภาษา VBScript, Java Script หรือ ASP, PHP, JSP นั้นทำให้เว็บเพจมีลักษณะคล้ายแอปพลิเคชันจึงถูกเรียกรวมกันว่าเว็บแอปพลิเคชัน

เว็บแอปพลิเคชัน นั้นสามารถตอบสนองความคิดคิสทริบิวต์โปรเซสซิง (Distributed Processing) ได้ในระดับหนึ่งซึ่งก็คือ การแบ่งการประมวลผลไว้ที่ฝั่งไคลเอนท์และฝั่งเซิร์ฟเวอร์และมักจะมีการใช้ดาต้าเบสควบคู่กับการทำเว็บแอปพลิเคชันไปด้วยตามความต้องการในการทำ E-Business และ E-Commerce ที่กำลังเป็นที่นิยมในปัจจุบัน และเกิดปัญหาที่ตามมาคือ เรื่องของการจ่ายเงินหรือที่เรียกว่า E-payment หรือ Payment-Gateway ซึ่งเว็บแอปพลิเคชันที่ทำ E-Commerce ต้องใช้บริการจากธนาคารออนไลน์ในการจัดเก็บเงินกับลูกค้า เพราะด้วยเทคโนโลยีนี้การใช้บริการเก็บเงินจากธนาคารออนไลน์จำเป็นที่ผู้ค้าต้องไปทำการตกลงกับธนาคารและเขียนโปรแกรมให้ตรงตามมาตรฐานที่ธนาคารออนไลน์กำหนดไว้

ด้วยปัญหายุ่งยากในการค้นหา ติดต่อและตกลงในการขอใช้บริการเก็บเงินจากธนาคารออนไลน์แนวคิดเว็บเซอร์วิสจึงดูเหมือนเป็นทางออกของปัญหานี้ ความเด่นของเทคโนโลยีเว็บเซอร์วิสก็คือ การทำระหว่างเว็บกับเว็บสามารถติดต่อสื่อสารกันได้ด้วยเอกสาร XML ที่ทั้งคนและคอมพิวเตอร์เข้าใจ และคอมพิวเตอร์ยังสามารถนำข้อมูลนั้นไปประมวลผลต่อได้ด้วยเอกสาร XML นี้เองทำให้เว็บสามารถส่งข้อมูลที่จำเป็นไปให้อีกเว็บหนึ่งทำงานบางอย่างให้หรือใช้บริการนั่นเอง ทำให้เป็นการง่ายที่จะเขียนโปรแกรมที่จะติดต่อสื่อสารหรือขอใช้บริการเก็บเงินจากธนาคารออนไลน์แต่สำหรับเว็บแอปพลิเคชันที่ใช้การส่งข้อมูลเป็น HTML ทำให้ข้อมูลนั้นไม่สามารถนำไปใช้ต่อได้ การเขียนโปรแกรมจึงยุ่งยากตามที่กล่าวด้านบน

แนวคิดของเว็บเซอร์วิส ก็คือ เว็บที่สามารถทำงานอะไรบางอย่างหรือก็คือให้บริการบางอย่างจากการร้องขอจากต่างเซิร์ฟเวอร์ ด้วยเหตุนี้ทำให้เทคโนโลยีเว็บเซอร์วิสเอื้อต่อแนว Distributed processing มากกว่าเว็บแอปพลิเคชัน และเมื่อประกอบกับการที่เว็บเซอร์วิส มี UDDI ทำให้เว็บเซอร์วิสสามารถค้นหาบริการต่างๆ ที่ต้องการได้จากทั่วทุกมุมโลก

### 2.2.3. ประโยชน์ของการทำงานร่วมกันระหว่างแอปพลิเคชันกับเว็บเซอร์วิส

เว็บเซอร์วิสช่วยให้การเข้าถึงข้อมูลสารสนเทศจากแอปพลิเคชันที่ต่างกันเป็นไปได้โดยง่าย โดยแอปพลิเคชันนั้นๆ สามารถเขียนด้วย Java และรันอยู่บน Sun Solaris Application Server หรืออาจจะเขียนด้วย C++ และรันอยู่บนวินโดวส์เอ็นที(Windows NT) หรืออาจจะเขียนด้วยเพิร์ล (Perl) และรันอยู่บนเครื่องลินุกซ์(Linux) ซึ่งมาตรฐานของเว็บเซอร์วิสทำให้อินเทอร์เฟซของแอปพลิเคชันเหล่านี้ ถูกอธิบายโดย WSDL และทำให้อยู่ในมาตรฐานของ UDDI หลังจากนั้นจึงสามารถติดต่อสื่อสารถึงกันโดย XML ผ่าน SOAP อินเทอร์เน็ต

เว็บเซอร์วิสสามารถถูกเรียกใช้ภายในองค์กรเองหรือจากภายนอกองค์กร โดยผ่านไฟร์วอลล์ ดังนั้นจึงมีองค์กรใหญ่ๆ มากมาย กำลังพัฒนาระบบที่มีอยู่ของตน ให้เข้ากับเว็บเซอร์วิสซึ่งนับเป็นการลงทุนที่คุ้มค่า เนื่องจากเว็บเซอร์วิสสามารถเพิ่มศักยภาพในการทำงานขององค์กร อีกทั้งลดค่าใช้จ่ายในการจัดการทรัพยากรขององค์กรได้อีกทางหนึ่ง

นอกจากนั้นเว็บเซอร์วิสยังสามารถใช้ร่วมกับเว็บแอปพลิเคชัน โดยส่งผ่านข้อมูลทางอินเทอร์เน็ตได้อีกด้วยซึ่งนับเป็นวิธีที่มีประสิทธิภาพในการติดต่อสื่อสารกับลูกค้าหรือหุ้นส่วน ถึงแม้จะต้องคำนึงถึงระบบรักษาความปลอดภัย และการจัดการรายการของข้อมูลอยู่ก็ตาม แต่เว็บเซอร์วิสได้ใช้มาตรฐานทั่วไปของอินเทอร์เน็ตเรื่องดังกล่าวจึงนับเป็นเรื่องธรรมดาของการสื่อสารผ่านระบบอิเล็กทรอนิกส์

### 2.2.4. การประยุกต์ใช้งาน

ขณะเดียวกัน เว็บเซอร์วิสเป็นพื้นฐานสำคัญของการบริการใหม่ๆ ในลักษณะที่ต้องการเชื่อมโยง ระบบสารสนเทศที่มีความแตกต่างเข้าด้วยกันอย่างอัตโนมัติ เช่น การพาณิชย์อิเล็กทรอนิกส์ระหว่างธุรกิจกับธุรกิจ (B2B), ระบบการบริการรัฐบาลอิเล็กทรอนิกส์ ในลักษณะที่มีการเชื่อมบริการของภาครัฐกับภาครัฐเข้าด้วยกัน เพื่อทำให้เกิดการบริการต่อประชาชนจากจุดเดียวได้ (One Stop Services) เป็นต้น

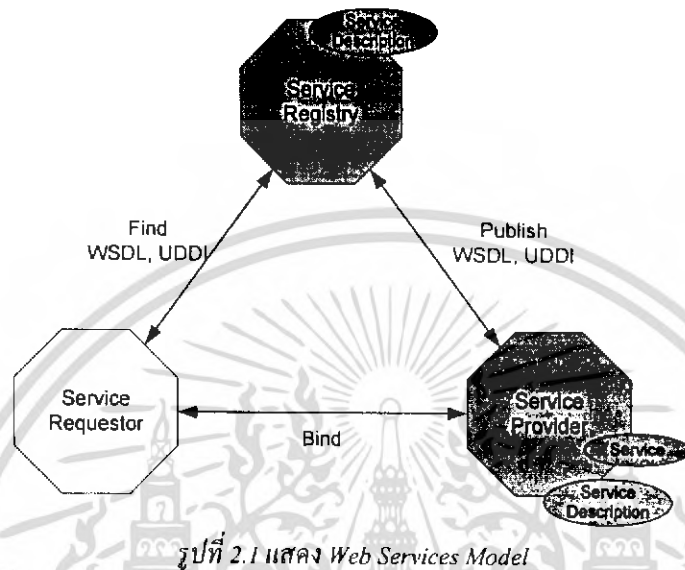
ตัวอย่างพื้นฐานหนึ่ง คือ การเชื่อมโยงระหว่างระบบการสั่งซื้อ ของบริษัทผู้ขายบริษัทหนึ่ง ไปยังระบบสารสนเทศการ จัดส่งของบริษัทจัดส่งอีกบริษัทหนึ่ง จะเป็นไปได้ง่ายมากขึ้นด้วยเทคโนโลยีของเว็บเซอร์วิส พื้นฐานดังกล่าวนี้ จะทำให้ "การจัดการห่วงโซ่อุปทานด้วยสื่ออิเล็กทรอนิกส์" (e-SCM:Supply Chain Management) และธุรกิจผ่านสื่ออิเล็กทรอนิกส์ที่มีประสิทธิภาพ และมีความคุ้มค่ามากขึ้น โดยมีการแลกเปลี่ยนข้อมูลธุรกิจ ระหว่างองค์กรในห่วงโซ่อุปทาน

ทั้งนี้แต่ละบริษัทจะมีระบบงาน (แอปพลิเคชัน) และใช้รูปแบบของข้อมูล (ฟอร์แมต คาดา) ที่แตกต่างกันได้ ซึ่งเมื่อมีการส่งข้อมูลจากองค์กรหนึ่งไปอี้องค์กรหนึ่ง ระบบคอมพิวเตอร์ของบริษัทผู้รับ สามารถนำข้อมูลที่ ได้ไปประมวลผลต่อได้ทันที การเชื่อมโยง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบอีคอมเมิร์ซของหน้าร้าน กับระบบหลังร้าน แล้วยังเชื่อมโยง กับระบบสารสนเทศของพันธมิตรทางการค้านี้ จะทำให้การทำธุรกิจผ่านสื่ออิเล็กทรอนิกส์ มีความคุ้มค่า มีประสิทธิภาพ และสร้างผลกำไรให้กับบริษัทได้ นั่นเอง

### 2.2.5. Web Services Model



<b>Requestor</b>	เป็นใครก็ตามที่ต้องการเรียกใช้บริการจาก Provider ซึ่งสามารถค้นหาบริการที่ต้องการได้จาก UDDI registry หรือ Service Registry หรือติดต่อจาก Provider โดยตรง
<b>Registry</b>	ทำหน้าที่เป็นตัวกลางให้ Provider มาลงทะเบียนไว้ โดยใช้ WSDL ไฟล์ บอกรายละเอียดของบริษัทและบริการที่มีให้ ซึ่งอาจจะใช้หรือไม่ใช้ก็ได้
<b>Provider</b>	เป็นผู้ให้บริการ มีหน้าที่ในการเปิดบริการเพื่อรองรับการขอใช้บริการจาก Requestor ที่เรียกเข้ามาขอใช้

ตาราง 2.1 ตารางแสดงโครงสร้างของ Web Service Model

### 2.2.6. XML (The Extensible Markup Language 1.0)

XML (The Extensible Markup Language 1.0) เป็นภาษา Markup ที่เป็นเท็กซ์เบส(text-based) ซึ่งทำให้เป็น มาตรฐานในการแลกเปลี่ยนข้อมูลบนอินเทอร์เน็ตอย่างรวดเร็ว ผู้ที่ทำหน้าที่รับผิดชอบ และกำหนดมาตรฐานของ XML คือ World Wide Web Consortium (W3C) ความแตกต่างระหว่าง XML กับ HTML คือ HTML ถูกนำมาใช้ในการสร้าง เว็บเพจ ที่สามารถแสดงผลได้โดยโปรแกรมบราวเซอร์ แต่ XML จะใส่แท็ก(tags) ได้อย่างอิสระ แล้วทำการส่ง XML ชุดนี้ไปประมวลผลยังแอปพลิเคชันใด ๆ ที่สามารถใช้ข้อมูลใน XML นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## XML & HTML

HTML ภาษาที่ใช้ในการเขียนเว็บมากที่สุดนั่นเป็นเพราะมีรูปแบบที่ง่ายต่อการแสดงผลของเบราว์เซอร์เนื่องจาก มีแท็กตายตัวที่สามารถบอกได้ว่าเมื่อเจอแท็กนี้จะแสดงผลอย่างไร เช่น เมื่อเจอแท็ก<B>...</B> ในเอกสารก็ให้แสดงข้อความที่อยู่ระหว่างแท็กเป็นตัวหนา แต่จะสังเกตได้ว่าคอมพิวเตอร์จะไม่เข้าใจว่าข้อความนั้นคืออะไร เพียงแต่รู้ว่าแสดงผลอย่างไร นั่นแสดงว่าไม่สามารถนำข้อมูลภายใน tag เหล่านี้ไปทำการประมวลใดๆ ได้เลย

XML เป็นภาษาที่มีลักษณะเป็น tag คล้าย HTML แต่ไม่ได้มุ่งที่การแสดงผล XML มุ่งที่การสื่อความหมายโดยอนุญาตให้ผู้ใช้สามารถกำหนดแท็กขึ้นได้เองเพื่อให้สื่อความหมายทางภาษาของมนุษย์ แต่คอมพิวเตอร์เองก็เข้าใจเช่นกัน ทำให้ข้อมูลระหว่างแท็กสามารถนำไปประมวลผลต่อได้ เช่น

```
<ComputerBook>
  <book>
    <name>เว็บเซอวิส</name>
    <price>10.00$</price>
  </book>
  <book>
    <name>xml</name>
    <price>10.00$</price>
  </book>
</ComputerBook>
```

จากตัวอย่าง จะเห็นได้ว่ารูปแบบเอกสาร XML ไม่ได้บอกวิธีแสดงผลไว้ แต่เอกสารสามารถสื่อความหมายให้คอมพิวเตอร์เข้าใจได้ และนำไปประมวลผลต่อได้ แต่ถ้าเราต้องการจะแสดงผลเราต้องใช้ควบคู่กันกับ HTML

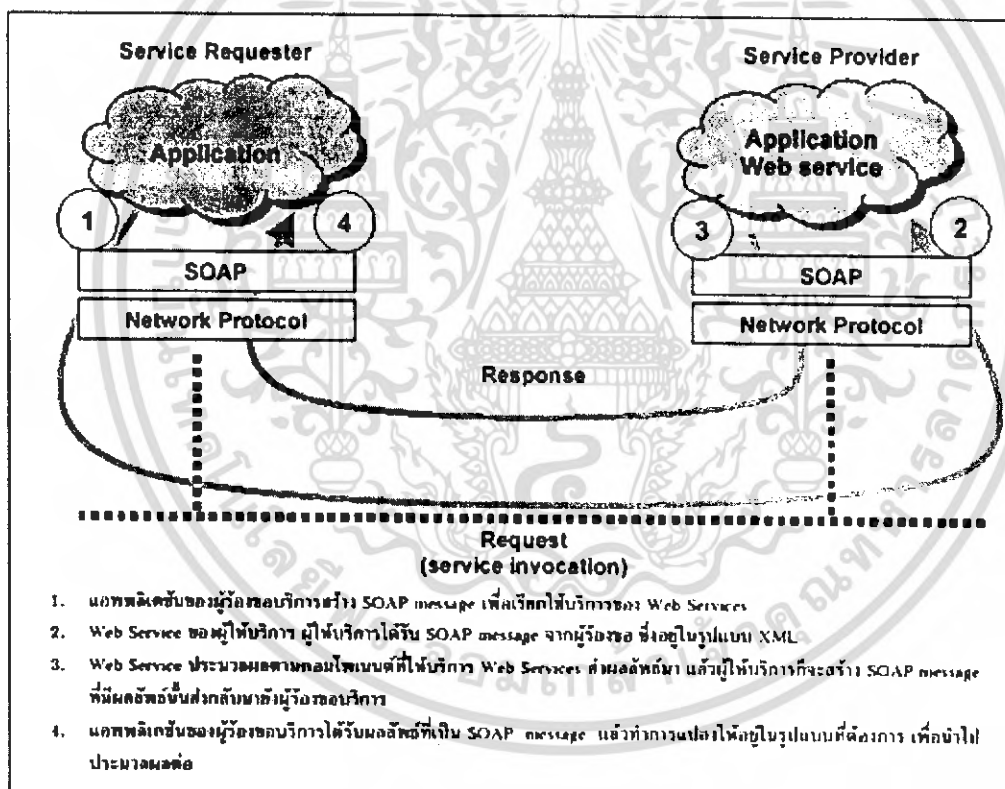
### 2.2.7. SOAP(Simple Object Access Protocol)

SOAP กลายเป็นสิ่งที่มีความสำคัญสำหรับเว็บเซอวิสอย่างรวดเร็ว เป็นโปรโตคอลที่ผู้จัดทำเว็บเซอวิสไว้เลือกใช้ที่จะส่งเมสเสจระหว่างเว็บเซอวิส ซึ่ง SOAP เป็นทรานสปอร์ตโปรโตคอลที่มี XML เป็นพื้นฐานและใช้ HTTP เป็นโปรโตคอลร่วมในการส่งผ่านเครือข่าย SOAP จะระบุวิธีในการเข้ารหัสส่วนหัว (Header Encoding) ของทั้ง HTTP และไฟล์ XML ivo อย่างชัดเจนทั้งใน ส่วนของการติดต่อไปยังคอมพิวเตอร์อีกเครื่องหนึ่งและส่งผ่านข้อมูลไปให้ รวมถึงระบุวิธีที่โปรแกรมซึ่งถูกเรียกนั้นจะส่งค่าคืนกลับมาด้วย

SOAP (Simple Object Access Protocol) เป็น XML-based โปรโตคอล (lightweight protocol)และใช้ HTTP เป็นโปรโตคอลร่วม สำหรับการแลกเปลี่ยนข้อมูลในสภาวะแวดล้อมแบบกระจายศูนย์ (decentralized, distributed environment) SOAP ได้ กำหนดเมสเสจจึงโปรโตคอล (Messaging Protocol) ระหว่างผู้ขอบริการ (requestor) กับผู้ให้บริการ (provider)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่น ผู้ขอบริการสามารถติดต่อแลกเปลี่ยนข้อมูลกับผู้ให้บริการโดยใช้ RMI (Remote Method Invocation) ตามวิธีการของ โปรแกรมแบบออปเจ็ค บริษัทไมโครซอฟท์, ไอบีเอ็ม, ล็อตัส, UserLand และ DeveloperMenter ได้ร่วมกันกำหนดมาตรฐานของ SOAP ขึ้น ซึ่งต่อมาได้มีบริษัทอีก 30 กว่าบริษัทเข้าร่วมและ จัดตั้งเป็น W3C XML Protocol Workgroup ขึ้น SOAP ได้กำหนดรูปแบบพื้นฐานของการสื่อสารแบบกระจายขึ้นโดย การพัฒนา SOA แม้ว่า SOA จะไม่ได้กำหนด Messaging Protocol ไว้ แต่ SOAP ได้ถูกกำหนด ให้เป็น Services-Oriented Architecture Protocol เรียบร้อยแล้ว เนื่องจากมันได้ถูกใช้ในการพัฒนา SOA อย่างแพร่ หลายแล้วนั่นเอง จุดเด่นของ SOAP ก็คือเป็นโปรโตคอลที่เป็นกลาง กล่าวคือ ไม่มีใครเป็นเจ้าของ และเป็นโปรโตคอล ที่ทำงานกับโปรโตคอลอื่นหลายชนิด การพัฒนาก็อนุญาตให้ทำได้ อย่างอิสระตามแพลตฟอร์มระบบปฏิบัติการ แบบจำลองทางวัตถุ (Object model) และภาษาโปรแกรมของผู้ที่ทำการพัฒนา



รูป 2.2 แสดง การส่ง XML โดยใช้ SOAP

เนื่องจากจุดประสงค์หลักของการใช้งานเว็บเซอร์วิส เราต้องการให้แอปพลิเคชัน มีการทำงานกับแอปพลิเคชันที่ทำงานอยู่ในเครื่องอื่นโดยผ่านทางเครือข่าย ซึ่งเทคโนโลยีที่มีอยู่ในปัจจุบันที่ใช้มีการสื่อสารระหว่าง Object ในระยะไกล (Remote Procedure Calls : RPC) เช่น DCOM,EJB หรือ CORBANั้นไม่ได้ถูกออกแบบมาใช้สำหรับโปรโตคอล HTTP (เครือข่ายที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัจจุบันใช้งานแพร่หลาย และสะดวกมากที่สุดได้แก่เครือข่าย Internet ซึ่งให้การสื่อสารผ่าน โพรโทคอล HTTP เป็นส่วนใหญ่)

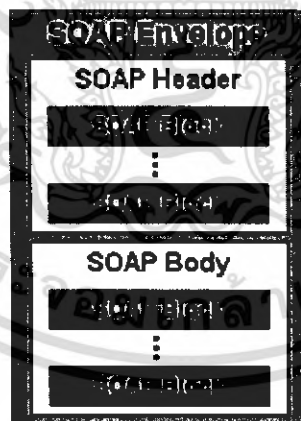
เทคนิค RPC ของเทคโนโลยีที่กล่าวข้างต้นนั้นต่างก็มีปัญหาในด้านการนำมาใช้งานในแง่ของความเข้ากันได้ของการเรียกใช้งานข้ามเทคโนโลยี เนื่องจากเป็นเทคโนโลยีเฉพาะของแต่ละค่าย ยกเว้น CORBA ผู้พัฒนาระบบจะต้องพัฒนาโปรแกรมที่มีความซับซ้อน และยังมีปัญหาในส่วนของ Firewalls และ Proxy Server ด้วยเนื่องจากโดยปกติ เซอร์เวอร์จะปิดการสื่อสารที่ไม่ใช่โพรโทคอล HTTP ออกไป เพื่อความปลอดภัยของระบบที่มีการติดต่อสื่อสารกับภายนอก

ดังนั้นทางเลือกของการสื่อสารที่จะนำมาใช้ในการทำบริการเว็บเซอร์วิส ก็คือให้ทำงานอยู่บน โพรโทคอล HTTP ไปเลย ซึ่ง SOAP นอกจากจะทำงานบนโพรโทคอล HTTP แล้วยังเป็นมาตรฐานเปิดที่จะทำให้สามารถติดต่อสื่อสารกับเครื่องคอมพิวเตอร์ที่มีความแตกต่างกันทั้งระบบปฏิบัติการ, เทคโนโลยี รวมไปถึงภาษาที่ใช้ในการพัฒนาด้วยก็ได้

### โครงสร้างของ SOAP

เอกสาร SOAP นั้นมีโครงสร้างในรูปแบบ XML ซึ่งเราสามารถแบ่งเป็นส่วนของเอกสารได้เป็น 3 ส่วนหลักดังนี้คือ

1. SOAP envelop เนื้อหาสาระ (Content) ของเอกสารทั้งหมด
2. SOAP header ส่วนเพิ่มเติมของเอกสาร SOAP ซึ่งจะมีก็ได้ หรือ ไม่มีก็ได้
3. SOAP body ส่วนที่ใช้ในการเรียกใช้งานเซอร์วิส และผลลัพธ์ที่ได้จากเซอร์วิส



รูป 2.3 โครงสร้างของเอกสาร SOAP

ตัวอย่างเอกสาร SOAP อย่างง่ายของการสอบถามราคา ดอกกุหลาบ จำนวน 100 ดอก

```
<soap:Envelope>
  <soap:Body>
    <GetPrice>
      <Item>Rose</Item>
      <Quantity>100</Quantity>
    </GetPrice>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</soap:Body>
</soap:Envelope>

```

เรามาดูอีกตัวอย่างหนึ่งของเอกสาร SOAP ที่จะแสดงในรูปแบบข้างเพื่อจะทำความเข้าใจในส่วนต่างๆ ที่มีความซับซ้อนขึ้น แต่ก็อย่าจนเกินไปที่จะทำความเข้าใจได้ดังรูปที่ 2 SOAP Envelope ต่อไปนี้

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"

  soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
  <soap:Header>
    <!-- ข้อมูลในส่วนของ Header -->
    <i:local xmlns:i="http://www.i3t.or.th/ws/">
      <i:currency>Bath</i:currency>
    </i:local>
  </soap:Header>
  <soap:Body>
    <!-- ข้อมูลในส่วนของ Body -->
    <GetPrice>
      <Item>Rose</Item>
      <Quantity>100</Quantity>
    </GetPrice>
  </soap:Body>
  <soap:Fault>
    <!-- ข้อมูลของ SOAP ในกรณีข้อผิดพลาด จาก SOAP Node -->
  </soap:Fault>
</soap:Envelope>

```

Element	Definition
<Envelope>	<p>จะเป็น Root ของเอกสาร XML SOAP เสมอ โดยจากตัวอย่างจะใช้ Namespaces จาก "http://schemas.xmlsoap.org/soap/envelope/" (Schema for the SOAP/1.1 envelope)</p> <p>ดังนั้น Element ต่างๆ ในตัวอย่างนี้ที่อยู่ในเอกสาร XML ส่วนที่เป็นมาตรฐานของ SOAP เราจะมี Namespaces ที่เราตั้งเป็น soap (&lt;soap:Envelop&gt;, &lt;soap:Header&gt;, &lt;soap:Body&gt;, &lt;soap:Fault&gt;)</p> <p>Attribute ใน Envelop</p> <p>Attribute encodingStyle ในตัวอย่างมีการอ้างการ encoding จาก "http://schemas.xmlsoap.org/soap/encoding/" (Schema for the SOAP/1.1 encoding)</p> <p>จะใช้ในกรณีที่เรามีการใช้ Parameter หลายชนิดข้อมูลในเอกสาร (Data Type) เช่น Boolean, String, Integer เป็นต้น</p>
<Header>	จากตัวอย่างมีการเพิ่มเนื้อหาของเอกสาร SOAP เข้าไปในส่วนของ Header ในที่นี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	<p>สกุลเงิน (Currency) ซึ่งส่วนนี้จะใช้ในส่วนของ Application ไม่ได้เป็นส่วนของมาตรฐานของ SOAP แต่ผู้ใช้งานเป็นผู้กำหนดใช้เอง (User-defined)</p> <p><b>Attribute ใน Header</b></p> <p>ใน &lt;Header&gt; Element อาจจะมีการใส่ Attribute mustUnderstand เพื่อให้ฝั่งที่รับเอกสาร SOAP ให้ทำการประมวลผลในส่วน Header ด้วย โดยกำหนดค่าเป็น Boolean เช่น</p>
	<pre>&lt;i:local xmlns:i="http://www.i3t.or.th/ws/"&gt;   &lt;i:country mustUnderstand="1"&gt;Thailand&lt;/i:country&gt;   &lt;i:currency mustUnderstand="0"&gt;Bath&lt;/i:currency&gt; &lt;/i:local&gt;</pre>
	<p>ค่าของ mustUnderstand ถ้าเป็น "0" แสดงว่าไม่ต้องประมวลผล ถ้าเป็น "1" : true คือจำเป็นต้องมีการประมวลผล (โดยค่า default = "0" : fault)</p>
<p>&lt;Body&gt;</p>	<p>โดยปกติแล้วเอกสาร SOAP จะต้องมีส่วนของ Body เพราะเป็นส่วนเป็นเนื้อหาของสาระจริงๆ ของ SOAP จากตัวอย่างเราต้องการสอบถามราคาของดอกกุหลาบจำนวน 100 ดอก ซึ่ง &lt;GetPrice&gt;, &lt;Item&gt; และ &lt;Quantity&gt; เป็น Element ที่ใช้งานใน Application ไม่ใช่มาตรฐานของ SOAP</p>
<p>&lt;Fault&gt;</p>	<p>เป็นส่วนที่จะถูกใช้เมื่อมีข้อผิดพลาดในการประมวลผลของเอกสาร SOAP ซึ่งโดยปกติจะเห็นเฉพาะที่เป็นเอกสารตอบกลับเท่านั้น (Reply Message หรือ Answer Message) ซึ่งตัวอย่างจะเป็นเอกสารที่เป็นการร้องขอบริการ (Request Message) ซึ่งไม่น่าจะมีส่วนของ &lt;Fault&gt; Element แต่ที่เขียนไว้เพื่อให้เห็นภาพรวมของโครงสร้างเอกสาร SOAP ทั้งหมดก่อน</p> <p>ตัวอย่างข้อมูลใน &lt;Fault&gt; Element ที่มีการเกิดข้อผิดพลาด</p>

ตาราง 2.2 ตารางแสดงโครงสร้างของ SOAP

```
<soap:Fault>
  <faultcode>Client</faultcode>
  <faultstring>Invalid Request</faultstring>
</soap:Fault>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางอธิบาย Fault Element

Sub Element	คำอธิบาย
<faultcode>	ชื่อของการเกิดข้อผิดพลาด (ดูตารางด้านล่าง)
<faultstring>	คำอธิบายของข้อผิดพลาด
<faultactor>	ชื่อของแหล่งที่เกิดข้อผิดพลาด
<detail>	ข้อมูลรายละเอียดเพิ่มเติมของข้อผิดพลาดนั้น

ตาราง 2.3 ตารางอธิบาย Element ของ SOAP

### ตารางอธิบาย faultcode Sub Element

Error (faultcode)	คำอธิบาย
VersionMismatch	มีข้อผิดพลาดของการอ้าง Namespace ของ Element
MustUnderstand	ในการประมวลผลของ Sub Element ใน Header ที่มีการกำหนด attribute mustUnderstand เป็น "1" ไม่สามารถประมวลผลได้ (not understood)
Client	เอกสาร SOAP มีข้อผิดพลาดอันเนื่องมาจากมีโครงสร้างทางผิด หรือมีข้อมูลที่ไม่ถูกต้อง
Server	มีข้อผิดพลาดจาก Server ไม่สามารถประมวลผลเอกสารได้

ตาราง 2.4 ตารางอธิบาย Sub Element ของ SOAP

ท่านจะเห็นว่า SOAP Envelope ก็เป็นเอกสารที่อยู่ในรูปแบบ XML ทั่วไป แต่สิ่งที่ทำให้ SOAP มีความสามารถมากขึ้นก็คือ SOAP ได้มีการกำหนดโครงสร้างของเอกสารเป็นส่วนๆ โดยอธิบายว่าส่วนใดมีหน้าที่อะไร (What) มีข้อมูลอะไรอยู่ในส่วนนั้น และใคร (Who) คือผู้ที่จะต้องสนใจในส่วนนั้น ๆ และกล่าวถึงว่าส่วนใดจำเป็นที่ต้องมีในเอกสาร และส่วนใดเป็นส่วนที่เพิ่มเติมอาจจะมีหรือไม่มีส่วนนั้นก็ได้ขึ้นอยู่กับความจำเป็นในการใช้งาน ทำให้การพัฒนาโปรแกรมมีทิศทางที่ค่อนข้างชัดเจน แต่สามารถเพิ่มขยายได้ในอนาคตอีกด้วย

#### 2.2.8. SOAP Intermediaries

จากตัวอย่างเอกสาร SOAP นั้นจะเห็นได้ว่าในส่วนของ <Header> Element จะเป็นส่วนที่เราจะขยายสารสนเทศ (information) ใหม่ๆ เข้าไปใน SOAP ได้ซึ่งตัวอย่างเรามีการอธิบายให้ผู้รับเอกสารตัวอย่างนี้ให้ทราบว่าในการสอบถามราคานี้ให้ใช้เงินในสกุลเงินบาท เป็นต้น ซึ่งเราจะเรียกการขยายในส่วนนี้ว่าเป็นการขยายในแนวตั้ง (Vertical extensibility) แต่ถ้าเราจะมาดูในด้านการขยายในแนวราบ (Horizontal extensibility) ของ SOAP นั่นก็คือทำอย่างไรที่จะให้มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การส่งเอกสาร SOAP ไปยังปลายทางโดยเส้นทางของการเดินทางของเอกสารอาจจะไม่เป็นลักษณะแบบจุดต่อจุด (Point-to-point) แต่อาจจะเป็นการส่งเอกสารตามเส้นทางโดยเอกสารยังคงเป็นเอกสารตัวเดิมดังรูป

### ความจำเป็นในการใช้ SOAP Intermediary

SOAP Intermediaries จะทำหน้าที่เป็นตัวผ่าน หรือตัวกลางในการส่งเอกสาร SOAP จากต้นทาง ในที่นี้คือ Requestor ไปยัง Provider เพื่อสร้างกระบวนการบางอย่างกับเอกสาร ซึ่งในการสร้างเส้นทางในการส่งแบบนี้อาจมีจุดประสงค์ดังต่อไปนี้คือ

