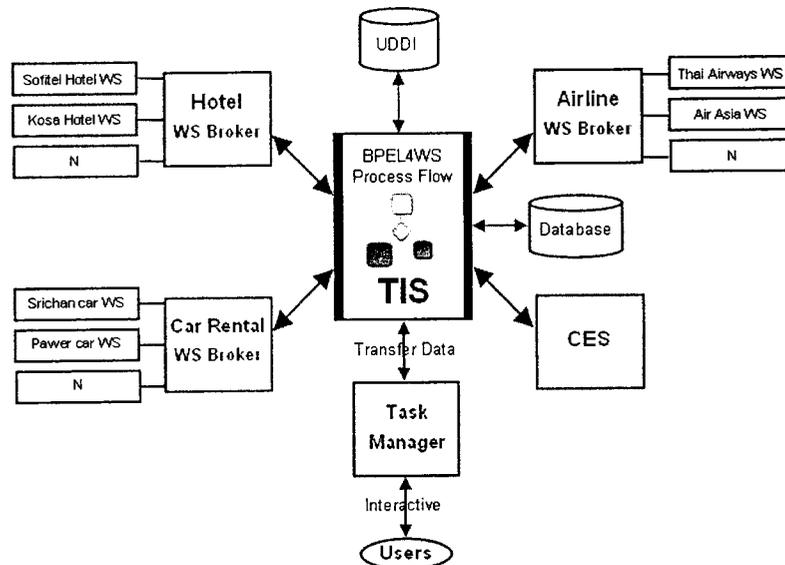


บทที่ 3 วิธีการดำเนินการวิจัย

1. ออกแบบสถาปัตยกรรมของธุรกิจการท่องเที่ยวแบบครบวงจรโดยใช้องค์ประกอบตัวบริการเว็บ

ออกแบบและพัฒนาระบบสถาปัตยกรรมของระบบธุรกิจการท่องเที่ยวแบบครบวงจร โดยใช้องค์ประกอบตัวบริการเว็บ เพื่อให้ตัวบริการเว็บของธุรกิจบริการการท่องเที่ยวต่างๆ สามารถติดต่อสื่อสารกันและทำงานร่วมกันได้อย่างอัตโนมัติ ซึ่งเป็นการอำนวยความสะดวกให้กับลูกค้าที่ใช้บริการโดยสามารถได้รับบริการข้อมูลการท่องเที่ยวและเลือกใช้บริการต่างๆ ได้โดยผ่านระบบสารสนเทศการท่องเที่ยว (Tourism Information System) หรือระบบ TIS



ภาพที่ 15 สถาปัตยกรรมของระบบการท่องเที่ยวแบบครบวงจร โดยวิธีแบบจำลององค์ประกอบตัวบริการเว็บ

ในแต่ละองค์ประกอบสามารถอธิบายหน้าที่ได้ดังต่อไปนี้

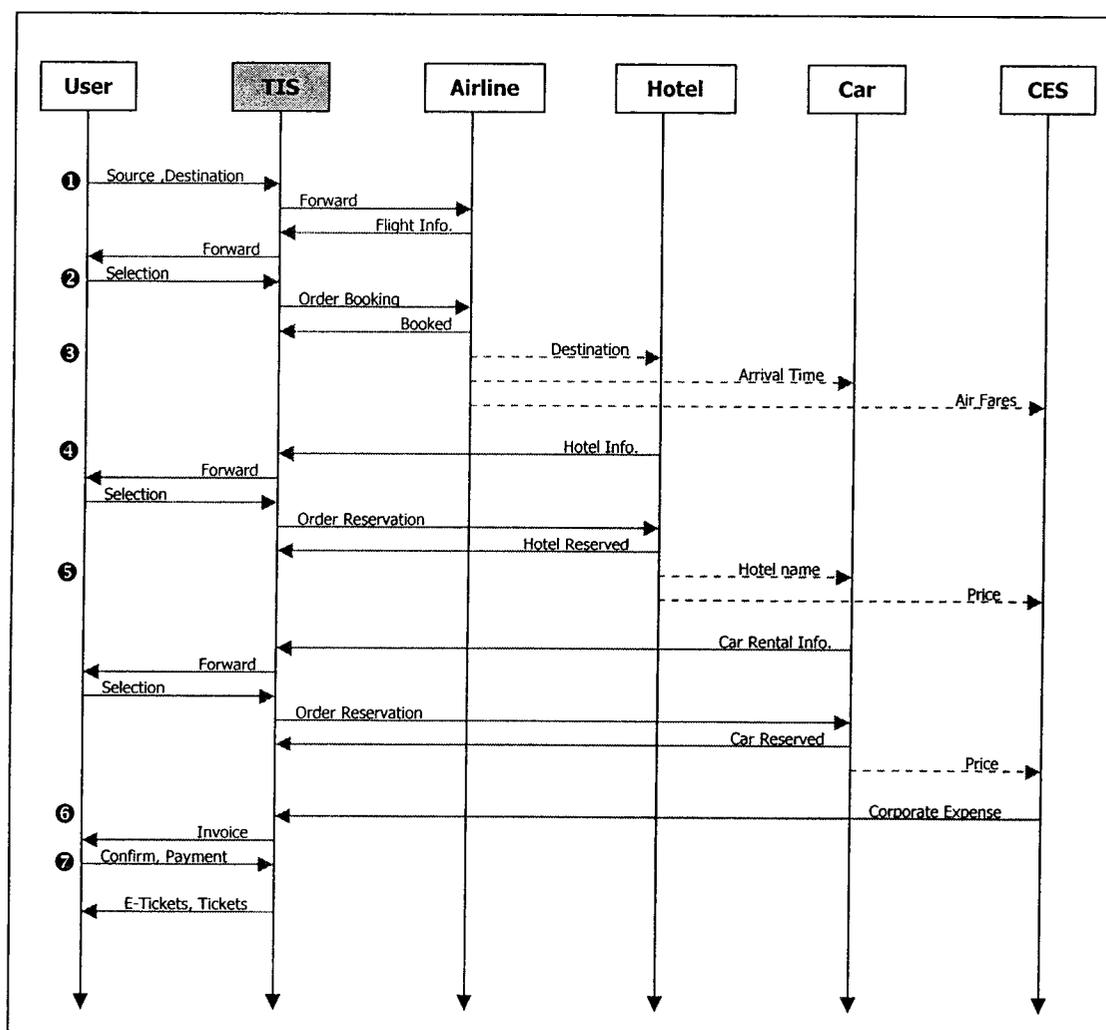
- ระบบสารสนเทศการท่องเที่ยว หรือ ระบบ TIS ทำหน้าที่เป็นศูนย์กลางในการประสานงานการทำงานร่วมกันของตัวบริการเว็บต่างๆ ของธุรกิจบริการการท่องเที่ยว
- ตัวแทนบริการท่องเที่ยวต่างๆ (Web Service Broker) ทำหน้าที่เป็นตัวแทนในการเรียกไปยังตัวบริการเว็บในกลุ่มบริการของตน เพื่อรวบรวมผลลัพธ์ส่งไปยังระบบ TIS เพื่อส่งข้อมูลผ่านไปยังตัวบริหารงานให้ลูกค้าเลือกรายการ
- ตัวบริหารงาน (Task Manager Service) ทำหน้าที่ประสานงานระหว่างตัว BPEL Process กับลูกค้าหรือผู้ใช้
- ตัวบริการจัดการค่าใช้จ่าย (Corporate Expense Service หรือ CES) ทำหน้าที่สรุปค่าใช้จ่ายที่เกิดขึ้นในระบบทั้งหมด และส่งกลับไปยัง TIS เพื่อประมวลผลต่อไป

