

การสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-พีเพิลจากกราฟการเรียกเชิงสถิติ



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2563

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

TEST CASE GENERATION FOR WS-BPEL FROM A STATIC CALL GRAPH



A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Software Engineering

Department of Computer Engineering

FACULTY OF ENGINEERING

Chulalongkorn University

Academic Year 2020

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส- บีเพลจากกราฟการเรียกเชิงสถิติ
โดย	น.ส.วาริรัตน์ บัวเสนาะ
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.ธราทิพย์ สุวรรณศาสตร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้หัวข้อวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่ง
ของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

.....	คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สุพจน์ เตชวรสินสกุล)	
คณะกรรมการสอบวิทยานิพนธ์	
.....	ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์)	
.....	อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.ธราทิพย์ สุวรรณศาสตร์)	
.....	กรรมการ
(รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)	
.....	กรรมการ
(รองศาสตราจารย์ ดร.พรศิริ หมั่นไชยศรี)	
.....	กรรมการภายนอกมหาวิทยาลัย
(ผู้ช่วยศาสตราจารย์ ดร.ภัทรชัย ลลิตโรจน์วงศ์)	

วาริรต์น์ บัวเสนาะ : การสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพล
จากกราฟการเรียกเชิงสถิติ. (TEST CASE GENERATION FOR WS-BPEL FROM A
STATIC CALL GRAPH) อ.ที่ปรึกษาหลัก : รศ. ดร.ธราทิพย์ สุวรรณศาสตร์

เว็บเซอร์วิสถูกใช้งานกันอย่างแพร่หลายและได้รับความนิยมเป็นอย่างมาก การพัฒนาเว็บเซอร์วิสนั้นไม่ยึดติดกับภาษาที่ใช้ในการพัฒนาจึงทำให้เว็บเซอร์วิสมีรูปแบบที่หลากหลายและแตกต่างกันออกไป จึงต้องมีมาตรฐานในการควบคุมและประสานกระบวนการธุรกิจผ่านทางเว็บเซอร์วิส มาตรฐานนี้ถูกเรียกว่าดับเบิลยูเอส-บีเพล การทดสอบเว็บเซอร์วิสนั้นก็เป็นการทดสอบที่สำคัญในการพัฒนาซอฟต์แวร์เพื่อให้ตรงตามความต้องการของกระบวนการธุรกิจ งานวิจัยในปัจจุบันมีการนำเสนอแนวคิดและเครื่องมือที่ใช้ในการสร้างกรณีทดสอบสำหรับดับเบิลยูเอส-บีเพล โดยมุ่งเน้นเพียงกระบวนการธุรกิจในไฟล์ดับเบิลยูเอส-บีเพลเพียงหนึ่งไฟล์เท่านั้น

วิทยานิพนธ์ฉบับนี้จัดทำขึ้นเพื่อนำเสนอแนวทางและเครื่องมือการสร้างกรณีทดสอบสำหรับดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ เพื่อรองรับกรณีที่มีการเรียกใช้กระบวนการย่อยระหว่างไฟล์ดับเบิลยูเอส-บีเพล โดยเครื่องมือจะวิเคราะห์หาความสัมพันธ์ของแต่ละไฟล์แทรกชุดคำสั่งในรหัสต้นทาง วิเคราะห์ความสัมพันธ์การเรียกใช้งานระหว่างไฟล์ดับเบิลยูเอส-บีเพลเพื่อสร้างกราฟการเรียกเชิงสถิติ จากนั้นสร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติเพื่อให้ได้ความครอบคลุมในระดับกึ่ง

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สาขาวิชา วิศวกรรมซอฟต์แวร์
ปีการศึกษา 2563

ลายมือชื่อนิสิต
ลายมือชื่อ อ.ที่ปรึกษาหลัก

5970958021 : MAJOR SOFTWARE ENGINEERING

KEYWORD: WS-BPEL, Static Call Graph, Test Case Generation

Wareerat Bousanoh : TEST CASE GENERATION FOR WS-BPEL FROM A STATIC CALL GRAPH. Advisor: Assoc. Prof. Taratip Suwannasart, Ph.D.

Web service is widely used and very popular. However, web service development has not adhered to any development languages in variety of platforms or frameworks. There must be a standard in controlling and coordinating a business process through a web service. This standard is called WS-BPEL. Web service testing is an important process in software development to meet the needs of the business process. Recently some researchers proposed concepts and tools used to generate test cases for WS-BPEL, while they focused only a business processes within a single WS-BPEL file.

This thesis proposes an approach and a test case generation tool for WS-BPEL from a static call graph in case that there are calling subprocesses between WS-BPEL files. The tool can analyze relationships among those WS-BPEL files, instrument code into the original WS-BPEL files, analyze calling relationships among WS-BPEL files in order to generate a static call graph. Then test cases that are satisfied branch coverage, are created from the graph.

CHULALONGKORN UNIVERSITY

Field of Study: Software Engineering

Student's Signature

Academic Year: 2020

Advisor's Signature

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ด้วยความอนุเคราะห์และความช่วยเหลืออย่างสูงจาก รองศาสตราจารย์ ดร.ธราทิพย์ สุวรรณศาสตร์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่สละเวลาชี้แนะและแนะนำแนวทางในการทำวิทยานิพนธ์และผลงานทางวิชาการ แม้ว่าผู้วิจัยจะติดภารกิจจากทางบ้านและงานประจำ ท่านก็ยังคงมีความกรุณาและสนับสนุนผู้วิจัยด้วยความเอาใจใส่ยิ่ง

ขอกราบขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์ ประธานกรรมการสอบวิทยานิพนธ์ รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ รองศาสตราจารย์ ดร.พรศิริ หมั่นไชยศรี และผู้ช่วยศาสตราจารย์ ดร.ภัทรชัย ลลิตโรจน์วงศ์ กรรมการสอบวิทยานิพนธ์ ที่ได้สละเวลาและให้คำแนะนำในการแก้ไขวิทยานิพนธ์ฉบับนี้ให้มีความสมบูรณ์มากยิ่งขึ้น

ขอกราบขอบพระคุณอาจารย์ทุกท่านในภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ที่ได้ให้ความรู้ทางวิชาการที่มีประโยชน์ต่อการจัดทำวิทยานิพนธ์ฉบับนี้ รวมถึงบุคลากรทุกท่านในภาควิชาฯ ที่ช่วยประสานงาน ให้คำแนะนำ และช่วยเหลือผู้วิจัยเรื่อยมา ตั้งแต่เริ่มเข้ามาศึกษาเล่าเรียนตลอดจนสอบวิทยานิพนธ์สำเร็จลุล่วง

ขอขอบคุณสมาชิกในครอบครัว เพื่อน ๆ พี่ ๆ น้อง ๆ และหัวหน้างาน ที่เข้าใจ เป็นกำลังใจ และให้การสนับสนุนแก่ผู้วิจัยเสมอมา

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

วาริรัตน์ บัวเสนาะ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ค
บทคัดย่อภาษาอังกฤษ.....	ง
กิตติกรรมประกาศ.....	จ
สารบัญ.....	ฉ
สารบัญตาราง.....	ฅ
สารบัญรูป.....	ฎ
บทที่ 1 บทนำ	1
1.1. ความเป็นมาและความสำคัญของงานวิจัย.....	1
1.2. วัตถุประสงค์ของงานวิจัย.....	2
1.3. ขอบเขตของงานวิจัย	2
1.4. ขั้นตอนและวิธีการดำเนินงานวิจัย	3
1.5. ประโยชน์ที่คาดว่าจะได้รับ.....	4
1.6. บทความวิชาการที่ได้รับการตีพิมพ์.....	4
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	5
2.1 ทฤษฎีที่เกี่ยวข้อง.....	5
2.1.1 ดัชนีเบิลยูเอส-ปีเพิล	5
2.1.2 ดัชนีเบิลยูเอสดีแอล.....	7
2.1.3 เอกซ์เอสดี.....	8
2.1.4 กราฟการเรียกเชิงสถิติ	8
2.2 งานวิจัยที่เกี่ยวข้อง	9

2.2.1 A Design of WS-BPEL Test Case Generation Tool Based on Path Conditions	9
2.2.2 Automatic Test Case Generation for Orchestration Languages at Service Oriented Architecture	10
2.2.3 An Approach for Test Case Generation from a Static Call Graph for Object-Oriented Programming	10
บทที่ 3 วิธีการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิง สถิติ.....	12
3.1 นำเข้าชุดไฟล์ข้อมูล	13
3.2 แทรกชุดคำสั่งในรหัสต้นทางดับเบิลยูเอส-บีเพล	20
3.3 สร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติ	23
3.4 วิเคราะห์ตัวแปรและข้อจำกัดของตัวแปร	25
3.4.1 วิเคราะห์ตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล.....	25
3.4.2 วิเคราะห์ตัวแปรในแต่ละโหนดของทางเดินทดสอบ.....	27
3.5 สร้างกรณีทดสอบ	28
3.5.1 สุ่มค่าของตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล.....	28
3.5.2 สร้างกรณีทดสอบ.....	29
3.6 สร้างรายงานการทดสอบดับเบิลยูเอส-บีเพล.....	29
บทที่ 4 การออกแบบและพัฒนาเครื่องมือ.....	31
4.1 การออกแบบเครื่องมือ.....	31
4.1.1 แผนภาพยูสเคส.....	31
4.1.2 แผนภาพกิจกรรม.....	33
4.1.3 แผนภาพคลาส	36
4.1.4 แผนภาพลำดับ.....	43
4.1.5 โครงสร้างฐานข้อมูล.....	51

4.2 การพัฒนาเครื่องมือ.....	52
4.2.1 สภาพแวดล้อมที่ใช้ในการพัฒนา.....	52
4.2.2 ส่วนต่อประสานผู้ใช้.....	53
4.2.3 แผนภาพการติดตั้ง.....	63
บทที่ 5 การทดสอบเครื่องมือ.....	66
5.1 สภาพแวดล้อมที่ใช้ในการทดสอบ.....	66
5.2 ขั้นตอนในการทดสอบเครื่องมือ.....	66
5.3 การทดสอบเครื่องมือ.....	66
5.1.1 กรณีศึกษาที่ 1 ระบบโปรโมชันรับเครดิตเงินคืนของแผนกเครื่องใช้ไฟฟ้า.....	67
5.1.2 กรณีศึกษาที่ 2 ระบบร้านขนส่งพัสดุ.....	76
5.1.3 กรณีศึกษาที่ 3 ระบบติดตามการทำงานตู้คืออส.....	86
5.4 ผลการทดสอบเครื่องมือ.....	94
5.5 สรุปผลการทดสอบเครื่องมือ.....	99
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ.....	101
6.1 สรุปผลการวิจัย.....	101
6.2 ข้อจำกัดงานวิจัย.....	101
6.3 แนวทางการพัฒนางานวิจัย.....	102
บรรณานุกรม.....	103
ภาคผนวก.....	105
ภาคผนวก ก รายละเอียดยูนิตของเครื่องมือ.....	106
ภาคผนวก ข รายละเอียดกรณีทดสอบของแต่ละกรณีศึกษา.....	111
ประวัติผู้เขียน.....	121

สารบัญตาราง

หน้า

ตารางที่ 3-1 ชุดตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเฟลและข้อจำกัดของตัวแปร	27
ตารางที่ 3-2 เงื่อนไขของตัวแปรในแต่ละโหนดของทางเดินทดสอบ P1	27
ตารางที่ 3-3 ค่าข้อมูลของตัวแปรนำเข้าบนทางเดินทดสอบ P1	28
ตารางที่ 3-4 กรณีทดสอบของทางเดินทดสอบ P1	29
ตารางที่ ก-1 รายละเอียดของยูสเคสนำเข้าข้อมูล.....	106
ตารางที่ ก-2 รายละเอียดของยูสเคสแทรกชุดคำสั่งในรหัสต้นทาง	107
ตารางที่ ก-3 รายละเอียดของยูสเคสร่างกรณีทดสอบ	107
ตารางที่ ก-4 รายละเอียดของยูสเคสดาวนโหนดรหัสต้นทางที่ถูกแทรกชุดคำสั่ง.....	108
ตารางที่ ก-5 รายละเอียดของยูสเคสดาวนโหนดกรณีทดสอบทั้งหมด.....	109
ตารางที่ ก-6 รายละเอียดของยูสเคสเรียกดูรายละเอียดกรณีทดสอบ	109
ตารางที่ ก-7 รายละเอียดของยูสเคสเรียกดูรายงานการทดสอบ.....	110
ตารางที่ ก-8 รายละเอียดของยูสเคสเรียกดูประวัติการทดสอบดับเบิลยูเอส-บีเฟล.....	110
ตารางที่ ข-1 กรณีทดสอบ TC01 ของกรณีศึกษาที่ 1.....	111
ตารางที่ ข-2 กรณีทดสอบ TC02 ของกรณีศึกษาที่ 1.....	111
ตารางที่ ข-3 กรณีทดสอบ TC03 ของกรณีศึกษาที่ 1.....	112
ตารางที่ ข-4 กรณีทดสอบ TC04 ของกรณีศึกษาที่ 1.....	112
ตารางที่ ข-5 กรณีทดสอบ TC05 ของกรณีศึกษาที่ 1.....	113
ตารางที่ ข-6 กรณีทดสอบ TC06 ของกรณีศึกษาที่ 1.....	113
ตารางที่ ข-7 กรณีทดสอบ TC01 ของกรณีศึกษาที่ 2.....	114
ตารางที่ ข-8 กรณีทดสอบ TC02 ของกรณีศึกษาที่ 2.....	115
ตารางที่ ข-9 กรณีทดสอบ TC03 ของกรณีศึกษาที่ 2.....	115

ตารางที่ ข-10	กรณีทดสอบ TC04 ของกรณีศึกษาที่ 2	116
ตารางที่ ข-11	กรณีทดสอบ TC05 ของกรณีศึกษาที่ 2	117
ตารางที่ ข-12	กรณีทดสอบ TC06 ของกรณีศึกษาที่ 2	117
ตารางที่ ข-13	กรณีทดสอบ TC07 ของกรณีศึกษาที่ 2	118
ตารางที่ ข-14	กรณีทดสอบ TC08 ของกรณีศึกษาที่ 2	118
ตารางที่ ข-15	กรณีทดสอบ TC09 ของกรณีศึกษาที่ 2	119
ตารางที่ ข-16	กรณีทดสอบ TC01 ของกรณีศึกษาที่ 3	119
ตารางที่ ข-17	กรณีทดสอบ TC02 ของกรณีศึกษาที่ 3	120
ตารางที่ ข-18	กรณีทดสอบ TC03 ของกรณีศึกษาที่ 3	120



สารบัญรูป

	หน้า
รูปที่ 2-1 ไฟล์บีเพลที่มีกระบวนการย่อยภายใน	6
รูปที่ 2-2 ไฟล์บีเพลย่อย sbp_VoucherCashback.....	6
รูปที่ 2-3 ไฟล์บีเพลที่มีการเรียกใช้บีเพลย่อย sbp_VoucherCashback.....	7
รูปที่ 2-4 ตัวอย่างไฟล์ดับเบิลยูเอสดีแอล	7
รูปที่ 2-5 ตัวอย่างไฟล์เอกซ์เอสดี	8
รูปที่ 2-6 (ก) การเรียกใช้ฟังก์ชันในภาษาจาวา.....	9
รูปที่ 2-6 (ข) กราฟการเรียกเชิงสถิติ	9
รูปที่ 3-1 ภาพรวมของการดำเนินงานวิจัย.....	12
รูปที่ 3-2 ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล.....	13
รูปที่ 3-3 ไฟล์บีเพลตั้งต้น.....	14
รูปที่ 3-4 ไฟล์ดับเบิลยูเอสดีแอล.....	15
รูปที่ 3-5 ไฟล์เอกซ์เอสดี	16
รูปที่ 3-6 ไฟล์บีเพลย่อย sbp_CheckCCAllow.....	17
รูปที่ 3-7 ไฟล์บีเพลย่อย sbp_VocherCashback.....	18
รูปที่ 3-8 เส้นทางการไหลของบีเพลในไฟล์บีเพลตั้งต้น	19
รูปที่ 3-9 เส้นทางการไหลของบีเพลในไฟล์บีเพลย่อย sbp_CheckCCAllow	19
รูปที่ 3-10 เส้นทางการไหลของบีเพลในไฟล์บีเพลย่อย sbp_VoucherCashback	20
รูปที่ 3-11 (ก) กราฟการไหลของบีเพลในรหัสต้นทางที่ถูกแทรกชุดคำสั่งด้วยโหนด <assign>	21
รูปที่ 3-11 (ข) รหัสต้นทางดับเบิลยูเอส-บีเพลที่ถูกแทรกชุดคำสั่งด้วยโหนด <assign>.....	22
รูปที่ 3-12 ไฟล์บีเพลตั้งต้นที่มีการเรียกใช้บีเพลย่อย sbp_CheckCCAllow.....	23
รูปที่ 3-13 ไฟล์บีเพลตั้งต้นที่มีการเรียกใช้บีเพลย่อย sbp_VoucherCashback.....	24

รูปที่ 3-14 (ก) ความสัมพันธ์ของเส้นทางการไหลระหว่างดับเบิลยูเอส-บีเพล	24
รูปที่ 3-14 (ข) กราฟการเรียกเชิงสถิติในการเรียกใช้งานระหว่างดับเบิลยูเอส-บีเพล	24
รูปที่ 3-15 ชุดตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล	25
รูปที่ 3-16 ชุดตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพลบนไฟล์เอกซ์เอสดี	26
รูปที่ 3-17 ตัวแปรนำเข้าของกรณีทดสอบ TC01 ในรูปแบบเอกซ์เอ็มแอล	29
รูปที่ 3-18 รายงานการทดสอบดับเบิลยูเอส-บีเพล	30
รูปที่ 4-1 แผนภาพยูสเคสของเครื่องมือ	32
รูปที่ 4-2 แผนภาพกิจกรรมการนำเข้าข้อมูล	33
รูปที่ 4-3 แผนภาพกิจกรรมการแทรกชุดคำสั่งในรหัสต้นทาง	34
รูปที่ 4-4 แผนภาพกิจกรรมการทดสอบดับเบิลยูเอส-บีเพล	35
รูปที่ 4-5 แผนภาพแพ็คเกจของเครื่องมือ	36
รูปที่ 4-6 คลาส TestCaseGenerator	37
รูปที่ 4-7 คลาส TestCaseDetail	37
รูปที่ 4-8 คลาส TestReportGenerator	37
รูปที่ 4-9 คลาส ProjectHistory	38
รูปที่ 4-10 คลาส ProjectHistoryDetail	38
รูปที่ 4-11 คลาส ProjectController	38
รูปที่ 4-12 คลาส BpelController	39
รูปที่ 4-13 คลาส VariableController	39
รูปที่ 4-14 คลาส TestPathController	39
รูปที่ 4-15 คลาส TestCaseController	40
รูปที่ 4-16 คลาส TestReportController	40
รูปที่ 4-17 คลาส ServiceController	40
รูปที่ 4-18 คลาส LocalFileController	41

รูปที่ 4-19 คลาส Project	41
รูปที่ 4-20 คลาส Bpel.....	42
รูปที่ 4-21 คลาส Variable	42
รูปที่ 4-22 คลาส TestPath.....	43
รูปที่ 4-23 คลาส TestCase	43
รูปที่ 4-24 แผนภาพลำดับการนำเข้าข้อมูลตั้งต้น	44
รูปที่ 4-25 แผนภาพลำดับการนำเข้าชุดไฟล์บีเพลย่อย	46
รูปที่ 4-26 แผนภาพลำดับการแทรกชุดคำสั่งในรหัสต้นทาง.....	48
รูปที่ 4-27 แผนภาพลำดับการสร้างกรณีทดสอบ	50
รูปที่ 4-28 แผนภาพอีอาร์ของเครื่องมือ	51
รูปที่ 4-29 แผนภาพวินโดว์เนวิเกชันของเครื่องมือ	53
รูปที่ 4-30 หน้าจอ index.html.....	54
รูปที่ 4-31 หน้าจอติดต่อสอบถามข้อมูลเพิ่มเติม	54
รูปที่ 4-32 ขั้นตอนนำเข้าข้อมูลตั้งต้นของหน้าจอ generateTC.html.....	55
รูปที่ 4-33 ขั้นตอนนำเข้าชุดไฟล์บีเพลย่อยของหน้าจอ generateTC.html	56
รูปที่ 4-34 ขั้นตอนนำเข้าบีเพลย่อย sbp_VISAOnTop ที่ถูกเรียกใช้งาน.....	57
รูปที่ 4-35 ขั้นตอนนำเข้าบีเพลย่อย sbp_ChildPromotion และ sbp_Masks ที่ถูกเรียกใช้งาน... 57	
รูปที่ 4-36 ขั้นตอนสร้างกรณีทดสอบของหน้าจอ generateTC.html	58
รูปที่ 4-37 กรณีทดสอบที่ดาวนโหลดจากเครื่องมือ	59
รูปที่ 4-38 กรณีทดสอบในรูปแบบไฟล์เอกซ์เซล	59
รูปที่ 4-39 กรณีทดสอบในรูปแบบไฟล์เอกซ์เอ็มแอล	59
รูปที่ 4-40 หน้าจอ testCaseDtls.html.....	60
รูปที่ 4-41 หน้าจอ generateTR.html.....	61
รูปที่ 4-42 หน้าจอ projectHistory.html.....	62

รูปที่ 4-43 หน้าจอ projectHistoryDtls.html	62
รูปที่ 4-44 แผนภาพการติดตั้งเครื่องมือ	63
รูปที่ 4-45 การติดตั้งรหัสต้นทางที่ถูกแทรกชุดคำสั่งบนปีเพลเอนจิน	64
รูปที่ 4-46 การนำเข้ากรณีทดสอบบนปีเพลเอนจิน	64
รูปที่ 4-47 การทดสอบดับเบิ้ลยูเอส-ปีเพลบนปีเพลเอนจิน	65
รูปที่ 5-1 กระบวนการดับเบิ้ลยูเอส-ปีเพลในไฟล์ปีเพลตั้งต้น CreditCardPromotion.....	67
รูปที่ 5-2 กระบวนการดับเบิ้ลยูเอส-ปีเพลในไฟล์ปีเพลย่อย sbpCheckCreditCard.....	67
รูปที่ 5-3 กระบวนการดับเบิ้ลยูเอส-ปีเพลในไฟล์ปีเพลย่อย sbpVoucherCashback.....	68
รูปที่ 5-4 ตัวแปรนำเข้าและตัวแปรส่งออกกระบวนการดับเบิ้ลยูเอส-ปีเพลของกรณีศึกษาที่ 1	68
รูปที่ 5-5 หน้าจอนำเข้าข้อมูลตั้งต้นของกรณีศึกษาที่ 1	69
รูปที่ 5-6 หน้าจอนำเข้าไฟล์ปีเพลย่อย sbpCheckCreditCard	70
รูปที่ 5-7 หน้าจอนำเข้าไฟล์ปีเพลย่อย sbpVoucherCashback เพิ่มเติม.....	71
รูปที่ 5-8 กราฟการเรียกเชิงสถิติของกรณีศึกษาที่ 1.....	72
รูปที่ 5-9 กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 1	73
รูปที่ 5-10 กรณีทดสอบทั้งหมดของกรณีศึกษาที่ 1 ในรูปแบบเอกซ์เซล	73
รูปที่ 5-11 กรณีทดสอบ TC01 ของกรณีศึกษาที่ 1 ในรูปแบบเอกซ์เอ็มแอล	74
รูปที่ 5-12 การนำเข้ากรณีทดสอบ TC01 ของกรณีศึกษาที่ 1 บนปีเพลเอนจิน.....	74
รูปที่ 5-13 ผลลัพธ์การทดสอบด้วยกรณีทดสอบ TC01 ของกรณีศึกษาที่ 1 บนปีเพลเอนจิน.....	75
รูปที่ 5-14 รายงานการทดสอบดับเบิ้ลยูเอส-ปีเพลของกรณีศึกษาที่ 1	75
รูปที่ 5-15 กระบวนการดับเบิ้ลยูเอส-ปีเพลในไฟล์ปีเพลตั้งต้น TransExpress.....	76
รูปที่ 5-16 กระบวนการดับเบิ้ลยูเอส-ปีเพลในไฟล์ปีเพลย่อย sbpRegister.....	77
รูปที่ 5-17 กระบวนการดับเบิ้ลยูเอส-ปีเพลในไฟล์ปีเพลย่อย sbpCalPkgSize.....	77
รูปที่ 5-18 กระบวนการดับเบิ้ลยูเอส-ปีเพลในไฟล์ปีเพลย่อย sbpCalPrice	78
รูปที่ 5-19 กระบวนการดับเบิ้ลยูเอส-ปีเพลในไฟล์ปีเพลย่อย sbpChkWeigh.....	78

รูปที่ 5-20	ตัวแปรนำเข้าและตัวแปรส่งออกกระบวนการดับเบิลยูเอส-ปีเพลของกรณีศึกษาที่ 2	79
รูปที่ 5-21	หน้าจอนำเข้าข้อมูลตั้งต้นของกรณีศึกษาที่ 2	80
รูปที่ 5-22	หน้าจอนำเข้าไฟล์ปีเพลย่อยของกรณีศึกษาที่ 2	81
รูปที่ 5-23	หน้าจอนำเข้าไฟล์ปีเพลย่อยเพิ่มเติมของกรณีศึกษาที่ 2	81
รูปที่ 5-24	กราฟการเรียกเชิงสถิติของกรณีศึกษาที่ 2	82
รูปที่ 5-25	กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 2	84
รูปที่ 5-26	กรณีทดสอบทั้งหมดของกรณีศึกษาที่ 2 ในรูปแบบเอกซ์เซล	84
รูปที่ 5-27	กรณีทดสอบ TC06 ของกรณีศึกษาที่ 2 ในรูปแบบเอกซ์เอ็มแอล	85
รูปที่ 5-28	การนำเข้ากรณีทดสอบ TC06 ของกรณีศึกษาที่ 2 บนปีเพลเอนจิน	85
รูปที่ 5-29	ผลลัพธ์การทดสอบด้วยกรณีทดสอบ TC06 ของกรณีศึกษาที่ 2 บนปีเพลเอนจิน	85
รูปที่ 5-30	รายงานการทดสอบดับเบิลยูเอส-ปีเพลของกรณีศึกษาที่ 2	86
รูปที่ 5-31	กระบวนการดับเบิลยูเอส-ปีเพลในไฟล์ปีเพลตั้งต้น KioskTracking	87
รูปที่ 5-32	กระบวนการย่อยภายใน Logger ในไฟล์ปีเพลตั้งต้น KioskTracking	87
รูปที่ 5-33	ตัวแปรนำเข้าและตัวแปรส่งออกกระบวนการดับเบิลยูเอส-ปีเพลของกรณีศึกษาที่ 3	88
รูปที่ 5-34	หน้าจอนำเข้าข้อมูลตั้งต้นของกรณีศึกษาที่ 3	89
รูปที่ 5-35	กราฟการเรียกเชิงสถิติของกรณีศึกษาที่ 3	90
รูปที่ 5-36	กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 3	91
รูปที่ 5-37	กรณีทดสอบทั้งหมดของกรณีศึกษาที่ 3 ในรูปแบบเอกซ์เซล	92
รูปที่ 5-38	กรณีทดสอบ TC01 ของกรณีศึกษาที่ 3 ในรูปแบบเอกซ์เอ็มแอล	92
รูปที่ 5-39	การนำเข้ากรณีทดสอบ TC01 ของกรณีศึกษาที่ 3 บนปีเพลเอนจิน	93
รูปที่ 5-40	ผลลัพธ์การทดสอบด้วยกรณีทดสอบ TC01 ของกรณีศึกษาที่ 3 บนปีเพลเอนจิน	93
รูปที่ 5-41	รายงานการทดสอบดับเบิลยูเอส-ปีเพลของกรณีศึกษาที่ 3	94

บทที่ 1

บทนำ

1.1. ความเป็นมาและความสำคัญของงานวิจัย

ปัจจุบันการใช้งานเว็บเซอร์วิส (Web Service) ถูกใช้งานกันอย่างแพร่หลายและได้รับความนิยมเป็นอย่างมาก เนื่องจากเว็บเซอร์วิสสามารถนำกลับมาใช้งานใหม่ได้และง่ายต่อการบำรุงรักษา โดยการทำงานของเว็บเซอร์วิสจะช่วยสนับสนุนการแลกเปลี่ยนข้อมูลและเรียกใช้บริการระหว่างเว็บเซอร์วิสผ่านระบบเครือข่ายโดยใช้ภาษาเอกซ์เอ็มแอล (XML: Extensible Markup Language) ในการติดต่อสื่อสารระหว่างกัน ซึ่งวิธีการพัฒนาเว็บเซอร์วิสเป็นวิธีการหนึ่งในการพัฒนาซอฟต์แวร์ (Software) ภายใต้หลักการของสถาปัตยกรรมเชิงบริการ (SOA: Service Oriented Architecture) [1] ที่มีแนวคิดในการออกแบบระบบซอฟต์แวร์ โดยแบ่งเป็นส่วนประกอบย่อย ทำให้ส่วนประกอบซอฟต์แวร์นั้นเป็นอิสระต่อกัน สามารถเรียกใช้บริการและทำงานร่วมกันได้ แต่เนื่องด้วยการพัฒนาเว็บเซอร์วิสนั้นไม่ยึดติดกับภาษาที่ใช้ในการพัฒนา ทำให้เว็บเซอร์วิสมีรูปแบบที่หลากหลายและแตกต่างกันออกไป จึงต้องมีมาตรฐานในการควบคุมการทำงานและประสานงานกันของกระบวนการทางธุรกิจ (Business Process) ผ่านเว็บเซอร์วิส โดยมาตรฐานที่กล่าวถึง คือดับเบิลยูเอส-บีเพล (WS-BPEL: Web Service Business Process Execution Language) ที่พัฒนาขึ้นโดยองค์กรโอเอซิส (OASIS: Advancing Open Standards for The Information Society) [2]

ดับเบิลยูเอส-บีเพลเป็นภาษาที่ใช้ในอธิบายกระบวนการทางธุรกิจในรูปแบบของเว็บเซอร์วิสที่สามารถเรียกใช้งานเว็บเซอร์วิสทั้งภายในและภายนอกได้ โดยมักจะมีการใช้งานควบคู่ไปกับการใช้ดับเบิลยูเอสดีแอล (WSDL: Web Service Description Language) ซึ่งเป็นไฟล์ที่อธิบายส่วนที่ให้บริการ และลักษณะของข้อมูลนำเข้าและข้อมูลส่งออก โดยสามารถอธิบายได้ทั้งลักษณะของดับเบิลยูเอส-บีเพลเอง และอธิบายถึงเว็บเซอร์วิสอื่นที่เรียกใช้งานได้

เพื่อให้การพัฒนาเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพล สามารถตอบสนองตามความต้องการตามกระบวนการทางธุรกิจขององค์กร การทดสอบเว็บเซอร์วิสจึงเป็นขั้นตอนหนึ่งที่มีความสำคัญในกระบวนการพัฒนาซอฟต์แวร์เพื่อลดข้อผิดพลาดก่อนการส่งมอบและนำไปใช้งานจริง นอกจากนี้ยังช่วยทำให้เกิดความมั่นใจว่าซอฟต์แวร์สามารถทำงานร่วมกันได้อย่างถูกต้อง งานวิจัยในปัจจุบันมีการนำเสนอแนวคิดและเครื่องมือที่ใช้ในการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพล ที่มุ่งเน้นเพียงกระบวนการทางธุรกิจภายในไฟล์ดับเบิลยูเอส-บีเพลเพียงไฟล์เดียวเท่านั้น ซึ่งในกรณีที่

ซอฟต์แวร์มีการเรียกใช้งานกันระหว่างไฟล์ดับเบิลยูเอส-บีเพลจำเป็นต้องมีการทดสอบการเรียกใช้บริการระหว่างกัน โดยการใช้กราฟการเรียกเชิงสถิต (Static Call Graph) เข้ามาช่วยให้ผู้ทดสอบสามารถทำความเข้าใจเกี่ยวกับไฟล์ดับเบิลยูเอส-บีเพลทั้งหมดที่มีการเรียกใช้งานต่อเนื่องกัน เพื่อที่จะสามารถรวบรวมตัวแปรที่เกี่ยวข้องและนำมาจัดทำเป็นกรณีทดสอบได้ ทำให้การสร้างกรณีทดสอบต้องใช้ผู้ที่มีความรู้ความเข้าใจเกี่ยวกับลักษณะของไฟล์ดับเบิลยูเอส-บีเพล เพื่อให้ได้กรณีทดสอบที่มีความครอบคลุมทั้งซอฟต์แวร์