1. Intermediary ทำหน้าที่เป็น Crossing trust domains เนื่องจากการทำงานของ Web Services เป็นการเรียกใช้งานแบบระยะไกล หรือมีการทำงานแบบระบบกระจาย (Distributed System) ดังนั้นวิธีการในด้านความปลอดภัยจึงเป็นส่วนหนึ่งที่ต้องคำนึงถึง ซึ่งวิธีการหนึ่งก็คือให้มีการส่งเอกสารมายังจุดหนึ่งที่มีความน่าเชื่อถือได้ก่อน แล้วส่งต่อไปยังปลายทางอีกที่หนึ่ง ซึ่งหลักการใช้ Trust Domains เราอาจจะดูตัวอย่างง่ายๆ เช่น ในองค์กรเราอาจจะมีเครือข่าย Intranet อยู่ภายในองค์กร เครื่องคอมพิวเตอร์แต่ละเครื่องอาจจะสามารถติดต่อไปยัง Server ได้โดยตรง แต่ถ้าต้องการออกไปยัง Internet หรือ Extranet ที่อยู่นอกองค์กรจะต้องผ่าน Trust Domains ก่อน ในที่นี้อาจจะเป็น Firewalls หรือ Virtual Private Network (VPN) gateways เพื่อความปลอดภัย

2. Intermediary ทำหน้าที่เป็นตัวขยายการทำงานแบบระบบกระจาย (Distributed System) ถ้าเรามองดูเฉพาะแง่ของการขอใช้บริการจาก Requestor ไปยังผู้ให้บริการ Provider โดยผู้ให้บริการจะมีการกระบวนการทำงานบางอย่างภายในและตอบกลับไปยังผู้ขอใช้ ซึ่งการทำงานแบบนี้ก็อยู่บนพื้นฐานของ Request and Respond นั้นเอง แต่ถ้าระบบของเรามีการขยายใหญ่ขึ้นสิ่งที่เราจะต้องดูเพิ่มขึ้นก็คือความสามารถในการรองรับงานของผู้ให้บริการด้วย ดังนั้นก่อนที่จะมีการส่งเอกสาร SOAP ไปยังเซิร์ฟเวอร์ ของผู้ให้บริการนั้นอาจจะมีการผ่าน Intermediary Server ก่อนเพื่อทำหน้าที่เป็น Buffer รับเอกสาร SOAP มาพักไว้และส่งเอกสารให้ผู้ให้บริการ (Provider) อีกทอดหนึ่ง การทำงานเช่นนี้จะช่วยให้ Server ของ ผู้ให้บริการไม่รับภาระงานมากจนเกินไปในช่วงเวลาใดเวลาหนึ่งจนอาจทำให้เกิดความเสียหายได้ และ Intermediary อาจจะมีการรวบรวมเอกสารแล้วส่งมาให้ผู้ให้บริการเป็นลักษณะ Batch เพื่อเพิ่ม Performance การใช้งาน Intermediary ลักษณะนี้ผู้ขอใช้บริการอาจจะไม่เห็น Intermediary แต่จะมองเห็นว่าติดต่อกับผู้ให้บริการ โดยตรง

3. การใช้งาน Intermediary ในลักษณะ Value-added ให้กับระบบ เช่น ใช้ Intermediary ในการเข้ารหัสข้อมูล (Encrypts) และเซ็นลายเซ็นอิเล็กทรอนิกส์ (Digital signs) ในการส่ง และรับ หรืออีกตัวอย่างหนึ่งเช่นต้องการเก็บข้อมูลเพิ่มเติมไปวิเคราะห์การทำงานของระบบโดยการรับ-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่งเอกสารจะผ่าน Intermediary แต่ไม่ได้ทำการประมวลผลเอกสารแต่จะเก็บของข้อมูล บางอย่างไว้เท่านั้นเช่นนำไปวิเคราะห์ค่า QoS (Quality of Service) หรือ Bottleneck ของระบบ เป็นต้น

### Intermediaries ใน SOAP

ในหัวข้อก่อนหน้านี้ได้อธิบายหลักการของการใช้ Intermediaries ไปแล้ว ดังนั้นในส่วนนี้จะกล่าวถึงว่าจะทราบได้อย่างไรว่าเอกสาร SOAP นั้นใครจะเป็นผู้ประมวลผล (ซึ่งจะอยู่ในส่วนของ <Body> Element ของเอกสาร)

ผังรูปที่แสดงการทำงานของ Intermediaries จะเห็นว่าเอกสาร SOAP จะถูกส่งจากผู้ขอใช้บริการผ่าน Intermediaries เป็นไปจนถึงปลายทางคือผู้ให้บริการ ดังนั้นเอกสาร SOAP ก็ถูกเปิดเผยทั้งหมดกับ Intermediaries ด้วย แต่ส่วนที่ Intermediaries สนใจจะอยู่เฉพาะส่วนที่เป็น Header เท่านั้น โดยใน Header จะมี Attribute หนึ่งเพื่อใช้ในการระบุว่าใครที่ควรจะทำงานกับ Header นี้ ซึ่ง Attribute นั้นก็คือ actor โดยค่าที่ actor จะต้องใส่ก็คือ URI (Universal Resource Indicator) ของ Intermediary นั้นเอง ดังตัวอย่าง

```
<soap:Header>
  <i:Authentication
    xmlns:i="http://www.i3t.or.th/ws/security"
    soap:actor="urn:I3T:SecurityGateWay"
    soap:mustUnderstand="1">
    <i:username>ACompany</i:username>
    <i:password>SOAPpath</i:password>
  </i:Authentication>
</soap:Header>
```

ดังนั้นเมื่อ Intermediary รับเอกสาร SOAP โดยถ้ามีชื่อ attribute actor ตรงกับของตัวเอง Intermediary จะเปลี่ยนเนื้อหาใน Header ที่ได้เพื่อส่งต่อไปยัง Intermediary อื่นหรือไปยังผู้ให้บริการจุดประสงค์หลักที่มีการออกแบบการทำงานแบบนี้ เพื่อลดความซับซ้อนของเอกสารลง นอกจากนี้แล้วยังมี URI ของ attribute actor ชนิดพิเศษอีกหนึ่งค่าคือ "http://schemas.xmlsoap.org/soap/actor/next" เพื่อบอกว่าผู้รับเอกสาร SOAP ในจุดต่อไปจะเป็นผู้ที่จะประมวลผลของเอกสาร ซึ่งกระบวนการนี้จะมีประโยชน์อย่างมากในการส่งเอกสาร SOAP เป็นทอดๆ (hop-by-hop)

### 2.2.9. WSDL(Web Services Description Language)

WSDL (Web Services Description Language) เป็นภาษาที่ใช้อธิบายคุณลักษณะการใช้บริการของ Web Services และวิธีการติดต่อกับ Web Services ความต้องการของนิยามนี้ เกี่ยวเนื่องกับความต้องการของ distributed system ที่จะกำหนด Interface Definition Language(IDL) โดยใช้ภาษา XML, WSDL เกิดจากการรวมแนวคิดของ NASSL (The

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Network Accessible Service Specification Language), WDS (Well-Defined Services) ของบริษัทไอบีเอ็ม, SDL (The Service Description Language) และ SCL (the SOAP Contract Language) ของบริษัทไมโครซอฟท์ ปัจจุบัน WSDL เป็นภาษา ที่ใช้ในการดูแลของ W3C (World Wide Web Consortium) ซึ่งยังไม่เป็นมาตรฐานที่สมบูรณ์ เวอร์ชันที่ใช้งานอยู่ในปัจจุบันคือ WSDL 1.1 (รายละเอียดเพิ่มเติมเกี่ยวกับ WSDL สามารถศึกษาเพิ่มเติมได้ที่ <http://www.w3c.org/TR/wsdl>)

WSDL คือ มาตรฐานสำหรับการประกาศโปรเซสที่จำเป็นในการเรียกใช้เซอร์วิส SOAP (Simple Object Access Protocol)

### โครงสร้างเอกสาร WSDL

WSDL เป็นภาษาที่อยู่ในความดูแลขององค์กร W3C (World Wide Web Consortium) version ที่มีอยู่ในปัจจุบัน คือ WSDL 1.1 ในการใช้งานจริง หากเราสร้างบริการเว็บเซอร์วิสก็จะมีเครื่องมือช่วยสร้างเอกสาร WSDL สำหรับเว็บเซอร์วิสอย่างอัตโนมัติ จุดภายในเอกสารที่เราควรรู้เกี่ยวกับการติดต่อและเรียกใช้บริการของเว็บเซอร์วิส มีจุดที่ควรรู้ ดังนี้

Element	Definition
<portType>	เป็นส่วนที่สำคัญที่สุดใน WSDL element อธิบาย operations ที่ web service มีให้บริการและ messages ที่เกี่ยวข้อง เทียบได้กับ function library หรือ module หรือ class ในการเขียนโปรแกรม
<operation>	อธิบาย method ที่ให้บริการ Web Services หนึ่งจะมี method จำนวนกี่ method ก็ได้
<message>	อธิบาย data elements ของ operation แต่ละ message อาจมีมากกว่าหนึ่งส่วนเทียบได้กับ parameter ของ function ในการเขียน โปรแกรม
<types>	อธิบายชนิดข้อมูลที่ web service ใช้ เพื่อความเป็นกลาง WSDL ใช้ XML Schema syntax ในการระบุชนิดข้อมูล
<binding>	อธิบาย format ของ message และ protocol details ในแต่ละ port
<service>	สำหรับ web server จะมี Web Services จำนวนกี่บริการก็ได้ และ ชื่อ Web Services ก็เป็นตัวจำแนกและบ่งบอกแต่ละบริการซึ่งห้ามมีชื่อซ้ำกัน

ตาราง 2.5 ตารางแสดงโครงสร้างของ WSDL

ตามทฤษฎีแล้ว ไฟล์เอกสาร WSDL แต่ละไฟล์ สามารถอธิบายคุณลักษณะของบริการเว็บเซอร์วิสได้มากกว่า 1 บริการ โดยแต่ละเว็บเซอร์วิสจะมีพอร์ตสื่อสารเฉพาะตัว ซึ่งบ่งบอกไว้ในเอกสาร WSDL อยู่แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## WSDL Example

```

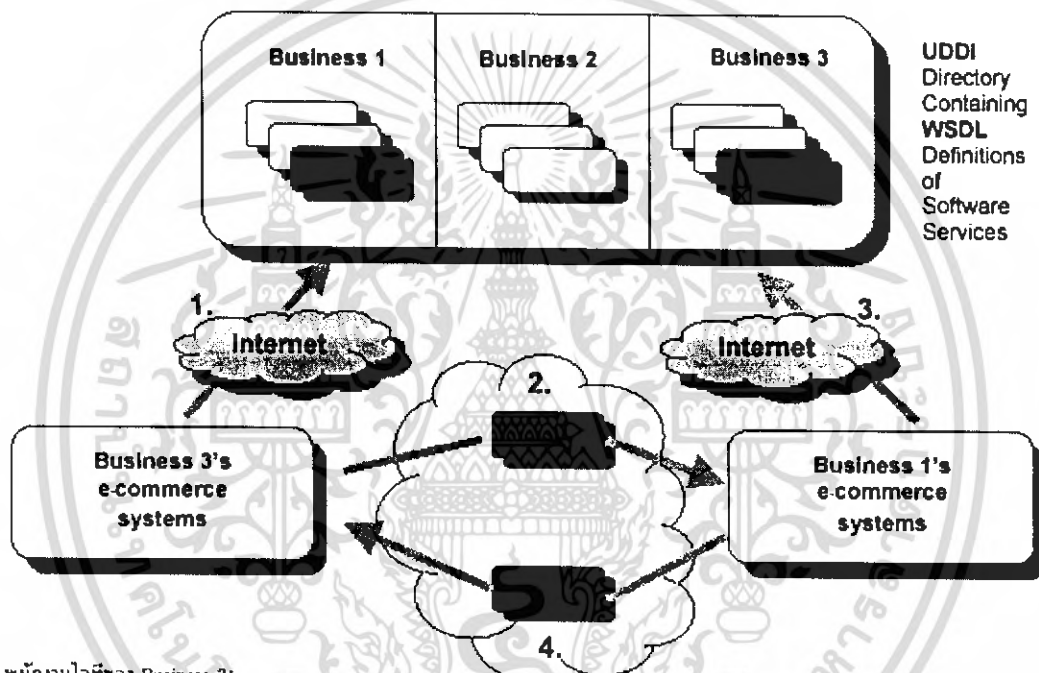
<wsdl:definitions name="nmtoken"? targetNamespace="uri">
  <import namespace="uri" location="uri"/> *
  <wsdl:documentation .... /> ?
  <wsdl:types> ?
    <wsdl:documentation .... /> ?
    <xsd:schema .... /> *
  </wsdl:types>
  <wsdl:message name="ncname"> *
    <wsdl:documentation .... /> ?
    <part name="ncname" element="qname"? type="qname"?/> *
  </wsdl:message>
  <wsdl:portType name="ncname"> *
    <wsdl:documentation .... /> ?
    <wsdl:operation name="ncname"> *
      <wsdl:documentation .... /> ?
      <wsdl:input message="qname"> ?
        <wsdl:documentation .... /> ?
      </wsdl:input>
      <wsdl:output message="qname"> ?
        <wsdl:documentation .... /> ?
      </wsdl:output>
      <wsdl:fault name="ncname" message="qname"> *
        <wsdl:documentation .... /> ?
      </wsdl:fault>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:serviceType name="ncname"> *
    <wsdl:portType name="qname"/> +
  </wsdl:serviceType>
  <wsdl:binding name="ncname" type="qname"> *
    <wsdl:documentation .... /> ?
    <!-- binding details --> *
    <wsdl:operation name="ncname"> *
      <wsdl:documentation .... /> ?
      <!-- binding details --> *
      <wsdl:input> ?
        <wsdl:documentation .... /> ?
        <!-- binding details -->
      </wsdl:input>
      <wsdl:output> ?
        <wsdl:documentation .... /> ?
        <!-- binding details --> *
      </wsdl:output>
      <wsdl:fault name="ncname"> *
        <wsdl:documentation .... /> ?
        <!-- binding details --> *
      </wsdl:fault>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="ncname" serviceType="qname"> *
    <wsdl:documentation .... /> ?
    <wsdl:port name="ncname" binding="qname"> *
      <wsdl:documentation .... /> ?
      <!-- address details -->
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.10. UDDI(Universal Description, Discovery and Integration protocol)

UDDI (Universal Description, Discovery, and Integration) เป็นมาตรฐานที่ให้ชุดพื้นฐาน APIs (Application Programming Interface) ของ SOAP ที่สามารถนำมาใช้ในการพัฒนาเป็นตัวแทนของผู้ให้บริการ (Service broker) UDDI ใช้สำหรับค้นหาเซอร์วิสที่ต้องการและเมื่อได้มาแล้ว UDDI ยังจัดหาข้อตกลงในวิธีการที่จะใช้งานเปรียบได้กับสมุดหน้าเหลือง เป็นมาตรฐานที่จัดตั้งขึ้นโดยบริษัทไอบีเอ็ม บริษัทไมโครซอฟต์ และบริษัทอริบา (Ariba) ปัจจุบันมีบริษัทที่ร่วมกันกำหนดมาตรฐานของ UDDI มากกว่า 70 บริษัท ซึ่งมาตรฐานของ UDDI ถูกกำหนดให้เป็นมาตรฐานสำหรับ B2B interoperability (รายละเอียดเพิ่มเติมเกี่ยวกับ UDDI สามารถศึกษาเพิ่มเติมได้ที่ <http://uddi.org>)



พนักงานไอทีของ Business 3:

1. ค้นหาข้อกำหนดการให้บริการใน WSDL ผ่านอินเทอร์เน็ตของ Business 1
2. ติดต่อผ่านอินเทอร์เน็ตของแอปพลิเคชันที่ให้บริการ (ด้วย SOAP) ผ่านทางอินเทอร์เน็ต
- 3,4. Business 1 รวมหรือบูรณาการกับ Business 3 ด้วยวิธีเดียวกัน

รูป 2.4 Web Services with SOAP, UDDI และ WSDL

## 2.3. .NET เทคโนโลยี

### 2.3.1. ความหมายของ .NET

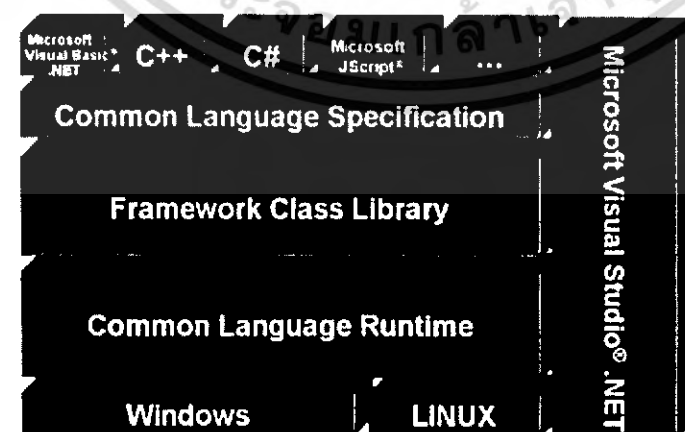
.NET (อ่านว่า ดอทเน็ต) คือแนวคิดหนึ่งที่บริษัทไมโครซอฟท์ภูมิใจนำเสนอ โดย .NET ตัวนี้ไม่ได้เกี่ยวข้องกับโดเมนเนมของเว็บใด ๆ ทั้งสิ้น แต่ .NET ตัวนี้ หมายถึง การนำเอาอุปกรณ์ทุกอย่างบนโลกมาเชื่อมโยงต่อกันเหมือนตาข่าย (net = ตาข่าย) ซึ่งหากว่าทำสำเร็จแล้วไม่ต้องนึกเลยว่าไมโครซอฟท์จะได้เป็นเจ้าของเทคโนโลยีโดยไม่ต้องสงสัย แต่เรื่องมันไม่ง่ายอย่างที่คิด เพราะอุปกรณ์ต่างๆเหล่านั้นล้วนถูกออกแบบมาแตกต่างกัน การที่มันจะติดต่อสื่อสารกันรู้เรื่องนั้น ย่อมเป็นเรื่องที่เป็นไปได้ยาก บริษัทไมโครซอฟท์เล็งเห็นจุดนี้ จึงได้พยายามที่จะคิดค้นสิ่ง ที่เป็นมาตรฐานขึ้น เพื่อให้อุปกรณ์ทุกชนิดทั่วโลกติดต่อสื่อสารกันได้อย่างรู้เรื่อง จึงไม่ใช่เรื่องแปลกถ้าในอนาคตเราเปิดเว็บไซต์เล่นอินเตอร์เน็ตด้วยอุปกรณ์อื่นๆนอกเหนือจากคอมพิวเตอร์

### 2.3.2. .NET Framework

จากที่กล่าวข้างต้นบริษัทไมโครซอฟท์ต้องการที่จะสร้างอะไรที่เป็นมาตรฐานขึ้นมา เพื่อให้ทุกสิ่งทุกอย่าง เพื่อให้ทุกสิ่งทุกอย่างสามารถติดต่อสื่อสารกันได้หมด โดยคิดค้นระบบ ซึ่งหมายถึงมันป็นมือว่าจะให้เป็นระบบมาตรฐาน ระบบนี้คือ .NET Framework ซึ่งระบบนี้ไม่ใช่ระบบปฏิบัติการ แต่เปรียบเสมือนโปรแกรมหนึ่งที่จะสามารถสร้างสภาวะแวดล้อมหนึ่ง ซึ่งสามารถทำงานในระบบ .NET นี้ได้

ในอนาคตบริษัทไมโครซอฟท์ก็มีแผนที่จะนำเอาระบบนี้ไปติดตั้งลงบนอุปกรณ์ทุกชนิด เพื่อให้ทุกสิ่งทุกอย่างมีระบบๆหนึ่งที่เหมือนกันหมด โดย .NET Framework นั้นมีส่วนประกอบภายในแบ่งออกเป็น 3 ชั้นใหญ่ๆคือ

#### Microsoft .NET Framework Architecture



รูป 2.5 แสดง โครงสร้างของ .NET Framework

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1) Programming Language: เป็นรูปแบบของภาษาที่ออกแบบมาเพื่อให้สามารถทำงานในสถานะที่เป็น .NET ได้โดยที่ทางบริษัทไมโครซอฟท์ได้เปิดตัวภาษาหลักๆที่จะใช้พัฒนามบน .NET นี้ 3 ภาษา
  - C# เป็นภาษาใหม่ที่บริษัทไมโครซอฟท์พัฒนามาจาก C++ กับ JAVA เป็นหลัก
  - VB.NET เป็นภาษาที่พัฒนามาจาก Visual Basic ในเวอร์ชัน 6.0
  - JScript.net เป็นภาษาที่พัฒนามาจาก JScript ซึ่งเป็น JavaScript ในเวอร์ชันของบริษัทไมโครซอฟท์
- 2) Base Classes Library: Library นั้นเปรียบเสมือนชุดคำสั่งสำเร็จรูปย่อยๆที่เพิ่มเข้ามา ซึ่งส่วนใหญ่จะเป็นชุดคำสั่งที่ต้องใช้งานอยู่เป็นประจำ ดังนั้นจึงมีผู้คิดค้นเพื่ออำนวยความสะดวกในการเขียนโปรแกรม ซึ่ง Library ในภาษาต่างๆส่วนใหญ่จะอยู่ในรูปแบบไฟล์ include แต่ถ้าเป็น ASP สิ่งที่เป็น library ก็คือ คอมโพเนนต์ต่างๆนั่นเอง ซึ่งภายในระบบ .NET จะสร้างสิ่งที่เรียกว่าเป็น Library พื้นฐานขึ้น ทำให้ไม่ว่าจะใช้ภาษาใดในการพัฒนาโปรแกรมก็สามารถที่จะเรียกใช้ Library ที่เป็นตัวเดียวกันได้หมด
- 3) Common Language Runtime (CLR): นับเป็นสิ่งสำคัญแทบจะที่สุดของระบบ .NET นี้ก็ว่าได้ เพราะ CLR ที่ว่านี้มีหน้าที่ทำให้โปรแกรมที่เขียนขึ้นมาด้วยภาษาต่างๆกัน กลายเป็นภาษารูปแบบมาตรฐานเดียวกัน ทั้งหมด เราเรียกภาษาที่ว่านี้ว่า Intermediate language (IL) ซึ่งเมื่อต้องการที่จะรันโปรแกรมใด CLR ที่ว่านี้จะตรวจสอบเครื่องที่รันว่ามีสถานะแวดล้อมการทำงานเช่นใดหลังจากนั้นก็จะคอมไพล์เป็นโปรแกรมที่เหมาะสมต่อการทำงานของเครื่องนั้น ทำให้เราสามารถใช้งานโปรแกรมต่างๆได้อย่างมีประสิทธิภาพสูงสุดในแต่ละเครื่อง

### 2.3.3. ข้อดีของ .NET Framework

- 1) ประโยชน์และข้อดีของ .NET Framework นั้นพอจะสรุปออกมาได้เป็นข้อๆดังนี้
- 2) เป็นระบบที่มีไลบรารีที่เป็นมาตรฐานเดียวกัน: เนื่องจากมีไลบรารีที่เป็นมาตรฐานเดียวกันทั้งหมดทำให้เราไม่ต้องกังวล ว่าภาษาที่ใช้เขียนนั้นมีไลบรารีตัวนั้นตัวนี้หรือไม่ รวมทั้งไม่ต้องคอยกังวลว่าถ้าใช้ไลบรารีของภาษาหนึ่งแล้วอีกภาษาหนึ่งจะไม่มีไลบรารีตัวนั้น
- 3) ไม่ขึ้นกับระบบปฏิบัติการ (OS): เนื่องจากระบบปฏิบัติการ ที่แต่ละบุคคลหรือองค์กรใช้นั้นย่อมไม่เหมือนกัน แต่ภายใน .NET Framework จะไม่มีปัญหาหนึ่งของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพียงแต่มีระบบ .NET Framework ก็จะทำให้สามารถใช้งานโปรแกรมต่างๆได้ ซึ่งเป็นข้อดีตรงที่เราจะสามารถใช้โปรแกรมต่างๆได้ทุกระบบปฏิบัติการ

- 4) ใช้ในการพัฒนาได้ทุกภาษา: ทำให้เราไม่ต้องคอยมาศึกษาภาษาใหม่ๆเมื่อต้องการสร้างโปรแกรมในแต่ละครั้ง นอกจากนั้นเรายังสามารถเลือก ใช้ภาษาที่เราถนัดที่สุดในการพัฒนาโปรแกรมต่างๆได้ด้วย
- 5) มีการควบคุมสิ่งแวดล้อมในการทำงานเป็นอย่างดี :เนื่องจากเป็นระบบที่เป็นมาตรฐานทำให้การควบคุมจัดสรรระบบต่างๆ ทำได้ง่ายขึ้น ไม่ว่าจะเป็นการจัดสรรหน่วยความจำ ด้านการใช้งานเครื่องก็มีความรวดเร็วมากขึ้น ลดโอกาสที่เครื่องจะแฮกก็ได้เป็นอย่างดี
- 6) ความปลอดภัยที่มีมากขึ้น: .NET Framework สามารถกำหนดสิทธิ์การใช้งานหรือ permission ของผู้ใช้งาน ได้มากขึ้นทำให้สามารถกำหนดว่า จะให้โปรแกรมในส่วนใดใช้งานได้หรือไม่ได้ แล้วแต่เฉพาะบุคคล

## 2.4. ASP.NET

### 2.4.1. ความเปลี่ยนแปลงที่เกิดขึ้นของ ASP.net

เราสามารถเลือกสร้างแอปพลิเคชัน ASP.net ได้ทันทีจาก Visual Studio.net ซึ่งการสร้าง

แอปพลิเคชันประเภทนี้ นอกจากจะสนับสนุนการทำงานบนอุปกรณ์ต่างๆ เช่น บราวเซอร์บนเครื่อง พีซี ,PDA ,Moblie phone แล้ว ตัว ASP.net มีสิ่งที่มีการเปลี่ยนแปลงมากที่สุด

### 2.4.2. นามสกุลของไฟล์ และการแยกส่วนออกแบบกับส่วนเขียนโค้ด

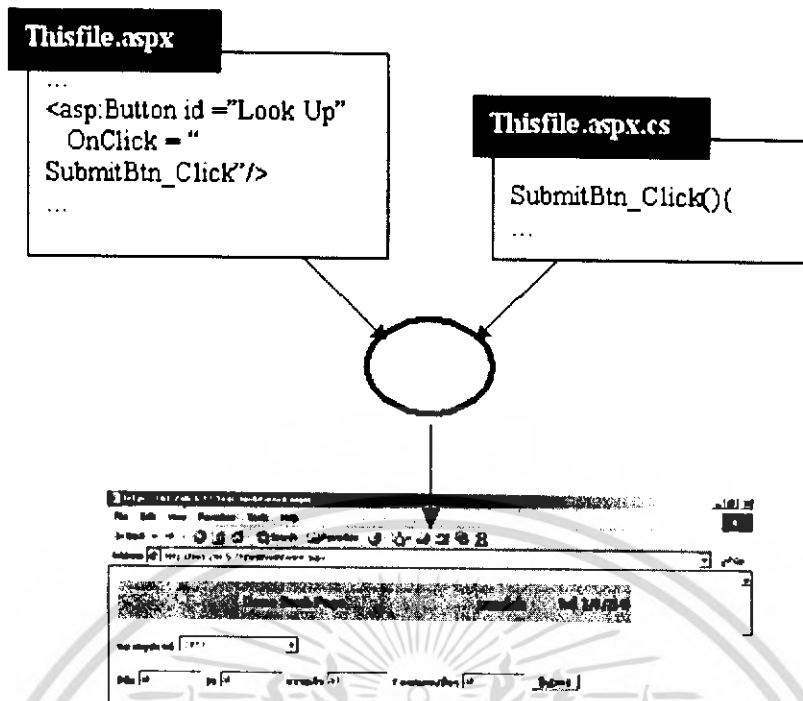
เริ่มจากนามสกุลของไฟล์ที่สร้าง จะเปลี่ยนจาก .asp เป็น .aspx และ 1 หน้าของเว็บเพจของไฟล์ .aspx

ประกอบด้วย 2 ส่วน คือ

1. ไฟล์ที่มีนามสกุล .aspx
2. ไฟล์ที่มีนามสกุลในแบบ Visual c# หรือ Visual Basic

เป็นการแยกระหว่างการออกแบบหน้าเว็บเพจกับการเขียนโปรแกรม ปัจจุบันการเขียนเว็บเพจจะรวมโค้ดทั้งหมดไว้ ไม่ว่าจะเขียนด้วย VBScript หรือ JavaScript และโครงสร้างของภาษา HTML จะรวมกันที่ไฟล์ ASP อย่างเดียวกัน นี้ทำให้ประสบปัญหา มาก แต่ไฟล์ .aspx จะแยกออกมา 2 ส่วนตามที่กล่าวมาแล้ว ทำให้การพัฒนาโปรแกรม เป็นเรื่องง่ายขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.6 แสดงเว็บแอปพลิเคชันที่สร้างด้วย ASP.net จะแยกไฟล์สำหรับออกแบบ(file.aspx) กับไฟล์สำหรับเขียนโค้ด โปรแกรมจากกัน(file.aspx.cs)

#### 2.4.3. การคอมไพล์โปรแกรม

ไฟล์ที่สร้างขึ้นจาก ASP.net เมื่อถึงตอนทำงาน คือ เมื่อทำการคอมไพล์แอปพลิเคชัน ทั้ง 2 ไฟล์จะถูกนำมารวมกัน และคอมไพล์ออกมาเป็น Common Language Runtime จากนั้นเมื่อมีการเรียกใช้งาน ก็จะถูกคอมไพล์เป็น ไบนารีโค้ดและนำไปรันเพื่อแสดงผลได้

ส่วนเว็บเพจที่เป็น ASP แบบเดิม จะมี ASP เป็นตัวแปลภาษา (interpret) คือทุกครั้ง ที่เว็บเพจจะถูกนำไปแสดงผล ก็จะต้องคอมไพล์ พอผู้ใช้คนที่ 2 เข้ามาเรียกเว็บเพจ ก็จะถูกคอมไพล์อีก ซึ่งต่างจาก ASP.net

โครงสร้างของไฟล์ .aspx ในบรรทัดแรก จะมีส่วนที่เป็นตัวบอกว่าเป็นไฟล์หนึ่งเป็น ไฟล์เขียนโค้ดโปรแกรม นั้น เป็นภาษาอะไร ไฟล์ชื่ออะไร ซึ่งเป็นการลิงค์ระหว่างไฟล์ Visual Basic หรือ Visual C# ที่ทำงานอยู่เบื้องหลังแอปพลิเคชันนี้

```
<%@Page language="vb"
Codebehind = "events_II.aspx.vb"%>
```

นอกจากนั้นยังมีแท็กคำสั่งพิเศษคือ runat="server" เพื่อบอกให้ทราบว่าการทำงานของโปรแกรมจะอยู่บนเซิร์ฟเวอร์

```
<Form runat="server"> ... </Form>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 2.4.4. คอนโทรลที่ใช้ออกแบบในแอปพลิเคชัน ASP.net

การเขียนโปรแกรมใน ASP.net จะใช้ภาษาใดก็ได้ใน .net ทั้งภาษาหลักใน Visual Studio.net อย่าง Visual C# , Visual Basic .net หรือภาษาอื่นๆ ที่สนับสนุน .net เช่น ภาษา Pascal ก็สามารเขียนได้ โดยแอปพลิเคชัน ที่สร้างขึ้นนี้สามารถรันได้บนทุกอุปกรณ์ที่สนับสนุน .net

แอปพลิเคชันที่สร้างด้วย ASP.net นั้น จะเป็นการแยกแยะระหว่างส่วนที่ใช้เขียนโค้ดโปรแกรม กับส่วนที่ใช้ออกแบบออกจากกัน โดยฟอร์มที่อยู่ใน ASP.net ก็จะคล้ายๆ กับ Windows Form ซึ่งคอนโทรล 3 ประเภทที่ใช้ในการสร้าง ASP.net มีดังนี้

#### 2.4.5. คอนโทรล HTML

คอนโทรล HTML นั้นจะมีโครงสร้างเหมือนเดิมที่ใช้บน ASP แต่เพิ่มบรรทัด Runat = "server" ขึ้นมา ก็บอกแล้วว่าคอนโทรลนี้ใช้กับ ASP.net เช่นเดียวกันหากในฟอร์มนี้ต้องการให้คอนโทรลบางตัวไม่ทำงานในแบบของ ASP.net เพียงแต่เอาคำว่า Runat "server" ออกเท่านั้นเอง