- ลูกค้าหรือผู้ใช้งานทั่วไป (Users) ทำหน้าที่เป็นผู้เรียกใช้บริการ (Service requester) จากระบบ TIS โดยลูกค้าสามารถเลือกได้ว่าจะใช้บริการใดบ้าง
- Database ทำหน้าที่เก็บ Transaction ที่เกิดขึ้นจากการทำงานของระบบ TIS
- UDDI เก็บรวบรวมการให้บริการของตัวบริการเว็บต่าง ๆ เพื่อให้ระบบ TIS ค้นหาไฟล์ WSDL เพื่อประกอบการสร้างองค์ประกอบตัวบริการเว็บ

2. ออกแบบลำดับขั้นตอนการดำเนินงานทางธุรกิจของธุรกิจบริการท่องเที่ยวแบบครบวงจร

การออกแบบกระบวนการทางธุรกิจของระบบธุรกิจบริการท่องเที่ยวแบบครบวงจร จะประกอบไปด้วยธุรกิจต่าง ๆ ที่ลูกค้าสามารถเลือกที่จะใช้บริการแบบครบวงจรได้ และเป็นธุรกิจที่ต้องทำงานร่วมกัน เช่น Airline, Hotel, Car Rental และ Corporate expense service ซึ่งทุกธุรกิจถูกออกแบบเป็นตัวบริการเว็บ ที่มีตัวกลางในการเรียกใช้บริการต่าง ๆ จากตัวบริการเว็บ ซึ่งได้แก่ ระบบ TIS โดยมีลำดับขั้นตอนการทำงานดังแสดงในภาพที่

16



ภาพที่ 16 แสดงลำดับขั้นตอนการทำงานของธุรกิจบริการท่องเที่ยวต่างๆ

จากภาพที่ 16 อธิบายได้ว่าเป็นระบบการท่องเที่ยวที่จะประกอบไปด้วยธุรกิจต่าง ๆ ที่ต้องทำงานร่วมกัน ได้แก่ Airline, Hotel, Car Rental และ Corporate expense service ซึ่งทุกธุรกิจถูกออกแบบเป็นตัวบริการเว็บ (Web Service) โดยมีตัวกลางในการเรียกใช้บริการต่าง ๆ จากตัวบริการเว็บนั้น ๆ หรือที่เรียกว่าระบบ TIS และสามารถอธิบายเป็นขั้นตอนได้ดังนี้

(1) ลูกค้าร้องขอบริการท่องเที่ยวจากตัวแทน หรือ TIS โดยเริ่มจากค้นหาเที่ยวบิน ซึ่งลูกค้าจะต้องส่งข้อมูลเบื้องต้น เช่น ต้นทาง ปลายทาง และวันเวลาที่ต้องการเดินทาง จากนั้น TIS จะส่งข้อมูลไปยัง Airline Web Service ทำการค้นหาและแสดงรายการข้อมูลเที่ยวบินกลับมาให้ลูกค้าทราบ

(2) ลูกค้าเลือกว่าต้องการไปกับสายการบินใด ก็ที่นั่น เลือกชั้นแบบธรรมดาหรือแบบธุรกิจ จากนั้น TIS ก็ส่งข้อมูลไปยัง Airline Web Service เพื่อทำการสำรองที่นั่ง

(3) ข้อมูลปลายทาง เวลาที่มาถึง และค่าโดยสารเครื่องบิน จะถูกส่งอัตโนมัติไปที่ Web Service ของ Hotel, Car Rental, Corporate Expense Service (CES) ตามลำดับ

(4) เมื่อ Hotel Web Service ได้รับข้อมูลปลายทาง ก็จะทำการค้นหาโรงแรมในบริเวณที่ลูกค้าต้องการไปออกมา พร้อมกับประเภทโรงแรมและราคามาให้ลูกค้าเลือกด้วย เช่น ปลายทางคือจังหวัดขอนแก่น ก็จะทำการค้นหาโรงแรมที่อยู่ภายในจังหวัดขอนแก่น

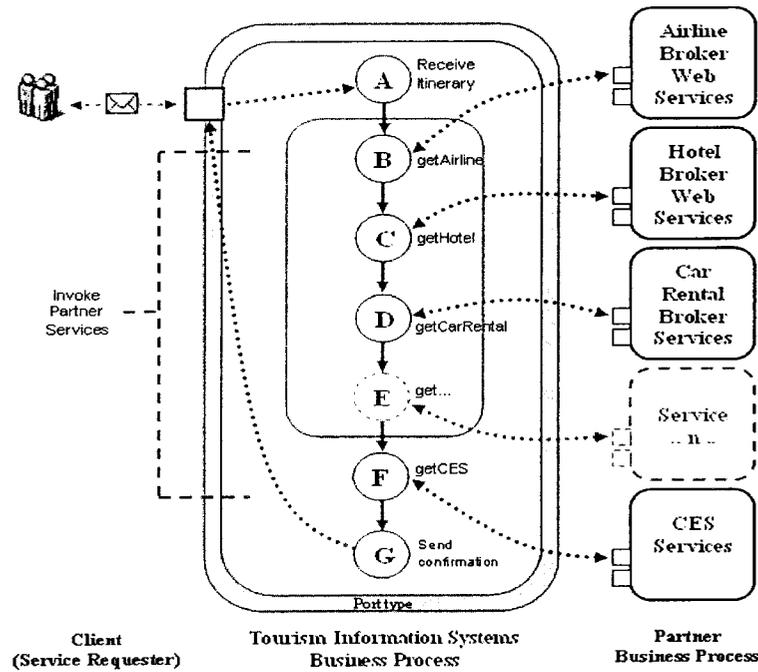
(5) เมื่อเลือกโรงแรมได้แล้ว TIS ก็จะส่งราคาค่าเช่าห้องพักโรงแรมไปยัง CES และส่งชื่อโรงแรมไปยัง Car Rental Web Service เพื่อแสดงประเภทของรถมาให้ลูกค้าเลือก โดยที่ Car Rental Web Service จะทราบอัตโนมัติแล้วว่าต้องไปรับลูกค้าที่ Airport ไต เวลาใด และจะต้องไปส่งลูกค้าที่โรงแรมใด เพราะได้รับข้อมูลมาแล้วจาก TIS

(6) Corporate Expense Service (CES) ทำการสรุปค่าใช้จ่ายทั้งหมด ซึ่งลูกค้าจะต้องเป็นสมาชิกอยู่ในบริการจ่ายเงินนี้ด้วย เพื่อสรุปส่งเป็น Invoice และ TIS ก็จะแสดงรายการค่าใช้จ่ายนั้นให้ลูกค้าทราบด้วย ลูกค้าทำการยืนยันรายการที่เลือกไป และชำระเงิน จากนั้น TIS ก็จะนำส่ง Tickets หรือจะส่งพิมพ์ตั๋วผ่านระบบ E-Tickets ทาง Web Browser ได้อีกด้วย