จากปัญหาข้างต้น งานวิจัยนี้จึงจัดทำขึ้นเพื่อศึกษาและออกแบบแนวทางการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลและพัฒนาเครื่องมือสำหรับใช้สร้างกรณีทดสอบจากกราฟการเรียกเชิงสถิตของดับเบิลยูเอส-บีเพล โดยรองรับกรณีที่เว็บเซอร์วิสมีการเรียกใช้งานกันระหว่างไฟล์ดับเบิลยูเอส-บีเพลได้ เพื่อช่วยลดระยะเวลาในการเตรียมการทดสอบให้แก่ผู้ทดสอบ ซึ่งกรณีทดสอบจะสร้างขึ้นจากเส้นทางทดสอบระหว่างไฟล์ดับเบิลยูเอส-บีเพลที่ครอบคลุมทางเดินทดสอบในระดับกิ่ง (Branch Coverage) ประกอบกับข้อมูลเพิ่มเติมจากไฟล์ดับเบิลยูเอสดีแอล (WSDL) และไฟล์เอกซ์เอสดี (XSD: XML Schema Definition Language) เพื่อกำหนดขอบเขตการสุ่มค่าข้อมูลนำเข้าสำหรับกรณีทดสอบที่สร้างขึ้น โดยเครื่องมือที่พัฒนาไม่สามารถสร้างกรณีทดสอบที่มีการเรียกใช้ฟังก์ชันหรือเว็บเซอร์วิสที่มีการตรวจสอบข้อมูลในฐานข้อมูล และการใช้งานข้อมูลนำเข้าที่มาจากข้อมูลส่งออกของเว็บเซอร์วิสก่อนหน้าได้ เนื่องจากขั้นตอนการสร้างกรณีทดสอบเกิดขึ้นจากการอ่านแท็กเอกซ์เอ็มแอลของไฟล์ดับเบิลยูเอส-บีเพลซึ่งไม่มีการเรียกใช้ฟังก์ชันหรือเว็บเซอร์วิสจริง

1.2. วัตถุประสงค์ของงานวิจัย

เพื่อศึกษา ออกแบบ และพัฒนาเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิต

1.3. ขอบเขตของงานวิจัย

- 1) สร้างกรณีทดสอบที่ครอบคลุมเส้นทางทดสอบระหว่างไฟล์ดับเบิลยูเอส-บีเพลที่ครอบคลุมทางเดินทดสอบในระดับกิ่ง
- 2) สร้างกรณีทดสอบที่มีจำนวนไซโคลเมติกภายในไฟล์ดับเบิลยูเอส-บีเพลไม่เกิน 10
- 3) เครื่องมือที่พัฒนารองรับการนำเข้าไฟล์ในรูปแบบเอกซ์เอ็มแอล ดังต่อไปนี้

- ไฟล์ดับเบิลยูเอส-บีเพล เวอร์ชัน 2.0
 - ไฟล์ดับเบิลยูเอสดีแอล เวอร์ชัน 1.1
 - ไฟล์เอกซ์เอสดี เวอร์ชัน 1.0
- 4) เครื่องมือที่พัฒนารองรับข้อมูลประเภทพื้นฐาน (Simple Type) ที่เป็นตัวอักษร (String) ตัวเลขจำนวนเต็ม (Integer) ตัวเลขทศนิยม (Decimal) บูลีน (Boolean) วันที่ (Date) และ เวลา (Time)
 - 5) เครื่องมือที่พัฒนารองรับข้อมูลประเภทซับซ้อน (Complex Type) ที่ประกอบไปด้วยข้อมูลประเภทพื้นฐานที่เป็นตัวอักษร ตัวเลขจำนวนเต็ม ตัวเลขทศนิยม บูลีน วันที่ และเวลา
 - 6) เครื่องมือที่พัฒนาสามารถสร้างกรณีทดสอบที่ไม่มีการเรียกใช้ฟังก์ชันหรือเว็บเซอร์วิสที่มีการตรวจสอบข้อมูลในฐานข้อมูล
 - 7) เครื่องมือที่พัฒนาสามารถสร้างกรณีทดสอบที่ไม่มีการใช้ฟังก์ชันหรือเว็บเซอร์วิสที่มีข้อมูลนำเข้ามาจากข้อมูลส่งออกของเว็บเซอร์วิสก่อนหน้า
 - 8) ผู้ทดสอบสามารถนำกรณีทดสอบที่สร้างขึ้นจากเครื่องมือที่พัฒนาไปรันบนบีเพลเอนจินของ Oracle JDeveloper
 - 9) ดำเนินการทดสอบเครื่องมือที่พัฒนา โดยใช้กรณีศึกษาอย่างน้อย 3 กรณีศึกษา

1.4. ขั้นตอนและวิธีการดำเนินงานวิจัย

- 1) ศึกษาการทำงานของดับเบิลยูเอส-บีเพล
- 2) ศึกษาวิธีสร้างกราฟการเรียกเชิงสถิติที่มีการเรียกใช้งานกันระหว่างไฟล์ดับเบิลยูเอส-บีเพล
- 3) ศึกษางานวิจัยที่เกี่ยวข้อง
- 4) รวบรวมข้อมูลและศึกษาความเป็นไปได้ของงานวิจัย
- 5) ออกแบบภาพรวมและแนวคิดการดำเนินงานวิจัย
- 6) กำหนดคุณลักษณะและขอบเขตของงานวิจัย
- 7) ดำเนินการวิจัย โดยทำการพัฒนาเครื่องมือเพื่อสร้างกรณีทดสอบจากกราฟการเรียกเชิงสถิติสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพล
- 8) ทดสอบเครื่องมือที่พัฒนา ประเมินผลที่ได้จากการทดสอบ และปรับปรุงการทำงานของเครื่องมือ
- 9) สรุปผลการดำเนินงานวิจัยและข้อเสนอแนะ
- 10) จัดทำรูปเล่มวิทยานิพนธ์

1.5. ประโยชน์ที่คาดว่าจะได้รับ

- 1) วิธีสร้างกรณีทดสอบจากกราฟการเรียกเชิงสถิติสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพล
- 2) เครื่องมือสร้างกรณีทดสอบจากกราฟการเรียกเชิงสถิติสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพล
- 3) ช่วยลดระยะเวลาในการเตรียมการทดสอบให้แก่ผู้ทดสอบซอฟต์แวร์

1.6. บทความวิชาการที่ได้รับการตีพิมพ์

งานวิจัยนี้ได้รับคัดเลือกและตีพิมพ์เป็นบทความวิชาการเรื่อง "Test Case Generation for WS-BPEL from a Static Call Graph" โดย Wareerat Bousanoh และ Taratip Suwannasart ในการประชุมวิชาการ "The 11th International Conference on Computer and Electrical Engineering (ICCEE 2018)" ระหว่างวันที่ 12 - 14 ตุลาคม 2561 ณ.กรุงโตเกียว ประเทศญี่ปุ่น



บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

เพื่อให้ทราบถึงแนวทางการวิจัยที่สามารถทำได้ จึงจำเป็นต้องศึกษาวิธีการหรือทฤษฎีต่าง ๆ ที่เป็นประโยชน์ต่อการดำเนินการวิจัย ซึ่งทฤษฎีต่าง ๆ ที่เกี่ยวข้องกับงานวิจัยมีดังต่อไปนี้

2.1.1 ดับเบิลยูเอส-บีเพล

ดับเบิลยูเอส-บีเพล [2] เป็นภาษาที่ใช้กำหนดโมเดลและหลักการในการอธิบายพฤติกรรมของกระบวนการธุรกิจที่มีกระบวนการทำงานร่วมกันระหว่างผู้ใช้งานกับเว็บเซอร์วิสที่เกี่ยวข้อง ถูกพัฒนาขึ้นเพื่อเป็นภาษามาตรฐานโดยองค์กรโอเอซิส และเริ่มดำเนินการเผยแพร่เมื่อปี ค.ศ. 2007 โดยมีโครงสร้างอยู่ในรูปแบบของภาษาเอกซ์เอ็มแอล

ดับเบิลยูเอส-บีเพลสามารถนำกระบวนการทางธุรกิจกลับมาใช้งานใหม่ได้ โดยมีฟังก์ชันในการจัดทำกระบวนการย่อยที่มีการแบ่งกระบวนการที่ลักษณะการทำงานเหมือน ๆ กันออกมาจากกระบวนการหลัก ซึ่งกระบวนการย่อยในดับเบิลยูเอส-บีเพลถูกกำหนดอยู่ในกระบวนการของบีเพลเวอร์ชัน 2.0 [3]

ในการใช้งานกระบวนการย่อยนั้น มีลักษณะการเรียกใช้งานระหว่างดับเบิลยูเอส-บีเพล ซึ่งในงานวิจัยนี้รองรับกระบวนการย่อยใน 2 รูปแบบ คือ กระบวนการย่อยภายใน (Inline Subprocess) และกระบวนการย่อยภายนอก (Standalone Subprocess) โดยกระบวนการย่อยทั้ง 2 รูปแบบนั้น จะปรากฏเป็นแท็กเอกซ์เอ็มแอล <bpelx> ในไฟล์บีเพล โดยมีรายละเอียดดังต่อไปนี้

- 1) **กระบวนการย่อยภายใน** เป็นกระบวนการย่อยที่รวมกันในไฟล์บีเพลเดียวกัน โดยจะปรากฏแท็กเอกซ์เอ็มแอล <bpelx:inlineSubprocess> ซึ่งอยู่ภายใต้แท็ก <process> แสดงดังรูปที่ 2-1
- 2) **กระบวนการย่อยภายนอก** เป็นไฟล์บีเพลย่อยที่มีนามสกุลของไฟล์เป็น .sbpel โดยจะปรากฏแท็กเอกซ์เอ็มแอล <bpelx:subProcess> ในไฟล์บีเพลย่อย แสดงดังรูปที่ 2-2 ส่วนไฟล์บีเพลที่เรียกใช้ไฟล์บีเพลย่อยนั้น จะปรากฏด้วยแท็กเอกซ์เอ็มแอล <bpel:call> ซึ่งจะอยู่ภายใต้แท็กเอกซ์เอ็มแอล <extensionActivity> แสดงดังรูปที่ 2-3

```

<process name="NCName" targetNamespace="anyURI"
  xmlns="http://docs.oasis-open.org/wsbpel/2.0/process/executable" ...>
  <!--
  ! All sub-process definitions must appear prior to the WS-BPEL artifacts of
  ! the process definition.
  -->

  <!-- Inline sub-process definition at process scope -->
  <bpelx:inlineSubProcess xmlns:bpelx="http://schemas.oracle.com/bpel/extension"
    name="NCName">*
    ...
    <!-- Partner links and variables serve as sub-process arguments -->
    <partnerLinks?>
      <partnerLink name="NCName" partnerLinkType="QName" myRole="NCName"?
        partnerRole="NCName"?
        bpelx:argumentRequired=["yes"|"no"]? />+
    </partnerLinks>
    ...
    <variables?>
      <variable name="BPELVariableName" messageType="QName"? type="QName"?
        element="QName"?
        bpelx:argumentRequired=["yes"|"no"]?>+
        from-spec?
      </variable>
    </variables>
    ...
    <!--
    ! Standard process activity graph here, except that no <receive> or <pick>
    ! activities with createInstance = "yes" are allowed.
    -->
    activity
  </bpelx:inlineSubProcess>

  <!--
  ! BPEL code stripped for brevity
  -->
</process>

```

รูปที่ 2-1 ไฟล์บีเพลที่มีกระบวนการย่อยภายใน

จุฬาลงกรณ์มหาวิทยาลัย

```

<?xml version = "1.0" encoding = "UTF-8" ?>
<!--
////////////////////////////////////
Oracle JDeveloper BPEL Designer

Created: Mon Jul 13 02:23:29 ICT 2020
Author: wareer
Type: BPEL 2.0 Subprocess
////////////////////////////////////
-->
<bpelx:subProcess name="sbp_VoucherCashback"
  targetNamespace="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_VoucherCashback"
  xmlns="http://docs.oasis-open.org/wsbpel/2.0/process/executable"
  xmlns:client="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_VoucherCashback"
  xmlns:bpelx="http://schemas.oracle.com/bpel/extension"
  xmlns:bpel="http://docs.oasis-open.org/wsbpel/2.0/process/executable"
  xmlns:ora="http://schemas.oracle.com/xpath/extension"
  xmlns:ui="http://xmlns.oracle.com/soa/designer"
  xsi:type="bpel:tProcess" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">

```

รูปที่ 2-2 ไฟล์บีเพลย่อย sbp_VoucherCashback

```

<sequence name="Sequence2">
  <extensionActivity>
    <bpelx:call name="Call_VoucherCashback"
      xmlns:sp1="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_VocherCashback"
      target="sp1:fbp_VocherCashback">
      <bpelx:param name="inp_price" copyByValue="yes" variable="tmp_NetPrice"/>
      <bpelx:param name="out_voucher" copyByValue="no" variable="tmp_VoucherVal"/>
      <bpelx:param name="inp_TP" copyByValue="no" variable="tmp_TestPath"/>
    </bpelx:call>
  </extensionActivity>
</sequence>

```

รูปที่ 2-3 ไฟล์บีเพลที่มีการเรียกใช้บีเพलय่อย sbp_VoucherCashback

2.1.2 ดับเบิลยูเอสดีแอล

ดับเบิลยูเอสดีแอล [4] คือภาษามาตรฐานที่ใช้ในการอธิบายรายละเอียดของเว็บเซอร์วิสและการติดต่อสื่อสารกันระหว่างเว็บเซอร์วิส โดยมีโครงสร้างอยู่ในรูปแบบของภาษาเอกซ์เอ็มแอล อยู่ภายใต้การดูแลจากกลุ่มดับเบิลยูทีทีซี (W3C: World Wide Web Consortium) [5] โดยในงานวิจัยจะใช้ดับเบิลยูเอสดีแอลเวอร์ชัน 1.1 ที่ถูกเผยแพร่ในปี ค.ศ. 2001 โดยตัวอย่างไฟล์ดับเบิลยูเอสดีแอลแสดงดังรูปที่ 2-4

```

<?xml version="1.0"?>
<definitions name="StockQuote"

  targetNamespace="http://example.com/stockquote/definitions"
  xmlns:tns="http://example.com/stockquote/definitions"
  xmlns:xsd1="http://example.com/stockquote/schemas"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns="http://schemas.xmlsoap.org/wsdl/">

  <import namespace="http://example.com/stockquote/schemas"
    location="http://example.com/stockquote/stockquote.xsd"/>

  <message name="GetLastTradePriceInput">
    <part name="body" element="xsd1:TradePriceRequest"/>
  </message>

  <message name="GetLastTradePriceOutput">
    <part name="body" element="xsd1:TradePrice"/>
  </message>

  <portType name="StockQuotePortType">
    <operation name="GetLastTradePrice">
      <input message="tns:GetLastTradePriceInput"/>
      <output message="tns:GetLastTradePriceOutput"/>
    </operation>
  </portType>
</definitions>

```

รูปที่ 2-4 ตัวอย่างไฟล์ดับเบิลยูเอสดีแอล

2.1.3 เอกซ์เอสดี

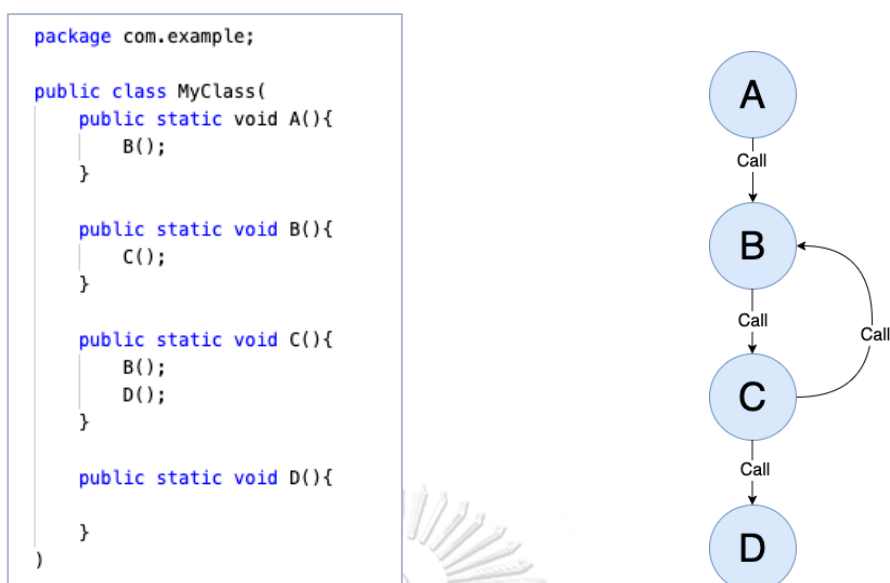
เอกซ์เอสดี [6] คือภาษามาตรฐานสำหรับการอธิบายรูปแบบและเงื่อนไขของข้อมูล โดยมีโครงสร้างอยู่ในรูปแบบของภาษาเอกซ์เอ็มแอล อยู่ภายใต้การดูแลจากกลุ่มดับเบิลยูทีซี [5] ปัจจุบันเป็นเวอร์ชันที่ 1.1 ถูกเผยแพร่ในปี ค.ศ. 2012 โดยตัวอย่างไฟล์เอกซ์เอสดี [7] แสดงดังรูปที่ 2-5

```
<?xml version="1.0"?>
<schema targetNamespace="http://example.com/stockquote/schemas"
  xmlns="http://www.w3.org/2000/10/XMLSchema">
  <element name="TradePriceRequest">
    <complexType>
      <all>
        <element name="tickerSymbol" type="string"/>
      </all>
    </complexType>
  </element>
  <element name="TradePrice">
    <complexType>
      <all>
        <element name="price" type="float"/>
      </all>
    </complexType>
  </element>
</schema>
```

รูปที่ 2-5 ตัวอย่างไฟล์เอกซ์เอสดี

2.1.4 กราฟการเรียกเชิงสถิติ

กราฟการเรียกเชิงสถิติ เป็นกราฟที่แสดงความสัมพันธ์ระหว่างฟังก์ชันที่เรียกใช้งานกัน [8] โดยจำลองความสัมพันธ์เป็นโหนดแทนการนำเสนอแต่ละฟังก์ชัน และเส้นเชื่อมแทนความสัมพันธ์ระหว่างโหนดที่มีการเรียกใช้งานกันระหว่างฟังก์ชัน ซึ่งกิจกรรม (Activity) ที่อยู่ในไฟล์บีเพลเปรียบเสมือนฟังก์ชันการทำงานในโปรแกรมทั่วไป ดังนั้นแต่ละโหนดของกราฟจะแทนกิจกรรมที่อยู่ในไฟล์บีเพล และเส้นเชื่อมจะแทนความสัมพันธ์ระหว่างโหนดที่มีการเรียกใช้งานกัน ตัวอย่างการเรียกใช้งานฟังก์ชันในภาษาจาวา แสดงดังรูปที่ 2-6 (ก) ที่มีฟังก์ชัน A เรียกใช้ฟังก์ชัน B ส่วนฟังก์ชัน B เรียกใช้ฟังก์ชัน C และฟังก์ชัน C เรียกใช้งานฟังก์ชัน B และ D ตามลำดับ จะสามารถสร้างกราฟการเรียกเชิงสถิติที่อธิบายความสัมพันธ์ของฟังก์ชันทั้งสามได้ แสดงดังรูปที่ 2-6 (ข)



รูปที่ 2-6 (ก) การเรียกใช้ฟังก์ชันในภาษาจาวา รูปที่ 2-6 (ข) กราฟการเรียกเชิงสถิติ

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 A Design of WS-BPEL Test Case Generation Tool Based on Path Conditions

งานวิจัยโดย Preecha Nakngern และ Taratip Suwannasart (2017) [9] นำเสนอแนวคิดการออกแบบ และฟังก์ชันการทำงานของเครื่องมือที่ใช้สร้างกรณีทดสอบสำหรับดับเบิลยูเอส-บีเพลโดยอัตโนมัติ ซึ่งมีขั้นตอนการทำงานเริ่มจาก นำเข้าข้อมูล ตรวจสอบไฟล์ดับเบิลยูเอส-บีเพลที่นำเข้า ตรวจสอบเว็บเซอร์วิสที่นำเข้า สร้างกราฟการไหลของการควบคุม (Control Flow Graph) สร้างกรณีทดสอบ แทรกรหัสต้นทางในไฟล์ดับเบิลยูเอส-บีเพล และทดสอบดับเบิลยูเอส-บีเพล

เนื่องจากงานวิจัยนี้เป็นงานวิจัยที่นำเสนอแนวคิดและเครื่องมือที่ใช้ในการสร้างกรณีทดสอบสำหรับดับเบิลยูเอส-บีเพล ซึ่งมุ่งเน้นเพียงแค่กระบวนการทางธุรกิจภายในไฟล์ดับเบิลยูเอส-บีเพลในไฟล์เดียวเท่านั้น ไม่สามารถสร้างกรณีทดสอบจากการเรียกใช้กระบวนการทางธุรกิจระหว่างไฟล์ดับเบิลยูเอส-บีเพลหลาย ๆ ไฟล์ได้ อย่างไรก็ตามแนวคิดของงานวิจัยนี้สามารถนำมาประยุกต์ใช้และต่อยอดแนวความคิดเพิ่มเติมได้

2.2.2 Automatic Test Case Generation for Orchestration Languages at Service Oriented Architecture

งานวิจัยโดย Ebrahim Shamsoddin-Motlagh (2013) [10] นำเสนอแนวทางการสร้างกรณีทดสอบของระบบสถาปัตยกรรมเชิงบริการโดยอัตโนมัติ แสดงดังรูปที่ 2-7 ซึ่งมีขั้นตอนการทำงาน 6 ขั้นตอน ดังนี้

- 1) สร้างกราฟการไหลของบีเพล โดยวิเคราะห์ไฟล์ดับเบิลยูเอส-บีเพล ไฟล์กับเบิลยูเอสดีแอล และไฟล์ดับเบิลยูเอสไอจี (WSIG: Web Service Invoke Graph) จากระบบเซอร์วิซหลัก
- 2) สร้างกรณีทดสอบอย่างสุ่ม
- 3) ดำเนินการอัลกอริทึมบีโอ (BO: Batch-Optimistic) เพื่อให้ได้กรณีทดสอบที่ครอบคลุมที่สุด
- 4) ดำเนินการอัลกอริทึมซีโอ (CO: Close-Up) สำหรับโหนดที่กรณีทดสอบยังไม่ครอบคลุม
- 5) กำจัดกรณีทดสอบบนทางเดินทดสอบที่ซ้ำกัน ถ้ามีกรณีทดสอบที่ซ้ำกันมากกว่า 1 กรณีทดสอบ จะเก็บข้อมูลไว้เพียง 1 กรณีทดสอบเท่านั้น
- 6) แสดงกรณีทดสอบที่ครอบคลุมในระดับโหนด

อย่างไรก็ตามงานวิจัยนี้เป็นงานวิจัยที่นำเสนอแนวทางการสร้างกรณีทดสอบของระบบสถาปัตยกรรมเชิงบริการจากกราฟการไหลของบีเพลและกราฟย่อยที่มีการเรียกใช้งานกันระหว่างเซอร์วิซที่เกี่ยวข้องจากแท็กเอกซ์เอ็มแอล <invoke> แต่สามารถนำงานวิจัยชิ้นนี้มาประยุกต์ใช้ในการสร้างกรณีทดสอบจากกราฟการเรียกเชิงสถิติที่มีการเรียกใช้กระบวนการทางธุรกิจระหว่างไฟล์ดับเบิลยูเอส-บีเพลจากแท็กเอกซ์เอ็มแอล <bpelx:call> ได้

2.2.3 An Approach for Test Case Generation from a Static Call Graph for Object-Oriented Programming

งานวิจัยโดย Sitdhibong Laokok และ Taratip Suwannasart (2017) [11] นำเสนอวิธีการสร้างกรณีทดสอบจากกราฟการเรียกเชิงสถิติสำหรับการโปรแกรมเชิงวัตถุ ซึ่งประกอบไปด้วยขั้นตอนการทำงานคือ ดึงข้อมูลรหัสต้นทางจากแหล่งที่เก็บข้อมูล เก็บรวบรวมข้อมูลค่าคงที่ สร้างกราฟการไหลของข้อมูลและกราฟการเรียกเชิงสถิติ แทรกชุดคำสั่งในรหัสต้นทาง สร้างกรณีทดสอบ ปรับค่าคาดหวังของผลลัพธ์ ทดสอบซอฟต์แวร์ และเปรียบเทียบผลลัพธ์การทำงาน

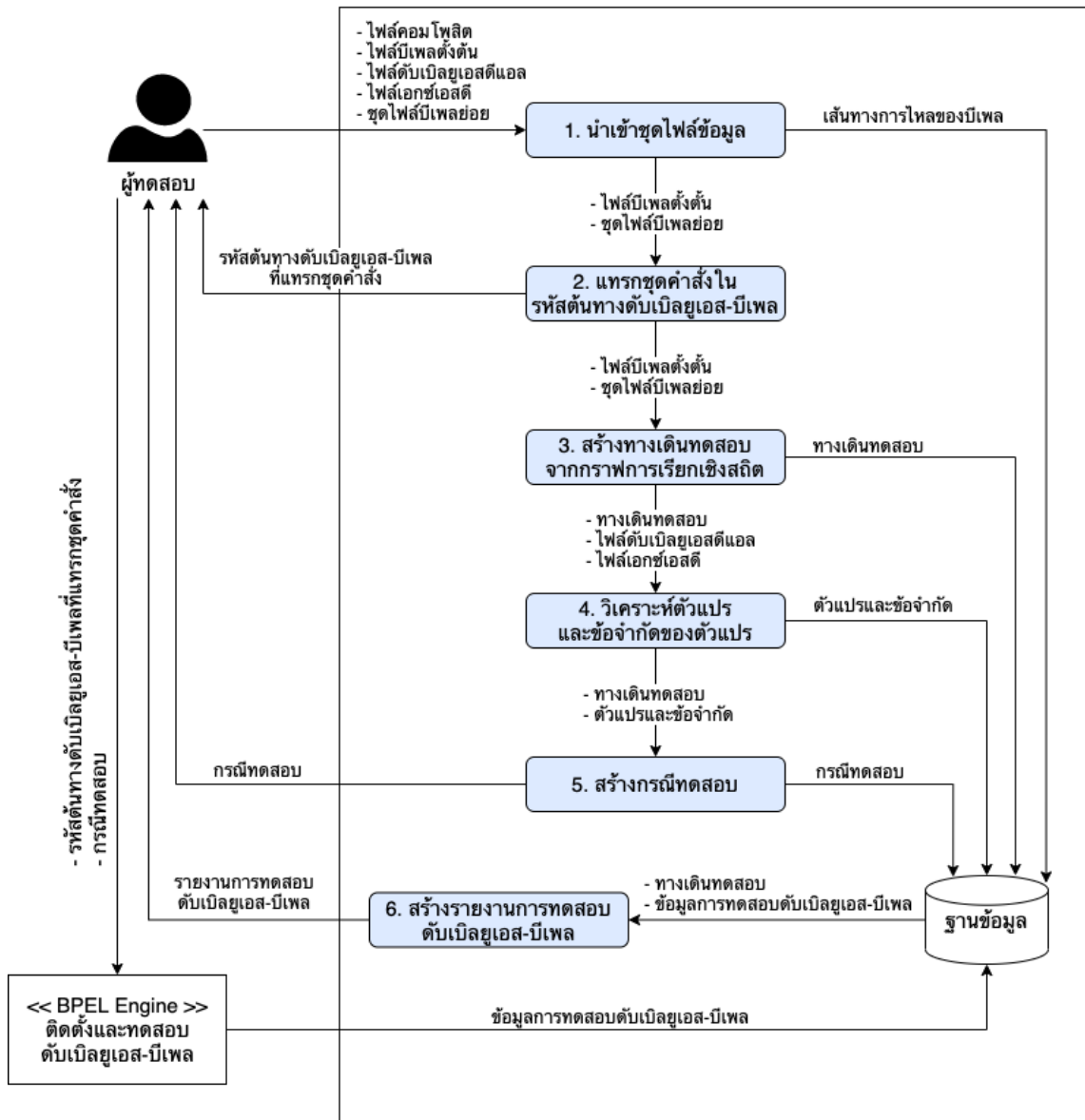
อย่างไรก็ตามงานวิจัยชิ้นนี้เป็นงานวิจัยที่นำเสนอวิธีการสร้างกรณีทดสอบจากกราฟการเรียก
เชิงสถิติสำหรับการโปรแกรมเชิงวัตถุเท่านั้น แต่สามารถนำงานวิจัยชิ้นนี้มาประยุกต์ใช้เพิ่มเติมในการ
นำเสนอวิธีการสร้างกรณีทดสอบจากกราฟเชิงสถิติสำหรับเว็บเซอร์วิสแบบดับเบิ้ลยูเอส-บีเพลได้



บทที่ 3

วิธีการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ

งานวิจัยนี้ได้นำเสนอวิธีการและพัฒนาเครื่องมือสำหรับการสร้างกรณีทดสอบที่ครอบคลุมความสัมพันธ์การเรียกใช้กระบวนการทางธุรกิจระหว่างไฟล์ดับเบิลยูเอส-บีเพลในรูปแบบของกราฟการเรียกเชิงสถิติ โดยภาพรวมของการดำเนินงานวิจัย แสดงดังรูปที่ 3-1 ประกอบไปด้วยขั้นตอนการดำเนินงานทั้งหมด 6 ขั้นตอน ดังนี้



รูปที่ 3-1 ภาพรวมของการดำเนินงานวิจัย

3.1 นำเข้าชุดไฟล์ข้อมูล

ผู้ทดสอบนำเข้าชุดไฟล์ข้อมูลสู่เครื่องมือ เพื่อสร้างโครงการดับเบิลยูเอส-บีเพลและจัดเก็บชุดไฟล์ข้อมูลนำเข้ลงในแฟ้มโครงการที่อยู่บนเว็บเซิร์ฟเวอร์ จากนั้นเครื่องมือจะบันทึกข้อมูลโครงการและชุดไฟล์ข้อมูลนำเข้ลงในฐานข้อมูล โดยชุดไฟล์ข้อมูลนำเข้แบ่งออกเป็น 2 ส่วน คือ ชุดไฟล์ข้อมูลตั้งต้น และชุดไฟล์บีเพลย่อย โดยผู้วิจัยจะใช้ชุดไฟล์นำเข้เหล่านี้เป็นตัวอย่งการทำงานในแต่ละขั้นตอนการดำเนินงาน ซึ่งรายละเอียดของไฟล์นำเข้แต่ละส่วน มีดังนี้

- ชุดไฟล์ข้อมูลตั้งต้น ประกอบไปด้วย ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล ไฟล์บีเพลตั้งต้น ไฟล์ดับเบิลยูเอสดีแอล และไฟล์เอกซ์เอสดี แสดงดังรูปที่ 3-2 ถึงรูปที่ 3-5 ตามลำดับ
- ชุดไฟล์บีเพลย่อย คือ ชุดของไฟล์บีเพลย่อยที่ถูกเรียกใช้โดยไฟล์บีเพลตั้งต้นหรือไฟล์บีเพลย่อยด้วยตัวเอง แสดงดังรูปที่ 3-6 และรูปที่ 3-7

```

1 <?xml version="1.0" encoding="UTF-8" ?>
2 <!-- Generated by Oracle SOA Modeler version 12.2.1.3.0.170820 at [7/13/20 1:33 AM]. -->
3 <composite name="Project1_Promotion" revision="1.0" label="2020-07-13-01-33-29_405" mode="active" state="on"
4   xmlns="http://xmlns.oracle.com/sca/1.0" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:wsp="http://xmlns.oracle.com/sca/1.0/wsp"
5   xmlns:orawsp="http://schemas.oracle.com/ws/2006/01/policy" xmlns:ui="http://xmlns.oracle.com/soa/di/ui/xml"
6   <import namespace="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept"
7     location="WSDLs/ElectronicDept.wsdl" importType="wsdl"/>
8   <service name="electronicdept_client_ep" ui:wSDLLocation="WSDLs/ElectronicDept.wsdl">
9     <interface.wsdl interface="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept#wsdl.in"
10     <binding.ws port="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept#wsdl.endpoint(electronicdept_client_ep)"/>
11   </service>
12   <property name="productVersion" type="xs:string" many="false">12.2.1.3.0.170820</property>
13   <property name="compositeID" type="xs:string" many="false">ed3d9f9f-56ec-41ad-b48d-960aa8c3d6cd</property>
14   <component name="ElectronicDept" version="2.0">
15     <implementation.bpel src="BPEL/ElectronicDept.bpel"/>
16     <componentType>
17       <service name="electronicdept_client" ui:wSDLLocation="WSDLs/ElectronicDept.wsdl">
18         <interface.wsdl interface="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept#wsdl.in"
19         </service>
20         <sca-ext:callReference name="sbp_CheckCCAllow"/>
21       </componentType>
22       <property name="bpel.config.transaction" type="xs:string" many="false">required</property>
23     </component>
24     <component name="sbp_CheckCCAllow">
25       <implementation.sbpel src="BPEL/sbp_CheckCCAllow.sbpel"/>
26       <componentType>
27         <sca-ext:callService name="sbp_CheckCCAllow"/>
28         <sca-ext:callReference name="sbp_VocherCashback"/>
29       </componentType>
30     </component>
31     <component name="sbp_VocherCashback">
32       <implementation.sbpel src="BPEL/sbp_VocherCashback.sbpel"/>
33       <componentType>
34         <sca-ext:callService name="sbp_VocherCashback"/>
35       </componentType>
36     </component>
37   <wire>--
40 </wire>
41 <!-- <sca-ext:callWire name="sbp_CheckCCAllow" wire="sbp_CheckCCAllow" /> -->
44 </sca-ext:callWire>
45 </composite>