(System.Web.UI.HtmlControls)

```
<input type=button id=button1
```

```
OnServerClick = "Button1_Click" runat="server"/>
```



รูป 2.7 คอนโทรล HTML ที่เราสามารถเลือกสร้างได้จาก Tool Box

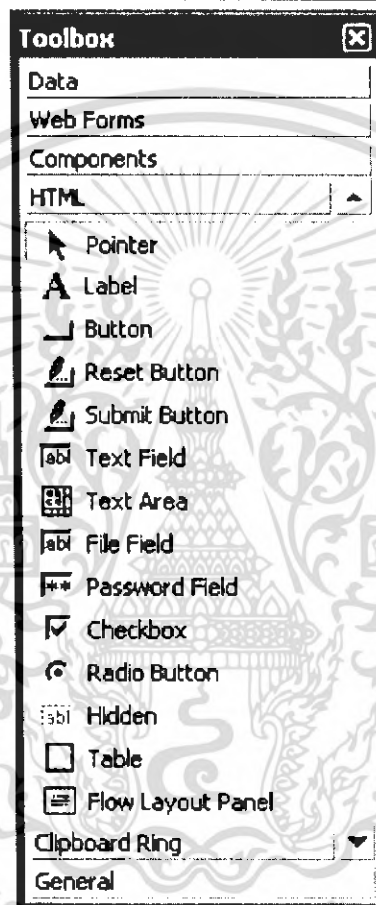
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 2.4.6. คอนโทรล Web Forms

คอนโทรล Web Forms เป็นคอนโทรลพิเศษ ที่มีเฉพาะใน ASP.net คอนโทรลพวกนี้ จะมีคำว่า ASP อยู่ข้างหน้าและมีคำว่า Runat = "Server"

(System.Web.UI.WebControls)

```
<asp: label id= " label " runat ="server"
backcolor = "red" ForeColor ="white">
Hello World </label>
```



รูป 2.8 คอนโทรล Web Forms ที่เราสามารถเลือกสร้างได้จาก Tool Box

#### 2.4.7. คอนโทรล Customer

คอนโทรลนี้มีนามสกุลเป็น .ascx เหมือน ActiveX control บน Windows Forms  
นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 2.4.8. คอนโทรลที่ทำงานฝั่งเซิร์ฟเวอร์

ตัวคอนโทรลที่ใช้สำหรับออกแบบบนเว็บเพจที่ทำงานทางฝั่งเซิร์ฟเวอร์นั้น มีกว่า 45 คอนโทรลตั้งแต่คอนโทรลมาตรฐานอย่าง TextBox, CheckBox จนถึงคอนโทรลพิเศษที่พัฒนาขึ้นมาเพื่อใช้กับASP.net

ตัวอย่างโค้ดของคอนโทรลพื้นฐานที่ทำงานบนเซิร์ฟเวอร์

```
<asp:textbox id =text1 runat=server/>
text1.text ="Hello World"

<asp:checkbox id=check1 runat=server/>
check1.checked=True

<asp:button id=button1 runat=server/>
button1_onClick()

<asp:DropDownList id=DropDownList1 runat=server>
DropDownList1.SelectedItem.Text = "Hello"
```

คอนโทรลพื้นฐานมีวิธีการใช้งานเหมือนกับคอนโทรล HTML ทั่วไป เพียงแต่เพิ่มคำว่า ASP และ : (โคลอน) ตามด้วยชนิดของคอนโทรล กับคำว่า runat = "server" รวมทั้งการอ้างถึงค่าอื่นๆ เช่นข้อความที่ต้องการให้แสดงบนคอนโทรล

ตัวอย่างของคอนโทรลพิเศษเช่น Data Grid ใช้แสดงข้อมูล สามารถรวมเข้ากับ ADO เพื่อนำข้อมูลจากฐานข้อมูลมาแสดง นอกจากนี้มีคอนโทรลประเภท Validation เพื่อตรวจสอบการป้อนข้อมูลของเรา

คอนโทรลประเภท Validation เป็นคอนโทรลที่เราพบบ่อยๆ เช่น การสร้างฟอร์มให้ผู้ใช้กรอกข้อมูล ในปัจจุบันหากเราสร้างคอนโทรลประเภทนี้เอง คงต้องมาทำการเขียนโค้ดเอง แต่สิ่งที่เกิดในASP.net คือมีคอนโทรลชื่อ RequireFieldValidator ให้เราสามารถตรวจสอบได้

```
<asp:RequiredFieldValidator ControlToValidate ="txtName"
ErrorMessage="Please Enter Your Name" runat ="server"/>
```

โดยเราสามารถกำหนดพร็อพเพอร์ตี้ที่ Error message หากไม่มีการกรอกข้อมูล ให้แสดงข้อความที่อยู่ต่อท้ายพร็อพเพอร์ตี้

#### 2.4.9. การเขียนโปรแกรมใน ASP.net

ASP นั้นมีการเขียนโปรแกรมเพื่อดักอีเวนต์ (Event Driven) ด้วย VBScript หรือ JavaScript โดยจะมี Event Driven 4 ตัวดังนี้

- Application\_OnStart เป็นอีเวนต์ที่เกิดขึ้น เมื่อมีการเรียกใช้งานเว็บเพจใดๆ ก็ได้ในเว็บไซด์เป็นเว็บเพจแรก
- Application\_OnEnd เกิดขึ้นเมื่อมีการปิดเครื่องเซิร์ฟเวอร์ หรือทำการ Stop Service ของเว็บเซิร์ฟเวอร์
- Session\_OnStart เกิดขึ้นเมื่อมีการเปิดเครื่องเซิร์ฟเวอร์ หรือทำการ Stop Services ของเว็บเซิร์ฟเวอร์
- Session\_OnEnd เกิดขึ้นเมื่อไคลเอนท์จบการทำงานบนเว็บไซด์

#### 2.4.10. สรุปตารางเปรียบเทียบระหว่าง ASP กับ ASP.net

คุณลักษณะ	ASP	ASP.NET
นามสกุลของไฟล์	.asp	.aspx , .ascx , .asmx
การทำงานของโปรแกรม	คอมไพล์โค้ดด้วยตัว Interpreter	คอมไพล์โค้ดด้วยตัว JIT Compiler
การออกแบบและการเขียนโปรแกรม	รวมส่วนการเขียนโปรแกรมและส่วนการออกแบบด้วยกัน	แยกส่วนการเขียนโปรแกรมและส่วนการออกแบบออกจากกัน
การจัดการเรื่องสถานะของโปรแกรม	มีการใช้ Cookies ที่ไคลเอนต์ไม่สามารถทำงานร่วมกันผ่านแพลตฟอร์มได้	สามารถฝังลง URL ได้เลยโดยไม่ต้องใช้ Cookies สามารถบันทึกข้อมูลที่ใช้ร่วมกันลงในฐานข้อมูล SQL ภายนอกได้
อัปเดตไฟล์ DLL Library	ต้องปิดเว็บเซิร์ฟเวอร์ก่อน	สามารถคัดลอกไฟล์ DLL ลงในไดเรกทอรี bin ได้ทันที (ไม่ต้องการลงทะเบีย่น)

ตาราง 2.6 แสดงตารางเปรียบเทียบระหว่าง ASP กับ ASP.net

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5. ภาษา C#

### 2.5.1. โครงสร้างพื้นฐาน

#### 1) Comment

เป็นข้อความที่ถูกแทรกเข้าไปในโปรแกรม เพื่ออธิบายให้ผู้อ่านสามารถเข้าใจโปรแกรมนั้นได้ง่ายขึ้นหรือเพื่อบันทึกข้อมูลที่เกี่ยวข้องกับโปรแกรมนั้น ข้อความที่เป็น comment จะไม่มีผลต่อการทำงานของโปรแกรม ภาษา C# มี comment อยู่ 3 แบบ

- เหมือนภาษา C คือ เริ่มต้นด้วย /\* และจบด้วย \*/ แบบนี้สามารถ comment ได้หลายบรรทัด เช่น `<font color="#006600" face="Courier New">/* This is multiple<br> line comment */`
- แบบเดียวกับ C++ คือเริ่มต้นด้วย // ไปจนจบบรรทัดนั้น โดยไม่ต้องมีตัวปิด
- แบบที่สามขึ้นต้นด้วย /// ข้อความนี้จะถูกนำไปทำเป็น xml documentation

#### 2) Keyword

คือคำ ที่ถูกกำหนดความหมายไว้แล้ว ในภาษาส่วนใหญ่ นั้นรวมทั้ง C# จะถือว่า keyword เป็น reserved word (คำสงวน) นั่นคือไม่สามารถนำไปใช้ใน ความหมายอื่น นั่นคือห้ามนำไปใช้ เป็นชื่อตัวแปร ชื่อฟังก์ชัน หรือชื่อคลาส ภาษา C# มี keyword ทั้งหมดดังนี้

abstract	as	base	bool	break
byte	case	catch	char	checked
class	const	continue	decimal	default
delegate	do	double	else	enum
event	explicit	extern	false	finally
fixed	float	for	foreach	goto
if	implicit	in	int	interface
internal	is	lock	long	namespace
new	null	object	operator	out
override	params	private	protected	public
readonly	ref	return	sbyte	sealed
short	sizeof	stackalloc	static	string
struct	switch	this	throw	true
try	typeof	uint	ulong	unchecked
unsafe	ushort	using	virtual	void
volatile	while			

ตาราง 2.7 ตารางแสดงชื่อคลาสในภาษา C#

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

C# คล้ายกับ Java ตรงที่รวมเอาคำว่า true, false และ null ซึ่งถือว่าเป็นค่าคงที่เข้ามาเป็น keyword

### 3) Identifier

คือชื่อที่ผู้เขียนโปรแกรมตั้งขึ้นได้ อาจเป็นชื่อของคลาส ตัวแปร ฟังก์ชัน ชื่อที่ถูกต้องจะต้องไม่เป็น keyword อาจประกอบด้วย ตัวอักษรตัวใหญ่, ตัวเล็ก, ตัวเลข และ underscore identifier ที่ถูกต้อง เช่น ThisIsABook, num1, \_x, System, Int32 โดยชื่อที่ถูกต้องห้ามขึ้นต้นด้วยตัวเลข และจะมีความยาวเท่าไรก็ได้ ภาษา C# นั้นเป็นภาษาที่ case-sensitive นั่นคือ ตัวเล็ก กับ ตัวใหญ่จะตีความไม่เหมือนกัน เช่น WriteLine กับ writeline นั้นเป็นคนละตัวกัน

#### Verbatim identifier

verbatim identifier คือ identifier ที่นำเอา keyword มาตั้งชื่อได้ โดยการเติมตัวอักษร @ เข้าไปข้างหน้า

#### Example :

```
class @class {
    public static void @static(bool @bool) {
        if (@bool)
            System.Console.WriteLine("true");
        else
            System.Console.WriteLine("false");
    }
}
```

หลักการตั้งชื่อ Identifier ที่ดีคือ

- ตั้งให้สื่อความหมาย
- ใช้ตัวเล็กตัวใหญ่เพื่อแบ่งคำ
- ถึงแม้ การตั้งชื่อด้วยคำเดียวกันแต่ต่างกันที่ตัวอักษรใหญ่ เล็ก ต่างกันจะทำได้ แต่เราก็ไม่ควร เนื่องจาก .NET นั้นสามารถเรียกใช้ คลาสหรือฟังก์ชันข้ามภาษากันได้ ดังนั้นภาษาที่ case-insensitive เช่น VB.NET อาจจะไม่สามารถใช้งาน identifier แบบนี้ได้

#### 4) Literal

คือลำดับของตัวอักษรที่ใช้แสดงค่าคงที่ใน source code แบ่งออกเป็น

- **Integral literal** คือค่าคงที่ของเลขจำนวนเต็ม เขียนได้ 3 แบบ
  - ตัวเลขล้วน ๆ ที่ไม่ขึ้นต้นด้วยเลข 0 จะถูกตีความเป็นเลขฐาน 10
  - ตัวเลข 0-7 ที่ขึ้นต้นด้วยเลข 0 จะถูกตีความเป็นเลขฐาน 8 เช่น 0400 คือ 256 ในเลขฐาน 10
  - ตัวเลข 0-9 และอักษร A-F(หรือ a-f) ที่ขึ้นต้นด้วย 0X(หรือ 0x) จะถูกตีความเป็นเลขฐาน 16 เช่น 0X7F คือ 127 ในเลขฐาน 10

โดย default แล้ว Integral literal จะถูกเก็บไว้ในหน่วยความจำขนาด 32-bit(ค่าคงที่ชนิด int) หากเราต้องการใช้ตัวเลขที่มากกว่านั้นคือตั้งแต่ 231 จนถึง  $2^{63} - 1$  จะต้องมีตัวอักษร l(แอลเล็ก) หรือ L ต่อท้ายเลขนั้น เพื่อบอก compiler ว่าเป็นเลข long integer(ค่าคงที่ชนิด long)

- **Floating-point Literal** คือค่าคงที่เลขทศนิยม เขียนได้สองแบบ
  - แบบมาตรฐาน เช่น 17.357810
  - แบบวิทยาศาสตร์ เช่น  $2.997E8(2.997 * 10^8)$  หรือ  $9.1093897e-31$  (จะใช้ e หรือ E ก็ได้)

ในภาษา C# มี floating-point สองแบบคือแบบ 32-bit(ค่าคงที่ชนิด float) ซึ่งจะมีความละเอียดของทศนิยมประมาณ 7 หลัก และแบบ 64-bit(ค่าคงที่ประเภท double) โดย default ถ้าเราไม่ระบุตัวอักษรต่อท้าย compiler จะตีความเป็น ค่าคงที่ชนิด double ซึ่งเราควรกำหนดไปให้ชัดเจนว่าเป็นตัวแปรขนาดใดโดยการเติมตัวอักษร f(หรือ F) เพื่อบอกว่ามีขนาด 32-bit หรือ d(หรือ D) เพื่อบอกว่ามีขนาด 64-bit

- **Boolean Literal** คือค่าคงที่ความจริง (truth value) มีค่าเป็น true หรือ false
- **Character Literal** คือค่าคงที่ตัวอักษร

โดยทั่วไปแล้ว จะถูกแสดงอยู่ใน single quote เช่น 'a' และ '0' หากเป็นอักษรพิเศษ เช่นพวก control character จะต้องถูกนำหน้าด้วย \ (backslash) โดยจะถูกเรียกว่า escaped character

- **String Literal** มีรูปแบบคล้าย ๆ กับ character ต่างกันที่จะถูกแสดงอยู่ใน double quote และการจะแสดงเครื่องหมาย single quote ไม่จำเป็นต้องใช้ escaped character

## 5) Operator

- **Assignment operator** ในภาษา C# นั้นก็ได้้นำความคิดมาจากภาษา C นั่นคือ เครื่องหมาย = เป็น operator ที่สามารถนำมาใช้ได้ ใน expression ปกติได้ เช่น  $x + (y = 1)$  นั่นคือมีการกำหนดค่า  $y$  ให้เท่ากับ 1 ก่อนแล้วจึงนำไป + กับ  $x$
- **Arithmetic operator** แบ่งย่อยออกเป็น
  - **Integer & Floating-point operator** ได้แก่ +, -, \*, /, %(modulus) โดยถ้า operand ทั้งสองของ operator / เป็น integer number, operator / จะถูกตีความเป็น div(หารแบบปัดเศษ) แต่ถ้ามี operand ตัวใดตัวหนึ่งเป็น floating-point number, operator / จะถูกกระทำแบบการหารปกติ
  - ในภาษา C# จะเหมือนกับ Java นั่นคือ modulus operator นั้นจะสามารถใช้กับ floating-point number ได้ด้วย
  - **Arithmetic assignment operator** ได้แก่ +=, -=, \*=, /=, %= มีไว้ช่วยเขียน ประโยค assignment ให้อ่านง่ายขึ้น เช่น  $x = x + 7$  สามารถเขียนได้เป็น  $x += 7$  อ่านได้ว่า เพิ่มค่า  $x$  อีก 7
  - **Increment and Decrement operator** ได้แก่ operator 2 ตัว คือ ++ และ -- ซึ่งเขียนได้ทั้งแบบ prefix และ postfix โดย operator นี้ เป็น operator ที่ใช้เพิ่มหรือลดค่าตัวแปรอีกทีละ 1 โดยจุดเริ่มต้นของ operator นี้มาจากภาษา C เพราะคำสั่งประเภทนี้จะถูกแปลเป็นคำสั่ง INC, DEC ในภาษา assembly ซึ่งจะทำงานได้เร็วกว่า ADD และ SUB แต่ในการใช้งานจริงแล้ว ประโยชน์ของมันคือการที่ code อ่านง่ายขึ้น
- **Bitwise operator** เป็น operator ที่จะกระทำกับทุก ๆ bit ของ operand ในชนิด integral type(sbyte, short, int, long, byte, ushort, uint, ulong)
  - **Boolean bitwise operator** ได้แก่ ~(bitwise unary NOT), &(bitwise AND), |(bitwise OR), ^(bitwise XOR), <<(shift left), >>(shift right)
  - **Assignment bitwise operator** เช่นเดียวกับ Arithmetic assignment operator คือนำเอา Boolean bitwise operator มารวมกับเครื่องหมาย = ได้เป็น operator ดังนี้ ~=, &=, |=, ^=, <<=, >>=

- **Relational operator** ได้แก่ ==(equal to), !=(not equal to), >(greater than), >= (greater than or equal), <(less than), <=(less than or equal) ใช้ในการเปรียบเทียบกับ operand ที่มีประเภท ตัวเลข, ตัวอักษร(char), ข้อความ(string), bool โดยข้อมูลประเภทที่ไม่ใช่ตัวเลขนั้น จะสามารถเปรียบเทียบโดยใช้ operator == และ != ได้เท่านั้น และต้องเปรียบเทียบข้อมูลประเภทเดียวกันเท่านั้น  
โดยผลลัพธ์ของการเปรียบเทียบนั้นจะได้ออกมาค่า boolean(true กับ false)
- **Logical operator** เนื่องจากว่าในภาษา C# มี boolean type แล้ว ดังนั้น operator ในกลุ่มนี้ จึงสามารถกระทำกับ operand ที่เป็น boolean เท่านั้น (เช่นเดียวกับ Java)
  - **Boolean Logical operator** ได้แก่ &(logical AND), |(logical OR), ^(logical XOR) และ !(logical NOT) จะสังเกตได้ว่า operator เหล่านี้ จะใช้เครื่องหมายเดียวกับ bitwise operator โดย compiler จะดูจาก operand ว่าเป็นชนิดใด จึงจะตัดสินใจใช้ operator ในความหมายเหล่านั้น
  - **Short-Circuit Logical operator** ได้แก่ &&(short-circuit AND) และ ||(short-circuit OR) ต่างกับ Boolean Logical operator ตรงที่ว่า จะหยุดคำนวณเมื่อ ผลลัพธ์ของทั้ง expression นั้นสามารถระบุค่าได้แล้ว เช่น
- **Conditional operator** จริง ๆ ไม่อยากเขียนเรื่องนี้เลยเพราะยังทำความเข้าใจกับรูปประโยคก่อนข้างหาก เขาเป็นว่ามันคือ if clause แบบใช้สัญลักษณ์ละกัน
- **String operator** มี +, =, ==, != และ += โดยเครื่องหมาย + มีไว้ต่อ string โดยที่จะต่างกับ + ทั่ว ๆ ไปตรงที่ไม่มีคุณสมบัติสลับที่ นอกนั้นก็มีความหมายเหมือนปกติ

## 6) Primitive Data Type

Data type	Description	Aliased From
sbyte	1-byte signed integer เก็บค่าตั้งแต่ -128 ถึง 127	public struct System.SByte
byte	1-byte unsigned integer เก็บค่าตั้งแต่ 0 ถึง 255	public struct System.Byte
short	2-byte signed integer เก็บค่าตั้งแต่ -32768( $-2^{15}$ ) ถึง 32767( $2^{15} - 1$ )	public struct System.Int16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data type	Description	Aliased From
ushort	2-byte unsigned integer เก็บค่าตั้งแต่ 0 ถึง $65536(2^{16} - 1)$	public struct System.UInt16
int	4-byte signed integer เก็บค่าตั้งแต่ -2,147,483,648( $-2^{31}$ ) ถึง 2,147,483,647( $2^{31} - 1$ )	public struct System.Int32
uint	4-byte unsigned integer เก็บค่าตั้งแต่ 0 ถึง $4,294,967,295(2^{32} - 1)$	public struct System.UInt32
long	8-byte signed integer เก็บค่าตั้งแต่ -9,223,372,036,854,775,808( $-2^{63}$ ) ถึง -9,223,372,036,854,775,807( $2^{63} - 1$ )	public struct System.Int64
ulong	8-byte signed integer เก็บค่าตั้งแต่ 0 ถึง $18,446,744,073,709,551,616(2^{64} - 1)$	public struct System.UInt64
float	4-byte floating-point number เก็บค่าในช่วง $\pm(1.401298E-453.4028235E+38$ ถึง $3.4028235E+38)$	public struct System.Single
double	8-byte floating-point number เก็บค่าในช่วง $\pm(4.94065645841246544E-324$ ถึง $1.79769313486231570E+308)$	public struct System.Double
bool	Boolean type เก็บค่า true และ false	public struct System.Boolean
char	2-byte Unicode character	public struct System.Char

ตาราง 2.8 แสดง Primitive Datatype

## 7) Data type อื่น ๆ ที่ใช้บ่อย

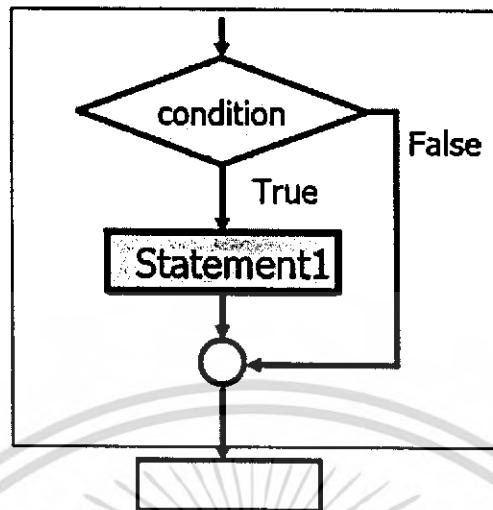
Data type	Description	Aliased From
decimal	floating-point แบบละเอียดเก็บค่าตั้งแต่ $(-2^{96}$ ถึง $2^{96})/10^{(0 \text{ ถึง } 28)}$ เวลาจะระบุค่า literal ใด ๆ ว่าเป็น decimal ให้ใช้ตัว M หรือ m ต่อท้ายเช่น 0.001m	public struct System.Decimal
DateTime	เก็บค่าข้อมูล วันที่และเวลาในช่วง 00:00:00 1 มกราคม 0001 จนถึง 23:59:59 31 ธันวาคม 9999 โดยที่มีความละเอียดในการเก็บ 0.1 millisecond(100 nanosecond)	public struct System.DateTime
string	ลำดับของ character ที่ถูก encode แบบ UTF-16	public class System.String
object	เป็น type ที่ใหญ่ที่สุดในนี้ คือทุก ๆ type สามารถเปลี่ยนมาเป็น type นี้ได้หมด (คล้าย ๆ กับ variant ใน VB® ตะครีบ (แต่ประชิด mem และ พลังงานในการใช้หน่วยอะ))	public class System.Object

ตาราง 2.9 แสดง Primitive Datatype อื่น ๆ ที่ใช้บ่อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5.2. Control Statement

### 1) If..Clause



รูป 2.9 แสดงรูปแบบการทำงานของ if..Clause

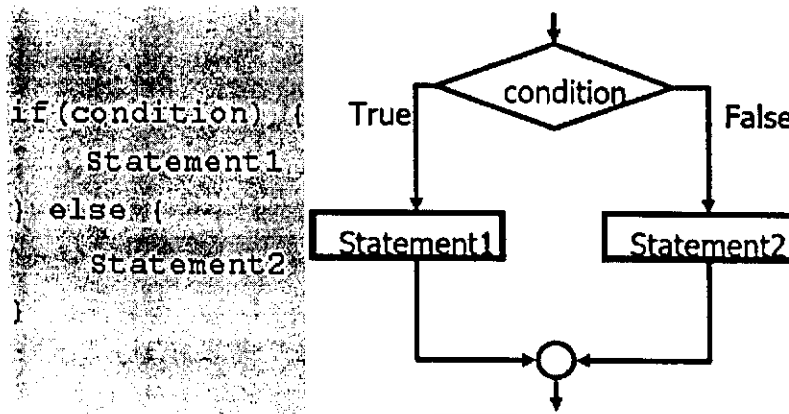
โดยปกติแล้ว โปรแกรมที่เขียนจะมีลำดับการทำงานจากบนไปล่าง ทีละคำสั่ง จบหนึ่งคำสั่งถึงไปทำอีกคำสั่งหนึ่ง แต่ถ้าต้องการให้มันทำงานเป็นเงื่อนไข ก็คือต้องตัดสินใจว่าจะทำคำสั่งหนึ่งหรือไม่ โดยขึ้นกับอีกคำสั่งหนึ่ง จะต้องใช้ control statement จำพวก if..else

โดยวงเล็บปิดจะเป็นตัวบอกว่าจบ condition แล้ว และเริ่ม statement ที่จะต้องทำงานเมื่อถูกเงื่อนไข และถ้าเราต้องการให้มีการ execute statement หลาย ๆ statement ในเงื่อนไขเดียว เราต้องใช้เครื่องหมาย '{' และ '}' (วงเล็บปีกกา) ครอบกลุ่มคำสั่งเหล่านั้นไว้ เช่น

```

if() {
// statement_1
// statement_2
// ...
// statement_n
}
  
```

ถึงแม้ว่า statement หลัง if จะมีแค่ statement เดียว เราก็ควรจะใส่วงเล็บปีกกาครอบไว้เสมอ เพื่อความเป็นระเบียบ



รูป 2.10 แสดงรูปแบบการทำงานของ If Statement

## 2) Switch Case

switch..case ก็เป็นการเขียน if clause แบบหนึ่งที่ยั่งยืน มีรูปแบบ syntax ดังนี้

```
switch(<expression>) {
    case <value> : <statement>
    case <value> : <statement>
    case <value> : <statement>
    .....
    [default : <statement>]
}
```

ที่ expression นั้นจะเป็นตัวแปร หรือ method ใด ๆ ก็ได้ โดย default นั้นเปรียบเสมือน else ... ซึ่งจะเป็น option ที่จะมีก็ได้ ไม่มีก็ได้

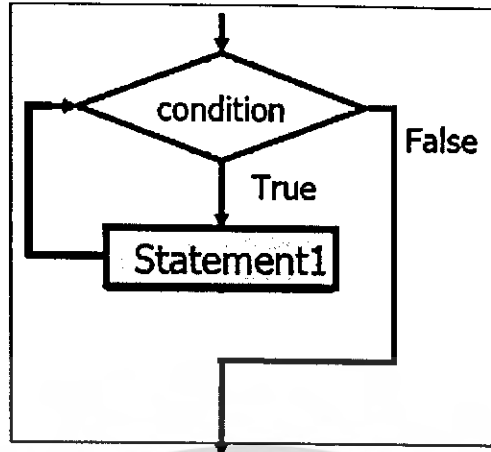
### 2.5.3. Loop Statement

การทำอะไรบางอย่างที่เป็นการทำซ้ำ ๆ กัน เราจะเรียกว่าการวนลูป คือการทำงานเป็นวงรอบนั่นเอง

โดย loop ในภาษานี้จะมีอยู่ 3 แบบคือ

- while( )..
- do..while( )
- for( ; ; )

## 1) while loop

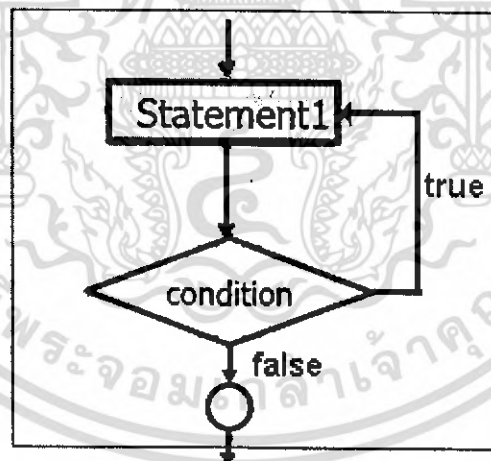


รูป 2.11 แสดงรูปแบบการทำงานของ While loop

จาก flow chart ข้างบน คือจะมีการเช็คเงื่อนไขก่อนเข้าลูป ถ้าเป็นจริง ก็  
จะทำงาน แล้วก็จะกลับมาเช็คเงื่อนไขใหม่ไปเรื่อยๆ โดยมี syntax ดังนี้

```
while(cond)
statement
```

## 2) do..while loop



รูป 2.12 แสดงรูปแบบการทำงานของ Do..While loop

มีลักษณะคล้ายๆกับ while loop แต่ต่างกันตรงที่ทำงาน statement ใน  
loop ก่อน1ครั้งแล้วจึงค่อยเช็คconditionมี syntax ดังนี้

```
do
statement
while(cond);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3) for..loop

for loop เป็นเหมือนกับ while loop ที่ลูปลงมา ไว้ใช้สำหรับการวนลูปที่กำหนดจำนวนครั้งเอาไว้ โดยมี รูปแบบ syntax ดังนี้

```
for(initial_expr; condition; update_expr)
    statement
```

## 2.6. MS SQL Server 2000

### 2.6.1. SQL Server 2000

SQL Server 2000 เป็นโปรแกรมจัดการฐานข้อมูลระดับเซิร์ฟเวอร์ที่มีขีดความสามารถในการรองรับข้อมูลขนาดใหญ่ที่ช่วยให้การบริหารจัดการฐานข้อมูลมีประสิทธิภาพ รวดเร็ว สนองตอบต่อความต้องการขององค์กรขนาดใหญ่

SQL Server 2000 เป็นระบบฐานข้อมูลและโซลูชันการวิเคราะห์ที่สมบูรณ์แบบ ซึ่งนำเสนอความน่าเชื่อถือและประสิทธิภาพ ด้านการขยายระบบที่เว็บและองค์กรธุรกิจต้องการ ด้วยการรองรับ XML และ HTTP ทำให้การเข้าถึงและการแลกเปลี่ยนข้อมูลนั้นทำได้ง่ายขึ้น ในขณะที่ความสามารถในการวิเคราะห์อันทรงพลังยังช่วยเพิ่มคุณค่าของข้อมูล และด้วยความพร้อมของระบบที่ดีขึ้น ทำให้ความสามารถในการทำงานอย่างต่อเนื่อง ของระบบนั้นสูงขึ้น, จัดการกับงานในแต่ละวันโดยอัตโนมัติได้ดีขึ้น รวมทั้งปรับปรุงเครื่องมือด้านการเขียน โปรแกรม และการพัฒนาความเร็วด้านการบริการได้ดีขึ้นเช่นกัน

SQL Server 2000 ประกอบไปด้วยเวอร์ชันต่าง ๆ ดังต่อไปนี้

- **Personal Edition** เป็นระบบจัดการฐานข้อมูลที่ใช้ได้เฉพาะเครื่องส่วนตัวเท่านั้น () ไม่สามารถใช้งานร่วมกับ SQL Server 2000 เวอร์ชันอื่นได้โดยตรง ส่วนใหญ่นิยมนำมาใช้ในการทดสอบการเขียนแอปพลิเคชัน ใช้ติดตั้งบน Windows 9x, ME, XP
- **Standard Edition** เป็นระบบจัดการฐานข้อมูลที่สามารถรองรับเซิร์ฟเวอร์ได้สูงสุดถึง 4 CPU และอ้างหน่วยความจำได้ถึง 2 GB นิยมนำมาใช้งานกับองค์กรขนาดใหญ่ ต้องติดตั้งบน Windows NT 4, 2000 หรือ 2003 Server แต่ถ้าติดตั้งบน Windows 9x, ME, XP จะติดตั้งให้เฉพาะส่วนที่เป็น Client Tools หรือ Connectivity เท่านั้น

- **Enterprise Edition** เป็นระบบจัดการฐานข้อมูลที่สามารถรองรับเซิร์ฟเวอร์ได้สูงสุดถึง 32 CPU และอ้างหน่วยความจำได้ถึง 64 GB นิยมนำมาใช้งานกับองค์กรที่มีขนาดใหญ่และมีสาขามากมาย เพราะมีระบบสนับสนุนการทำงานมากมาย เช่น การทำ Data Mining, Data Warehouse เป็นต้น ต้องติดตั้งบน Windows NT 4, 2000 หรือ 2003 Server เท่านั้น

### 2.6.2. TSQL (Transact-SQL)

เป็นคำสั่งที่ใช้ในการคิวรี (Query) ข้อมูลที่ต้องการ ประกอบไปด้วยคำสั่ง 3 ประเภทคือ

1. **Data Definition Language (DDL)** เป็นคำสั่งที่ใช้ในการจัดการอ็อบเจกต์ต่าง ๆ ของฐานข้อมูล เช่น ฐานข้อมูล ตาราง วิว ได้แก่คำสั่ง CREATE, ALTER, DROP
2. **Data Manipulation Language (DML)** เป็นคำสั่งที่ใช้ในการจัดการกับข้อมูลที่อยู่ในฐานข้อมูล ได้แก่คำสั่ง SELECT, INSERT, UPDATE และ DELETE
3. **Data Control Language (DCL)** เป็นคำสั่งที่ใช้ในการควบคุมการทำงานของเครื่อง เช่นการกำหนดสิทธิให้กับผู้ใช้งาน ได้แก่คำสั่ง GRANT เป็นต้น

SQL Server 2000 ได้ยึดมาตรฐาน ANSI SQL-92 ในการกำหนดคำสั่งที่ใช้งาน แต่ก็มีเพิ่มเติมขีดความสามารถในการทำงานให้ดียิ่งขึ้น นอกเหนือไปจากคำสั่งมาตรฐาน ทำให้มีขีดความสามารถสูง มีความยืดหยุ่นสูง โดยเรียกคำสั่ง SQL นี้ว่า Transact-SQL หรือ T-SQL

#### การเขียนคำสั่ง SELECT

ใช้สำหรับเลือกข้อมูลในตาราง มีรูปแบบการเขียนดังนี้