2.1 การออกแบบกระบวนการทางธุรกิจของระบบในระดับบน : Level 0

การออกแบบกระบวนการทางธุรกิจในระดับบนหรือ Level 0 เป็นการแสดงการทำงานและการประกอบร่วมกันของตัวบริการเว็บของกลุ่มธุรกิจต่าง ๆ ได้แก่ กลุ่มธุรกิจจองตั๋วเครื่องบิน กลุ่มธุรกิจโรงแรม และกลุ่มธุรกิจรถเช่า ซึ่งสถาปัตยกรรมของระบบสามารถรองรับกับกลุ่มธุรกิจที่จะเกิดขึ้นในอนาคตได้ เช่น กลุ่มธุรกิจร้านอาหาร หรือธุรกิจสปา เป็นต้น โดยมีตัวกลางในการประสานงานระหว่างกลุ่มธุรกิจต่าง ๆ เรียกว่าระบบ TIS ซึ่งเป็นตัวจัดการกระบวนการทางธุรกิจ (Business Process) ที่จะเกิดขึ้น ทำให้การติดต่อสื่อสารและการรับส่งค่าระหว่างกลุ่มธุรกิจเป็นไปอย่างอัตโนมัติ

กระบวนการทางธุรกิจจะเริ่มตั้งแตรับข้อมูลความต้องการเบื้องต้นจากลูกค้า (Activity A) จากนั้นทำการส่งข้อมูลต่อไปยัง Airline Broker Web Services (Activity B) ซึ่งจะมีการประมวลผลจาก Activity A ถึง Activity G ตามลำดับ ดังภาพที่ 17 โดยเริ่มแรกลูกค้าสามารถเลือกได้ว่าจะใช้บริการใดบ้าง และจะไม่ใช้บริการใดบ้าง เช่น ลูกค้าเลือกบริการจองตั๋วเครื่องบินและจองบริการจองโรงแรม แต่อาจไม่ได้เลือกบริการจองรถเช่า เป็นต้น



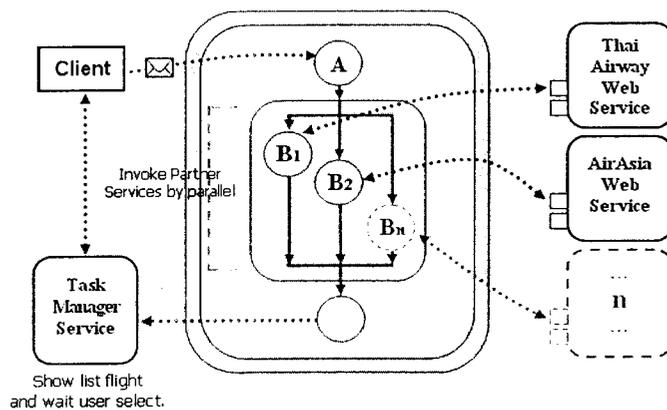
ภาพที่ 17 แผนภาพกระบวนการทางธุรกิจของระบบการท่องเที่ยวแบบครบวงจรในระดับบน : Level 0

2.2 การออกแบบกระบวนการทางธุรกิจของตัวแทนการบริการในระดับล่าง : Level 1

การออกแบบกระบวนการทางธุรกิจในระดับล่างหรือ Level 1 แบ่งตามธุรกิจบริการต่างๆ ออกเป็น 4 กลุ่ม ได้แก่ Airline Broker Web Services, Hotel Broker Web Services, Car Rental Broker Web Services และ Corporate Expense Service สามารถอธิบายในแต่ละกลุ่มได้ดังนี้

2.2.1 ตัวแทนบริการจองตั๋วเครื่องบิน (Airline Broker Web Services)

เป็นการออกแบบกระบวนการทางธุรกิจของตัวแทนบริการจองตั๋วเครื่องบิน ซึ่งจะเรียกไปยัง Partner อื่นๆ ประกอบไปด้วยตัวบริการเว็บของธุรกิจการจองตั๋วเครื่องบิน ได้แก่ ThaiAirway web service และ AirAsia web service และพร้อมสนับสนุนสำหรับการเกิดขึ้นของธุรกิจจองตั๋วเครื่องบินของบริษัทอื่นๆ ในอนาคต ดังภาพที่ 18



ภาพที่ 18 แผนภาพการทำงานของตัวแทนการบริการจองตั๋วเครื่องบินในระดับล่าง : Level 1

การประมวลผลจะเริ่มตั้งแต่รับค่าชื่อลูกค้า (Customer Name) สถานีต้นทาง (Source) สถานีปลายทาง (Destination) วันที่เดินทาง (Departure Date) และจำนวนผู้โดยสาร (Passenger) จาก Client (Activity A) หลังจากนั้นระบบจะประมวลผลโดยเรียกไปยังตัวบริการเว็บจองตั๋วเครื่องบินต่าง ๆ (Activity B) และส่งค่ารายการเที่ยวบินที่ได้ไปยังตัวบริหารงานหรือ Task Manager Service เพื่อแสดงรายการให้ลูกค้าได้เลือกผ่านทางเว็บเบราว์เซอร์ โดยค่าผลลัพธ์ที่ได้จากบริการนี้ก็จะถูกส่งต่อไปยังธุรกิจบริการอื่น ๆ ต่อไป เช่น ตัวแทนบริการโรงแรม เป็นต้น ค่าผลลัพธ์ที่ได้จากบริการนี้ ดังแสดงในตารางที่ 1

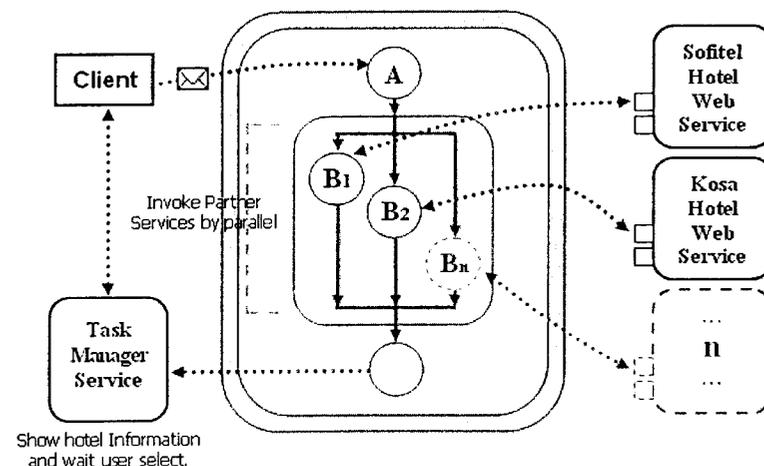
ตารางที่ 1 Input / Output of Airline Broker Web Services

Input	Output
1. Customer Name	1. <u>Customer Name</u>
2. Source	2. Source
3. Destination	3. <u>Destination</u>
4. Departure Date	4. Departure Date
5. Passenger	5. Passenger
	6. Company
	7. Flight Number
	8. Cabin Class
	9. Departing Time
	10. <u>Arriving Time</u>
	11. <u>Air Fares</u>