```

รูปที่ 3-2 ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล

```

1 <?xml version = "1.0" encoding = "UTF-8" ?>
2 <!-- ...
12 <process name="ElectronicDept" targetNamespace="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept"
13 xmlns="http://docs.oasis-open.org/wsbpel/2.0/process/executable"
14 xmlns:client="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept"
15 xmlns:ora="http://schemas.oracle.com/xpath/extension"
16 xmlns:ui="http://xmlns.oracle.com/soa/designer"
17 xmlns:bpelx="http://schemas.oracle.com/bpel/extension"
18 xmlns:bpel="http://docs.oasis-open.org/wsbpel/2.0/process/executable"
19 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
20 xmlns:oraext="http://www.oracle.com/XSL/Transform/java/oracle.tip.pc.services.functions.ExtFunc"
21 xmlns:bpm="http://xmlns.oracle.com/bpmn20/extensions"
22 xmlns:xp20="http://www.oracle.com/XSL/Transform/java/oracle.tip.pc.services.functions.Xpath20"
23 xmlns:ess="http://xmlns.oracle.com/scheduler"
24 xmlns:hwf="http://xmlns.oracle.com/bpel/workflow/xpath"
25 xmlns:xref="http://www.oracle.com/XSL/Transform/java/oracle.tip.xref.xpath.XRefXPathFunctions"
26 xmlns:dvm="http://www.oracle.com/XSL/Transform/java/oracle.tip.dvm.LookupValue"
27 xmlns:bpws="http://schemas.xmlsoap.org/ws/2003/03/business-process/"
28 xmlns:xdk="http://schemas.oracle.com/bpel/extension/xpath/function/xdk"
29 xmlns:ids="http://xmlns.oracle.com/bpel/services/IdentityService/xpath"
30 xmlns:ldap="http://schemas.oracle.com/xpath/extension/ldap">
31
32 >
35 | importType="http://schemas.xmlsoap.org/wsdl/" />
36 >
42 > <partnerLinks>--
49 </partnerLinks>
50
51 > <!-- ...
57 > <variables>--
63 </variables>
64
65 > <!-- ...
72 <sequence name="main">
73 <!-- Receive input from requestor. (Note: This maps to operation defined in ElectronicDept.wsdl) -->
74 <receive name="receiveInput" partnerLink="electronicdept_client" portType="client:ElectronicDept"
75 | operation="process" variable="inputVariable" createInstance="yes"/>
76 > <assign name="Discount5Percent">--
81 </assign>
82 <extensionActivity>
83 <bpelx:call name="Call_CheckCCAllow"
84 | xmlns:sp1="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_CheckCCAllow"
85 | target="sp1:sbp_CheckCCAllow">
86 | <bpelx:param name="inp_sof" copyByValue="yes" variable="$inputVariable.payload/client:SOF"/>
87 | <bpelx:param name="inp_ccType" copyByValue="yes" variable="$inputVariable.payload/client:CCType"/>
88 | <bpelx:param name="inp_price" copyByValue="yes" variable="$inputVariable.payload/client:Price"/>
89 | <bpelx:param name="out_isCCAllow" copyByValue="no" variable="$outputVariable.payload/client:IsAllow"/>
90 | <bpelx:param name="out_voucherValue" copyByValue="no" variable="$outputVariable.payload/client:VoucherValue"/>
91 | </bpelx:call>
92 </extensionActivity>
93 <!-- Generate reply to synchronous request -->
94 <reply name="replyOutput" partnerLink="electronicdept_client" portType="client:ElectronicDept"
95 | operation="process" variable="outputVariable"/>
96 </sequence>
97 </process>

```

รูปที่ 3-3 ไฟล์บีเพลดั้งเดิม

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <wsp:definitions name="ElectronicDept"
3      targetNamespace="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept"
4      xmlns:wsp="http://schemas.xmlsoap.org/wsdl/"
5      xmlns:client="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept"
6      xmlns:plnk="http://docs.oasis-open.org/wsbpel/2.0/plnktype">
7
8      <!-- ~~~~~
9      TYPE DEFINITION - List of services participating in this BPEL process
10     The default output of the BPEL designer uses strings as input and
11     output to the BPEL Process. But you can define or import any XML
12     Schema type and use them as part of the message types.
13     ~~~~~ -->
14     <wsp:types>
15         <schema xmlns="http://www.w3.org/2001/XMLSchema"
16             <import namespace="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept"
17                 schemaLocation="../Schemas/ElectronicDept.xsd" />
18         </schema>
19     </wsp:types>
20
21     <!-- ~~~~~
22     MESSAGE TYPE DEFINITION - Definition of the message types used as
23     part of the port type definitions
24     ~~~~~ -->
25     <wsp:message name="ElectronicDeptRequestMessage">
26         <wsp:part name="payload" element="client:process"/>
27     </wsp:message>
28     <wsp:message name="ElectronicDeptResponseMessage">
29         <wsp:part name="payload" element="client:processResponse"/>
30     </wsp:message>
31
32     <!-- ~~~~~
33     PORT TYPE DEFINITION - A port type groups a set of operations into
34     a logical service unit.
35     ~~~~~ -->
36
37     <!-- portType implemented by the ElectronicDept BPEL process -->
38     <wsp:portType name="ElectronicDept">
39         <wsp:operation name="process">
40             <wsp:input message="client:ElectronicDeptRequestMessage" />
41             <wsp:output message="client:ElectronicDeptResponseMessage"/>
42         </wsp:operation>
43     </wsp:portType>
44
45     <!-- ~~~~~
46     PARTNER LINK TYPE DEFINITION
47     ~~~~~ -->
48     <plnk:partnerLinkType name="ElectronicDept">
49         <plnk:role name="ElectronicDeptProvider" portType="client:ElectronicDept"/>
50     </plnk:partnerLinkType>
51 </wsp:definitions>

```

รูปที่ 3-4 ไฟล์ดับเบิลยูเอสดีแอล

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <schema attributeFormDefault="unqualified"
3    elementFormDefault="qualified"
4    targetNamespace="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept"
5    xmlns="http://www.w3.org/2001/XMLSchema"
6    xmlns:kt="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept">
7    <simpleType name="CreditCardType" id="CreditCardType" final="restriction">
8      <restriction base="string">
9        <enumeration value="VISA" />
10       <enumeration value="MASTER" />
11       <enumeration value="JCB" />
12       <enumeration value="AMEX" />
13       <enumeration value="UNIONPAY" />
14     </restriction>
15   </simpleType>
16   <simpleType name="SourceOfFund" id="SourceOfFund" final="restriction">
17     <restriction base="string">
18       <enumeration value="CASH" />
19       <enumeration value="CREDITCARD" />
20     </restriction>
21   </simpleType>
22   <element name="process">
23     <complexType>
24       <sequence>
25         <element name="Price">
26           <simpleType>
27             <restriction base="decimal">
28               <totalDigits value="10" />
29               <fractionDigits value="2" />
30             </restriction>
31           </simpleType>
32         </element>
33         <element name="SOF" type="kt:SourceOfFund" />
34         <element name="CCNo">
35           <simpleType>
36             <restriction base="string">
37               <pattern value="[0-9]{16}" />
38             </restriction>
39           </simpleType>
40         </element>
41         <element name="CCType" type="kt:CreditCardType" />
42       </sequence>
43     </complexType>
44   </element>
45   <element name="processResponse">
46     <complexType>
47       <sequence>
48         <element name="IsAllow" type="boolean" />
49         <element name="NetPrice" type="decimal" default="0" />
50         <element name="VoucherValue" type="integer" default="0" />
51       </sequence>
52     </complexType>
53   </element>
54 </schema>

```

รูปที่ 3-5 ไฟล์เอกซ์เอสดี


```

1  <?xml version = "1.0" encoding = "UTF-8" ?>
2  > <!-- ...
11 <bpelx:subProcess name="sbp_CheckCCAllow"
12   targetNamespace="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_CheckCCAllow" xmlns="http://docs.c
13   xmlns:client="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_CheckCCAllow" xmlns:bpelx="http://sch
14   xmlns:bpel="http://docs.oasis-open.org/wsbpel/2.0/process/executable" xmlns:ora="http://schemas.oracle.com/xpat
15   xmlns:ui="http://xmlns.oracle.com/soa/designer" xsi:type="bpel:tProcess" xmlns:xsi="http://www.w3.org/2001/XMLS
16   xmlns:xsd="http://www.w3.org/2001/XMLSchema">
17
18 > <variables> --
24 </variables>
25
26 <sequence name="main">
27 <if name="IfCreditCard">
28 <documentation> --
30 </documentation>
31 <condition>$inp_sof == 'CREDITCARD'</condition>
32 <sequence name="Sequence1">
33 <if name="IfVISA">
34 <documentation> --
36 </documentation>
37 <condition>$inp_ccType == 'VISA'</condition>
38 <sequence name="Sequence4">
39 <assign name="assAllowVisa"> --
44 </assign>
45 <extensionActivity>
46 <bpelx:call name="Call_VoucherCashback"
47   xmlns:sp1="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_VocherCashback"
48   target="sp1:sbp_VocherCashback">
49   <bpelx:param name="inp_price" copyByValue="yes" variable="$inp_price"/>
50   <bpelx:param name="out_voucher" copyByValue="no" variable="$out_vocherValue"/>
51 </bpelx:call>
52 </extensionActivity>
53 </sequence>
54 <else>
55 <documentation> --
57 </documentation>
58 <sequence name="Sequence3"> --
65 </sequence>
66 </else>
67 </if>
68 </sequence>
69 <else>
70 <documentation> --
72 </documentation>
73 <sequence name="Sequence2"> --
80 </sequence>
81 </else>
82 </if>
83 </sequence>
84 </bpelx:subProcess>

```

รูปที่ 3-6 ไฟล์บีเพลย่อย sbp_CheckCCAllow

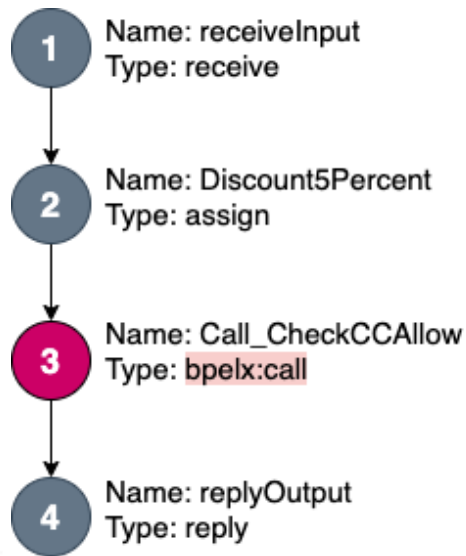
```

1  <?xml version = "1.0" encoding = "UTF-8" ?>
2  > <!--
11 <bpelx:subProcess name="sbp_VocherCashback"
12   targetNamespace="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_VocherCashback"
13   xmlns="http://docs.oasis-open.org/wsbpel/2.0/process/executable"
14   xmlns:client="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_VocherCashback"
15   xmlns:bpelx="http://schemas.oracle.com/bpel/extension"
16   xmlns:bpel="http://docs.oasis-open.org/wsbpel/2.0/process/executable"
17   xmlns:ora="http://schemas.oracle.com/xpath/extension"
18   xmlns:ui="http://xmlns.oracle.com/soa/designer"
19   xsi:type="bpel:tProcess" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
20   xmlns:xsd="http://www.w3.org/2001/XMLSchema">
21
22 <variables>
23   <variable name="inp_price" type="xsd:decimal"/>
24   <variable name="out_voucher" type="xsd:integer"/>
25 </variables>
26
27 <sequence name="main">
28   <if name="IfPrice2500Up">
29     <documentation>--
31     </documentation>
32     <condition>$inp_price &gt;= 2500</condition>
33     <sequence name="Sequence1">
34       <if name="IfPrice5000Up">
35         <documentation>--
37         </documentation>
38         <condition>$inp_price &gt;= 5000</condition>
39         <sequence name="Sequence4">
40           <if name="IfPrice10000Up">--
42           </if>
44         </sequence>
45       </else>
46       <documentation>--
48       </documentation>
49       <sequence name="Sequence3">--
51       </sequence>
52     </else>
53   </if>
54 </sequence>
55 <else>
56   <documentation>--
58   </documentation>
59   <sequence name="Sequence2">--
61   </sequence>
62 </else>
63 </if>
64 </sequence>
65 </bpelx:subProcess>

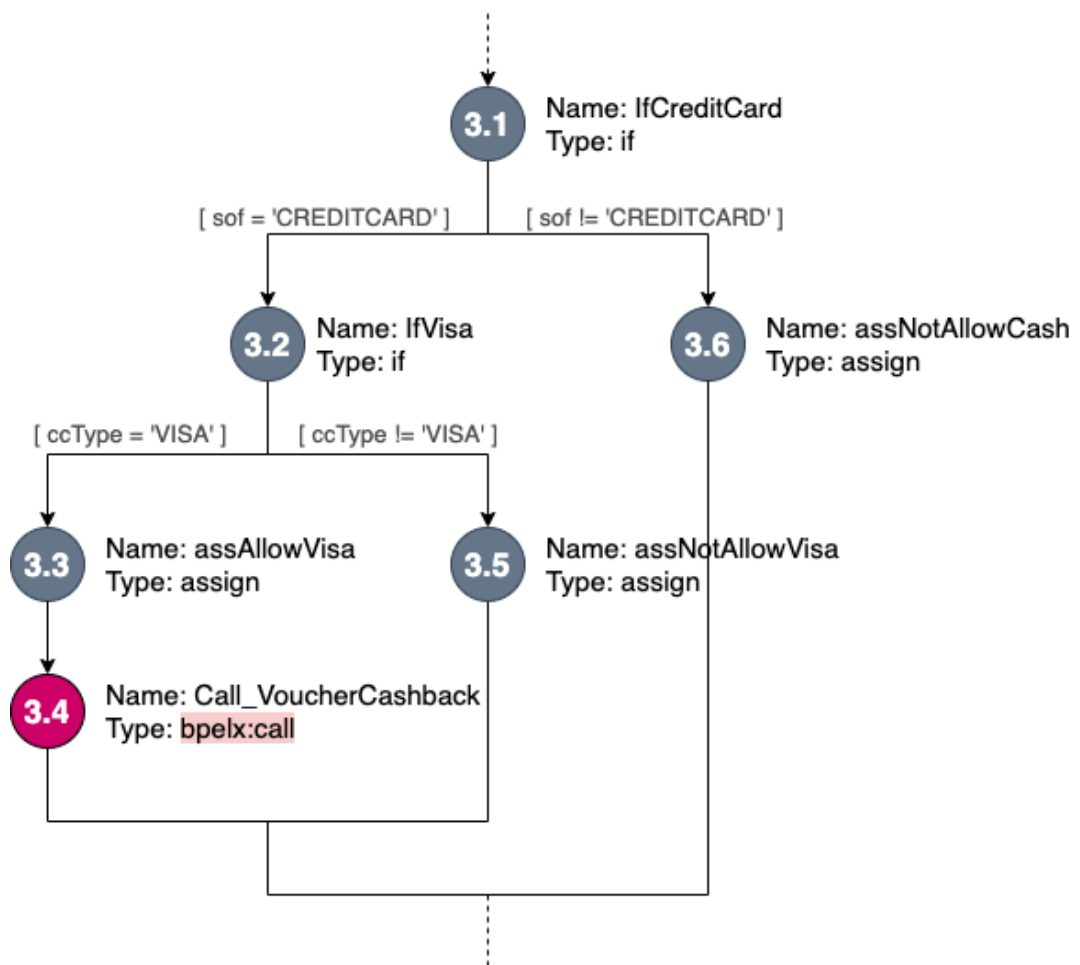
```

รูปที่ 3-7 ไฟล์บีเพलय่อย sbp_VocherCashback

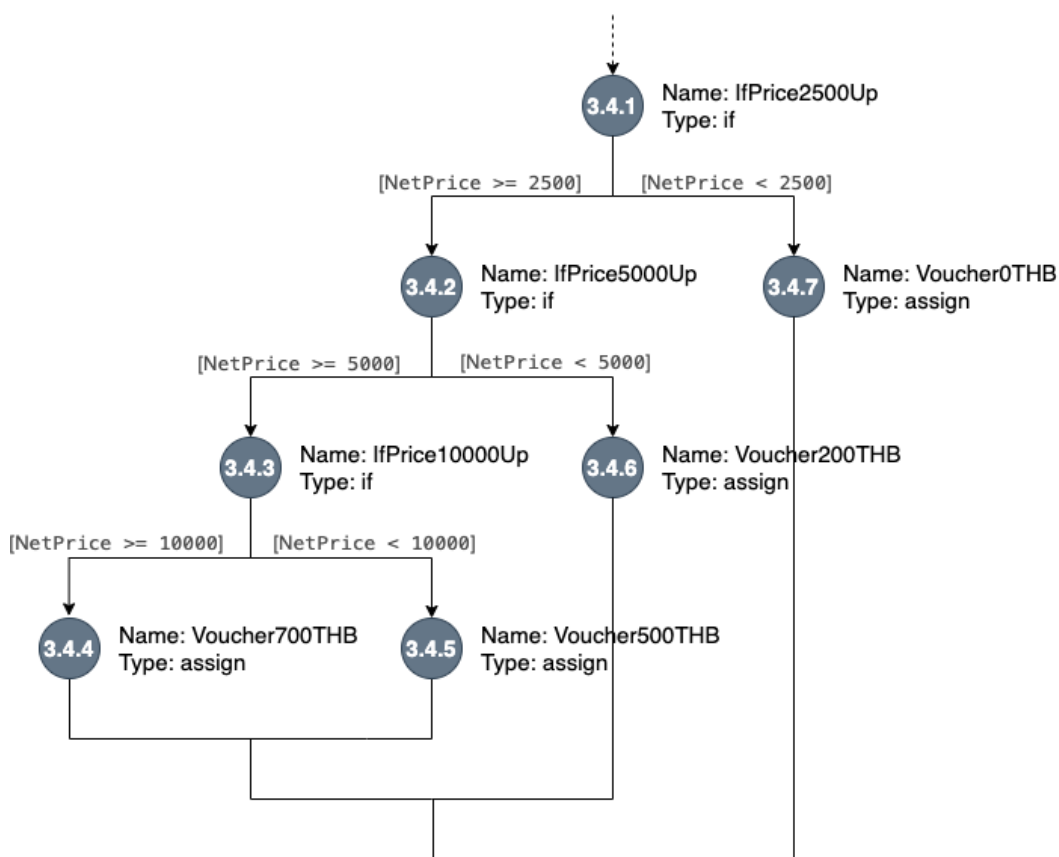
หลังจากผู้ทดสอบนำเข้าสู่ชุดไฟล์ข้อมูลเรียบร้อยแล้ว เครื่องมือจะอ่านแท็กเอกซ์เอ็มแอลของไฟล์บีเพลดั้งเดิม ดังรูปที่ 3-3 และชุดไฟล์บีเพलय่อย ดังรูปที่ 3-6 และรูปที่ 3-7 พร้อมทั้งพิจารณาความสัมพันธ์ในแต่ละโหนด เพื่อค้นหาเส้นทางการไหลของบีเพล และบันทึกเส้นทางการไหลของบีเพลลงในฐานข้อมูล โดยเส้นทางการไหลของบีเพลในไฟล์บีเพลดั้งเดิมและชุดไฟล์บีเพलय่อย แสดงดังรูปที่ 3-8 ถึงรูปที่ 3-10



รูปที่ 3-8 เส้นทางการไหลของบีเพลในไฟล์บีเพลตั้งต้น



รูปที่ 3-9 เส้นทางการไหลของบีเพลในไฟล์บีเพलय่อย sbp_CheckCCAllow



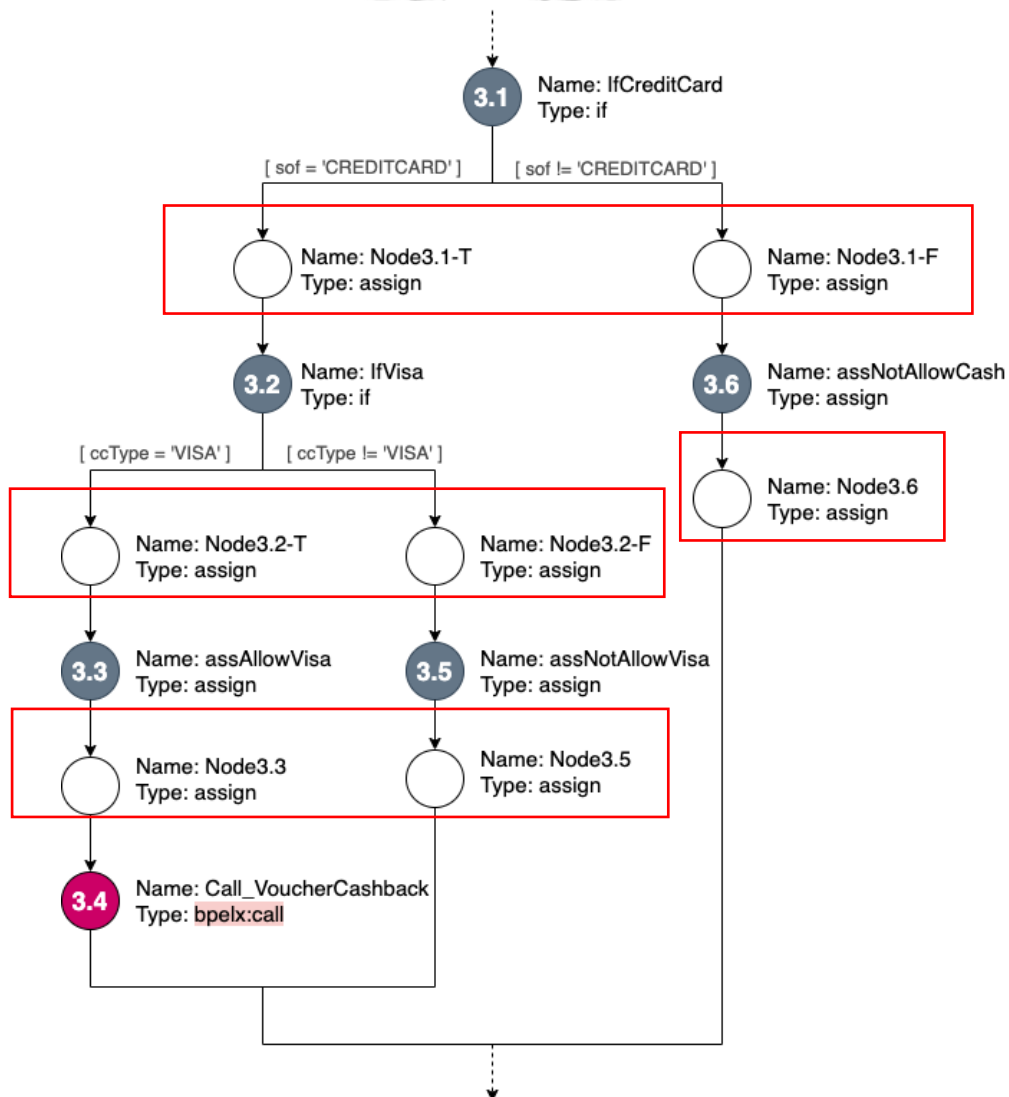
รูปที่ 3-10 เส้นทางการไหลของบีเพลในไฟล์บีเพลย่อย sbp_VoucherCashback

3.2 แทรกชุดคำสั่งในรหัสต้นทางดับเบิลยูเอส-บีเพล

การแทรกชุดคำสั่งในรหัสต้นทางเพื่อจัดเก็บข้อมูลการทดสอบดับเบิลยูเอส-บีเพล โดยในขั้นตอนนี้เครื่องมือจะแทรกชุดคำสั่งในรหัสต้นทางของไฟล์บีเพลตั้งต้นและไฟล์บีเพลย่อยที่ถูกเรียกใช้งาน โดยเมื่อเครื่องมือทำการแทรกชุดคำสั่งในรหัสต้นทางเสร็จสิ้น ผู้ทดสอบสามารถดาวน์โหลดรหัสต้นทางที่แทรกชุดคำสั่งไปติดตั้งบนบีเพลเอนจิน

ตัวอย่างการทำงานในขั้นตอนนี้ เครื่องมืออ่านรหัสต้นทางดับเบิลยูเอส-บีเพล และแทรกรหัสต้นทางด้วยโหนด <assign> เพื่อเก็บค่าการทำงานของกระบวนการดับเบิลยูเอส-บีเพล รหัสต้นทางที่ถูกแทรกชุดคำสั่ง แสดงดังรูปที่ 3-11 (ก) และรูปที่ 3-11 (ข) โดยเครื่องมือจะทำการแทรกโหนด <assign> ภายใต้ 2 เงื่อนไข ดังนี้

- 1) แทรกโหนด <assign> หลังจากการเรียกใช้งานโหนดต่าง ๆ ดังนี้
 - โหนด <receive> โหนดรับค่าข้อมูลนำเข้ากระบวนการดับเบิลยูเอส-บีเพล
 - โหนด <invoke> โหนดที่เรียกใช้งานเว็บเซอร์วิส
 - โหนด <assign> โหนดตั้งค่าข้อมูล
 - โหนด <if> โหนดเงื่อนไขที่เป็นจริง
 - โหนด <else> โหนดเงื่อนไขอื่น ๆ ที่นอกเหนือจากโหนด <if>
 - โหนด <receive> เป็นโหนดที่รับค่าข้อมูลนำเข้าจากผู้ทดสอบ
- 2) แทรกโหนด <assign> ก่อนการเรียกใช้งานโหนดต่าง ๆ ดังนี้
 - โหนด <reply> โหนดส่งค่าข้อมูลส่งออกกระบวนการดับเบิลยูเอส-บีเพล



รูปที่ 3-11 (ก) กราฟการไหลของบีเพลในรหัสต้นทางที่ถูกแทรกชุดคำสั่งด้วยโหนด <assign>

```

48 <sequence name="main">
49   <if name="IfCreditCard">
50     <documentation>
51       <![CDATA[SOFCREDITCARD]]>
52     </documentation>
53     <condition>$inp_sof='CREDITCARD'</condition>
54     <sequence>
55       <assign name="Node3.1-T">
56         <copy>
57           <from>concat($inp_TP,"-", "3.1(T)")</from>
58           <to expressionLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang:xpath1.0">$inp_TP</to>
59         </copy>
60       </assign>
61     <if name="IfVisa">
62       <documentation>--
64     </documentation>
65     <condition>$inp_ccType='VISA'</condition>
66     <assign name="Node3.2-T">
67       <copy>
68         <from>concat($inp_TP,"-", "3.2(T)")</from>
69         <to expressionLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang:xpath1.0">$inp_TP</to>
70       </copy>
71     </assign>
72   <assign name="assAllowVisa">--
77 </assign>
78   <assign name="Node3.3">
79     <copy>
80       <from>concat($inp_TP,"-", "3.3")</from>
81       <to expressionLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang:xpath1.0">$inp_TP</to>
82     </copy>
83   </assign>
84   <extensionActivity>
85     <bpelx:call name="Call_VoucherCashback"--
91   </bpelx:call>
92   </extensionActivity>
93   <else>
94     <documentation>--
96   </documentation>
97   <assign name="Node3.2-F">
98     <copy>
99       <from>concat($inp_TP,"-", "3.2(F)")</from>
100      <to expressionLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang:xpath1.0">$inp_TP</to>
101    </copy>
102  </assign>
103  <assign name="assNotAllowVisa">--
108 </assign>
109  <assign name="Node3.5">
110    <copy>
111      <from>concat($inp_TP,"-", "3.5")</from>
112      <to expressionLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang:xpath1.0">$inp_TP</to>
113    </copy>
114  </assign>
115  </else>
116 </if>
117 </sequence>
118 <else>
119 >
121 </documentation>
122 <assign name="Node3.1-F">
123   <copy>
124     <from>concat($inp_TP,"-", "3.1(F)")</from>
125     <to expressionLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang:xpath1.0">$inp_TP</to>
126   </copy>
127 </assign>
128 >
133 </assign>
134 <assign name="Node3.6">
135   <copy>
136     <from>concat($inp_TP,"-", "3.6")</from>
137     <to expressionLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang:xpath1.0">$inp_TP</to>
138   </copy>
139 </assign>
140 </else>
141 </if>
142 </sequence>
143 </bpelx:subProcess>

```

รูปที่ 3-11 (ข) รหัสต้นทางดับเบิลยูเอส-บีเพลที่ถูกแทรกชุดคำสั่งด้วยโหนด <assign>

จากรูปที่ 3-11 (ก) จะเห็นว่ามีการแทรกโหนด <assign> จำนวน 7 โหนด หลังการเรียกใช้งานโหนดต่าง ๆ และจากรูปที่ 3-11 (ข) จะเห็นว่ามีการแทรกโหนด <assign> ปรากฏอยู่ในบรรทัดที่ 55 66 78 97 109 122 และ 134 เพื่อเก็บค่าการทำงานของกระบวนการดับเบิลยูเอส-บีเพลในตัวแปรที่ชื่อว่า inp_TP และสุดท้ายเมื่อทำการทดสอบดับเบิลยูเอส-บีเพล บีเพลเอนจินจะส่งค่าข้อมูลการทดสอบมาให้เครื่องมือ โดยผ่านเว็บเซอร์วิสตามที่เครื่องมือได้แทรกชุดคำสั่งไว้

3.3 สร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติ

เครื่องมือนำเส้นทางการไหลของบีเพลมาวิเคราะห์และหาความสัมพันธ์การเรียกใช้งานกันระหว่างดับเบิลยูเอส-บีเพล เพื่อสร้างกราฟการเรียกเชิงสถิติและสร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติ พร้อมบันทึกข้อมูลทางเดินทดสอบลงในฐานข้อมูล

การทำงานในขั้นตอนนี้ เริ่มจากการพิจารณาเส้นทางการไหลของบีเพลในไฟล์บีเพลตั้งต้นและชุดไฟล์บีเพลย่อย โดยเมื่อตรวจสอบโหนดการทำงานในไฟล์บีเพลตั้งต้นพบว่าบรรทัดที่ 83 ในรูปที่ 3-12 มีแท็กเอกซ์เอ็มแอล <bpelx:call> ที่เรียกใช้บีเพลย่อย sbp_CheckCCAllow ปรากฏอยู่ และเมื่อตรวจสอบโหนดการทำงานในไฟล์บีเพลย่อย sbp_CheckCCAllow พบว่าบรรทัดที่ 46 ในรูปที่ 3-13 มีแท็กเอกซ์เอ็มแอล <bpelx:call> ที่เรียกใช้บีเพลย่อย sbp_VoucherCashback ปรากฏอยู่ จากนั้นเครื่องมือวิเคราะห์หาความสัมพันธ์ของเส้นทางการไหลระหว่างดับเบิลยูเอส-บีเพล แสดงดังรูปที่ 3-14 (ก) และสร้างกราฟการเรียกเชิงสถิติที่แสดงความสัมพันธ์การเรียกใช้งานกันระหว่างดับเบิลยูเอส-บีเพล แสดงดังรูปที่ 3-14 (ข)

```

65 > <!-- --
72 > <sequence name="main">
73 > <!-- Receive input from requestor. (Note: This maps to operation defined in ElectronicDept.wsdl) -->
74 > <receive name="receiveInput" partnerLink="electronicdept_client" portType="client:ElectronicDept"
75 > | operation="process" variable="inputVariable" createInstance="yes"/>
76 > <assign name="Discount5Percent">--
81 > </assign>
82 > <extensionActivity>
83 > <bpelx:call name="Call_CheckCCAllow"
84 > | xmlns:sp1="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_CheckCCAllow"
85 > | target="sp1:sbp_CheckCCAllow">
86 > | <bpelx:param name="inp_sof" copyByValue="yes" variable="$inputVariable.payload/client:SOF"/>
87 > | <bpelx:param name="inp_ccType" copyByValue="yes" variable="$inputVariable.payload/client:CCType"/>
88 > | <bpelx:param name="inp_price" copyByValue="yes" variable="$inputVariable.payload/client:Price"/>
89 > | <bpelx:param name="out_isCCAllow" copyByValue="no" variable="$outputVariable.payload/client:IsAllow"/>
90 > | <bpelx:param name="out_voucherValue" copyByValue="no" variable="$outputVariable.payload/client:VoucherValue"/>
91 > </bpelx:call>
92 > </extensionActivity>
93 > <!-- Generate reply to synchronous request -->
94 > <reply name="replyOutput" partnerLink="electronicdept_client" portType="client:ElectronicDept"
95 > | operation="process" variable="outputVariable"/>
96 > </sequence>
97 > </process>

