

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เว็บเซอร์วิส แอปพลิเคชัน ดีเวลอปเมนต์ I

Web Services Application Development I



นายสรายุติ เรืองสุธรรม
นางสาวสวรรค์ อารมณรัตน์
นางสาวศิริกานต์ วงศ์ตั้งใจ

เลขหมู่.....
เลขทะเบียน..... 72938
วัน,เดือน,ปี..... 26 ส.ย. 2550

b. 11274964
i.....

รายงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เว็บเซอร์วิส แอปพลิเคชัน ดีเวล็อปเมนต์ I

Web Services Application Development I

โดย

นายสรวิทย์ เรืองสุวรรณ รหัสประจำตัว 46010813

นางสาวสวรัตน์ อารมย์รัตน์ รหัสประจำตัว 46010816

นางสาวสิริภรณ์ ดวงสีตังใจ รหัสประจำตัว 46010836



อาจารย์ที่ปรึกษา

ดร.วรัฒน์ ถิ่นโกศา

รายงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2549

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เว็บเซอร์วิส แอปพลิเคชัน ดีเวลอปเมนท์ I

Web Services Application Development I

ผู้จัดทำ

1. นายสรวิชัย เรืองสุธรรม รหัสนักศึกษา 46010813
2. นางสาวสวรัตน์ อารมณรัตน์ รหัสนักศึกษา 46010816
3. นางสาวสิริกานต์ วงศ์ตั้งใจ รหัสนักศึกษา 46010836



อาจารย์ที่ปรึกษา

(ดร. วรวัฒน์ ลิ้มโกศา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เว็บเซอร์วิส แอปพลิเคชัน ดีเวลอปเมนต์ I

นายสรายุทธิ	เรืองสุธรรม	46010813
นางสาวสวรรค์	อารมณรัตน์	46010816
นางสาวสิริกานต์	วงศ์ตั้งใจ	46010836
ดร.วรวัฒน์	ลี้ม โภคา	อาจารย์ที่ปรึกษา
ปีการศึกษา 2549		

บทคัดย่อ

ในการทำโครงงานครั้งนี้ ได้นำเสนอวิธีการพัฒนาเว็บเซอร์วิสบนจาวาแพลตฟอร์ม โดยใช้โปรแกรม IBM Rational Application Developer 6.0 ในการพัฒนา ซึ่งในการพัฒนาเว็บเซอร์วิสมีการนำมาตรฐานต่างๆมาใช้ เช่น การรักษาความปลอดภัยของข้อมูลที่ใช้ในการติดต่อเว็บเซอร์วิสโดยยึดตามมาตรฐาน WS-Security และมีการนำเอา WS-Transaction มาใช้เพื่อช่วยในการทำธุรกรรม (transaction) ใดๆให้สามารถดำเนินไปได้ โดยถ้าสำเร็จให้สำเร็จหมดและถ้าไม่สำเร็จให้เหมือนกับธุรกรรมนั้นไม่เคยเกิดขึ้นเลย ทั้งนี้เพื่อตอบสนองของระบบงานที่หลากหลายรูปแบบ และมีการนำเอา Business Process Execution Language (BPEL) มาใช้ในการสร้าง workflow เพื่อเป็นตัวกลางในการจัดการเรียกใช้เว็บเซอร์วิสหลายๆตัวที่สร้างขึ้น เพื่อช่วยลดความยุ่งยากในการเรียกใช้งานเว็บเซอร์วิสผ่าน application โดยตรง นอกเหนือจากนั้นปริยญาณีพนธ์เล่มนี้ยังอธิบายถึงรายละเอียดของเว็บเซอร์วิสทั้งหมด ซึ่งประกอบไปด้วย การทำงานของเว็บเซอร์วิส, สถาปัตยกรรมของเว็บเซอร์วิส, เทคโนโลยีที่ใช้ในเว็บเซอร์วิส อันได้แก่ XML, SOAP, WSDL และ UDDI โดยดูว่าในแต่ละเทคโนโลยีที่นำมาใช้นั้นมีการนำมาใช้ในลักษณะใด พร้อมทั้งนำเสนอระบบ Online Booking System โดยพัฒนาให้ผู้ใช้จากระบบมีการใช้งานผ่านเว็บแอปพลิเคชัน ซึ่งจะมีการเรียกใช้ต่อไปยัง BPEL เพื่อใช้เป็นตัวกลางในการเรียกเว็บเซอร์วิส โดยนำเอาหลักการต่างๆดังที่ได้กล่าวมาแล้วมาประยุกต์ใช้ในระบบดังกล่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Web Services Application Development I

Mr. Saravuth Ruangsutham 46010813

Miss Sawarat Arromratana 46010816

Miss Sirikarn Wongtangjai 46010836

Dr. Vorawat Limpoka Advisor

Academic Year 2006

ABSTRACT

This thesis presents a method for web service development on JAVA platform. Nowadays, there are various types of transactions occurring on website; security is then very important. Therefore, to maintain the highest security, while still provide the best flow of transaction and finest responses to users' needs, IBM Rational Application Developer 6.0 program, of which the security system follows WS-Security standard, and the program of WS-transaction will be applied. Each transaction will be set to automatically cancel itself if any errors would happen during transaction steps. In addition, Business Process Execution Language (BPEL) is used for creating workflow which will be used as a connecting subject of web service in order to decrease complexity in calling web service via application directly. Moreover, this thesis also provides other deep knowledge about web-service. Topics are divided into three major parts that are web service process, web-service architecture, and web service technology. For web service technology, the thesis will explain in details about the application of four major technologies, which are XML, SOAP, WSDL and UDDI. Lastly, a case study of the system of Online Booking System, which has already developed its web service and applied all mentioned principles into the system, will also be presented at the end of the thesis.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญาโทฉบับนี้คงไม่อาจสำเร็จได้ด้วยดี หากขาดการช่วยเหลือ สนับสนุนและให้คำปรึกษาจาก ดร. วรวัฒน์ ลิ้ม โภคา ซึ่งเป็นผู้ควบคุมดูแลในการทำปริญญาโทในครั้งนี้ ข้าพเจ้ารู้สึกทราบบ้างในความอนุเคราะห์ของอาจารย์ และขอกราบขอบพระคุณท่านเป็นอย่างสูง

ขอกราบขอบพระคุณคณาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกๆท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า รวมไปถึงบุคลากรทุกท่านที่คอยอำนวยความสะดวกและประสานงานในเรื่องต่างๆ

ขอขอบคุณรุ่นพี่ และบริษัท IBM ที่อนุเคราะห์โปรแกรมที่ใช้ในการทำปริญญาโท รวมไปถึงคำแนะนำต่างๆ ในครั้งนี้เป็นอย่างสูง

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่คอยให้กำลังใจ และให้คำแนะนำที่ดีเสมอมา

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจ และให้การสนับสนุนช่วยเหลือในทุกๆเรื่อง ทำให้ข้าพเจ้าสามารถทำปริญญาโทในครั้งนี้สำเร็จ ลุล่วงไปได้ด้วยดี

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์เล่มนี้ ข้าพเจ้าขอมอบให้แก่ผู้มีพระคุณทุกท่าน

นายสรวิทย์ เรืองสุธรรม
นางสาวสวรัตน์ อารมณรัตน์
นางสาวศิริกานต์ วงศ์ตั้งใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูปภาพ	VIII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	2
1.4 ขอบเขตของโครงการ	2
1.5 วิธีการดำเนินงาน	3
บทที่ 2 ทฤษฎีพื้นฐานเกี่ยวกับเว็บเซอร์วิส	4
2.1 เว็บเซอร์วิสคืออะไร	4
2.2 เปรียบเทียบระหว่างเว็บแอปพลิเคชันกับเว็บเซอร์วิส	4
2.3 Service Oriented Architecture (SOA)	5
2.3.1 ส่วนประกอบเบื้องต้นของ SOA	5
2.4 เทคโนโลยีพื้นฐานของเว็บเซอร์วิส	6
2.4.1 XML (Extensible Markup Language)	7
2.4.1.1 XML และ HTML	7
2.4.2 SOAP (Simple Object Access Control)	8
2.4.2.1 โครงสร้างของ SOAP	9
2.4.3 WSDL (Web Service Description Language)	12
2.4.3.1 โครงสร้างเอกสาร WSDL	12
2.4.4 UDDI (Universal Description, Discovery, and Integration)	15
2.4.4.1 โครงสร้างข้อมูลของ UDDI	16
2.5 ประเภทของเว็บเซอร์วิสและการเรียกใช้	17
2.6 JavaServer Pages (JSP)	18
2.6.1 ตัวอย่างการเขียน JSP	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.6.2 เปรียบเทียบ JSP กับเทคโนโลยีอื่นๆ	19
2.6.2.1 แอคทีฟเซิร์ฟเวอร์เพจ	19
2.6.2.2 จาวาเซิร์ฟเล็ต	19
2.6.2.3 จาวาสคริปต์	19
บทที่ 3 Business Execution Language (BPEL)	20
3.1 Business Execution Language (BPEL)	20
3.2 ตัวอย่าง Business process	22
3.3 BPEL Language Structure	23
3.4 BPEL activity	23
บทที่ 4 WS-Security	25
4.1 นิยามความมั่นคงปลอดภัยในคอมพิวเตอร์	25
4.2 การพิสูจน์ตัวตน (Authentication)	28
4.2.1 หลักการทำงานของ Username Token	29
4.2.2 อีลีเมนต์ที่ใช้ในการส่งที่เกี่ยวกับโทเคนแบบระบุรหัสผ่าน โดยใช้ nonce	30
4.3 การรักษาความลับของข้อมูล (Confidentiality)	31
4.3.1 หลักการทำงานของ XML Encryption	31
4.3.2 อีลีเมนต์ที่ใช้ใน XML Encryption	33
4.4 การรักษาความสมบูรณ์ของข้อมูล (Integrity)	36
4.4.1 หลักการทำงานของ XML Digital Signature	36
4.4.2 อีลีเมนต์ที่ใช้ใน XML Digital Signature	37
4.5 ใบรับรองอิเล็กทรอนิกส์ (Certification)	38
4.5.1 ขั้นตอนการขอใบรับรองของเว็บ http://www.verisign.com (CA International)	40
4.5.2 ขั้นตอนการขอใบรับรองของเว็บ http://ca.khonthai.com (CA Thai)	40
4.6 Secure Socket Layer (SSL)	41
4.7 ความจำเป็นที่จะต้องใช้ WS-Security ควบคู่ไปกับ Secure Socket Layer (SSL)	48
บทที่ 5 ทรานแซกชัน	50
5.1 ทรานแซกชัน คืออะไร	50
5.2 WS-Coordination (WS-COOR)	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
5.2.1 Model	51
5.2.2 Extensibility	52
5.2.3 Coordination Context	53
5.2.4 Coordination Service (Coordinator)	54
5.2.5 Coordination Fault	59
5.2.6 Security Model	62
5.3 WS-AtomicTransaction (WS-AT)	63
5.3.1 การcommit แบบสองระยะ (Two-Phase Commit Protocol)	64
5.3.2 Atomic Transaction Context	65
5.3.3 Atomic Transaction Protocols	65
5.3.4 Preconditions	66
5.3.5 Completion Protocol	66
5.3.6 ลักษณะของ Two-Phase Commit Protocol ใน WS-AtomicTransaction	67
5.3.7 Transaction Faults	69
5.4 ความสามารถในการสนับสนุน WS-Atomic Transaction บน WebSphere Application Server	71
5.4.1 การตั้งค่า Deployment descriptors สำหรับ atomic transactions	71
บทที่ 6 การออกแบบและพัฒนาระบบ Online Booking System	73
6.1 รายละเอียดของแอปพลิเคชัน	73
6.2 Application Model with Security Model	74
6.3 Application Model with Transaction Model	75
6.4 Use Case Diagram	76
6.5 Sequence Diagram	77
6.6 Activity Diagram	82
6.7 ความสัมพันธ์ของตารางในฐานข้อมูลของโรงแรม	90
6.8 ความสัมพันธ์ของตารางในฐานข้อมูลของสายการบิน	91
6.9 ความสัมพันธ์ของตารางในฐานข้อมูลของแพคเกจทัวร์	92
6.10 ความสัมพันธ์ของตารางในฐานข้อมูลของธนาคาร	92
6.11 Class Diagram	93

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 7 การทดลองและผลการทดลอง	94
7.1 เครื่องมือที่ใช้ในการทดลอง	94
7.2 ฟังก์ชันการทำงานของเว็บเซอร์วิส	94
7.2.1 ฟังก์ชันการทำงานของเว็บเซอร์วิสตัวหลักของสายการบิน	94
7.2.2 ฟังก์ชันการทำงานของเว็บเซอร์วิสตัวหลักของโรงแรม	95
7.2.3 ฟังก์ชันการทำงานของเว็บเซอร์วิสตัวหลักของแพคเกจทัวร์	95
7.3 การใช้งานแอปพลิเคชัน	95
7.3.1 การสมัครสมาชิก	95
7.3.2 การเลือกจองประเภทต่างๆจากหน้า index	96
7.3.3 การค้นหาและจองโรงแรม	97
7.3.4 การค้นหาและจองสายการบิน	99
7.3.5 การค้นหาและจองทั้งสายการบินและโรงแรม	103
7.3.6 การค้นหาและจองแพคเกจทัวร์	104
7.3.7 การดูรายละเอียดและแก้ไขรายละเอียดการจอง	106
7.4 ผลการทดลองที่ได้จากการเรียกใช้เว็บเซอร์วิสที่มีการทำทรานแซกชัน	107
7.5 ผลการทดลองที่ได้จากการทำ WS-Security	108
7.5.1 การเพิ่มความปลอดภัยในการพิสูจน์ตัวตน โดยใช้ Username Token	109
7.5.2 การรักษาความลับของข้อมูล โดยการใช้ XML Encryption	109
7.5.3 การรักษาความสมบูรณ์ของข้อมูล โดยการใช้ Digital Signature	110
7.6 ผลการทดลองจาก Business process execution language (BPEL)	111
บทที่ 8 บทวิจารณ์และสรุป	113
8.1 บทวิจารณ์และสรุปผล	113
8.2 ปัญหาอุปสรรคและแนวทางแก้ไข	113
8.3 แนวทางการพัฒนาต่อ	115
บรรณานุกรม	116
ภาคผนวก ก คู่มือการติดตั้ง	118
ภาคผนวก ข การคอนฟิก WS-Security	140

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 การเปรียบเทียบระหว่างเว็บแอปพลิเคชันกับเว็บเซอร์วิส	5
2.2 Sub Element ของ Fault Element	12
2.3 คำอธิบายของแต่ละอีลีเมนต์ในเอกสาร WSDL	14
5.1 คุณสมบัติของ [Code] สำหรับ SOAP 1.2	59
7.1 รายละเอียดของฮาร์ดแวร์ที่ใช้	94
7.2 รายละเอียดของซอฟต์แวร์ที่ใช้	94



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 สถาปัตยกรรมแบบ SOA	6
2.2 ตัวอย่างเอกสาร XML	8
2.3 XML messaging using SOAP	8
2.4 โครงสร้างของเอกสาร SOAP	9
2.5 SOAP Envelope Element	10
2.6 SOAP Header Element	10
2.7 ตัวอย่าง SOAP รีเควสต์เมสเสจ	11
2.8 ตัวอย่าง SOAP เรสพอนส์เมสเสจ	11
2.9 แสดงโครงสร้างเอกสาร WSDL	12
2.10 ตัวอย่างเอกสาร WSDL	15
2.11 ภาพรวมโครงสร้างข้อมูลของ UDDI	16
2.12 ความสัมพันธ์ของโครงสร้างของ UDDI กับ tModel	17
2.13 การทำงานของ JavaServer Pages (JSP)	18
3.1 Web services technology stack	20
3.2 BPEL Components	21
3.3 ตัวอย่าง Business Process	22
3.4 BPEL Language Structure	23
3.5 Structured activities	24
4.1 การส่ง SOAP Message ที่ไม่มีการรักษาความปลอดภัย	26
4.2 การส่ง SOAP Message ที่มีการรักษาความปลอดภัย	27
4.3 การรักษาความปลอดภัยข้อมูลแบบ End-to-End โดยใช้ WS-security	27
4.4 โมเดล Securing Web Services	28
4.5 ประเภทของ Token	28
4.6 การส่ง SOAP Message ที่มีการใช้ nonce และ created	30
4.7 ตัวอย่างการใช้ Username Token แบบใช้ nonce	31
4.8 การเข้ารหัสและถอดรหัส	32
4.9 การเข้ารหัสและถอดรหัสแบบสมมาตร	32
4.10 การเข้ารหัสแบบอสมมาตร	32
4.11 อีลีเมนต์ที่ใช้ใน XML Encryption	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.12	34
4.13	35
4.14	36
4.15	37
4.16	38
4.17	39
4.18	42
4.19	43
4.20	44
4.21	44
4.22	45
4.23	45
4.24	46
4.25	47
4.26	48
4.27	48
4.28	48
5.1	52
5.2	55
5.3	58
5.4	63
5.5	66
5.6	68
6.1	74
6.2	75
6.3	76
6.4	77
6.5	77
6.6	78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
6.7 Sequence Diagram (Reserve Airline)	79
6.8 Sequence Diagram (Reserve Package)	80
6.9 Sequence Diagram (View Reservation)	81
6.10 Sequence Diagram (Edit)	81
6.11 Activity diagram (Register)	82
6.12 Activity diagram (Login)	83
6.13 Activity diagram (View Info)	84
6.14 Activity diagram (Reserve Hotel)	85
6.15 Activity diagram (Reserve Airline)	86
6.16 Activity diagram (Reserve Hotel_Airline)	87
6.17 Activity diagram (Reserve Package Tour)	88
6.18 Activity diagram (View Reservation)	89
6.19 Activity diagram (Edit Reservation)	89
6.20 ความสัมพันธ์ของตารางในฐานข้อมูลของโรงแรม	90
6.21 ความสัมพันธ์ของตารางในฐานข้อมูลของสายการบิน	91
6.22 ความสัมพันธ์ของตารางในฐานข้อมูลของแพคเกจทัวร์	92
6.23 ความสัมพันธ์ของตารางในฐานข้อมูลของธนาคาร	92
6.25 Class diagram	93
7.1 ฟังก์ชันที่มีให้เลือกใช้ในเว็บเซอร์วิสตัวหลักของสายการบิน	94
7.2 ฟังก์ชันที่มีให้เลือกใช้ในเว็บเซอร์วิสตัวหลักของโรงแรม	95
7.3 ฟังก์ชันที่มีให้เลือกใช้ในเว็บเซอร์วิสตัวหลักของแพคเกจทัวร์	95
7.4 การสมัครสมาชิก	96
7.5 หน้า index	96
7.6 หน้าค้นหาโรงแรม	97
7.7 รายชื่อโรงแรมที่ได้จากการค้นหา	97
7.8 หน้าให้ผู้ใช้บริการเลือกชนิดของห้องพัก	98
7.9 หน้าการคิดราคารวมทั้งหมด	98
7.10 หน้าแสดงรหัสการจอง	99
7.11 หน้าค้นหาเที่ยวบิน	99

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญรูป (ต่อ)

รูปที่	หน้า	
7.12	รายละเอียดของสายการบินที่ได้จากการค้นหา	100
7.13	หน้าให้ผู้ใช้บริการเลือกประเภทที่นั่ง	101
7.14	หน้าการคิดราคารวมทั้งหมด	101
7.15	หน้าให้ผู้ผู้ใช้ใส่รายชื่อผู้ที่จะเดินทาง และรายละเอียดบัตรเครดิต	102
7.16	หน้าแสดงรหัสการจอง	102
7.17	การค้นหาสายการบินและโรงแรม	103
7.18	หน้าราคารวมทั้งหมดของทั้งการจองเครื่องบินและโรงแรม	104
7.19	หน้าให้ผู้ผู้ใช้ใส่รายละเอียดต่างๆสำหรับการจองแพคเกจ	105
7.20	หน้ารายละเอียดของแพคเกจทัวร์	105
7.21	หน้ารายละเอียดการจองของของผู้ใช้งาน	106
7.22	หน้ารายละเอียดการจองของรายการที่ได้เลือกไว้	106
7.23	หน้าสำหรับให้ผู้ผู้ใช้เปลี่ยนแปลงรายละเอียดการจอง	107
7.24	หน้าสุดท้ายของการจองตั๋วเครื่องบิน (ทำการจองไม่สำเร็จ)	107
7.25	SOAP Message ที่มีการระบุว่ามีการ roll back เกิดขึ้น	108
7.26	SOAP Message ก่อนการทำ WS-security	108
7.27	SOAP ที่มีการทำ Username Token	109
7.28	SOAP ที่มีการทำ XML Encryption	110
7.29	SOAP ที่มีการทำ XML Digital Signature	110
7.30	ผลการทดลองจาก Business process execution language (BPEL)	112
8.1	โค้ดที่ต้องเขียนในการเรียกใช้ BPEL จาก Rational	114

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญและที่มาของโครงการ

เป็นที่ยอมรับกันว่าเทคโนโลยีสารสนเทศและเครือข่ายอินเทอร์เน็ตเป็นเครื่องมือธุรกิจที่จำเป็นในการดำเนินธุรกิจยุคใหม่ ซึ่งเว็บแอปพลิเคชันเป็นเครื่องมือที่สำคัญในการทำธุรกิจบนโลกอินเทอร์เน็ต แต่ในปัจจุบันเว็บแอปพลิเคชันกำลังจะกลายเป็นเพียงตัวประกอบ เนื่องจากเว็บแอปพลิเคชันสามารถตอบสนองความคิด Distributed Processing ได้ในระดับหนึ่งเท่านั้น ด้วยเหตุนี้องค์กรธุรกิจจึงเริ่มหันมาให้ความสนใจกับเทคโนโลยีเว็บเซอร์วิส ซึ่งเข้ามาช่วยเพิ่มประสิทธิภาพให้กับระบบการดำเนินธุรกิจขององค์กร และยังสามารถทำงานร่วมกับเว็บแอปพลิเคชันเดิมได้เป็นอย่างดี

เว็บเซอร์วิสเป็นเทคโนโลยีที่ถูกนำมาใช้ในการพัฒนาอย่างรวดเร็วและแพร่หลาย เนื่องจากเว็บเซอร์วิสสามารถเข้าถึงได้ทุกที่ผ่านทางเว็บ ไม่ขึ้นกับแพลตฟอร์มและรูปแบบของภาษา โดยเว็บเซอร์วิสทำให้แอปพลิเคชันต่าง ๆ นั้นสามารถติดต่อกันได้โดยใช้ภาษาเอ็กซ์เอ็มแอล (XML) ซึ่งเป็นภาษากลางในการสื่อสารผ่านเครือข่ายคอมพิวเตอร์ เว็บเซอร์วิสสามารถที่จะทำงานในสถานะแวดล้อมที่แตกต่างกันได้โดยผ่าน โพรโตคอลกลาง แต่เนื่องจากบางระบบจะต้องใช้ความปลอดภัยค่อนข้างสูงอย่าง เช่น ระบบเกี่ยวกับการเงิน จึงจำเป็นต้องมีการรักษาความปลอดภัยให้กับข้อมูลด้วย เทคโนโลยีที่ใช้ในด้านความปลอดภัยมีหลายอย่างแต่เทคโนโลยีในด้านความปลอดภัยที่ใช้แพร่หลายอยู่ในปัจจุบันนี้ คือ เอสเอสแอล (SSL) แต่ก็ยังไม่สามารถตอบสนองความต้องการด้านความปลอดภัยในเว็บเซอร์วิสได้ทั้งหมด จึงได้มีการจัดทำเว็บเซอร์วิสซีเคียวริตี้ขึ้นมา ซึ่งสามารถตอบสนองความต้องการในด้านรักษาความปลอดภัยบนเว็บเซอร์วิสได้มากกว่าแบบเดิม นอกจากนี้ยังมีเว็บเซอร์วิสบางตัวที่ต้องเรียกใช้เว็บเซอร์วิสอื่น เพื่อสร้างเซอร์วิสของตัวเอง เช่น เว็บเซอร์วิสของบริษัททัวร์ ที่ต้องเรียกใช้เว็บเซอร์วิสของโรงแรมเพื่อจองห้องพัก, สายการบินเพื่อจองตั๋ว และธนาคารเพื่อตัดยอดเงิน แต่เนื่องจากเว็บเซอร์วิสประเภทนี้อาจจะมีการเปลี่ยนแปลง business logic อยู่บ่อยครั้งดังนั้นการผูกแต่ละเว็บเซอร์วิสไว้ด้วยการ coding แบบเก่านั้น จึงไม่เหมาะกับการสร้าง business process ด้วยเว็บเซอร์วิสตามหลักการของ SOA ดังนั้นจึงมีการใช้ Business Process Execution Language (BPEL) เพื่อช่วยในการแก้ปัญหาดังกล่าว

1.2 วัตถุประสงค์ของโครงการ

1.2.1 ศึกษาการทำงาน, สถาปัตยกรรม และส่วนประกอบต่างๆของเว็บเซอร์วิส

1.2.2 ศึกษาการรักษาความปลอดภัยของข้อมูลในการส่งผ่านเครือข่ายโดยใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WS-security และ SSL

- 1.2.3 เพื่อศึกษาการนำเอาเทคโนโลยี WS-transaction มาใช้ในการเรียกเว็บเซอร์วิส
- 1.2.4 ศึกษาการสร้างเว็บเซอร์วิสที่พัฒนาโดยจาวารวมไปถึงวิธีการที่จะเรียกใช้เว็บเซอร์วิสดังกล่าว
- 1.2.5 ศึกษาการสร้าง Business Process Execution Language (BPEL) เพื่อช่วยเป็นตัวกลางในการเรียกใช้เว็บเซอร์วิส
- 1.2.6 พัฒนาระบบ Online Booking System เพื่อทดลองเรียกใช้เว็บเซอร์วิส โดยมีการรับความต้องการของผู้ใช้ผ่านทางเว็บแอปพลิเคชัน

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1.3.1 ได้รับความรู้ความเข้าใจเกี่ยวกับการสร้างเว็บเซอร์วิสและสามารถนำไปใช้อย่างถูกต้อง
- 1.3.2 เข้าใจหลักการทำงานพื้นฐานของการสร้างความปลอดภัยให้กับข้อมูลที่มีในเว็บเซอร์วิส
- 1.3.3 เข้าใจหลักการของ Business Process Execution Language (BPEL) และสามารถสร้าง BPEL เพื่อเรียกใช้เว็บเซอร์วิสอื่นได้
- 1.3.4 สามารถสร้างเว็บเซอร์วิสขึ้นมาใช้งานได้ถูกต้องและมีประสิทธิภาพ
- 1.3.5 สามารถสร้างเว็บเซอร์วิสที่มีความปลอดภัยในการส่งข้อมูล โดยการเพิ่มคุณสมบัติที่มีในเว็บเซอร์วิสซีเคียวริตี้เพิ่มเข้าไป
- 1.3.6 สามารถสร้างเว็บเซอร์วิสที่รองรับการใช้งานทรานแซกชัน
- 1.3.7 สามารถสร้างเว็บแอปพลิเคชันเพื่อเป็นส่วนติดต่อกับผู้ใช้ เพื่อเรียกใช้งานเว็บเซอร์วิส

1.4 ขอบเขตของโครงการ

โครงการนี้จะศึกษาเกี่ยวกับรายละเอียดของเว็บเซอร์วิสทั้งหมด ซึ่งประกอบไปด้วย การทำงานของเว็บเซอร์วิส, สถาปัตยกรรมของเว็บเซอร์วิส, เทคโนโลยีที่ใช้ในเว็บเซอร์วิส อันได้แก่ XML, SOAP, WSDL และ UDDI นอกจากนี้มีการพัฒนาระบบที่ให้บริการเกี่ยวกับการจองออนไลน์ โดยมีการสร้างเว็บเซอร์วิสสำหรับการจองตั๋วเครื่องบิน, จองโรงแรม และการจองแพคเกจทัวร์ท่องเที่ยว โดยใช้โปรแกรม Rational Application Developer 6.0 ในการพัฒนาเว็บเซอร์วิส นอกจากนี้ยังมีการนำเอามาตรฐานต่างๆ ของเว็บเซอร์วิสมาประยุกต์ใช้ในระบบ เช่น การใช้ ws-security เพื่อรักษาความปลอดภัยของข้อมูลที่ใช้สำหรับเว็บเซอร์วิส การใช้ ws-transaction ในการทำธุรกรรมต่างๆของระบบ รวมถึงการสร้าง Business Process Execution Language (BPEL) เพื่อเป็นตัวกลางในการเรียกใช้งานเว็บเซอร์วิส ซึ่ง BPEL จะถูกใช้โดยเว็บแอปพลิเคชันที่สร้างจาก JavaServer Pages (JSP) อีกทีหนึ่ง เพื่อรับความความต้องการของผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 วิธีการดำเนินงาน

- 1.5.1 ศึกษาสถาปัตยกรรม และรายละเอียดของเว็บเซอร์วิสในส่วนต่างๆ เช่น XML, SOAP, WSDL, UDDI รวมไปถึงมาตรฐานต่างๆของเว็บเซอร์วิส เช่น ws-security, ws-transaction, และ BPEL (Bpel Execution Language)
- 1.5.2 เลือกเครื่องมือที่จะนำมาใช้ในการสร้างเว็บเซอร์วิส ซึ่งในโครงการนี้ได้เลือกใช้โปรแกรม Rational Application Developer 6.0 และศึกษาวิธีการสร้างเว็บเซอร์วิสโดยใช้เครื่องมือนี้
- 1.5.3 เลือกเครื่องมือที่จะนำมาใช้ในการสร้าง BPEL (Business Execution Language) ซึ่งในโครงการนี้ได้เลือกใช้ โปรแกรม ActiveBpel Designer และศึกษาวิธีการใช้เครื่องมือดังกล่าว
- 1.5.4 พัฒนาเว็บเซอร์วิสที่ให้บริการในการจองตั๋วเครื่องบิน, การจองโรงแรม และแพคเกจทัวร์ โดยเพิ่มการรักษาความปลอดภัยของข้อมูลโดยใช้ ws-security และ SSL และเรื่องของ ws-transaction ในการทำธุรกรรมต่างๆของระบบ
- 1.5.5 ศึกษารายละเอียดและวิธีการสร้าง Business Process Execution Language (BPEL)
- 1.5.6 สร้าง Business Process Execution Language (BPEL) เพื่อใช้เป็นตัวกลางในการไปเรียกใช้เว็บเซอร์วิสอื่น
- 1.5.7 สร้างเว็บแอปพลิเคชันโดยใช้ JavaServer Pages (JSP) สำหรับให้ผู้ใช้สามารถใช้งานระบบที่สร้างขึ้นมา โดยเป็นตัวกลางไปเรียกใช้ BPEL (Business Process Execution Language)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีพื้นฐานเกี่ยวกับเว็บเซอร์วิส

2.1 เว็บเซอร์วิสคืออะไร

เว็บเซอร์วิส คือ แอปพลิเคชันหรือซอฟต์แวร์ที่ทำงานอย่างใดอย่างหนึ่งในลักษณะให้บริการ โดยจะถูกเรียกใช้งานจากแอปพลิเคชันอื่นๆ ผ่านทางหน้าเว็บ ในรูปแบบ RPC (Remote Procedure call) โดยที่ภาษาที่ใช้ในการติดต่อสื่อสารระหว่างคอมพิวเตอร์คือภาษาเอ็กซ์เอ็มแอล (XML)

การอธิบายวิธีการใช้งานของเว็บเซอร์วิส นั้น จะอธิบายโดยใช้ภาษาวิสเดิล WSDL (Web Service Description Language) ซึ่งเป็นภาษา XML ประเภทหนึ่ง

ระบบอื่นๆ จะสามารถติดต่อและทำงานกับเว็บเซอร์วิสได้ โดยใช้โปรโตคอลที่ชื่อว่า SOAP (Simple Object Protocol) ซึ่งใช้ภาษา XML เป็นมาตรฐานในการติดต่อระหว่างระบบผ่านทางโปรโตคอลอื่นที่ใช้ในการส่งข้อมูลบนเว็บ อาทิเช่น โปรโตคอล HTTP

2.2 เปรียบเทียบระหว่างเว็บแอปพลิเคชันกับเว็บเซอร์วิส

ทั้งเว็บเซอร์วิสและเว็บแอปพลิเคชันต่างใช้โปรโตคอล HTTP หรือ อินเทอร์เน็ต เป็นช่องทางในการสื่อสารเหมือนกัน แต่มีวัตถุประสงค์ต่างกัน โดย เว็บแอปพลิเคชันใช้เพื่อการแลกเปลี่ยน HTML ระหว่างเว็บเซิร์ฟเวอร์ แต่ เว็บเซอร์วิสเป็นการแลกเปลี่ยน “บริการ” (ก้อนโปรแกรม Software Components) ระหว่างระบบสารสนเทศ ผ่านเว็บเซิร์ฟเวอร์

ในเรื่องของความสามารถโดยส่วนใหญ่จะใช้เว็บแอปพลิเคชัน ในการติดต่อกับผู้ใช้ผ่านทาง Internet browser เพื่อนำเสนอข้อมูลและการทำธุรกรรมต่างๆ ส่วนเว็บเซอร์วิส จะทำหน้าที่ในการติดต่อกับเว็บเซิร์ฟเวอร์ เพื่อแลกเปลี่ยนข้อมูลและการทำงานหรือให้บริการข้ามระบบกันโดยใช้เว็บแอปพลิเคชัน หรือ Application Interface ในการติดต่อกับผู้ใช้ นอกจากนี้ เว็บเซอร์วิสยังสามารถทำงานกับระบบต่างๆ ได้มากกว่า 1 ระบบ ในขณะที่เว็บแอปพลิเคชันไม่สามารถทำได้โดยตรง ซึ่งสามารถสรุปการเปรียบเทียบได้ดังตาราง

หัวข้อ	เว็บเซอร์วิส	เว็บแอปพลิเคชัน
การเชื่อมต่อ	program-program	human-program
ภาษาที่ใช้	XML	HTML, XML
รายชื่อการให้บริการ	ค้นหาผ่าน UDDI	ค้นหาผ่าน search engine
ขอบเขตการใช้งาน	Business-to-Business (B2B) Business-to-Customer (B2C)	Business-to-Customer (B2C)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	เว็บเซอร์วิส	เว็บแอปพลิเคชัน
โปรโตคอล(Protocol)	SOAP+HTTP	HTTP

ตารางที่ 2.1 การเปรียบเทียบระหว่าง เว็บแอปพลิเคชันกับเว็บเซอร์วิส

หลังจากที่ได้ทำความเข้าใจความหมายและข้อเปรียบเทียบระหว่างเว็บแอปพลิเคชันและเว็บเซอร์วิสแล้ว ความแตกต่างระหว่างเว็บทั้งสองจะเห็นได้ชัดในเรื่องของความสามารถ แต่ในการนำไปใช้งานจำเป็นจะต้องประยุกต์ใช้เว็บทั้งสองชนิดร่วมกัน เพื่อให้เกิดประสิทธิภาพมากที่สุด

2.3 Service Oriented Architecture (SOA)

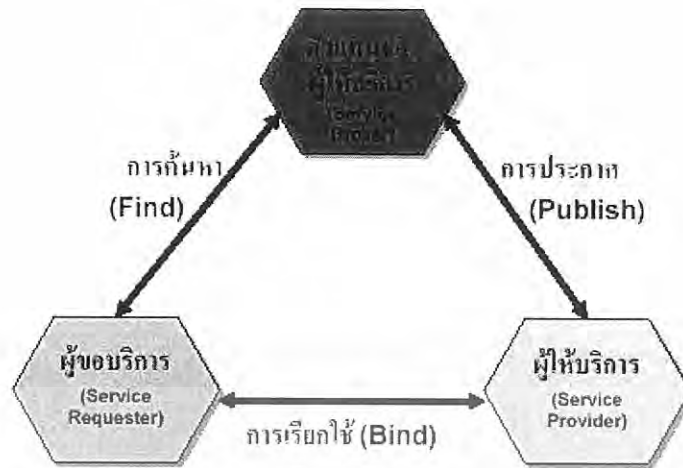
เว็บเซอร์วิสมีสถาปัตยกรรมแบบ Service Oriented Architecture (SOA) ซึ่งเป็นสถาปัตยกรรมที่เน้นการให้บริการ โดย SOA เป็นรูปแบบในการออกแบบระบบ (system) หรือ application แบบหนึ่ง อย่าง object-oriented เวลาออกแบบระบบเราจะมองเป็น object หรือ class แยกแยะระบบออกมาเป็น class และความสัมพันธ์ แต่ SOA จะมองระบบประกอบด้วยการทำงานหรือบริการ (service) ต่างๆ ซึ่งเซอร์วิสก้คล้ายกับฟังก์ชันหรือเมธอด กล่าวคือ มีหน้าที่ทำอะไรสักอย่างเมื่อมีการเรียกใช้ แต่เซอร์วิสมีลักษณะเป็น high-level และเป็นมุมมองจาก business จริงๆ มากกว่า เช่น มองระบบธนาคารประกอบด้วย withdrawal service, Deposit service แทนที่จะมองเป็น class Customer หรือ class Teller ที่มี operation withdrawal หรือ deposit นอกเหนือจากนั้นสิ่งที่เซอร์วิสทำได้ แต่เมธอดทั่วไปทำไม่ได้ คือ distributed system นั่นคือแต่ละ service สามารถถูกเรียกจากต่างระบบ, host หรือ JVM กันได้ และแต่ละ service มีอิสระต่อกัน

2.3.1 ส่วนประกอบเบื้องต้นของ SOA

SOA มีส่วนประกอบหลักด้วยกันสามส่วน คือ

- **Service Provider** (ผู้ให้บริการ) มีหน้าที่ในการสร้างบริการที่เป็นประโยชน์ และสร้างรายละเอียดของบริการนั้นๆ แล้วกระจายรายละเอียดนั้นออกไปยังระบบลงทะเบียนต่างๆ และรับการร้องขอบริการจากผู้ร้องขอบริการต่างๆ
- **Service Request** (ผู้ร้องขอบริการ) มีหน้าที่ในการค้นหาบริการที่อยู่ภายในระบบลงทะเบียนต่างๆ เพื่อนำรายละเอียดบริการมาร้องขอบริการที่กำหนดโดยผู้ให้บริการ
- **Service Broker** (ตัวแทนของผู้ให้บริการ) มีหน้าที่ในการกระจายข่าวรายละเอียดของบริการที่เกิดขึ้นจากผู้ให้บริการและอนุญาตให้ผู้ร้องขอบริการใช้การค้นหารายละเอียดของบริการที่อยู่ภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 สถาปัตยกรรมแบบ SOA

องค์ประกอบทั้งสามจะทำงานร่วมกัน โดยใช้ฟังก์ชันพื้นฐาน ดังนี้

- Publish (การประกาศ) เป็นการติดต่อระหว่างผู้ให้บริการกับตัวแทนของผู้ให้บริการ โดยผู้ให้บริการจะประกาศบริการของตนไว้กับตัวแทนของผู้ให้บริการ เพื่อให้ผู้ขอบริการมาทำการค้นหาบริการที่ต้องการได้
- Find (การค้นหา) เป็นการติดต่อระหว่างผู้ขอบริการกับตัวแทนของผู้ให้บริการ โดยผู้ขอบริการจะใช้ขั้นตอนในการค้นหาเพื่อค้นหาบริการที่ตรงกับความต้องการ
- Bind (การเรียกใช้) เป็นการติดต่อระหว่างผู้ขอบริการกับผู้ให้บริการ โดยผู้ขอบริการจะทำการเรียกใช้บริการจากผู้ให้บริการ

2.4 เทคโนโลยีพื้นฐานของเว็บเซอร์วิส

Service Directory	: UDDI
Service Description	: WSDL
Service Interactive	: SOAP
Data Format	: XML
Communication Protocol	: HTTP
Communication Network	: Internet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 XML (Extensible Markup Language)

XML เป็นภาษา Markup ที่เป็น text-based ซึ่งเป็นมาตรฐานที่ทุกระบบสนับสนุน ทำให้ข้อมูลที่มีโครงสร้างของภาษา XML จะถูกนำไปประมวลผลต่ออย่างอัตโนมัติได้อย่างง่ายดาย ภาษา XML จึงถูกนำมาใช้เป็นภาษามาตรฐานในการแลกเปลี่ยนข้อมูลของ Web Services

ลักษณะที่สำคัญของภาษา XML

- สิ่งที่แตกต่างกันระหว่าง XML และ HTML คือ XML ถูกออกแบบมาเพื่อใช้แลกเปลี่ยนข้อมูล ไม่ใช่มาแทนที่ HTML ซึ่ง XML และ HTML ถูกออกแบบมาเพื่อจุดประสงค์ที่ต่างกัน คือ HTML จะเกี่ยวข้องกับการแสดงข้อมูล ส่วน XML จะเกี่ยวข้องกับการอธิบายข้อมูล
- XML ไม่มีแท็ก (tag) ที่สร้างขึ้น เราจะเป็นผู้กำหนดนิยาม และความหมายของแท็กตามข้อตกลงของ W3C ซึ่งทำให้การอ่านเอกสารเป็นไปได้ง่ายและเป็นมาตรฐาน
- XML สามารถรองรับการแลกเปลี่ยนข้อมูลระหว่างแอปพลิเคชัน โดยไม่ขึ้นอยู่กับแพลตฟอร์ม
- XML เป็นภาษาที่ได้รับการออกแบบมาเพื่อให้สามารถนิยามความหมายของข้อมูลได้ จึงมีการจัดโครงสร้างข้อมูล แบ่งข้อมูลออกเป็นหมวดหมู่ และส่วนประกอบย่อย
- ส่วนของข้อมูล และส่วนการแสดงผลของเอกสาร XML ถูกแยกออกจากกันอย่างชัดเจน โดยที่ในเอกสาร XML นั้นจะมีแต่ตัวเนื้อข้อมูล จึงทำให้การเปลี่ยนแปลงข้อมูลไม่ส่งผลกระทบต่อการแสดงผล และการแก้ไขส่วนแสดงผลก็จะไม่ส่งผลกระทบต่อตัวเนื้อข้อมูล

2.4.1.1 XML และ HTML

HTML เป็นภาษาที่ใช้ในการเขียน Web มากที่สุดนั่นเป็นเพราะมีรูปแบบที่ง่ายต่อการแสดงผลของ Browser เนื่องจาก มี tag ด้ายตัวที่สามารถบอกได้ว่าเมื่อเจอ tag นี้จะแสดงผลอย่างไร เช่น เมื่อเจอ tag ... ในเอกสารก็ให้แสดงข้อความที่อยู่ระหว่าง tag เป็นตัวหนา แต่จะสังเกตเห็นได้ว่าคอมพิวเตอร์จะไม่เข้าใจว่าข้อความนั้นคืออะไร เพียงแต่รู้ว่าจะแสดงผลอย่างไร นั่นแสดงว่าไม่สามารถนำข้อมูลภายใน tag เหล่านี้ไปทำการประมวลใดๆ ได้เลย

XML เป็นภาษาที่มีลักษณะเป็น tag คล้าย HTML แต่ไม่ได้มุ่งที่การแสดงผล XML มุ่งที่การสื่อความหมาย โดยอนุญาตให้ผู้ใช้สามารถกำหนด tag ขึ้นได้เอง เพื่อให้สื่อความหมายทางภาษาของมนุษย์ แต่คอมพิวเตอร์เองก็เข้าใจเช่นกัน ทำให้ข้อมูลระหว่าง tag สามารถนำไปประมวลผลต่อได้ เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<note date = "12/11/2005">
  <to>Tove</to>
  <address>
    <street>123 Main St.</street>
    <city>San Francisco</city>
    <state>CA</state>
    <zip>94117</zip>
  </address>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>

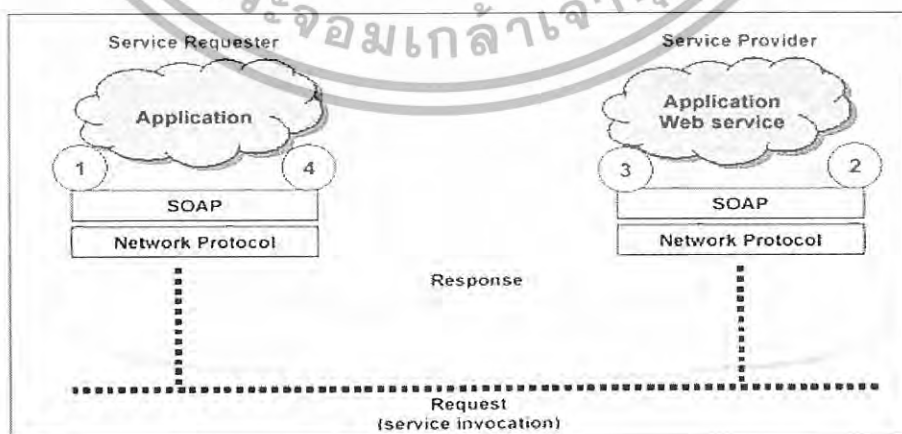
```

รูปที่ 2.2 ตัวอย่างเอกสาร XML

2.4.2 SOAP (Simple Object Access Control)

SOAP เป็นมาตรฐานที่ใช้เพื่อกำหนดการส่งข้อมูลของผู้ให้บริการและรับบริการ เป็นโปรโตคอลที่ผู้จัดทำ Web Services เรียกใช้ในการส่ง message ระหว่าง Web Services SOAP เป็น Transport Protocol ที่มี XML เป็นพื้นฐานและใช้ HTTP เป็นโปรโตคอลร่วมในการส่งผ่านเครือข่าย SOAP จะระบุวิธีในการเข้ารหัสส่วนหัว (Header Encoding) ของทั้ง HTTP และไฟล์ XML ให้อย่างชัดเจน ในส่วนของการติดต่อไปยังคอมพิวเตอร์เครื่องหนึ่งและส่งผ่านข้อมูลไปให้ รวมถึงระบุวิธีที่โปรแกรมซึ่งถูกเรียกนั้นจะส่งค่าคืนกลับมาด้วย SOAP เป็น XML-based โปรโตคอล และใช้ HTTP เป็นโปรโตคอลร่วม สำหรับการแลกเปลี่ยนข้อมูล

SOAP ได้กำหนดคุณสมบัติของโปรโตคอล (Messaging Protocol) ระหว่างผู้ขอบริการกับผู้ให้บริการในการติดต่อสื่อสารกัน เช่น กำหนดให้ผู้ขอบริการต้องส่งข้อมูลที่ระบุฟังก์ชันและค่าพารามิเตอร์ต่างๆ ที่จำเป็นต้องใช้ในแอปพลิเคชันที่ร้องขอ ส่งไปให้กับผู้ให้บริการ ซึ่งแอปพลิเคชันของผู้ให้บริการก็จะทำงานตามกระบวนการที่ถูกร้องขอมา โดยมีการทำงานดังรูป



รูปที่ 2.3 XML messaging using SOAP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนในการส่ง SOAP Message เพื่อทำการติดต่อ ระหว่างผู้เรียกใช้บริการกับผู้ให้บริการมีดังนี้

1. แอปพลิเคชันของผู้ขอบริการสร้าง SOAP Message เพื่อเรียกใช้บริการของเว็บเซอร์วิส
2. เว็บเซอร์วิสของผู้ให้บริการได้รับ SOAP Message จากผู้ร้องขอ ซึ่งอยู่ในรูปแบบ XML
3. เว็บเซอร์วิสประมวลผลตามคอมพิวเตอร์ที่ให้บริการแล้วผู้ให้บริการก็จะสร้าง SOAP Message ที่มีผลลัพธ์นั้นกลับไปยังผู้ร้องขอบริการ
4. แอปพลิเคชันของผู้ขอบริการได้รับผลลัพธ์ที่เป็น SOAP Message แล้วทำการแปลงให้อยู่ในรูปแบบที่ต้องการ เพื่อนำไปประมวลผลต่อ

2.4.2.1 โครงสร้างของ SOAP

เอกสาร SOAP นั้นมีโครงสร้างในรูปแบบ XML ซึ่งเราสามารถแบ่งส่วนของเอกสารได้เป็น 3 ส่วนหลัก ดังนี้

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



รูปที่ 2.4 โครงสร้างของเอกสาร SOAP

1. Envelope Element

SOAP เมสเสจทั้งหมดจะถูกห่อหุ้มในเอ็นวีโลบ (Envelope) ซึ่งมันจะเป็นอีลีเมนต์ที่อยู่บนสุดของ SOAP เมสเสจ นอกจากนี้ในส่วนของเอ็นวีโลบนั้นจะมีการใช้แอทริบิวต์ encodingStyle ซึ่งจากตัวอย่างด้านล่างมีการอ้างการ encoding จาก

"<http://www.w3.org/2001/12/soap-encoding>"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
soap:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">
...
  Message information goes here
...
</soap:Envelope>
```

รูปที่ 2.5 SOAP Envelope Element

2. Header Element

เฮดเดอร์อิตีเม้นต์จะถูกใช้ในการส่งเมตาอินฟอร์เมชัน (meta information) ที่เกี่ยวกับ SOAP เมสเสจ แต่เมตาอินฟอร์เมชันนี้จะไม่มีความจำเป็นสำหรับส่วนของการเรียกใช้งานฟังก์ชัน ใน SOAP เมสเสจจะมีการกำหนดค่า `mustUnderstand` attribute ให้อยู่ในส่วนของ เฮดเดอร์อิตีเม้นต์ ซึ่งจะใช้กรณีที่เซิร์ฟเวอร์เกิดไม่เข้าใจอิตีเม้นต์นั้นก็จะมีการส่งเมสเสจแสดงความผิดพลาด (fault message) กลับมาบอก ซึ่งกระบวนการนี้จะทำให้เครื่องไคลเอนต์มั่นใจได้ว่า เครื่องเซิร์ฟเวอร์สามารถที่จะประมวลผลได้ โดยค่าแอททริบิวต์ตัวนี้มีค่าได้ 2 ค่า คือ 0 แสดงว่าไม่ต้องประเมินผล และ 1 แสดงว่าจำเป็นต้องมีการประเมินผล ดังนั้นตัวอย่างข้างล่างนี้จะใช้เฮดเดอร์สำหรับส่งค่าไอดี(ID) จากไคลเอนต์ไปยังเซิร์ฟเวอร์ ซึ่งเกี่ยวกับเรื่องระบบรักษาความปลอดภัย

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
soap:encodingStyle="
http://schemas.xmlsoap.org/soap/envelope/http://www.w3.org/2001/12/soap-
encoding">
<soap:Header>
<m:Trans
xmlns:m="http://www.w3schools.com/transaction/"
soap:mustUnderstand="1">234</m:Trans>
</soap:Header>
...
```

รูปที่ 2.6 SOAP Header Element

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Body Element

เป็นส่วนที่ต้องมีอยู่ใน SOAP เมสเสจเสมอ และจะบรรจุใจความสำคัญของ SOAP เมสเสจ ซึ่งชนิดของเมสเสจนั้นจะบรรจุส่วนของคำร้องขอใช้งานเมธอดโดยใช้ชื่อของเมธอดและพารามิเตอร์ของเมธอดนั้น ซึ่งจะอยู่ในรูปแบบของ XML โดยเฉพาะรูปแบบของพารามิเตอร์ที่ส่งจะเป็นส่วนสำคัญมากสำหรับการส่งข้อมูลที่ประมวลผลที่ได้จากผู้ใช้บริการกลับมาในรูปแบบ XML และถ้าหากมีข้อผิดพลาดก็จะบอกในส่วนนี้

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
<soap:Body>
  <m:GetPrice
xmlns:m="http://www.w3schools.com/prices">
    <m:Item>Apples</m:Item>
  </m:GetPrice>
</soap:Body>
</soap:Envelope>
```

รูปที่ 2.7 ตัวอย่าง SOAP รีเควสต์เมสเสจ

```
<?xml version="1.0"?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
<soap:Body>
  <m:GetPriceResponse
xmlns:m="http://www.w3schools.com/prices">
    <m:Price>1.90</m:Price>
  </m:GetPriceResponse>
</soap:Body>
</soap:Envelope>
```

รูปที่ 2.8 ตัวอย่าง SOAP เรสพอนส์เมสเสจ

3.1 SOAP Fault Element

เป็นส่วนที่จะถูกใช้เมื่อมีข้อผิดพลาดในการประมวลผลของเอกสาร SOAP ซึ่งโดยปกติจะเห็นเฉพาะที่เป็นเอกสารตอบกลับเท่านั้น (Reply Message หรือ Answer Message) โดยถ้ามี Fault Element ส่วนนี้จะอยู่ในตำแหน่งของ child element ของส่วน Body

เอกสารนี้เป็นเอกสารที่ส่งวนไว้ SOAP Fault Element มี Sub Element ดังนี้ อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

ตารางที่ 2.2 Sub Element ของ Fault Element

2.4.3 WSDL (Web Services Description Language)

WSDL นั้นเป็นภาษาที่ใช้ XML เป็นพื้นฐาน ซึ่งใช้ในการบรรยายเว็บเซอร์วิสหรือเน็ตเวิร์คเอ็นพอยต์ (network endpoint) เพื่อเป็นการเผยแพร่ข้อมูลการให้บริการแก่ระบบภายนอก โดยที่ทำงานผ่านระบบเครือข่าย โดยปราศจากการแทรกแซงของคน นอกจากนี้ WSDL นั้นยังสามารถที่จะบรรยายการส่งเมสเสจระหว่างเว็บเซอร์วิสระบุตำแหน่งที่อยู่ของเว็บเซอร์วิสและรวมไปถึงโปรโตคอลที่ใช้ในการติดต่อสื่อสารกันของเว็บเซอร์วิส WSDL นั้นจะทำงานร่วมกับ SOAP และ UDDI เพื่อที่จะทำให้เว็บเซอร์วิสติดต่อกับเว็บเซอร์วิสอื่นๆ ได้บนระบบอินเทอร์เน็ต ถ้าไม่มี WSDL แล้วการกระทำเมสเสจจิงอินเทอร์เฟซ (messaging interface) นั้นจะต้องทำเอง WSDL คือ มาตรฐานสำหรับการประกาศโปรเซสที่จำเป็นในการเรียกใช้เซอร์วิส

2.4.3.1 โครงสร้างเอกสาร WSDL

```

<definitions>
<types>
  definition of types.....
</types>
<message>
  definition of a message.....
</message>
<portType>
  <operation>
    definition of a port.....
  </operation>
</portType>
<binding>
  <operation>...</operation>
  definition of a binding....
</binding>
<service>
  definition of a service....
</service>
</definitions>

```

รูปที่ 2.9 โครงสร้างเอกสาร WSDL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Element	Definition
Definitions	<p>กำหนดรายละเอียดของเซอร์วิส โดยอ็ลิมেন্টนี้จะสนับสนุนแอทริบิวต์ (attributes) ต่างๆดังนี้ :</p> <ul style="list-style-type: none"> • name (optional) • targetNamespace : เป็น logical namespace สำหรับข้อมูลเกี่ยวกับเซอร์วิส เอกสาร WSDL สามารถ import เอกสาร WSDL อื่น และกำหนด targetNamespace ให้เป็นค่าที่ unique เพื่อไม่ให้เกิดปัญหาNamespace ซ้ำกัน • xmlns : เป็น default namespace ของเอกสาร WSDL และจะถูกกำหนดค่าให้เป็น http://schemas.xmlsoap.org/wsdl/ เพื่อใช้ WSDL elements มาตรฐาน เช่น <definitions>, <types> และ <message> ที่อยู่ใน namespace นี้ • xmlns:xsd และ xmlns:soap ; เป็น standard namespace definitions ที่ถูกใช้สำหรับการระบุข้อมูล SOAP-specific เช่นเดียวกับ data types. • xmlns:tns หมายถึง this namespace.
Types	<p>ให้ข้อมูลเกี่ยวกับ data types ใดๆที่ซับซ้อนซึ่งถูกใช้ในเอกสาร WSDL แต่ถ้าใช้ data types ที่ไม่ซับซ้อน เอกสาร WSDL ก็ไม่มีความจำเป็นที่จะต้องใช้ในส่วนนี้</p>
message	<p>อธิบาย data elements ของ operation แต่ละ message อาจมีมากกว่าหนึ่งส่วนเทียบได้กับ parameter ของ function ในการเขียน โปรแกรม</p>
portType	<p>เป็นส่วนที่สำคัญที่สุดใน WSDL element อธิบาย operations ที่ web service มีให้บริการและ messages ที่เกี่ยวข้อง เทียบได้กับ function library หรือ module หรือ class ในการเขียน โปรแกรม</p>
operation	<p>อธิบาย method ที่ให้บริการ Web Services หนึ่งจะมี method จำนวนกี่ method ก็ได้</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

binding	อธิบาย format ของ operation และ protocol details ในแต่ละ port
service	สำหรับ web server จะมี Web Services จำนวนที่บริการก็ได้ และ ชื่อ Web Services ก็เป็นตัวจำแนกและบ่งบอกแต่ละบริการซึ่งห้ามมีชื่อซ้ำกัน ซึ่งภายใน service จะระบุ port address(es) ของการ binding โดย service จะเก็บรายละเอียดของ network endpoints หรือ ports.
port	กำหนด single endpoint ให้เป็นเหมือน address สำหรับ binding ดังนั้นมันจึงเป็นเหมือนตัวที่ใช้ในการกำหนด single communication endpoint.