หมายเหตุ ค่าที่ขีดเส้นใต้คือค่าที่ถูกส่งไปประมวลผลยังตัวบริการเว็บอื่น

2.2.2 ตัวแทนบริการโรงแรม (Hotel Broker Web Services)

เป็นการออกแบบกระบวนการทางธุรกิจของตัวแทนบริการจองโรงแรมและห้องพัก ซึ่งจะเรียกไปยัง Partner อื่น ๆ ประกอบไปด้วยตัวบริการเว็บของธุรกิจโรงแรม ได้แก่ Sofitel Hotel web service และ Kosa Hotel web service และพร้อมสนับสนุนสำหรับการเกิดขึ้นของโรงแรมของบริษัทอื่น ๆ ในอนาคต ดังภาพที่ 19



ภาพที่ 19 แผนภาพการทำงานของตัวแทนการบริการโรงแรมในระดับล่าง : Level 1

การประมวลผลจะเริ่มตั้งแต่รับค่าชื่อลูกค้า (Customer Name) และสถานีปลายทาง (Destination) จาก Client หรือจากตัวแทนบริการจองตั๋วเครื่องบิน (Activity A) หลังจากนั้นระบบจะประมวลผล โดยเรียกไปยังตัวบริการเว็บของโรงแรมต่าง ๆ (Activity B) และส่งค่ารายชื่อโรงแรมและรายการห้องพักที่ได้ไปยังตัวบริหารงานหรือ Task Manager Service เพื่อแสดงรายการให้ลูกค้าได้เลือกผ่านทางเว็บเบราว์เซอร์ โดยผลลัพธ์ที่ได้จากบริการนี้ ดังแสดงในตารางที่ 2

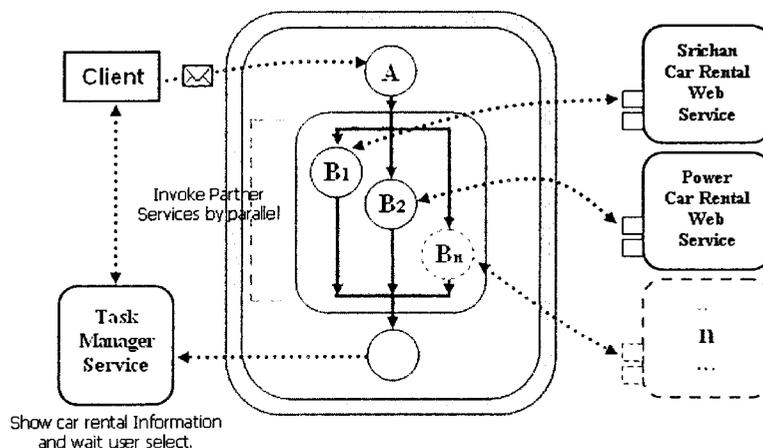
ตารางที่ 2 Input / Output of Hotel Broker Web Services

Input	Output
1. Customer Name	1. Customer Name
2. Destination	2. Destination
	3. <u>Hotel Name</u>
	4. Rating
	5. Class Room
	6. Check-in Date
	7. Check-out Date
	8. No. of Night
	9. No. of Room
	10. <u>Price hotel</u>

หมายเหตุ ค่าที่ขีดเส้นใต้คือค่าที่ถูกส่งไปประมวลผลยังตัวบริการเว็บอื่น

2.2.3 ตัวแทนบริการรถเช่า (Car Rental Broker Web Service)

เป็นการออกแบบกระบวนการทางธุรกิจของตัวแทนบริการจองรถเช่า ซึ่งจะเรียกไปยัง Partner อื่น ๆ ประกอบไปด้วยตัวบริการเว็บของธุรกิจรถเช่า ได้แก่ Srichan Car Rental web service และ Power Car Rental web service และพร้อมสนับสนุนสำหรับการเกิดขึ้นของธุรกิจรถเช่าของบริษัทอื่น ๆ ในอนาคต ดังสถาปัตยกรรมในภาพที่ 20



ภาพที่ 20 แผนภาพการทำงานของตัวแทนการบริการรถเช่าในระดับล่าง : Level 1

การประมวลผลจะเริ่มตั้งแต่รับค่าชื่อลูกค้า (Customer Name) สถานีปลายทาง (Destination) เวลาที่มาถึง (Arriving Time) และชื่อโรงแรม (Hotel Name) จาก Client หรือจากตัวแทนบริการจองตั๋วเครื่องบิน และตัวแทนบริการโรงแรม (Activity A) หลังจากนั้นระบบจะประมวลผลโดยเรียกไปยังตัวบริการเว็บของบริษัท รถเช่าต่าง ๆ (Activity B) และส่งคำรายการรถเช่าที่ได้ไปยังตัวบริหารงานหรือ Task Manager Service เพื่อแสดงรายการให้ลูกค้าได้เลือกผ่านทางเว็บเบราว์เซอร์ โดยค่าผลลัพธ์ที่ได้จากบริการนี้ ดังแสดงในตารางที่ 3

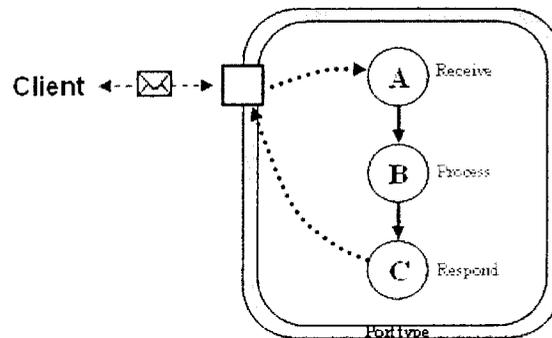
ตารางที่ 3 Input / Output of Car Rental Web Services

Input	Output
1. Customer Name	1. Customer Name
2. Destination	2. Destination
3. Arriving Time	3. Arriving Time
4. Hotel Name	4. Hotel Name
	5. Company
	6. Car Type
	7. Start Date
	8. End Date
	9. No. of Day
	10. No. of Car
	11. Price car rental

หมายเหตุ ค่าที่ขีดเส้นใต้คือค่าที่ถูกส่งไปประมวลผลยังตัวบริการเว็บอื่น

2.2.4 Corporate Expense Service

เป็นบริการสำหรับคำนวณและสรุปค่าใช้จ่ายต่างๆ ที่เกิดขึ้นในระบบธุรกิจการท่องเที่ยวแบบครบวงจรทั้งหมด โดยรับค่าของค่าใช้จ่ายในบริการต่างๆ ที่ลูกค้าได้ใช้บริการ ทำการประมวลผลและส่งกลับผลลัพธ์ ดังกระบวนการทางธุรกิจภาพที่ 21



ภาพที่ 21 แผนภาพการทำงานของตัวบริการจัดการค่าใช้จ่าย : Level 1

การประมวลโดยจะรับค่า ค่าโดยสารเครื่องบิน (Air Fares) ค่าห้องพักโรงแรม (Price Hotel) และค่าเช่ารถ (Price Car Rental) จาก Client หรือจากตัวบริการเว็บต่าง ๆ (Activity A) หลังจากนั้นระบบจะประมวลผล (Activity B) และส่งกลับผลลัพธ์ (Activity C) โดยค่าผลลัพธ์ที่ได้จากบริการนี้แสดงในตารางที่ 4