```

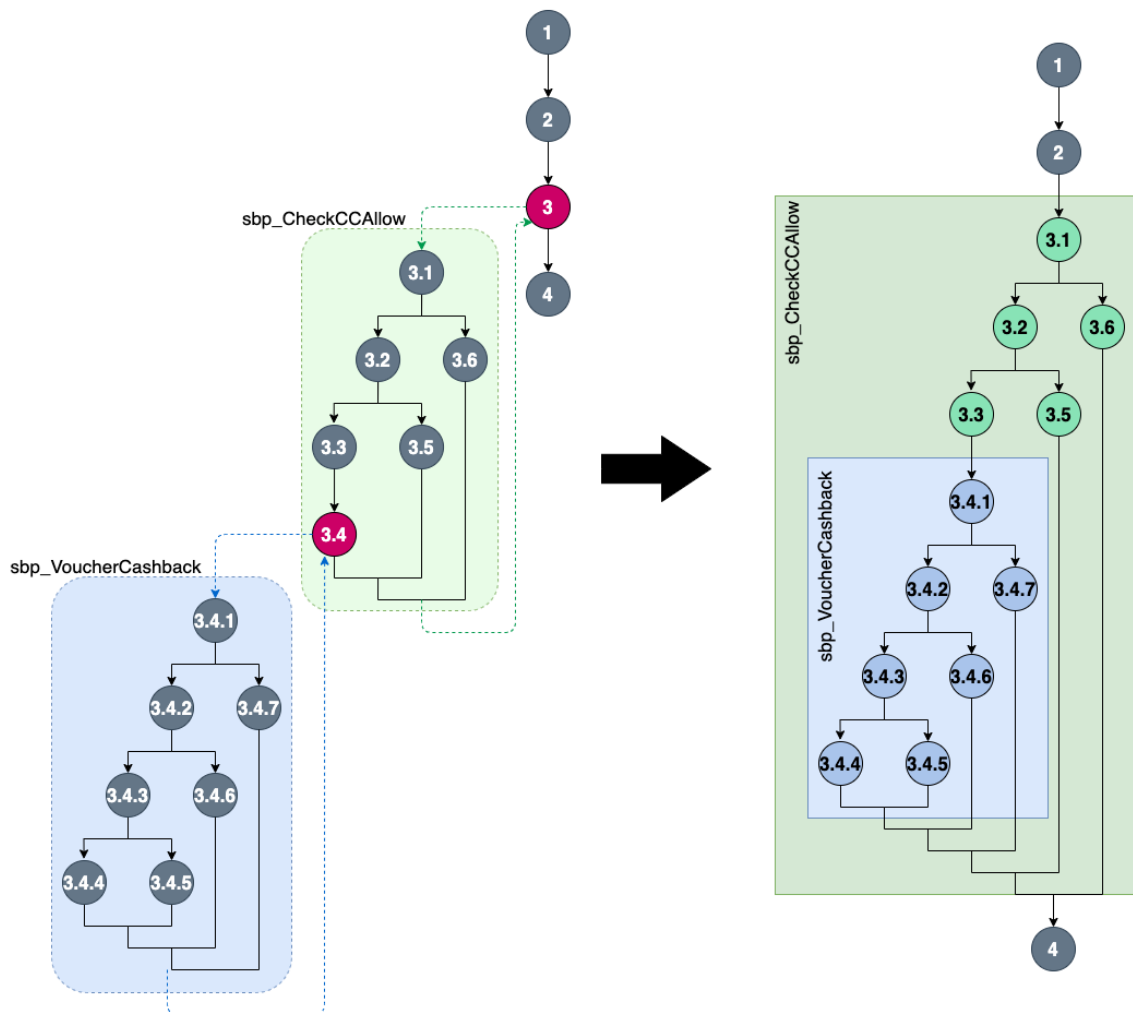
รูปที่ 3-12 ไฟล์บีเพลตั้งต้นที่มีการเรียกใช้บีเพลย่อย sbp_CheckCCAllow

```

26 <sequence name="main">
27   <if name="IfCreditCard">
28     <documentation>-
30     </documentation>
31     <condition>$inp_sof == 'CREDITCARD'</condition>
32     <sequence name="Sequence1">
33       <if name="IfVISA">
34         <documentation>-
36         </documentation>
37         <condition>$inp_ccType == 'VISA'</condition>
38         <sequence name="Sequence4">
39           <assign name="assAllowVisa">-
44           </assign>
45           <extensionActivity>
46             <bpelx:call name="Call_VoucherCashback"
47               xmlns:sp1="http://xmlns.oracle.com/Application1/Project1_Promotion/sbp_VoucherCashback"
48               target="sp1:spb_VoucherCashback">
49               <bpelx:param name="inp_price" copyByValue="yes" variable="$inp_price"/>
50               <bpelx:param name="out_voucher" copyByValue="no" variable="$out_voucherValue"/>
51             </bpelx:call>
52           </extensionActivity>
53         </sequence>

```

รูปที่ 3-13 ไฟล์บีเพलय่อยที่มีการเรียกใช้บีเพलय่อย sbp_VoucherCashback



รูปที่ 3-14 (ก) ความสัมพันธ์ของเส้นทางการไหล
ระหว่างดับเบิลยูเอส-บีเพล

รูปที่ 3-14 (ข) กราฟการเรียกเชิงสถิติในการ
เรียกใช้งานระหว่างดับเบิลยูเอส-บีเพล

จากกราฟการเรียกเชิงสถิติในรูปที่ 3-14 (ข) เครื่องมือจะสร้างทางเดินทดสอบให้ครอบคลุมในระดับกิ่งและบันทึกข้อมูลทางเดินทดสอบลงในฐานข้อมูล ซึ่งจากกราฟการเรียกเชิงสถิติจะได้ทางเดินทดสอบทั้งหมด 6 เส้นทาง ดังนี้

P1: 1 – 2 – 3.1 – 3.2 – 3.3 – 3.4.1 – 3.4.2 – 3.4.3 – 3.4.4 – 4

P2: 1 – 2 – 3.1 – 3.2 – 3.3 – 3.4.1 – 3.4.2 – 3.4.3 – 3.4.5 – 4

P3: 1 – 2 – 3.1 – 3.2 – 3.3 – 3.4.1 – 3.4.2 – 3.4.6 – 4

P4: 1 – 2 – 3.1 – 3.2 – 3.3 – 3.4.1 – 3.4.7 – 4

P5: 1 – 2 – 3.1 – 3.2 – 3.5 – 4

P6: 1 – 2 – 3.1 – 3.6 – 4

3.4 วิเคราะห์ตัวแปรและข้อจำกัดของตัวแปร

ในขั้นตอนนี้เครื่องมือจะอ่านไฟล์ดับเบิลยูเอสดีแอลเพื่อค้นหาตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล จากนั้นวิเคราะห์ข้อมูลตัวแปรและข้อจำกัดจากการอ่านข้อมูลในไฟล์เอกซ์เอสดีและบันทึกข้อมูลตัวแปรและข้อจำกัดลงในฐานข้อมูล รวมถึงการวิเคราะห์เงื่อนไขของตัวแปรในแต่ละโหนดของทางเดินทดสอบ โดยในแต่ละขั้นตอนย่อยมีรายละเอียด ดังนี้

3.4.1 วิเคราะห์ตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล

ในขั้นตอนนี้เครื่องมือจะอ่านไฟล์ดับเบิลยูเอสดีแอล และค้นหาแท็กเอกซ์เอ็มแอล <input> ที่เป็นชุดตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล แสดงดังรูปที่ 3-15

```

21 <!-- ~~~~~
22 MESSAGE TYPE DEFINITION - Definition of the message types used as
23 part of the port type definitions
24 ~~~~~ -->
25 <wsdl:message name="ElectronicDeptRequestMessage">
26   <wsdl:part name="payload" element="client:process"/>
27 </wsdl:message>
28 <wsdl:message name="ElectronicDeptResponseMessage">
29   <wsdl:part name="payload" element="client:processResponse"/>
30 </wsdl:message>
31
32 <!-- ~~~~~
33 PORT TYPE DEFINITION - A port type groups a set of operations into
34 a logical service unit.
35 ~~~~~ -->
36
37 <!-- portType implemented by the ElectronicDept BPEL process -->
38 <wsdl:portType name="ElectronicDept">
39   <wsdl:operation name="process">
40     <wsdl:input message="client:ElectronicDeptRequestMessage" />
41     <wsdl:output message="client:ElectronicDeptResponseMessage"/>
42   </wsdl:operation>
43 </wsdl:portType>

```

รูปที่ 3-15 ชุดตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล

จากการอ่านไฟล์ดับเบิลยูเอสดีแอล ดังรูปที่ 3-15 พบว่าบรรทัดที่ 40 ปรากฏชุดตัวแปร นำเข้า ElectronicDeptRequestMessage ส่วนบรรทัดที่ 25 ปรากฏรายละเอียดของชุดตัวแปร นำเข้า ElectronicDeptRequestMessage อยู่ในอิลิเมนต์ process

จากนั้นเครื่องมือจะอ่านไฟล์เอกซ์เอสดี ดังรูปที่ 3-16 เพื่อค้นหาชุดตัวแปรนำเข้า กระบวนการดับเบิลยูเอส-บีเพล โดยค้นหาแท็กเอกซ์เอ็มแอลที่มีอิลิเมนต์ชื่อ process โดยเครื่องมือ พบว่าในบรรทัดที่ 20 ในไฟล์เอกซ์เอสดี มีการปรากฏแท็กเอกซ์เอ็มแอลชื่อ process

```

5 <simpleType name="CreditCardType" id="CreditCardType" final="restriction">
6 <restriction base="string">
7   <enumeration value="VISA" />
8   <enumeration value="MASTER" />
9   <enumeration value="JCB" />
10  <enumeration value="AMEX" />
11  <enumeration value="UNIONPAY" />
12 </restriction>
13 </simpleType>
14 <simpleType name="SourceOfFund" id="SourceOfFund" final="restriction">
15 <restriction base="string">
16   <enumeration value="CASH" />
17   <enumeration value="CREDITCARD" />
18 </restriction>
19 </simpleType>
20 <element name="process">
21 <complexType>
22 <sequence>
23   <element name="Price">
24     <simpleType>
25       <restriction base="decimal">
26         <totalDigits value="8" />
27         <fractionDigits value="2" />
28       </restriction>
29     </simpleType>
30   </element>
31   <element name="SOF" type="kt:SourceOfFund" />
32   <element name="CCNo">
33     <simpleType>
34       <restriction base="integer">
35         <minInclusive value="1000000000000000" />
36         <maxInclusive value="999999999999999" />
37       </restriction>
38     </simpleType>
39   </element>
40   <element name="CCType" type="kt:CreditCardType" />
41 </sequence>
42 </complexType>
43 </element>
44 <element name="processResponse">
45 <complexType>
46 <sequence>
47   <element name="IsAllow" type="boolean" />
48   <element name="NetPrice" type="decimal" default="0.0" />
49   <element name="VoucherValue" type="integer" default="0" />
50 </sequence>
51 </complexType>
52 </element>

```

รูปที่ 3-16 ชุดตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพลบนไฟล์เอกซ์เอสดี

ชุดตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพลและข้อจำกัดของตัวแปร ที่อ่านได้จากไฟล์
ดับเบิลยูเอสดีแอลในรูปแบบที่ 3-15 และไฟล์เอกซ์เอสดีในรูปแบบที่ 3-16 มีรายละเอียดแสดงดังตารางที่ 3-1

ตารางที่ 3-1 ชุดตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพลและข้อจำกัดของตัวแปร

ลำดับ	ตัวแปรนำเข้า	ชนิดข้อมูล	ข้อจำกัดของตัวแปรนำเข้า
1	Price	decimal	ตัวเลขทั้งหมด 8 หลักโดยเป็นจุดทศนิยม 2 หลัก
2	SOF	string	'CASH' หรือ 'CREDITCARD'
3	CCNo	integer	ค่าต่ำสุด = 1000000000000000 ค่าสูงสุด = 9999999999999999
4	CCType	string	'VISA' หรือ 'MASTER' หรือ 'JCB' หรือ 'AMEX' หรือ 'UNIONPAY'

3.4.2 วิเคราะห์ตัวแปรในแต่ละโหนดของทางเดินทดสอบ

ในขั้นตอนนี้เครื่องมือจะวิเคราะห์หาเงื่อนไขและข้อจำกัดของตัวแปรที่อยู่ในแต่ละโหนดของ
ทางเดินทดสอบที่ได้จากขั้นตอนที่ 3.3 โดยจะใช้ทางเดินทดสอบ P1 เป็นตัวอย่างการทำงานขั้นตอนนี้
ซึ่งจากทางเดินทดสอบ P1: 1 – 2 – 3.1 – 3.2 – 3.3 – 3.4.1 – 3.4.2 – 3.4.3 – 3.4.4 – 4 พบว่ามี
เงื่อนไขและข้อจำกัดของตัวแปรในแต่ละโหนดของทางเดินทดสอบ แสดงดังตารางที่ 3-3

ตารางที่ 3-2 เงื่อนไขของตัวแปรในแต่ละโหนดของทางเดินทดสอบ P1

โหนด	ชื่อโหนด	ชนิดของโหนด	ตัวแปร	เงื่อนไขและข้อจำกัด
1	receiveInput	<receive>	-	-
2	Discount5Percent	<assign>	NetPrice	= Price - ((Price * 5)/100)
3.1	IfCreditCard	<if>	SOF	= 'CREDITCARD'
3.2	IfVISA	<if>	CCType	= 'VISA'
3.3	assAllowVisa	<assign>	IsAllow	= true
3.4.1	IfPrice2500Up	<if>	Price	>= 2500
3.4.2	IfPrice5000Up	<if>	Price	>= 5000
3.4.3	IfPrice10000Up	<if>	Price	>= 10000
3.4.4	Voucher700THB	<assign>	VoucherValue	= 700
4	replyOutput	<reply>	-	-

3.5 สร้างกรณีทดสอบ

เครื่องมือจะสร้างกรณีทดสอบในแต่ละทางเดินทดสอบ โดยเริ่มจากการสุ่มค่าของตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล จากนั้นจึงรวบรวมข้อมูลตัวแปรนำเข้ามาสร้างเป็นกรณีทดสอบ โดยเมื่อเครื่องมือทำการสร้างกรณีทดสอบเสร็จสิ้น ผู้ใช้งานสามารถดาวน์โหลดกรณีทดสอบและนำไปทดสอบดับเบิลยูเอส-บีเพลบนบีเพลเอนจิน โดยการสร้างกรณีทดสอบ มีรายละเอียดแต่ละขั้นตอน ดังนี้

3.5.1 สุ่มค่าของตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล

เครื่องมือทำการสุ่มค่าของตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล โดยพิจารณาจากรายการตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล แสดงดังตารางที่ 3.1 และเงื่อนไขของตัวแปรในแต่ละโหนดของทางเดินทดสอบ P1 แสดงดังตารางที่ 3.2 ซึ่งจะพิจารณาเฉพาะตัวแปรที่อยู่บนโหนด <if> เท่านั้น โดยตัวอย่างการสุ่มค่าของตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพลบนทางเดินทดสอบ P1 และเงื่อนไขต่าง ๆ ของตัวแปรนำเข้าที่เครื่องมือสามารถวิเคราะห์ได้ แสดงดังตารางที่ 3-3

ตารางที่ 3-3 ค่าข้อมูลของตัวแปรนำเข้าบนทางเดินทดสอบ P1

ลำดับ	ตัวแปรนำเข้า	ข้อจำกัดและเงื่อนไขของตัวแปรนำเข้า	ค่าข้อมูลที่สุ่มได้
1	Price	ตัวเลขสูงสุด 8 หลักโดยเป็นจุดทศนิยม 2 หลัก	34690.42
		≥ 2500	
		≥ 5000	
		≥ 10000	
2	SOF	'CASH' หรือ 'CREDITCARD'	CREDITCARD
		= 'CREDITCARD'	
3	CCNo	ค่าต่ำสุด = 1000000000000000	5059232372440152
		ค่าสูงสุด = 9999999999999999	
4	CCType	'VISA' หรือ 'MASTER' หรือ 'JCB' หรือ 'AMEX' หรือ 'UNIONPAY'	VISA
		= 'VISA'	

3.5.2 สร้างกรณีทดสอบ

เครื่องมือจะใช้ข้อมูลตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพลที่ได้จากขั้นตอนที่ 3.5.1 มาทำการสร้างกรณีทดสอบของแต่ละทางเดินทดสอบ โดยเครื่องมือจะสร้างกรณีทดสอบในรูปแบบไฟล์เอกซ์เซลและเอกซ์เอ็มแอลสำหรับผู้ทดสอบนำเข้ากรณีทดสอบเพื่อทดสอบดับเบิลยูเอส-บีเพลบนบีเพลเอนจิน ตัวอย่างกรณีทดสอบของทางเดินทดสอบ P1 แสดงดังตารางที่ 3-4 และกรณีทดสอบในรูปแบบไฟล์เอกซ์เอ็มแอลแสดงดังรูปที่ 3-17

ตารางที่ 3-4 กรณีทดสอบของทางเดินทดสอบ P1

โครงการ	ElectronicDept	
รหัสกรณีทดสอบ	TC01	
ทางเดินทดสอบ	1 – 2 – 3.1 – 3.2 – 3.3 – 3.4.1 – 3.4.2 – 3.4.3 – 3.4.4 – 4	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Price	decimal	34690.42
SOF	string	CREDITCARD
CCNo	String	5059232372440152
CreditCardType	string	VISA

```

1 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
2   <soap:Body>
3     <ns1:process xmlns:ns1="http://xmlns.oracle.com/Applica
4       <ns1:Price>34690.42</ns1:Price>
5       <ns1:SOF>CREDITCARD</ns1:SOF>
6       <ns1:CCNo>5059232372440152</ns1:CCNo>
7       <ns1:CCType>VISA</ns1:CCType>
8     </ns1:process>
9   </soap:Body>
10 </soap:Envelope>

```

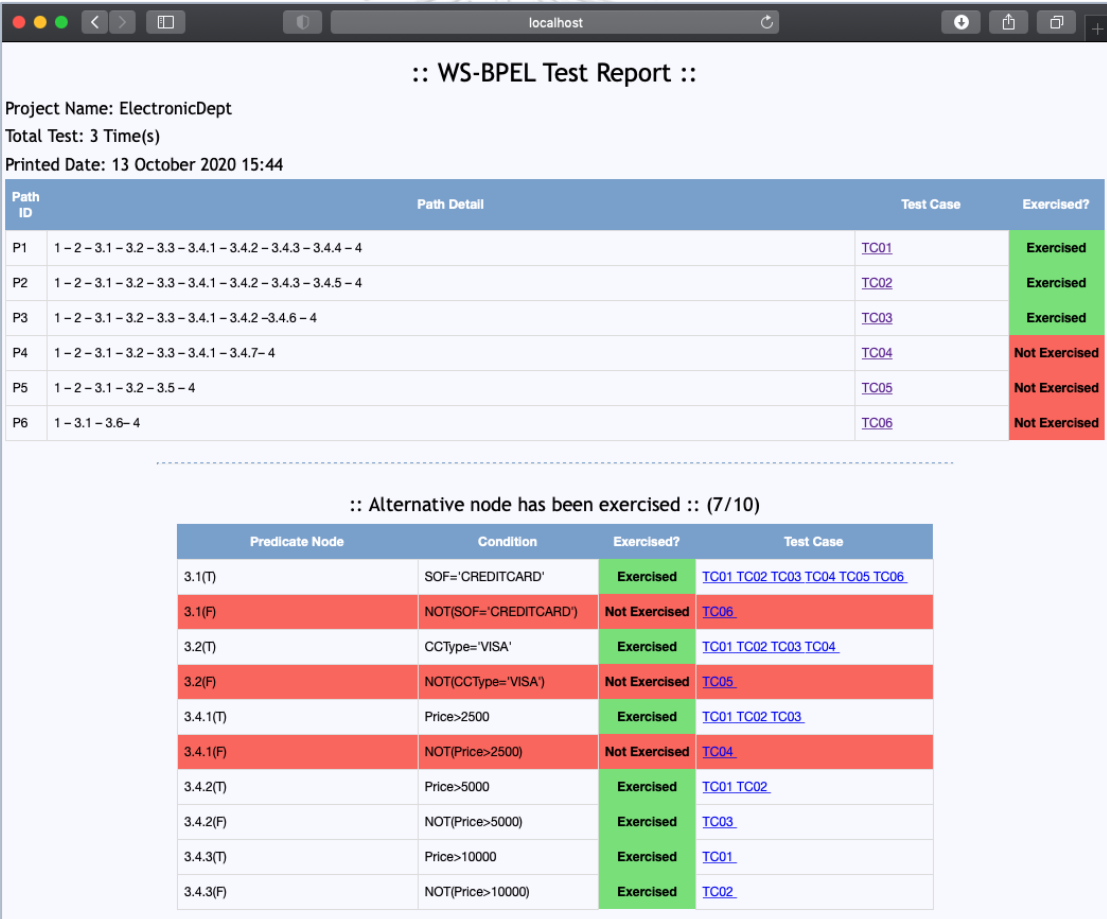
รูปที่ 3-17 ตัวแปรนำเข้าของกรณีทดสอบ TC01 ในรูปแบบเอกซ์เอ็มแอล

3.6 สร้างรายงานการทดสอบดับเบิลยูเอส-บีเพล

เมื่อผู้ทดสอบนำรหัสต้นทางที่แทรกชุดคำสั่งแล้วจากขั้นตอนที่ 3.2 ไปติดตั้งบนบีเพลเอนจิน จากนั้นนำกรณีทดสอบที่ได้รับจากขั้นตอนที่ 3.5 ไปทำการทดสอบดับเบิลยูเอส-บีเพลบนบีเพลเอนจิน

บีเพลเอนจินจะส่งค่าข้อมูลการทดสอบมาให้เครื่องมือ โดยผ่านเว็บเซอร์วิสตามที่เครื่องมือได้แทรกชุดคำสั่งไว้ โดยผู้ทดสอบสามารถเรียกดูรายงานการทดสอบดับเบิลยูเอส-บีเพลได้ ตัวอย่างรายงานการทดสอบดับเบิลยูเอส-บีเพล แสดงดังรูปที่ 3-18 โดยในรายงานการทดสอบดับเบิลยูเอส-บีเพล ประกอบไปด้วยรายละเอียด ดังต่อไปนี้

- 1) ตารางแสดงความครอบคลุมทางเดินทดสอบ โดยแสดงกรณีทดสอบที่ครอบคลุมในแต่ละทางเดินทดสอบ พร้อมทั้งแสดงรายการทางเดินทดสอบที่ถูกทดสอบและยังไม่ถูกทดสอบ
- 2) ตารางแสดงความครอบคลุมโหนดเงื่อนไขของกราฟการเรียกเชิงสถิติ โดยแสดงกรณีทดสอบที่ครอบคลุมในแต่ละโหนดเงื่อนไข พร้อมทั้งแสดงรายการโหนดเงื่อนไขที่ถูกทดสอบและยังไม่ถูกทดสอบ



:: WS-BPEL Test Report ::

Project Name: ElectronicDept
 Total Test: 3 Time(s)
 Printed Date: 13 October 2020 15:44

Path ID	Path Detail	Test Case	Exercised?
P1	1-2-3.1-3.2-3.3-3.4.1-3.4.2-3.4.3-3.4.4-4	TC01	Exercised
P2	1-2-3.1-3.2-3.3-3.4.1-3.4.2-3.4.3-3.4.5-4	TC02	Exercised
P3	1-2-3.1-3.2-3.3-3.4.1-3.4.2-3.4.6-4	TC03	Exercised
P4	1-2-3.1-3.2-3.3-3.4.1-3.4.7-4	TC04	Not Exercised
P5	1-2-3.1-3.2-3.5-4	TC05	Not Exercised
P6	1-3.1-3.6-4	TC06	Not Exercised

:: Alternative node has been exercised :: (7/10)

Predicate Node	Condition	Exercised?	Test Case
3.1(T)	SOF='CREDITCARD'	Exercised	TC01 TC02 TC03 TC04 TC05 TC06
3.1(F)	NOT(SOF='CREDITCARD')	Not Exercised	TC06
3.2(T)	CCType='VISA'	Exercised	TC01 TC02 TC03 TC04
3.2(F)	NOT(CCType='VISA')	Not Exercised	TC05
3.4.1(T)	Price>2500	Exercised	TC01 TC02 TC03
3.4.1(F)	NOT(Price>2500)	Not Exercised	TC04
3.4.2(T)	Price>5000	Exercised	TC01 TC02
3.4.2(F)	NOT(Price>5000)	Exercised	TC03
3.4.3(T)	Price>10000	Exercised	TC01
3.4.3(F)	NOT(Price>10000)	Exercised	TC02

รูปที่ 3-18 รายงานการทดสอบดับเบิลยูเอส-บีเพล

บทที่ 4

การออกแบบและพัฒนาเครื่องมือ

ผู้วิจัยได้นำวิธีการที่นำเสนอในบทที่ 3 มาพัฒนาเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ สำหรับบทนี้เป็นการนำเสนอการออกแบบและพัฒนาเครื่องมือโดยใช้การออกแบบโปรแกรมเชิงวัตถุด้วยภาษายูเอ็มแอล (UML) มาอธิบายฟังก์ชันการทำงานของเครื่องมือ พร้อมทั้งนำเสนอสภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ และส่วนต่อประสานผู้ใช้ของเครื่องมือ

4.1 การออกแบบเครื่องมือ

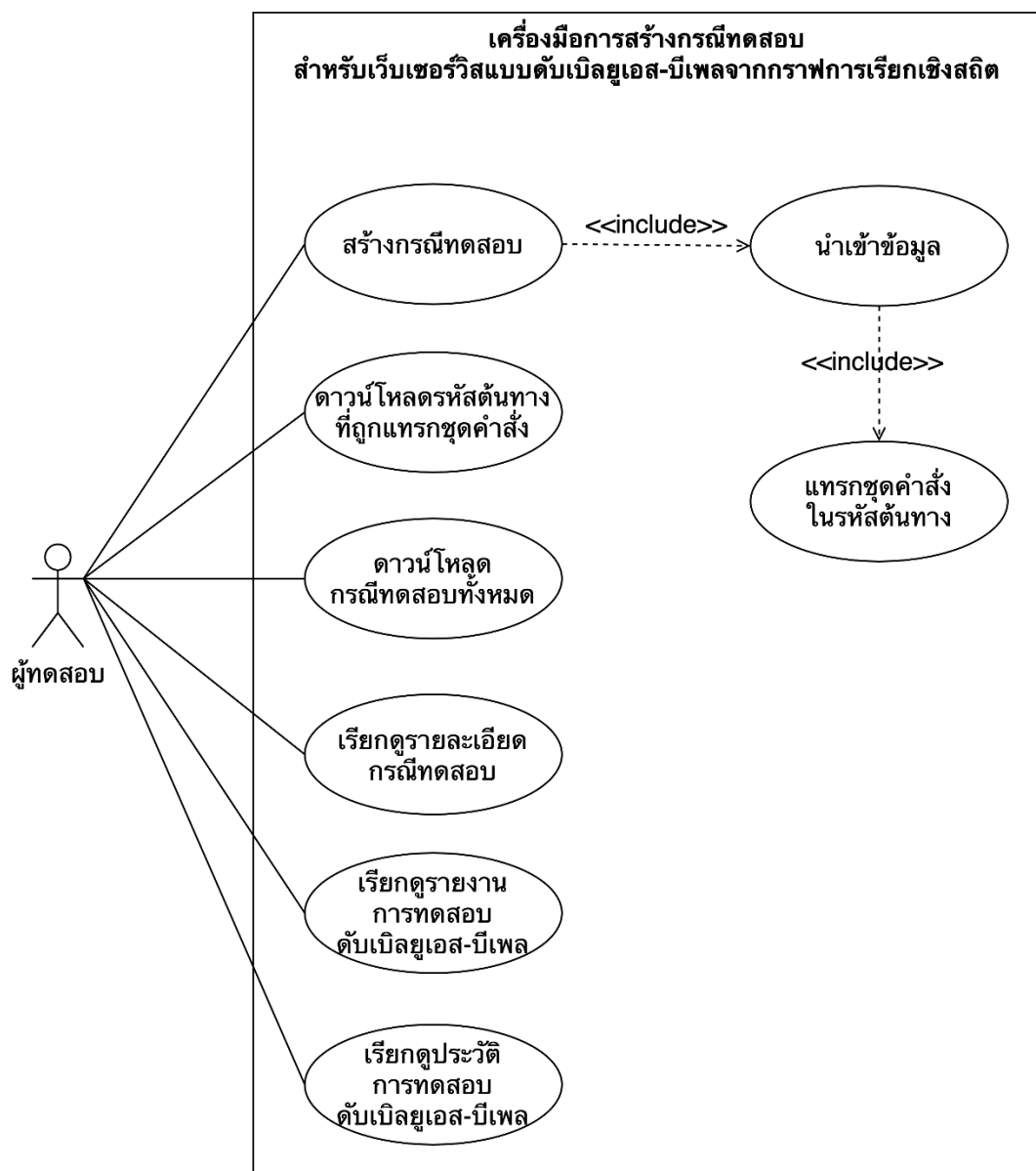
การออกแบบเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ สามารถแสดงได้ด้วยแผนภาพยูสเคส (Use Case Diagram) แผนภาพกิจกรรม (Activity Diagram) แผนภาพคลาส (Class Diagram) แผนภาพลำดับ (Sequence Diagram) และโครงสร้างฐานข้อมูลด้วยแผนภาพอีอาร์

4.1.1 แผนภาพยูสเคส

แผนภาพยูสเคสดังรูปที่ 4-1 เป็นแผนภาพที่แสดงถึงฟังก์ชันการทำงานของเครื่องมือ ซึ่งประกอบไปด้วย 8 ยูสเคส ดังนี้ โดยรายละเอียดของแต่ละยูสเคสสามารถดูเพิ่มเติมได้ที่ภาคผนวก ก

- 1) ยูสเคสนำเข้าข้อมูลตั้งต้น เป็นยูสเคสที่ผู้ทดสอบนำข้อมูลตั้งต้นเข้าสู่เครื่องมือ โดยข้อมูลตั้งต้นที่ผู้ทดสอบจะต้องนำเข้าได้แก่ ชื่อโครงการดับเบิลยูเอส-บีเพล ไฟล์คอมโพสิต ไฟล์บีเพลตั้งต้น ไฟล์ดับเบิลยูเอสดีแอล ไฟล์เอกซ์เอสดี และชุดไฟล์บีเพลย่อยที่ถูกเรียกใช้งาน
- 2) ยูสเคสแทรกชุดคำสั่งในรหัสต้นทาง เป็นยูสเคสที่เครื่องมือแทรกชุดคำสั่งในรหัสต้นทาง
- 3) ยูสเคสสร้างกรณีทดสอบ เป็นยูสเคสที่เครื่องมือทำการสร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติ สุ่มข้อมูลนำเข้าตามข้อจำกัดของตัวแปร และสร้างกรณีทดสอบแต่ละทางเดินทดสอบ
- 4) ยูสเคสดาวนโหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่ง เป็นยูสเคสที่ผู้ทดสอบดาวนโหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่ง เพื่อนำไปติดตั้งบนบีเพลเอนจิน
- 5) ยูสเคสดาวนโหลดกรณีทดสอบทั้งหมด เป็นยูสเคสที่ผู้ทดสอบดาวนโหลดไฟล์กรณีทดสอบของโครงการดับเบิลยูเอส-บีเพล เพื่อนำไปทดสอบร่วมกับรหัสต้นทางที่ถูกแทรกชุดคำสั่งที่ติดตั้งอยู่บนบีเพลเอนจิน

- 6) ยูสเคสรายเรียกดูละเอียดกรณีทดสอบ เป็นยูสเคสที่ผู้ทดสอบเรียกดูรายละเอียดกรณีทดสอบของโครงการดับเบิลยูเอส-บีเพลบนทางเดินทดสอบที่สนใจ
- 7) ยูสเคสเรียกดูรายงานการทดสอบ เป็นยูสเคสที่ผู้ทดสอบเรียกดูรายงานการทดสอบดับเบิลยูเอส-บีเพลของโครงการ
- 8) ยูสเคสเรียกดูประวัติการทดสอบดับเบิลยูเอส-บีเพล เป็นยูสเคสที่ผู้ทดสอบเรียกดูประวัติการทดสอบโครงการดับเบิลยูเอส-บีเพลย้อนหลัง

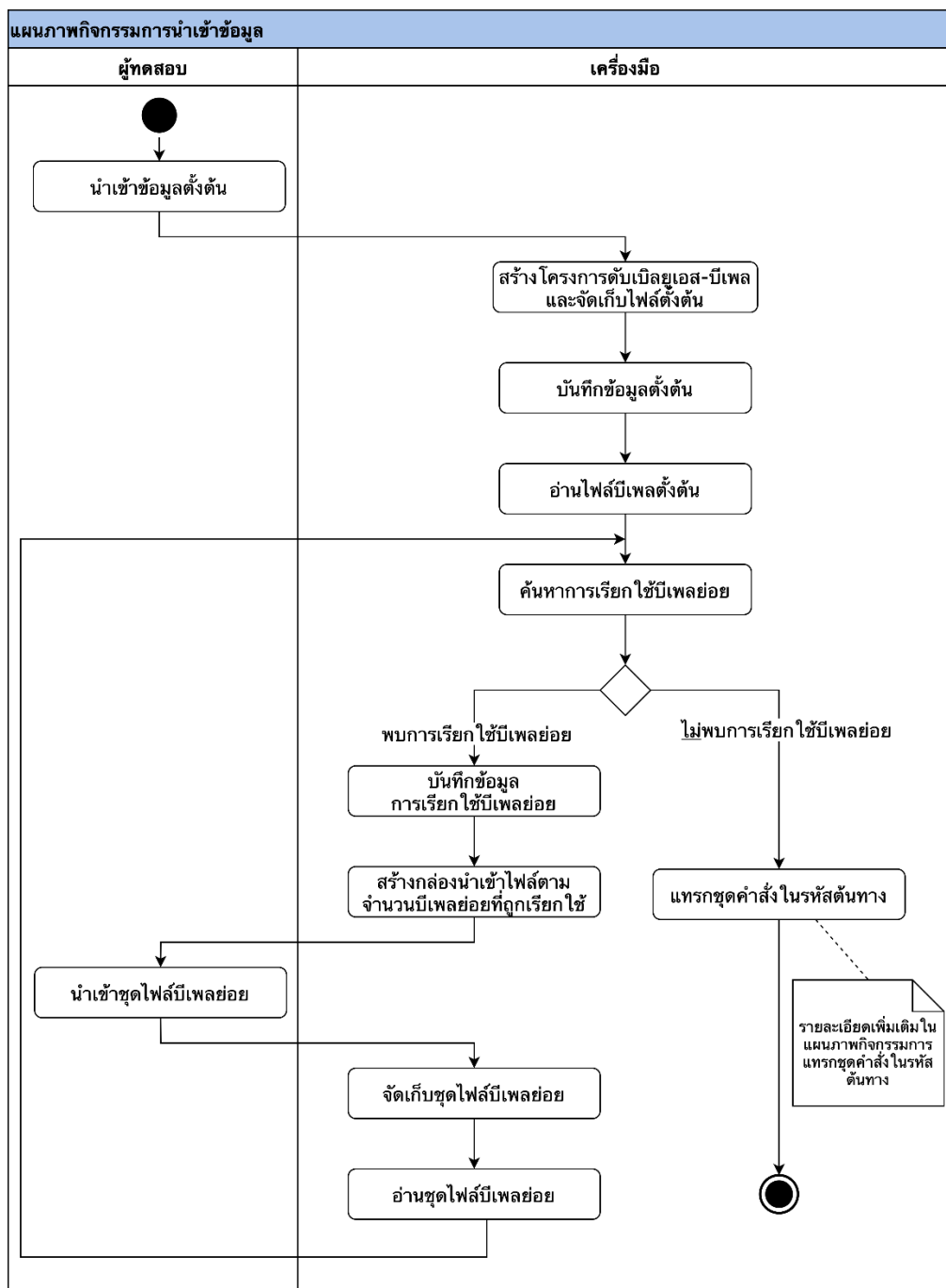


รูปที่ 4-1 แผนภาพยูสเคสของเครื่องมือ

4.1.2 แผนภาพกิจกรรม

แผนภาพกิจกรรมเป็นแผนภาพที่ใช้แสดงขั้นตอนการทำงานของเครื่องมือเพื่ออธิบายกิจกรรมที่เกิดขึ้น ซึ่งเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ สามารถอธิบายได้ด้วยแผนภาพกิจกรรม ดังต่อไปนี้