```
SELECT [Predicate] { * | [Table.]Field1 [AS Alias1]
[, [Table.]Field2 [AS Alias2] [, ...]]}
FROM Table[, ...]
[WHERE Search-Criteria]
[GROUP BY Group-List]
[HAVING Search-Criteria]
[ORDER BY Sort-Criteria]
```

**SELECT** เป็นคำสั่งที่ใช้กำหนดรายชื่อ Attribute (Field) ของ Relation (Table) ที่ต้องการให้แสดง

**FROM** เป็นคำสั่งที่ใช้สำหรับกำหนดรายชื่อ Relation (Table) ที่เป็นแหล่งข้อมูลที่ต้องการนำมาแสดงผล

**WHERE** เป็นคำสั่งที่ใช้สำหรับกำหนดเงื่อนไขในการเลือกข้อมูลจาก Relation (Table) ที่กำหนดจากส่วนของ FROM ขึ้นมาแสดง คำสั่งในส่วนนี้จะกำหนดหรือไม่ก็ได้ แต่ถ้าไม่กำหนดจะถือว่าต้องการทำงานกับทุก Tuple (Record) ใน Relation (Table) นั้น การกำหนดเงื่อนไขนั้นประกอบไปด้วยโอเปอเรเตอร์ Comparison, BETWEEN..AND, IN, LIKE, IS NULL, AND, OR, DISTINCT, TOP, ASC, DESC, GROUP BY, HAVING, ALL, COMPUTE, COMPUTE BY, ROLLUP, CUBE

### วิธีการเขียนภาษา SQL

การเขียนภาษา SQL มีข้อกำหนดดังนี้

1. เขียนได้ตั้งแต่ 1 บรรทัดเป็นต้นไป
2. ส่วนต่าง ๆ ของคำสั่งสามารถเขียนแยกบรรทัดได้
3. ภายใน 1 อนุประโยคไม่สามารถแยกบรรทัดได้
4. คำสั่ง SQL จะเรียกใช้ที่ SQL Prompt (SQL>)
5. ประโยคคำสั่งสามารถทำงานได้หลาย ๆ ครั้งในขณะที่ยังอยู่ในบัฟเฟอร์
6. ทุก ๆ ประโยคคำสั่งต้องจบด้วยเครื่องหมายเซมิโคลอน (;)

ตัวอย่างของการเขียนภาษา SQL มีดังนี้

แบบที่ 1

เมื่อ \* หมายความว่าถึง ข้อมูลจากทุกฟิลด์ที่ต้องการนำมาแสดงผล

```
SELECT * FROM LAWS;
```

แบบที่ 2

```
SELECT
*
FROM
LAWS;
```

## แบบที่ 3

```
SELECT *
FROM LAWS;
```

ไม่ว่าจะเขียนภาษา SQL แบบใดก็ตาม (แบบที่ 1 – แบบที่ 3) สามารถทำงานได้ ผลลัพธ์ออกมาเป็นเช่นเดียวกันทั้งหมด

สังเกตว่าทุกประโยคคำสั่ง SQL มาตรฐานจะสิ้นสุดลงด้วยเครื่องหมายเซมิโคลอน (;) แต่สำหรับ SQL Server จะไม่มีการใช้เครื่องหมายดังกล่าว ดังนั้นเวลาใช้งานต้องระมัดระวังด้วย เพราะถ้าใส่ลงไปจะมี Error เวลาสั่ง Run

## คุณสมบัติอื่นของภาษา SQL

เพื่อให้ภาษา SQL มีคุณสมบัติในการทำงานเพียงพอต่อการนำไปใช้งาน จึงมีการกำหนดคุณสมบัติต่าง ๆ ของภาษา SQL เพิ่มเติมขึ้นมา ดังนี้

1. การเพิ่มนิพจน์ (Expression) ขึ้นมาเพื่อใช้เป็นเงื่อนไขในการค้นหา
2. การสร้างชื่อคอลัมน์ใหม่ขึ้นมา (Alias)
3. การเชื่อมต่อคอลัมน์ (Concatenation Operator)
4. การแสดงค่าเฉพาะ (Literals)
5. การทำงานกับค่าที่เป็นค่าว่าง (Null)
6. การเรียกดูข้อมูลที่มีค่าไม่ซ้ำกัน (Distinct)
7. การเรียงลำดับข้อมูล (Order By)
8. การเรียกดูข้อมูล โดยมีเงื่อนไข (Where Clause)

## JOIN

เป็นการนำตารางหลาย ๆ ตารางมาควิรพร้อมกัน เพื่อหาข้อมูลที่ต้องการ

**Inner Join** เป็นการนำตาราง 2 ตารางมา join กัน ตามเงื่อนไขที่กำหนด

- Equi JOIN จะมีการแสดงข้อมูลซ้ำกันในคอลัมน์ที่นำมาเปรียบเทียบกัน
- Natural JOIN จะไม่นำข้อมูลคอลัมน์ที่เปรียบเทียบมาแสดง

**Cross Join (Unrestricted JOIN)** เป็นการ JOIN ที่แสดงถึงความเป็นไปได้ของผลลัพธ์ที่สามารถเกิดขึ้นได้ (Cartesian Product)

**Outer Join** การ JOIN ที่อยู่นอกเหนือเงื่อนไขที่กำหนด ประกอบไปด้วย

- LEFT OUTER JOIN (\* =) ขีดรายการจากตารางหลักเป็นหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- RIGHT OUTER JOIN (= \*) ขีดรายการจากตารางที่สองเป็นหลัก Outer Join เป็นการ JOIN ตารางเดียวกัน แต่ตั้งชื่อ alias ต่างกัน

### รูปแบบความสัมพันธ์

เนื่องจากตารางความสัมพันธ์กำหนดให้แต่ละตารางต้องมีความสัมพันธ์กัน ซึ่งเราสามารถแยกความสัมพันธ์ของตารางออกเป็น 4 แบบ ดังนี้

1. **หนึ่งรายการต่อหนึ่งรายการ (One to One)** ข้อมูลรายการหนึ่งไปสัมพันธ์กับข้อมูลของอีกตารางหนึ่งได้เพียงรายการเดียวเท่านั้น เช่น คนหนึ่งคนมีจังหวัดเกิดได้เพียงจังหวัดเดียวเท่านั้น
2. **หนึ่งรายการต่อหลายรายการ (One to Many)** ข้อมูลรายการหนึ่งไปสัมพันธ์กับข้อมูลของอีกตารางหนึ่งได้หลายรายการ เช่น ตำแหน่งพนักงานขายสามารถมีได้หลายคน
3. **หลายรายการต่อหนึ่งรายการ (Many to One)** ข้อมูลหลายรายการไปสัมพันธ์กับข้อมูลของอีกตารางหนึ่งได้เพียงรายการเดียวเท่านั้น เช่น พนักงานขายหลายคนทำงานอยู่บริษัทเดียวกัน
4. **หลายรายการต่อหลายรายการ (Many to Many)** ข้อมูลหลายรายการไปสัมพันธ์กับข้อมูลของอีกตารางหนึ่งได้หลายรายการ เช่น พนักงานคนหนึ่งสามารถรับปริญญาตรีจากมหาวิทยาลัยได้หลายใบ ในขณะที่เดียวกัน มหาวิทยาลัยก็สามารถผลิตบัณฑิตได้หลายคน

ในทางปฏิบัติพบว่าในการออกแบบฐานข้อมูลที่ดี จะมีความสัมพันธ์ของตารางเป็นแบบหนึ่งรายการต่อหนึ่งรายการ หรือแบบหนึ่งรายการต่อหลายรายการ และไม่ค่อยพบเห็นตารางที่มีความสัมพันธ์แบบหลายรายการต่อหลายรายการโดยตรง แต่จะใช้วิธีการสร้างตารางความสัมพันธ์แบบหนึ่งรายการต่อหนึ่งรายการหรือแบบหนึ่งรายการต่อหลายรายการเข้ามาทำหน้าที่เชื่อมความสัมพันธ์แทน

### การใช้ Sub Query

ในบางครั้งผลลัพธ์ที่ได้จากการ JOIN สามารถใช้ Subqueries แทนได้เช่นกัน แต่ก็มีบ้างที่เพียง Subqueries เท่านั้นจึงจะสามารถหาผลลัพธ์ที่ต้องการได้ แต่หากพูดถึงประสิทธิภาพในการทำงานของทั้งสองก็ไม่แตกต่างกันมากนัก โดยทั้งนี้ขึ้นอยู่กับเงื่อนไขในการระบุอีกด้วย

ตามปกติในการใช้งาน Subqueries จะถูกรออบไว้ด้วยวงเล็บเพื่อให้เห็นถึงส่วนของคิวรีย่อยไว้อย่างชัดเจน โดยที่คิวรีย่อยสามารถมีคิวรีย่อยได้อีกต่อไปเรื่อย ๆ หลายชั้นไม่จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Union**

เป็นตัวเชื่อมผลลัพธ์ที่เกิดจากต่างคิวรีกัน แต่มีลักษณะการแสดงตามลำดับของคอลัมน์คล้าย ๆ กันแล้วนำข้อมูลเหล่านี้มาจัดแสดงได้พร้อมกัน

**VIEW**

คือตารางเสมือนที่เกิดจากการจำลองตารางข้อมูล โดยนำเฉพาะบางคอลัมน์ บางแถว หรือแม้แต่สามารถนำข้อมูลบางส่วนจาก 2 ตารางขึ้นไป มาแสดงไว้ภายในตารางที่ว่านี้ได้ เช่น

ลำดับที่	ชื่อ-สกุล	โทรศัพท์	E-Mail
1	นายจิตเกษม พัฒนาศิริ	014241742	chitpatt@oalweb.com
2	นางสาวจิตติมา มโนหมั่นศรีตรา	040100464	thitima@oalweb.com
3	นายสมชาย พงษ์เกษม	029885025	somchaip@oalweb.com

} ตาราง

ตาราง 2.10 แสดงผล ที่ได้จากการ Query

ชื่อ-สกุล	E-Mail
นายจิตเกษม พัฒนาศิริ	chitpatt@oalweb.com
นางสาวจิตติมา มโนหมั่นศรีตรา	thitima@oalweb.com

} วิว

ตาราง 2.11 แสดงวิว ที่ได้จากการ Query

จะเห็นได้ว่าการสร้างวิว คือ การนำคิวรีมาสร้างให้เป็นออบเจกต์หนึ่งของฐานข้อมูล มีประโยชน์กับคิวรีที่มีการใช้งานบ่อย ๆ ทำให้สามารถแสดงข้อมูลได้อย่างรวดเร็ว

**SELECT INTO**

ใช้สำหรับการทำสำเนาข้อมูลจากตารางหนึ่งมาเป็นอีกตารางหนึ่ง มีรูปแบบการเขียนดังนี้

```
SELECT Column_list (Column1,Column 2, ...)
INTO new_TableName
FROM Table_list (TableName 1, TableName 2, ...)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**INSERT**

ใช้สำหรับเพิ่มแถวข้อมูลลงในตาราง มีรูปแบบการเขียนดังนี้

```
INSERT INTO TableName (Field1, Field2, ...) VALUES (Value1, Value2, ...)
```

```
INSERT INTO TableName SELECT-STATEMENT
```

**DELETE**

ใช้สำหรับลบแถวข้อมูลในตาราง มีรูปแบบการเขียนดังนี้

```
DELETE FROM TableName WHERE Search-Criteria
```

**UPDATE**

ใช้สำหรับแก้ไขข้อมูลในตาราง มีรูปแบบการเขียนดังนี้

```
UPDATE TableName SET ColumnName = Value, ... [WHERE Search-Criteria]
```

ในทางปฏิบัติจะไม่ค่อยเห็นการใช้งาน Transact-SQL โดยตรงมากนัก ทั้งนี้เพราะส่วนใหญ่จะนำคำสั่ง SQL ไปฝังไว้กับ Application ที่พัฒนาขึ้นมาจาก VB, Delphi, PB เป็นต้น

**2.6.3. Stored Procedure คืออะไร**

Stored procedures เป็นกลุ่มคำสั่ง SQL ร่วมกับคำสั่งที่เป็นพวก control flow พื้นฐานอย่างพวก if then else, for loop, while loop เป็นต้น Stored procedures จะแตกต่างกับคำสั่ง SQL หรือกลุ่มคำสั่ง SQL ตรงที่มันมีการทำ pre-compiled ไว้ก่อนล่วงหน้า คือทุกครั้งที่เรารันคำสั่ง store procedure ในครั้งแรก ตัว SQL Server จะคอมไพล์และเก็บคำสั่งนั้นไว้ในตารางระบบ (system table) ของตัว SQL Server แล้วจึงนำคำสั่งนั้นไป execute หลังคอมไพล์

คำสั่งที่อยู่ใน Stored procedures ก็คล้ายๆกับคำสั่งที่เราเห็นทั่วไปในภาษาโปรแกรม แต่ syntax ของมันค่อนข้างจะผสมระหว่างภาษา C กับ Pascal ซึ่งมันสามารถรับและรีเทิร์นค่าพารามิเตอร์เหมือนกับฟังก์ชันต่างๆไปได้ ซึ่งผลลัพธ์ของการเรียกใช้ Store Procedure มีความเป็นไปได้ดังนี้

1. ไม่ส่งอะไรกลับมาเลย กรณีนี้คำสั่งใน Store Procedure จะสั่งให้ update, addnew หรือ delete หรือทำอะไรก็ตามที่ไม่ต้องการส่งค่ากลับมา
2. ส่งค่ากลับมาเป็น recordset
3. ส่งค่ากลับมาเป็นตัวแปรเช่นตัวเลขหรือตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ข้อดีของ store procedure

Stored procedure มีคุณสมบัติดีกว่าการคิวรี (query) ข้อมูลธรรมดา เพราะว่าการคิวรี ข้อมูลธรรมดาจากหน้า ASP, ASP.NET , PHP หรือ script อื่นๆ หรือจากวินโดวส์ แอปพลิเคชันนั้น ทุกครั้งที่มีการส่งคำสั่ง sql ไปที่เครื่องที่ทำหน้าที่เป็น Database Server มันจะทำการคอมไพล์คำสั่ง query นั้นใหม่ทุกครั้งที่มีการเรียก นั่นหมายถึงถ้าเรียกบ่อยครั้งก็คอมไพล์คำสั่ง query บ่อยครั้งแล้วค่อยนำคำสั่ง query นั้นไป execute อีกที

แนวคิดของ Store Procedure จึงถูกนำมาแก้ไขปัญหาดังกล่าว โดยการเขียนชุดคำสั่ง query รวบรวมเอาไว้บน Database Server แล้วคอมไพล์ชุดคำสั่งเหล่านั้นไว้ครั้งแรกครั้งเดียว เมื่อเราต้องการ query ข้อมูล ก็สั่งให้ Store Procedure ที่เราต้องการให้ทำงานแทนการส่งชุดคำสั่ง sql ไปคอมไพล์และ execute ใหม่ นอกจากนี้ข้อดีอีกอย่าง Store Procedure ก็เราสามารถส่งพารามิเตอร์ทั้งเข้าและออก store procedure เพื่อมองให้เหมือนกับฟังก์ชันทั่วไปที่เราเขียน จะต่างกันก็ในด้านของความหมายเพราะ store procedure จะเป็นฟังก์ชันการทำงานทางด้านฐานข้อมูล

### ลักษณะที่คล้ายกับ Programming Language

Stored procedures ของ SQL Server จะมีคุณสมบัติที่คล้ายคลึงกับภาษาโปรแกรมทั่วไปแยกเป็นข้อๆ ได้ดังนี้

- พารามิเตอร์ที่ส่งเข้าไปหรือรีเทิร์นกลับมามี data type คล้ายกับภาษาทั่วๆไป เช่น int , char ,nvarchar เป็นต้น
- มีชุดคำสั่งที่ทำงานเกี่ยวกับ database และคำสั่งพื้นฐานอย่างพวก if then else, loop ต่างๆ รวมถึงคำสั่งที่เรียกเอา store อื่นมาทำงานด้วย
- มีการส่งค่า status กลับในกรณีที่เรียก procedure อื่นซึ่งระบุว่าทำงานสำเร็จหรือไม่สำเร็จ
- มีการ Precompiled คือคอมไพล์ครั้งแรกแล้วเก็บไว้ใน code ที่คอมไพล์ไว้เพื่อไม่ต้องคอมไพล์อีก
- ลดปัญหาการจราจรบนเน็ตเวิร์กเพื่อไม่ต้องส่งคำสั่ง query ยาวๆ ไปหาเครื่องที่เป็น SQL Server เพราะจะส่งเพียงชื่อและพารามิเตอร์ไปเท่านั้น
- เพิ่มประสิทธิภาพในการเรียกใช้โค้ดที่ทำงานซ้ำๆกัน เพื่อจะได้ไม่ต้องเขียนโค้ดในหลายๆที่
- กำหนดสิทธิได้ว่าจะให้ user คนไหนมีสิทธิรัน Store Procedure ไหนบ้าง

- ความเร็วในการคิวรีที่รวดเร็วมก เพราะในตอน precompile ถ้าโค้ดที่เป็น Transact-SQL ของเรามีความสลับซับซ้อนตัวคอมไพเลอร์จะทำการ Optimize โค้ดเหล่านั้นให้เราด้วย
- สามารถกำหนดได้ว่าจะให้ execute Store Procedure บนเครื่อง local หรือบนเครื่อง SQL ซึ่งมันจะช่วยในเรื่องของการประมวลผลข้าม server ในกรณีที่เป็น Farm Server

### รูปแบบของ Store Procedure

สามารถกำหนดชื่อของ Store Procedure ได้ความยาวสูงสุด 30 ตัวอักษรด้วยการกำหนด syntax ของ Store Procedure ดังนี้

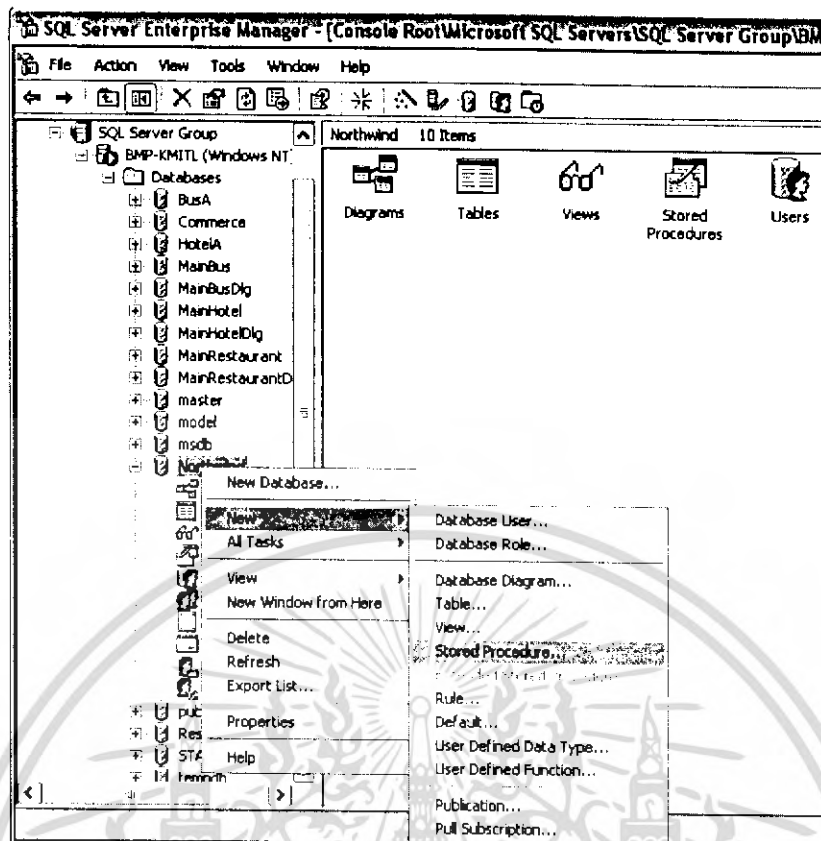
```
CREATE PROCEDURE [owner,] procedure_name [;number]
[@parameter_name datatype [=default] [OUTput]
...
[@parameter_name datatype [=default] [OUTput]
[FOR REPLICATION] | [WITH RECOMPILE] , ENCRYPTION
AS sql_statements
```

ภายหลังที่สร้าง Store Procedure เรียบร้อยแล้ว สามารถทดสอบ Store Procedure โดยผ่านทางกรเขียนโค้ดที่ใช้ ADO หรือผ่านทางโปรแกรม Query Analyzer ที่มีมากับ SQL Server ได้

### เริ่มทดลองสร้าง Store Procedure

- 1) ให้เปิด Enterprise Manager ของ SQL-Server ขึ้นมา
- 2) เลือกไปที่ database NorthWind
- 3) Click ขวาที่ Store Procedure ดังรูป
- 4) เลือก New Store Procedure
- 5) ป้อนโค้ดต่อไปนี้ลงไปหน้าต่างที่โผล่ขึ้นมา

```
create procedure all_employees
as select * from employees
GO
```

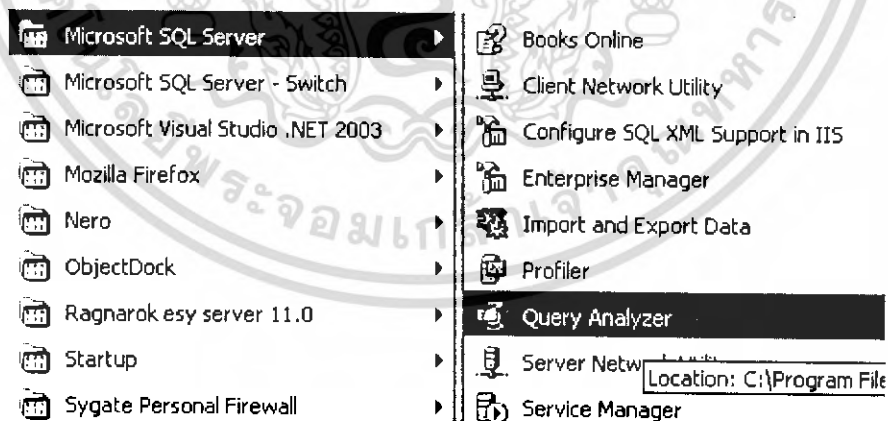


รูป 2.13 แสดงการสร้าง Stored Procedure

6) คลิก OK

### การเรียกใช้ Store Procedure ผ่าน Query Analyzer

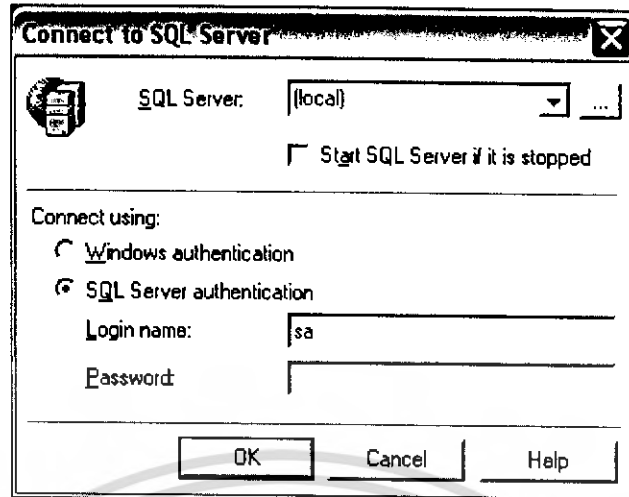
1) เปิดโปรแกรม Query Analyzer



รูป 2.14 แสดงการเปิดโปรแกรม Query Analyzer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 2) เลือก SQL Server และป้อน username password



รูป 2.15 แสดงการเปิดเลือกเปิด SQL Server

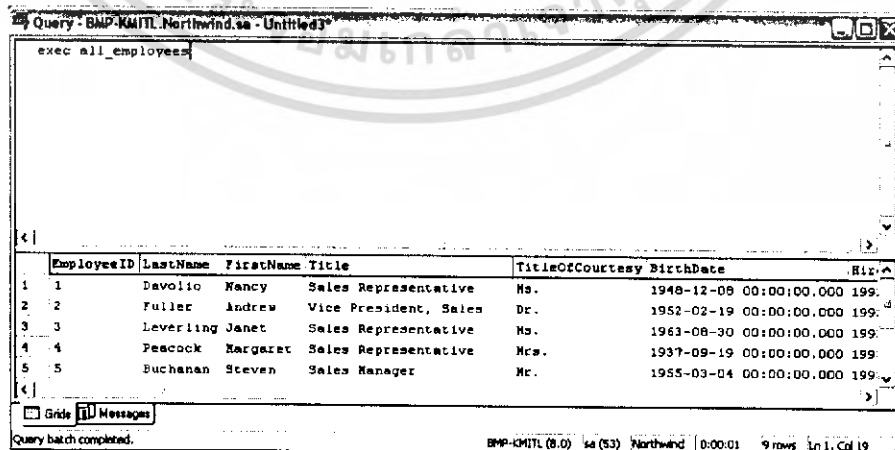
- 3) เลือก Database ไปที่ NorthWind



รูป 2.16 แสดงการเปิดเลือกเปิด Database NorthWind ใน SQL Server

- 4) พิมพ์คำสั่งต่อไปนี้

exec all\_employees แล้ว click ปุ่ม ▶ จะได้ผลลัพธ์เป็น recordset คืนกลับมา



รูป 2.17 แสดงผลจากการ Query โดยใช้ Store Procedure

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ 3

#### การออกแบบและพัฒนา

##### 3.1 บทนำ

ระบบ Smart Tour Agency เป็นระบบจัดการสร้างแพ็คเกจ ซึ่งเป็นเว็บเซอร์วิสให้บริการ จัดหา และติดต่อของ การขนส่ง ที่พัก ร้านอาหาร และสถานที่ท่องเที่ยว โดยผู้ให้บริการสามารถ ค้นหาข้อมูลต่าง ๆ ได้ตามความต้องการของผู้ใช้บริการเช่น ลักษณะการเดินทางโดยรถ เรือ หรือ เครื่องบิน ช่วงเวลาการท่องเที่ยว งบประมาณต่อคน เป็นต้น จากนั้นระบบนำข้อมูลเหล่านี้ไปใช้ในการ ค้นหาข้อมูลจากผู้ให้บริการเว็บเซอร์วิส และนำข้อมูลที่ได้มาประมวลผล และจัดสร้างแพ็คเกจ ที่เป็นไปได้ตามความต้องการของผู้ใช้บริการ เพื่อเป็นทางเลือกให้กับผู้ให้บริการ

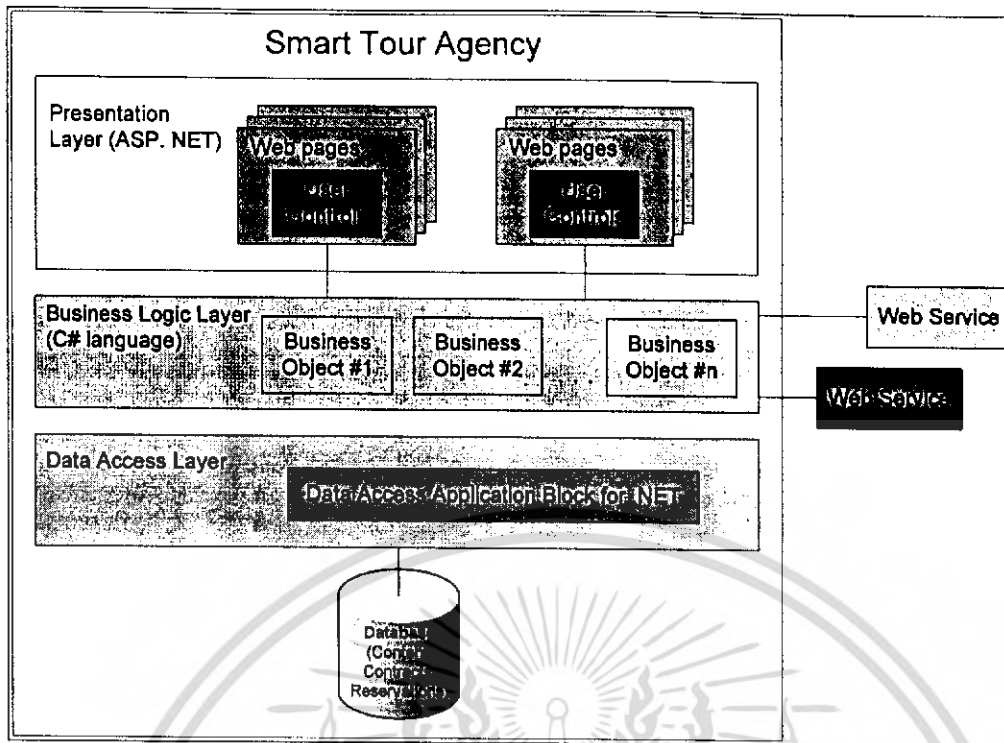
เมื่อผู้ให้บริการทำการจองแพ็คเกจ ระบบจะทำการติดต่อกับเซิร์ฟเวอร์ ของผู้ให้บริการของ ทั้งที่พัก และการเดินทาง เพื่อทำการสำรองไว้ให้กับผู้ให้บริการ หลังจากทำการจองแพ็คเกจแล้ว ผู้ให้บริการสามารถตรวจสอบรายละเอียด และสามารถยกเลิกการจองแพ็คเกจนั้นได้

ระบบสามารถสร้างรายงานจากข้อมูลการตั้งจองของผู้ให้บริการให้กับผู้ดูแลระบบและผู้ ให้บริการ เพื่อนำมาแสดงผลและใช้เป็นสถิติในการเปรียบเทียบต่างๆ ตามความต้องการของผู้ที่ ให้บริการ

##### 3.2 สถาปัตยกรรมที่ 3 เทียร์

ระบบงานที่ได้สร้างขึ้น ได้มีการใช้เทคโนโลยีต่าง ๆ ที่ได้ศึกษามา โดยได้สร้างระบบ เป็นแบบ 3 เทียร์ คือ

1. Presentation - เว็บเพจทั้งหมดสร้างขึ้นบน Web Forms Designer ใน Visual Studio .NET และพัฒนาโดยใช้ Microsoft ASP.NET
2. Business - พัฒนาโดยใช้ Microsoft Visual C# 2003 development tool ซึ่งจะ ทำงานเกี่ยวกับการจัดการ การจองต่าง ๆ การจัดการการสร้างแพ็คเกจที่เป็นไปได้ตามความต้องการของผู้ใช้
3. Data Access – เข้าถึงข้อมูลผ่านคลาส Microsoft ADO.NET และเพื่อเพิ่ม ประสิทธิภาพ และความปลอดภัย การเข้าถึงฐานข้อมูลจะทำงานผ่าน Stored Procedures (no inline Structured Query Language (SQL)) และใช้ดาต้าเบส เซิร์ฟเวอร์เป็น Microsoft SQL Server 2000