ตารางที่ 4 Input / Output of Corporate Expense Services

Input	Output
1. Air Fares	1. Air Fares
2. Price hotel	2. Price hotel
3. Price car rental	3. Price car rental
	4. Total

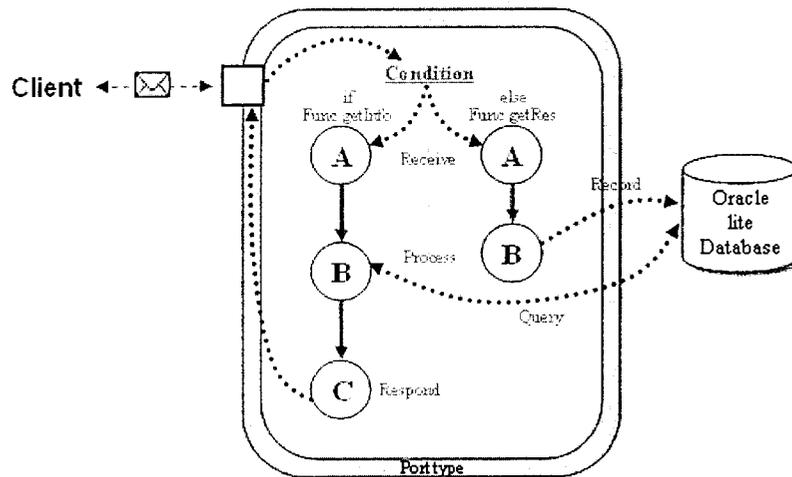
2.3 การออกแบบกระบวนการทางธุรกิจของระบบในระดับ Level 2

ออกแบบกระบวนการทางธุรกิจ ใน Level 2 เป็นการออกแบบการให้บริการของตัวบริการเว็บของผู้ให้บริการในกลุ่มต่าง ๆ เช่น ตัวบริการเว็บในกลุ่มที่ให้บริการจองตั๋วเครื่องบิน ได้แก่ ThaiAirway Web Service และ AirAsia Web Service ตัวบริการเว็บในกลุ่มที่ให้บริการจองโรงแรม ได้แก่ Sofitel Hotel web service และ Kosa Hotel web service และตัวบริการเว็บในกลุ่มที่ให้บริการจองรถเช่า ได้แก่ Srichan Car Rental web service และ Power Car Rental web service

การออกแบบกระบวนการทางธุรกิจในระดับ Level 2 ตัวบริการเว็บในแต่ละกลุ่มมีการออกแบบที่คล้ายกัน จึงขออธิบายกลุ่มบริการละ 1 ตัวบริการเว็บ ดังนี้

2.3.1 ThaiAirway Web Service

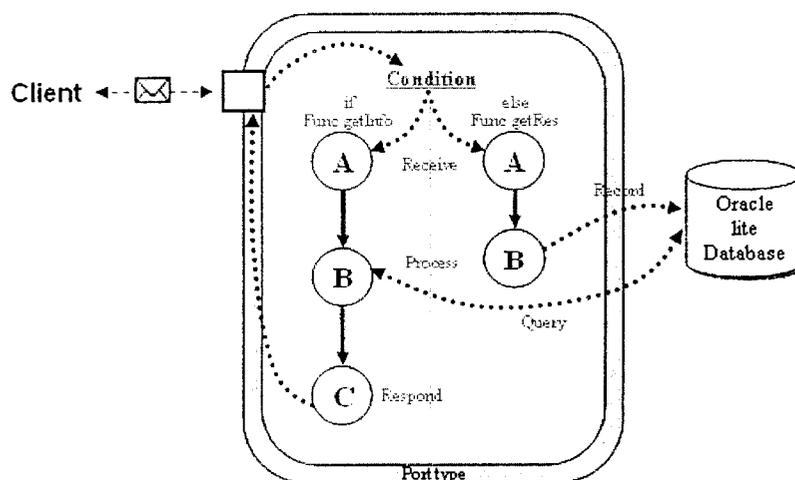
เป็นการออกแบบสถาปัตยกรรมของตัวบริการเว็บที่ให้บริการในการจองตั๋วเครื่องบิน ซึ่งตัวบริการเว็บนี้ทำงานใน 2 ฟังก์ชัน คือ getInformation สำหรับแสดงรายการของเที่ยวบิน โดยจะรับค่าต้นทาง (Source) และปลายทาง (Destination) มาประมวลผลและส่งกลับรายละเอียดเกี่ยวกับรายการเที่ยวบิน และ getReservation สำหรับรับรายการจอง โดยรับค่าชื่อลูกค้า (Customer Name) ต้นทาง (Source) ปลายทาง (Destination) หมายเลขเที่ยวบิน (Flight Number) วันที่เดินทาง (Departure Date) ค่าโดยสาร (Air Fares) และรายละเอียดเกี่ยวกับเที่ยวบินที่ลูกค้าได้ทำการเลือก โดยใช้ภาษา BPEL ในการพัฒนาและใช้ Oracle lite Database เป็นฐานข้อมูล (รายละเอียดการจำลองตัวบริการเว็บนี้ อยู่ในภาคผนวก ข) ดังสถาปัตยกรรมในภาพที่



ภาพที่ 22 แผนภาพการทำงานของ ThaiAirway Web Service

2.3.2 Sofitel Hotel Web Service

เป็นการออกแบบสถาปัตยกรรมของตัวบริการเว็บที่ให้บริการในการจองโรงแรมและห้องพัก ซึ่งตัวบริการเว็บนี้ทำงานใน 2 ฟังก์ชัน คือ getInformation สำหรับแสดงรายการของโรงแรมและประเภทห้องพัก โดยจะรับค่าและปลายทาง (Destination) มาประมวลผลและส่งกลับรายละเอียดเกี่ยวกับรายการโรงแรมและประเภทห้องพัก และ getReservation สำหรับรับรายการจอง โดยรับค่าชื่อลูกค้า ประเภทห้องพัก จำนวนห้องพัก จำนวนวันที่เข้าพัก และราคาห้องพัก โดยใช้ภาษา BPEL ในการพัฒนาและใช้ Oracle lite Database เป็นฐานข้อมูล (รายละเอียดการจำลองตัวบริการเว็บนี้ อยู่ในภาคผนวก ข) ดังสถาปัตยกรรมในภาพที่ 23