1) แผนภาพกิจกรรมการนำเข้าข้อมูล



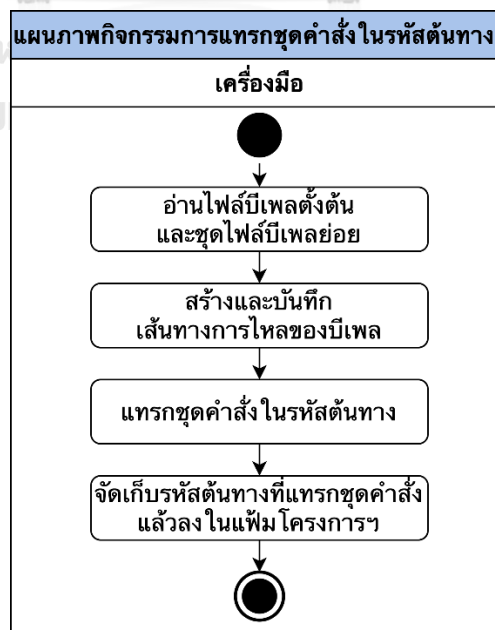
รูปที่ 4-2 แผนภาพกิจกรรมการนำเข้าข้อมูล

แผนภาพกิจกรรมการนำเข้าข้อมูล แสดงดังรูปที่ 4-2 โดยเริ่มจากผู้ทดสอบนำเข้าข้อมูล ตั้งต้น ซึ่งประกอบไปด้วย ชื่อโครงการดับเบิลยูเอส-บีเพล ไฟล์คอมพิวเตอร์ของดับเบิลยูเอส-บีเพล ไฟล์บีเพลตั้งต้น ไฟล์ดับเบิลยูเอสดีแอล และไฟล์เอกซ์เอสดี จากนั้นเครื่องมือจะสร้างแฟ้มโครงการดับเบิลยูเอส-บีเพลลงบนเว็บเซิร์ฟเวอร์ และจัดเก็บไฟล์ข้อมูลตั้งต้นลงในแฟ้มโครงการดับเบิลยูเอส-บีเพลที่สร้างขึ้น พร้อมทั้งบันทึกข้อมูลตั้งต้นต่าง ๆ ลงในฐานข้อมูล

หลังจากที่เครื่องมือจัดเก็บและบันทึกข้อมูลตั้งต้นเรียบร้อยแล้ว เครื่องมือจะค้นหาการเรียกใช้บีเพลย่อยโดยการอ่านไฟล์บีเพลตั้งต้น หากพบว่าในไฟล์บีเพลตั้งต้นมีการเรียกใช้บีเพลย่อย เครื่องมือจะบันทึกการเรียกใช้บีเพลย่อยลงในฐานข้อมูล และสร้างแถบอัปโหลดไฟล์บนหน้าจอตามจำนวนบีเพลย่อยที่ถูกเรียกใช้ จากนั้นผู้ทดสอบจะต้องนำเข้าชุดไฟล์บีเพลย่อยให้ถูกต้องและครบถ้วนตามที่เครื่องมือแสดงบนหน้าจอ เมื่อผู้ทดสอบนำเข้าชุดไฟล์บีเพลย่อยเรียบร้อยแล้ว เครื่องมือจะจัดเก็บชุดไฟล์บีเพลย่อยลงในแฟ้มโครงการดับเบิลยูเอส-บีเพล จากนั้นเครื่องมือค้นหาการเรียกใช้บีเพลย่อยโดยการอ่านชุดไฟล์บีเพลย่อยที่ผู้ทดสอบนำเข้า

หลังจากที่เครื่องมือค้นหาการเรียกใช้บีเพลย่อย และพบว่ายังมีไฟล์บีเพลย่อยที่ถูกเรียกใช้อยู่ เครื่องมือจะวนลูปบันทึกข้อมูลการเรียกใช้บีเพลย่อย และสร้างแถบอัปโหลดไฟล์บนหน้าจอไปเรื่อย ๆ จนกว่าจะไม่พบการเรียกใช้บีเพลย่อยเพิ่มเติม เครื่องมือจึงยุติการวนลูป

2) แผนภาพกิจกรรมการแทรกชุดคำสั่งในรหัสต้นทาง

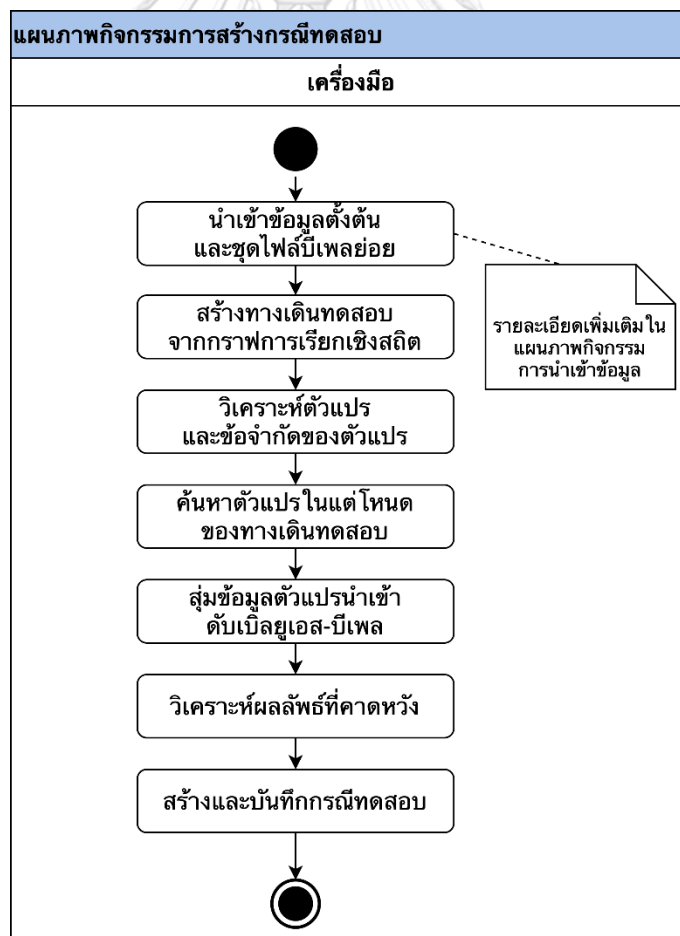


รูปที่ 4-3 แผนภาพกิจกรรมการแทรกชุดคำสั่งในรหัสต้นทาง

แผนภาพกิจกรรมการแทรกชุดคำสั่งในรหัสต้นทางแสดงดังรูปที่ 4-3 โดยเริ่มจากการที่เครื่องมืออ่านไฟล์บีเพลดั้งเดิมและชุดไฟล์บีเพลด้อยที่จัดเก็บในแฟ้มโครงการดับเบิลยูเอส-บีเพล ซึ่งในขณะที่เครื่องมืออ่านข้อมูลแต่ละแท็กของไฟล์บีเพล เครื่องมือจะสร้างและบันทึกเส้นทางการไหลของบีเพลลงในฐานข้อมูล และแทรกชุดคำสั่งในรหัสต้นทาง จากนั้นจะจัดเก็บรหัสต้นทางที่แทรกชุดคำสั่งแล้วลงในแฟ้มโครงการดับเบิลยูเอส-บีเพล

3) แผนภาพกิจกรรมการสร้างกรณีทดสอบ

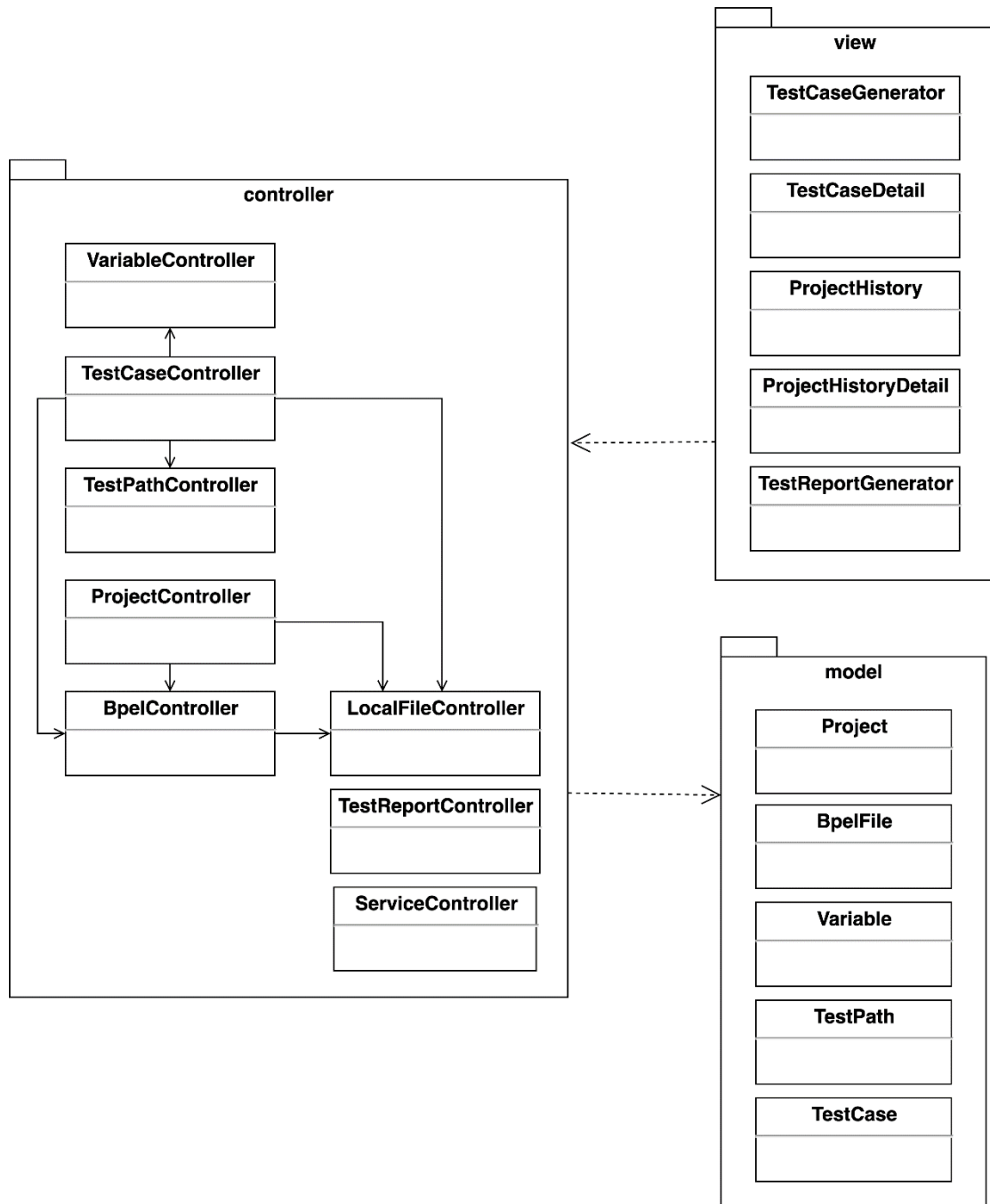
แผนภาพกิจกรรมการสร้างกรณีทดสอบ แสดงดังรูปที่ 4-4 โดยเริ่มจากการที่เครื่องมือจะสร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติที่วิเคราะห์มาจากการไหลของบีเพล และทำการบันทึกทางเดินทดสอบลงฐานข้อมูล จากนั้นเครื่องมือจะวิเคราะห์ตัวแปรและข้อจำกัดของตัวแปร ค้นหาตัวแปรในแต่ละโหนดของทางเดินทดสอบ สุ่มค่าข้อมูลนำเข้า เพื่อสร้างเป็นกรณีทดสอบของแต่ละทางเดินทดสอบ พร้อมทั้งบันทึกกรณีทดสอบลงในฐานข้อมูล



รูปที่ 4-4 แผนภาพกิจกรรมการทดสอบดับเบิลยูเอส-บีเพล

4.1.3 แผนภาพคลาส

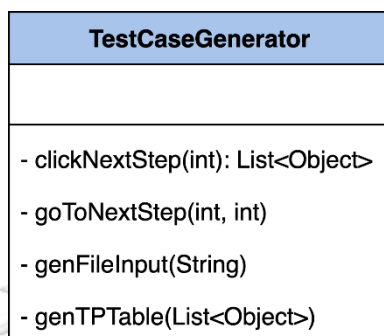
แผนภาพคลาสเป็นแผนภาพที่แสดงรายละเอียดของคลาสและความสัมพันธ์ระหว่างคลาส โดยแผนภาพคลาสของเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ แสดงดังรูปที่ 4-5 โดยแต่ละคลาสที่จำแนกตามแพ็คเกจมีรายละเอียด ดังนี้



รูปที่ 4-5 แผนภาพแพ็คเกจของเครื่องมือ

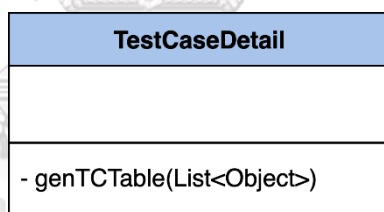
1) แฟ้มเอกสาร form เป็นแฟ้มเอกสารที่ประกอบไปด้วยคลาสที่ทำหน้าที่ติดต่อกับผู้ทดสอบ และแสดงผล ส่วนต่อประสานผู้ใช้ โดยในแฟ้มเอกสาร form ประกอบไปด้วยคลาสต่าง ๆ ดังนี้

1.1) คลาส TestCaseGenerator ทำหน้าที่แสดงข้อมูลในแต่ละขั้นตอนการสร้างกรณีทดสอบ และสร้างแถบอัปโหลดไฟล์ตามจำนวนการเรียกใช้บีเฟลย์ย่อย โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-6



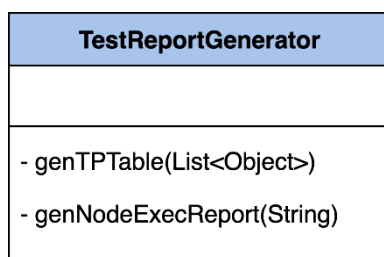
รูปที่ 4-6 คลาส TestCaseGenerator

1.2) คลาส TestCaseDetail ทำหน้าที่แสดงรายละเอียดของกรณีทดสอบ โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-7



รูปที่ 4-7 คลาส TestCaseDetail

1.3) คลาส TestReportGenerator ทำหน้าที่แสดงรายงานการทดสอบดับเบิลยูเอส-บีเฟล ซึ่งในรายงานการทดสอบมีการแสดงตารางทางเดินทดสอบ และตารางหนดเงื่อนไขที่ยังไม่ผ่านการทดสอบ โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-8



รูปที่ 4-8 คลาส TestReportGenerator

- 1.4) คลาส ProjectHistory ทำหน้าที่แสดงประวัติโครงการดับเบิลยูเอส-บีเพลที่เคยถูกสร้าง
กรณีทดสอบ โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-9

ProjectHistory
- genHistoryTable(List<Object>)

รูปที่ 4-9 คลาส ProjectHistory

- 1.5) คลาส ProjectHistoryDetail ทำหน้าที่แสดงรายละเอียดโครงการดับเบิลยูเอส-บีเพล
โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-10

ProjectHistoryDetail
- genHisDtlsTable(List<Object>)

รูปที่ 4-10 คลาส ProjectHistoryDetail

- 2) แพ็กเกจ controller เป็นแพ็กเกจที่ประกอบไปด้วยคลาสที่ทำหน้าที่ประมวลผล จัดการ และ
ตรวจสอบข้อมูลที่น่าเข้ามาจากส่วนต่อประสานผู้ใช้ โดยในแพ็กเกจ controller ประกอบไปด้วย
คลาสต่าง ๆ ดังนี้

- 2.1) คลาส ProjectController ทำหน้าที่สร้างโครงการดับเบิลยูเอส-บีเพล จะถูกเรียกใช้งาน
เมื่อผู้ทดสอบสร้างโครงการและนำเข้าข้อมูลตั้งต้น โดยรายละเอียดของคลาสแสดงดังรูป
ที่ 4-11

ProjectController
- intProjectId: integer
- objProject: Object
+ createBpelProj(string, List<String>): JSON

รูปที่ 4-11 คลาส ProjectController

- 2.2) คลาส BpelController ทำหน้าที่ค้นหาการเรียกใช้บีเพลย่อย และแทรกชุดคำสั่งในรหัส
ต้นทาง โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-12

BpelController
<ul style="list-style-type: none"> - searchSbpelCalled(String): List<String> - addArrFileList(List<String>) - getNewBpelList(List<Object>, List<Object>): List<Object> - getArrFileList(): List<String> - getInstrumentList(): String - instrumentTag(String) - instrumentSeqTag(int, String) + createSbpelFile(int, List<Object>): JSON + readBpel(String)

รูปที่ 4-12 คลาส BpelController

- 2.3) คลาส VariableController ทำหน้าที่อ่านไฟล์เอกซ์เอสดีเพื่อค้นหาข้อมูลตัวแปรและข้อจำกัด โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-13

VariableController
<ul style="list-style-type: none"> + loadXsd(String, List<Object>)

รูปที่ 4-13 คลาส VariableController

- 2.4) คลาส TestPathController ทำหน้าที่ค้นหาทางเดินทดสอบจากกราฟการเรียกเชิงสถิติที่วิเคราะห์ความสัมพันธ์จากเส้นทางการไหลของบีเพลในแต่ละไฟล์ โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-14

TestPathController
<ul style="list-style-type: none"> - generateTP (List<Object>): List<Object> + createTP(int): List<Object>

รูปที่ 4-14 คลาส TestPathController

- 2.5) คลาส TestCaseController ทำหน้าที่สร้างกรณีทดสอบ โดยมีการวิเคราะห์ข้อมูลตัวแปรและข้อจำกัดที่อยู่ในแต่ละทางเดินทดสอบ สุ่มข้อมูลนำเข้า และสร้างกรณีทดสอบ โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-15

TestCaseController
- generateTC(List<Object>): List<Object> + genTC(int): List<Object>

รูปที่ 4-15 คลาส TestCaseController

- 2.6) คลาส TestReportController ทำหน้าที่สร้างรายงานการทดสอบดับเบิลยูเอส-บีเพลของโครงการ โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-16

TestReportController
+ generateTR(int): Object

รูปที่ 4-16 คลาส TestReportController

- 2.7) คลาส ServiceController ทำหน้าที่รับข้อมูลการทดสอบดับเบิลยูเอส-บีเพลที่ผู้ทดสอบกระทำการทดสอบผ่านบีเพลเอนจิน โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-17

ServiceController
+ strIpAddress: String
+ setTpExecuted(int,String)

รูปที่ 4-17 คลาส ServiceController

- 2.8) คลาส LocalFileController ทำหน้าที่สร้างแฟ้มข้อมูลโครงการดับเบิลยูเอส-บีเพลบนเว็บเซิร์ฟเวอร์ จัดเก็บไฟล์นำเข้าลงในแฟ้มโครงการ โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-18

LocalFileController
+ createProjFolder(String): String + uplFile(String, List<String>) + writeInstrumentFile(String, String, String)

รูปที่ 4-18 คลาส LocalFileController

3) แพ็กเกจ model เป็นแพ็กเกจที่ประกอบไปด้วยคลาสที่ทำหน้าที่เข้าถึงฐานข้อมูลเพื่ออ่านค่าข้อมูล บันทึกข้อมูล และแก้ไขข้อมูลที่อยู่ในฐานข้อมูล โดยในแพ็กเกจ model ประกอบไปด้วยคลาสต่าง ๆ ดังนี้

3.1) คลาส Project ทำหน้าที่บันทึกและค้นหาโครงการดับเบิลยูเอส-บีเพล โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-19

Project
- id: int - name: String - path: String - xsdFname: String - wsdlFname: String - compositeFname: String - createDate: datetime
+ getProjectById(int): Object + addProject(String, String, String, String): int

รูปที่ 4-19 คลาส Project

3.2) คลาส Bpel ทำหน้าที่บันทึก ค้นหา และแก้ไขข้อมูลที่เกี่ยวข้องกับบีเพลในฐานข้อมูล ข้อมูลที่เกี่ยวข้องกับบีเพล ได้แก่ ไฟล์บีเพลตั้งต้น ชุดไฟล์บีเพลย่อย และโหนดกิจกรรมของบีเพล โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-20

Bpel
- id: int - projectId: int - fname: String - parentId: int
+ getBpelFileListByProjId(int): List<Object> + getBpelUnreadListByProjId(int): List<Object> + getBpelNodeListByProjId(int): List<Object> + addBpelNode(int, int): int + addSbpelFileList(int, int, List<String>) + updateBpelReadById(int)

รูปที่ 4-20 คลาส Bpel

- 3.3) คลาส Variable ทำหน้าที่บันทึกและค้นหาข้อมูลที่เกี่ยวข้องกับตัวแปรและข้อจำกัดของตัวแปร โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-21

Variable
- id: int - fileId: int - varName: String - varConstraint: String
+ getNodeVarByNodeList(Object): List<Object> + addVarList(int, List<Object>) + getVarByName(int, String): Object + addNodeVarList(int, int, Object)

รูปที่ 4-21 คลาส Variable

- 3.4) คลาส TestPath ทำหน้าที่ค้นหาทางเดินทดสอบจาก บันทึกข้อมูลทางเดินทดสอบ และบันทึกข้อมูลทางเดินทดสอบที่ถูกทดสอบผ่านปีเพลเอนจิน โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-22

TestPath
- id: int - projectId: int - tpName: String - tpNodeName: String
+ getTestPathByName(int, String): int + addTestPathLog(int, List<String>) + addTPLList(List<Object>)

รูปที่ 4-22 คลาส TestPath

- 3.5) คลาส TestCase ทำหน้าที่บันทึกกรณีทดสอบและรายละเอียดของกรณีทดสอบ โดยรายละเอียดของคลาสแสดงดังรูปที่ 4-23

TestCase
- id: int - tpld: int - tcName: String
+ addTC(int, String): int + addTCDetailList(List<Object>)

รูปที่ 4-23 คลาส TestCase

4.1.4 แผนภาพลำดับ

แผนภาพลำดับเป็นแผนภาพที่อธิบายขั้นตอนการทำงานของคลาสต่าง ๆ อย่างเป็นลำดับขั้น โดยการทำงานของเครื่องมือประกอบไปด้วยการสร้างกรณีทดสอบและการทดสอบดับเบิลยูเอส-พีเพิล ซึ่งในแต่ละขั้นตอนการทำงานมีรายละเอียด ดังนี้

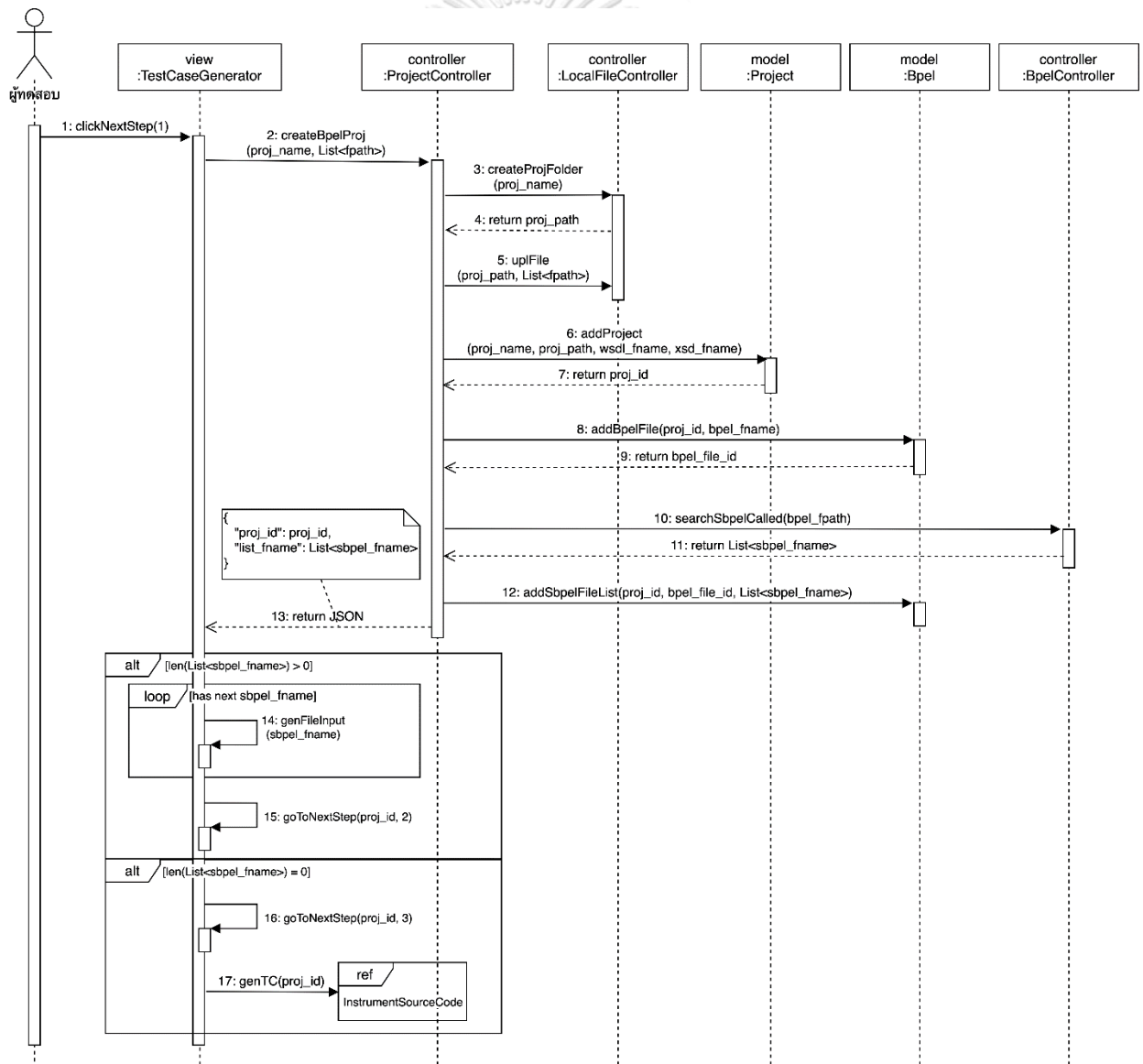
1) การสร้างกรณีทดสอบ

ในการสร้างกรณีทดสอบนั้น ผู้วิจัยได้ออกแบบแผนภาพลำดับเป็น 5 ส่วนตามขั้นตอนการทำงาน ได้แก่ แผนภาพลำดับการนำเข้าข้อมูลตั้งต้น แผนภาพลำดับการนำเข้าชุดไฟล์บีบอัด

แผนภาพลำดับการแทรกชุดคำสั่งในรหัสต้นทาง แผนภาพลำดับการสร้างกรณีทดสอบ และแผนภาพลำดับการจัดเก็บผลการทดสอบด้วยเบิลยูเอส-บีเพล โดยแต่ละแผนภาพลำดับมีรายละเอียด ดังนี้

1.1) แผนภาพลำดับการนำเข้าข้อมูลตั้งต้น

แผนภาพลำดับการนำเข้าข้อมูลตั้งต้น ดังรูปที่ 4-24 เป็นแผนภาพลำดับที่แสดงถึงการนำเข้าข้อมูลตั้งต้นของผู้ทดสอบ โดยเมื่อผู้ทดสอบกดปุ่ม Next ที่อยู่บนหน้าจอแสดงผล จะเป็นการเรียกใช้ฟังก์ชัน clickNextStep(1) ของคลาส TestCaseGenerator ต่อมาเครื่องมือจะสร้างโครงการด้วยเบิลยูเอส-บีเพลและจัดเก็บข้อมูลตั้งต้นโดยการเรียกใช้ฟังก์ชัน createBpelProj ของคลาส ProjectController ที่เป็นฟังก์ชันหลักในการนำเข้าข้อมูลตั้งต้น



รูปที่ 4-24 แผนภาพลำดับการนำเข้าข้อมูลตั้งต้น

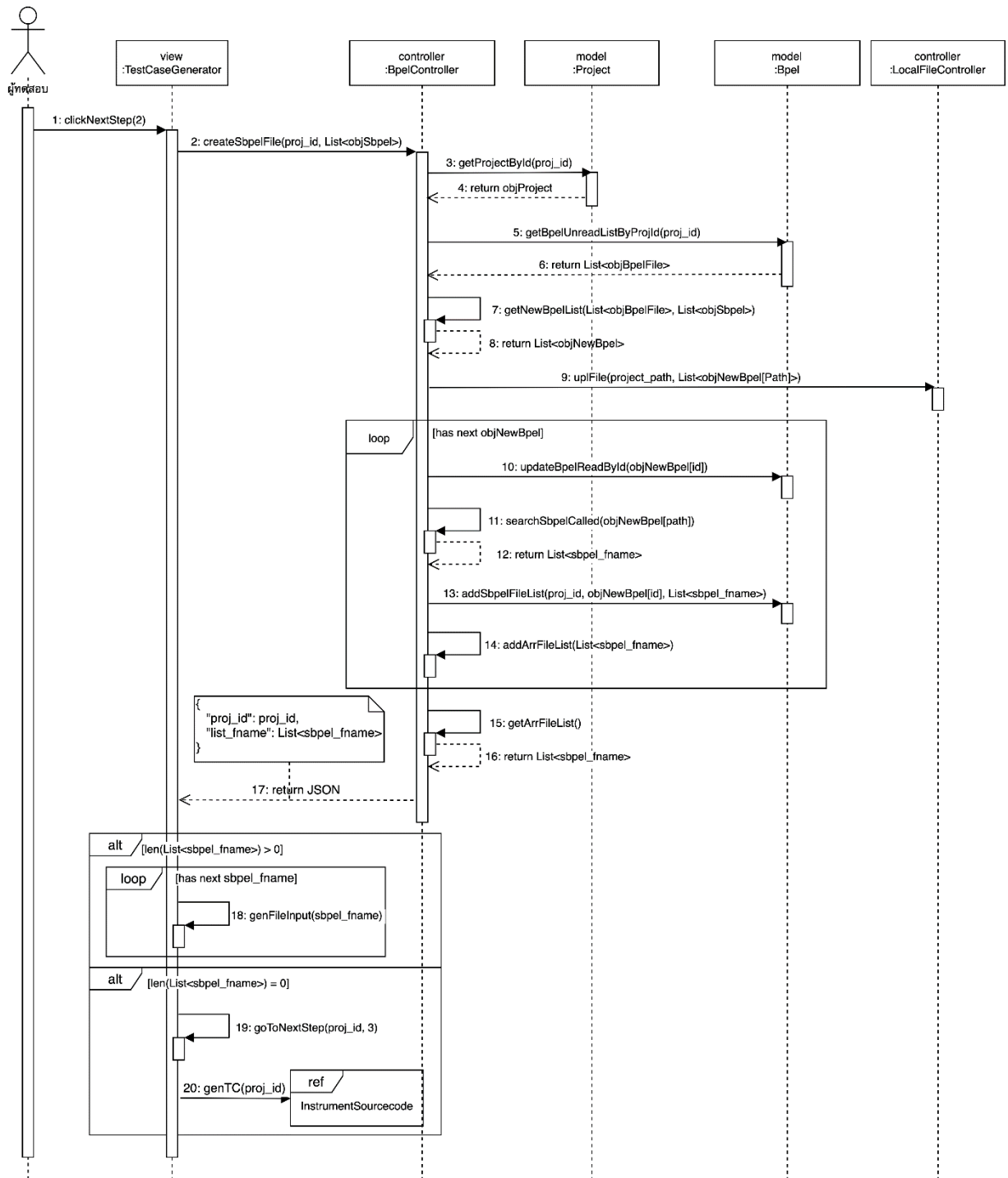
ภายในฟังก์ชัน createBpelProj มีการเรียกใช้ฟังก์ชัน createProjFolder และ uplFile ของคลาส LocalFileController ตามลำดับ เพื่อสร้างแฟ้มโครงการดับเบิลยูเอส-บีเพลบนเว็บเซิร์ฟเวอร์ และอัปโหลดไฟล์ตั้งต้นลงในแฟ้มโครงการ จากนั้นเครื่องมือจะเรียกใช้ฟังก์ชัน addProject ของคลาส Project เพื่อบันทึกข้อมูลโครงการดับเบิลยูเอส-บีเพลลงในฐานข้อมูล และเรียกใช้ฟังก์ชัน addBpelFile ของคลาส BpelFile เพื่อบันทึกข้อมูลรายละเอียดไฟล์บีเพลลงในฐานข้อมูล เมื่อข้อมูลนำเข้าต่าง ๆ ถูกจัดเก็บและบันทึกข้อมูลเรียบร้อยแล้ว เครื่องมือจะเรียกใช้ฟังก์ชัน searchSBpelCalled ของคลาส BpelController เพื่อค้นหาการเรียกใช้บีเพลย่อยจากการอ่านไฟล์บีเพลตั้งต้น และเมื่อเครื่องมือพบข้อมูลการเรียกใช้บีเพลย่อย เครื่องมือจะเรียกใช้ฟังก์ชัน addSbpelFileList ของคลาส Bpel เพื่อบันทึกข้อมูลการเรียกใช้บีเพลย่อยลงฐานข้อมูล