ตารางที่ 2.3 คำอธิบายของแต่ละอีลิเมนต์ในเอกสาร WSDL

```
<?xml version = '1.0' encoding = 'windows-1252'?>
<!--Generated by the Oracle9i JDeveloper Web Services WSDL Generator-->
<!--Date Created: Wed Mar 06 12:59:39 IST 2002-->
<definitions
  name="CCServicesImpl"
  targetNamespace=
    "http://tempuri.org/jdeveloper/generated/oracle/otnsamples/ws/CCServicesImpl"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns=
    "http://tempuri.org/jdeveloper/generated/oracle/otnsamples/ws/CCServicesImpl"
  xmlns:ns1=
    "http://tempuri.org/jdeveloper/generated/oracle/otnsamples/ws/CCServicesImpl/schema">
  <types>
    <schema
      targetNamespace=
        "http://tempuri.org/jdeveloper/generated/oracle/otnsamples/ws/CCServicesImpl/schema"
      xmlns="http://www.w3.org/2001/XMLSchema"
      xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding"/>
    </types>
    <message name="validateCard0Request">
      <part name="cardNumber"
        type="xsd:string"/>
    </message>
    <message name="validateCard0Response"/>
    <portType name="CCServicesImplPortType">
      <operation name="validateCard">
        <input name="validateCard0Request" message="tns:validateCard0Request"/>
        <output name="validateCard0Response" message="tns:validateCard0Response"/>
      </operation>
    </portType>
    <binding name="CCServicesImplBinding" type="tns:CCServicesImplPortType">
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ในวงจำกัดเท่านั้น การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาต
 ไม่มีการเผยแพร่ทั้งต้นฉบับที่แก้ไขเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
<operation name="validateCard">
  <soap:operation soapAction="" style="rpc"/>
  <input name="validateCard0Request">
    <soap:body use="encoded" namespace="CCServices"
      encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
  </input>

  <output name="validateCard0Response">
    <soap:body use="encoded" namespace="CCServices"
      encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
  </output>
</operation>
</binding>
<service name="CCServicesImpl">
  <documentation>
Class providing credit card validation methods. This implementation
uses LUHN algorithm to validate 16 digit number cards.
</documentation>
  <port name="CCServicesImplPort" binding="tns:CCServicesImplBinding">
    <soap:address location="http://127.0.0.1:8988/vsm/CCServices"/>
  </port>
</service>
</definitions>

```

รูปที่ 2.10 ตัวอย่างเอกสาร WSDL

2.4.4 UDDI (Universal Description, Discovery, and Integration)

UDDI เป็นมาตรฐานที่ให้จุดพื้นฐาน APIs (Application Programming Interface) ของ SOAP ที่สามารถนำมาใช้ในการพัฒนาเป็นตัวแทนของผู้ให้บริการ (Service broker) UDDI ใช้สำหรับค้นหาบริการ ที่ต้องการและเมื่อได้มาแล้ว UDDI ยังจัดหาข้อตกลงในวิธีการที่จะใช้งาน รูปแบบการทำงานของ UDDI จะเปลี่ยนแปลงวิธีการค้นหาข้อมูลบนอินเทอร์เน็ต ที่เราใช้อยู่ในปัจจุบันจากการพิมพ์ชื่อ โดเมนเนมในช่องแอดเดรสบาร์ ไปเป็นการใช้คำสามัญหรือชื่อเฉพาะในการค้นหา และเข้าสู่เว็บไซต์แทน UDDI จะทำหน้าที่เหมือนสมุดหน้าเหลืองที่จำแนกผู้จดทะเบียนตามรูปแบบของธุรกิจ แยกประเภทตามกิจการ

เมื่อมีผู้ที่ต้องการลงทะเบียนเว็บไซต์ ในครั้งแรกจะต้องทำการลงทะเบียนเกี่ยวกับชื่อองค์กรผู้ให้บริการ, คำอธิบายรายละเอียดของเว็บไซต์, ข้อมูลการติดต่อ และหมายเลขอุตสาหกรรม จากนั้นจึงจะทำการลงทะเบียนเว็บไซต์ได้ ซึ่งลงทะเบียนโดยระบุ URL ของ WSDL และคำอธิบายรายละเอียดเว็บไซต์นั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

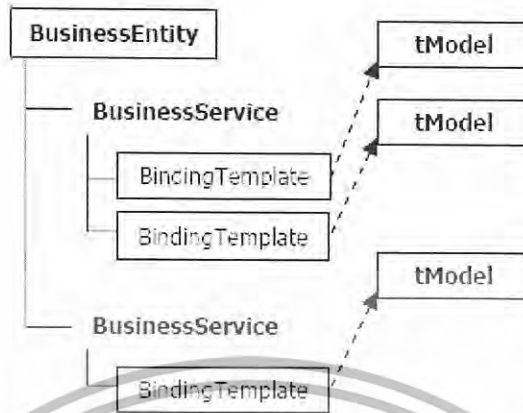
2.4.4.1 โครงสร้างข้อมูลของ UDDI

UDDI ประกอบไปด้วย 2 ส่วน คือส่วนของรายละเอียดข้อมูลของผู้ให้บริการ หรือผู้ลงทะเบียนเว็บเซอร์วิส ซึ่งจะอยู่ในรูปแบบของธุรกิจ (BusinessEntity) ที่อธิบายถึงบริการต่างๆ ที่มีอยู่ในธุรกิจนั้นๆ (BusinessService) โดยจะมีตัวแม่แบบ (BindingTemplate) ทำหน้าที่เชื่อมโยงไปยังส่วนที่สอง นั่นคือส่วนในการค้นหาที่เป็นรูปแบบของ tModel ซึ่งเป็นหัวใจหลักในการค้นหาข้อมูลที่ใช้มาตรฐาน UDDI



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรืออื่นๆความสัมพันธ์ของโครงสร้าง UDDI ในการลงทะเบียนและการค้นหาเว็บเซอร์วิส
 ดังรูป



รูปที่ 2.12 ความสัมพันธ์ของโครงสร้างของ UDDI กับ tModel

2.5 ประเภทของเว็บเซอร์วิสและวิธีการเรียกใช้

เราสามารถแบ่งเว็บเซอร์วิสออกได้เป็น 2 ประเภท คือ

1. Atomic web service – เป็นเว็บเซอร์วิสทำงานด้วยตัวเองไม่จำเป็นต้องพึ่งพาเว็บเซอร์วิสอื่น ซึ่งจุดประสงค์โดยส่วนใหญ่ของการเรียกใช้เว็บเซอร์วิสประเภทนี้ คือ เรียกใช้เพราะเราต้องการให้บริการจริงๆ ผู้เรียกมักเป็น end user client เช่น การที่ผู้ใช้ส่งข้อมูลไปยังเว็บเซอร์วิสของบริษัททัวร์ เพื่อซื้อ package ทัวร์
2. Composite web service – เป็นเว็บเซอร์วิสที่ต้องเรียกใช้เว็บเซอร์วิสอื่น เพื่อสร้างเซอร์วิสของตนเอง จุดประสงค์โดยส่วนใหญ่ของการเรียกใช้เว็บเซอร์วิสประเภทนี้ คือ เรียกใช้เพราะต้องการไปใช้เว็บเซอร์วิสอื่นต่อ ซึ่งก็คือ Composite Web Service นั่นเอง เช่น เว็บเซอร์วิสของบริษัททัวร์ เรียกใช้เว็บเซอร์วิสของโรงแรมเพื่อจองห้อง, สายการบินเพื่อจองตั๋ว และธนาคารเพื่อตัดยอดเงิน

การเรียกใช้เว็บเซอร์วิสมี 2 วิธีหลักๆ ได้แก่

1. ใช้ Web service client API เช่น JAVA proxy, WSIF, WSE เป็นต้น ก็คือเราเขียน application (ด้วย Java หรือ .NET) ของเราไปตามปกติ เมื่อถึงเวลาต้องเรียกใช้เว็บเซอร์วิสก็เรียกใช้ API แทน เหมือนกับการเขียนโปรแกรมปกติทั่วไป วิธีนี้เหมาะกับ end user client ที่เป็นผู้ใช้ปลายทางจริงๆ เพราะไม่ค่อยมีการเปลี่ยนแปลงบ่อยและมักเรียกเพียงไม่กี่เซอร์วิส อย่างไรก็ตามวิธีนี้ไม่เหมาะกับการสร้าง Composite Web Service เพราะเรามักใช้ Composite Web Service เพื่อสร้าง business process (business process คือ บริการที่เห็นหรือสัมผัสได้โดยตรงจากผู้ใช้หรือลูกค้า และให้ผลตอบแทนกับองค์กร นั่นก็คือ service นอกสุดที่ให้บริการลูกค้าโดยตรงนั่นเอง) ซึ่งมี business logic จึงอาจเกิดการเปลี่ยนแปลงได้บ่อยๆ นอกจากนี้

เอกสารนี้เป็นลิขสิทธิ์ของสำนักหอสมุดกลางพระจอมเกล้าลาดกระบัง การนำเอกสารนี้ไปเผยแพร่โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย

business goal ของการสร้าง application ด้วยเว็บเซอร์วิส ก็เพื่อความคล่องตัว (agility) ปรับเปลี่ยนง่าย จะได้สอดคล้องกับสภาพธุรกิจปัจจุบันที่มีการแข่งขันสูง ดังนั้นการผูกแต่ละเว็บเซอร์วิสไว้ด้วยการcoding แบบเก่านั้น จึงไม่เหมาะกับการสร้าง business process ด้วยเว็บเซอร์วิสตามหลักการของ SOA

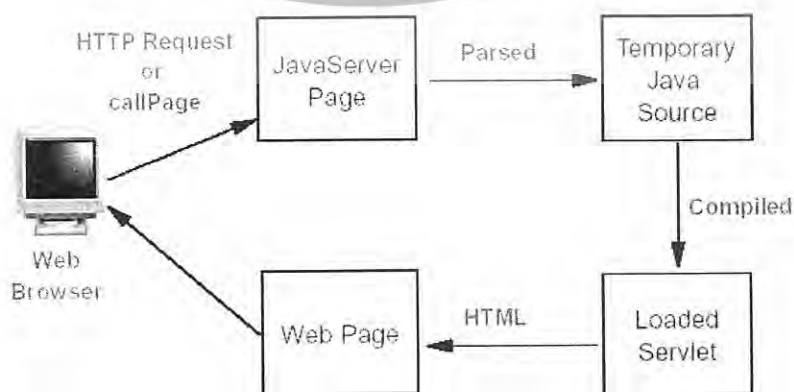
- ใช้เว็บเซอร์วิสตัวกลาง (mediator) มาเรียกใช้ Sub-Web Service แต่วิธีการนี้ไม่เหมาะสมตามที่ได้ออกมาแล้วข้างต้น จึงมีการนำ BPEL (Business Process Execution Language) มาใช้เพื่อสร้าง business process หรือ composite Web Service นั้นเอง

2.6 JavaServer Pages (JSP)

JavaServer Pages หรือ JSP นั้นมีการทำงานที่อิงบนพื้นฐานของเทคโนโลยีจาวาที่สร้างโค้ดออกมาในรูปแบบของไดนามิกเว็บเพจ โดยสามารถทำงานได้บนทุกแพลตฟอร์มและบนเว็บเซิร์ฟเวอร์

JSP นั้นมีการเขียนคล้ายกับภาษา Active Server Pages (ASP) ของทางไมโครซอฟท์ แต่ใช้งานร่วมกับคำสั่ง HTML และจาวาโค้ดที่คล้ายกับ VBScript แต่ภาษา JSP นั้นจะทำงานบนเว็บเซิร์ฟเวอร์ที่ ASP ไม่สามารถทำงานได้ เช่นเซิร์ฟเวอร์อย่าง Apache หรือ Netscape ซึ่งไลเซนส์ของ JSP ถูกพัฒนาโดยทาง Sun ซึ่งสามารถใช้งานได้ฟรี ซึ่งสามารถทำงานได้บนระบบปฏิบัติการ Solaris, Linux และบนวินโดวส์

JSP นั้นจะทำงานทางฝั่งของเซิร์ฟเวอร์คล้ายกับ จาวาสคริปต์ อาจเรียกได้ว่าเป็นการทำงานแบบ Server side ขั้นตอนการทำงานจะเริ่มตั้งแต่การร้องขอ หรือ เกิด Request จาก Browser หรือ Client มาที่ JSP บนฝั่ง Server จากนั้น Server ก็ทำการประมวลผล JSP เป็น servlet ก่อน แล้วส่ง Response กลับไปให้ Client ในรูปของ HTML ดังนั้นภาษา JSP จึงสามารถควบคุมความสามารถได้มากกว่าภาษาอื่นที่ออกมาก่อนหน้านี้เช่น Perl scripts หรือ API ทางฝั่งเซิร์ฟเวอร์อย่าง ASP เป็นต้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 2.13 การทำงานของ JavaServer Pages (JSP) ไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1 ตัวอย่างการเขียน JSP

```
File test.jsp
<html><body>
<%
JSP Command1;
JSP Command2;
JSP Command3;
%>
</body>
</html>
```

JSP มีลักษณะรูปแบบการเขียนเป็น HTML ผสมกับ tag ของจาวาแทรกกัน และไฟล์ที่เขียนด้วยเทคโนโลยีนี้ จะต้องกำหนดให้ไฟล์มีนามสกุลเป็น *.jsp เสมอ เพื่อบอกให้ server ทราบว่าจะต้องทำการประมวลผลกับไฟล์ที่ถูกร้องขออย่างไร การทำงานเมื่อทำการประมวลผล jsp ด้วย browser ใดๆ จะยังไม่สามารถทำการแสดงผลได้ในทันที แต่ jsp จะถูกทำการประมวลผลเป็น servlet ก่อน จากนั้น server จะทำการส่งผลลัพธ์หรือออกมาให้ browser เพื่อทำการแสดงผลอีกทีหนึ่ง (จะเห็นว่าการทำงานจะแตกต่างกับ HTML คือ JSP จะไม่สามารถทำการแสดงผลได้ในทันที แต่จะต้องมีการประมวลผลบน server ก่อน)

2.6.2 เปรียบเทียบ JSP กับเทคโนโลยีอื่นๆ

2.6.2.1 แอคทีฟเซิร์ฟเวอร์เพจ

แอคทีฟเซิร์ฟเวอร์เพจเป็นเทคโนโลยีที่ใช้หลักการคล้ายกับ JSP แต่เป็นเทคโนโลยีที่พัฒนาโดยบริษัทไมโครซอฟต์ โดยจะมีรูปแบบที่แตกต่างกันอย่างเด่นชัด 2 ประการคือ ประการแรกภาษาที่ใช้ใน JSP คือภาษาจาวา ส่วนภาษาที่ใช้ใน ASP คือภาษา VisualBasic หรือภาษาอื่นๆ ที่กำหนดโดยไมโครซอฟต์

ประการที่สองเป็นผลสืบเนื่องมาจากประการแรก คือ เนื่องจากใช้ภาษาจาวาทำให้ JSP สามารถนำไปใช้งานบนระบบปฏิบัติการใดก็ได้ แต่ ASP ต้องใช้ในระบบปฏิบัติการที่ไมโครซอฟต์กำหนดเท่านั้น

2.6.2.2 จาวาเซิร์ฟเล็ต

ตัว JSP สามารถทำงานได้เช่นเดียวกับที่เซิร์ฟเล็ตทำได้ แต่จุดเด่นที่สำคัญของ JSP คือ สะดวกในการสร้างและเปลี่ยนแปลงมากกว่า เพราะสามารถแก้ไขที่ไฟล์ได้โดยตรง และไม่จำเป็นต้องคอมไพล์ใหม่ก่อนนำไปใช้งาน

2.6.2.3 จาวาสคริปต์

เป็นการทำเนื้อหาไดนามิกให้กับเว็บเหมือนกัน แต่จาวาสคริปต์จะเกิดการประมวลผลที่ไคลเอ็นต์ ถ้าเว็บเบราว์เซอร์ไม่สนับสนุนจาวาสคริปต์ก็จะทำให้ไม่สามารถทำงานได้ แต่ JSP จะเกิดการประมวลผลที่ตัวเซิร์ฟเวอร์โดยตรงทำให้ไม่มีปัญหาเกี่ยวกับ

เบราว์เซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

Business Execution Language (BPEL)

3.1 Business Process Execution Language (BPEL)

BPEL มีรูปแบบอยู่ในรูปของภาษา XML กล่าวคือ BPEL ใช้ Syntax ทั้งหมดของ XML โดย BPEL เป็น orchestration language ไม่ใช่ choreography language ซึ่งข้อแตกต่างหลักระหว่างภาษาทั้งสองคือ orchestration จะเป็นภาษาที่ทำงานภายใน Business Process ส่วน choreography เป็นภาษาที่ทำงานระหว่าง business process นอกจากนี้ยังเป็นภาษาที่รวมเอาลักษณะเด่นจาก IBM's Web Service Flow Language (WSFL) ซึ่งใช้วิธีการของ directed graph และ Microsoft's XLANG ซึ่งเป็น orchestration language ที่ถูกใช้ใน BizTalk server และมีวิธีการแบบ block-structured เพื่อใช้ในการสร้าง business process ขนาดใหญ่ภายในเว็บเซอร์วิส BPEL มักจะถูกนำมาใช้งานภายในองค์กรหรือระหว่างองค์กร เพื่ออธิบายขบวนการการทำงานของระบบที่เป็น SOA เพื่อให้รวมเซอร์วิสต่างๆ เข้าด้วยกัน (assembling) ทำให้ขบวนการต่างๆ ทางเซอร์วิสที่มีความซับซ้อนสามารถทำงานร่วมกัน ได้อย่างมีประสิทธิภาพ

BPEL มักจะถูกเปรียบเทียบให้เป็น Music sheet ซึ่งก็คือ กระดาษที่บันทึกขั้นตอนการบรรเลงของอุปกรณ์ดนตรี ให้สามารถบรรเลงร่วมกันได้อย่างไพเราะ การจะทำให้เพลงใดๆ ออกมาได้ไพเราะนั้นขึ้นอยู่กับผู้เขียนบทบรรเลงบทเพลง เครื่องเล่นดนตรีเองก็เปรียบเสมือนเว็บเซอร์วิส ที่จะถูกเล่นโดยนักดนตรี นักดนตรีก็จะทำการบรรเลงโดยอาศัย music sheet ถ้า music sheet เขียนขึ้นมาดี อุปกรณ์ทางดนตรีต่างๆ ก็จะเล่นได้อย่างสอดคล้องประสานกัน แต่ถ้า music sheet เขียนได้ไม่ดี ก็อย่าหวังว่าเพลงที่บรรเลงออกมานักดนตรีชั้นยอดจะไพเราะได้

BPEL เป็นพื้นฐานหนึ่งของเว็บเซอร์วิส โดยภาพด้านล่างจะแสดงความสัมพันธ์ระหว่างมาตรฐานต่างๆ

BPEL4WS		Process
WS-Reliable Messaging	WS-Security	Quality Of Service
	UDDI	Discovery
	XSD, WSDL, WS-Policy	Description
	SOAP	Messaging
	XML	
	HTTP, MQ, SMTP	Transport

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3.1 Web services technology stack ให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจะสร้าง business process หรือ composite Web Service จาก BPEL ต้องประกอบด้วย 3 ส่วน (BPEL Component) ได้แก่

1. BPEL Designer

BPEL Designer (ซึ่งเป็น Graphic User Interface) จะใช้ในการกำหนด, ออกแบบ business process ซึ่ง BPEL Designer จะทำการสร้าง (generate) BPEL file หรือ Process flow template ซึ่งจะอยู่ในรูปของภาษา XML ขึ้นมาให้

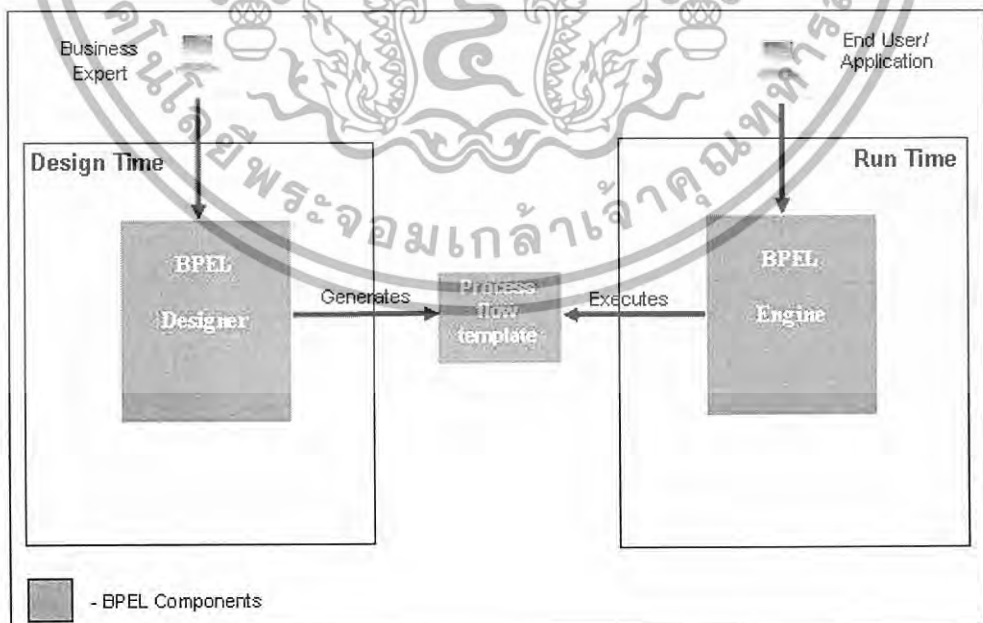
2. BPEL file

BPEL เป็นภาษาที่ไว้ใช้กำหนด business process ซึ่งจริงๆ แล้วเป็นภาษา XML ลักษณะของ BPEL คือ เป็น procedural language คล้ายกับ flow chart ทำหน้าที่กำหนดว่าจะเรียกเว็บเซอร์วิสไหน, เมื่อไหร่ และอาจจะมีการเก็บตัวแปรด้วย การทำงานจะไปข้างหน้าเรื่อยๆ จนจบไฟล์

3. BPEL engine

ตัวที่จะมาอ่าน BPEL ที่เราเขียน และสร้าง composite Web Service ให้ทำงานตามที่กำหนดใน BPEL นั้นเอง โดย client ที่เรียกใช้จะเห็น composite Web Service ที่สร้างขึ้นเป็นเหมือนเว็บเซอร์วิสทั่วไป

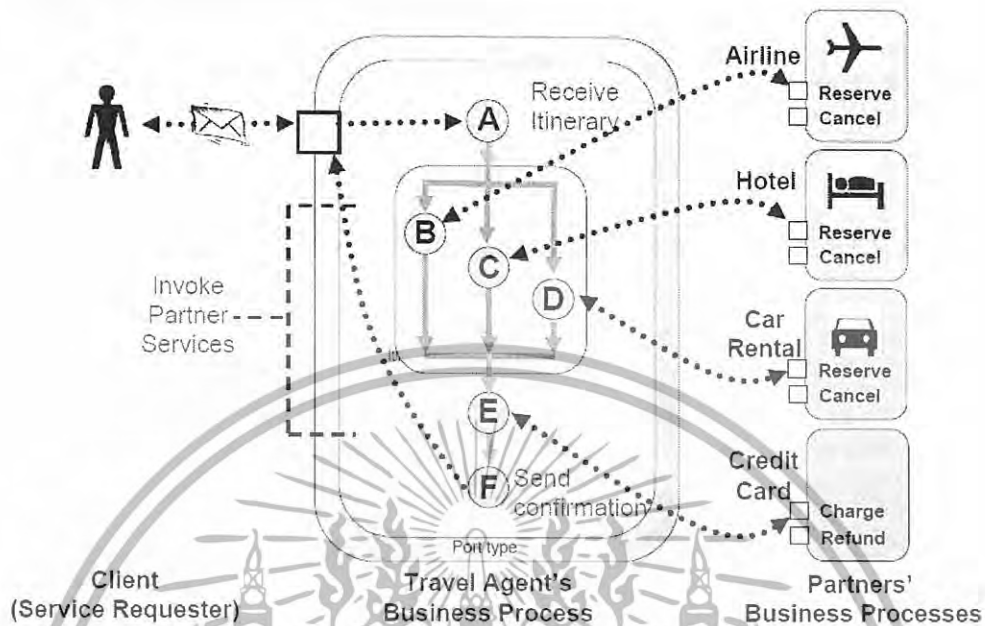
BPEL engine ปัจจุบันที่มีให้ใช้ เช่น BPEL manager ของ oracle ซึ่ง vendor ส่วนใหญ่จะมี BPEL designer ติดมาด้วย เพื่อสร้างและแปลง flow chart ให้เป็น BPEL file นอกจากนี้ BPEL ยังสนับสนุนการทำงานแบบ concurrent, asynchronous และ exception handling ด้วย



รูปที่ 3.2 BPEL Components

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ตัวอย่าง Business process



รูปที่ 3.3 ตัวอย่าง Business Process

จากตัวอย่างด้านบนเป็น Business Process ของ Travel Agent โดย Process ของเราจะถูกมองเหมือนเซอร์วิสหนึ่งซึ่งภายในจะมีการทำตาม activity ต่างๆที่ได้ออกแบบไว้ โดย process ที่ออกแบบไว้นั้นสามารถที่จะไปเรียกใช้เซอร์วิสอื่นๆได้ จากตัวอย่างภาพด้านบนสามารถอธิบายได้ดังนี้

1. Node A จะเป็นตัว receive ซึ่งจะทำหน้าที่ในการรับ message ที่มาจากผู้เรียกใช้(client)
2. Node B, Node C และ Node D จะเป็นตัว invoke ซึ่งทำงานแบบขนาน โดย Node B จะไปเรียกเซอร์วิสของการจองตั๋วเครื่องบิน, Node C จะไปเรียกเซอร์วิสของการจองโรงแรม และ Node D จะไปเรียกของการจองรถ
3. เมื่อทำทั้งสาม activity เสร็จแล้ว จะมาทำ activity ที่ Node E
4. Node E จะไปทำการเรียกเซอร์วิสของ Credit Card เมื่อคำนวณข้อมูลทางการเงิน แล้วส่งผลไปยัง Node F
5. Node F จะเป็นตัว Reply ซึ่งจะส่ง message ของผลที่ได้จาก process กลับไปให้กับ Client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 BPEL Language Structure

```

<process>
  <!-- Definition and roles of process participants -->
  <partnerLinks> ... </partnerLinks>
  <!-- Data/state used within the process -->
  <variables> ... </variables>
  <!-- Properties that enable asynchronous interaction -->
  <correlationSets> ... </correlationSets>
  <!-- Exception handling -->
  <faultHandlers> ... </faultHandlers>
  <!-- Error recovery - undoing actions -->
  <compensationHandlers> ... </compensationHandlers>
  <!-- Concurrent events with process itself -->
  <eventHandlers> ... </eventHandlers>
  <!-- Business process flow -->
  (activities)*
</process>

```

รูปที่ 3.4 BPEL Language Structure

- <process> เป็นส่วนนอกสุดที่ห่อหุ้ม BPEL element
- <partnerLinks> กำหนด partner และบทบาท (role) ของ partner นั้น
- <variables> ข้อมูลตัวแปรที่จะถูกใช้ใน process
- <correlationSets> (optional) เป็น set ของ property ที่ message ใช้
- <faultHandlers> (optional) จัดการกับ exception ที่เกิดขึ้น
- <compensationHandlers> (optional) แก้ไข error ที่เกิดขึ้น
- ส่วน business process flow ซึ่งภายในจะเป็น step การทำงานของ process ซึ่งจะถูกระบุอธิบายโดยใช้ activity

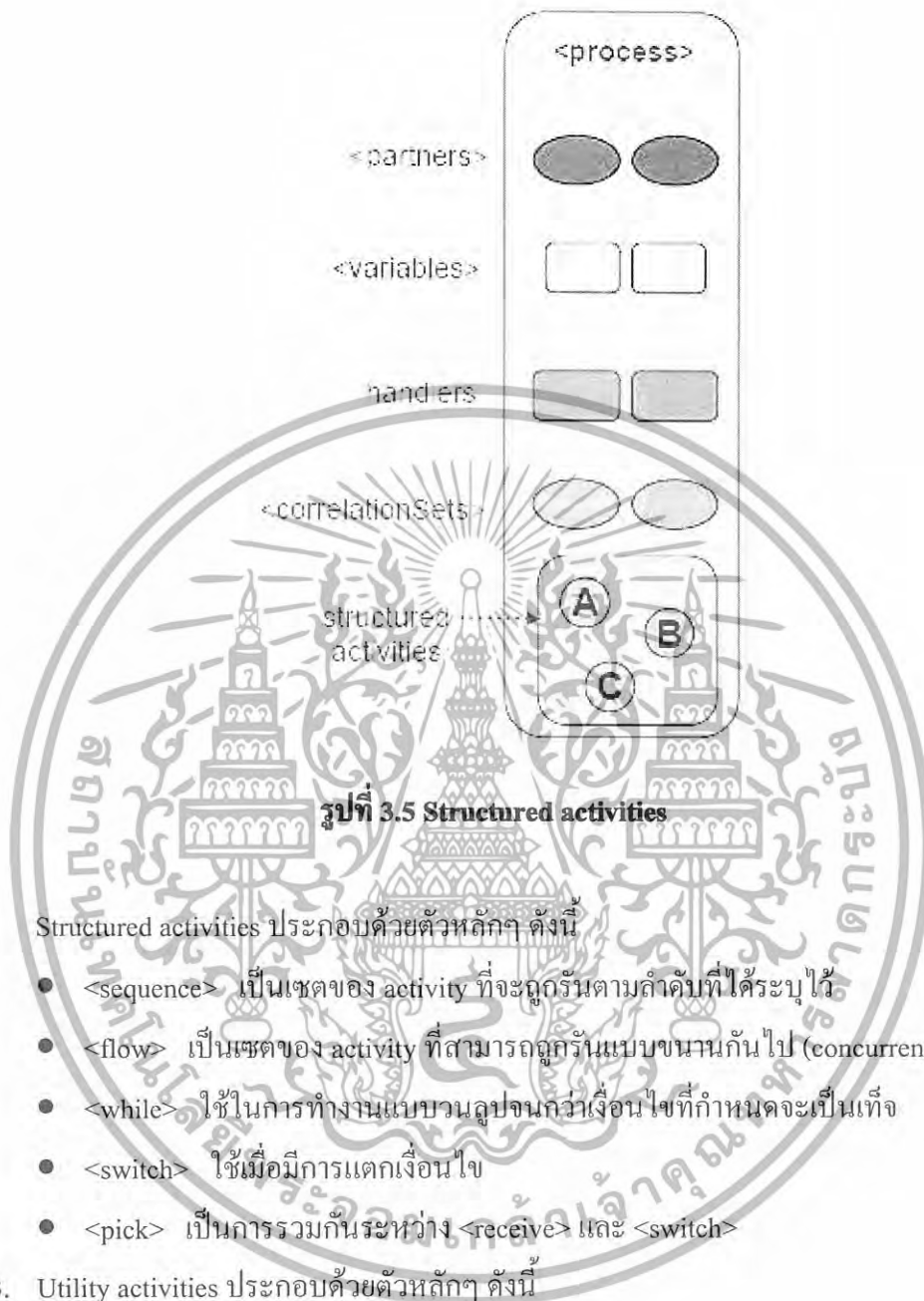
3.4 BPEL activity

Activity ใน BPEL แบ่งได้เป็น 3 ส่วนดังนี้

1. Service activities ประกอบด้วยตัวหลักๆ ดังนี้
 - <receive> ทำหน้าที่ในการรับ message ที่ส่งมาจาก client
 - <reply> ทำหน้าที่ในการส่ง response message กลับไปยัง client
 - <invoke> ทำหน้าที่ในการเรียกใช้เซอร์วิสอื่นและรอ response ที่ส่งกลับมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Structured activities เป็น activity ที่สามารถมี activity อื่นๆบรรจุอยู่ภายในได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

WS-Security

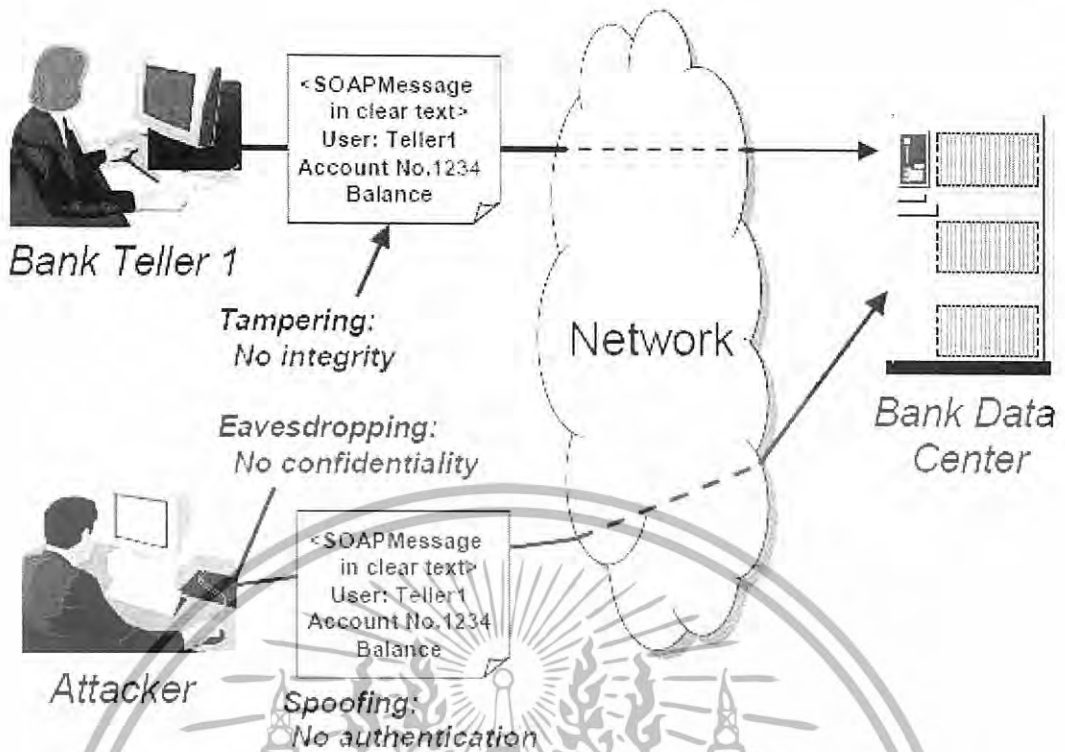
4.1 นิยามของความมั่นคงปลอดภัยในคอมพิวเตอร์

ในปัจจุบันระบบคอมพิวเตอร์ได้ถูกคุกคามมากขึ้นทั้งจากไวรัสคอมพิวเตอร์หรือจากผู้ไม่ตีซึ่งความมั่นคงปลอดภัยคอมพิวเตอร์ (Computer Security) ช่วยปกป้องเครื่องคอมพิวเตอร์รวมไปถึงอุปกรณ์ต่างๆที่เกี่ยวข้อง และที่สำคัญยังสามารถช่วยปกป้องข้อมูลที่ได้จัดเก็บไว้ภายในระบบหรือใช้ในความปลอดภัยทางข้อมูลสารสนเทศ (Information Security) ก็ได้

จุดประสงค์หลักของความมั่นคงปลอดภัยทางข้อมูลคือ

- **การรักษาความลับ (Confidentiality)** คือการรับรองว่าจะมีการเก็บข้อมูลไว้เป็นความลับ และผู้มีสิทธิเท่านั้นจึงจะเข้าถึงข้อมูลนั้นได้
- **การรักษาความสมบูรณ์ (Integrity)** คือการรับรองว่าข้อมูลจะไม่ถูกเปลี่ยนแปลงหรือทำลายไม่ว่าจะเป็นโดย อุบัติเหตุหรือโดยเจตนา
- **ความพร้อมใช้ (Availability)** คือการรับรองว่าข้อมูลและบริการการสื่อสารต่างๆ พร้อมทั้งจะใช้ได้ในเวลาที่ต้องการใช้งาน
- **การห้ามปฏิเสธความรับผิดชอบ (Non-Repudiation)** คือวิธีการสื่อสารซึ่งผู้ส่งข้อมูลได้รับหลักฐานว่า ได้มีการส่งข้อมูลแล้วและผู้รับก็ได้รับการยืนยันว่าผู้ส่งเป็นใคร ดังนั้นทั้งผู้ส่งและผู้รับจะไม่สามารถปฏิเสธได้ว่าไม่มีความเกี่ยวข้องกับข้อมูลดังกล่าวในภายหลัง
- **การพิสูจน์ตัวตน (Authentication)** คือขั้นตอนการยืนยันความถูกต้องของหลักฐาน (identity) ที่แสดงว่าเป็นบุคคลที่กล่าวอ้างจริง
- **การกำหนดสิทธิ์ (Authorization)** คือขั้นตอนในการอนุญาตให้แต่ละบุคคลสามารถเข้าถึงข้อมูลหรือระบบใดได้บ้าง ก่อนอื่นต้องทราบก่อนว่าบุคคลที่กล่าวอ้างนั้นคือใครตามขั้นตอนการพิสูจน์ตัวตนและต้องให้แน่ใจด้วยการพิสูจน์ตัวตนนั้นถูกต้อง
- **การตรวจสอบ (Audit)** คือการตรวจสอบหลักฐานทางอิเล็กทรอนิกส์ ซึ่งสามารถใช้ในการติดตามการดำเนินการเพื่อตรวจสอบความถูกต้องและแม่นยำ ตัวอย่างเช่นการตรวจสอบบัญชีข้อผู้ใช้ โดยผู้ตรวจบัญชี ซึ่งการตรวจสอบความถูกต้องของการดำเนินการเพื่อให้แน่ใจว่าหลักฐานทางอิเล็กทรอนิกส์นั้น ได้ถูกสร้างและสั่งให้ทำงาน โดยบุคคลที่ได้รับอนุญาต และในการเชื่อมต่อเหตุการณ์เข้ากับบุคคลจะต้องทำการตรวจสอบหลักฐานของบุคคลนั้นด้วย ซึ่งถือเป็นหลักการพื้นฐานของขั้นตอนการทำงานของการพิสูจน์ตัวตนด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 การส่ง SOAP Message ที่ไม่มีการรักษาความปลอดภัย

จากภาพเป็นการจำลองเหตุการณ์ที่พนักงานแบงก์ (bank teller) ซึ่งเปรียบได้กับ web service client จะส่งข้อมูลไปยังแบงก์สำนักงานใหญ่ (bank data center) ซึ่งเปรียบได้กับ web service provider โดยข้อมูลจะอยู่ในรูปของ SOAP message ที่ไม่มีการรักษาความปลอดภัยใดๆ จะเห็นได้ว่าจากสถานการณ์นี้จะเกิดความเสียหายได้ 3 ประการ คือ

- **Spoofing : No authentication** ในกรณีนี้ attacker สามารถที่เปลี่ยนแปลงแก้ไข SOAP message และส่ง message ดังกล่าวไปยัง web service provider ได้ เสมือนตัวเองเป็นพนักงานแบงก์ (bank teller) โดย attacker อาจจะแอบอ้างว่าตนเองเป็นพนักงานแบงก์ (bank teller) แล้วทำการส่ง message เช่น message สำหรับการถอนเงินจากบัญชีของลูกค้ารายอื่น ไปยังแบงก์สำนักงานใหญ่ (bank data center) ได้

วิธีการแก้ไข สามารถใช้การพิสูจน์ตัวตน (Authentication) ในการแก้ปัญหาดังกล่าว เพื่อใช้ในการตรวจว่า message ที่ส่งมานั้นมาจากบุคคลนั้นจริงๆหรือไม่

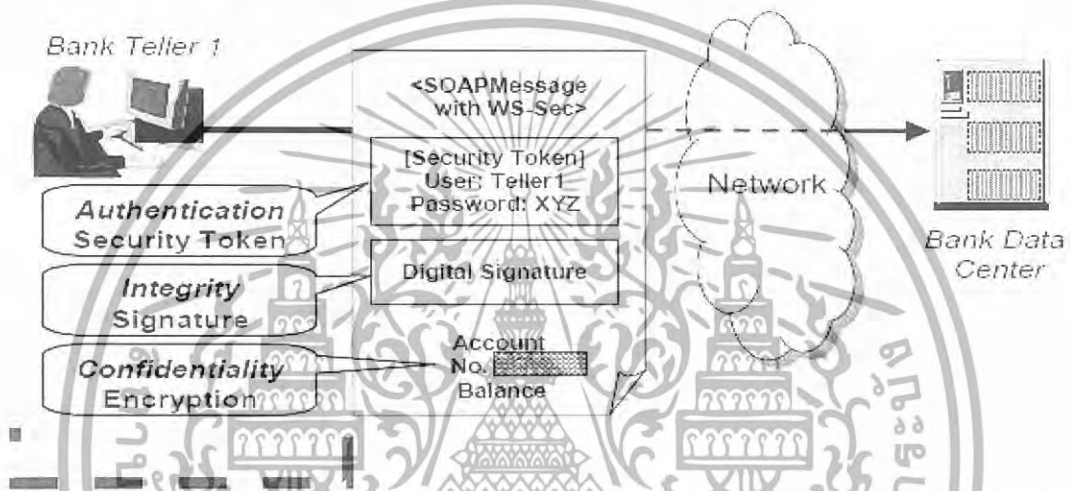
- **Eavesdropping : No Confidentiality** ในกรณีนี้ attacker สามารถดักจับ SOAP message และอ่านข้อมูลที่อยู่ใน message นั้นได้ เพราะ message ไม่ได้ทำการเข้ารหัสเอาไว้ ดังนั้นข้อมูลต่างๆที่สำคัญของลูกค้าหรือข้อมูลของแบงก์อาจจะไปอยู่ในมือของบุคคลอื่นได้ เช่น หมายเลขของบัญชี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการแก้ไข สามารถใช้การรักษาความลับ (Confidentiality) ในการแก้ปัญหาดังกล่าว เพื่อใช้ในการเข้ารหัสข้อมูลที่ส่งไป

- Tampering : No Integrity ในกรณีนี้ attacker สามารถดักจับ SOAP message และทำการเปลี่ยนแปลงข้อมูลใน message เช่น การฝากเงิน ไปยังบัญชีอื่น โดยการเปลี่ยนแปลงหมายเลขบัญชี ซึ่งที่ web service server เองก็ไม่มีตรวจสอบว่า message ที่ส่งมาถูกเปลี่ยนแปลงไปหรือไม่

วิธีการแก้ไข สามารถใช้การรักษาความสมบูรณ์ (Integrity) ในการแก้ปัญหาดังกล่าว เพื่อทำการตรวจสอบ message ที่ส่งมาว่าถูกทำการเปลี่ยนแปลงหรือไม่



รูปที่ 4.2 การส่ง SOAP Message ที่มีการรักษาความปลอดภัย

WS-Security ออกแบบมาเพื่อเพิ่มความปลอดภัยของข้อมูลในระดับ Message ในการส่งข้อมูลระหว่าง เว็บเซอร์วิส แบบ End-to-End นั้นหมายความว่า message จะมีความปลอดภัยถึงแม้จะมีการส่งผ่านหลายๆเซิร์ฟวิส (ผ่าน intermediaries) แสดงดังรูป และเพิ่มประสิทธิภาพในการส่ง SOAP Message ให้มีความน่าเชื่อถือ มีความถูกต้อง และ มีความปลอดภัย (integrity, confidentiality, authentication) ทำให้ข้อมูลมีความปลอดภัยตลอดเวลาไม่ว่าจะถูกส่งไปให้ Service ไหนก็ตาม