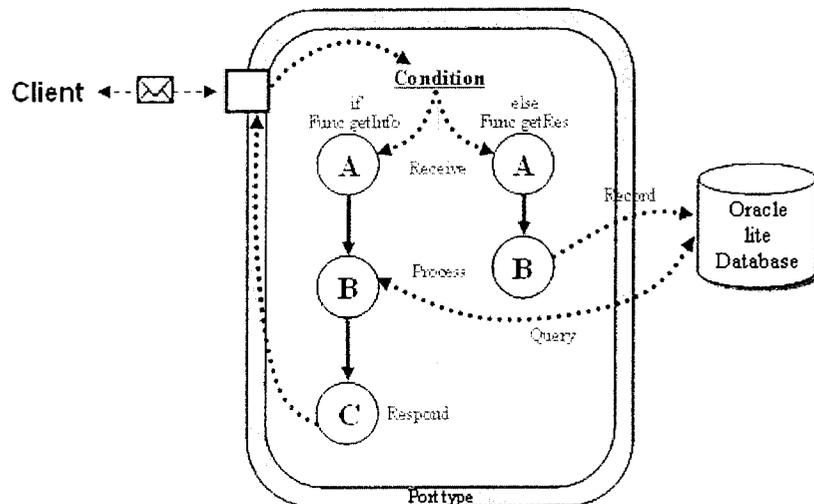


ภาพที่ 23 แผนภาพการทำงานของ Sofitel Hotel Web Service

2.3.3 Khon Kaen Car Rental web service

เป็นการออกแบบสถาปัตยกรรมของตัวบริการเว็บที่ให้บริการในการจองรถเช่า ซึ่งตัวบริการเว็บนี้ทำงานใน 2 ฟังก์ชัน คือ getInformation สำหรับแสดงรายการรถเช่า โดยจะรับค่าและปลายทางมา

ประมวลผลและส่งกลับรายละเอียดเกี่ยวกับรายการรถเช่าและประเภทของรถเช่า และ getReservation สำหรับรับรายการจอง โดยรับค่าชื่อลูกค้า ประเภทรถเช่า จำนวนวันที่เช่ารถ และค่าเช่ารถ โดยใช้ภาษา BPEL ในการพัฒนาและใช้ Oracle lite Database เป็นฐานข้อมูล ดังสถาปัตยกรรมในภาพที่ 24



ภาพที่ 24 แผนภาพการทำงานของ Khon Kaen Car Rental web service

3. จำลองตัวบริการเว็บที่เกี่ยวข้องกับธุรกิจการท่องเที่ยว

ในการสร้างองค์ประกอบตัวบริการเว็บ (Web Service Composition) ในงานวิจัยครั้งนี้ได้ทำการจำลองตัวบริการเว็บขึ้นมาเพื่อประกอบการทำกระบวนการทางธุรกิจ มีขั้นตอนดังนี้

3.1 จำลองตัวบริการเว็บของผู้ให้บริการ (Web Service Providers) ใน Level 2

จำลองตัวบริการเว็บตามสถาปัตยกรรมที่ได้ทำการออกแบบไว้ ประกอบด้วยผู้ให้บริการในกลุ่มต่างๆ ได้แก่ กลุ่มที่ให้บริการจองตั๋วเครื่องบิน เช่น ThaiAirway web service และ AirAsia web service กลุ่มที่ให้บริการจองโรงแรมที่พัก ได้แก่ Sofitel hotel web service และ Kosa hotel web service กลุ่มที่ให้บริการจองรถเช่า ได้แก่ Srichan car rental web service และ Power car rental web service

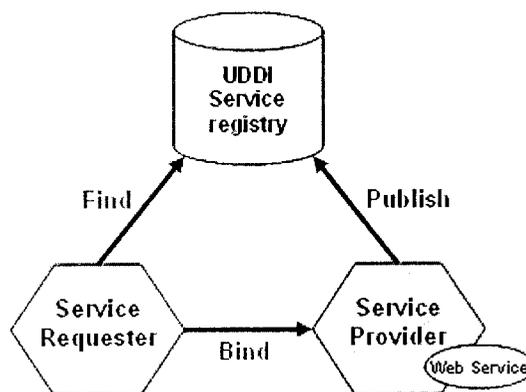
บริการของแต่ละ Web Service Providers ประกอบด้วย 2 Operations ได้แก่ getInformation สำหรับให้ข้อมูลเกี่ยวกับบริการของตนเอง เช่น ThaiAirways web service เมื่อร้องขอบริการไปจะได้ข้อมูลเกี่ยวกับรายการเที่ยวบินกลับมา และ getReservation สำหรับการจองรายการที่ได้เลือกไป สำหรับตัวบริการเว็บที่ได้จำลองขึ้นมา มีดังนี้

- (1) จำลอง ThaiAirway web service
- (2) จำลอง AirAsia web service
- (3) จำลอง Sofitel hotel web service
- (4) จำลอง Kosa hotel web service
- (5) จำลอง Power car rental web service
- (6) จำลอง Srichan car rental web service

(ตัวอย่าง Source Code อยู่ในภาคผนวก ข)

3.2 ลงทะเบียนบริการบน UDDI

เมื่อสร้าง Web Service Providers แต่ละบริการเสร็จเรียบร้อยแล้ว ทำการลงทะเบียนบริการของตนไว้ใน UDDI Service registry เพื่อให้ Service Requester ค้นหาบริการที่ต้องการเรียกใช้ ในงานวิจัยนี้ในเบื้องต้นได้ทำการประกาศการให้บริการของตัวบริการเว็บไว้น UDDI ของ localhost ซึ่งตัวโปรแกรม Oracle Jdeveloper ซึ่งเป็นโปรแกรมที่ใช้ในการสร้างกระบวนการทางธุรกิจจะทำการค้นหาบริการนั้นแล้วนำมาประกอบการสร้างกระบวนการทางธุรกิจ



ภาพที่ 25 สถาปัตยกรรมของเว็บเซอร์วิส

จากภาพที่ 25 สามารถอธิบายได้ดังนี้

Requestor เป็นใครก็ตามที่ต้องการเรียกใช้บริการจาก Provider ซึ่งสามารถค้นหาบริการที่ต้องการได้จาก UDDI registry หรือ Service Registry หรือติดต่อจาก Provider โดยตรง

Registry ทำหน้าที่เป็นตัวกลางให้ Provider มาลงทะเบียนไว้ โดยใช้ WSDL ไฟล์ บอกรายละเอียดของบริษัทและบริการที่มีให้ ซึ่งอาจจะใช้หรือไม่ใช้ก็ได้

Provider เป็นผู้ให้บริการ มีหน้าที่ในการเปิดบริการเพื่อรองรับการขอใช้บริการจาก Requestor ที่เรียกเข้ามาขอใช้

4. สร้างแบบจำลององค์ประกอบตัวบริการเว็บตามมาตรฐานของ BPEL4WS

สร้างแบบจำลององค์ประกอบตัวบริการเว็บของธุรกิจการท่องเที่ยวแบบครบวงจรโดยใช้ภาษา BPEL พัฒนาขึ้นจากโปรแกรม Oracle Jdeveloper โดยจะ Composite ใน 2 ระดับ คือ สถาปัตยกรรมของ Level 0 จะ Composite ตัว Web Service Brokers (ดังภาพที่ 15) และ สถาปัตยกรรมของ Level 1 จะ Composite ตัว Web Service Providers (ดังภาพที่ 16 ภาพที่ 17 และภาพที่ 18)