จากนั้นเครื่องมือจะตรวจสอบชุดไฟล์บีเพลย่อยที่ถูกเรียกใช้งาน จากผลลัพธ์ที่ได้จากการเรียกใช้ฟังก์ชัน createBpelProj โดยมีเงื่อนไข ดังนี้

- หากเครื่องมือตรวจพบที่มีการเรียกใช้บีเพลย่อยมากกว่า 1 ไฟล์ เครื่องมือจะวนลูปเรียกใช้ฟังก์ชัน genFileInput เพื่อสร้างแถบอัปโหลดไฟล์ตามจำนวนชุดไฟล์บีเพลย่อย และเรียกใช้ฟังก์ชัน goToNextStep โดยการส่งค่าขั้นตอนเป็น 2 เพื่อนำผู้ทดสอบไปสู่ขั้นตอนการนำเข้าชุดไฟล์บีเพลย่อย
- หากเครื่องมือไม่พบการเรียกใช้งานบีเพลย่อย เครื่องมือจะเรียกใช้ฟังก์ชัน goToNextStep โดยการส่งค่าขั้นตอนเป็น 3 เพื่อนำผู้ทดสอบไปสู่ขั้นตอนการแทรกชุดคำสั่งในรหัสต้นทาง

1.2) แผนภาพลำดับการนำเข้าชุดไฟล์บีเพลย่อย

แผนภาพลำดับการนำเข้าชุดไฟล์บีเพลย่อย ดังรูปที่ 4-25 โดยเมื่อผู้ทดสอบได้นำเข้าข้อมูลตั้งต้นเรียบร้อยแล้วและเครื่องมือพบที่มีการเรียกใช้บีเพลย่อย ผู้ใช้งานต้องนำเข้าชุดไฟล์บีเพลย่อยให้ถูกต้องและครบถ้วน และเมื่อผู้ทดสอบกดปุ่ม Next ที่อยู่บนหน้าจอแสดงผล จะเป็นการเรียกใช้ฟังก์ชัน clickNextStep(2) ของคลาส TestCaseGenerator ต่อมาเครื่องมือจะทำการจัดเก็บข้อมูลชุดไฟล์บีเพลย่อยที่ผู้ทดสอบนำเข้า โดยการเรียกใช้ฟังก์ชัน createSbpelFile ของคลาส BpelController ที่เป็นฟังก์ชันหลักในการนำเข้าชุดไฟล์บีเพลย่อย



รูปที่ 4-25 แผนภาพลำดับการนำเข้าสู่ชุดไฟล์บีเพลย่อย

ภายในฟังก์ชัน createSbpelFile มีการเรียกใช้ฟังก์ชัน getProjectById จากคลาส Project เพื่อดึงข้อมูลโครงการดับเบิ้ลยูเอส-บีเพลที่กำลังใช้งานอยู่ในปัจจุบัน จากนั้นเรียกใช้ฟังก์ชัน getBpelUnreadListByProjId จากคลาส Bpel เพื่อดึงข้อมูลชุดไฟล์บีเพลย่อยที่ยัง

ไม่ถูกอ่านข้อมูลขึ้นมาเปรียบเทียบกับข้อมูลชุดไฟล์บีเพलय่อยที่ส่งมาจากคลาส TestCaseGenerator ว่ามีไฟล์บีเพलय่อยใดบ้างที่ยังไม่ถูกอ่านข้อมูล โดยเรียกใช้ฟังก์ชัน getNewBpelList ในการเปรียบเทียบชุดไฟล์บีเพलय่อย และเรียกใช้ฟังก์ชัน uplFile ของคลาส LocalFileController เพื่อทำการจัดเก็บชุดไฟล์บีเพलय่อยลงในแฟ้มโครงการ จากนั้นเครื่องมือจะวนลูปเพื่ออ่านไฟล์บีเพलय่อยแต่ละไฟล์ โดยมีการเรียกใช้ฟังก์ชัน updateBpelReadById ของคลาส Bpel เพื่อบันทึกข้อมูลการอ่านไฟล์บีเพलय่อยลงในฐานข้อมูล จากนั้นมีการค้นหาข้อมูลการเรียกใช้บีเพलय่อยจากฟังก์ชัน searchSbpelCalled และเมื่อค้นพบการเรียกใช้งานบีเพलय่อย เครื่องมือจะเรียกใช้งานฟังก์ชัน addSbpelFile ของคลาส Bpel เพื่อบันทึกข้อมูลไฟล์บีเพलय่อยลงฐานข้อมูล พร้อมทั้งเก็บสะสมข้อมูลไฟล์บีเพलय่อยที่ถูกเรียกใช้งานผ่านฟังก์ชัน addArrFileList เมื่อวนลูปอ่านไฟล์บีเพलय่อยจนครบทุกไฟล์แล้ว เครื่องมือจะคืนค่าชุดไฟล์บีเพलय่อยที่ถูกเรียกใช้งานผ่านฟังก์ชัน getArrFileList

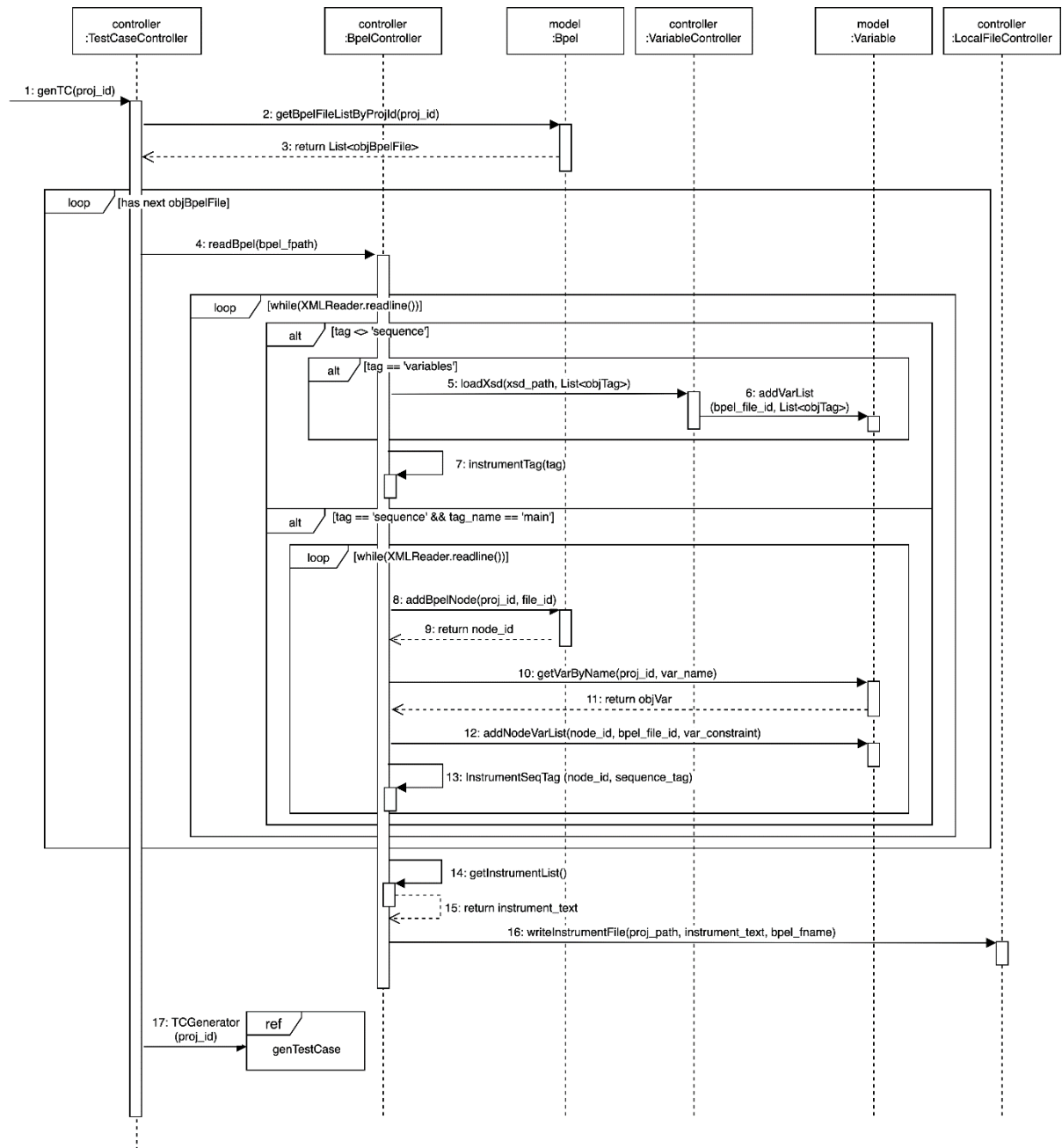
จากนั้นเครื่องมือจะตรวจสอบชุดไฟล์บีเพलय่อยที่ถูกเรียกใช้งาน จากผลลัพธ์ที่ได้จากการเรียกใช้ฟังก์ชัน createSbpelFile โดยมีเงื่อนไข ดังนี้

- หากเครื่องมือตรวจพบที่มีการเรียกใช้บีเพलय่อยมากกว่า 1 ไฟล์ เครื่องมือจะวนลูปเรียกใช้ฟังก์ชัน genFileInput เพื่อสร้างแถบอัปโหลดไฟล์ตามจำนวนชุดไฟล์บีเพलय่อย
- หากเครื่องมือไม่พบการเรียกใช้งานบีเพलय่อย เครื่องมือจะเรียกใช้ฟังก์ชัน goToNextStep โดยการส่งค่าขั้นตอนเป็น 3 เพื่อนำผู้ทดสอบไปสู่ขั้นตอนการแทรกชุดคำสั่งในรหัสต้นทาง

1.3) แผนภาพลำดับการแทรกชุดคำสั่งในรหัสต้นทาง

แผนภาพลำดับการแทรกชุดคำสั่งในรหัสต้นทาง ดังรูปที่ 4-26 เป็นแผนภาพลำดับที่แสดงถึงการแทรกชุดคำสั่งในรหัสต้นทางของไฟล์บีเพลและไฟล์บีเพलय่อย โดยเป็นการทำงานต่อจากขั้นตอนการนำเข้าข้อมูลตั้งต้นและการนำเข้าชุดไฟล์บีเพलय่อย ซึ่งเริ่มจากการเรียกใช้ฟังก์ชัน genTC ของคลาส TestCaseController

เครื่องมือเรียกใช้ฟังก์ชัน getBpelFileListByProjId ของคลาส Bpel เพื่อทำการดึงข้อมูลไฟล์บีเพลและชุดไฟล์บีเพलय่อยทั้งหมดของโครงการ จากนั้นมีการวนลูปที่ละไฟล์เพื่ออ่านค่าแท็กดับเบิลยูเอส-บีเพล โดยการเรียกใช้ฟังก์ชัน readBpelFile ของคลาส BpelController โดยในการอ่านแต่ละแท็กของไฟล์บีเพลมีเงื่อนไข ดังนี้



รูปที่ 4-26 แผนภาพลำดับการแทรกชุดคำสั่งในรหัสต้นทาง

- หากเป็นแท็กตัวแปร <variables> เครื่องมือจะเรียกใช้ฟังก์ชัน LoadXsd ของคลาส VariableController เพื่อค้นหาข้อมูลตัวแปรและข้อจำกัด และเมื่อวิเคราะห์และได้ข้อมูลตัวแปรเรียบร้อยแล้ว ให้เรียกใช้งานฟังก์ชัน addVarList

ของคลาส Variable เพื่อบันทึกข้อมูลตัวแปรและข้อจำกัดลงฐานข้อมูล และให้เก็บสะสมแท็กของรหัสต้นทางผ่านฟังก์ชัน instrumentTag

- หากไม่ใช่แท็กกิจกรรมของดับเบิลยูเอส-บีเพล ให้เก็บสะสมแท็กของรหัสต้นทางผ่านฟังก์ชัน instrumentTag
- หากเป็นแท็กกิจกรรมของดับเบิลยูเอส-บีเพล เครื่องมือจะเรียกใช้ฟังก์ชัน addBpelNode ของคลาส Bpel เพื่อเก็บข้อมูลเส้นทางการไหลของบีเพลลงฐานข้อมูล จากนั้นทำการดึงค่าข้อมูลตัวแปรและบันทึกข้อจำกัดของตัวแปรในแต่ละโหนดกิจกรรม ผ่านการเรียกใช้ฟังก์ชัน getVarByName และ addNodeList ของคลาส Variable ตามลำดับ จากนั้นเรียกใช้ฟังก์ชัน InstrumentSegTag เพื่อเก็บสะสมแท็กชุดคำสั่งและแท็กรหัสต้นทาง ผ่านฟังก์ชัน instrumentTag

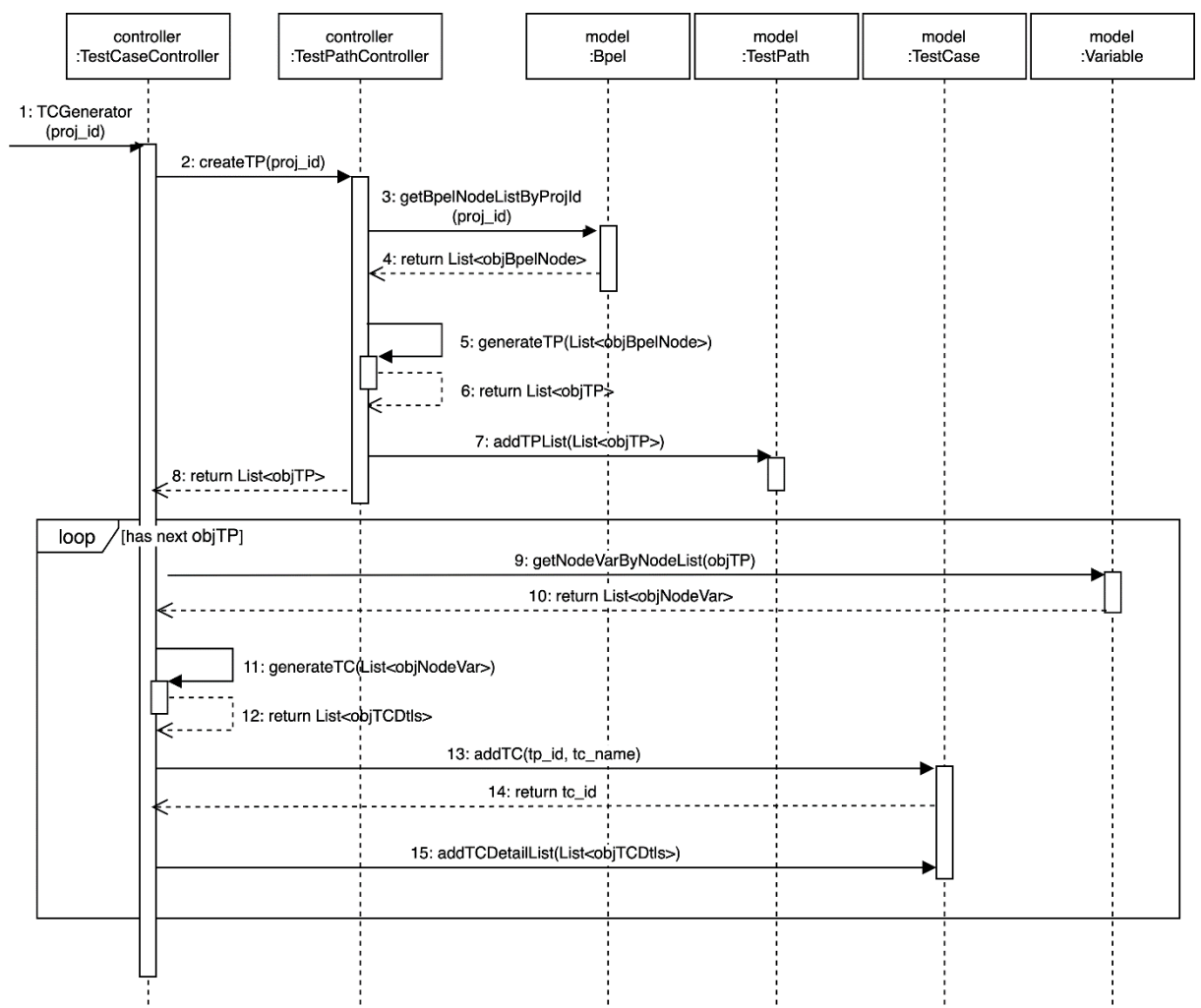
เมื่อวนลูปอ่านค่าข้อมูลครบทุกไฟล์บีเบลและไฟล์บีเพลย่อยแล้ว เครื่องมือเรียกใช้งานฟังก์ชัน getInstrumentList เพื่อนำข้อมูลการแทรกชุดคำสั่งในรหัสต้นทางไปเขียนไฟล์และจัดเก็บลงในแฟ้มโครงการดับเบิลยูเอส-บีเพลผ่านฟังก์ชัน writeInstrumentFile ของคลาส LocalFileController และสุดท้ายเครื่องมือจะทำการสร้างกรณีทดสอบ โดยจะมีการอธิบายเพิ่มเติมที่หัวข้อ 1.4

1.4) แผนภาพลำดับการสร้างกรณีทดสอบ

แผนภาพลำดับการสร้างกรณีทดสอบ เป็นแผนภาพที่แสดงถึงลำดับการสร้างกรณีทดสอบของเครื่องมือ โดยแผนภาพลำดับการสร้างกรณีทดสอบ แสดงดังรูปที่ 4-27 จะเป็นลำดับการทำงานที่ต่อจากหัวข้อ 1.3 การแทรกชุดคำสั่งในรหัสต้นทาง โดยเริ่มจากคลาส TestCaseController เรียกใช้ฟังก์ชัน createTP ของคลาส TestPathController เพื่อทำการสร้างทางเดินทดสอบ

โดยในฟังก์ชัน createTP มีการเรียกใช้ฟังก์ชัน getBpelNodeListByProjId ของคลาส Bpel เพื่อดึงค่าข้อมูลการไหลของบีเพลมาทำการสร้างกราฟการเรียกเชิงสถิติและสร้างทางเดินทดสอบผ่านฟังก์ชัน generateTP เมื่อสร้างข้อมูลทางเดินทดสอบถูกสร้างขึ้น เครื่องมือจึงเรียกใช้งานฟังก์ชัน addTPList ของคลาส TestPath เพื่อบันทึกทางเดินทดสอบลงฐานข้อมูล และคืนค่าข้อมูลทางเดินทดสอบกลับไปยังคลาส TestCaseController

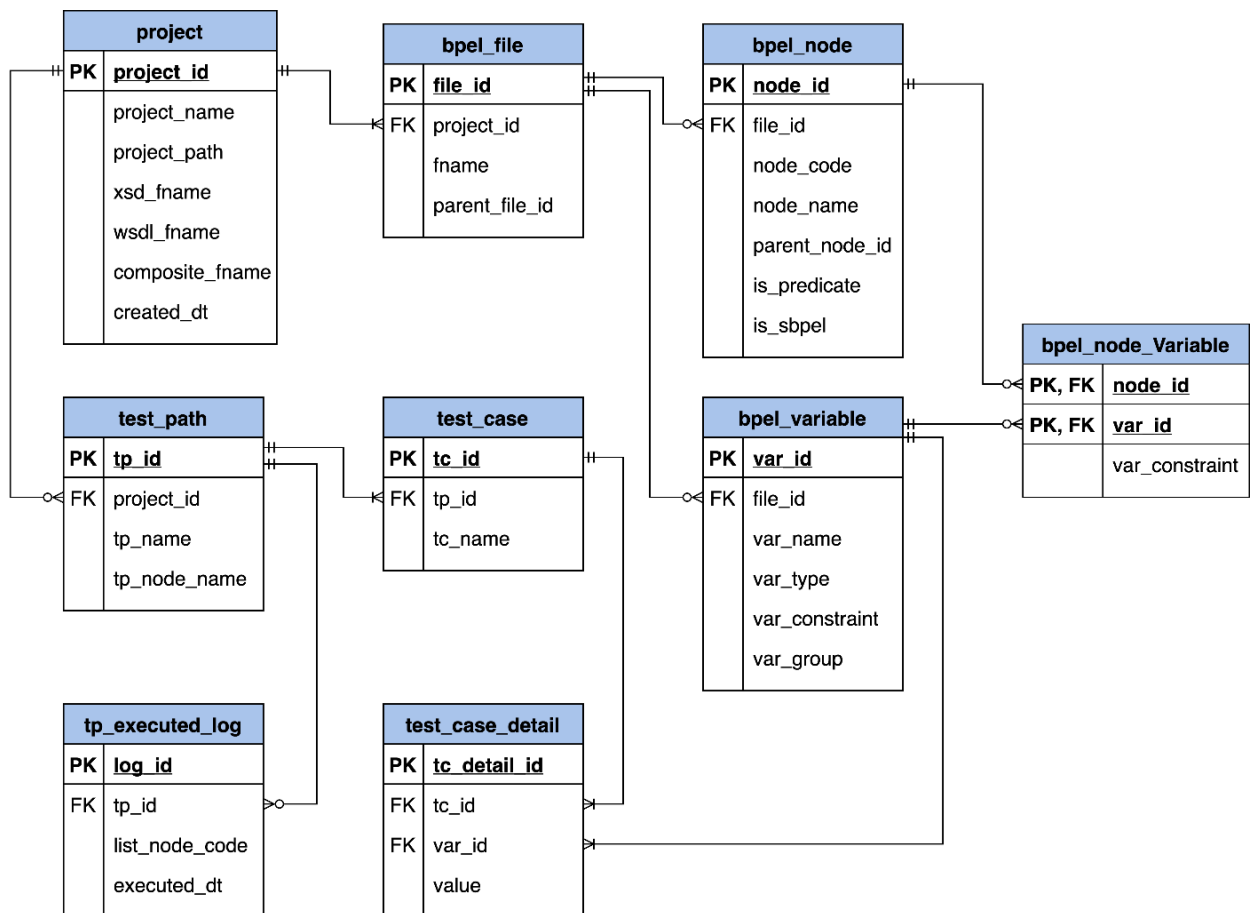
ที่คลาส TestCaseController มีการวนลูปสร้างกรณีทดสอบในแต่ละทางเดินทดสอบ โดยการเรียกใช้ฟังก์ชัน getNodeVarByNodeList ของคลาส Variable เพื่อดึงชุดข้อมูลตัวแปรนำเข้าและข้อจำกัดของแต่ละทางเดินทดสอบ จากนั้นเรียกใช้งานฟังก์ชัน addTC เพื่อสร้างกรณีทดสอบ โดยการส่งข้อมูลนำเข้าภายใต้ข้อจำกัดตัวแปรเพื่อสร้างเป็นกรณีทดสอบ และสุดท้ายมีการเรียกใช้งานฟังก์ชัน addTCDetailList เพื่อบันทึกข้อมูลกรณีทดสอบลงในฐานข้อมูล



รูปที่ 4-27 แผนภาพลำดับการสร้างกรณีทดสอบ

4.1.5 โครงสร้างฐานข้อมูล

โครงสร้างฐานข้อมูลเครื่องมือสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพล จากกราฟการเรียกเชิงสถิติ แสดงโดยแผนภาพอีอาร์ (ER: Entity Relationship Diagram) ดังรูปที่ 4-28 โดยมีรายละเอียดในแต่ละเอนทิตี ดังนี้



รูปที่ 4-28 แผนภาพอีอาร์ของเครื่องมือ

- 1) เอนทิตี project เป็นตารางที่ใช้จัดเก็บข้อมูลโครงการดับเบิลยูเอส-บีเพล
- 2) เอนทิตี bpel_file เป็นตารางที่ใช้จัดเก็บข้อมูลไฟล์บีเพลและลำดับการเรียกใช้บีเพลย่อยในแต่ละโครงการดับเบิลยูเอส-บีเพล
- 3) เอนทิตี bpel_activity เป็นตารางที่ใช้จัดเก็บรายละเอียดและลำดับของโหนดกิจกรรมในแต่ละไฟล์บีเพล
- 4) เอนทิตี bpel_var เป็นตารางที่ใช้จัดเก็บตัวแปรและข้อจำกัดของตัวแปรในแต่ละไฟล์บีเพล

- 5) เอนทิตี `bpel_act_var` เป็นตารางที่ใช้จัดเก็บรายละเอียดของตัวแปรที่อยู่ในแต่ละขั้นตอนกิจกรรม
- 6) เอนทิตี `test_path` เป็นตารางที่ใช้จัดเก็บทางเดินทดสอบแต่ละโครงการดับเบิลยูเอส-บีเพล
- 7) เอนทิตี `test_path_log` เป็นตารางที่ใช้จัดเก็บผลการทดสอบดับเบิลยูเอส-บีเพลในแต่ละทางเดินทดสอบ
- 8) เอนทิตี `test_case` เป็นตารางที่ใช้จัดเก็บกรณีทดสอบที่เครื่องมือสร้างขึ้นในแต่ละทางเดินทดสอบ
- 9) เอนทิตี `test_case_detail` เป็นตารางที่จัดเก็บค่าข้อมูลนำเข้าในแต่ละกรณีทดสอบ

4.2 การพัฒนาเครื่องมือ

จากการออกแบบเครื่องมือในหัวข้อ 4.1 ผู้วิจัยได้พัฒนาเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ โดยกำหนดสภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ อธิบายส่วนต่อประสานผู้ใช้ (User Interface) ด้วยแผนภาพวินโดว์เนวิกชัน (Window Navigation Diagram) และอธิบายการติดตั้งเครื่องมือด้วยแผนภาพการติดตั้ง (Deployment Diagram) ดังนี้

4.2.1 สภาพแวดล้อมที่ใช้ในการพัฒนา

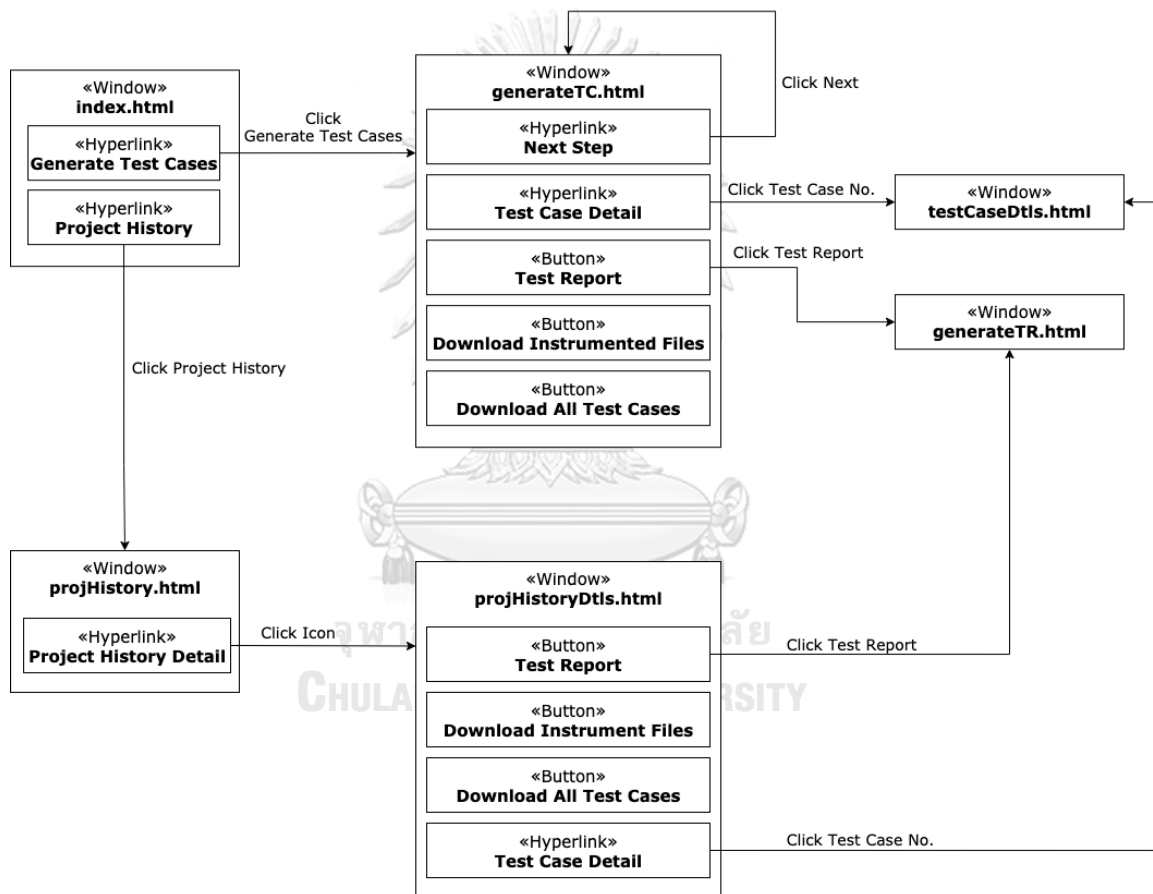
สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ แบ่งออกเป็น 2 ส่วน ได้แก่ ฮาร์ดแวร์ (Hardware) และซอฟต์แวร์ โดยมีรายละเอียด ดังนี้

- 1) ฮาร์ดแวร์
 - เครื่องคอมพิวเตอร์โน้ตบุ๊ก (Notebook) หน่วยประมวลผลอินเทลคอร์ไอโพลี รุ่นที่ 5 ความเร็ว 1.8 กิกะเฮิรซ์ (Intel Core i5-5350U 1.8 GHz)
 - หน่วยความจำหลัก (RAM) 8 กิกะไบต์ (8 GB)
 - ฮาร์ดดิสก์ (Hard Disk) 256 กิกะไบต์ (256 GB)
- 2) ซอฟต์แวร์
 - ระบบปฏิบัติการ (Mac OS) Mojave เวอร์ชัน 10.14.6
 - ภาษาที่ใช้พัฒนา คือ ภาษาพีเอชพี (PHP) เวอร์ชัน 7.3.8

- ฐานข้อมูลมายเอสคิวแอล (MySQL) เวอร์ชัน 5.7
- เครื่องมือบีเฟลเอนจินของ Oracle JDeveloper เวอร์ชัน 12c

4.2.2 ส่วนต่อประสานผู้ใช้

ส่วนต่อประสานผู้ใช้ของเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเฟลจากกราฟการเรียกเชิงสถิติ สามารถอธิบายได้ด้วยแผนภาพวินโดว์เนวิกชัน ดังรูปที่ 4-29 ซึ่งเป็นการอธิบายถึงความสัมพันธ์ของส่วนต่อประสานผู้ใช้ทั้งหมดของเครื่องมือ โดยมีรายละเอียดของแต่ละส่วนต่อประสาน ดังนี้

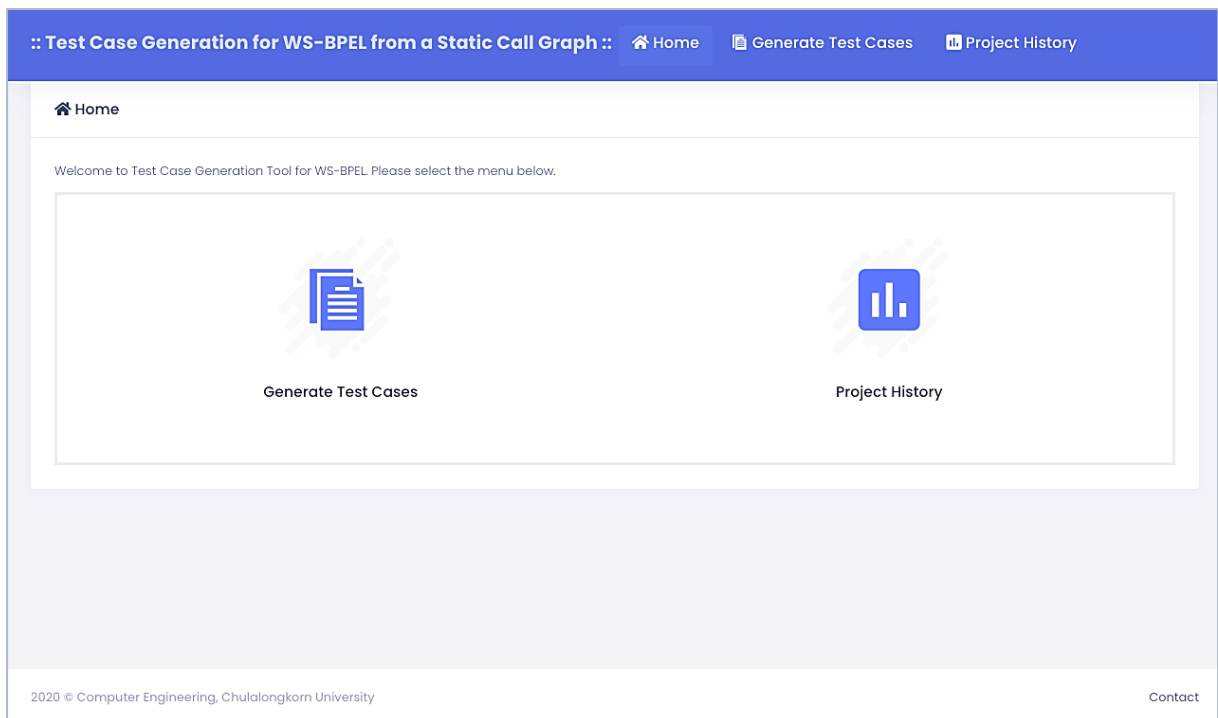


รูปที่ 4-29 แผนภาพวินโดว์เนวิกชันของเครื่องมือ

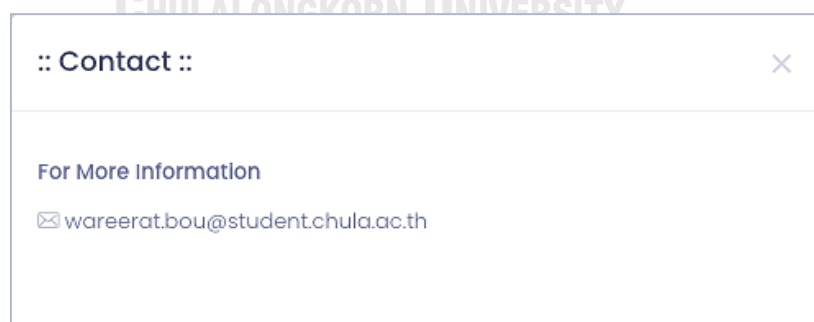
1) หน้าจอ index.html

หน้าจอ index.html ดังรูปที่ 4-30 เป็นหน้าจอเริ่มต้นของการใช้งานเครื่องมือ ผู้ทดสอบสามารถเลือกที่จะสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเฟล หรือเรียกดู