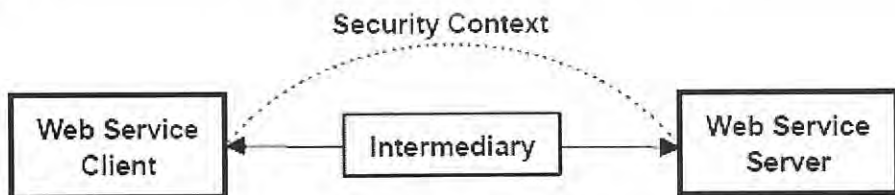
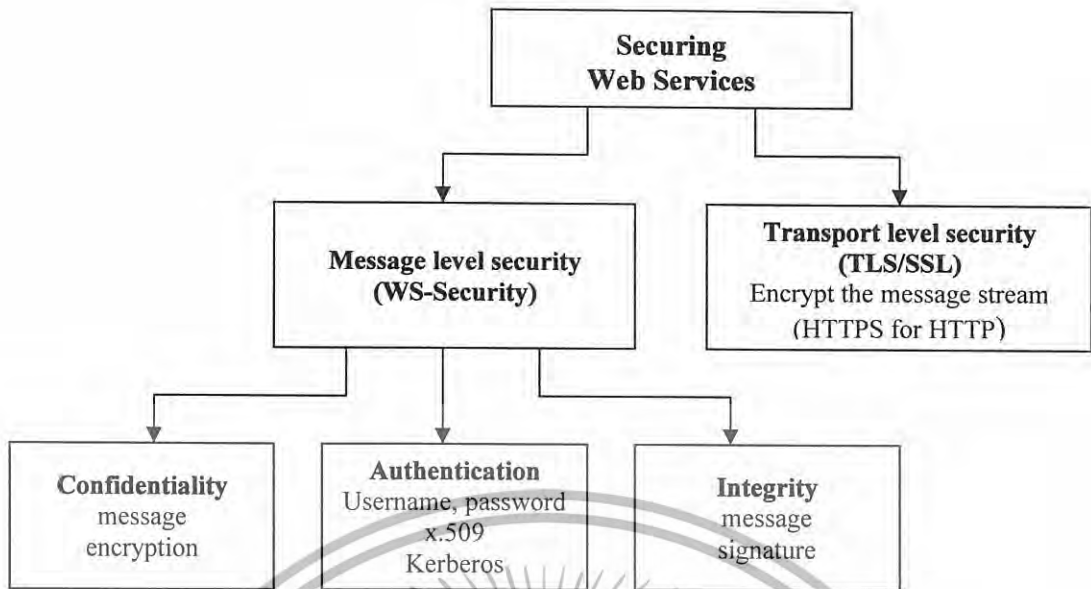


Figure 8-3 End-to-end security with message-level security

รูปที่ 4.3 การรักษาความปลอดภัยข้อมูลแบบ End-to-End โดยใช้ WS-security

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



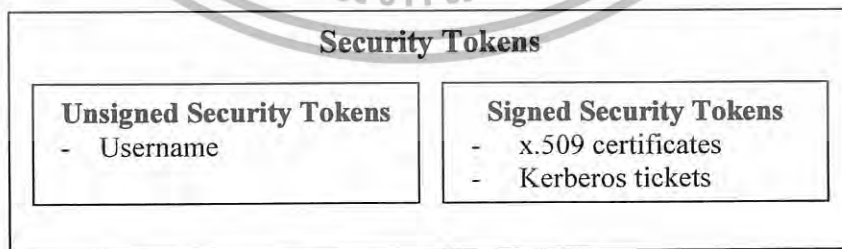
รูปที่ 4.4 โมเดล Securing Web Services

4.2 การพิสูจน์ตัวตน (Authentication)

WS-Security จะใช้ token ในการพิสูจน์ตัวตน เช่น การใช้ username token โดยจะมีการส่ง username และ password ไปพร้อมกับ SOAP Message เมื่อเว็บเซิร์ฟเวอร์ได้รับ username, password จะทำการตรวจสอบ ถ้าข้อมูลที่ได้ออกคือ message จะถูกนำไปประมวลผลต่อไป

Security Token ถูกเป็นออกเป็น 2 ประเภท คือ

1. Unsigned Security Token เป็น Token ที่ไม่มีผู้รับรอง เป็นเพียงการแสดงสิทธิ์ของตนเอง ไม่มีการรับรองแต่อย่างใด เช่น Username Token
2. Signed Security Token (Binary Security Token) เป็น Token ที่ได้รับการรับรองจากผู้ที่มีความน่าเชื่อถือ เช่น Kerberos Ticket หรือ x.509 Certificate



รูปที่ 4.5 ประเภทของ Token

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 หลักการทำงานของ Username Token

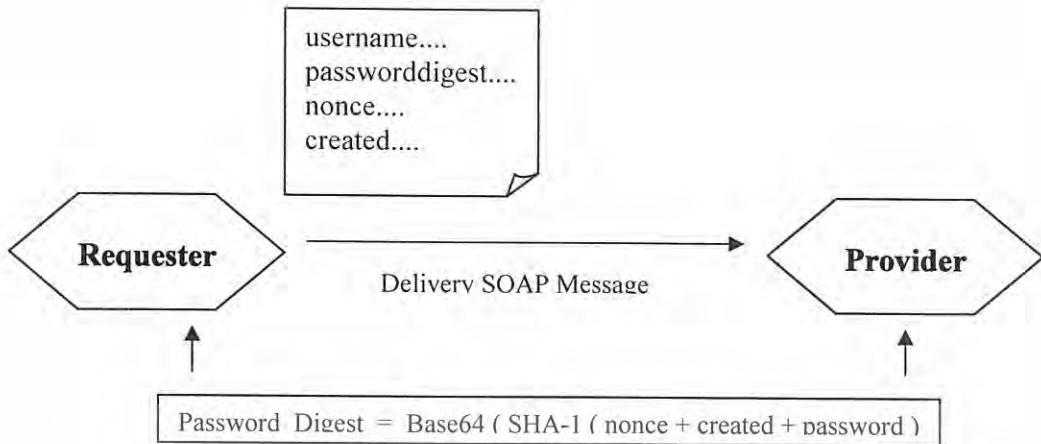
Username Token สามารถส่งได้ทั้งแบบระบุรหัสผ่านโดยตรง (clear text) และแบบ digest โดยใช้ nonce โดยการส่งในแบบแรก SOAP Message ที่ส่งไปจะไม่ปลอดภัย เพราะแฮกเกอร์สามารถเข้าดักจับ (trap) password ได้ ยกเว้นว่า SOAP Message นั้นจะถูกส่งอยู่ในช่องทางที่ปลอดภัยอยู่แล้วเช่น SSL สำหรับวิธีแบบ digest โดยใช้ nonce เป็นวิธีการป้องกันการโจมตีแบบสแนก โดยผู้ส่งเป็นผู้สร้าง nonce ซึ่งเป็นค่าแบบสุ่มขึ้นมา โดยวิธีการนี้ทำให้ต้องมีเซิร์ฟเวอร์ที่มีแคช (cache) เพียงพอต่อการเก็บค่า nonce ที่ถูกใช้แล้วเอาไว้ทำให้สิ้นเปลืองรีซอร์ส (resource) จึงมีการใช้ควบคู่ไปกับ timestamp (created) วิธีนี้มีข้อดีคือ สามารถกำหนดขีดจำกัดของ nonce ให้อยู่ในช่วงเวลาที่ใหม่ (freshness) เท่านั้น ถ้ามีการใช้ nonce เกิดขึ้นจะมีการเพิ่มค่าของพาสเวิร์ดไดเจสต์ (Password_Digest value) เข้าไปใน SOAP Message ด้วย โดยค่าพาสเวิร์ดไดเจสต์ทำได้โดย นำค่า nonce, created และ password มาทำการ concatenate แล้วทำการแฮชด้วยอัลกอริทึม SHA-1 หลังจากนั้นทำการ encode ด้วย BASE64 ดังสมการนี้

$$\text{Password_Digest} = \text{Base64} (\text{SHA-1} (\text{nonce} + \text{created} + \text{password}))$$

ขั้นตอนการส่ง SOAP Message ที่มีการใช้ nonce และ created

1. Requester สร้าง nonce ขึ้นมาพร้อมทั้งระยะเวลา timestamp (created) แล้วนำค่า password, nonce และ created มา concatenate แล้วทำการแฮชด้วย อัลกอริทึม SHA-1 หลังจากนั้นทำการ encode ด้วย BASE64 เพื่อหาค่าของ PasswordDigest แล้วนำค่าที่ได้ใส่ลงใน SOAP Message ที่จะส่งด้วย
2. เมื่อฝั่ง Provider ได้รับ SOAP Message จะตรวจสอบความถูกต้องของค่า PasswordDigest ที่ส่งมา โดยนำค่า password ที่เก็บไว้มา concatenate กับค่า nonce และ created ที่ส่งมากับ SOAP Message แล้วทำการแฮชด้วยอัลกอริทึม SHA-1 หลังจากนั้นทำการ encode ด้วย BASE64 ถ้าค่าที่ได้ตรงกันกับค่า PasswordDigest ที่ส่งมา message จะถูกนำไปประมวลผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 การส่ง SOAP Message ที่มีการใช้ nonce และ created

เมื่อมีการส่ง username token จากระบบหนึ่งไปอีกระบบหนึ่งโดยไม่มีการรักษาความปลอดภัยในการขนส่งข้อมูลเช่น HTTP อาจทำให้ token ถูกดักจับได้ทำให้เป็นจุดอ่อนในการโจมตีระบบได้ เว้นแต่จะใช้ XML digital signature หรือ XML Encryption เข้ามาช่วย

4.2.2 อีลีเมนต์ที่ใช้ในการส่งที่เกี่ยวข้องวิธีโทเคนแบบระบุรหัสผ่านโดยใช้ nonce

- <wsse:UsernameToken> แสดงถึงการใช้ Username Token
- <wsse:Username> แสดงถึงยูสเซอร์เนมที่ใช้
- <wsse:Password> แสดงถึงรหัสผ่านของยูสเซอร์
- แอททริบิวต์ PasswordType ระบุว่าใช้รหัสผ่านเป็นแบบใดเจสต์
- <wsse:Nonce> เป็นค่าที่ไว้ป้องกันการโจมตีแบบเสก (replay attack)
- <wsu:Created> เป็นค่าที่บอกว่าเมสเสจนี้ถูกสร้างขึ้นเมื่อไร (timestamp)

```
<S:Envelope xmlns:S="http://www.w3.org/2001/12/soap-envelope"
  xmlns:wsse="http://schemas.xmlsoap.org/ws/2003/06/secext">
  <S:Header>
    ...
    <wsse:Security>
      <wsse:UsernameToken
        xmlns:wsse="http://schemas.xmlsoap.org/ws/2003/06/secext"
        xmlns:wsu="http://schemas.xmlsoap.org/ws/2003/06/utility">
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<wsse:Username>NNK</wsse:Username>
<wsse:Password Type="wsse:PasswordDigest">
  weYI3nXd8LjMNVksCKFV8t3rgHh3Rw==
</wsse:Password>
<wsse:Nonce>WScqanjCEAC4mQoBE07sAQ==</wsse:Nonce>
<wsu:Created>2003-07-16T01:24:32Z</wsu:Created>
</wsse:UsernameToken>
</wsse:Security>
...
</S:Header>
...
</S:Envelope>

```

รูปที่ 4.7 ตัวอย่างการใช้ Username Token แบบใช้ nonce

4.3 การรักษาความลับของข้อมูล (Confidentiality)

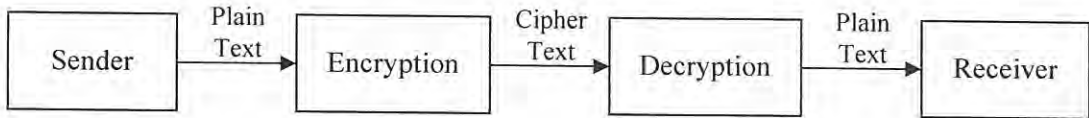
การรักษาความลับของข้อมูล คือ การรับรองว่าจะมีการเก็บข้อมูลไว้เป็นความลับ และผู้มีสิทธิเท่านั้นจึงจะเข้าถึงข้อมูลนั้นได้  สามารถจัดการเกี่ยวกับการเข้ารหัสข้อมูลบางส่วน หรือทั้งหมดของ SOAP message ก่อนที่จะทำการส่งต่อไป ซึ่งมาตรฐานหนึ่งที่ถูกนำมาใช้คือ XML Encryption ซึ่งมาจาก W3C

4.3.1 หลักการทำงานของ XML Encryption

XML Encryption คล้ายกับการเข้ารหัสแบบต่างๆ ไป แต่ต่างกันตรงที่ว่า XML Encryption สามารถเลือกทำการเข้ารหัสเอกสารทั้งหมด หรือเลือกเฉพาะบางส่วนของข้อมูลในส่วนของเฮดเดอร์, บอดี หรือข้อมูลที่แนบมาของ SOAP message นั้น

การเข้ารหัส คือ กระบวนการในการเปลี่ยนข้อมูลต้นฉบับให้อยู่ในอีกรูปแบบหนึ่งที่ไม่สามารถเข้าใจได้โดยง่าย ทำให้ผลลัพธ์ที่ได้จากการเข้ารหัสจะสามารถนำกลับไปเป็นข้อมูลต้นฉบับได้นั้นต้องใช้ในการถอดรหัส (Decryption) โดยเราจะเรียกข้อมูลต้นฉบับที่จะทำการเข้ารหัสว่า เคลียร์เท็กซ์ (Clear text) หรือ เพลนเท็กซ์ (Plain text) และเรียกข้อมูลที่ทำการเข้ารหัสเรียบร้อยแล้ว ไชเฟอร์เท็กซ์ (Cipher text)

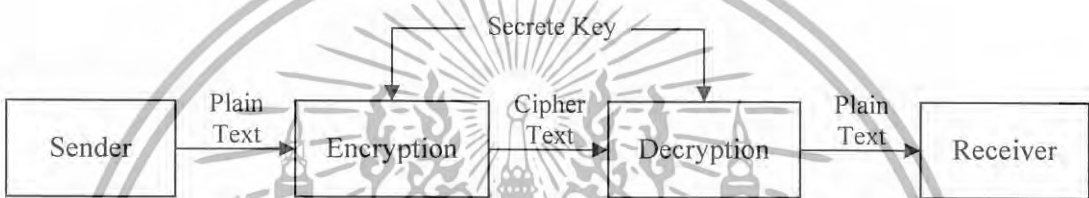
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 การเข้ารหัสและถอดรหัส

การเข้ารหัสนั้นวิธีการหนึ่งที่ได้คือการเข้ารหัสและถอดรหัส โดยใช้กุญแจ มี 2 แบบ คือ **การเข้ารหัสแบบสมมาตร (Symmetric Key Cryptography)**

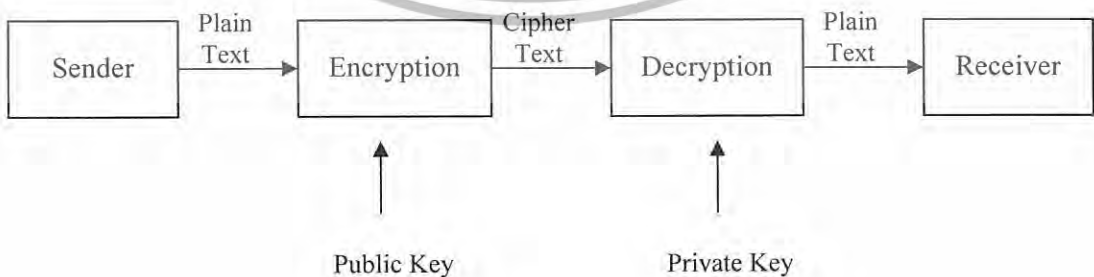
อัลกอริทึมแบบนี้จะใช้กุญแจที่เรียกว่า กุญแจลับ (Secret key) ซึ่งมีเพียงหนึ่งเดียวเพื่อใช้ในการเข้ารหัสและถอดรหัสข้อความที่ส่งไป



รูปที่ 4.9 การเข้ารหัสและถอดรหัสแบบสมมาตร

การเข้ารหัสแบบอสมมาตร (Asymmetric Key Cryptography)

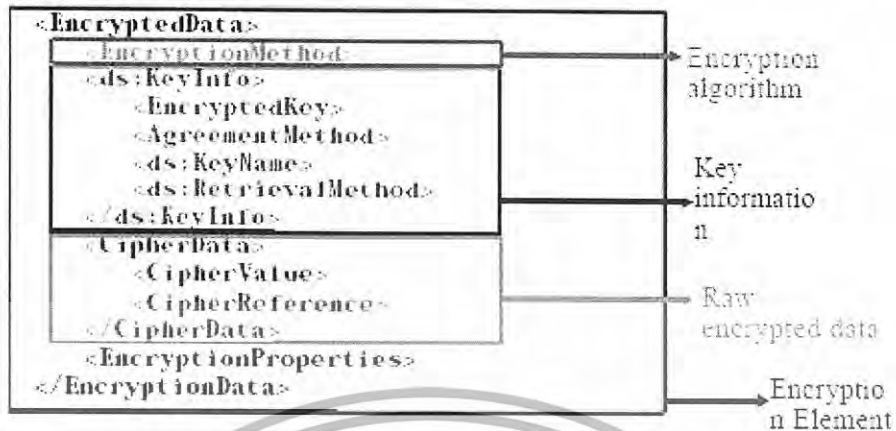
การเข้ารหัสแบบนี้จะใช้หลักกุญแจคู่ทำการเข้ารหัสและถอดรหัส ซึ่งกุญแจคู่ที่ว่านี้จะประกอบไปด้วย กุญแจส่วนตัว (private key) และกุญแจสาธารณะ (public key) โดยกุญแจสาธารณะจะใช้ในการเข้ารหัส ส่วนกุญแจส่วนตัวจะใช้ในการถอดรหัส กุญแจสาธารณะนี้สามารถส่งมอบให้กับผู้อื่นได้ เช่น เพื่อนร่วมงานที่เราต้องการติดต่อด้วย หรือแม้กระทั่งวางไว้บนเว็บไซต์เพื่อให้ผู้อื่นสามารถดาวน์โหลดไปใช้งานได้ สำหรับกุญแจส่วนตัวนั้นต้องเก็บไว้กับผู้เป็นเจ้าของกุญแจส่วนตัวเท่านั้น และห้ามเปิดเผยให้ผู้อื่นทราบโดยเด็ดขาด



รูปที่ 4.10 การเข้ารหัสแบบอสมมาตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 อีลีเมนต์ที่ใช้ใน XML Encryption



รูปที่ 4.11 อีลีเมนต์ที่ใช้ใน XML Encryption

อีลีเมนต์ของ XML ในการอธิบายมาตรฐานนี้คือ <EncryptedData>, <EncryptedKey> ในส่วนของบอดีของเมสเสจจะประกอบด้วย <EncryptedData> กับอีลีเมนต์ย่อยอีก 3 อีลีเมนต์คือ

1. อีลีเมนต์แรกคือ <EncryptionMethod> จะบอกถึงอัลกอริทึมในการเข้ารหัสข้อมูล ในกรณีนี้ อัลกอริทึมชนิด TripleDES จะถูกเลือก และกุญแจที่เหมือนกันจะถูกใช้ในการเข้ารหัสและถอดรหัส
2. อีลีเมนต์ที่สองคือ <KeyInfo> มันจะสมมติว่าทั้งสองฝั่งที่ติดต่อกันนั้นรู้กุญแจที่ใช้ในการเข้ารหัส และถอดรหัส ตัวอย่างชื่อของกุญแจ เช่น "CN=Key13, C=US" ไม่ว่าชื่อใดก็ตามที่ถูกใช้นั้น สิ่งที่สำคัญที่สุดคือทั้งสองฝั่งต้องเข้าใจถึงที่กุญแจนี้อย่างถึง
3. อีลีเมนต์สุดท้ายคือ <CipherData> เป็นอีลีเมนต์ย่อยที่จะมี <CipherValue> เป็นอีลีเมนต์ย่อยอีกทีที่จะเป็นการเข้ารหัสข้อมูลที่แท้จริง

ตัวอย่างในกรณีนี้เมื่อข้อมูลถูกเข้ารหัสโดยใช้กุญแจสมมาตร โดยใช้ TripleDES ดังรูปด้านล่าง

```

<s:Envelope
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
  <s:Body>
    <xenc:EncryptedData>
      <xenc:EncryptionMethod
        Algorithm='http://www.w3.org/2001/04/xmlenc#tripleDES-cbc'/>
      <ds:KeyInfo>
        <ds:KeyName> CN=Key13, C=US </ds:KeyName>
      </ds:KeyInfo>
      <xenc:CipherData>
        <xenc:CipherValue> r5KipsDV . . . </xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </s:Body>
</s:Envelope>

```

รูปที่ 4.12 การเข้ารหัสในส่วนบอดี้ของ SOAP MESSAGE

เมื่อคุณเจสสมมาตรถูกเข้ารหัสโดยใช้กุญแจสาธารณะแล้วทำการส่งข้อมูลต่อไป เมื่อได้รับเมสเสจแล้ว ผู้รับสามารถที่จะใช้กุญแจส่วนตัวในการถอดรหัส เพื่อดึงเอาคุณเจสสมมาตรออกมา แล้วใช้กุญแจสมมาตรนี้ในการถอดรหัสข้อมูลที่แท้จริง การส่งกุญแจที่เข้ารหัสนั้น XML Encryption จะกำหนดอ็ลิตีเมนต์ <EncryptedKey> เพื่อใช้กับ SOAP เมสเสจ ซึ่ง WS-Security จะกำหนดว่าอ็ลิตีเมนต์นี้จะอยู่ใน <Security> เฮดเคอร์ ดังนั้น SOAP เมสเสจจะมีทั้งข้อมูลที่เข้ารหัสแล้ว และกุญแจสมมาตรที่เข้ารหัสแล้ว แสดงดังรูปด้านล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<s:Envelope
  xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/12/secext"
  xmlns:wsu="http://schemas.xmlsoap.org/ws/2002/07/utility"
  xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
  xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
  <s:Header>
    <wsse:Security>
      <xenc:EncryptedKey>
        <xenc:EncryptionMethod
          Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
        <ds:KeyInfo>
          <ds:KeyName> CN=Key13, C=US </ds:KeyName>
        </ds:KeyInfo>
        <xenc:CipherData>
          <xenc:CipherValue> ir4DfG . . . </xenc:CipherValue>
        </xenc:CipherData>
        <xenc:ReferenceList>
          <xenc:DataReference URI="#EncryptedBody"/>
        </xenc:ReferenceList>
      </xenc:EncryptedKey>
    </wsse:Security>
  </s:Header>
  <s:Body>
    <xenc:EncryptedData wsu:Id="EncryptedBody">
      <xenc:EncryptionMethod
        Algorithm='http://www.w3.org/2001/04/xmlenc#tripleDES-cbc'/>
      <xenc:CipherData>
        <xenc:CipherValue> r5KipsDV . . . </xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>
  </s:Body>
</s:Envelope>

```

รูปที่ 4.13 การเข้ารหัส SOAP Message ด้วยกุญแจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

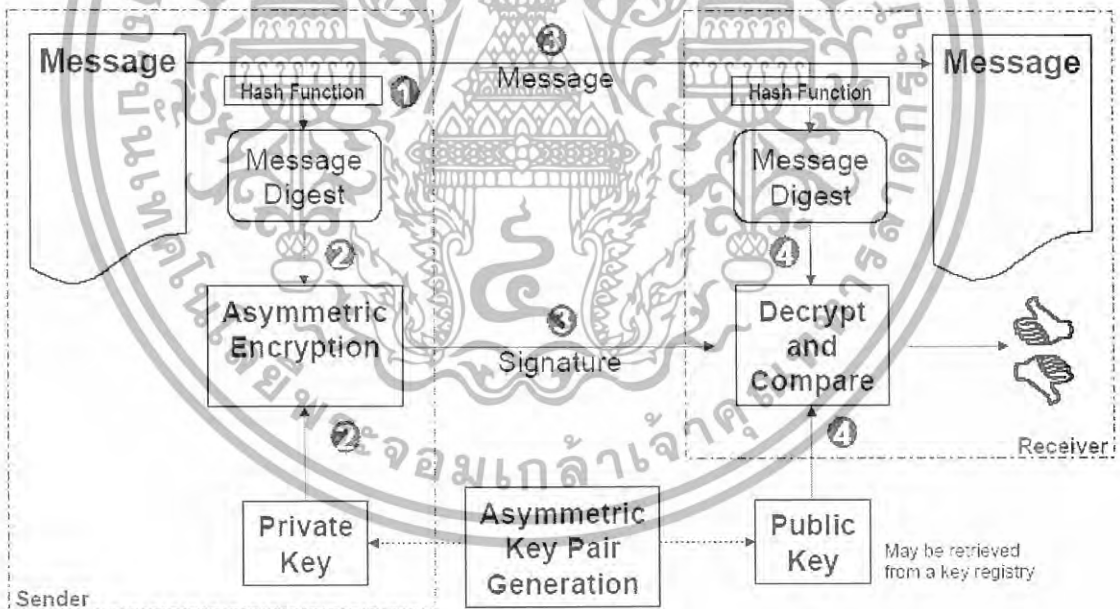
4.4 การรักษาความสมบูรณ์ของข้อมูล (Integrity)

การรักษาความสมบูรณ์ของข้อมูล คือ การรับรองว่าข้อมูลจะไม่ถูกเปลี่ยนแปลงหรือทำลายไปจากต้นฉบับ (Source) ไม่ว่าจะโดยบังเอิญหรือดัดแปลงโดยเจตนาที่อาจส่งผลเสียต่อข้อมูล การคุกคามความสมบูรณ์ของข้อมูล คือ การที่บุคคลที่ไม่ได้รับอนุญาตสามารถที่จะเข้าควบคุมการจัดการของข้อมูลได้ ใน WS-Security การรักษาความสมบูรณ์ของ SOAP Message จะทำผ่าน XML Digital Signature ซึ่งเป็นมาตรฐานของ W3C

4.4.1 หลักการทำงานของ XML Digital Signature

XML Digital Signature เป็นการระบุลายเซ็นดิจิทัลลงบน SOAP Message เพื่อยืนยันตัวตนของผู้ส่งเพื่อพิสูจน์ว่าเอกสารที่ส่งมาไม่ถูกเปลี่ยนแปลงซึ่งถ้าหากเอกสารถูกเปลี่ยนแปลงระหว่างการส่งเมื่อทำการตรวจสอบจะได้ผลไม่ตรงกัน ทำให้บอกได้ว่าเอกสารนี้ถูกส่งจากผู้ส่งที่เราคาดหวังจริงหรือไม่

Digital Signature เป็นการนำหลักการของการทำงานของระบบการเข้ารหัสแบบใช้คู่รหัสกุญแจเพื่อการพิสูจน์ตัวตนมาประยุกต์ใช้ ระบบของลายเซ็นดิจิทัลสามารถแบ่งเป็นขั้นตอนได้ดังนี้



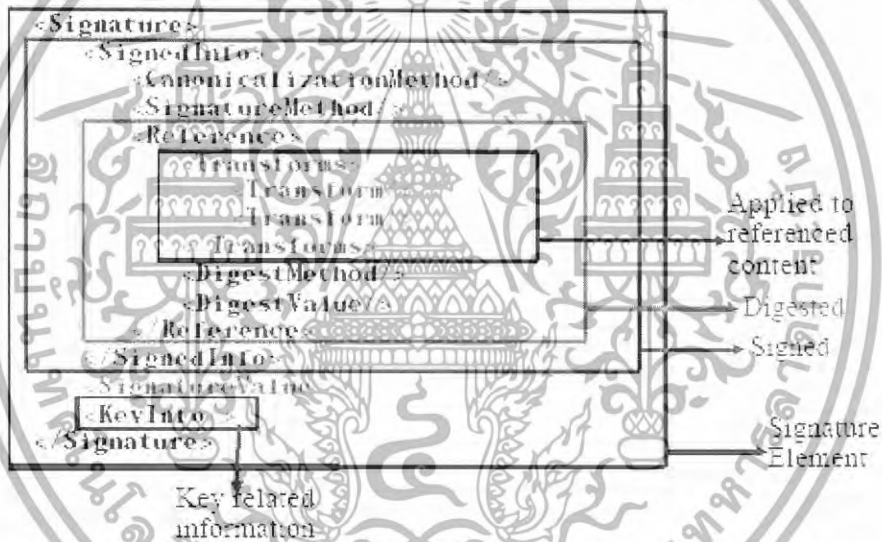
รูปที่ 4.14 ขั้นตอนการทำ Digital Signature

1. เมื่อผู้ใช้ต้องการจะส่งข้อมูลไปยังผู้รับ ข้อมูลนั้นจะถูกนำไปเข้าฟังก์ชันทางคณิตศาสตร์ที่เรียกว่า "แฮชฟังก์ชัน" ได้เป็นเมสเสจไดเจสต์ (Message Digest) ออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. นำเมสเสจไคเจสต์ที่ได้มาทำการเข้ารหัสด้วยกุญแจส่วนตัว ผลลัพธ์ที่ได้ออกมาจะอยู่ในรูป Signature
3. ส่ง Signature และ message ไปยังฝั่งผู้รับ
4. ในฝั่งผู้รับจะมีการตรวจสอบความถูกต้องของข้อมูลและตรวจสอบว่าข้อมูลถูกส่งมาจากผู้ส่งคนนั้นจริงหรือไม่ โดยการนำ message มาผ่านแฮชฟังก์ชันเพื่อคำนวณค่าเมสเสจไคเจสต์ และทำการถอดรหัสลายเซ็นอิเล็กทรอนิกส์ด้วยกุญแจสาธารณะของผู้ส่ง ถ้าสามารถถอดรหัสได้อย่างถูกต้อง จะเป็นการยืนยันข้อมูลว่ามาจากผู้ส่งคนนั้นจริง และถ้าข้อมูลเมสเสจไคเจสต์ที่ได้จากการถอดรหัสเท่ากับค่าเมสเสจไคเจสต์ที่ได้จากการเข้าแฮชฟังก์ชัน จะถือว่าข้อมูลดังกล่าวนั้นถูกต้อง

4.4.2 อีลีเมนต์ที่ใช้ใน XML Digital Signature



รูปที่ 4.15 อีลีเมนต์ที่ใช้ใน XML Digital Signature

อีลีเมนต์ที่สำคัญในการทำลายเซ็นอิเล็กทรอนิกส์คือ < Signature> เป็นอีลีเมนต์เริ่มต้นที่จะประกอบด้วยอีลีเมนต์ย่อย ที่จะใช้ในการทำลายเซ็นอิเล็กทรอนิกส์ โดยมีอีลีเมนต์ภายใน เช่น

- <SignedInfo> ประกอบด้วยรายละเอียดของอินฟอร์เมชันที่เกี่ยวข้องกับการ signed
- <SignatureValue> เป็นตัวระบุถึงของ signature ที่ได้
- <KeyInfo> เป็นตัวที่จะบอกว่าควรจะใช้กุญแจดอกไหนในการในการทำลายเซ็นอิเล็กทรอนิกส์ภายในจะมีอีลีเมนต์ <SecurityTokenReference> ที่ใช้ในการอ้างอิงถึง security token ที่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการเชิงงานเพื่อการศึกษาเท่านั้น ไม่นิยามให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<?xml version="1.0" encoding="utf-8"?>
  <S11:Envelope xmlns:S11="..." xmlns:wse="..." xmlns:wsu="..."
  xmlns:Ss="...">
    <S11:Header>
      <wse:Security>
        <wse:BinarySecurityToken
          ValueType="...#X509v3"
          EncodingType="...#Base64Binary"
          wsu:Id="X509Token">
          MIIBDzCCAg9CgAwIBAgIQEmsJZ61gqrKk51...
        </wse:BinarySecurityToken>
        <ds:Signature>
          <ds:SignedInfo>
            <ds:CanonicalizationMethod Algorithm=
              "http://www.w3.org/2001/10/xml-exc-c14n#" />
            <ds:SignatureMethod Algorithm=
              "http://www.w3.org/2001/09/xmldsig#rsa-sha1" />
            <ds:Reference URI="#myBody">
              <ds:Transforms>
                <ds:Transform Algorithm=
                  "http://www.w3.org/2001/10/xml-exc-c14n#" />
                <ds:Transform
                  Algorithm="http://www.w3.org/2001/09/xmldsig#sha1" />
                <ds:DigestMethod Algorithm=
                  "http://www.w3.org/2001/09/xmldsig#sha1" />
                <ds:DigestValue EVMkDy5S1...>
              </ds:Reference>
            <ds:SignatureValue
              EEPj55T:EbLd:VXcNMZMj:FOV...>
            <ds:KeyInfo>
              <wse:SecurityTokenReference>
                <wse:Reference URI="#X509Token">
                </wse:Reference>
              </wse:SecurityTokenReference>
            </ds:KeyInfo>
          </ds:Signature>
        </wse:Security>
      </S11:Header>
      <S11:Body wsu:Id="#myBody">
        <xml:StockSymbol xmlns:xml="http://www.fabrikam13.com/payloads">
          000
        </xml:StockSymbol>
      </S11:Body>
    </S11:Envelope>
  
```

รูปที่ 4.16 การสร้างลายเซ็นอิเล็กทรอนิกส์


4.5 ใบรับรองอิเล็กทรอนิกส์ (Certification)

ในโลกของอิเล็กทรอนิกส์ เราสามารถทำการตรวจสอบสถานะของบุคคลที่ประสงค์จะติดต่อได้จากใบรับรองอิเล็กทรอนิกส์ (Certificate) ซึ่งใช้สำหรับการยืนยันตัวตนบุคคล ใบรับรองฯ ดังกล่าวเป็นข้อมูลในรูปแบบอิเล็กทรอนิกส์ที่ออกให้โดยผู้ให้บริการออกใบรับรอง (Certification Authority - CA) ซึ่งต้องเป็นหน่วยงานหรือองค์กรที่เชื่อถือได้และมีความรู้ความเชี่ยวชาญในเทคโนโลยีความปลอดภัย โดยอาศัยเทคโนโลยีหลัก เรียกว่า เทคโนโลยีโครงสร้างพื้นฐานระบบ

กุญแจสาธารณะ (Public Key Infrastructure - PKI) ซึ่งมีระบบกุญแจคู่ อันประกอบด้วยกุญแจเอกสารเป็นเอกสารที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนตัว (Private Key) และกุญแจสาธารณะ (Public Key) เป็นพื้นฐานสำคัญ โดยที่ผู้ให้บริการออกใบรับรองจะทำการรับรองข้อมูลต่างๆ รวมทั้งกุญแจสาธารณะของบุคคลนั้น ซึ่งกุญแจสาธารณะดังกล่าว จะใช้ในการเข้ารหัสข้อมูล (Encryption) และตรวจสอบลายมือชื่อดิจิทัล (Digital Signature) เพื่อเป็นการรักษาความลับ ตรวจสอบความถูกต้อง และที่มาของข้อมูลว่าบุคคลใดเป็นผู้ทำการส่งข้อมูลหรือลงลายมือชื่อดิจิทัล โดยที่ผู้ใช้งานควรที่จะมีความรู้ความเข้าใจเกี่ยวกับกระบวนการในการใช้งานใบรับรองอิเล็กทรอนิกส์ และข้อมูลที่ปรากฏอยู่ในใบรับรองอิเล็กทรอนิกส์ เพื่อก่อให้เกิดประสิทธิภาพในการใช้งาน รวมทั้งยังเป็นประโยชน์แก่ผู้ใช้งานอีกด้วย

รูปแบบของใบรับรองอิเล็กทรอนิกส์ อิงตามมาตรฐาน X.509 Certificate ซึ่งเป็นมาตรฐานของใบรับรองอิเล็กทรอนิกส์ที่กำหนดโดย ITU-T X.509 International Standard ที่บอกถึงรูปแบบของใบรับรองๆ โดยประกอบไปด้วยข้อมูลหลักๆ 2 ส่วน คือ ข้อมูลพื้นฐานของใบรับรองอิเล็กทรอนิกส์ (Basic Certificate Fields) และข้อมูลเพิ่มเติมของใบรับรองอิเล็กทรอนิกส์ (Certificate Extension) ดังรูป

Certificate format version		
Certificate serial number		
Signature algorithm identifier for CA		
Issuer X.509 name		
Validity period		
Subject X.509 Name		
Subject public key information		
Type	Criticality	Value
Type	Criticality	Value 
CA Signature		

รูปที่ 4.17 รูปแบบของใบรับรองอิเล็กทรอนิกส์

ในการขอเอกสารสิทธิ์นั้นจะต้องกระทำการบนกระบวนการที่ปลอดภัย ซึ่งปัจจุบันมี 2 วิธีคือ

1. การขอเอกสารสิทธิ์ผ่านโปรแกรมประเภทบราวเซอร์ เช่น อินเทอร์เน็ตเอ็กซ์พลอเรอร์

เน็ตสเคป เป็นต้น โดยผู้ใช้เข้าไปยังโฮสต์ที่ให้บริการการขอเอกสารสิทธิ์หรือเว็บที่เปิดเอกสารนี้เป็นเอกสารที่ส่งงานไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้บริการดังกล่าวอยู่ ตัวโปรแกรมประเภทบราวเซอร์จะทำการสร้างกุญแจคู่ กุญแจส่วนตัวจะเก็บไว้ที่เครื่องของเรา จากนั้นจะส่งกุญแจสาธารณะและข้อมูลส่วนตัวของเราไปยังโฮสต์หรือไชต์ที่เป็นขององค์กรพิสูจน์ จากนั้นองค์กรพิสูจน์จะทำการรับรองข้อมูลแล้วจึงส่งเอกสารสิทธิ์กลับมาที่เครื่องเรา

2. การขอเอกสารสิทธิ์อีกรูปแบบหนึ่งคือการส่งเอกสารสิทธิ์ที่ไม่ผ่านระบบเครือข่าย กล่าวคือผู้ขอต้องไปทำการขอที่สำนักงานขององค์กรพิสูจน์สิทธิ์โดยตรง โดยทำเรื่องขอเอกสารสิทธิ์และกรอกข้อมูลส่วนตัว องค์กรพิสูจน์สิทธิ์จะทำการรับรองและบันทึกข้อมูลของผู้ขอแล้วทำการส่งเอกสารสิทธิ์พร้อมทั้งกุญแจส่วนตัวให้กับผู้ขอในรูปแบบของไฟล์แผ่นดิสก์ให้ผู้ขอนำไปบันทึกในเครื่องของตนต่อไป

4.5.1 ขั้นตอนการขอใบรับรองของเว็บ <http://www.verisign.com> (CA International)

1. สร้าง CSR(Certificate Signing Request) ขึ้นมา
2. ขั้นตอนการพิจารณาข้อมูลของผู้ที่จะทำการขอ Certificate ซึ่งใช้เวลาตั้งแต่ 1-30 วัน ในการ พิจารณาขึ้นอยู่กับชนิดของ Certificate และข้อมูล เอกสารที่ผู้ร้องขอส่งไปให้ทาง CA
3. จะได้รับ Certificate เป็น Text file จากทาง email

4.5.2 ขั้นตอนการขอใบรับรองของเว็บ <http://ca.khonthai.com> (CA Thai)

1. ขอรหัส PIN ที่ website www.dopa.go.th หรือ www.khonthai.com ซึ่งเป็น website ของกรมการปกครอง
 ** ในกรณีที่ได้ลงทะเบียนขอรหัส PIN แล้วมีเพียงรหัส PIN1 ให้ใส่รหัส PIN2 เป็นรหัสเดียวกับ PIN1 แล้วสามารถเข้าไปเปลี่ยนรหัส PIN2 ได้ใน website ของกรมการปกครอง ***
2. ลงทะเบียนที่หน้าลงทะเบียนของ website ของ Khonthai CA
3. ใส่ PID (หมายเลขบัตรประชาชน 13 หลัก) และใส่รหัส PIN1, PIN2 ให้ถูกต้อง
4. ถ้า PID และรหัส PIN ถูกต้องจะปรากฏข้อมูลของคุณ
5. ยืนยันการลงทะเบียน โดย

- ใส่ E-mail address ที่ต้องการใช้กับใบรับรองนี้ กล่าวคือใบรับรอง 1 ใบจะใช้ได้กับ 1 E-mail address เท่านั้น

- ใส่องค์กรที่คุณสังกัดอยู่ ในกรณีไม่ได้สังกัดองค์กรใดไม่ต้องใส่ข้อมูล

- ตั้งรหัสผ่านและยืนยันรหัสผ่าน รหัสผ่านนี้จะต้องนำไปใช้เมื่อต้องการนำกุญแจส่วนตัวมาใช้หรือขอใช้ บริการอื่นๆเพื่อยืนยันว่าคุณเป็นเจ้าของใบรับรองที่แท้จริง

เมื่อทำการลงทะเบียนเรียบร้อยทาง Khonthai CA จะอนุญาตให้ผู้ขอใบรับรองใช้

ใบรับรองดังกล่าวได้ทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สิ่งที่ได้รับหลังจากลงทะเบียน

- สิทธิในการใช้ใบรับรอง โดยใบรับรองจะมีกุญแจสาธารณะของคุณอยู่
- สิทธิในการใช้กุญแจส่วนตัว (Private Key)

ทั้งใบรับรองและกุญแจส่วนตัวนั้นจะยังอยู่บน Server ของ Khonthai CA เพื่อความปลอดภัยจนกระทั่งเจ้าของใบรับรองต้องการใช้งาน Khonthai CA จะมีวิธีการนำออกมาใช้งานแบบเฉพาะของ Khonthai CA ที่มีความปลอดภัยสูงซึ่งจะกล่าวถึงต่อไป

สิ่งที่ควรปฏิบัติหลังจากได้ใบรับรองแล้ว

- ตรวจสอบรายละเอียดและสถานะของใบรับรองโดยการ "ค้นหาใบรับรอง" จากหน้าค้นหา

4.6 Secure Sockets Layer (SSL)

HTTPS จะใช้การพิสูจน์ตัวตนระหว่างฝั่ง client และฝั่ง server ผ่านเอกสารรับรอง certification ไม่ว่าจะแบบตั้งขึ้นเอง (self-signed) หรือแบบที่ขอจากผู้ที่มีความน่าเชื่อถือ (certification agency) สำหรับเว็บเซิร์ฟเวอร์ที่ทำงานอยู่บนโปรโตคอล HTTP นั้น HTTPS/SSL สามารถที่จะถูกนำมาใช้ร่วมกับการรักษาความปลอดภัยในระดับ message (message-level) ซึ่ง SSL ถือว่าเป็นความปลอดภัยในระดับ transport (transport-level) โดย HTTPS จะทำการเข้ารหัส HTTP data packet ทั้งหมด

SSL เป็นโปรโตคอลระดับ Application นิยมใช้ในการเพิ่มความปลอดภัยของการสื่อสารผ่านโปรโตคอล HTTP ซึ่งเป็นการสื่อสารผ่านเว็บ ทำให้เกิดการสื่อสารอย่างปลอดภัยระหว่าง Client และ Server โดยหลักการทำงานแบ่งออกเป็น 3 ส่วนใหญ่ๆ คือ

1. การตรวจสอบ server ว่าเป็นความจริง : ตัวโปรแกรม client ที่มีขีดความสามารถในการสื่อสารแบบ SSL จะสามารถตรวจสอบเครื่อง server ที่ตนกำลังจะไปเชื่อมต่อได้ว่า server นั้นเป็น server ตัวจริงหรือไม่ โดยใช้เทคนิคการเข้ารหัสแบบ public key ในการตรวจสอบใบรับรอง (certificate) และ public ID ของ server นั้น (โดยที่เมื่อครั้งที่ client เชื่อมถือเป็นผู้ถือใบรับรองและ public ID ให้แก่ server นั้น)

หน้าที่นี้ของ SSL เป็นหน้าที่ที่สำคัญ โดยเฉพาะอย่างยิ่งในกรณีที่ client ต้องการที่จะส่งข้อมูลที่เป็นความลับ (เช่น หมายเลข credit card) ให้กับ server ซึ่ง client จะต้องตรวจสอบก่อนว่า server เป็นตัวจริงหรือไม่

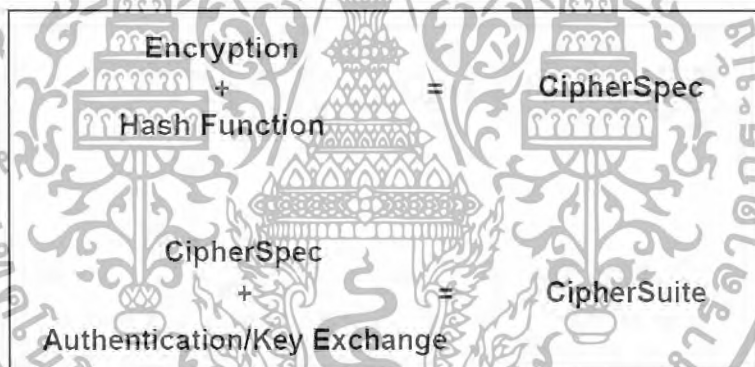
2. การตรวจสอบว่า client เป็นตัวจริง: server ที่มีขีดความสามารถในการสื่อสารแบบ SSL จะใช้เทคนิคเช่นเดียวกับในหัวข้อที่แล้วในการตรวจสอบ client หรือผู้ใช้ว่าเป็นตัวจริงหรือไม่ โดยจะตรวจสอบใบรับรองและ public ID (ที่เมื่อครั้งที่ server เชื่อมถือเป็นผู้ถือใบรับรอง) ของ client หรือผู้ใช้นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่นี้ของ SSL จะมีประโยชน์ในกรณีเช่น ธนาคารต้องการที่จะส่งข้อมูลลับทางการเงิน ให้แก่ลูกค้าของตนผ่านทางเครือข่าย Internet (server ก็จะต้องตรวจสอบ client ก่อนว่าเป็น client นั้นจริง)

3. การเข้ารหัสลับการเชื่อมต่อ: ในกรณีนี้ ข้อมูลทั้งหมดที่ถูกส่งระหว่าง client และ server จะถูกเข้ารหัสลับ โดยโปรแกรมที่ส่งข้อมูลเป็นผู้เข้ารหัสและโปรแกรมที่รับข้อมูลเป็นผู้ถอดรหัส (โดยใช้วิธี public key) นอกจากการเข้ารหัสลับในลักษณะนี้แล้ว SSL ยังสามารถปกป้องความถูกต้องสมบูรณ์ของข้อมูลได้อีกด้วย กล่าวคือ ตัวโปรแกรมรับข้อมูลจะทราบได้ หากข้อมูลถูกเปลี่ยนแปลงไปในขณะกำลังเดินทางจากผู้ส่ง ไปยังผู้รับ

เมื่อมีการใช้ SSL ผู้ใช้สามารถระบุอัลกอริทึมที่ต้องการในการเข้ารหัส, กำหนด hash function ที่ใช้ในการสร้าง message digest หรือ MAC (message authentication code) ซึ่งทั้งสองสิ่งจะถูกเรียกว่า CipherSpec นอกจากนี้แล้วยังต้องรู้อัลกอริทึมที่ใช้สำหรับการพิสูจน์ตัวตน (Authentication) และ key ที่แลกเปลี่ยนกัน โดย authentication/key exchange และ cipherspec จะถูกเรียกว่า CipherSuite แสดงดังภาพ



รูปที่ 4.18 ความหมายของ CipherSpec และ CipherSuite

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยกระบวนการในการเริ่มต้นการสื่อสารผ่านชั้น SSL มี ขั้นตอนดังนี้

1. ประกาศชุดวิธีการเข้ารหัส โคลเ็นต์และลายเซ็นดิจิทัลที่สนับสนุนของทั้ง โคลเ็นต์และเซิร์ฟเวอร์

โคลเ็นต์และเซิร์ฟเวอร์ส่งข้อความเริ่มต้นการสื่อสาร (Hello message) ซึ่งประกอบไปด้วยเวอร์ชันของ โพรโตคอลที่ใช้ วิธีการเข้ารหัสที่เว็บเซิร์ฟเวอร์และโคลเ็นต์สนับสนุน หมายเลขระบุการสื่อสาร (Session identifier) รวมถึงวิธีการบีบอัดข้อมูลในการสื่อสารที่สนับสนุนหมายเลขระบุการสื่อสารที่เกิดขึ้น ใช้สำหรับตรวจสอบการเชื่อมต่อระหว่างโคลเ็นต์และเซิร์ฟเวอร์ ถ้ามีการเชื่อมต่อก่อนหน้านี้เกิดขึ้น แสดงว่าได้มีการตกลงวิธีการสื่อสารแล้ว สามารถเริ่มต้นส่งข้อมูล ได้ทันที เป็นการลดเวลาติดต่อสื่อสารลง

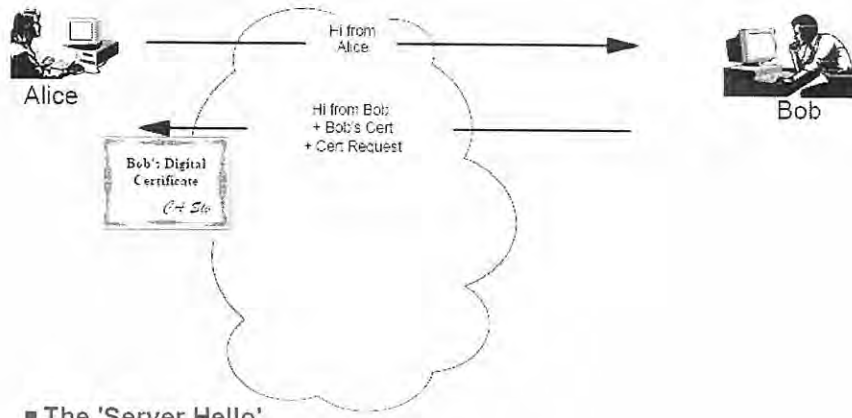


รูปที่ 4.19 การเริ่มต้นสื่อสารด้วย Hello Message

2. การพิสูจน์ตัวตนของเซิร์ฟเวอร์ต่อ โคลเ็นต์

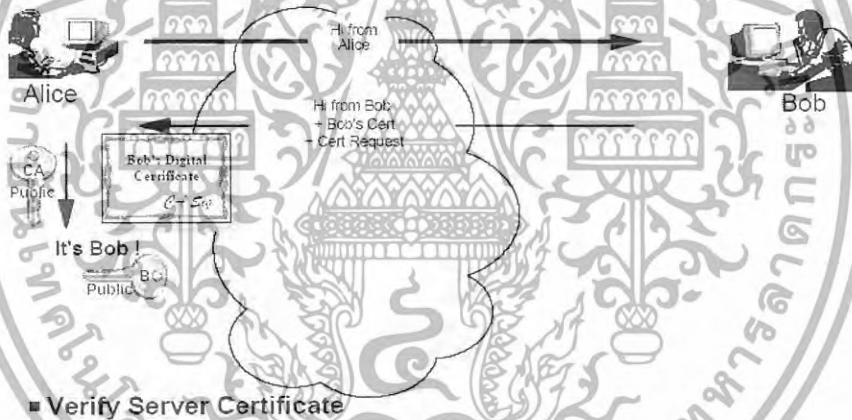
ถัดมาเว็บเซิร์ฟเวอร์ทำการส่ง Certificate หรือใบยืนยันความมีตัวตนของเซิร์ฟเวอร์ โคลเ็นต์จะทำการตรวจสอบ Certificate กับผู้ให้บริการ Certificate Authority ที่ได้ตั้งค่าไว้ เพื่อยืนยันความถูกต้องของ Certificate ของเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- The 'Server Hello'
- The Server Certificate
- The Client Certificate Request