รูป 3.1 ภาพแสดงโครงสร้างสถาปัตยกรรม 3 เทียร์ของระบบ Smart Tour Agency

### 3.3 รายละเอียดของแอปพลิเคชัน

ระบบของ Smart Tour Agency ประกอบด้วยบริการต่างๆ ดังนี้

1. การลงทะเบียน
2. การเข้าระบบ
3. การค้นหาข้อมูลต่างๆ
  - การขนส่ง
  - โรงแรม
  - ร้านอาหาร
  - สถานที่ท่องเที่ยว
4. การประมวลผลต่างๆ
5. การจอง
6. การยกเลิกการจอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การลงทะเบียน

สามารถที่จะลงทะเบียนผ่านทางหน้าเว็บไซต์ได้ โดยจะต้องกรอกรายละเอียดต่าง ๆ ใ้ครบทั้งหมด

### 1. การเข้าสู่ระบบ

ได้มีการแบ่งระดับของผู้ใช้ที่จะทำการเข้าสู่ระบบมา โดยจะแบ่งไว้ 3 ระดับ คือ

- ผู้ดูแลระบบ (Administrator)
- ผู้ให้บริการ (Service Provider)
- ผู้ใช้บริการ (Client)

### 2. การค้นหาข้อมูล

ระบบจะทำการค้นหาตามข้อมูลที่ผู้ให้บริการระบุมาจากผู้ที่ให้บริการ ซึ่งจะทำการค้นหาทั้งการขนส่ง ที่พัก ร้านอาหาร ที่มีส่วนเกี่ยวข้องกับข้อมูลของผู้ให้บริการระบุมา โดยข้อมูลที่จะแสดงออกมานั้นจะถูกทำการประมวลผล เพื่อเป็นการกรองข้อมูลออกมาตามความเหมาะสมในแต่ละช่วงเวลา

### 3. การประมวลผลข้อมูล

ระบบจะนำข้อมูลทั้งหมดที่ได้จากการระบุของผู้ใช้ทั้งหมดมาทำเป็นสถิติเพื่อเป็นส่วนในการตัดสินใจเลือกของแพ็คเกจของผู้ใช้ และในการสร้างแพ็คเกจระบบจะนำข้อมูลอื่น ๆ ที่เกี่ยวข้อง เช่น ระยะเวลา สถานที่ตั้ง นำมาใช้ในการประมวลผลสร้างแพ็คเกจที่มีความเป็นไปได้จริง ในด้านการขนส่งระบบจะนำมารวมในการประมวลผลด้วยโดยระบบจะประมวลจากงบประมาณ จำนวนคน ความเป็นไปได้

### 4. การจอง

ในการจองที่พัก และร้านอาหาร ระบบจะให้ผู้ใช้บริการทำการเลือกสถานที่สำรองเพื่อเพิ่มทางเลือกในการสร้างแพ็คเกจให้กับผู้ใช้บริการ ในด้านสถานที่ท่องเที่ยวผู้ใช้บริการจะสามารถเลือกสถานที่ท่องเที่ยวได้ตามจำนวนวัน โดยภายในหนึ่งวันสามารถเลือกสถานที่ท่องเที่ยวได้ไม่เกิน 2 ที่ ในการจองแพ็คเกจแต่ละครั้งจำเป็นต้องชำระเงินเป็น 30 เปอร์เซ็นต์ของทั้งหมดในทันทีผ่านทางบัตรเครดิต และชำระเงินในส่วนที่เหลือภายในระยะเวลาที่กำหนดไว้

### 5. การยกเลิกการจอง

การยกเลิกการจองแพ็คเกจผู้ใช้สามารถทำได้ผ่านทางเว็บไซต์ ถ้าผู้ใช้บริการทำการยกเลิกการจองก่อน ทางบริษัทจะไม่คืนจำนวนเงินที่ชำระก่อนหน้า

### 3.4 ขั้นตอนการจัดทำแพ็คเกจ

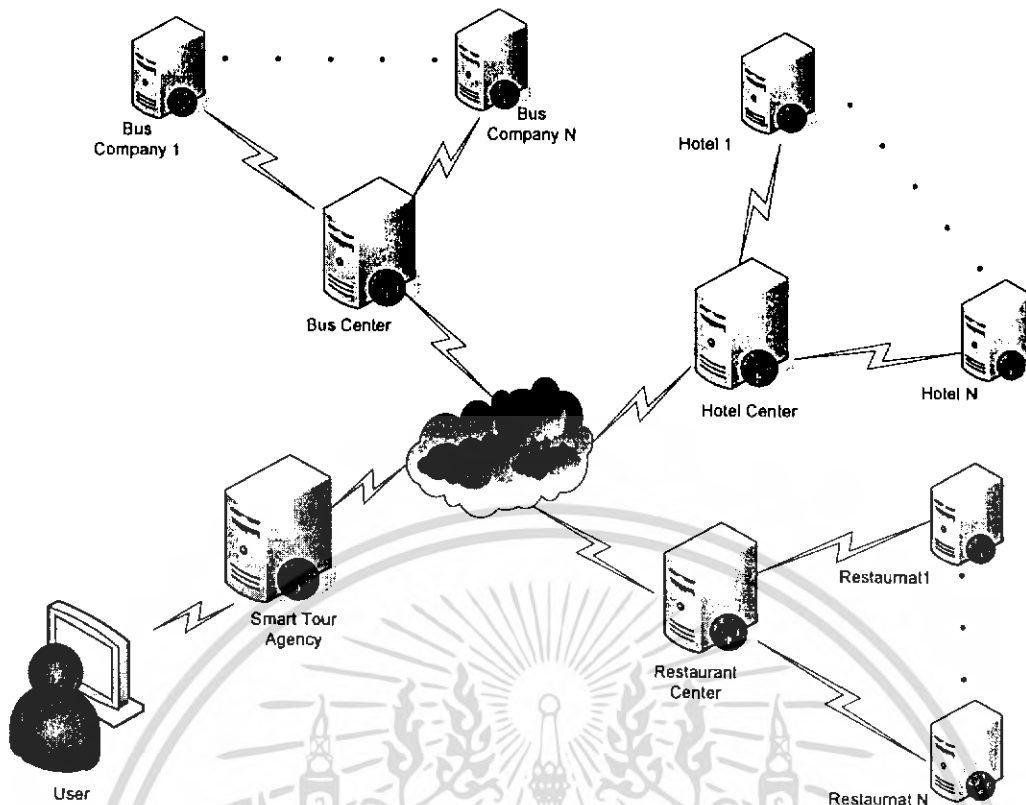
1. ผู้ใช้กรอกข้อมูลการท่องเที่ยว เช่น จังหวัดที่จะไป สถานที่ท่องเที่ยว วันออกเดินทาง ลักษณะการเดินทาง งบประมาณ จำนวนคน สถานที่พัก จำนวนห้องพัก ร้านอาหาร เป็นต้น
2. นักท่องเที่ยว Submit ฟอร์ม เพื่อนำข้อมูลไปใช้ หาห้องพัก สถานที่ท่องเที่ยว ร้านอาหาร และจัดทำเป็น รายการของแพ็คเกจ
3. ระบบทำการหาห้องพัก โรงแรมที่ตรงตามเงื่อนไขที่ผู้ใช้ระบุ
  - 3.1. ระบบติดต่อ ไปยังระบบ โรงแรมเพื่อตรวจสอบห้องพักที่เป็นไปตามเงื่อนไขของผู้ใช้
  - 3.2. ระบบโรงแรมตอบกลับ พร้อมส่งข้อมูลห้องพักกลับมา
4. ระบบแสดงรายการห้องพักที่ตรงตามเงื่อนไขเพื่อให้ผู้ใช้เลือก
5. ผู้ใช้เลือกห้องพักตามที่ต้องการ
6. ระบบทำการค้นหาสถานที่ท่องเที่ยวที่ตรงตามเงื่อนไขจากฐานข้อมูลระบบ และแสดงรายการสถานที่ท่องเที่ยว
7. ผู้ใช้เลือกสถานที่ท่องเที่ยวที่ต้องการ ไปในแต่ละวันของการท่องเที่ยว
8. ระบบทำการหาร้านอาหารที่ตรงตามเงื่อนไขที่ผู้ใช้ระบุจากฐานข้อมูลระบบ
9. ผู้ใช้เลือกร้านอาหารที่ต้องการ ไปในแต่ละวัน
10. ระบบนำข้อมูลการเลือกทั้งหมดจากข้อ 5, 7, 9 และข้อมูลการท่องเที่ยวจากข้อ 1 มาจัดทำรายการแพ็คเกจที่เป็นไปได้ตามเงื่อนไขต่างๆ
11. ผู้ใช้เลือกแพ็คเกจที่ต้องการ จากที่ระบบได้ประมวลผลขึ้นมา
12. ผู้ใช้ต้องชำระเงินล่วงหน้าจำนวน 30 เปอร์เซ็นต์ จากราคาทั้งหมดของแพ็คเกจที่เลือกไว้แล้วก่อน ผ่านทางเว็บไซต์ได้เลย
13. ระบบจะทำการติดต่อไปยังระบบ โรงแรม ระบบรถเช่า ระบบร้านอาหาร และสถานที่ท่องเที่ยว เพื่อติดต่อจอง
14. ระบบจะกำหนดวันสุดท้ายที่จะต้องชำระเงิน
15. ผู้ใช้ชำระเงินตามราคาแพ็คเกจที่ได้เลือกไว้

### 3.5 โครงสร้างระบบ

ผู้ที่เกี่ยวข้องในระบบ

- โรงแรม
- ร้านอาหาร
- บริษัทขนส่ง
- นักท่องเที่ยว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.2 แสดงโครงสร้างระบบ Smart Tour Agency

### 3.6 Use Case

Use Case ของระบบมีดังนี้

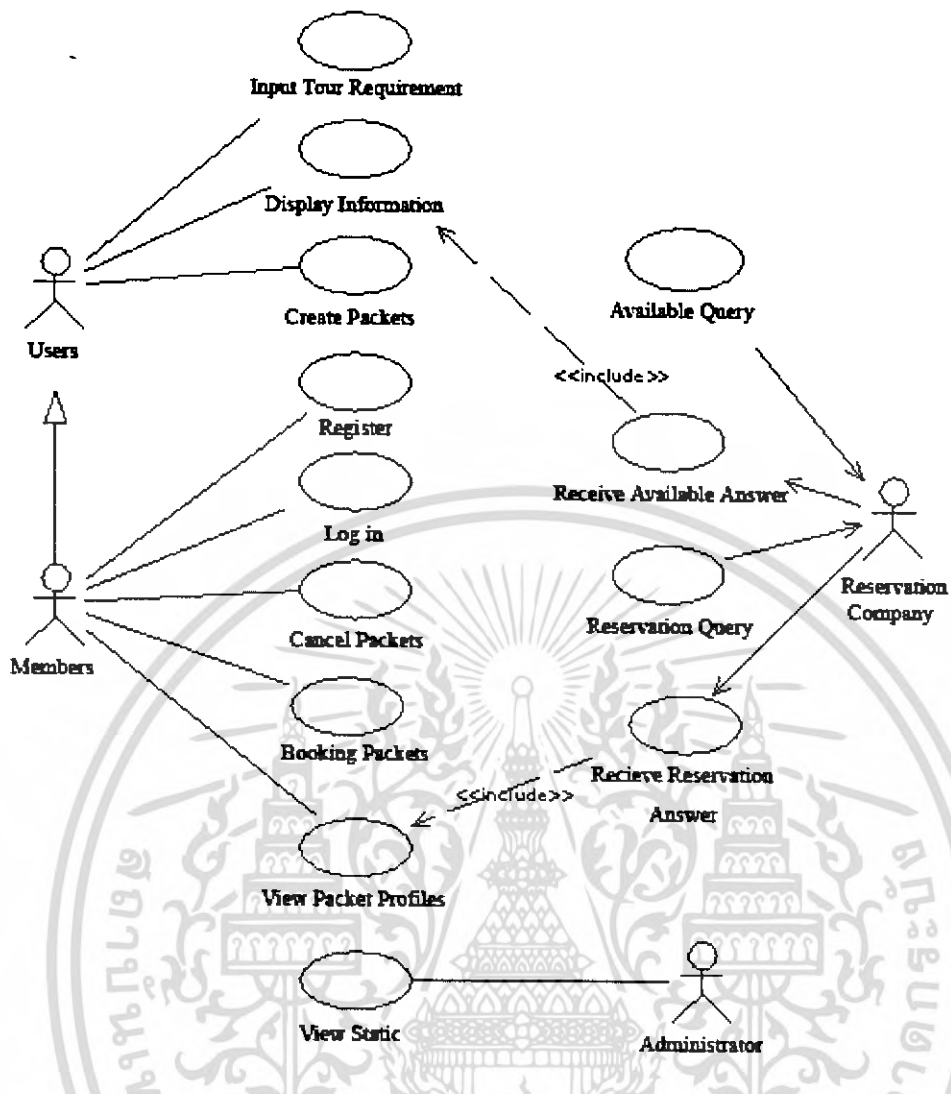
นักท่องเที่ยว

- ลงทะเบียนเพื่อเป็นสมาชิก
- ทำการเชื่อมต่อกับระบบ เพื่อเข้ามาใช้บริการ
- การระบุข้อมูลในการเดินทาง เช่น
  - การกำหนดจุดหมายปลายทาง และจำนวนของคนในหมู่คณะ
  - การกำหนดลักษณะการเดินทาง
  - การกำหนดสถานที่พักแรม รวมถึงลักษณะของห้องพัก
  - การกำหนดสถานที่ท่องเที่ยว
- ทำการแก้ไขข้อมูลที่ได้กำหนดไว้ได้
- สั่งจองแพคเกจที่ได้สร้างไว้

เจ้าของระบบ

- ดูข้อมูลสถิติการจองแพคเกจ

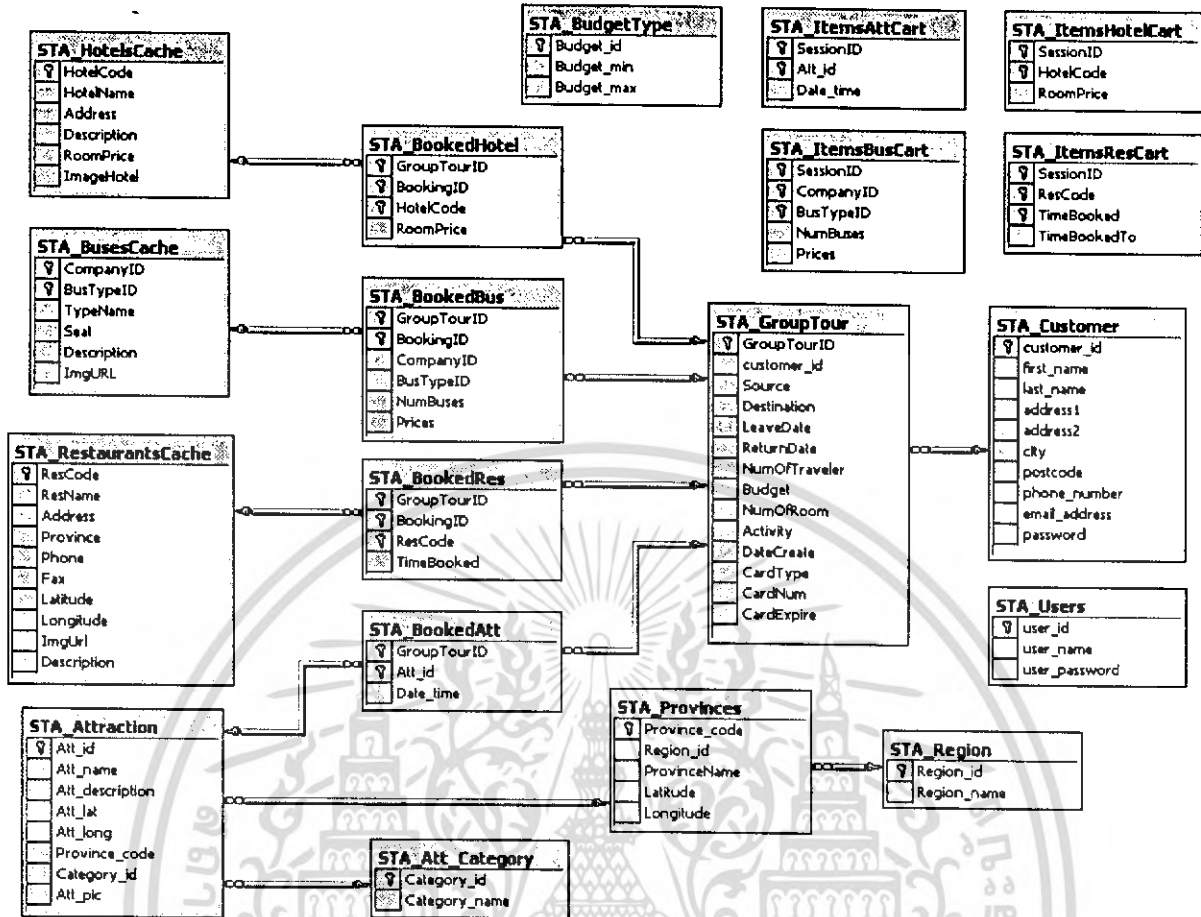
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



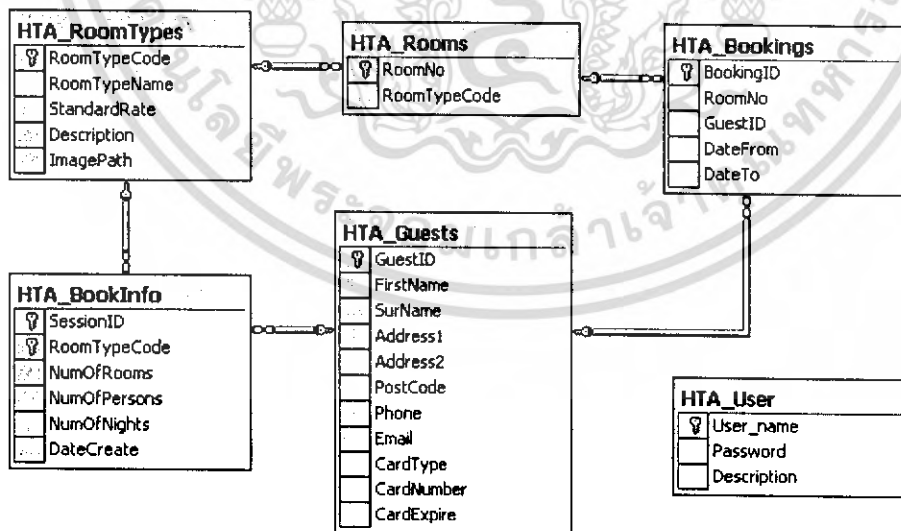
รูป 3.3 แสดง Use Case Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.7 Database Model Diagram

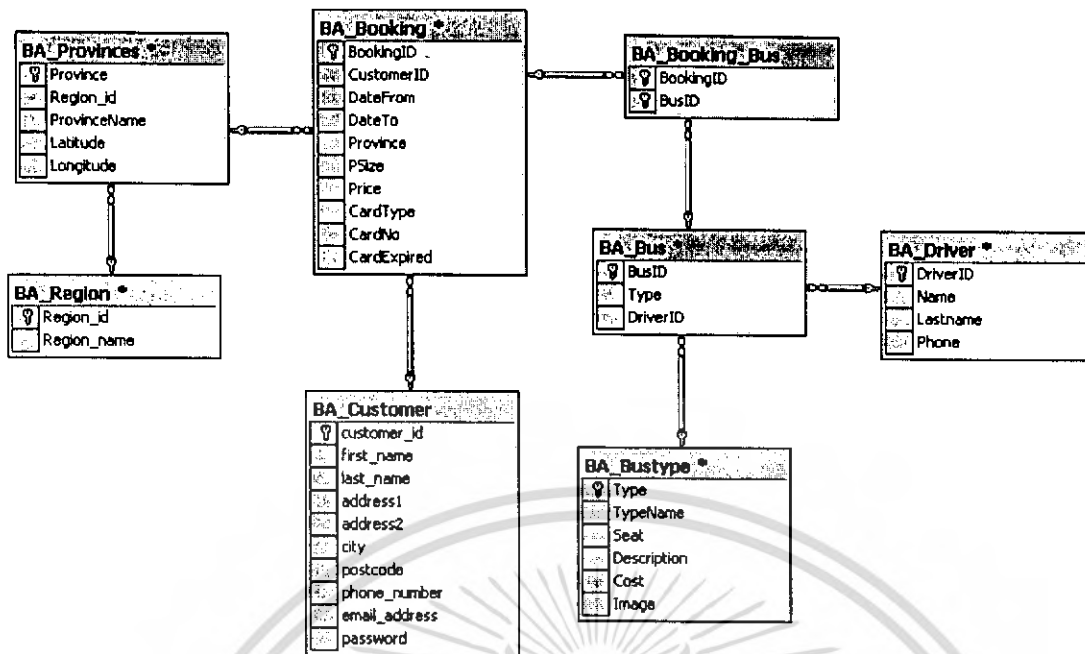


รูป 3.4 แสดง STA Database Model Diagram

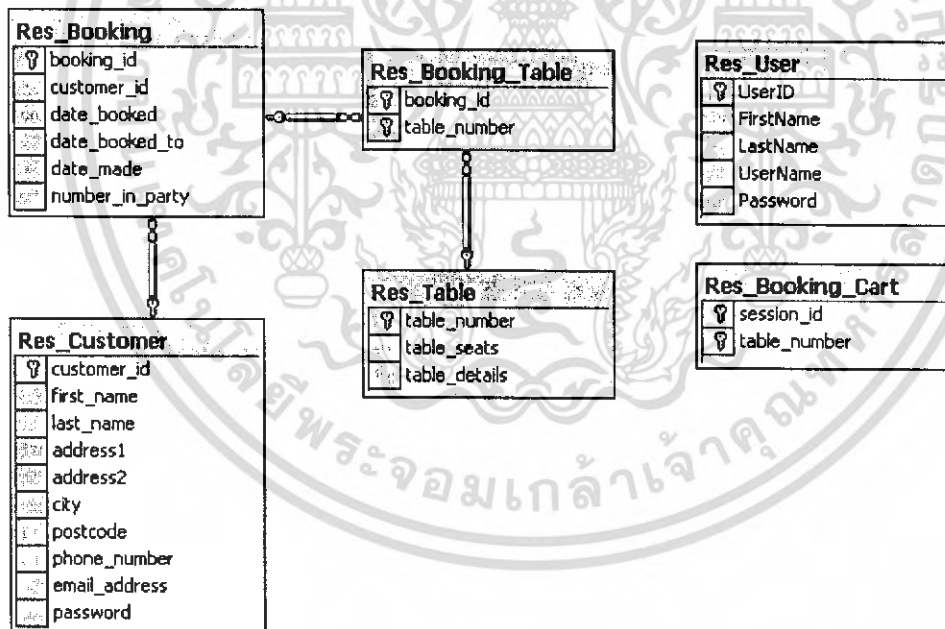


รูป 3.5 แสดง Hotel Database Model Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

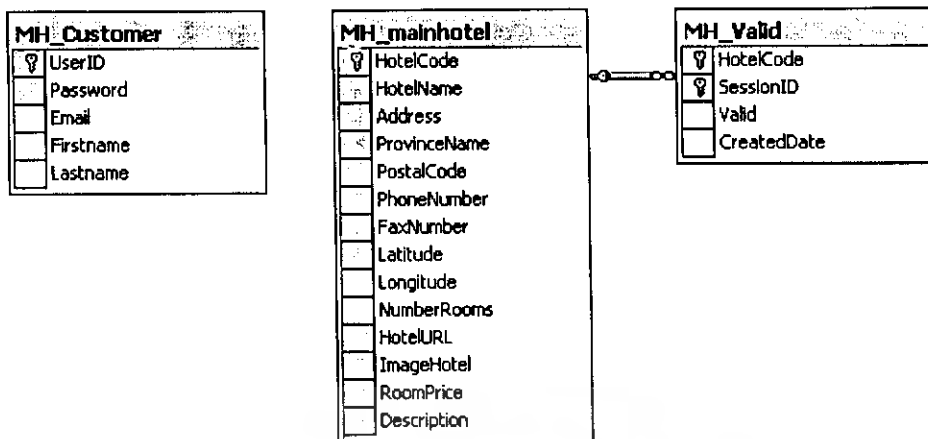


รูป 3.6 แสดง Bus Database Model Diagram

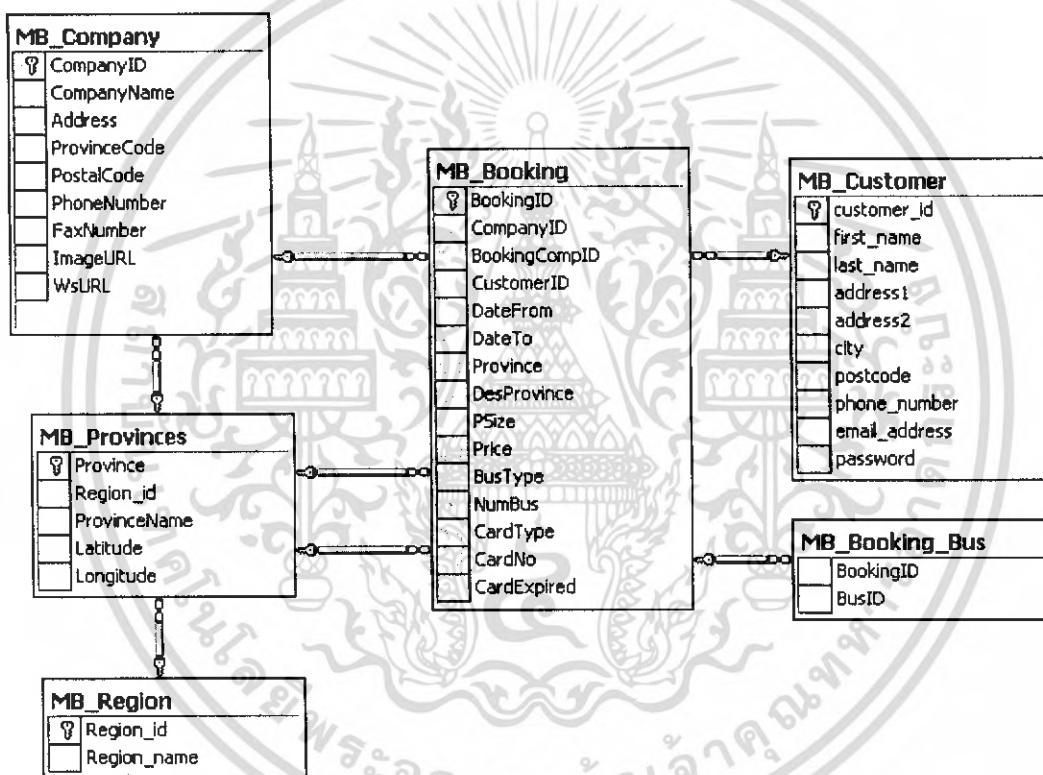


รูป 3.7 แสดง Restaurant Database Model Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

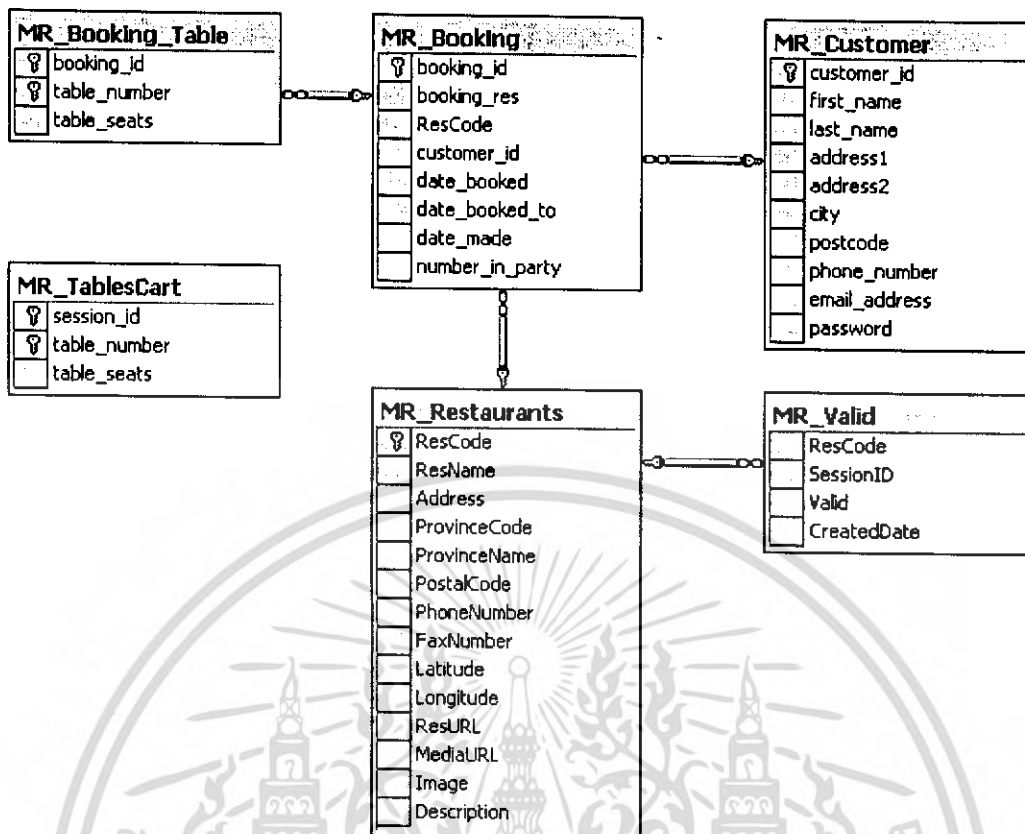


รูป 3.8 แสดง HotelCenter Database Model Diagram



รูป 3.9 แสดง BusCenter Database Model Diagram

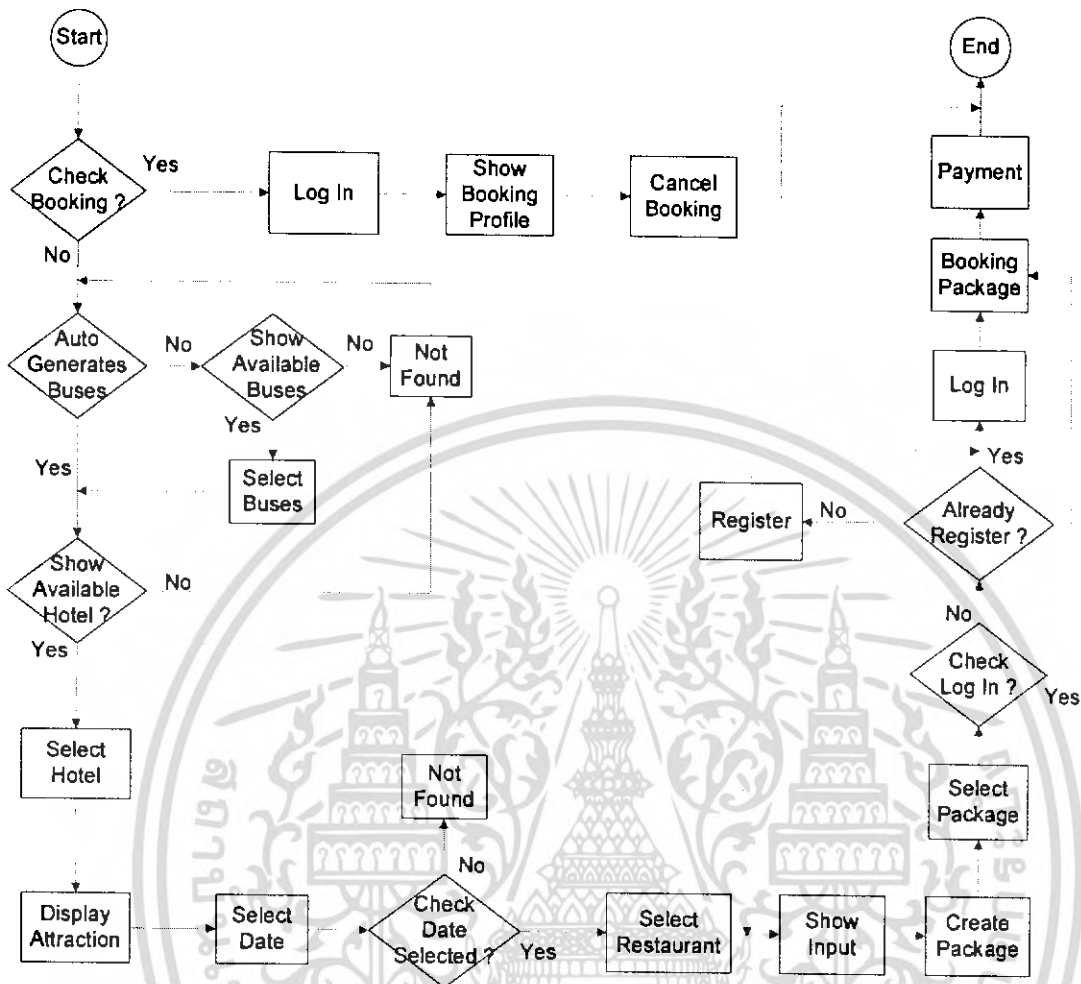
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.10 แสดง RestaurantCenter Database Model Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Flowchart