ประวัติการทดสอบโครงการดับเบิลยูเอส-พีเพิล และจากหน้าจอ index.html เมื่อผู้ทดสอบคลิก
ลิงก์ Contact ที่อยู่ด้านล่างขวาของหน้าจอจะพบกับหน้าจอติดต่อสอบถามข้อมูลเพิ่มเติม ดังรูป
ที่ 4-31



รูปที่ 4-30 หน้าจอ index.html



รูปที่ 4-31 หน้าจอติดต่อสอบถามข้อมูลเพิ่มเติม

2) หน้าจอ generateTC.html

หน้าจอ generateTC.html เป็นหน้าจอสำหรับให้ผู้ทดสอบสร้างกรณีทดสอบ โดยในการสร้างกรณีทดสอบนั้น ผู้ทดสอบจะต้องทำ 3 ขั้นตอนตามลำดับ คือ นำเข้าข้อมูลตั้งต้น นำเข้าชุดไฟล์บีเพलय่อย และสร้างกรณีทดสอบ

- 2.1) ขั้นตอนนำเข้าข้อมูลตั้งต้นของหน้าจอ generateTC.html แสดงดังรูป 4-32 โดยข้อมูลตั้งต้นประกอบไปด้วย ชื่อโครงการดับเบิลยูเอส-บีเพล ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล ไฟล์บีเพลตั้งต้น ไฟล์ดับเบิลยูเอสดีแอล และไฟล์เอกซ์เอสดี โดยเมื่อผู้ทดสอบกดปุ่ม Next Step เครื่องมือจะนำผู้ทดสอบไปยังขั้นตอนนำเข้าชุดไฟล์บีเพलय่อย

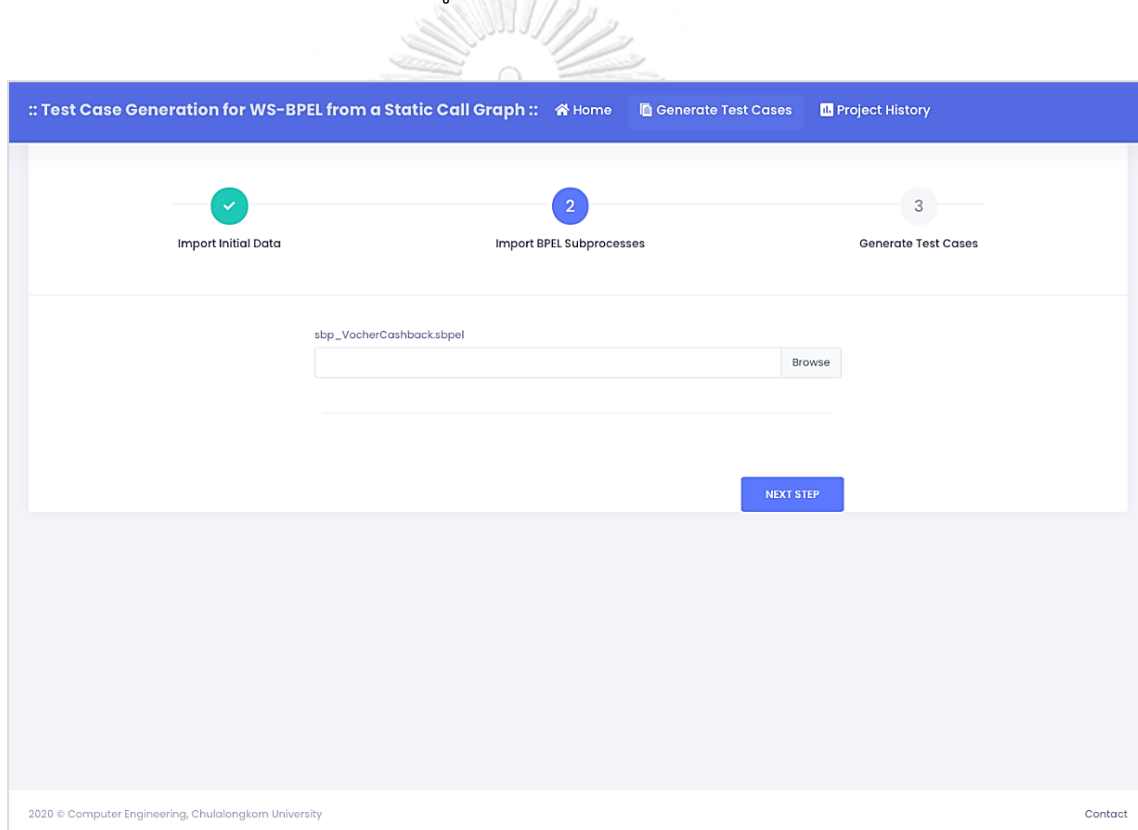
The screenshot displays the 'Test Case Generation for WS-BPEL from a Static Call Graph' web application. The interface is divided into three main steps: 1. Import Initial Data, 2. Import BPEL Subprocesses, and 3. Generate Test Cases. The first step is currently active. The form includes several input fields and buttons: 'Project Name' (text input), 'Composite File' (text input with a 'Browse' button), 'BPEL File' (text input with a 'Browse' button), 'WSDL File' (text input with a 'Browse' button), and 'XSD File' (text input with a 'Browse' button). A 'NEXT STEP' button is located at the bottom right of the form area. The footer contains the text '2020 © Computer Engineering, Chulalongkorn University' and a 'Contact' link.

รูปที่ 4-32 ขั้นตอนนำเข้าข้อมูลตั้งต้นของหน้าจอ generateTC.html

ขั้นตอนนำเข้าชุดไฟล์บีเพलय่อยของหน้าจอ generateTC.html แสดงดังรูป 4-33 โดยเมื่อผู้ทดสอบนำเข้าชุดไฟล์บีเพलय่อยและกดปุ่ม Next Step เครื่องมือจะค้นหาการเรียกใช้บีเพล

ย่อยจากไฟล์ที่ผู้ทดสอบนำเข้า หากเครื่องมือพบการเรียกใช้บีเพลย่อย เครื่องมือจะแสดงแถบอัปโหลดไฟล์เพิ่มเติมตามจำนวนบีเพลย่อยที่ถูกเรียกใช้ ดังรูปที่ 4-34 และผู้ทดสอบจะต้องนำเข้าไฟล์บีเพลย่อยเพิ่มเติมไปเรื่อย ๆ ตามจำนวนแถบอัปโหลดไฟล์ที่เครื่องมือแสดง ดังรูปที่ 4-35 จนกระทั่งเครื่องมือไม่พบการเรียกใช้งานบีเพลย่อยเพิ่มเติมอีก

- 2.2) ขั้นตอนสร้างกรณีทดสอบของหน้าจอ generateTC.html แสดงดังรูป 4-36 โดยเครื่องมือจะแสดงทางเดินทดสอบและกรณีทดสอบที่อยู่บนทางเดินทดสอบในรูปแบบตาราง และเมื่อผู้ทดสอบกดรหัสกรณีทดสอบที่สนใจที่อยู่ในแต่ละแถวของตาราง เครื่องมือจะแสดงหน้าจอรายละเอียดของกรณีทดสอบที่ผู้ทดสอบเลือก



รูปที่ 4-33 ขั้นตอนนำเข้าชุดไฟล์บีเพลย่อยของหน้าจอ generateTC.html

Test Case Generation for WS-BPEL from a Static Call Graph :: Home Generate Test Cases Project History

1 Import Initial Data 2 Import BPEL Subprocesses 3 Generate Test Cases

sbp_VoucherCashback.sbpel Browse

sbp_VISAOnTop.sbpel Browse

NEXT STEP

2020 © Computer Engineering, Chulalongkorn University Contact

รูปที่ 4-34 ขั้นตอนนำเข้าบีเพलय่อย sbp_VISAOnTop ที่ถูกเรียกใช้งาน

Test Case Generation for WS-BPEL from a Static Call Graph :: Home Generate Test Cases Project History

1 Import Initial Data 2 Import BPEL Subprocesses 3 Generate Test Cases

sbp_VoucherCashback.sbpel Browse

sbp_VISAOnTop.sbpel Browse

sbp_ChildPromotion.sbpel Browse

sbp_Masks.sbpel Browse

NEXT STEP

2020 © Computer Engineering, Chulalongkorn University Contact

รูปที่ 4-35 ขั้นตอนนำเข้าบีเพलय่อย sbp_ChildPromotion และ sbp_Masks ที่ถูกเรียกใช้งาน

:: Test Case Generation for WS-BPEL from a Static Call Graph :: [Home](#) [Generate Test Cases](#) [Project History](#)

✓ Import Initial Data
 ✓ Import BPEL Subprocesses
 3 Generate Test Cases

[TEST REPORT](#)
[DOWNLOAD](#)

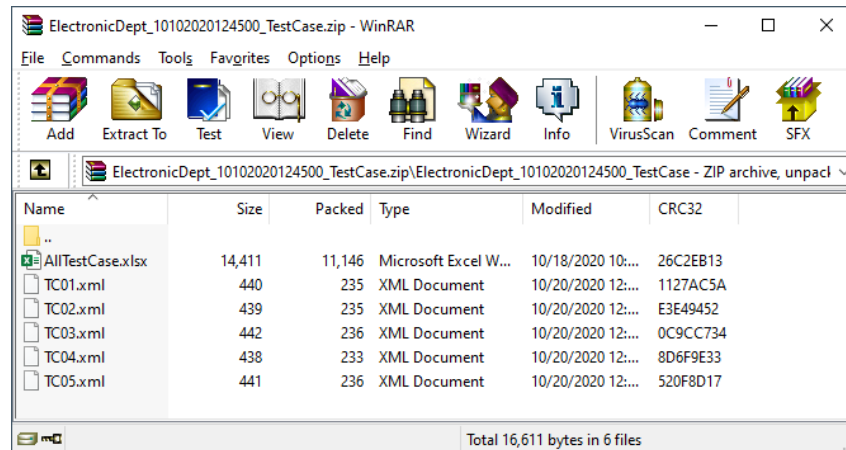
#	Test Path Description	Test Case No.
1	RecieveInput - Discount5Percent - IfCCAllow(T) - IfPrice2500Up(T) - IfPrice5000Up(T) - IfPrice10000Up(T) - Voucher700THB - AssignOutput - ReplyOutput	TC01
2	RecieveInput - Discount5Percent - IfCCAllow(T) - IfPrice2500Up(T) - IfPrice5000Up(T) - IfPrice10000Up(F) - Voucher200THB - AssignOutput - ReplyOutput	TC02
3	RecieveInput - Discount5Percent - IfCCAllow(T) - IfPrice2500Up(T) - IfPrice5000Up(F) - Voucher200THB - AssignOutput - ReplyOutput	TC03
4	RecieveInput - Discount5Percent - IfCCAllow(T) - IfPrice2500Up(F) - Voucher0THB - AssignOutput - ReplyOutput	TC04
5	RecieveInput - Discount5Percent - IfCCAllow(F) - AssignVoucher0 - AssignOutput - ReplyOutput	TC05

2020 © Computer Engineering, Chulalongkorn University Contact

รูปที่ 4-36 ขั้นตอนสร้างกรณีทดสอบของหน้าจอ generateTC.html

จากหน้าจอ generateTC.html ดังรูปที่ 4-36 ผู้ทดสอบสามารถเรียกดูรายงานและดาวน์โหลดข้อมูลต่าง ๆ ดังนี้

- กดปุ่ม Test Report เพื่อเรียกดูรายงานการทดสอบฉบับเบิลยูเอส-บีเพล โดยรายละเอียดของหน้าจอแสดงรายงานการทดสอบฉบับเบิลยูเอส-บีเพล
- กดปุ่ม Download จากนั้นเลือก Instrumented Source Code เพื่อดาวน์โหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่งลงในเครื่องคอมพิวเตอร์ของผู้ทดสอบ
- กดปุ่ม Download จากนั้นเลือก Instrumented Files เพื่อดาวน์โหลดกรณีทดสอบลงในเครื่องคอมพิวเตอร์ของผู้ทดสอบ ข้อมูลกรณีทดสอบที่ดาวน์โหลดแสดงดังรูปที่ 4-37 ประกอบไปด้วยกรณีทดสอบที่อยู่ในรูปแบบไฟล์เอกซ์เซล ดังรูปที่ 4-38 และกรณีทดสอบที่อยู่ในรูปแบบไฟล์เอกซ์เอ็มแอล ดังรูปที่ 4-39



รูปที่ 4-37 กรณีทดสอบที่ดาวน์โหลดจากเครื่องมือ

Variable Name	Data Type	Value
Price	decimal	34690.42
SOF	String	CREDITCARD
CCNo	String	5059232372440152
CreditCardType	String	JCB

รูปที่ 4-38 กรณีทดสอบในรูปแบบไฟล์เอกซ์เซล

```

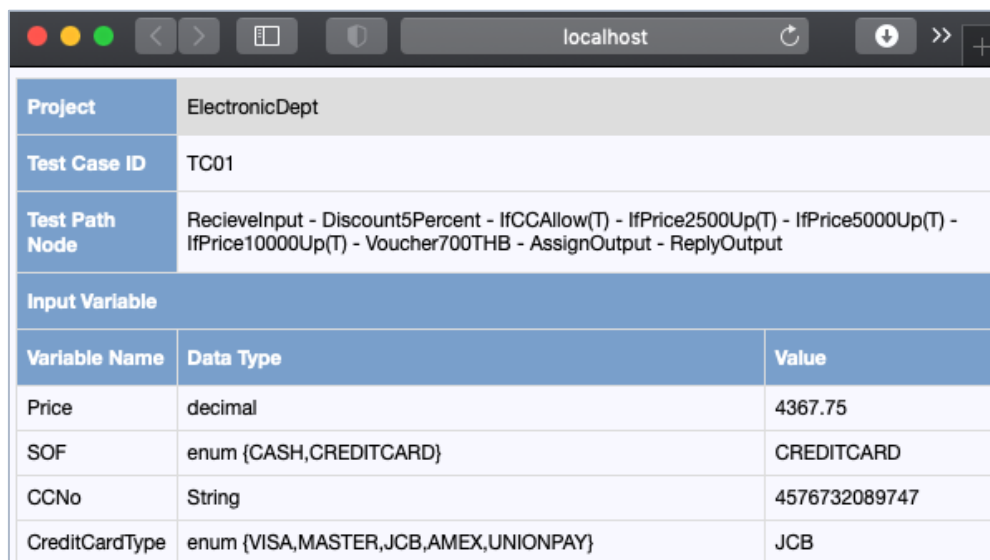
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
3   <soap:Body>
4     <ns1:process xmlns:ns1="http://xmlns.oracle.com/Application1/Project1_Promotion/ElectronicDept">
5       <ns1:Price>34690.42</ns1:Price>
6       <ns1:SOF>CREDITCARD</ns1:SOF>
7       <ns1:CCNo>5059232372440152</ns1:CCNo>
8       <ns1:CCType>JCB</ns1:CCType>
9     </ns1:process>
10  </soap:Body>
11 </soap:Envelope>

```

รูปที่ 4-39 กรณีทดสอบในรูปแบบไฟล์เอกซ์เอ็มแอล

3) หน้าจอ testCaseDtls.html

หน้าจอ testCaseDtls.html ดังรูปที่ 4-40 เป็นหน้าจอสำหรับผู้ทดสอบเรียกดูรายละเอียดกรณีทดสอบในแต่ละเส้นทาง



Project	ElectronicDept	
Test Case ID	TC01	
Test Path Node	RecieveInput - Discount5Percent - IfCCAllow(T) - IfPrice2500Up(T) - IfPrice5000Up(T) - IfPrice10000Up(T) - Voucher700THB - AssignOutput - ReplyOutput	
Input Variable		
Variable Name	Data Type	Value
Price	decimal	4367.75
SOF	enum {CASH,CREDITCARD}	CREDITCARD
CCNo	String	4576732089747
CreditCardType	enum {VISA,MASTER,JCB,AMEX,UNIONPAY}	JCB

รูปที่ 4-40 หน้าจอ testCaseDtls.html

4) หน้าจอ generateTR.html

หน้าจอ generateTR.html ดังรูปที่ 4-41 เป็นหน้าจอสำหรับผู้ทดสอบเรียกดูรายงานการทดสอบดับเบิลยูเอส-บีเพล

5) หน้าจอ projectHistory.html

หน้าจอ projectHistory.html ดังรูปที่ 4-42 เป็นหน้าจอสำหรับผู้ทดสอบเรียกดูประวัติการทดสอบโครงการดับเบิลยูเอส-บีเพล เมื่อผู้ทดสอบกดไอคอนแว่นขยายในแต่ละแถวของโครงการดับเบิลยูเอส-บีเพลที่สนใจ เครื่องมือจะแสดงรายละเอียดการทดสอบของโครงการดับเบิลยูเอส-บีเพลที่เลือก

6) หน้าจอ projectHistoryDtls.html

หน้าจอ projectHistoryDtls.html ดังรูปที่ 4-43 เป็นหน้าจอสำหรับผู้ทดสอบเรียกดูรายละเอียดการทดสอบดับเบิลยูเอส-บีเพลของโครงการ เครื่องมือจะแสดงทางเดินทดสอบและกรณีทดสอบที่อยู่บนทางเดินทดสอบในรูปแบบตาราง ผู้ทดสอบสามารถเรียกดูรายละเอียดของ

แต่ละกรณีทดสอบโดยการกดที่หมายเลขกรณีทดสอบที่สนใจที่อยู่ในแต่ละแถวของตาราง หรือ กดปุ่ม Test Report เมื่อผู้ทดสอบต้องการเรียกดูรายงานการทดสอบฉบับเบสิลยูเอส-บีเพล หาก ต้องการดาวน์โหลดข้อมูล ให้ผู้ทดสอบกดปุ่ม Download จากนั้นเลือก Instrumented Files เพื่อดาวน์โหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่ง หรือเลือก All Test Cases เพื่อดาวน์โหลดกรณี ทดสอบทั้งหมด

:: WS-BPEL Test Report ::

Project Name: ElectronicDept
 Total Test: 3 Time(s)
 Printed Date: 13 October 2020 15:44

Path ID	Path Detail	Test Case	Exercised?
P1	1-2-3.1-3.2-3.3-3.4.1-3.4.2-3.4.3-3.4.4-4	TC01	Exercised
P2	1-2-3.1-3.2-3.3-3.4.1-3.4.2-3.4.3-3.4.5-4	TC02	Exercised
P3	1-2-3.1-3.2-3.3-3.4.1-3.4.2-3.4.6-4	TC03	Exercised
P4	1-2-3.1-3.2-3.3-3.4.1-3.4.7-4	TC04	Not Exercised
P5	1-2-3.1-3.2-3.5-4	TC05	Not Exercised
P6	1-3.1-3.6-4	TC06	Not Exercised

:: Alternative node has been exercised :: (7/10)

Predicate Node	Condition	Exercised?	Test Case
3.1(T)	SOF='CREDITCARD'	Exercised	TC01 TC02 TC03 TC04 TC05 TC06
3.1(F)	NOT(SOF='CREDITCARD')	Not Exercised	TC06
3.2(T)	CCType='VISA'	Exercised	TC01 TC02 TC03 TC04
3.2(F)	NOT(CCType='VISA')	Not Exercised	TC05
3.4.1(T)	Price>2500	Exercised	TC01 TC02 TC03
3.4.1(F)	NOT(Price>2500)	Not Exercised	TC04
3.4.2(T)	Price>5000	Exercised	TC01 TC02
3.4.2(F)	NOT(Price>5000)	Exercised	TC03
3.4.3(T)	Price>10000	Exercised	TC01
3.4.3(F)	NOT(Price>10000)	Exercised	TC02

รูปที่ 4-41 หน้าจอ generateTR.html

:: Test Case Generation for WS-BPEL from a Static Call Graph :: [Home](#) [Generate Test Cases](#) [Project History](#)

Project History

#	Project Name	Project Created Date	
1	ElectronicDept	10-Oct-2020 12:45	Q
2	Demo-Project	10-Oct-2020 10:32	Q
3	Kiosk Process 02	10-Oct-2020 09:17	Q
4	Kiosk Process	9-Oct-2020 12:49	Q
5	ElectronicDept	9-Oct-2020 06:33	Q
6	Project01	8-Oct-2020 01:29	Q

2020 © Computer Engineering, Chulalongkorn University Contact

รูปที่ 4-42 หน้าจอ projectHistory.html

:: Test Case Generation for WS-BPEL from a Static Call Graph :: [Home](#) [Generate Test Cases](#) [Project History](#)

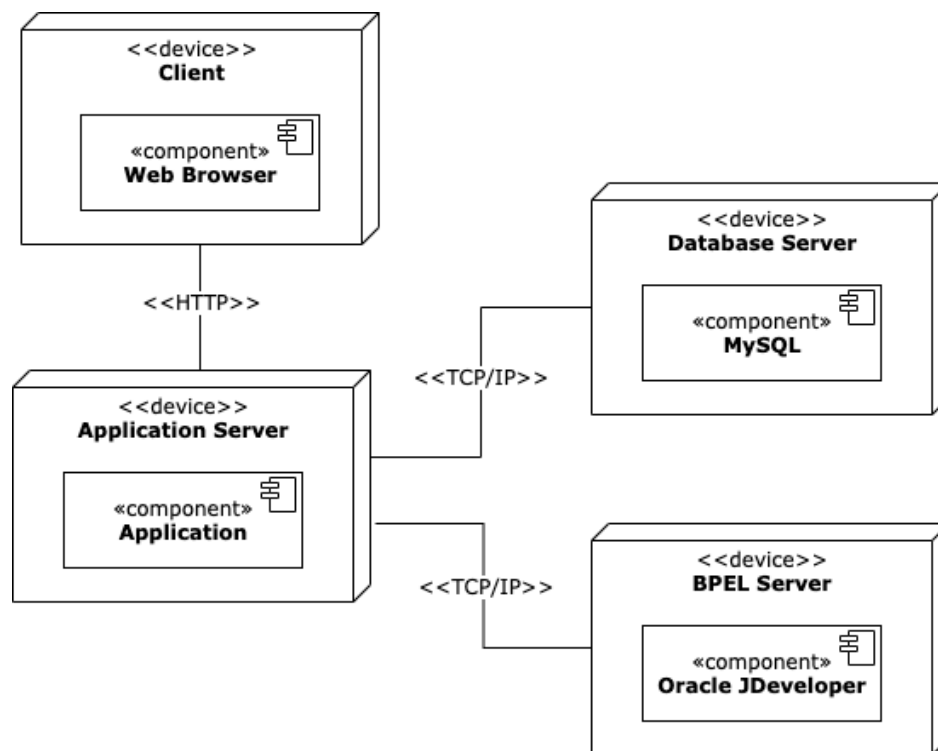
Project History : ElectronicDept : [TEST REPORT](#) [DOWNLOAD](#)

#	Test Path Description	Test Case No.
1	RecieveInput - Discount5Percent - IfCCAllow(T) - IfPrice2500Up(T) - IfPrice5000Up(T) - IfPrice10000Up(T) - Voucher700THB - AssignOutput - ReplyOutput	TC01
2	RecieveInput - Discount5Percent - IfCCAllow(T) - IfPrice2500Up(T) - IfPrice5000Up(T) - IfPrice10000Up(F) - Voucher200THB - AssignOutput - ReplyOutput	TC02
3	RecieveInput - Discount5Percent - IfCCAllow(T) - IfPrice2500Up(T) - IfPrice5000Up(F) - Voucher200THB - AssignOutput - ReplyOutput	TC03
4	RecieveInput - Discount5Percent - IfCCAllow(T) - IfPrice2500Up(F) - Voucher0THB - AssignOutput - ReplyOutput	TC04
5	RecieveInput - Discount5Percent - IfCCAllow(F) - AssignVoucher0 - AssignOutput - ReplyOutput	TC05

2020 © Computer Engineering, Chulalongkorn University Contact

รูปที่ 4-43 หน้าจอ projectHistoryDtls.html

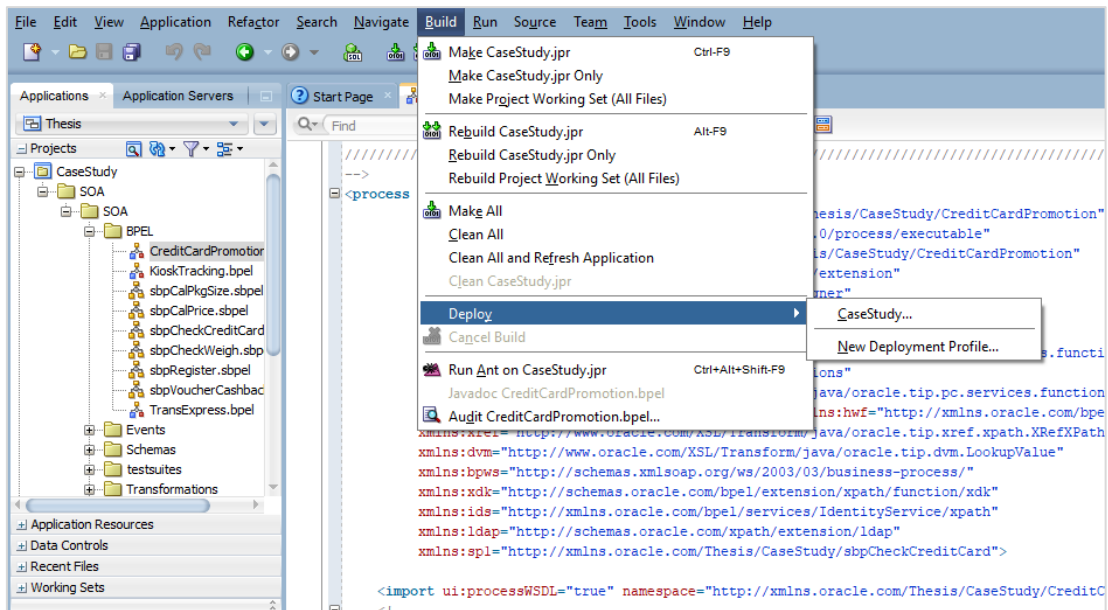
4.2.3 แผนภาพการติดตั้ง



รูปที่ 4-44 แผนภาพการติดตั้งเครื่องมือ

แผนภาพการติดตั้งเครื่องมือสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพล จากกราฟการเรียกเชิงสถิติ ดังรูปที่ 4-44 แสดงให้เห็นถึงสถาปัตยกรรมของซอฟต์แวร์และการติดตั้งซอฟต์แวร์บนอุปกรณ์ต่าง ๆ โดยเครื่องลูกข่าย (Client) มีการเรียกใช้งานเครื่องมือซึ่งถูกติดตั้งอยู่บนแอปพลิเคชันเซิร์ฟเวอร์ (Application Server) และเครื่องมือมีการเรียกใช้งานฐานข้อมูลมายเอสคิวแอลซึ่งถูกติดตั้งอยู่บนเซิร์ฟเวอร์ฐานข้อมูล (Database Server)

การติดตั้งรหัสต้นทางที่ถูกแทรกชุดคำสั่งและการทดสอบดับเบิลยูเอส-บีเพลผ่านบีเพลเอนจินของ Oracle JDeveloper ที่ถูกติดตั้งอยู่บนบีเพลเอนจินเซิร์ฟเวอร์ (BPEL Engine Server) ทำได้โดยผู้ทดสอบจะต้องดาวน์โหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่งจากเครื่องมือ จากนั้นทำการติดตั้งรหัสต้นทางที่ถูกแทรกชุดคำสั่งบนบีเพลเอนจินโดยผ่านการใช้งานเมนู Build และเลือก Deploy ดังรูปที่ 4-45



รูปที่ 4-45 การติดตั้งรหัสต้นทางที่ถูกแทรกชุดคำสั่งบนบีเพลเอนจิน

เมื่อติดตั้งรหัสต้นทางบนบีเพลเอนจินเรียบร้อยแล้ว ให้ผู้ทดสอบทำการทดสอบดับเบิลยูเอส-บีเพลโดยการเข้าไปที่ Oracle Enterprise Manager ของบีเพลเอนจิน และนำเข้ากรณีทดสอบต่าง ๆ ที่ดาวน์โหลดจากเครื่องมือมาทำการทดสอบร่วมกับรหัสต้นทางที่ถูกแทรกชุดคำสั่ง ดังรูปที่ 4-46 และรูปที่ 4-47 โดยบีเพลเอนจินจะส่งข้อมูลการทดสอบให้กับเครื่องมือผ่านทางเว็บเซอร์วิสที่เครื่องมือได้แทรกชุดคำสั่งไว้ และเมื่อเครื่องมือได้รับค่าข้อมูลการทดสอบที่ส่งมาจากบีเพลเอนจิน เครื่องมือจะทำการบันทึกผลการทดสอบของดับเบิลยูเอส-บีเพลลงในฐานข้อมูล

จ. ภาลกรณัฒมหาวิทยาลัย

Input Arguments

Tree View Enable Validation Load Payload

TC01.xml

SOAP Body


View

Name	Type	Value
* payload	payload	
* Price	decimal	34690.42
* SOF	string	CREDITCARD
* CCNo	integer	5059232372440152
* CCType	string	VISA

Request Response

รูปที่ 4-46 การนำเข้ากรณีทดสอบบนบีเพลเอนจิน

Request **Response**

Test Status Request successfully received. 

Response Time (ms) 10308

Tree View

A new flow instance was generated.