รูปที่ 4.20 ขั้นตอน Server Hello



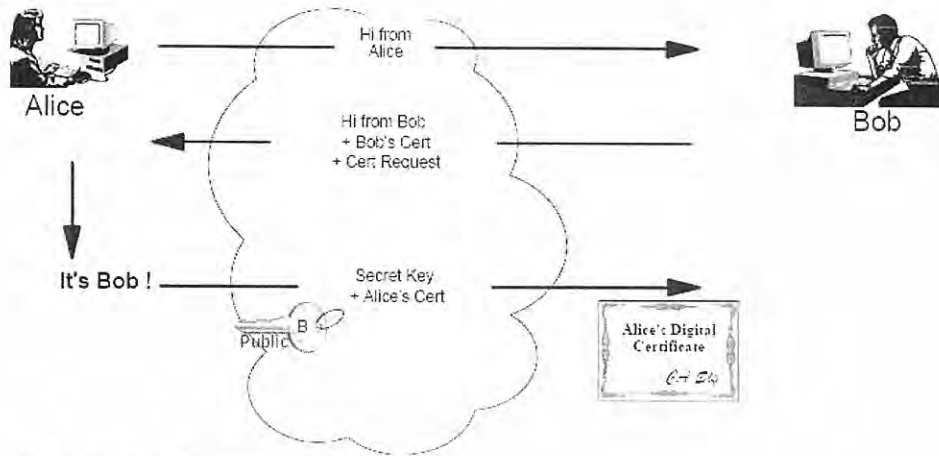
- Verify Server Certificate

รูปที่ 4.21 ขั้นตอนการตรวจสอบ certificate ของ server

3. การพิสูจน์ตัวตนของไคลเอ็นต์ต่อเซิร์ฟเวอร์ ถ้าจำเป็น

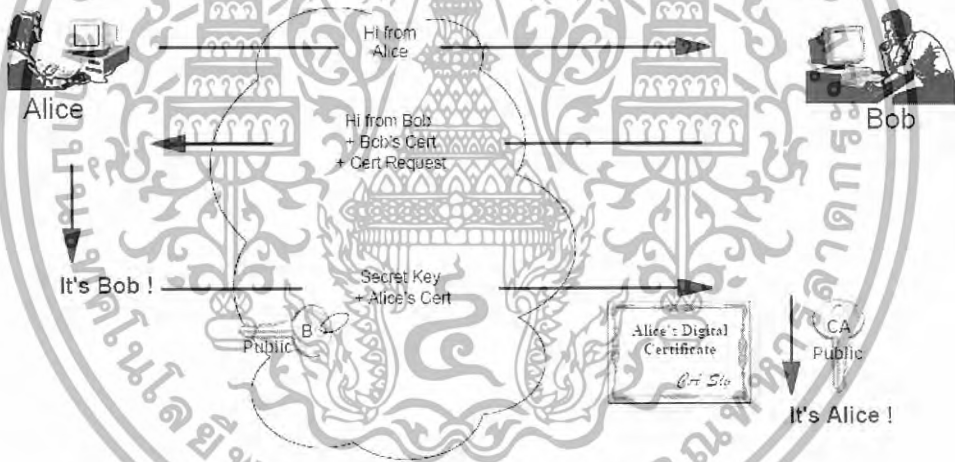
เซิร์ฟเวอร์สามารถร้องขอ Certificate จากไคลเอ็นต์เพื่อตรวจสอบความถูกต้องของ Client ด้วยก็ได้ โดยใช้ในกรณีที่มีการจำกัดการใช้งานเฉพาะไคลเอ็นต์ที่ต้องการเท่านั้น ซึ่ง SSL สนับสนุนการตรวจสอบที่ได้จากทั้งเซิร์ฟเวอร์และไคลเอ็นต์ ขึ้นอยู่กับการเลือกใช้งาน ในขณะติดต่อสื่อสารที่เกิดขึ้นนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Client Key Exchange

รูปที่ 4.22 ขั้นตอน Client Key Exchange

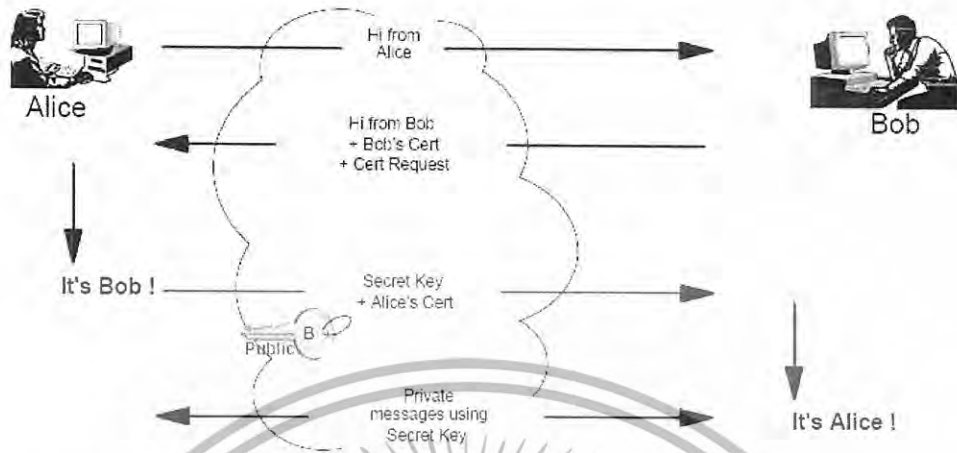


Verify Client Certificate

รูปที่ 4.23 ขั้นตอนตรวจสอบ Client Certification

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. โคลเอนต์และเซิร์ฟเวอร์ตกลงชุดวิธีการเข้ารหัสการสร้างไคเจสต์ และการใช้ลายเซ็นดิจิทัล



- Send Information using agreed Secret Key
- This is now a 'secure line'

รูปที่ 4.24 การขั้นตอนสุดท้ายของ SSL

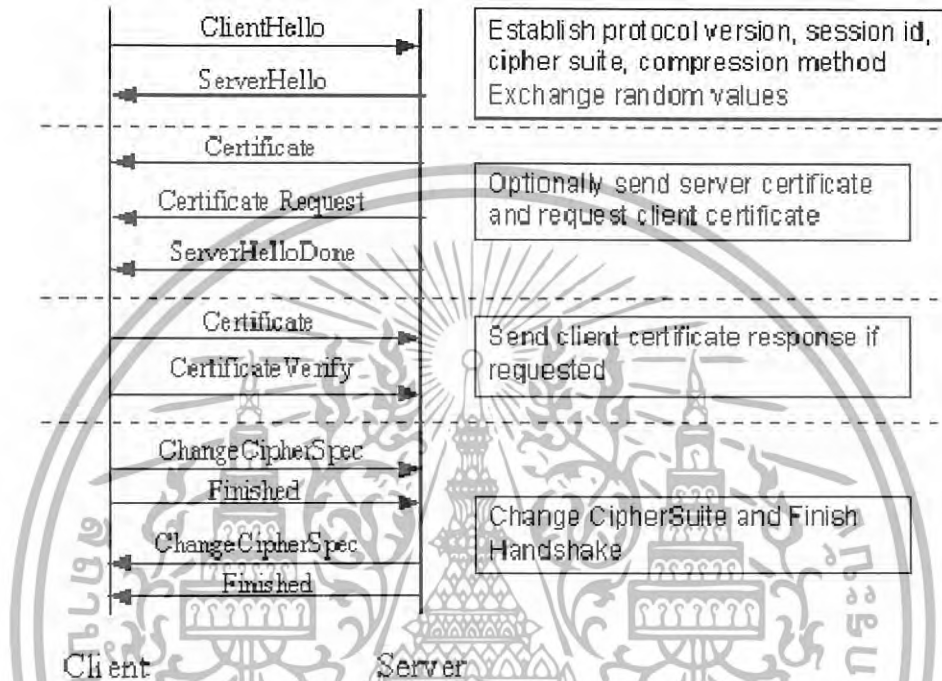
วิธีการแลกเปลี่ยนกุญแจในการเข้ารหัส (Key exchange method) คือการกำหนดกลไกการแลกเปลี่ยนกุญแจที่ใช้ในการเข้ารหัสระหว่างการสื่อสาร โดยทั้งโคลเอนต์และเซิร์ฟเวอร์จะใช้กุญแจนี้ในการเข้ารหัสและถอดรหัสข้อมูล ใน SSL เวอร์ชัน 2.0 จะสนับสนุนวิธีการแลกเปลี่ยนกุญแจแบบ RSA ส่วน SSL เวอร์ชัน 3.0 ขึ้นไปจะสนับสนุนวิธีการอื่นๆ เพิ่มเติมเช่นการใช้ RSA ร่วมกับการใช้ Certificate หรือ Diffie-Hellman เป็นต้น

วิธีการเข้ารหัสในปัจจุบันแบ่งเป็นสองวิธีคือ การใช้กุญแจเดียวกันในการเข้ารหัสและถอดรหัส อาจเรียกกุญแจนี้ว่า Session key หรือ Secret key ส่วนอีกวิธีการคือ การใช้กุญแจคนละตัวในการเข้ารหัสและถอดรหัส ประกอบไปด้วยกุญแจสาธารณะและกุญแจส่วนตัวซึ่งเป็นคู่กันเสมอ การเข้ารหัสด้วยกุญแจใด จะต้องถอดรหัสด้วยกุญแจที่คู่กันและตรงกันข้ามเท่านั้น มักใช้วิธีการเข้ารหัสด้วยกุญแจคนละตัวมาใช้ในการเข้ารหัส Session key และส่งไปให้ฝั่งตรงข้ามก่อนการสื่อสารจะเกิดขึ้นรวมเรียกว่าวิธีการแลกเปลี่ยนกุญแจในการเข้ารหัส

SSL ใช้วิธีการเข้ารหัสด้วยกุญแจสมมาตร หรือกุญแจเดียวในการเข้ารหัสและถอดรหัส ตามที่กล่าวข้างต้น วิธีการเข้ารหัสคือ การเข้ารหัสด้วย DES และ 3DES (Data Encryption Standard), วิธีการเข้ารหัสด้วย IDEA ส่วน RC2 และ RC4 เป็นวิธีการเข้ารหัสของ RSA รวมถึงวิธีการเข้ารหัสแบบ Fortezza สำหรับความยาวของการเข้ารหัสที่ใช้คือ 40 บิต, 96 บิต และ 128 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้าง Message Authentication Code (MAC) เพื่อใช้สำหรับการยืนยันความถูกต้องของข้อมูลระหว่างการสื่อสารและป้องกันการปลอมข้อมูล ส่วนฟังก์ชันสร้างไคเจสต์ที่ SSL สนับสนุนและเลือกใช้ได้ในปัจจุบันคือ MD5 ขนาด 128 บิต และ SHA-1 (Secure Hash Algorithm) ขนาด 160 บิต ซึ่งจะได้วิธีการที่ทั้งสองฝ่ายสนับสนุนและเหมาะสมซึ่งเป็นขั้นตอนสุดท้ายก่อนการสื่อสารที่มีการเข้ารหัสจะเริ่มต้นขึ้น



รูปที่ 4.25 ภาพรวมกระบวนการเริ่มต้นการติดต่อสื่อสารของโพรโตคอล SSL

ข้อดีของ SSL

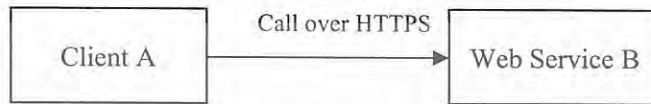
- มีการเตรียมโพรโตคอลสำหรับฟังก์ชันที่เราต้องการ
 - Encryption
 - Message Integrity Checking
 - Authentication
- ใช้ Public/Private Key
 - ไม่มีปัญหาเรื่องการกระจาย key
- ได้รับการยอมรับอย่างกว้างขวางในสังคม Internet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7 ความจำเป็นที่จะต้องใช้ WS-Security ทั่วไปกับ Secure Sockets Layer (SSL)

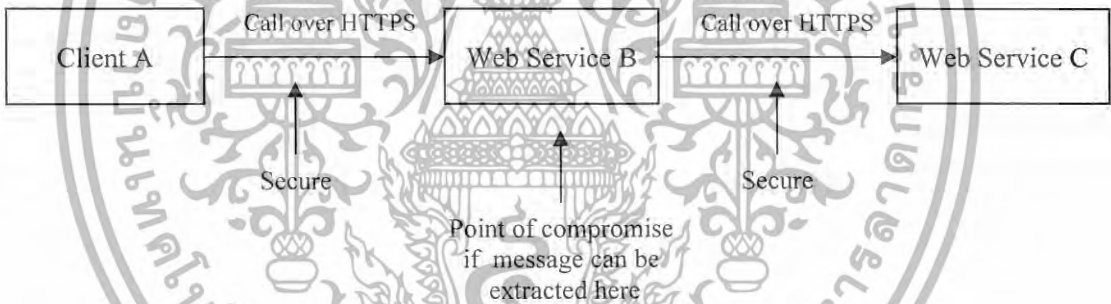
เนื่องจาก SSL นั้นยังมีข้อเสียอยู่บางอย่างที่ไม่เหมาะสมจะนำมาใช้กับเว็บเซอร์วิส โดยมีเหตุผลหลัก 2 ข้อ ดังต่อไปนี้

1. SSL นั้นเป็นการรักษาความปลอดภัยแบบ จุด-จุด (point-to-point security) นั้นหมายถึงว่าจะมีความปลอดภัยแต่จากจุดเริ่มถึงจุดที่มันส่งไปหา ซึ่งการรักษาความปลอดภัยแบบนี้จะใช้ได้แค่กรณีแบบติดต่อกันทีละแค่ 2 เครื่องเท่านั้น ดังรูปด้านล่าง



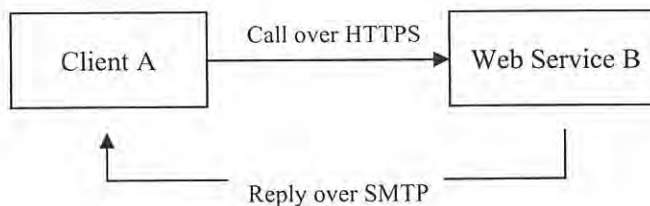
รูปที่ 4.26 การรักษาความปลอดภัยแบบ จุด-จุด บน HTTPS

แต่ถ้าเว็บเซอร์วิสส่งต่อไปให้เว็บเซอร์วิสอื่น กรณีนี้ถึงแม้ว่าจะมีความปลอดภัยจากเอไปบี และจากบีไปซีก็ตามแต่เราก็ไม่สามารถมั่นใจได้เลยว่าข้อมูลที่ถูส่งจากเอไปซีจะมีความปลอดภัย เพราะข้อมูลสามารถเกิดการเปลี่ยนได้ในเว็บเซอร์วิส ดังรูปด้านล่าง



รูปที่ 4.27 การรักษาความปลอดภัยแบบ จุด-จุดบน HTTP และการส่งต่อ

2. เว็บเซอร์วิสนั้นมีความจำเป็นที่ต้องทำงานบนโปรโตคอลหลายอย่าง ซึ่ง SSL นั้นทำงานอยู่บน HTTP เท่านั้น เพราะฉะนั้นเมื่อมีการทำงานบนโปรโตคอลอื่น เช่น SMTP ก็ไม่สามารถที่จะมีความปลอดภัยได้ ดังรูปด้านล่าง



รูปที่ 4.28 การรักษาความปลอดภัยแบบ จุด-จุดบน HTTP และไม่ปลอดภัยบน SMTP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากข้อเสียข้างต้นนั้น สามารถแก้ไขได้โดยการใช้ควบคู่ไปกับ WS-Security เนื่องจาก WS-Security เป็นวิธีการรักษาความปลอดภัยแบบ จบ-จบ (end-to-end) และยังไม่ยึดติดกับโปรโตคอลใดอีกด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

ทรานแซกชัน

ในบทนี้จะพูดถึงเกี่ยวกับ Web Services Transaction ที่เราได้นำมาใช้ในโครงการและแสดงให้เห็นการทำงานของ WS-Transaction และการนำมาใช้งานในโครงการ

5.1 ทรานแซกชัน คืออะไร

Transaction หมายถึง ชุดลำดับของคำสั่งในการประมวลผลข้อมูลที่ถูกจัดรวมเป็นหนึ่งงาน ซึ่งคุณสมบัติ ดังนี้

1. Atomicity จะทำการประมวลผลอย่างแยกออกจากกันไม่ได้ ถ้าทำงานไม่สำเร็จทั้งหมดก็ต้องยกเลิกทั้งหมด
2. Consistency ฐานข้อมูลจะต้องอยู่ในสภาพที่ถูกต้องเสมอ ตราบใดที่โปรแกรมของผู้ใช้ไม่มีข้อผิดพลาดในทางตรรกะธุรกิจ (business logic) การประมวลผล Transaction จะต้องไม่ทำให้ข้อมูลในฐานข้อมูลผิดพลาด
3. Isolation ไม่ว่าในระบบจะมี Transaction ทำงานอยู่ในจำนวนเท่าไรก็ตาม การทำงานของ Transaction หนึ่งจะต้องทำงานร่วมกับ Transaction อื่นเสมือนว่าแต่ละ Transaction นั้นทำงานเป็นอิสระซึ่งกันและกัน
4. Durability เมื่อ Transaction หนึ่งทำงานสิ้นสุดลงแล้ว ผลของการประมวลผลของ Transaction จะต้องไม่สูญหาย ถึงแม้ว่าระบบมีเหตุขัดข้องหลังจากที่ Transaction จะเสร็จสิ้น เช่น เครื่องหยุดการทำงานเนื่องจากไฟฟ้าดับ ระบบจัดการฐานข้อมูลจะต้องรักษาสถานะของฐานข้อมูลที่ถูกต้อง

คุณสมบัติทั้ง 4 ข้อ เรียกว่ากฎ ACID เพื่อให้บรรลุตามคุณสมบัตินี้ ระบบจัดการฐานข้อมูลต้องทำงาน ดังต่อไปนี้

1. ระบบจัดการฐานข้อมูลจะทำงานในลักษณะของ Transaction โดยสามารถรับการกำหนดจากโปรแกรมระบบงานของผู้ใช้ว่า Transaction นั้นเริ่มต้นและสิ้นสุดเมื่อไร
2. เก็บรายการเปลี่ยนแปลงในฐานข้อมูล (Logging) ทุกรายการที่มีการเปลี่ยนแปลงจะต้องมีการเก็บบันทึกลงในสื่อ เช่น Hard disk เมื่อระบบมีข้อผิดพลาดเกิดขึ้น ระบบจัดการฐานข้อมูลสามารถนำข้อมูลดังกล่าวมาปรับปรุงข้อมูลให้อยู่ในสภาพที่ถูกต้องดังเดิม
3. ความคุมการทำงานพร้อมกัน (Concurrency Control) ถ้า Transaction มากกว่าหนึ่งงานจะต้องทำงานเสมือนแต่ละงานเป็นอิสระต่อกัน ระบบจัดการฐานข้อมูลจะทำการควบคุมการทำงานพร้อมกันได้ด้วยการควบคุมการเข้าถึงข้อมูลในระบบฐานข้อมูล โดยระบบจัดการฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะต้องจัดลำดับในการประมวลผล เพื่อให้การทำงาน โดยรวมเทียบเท่ากับแต่ละงานทำงานเป็นหน่วยเดียวกันตั้งแต่ต้นจนจบ

4. การแก้ไขการติดขัด (Deadlock Resolution) ในกรณีที่เกิดการแย่งกันเข้าถึงข้อมูล จนทำให้ระบบไม่สามารถทำงานต่อไปได้ ระบบจัดการฐานข้อมูลจะต้องทำการแทรกแซงและยกเลิกการทำงานของบางรายการเพื่อให้ระบบ โดยรวมสามารถทำงานต่อไปได้

Specification ของ Web Services Transactions กำหนดวิธีการทำงานของ Transaction ในการติดต่อสื่อสารระหว่าง Web Services ที่เป็นผู้เรียกและ Web Services ที่เป็นผู้ให้บริการในการดึงเอา Transaction มาใช้ในการเพิ่มคุณภาพของการติดต่อสื่อสารบน Web Services

Specification ของ Web Services Transactions ได้ทำการระบุรายละเอียดสำหรับ

- Short duration, ACID transactions (WS-AtomicTransaction)
- Long running business transactions (WS-BusinessActivity)

5.2 WS-Coordination (WS-COOR)

มาตรฐานนี้จะระบุ Framework สำหรับการจัดหา protocol ซึ่งคอยประสานการทำงานของ application ที่อยู่บนระบบแบบ distributed โดย protocol สำหรับการประสานงานนั้นจะใช้สำหรับรองรับจำนวนของ application หลายๆ application โดยจะรวมทุกๆ อย่างที่จำเป็นในการติดต่อสื่อสารกันของระบบแบบ distributed

Framework ที่ถูกกำหนดไว้ใน specification นี้จะทำให้ application ที่เป็นผู้ประสานงานสามารถสร้างสภาพแวดล้อมเพื่อประกาศการกระทำในขณะนั้นให้กับ services อื่นๆ และยังเป็น การลงทะเบียนสำหรับ protocol ในการประสานงานและ framework ยังคงทำให้การดำเนินการกับ transaction ในปัจจุบัน, การไหลของงานที่ทำและระบบอื่นๆ สำหรับการประสานงานสามารถที่จะ ซ่อน protocol ที่เป็นเจ้าของเอาไว้และยังสามารถทำงานได้ในระบบที่ต่างกัน

นอกจากนี้ Specification นี้ยังอธิบายเกี่ยวกับคำนิยามของ โครงสร้างสภาพแวดล้อมและความ ต้องการของระบบในการสนับสนุนการปฏิบัติงานร่วมกันของ services ต่างๆ

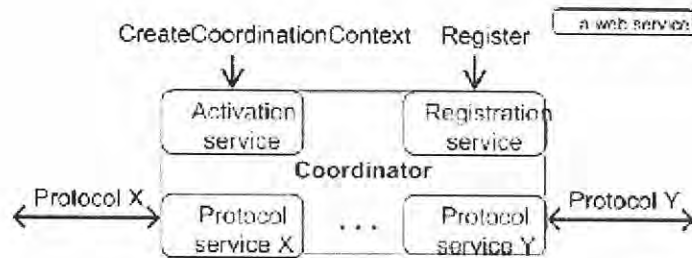
5.2.1 Model

WS-Coordination ได้ทำการบรรยายถึงโครงของ Coordination Service (Coordinator) เอาไว้ซึ่งประกอบไปด้วย Service ดังนี้

- Activation Service, ซึ่งเป็นกระบวนการในการสร้าง Coordination instance (Context)
- Registration Service, ซึ่งเป็นกระบวนการในการให้ แอปพลิเคชัน register เข้ากับ coordination protocol

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Coordination Type ซึ่งเป็นเซตของ Coordination protocol



รูปที่ 5.1 เซอร์วิสที่ประกอบภายในCoordinator

แอปพลิเคชัน มีการใช้ Activation Service ในการที่จะสร้าง Coordination Context สำหรับ Activity ของตัวเอง เมื่อ Coordination Context ถูกนำไปโดย แอปพลิเคชัน อื่น Coordination Context จะถูกส่งไปด้วยอะไรก็ตามที่เหมาะสมสำหรับ แอปพลิเคชัน อื่นๆ นั้น

Context จะประกอบไปด้วยข้อมูลที่สำคัญในการที่จะให้ แอปพลิเคชัน อื่นๆ นั้นสามารถที่จะเข้ามา Register สำหรับ Activity ได้โดยการกำหนดพฤติกรรมของ Coordination ที่แอปพลิเคชันนั้นต้องทำตาม

นอกจากนั้น แอปพลิเคชัน ที่ได้รับ Coordination Context ยังสามารถที่จะใช้ Registration Service กับตัวมันเอง หรืออาจจะใช้กับ Trusted Coordinator อื่นก็ได้ ในกรณีนี้ กลุ่มของ Web Service อาจจะเข้ามาสัมพันธ์กันในกระบวนการเชื่อมต่อ

5.2.2 Extensibility

โครงสร้างของ WS-Coordination มีความสามารถในการที่จะต่อขยายรวมไปถึงมีความยืดหยุ่นสำหรับการใช้งาน โดยโครงสร้างจะอนุญาตให้

- สามารถทำการตีพิมพ์ เผยแพร่ Coordination Protocols ใหม่ๆ ได้
- Protocol ที่ถูกเลือกมาจาก Coordination Type และคำจำกัดความของส่วนที่ต่อขยาย สามารถทำการเพิ่มเข้าไปใน Protocol และ message flows

ส่วนที่ทำการต่อขยายเข้ามาสามารถใช้ในการแลกเปลี่ยนข้อมูลได้ใน message flow ที่ถูกจำกัดความไว้ใน WS-Coordination ซึ่งนำไปสู่ความต้องการในการแลกเปลี่ยนข้อมูลใน Isolation-level หรือใน Business-Level Coordination protocols ข้อมูลที่แลกเปลี่ยนกันสามารถทำการบันทึกไว้ได้เพื่อทำการตรวจสอบในภายหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.3 Coordination Context

Coordination Context คือ ชนิดของ Context ชนิดหนึ่งที่ใช้ในการส่งข้อมูลของ Coordination ให้กับ Participants อื่นๆทุกตัวที่เกี่ยวข้องใน Coordination Service (Coordinator) ซึ่งตัว Coordination Context จะถูกใส่รวมไว้กับ แอปพลิเคชัน (Conveying a context-flowing the context)

CoordinationContext จะให้ความสามารถในการ coordination registration service, Coordination Type และ relevant extensions ตัวอย่างของ *CoordinationContext*

```
<?xml version="1.0" encoding="utf-8"?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
<S:Header>
... <wscoor:CoordinationContext
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns:wscoor="http://schemas.xmlsoap.org/ws/2004/10/wscoor"
xmlns:myApp="http://fabrikam123.com/myApp"
S:mustUnderstand="true">
<wscoor:Identifier>
http://Fabrikam123.com/SS/1234
</wscoor:Identifier>
<wscoor:Expires>3000</wscoor:Expires>
<wscoor:CoordinationType>
http://schemas.xmlsoap.org/ws/2004/10/wsat
</wscoor:CoordinationType>
<wscoor:RegistrationService>
<wsa:Address>
http://Business456.com/mycoordinationsservice/registration
</wsa:Address>
<wsa:ReferenceProperties>
<myApp:BetaMark> ... </myApp:BetaMark>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<myApp:EBDCode> ... </myApp:EBDCode>

</wsa:ReferenceProperties>

</wscoor:RegistrationService>

<myApp:IsolationLevel>
RepeatableRead
</myApp:IsolationLevel>

</wscoor:CoordinationContext>

...

</S:Header>

</S:Body>

...

</S:Body>

</S:Envelope>

```

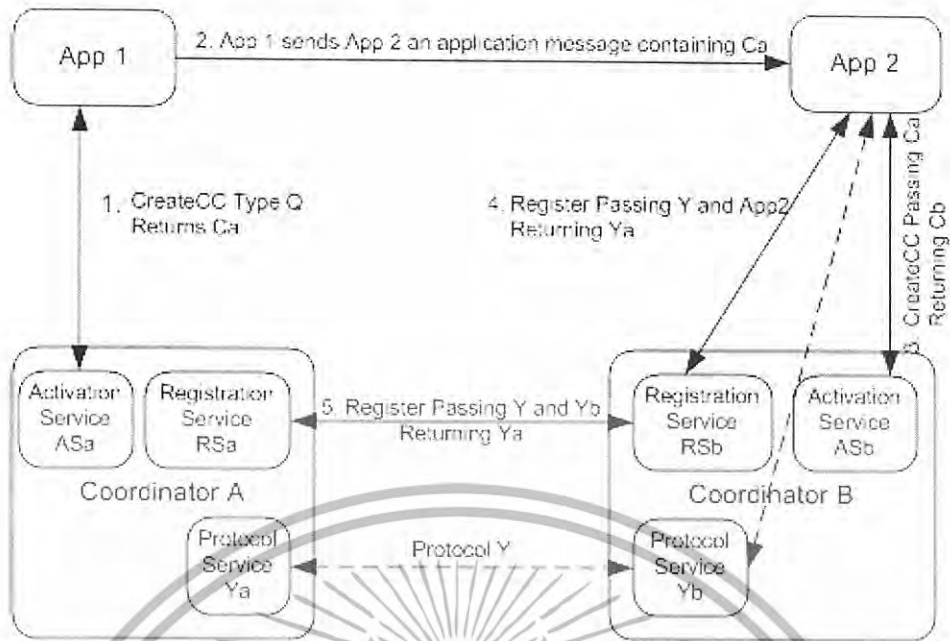
แอปพลิเคชัน ต้องการใช้ Activity ที่เหมาะสมจาก Coordination Service (Coordinator) แอปพลิเคชัน ต้องมีการรวมเอา Coordination Context เอาไว้ใน message ด้วย Context ถูกแลกเปลี่ยนกันในลักษณะของ SOAP header 필ด์ mustUnderstand จะต้องถูกเซทให้มีค่าเป็นจริง เพื่อให้ แอปพลิเคชัน ต้องทำการรับ SOAP header (เนื่องจาก SOAP header เป็นเพียงตัวเลือก แอปพลิเคชัน อาจจะไม่ต้องสนใจก็ได้ แต่ถ้ามีการเซท mustUnderstand ให้เป็นจริง แอปพลิเคชัน จะต้องรับ SOAP header)

5.2.4 Coordination Service (Coordinator)

Coordination Service (Coordinator) เป็นการประกอบไปด้วย Service ดังต่อไปนี้

- Activation Service มีการกำหนดให้ CreateCoordinationContext เป็นการสร้าง CoordinationContext โดย Coordination Service อาจจะรองรับ Activation Service
- Registration Service เป็นการกำหนดขั้นตอนวิธีการของการ Register ซึ่งมีส่วนในการอนุญาตให้ Web Service เข้าร่วมใน Coordination protocol โดย Coordination service ต้องรองรับ Registration Service
- เซทของ Coordination protocol สำหรับแต่ละ Coordination type ที่รองรับ ซึ่งทั้งหมดนี้จะถูกกำหนดในข้อกำหนดในส่วนของ Coordination Type

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 แอปพลิเคชัน และ Coordinator

จากรูปจะเป็นการแสดงการที่แอปพลิเคชัน Services (App1 และ App2) กับ Coordinator ของตัวเอง (CoordinatorA และ CoordinatorB) ติดต่อกัน โดยมี Protocol Y และ Services Ya และ Yb เป็น Coordination Type ซึ่งไม่ได้ระบุใน WS-Coordination โดยจะมีขั้นตอนดังนี้

1. App1 ทำการส่ง CreateCoordinationContext สำหรับ Coordination Type Q โดยได้รับ Context Ca กลับมาซึ่งประกอบไปด้วยตัวระบุถึง Activity (Activity identifier) A1 โดยที่ Coordination type Q และมี Endpoint ที่อ้างถึง Registration Service ของ CoordinatorA (RSa)
2. จากนั้น App1 จะส่ง แอปพลิเคชัน message ไปยัง App2 ซึ่งประกอบไปด้วย Context Ca
3. แต่เนื่องจาก App2 ต้องการใช้ CoordinatorB เพราะฉะนั้น App2 จึงใช้ CreateCoordinationContext โดยมี Ca เป็น input แล้วทำการเสนอให้กับ CoordinatorB จากนั้น CoordinatorB จึงทำการสร้าง CoordinationContext Cb ของมันเองซึ่งประกอบไปด้วยตัวระบุถึง Activity (Activity identifier) เช่นเดียวกัน และ Coordination Type เหมือนกับ Ca แต่เป็นการ Register เข้ากับ Service ของตัวมันเอง RSb
4. App2 ทำการพิจารณา Coordination protocol ที่รองรับของ Coordination type Q จากนั้นทำการ Register สำหรับ Coordination protocol Y ที่ CoordinatorB จากนั้นทำการเปลี่ยนจุด Endpoint ให้อ้างถึง App2 และ Protocol Service Yb ซึ่งนี้จะทำให้เกิดการเชื่อมต่อเสมือนระหว่างจุด Endpoint ที่ Protocol Y สามารถใช้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การ Register นี้จะทำให้ CoordinatorB ทำการส่งต่อการ register ไปยัง Registration Service ของ CoordinatorA (RSa) และทำการแลกเปลี่ยนจุด Endpoint สำหรับ Yb และ Protocol Service Ya ซึ่งจะทำให้เสมือนเกิดการเชื่อมต่อระหว่าง Endpoint ที่ protocol Y สามารถใช้ได้

Activation Service

สำหรับ Activation Service จะทำการสร้าง Activity ใหม่และคืนค่าเป็น Coordination Context ของมันกลับมา

แอปพลิเคชัน ส่ง CreateCoordinationContext จากนั้น Activation Service จะทำการคืนค่าของ CreateCoordinationContextRespond

- CreateCoordinationContext การร้องขออันนี้จะทำให้มีการสร้าง coordination context ที่มีการรองรับ coordination type (service ที่จัดหาเซทของ coordination protocol เช่น WS-AtomicTransaction) ซึ่งคำสั่งที่มีความจำเป็นเมื่อมีความต้องการที่จะเข้าถึงเครือข่ายที่มีสถานะแวดล้อมแตกต่างกัน (ต่าง platform กัน)

```
<CreateCoordinationContext ...>
```

```
<Expires> ... </Expires>? <CurrentContext> ... </CurrentContext>?
```

```
<CoordinationType> ... </CoordinationType>
```

```
...
```

```
</CreateCoordinationContext>
```

```
/CreateCoordinationContext/CoordinationType
```

ตัวบ่งบอกเฉพาะที่แสดงถึง coordination type ที่ต้องการสำหรับ activity (เช่น Atomic Transaction)

```
/CreateCoordinationContext/Expires
```

แสดงถึงเวลาที่หมดอายุ โดยอยู่ในรูปของตัวเลขที่ไม่มีเครื่องหมายในหน่วย milliseconds ซึ่งอาจจะมีหรือไม่มีก็ได้

```
/CreateCoordinationContext/CurrentContext
```

อาจจะใช้ได้อย่างหลากหลายรวมไปถึงการ recovery และ subordinate ซึ่งอาจจะมีหรือไม่มีก็ได้

```
/CreateCoordinationContext /{any}
```

Element ที่อาจจะใช้สำหรับรายละเอียดเพิ่มเติม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/CreateCoordinationContext /@{any}

Attributes ที่อาจจะใช้สำหรับรายละเอียดเพิ่มเติม

ซึ่ง message CreateCoordinationContext อาจจะมีโครงสร้างง่าย ๆ ดังนี้ก็ได้

<CreateCoordinationContext>

<CoordinationType>

<http://schemas.xmlsoap.org/ws/2004/10/wsat>

</CoordinationType>

</CreateCoordinationContext>

- CreateCoordinationContextResponse จะเป็นการคืนค่า CoordinationContext ที่ถูกสร้างขึ้นมา โดยจะมีโครงสร้างดังนี้

<CreateCoordinationContextResponse ...>

<CoordinationContext> ... </CoordinationContext>

...

</CreateCoordinationContextResponse>

/CreateCoordinationContext/CoordinationContext

ส่วนนี้คือ CoordinationContext ที่ถูกสร้างขึ้นมา

/CreateCoordinationContext /{any}

Element ที่อาจจะใช้สำหรับรายละเอียดเพิ่มเติม

/CreateCoordinationContext /@{any}

Attributes ที่อาจจะใช้สำหรับรายละเอียดเพิ่มเติม

<CreateCoordinationContextResponse>

<CoordinationContext>

<Identifier>

<http://Business456.com/tm/context1234>

</Identifier>

<CoordinationType>

<http://schemas.xmlsoap.org/ws/2004/10/wsat>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

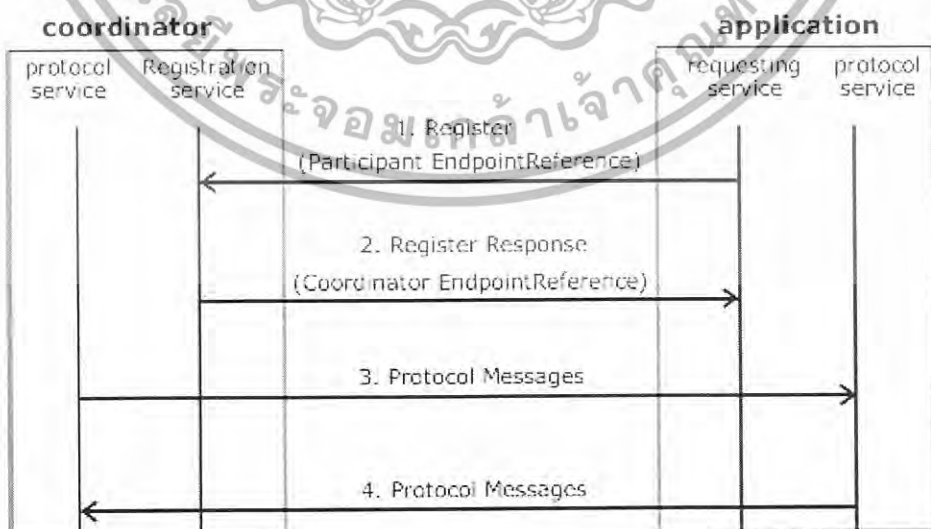
```

</CoordinationType>
<RegistrationService>
<wsa:Address>
http://Business456.com/tm/registration
</wsa:Address>
<wsa:ReferenceProperties>
<myapp:PrivateInstance>
1234
</myapp:PrivateInstance>
</wsa:ReferenceProperties>
</RegistrationService>
</CoordinationContext>
</CreateCoordinationContextResponse>

```

Registration Service

หลังจากที่ แอปพลิเคชัน มี coordination context จาก coordinator ที่ถูกเลือก แอปพลิเคชัน จะสามารถเข้าไป register สำหรับ Activity นั้นๆ ได้ โดย interface ที่ถูกจัดหาให้ สำหรับการ register ของ แอปพลิเคชัน เพื่อเข้าถึง activity และสำหรับ coordinator ในการ register เพื่อเข้าถึง activity นั้นเหมือนกัน



รูปที่ 5.3 การใช้จุดอ้างอิง Endpoint ระหว่างการลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 5.3 coordinator จัดหาจุดอ้างอิง endpoint ในการ registration ให้ใน CoordinationContext ในขั้นตอน CreateCoordinationContext ซึ่ง service ที่ทำการร้องขอจุดอ้างอิง endpoint จะได้รับใน CoordinationContext ซึ่งอยู่ใน แอปพลิเคชัน message

1. register message จะทำการอ้างอิงถึงจุดอ้างอิง endpoint นี้ และทำการรวมเอาจุดอ้างอิง endpoint ของ service participant protocol เอาไว้เป็น parameter
2. RegisterResponse ทำการรวมเอาจุดอ้างอิง endpoint ของ protocol service ของ coordinator
3. & 4. ในจุดนี้ทั้งสองฝั่งต่างมีจุดอ้างอิง endpoint ของ protocol service ของกันและกัน ดังนั้น protocol message สามารถ target ไปยังอีกฝั่งได้

จุดอ้างอิง endpoint เหล่านี้อาจจะมี wsa:ReferenceProperties เป็นส่วนประกอบด้วยก็ได้ จากกฎของการ mapping ที่ถูกระบุไว้ใน WS-Addressing คุณสมบัติการอ้างอิงทั้งหมดต้องถูก copy เหมือนเป็น header ในทุกๆ message ที่ target ไปยัง endpoint

- Register Message
- RegistrationResponse Message

5.2.5 Coordination Fault

WS-Coordination Fault ต้องมีการ include URI:

<http://schemas.xmlsoap.org/ws/2004/10/wxcoor/fault>

โดยใน WS-Coordination จะมีการนิยามความผิดพลาดไว้โดยมีคุณสมบัติดังนี้

[Code] Fault code

[Subcode] Fault subcode

[Reason] ส่วนของเหตุผล

[Detail] รายละเอียดต่างๆ ถ้าว่างคือไม่มีรายละเอียดของความผิดพลาด

สำหรับ SOAP 1.2 [Code] ต้องเป็น Sender หรือ Receiver โดยคุณสมบัติเหล่านี้จะถูกทำให้เป็น XML ดังตาราง