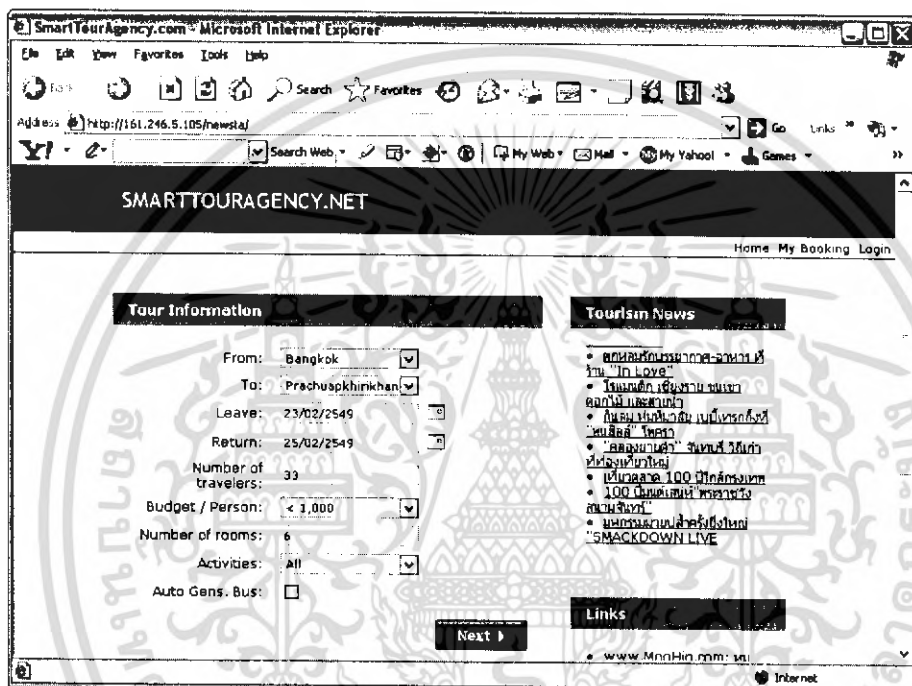
รูป 3.11 แสดงกระบวนการการจัดสร้างแพคเกจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### การทดลองและผลการทดลอง

ผู้ใช้ระบบสามารถทำการสร้างแพคเกจทัวร์ขึ้นได้เอง โดยผ่านทางเว็บเซอร์วิสของ Smart Tour Agency ซึ่งเป็นตัวแทนที่ทำการติดต่อและจัดการจองกับการเดินทาง ที่พักและร้านอาหารให้แทนผู้ใช้ระบบ โดยที่ระบบของ Smart Tour Agency นั้น จะทำการแสดงและเลือกติดต่อกับเซิร์ฟเวอร์ที่เปิดให้บริการตามข้อมูลที่ใช้ได้กรอกเข้ามา ดังรูป



รูป 4.1 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้ระบบทำการกรอกข้อมูลเข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

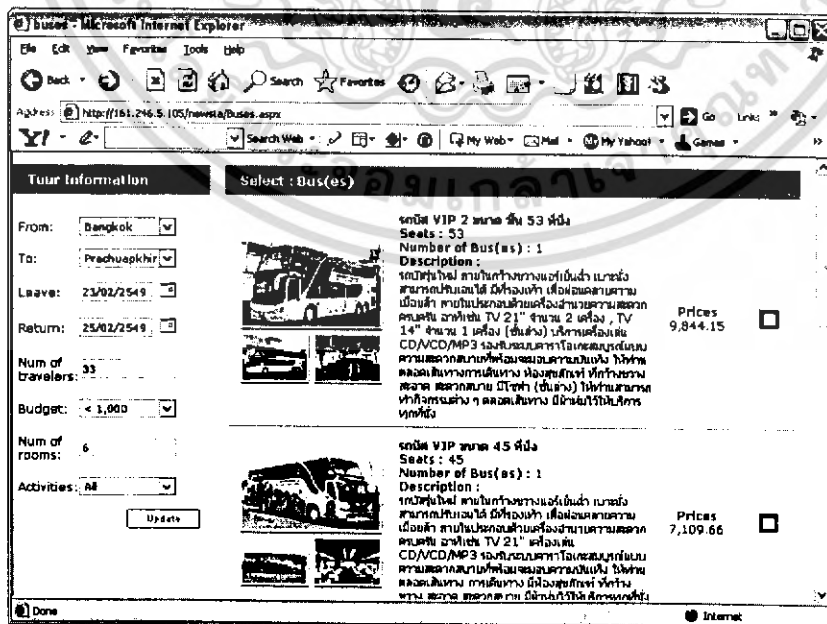
#### 4.1. การติดต่อแสดงข้อมูลการจองผ่านทางเว็บไซต์ของ Smart Tour Agency

##### 4.1.1. การติดต่อแสดงข้อมูลการจองรถโดยสาร

เว็บไซต์ Smart Tour Agency ทำการร้องขอข้อมูลจากเว็บไซต์ที่ได้ติดต่ออยู่ เพื่อนำมาแสดงข้อมูลรายละเอียดของรถที่เป็นไปได้ โดยที่นำข้อมูลของผู้ใช้ระบบที่ได้กรอกไว้จากหน้าแรกไว้แล้วนำมาเลือกในการแสดงชนิดของรถที่สามารถทำการจองได้ ซึ่งขึ้นอยู่กับระยะเวลาเดินทางจากไปและกลับ, ช่วงเวลาที่ใช้ในการเดินทาง, จำนวนคนของคณะทัวร์ และงบประมาณต่อคนของคณะทัวร์



รูป 4.2 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลในการให้บริการในการจองรถที่เป็นไปได้

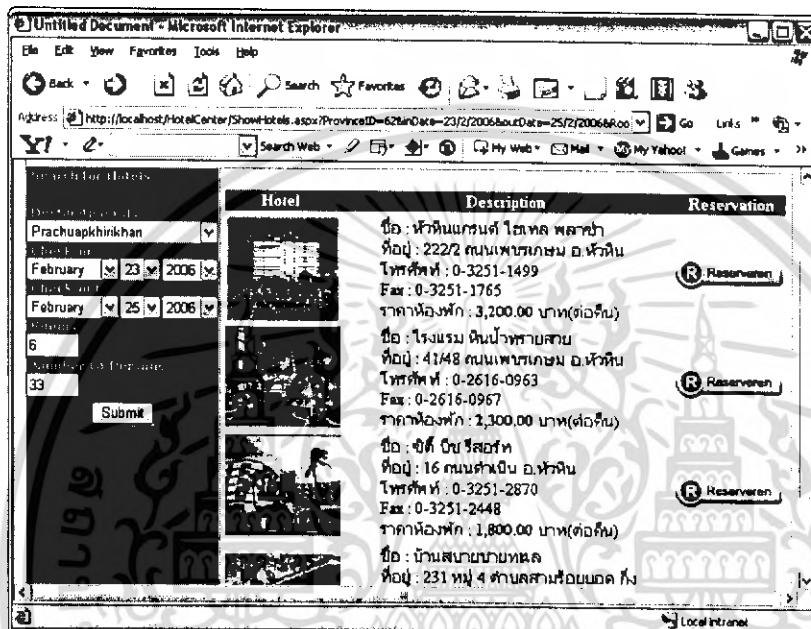


รูป 4.3 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลของรถที่เป็นไปได้ตามที่ผู้ใช้ระบบทำการกรอกข้อมูลเข้ามา

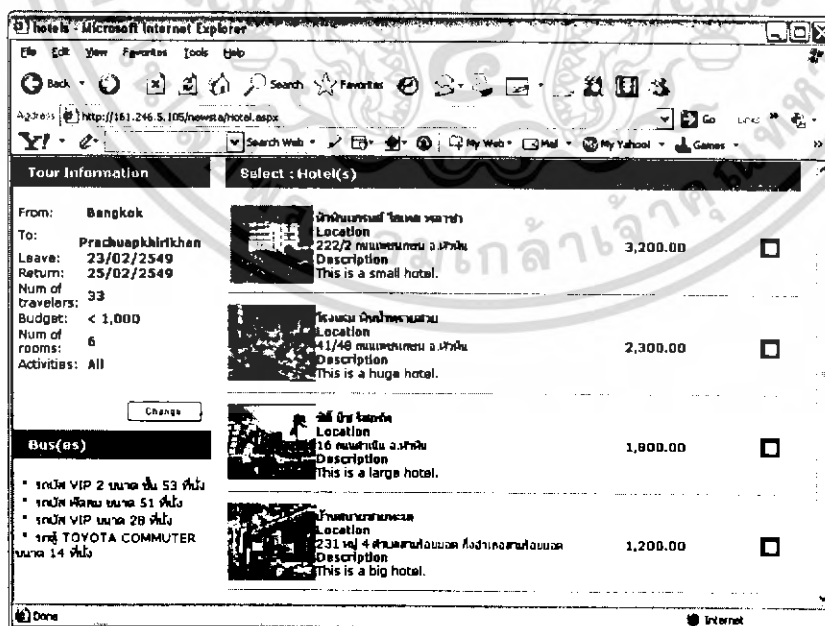
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.2. การติดต่อแสดงข้อมูลการจองห้องพัก

เว็บ Smart Tour Agency ทำการร้องขอข้อมูลจากเว็บเซิร์ฟเวอร์ที่ได้ติดต่ออยู่ เพื่อนำมาแสดงข้อมูลรายละเอียดของ โรงแรมที่เป็นไปได้ โดยที่นำข้อมูลของผู้ใช้ระบบที่ได้กรอกไว้จากหน้าแรกไว้แล้วนำมาเลือกในการแสดงโรงแรมที่สามารถทำการจองได้ ซึ่งขึ้นอยู่กับจังหวัดปลายทางที่ไปพักผ่อน, วันที่จะทำการเข้าพักและออกจากที่พัก, จำนวนห้องทั้งหมดของคณะทัวร์ และงบประมาณต่อคนของคณะทัวร์



รูป 4.4 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูล ในการให้บริการ ในการจอง โรงแรมที่เป็นไปได้

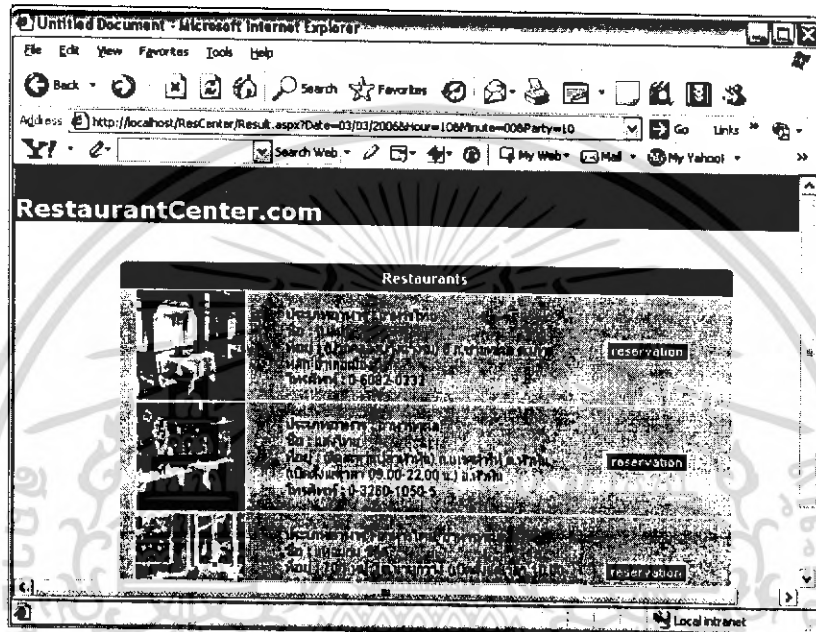


รูป 4.5 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลของ โรงแรมที่เป็นไปได้ตามที่  
ผู้ใช้ระบบทำการกรอกข้อมูลเข้ามา

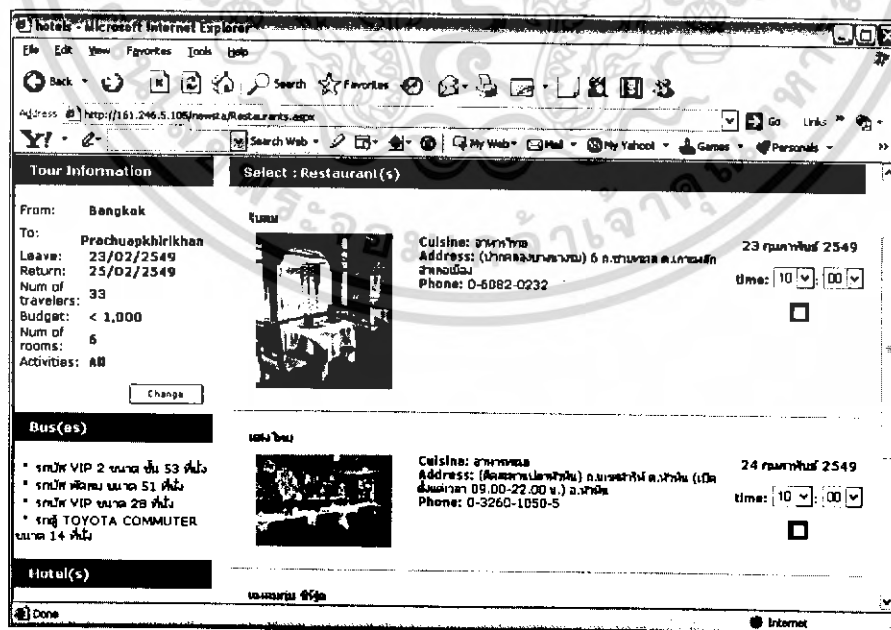
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.3. การติดต่อแสดงข้อมูลการจองร้านอาหาร

เว็บ Smart Tour Agency ทำการร้องขอข้อมูลจากเว็บเซอร์วิสที่ได้ติดต่ออยู่ เพื่อนำมาแสดงข้อมูลรายละเอียดของร้านอาหารที่เป็นไปได้ โดยที่นำข้อมูลของผู้ใช้ระบบที่ได้กรอกไว้จากหน้าแรกไว้แล้วและสถานที่ท่องเที่ยวที่ผู้ใช้ระบบต้องการไปในแต่ละวัน นำมาเลือกในการแสดงร้านอาหารที่ใกล้ที่สุดและเหมาะสมที่ใช้ในการท่องเที่ยวแต่ละวัน ซึ่งขึ้นอยู่กับ จังหวัดปลายทางที่ไปพักผ่อน, วันและเวลาที่จองโต๊ะอาหาร และจำนวนคนของคณะทัวร์



รูป 4.6 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลในการให้บริการ ในการจองร้านอาหารที่เป็นไปได้

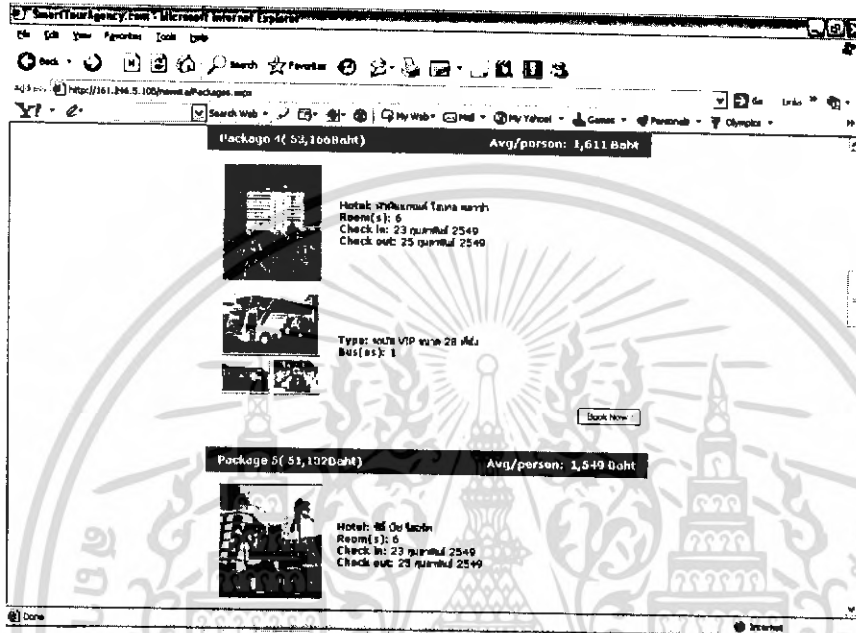


รูป 4.7 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลของโรงแรมที่เป็นไปได้ตามที่ผู้ใช้ระบบทำการกรอกข้อมูลเข้ามา

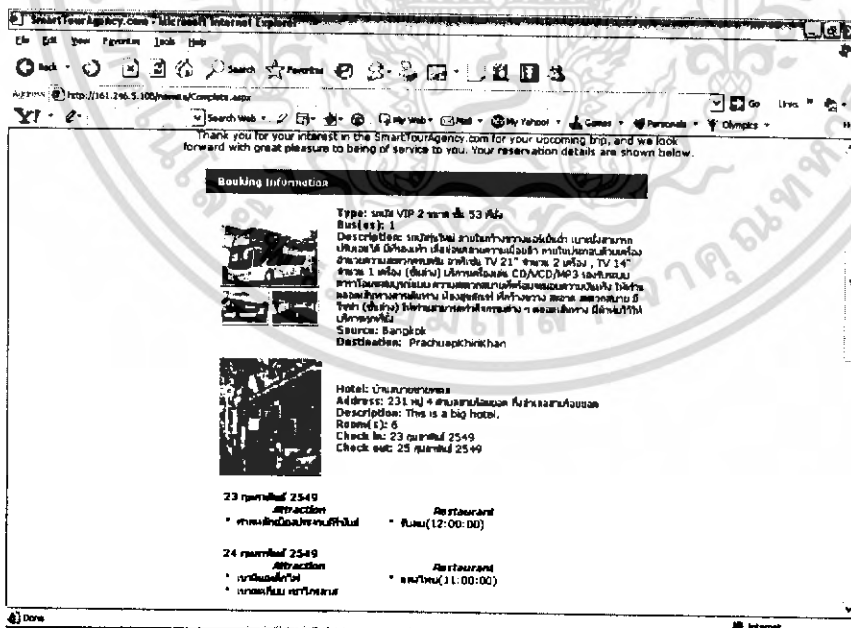
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2. การติดต่อการจองผ่านทางเว็บไซต์ของ Smart Tour Agency

เมื่อผู้ใช้ระบบผ่านการเลือกรถเดินทาง, ที่พัก และร้านอาหาร ระบบก็จะทำการสร้างแพ็คเกจทัวร์ขึ้นมา ซึ่งแพ็คเกจทัวร์จะถูกสร้างขึ้นในหลายๆลักษณะ ตามที่ผู้ใช้ระบบได้ทำการเลือกไว้ข้างต้น เมื่อผู้ใช้ระบบทำการเลือกแพ็คเกจทัวร์จากที่ได้แสดงไว้มาอันหนึ่ง ระบบก็จะทำการติดต่อจองกับรถเดินทาง, ที่พัก และร้านอาหารตามที่ได้แสดงไว้ในแพ็คเกจทัวร์



รูป 4.8 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงแพ็คเกจทัวร์ที่เป็นไปได้ตามที่ผู้ใช้ระบบทำการกรอกข้อมูลเข้ามา



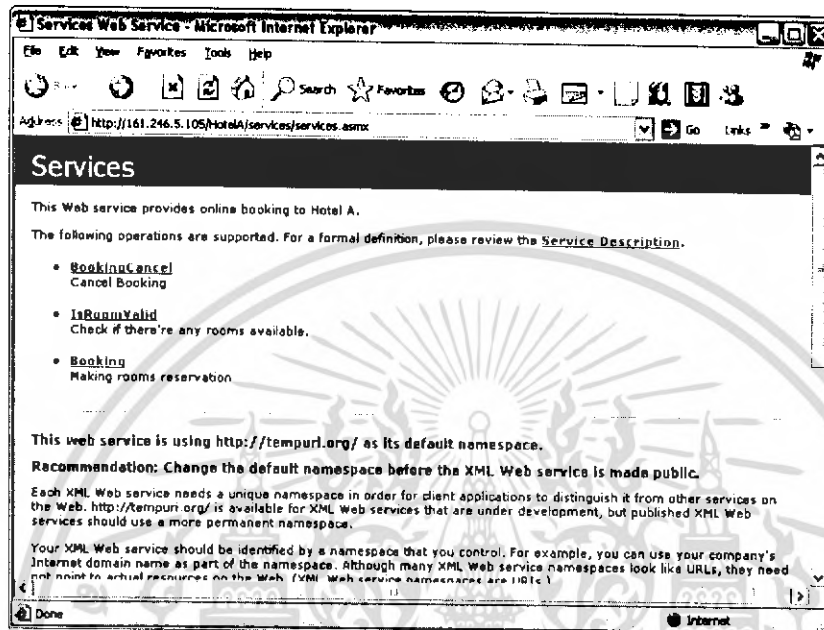
รูป 4.9 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงแพ็คเกจทัวร์ที่ได้ทำการจองไว้เรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.3. เซอร์วิสที่ได้เปิดให้ใช้บริการกับเว็บแอปพลิเคชัน

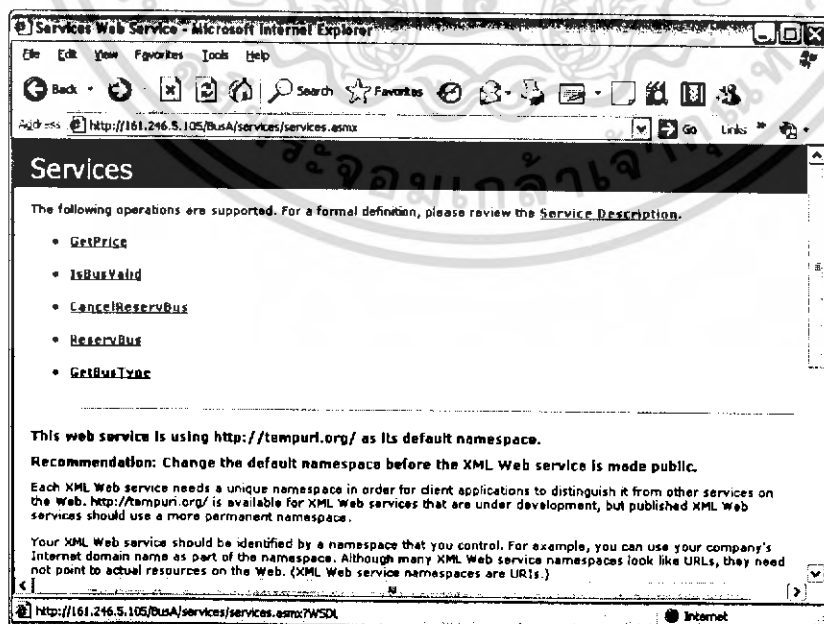
เป็นฟังก์ชันที่สร้างขึ้นเพื่อเปิดให้มีการใช้บริการจากหน้าเว็บแอปพลิเคชันภายนอก โดยที่เว็บแอปพลิเคชันทำการส่งข้อมูลของผู้ใช้บริการเข้ามาไว้ในฟังก์ชัน แล้วเซิร์ฟเวอร์ที่เปิดให้บริการก็จะทำการประมวลผลและส่งผลกลับไปยังเว็บแอปพลิเคชัน

##### 4.3.1. เซอร์วิสที่เซิร์ฟเวอร์ได้เปิดให้ใช้บริการกับเว็บเซอร์วิสตัวกลาง



รูป 4.10 ภาพแสดงตัวอย่างเซอร์วิสของเซิร์ฟเวอร์โรงแรมที่เปิดให้เว็บเซอร์วิสเรียกใช้

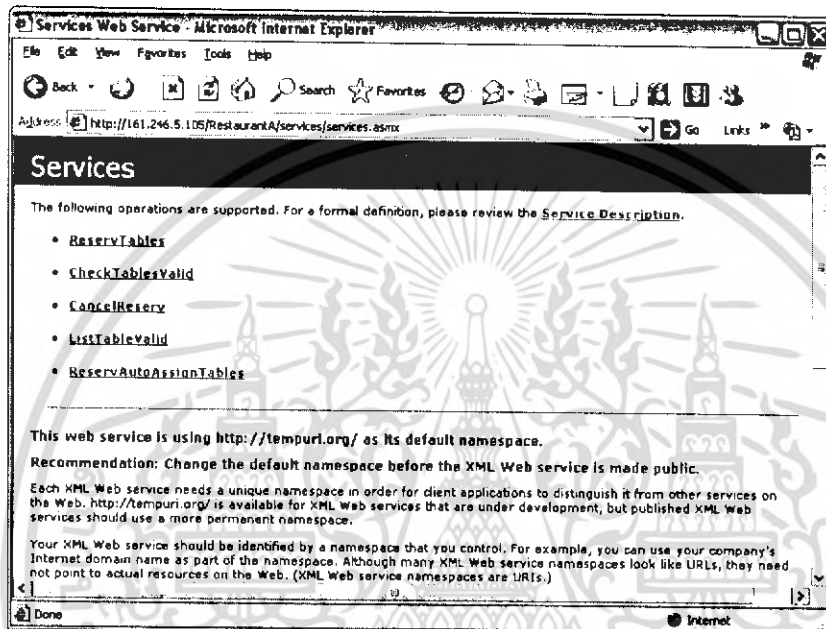
- BookingCancel – ยกเลิกการจองห้องพักที่ได้จองไว้ล่วงหน้า
- IsRoomValid – ตรวจสอบสถานะห้องพักว่าว่างตามช่วงวันที่กำหนดหรือไม่
- Booking – จองห้องพักตามช่วงวันที่กำหนด



รูป 4.11 ภาพแสดงตัวอย่างเซอร์วิสของเซิร์ฟเวอร์รถเดินทางที่เปิดให้เว็บเซอร์วิสเรียกใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- GetPrice – ตรวจสอบราคาเช่าซึ่งขึ้นอยู่กับชนิดของรถ จำนวนรถ และระยะทาง
- IsBusValid – ตรวจสอบรถว่ามีพอดตามเวลาที่กำหนดหรือไม่
- CancelReservBus – ยกเลิกการจองที่ได้จองไว้ล่วงหน้า
- ReservBus – จองรถตามเวลาที่กำหนด
- GetBusType – แสดงรายการชนิดของรถทั้งหมด

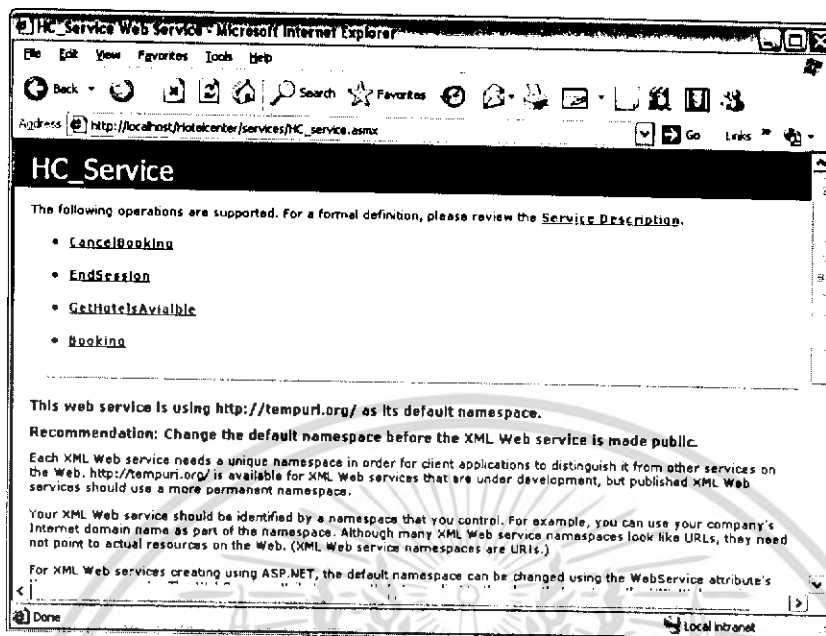


รูป 4.12 ภาพแสดงตัวอย่างเซอร์วิสของเซิร์ฟเวอร์ร้านอาหารที่เปิดให้เว็บเซอร์วิสเรียกใช้

- ReservTables – จองโต๊ะอาหารที่ได้เลือกไว้ตามเวลาที่กำหนด
- CheckTablesValid – ตรวจสอบจำนวนโต๊ะที่ว่างตามเวลาที่กำหนดหรือไม่
- CancelReserv – ยกเลิกการจองโต๊ะที่ได้จองไว้ล่วงหน้า
- ListTableValid – แสดงรายการของโต๊ะที่ว่างตามเวลาที่กำหนด
- ReservAutoAssignTables – จองโต๊ะตามเวลาที่กำหนดโดยมีการเลือกโต๊ะให้อัตโนมัติ

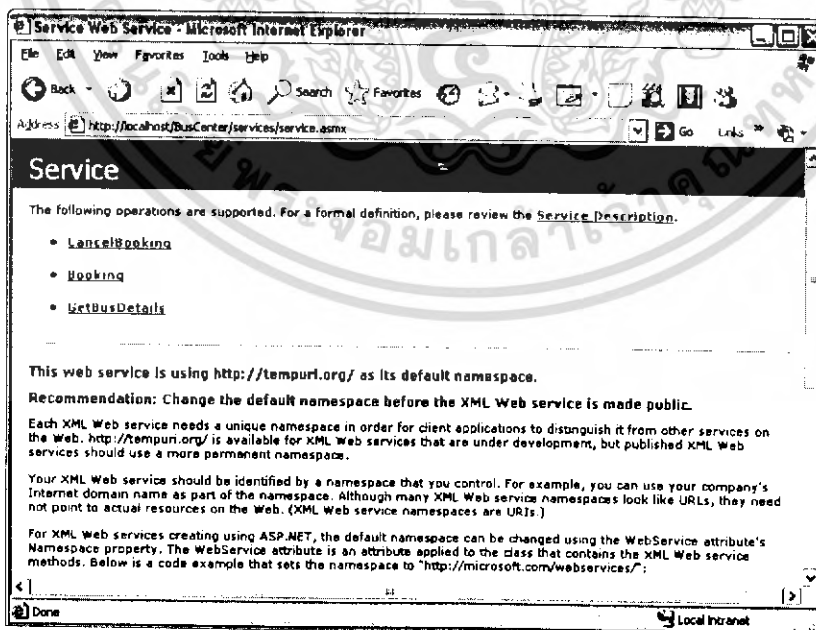
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.3.2. เซอร์วิสที่เว็บเซอร์วิสเปิดให้บริการกับเว็บ Smart Tour Agency



รูป 4.13 ภาพแสดงตัวอย่างเซอร์วิสของเว็บเซอร์วิสโรงแรมที่เปิดให้เว็บเซอร์วิส Smart Tour Agency เรียกใช้