4.1 ค้นหา WSDL ของบริการที่ต้องการเรียกใช้งานจาก UDDI

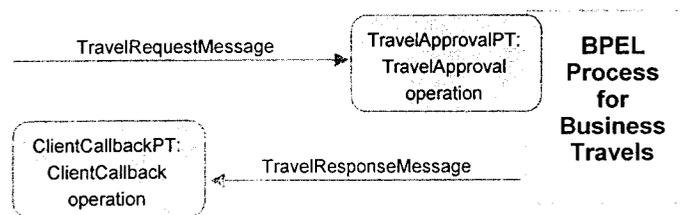
WSDL (Web Service Description Language) เป็นภาษาที่ใช้อธิบายคุณลักษณะการให้บริการของ Web Services และวิธีการติดต่อกับ Web Services ฉะนั้นก่อนที่จะทำการสร้าง Web Service Composition จะต้อง

ทราบที่อยู่ของ WSDL ของบริการที่ต้องการจะเรียกใช้บริการ โดยสามารถค้นหาได้จาก UDDI หรือในกรณีที่ทราบที่อยู่ของ WSDL อยู่แล้วก็อาจจะไม่ต้องค้นหาจาก UDDI ก็ได้

4.2 สร้างกระบวนการทางธุรกิจตามสถาปัตยกรรมในระดับ Level 0

4.2.1 Define WSDL for the BPEL Process

กำหนดค่าของ WSDL สำหรับเตรียมรับ message จาก Clients และตอบกลับค่าผลลัพธ์ โดยได้กำหนดไว้ในรูปแบบของ XML Schema แล้วจึงทำการ import ไฟล์ xsd นั้นเข้ามาใน WSDL โดยติดต่อผ่านทาง Port Type กำหนด Partner Link Types สำหรับติดต่อกับ Web Services อื่นๆ และเป็นลักษณะการประมวลผลแบบไม่ประสานเวลา (Asynchronously) ดังภาพที่ 26 ซึ่งตัวอย่างไฟล์ TravelAgent.wsdl และ TravelAgent.xsd อยู่ในภาคผนวก



ภาพที่ 26 The WSDL for the BPEL process

4.2.2 Create the Business Process

การสร้างกระบวนการทางธุรกิจ (Business Process) ตามมาตรฐานของ BPEL4WS โครงสร้างพื้นฐานของไฟล์ BPEL ประกอบด้วย 3 ส่วนหลัก ได้แก่ <partnerLinks> <variables> และ <sequence> ดังภาพที่ 27

```
<process name="BusinessTravelProcess" ... >
  <partnerLinks>
    <!-- The declaration of partner links -->
  </partnerLinks>
  <variables>
    <!-- The declaration of variables -->
  </variables>
  <sequence>
    <!-- The definition of the BPEL business process main body -->
  </sequence>
</process>
```

ภาพที่ 27 Basic structure of BPEL process definition document

การสร้างกระบวนการทางธุรกิจตามสถาปัตยกรรมในระดับ Level 0 สามารถอธิบายถึงการสร้างกระบวนการทางธุรกิจได้เป็นขั้นตอนดังนี้

- (1) กำหนดชื่อของ Process และ namespaces

```
<process name="TravelAgent"
  targetNamespace="http://xmlns.oracle.com/TravelAgent"
  xmlns="http://schemas.xmlsoap.org/ws/2003/03/business-process/"
  xmlns:ns1="http://xmlns.oracle.com/AirlineBrokerWS"
  xmlns:ns2="http://xmlns.oracle.com/HotelBrokerWS"
```

```

xmlns:ns3="http://xmlns.oracle.com/CarRentalBrokerWS"
xmlns:ns4="http://xmlns.oracle.com/CESservice" >

```

...

(2) กำหนด Partner Links

```

<partnerLinks>
  <partnerLink name="client"
    partnerLinkType="client:TravelAgent"
    myRole="TravelAgentProvider"
    partnerRole="TravelAgentRequester"/>

  <partnerLink name="AirlineBrokerWS"
    partnerLinkType="ns1:AirlineBrokerWS"
    myRole="AirlineBrokerWSRequester"
    partnerRole="AirlineBrokerWSProvider"/>

  <partnerLink name="HotelBrokerWS"
    partnerLinkType="ns2:HotelBrokerWS"
    myRole="HotelBrokerWSRequester"
    partnerRole="HotelBrokerWSProvider"/>

  <partnerLink name="CarRentalBrokerWS"
    partnerLinkType="ns3:CarRentalBrokerWS"
    myRole="CarRentalBrokerWSRequester"
    partnerRole="CarRentalBrokerWSProvider"/>

  <partnerLink name="CESservice"
    partnerLinkType="ns4:CESservice"
    partnerRole="CESserviceProvider"/>
</partnerLinks>

```

(3) กำหนด Variable

```

<variable name="inputVariable" messageType="client:TravelAgentRequestMessage"/>
<variable name="outputVariable"
  messageType="client:TravelAgentResponseMessage"/>
<variable name="invoke_AB"
  messageType="ns1:AirlineBrokerWSRequestMessage"/>
<variable name="receive_AB"
  messageType="ns1:AirlineBrokerWSResponseMessage"/>
<variable name="invoke_HB" messageType="ns2:HotelBrokerWSRequestMessage"/>
<variable name="receive_HB"
  messageType="ns2:HotelBrokerWSResponseMessage"/>
<variable name="invoke_CB"
  messageType="ns3:CarRentalBrokerWSRequestMessage"/>

```

```

<variable name="receive_CB"
  messageType="ns3:CarRentalBrokerWSResponseMessage"/>
<variable name="invoke_CES" messageType="ns4:CESserviceRequestMessage"/>
<variable name="receive_CES" messageType="ns4:CESserviceResponseMessage"/>
</variables>

```

(4) กำหนด Flow ของระบบ

```

<sequence name="main">
  <receive name="receiveInput" partnerLink="client" portType="client:TravelAgent"
    operation="initiate" variable="inputVariable" createInstance="yes"/>
  <scope name="getAirline">
    <switch name="Switch_1">
      <case
        condition="bpws:getVariableData('inputVariable','payload','/ns9:TravelAgent/ns9:
          Services/ns9:airline') = 'yes'">
        <sequence name="Sequence_1">
          <assign name="Assign_AB1">
            <copy>
              <from variable="inputVariable" part="payload"
                query="/ns9:TravelAgent/ns9:AirlineService/ns9:source"/>
              <to variable="invoke_AB" part="payload"
                query="/ns1:AirlineBrokerWSProcessRequest/ns1:source"/>
            </copy>
            <copy>
              <from variable="inputVariable" part="payload"
                query="/ns9:TravelAgent/ns9:AirlineService/ns9:destination"/>
              <to variable="invoke_AB" part="payload"
                query="/ns1:AirlineBrokerWSProcessRequest/ns1:destination"/>
            </copy>
          </assign>
          <invoke name="Invoke_AB" partnerLink="AirlineBrokerWS"
            portType="ns1:AirlineBrokerWS" operation="initiate"
            inputVariable="invoke_AB"/>
          <receive name="Receive_AB" createInstance="no"
            partnerLink="AirlineBrokerWS" portType="ns1:AirlineBrokerWSCallback"
            operation="onResult" variable="receive_AB"/>
          <assign name="Assign_AB2">
            <copy>
              <from variable="receive_AB" part="payload"
                query="/ns1:AirlineBrokerWSProcessResponse/ns1:airlineName"/>