SOAP Version	Sender	Receiver
SOAP 1.2	S:Sender	S:Receiver

ตาราง 5.1 คุณสมบัติของ [Code] สำหรับ SOAP 1.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยสามารถนำเอาคุณสมบัติในตารางข้างต้นมาเชื่อมเป็น SOAP 1.2 ได้ดังนี้

```

<S:Envelope>
<S:Header>
<wsa:Action>
http://schemas.xmlsoap.org/ws/2004/10/wscoor/fault
</wsa:Action>
<!-- Headers elided for clarity. -->
</S:Header>
<S:Body>
<S:Fault>
<S:Code>
<S:Value>[Code]</S:Value>
<S:Subcode>
<S:Value>[Subcode]</S:Value>
</S:Subcode>
</S:Code>
<S:Reason>
<S:Text xml:lang="en">[Reason]</S:Text>
</S:Reason>
<S:Detail>
[Detail]
...
</S:Detail>
</S:Fault>
</S:Body>
</S:Envelope>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Invalid State

สำหรับความผิดพลาดประเภทนี้จะถูกส่งได้ทั้ง Coordinator หรือ Participant เพื่อจะกำหนดจุด Endpoint ที่ทำให้เกิดความผิดพลาดนั้น ซึ่งได้รับ message ที่ไม่ valid สำหรับสถานะตอนนั้นของมัน ซึ่งสถานการณ์นี้ไม่สามารถทำการกู้คืนได้

คุณสมบัติ

[Code] Sender

[Subcode] wscoor:InvalidState

[Reason] The message was invalid for the current state of the activity.

[Detail] unspecified

Invalid Protocol

สำหรับความผิดพลาดประเภทนี้จะถูกส่งได้ทั้ง Coordinator หรือ Participant เพื่อจะกำหนดจุด Endpoint ที่ทำให้เกิดความผิดพลาดนั้น ซึ่งได้รับ message ที่ถูกส่งมาด้วย Protocol ที่ไม่ถูกต้อง ซึ่งสถานการณ์นี้ไม่สามารถทำการกู้คืนได้

คุณสมบัติ

[Code] Sender

[Subcode] wscoor:InvalidProtocol

[Reason] The protocol is invalid or is not supported by the coordinator

[Detail] unspecified

Invalid Parameters

สำหรับความผิดพลาดประเภทนี้จะถูกส่งได้ทั้ง Coordinator หรือ Participant เพื่อจะกำหนดจุด Endpoint ที่ทำให้เกิดความผิดพลาดนั้น ซึ่งได้รับ message ที่มี parameter ที่ invalid ซึ่งสถานการณ์นี้ไม่สามารถทำการกู้คืนได้

คุณสมบัติ

[Code] Sender

[Subcode] wscoor:InvalidParameters

[Reason] The message contained invalid parameters and could not be processed

[Detail] unspecified

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

No Activity

สำหรับความผิดพลาดประเภทนี้จะถูกส่งได้จาก Coordinator ถ้า participant ไม่มีการติดต่อมาเป็นระยะเวลาที่นานเกินไป

คุณสมบัติ

[Code] Sender

[Subcode] wscoor:ContextRefused

[Reason] The coordination context that was provided could not be accepted

[Detail] unspecified

Context Refused

สำหรับความผิดพลาดประเภทนี้จะถูกส่งไปยัง Coordinator เพื่อจะกำหนดจุด Endpoint ที่ไม่สามารถทำการรับ Context ที่ถูกส่งมาได้

คุณสมบัติ

[Code] Sender

[Subcode] wscoor:ContextRefused

[Reason] The coordination context that was provided could not be accepted

[Detail] unspecified

Already Registered

สำหรับความผิดพลาดประเภทนี้จะถูกส่งไปยัง participant ถ้า coordinator ตรวจสอบว่า participant พยายามที่จะ Register สำหรับ protocol ตัวเดียวกันให้กับ activity ตัวเดียวกันมากกว่า 1 ครั้ง

คุณสมบัติ

[Code] Sender

[Subcode] wscoor:AlreadyRegistered

[Reason] The participant has already registered for the same protocol

[Detail] unspecified

5.2.6 Security Model

จุดประสงค์หลักของ Security โดยอ้างอิงถึง WS-Coordination คือ

- ให้แน่ใจว่ามีเพียงขั้นตอนที่ได้รับการอนุมัติแล้วทำการสร้าง Coordination context
- ให้แน่ใจว่ามีเพียงขั้นตอนที่ได้รับการอนุมัติแล้วทำการ Register เข้ากับ Activity

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

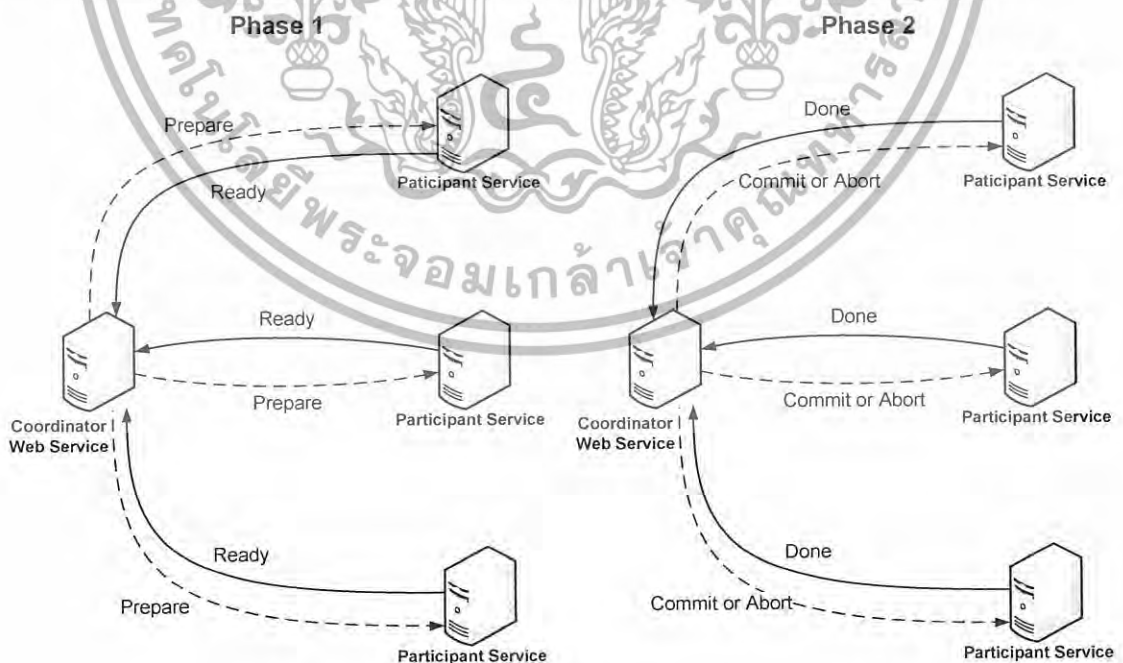
- ให้แน่ใจว่ามีเพียง Coordination Context ที่ถูกต้องตามกฎหมายทำการ Register
- ทำให้การทำงานของ Security ที่มีอยู่มีประสิทธิภาพมากขึ้น
- อนุญาตให้มีการจำกัดสิทธิ์โดยอ้างอิงถึง federated identities

จุดประสงค์หลักเหล่านี้ถูกกำหนดขึ้นมาจากความต้องการของ Security เพื่อ ความถูกต้องของข้อมูล การรักษาความลับ และการแสดงความเป็นเจ้าของ ซึ่งความต้องการแต่ละอย่างจะถูกจัดทำให้โดยองค์กรที่สร้างข้อกำหนดเกี่ยวกับ Web Service Security เช่น WS-Security และ WS-Trust

5.3 WS-AtomicTransaction (WS-AT)

Specification นี้จะให้คำจำกัดความของ atomic transaction ซึ่งเป็นรูปแบบของการประสานงานกันของระบบแบบ distributed ชนิดหนึ่งโดยจะใช้กับ framework ที่เราได้พูดถึงไปใน specification ของ WS-Coordination โดย specification นี้จะทำให้การให้นิยามของ protocol ที่จะทำการประสานงานเป็นไปอย่างสอดคล้องที่จะถูกใช้โดยการประสานงานแบบ atomic transaction ได้ 3 ชนิดด้วยกันคือ

- Completion
- Volatile two-phase commit
- Durable two-phase commit



รูปที่ 5.4 การทำงานของ Two-phase commit protocol

ซึ่งเป็น Protocol ที่ WS-Atomic Transaction ใช้ในการทำ Transaction บน Web Services

เอกสารนี้เป็นเอกสารลิขสิทธิ์การสงวนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่เชิงพาณิชย์
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยผู้พัฒนาระบบจะสามารถใช้ Protocol ใหม่นี้ได้ใน 3 ชนิดนี้ในการสร้าง application ซึ่งต้องการความสอดคล้องกันของผลลัพธ์สำหรับกิจกรรมบนระบบแบบ distributed ที่มีอายุสั้นๆ ที่จะต้องมีคุณสมบัติ **“ทำสำเร็จทั้งหมดหรือไม่ทำเลย”**

5.3.1 การcommit แบบสองระยะ (Two-Phase Commit Protocol)

การCommit แบบสองระยะเป็นกลไกที่จะช่วยให้การทำงานของแอปพลิเคชันแบบกระจายสามารถทำการ Commit Transaction ร่วมกันได้ และถ้ามีความผิดพลาดใดๆเกิดขึ้น Transaction นั้นๆจะต้องเหมือนกับไม่เคยเกิดขึ้นมาก่อน (Rollback) ซึ่งเป็นการช่วยให้คงความถูกต้องของข้อมูล การCommit แบบสองระยะจะประกอบไปด้วยการทำงานสองระยะ ดังนี้

ระยะที่ 1

1. Coordinator ทำการส่ง <Prepare T> ไปให้กับเซอร์วิสที่เกี่ยวข้องใน Transaction ทั้งหมด
2. จากนั้นเซอร์วิสจะทำการดูว่างานที่ได้รับมานั้นสามารถทำงานได้สำเร็จหรือไม่
 - a. ถ้าสำเร็จเซอร์วิสจะทำการส่ง <Ready T> ไปให้กับ Coordinator
 - b. ถ้าไม่สำเร็จเซอร์วิสจะทำการส่ง <Abort T> ไปให้กับ Coordinator
3. Coordinator รอจนกระทั่งได้รับเมสเซจจากทุกๆเซอร์วิสครบแล้ว

ระยะที่ 2

ถ้าทุกๆเมสเซจที่ Coordinator ได้รับคือ <Ready T> แล้ว

1. Coordinator ทำการเขียน <Commit> ลงในไฟล์บันทึก
2. Coordinator ทำการส่ง <Commit T> ไปให้กับเซอร์วิสทุกตัวที่เกี่ยวข้องใน Transaction
3. เซอร์วิสที่เกี่ยวข้องทุกตัวทำการ Commit งานของตัวเองจากนั้นทำการคืนทรัพยากรทั้งหมดที่ใช้ให้กับระบบ
4. เซอร์วิสทุกตัวทำการตอบรับการ Commit ไปยัง Coordinator
5. Coordinator สิ้นสุดการทำงานใน Transaction

ถ้า Coordinator ได้รับ <Abort T> จากเซอร์วิสตัวใดตัวหนึ่ง

1. Coordinator ทำการเขียน <Rollback> ลงในไฟล์บันทึก
2. Coordinator ทำการส่ง <Rollback T> ไปให้กับเซอร์วิสทุกตัวที่เกี่ยวข้องใน Transaction
3. เมื่อเซอร์วิสได้รับ <Rollback T> จะทำการ Rollback งานที่ทำของตน แล้วตอบรับกลับไปยัง Coordinator
4. Coordinator สิ้นสุดการทำงานใน Transaction

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.2 Atomic Transaction Context

Atomic Transaction สร้างขึ้นมาเพื่อทำการอธิบายถึง Coordination Type ที่ได้รับการกำหนดไว้บน WS-Coordination

Atomic Transaction Coordination Context ต้องส่งผ่านอยู่ในทุก แอปพลิเคชัน Message ที่เกี่ยวข้องกับ Transaction นอกจากนั้นยังมีการเพิ่มความหมายของขั้นตอน

CreateCoordinationContext ในส่วนของ Activation Service

- ถ้า request มีการรวมเอาส่วนของ CurrentContext Coordinator จะทำการแทรกเข้าไป เหมือนกับ subordinate ให้กับ coordinator ใน CurrentContext
- ถ้า request ไม่มีการรวมเอาส่วนของ CurrentContext Coordinator จะทำการสร้าง Transaction ใหม่ที่จะทำหน้าที่เหมือนเป็น root

Coordination Context อาจจะมี Expires attribute ประกอบอยู่ด้วยซึ่งเป็นการกำหนดเวลาที่เร็วที่สุดที่ Transaction อาจจะถูก terminate ได้จากการใช้เวลานานเมื่อถึงจุดนั้นแล้ว Transaction Manager อาจจะมีการทำการเลือกว่าจะทำการ roll back transaction นั้นหรือไม่ ถ้ายังไม่มีการ Transmit , Commit หรือ Prepare notification

Atomic Transaction protocol ถูกระบุโดย Coordination Type:

<http://schemas.xmlsoap.org/ws/2004/10/wsat>

5.3.3 Atomic Transaction Protocols

ในส่วนของ WS-AtomicTransaction จะทำการระบุถึง Protocol ต่อไปนี้สำหรับการทำ Atomic Transaction

- Completion: protocol นี้จะทำการเริ่มต้นกระบวนการในการ commit โดยอ้างจากแต่ละ protocol ที่ถูก register โดย participants ซึ่ง Coordinator จะเริ่มต้นด้วย Volatile 2PC จากนั้นจะเข้าไปสู่ Durable 2PC โดยผลลัพธ์สุดท้ายจะถูกส่งให้กับ initiator
- Two-Phase Commit (2PC): 2PC Protocol จะกำหนดให้ participant จะทำการ commit หรือว่า abort และให้แน่ใจว่า participant ทุกตัวได้รับการแจ้งผลลัพธ์สุดท้าย โดย 2PC protocol ประกอบไปด้วย
 - Volatile 2PC: Participant ที่มีการจัดการกับ Volatile resource เช่น Cache ควรจะทำการ register เข้ากับ Protocol
 - Durable 2PC: Participant ที่มีการจัดการกับ Durable resource เช่น Database ควรจะทำการ register เข้ากับ Protocol

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Participant สามารถทำการ register กับ Protocol ได้มากกว่า 1 protocol ซึ่งทำได้โดยการส่ง Register message หลายๆ message

5.3.4 Preconditions

ในการที่ Protocol จะทำงานได้ตามขั้นตอนอย่างถูกต้องจำเป็นต้องมีการทำงานก่อนที่จะเริ่มต้นการประมวลผลดังนี้

1. Source ต้องรู้ว่ากฎของ destination เป็นอย่างไร และถ้ามี source ต้องมีความสามารถในการที่จะสร้าง message ให้สอดคล้องตามกฎของ destination
2. ถ้ามีความต้องการของ secure exchange of message ดังนั้น source และ destination ต้องมี Security Context

5.3.5 Completion Protocol

Completion Protocol ถูกใช้โดย แอปพลิเคชัน เพื่อทำการบอก coordinator ว่าจะพยายาม Commit หรือจะ Abort Atomic Transaction หลังจากที่ transaction นั้นทำงานจนเสร็จแล้วสถานะของมันจะถูกส่งกลับมาให้ยัง แอปพลิเคชัน

Initiator register ของ Protocol นี้ใช้ Protocol identifier ดังนี้:

<http://schemas.xmlsoap.org/ws/2004/10/wsat/Completion>



รูปที่ 5.5 ภาพรวมของ Completion Protocol

Coordinator accepts:

- Commit - หลังจากที่ได้รับการแจ้งมาทำให้ Coordinator รู้ว่า Participant มีความสามารถในการที่จะทำงานให้เสร็จ และมันควรพยายามที่จะ Commit Transaction นั้น
- Rollback - หลังจากที่ได้รับการแจ้งมา Coordinator รู้ว่า Participant ได้ทำการ terminate การประมวลผลของการทำงานใน แอปพลิเคชัน นั้น และมันควรที่จะทำการ Abort

Transaction

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Initiator accepts:

- Committed - หลังจากที่ได้รับการแจ้งมา Initiator รู้ว่า coordinator ถึงเวลาที่ต้องทำการ Commit transaction
- Aborted - หลังจากที่ได้รับการแจ้งมา Initiator รู้ว่า Coordinator ถึงเวลาที่ต้องทำการ abort transaction

5.3.6 ลักษณะของ Two-Phase Commit Protocol ใน WS-AtomicTransaction

Two-Phase Commit Protocol (2PC) เป็น Coordination Protocol ที่ทำการกำหนดว่า Participants หลายตัวจะทำงานร่วมกันตามข้อตกลงของผลลัพธ์ที่ใน Atomic Transaction ได้ อย่งไร 2PC Protocol แบ่งออกได้เป็น Volatile 2PC และ Durable 2PC

■ Volatile Two-Phase Commit Protocol

หลังจากที่ได้รับการแจ้ง Commit มาในส่วนของ Completion Protocol จากนั้น Root Coordinator จะทำการเตรียมพร้อมให้ทุกๆ participant ทำการ register สำหรับ Volatile 2PC Protocol ทุกๆ Participant ที่ถูก register เข้า Protocol (Volatile 2PC) นี้จะต้องทำการ respond ก่อนที่จะ Register เข้ากับ Durable 2PC

Volatile Recipient จะไม่มีการรับรองว่าจะได้รับการแจ้งเมื่อมีผลลัพธ์ของ Transaction Participant ที่ register กับ Volatile 2PC สามารถใช้ Protocol identifier ดังนี้ :

<http://schemas.xmlsoap.org/ws/2004/10/wsat/Volatile2PC>

■ Durable Two-Phase Commit Protocol

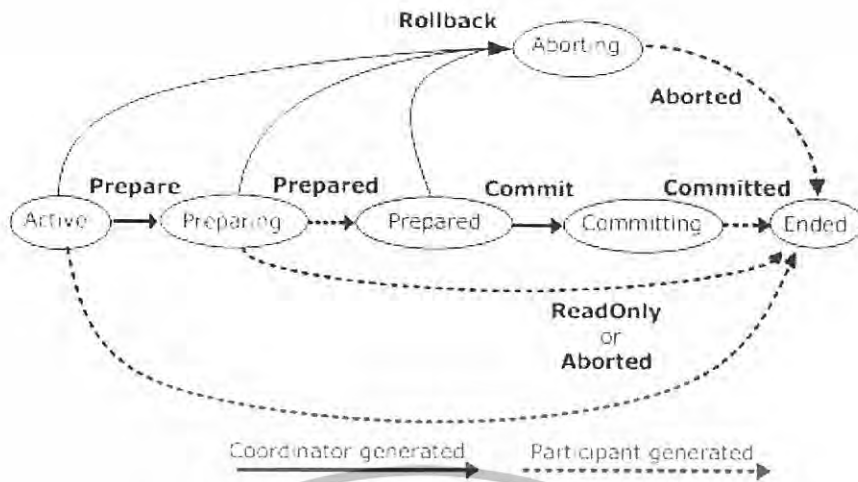
หลังจากที่ได้รับการแจ้ง Commit ในส่วนของ Completion Protocol และเสร็จสิ้นจากการเตรียมสถานะของ Volatile 2PC แล้ว Root Coordinator จะเริ่มทำการเตรียมสำหรับให้ Participant Register เข้าสู่ Durable 2PC Protocol โดยจะต้องมีการ respond ว่า Prepare หรือ ReadOnly ก่อนที่จะมีการแจ้ง Commit ให้กับ participant เพื่อให้มีการ register กับ protocol ตัวใดตัวหนึ่ง

Participant ที่ register กับ Durable 2PC สามารถใช้ Protocol identifier ดังนี้ :

<http://schemas.xmlsoap.org/ws/2004/10/wsat/Durable2PC>

■ 2PC Diagram and notifications

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6 ภาพรวมของการ Commit แบบสองระยะ

Participant accepts:

- Prepare

หลังจากที่ได้รับการแจ้งมา Participant จะรู้ว่าต้องเข้าสู่ Phase 1 และทำการ vote ให้กับผลลัพธ์ของ Transaction ถ้า Participant ไม่รู้เกี่ยวกับ Transaction มันต้องทำการ abort แต่ถ้า Participant นั้นได้ทำการ vote ไปแล้วมันควรจะทำการ resend ค่าที่เคย vote ให้

- Rollback

หลังจากที่ได้รับการแจ้งมา Participant จะรู้ว่าต้องทำการ abort และทำการทิ้ง Transaction นั้นไป การแจ้งนี้สามารถส่งได้ทั้ง phase 1 และ phase 2 เมื่อมีการส่งแล้ว coordinator จะไม่สนใจ และ clear ที่ทั้งหมดของ Transaction

- Commit

หลังจากที่ได้รับการแจ้งมา Participant จะรู้ว่าต้องทำการ commit Transaction โดยการแจ้งนี้จะสามารถส่งได้หลังจาก phase 1 และ Participant ทำการ vote commit ถ้า participant ไม่รู้เกี่ยวกับ Transaction มันต้องทำการส่งการแจ้งว่า committed ให้กับ Coordinator

Coordinator accepts:

- Prepared

หลังจากที่ได้รับการแจ้งมา Coordinator จะรู้ว่า participant ได้ถูกเตรียม และทำการ commit Transaction

- Read Only

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่ได้รับการแจ้งมา Coordinator จะรู้ว่า participant ได้ทำการ commit Transaction แล้ว และไม่สนใจใน Transaction อีก Participant ไม่ต้องการที่จะเข้ามามีส่วนร่วมใน phase 2

- Aborted

หลังจากที่ได้รับการแจ้งมา Coordinator จะรู้ว่า participant ได้ทำการ abort แล้ว และไม่สนใจใน Transaction อีก

- Committed

หลังจากที่ได้รับการแจ้งมา Coordinator จะรู้ว่า Participant ได้ทำการ committed transaction แล้วทำให้ Participant สามารถไม่ต้องสนใจกับ Transaction อีก

- Replay

หลังจากที่ได้รับการแจ้งมา Coordinator จะทำการสรุปได้ว่า Participant มีความผิดพลาดขึ้นซึ่งไม่สามารถทำการกู้คืนได้ Coordinator จึงควรที่จะ resend Protocol ที่เหมาะสมที่สุด

5.3.7 Transaction Faults

WS-AtomicTransaction fault ต้องมีการ include เอา [action] จาก URI:

<http://schemas.xmlsoap.org/ws/2004/10/wsati/fault>

Faults จะเน้นไปที่ destination endpoint ซึ่งเกี่ยวข้องกับกลไกของการรองรับการเกิด faults ซึ่งได้รับการระบุไว้ใน WS-Addressing

Faults ในส่วนนี้จะถูกระบุไว้ด้วยคุณสมบัติดังนี้

[Code] Fault code

[Subcode] Fault subcode

[Reason] เหตุผลของการเกิด fault (ภาษาอังกฤษ)

[Detail] รายละเอียดต่างๆ ถ้าว่างคือ ไม่มีรายละเอียดของความผิดพลาด

โดยสามารถนำเอาคุณสมบัติในตารางข้างต้นมาเชื่อมเป็น SOAP 1.2 ได้ดังนี้

<S:Envelope>

<S:Header>

<wsa:Action>

<http://schemas.xmlsoap.org/ws/2004/10/wsati/fault>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</wsa:Action>
<!-- Headers elided for clarity. -->
</S:Header>
<S:Body>
<S:Fault>
<S:Code>
<S:Value>[Code]</S:Value>
<S:Subcode>
<S:Value>[Subcode]</S:Value>
</S:Subcode>
</S:Code>
<S:Reason>
<S:Text xml:lang="en">[Reason]</S:Text>
</S:Reason>
<S:Detail>
[Detail]
...
</S:Detail>
</S:Fault>
</S:Body>
</S:Envelope>

```

InconsistentInternalState

Fault ประเภทนี้จะถูกส่งโดย Participant เพื่อทำการบอกว่าไม่สามารถทำตาม วัตถุประสงค์ซึ่งจะเป็นการบอกถึง Global consistency failure และเป็นสถานะที่ไม่สามารถ ทำการกู้คืนได้

[Code] Sender

[Subcode] wsat:InconsistentInternalState

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[Reason] A global consistency failure has occurred. This is an unrecoverable condition.

[Detail] unspecified

5.4 ความสามารถในการสนับสนุน WS-AtomicTransaction บน WebSphere

Application Server

WebSphere Application Server มีการนำ Web Services Atomic Transaction (WS-AT) มาใช้ โดย specification นี้จะทำให้ Web service สามารถที่จะมีส่วนร่วมใน global transaction ซึ่งถูกทำ ให้กระจายไปยังสภาพแวดล้อมของ Web Services ที่ต่างกัน

- Web Services Atomic Transaction สำหรับ WebSphere Application Server ให้บริการ เกี่ยวกับคุณภาพของบริการทางด้าน transaction บนสภาพแวดล้อมของ Web Services ซึ่ง จะทำให้ Web Services ที่อยู่บนระบบแบบ distributed และทรัพยากรที่ใช้กันสามารถเป็นส่วนหนึ่งของ global transaction
- ความสามารถในการใช้งาน WS-AT ที่สนับสนุนนั้นเป็นรูปแบบที่สามารถใช้งานข้าม ระบบได้โดยไม่ต้องมีการเขียนโปรแกรมติดต่อเพิ่มเติมโดยการกำหนดเขตของ Global transaction นั้นจะถูกจัดการ โดยมาตรฐานบน J2EE ซึ่งใช้ JTA UserTransaction โดยถ้ามีการเรียกใช้งาน Web Services จาก application ที่ทำงานอยู่ภายใต้ global transaction สภาพแวดล้อมของ WS-AT Coordinator จะทำการส่งคำร้องนั้นไปยัง Web Service เป้าหมาย

ถ้า WebSphere Application Server นั้นเป็นเจ้าของระบบที่รองรับการร้องขอที่อยู่ภายใต้ สภาพแวดล้อมของ WS-AT แล้ว WebSphere จะทำการเรียกใช้งานและสร้าง JTA transaction ใน ระบบปลายทางที่กำลังจะกลายเป็นสภาพแวดล้อมของ Transaction โดยอัตโนมัติ

5.4.1 การตั้งค่า Deployment descriptors สำหรับ atomic transactions

ในการเปิดใช้งาน WS-AT จะต้องมีการปรับปรุง deployment descriptors ทั้งในส่วน ของ Web และ EJB ดังนี้

- ในส่วนของเว็บที่เรียกใช้งาน Web Services นั้นจะต้องมีการกำหนดค่า Send Web Services Atomic Transaction on requests เพื่อทำการใช้งาน Transaction บน Web Service ปลายทาง
- ในส่วนของเว็บที่เป็น Web Service นั้นต้องมีการกำหนดค่า Execute using Web Services Atomic Transaction on incoming requests เพื่อให้ Web Service นั้นวิ่งอยู่บน สภาพแวดล้อมของ Transaction

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในส่วนของ EJB ที่เรียกใช้งาน Web Service นั้นกำหนดค่า Use Web Services Atomic Transaction เพื่อกำหนดให้ใช้งาน Atomic Transaction บน Service ปลายทาง
- ในส่วนของ EJB นั้นจำเป็นต้องมีการกำหนดค่าชนิดของ Transaction เป็น Required ซึ่งเป็นค่ามาตรฐานเพื่อการใช้งาน Atomic Transaction



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การออกแบบและพัฒนาระบบ Online Booking System

ระบบ Online Booking System เป็นระบบการจองออนไลน์ โดยมีเว็บเซอร์วิสหลักที่ให้บริการกับผู้ให้บริการ ซึ่งผู้ให้บริการสามารถใช้งานผ่านตัวเว็บแอปพลิเคชัน โดยผู้ให้บริการสามารถทำการจองตั๋วเครื่องบิน, โรงแรม และแพคเกจทัวร์ ได้ตามความต้องการของผู้ให้บริการ โดยระบบจะนำข้อมูลความต้องการที่ผู้ให้บริการใส่เข้ามาในเว็บแอปพลิเคชันไปใช้ในการค้นหาข้อมูลจากผู้ให้บริการเว็บเซอร์วิส และนำข้อมูลที่ได้มาประมวลผล เพื่อเป็นทางเลือกสำหรับผู้ให้บริการ

เมื่อผู้ใช้เลือกทำการใดๆก็ตาม เว็บแอปพลิเคชันจะทำการติดต่อไปยัง BPEL ซึ่งเป็นตัวกลางในการจัดการการเรียกใช้เว็บเซอร์วิสให้ตรงกับความต้องการของผู้ใช้ ว่าผู้ให้บริการต้องการจองอะไร หลังจากนั้นเว็บเซอร์วิสจะทำการจองเพื่อทำการสำรองไว้ให้กับผู้ให้บริการ หลังจากทำการจองแล้ว ผู้ให้บริการสามารถตรวจสอบรายละเอียด และสามารถเลื่อนการจองนั้นได้ โดยการทำงานของเว็บเซอร์วิสทั้งหมดจะอยู่ในสโคปของทรานแซกชัน

นอกเหนือจากนี้ระบบยังมีการนำเอามาตรฐานรักษาความปลอดภัยมาใช้ เพื่อป้องกันข้อมูลของผู้ให้บริการ เพื่อเป็นการสร้างความเชื่อใจให้กับผู้ให้บริการ

6.1 รายละเอียดของแอปพลิเคชัน

ระบบ Online Booking System ประกอบด้วยบริการต่างๆ ดังนี้

1. การลงทะเบียน

ผู้ให้บริการสามารถที่จะลงทะเบียนผ่านหน้าเว็บไซต์ได้ โดยจะต้องกรอกรายละเอียดต่างๆให้ครบทั้งหมด

2. การเข้าสู่ระบบของสมาชิก

สมาชิกสามารถเข้าสู่ระบบโดยใช้ username และ password

3. การค้นหาข้อมูลต่างๆ

- โรงแรม
- สายการบิน
- แพคเกจทัวร์

ระบบจะทำการค้นหาตามข้อมูลที่ผู้ให้บริการระบุมา โดยข้อมูลที่จะแสดงออกมานั้น จะถูกทำการประมวลผล เพื่อเป็นการกรองข้อมูลออกตามความเหมาะสมในแต่ละช่วงเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. การจอง

ในการจองใดๆ ผู้ใช้บริการจะต้องใส่ข้อมูลตามที่ได้ระบุไว้ โดยถ้ามีการจองตัวเครื่องบินเข้ามาเกี่ยวข้อง ผู้ใช้จะต้องใส่ชื่อผู้ที่ทำการเดินทางทั้งหมด รวมไปถึงจะต้องจ่ายเงินค่าตัวเครื่องบินล่วงหน้าด้วย โดยในการจองแต่ละครั้งผู้ใช้บริการจะได้รับรหัสการจอง

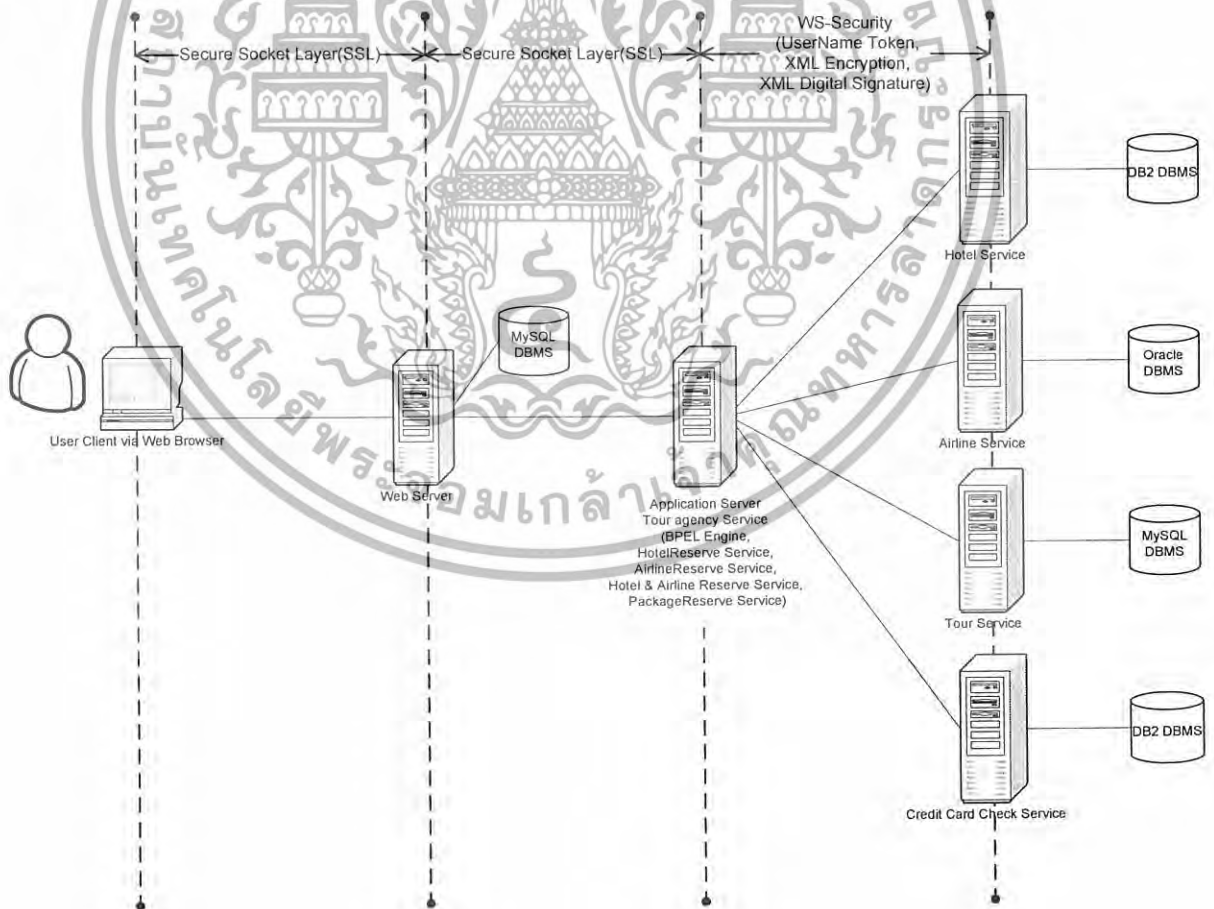
5. การเลื่อนการจอง

ผู้ใช้บริการสามารถเลื่อนการจองได้ผ่านทางเว็บไซต์ โดยเลือกรายการที่ต้องการเลื่อนการจอง

6. การยกเลิกการจอง

ผู้ใช้บริการสามารถยกเลิกการจองที่เกี่ยวข้องกับโรงแรมได้ผ่านทางเว็บไซต์ โดยเลือกรายการที่ต้องการเลื่อนการจอง

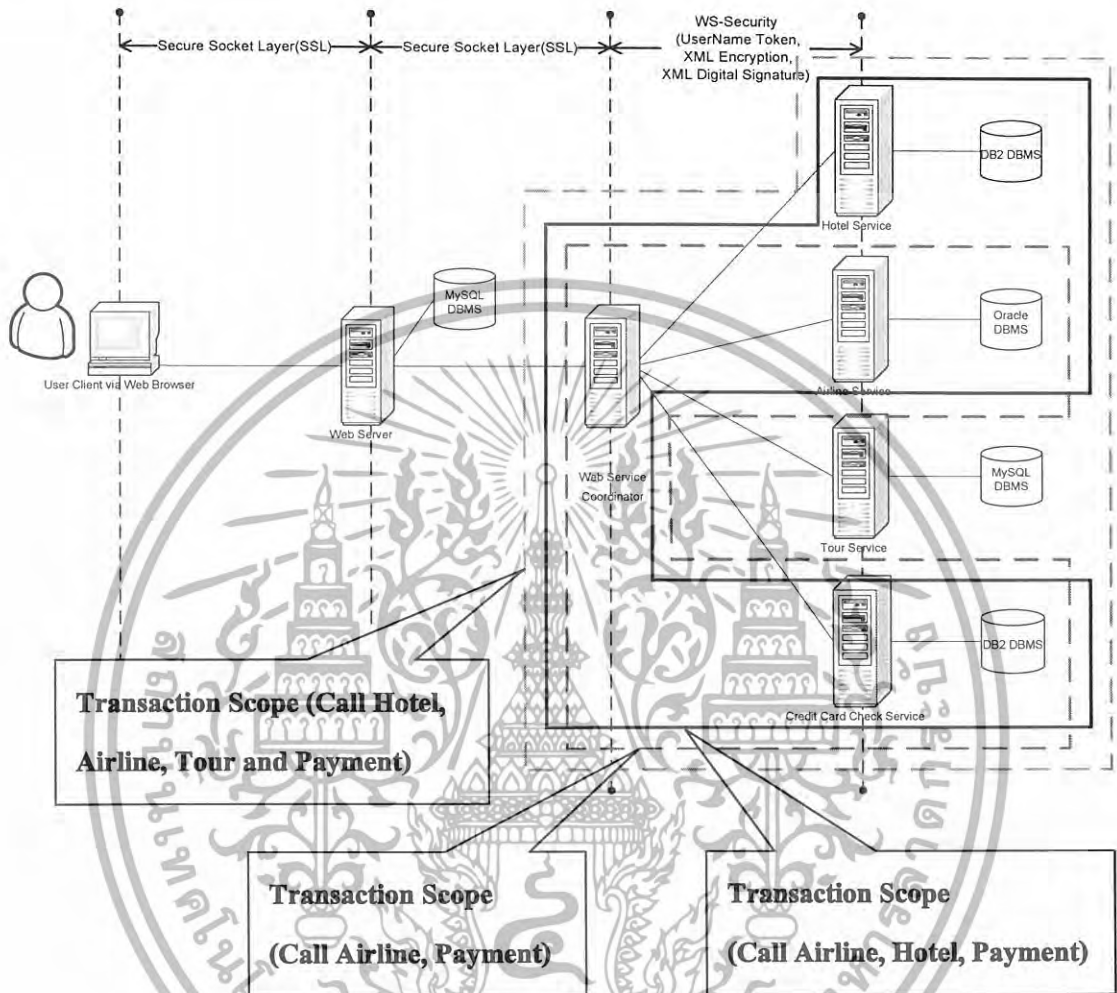
6.2 Application Model with Security Model



รูปที่ 6.1 Application Model with Security Model

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

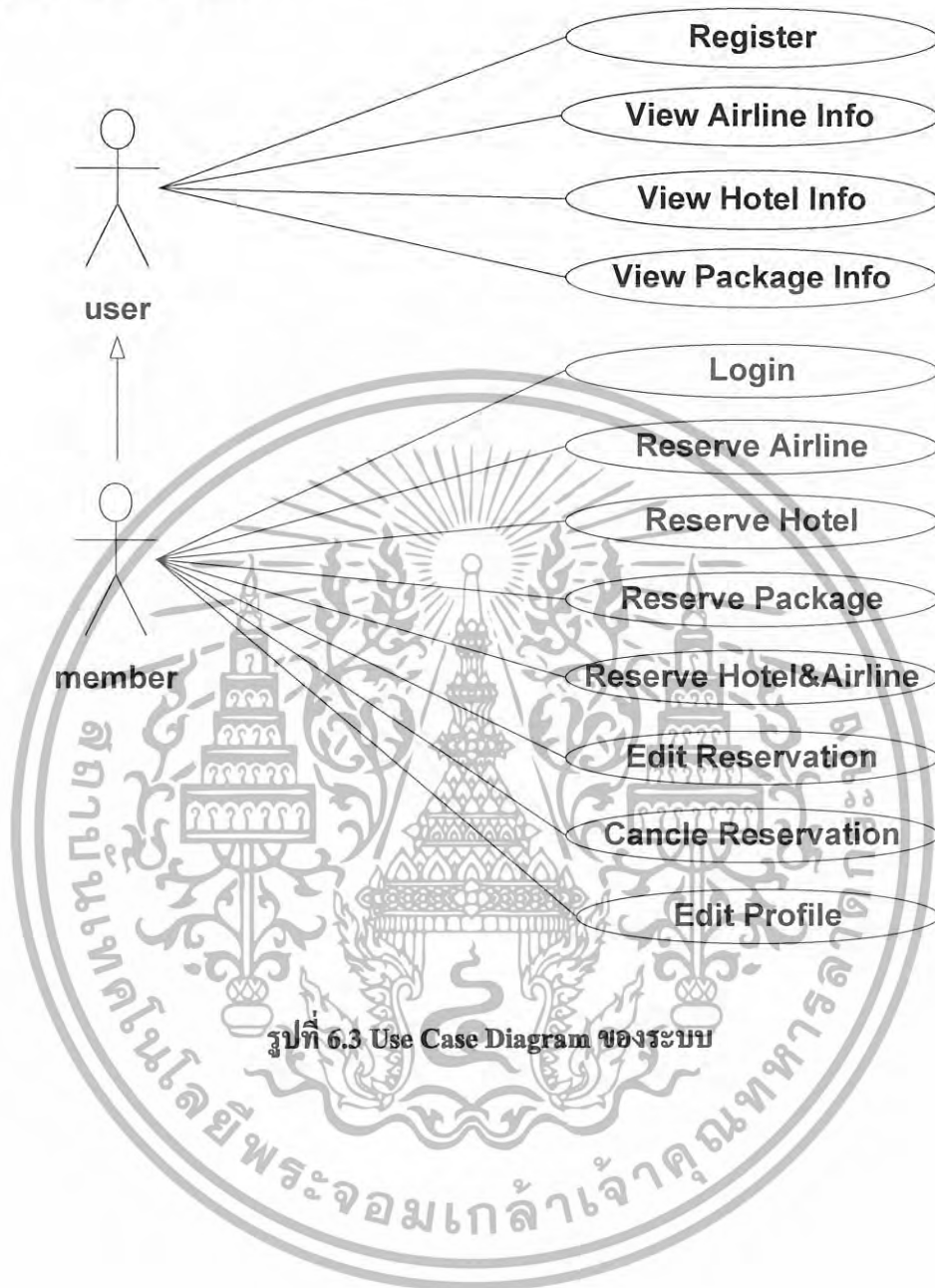
6.3 Application Model with Transaction Model



รูปที่ 6.2 Application Model with Transaction Model

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4 Use Case Diagram

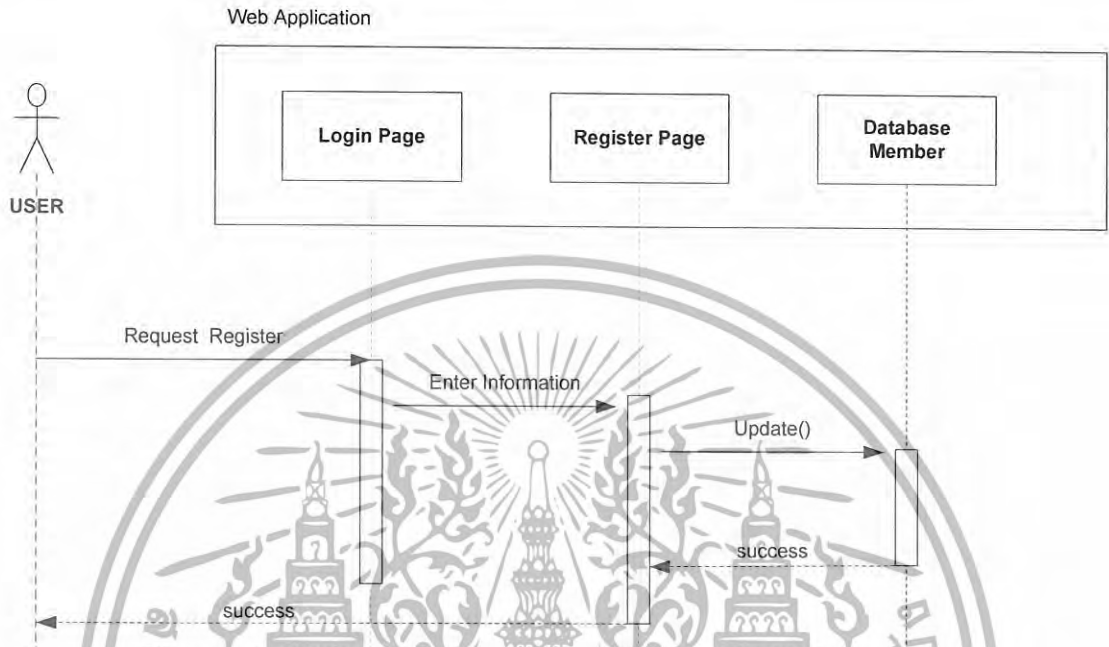


รูปที่ 6.3 Use Case Diagram ของระบบ

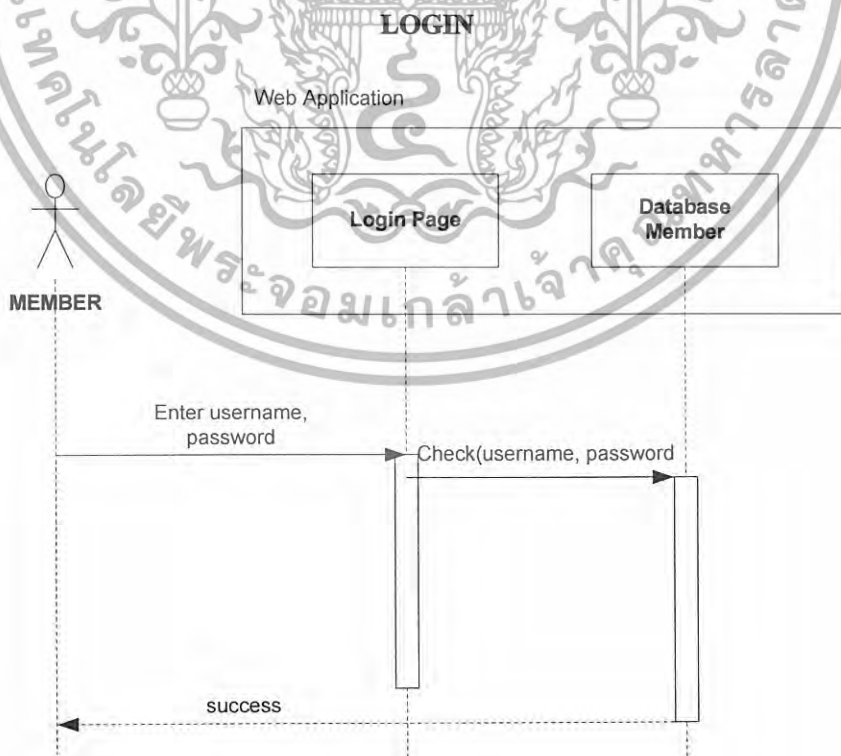
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.5 Sequence Diagram

REGISTER



รูปที่ 6.4 Sequence diagram (Register)



รูปที่ 6.5 Sequence diagram (Login)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรณียกเว้นกรณีพิเศษเท่านั้น เมื่อผู้ยัดให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RESERVE HOTEL



รูปที่ 6.6 Sequence diagram (Reserve Hotel)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RESERVE AIRLINE



รูปที่ 6.7 Sequence diagram (Reserve Airline)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการเชิงนิเวศเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

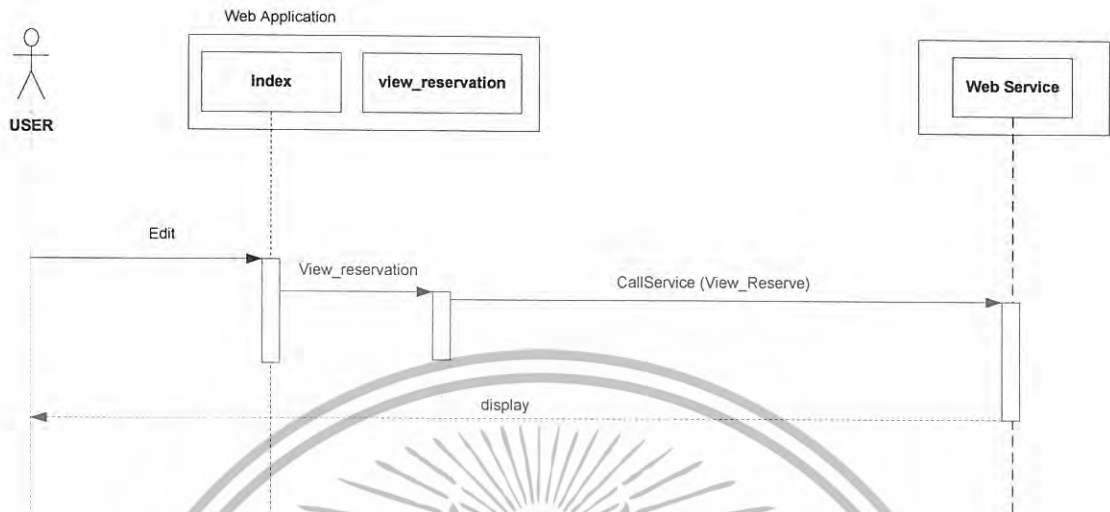
RESERVE PACKAGE



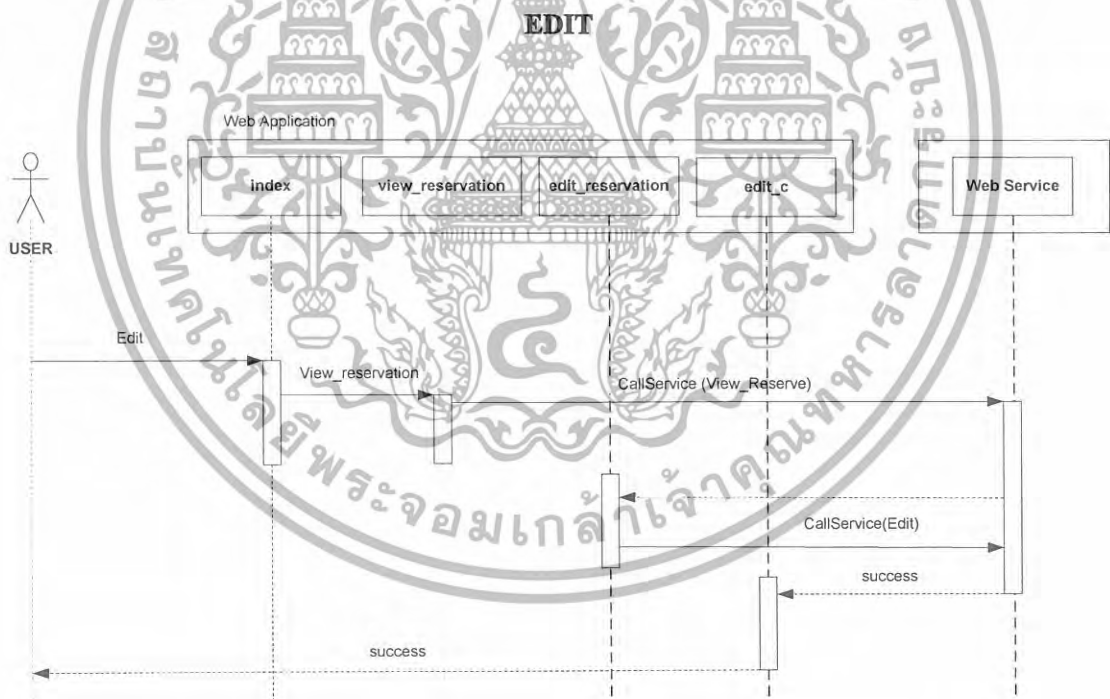
รูปที่ 6.8 Sequence diagram (Reserve Package)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VIEW RESERVATION



รูปที่ 6.9 Sequence diagram (View Reservation)



รูปที่ 6.10 แสดง Sequence diagram (Edit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.6 Activity diagram



รูปที่ 6.11 Activity diagram (Register)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.12 Activity diagram (Login)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RESERVE HOTEL



รูปที่ 6.14 Activity diagram (Reserve Hotel)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RESERVE AIRLINE



รูปที่ 6.15 Activity diagram (Reserve Airline)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RESERVE HOTEL_AIRLINE



รูปที่ 6.16 Activity diagram (Reserve Hotel_Airline)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

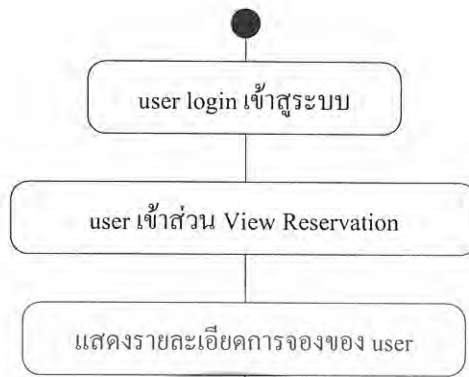
RESERVE PACKAGE TOUR



รูปที่ 6.17 Activity diagram (Reserve Package Tour)

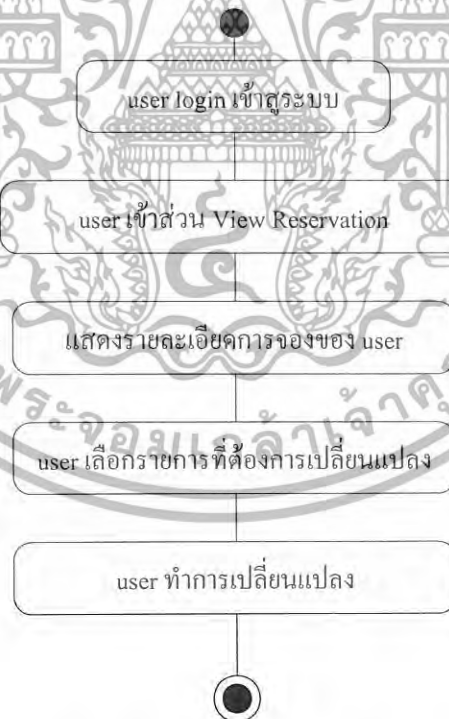
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VIEW RESERVATION



รูปที่ 6.18 Activity diagram (View Reservation)

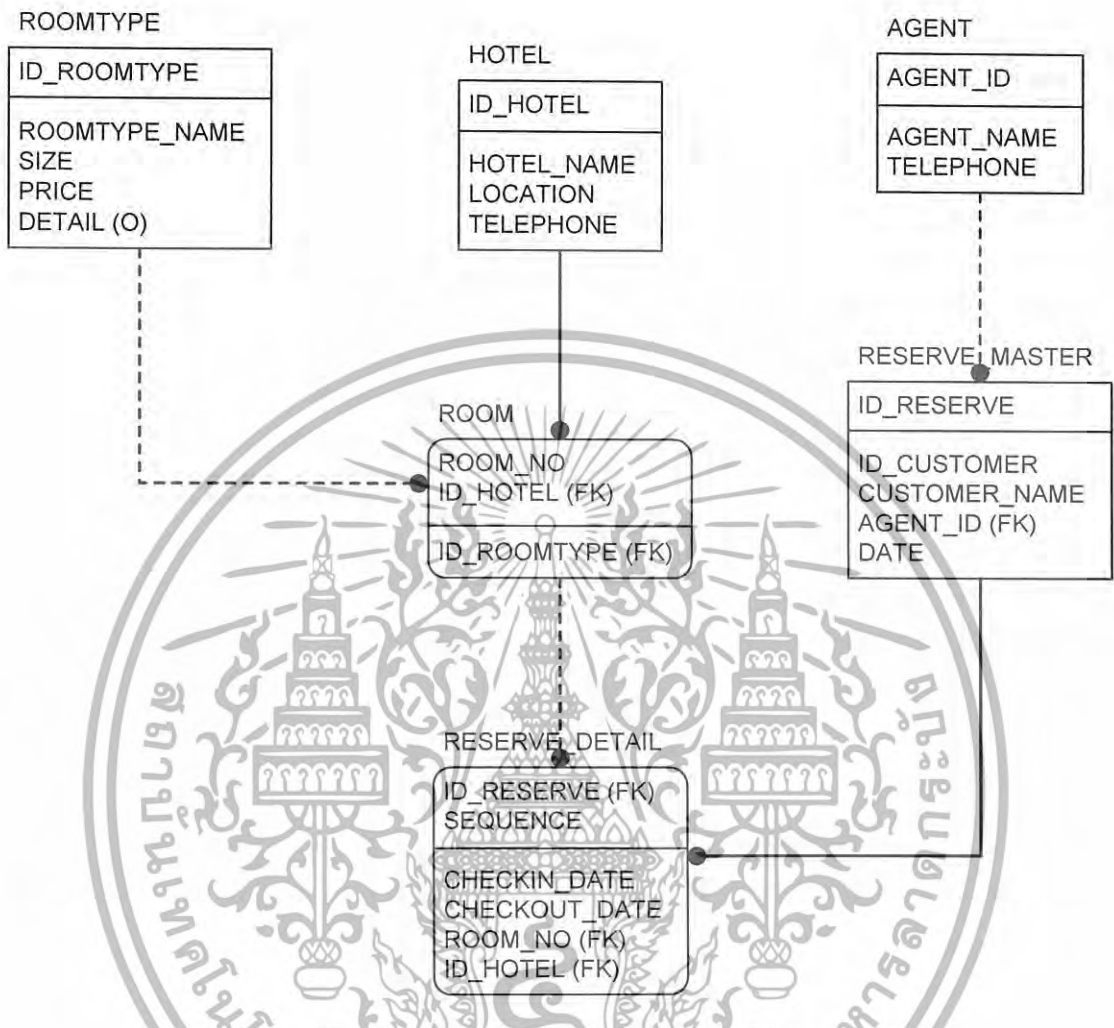
EDIT RESERVATION



รูปที่ 6.19 Activity diagram (Edit Reservation)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

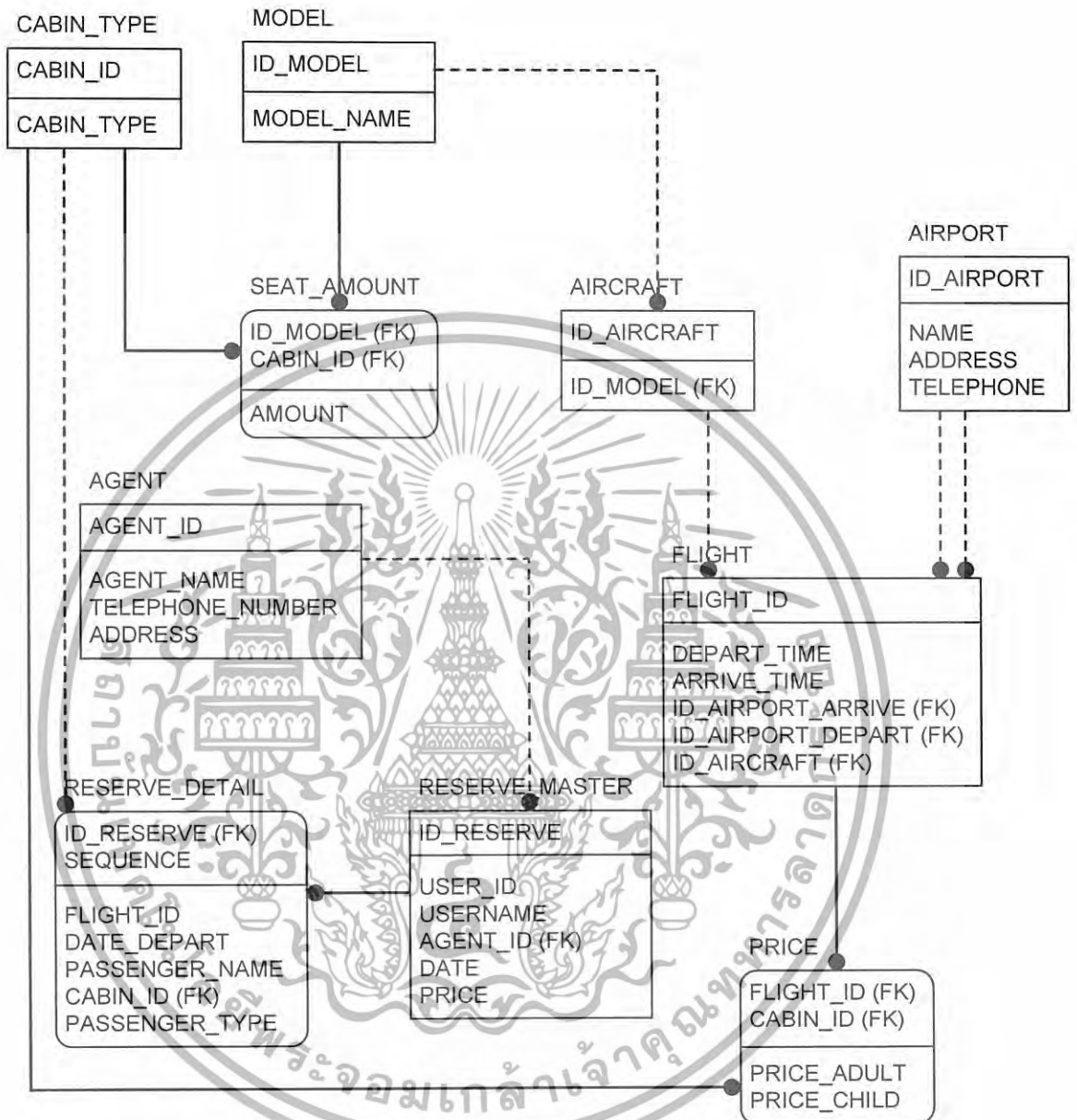
6.7 ความสัมพันธ์ของตารางในฐานข้อมูลของโรงแรม



รูปที่ 6.20 ความสัมพันธ์ของตารางในฐานข้อมูลของโรงแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

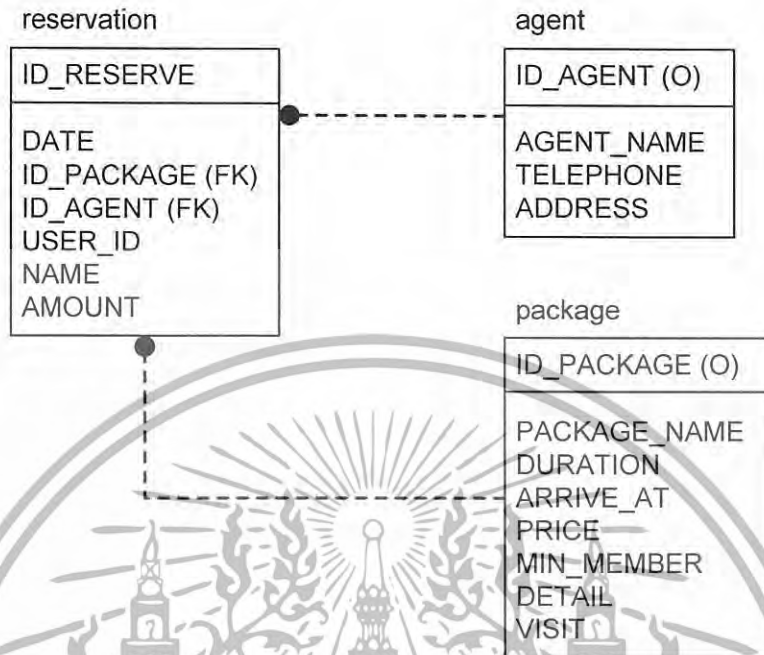
6.8 ความสัมพันธ์ของตารางในฐานข้อมูลของสายการบิน