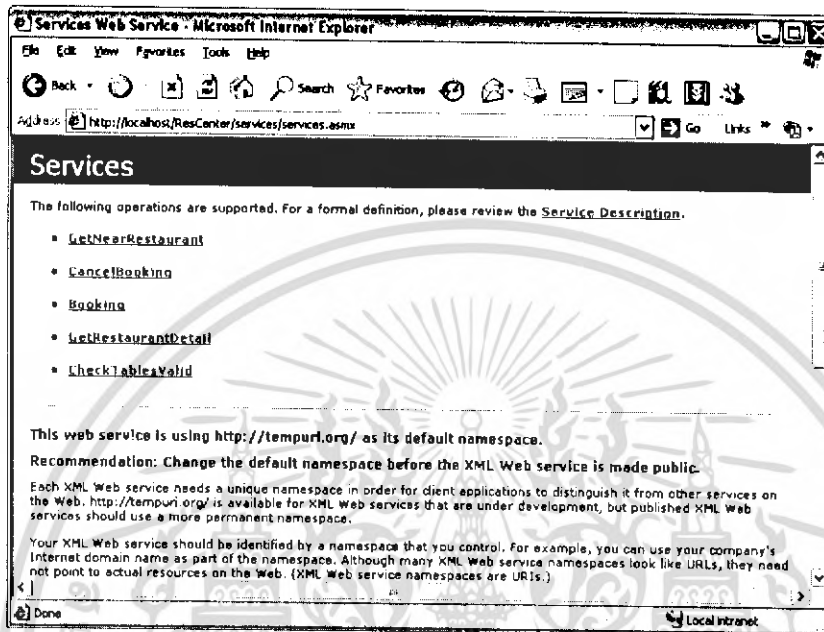
- CancelBooking – ยกเลิกการจองที่ได้จองไว้ล่วงหน้า
- EndSession – ทำลาย Session ที่มีการเรียกใช้ผ่านเว็บเซอร์วิส
- GetHotelsAvialble – แสดงรายการโรงแรมที่มีจำนวนห้องพักว่างตามช่วงวันที่กำหนด
- Booking - จองห้องพักของ โรงแรมที่เลือกตามช่วงวันที่กำหนด



รูป 4.14 ภาพแสดงตัวอย่างเซอร์วิสของเว็บเซอร์วิสรถเดินทางที่เปิดให้เว็บเซอร์วิส Smart Tour Agency เรียกใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- CancelBooking – ยกเลิกการจองรถที่ได้จองไว้ล่วงหน้า
- Booking – จองรถกับบริษัทที่เลือกในช่วงวันที่กำหนด
- GetBusDetails – แสดงชนิดของรถที่ว่างตาม บริษัท จำนวนคัน และช่วงวันที่กำหนด



รูป 4.15 ภาพแสดงตัวอย่างเซอร์วิสของเว็บเซอร์วิสร้านอาหารที่เปิดให้เว็บเซอร์วิส Smart Tour Agency เรียกใช้

- GetNearRestaurant – แสดงร้านอาหารที่ใกล้กับค่า Latitude และ Longitude ที่กำหนดมากที่สุด
- CancelBooking – ยกเลิกการจองร้านอาหารที่ได้จองไว้ล่วงหน้าแล้ว
- Booking – จองโต๊ะกับร้านอาหารที่เลือกตามช่วงเวลาที่กำหนด
- GetRestaurantDetail – แสดงรายการของร้านอาหารทั้งหมดที่มีอยู่ในจังหวัดที่กำหนด
- CheckTableValid - ตรวจสอบจำนวนโต๊ะของร้านค้าที่ระบุว่าว่างตามช่วงเวลาที่กำหนดหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4. ขั้นตอนการทำงานของระบบ

##### 4.4.1. กรอกข้อมูลของผู้ใช้เข้าสู่ระบบ

กรอกข้อมูลการเดินทางของผู้ใช้ระบบ ประกอบด้วย จังหวัดที่ออกเดินทาง, จังหวัดจุดหมายปลายทาง, วันที่ออกเดินทาง, วันที่เดินทางกลับ, จำนวนของคณะทัวร์, งบประมาณการเดินทางต่อบุคคล, จำนวนห้องพักที่ต้องการ, ประเภทของกิจกรรมในการเดินทาง และความประสงค์ที่ต้องการจะเลือกรถด้วยตนเองเป็นอันดับแรก

The screenshot shows a web browser window with the URL <http://161.246.5.105/newsta/>. The page title is "SmartTourAgency.com - Microsoft Internet Explorer". The main content area is titled "Tour Information" and contains the following form elements:

- From:** Select dropdown menu
- To:** Select dropdown menu
- Leave:** Calendar icon
- Return:** Calendar icon
- Number of travelers:** Input field
- Budget / Person:** Input field with a dropdown menu showing "< 1,000"
- Number of rooms:** Input field
- Activities:** Dropdown menu showing "All"
- Auto Gens. Bus:** Checkbox
- Next** button

On the right side, there is a "Tourism News" section with a link to "SMACKDOWN LIVE" and a "Links" section with several links:

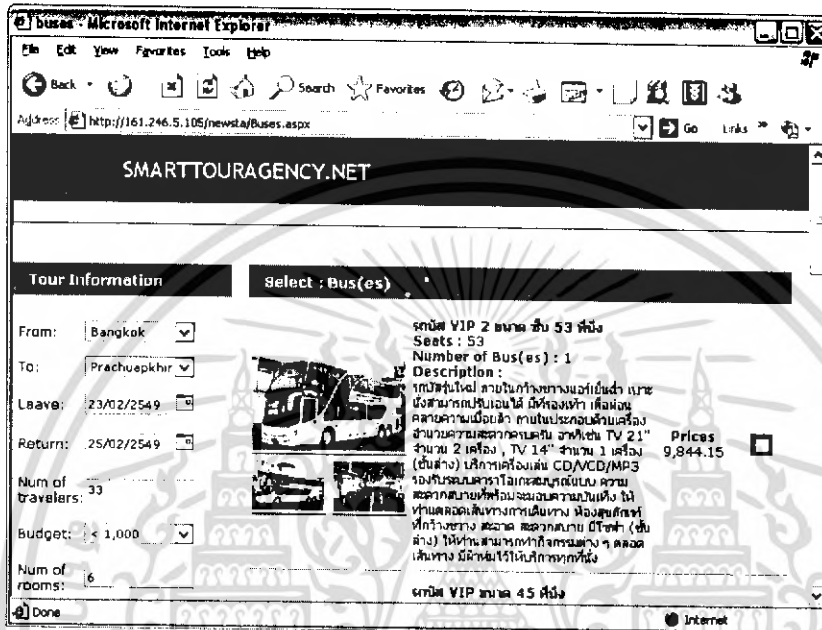
- [www.Machin.com](http://www.Machin.com)
- ...
- ...
- ...

รูป 4.16 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้กรอกข้อมูลเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.2. เลือกประเภทของรถที่จะใช้ในการเดินทาง

ผู้ใช้งานทำการเลือกชนิดของรถที่ใช้ในการเดินทางครั้งนี้ โดยที่ระบบจะทำการแสดงรถโดยสารที่เพียงพอที่จะรองรับกับจำนวนคนของคณะทัวร์ และผู้ใช้งานระบบนั้นสามารถเลือกชนิดของรถโดยสารได้มากกว่า 1 ชนิด เพื่อให้เกิดเป็นแพคเกจทัวร์ที่หลากหลายในขั้นตอนสุดท้าย

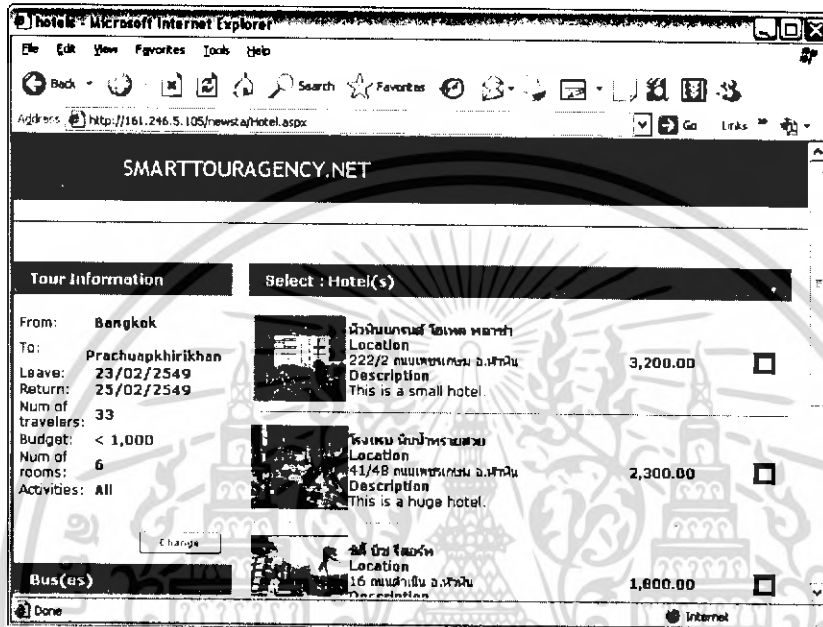


รูป 4.17 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้งานทำการเลือกชนิดของรถโดยสารที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.3. เลือกโรงแรมที่จะใช้เป็นที่พักในการเดินทาง

ผู้ใช้งานทำการเลือกโรงแรมที่ใช้เป็นที่พักในการเดินทางครั้งนี้ โดยที่ระบบจะทำการแสดงโรงแรมที่มีที่พักเพียงพอที่จะรองรับกับจำนวนคนของคณะทัวร์ และผู้ใช้งานระบบนั้นสามารถเลือกโรงแรมได้มากกว่า 1 แห่ง เพื่อให้เกิดเป็นแพคเกจทัวร์ที่หลากหลายในขั้นตอนสุดท้ายเช่นกัน

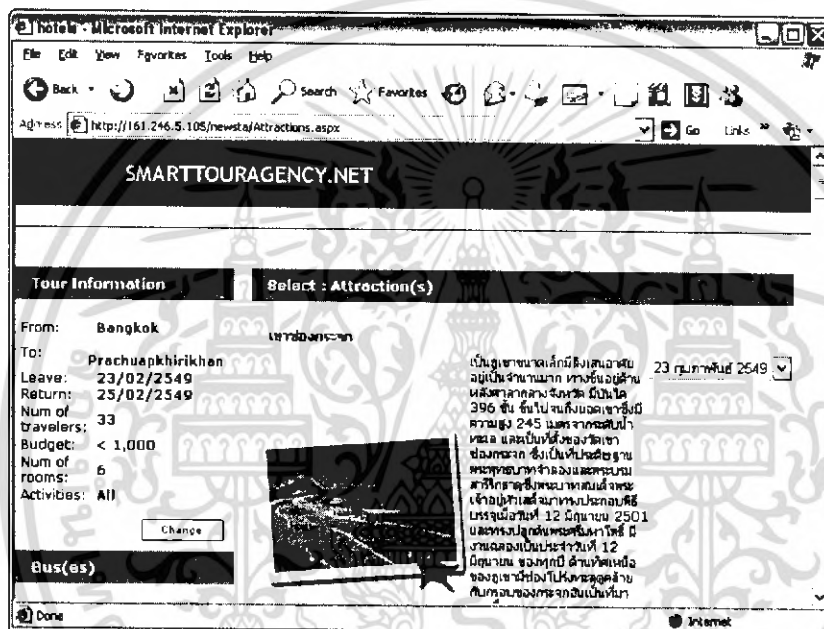


รูป 4.18 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้งานทำการเลือกโรงแรมที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.4. เลือกสถานที่ท่องเที่ยวที่ต้องการในแต่ละวัน

ผู้ใช้ระบบจะสามารถเลือกสถานที่ท่องเที่ยวในแต่ละวันได้ด้วยตนเอง เพื่อที่เป็นกำหนดการเดินทางให้กับแพคเกจทัวร์ ซึ่งผู้ใช้ระบบสามารถเลือกหรือไม่เลือกสถานที่ท่องเที่ยวในแต่ละวันก็ได้ โดยที่สถานที่ท่องเที่ยววันนั้นจะส่งผลไปถึงร้านอาหารที่จะเลือกในหน้าถัดไป เพื่อที่จะได้ไม่เป็นการเสียเวลาในการเดินทางไปสถานที่ท่องเที่ยวแต่ละวัน แต่ถ้าวันไหนไม่ได้มีการเลือกสถานที่ท่องเที่ยวในหน้าถัดไปที่จะมีร้านอาหารให้จองก็จะไม่ปรากฏร้านอาหารให้เลือกในวันนั้นเช่นกัน โดยถือว่าเป็นวันที่อิสระให้คณะท่องเที่ยวสามารถเลือกได้เองว่าจะไปไหน

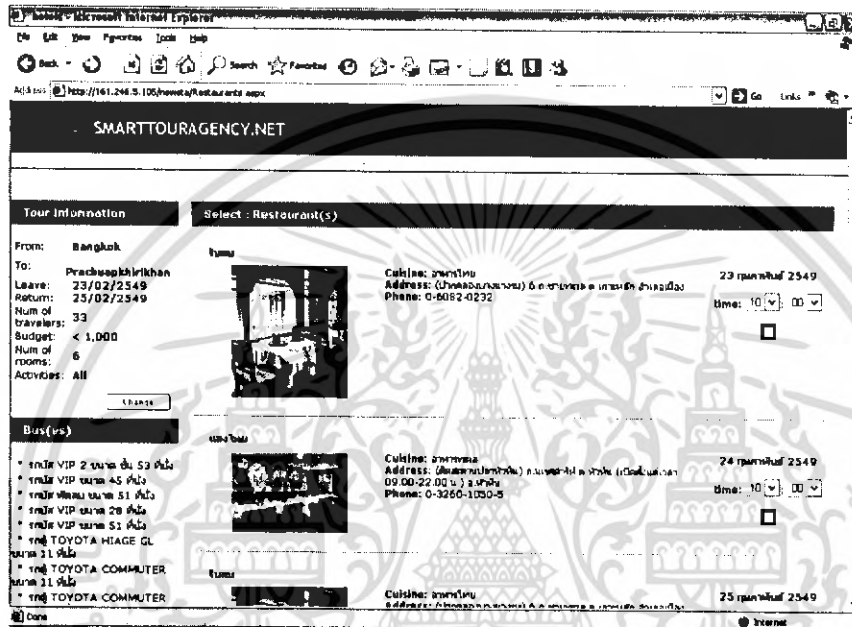


รูป 4.19 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้จะทำการเลือกสถานที่ท่องเที่ยวในแต่ละวัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.5. เลือกร้านอาหารที่ต้องการในแต่ละวัน

ผู้ใช้งานจะสามารถเลือกเวลาที่เข้าไปรับประทานอาหารของร้านอาหารที่เลือกไว้ที่ได้มาจากการที่ผู้ใช้งานทำการเลือกสถานที่ท่องเที่ยวไว้ที่หน้าก่อนหน้านี้ ซึ่งผู้ใช้นั้นสามารถจะเลือกหรือไม่เลือกก็ได้ โดยถ้าผู้ใช้งานเลือกที่ของร้านอาหารทางระบบก็จะทำการจองร้านอาหารไว้ให้ แต่ถ้าไม่ทำการเลือกในวันใดๆก็ถือว่าเป็นอิสระในการที่ผู้ใช้งานจะเลือกร้านอาหารแต่จะไม่มีการจองไว้ให้



รูป 4.20 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้งานทำการเลือกร้านอาหารในแต่ละวัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



#### 4.4.7. เลือกแพ็คเกจทัวร์ที่ระบบได้สร้างขึ้น



ระบบจะทำการสร้างแพ็คเกจทัวร์ในหลายๆลักษณะตามที่ผู้ใช้ระบบได้ป้อนข้อมูลเข้ามา ซึ่งจะบอกถึงโรงแรมที่พัก, จำนวนห้อง, วันเช็คอิน-เช็คเอาท์, ชนิดของรถโดยสาร, จำนวนคัน, ราคารวมของแพ็คเกจ และราคาต่อบุคคล

**SMARTTOURAGENCY.NET**

---

Package 1 (67,932 Baht)

Avg/person: 2,059 Baht



**Hotel:** ฟ้าดินแอนด์ โฮเทล หลานา  
**Room(s):** 6  
**Check In:** 23 กุมภาพันธ์ 2549  
**Check out:** 25 กุมภาพันธ์ 2549

**Type:** รถตู้ VIP 2 ขนาด ขึ้น 53 ที่นั่ง  
**Bus(es):** 1

[Book Now!](#)

Package 2 (61,370 Baht)

Avg/person: 1,860 Baht

**Hotel:** ฟ้าดินแอนด์ โฮเทล หลานา  
**Room(s):** 6  
**Check In:** 23 กุมภาพันธ์ 2549  
**Check out:** 25 กุมภาพันธ์ 2549

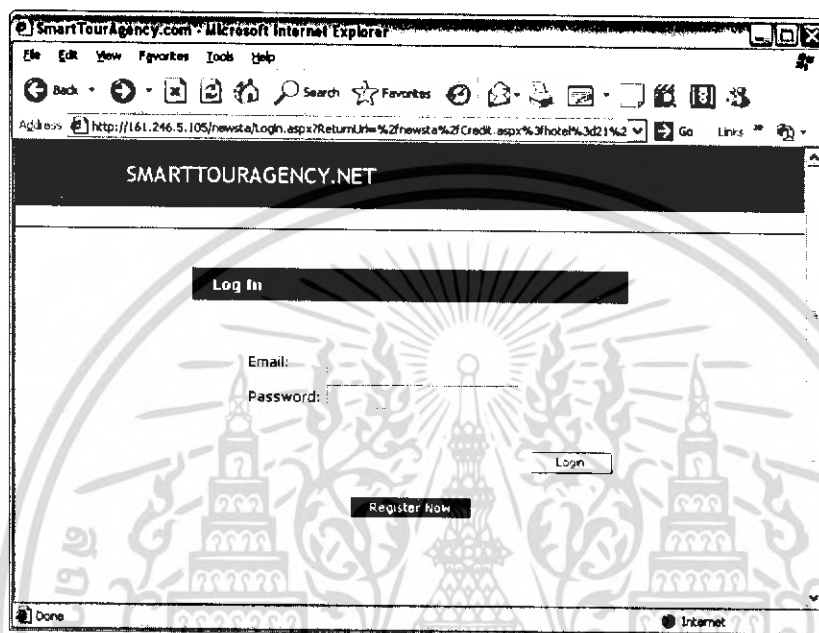
**Type:** รถตู้ VIP ขนาด 51 ที่นั่ง  
**Bus(es):** 1

รูป 4.22 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงแพ็คเกจให้ผู้ใช้ทำการเลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.8. ทำการเข้าสู่ระบบ

หากผู้ใช้ระบบได้ทำการเลือกแพคเกจที่ต้องการได้แล้ว แต่ยังไม่สามารถล็อกอินก็จะเข้าสู่หน้านี้ เพื่อทำการเข้าสู่ระบบ โดยจะให้ผู้ใช้กรอกอีเมลแอดเดรสและพาสเวิร์ด หากผู้ใช้ระบบนั้นยังไม่ได้ทำการลงทะเบียนกับระบบ ก็จะสามารถคลิกไปยังหน้าลงทะเบียนได้



รูป 4.23 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ให้ผู้ใช้งานเข้าสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.9. ทำการลงทะเบียนลงสู่ระบบ

ให้ผู้ใช้กรอกข้อมูลส่วนตัวลงสู่ระบบ เพื่อที่จะเป็นสมาชิกกับระบบ โดยส่วนที่ต้องกรอกข้อมูลนั้น ประกอบด้วย ชื่อ, นามสกุล, ที่อยู่, จังหวัด, รหัสไปรษณีย์, เบอร์โทรศัพท์, อีเมลล์แอดเดรส และพาสเวิร์ด

The screenshot shows a web browser window displaying the 'SMARTTOURAGENCY.NET' website. The main content area features a 'Customer Registration' form with the following fields: First Name, Last Name, Address1, Address2, City, Postal, Phone Number, Email Address, Password, and Confirm Password. A 'Submit' button is located at the bottom right of the form. The background of the page has a large, faint watermark of the Thai national emblem.

รูป 4.24 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ผู้ใช้ทำการลงทะเบียนสู่ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.10. กรอกข้อมูลที่ต้องใช้ชำระเงิน

ระบบจะให้ผู้ใช้ระบบทำการกรอกชนิดของบัตรเครดิต, หมายเลขบัตรเครดิต, ชื่อผู้ถือบัตร และวันหมดอายุของบัตรเครดิต เพื่อทำการชำระเงินค่าแพคเกจเป็นราคา 30% เพื่อเป็นค้ำมัดจำหากมีการยกเลิกการจองโดยไม่ได้แจ้งไว้ล่วงหน้าเป็นเวลา 2 วันก่อนถึงวันตามแพคเกจ

The screenshot shows a web browser window with the address bar displaying "http://161.246.5.105/newsta/Credit.aspx?hotel=21&comp=2&bbtype=4". The page title is "SMARTTOURAGENCY.NET". The main content area contains the following text:

Policy  
30% OF TOTAL AMOUNT WILL BE CHARGED IMMEDIATELY AND THE REMAIN SHOULD BE PAID WITHIN THE CANCELLATION PERIOD.  
Cancellation Policy and Important Notice  
A CANCELLATION FEE DEPENDS ON THE DATE OF YOUR CHANGE/CANCELLATION NOTICE. IF THE DATE OF YOUR CHANGE/CANCELLATION NOTICE FALLS WITHIN THE CANCELLATION PERIOD, YOU WILL BE CHARGED THE CANCELLATION FEES.  
ON THE DATE AFTER CANCELLATION PERIOD: 100% OF TOTAL AMOUNT CANCELED WILL BE CHARGED.

\*Credit Card Information A valid credit card is required  
A valid credit card is required.

Credit Card  
Select [dropdown menu]  
Card Number [input field]

รูป 4.25 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ให้ผู้ใช้งานชำระค่าใช้จ่ายผ่านบัตรเครดิต

#### 4.4.11. แสดงผลสรุปทั้งหมดของการจอง

ระบบจะทำการแสดงผลของการจองทั้งหมดออกมาในรูปแบบแพคเกจตามที่ผู้ใช้ระบบได้เลือกไว้ และยังแสดงสถานที่ท่องเที่ยว, ร้านอาหารที่ได้เลือกไว้ในแต่ละวันด้วย ซึ่งบอกถึงข้อมูลของผู้ที่เข้ามาใช้บริการระบบด้วย

## Complete

Thank you for your interest in the SmartTourAgency.com for your upcoming trip, and we look forward with great pleasure to being of service to you. Your reservation details are shown below.

## Booking Information



Type: รถ VIP ขนาด 28 ที่นั่ง  
 Bus(es): 1  
 Description: รถ VIP ใหม่ ตามใน枉วังทางแฉ่งเป็นฉ่า แฉ่งวังสามารถ  
 เป็นใช้ได้ เพื่อไม่เปลืองเวลาความเมือฉ่า สายไปปะหลกบฉิ่งฉิ่งฉ่า  
 ความสะดวกครบถ้วน อาทิเช่น TV 14" เครื่องเล่น บริการเครื่องเล่น  
 CD/VCD/MP3 รองรับระบบคาราโอเกะพร้อมที่นั่งที่ฉิ่งฉิ่งฉิ่งฉิ่ง  
 ปันฉิ่ง ให้ท่านพักผ่อนสบาย มีฉิ่งฉิ่งให้บริการ ทุกที่ฉิ่ง  
 Source: Bangkok  
 Destination: Prachuapkhirikhan



Hotel: โรงแรมโรงแรม  
 Address: 16 ถนนฉิ่งฉิ่ง ฉิ่งฉิ่ง  
 Description: This is a large hotel.  
 Room(s): 6  
 Check In: 23 กุมภาพันธ์ 2549  
 Check out: 25 กุมภาพันธ์ 2549

23 กุมภาพันธ์ 2549

Attraction

▪ เขาช่องกระเจก

Restaurant

▪ ภัตตาคาร(14:00)

24 กุมภาพันธ์ 2549

Attraction

▪ เขาดินเหล็กไฟ  
 ▪ เขามะลิฉิ่ง เขาศิลาฉิ่ง

Restaurant

▪ ภัตตาคาร(12:00)

25 กุมภาพันธ์ 2549

Attraction

▪ เขาช่องฉิ่ง  
 ▪ อุทยานแห่งชาติเขาสฉิ่งฉิ่ง

Restaurant

▪ ภัตตาคาร(13:00)

## Customer Information

First Name: Nuttapon  
 Last Name: Khewprasert  
 Address1: 112/5 mu3 Khomgsampravat  
 Address2: Ladkrabang  
 City: Bangkok  
 Postal: 10540  
 Phone Number: 09 783 192  
 Email Address: nut\_cd32@hotmail.com

## Cancellation Policy

- o To avoid the cancellation penalties, Reservation must be cancelled 2 days prior to arrival date.
- o Late cancellation will be charged at 1 night fee to credit card.
- o No-show and early departure will be charged at 1 night fee to credit card.

We reserve the right to cancel or modify reservation where it appears that a customer has engaged in fraudulent or inappropriate activity or under other circumstances where it appears that the reservation contains or is resulted from a mistake or through the provision of wrongful information.

By Confirmation of this reservation, you acknowledge that you have read and agreed to the guarantee and cancellation policy associated with this reservation.

## Credit Card

Credit Card Type: Visa  
 Card Number: 123-123-123-123  
 Expire: 3/2006

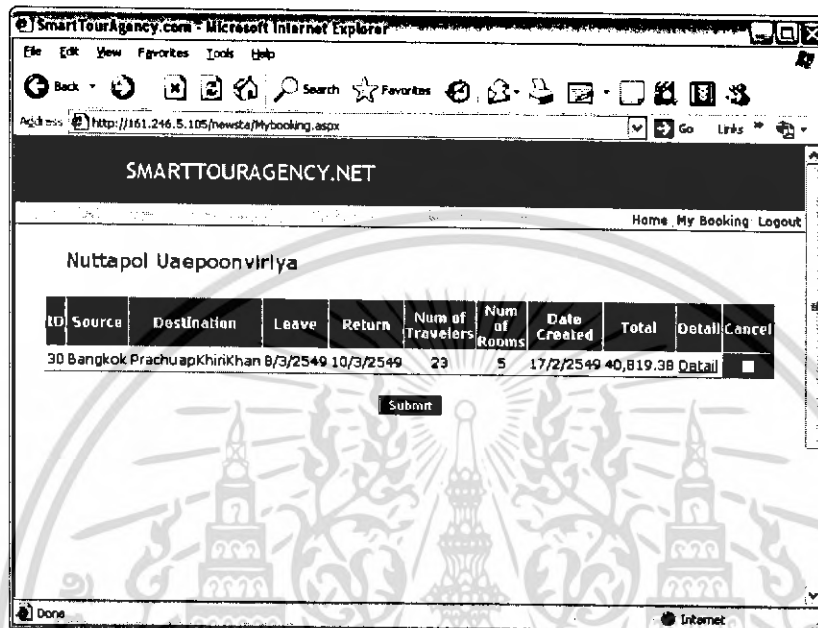
We look forward to sincerely welcoming you soon to the Hotel Serv.com, for further assistance, please do not hesitate to contact us.

รูป 4.26 ภาพแสดงตัวอย่างหน้าเว็บเพจที่ระบบแสดงผลการจองให้กับผู้ใช้

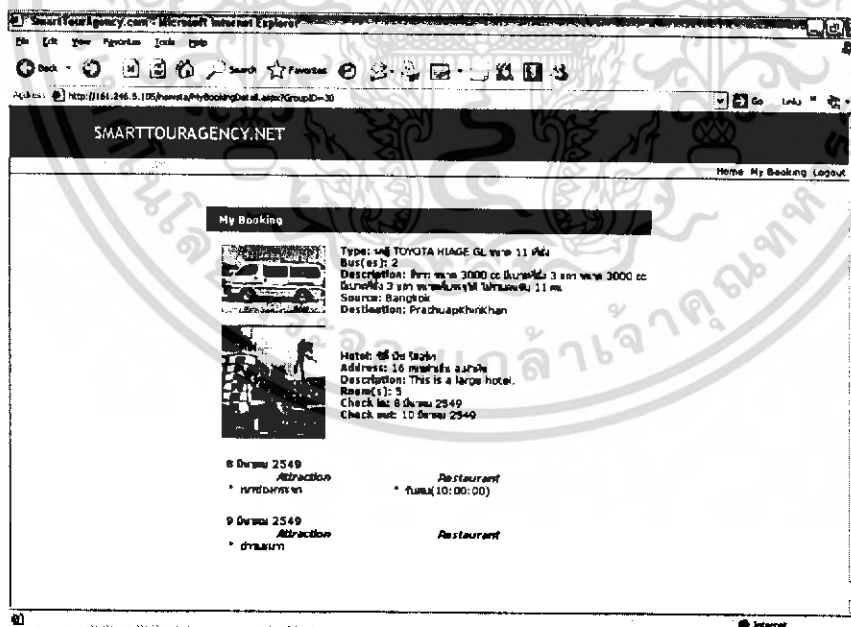
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.12. แสดงข้อมูลและทำการยกเลิกการจองแพ็คเกจที่ได้จองเอาไว้ของสมาชิก

ระบบจะแสดงข้อมูลการจองของสมาชิกที่ได้ทำการเข้าสู่ระบบไว้แล้ว โดยที่สามารถเข้าไปดูรายละเอียดของการจองในแต่ละครั้งได้ และยังสามารถที่จะทำการยกเลิกการจองได้ด้วย



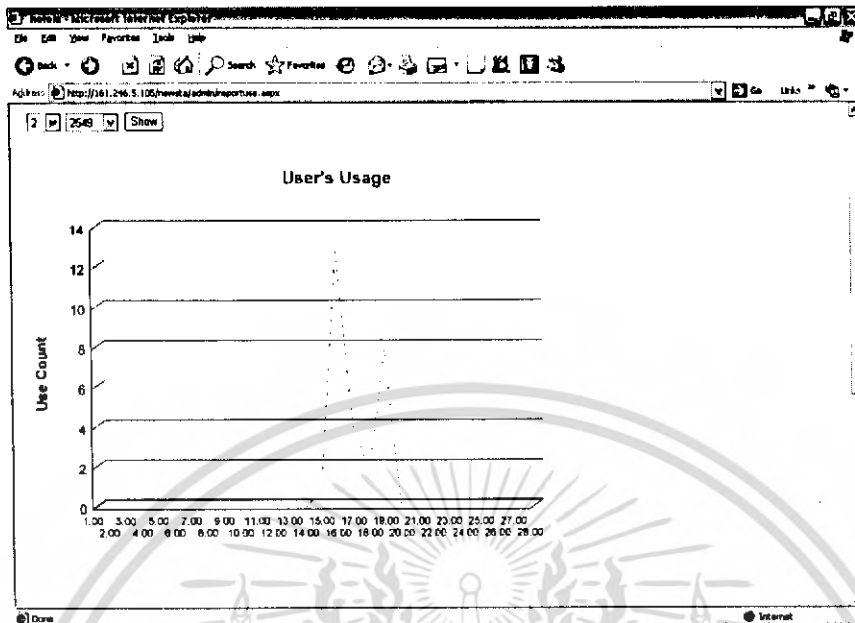
รูป 4.27 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงข้อมูลการจองและยกเลิกการจองของผู้ใช้ระบบ



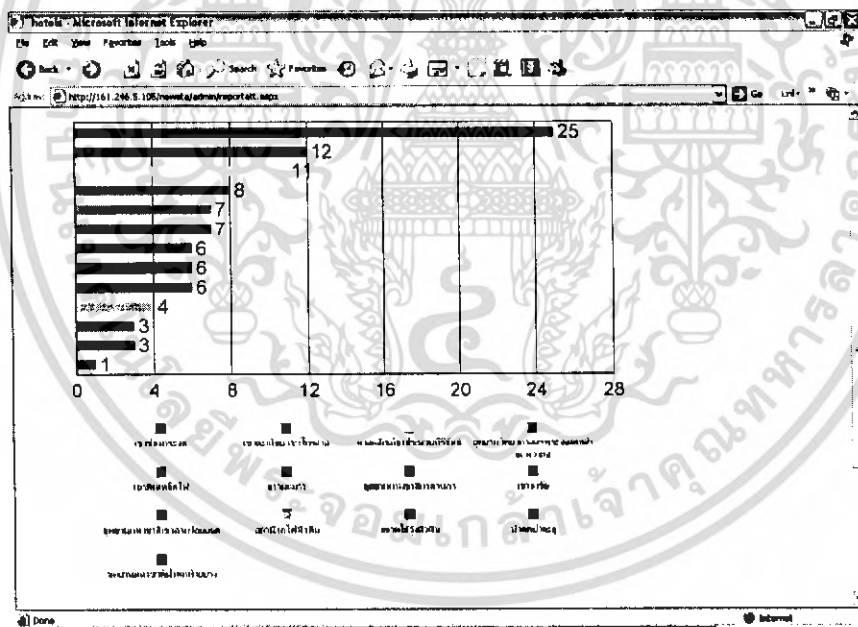
รูป 4.28 ภาพแสดงตัวอย่างหน้าเว็บเพจที่แสดงรายละเอียดการจองของผู้ใช้ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.5. การแสดงสถิติด้วย Crystal Report