```

```

        <to variable="outputVariable" part="payload"
        query="/ns9:TravelAgent/ns9:AirlineService/ns9:airlineName"/>
        </copy>
    </assign>
</sequence>
</case>
<otherwise>
    <assign name="Assign_AB3">
        <copy>
            <from expression="0"/>
            <to variable="invoke_CES" part="payload"
            query="/ns4:CESServiceProcessRequest/ns4:priceAirline"/>
        </copy>
    </assign>
</otherwise>
</switch>
</scope>
...
</sequence>

```

4.2.3 Deploy to Web Server

หลังจากการสร้างกระบวนการทางธุรกิจแล้ว จะต้องทำการ Deploy ไปยัง Web Server เพื่อเป็นการนำเอาแบบจำลององค์ประกอบตัวบริการเว็บที่สร้าง ขึ้นสู่สถานะบน Server จริงพร้อมทำงาน โดยมี User Interface เป็นตัวกลางในการประสานงานรับส่งค่าระหว่างผู้ใช้และแบบจำลององค์ประกอบตัวบริการเว็บ และก่อนจะทำการ Deploy ต้องกำหนดตัว Descriptor โดยมีการระบุถึงที่อยู่ของ WSDL ของแต่ละตัวบริการเว็บดังนี้

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<BPELSuitcase>
    <BPELProcess id="TravelAgent" src="TravelAgent.bpel">
        <partnerLinkBindings>
            <partnerLinkBinding name="client">
                <property name="wsdlLocation">TravelAgent.wsdl</property>
            </partnerLinkBinding>
            <partnerLinkBinding name="AirlineBrokerWS">
                <property
name="wsdlLocation">http://ekkachai:9700/orabpel/AirlineBrokerWS/AirlineBrokerWS?wsdl</property>
            </partnerLinkBinding>
            <partnerLinkBinding name="HotelBrokerWS">
                <property
name="wsdlLocation">http://ekkachai:9700/orabpel/HotelBrokerWS/HotelBrokerWS?wsdl</property>
            </partnerLinkBinding>
        </partnerLinkBindings>
    </BPELProcess>
</BPELSuitcase>

```

```

<partnerLinkBinding name="CarRentalBrokerWS">
  <property
name="wsdlLocation">http://ekkachai:9700/orabpel/CarRentalBrokerWS/CarRentalBrokerWS?wsdl</property>
</partnerLinkBinding>
<partnerLinkBinding name="CESservice">
  <property
name="wsdlLocation">http://ekkachai:9700/orabpel/default/CESservice/CESservice?wsdl</property>
</partnerLinkBinding>
</partnerLinkBindings>
</BPELProcess>
</BPELSuitcase>

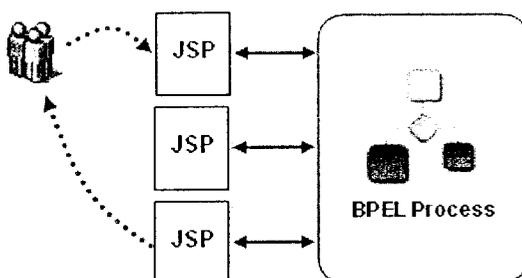
```

5. พัฒนาโปรแกรมประยุกต์เพื่อทดสอบแบบจำลององค์ประกอบตัวบริการเว็บ

หลังจากทำการพัฒนาแบบจำลององค์ประกอบตัวบริการเว็บเสร็จเรียบร้อยแล้ว เพื่อเป็นการทดสอบแบบจำลองดังกล่าว จึงได้ทำการพัฒนาโปรแกรมประยุกต์เพื่อใช้เป็น User Interface ใช้ภาษา JSP ในการพัฒนา โดยมีขั้นตอนการพัฒนาดังนี้

5.1 สร้าง User Interface

สร้าง User Interface เพื่อเป็นตัวกลางในการประสานงานรับค่าและแสดงรายการข้อมูลระหว่างผู้ใช้และ BPEL Process ดังแสดงในภาพที่ 28



ภาพที่ 28 แผนภาพการทำงานของ User Interface

ในการสร้าง User Interface ประกอบไปด้วยไฟล์ต่าง ๆ ดังนี้

(1) TravelAgentUI\index.jsp เป็นหน้าเริ่มต้นสำหรับแสดงรายการบริการต่าง ๆ ลูกค้าสามารถเลือกได้ว่าจะใช้บริการใดบ้าง

(2) AirlineBrokerWSUI\bookFlight.jsp สำหรับให้ลูกค้าเลือกต้นทาง ปลายทาง วันที่เดินทาง และจำนวนผู้โดยสาร สำหรับการจองเที่ยวบิน

(3) AirlineBrokerWSUI\listFlight.jsp สำหรับแสดงรายละเอียดเกี่ยวกับรายการเที่ยวบิน และชั้นโดยสารให้ลูกค้าเลือก

(4) AirlineBrokerWSUI\completeFlight.jsp สำหรับดำเนินการจองตั๋วเครื่องบิน

(5) HotelBrokerWSUI\listHotel.jsp สำหรับแสดงรายละเอียดรายชื่อโรงแรม ประเภทห้องพัก และวันที่ Check-in/out ให้ลูกค้าทำการเลือก

(6) HotelBrokerWSUI\completeHotel.jsp สำหรับดำเนินการจองโรงแรมและห้องพัก

(7) CarRentalWSUI\listCarRental.jsp สำหรับแสดงรายชื่อบริษัทที่ให้บริการรถเช่า และประเภทรถเช่า ในจังหวัดปลายทางที่ลูกค้าเดินทางไป

(8) CarRentalWSUI\completeCarRental.jsp สำหรับดำเนินการจองรถเช่า

(9) CESserviceUI\index.jsp สำหรับแสดงรายละเอียดของค่าใช้จ่ายทั้งหมด เพื่อให้ลูกค้ายืนยันการใช้บริการ

หมายเหตุ : Source Code แสดงอยู่ในภาคผนวก

5.2 Compile และ Deploy ไปยัง Web Server

หลังจากสร้าง User Interface เสร็จต้อง Compile และ Deploy ไปยัง Web Server มีขั้นตอนดังนี้

(1) เปิดโปรแกรม Command Prompt

(2) เข้าไปยังไดเรกทอรีที่เก็บไฟล์ ใช้คำสั่ง cd D:\OraBPELPM_1\integration\TravelAgent

(3) ใช้คำสั่ง obant

6. เครื่องมือที่ใช้ในการพัฒนาระบบ

6.1 ฮาร์ดแวร์

6.1.1 เครื่องคอมพิวเตอร์เซิร์ฟเวอร์ CPU Xeon 3.0 GHz 1 เครื่อง

6.1.2 เครื่องคอมพิวเตอร์ CPU P4^{HT} 2.8 GHz / RAM 512 MB 1 เครื่อง

6.2 ซอฟต์แวร์

6.2.1 Oracle BPEL Process Manager 10.1.2

6.2.2 Oracle Application Server Containers for J2EE 10g

6.2.3 Oracle Jdeveloper BPEL Designer 10g

6.2.4 Oracle Lite Database

6.2.5 nusoap V0.6.7

6.2.6 Macromedia DreamWeaver MX 2004