รูปที่ 6.21 ความสัมพันธ์ของตารางในฐานข้อมูลของสายการบิน

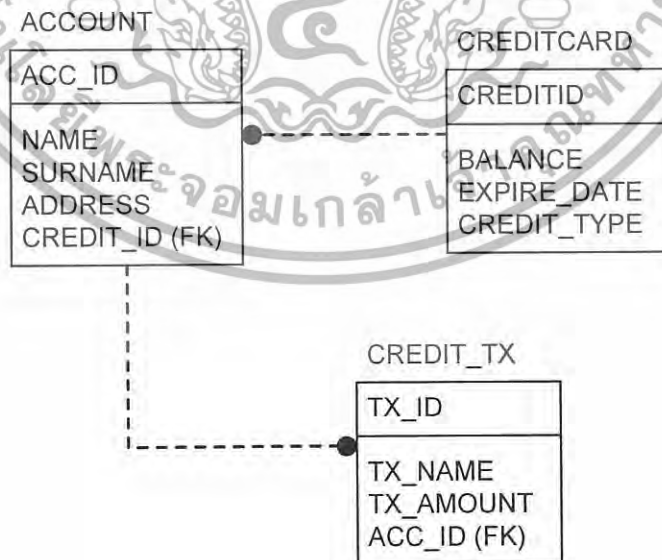
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.9 ความสัมพันธ์ของตารางในฐานข้อมูลของแพคเกจทัวร์



รูปที่ 6.22 ความสัมพันธ์ของตารางในฐานข้อมูลของแพคเกจทัวร์

6.10 ความสัมพันธ์ของตารางในฐานข้อมูลของธนาคาร



รูปที่ 6.23 ความสัมพันธ์ของตารางในฐานข้อมูลของธนาคาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.11 Class Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ **รูปที่ 6.20 Class diagram** เป็นอนุญาติให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

การทดลองและผลการทดลอง

7.1 เครื่องมือที่ใช้ในการทดลอง

รายละเอียด	Computer 1	Computer 2	Computer 3
หน่วยประมวลผล	Pentium M 1.86	Pentium 4 3.0	Pentium M 1.60
หน่วยความจำ	1.5 GB	1.0 GB	1.0 GB
ฮาร์ดดิสก์	80 GB	100 GB	40 GB

ตารางที่ 7.1 รายละเอียดของฮาร์ดแวร์ที่ใช้

รายละเอียด	Computer 1	Computer 2	Computer 3
ระบบปฏิบัติการ	Window XP SP2	Window XP SP2	Window XP SP2
ฐานข้อมูล	Oracle 10g	MySQL 4.1	DB2
เซิร์ฟเวอร์	Tomcat 5.5	WebSphere Application Server 5.0	WebSphere Application Server 5.0
ทูลที่ใช้พัฒนา	ActiveBPEL 2.1 (for create BPEL)	Rational Application Developer 6.0 (for create JSP web)	Rational Application Developer 6.0 (for create web service)

ตารางที่ 7.2 รายละเอียดของซอฟต์แวร์ที่ใช้

7.2 ฟังก์ชันการทำงานของเว็บเซอร์วิส

7.2.1 ฟังก์ชันการทำงานของเว็บเซอร์วิสตัวหลักของสายการบิน

Operations

Name	Documentation
getReserveAirline	--
getPriceAirline	--
viewInfoAirline	--
reserveAirline	--
editReserveAirline	--

รูปที่ 7.1 ฟังก์ชันที่มีให้เลือกใช้ในเว็บเซอร์วิสตัวหลักของสายการบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.2 ฟังก์ชันการทำงานของเว็บเซอวีส์ตัวหลักของโรงแรม

Operations

Name	Documentation
editReserveHotel	--
cancelReserveHotel	--
reserveHotel	--
getReserveHotel	--
viewInfoHotel	--

รูปที่ 7.2 ฟังก์ชันที่มีให้เลือกใช้ในเว็บเซอวีส์ตัวหลักของโรงแรม

7.2.3 ฟังก์ชันการทำงานของเว็บเซอวีส์ตัวหลักของแพคเกจทัวร์

Operations

Name	Documentation
editReservePackage	--
getReservePackage	--
reservePackage	--
viewInfoPackage	--

รูปที่ 7.3 ฟังก์ชันที่มีให้เลือกใช้ในเว็บเซอวีส์ตัวหลักของแพคเกจทัวร์

7.3 การใช้งานแอปพลิเคชัน

7.3.1 การสมัครสมาชิก

ผู้ใช้งานสามารถสมัครเข้าเป็นสมาชิกของทาง Web site โดยกรอกรายละเอียดต่างๆ ซึ่งผู้ที่
เป็นสมาชิกจะสามารถทำการจองโรงแรม, สายการบิน หรือแพคเกจทัวร์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Registration

Username :
 Password :
 Re-Password :
 Name :
 Surname :
 E-mail :
 Gender : Male Female
 Birthday (dd/mm/yyyy): 01 / 01 / 1907
 Age : 15
 Address :
 Province :
 Zipcode :
 Telephone :
 Mobile :
 Question : What's your mother name?
 Answer :

Register

รูปที่ 7.4 การสมัครสมาชิก

7.3.2 การเลือกจองประเภทต่างๆจากหน้า index

ผู้ใช้สามารถเลือกการจองประเภทต่าง ได้ผ่านทางหน้า index โดยมีให้เลือกด้วยกัน 4 ประเภท คือ การจองโรงแรม, การจองตั๋วเครื่องบิน, การจองโรงแรมและตั๋วเครื่องบิน และการจองแพคเกจทัวร์



LOGIN:
 Username
 Password
 Login
 ลืม Password
 ลืมชื่อผู้ใช้งาน

ข้อมูลท่องเที่ยว:
 ภาคเหนือ
 ภาคกลาง
 ภาคอีสาน
 ภาคใต้
 ภาคตะวันออก

Hotel Reservation
 จองโรงแรม

Airline Reservation
 จองตั๋วเครื่องบิน

Hotel & **A**irline Reservation
 จองโรงแรมและตั๋วเครื่องบิน

Package Reservation
 จองแพคเกจ

Copyright © 2012. All Rights Reserved. | E-mail: agenttravel@agenttravel.com | Contact Us: 08-1000-10000 | 08-1000-10000

รูปที่ 7.5 หน้า index

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.3.3 การค้นหาและจองโรงแรม

ผู้ใช้งานระบบสามารถทำการค้นหาโรงแรมที่ต้องการ โดยระบุจังหวัดที่จะเดินทางไปพัก และวันที่เข้าพัก-วันที่ออก รวมไปถึงจำนวนห้องพักที่ต้องการ หลังจากกดค้นหาโรงแรมแล้ว จะขึ้นรายชื่อของ โรงแรมทั้งหมดที่สัมพันธ์กับข้อมูลที่ใส่ไป หลังจากนั้นเลือกรายชื่อของ โรงแรมที่ต้องการ



รูปที่ 7.6 หน้าค้นหาโรงแรม

รูปที่ 7.7 รายชื่อโรงแรมที่ได้จากการค้นหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ใช้เลือกชนิดของห้องที่ต้องการ หลังจากกด Select แล้วจะขึ้นหน้าการคำนวณราคาของ
โรงแรมทั้งหมดที่จอง หลังจากผู้ใช้กด Reserve แล้วถ้าการจองสำเร็จจะขึ้นรหัสการจองให้กับ
ผู้ใช้



รูปที่ 7.8 หน้าที่ทำให้ผู้ใช้บริการเลือกชนิดของห้องพัก

รูปที่ 7.9 หน้าการคิดราคารวมทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 7.10 หน้าแสดงรหัสการจอง

7.3.4 การค้นหาและจองสายการบิน

ผู้ใช้งานระบบสามารถทำการค้นหาสายการบินที่ต้องการ โดยสามารถระบุได้ว่าต้องการเลือกดูทั้งขาไป-ขากลับ หรือเฉพาะขาไปเพียงอย่างเดียว โดยระบุจังหวัดต้นทาง-ปลายทาง รวมไปถึงวันที่เดินทางและจำนวนผู้โดยสาร หลังจากนั้นจะปรากฏหน้าสำหรับให้ผู้ใช้เลือกเที่ยวบินที่ต้องการ

รูปที่ 7.11 หน้าค้นหาเที่ยวบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Logout
 ยินดีลงชื่อ
 คุณ Sawarat
 Arrommatana
 Logout

Edit Profile
Edit Reservation

ข้อมูลท่องเที่ยว
 ภาคเหนือ
 ภาคกลาง
 ภาคอีสาน
 ภาคใต้
 ภาคตะวันออก

ค้นหาเที่ยวบิน

ออกเดินทางจาก : กรุงเทพฯ เดินทางไปยัง : เชียงใหม่
 วันที่ออกเดินทาง : 02-03-2007 วันที่เดินทางกลับ : 04-03-2007
 ผู้ใหญ่ : 2 เด็ก : 1

Outbound

- เที่ยวบินที่ : TG 102
 เวลาออกเดินทาง : 8.15
 เวลาถึงจอด : 9.25
 สนามบินต้นทาง : Suvarnabhumi International Airport (BKK)
 สนามบินปลายทาง : Chiang Mai International (CNX)
- เที่ยวบินที่ : TG 106
 เวลาออกเดินทาง : 10.50
 เวลาถึงจอด : 11.40
 สนามบินต้นทาง : Suvarnabhumi International Airport (BKK)
 สนามบินปลายทาง : Chiang Mai International (CNX)
- เที่ยวบินที่ : TG 112
 เวลาออกเดินทาง : 13.15
 เวลาถึงจอด : 14.25
 สนามบินต้นทาง : Suvarnabhumi International Airport (BKK)
 สนามบินปลายทาง : Chiang Mai International (CNX)
- เที่ยวบินที่ : TG 118
 เวลาออกเดินทาง : 16.00
 เวลาถึงจอด : 17.10
 สนามบินต้นทาง : Suvarnabhumi International Airport (BKK)
 สนามบินปลายทาง : Chiang Mai International (CNX)
- เที่ยวบินที่ : TG 641
 เวลาออกเดินทาง : 17.15
 เวลาถึงจอด : 18.25
 สนามบินต้นทาง : Suvarnabhumi International Airport (BKK)
 สนามบินปลายทาง : Chiang Mai International (CNX)

Select

รูปที่ 7.12 รายละเอียดของสายการบินที่ได้จากการค้นหา

เมื่อผู้ใช้งานกดเลือกสายการบินที่ต้องการแล้ว จะปรากฏหน้าต่างแสดงประเภทที่นั่งของสายการบินนั้น ผู้ใช้เลือกชนิดของเก้าอี้ที่นั่งแล้วกด Select หลังจากนั้นจะปรากฏหน้าต่างสรุปค่าเดินทางทั้งหมด เมื่อผู้ใช้งานกด Reserve

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Logout **ค้นหาเที่ยวบิน**

เข้าสู่ระบบ
คุณ Sawarat
Arromratana
[Logout](#)

ออกเดินทางจาก : กรุงเทพฯ
วันที่ออกเดินทาง : 02 03 2007
ผู้ใหญ่ : 2

เดินทางไปยัง : เชียงใหม่
วันที่เดินทางกลับ : 04 03 2007
เด็ก : 1

Edit Profile **Edit Reservation**

Outbound เที่ยวบินที่ : TG 102
ออกเดินทางจาก Suvarnabhumi International Airport (BKK)
เวลาเครื่องขึ้น 8:15

เดินทางไปยัง Chiang Mai International (CNX)
เวลาเครื่องลง 9:25

ข้อมูลท่องเที่ยว	ที่นั่งที่ 1	ผู้โดยสาร	ประเภทที่นั่ง	ราคา
ภาคเหนือ	<input type="radio"/>	ผู้ใหญ่	first	4975
ภาคกลาง	<input type="radio"/>	ผู้ใหญ่	business	4075
ภาคอีสาน	<input type="radio"/>	ผู้ใหญ่	economy	3075
ภาคใต้	<input type="radio"/>	ผู้ใหญ่	first	4975
ภาคตะวันออก	<input type="radio"/>	ผู้ใหญ่	business	4075
	<input type="radio"/>	ผู้ใหญ่	economy	3075
	<input type="radio"/>	ผู้ใหญ่	first	3190
	<input type="radio"/>	เด็ก	business	2390
	<input type="radio"/>	เด็ก	economy	1790

[Select](#)

กรุณาอย่าลืมกดปุ่ม "เลือกประเภทที่นั่ง" เพื่อเลือกประเภทที่นั่งที่ท่านต้องการ
กรุณาอย่าลืมกดปุ่ม "เลือกประเภทที่นั่ง" เพื่อเลือกประเภทที่นั่งที่ท่านต้องการ

รูปที่ 7.13 หน้าให้ผู้ให้บริการเลือกประเภทของที่นั่ง

Logout **ค้นหาเที่ยวบิน**

เข้าสู่ระบบ
คุณ Sawarat
Arromratana
[Logout](#)

ออกเดินทางจาก : กรุงเทพฯ
วันที่ออกเดินทาง : 02 03 2007
ผู้ใหญ่ : 2

เดินทางไปยัง : เชียงใหม่
วันที่เดินทางกลับ : 04 03 2007
เด็ก : 1

Edit Profile **Edit Reservation**

Outbound เที่ยวบินที่ : TG 102
ออกเดินทางจาก Suvarnabhumi International Airport (BKK)
เวลาเครื่องขึ้น 8:15

เดินทางไปยัง Chiang Mai International (CNX)
เวลาเครื่องลง 9:25

ข้อมูลท่องเที่ยว	ที่นั่งที่	ผู้โดยสาร	ประเภทที่นั่ง	ราคา
ภาคเหนือ	1	ผู้ใหญ่	economy	3075
ภาคกลาง	2	ผู้ใหญ่	economy	3075
ภาคอีสาน	3	เด็ก	economy	1790
ภาคใต้	รวมเป็นเงิน			7940 บาท

รวมเป็นเงินทั้งสิ้น 7940 บาท

[Reserve](#)

กรุณาอย่าลืมกดปุ่ม "เลือกประเภทที่นั่ง" เพื่อเลือกประเภทที่นั่งที่ท่านต้องการ
กรุณาอย่าลืมกดปุ่ม "เลือกประเภทที่นั่ง" เพื่อเลือกประเภทที่นั่งที่ท่านต้องการ

รูปที่ 7.14 หน้าการคิดราคารวมทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากนั้น จะปรากฏหน้าให้ผู้ใช้ใส่รายชื่อของผู้เดินทางทั้งหมด และรายละเอียดของข้อมูลบัตรเครดิต หลังจากผู้ใช้ใส่รายละเอียดทั้งหมดให้กด Reserve ระบบจะทำการจองตั๋วเครื่องบินให้ ถ้าการจองสำเร็จจะแสดงรหัสการจองให้กับผู้ใช้



รูปที่ 7.16 หน้าแสดงรหัสการจอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

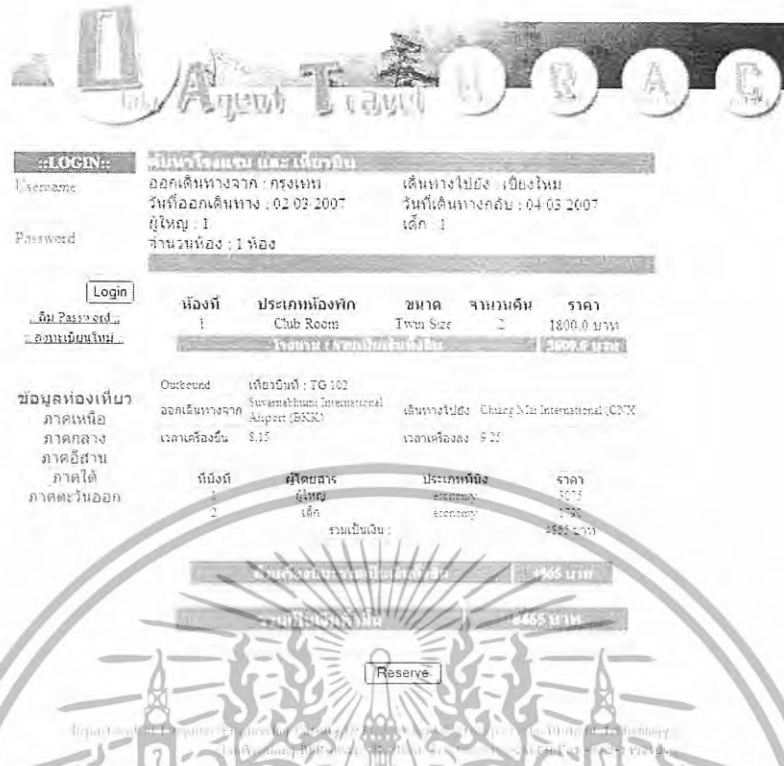
7.3.5 การค้นหาและจองทั้งสายการบินและโรงแรม

ผู้ใช้งานระบบสามารถทำการค้นหาสายการบินและ โรงแรมที่ต้องการ และทำการจองได้ โดยสามารถระบุได้ว่าต้องการเลือกตั๋วขาไป-จากกลับ หรือเฉพาะขาไปเพียงอย่างเดียว โดยระบบจังหวัดต้นทาง-ปลายทาง รวมไปถึงวันที่เดินทางและจำนวนผู้เดินทาง รวมไปถึงระบุจำนวนห้อง หลังจากนั้นทำการเลือก โรงแรมและสายการบินที่ต้องการซึ่งทำเช่นเดียวกับการจองเครื่องบินและโรงแรม โดยถ้าการจองทั้งหมดสำเร็จจะแสดงรหัสการจองให้กับผู้ใช้



รูปที่ 7.17 การค้นหาสายการบินและโรงแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

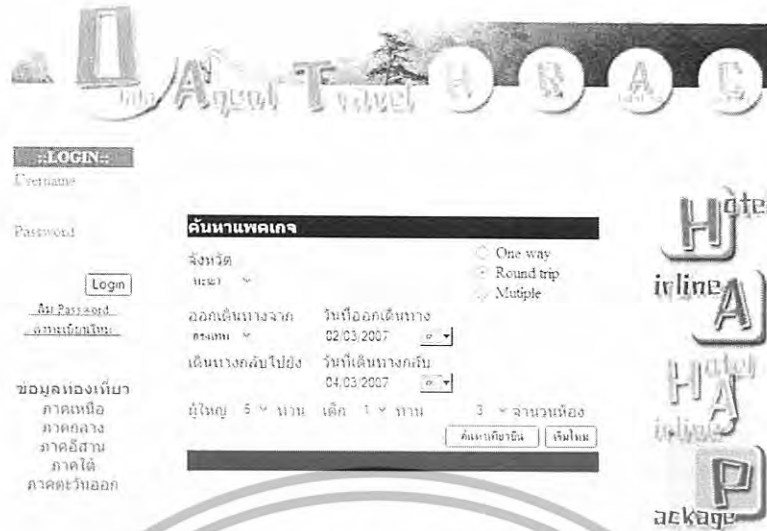


รูปที่ 7.18 หน้าราคา รวมทั้งหมดของทั้งการจองเครื่องบินและโรงแรม

7.3.6 การค้นหาและจองแพคเกจทัวร์

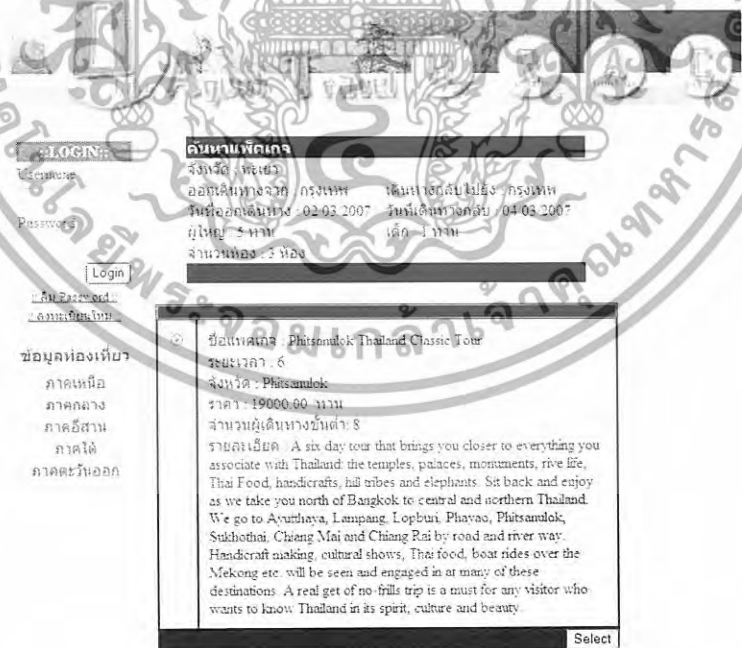
ในการค้นหาและจองแพคเกจทัวร์ ผู้ใช้ใส่รายละเอียดต่างๆ โดยจะมีการจองโรงแรม และตัวเครื่องบินควบคู่ไปด้วย ซึ่งในการจองเครื่องบินสามารถเลือกได้ทั้งแบบ One way คือ การจองตัวเครื่องบินขาไปเพียงอย่างเดียว, Round trip คือการจองตัวเครื่องบินทั้งขาไปและกลับ ส่วน multiple คือการจองตัวแบบระบุปลายทางที่จะเดินทางกลับได้ หลังจากนั้นจะปรากฏหน้าแสดงรายละเอียดของแต่ละแพคเกจที่เกี่ยวข้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.19 หน้าให้ผู้ใส่รายละเอียดต่างๆสำหรับการจองแพคเกจ

ผู้ซื้อเลือกชุดแพคเกจทั่วสัปดาห์ที่ต้องการหลังจากนั้นกด Select เพื่อทำการเลือกโรงแรมและเที่ยวบินที่ต้องการ โดยถ้าทำการจองสำเร็จทั้งหมดจะแสดงรหัสการจองให้กับผู้ซื้อ



รูปที่ 7.20 หน้ารายละเอียดของแพคเกจทั่วสัปดาห์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.3.7 การดูรายละเอียดและแก้ไขรายละเอียดการจอง

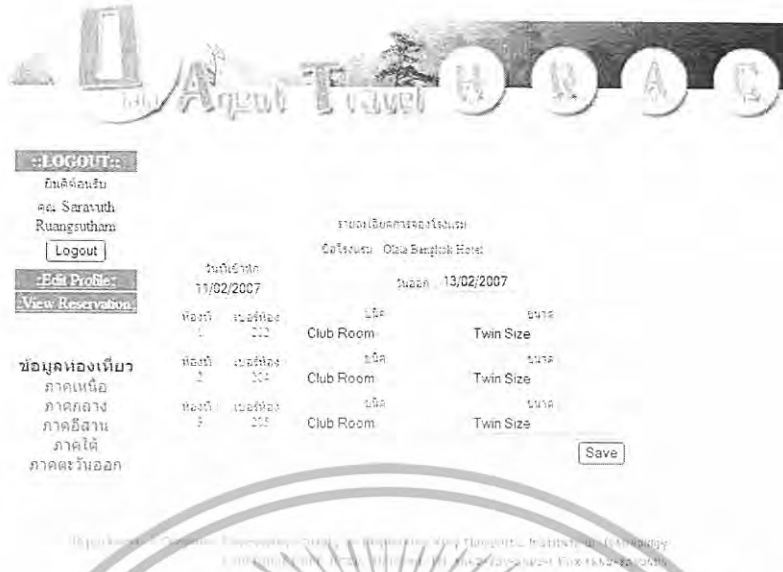
ผู้ใช้สามารถดูรายละเอียดและทำการแก้ไขรายละเอียดการจองได้ โดยแก้ไขได้เพียงเฉพาะวันที่ โดยผู้ใช้ต้องทำการ log in เข้าสู่ระบบ หลังจากนั้นเลือกที่ View Reservation เลือกที่รายการที่ต้องการแก้ไข หลังจากนั้นทำการแก้ไขให้เป็นวันที่ที่ต้องการ จากนั้นผู้ใช้กด save เพื่อบันทึกข้อมูลลงฐานข้อมูล ถ้าการเปลี่ยนแปลงสำเร็จจะขึ้นข้อความว่าการแก้ไขการจองสำเร็จ



รูปที่ 7.21 หน้ารายละเอียดการจองของผู้ใช้งาน

รูปที่ 7.22 หน้ารายละเอียดการจองของรายการที่ได้เลือกไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.23 หน้าสำหรับให้ผู้ใช้เปลี่ยนแปลงรายละเอียดการจอง

7.4 ผลการทดลองที่ได้จากการเรียกใช้เว็บเซอร์วิสที่มีการทำทรานแซกชัน

ในการทำการจองแพคเกจทัวร์, การจองตั๋วเครื่องบิน หรือการจองทั้งตัวเครื่องบินและโรงแรม จะมีการนำแทรกแซกชันเข้ามาใช้งาน กล่าวคือถ้าผู้ใช้ต้องการจองตั๋วเครื่องบิน แต่ในขณะการจองระบบไม่สามารถจองเครื่องบิน หรือโรงแรม หรือแม้แต่กระทั่งไม่สามารถชำระเงินค่าเครื่องบินได้ การจองนั้นจะถือว่าไม่สำเร็จ โดยตัวโคออดิเนเตอร์จะสั่งให้มีการ roll back เกิดขึ้น ซึ่งจะส่งผลกลับมาให้ผู้ใช้ว่าไม่สามารถทำการจองได้ แต่ถ้าการจองสำเร็จจะแสดงรหัสการจองให้กับผู้ใช้



รูปที่ 7.24 หน้าสุดท้ายของการจองตั๋วเครื่องบิน (ทำการจองไม่สำเร็จ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการเชิงงานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยภายใน SOAP Message ในกรณีที่ไม่สามารถทำการจองได้ จะมีส่วนที่บอกว่าเกิดการ roll back โดยมีการระบุปัญหาว่าเกิดขึ้นในส่วนใด โดยจากภาพด้านล่างเป็นการทดลองการจองตั๋วเครื่องบิน แต่เกิดปัญหาในขั้นตอนการชำระเงิน ทำให้เกิดการ roll back การทำงานทั้งหมด

```

Documents and Settings\War\Desktop\new work soap and report\rollback_response.xml
h Document Project Tools Window Help
B I U F
<soapenv:Envelope xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<soapenv:Header/>
<soapenv:Body>
<soapenv:Fault>
<faultcode>soapenv:Server.generalException</faultcode>
<faultstring><![CDATA[javax.transaction.TransactionRolledbackException: ; nested exception is:
java.rmi.RemoteException: Exception:java.rmi.RemoteException: Credit card invalid]]></faultstring>
</soapenv:Fault>
</soapenv:Body>
</soapenv:Envelope>

```

รูปที่ 7.25 SOAP Message ที่มีการระบุว่ามี การ roll back เกิดขึ้น

7.5 ผลการทดลองที่ได้จากการทำ WS-Security

จากทฤษฎีเกี่ยวกับเว็บเซอร์วิสที่ผ่านมาได้ทำการทดลองใช้งานในส่วนของ WS-Security แสดงให้เห็นใน SOAP ได้ดังตัวอย่าง

```

EditPlus - [Noname1]
File Edit View Search Document Project Tools Window Help
Directory Cliptext
Request
<?xml version="1.0" encoding="UTF-8" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Body>
<?echo>
<>Hello world!</>
</?echo>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

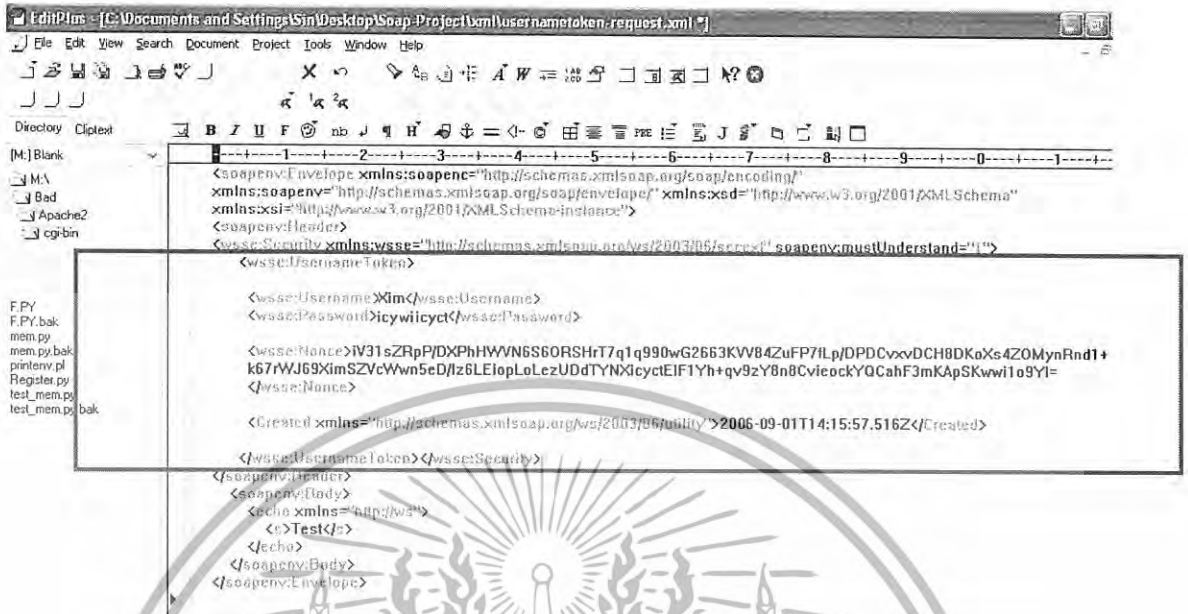
Response
<soapenv:Envelope xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<soapenv:Header />
<soapenv:Body>
<p694:echoResponse xmlns:p694="http://tut4">
<echoResult>Hello world!</echoResult>
</p694:echoResponse>
</soapenv:Body>
</soapenv:Envelope>

```

รูปที่ 7.26 SOAP Message ก่อนการทำ WS-security

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.5.1 การเพิ่มความปลอดภัยในการพิสูจน์ตัวตน โดยใช้ Username Token



รูปที่ 7.27 SOAP ที่มีการทำ Username Token

ใน SOAP ที่มีการทำ Username Token แล้วจะมี <wsse:UsernameToken> แสดงถึงการ ใช้ Username Token โดยภายในจะประกอบด้วยอิลิเมนต์ต่างๆ ดังนี้

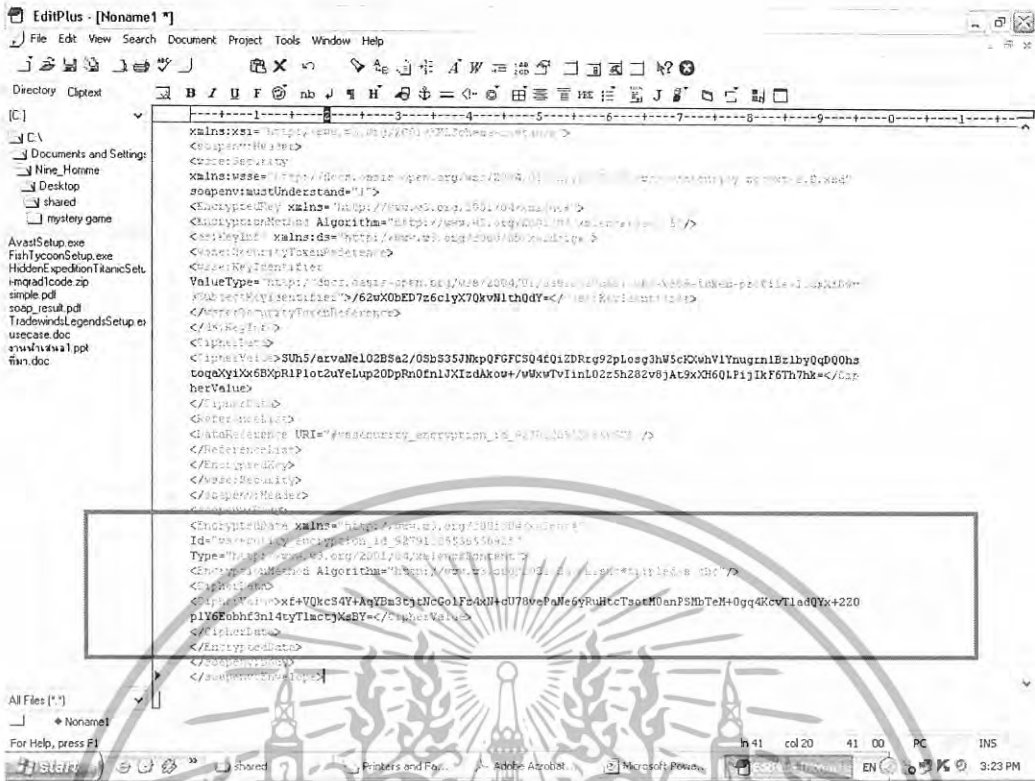
- <wsse:Username> แสดงถึงยูสเซอร์นามที่ใช้
- <wsse:Password> แสดงถึงรหัสผ่านของยูสเซอร์ ซึ่งภายในจะมีแอทริบิวต์ PasswordType ที่ใช้ระบุว่าใช้รหัสผ่านเป็นแบบใด
- <wsse:Nonce> เป็นค่าที่ไว้ป้องกันการโจมตีเมสเสจ (replay attack)
- <wsu:Created> เป็นค่าที่บอกว่าเมสเสจนี้ถูกสร้างขึ้นเมื่อไร (timestamp)

7.5.2 การรักษาความลับของข้อมูล โดยการใช้ XML Encryption

จากภาพด้านล่าง อิลิเมนต์ของ XML ที่ได้หลังการเข้ารหัส คือ <EncryptedData> ซึ่งอิลิเมนต์ย่อยอีก 3 อิลิเมนต์ คือ

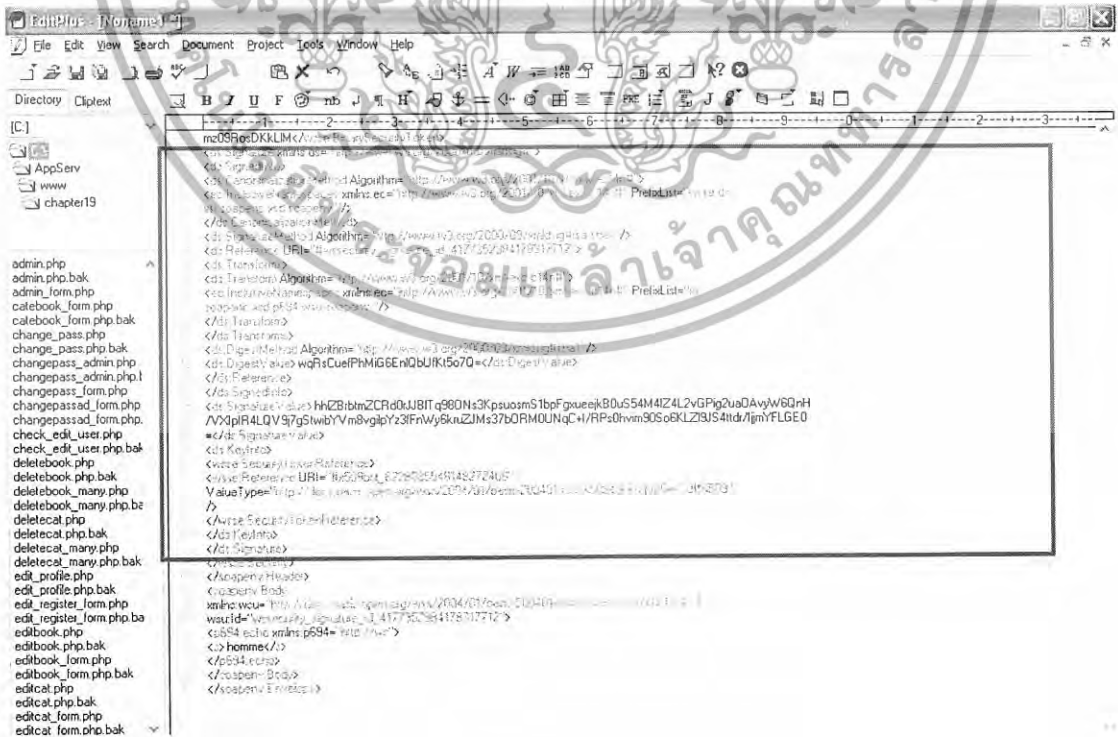
- <EncryptionMethod> จะบอกถึงอัลกอริทึมในการเข้ารหัสข้อมูล ในกรณีนี้อัลกอริทึม ชนิด TripleDES จะถูกเลือก และกุญแจที่เหมือนกันจะถูกใช้ในการเข้ารหัส และถอดรหัส
- <CipherData> เป็นอิลิเมนต์ย่อยที่จะมี <CipherValue> เป็นอิลิเมนต์ย่อยอีกทีที่จะเป็นการเข้ารหัสข้อมูลที่แท้จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.28 SOAP มีการทำ XML Encryption

7.5.3 การรักษาความสมบูรณ์ของข้อมูล โดยการใช้ Digital Signature



เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ 7.29 SOAP มีการทำ XML Digital Signature
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพแสดงให้เห็นส่วนประกอบที่เพิ่มขึ้นมาหลังจากมีการ ทำ Digital signature โดยส่วนแรกคือ <Signature> ซึ่งเป็นอีลิเมนต์เริ่มต้นที่จะประกอบด้วยอีลิเมนต์ย่อย ที่จะใช้ในการทำลายเซ็นอิเล็กทรอนิกส์ โดยเป็นส่วนที่เก็บข้อมูลทั้งหมดซึ่งกล่าวถึง information ที่ใช้ในการ sign เช่น signature , key ที่ใช้ sign , ตำแหน่งที่ใช้ในการเก็บ information โดยภายในจะประกอบด้วยอีลิเมนต์ต่างๆดังนี้

- <SignedInfo>, <SignatureMethod>, <CanonicalizationMethod> เป็นส่วนที่ระบุชนิดของ signature เช่น Kerberos หรือ RSA ซึ่งใช้ในการสร้าง signature element เหล่านี้ เป็นส่วนที่แสดงให้เห็นว่ามีการแบ่งการทำงานเป็นส่วนๆ และสามารถแสดงให้เห็นถึงการป้องกันการเปลี่ยนแปลงแก้ไขข้อมูล จากบุคคลที่ 3
- <SignatureValue> เป็นตัวระบุถึงของ signature ที่ได้ เป็นส่วนที่เก็บข้อมูลรายละเอียดของ application
- <KeyInfo> เป็นตัวที่จะบอกว่า ควรจะใช้กุญแจดอกไหนในการในการทำลายเซ็นอิเล็กทรอนิกส์ ภายในจะมีอีลิเมนต์ <SecurityTokenReference> ที่ใช้ในการอ้างอิงถึง security token ที่ใช้

7.6 ผลการทดลองจาก Business process execution language (BPEL)

จากที่กล่าวแล้วข้างต้นว่า BPEL จะมีลักษณะเป็น work flow ที่ควบคุมการทำงานให้เป็นไปตามเงื่อนไขในแผนภาพ ภาพด้านล่างเป็นตัวอย่าง BPEL ซึ่งเป็นขั้นตอนของการค้นหาราคาขึ้นมาให้ผู้ซื้อในส่วนของ การจองสายการบิน โดยเว็บแอปพลิเคชันจะมาเรียกใช้ BPEL ซึ่งใน BPEL จะมีการกำตวรจสอบว่าจะดำเนินการไปทางไหนตามเงื่อนไขที่ได้ระบุไว้ แล้วส่งค่ากลับไปยังเว็บผ่าน activity reply

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8

บทวิจารณ์และสรุป

8.1 บทวิจารณ์และสรุปผล

การสร้างเว็บเซอร์วิสขึ้นมาโดยปราศจากการทำ security จะทำให้การใช้งานไม่ปลอดภัย เพราะอาจถูกดักจับข้อมูลที่อยู่ใน SOAP message ได้ ซึ่งจะสร้างความเสียหายให้กับผู้ใช้งาน ดังนั้น สิ่งที่สำคัญที่สุดของเว็บเซอร์วิสคือ เรื่องของความปลอดภัย จึงมีการนำ WS-Security มาประยุกต์ใช้ร่วมกับแอปพลิเคชันในการรักษาความปลอดภัยให้กับการใช้งานเว็บเซอร์วิส โดย WS-Security ออกแบบมาเพื่อเพิ่มความปลอดภัยของข้อมูลในระดับ Message ในการส่งข้อมูลระหว่าง เว็บเซอร์วิส แบบ End-to-End ช่วยเพิ่มประสิทธิภาพในการส่ง SOAP Message ให้มีความน่าเชื่อถือ มีความถูกต้องและมีความปลอดภัย (integrity, confidentiality, authentication) ส่งผลให้ข้อมูลมีความปลอดภัยตลอดเวลาไม่ว่าจะถูกส่งไปให้ Service ใหนก็ตาม

นอกจากนี้การสร้าง Business Process Execution Language ขึ้นมาจะทำให้ง่ายต่อการแก้ไข Business Logic ในอนาคต ประกอบกับการใช้ WS-Transaction จะทำให้ธุรกรรมใดๆที่เกิดขึ้นมีความสมบูรณ์ กล่าวคือ ถ้าเกิดปัญหาเกี่ยวกับบริการใดบริการหนึ่งเกิดขึ้น บริการทั้งหมดจะถูกยกเลิกไป ซึ่งจะไม่ก่อให้เกิดปัญหาให้กับผู้ใช้ และปัญหาต่อระบบ

โครงการนี้ได้ทำการสร้างเว็บเซอร์วิส ซึ่งมีการรวมเอาเทคโนโลยีต่างๆข้างต้นมาประยุกต์ใช้อีกทั้งมีการทดสอบโดยการนำไปใช้ในระบบจริง โดยมีการสร้างเว็บแอปพลิเคชันที่ใช้ติดต่อเว็บเซอร์วิส เพื่อช่วยในการจองตั๋วเครื่องบิน, จองโรงแรม, จองแพคเกจทัวร์ ซึ่งเป็นตัวแทนในการติดต่อสื่อสารกับบริษัทต่างๆที่เกี่ยวข้อง โดยผ่านการติดต่อสื่อสารผ่านเว็บเซอร์วิส

8.2 ปัญหาอุปสรรคและแนวทางแก้ไข

1. ปัญหาในการเลือกใช้ tool มาใช้ในพัฒนาเว็บเซอร์วิส เนื่องจากมี tool ให้เลือกใช้หลากหลาย จึงจำเป็นต้องเลือก tool ที่สามารถ implement มาตรฐานของเว็บเซอร์วิสที่เราต้องการ ได้ จึงจำเป็นต้องทดลองใช้ tool ต่างๆ และเลือกอันที่ดีที่สุดมาใช้งาน
2. ด้วยความใหม่ของเทคโนโลยีที่เป็นมาตรฐานของเว็บเซอร์วิส ทำให้เสียเวลาในการค้นคว้าหาข้อมูล ซึ่งไม่มีข้อมูลที่แน่นอนที่จะนำมา implement ได้ในทันที และไม่มีความชัดเจนในการนำไปใช้ ดังนั้นจึงต้องใช้มาตรฐานตัวล่าสุดที่มีการ implement เป็นเกณฑ์ในการหาเอกสารอ้างอิง และนำข้อมูลดังกล่าวมาประยุกต์ใช้
3. ความไม่เข้ากันของ tool ที่ใช้ในการพัฒนาหลายๆตัว เช่น IBM Rational Application

Developer 6.0 กับ ActiveBpel Designer 2.1 ทำให้เกิดปัญหาในการเรียกใช้งานระหว่างกันแล้ว เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กัน เช่น การใช้โปรแกรม IBM Rational Application Developer 6.0 ในการสร้าง application เพื่อเรียกใช้ service ของ BPEL ที่มีการใช้ SSL นั้นไม่สามารถกระทำได้ ทำให้ต้องมีการเขียนโปรแกรมในการติดต่อเองโดยตรงโดยใช้ library มาตรฐานของ axis ดังนี้ โดยก่อนอื่นเราต้องทำการ Import library ของ axis เข้ามาใน project ของเราก่อนดังนี้

- activation.jar
- axis-ant.jar
- axis.jar
- commons-discovery-0.2.jar
- commons-logging-1.0.4.jar
- jaxrpc.jar
- log4j-1.2.8.jar
- mail.jar
- saaj.jar
- wsdl4j-1.5.1.jar

```
<%@ page import="org.apache.axis.client.Call
org.apache.axis.client.Service,
javax.xml.namespace.QName,
javax.xml.rpc.ParameterMode,
org.apache.axis.encoding.*" %>
<%!
public String callService(String s,int i,int j)
{
try{
String endpoint =
"https://161.246.6.118:8443/active-bpel/services/olala_bpelPLService1";
System.setProperty("javax.net.ssl.keyStore", "c:\\key\\ClientKeyStore.jks");
System.setProperty("javax.net.ssl.keyStorePassword", "longly15");
System.setProperty("javax.net.ssl.keyStoreType", "JKS");
System.setProperty("javax.net.ssl.trustStore", "c:\\key\\ClientTrustStore.jks");
System.setProperty("javax.net.ssl.trustStorePassword", "longly15");
Service service = new Service();
Call call = (Call)service.createCall();
call.setTargetEndpointAddress(new java.net.URL(endpoint));
call.setOperationName(new QName("olala_bpel", "operation"));
call.setPortTypeName(new QName("olala_bpelPT"));
call.addParameter("string_xml_in",XMLType.XSD_STRING,ParameterMode.IN);
call.addParameter("type_call",XMLType.XSD_INT,ParameterMode.IN);
call.addParameter("subType_call",XMLType.XSD_INT,ParameterMode.IN);
call.setReturnType(XMLType.XSD_STRING);
Integer i1 = new Integer(i);
Integer i2 = new Integer(j);
String s1 = new String(s);
String result = (String)call.invoke(new Object[] {s1,i1,i2});
return result;
}
catch(Exception e){
return e.getMessage();
}
}
%>
```

รูปที่ 8.1 โค้ดที่ต้องเขียนในการเรียกใช้ BPEL จาก Rational

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากนั้นเราทำการเขียน Code ข้างต้นซึ่งมีรายละเอียดดังนี้

- บรรทัดที่ 1-5 เป็นการ import library ของ axis มาใช้
 - บรรทัดที่ 10 เป็นการตั้งค่า endpoint ของ service
 - บรรทัดที่ 12-16 เป็นการตั้งค่า keystore ที่จะใช้และ trustkey ที่เราเชื่อถือ
 - บรรทัดที่ 20 เป็นการกำหนดชื่อของ operation ที่เราสร้างไว้
 - บรรทัดที่ 21 เป็นการกำหนดชื่อของ port type
 - บรรทัดที่ 22-25 เป็นการกำหนด parameter และ return type ของ service
 - บรรทัดที่ 29 เป็นการเรียก Web Service
 - บรรทัดที่ 30 เป็นค่าที่ return กลับมา
4. ไม่สามารถทำการติดต่อระหว่าง platform (JAVA และ .NET) ตามมาตรฐาน WS-Security เนื่องจาก JAVA และ .NET ใช้ WS-Addressing คนละเวอร์ชันกัน
5. เนื่องจาก BPEL ยังเป็นเทคโนโลยีที่ค่อนข้างใหม่ ดังนั้นจึงมี tool ที่ให้เลือกใช้ไม่หลากหลายนัก บาง tool เช่น ActiveBPEL ไม่สนับสนุนเทคโนโลยีพื้นฐานของเว็บเซอร์วิส เช่น ws-security ทำให้ต้องหันมาใช้การรักษาความปลอดภัยแบบ SSL ดังนั้นควรมีศึกษาหาข้อมูลใหม่สำหรับ tool ต่างๆอยู่เสมอ
7. ข้อมูลที่เว็บเซอร์วิสรับและส่งกลับมายังผู้ใช้เรียกใช้จะอยู่ในรูปของ complexType ดังนั้นเพื่อจะใช้ BPEL เรียกใช้เว็บเซอร์วิสและส่งข้อมูลดังกล่าวให้เว็บเซอร์วิส ได้จะต้องมีการคอนฟิกรเพิ่มเติมที่ตัว ActiveBpel Engine

8.3 แนวทางการพัฒนาต่อ

1. พัฒนาระบบโดยมีทางเลือกให้กับผู้ใช้ที่หลากหลายมากขึ้น เช่น สามารถของสายการบินหรือค้นหาข้อมูลได้มากกว่าหนึ่งสายการบิน โรงแรมมากกว่าหนึ่งโรงแรม
2. พัฒนาเว็บเซอร์วิสที่มีการนำมาตราฐานอื่นๆของเว็บเซอร์วิสมาใช้
3. พัฒนาในส่วนของเว็บแอปพลิเคชันให้มีความน่าใช้มากขึ้นและมีข่าวสารแนะนำข้อมูลการท่องเที่ยวให้แก่ผู้ใช้งาน เพื่อเป็นแนวทางให้ผู้ใช้ตัดสินใจเลือกได้มากยิ่งขึ้น
4. ทำการพัฒนาในส่วนของแพคเกจจิ้งทำให้สามารถเดินทางไปยังสถานที่ที่หลากหลายมากขึ้น ซึ่งก็คือสามารถไปท่องเที่ยวและหาที่พักในหลายๆแห่งได้ด้วย อย่างเช่น การแพคเกจเที่ยวภาคเหนือ สามารถที่จะแวะท่องเที่ยวและพักผ่อนได้ในหลายจังหวัด ทั้งลำพูน, ลำปาง, เชียงใหม่, เชียงราย ได้ในการเดินทางเพียงครั้งเดียว
5. พัฒนาเว็บเซอร์วิสให้สามารถทำงานร่วมกันได้ระหว่าง JAVA และ .NET ซึ่งอาจจะมองได้ว่าเป็นแนวทางหลักที่สำคัญอย่างยิ่ง เพราะถ้าสามารถทำให้ JAVA และ .NET ทำงานร่วมกันได้จะ

สามารถขยายขอบเขตการใช้งานของเว็บเซอร์วิสเพิ่มขึ้นได้อีกมาก เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

เอกสารอ้างอิงที่เป็นหนังสือ

- [1] ทินกร วัฒนเกษมสกุล 2549 “คัมภีร์ JSP” สำนักพิมพ์เคทีพี กรุงเทพฯ

เอกสารอ้างอิงที่เป็นปริญญาานิพนธ์

- [2] สุรเชษฐ เกียรติเรืองชัย, วราวุธ โรจนศิริบุตร , โอฬาร ลินะอังกรุจิ “ระบบสารสนเทศผ่านเว็บเซิร์ฟวิส” ปริญญาานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2547
- [3] นายวรมธ วุฒชะวนิช, นายวิชาติ อำไพภิญโญกุล “ระบบสารสนเทศผ่านเว็บเซิร์ฟวิส” ปริญญาานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2547
- [4] นายวรงค์ แสงโสภณ, นายอริชา เกษมสุขวิช, นายอาศุพล ปรียกนก “การพัฒนาแอปพลิเคชันด้วยเว็บเซิร์ฟวิส” ปริญญาานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2548
- [5] นายกฤษณ์ วรสุวรรณรักษ์, นายกฤษดา วัฒนาจารุพงศ์, นายชลธรรม ประภากรวรกุล “การพัฒนาแอปพลิเคชันด้วยเว็บเซิร์ฟวิส” ปริญญาานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2548

เอกสารอ้างอิงที่เป็น web-site

- [6] SUN. “Securing Web Services.” [Online].
Available :
http://www.sun.com/software/whitepapers/webservices/securing_webservices.pdf. 2003.
- [7] นัศรชัย สุขสะอาด. “Web Services abc.” [Online].
Available : <http://www.wsiam.com/document/abcwebservices/webservicesabc.jsp>. 2004.
- [8] OASIS. “Username Token.” [Online]. Available : <http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0.pdf>. 2004.
- [9] Nectec. “ความรู้เบื้องต้นการเข้ารหัสข้อมูล.” [Online].
Available : http://thaicert.nectec.or.th/paper/encryption/intro_crypt.php. 2004.
- [10] Nectec. “ความรู้เบื้องต้นการพิสูจน์ตัวตน.” [Online]. Available :
http://thaicert.nectec.or.th/paper/authen/authentication_guide.php. 2004.