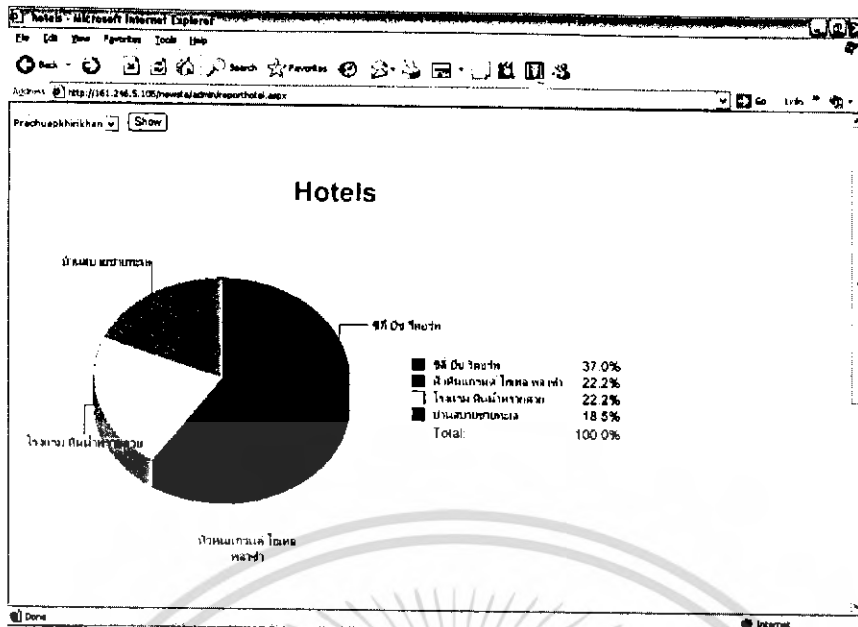


รูป 4.29 กราฟเส้นแสดงแนวโน้มการเข้ามาใช้บริการของผู้ใช้ระบบ



รูป 4.30 กราฟแท่งแสดงสถิติจำนวนของผู้ใช้ระบบที่เลือกไปสถานที่ท่องเที่ยวต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.31 กราฟวงกลมแสดงสถิติที่ผู้ใช้ระบบได้เลือกของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5 บทวิจารณ์และสรุป

### 5.1. ข้อสรุป

ในปัจจุบันได้มีการนำอินเทอร์เน็ตเทคโนโลยีเข้ามาใช้ในชีวิตประจำวันเพิ่มขึ้นเรื่อย อันเห็นได้จากการนำเอาอินเทอร์เน็ตเข้ามาช่วยงานทั้งการติดต่อสื่อสาร การทำธุรกรรมและความบันเทิง ทำให้ได้มีการพัฒนาอินเทอร์เน็ตเทคโนโลยีเหล่านี้กันอย่างต่อเนื่องขึ้นเรื่อยๆ จนทุกวันนี้ได้มีการพัฒนาเกิดเป็นเว็บเซอร์วิสที่ทำหน้าที่ติดต่อสื่อสารกันระหว่างคอมพิวเตอร์กับคอมพิวเตอร์ เพื่อให้เกิดความสะดวกสบายแก่นมนุษย์ในการจัดการต่างๆ มากยิ่งขึ้น

ในโครงการนี้จึงได้พัฒนาเว็บแอปพลิเคชันที่ใช้ติดต่อกับเว็บเซอร์วิส เพื่อช่วยทำการสร้างแพคเกจทัวร์ภายในประเทศไทยขึ้น ซึ่งเป็นตัวแทนในการติดต่อสื่อสารกับบริษัทต่างๆ ที่เกี่ยวข้องอันได้แก่ บริษัทขนส่ง, ที่พัก, ร้านอาหาร และสถานที่ท่องเที่ยว แทนผู้ใช้บริการ เพื่อให้ผู้บริการสามารถค้นหาข้อมูล และเลือกตามความพึงพอใจของผู้ใช้บริการได้ด้วยตนเอง แล้วสร้างเป็นแพคเกจทัวร์ออกมาจากการประมวลผลของระบบ โดยผ่านการติดต่อสื่อสารระหว่างเว็บเซอร์วิส

### 5.2. ปัญหาอุปสรรคและแนวทางการแก้ไข

- 1) เนื่องจากโครงการนี้เป็นระบบที่มีขนาดใหญ่ เพราะต้องมีการเรียกใช้เซอร์วิสจากทั้งบริษัทรถโดยสาร, โรงแรม และร้านอาหาร เมื่อมีการนำไปใช้จริงจำเป็นที่จะต้องมีการโครงสร้างของการทำงานรวมถึงรูปแบบของฟังก์ชันที่เรียกใช้เหมือนกัน ซึ่งจะเป็นไปไม่ได้ เพราะว่าในความเป็นจริงแล้ว แต่ละบริษัทที่เปิดเซิร์ฟเวอร์ให้บริการนั้น ได้ให้บุคลากรที่เป็นผู้ดูแลและออกแบบระบบเป็นคนจัดการทำให้ระบบที่ออกมานั้นจะแตกต่างกันออกไป ทำให้การเรียกใช้เซอร์วิสไปแต่ละแห่งจะมีความต้องการข้อมูลที่ต่างกันและผลลัพธ์ที่ได้ก็ออกมาจะไม่เหมือนกัน แนวทางการแก้ไขปัญหานั้น ก็คือ จะต้องเป็นองค์กรขนาดใหญ่ที่จะรวบรวมข้อมูลทั้งหมดไว้ แล้วให้บริษัทต่างๆ เข้ามาอัปเดตข้อมูล เรียกว่า C-Commerce (Collaborative-Commerce) และทำการสร้างเซอร์วิสที่จะให้มีการเรียกใช้ขึ้นภายในองค์กรนี้
- 2) การคำนวณระยะเวลาในการเดินทางไปยังแต่ละแห่งยังคำนวณได้ไม่แน่นอน เนื่องจากการเดินทางนั้นเป็นการเดินทางตามถนนที่มีการคดเคี้ยวไปมาต่างๆ กันไปแล้วแต่สภาพภูมิประเทศทำให้ไม่สามารถหาสูตรคำนวณออกมาได้จากสถานที่แต่ละแห่งมีระยะห่างกันเท่าไร และในส่วนของระยะเวลาในการเดินทางนั้นก็จะเป็นขึ้นอยู่กับการเดินทางด้วย รวมถึงสิ่งที่มีผลต่อการเดินทางอย่างอื่น เช่น การจราจร สภาพอากาศ และอุบัติเหตุ เป็นต้น จึงเป็นเหตุให้ไม่สามารถคำนวณระยะเวลาในการเดินทางได้อย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แน่นอน ซึ่งในโครงการนี้ได้ใช้การคำนวณระยะทางจากสถานที่หนึ่งไปยังอีกสถานที่หนึ่งโดยใช้การจัด แนวทางการแก้ไขนั้น ทำให้ได้โดยการค้นคว้าและเก็บข้อมูลเป็นสถิติในการเดินทางจากสถานที่แต่ละแห่ง แล้วนำข้อมูลที่ได้อามาหาแค่เฉลี่ยของในแต่ละช่วงเวลา ทำให้เกิดความแน่นอนในการคำนวณหาระยะเวลาในการเดินทางมากยิ่งขึ้น

- 3) การแสดงร้านอาหารที่ให้บริการจอง เนื่องจากการเดินทางจากสถานที่หนึ่งไปยังอีกสถานที่หนึ่งนั้นต้องรู้ถึงเส้นทางในการเดินทาง ซึ่งในโครงการนี้ได้ทำการคำนวณระยะทางโดยใช้การจัด ทำให้ไม่รู้ถึงเส้นทางในการเดินทาง และในการเดินทางจากสถานที่หนึ่งไปยังสถานที่หนึ่งนั้นอาจมีเส้นทางได้มากกว่าหนึ่งสายก็เป็นไปได้ จึงส่งผลต่อร้านอาหารที่จะปรากฏออกมาให้บริการจอง แนวทางในการแก้ไขนั้น ทำให้ได้โดยการค้นคว้าหาข้อมูลเส้นทางในการเดินทางจากสถานที่แต่ละแห่ง ซึ่งทำให้รู้ถึงร้านอาหารที่จะมีโอกาสผ่านในการเดินทางไปในแต่ละแห่งได้

### 5.3. แนวทางการพัฒนาต่อ

- 1) ทำการพัฒนาในส่วนคมนาคมขนส่งให้มีความหลากหลายมากยิ่งขึ้น เพื่อรองรับความต้องการของผู้ใช้ระบบ ทั้งทางรถ, รถไฟ, เรือ, เครื่องบิน ซึ่งจะต้องคำนึงถึงความเป็นจริงจะเกิดขึ้นได้ด้วย
- 2) ทำการพัฒนาระบบทัวร์ให้สามารถที่มีการเดินทางไปยังสถานที่หลายแห่งมากยิ่งขึ้น ซึ่งก็คือสามารถไปท่องเที่ยวและหาที่พักในหลายแห่งได้ด้วย อย่างเช่นการสร้างแพ็คเกจขึ้นภาคเหนือ สามารถที่จะแวะท่องเที่ยวและพักผ่อนได้ในหลายจังหวัด ทั้งลำพูน, ลำปาง, เชียงใหม่, เชียงราย ได้ในการเดินทางครั้งเดียว
- 3) พัฒนาระบบให้สามารถทำงานแบบ Long Time Durations ได้ เนื่องจากระบบมีการทำงานที่มีลักษณะทำงานหลายขั้นตอน และมีการ commit แบบ separate committed ทำให้เหมาะสมที่จะสามารถทำงานต่อเนื่องได้ แม้ว่า ได้มีการเข้าสู่ระบบเป็นระยะเวลานานแล้วก็ตาม

### บรรณานุกรม

- [1] EXBIZ (THAILAND) Co., Ltd.. “Siam WebServices.” [Online]. Available :  
<http://www.wsiam.com/document/abcwebservices/webservicesabc.jsp#what>
- [2] ASPThai.Net Full Edition v1. “ASP.NET Tutorial.” [Online]. Available :  
<http://www.aspthai.net/aspnet/default.asp>
- [3] พีรชสิทธิ์ ศุภจรรยาพงศ์, ไพโรจน์ แซ่เล้า, ภัทรภรณ์ อริยปรีชากุล “การพัฒนาซอฟต์แวร์เชิงคอมโพเนนต์โดยใช้ EJB1” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2545
- [4] พิสุทธิพงษ์ แสงอ่อน, เมธา ไพบุลย์ภิญญาเลิศ “การพัฒนาเว็บเซอร์วิสแอปพลิเคชัน โดยใช้ค็อตเน็ตเทคโนโลยี 1” วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2545

**ภาคผนวก ก**  
**การติดตั้ง SQL Server 2000**

ใส่แผ่น CD โปรแกรม SQL Server 2000 ลงในเครื่องอ่าน CD-ROM จากนั้นจะมีไดอะล็อกบ็อกซ์ปรากฏขึ้นมา

สำหรับผู้ที่ใช้ Windows 98 จะต้องติดตั้งโปรแกรมอัปเดตบางโปรแกรมลงไปเพิ่มเติม การติดตั้งโปรแกรม SQL Server 2000 สามารถติดตั้งได้ 3 แบบดังนี้

- **Local Computer** เป็นการติดตั้งโปรแกรมลงบนเครื่องคอมพิวเตอร์ที่กำลังใช้งานอยู่
- **Remote Computer** เป็นการติดตั้งโปรแกรมลงบนเครื่องคอมพิวเตอร์เครื่องอื่น
- **Virtual Server** เป็นการติดตั้งโปรแกรมลงบนเครื่องคอมพิวเตอร์ในระบบคลัสเตอร์

SQL Server 2000 กำหนดการติดตั้งไว้ 3 รูปแบบ ดังนี้

- **Client Tools Only** ติดตั้งเฉพาะเครื่องมือจัดการระบบเท่านั้น
- **Server and Client Tools** ติดตั้งโปรแกรมทั้งหมดที่เป็นเซิร์ฟเวอร์และเครื่องมือจัดการระบบ
- **Connectivity Only** ติดตั้งเฉพาะส่วนติดต่อระหว่างไคลเอ็นต์กับ SQL Server 2000 เท่านั้น

SQL Server 2000 สามารถติดตั้งหลายๆ ระบบ (Instance) ภายในเครื่องเดียวกันได้ (Multiple Instance) โดยสามารถติดตั้งได้สูงสุดถึง 16 อินสแตนซ์ ซึ่งแตกต่างจากเวอร์ชันก่อนหน้า ที่ติดตั้งได้เพียงอินสแตนซ์เดียวเท่านั้น

การติดตั้งอินสแตนซ์เพิ่มเติมเป็นการแยกสภาพแวดล้อมการทำงานของ SQL Server 2000 ออกจากอินสแตนซ์ปกติ ซึ่งมีประโยชน์ในการแยกการทำงานที่แตกต่างกัน เช่น อินสแตนซ์หนึ่งอาจใช้ในการทำงานตามปกติ แต่อีกอินสแตนซ์อาจนำไปใช้ในการทดสอบการทำงานของโปรแกรมได้

สำหรับผู้ที่ใช้ SQL Server เวอร์ชันก่อนหน้า สามารถเป็นเวอร์ชัน 2000 ทำให้สามารถใช้เครื่องมือและคำสั่งใหม่ๆ ได้

## Database

หลังจากติดตั้ง SQL Server 2000 เรียบร้อยแล้ว โปรแกรมจะให้ฐานข้อมูล 2 ประเภท ดังนี้

### - System Database ประกอบด้วย (ห้ามลบ)

- **master** เก็บข้อมูลระบบ, Error Message, Login Information, System Stored Procedure
- **model** เท็มเพลตที่ใช้กับฐานข้อมูลใหม่
- **msdb** เก็บข้อมูล Jobs, Alert, Event, Replication, History ของการ Backup /Restore
- **tempdb** เป็นที่พักข้อมูลชั่วคราว โดยจะถูก Initial ทุกครั้งที่มีการสตาร์ท เซอร์วิส MSSQLSERVER ขึ้นมา

### - Example Database ประกอบด้วย (ลบได้)

- **Pubs** ฐานข้อมูลสำนักพิมพ์
- **Northwind** ฐานข้อมูล Northwind

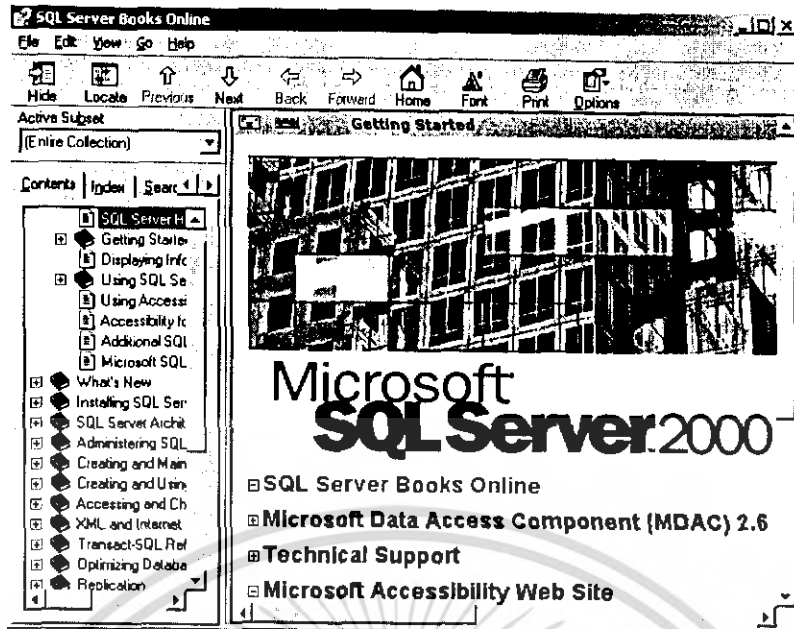
## SA (System Administration)

sa เป็นชื่อ Login ของผู้ดูแลระบบ SQL Server 2000 เป็นผู้ที่สามารถทำงานได้ทุกอย่างที่อยู่ภายใน SQL Server 2000 เช่นการสร้าง-ลบ-แก้ไขฐานข้อมูล เพิ่มเติม-ลบ-แก้ไขข้อมูล กำหนดสิทธิ์และผู้ใช้งานฐานข้อมูล

## เครื่องบริหารระบบงาน

หลังจากติดตั้ง SQL Server 2000 เรียบร้อยแล้ว จะมีเครื่องมือทำงานให้ดังนี้

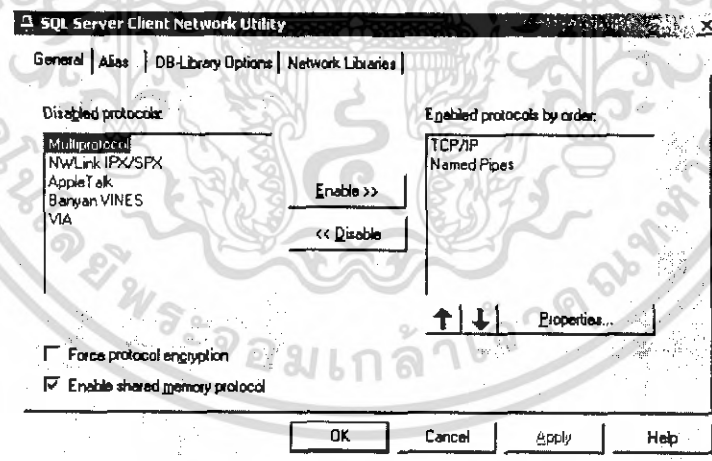
- **Book Online** คู่มือการใช้งานคำสั่งต่าง ๆ ของ SQL Server 2000



รูป ก-1 แสดงรูป Book Online

- Client Network Utility เป็นเครื่องมือที่ใช้ติดต่อกับเซิร์ฟเวอร์ที่มีเน็ตเวิร์คไลบรารีตรงกัน (ใช้กับฝั่งไคลเอนท์)

- **General** ใช้กำหนด protocol ที่ใช้งาน (ขึ้นอยู่กับระบบคอมพิวเตอร์ที่ใช้งานด้วย)

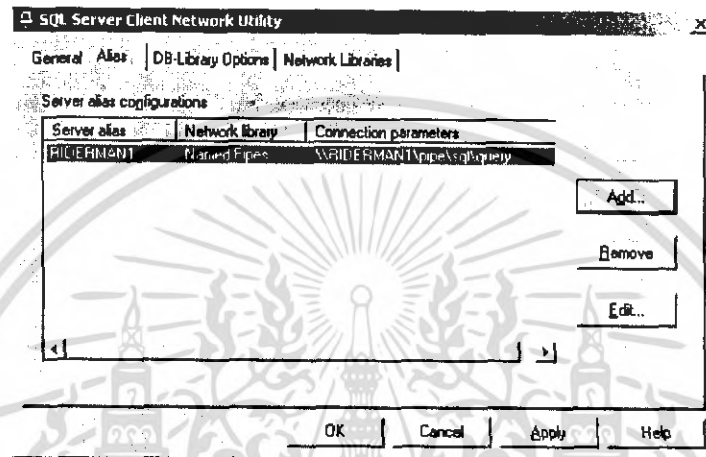


รูป ก-2 แสดงรูป Network Utility

- TCP/IP
- Name Pipe
- Multi Protocol
- NW link IPX/SPX

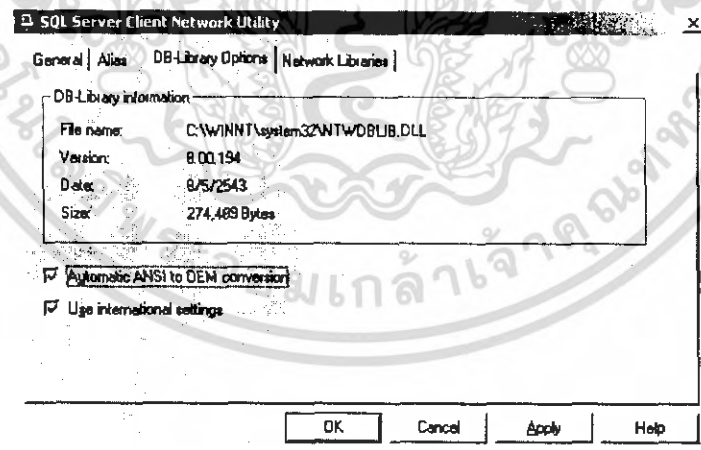
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Apple talk
  - Banyan Vines
  - VIA
- Alias กำหนดชื่อเล่นให้กับ server ที่ติดต่อกับ SQL Server 2000 พร้อมระบุโปรโตคอลที่ติดต่อ



รูป ก-3 แสดงรูปการกำหนด Alias ใน Network Utility

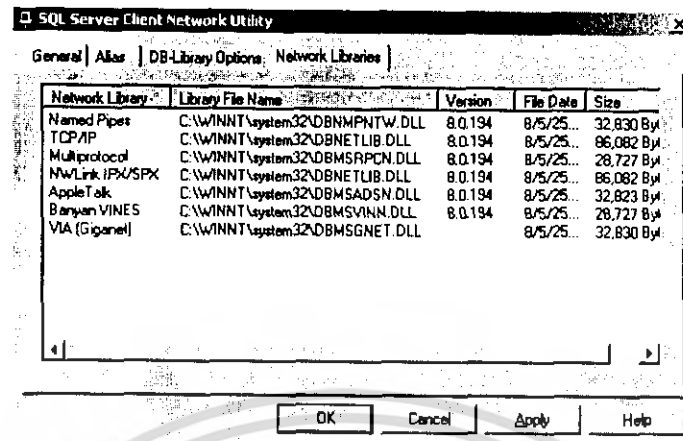
- DB-Library Options แสดงข้อมูลของไลบรารีดาตาเบส



รูป ก-4 แสดงรูป DB-Library Options ใน Network Utility

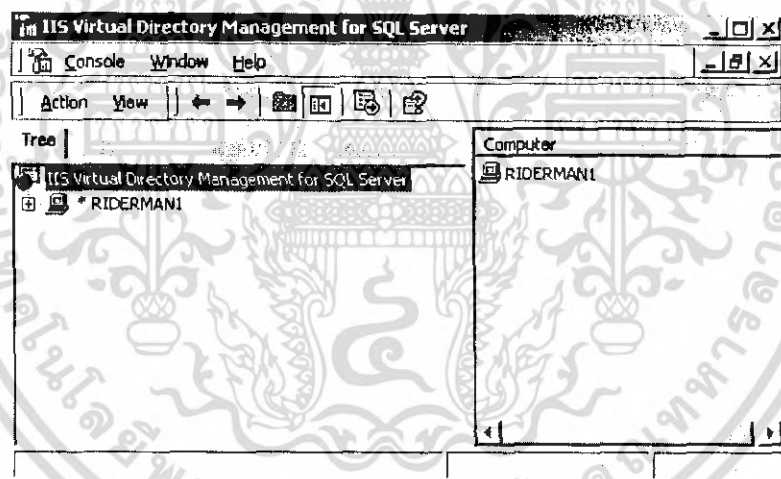
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Network Library แสดงข้อมูลไลบรารีของเน็ตเวิร์คที่ระบบใช้อยู่



รูป ก-5 แสดงรูป Network Library ใน Network Utility

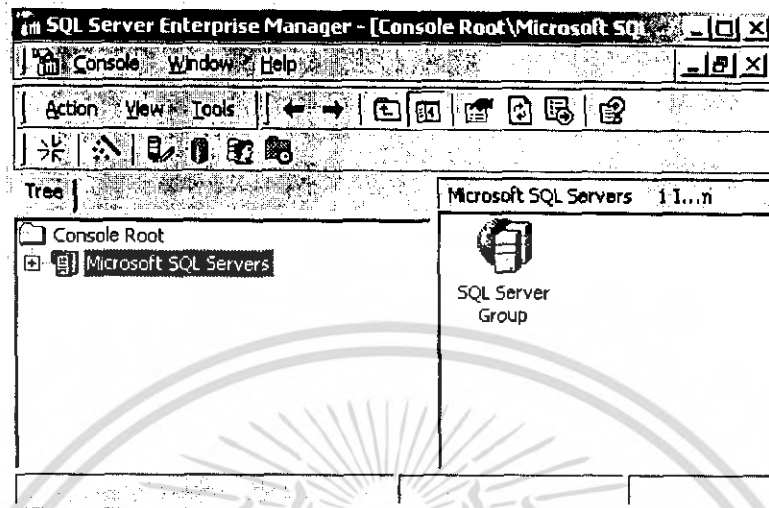
- Configure SQL XML Support in IIS เครื่องมือนำข้อมูลจาก SQL Server 2000 ออก Internet ผ่าน IIS web server แบบภาษา XML.



รูป ก-6 แสดงรูปการ Config IIS ใน SQL Server

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Enterprise Manager** เครื่องมือบริหารฐานข้อมูลที่สำคัญ ใช้สำหรับจัดการฐานข้อมูล, Backup/Restore, สร้าง package สำหรับโอนข้อมูล , กำหนดระบบความปลอดภัย ฯลฯ



รูป ก-7 แสดงรูป SQL Enterprise Manager

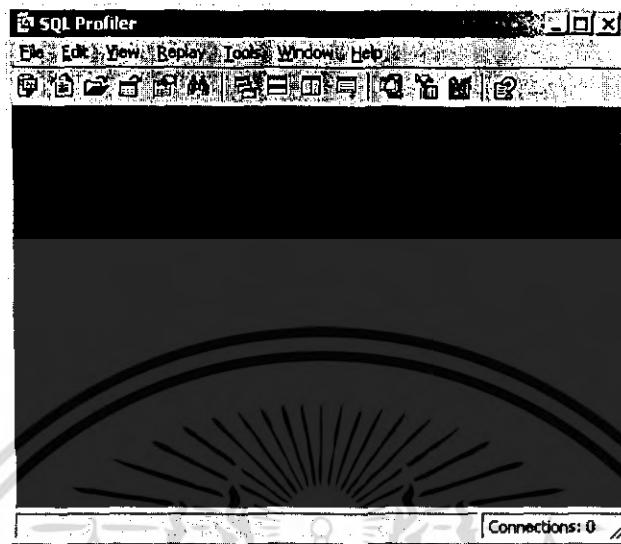
- **Import and Export Data** โอนถ่ายข้อมูลจาก SQL Server ไปยัง โปรแกรมอื่น หรือดึงข้อมูลจากโปรแกรมอื่นมาใช้กับ SQL Server



รูป ก-8 แสดงรูปการ Import และ Export ข้อมูล

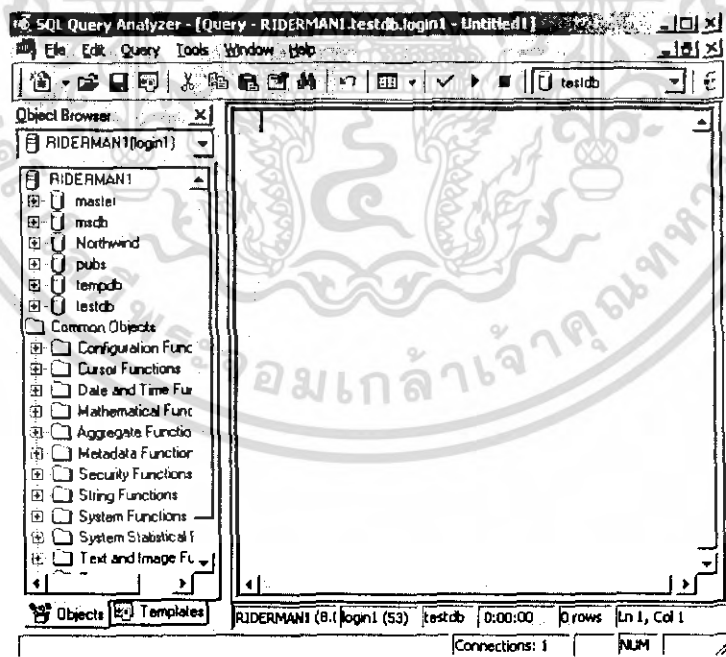
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Profiler ตรวจสอบการทำงานของ SQL Server ชื่อเดิมคือ SQL Trace ใช้สำหรับตรวจสอบและปรับปรุงประสิทธิภาพของระบบ



รูป ก-9 แสดงรูป SQL Profiler

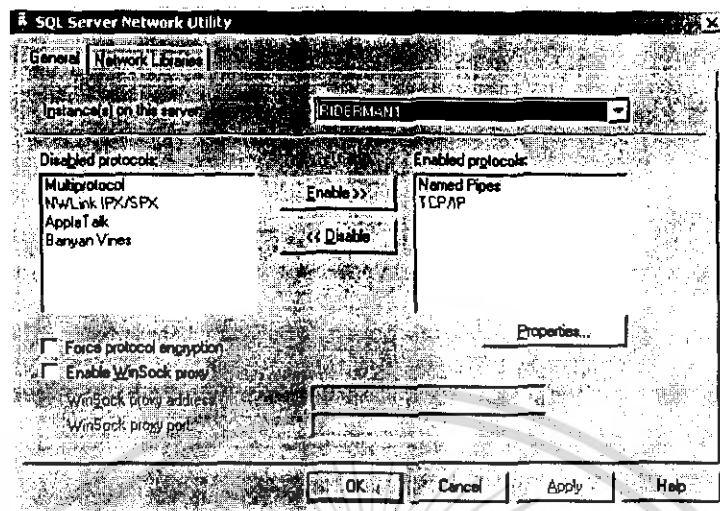
- Query Analyzer เครื่องมือหลักในการเขียนคำสั่ง SQL (Transact - SQL) ชื่อเดิมคือ ISQL/W



รูป ก-10 แสดงรูป SQL Query Analyzer

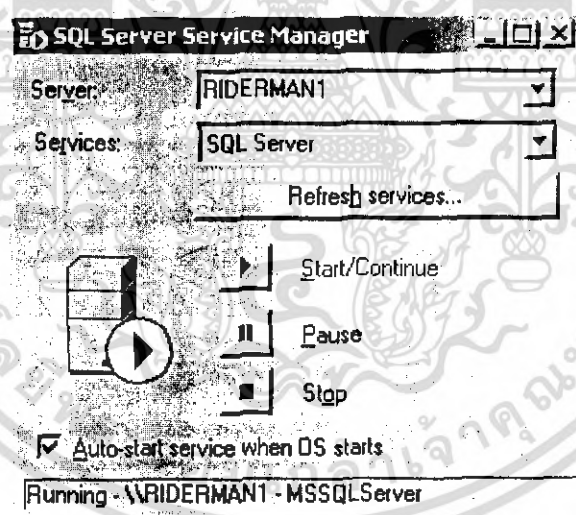
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Server Network Utility เครื่องมือกำหนด Network Library ให้กับฝั่ง Server



รูป ก-11 แสดงรูป SQL Server Network Utility

- Service Manager เครื่องมือควบคุมและแสดงสถานะของเซอร์วิสที่เกี่ยวข้องกับ SQL Server



รูป ก-12 แสดงรูป SQL Server Service Manager

ภายในไดอะล็อกบ็อกซ์ SQL Server Service Manager ประกอบไปด้วย

- **Server** เป็นชื่อเซิร์ฟเวอร์ที่ติดตั้ง SQL Server ที่ผ่านการ Register แล้ว
- **Services** เป็นชื่อเซอร์วิสที่เลือกใช้บริการ มีด้วยกันดังนี้ (ขึ้นอยู่กับเวอร์ชันและ คอนโพลเมนต์ ที่ติดตั้งลงไป)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- MSSQLSERVER เป็นเซอร์วิสหลักของการทำงาน
- SQLSERVERAGENT เป็นเซอร์วิสที่คอยทำหน้าที่ Schedule Jobs, Alerts, Event Management และ Replication
- Microsoft Search เป็นเซอร์วิสที่เกิดขึ้นเมื่อติดตั้งคอม โฟเนนด์ Full-Text Search โดยทำงานเกี่ยวกับ Index ของข้อมูล
- Microsoft Distributed Transaction Coordinator (MSDTC) เป็นเซอร์วิสที่ทำหน้าที่ควบคุม ทรานแซกชันสำหรับการโอนถ่ายข้อมูลระหว่างเซิร์ฟเวอร์
- Microsoft Analysis Service เป็นเซอร์วิสที่ทำหน้าที่เกี่ยวกับการบริหารและจัดการคลังข้อมูล เซอร์วิสนี้ทำงานเมื่อมีการติดตั้งคอม โฟเนนด์ Analysis Service

### MSDE

สำหรับผู้ติดตั้ง SQL Server 2000 บนระบบปฏิบัติการ Windows 9x, ME หรือ XP จะต้องติดตั้ง MSDE (Microsoft Database Engine) เพื่อให้สามารถใช้งานเซอร์วิสได้