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [11] ฉัตรชัย สุขสะอาด. “Web services เครื่องมือธุรกิจยุคใหม่.” [Online].
Available : http://www.wsiam.com/document/WebServices_BusinessTool.pdf. 2004
- [12] Mark Colan. “Understanding Web Services Security.” [Online].
Available : <http://ibm.com/developerworks/speakers/colan>. 2004.
- [13] Morag Hughson. “WebSphere MQ Channel Security with SSL.” [Online].
Available :
<ftp://ftp.software.ibm.com/software/integration/library/presentations/sslweb.pdf>. 2005.
- [14] IBM. “Introduction to Cryptography.” [Online].
Available : <http://www-128.ibm.com/developerworks/security/library/s-crypt02.html>. 2001.
- [15] Keytool. “Key and Certificate Management tool.” [Online].
Available : <http://java.sun.com/j2se/1.5.0/docs/tooldocs/windows/keytool.html#CertChains>.
2004.

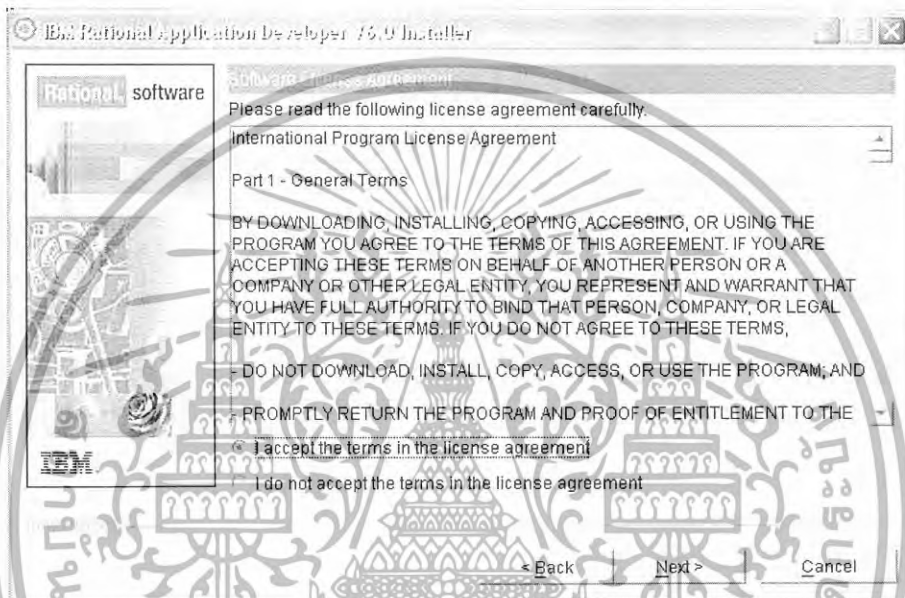


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก คู่มือการติดตั้ง

1. การติดตั้งโปรแกรม IBM Rational Application Developer 6.0

1.1 ทำการรันตัว setup โปรแกรมหลังจากนั้นเมื่อเข้าสู่หน้า Software License Agreement ให้เลือกที่ I accept และคลิก Next



รูปที่ ก.1 หน้า Software License Agreement

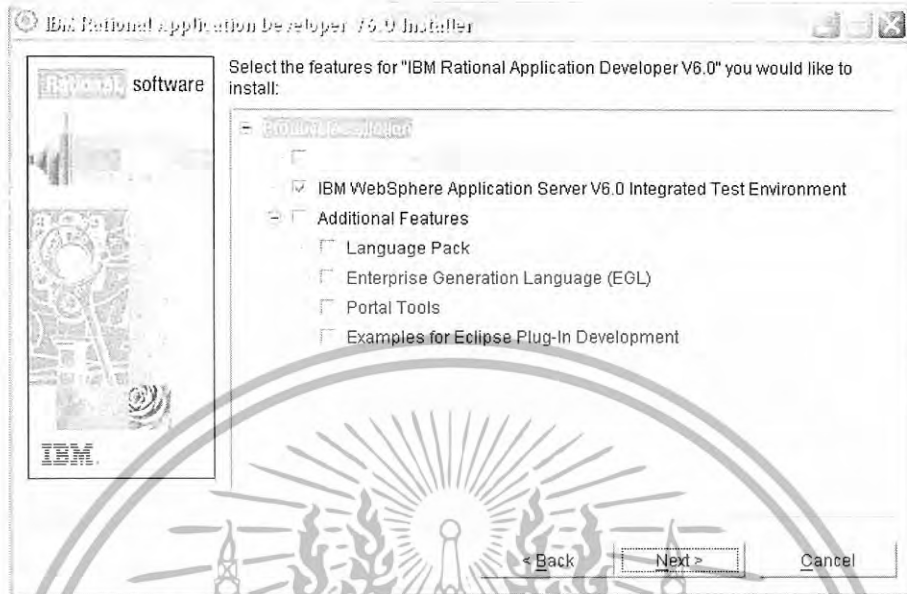
1.2 ทำการเลือก directory ที่จะใช้ในการติดตั้งโปรแกรม และคลิก Next



รูปที่ ก.2 หน้า Directory Name for Set Up

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ... ให้นำไปใช้ประโยชน์ด้านการค้า... ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ทำการเลือก feature ตามที่ต้องการ ในที่เลือก IBM WebSphere Application Server V6.0 Integrated Test Environment



รูปที่ ก.3 หน้า Select the feature

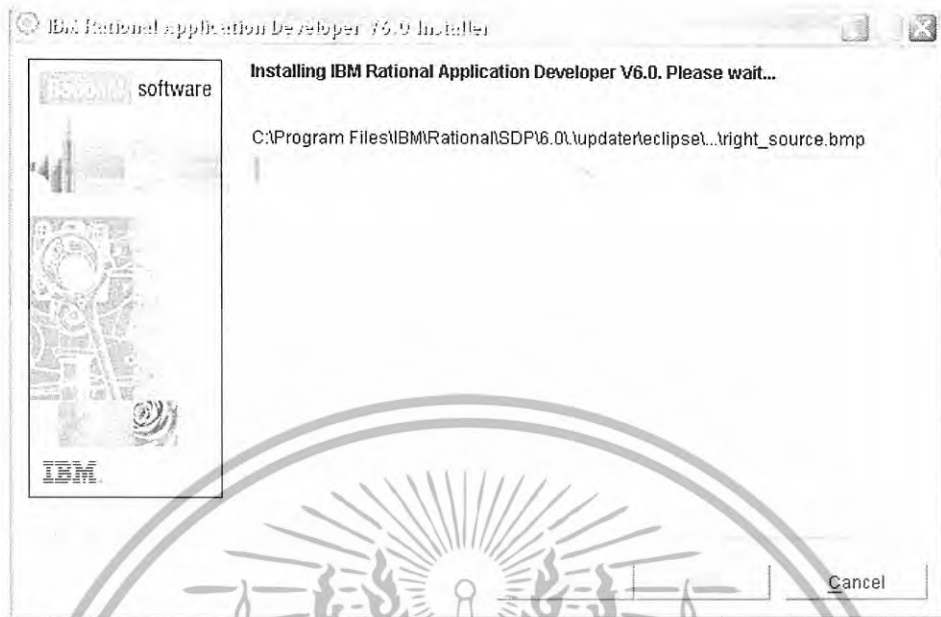
1.4 หลังจากนั้นก็ขึ้นหน้าสรุปให้ หลังจากนั้นคลิก Next



รูปที่ ก.4 หน้า Summary Information

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 หลังจากนั้นจะทำการติดตั้งโปรแกรมให้ตาม directory ที่ได้ระบุไว้



รูปที่ ก.5 หน้า Installing

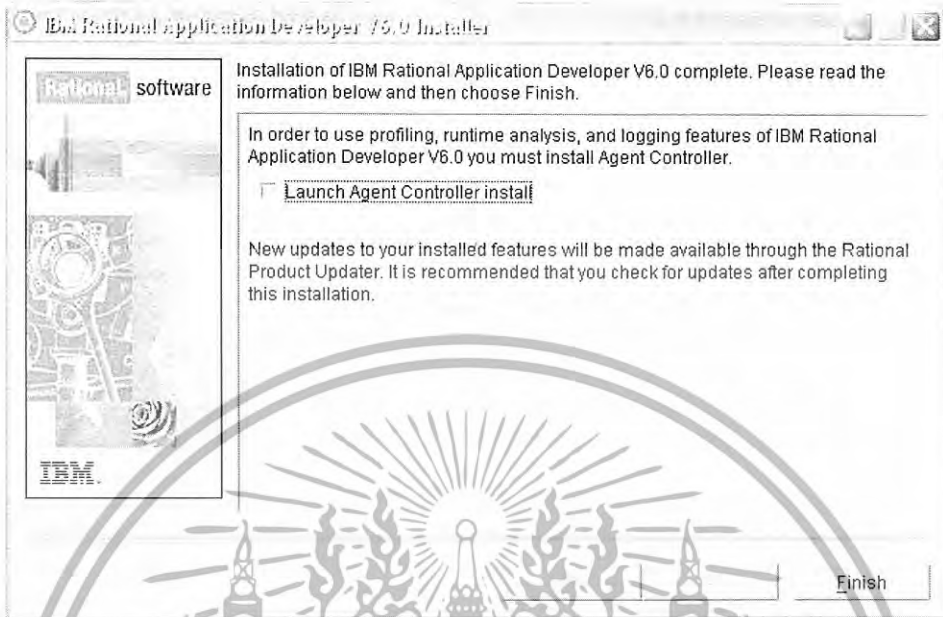
1.6 หลังจากทำการติดตั้งโปรแกรมเสร็จ จะปรากฏหน้าต่างที่ระบุว่าการติดตั้งโปรแกรมได้สำเร็จแล้ว



รูปที่ ก.6 หน้า Install Successful

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

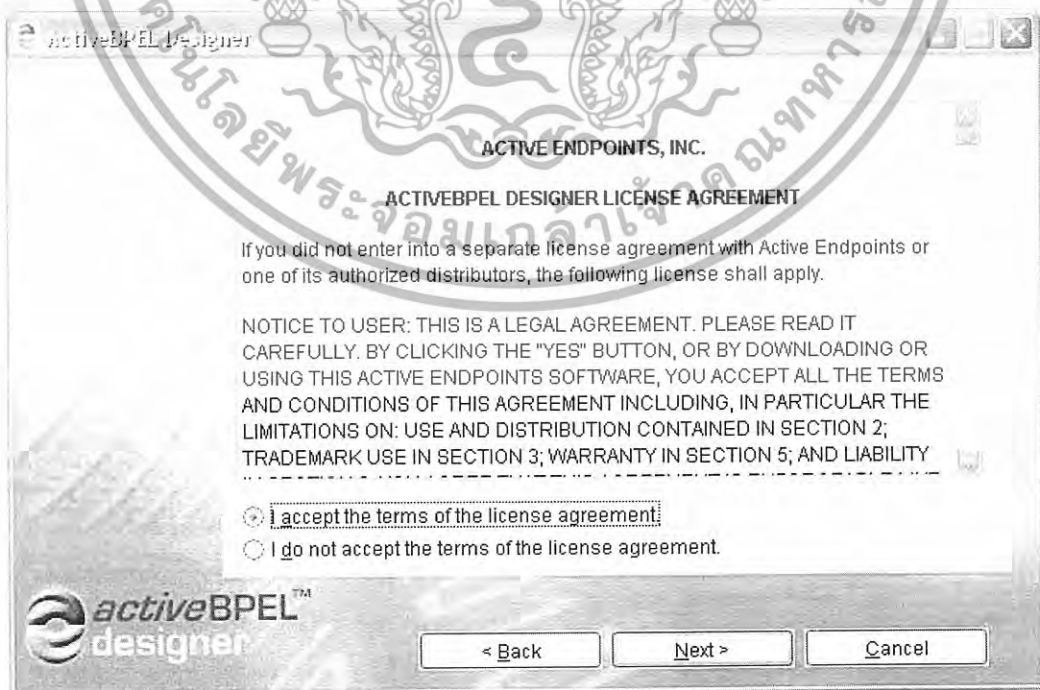
1.7 หลังจากจะปรากฏหน้าต่างการลงโปรแกรม ให้คลิกที่ Lunch Agent Controller Install เพื่อเอาเครื่องหมายออก



รูปที่ ก.7 หน้า Finish Installation

2. การติดตั้งโปรแกรม ActivePEL Designer 2.0

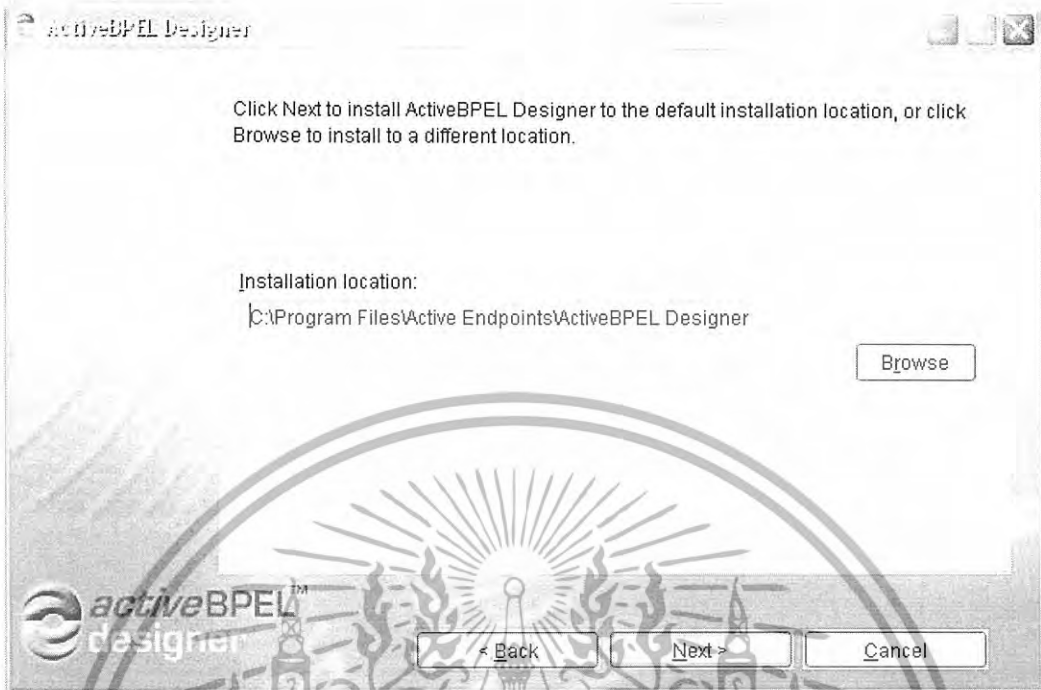
2.1 ทำการรันตัว setup โปรแกรมหลังจากนั้นเมื่อเข้าสู่หน้า Software License Agreement ให้เลือกที่ I accept และคลิก Next



รูปที่ ก.8 หน้า Software License Agreement

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ทำการเลือก directory ที่จะใช้ในการติดตั้งโปรแกรม และคลิก Next



รูปที่ ก.9 หน้า Installation Location

2.3 คลิกเมาส์ที่เพื่อทำการเลือกที่ ActiveBPEL Designer Samples เพื่อทำการลงตัวอย่างการใช้งานโปรแกรม หลังจากนั้นคลิกที่ Next



รูปที่ ก.10 หน้า Installation Location

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 คลิกหลังจากนั้นจะขึ้นหน้าสรุปให้ หลังจากนั้นคลิก Next



รูปที่ ก.11 หน้า Summary Information

2.5 หลังจากนั้นจะทำการติดตั้ง โปรแกรมให้ตาม directory ที่ได้ระบุไว้



รูปที่ ก.12 หน้า Installing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 หลังจากจะปรากฏหน้าต่างการลงโปรแกรม



รูปที่ ก.13 หน้า Finish Installation

3. การติดตั้ง Oracle Database Server 10g release 2 แบบ Basic

3.1 double-click file “setup.exe” มีสัญลักษณ์ดังรูป  เพื่อ start Oracle Universal Installer หรือ รอให้ Autorun ทำงานซึ่งจะแสดงหน้าต่างดังภาพ ให้เลือก Install/Deinstall Product

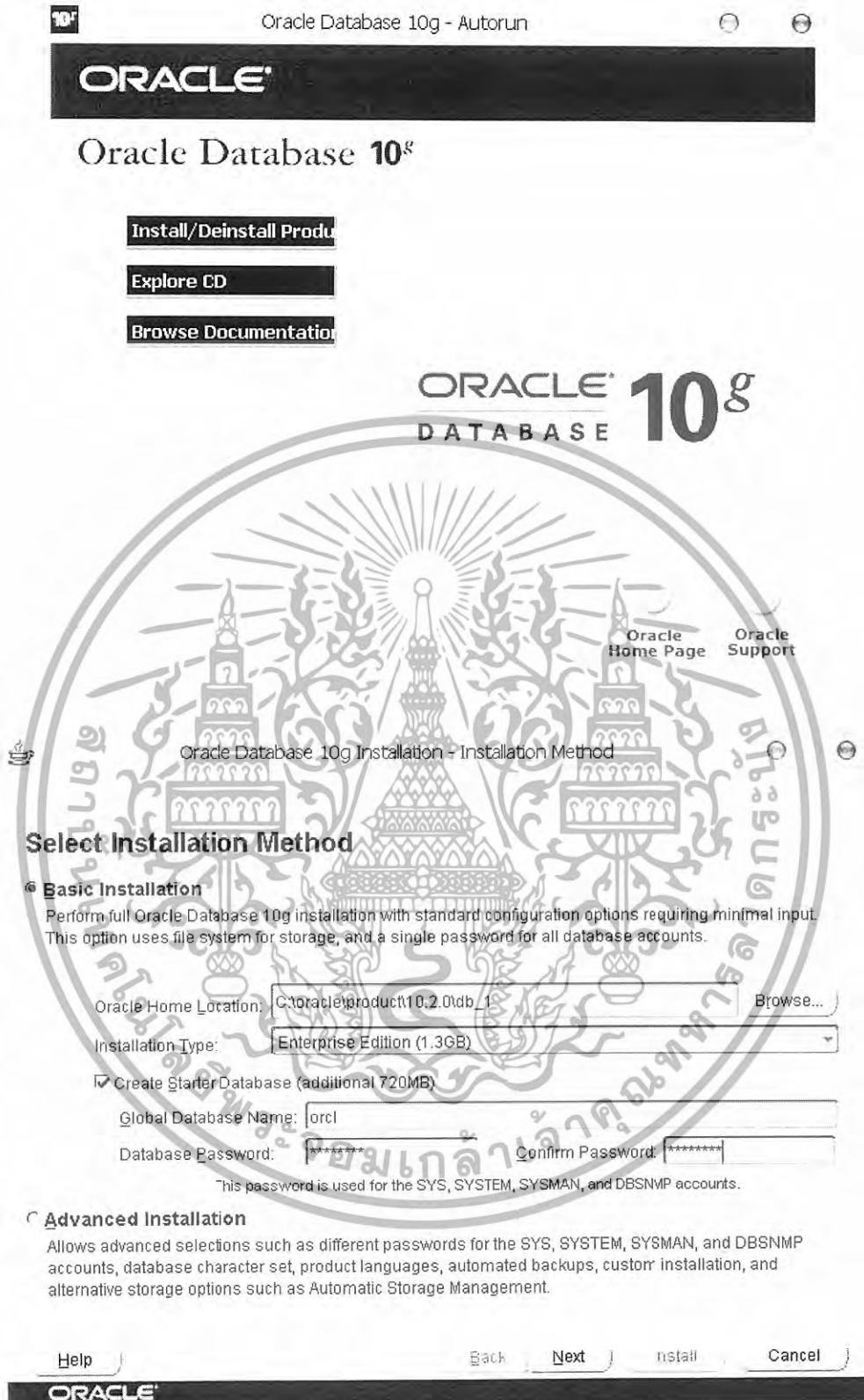
3.2 เข้าสู่ขั้นตอนการเลือก mode ของการ install ให้เลือก Basic Installation และในช่องของ Database Password และ Confirm Password ให้ใส่ Password ที่มีความยาวระหว่าง 4 ถึง 30 ตัวอักษร โดยสามารถมีสัญลักษณ์ต่อไปนี้ได้ คือ underscore(_), dollar(\$), pound sign(#) และ

- ห้ามขึ้นต้นด้วยตัวเลข
- ห้ามใส่ password ที่เหมือนกับ user name
- ห้ามใช้คำสงวนของ Oracle
- ห้ามใช้คำว่า change_on_install, sysman, dbsnmp เป็น password

หมายเหตุ ในที่นี้ให้ใส่คำว่า password ทั้งในช่อง Database Password และ Confirm Password)

แล้ว click ปุ่ม Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

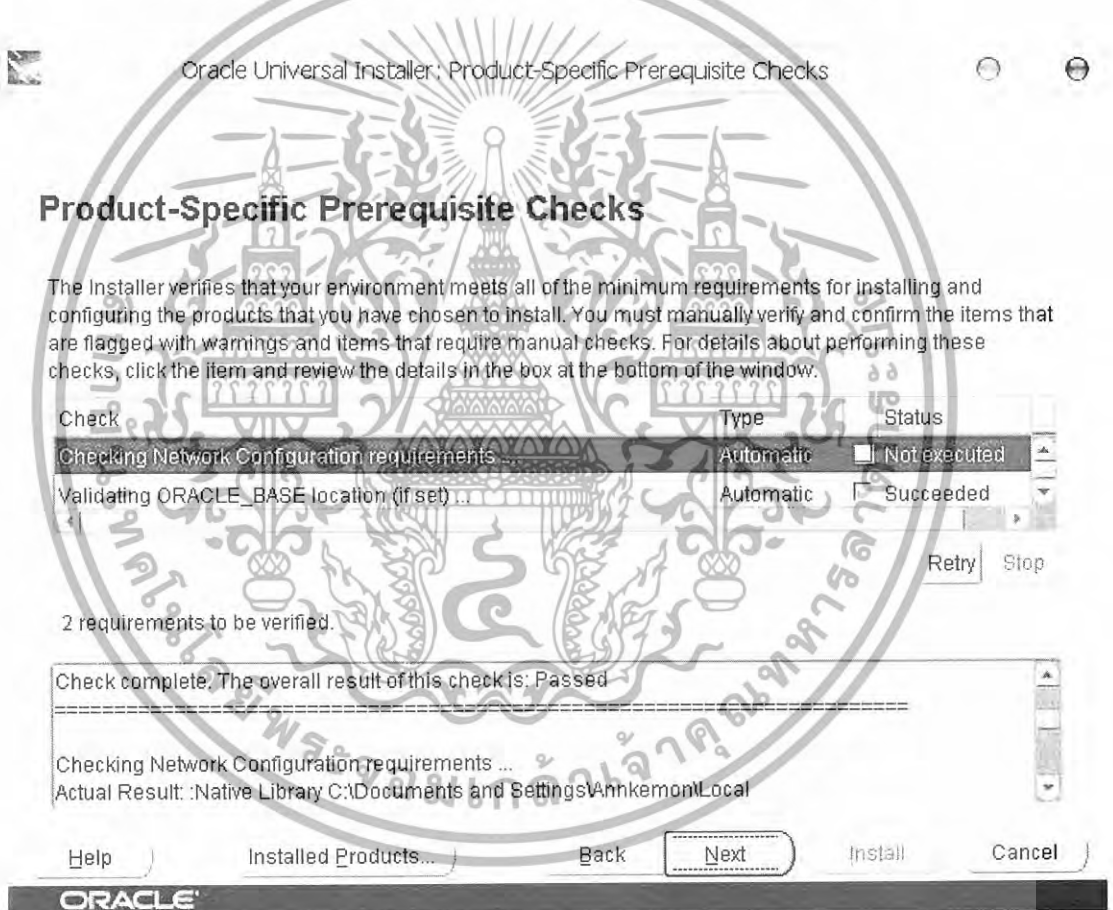


หลังจากนั้นระบบจะเตรียมการ Install ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

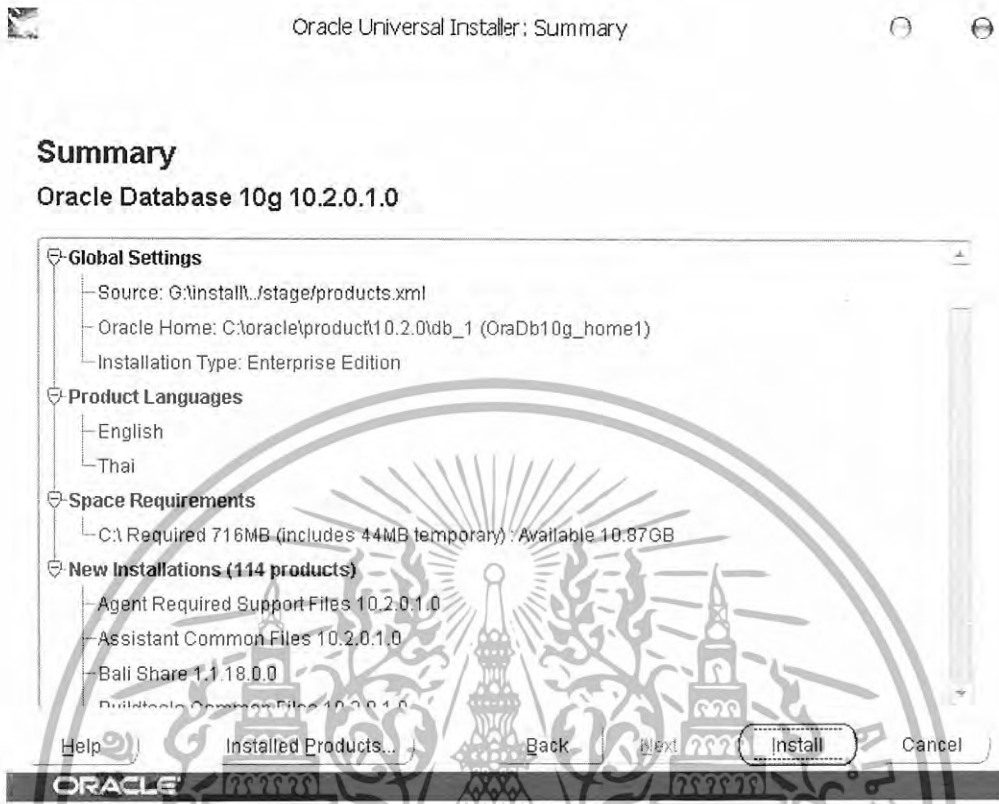


3.3 เมื่อแสดงหน้าต่าง Product-Specific Prerequisite Checks ให้ click ปุ่ม next

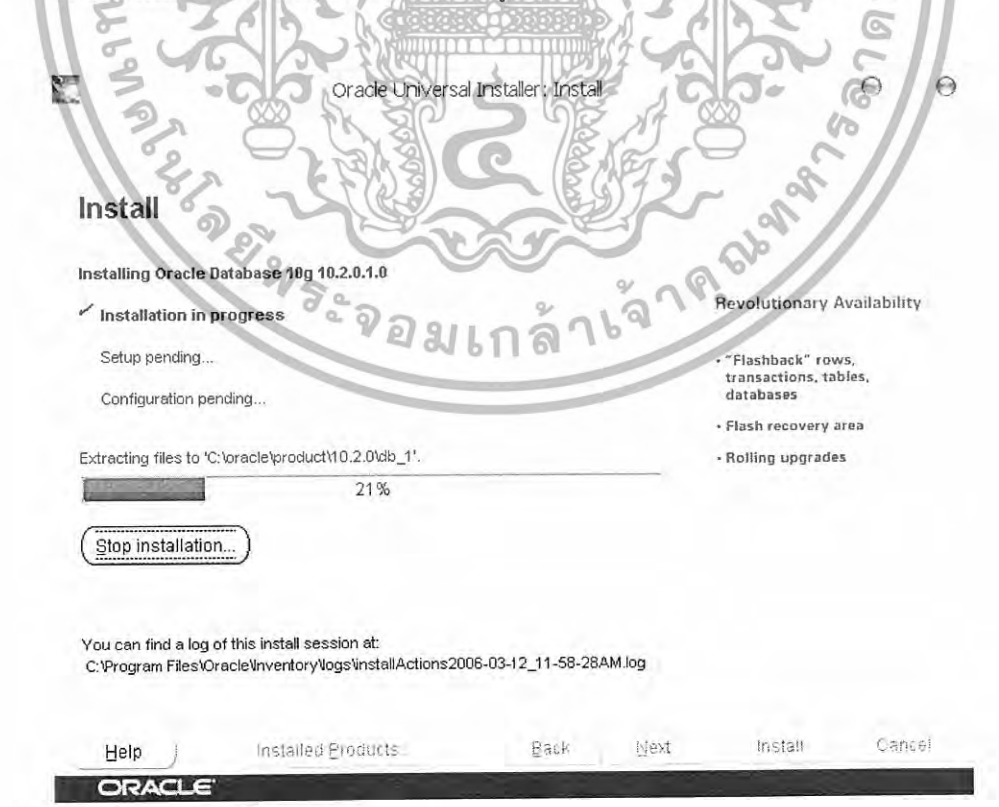


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 จากหน้าต่างสรุปผลให้เลือก Install

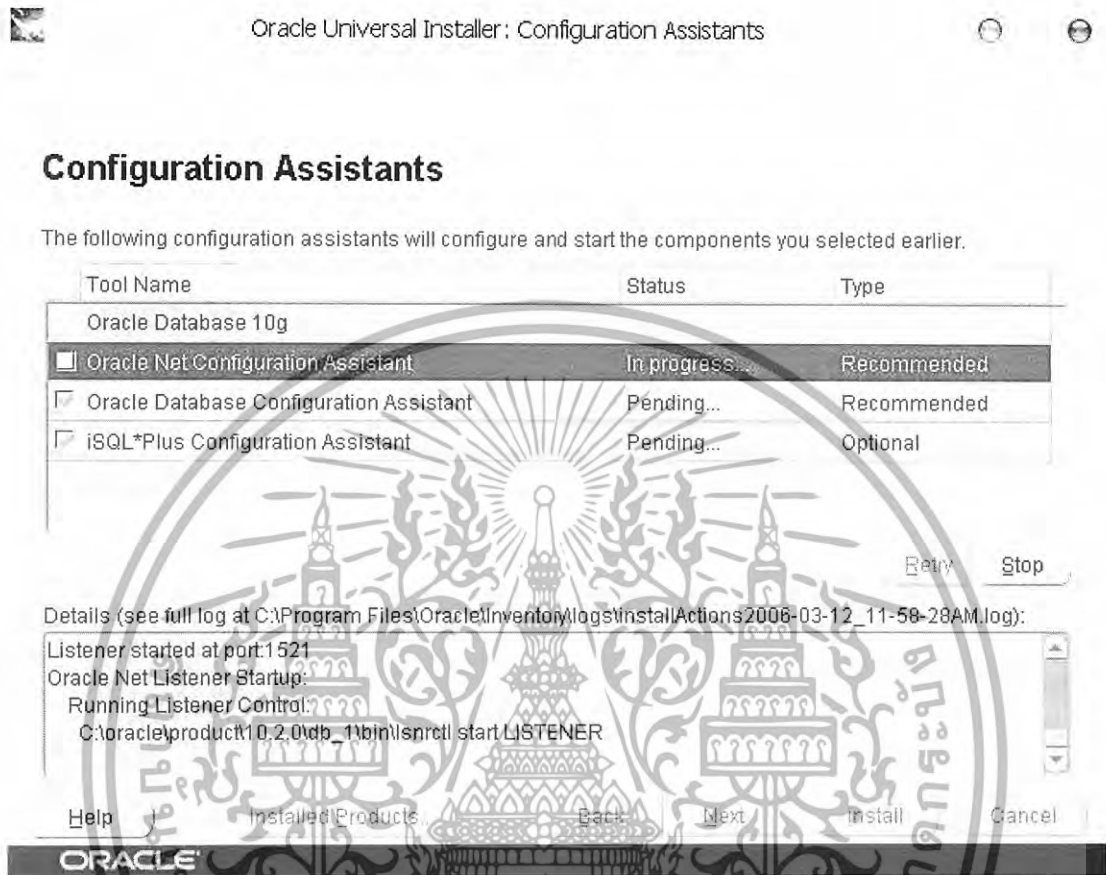


ขณะ install จะแสดงหน้าต่าง Install ดังรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

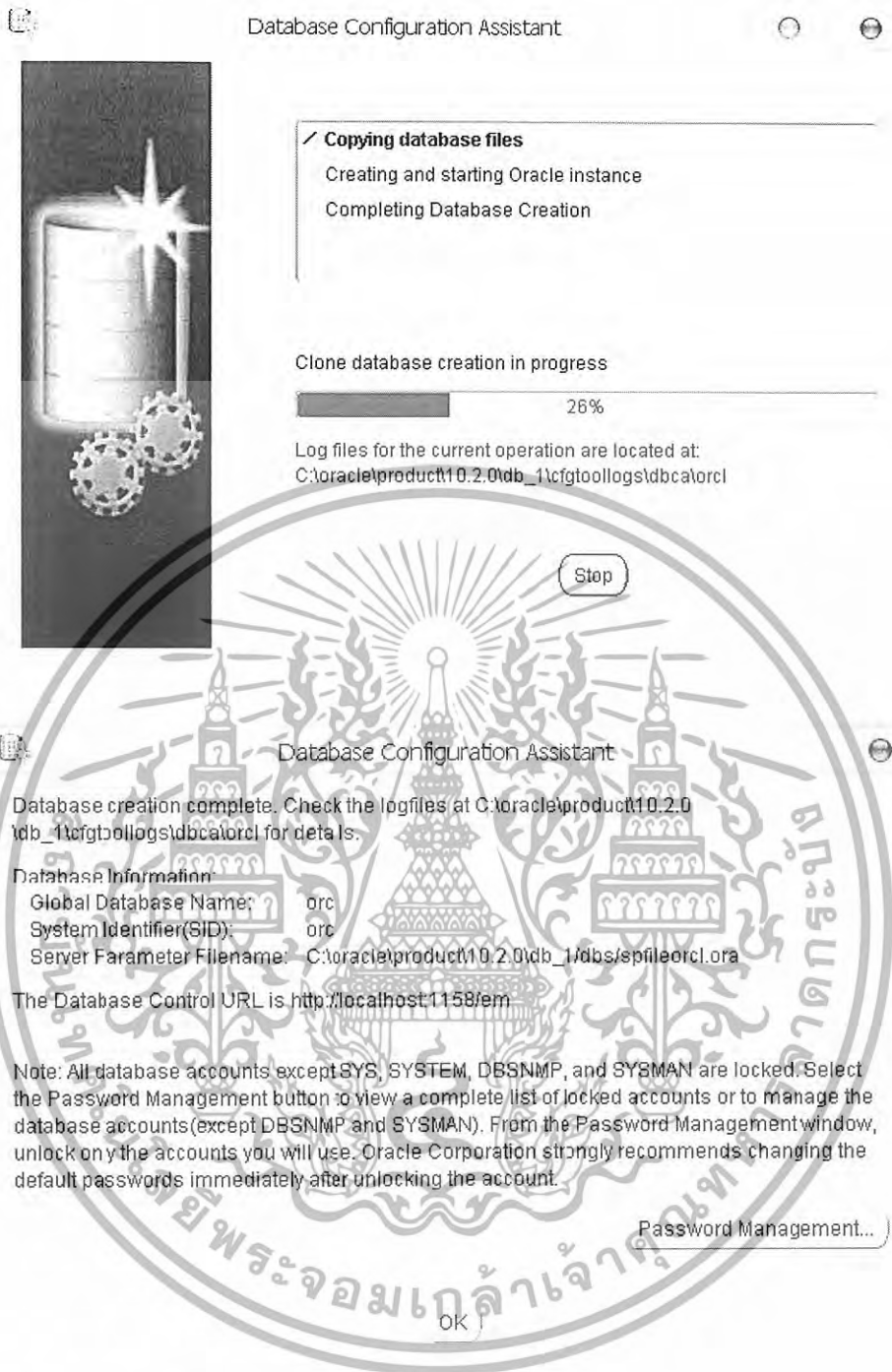
หลังจากนั้นจะแสดงหน้าต่าง Oracle Universal Installer : Configuration Assistants (ซึ่งผู้ติดตั้งไม่ต้องทำอะไร) ดังรูป



ต่อมาจะแสดงหน้าต่าง Database Configuration Assistant (ซึ่งผู้ติดตั้งไม่ต้องทำอะไร) ดังรูป

3.5 ในขั้นตอนต่อมาเป็นการแสดงรายละเอียดผลของกร install ให้ click ปุ่ม OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



3.6 ในขั้นตอนสุดท้ายของการ install ให้ click ปุ่ม Exit



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End of Installation

The installation of Oracle Database 10g was successful.

Please remember...

Enterprise Manager Database Control URL - (orcl) :
 http://localhost:1158/em

Your database configuration files have been installed in C:\oracle\product\10.2.0 while other components selected for installation have been installed in C:\oracle\product\10.2.0\db_1. Be cautious not to accidentally delete these configuration files.

The iSQL*Plus URL is:
 http://localhost:5560/isqlplus

The iSQL*Plus DBA URL is:
 http://localhost:5560/isqlplus/dba

Help Installed Products... Back Next Install Exit

4. การติดตั้ง MySQL Server 4.1

MySQL Server 4.1 - Setup Wizard

Welcome to the Setup Wizard for MySQL Server 4.1

The Setup Wizard will install MySQL Server 4.1 release 4.1.11 on your computer. To continue, click Next.

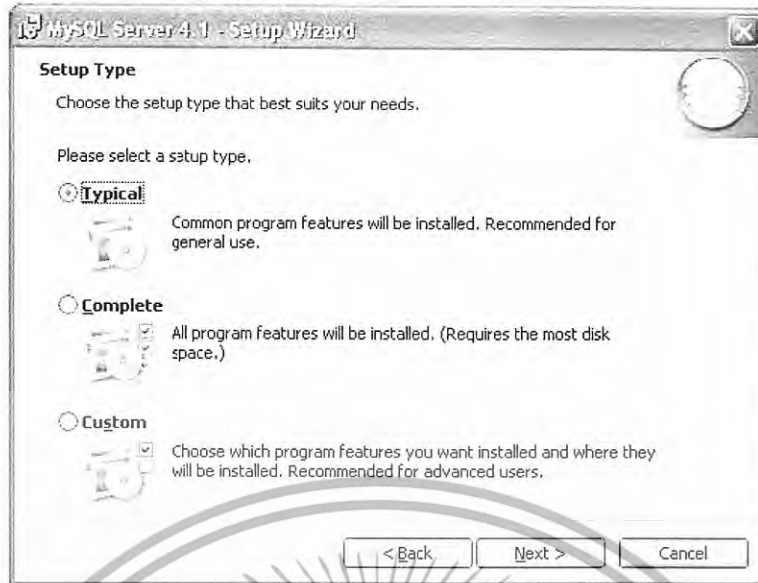
WARNING: This program is protected by copyright law.

MySQL

Next > Cancel

1. ปรากฏหน้าจอ Welcome to the Setup Wizard for MySQL Server 4.1 คลิกปุ่ม Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

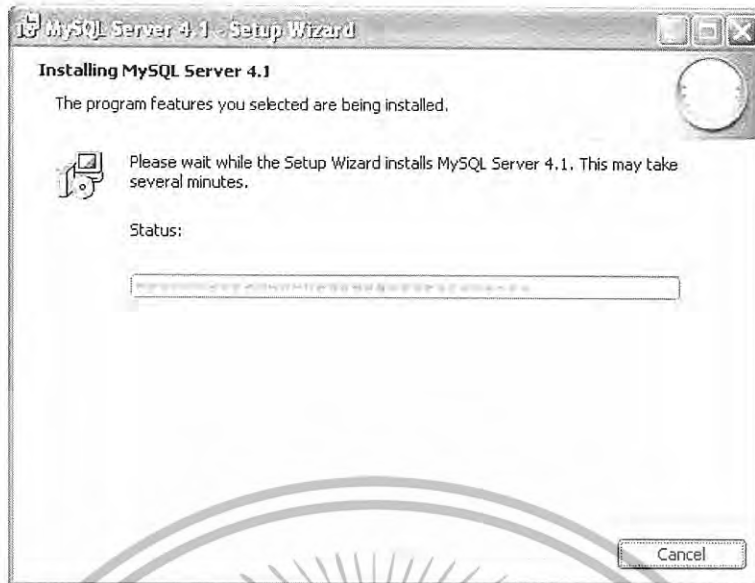


2. ปรากฏหน้าจอ Setup Type เลือก Typical จากนั้นคลิกปุ่ม Next



3. ปรากฏหน้าจอ Ready to Install the Program จากนั้นคลิกปุ่ม Install

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

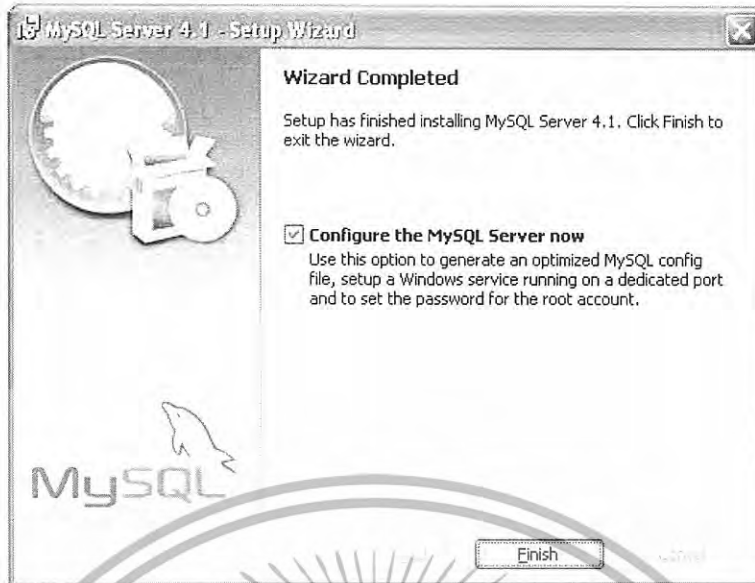


4. ปราบกฏหน้าจอ Installing MySQL SQL Server



5. ปราบกฏหน้าจอ MySQL.com Sign-Up เลือก Skip Sign-Up จากนั้นคลิกปุ่ม Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

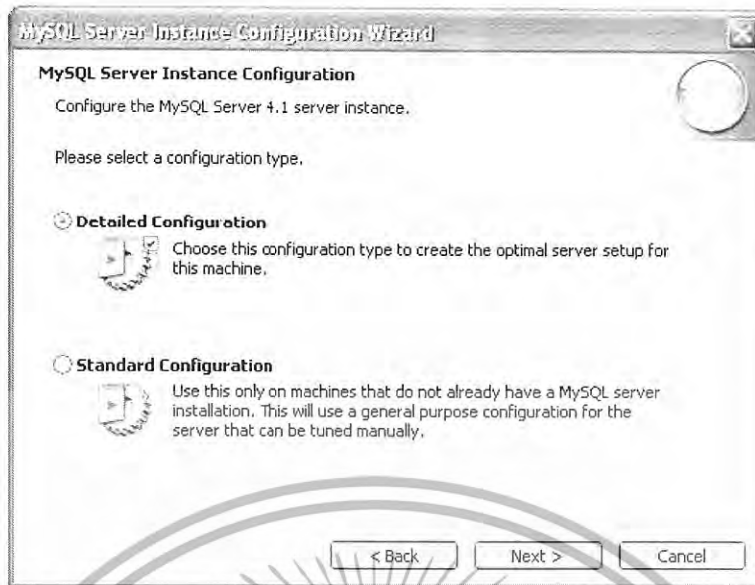


6. ปรากฏหน้าจอ Wizard Completed เลือกทำการ Configure the MySQL Server now จากนั้นคลิกปุ่ม Finish



7. ปรากฏหน้าจอ Welcome to the MySQL Server Instance Configuration Wizard จากนั้นคลิกปุ่ม Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

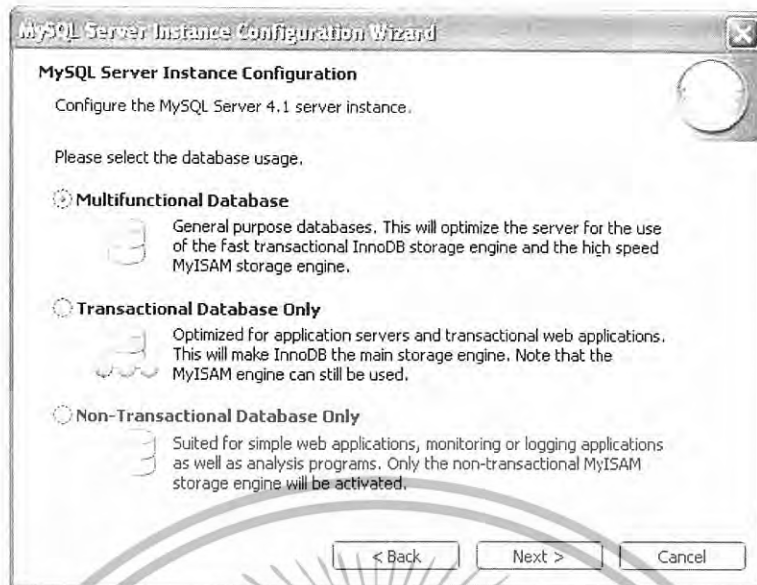


8. ปราบกฏหน้าจอ MySQL Server Instance Configuration เลือก Detailed Configuration จากนั้นคลิกปุ่ม Next



9. หน้า Please select a server type เลือก Developer Machine จากนั้นคลิกปุ่ม Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

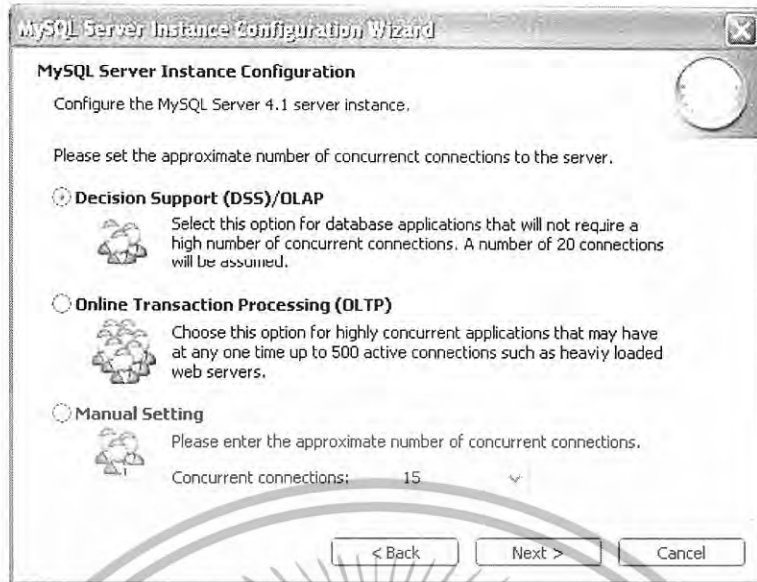


10. หน้า Please select the database usage. เลือก Multifunctional Database จากนั้นคลิกปุ่ม Next



11. หน้า Please select the drive for the InnoDB datafile จากนั้นคลิกปุ่ม Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

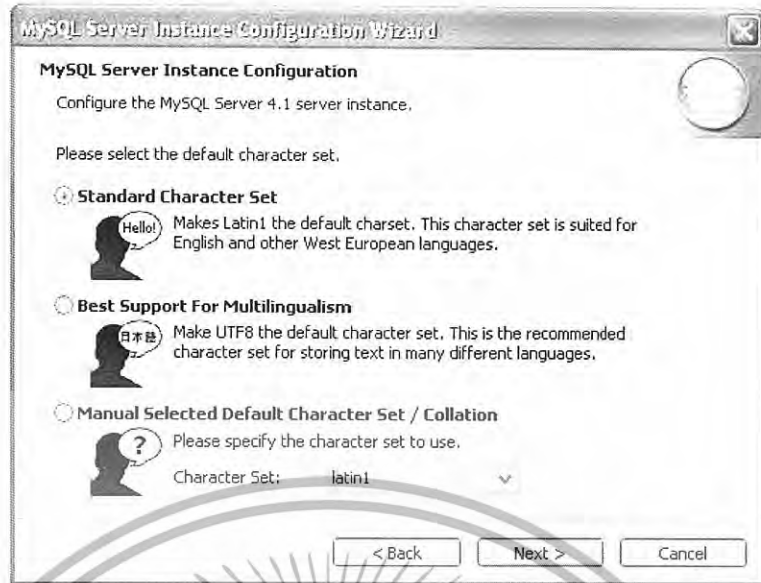


12. หน้า Please set the approximate number of concurrent connections to the server.
เลือก Decision Support (DSS)/OLAP จากนั้นคลิกปุ่ม Next



13. หน้า Please set the networking options. ทำการเลือก Enable TCP/IP Networking
ให้ Port Number เป็น 3307 จากนั้นคลิกปุ่ม Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

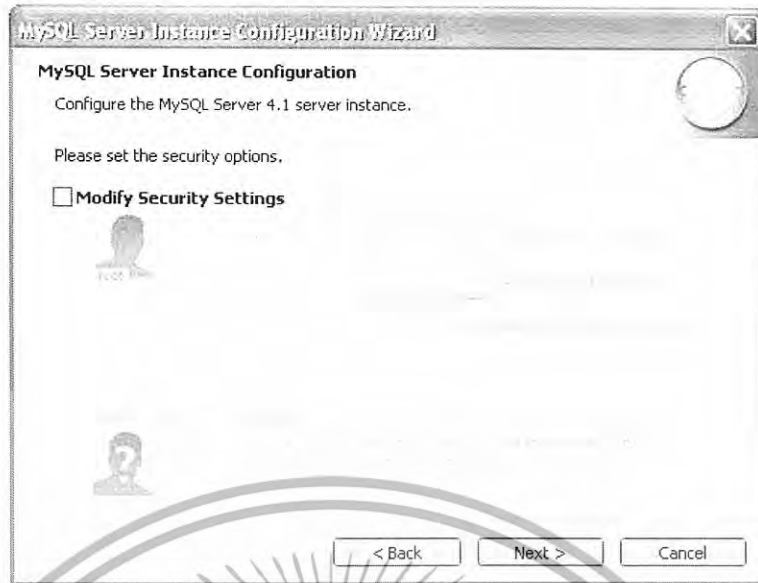


14. หน้า Please select the default character set. เลือก Standard Character Set จากนั้นคลิกปุ่ม Next



15. หน้า Please set the Windows options. เลือกทำการ Install As Windows Service ตั้งค่า Service Name เป็น MySQL4 เลือกทำ Launch the MySQL Server automatically จากนั้นคลิกปุ่ม Next

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

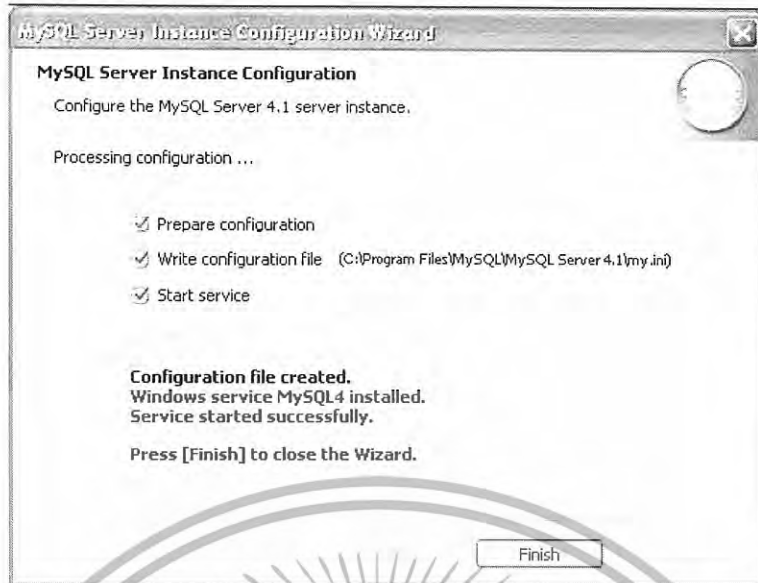


16. หน้า Please set the security options. ทำการ Disable ที่ Modify Security Settings จากนั้นคลิกปุ่ม Next



17. หน้า Ready to execute คลิกปุ่ม Execute

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



18.หน้า Processing Configuration คลิกปุ่ม Finish



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

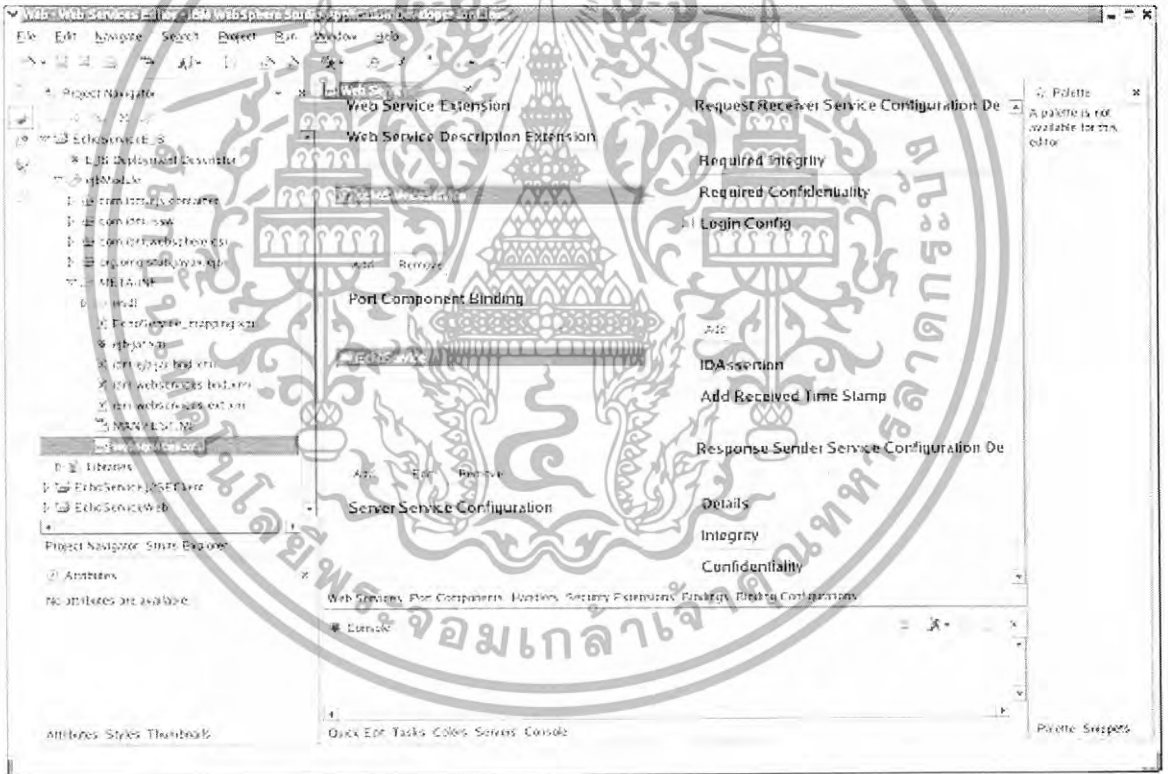
ภาคผนวก ข

การคอนฟิก WS-Security

1. ขั้นตอนการคอนฟิก User Name Tokens

การคอนฟิกฝั่งเซิร์ฟเวอร์

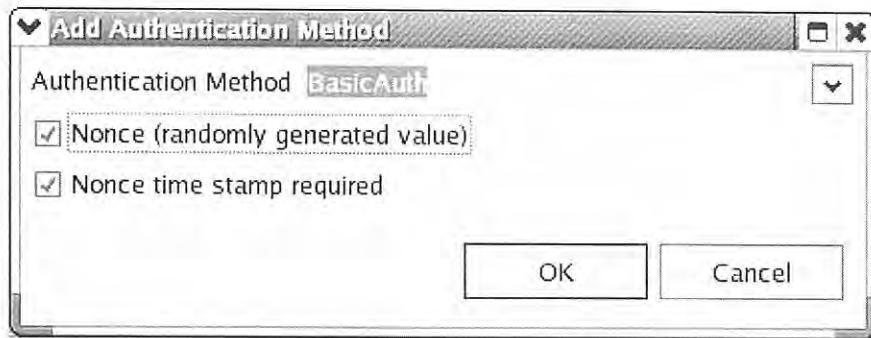
1. ใน Web perspective, กดเลือกเซอร์วิสที่ต้องการทำ security แล้ว browse จนเจอไฟล์ xml ดังตัวอย่าง directory => META-INF => wsdl => webservicex.xml
2. เปลี่ยนไปยังแท็บ Security Extensions
3. เปิดส่วน **Login Config** section
4. คลิก **Add** และเลือก **BasicAuth** จาก drop-down list.



รูปที่ ข.1 การคอนฟิก username token ฝั่งเซิร์ฟเวอร์

5. ในไดอะล็อก Add Authentication Method, ให้เลือกทั้ง Nonce ทั้งคู่ หลังจากนั้นคลิก OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

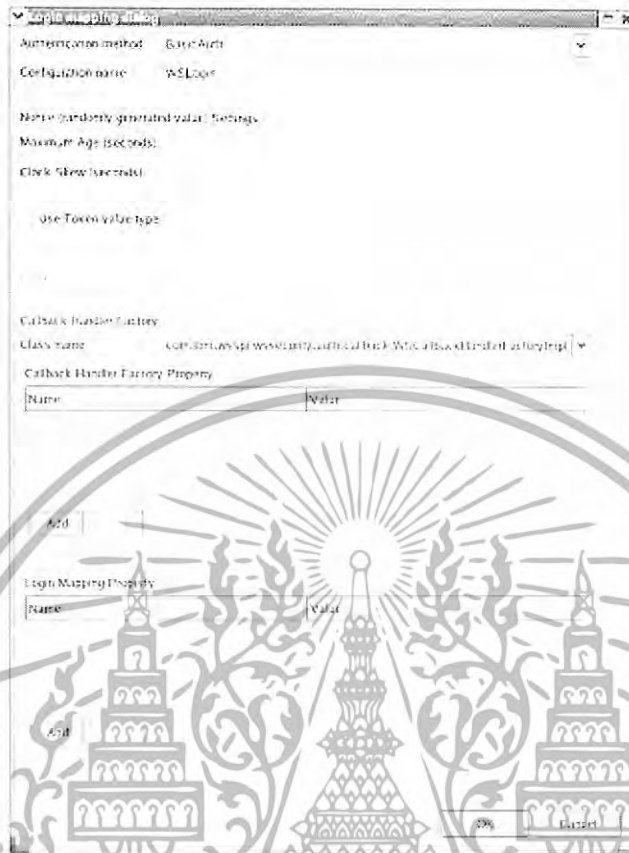


รูปที่ ข.2 การเพิ่ม Authentication method

6. เปลี่ยนไปยังแท็บ Binding Configurations และคลิกที่ add ที่อยู่ในส่วนของ **Login Mapping**
7. ในส่วนไดอะล็อก Login Mapping ให้ทำตามขั้นตอนดังนี้ :
 1. ระบุ **BasicAuth** สำหรับ **Authentication method**
 2. ใส่ configuration name ของ WSLLogin (case sensitive)
 3. Set **Callback handler** ให้เป็น `com.ibm.wsspi.wssecurity.auth.callback.WSCallbackHandlerFactoryImpl`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ลบข้อมูลอื่นๆที่เป็นค่า default ออก หลังจากนั้นคลิก OK



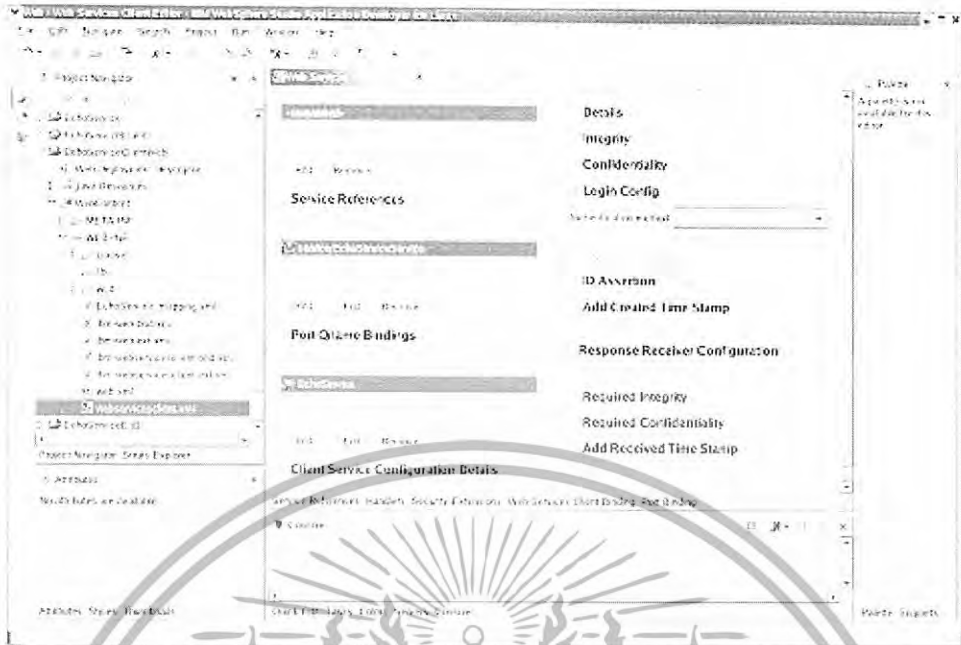
รูปที่ น.3 การ Configure server login mapping

8. Save การคอนฟิก แล้วเลือก ไปที่เพื่อทำการ restart **EchoService => Restart Project.**

การคอนฟิกฝั่งไคลเอนต์

1. ใน Web perspective, กดเลือกเซอร์วิสที่ต้องการทำ security แล้ว browse จนเจอไฟล์ xml **directory => Web Content => WEB-INF => wsdl > webservicessclient.xml.**
2. เปลี่ยนไปยังแท็บ Security Extensions
3. ในส่วนของ **Login Config** , เลือก **BasicAuth** จาก drop-down list ในฟิลด์ **Authentication method**, และกดเลือก **Nonce** ทั้งสองแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.4 หน้าการ Configure client login authentication

4. เปลี่ยนไปยังแท็บ Port Binding แล้วคลิก **Enable** ในส่วนของ **Login Binding**
5. ในไดอะล็อก Login Binding, ให้ทำตามขั้นตอน ดังนี้ :
 1. เลือก **BasicAuth** สำหรับ **Authentication method**
 2. เปลี่ยน **Callback handler** ให้เป็นจาก drop-down list
com.ibm.wsspi.wssecurity.auth.callback.NonPromptCallbackHandler
 3. ใส่ user ID และ password แล้วคลิก OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



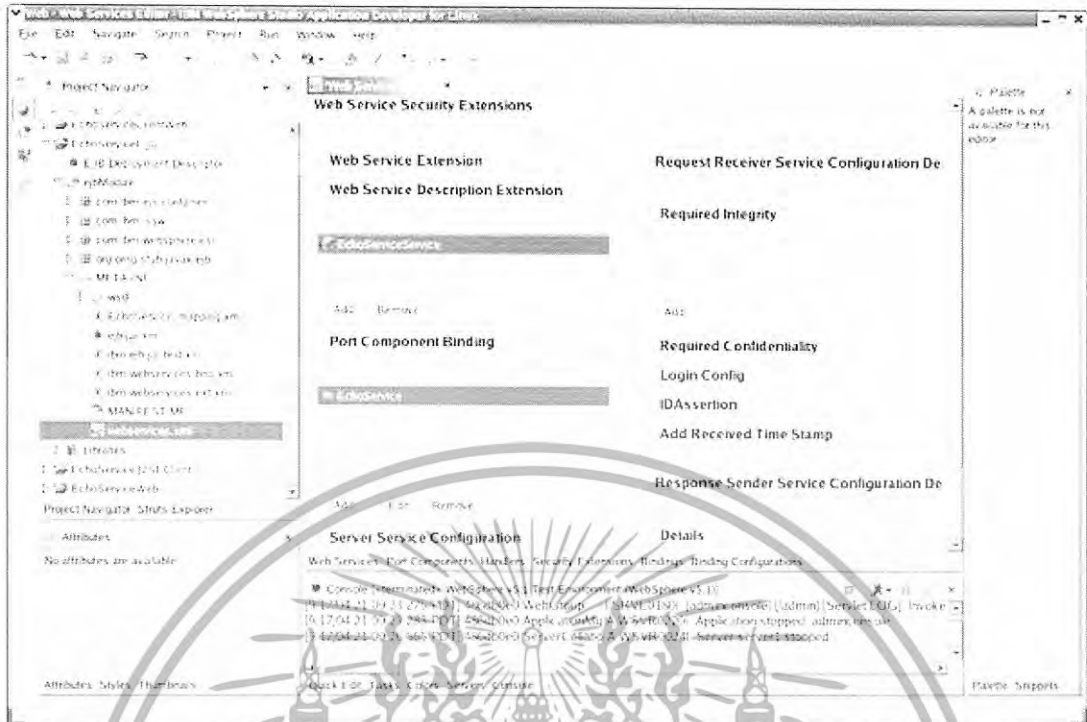
รูปที่ ข.5 หน้าการ Configure client login binding

2. การคอนฟิก Digital signatures

การคอนฟิกที่ฝั่งเซิร์ฟเวอร์

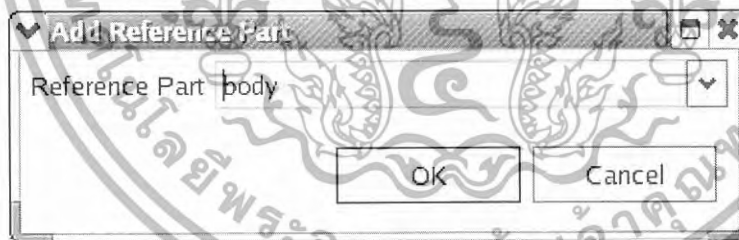
1. ใน Web perspective, เข้าไปแก้ไขในไฟล์ webservices.xml ซึ่งอยู่ใน **directory => ejbModule => META-INF => webservices.xml.**
2. เปลี่ยนไปยังแท็บ Security Extensions

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.6 หน้า Required integrity

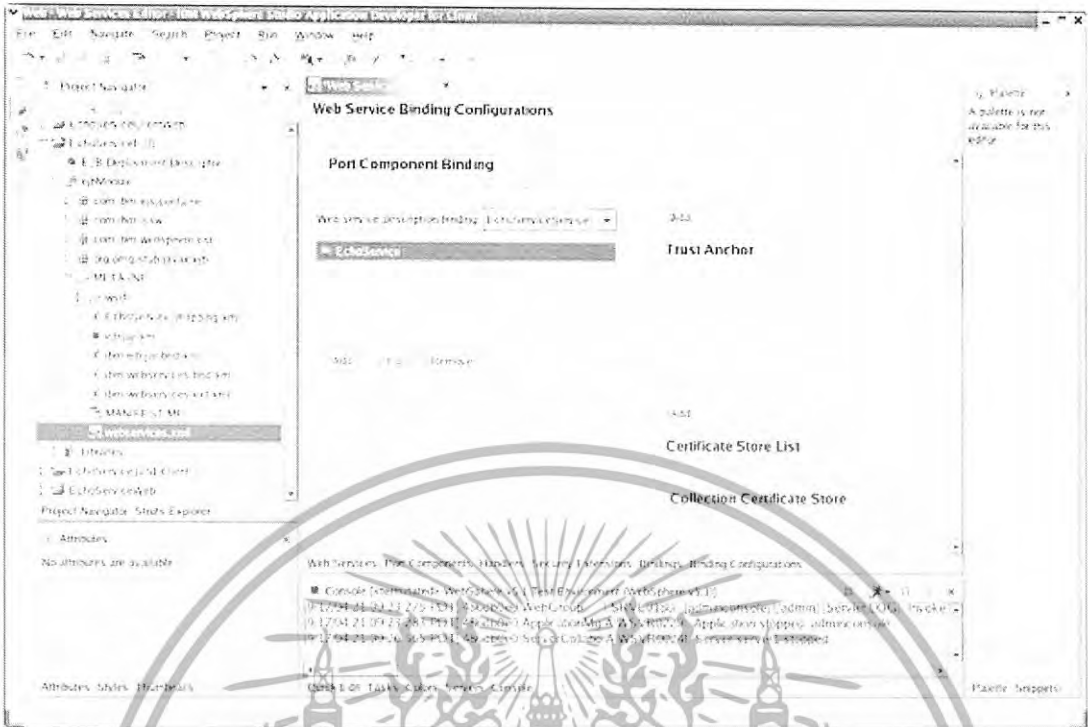
3. ในส่วนของ **Required Integrity** ให้คลิก **Add** เพื่อ add body and securitytoken. ซึ่งการ add ต้องทำสองครั้ง



รูปที่ ข.7 หน้า Add reference part dialog

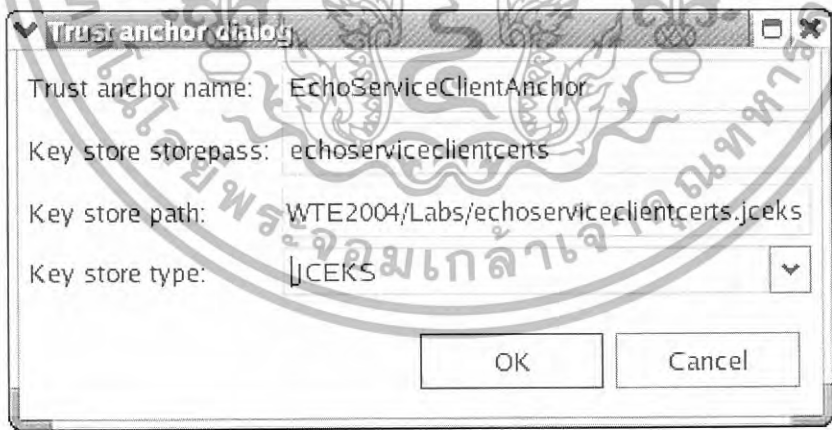
4. เปลี่ยนไปยัง Binding Configurations
5. ในส่วนของ **Trust Anchor**, คลิก **Add** เพื่อเพิ่ม trust anchor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.8 หน้า Trust anchor

- 6. ในไดอะล็อก trust anchor, ให้ใส่ keystore ซึ่งเก็บ trusted signer certificates แสดงดังรูปด้านล่าง

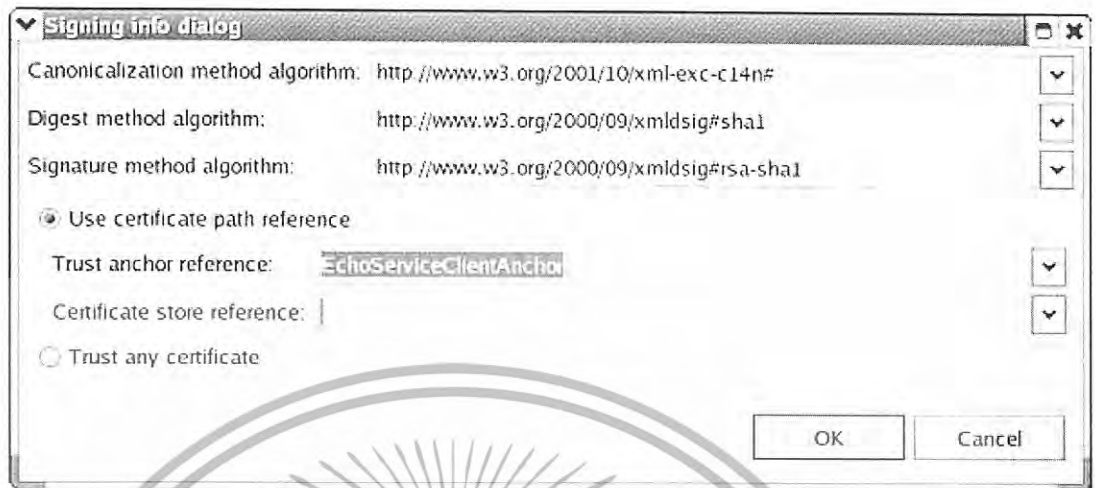


รูปที่ ข.9 หน้า Trust Anchor dialog

- 7. ในส่วนของ **Signing Information** ให้คลิก **Add**
- 8. ในไดอะล็อกของ Signing Info ให้ใช้ default algorithms ถ้าคุณไม่ทราบ algorithm อื่นที่แตกต่างออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. เลือก **Use certificate path reference**, แล้วเลือก new trust anchor หลังจากนั้นคลิก **OK**.

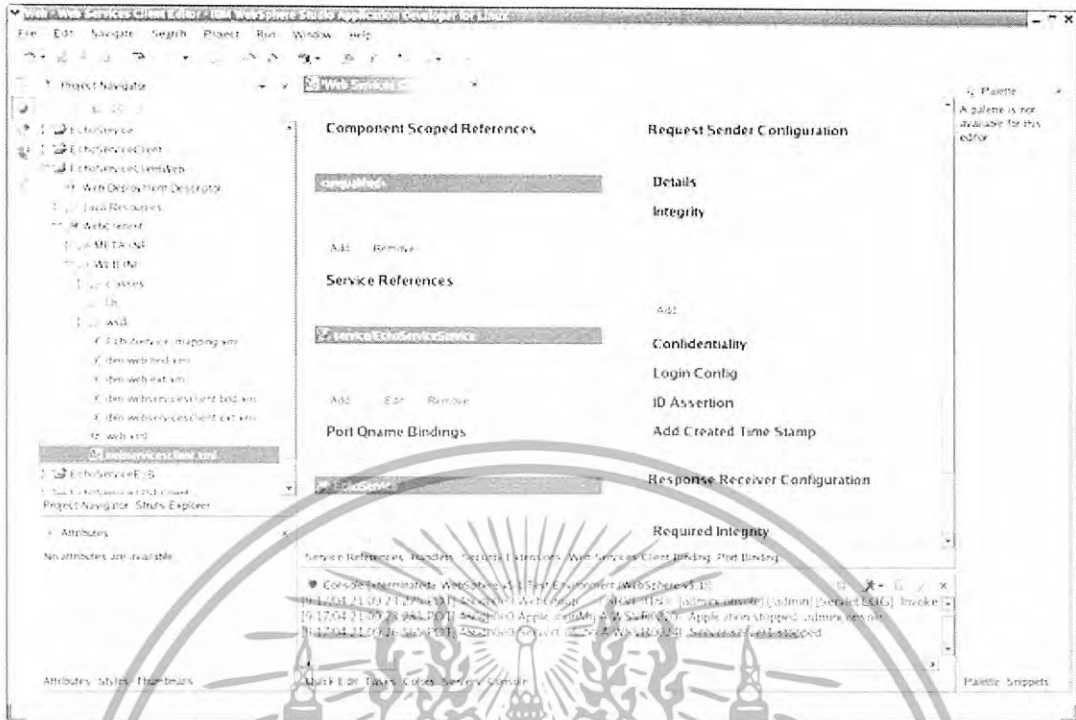


รูปที่ ข.10 หน้า Signing Information dialog

การคอนฟิกที่ฝั่งไคลเอนต์

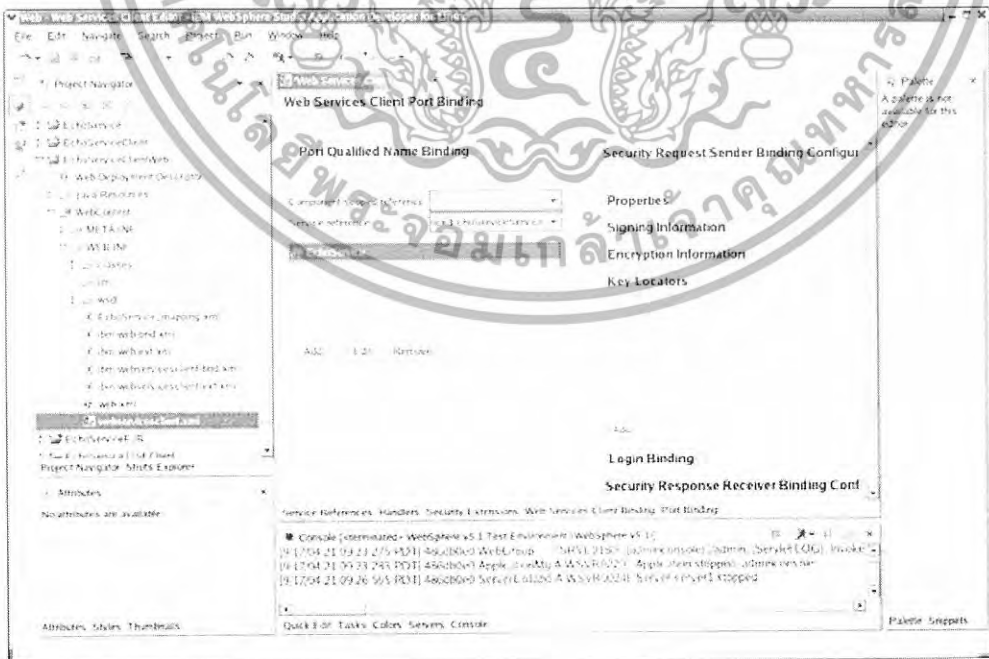
1. ใน Web perspective ให้เปิด ไฟล์ webservicessclient.xml เพื่อทำการแก้ไข โดยเข้าไปที่ **directory => Web Content => WEB-INF => webservicessclient.xml**.
2. เลือกแท็บ Security Extensions
3. ในส่วนของ **Integrity** ให้คลิก **Add** เพื่อเพิ่มส่วน body และ securitytoken.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.11 หน้า Integrity

4. เปลี่ยนไปยังแท็บ Port Binding
5. ในส่วนของ **Key Locators** field, ให้ add ที่อยู่ของ key แล้วคลิก add



รูปที่ ข.12 หน้า Key locators

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ในไดอะล็อกของ Key Locator, ให้ทำการตรวจสอบ **Use key store** และระบุ keyStore storepass (password), path and type.
7. ในฟิลด์ของ **Key** ให้เพิ่ม key reference และทำการระบุ alias และ key password ของ key ใน keystore
8. คลิก **OK**.



รูปที่ ข.13 Key Locator dialog

9. ในส่วนของ **Signing Information** คลิก **Enable**.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ในไดอะล็อกของ Signing Info ให้ใส่ string ที่คุณใช้สำหรับเป็นชื่อของ key เช่น **Signing key name**
11. ในฟิลด์ของ **Signing key locator** ให้ใส่ชื่อของ location ที่ได้ทำการสร้างก่อนหน้า หลังจากนั้นคลิก OK



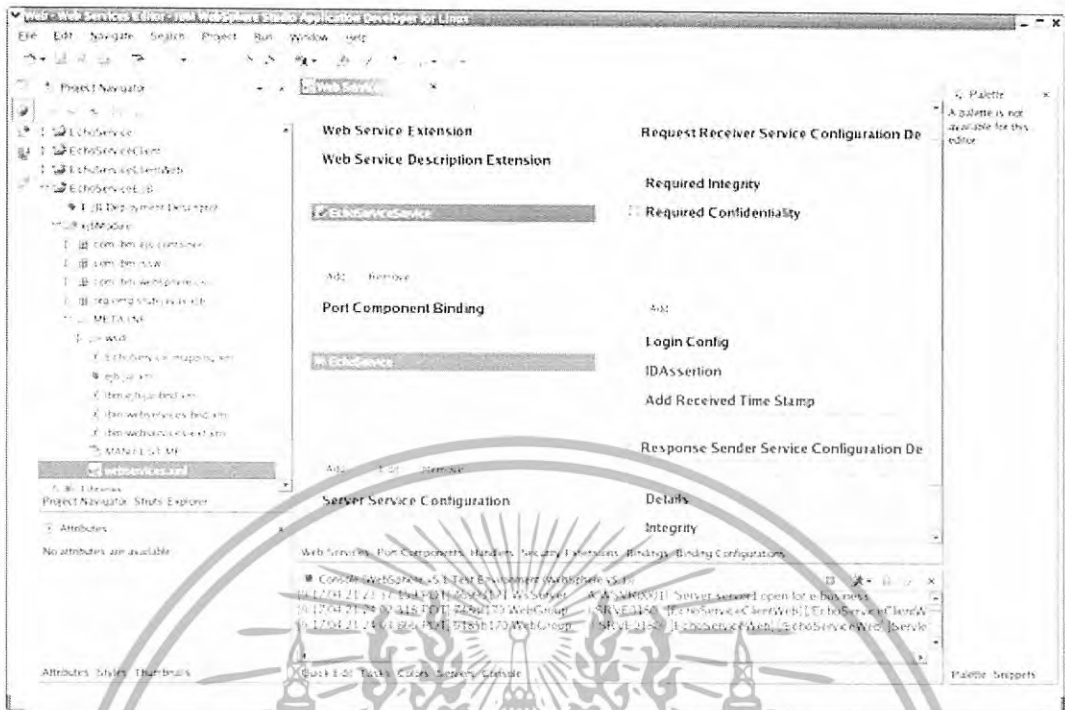
รูปที่ ข.14 Signing Information dialog

3. ขั้นตอนการคอนฟิก Encryption

การคอนฟิกฝั่งเซิร์ฟเวอร์

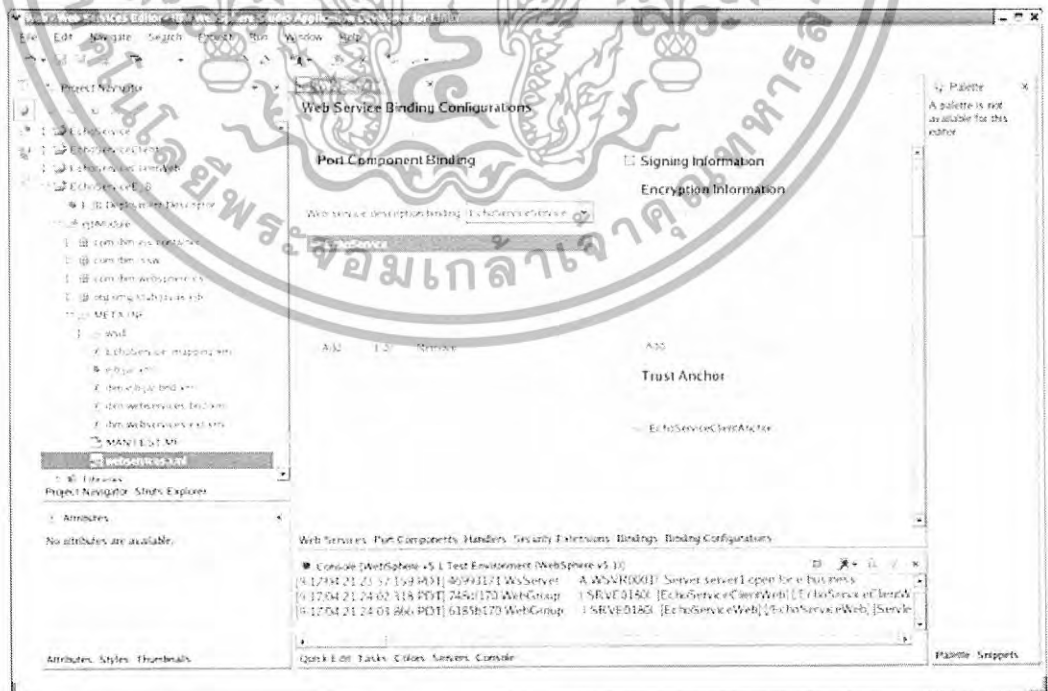
1. ใน Web perspective ทำการแก้ไขไฟล์ webservices.xml โดยเลือกเข้าไปที่ **directory => ejbModule => META-INF => webservices.xml**
2. เลือกแท็บ Security Extensions
3. ในส่วนของ **Required Confidentiality** คลิก **Add** เพื่อเพิ่ม bodycontent และ usernametoken

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.15 Required confidentiality

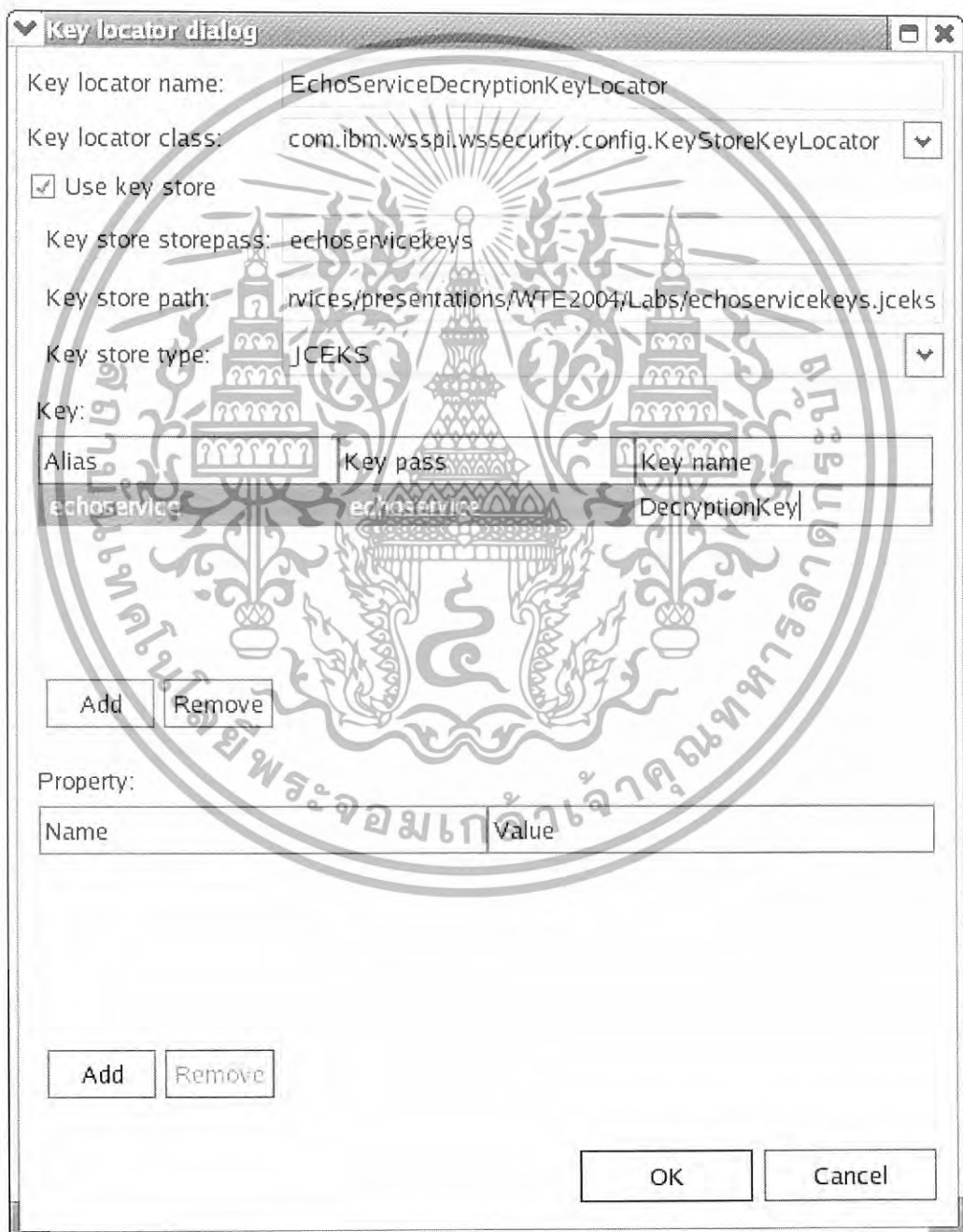
4. เปลี่ยนไปยังแท็บ Binding Configurations
5. ในส่วนของ **Key Locators** คลิก **Add** เพิ่มเพิ่ม key locator



รูปที่ ข.16 Key locator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

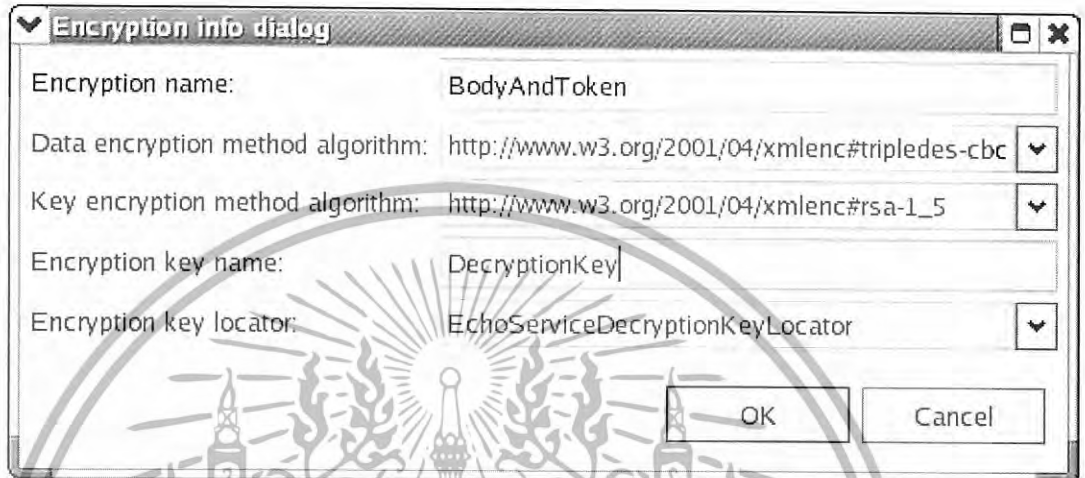
6. ในไดอะล็อกของ Key Locator ให้ระบุ **Key locator name**. ในตัวอย่างนี้จะใช้ EchoServiceDecryptionKeyLocator
7. สำหรับ **Key locator class** ระบุ ...KeyStoreKeyLocator
8. เลือก **Use key store** และทำการเพิ่มค่าที่เหมาะสมสำหรับ keystore แสดงดังภาพด้านล่าง
9. ในส่วนของ **Key** section คลิก **Add** เพื่อเพิ่ม key ลงใน list และทำการ set **Alias** และ **Key pass** fields ให้เป็นค่าที่ตรงกับใน keystore และระบุ **Key name** ซึ่งในตัวอย่างจะใช้ DecryptionKey หลังจากนั้นคลิก OK



รูปที่ ข.17 Key locator dialog

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เฉพาะภายในเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ในส่วนของ **Encryption Information** section, คลิก **Add** เพื่อเพิ่ม entry ต่างๆ
11. ในไดอะล็อกของ Encryption Info ให้ลบ default algorithms ออก, และใส่ **Encryption key name** and **Encryption key locator** ของคุณที่คุณพึงระบุไว้ (DecryptionKey and EchoServiceDecryptionKeyLocator)

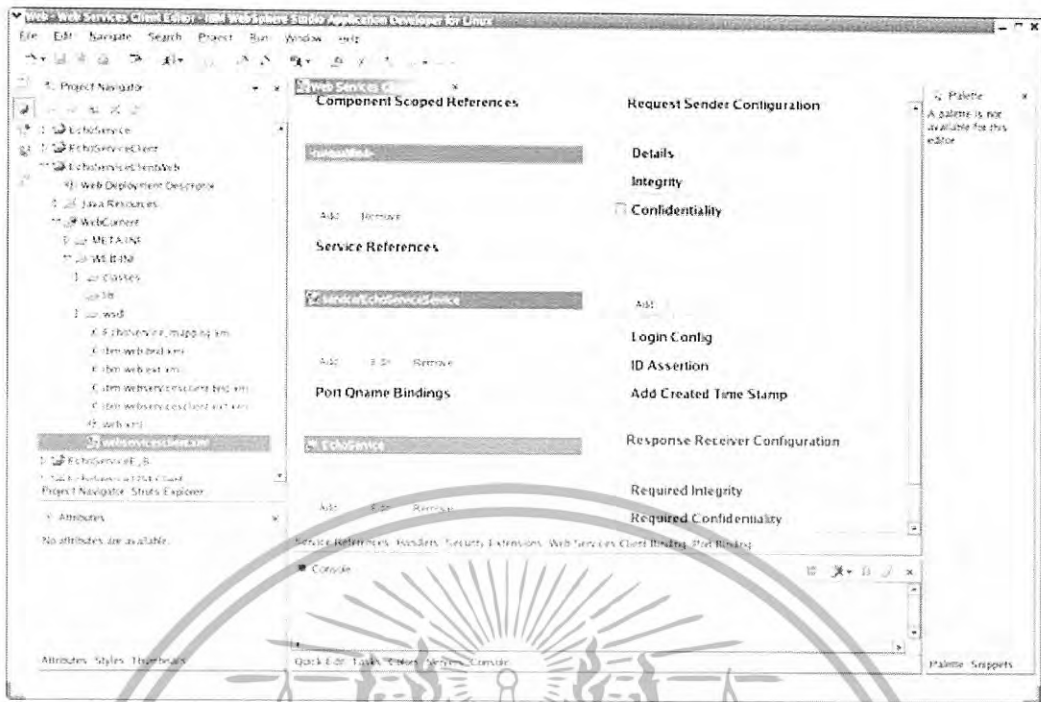


รูปที่ ข.18 Encryption Information dialog

การคอนฟิกฝั่งเซิร์ฟเวอร์

1. ใน Web perspective ให้เปิดไฟล์ webservicessclient.xml เพื่อทำการแก้ไข โดยเข้าไปยัง **directory => Web Content => WEB-INF => webservicessclient.xml**
2. เปลี่ยนไปยัง Security Extensions tab
3. ในส่วนของ **Confidentiality** section คลิก **Add** เพื่อเพิ่ม bodycontent และ usernametoken

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.19 Confidentiality

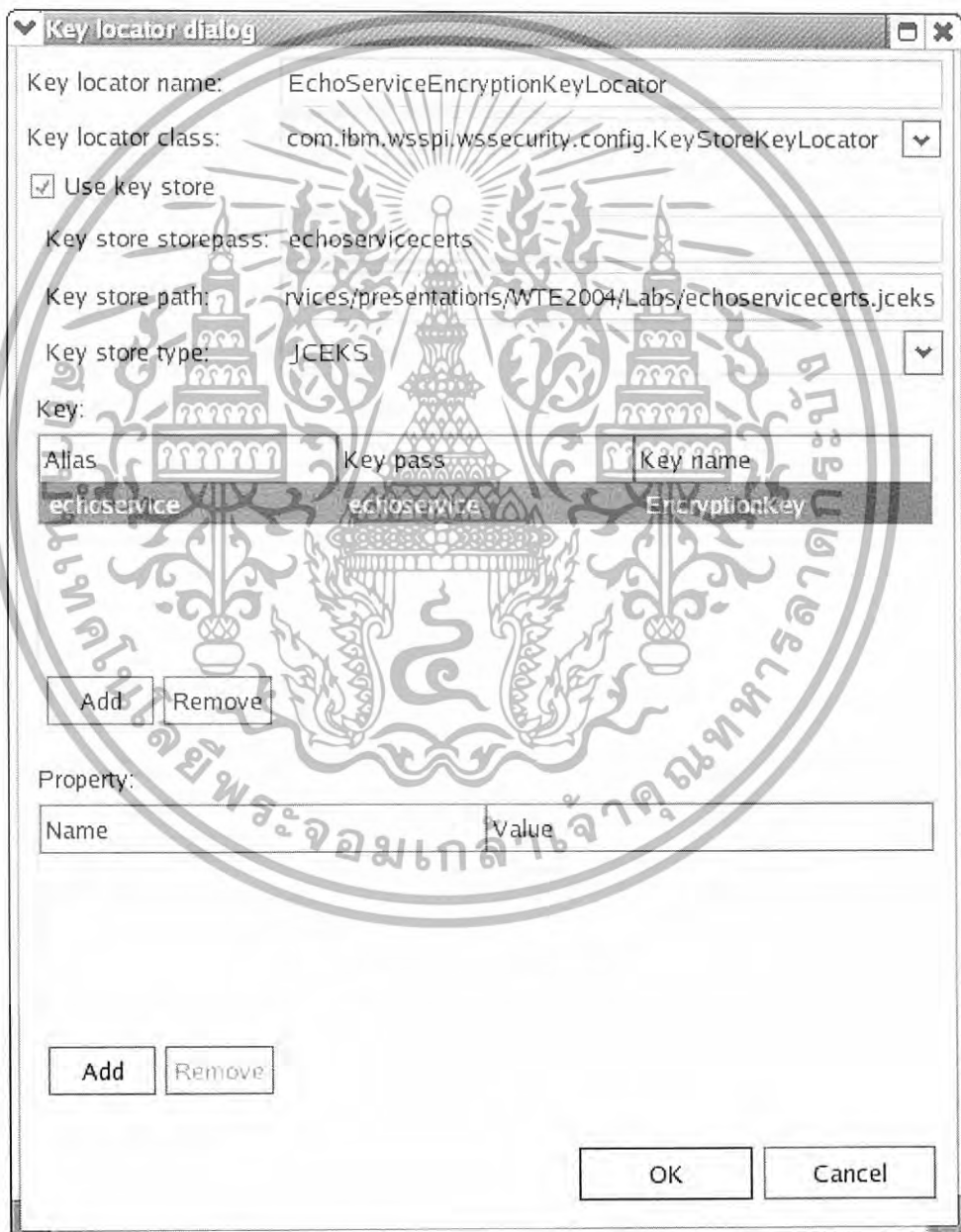
4. เปลี่ยนไปยัง Port Binding tab.
5. ในส่วนของ **Key Locators** section คลิกเพื่อเพิ่ม Key locator



รูปที่ ข.20 Key locator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

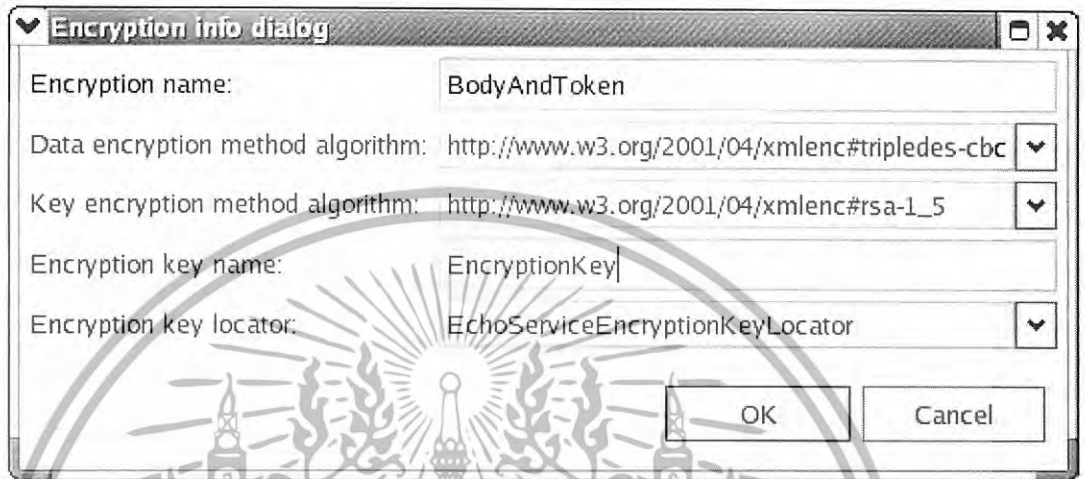
6. ใน Key Locator dialog ให้ระบุ **Key locator name** ในตัวอย่างจะใช้ EchoServiceEncryptionKeyLocator
7. สำหรับ **Key locator class** ให้ระบุ ...KeyStoreKeyLocator
8. เลือก **Use key store** และเพิ่มค่าที่เหมาะสมของ keystore, แสดงดังตัวอย่างด้านล่าง
9. ในส่วนของ **Key** section, คลิก **Add** เพื่อเพิ่ม key ใน list และ set **Alias** และ **Key pass** fields ให้เป็นค่าที่มาจาก keystore, และระบุ **Key name** ในตัวอย่างจะใช้ EncryptionKey หลังจากนั้นคลิก OK



รูปที่ ข.21 Key Locator dialog

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ในส่วนของ **Encryption Information** section, คลิก **Enable**.
11. ใน Encryption Info dialog, ให้ลบค่า defaults ของ algorithms, และใส่ **Encryption key name** and **Encryption key locator** ที่คุณ ได้กำหนดไว้ข้างต้น (EncryptionKey and EchoServiceEncryptionKeyLocator)



รูปที่ ๖.22 Encryption Information dialog

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้