Name	Type	Value
payload	payload	
IsAllow	boolean	true
NetPrice	decimal	32955.9
VoucherValue	integer	700

Request **Response**

รูปที่ 4-47 การทดสอบด้วยเบส-บีเพลนบีเพลนเงิน



บทที่ 5

การทดสอบเครื่องมือ

ในหัวข้อนี้จะกล่าวถึงการทดสอบเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ ซึ่งจะกล่าวถึงสภาพแวดล้อมที่ใช้ในการทดสอบขั้นตอนในการทดสอบเครื่องมือ และการทดสอบเครื่องมือกับ 3 กรณีศึกษา ได้แก่ กรณีศึกษาระบบโปรมิชั่นรับเครดิตเงินคืนของแผนกเครื่องใช้ไฟฟ้า ระบบร้านรับขนส่งพัสดุ และระบบติดตามการทำงานตู้คีออส (Kiosk) รวมถึงสรุปผลการทดสอบเครื่องมือ โดยมีรายละเอียด ดังต่อไปนี้

5.1 สภาพแวดล้อมที่ใช้ในการทดสอบ

สภาพแวดล้อมที่ใช้ในการทดสอบเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ ผู้วิจัยได้ใช้สภาพแวดล้อมเดียวกับการพัฒนาเครื่องมือตามที่ได้อธิบายไว้ในหัวข้อ 4.2.1

5.2 ขั้นตอนในการทดสอบเครื่องมือ

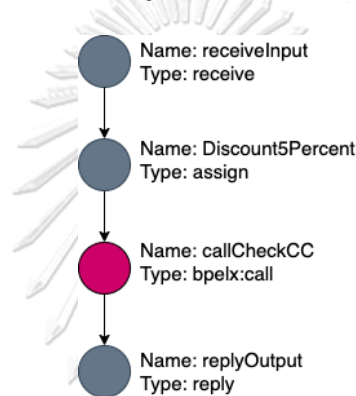
- 1) ผู้ทดสอบนำเข้าชุดไฟล์ตั้งต้น ได้แก่ ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล ไฟล์บีเพลตั้งต้น ไฟล์ดับเบิลยูเอสดีแอล และไฟล์เอกซ์เอสดี
- 2) ผู้ทดสอบนำเข้าชุดไฟล์บีเพลย่อยที่ถูกเรียกใช้งานให้ถูกต้องและครบถ้วนตามจำนวนแถบอัปโหลดไฟล์ที่เครื่องมือแสดง
- 3) ผู้ทดสอบดาวน์โหลดรหัสต้นทางที่แทรกชุดคำสั่งและกรณีทดสอบดับเบิลยูเอส-บีเพล
- 4) ผู้ทดสอบทำการทดสอบดับเบิลยูเอส-บีเพลบนบีเพลเอนจินที่มีการติดตั้งรหัสต้นทางที่แทรกชุดคำสั่ง โดยทดสอบร่วมกับกรณีทดสอบที่ดาวน์โหลดจากเครื่องมือ
- 5) ผู้ทดสอบเรียกดูรายงานการทดสอบดับเบิลยูเอส-บีเพล

5.3 การทดสอบเครื่องมือ

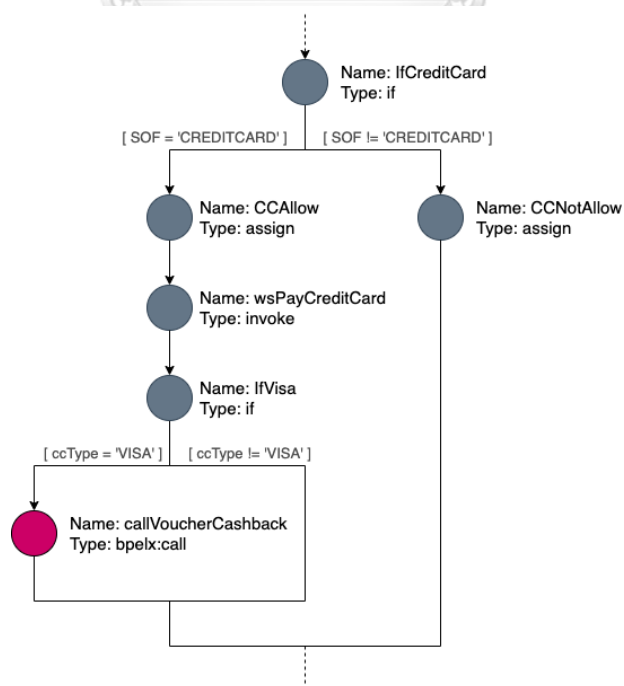
การทดสอบเครื่องมือการสร้างกรณีทดสอบสำหรับดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ เพื่อตรวจสอบความถูกต้องของเครื่องมือว่าสามารถทำงานได้ถูกต้องและสามารถสร้างกรณีทดสอบได้ครอบคลุมทางเดินทดสอบและโหนดเงื่อนไขในกราฟการเรียกเชิงสถิติหรือไม่ โดยผู้วิจัยได้ทำการทดสอบเครื่องมือกับ 3 กรณีศึกษา ดังต่อไปนี้

5.1.1 กรณีศึกษาที่ 1 ระบบโปรโมชั่นรับเครดิตเงินคืนของแผนกเครื่องใช้ไฟฟ้า

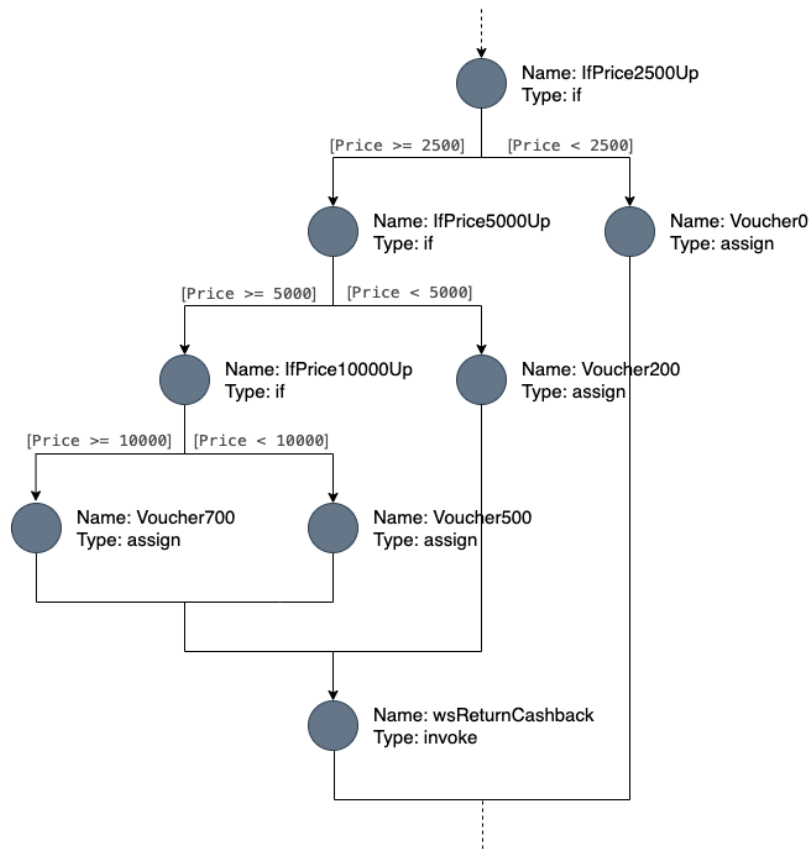
ระบบโปรโมชั่นรับเครดิตเงินคืนของแผนกเครื่องใช้ไฟฟ้าในห้างสรรพสินค้าแห่งหนึ่ง มีขั้นตอนเริ่มจากลูกค้าชำระสินค้าและบริการที่แผนกเครื่องใช้ไฟฟ้า โดยลูกค้าจะต้องแจ้งว่าจะชำระสินค้าด้วยช่องทางใด เงินสดหรือบัตรเครดิต หากเป็นบัตรเครดิตให้ระบุหมายเลขบัตรเครดิตและประเภทของบัตรเครดิต เพื่อที่จะตัดเงินค่าสินค้าและบริการ โดยหักส่วนลดตามโปรโมชั่นประจำเดือนของแผนก และจ่ายเครดิตเงินคืนให้กับลูกค้าตามเงื่อนไขการชำระเงินที่กำหนด ซึ่งกระบวนการดับเบิลยูเอส-บีเพลของกรณีศึกษานี้ แสดงดังรูปที่ 5-1 ถึงรูปที่ 5-3 โดยตัวแปรนำเข้าและตัวแปรส่งออกกระบวนการดับเบิลยูเอส-บีเพลที่อยู่ในไฟล์เอกซ์เอสดี แสดงดังรูปที่ 5-4



รูปที่ 5-1 กระบวนการดับเบิลยูเอส-บีเพลในไฟล์บีเพลตั้งต้น CreditCardPromotion



รูปที่ 5-2 กระบวนการดับเบิลยูเอส-บีเพลในไฟล์บีเพลย่อย sbpCheckCreditCard



รูปที่ 5-3 กระบวนการดับเบิลยูเอส-บีเพลในไฟล์บีเพลย่อย sbpVoucherCashback

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <schema attributeFormDefault="unqualified" elementFormDefault="qualified" targ
3 >
4 <simpleType name="CreditCardType" id="CreditCardType" final="restriction">
5 </simpleType>
6 </simpleType>
7 <simpleType name="SourceOfFund" id="SourceOfFund" final="restriction">--
8 </simpleType>
9 </simpleType>
10 <simpleType name="SourceOfFund" id="SourceOfFund" final="restriction">--
11 </simpleType>
12 </simpleType>
13 <element name="process">
14 <complexType>
15 <sequence>
16 <element name="Price">--
17 </element>
18 <element name="SOF" type="kt:SourceOfFund" />
19 <element name="CCNo">
20 <simpleType--
21 </simpleType>
22 </element>
23 <element name="CCType" type="kt:CreditCardType" />
24 </sequence>
25 </complexType>
26 </element>
27 <element name="processResponse">
28 <complexType>
29 <sequence>
30 <element name="IsAllow" type="boolean" />
31 <element name="NetPrice" type="decimal" default="0" />
32 <element name="VoucherValue" type="integer" default="0" />
33 </sequence>
34 </complexType>
35 </element>
36 </schema>
  
```

รูปที่ 5-4 ตัวแปรนำเข้าและตัวแปรส่งออกกระบวนการดับเบิลยูเอส-บีเพลของกรณีศึกษาที่ 1

การทดสอบกรณีศึกษาที่ 1 ผู้ทดสอบจะต้องทำตามขั้นตอนการทดสอบเครื่องมือ โดยมีขั้นตอนการทดสอบ ดังนี้

1) นำเข้าข้อมูลตั้งต้น ผู้ทดสอบนำเข้าข้อมูลตั้งต้นผ่านหน้าจอของเครื่องมือ แสดงดังรูปที่ 5-5 โดยข้อมูลตั้งต้นที่จะนำเข้าประกอบไปด้วยข้อมูล ดังนี้

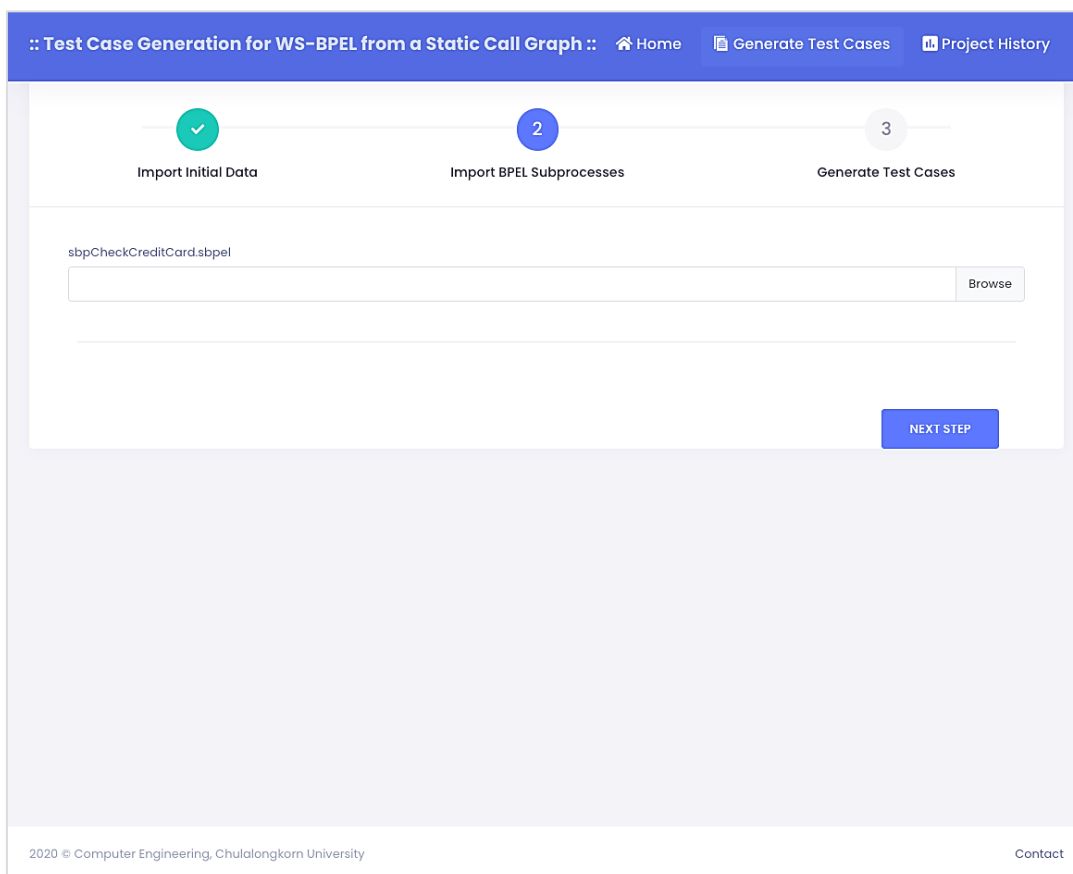
- ชื่อโครงการดับเบิลยูเอส-บีเพล CreditCardPromotion
- ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล Composite.xml
- ไฟล์บีเพลตั้งต้น CreditCardPromotion.bpel
- ไฟล์ดับเบิลยูเอสดีแอล CreditCardPromotion.wsdl
- ไฟล์เอกซ์เอสดี CreditCardPromotion.xsd

เมื่อผู้ทดสอบนำเข้าข้อมูลตั้งต้นเสร็จเรียบร้อยแล้ว ผู้ทดสอบกดปุ่ม Next Step โดยเครื่องมือจะนำผู้ทดสอบไปยังหน้าจอนำเข้าชุดไฟล์บีเพลย่อย

The screenshot displays the 'Test Case Generation for WS-BPEL from a Static Call Graph' web application. The interface is divided into three main steps: 1. Import Initial Data, 2. Import BPEL Subprocesses, and 3. Generate Test Cases. The first step is currently active. It includes a 'Project Name' field with the value 'CreditCardPromotion'. Below this are five file selection fields, each with a 'Browse' button: 'Composite File' (\\Mac\Home\Documents\JDeveloper\Thesis\ElectronicDeptPromotion\SOA\composite.xml), 'BPEL File' (\\Mac\Home\Documents\JDeveloper\Thesis\ElectronicDeptPromotion\SOA\BPEL\CreditCardPromotion.bpel), 'WSDL File' (\\Mac\Home\Documents\JDeveloper\Thesis\ElectronicDeptPromotion\SOA\WSDLs\CreditCardPromotion.wsdl), and 'XSD File' (\\Mac\Home\Documents\JDeveloper\Thesis\ElectronicDeptPromotion\SOA\Schemas\CreditCardPromotion.xsd). A 'NEXT STEP' button is located at the bottom right of the form area. The footer of the application shows '2020 © Computer Engineering, Chulalongkorn University' and a 'Contact' link.

รูปที่ 5-5 หน้าจอนำเข้าข้อมูลตั้งต้นของกรณีศึกษาที่ 1

- 2) นำเข้าสู่ไฟล์บีเพलय่อย เครื่องมือตรวจสอบพบการเรียกใช้บีเพलय่อยจากการอ่านไฟล์บีเพเล
ตั้งต้น จึงแสดงแถบอัปโหลดไฟล์บีเพलय่อย sbpCheckCreditCard แสดงดังรูปที่ 5-6 จากนั้น
ผู้ทดสอบนำเข้าไฟล์ sbpCheckCreditCard.sbpel และกดปุ่ม Next Step

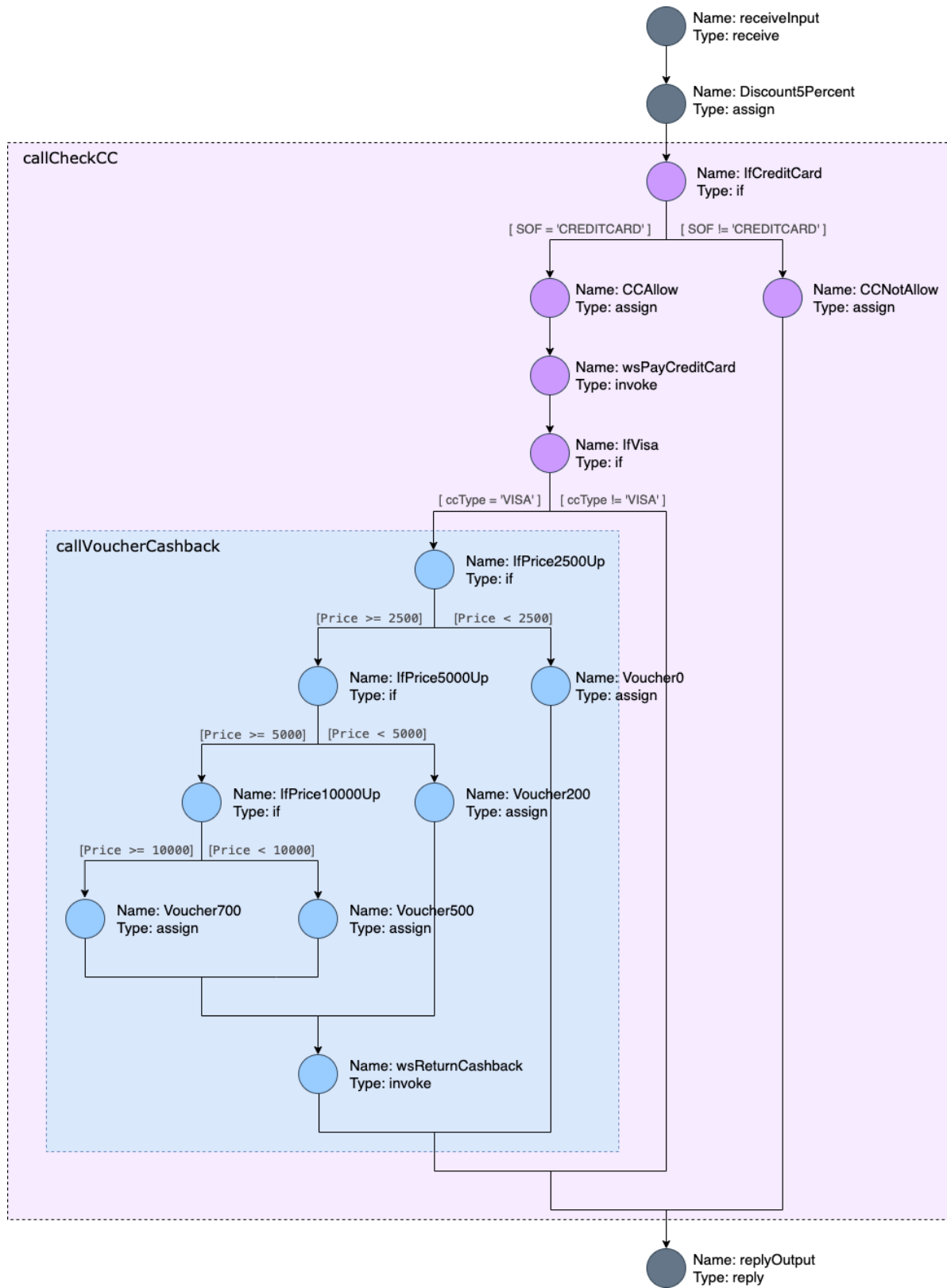


รูปที่ 5-6 หน้าจอนำเข้าไฟล์บีเพलय่อย sbpCheckCreditCard

เครื่องมือตรวจสอบพบการเรียกใช้งานบีเพलय่อยเพิ่มเติมจากการอ่านไฟล์บีเพलय่อยที่ผู้ทดสอบ
นำเข้า จึงแสดงแถบอัปโหลดไฟล์บีเพलय่อย sbpVoucherCashback ดังรูปที่ 5-7 เพื่อให้
ผู้ใช้งานนำเข้าไฟล์บีเพलय่อย sbpVoucherCashback.sbpel และกดปุ่ม Next Step โดยเมื่อ
เครื่องมือไม่พบการเรียกใช้งานบีเพलय่อยเพิ่มเติมแล้ว เครื่องมือจะนำผู้ทดสอบไปยังหน้าจอ
สร้างกรณีทดสอบ

รูปที่ 5-7 หน้าจอนำเข้าไฟล์บีเพलय่อย sbpVoucherCashback เพิ่มเติม

- 3) **สร้างกรณีทดสอบ** จากเส้นทางการไหลของไฟล์บีเพลดั้งเดิมและชุดไฟล์บีเพलय่อย ดังรูปที่ 5-1 ถึงรูปที่ 5-3 พบว่าไฟล์บีเพลดั้งเดิมมีการเรียกใช้งานบีเพलय่อย sbpCheckCreditCard และไฟล์บีเพलय่อย sbpCheckCreditCard ก็มีการเรียกใช้งานบีเพलय่อย CreditCardPromotion อีกต่อหนึ่ง โดยจากเส้นทางการไหลของบีเพลทั้ง 3 ไฟล์และความสัมพันธ์การเรียกใช้งานกันระหว่างดับเบิลยูเอส-บีเพล สามารถสร้างเป็นกราฟการเรียกเชิงสถิติ แสดงดังรูปที่ 5-8 โดยเครื่องมือสามารถสร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติได้ทั้งหมด 6 ทางเดินทดสอบ พร้อมทั้งสร้างกรณีทดสอบในแต่ละทางเดินทดสอบ ซึ่งรายละเอียดทางเดินทดสอบและกรณีทดสอบในแต่ละทางเดินทดสอบ แสดงดังรูปที่ 5-9



รูปที่ 5-8 กราฟการเรียกเชิงสถิติของกรณีศึกษาที่ 1

:: Test Case Generation for WS-BPEL from a Static Call Graph :: Home Generate Test Cases Project History

Import Initial Data Import BPEL Subprocesses **3** Generate Test Cases

TEST REPORT DOWNLOAD

Test Path	Test Path Description	Test Case ID
P1	receiveInput - Discount5Percent - IfCreditCard - CCAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - IfPrice5000Up - IfPrice10000Up - Voucher700 - wsReturnCashback - replyOutput	TC01
P2	receiveInput - Discount5Percent - IfCreditCard - CCAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - IfPrice5000Up - IfPrice10000Up - Voucher500 - wsReturnCashback - replyOutput	TC02
P3	receiveInput - Discount5Percent - IfCreditCard - CCAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - IfPrice5000Up - Voucher200 - wsReturnCashback - replyOutput	TC03
P4	receiveInput - Discount5Percent - IfCreditCard - CCAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - Voucher0 - wsReturnCashback - replyOutput	TC04
P5	receiveInput - Discount5Percent - IfCreditCard - CCAllow - wsPayCreditCard - IfVisa - replyOutput	TC05
P6	receiveInput - Discount5Percent - IfCreditCard - CCNotAllow - replyOutput	TC06

รูปที่ 5-9 กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 1

จากหน้าจอตั้งรูปที่ 5-9 ผู้ทดสอบกดปุ่ม “Download” และเลือก “Instrumented Source Code” เพื่อดาวน์โหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่ง โดยผู้ทดสอบนำรหัสต้นทางที่ถูกแทรกชุดคำสั่งไปติดตั้งบนพีแอลเอ็นจิน จากนั้นผู้ทดสอบดาวน์โหลดกรณีทดสอบทั้งหมด โดยกดปุ่ม “Download” และเลือก “All Test Cases” ตัวอย่างกรณีทดสอบที่ผู้ทดสอบดาวน์โหลดจากเครื่องมือ แสดงดังรูปที่ 5-10 และรูปที่ 5-11

AllTestCase.xlsx				
	A	B	C	D
1	Project	CreditCardPromotion		
2	Test Case ID	TC01		
3	Path Detail	receiveInput - Discount5Percent - IfCreditCard - CCAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - IfPrice5000Up - IfPrice10000Up - Voucher700 - wsReturnCashback - replyOutput		
4	Input Variable			
5	Variable Name	Data Type	Value	
6	Price	decimal	34690.42	
7	SOF	String	CREDITCARD	
8	CCNo	String	5059232372440152	
9	CCType	String	VISA	
10				

รูปที่ 5-10 กรณีทดสอบทั้งหมดของกรณีศึกษาที่ 1 ในรูปแบบเอกซ์เซล

```

1 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
2   <soap:Body>
3     <ns1:process xmlns:ns1="http://xmlns.oracle.com/Thesis/ElectronicDeptPromotion/CreditCardPromotion">
4       <ns1:Price>34690.42</ns1:Price>
5       <ns1:SOF>CREDITCARD</ns1:SOF>
6       <ns1:CCNo>5059232372440152</ns1:CCNo>
7       <ns1:CCType>VISA</ns1:CCType>
8     </ns1:process>
9   </soap:Body>
10 </soap:Envelope>

```

รูปที่ 5-11 กรณีทดสอบ TC01 ของกรณีศึกษาที่ 1 ในรูปแบบเอกซ์เอ็มแอล

- 4) ทดสอบดับเบิลยูเอส-บีเพล หลังจากผู้ทดสอบทำการติดตั้งรหัสต้นทางที่แทรกชุดคำสั่งบนบีเพลเอนจินแล้ว ผู้ทดสอบต้องการทดสอบทางเดินทดสอบ P1 P2 และ P3 จึงนำกรณีทดสอบ TC01 TC02 และ TC03 มาทดสอบดับเบิลยูเอส-บีเพลบนบีเพลเอนจิน หน้าจอการนำเข้ากรณีทดสอบ TC01 บนบีเพลเอนจิน แสดงดังรูปที่ 5-12 โดยผลลัพธ์การทดสอบดับเบิลยูเอส-บีเพลด้วยกรณีทดสอบ TC01 บนบีเพลเอนจิน แสดงดังรูปที่ 5-13 โดยในการทดสอบครั้งนี้ ผู้ทดสอบจะทำการทดสอบดับเบิลยูเอส-บีเพลกับกรณีทดสอบ TC01 TC02 และ TC03

Input Arguments

Tree View Enable Validation Load Payload TC01.xml

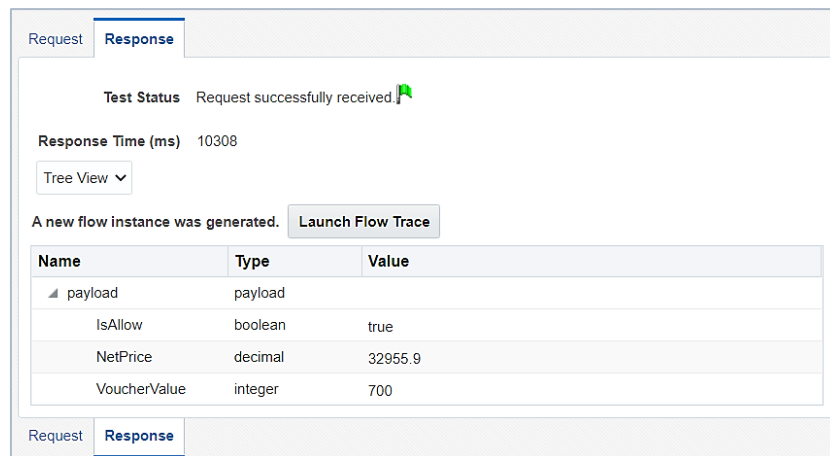
SOAP Body

View

Name	Type	Value
* payload	payload	
* Price	decimal	34690.42
* SOF	string	CREDITCARD
* CCNo	integer	5059232372440152
* CCType	string	VISA

Request Response

รูปที่ 5-12 การนำเข้ากรณีทดสอบ TC01 ของกรณีศึกษาที่ 1 บนบีเพลเอนจิน



Request **Response**

Test Status Request successfully received

Response Time (ms) 10308

Tree View ▾

A new flow instance was generated. **Launch Flow Trace**

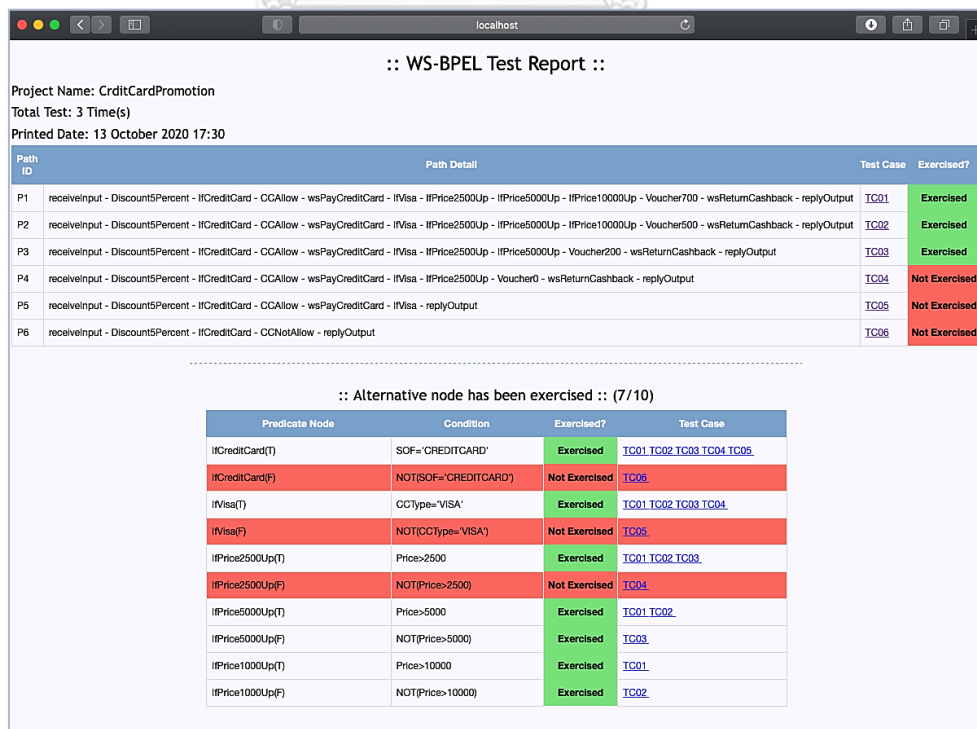
Name	Type	Value
payload	payload	
IsAllow	boolean	true
NetPrice	decimal	32955.9
VoucherValue	integer	700

Request **Response**

รูปที่ 5-13 ผลลัพธ์การทดสอบด้วยกรณีทดสอบ TC01 ของกรณีศึกษาที่ 1 บนบีเพลเอนจิน

5) เรียกดูรายงานการทดสอบดับเบิลยูเอส-บีเพล

หลังจากผู้ทดสอบทำการทดสอบดับเบิลยูเอส-บีเพลด้วยกรณีทดสอบ TC01 TC02 และ TC03 บนบีเพลเอนจินเรียบร้อยแล้ว ผู้ทดสอบกลับมาที่หน้าจอ ดังรูปที่ 5-9 และกดปุ่ม “Test Report” เพื่อเรียกดูรายงานการทดสอบ โดยรายละเอียดของรายงานการทดสอบดับเบิลยูเอส-บีเพลของกรณีศึกษาที่ 1 แสดงดังรูปที่ 5-14 และพบว่าทางเดินทดสอบ P1 P2 และ P3 ได้ถูกทดสอบแล้ว



localhost

:: WS-BPEL Test Report ::

Project Name: CrditCardPromotion
Total Test: 3 Time(s)
Printed Date: 13 October 2020 17:30

Path ID	Path Detail	Test Case	Exercised?
P1	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IVisa - IfPrice2500Up - IfPrice5000Up - IfPrice10000Up - Voucher700 - wsReturnCashback - replyOutput	TC01	Exercised
P2	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IVisa - IfPrice2500Up - IfPrice5000Up - IfPrice10000Up - Voucher500 - wsReturnCashback - replyOutput	TC02	Exercised
P3	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IVisa - IfPrice2500Up - IfPrice5000Up - Voucher200 - wsReturnCashback - replyOutput	TC03	Exercised
P4	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IVisa - IfPrice2500Up - Voucher0 - wsReturnCashback - replyOutput	TC04	Not Exercised
P5	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IVisa - replyOutput	TC05	Not Exercised
P6	receiveInput - Discount5Percent - IfCreditCard - CCNotAllow - replyOutput	TC06	Not Exercised

.....

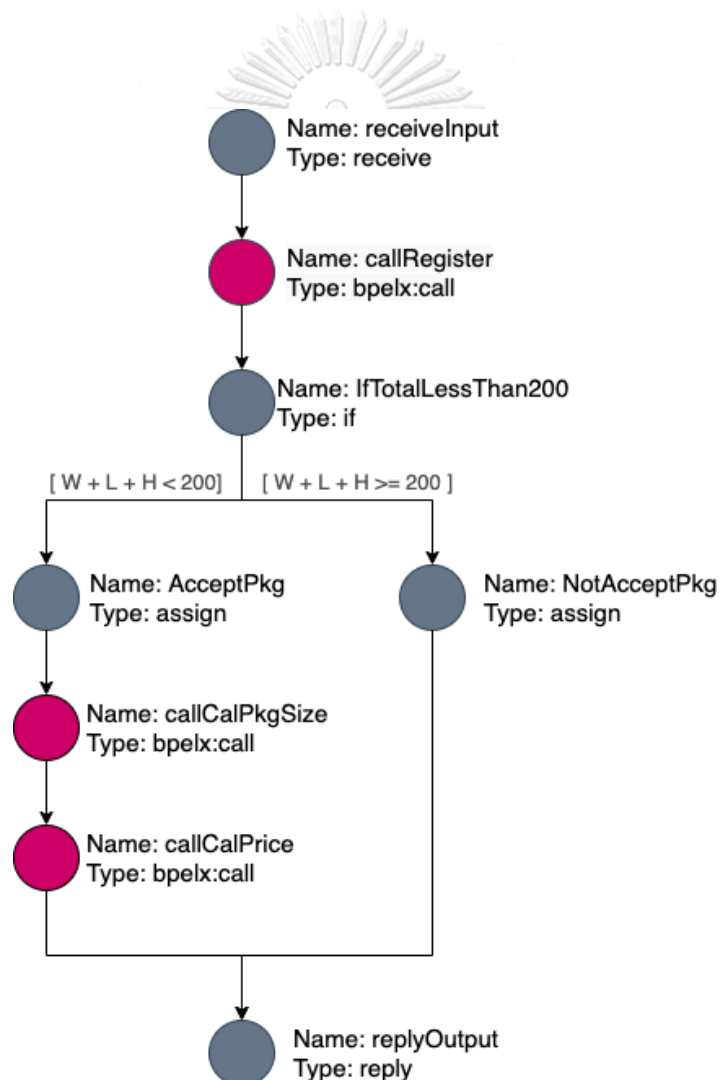
:: Alternative node has been exercised :: (7/10)

Predicate Node	Condition	Exercised?	Test Case
IfCreditCard(T)	SOF='CREDITCARD'	Exercised	TC01 TC02 TC03 TC04 TC05
IfCreditCard(F)	NOT(SOF='CREDITCARD')	Not Exercised	TC06
IVisa(T)	CCType='VISA'	Exercised	TC01 TC02 TC03 TC04
IVisa(F)	NOT(CCType='VISA')	Not Exercised	TC05
IfPrice2500Up(T)	Price>2500	Exercised	TC01 TC02 TC03
IfPrice2500Up(F)	NOT(Price>2500)	Not Exercised	TC04
IfPrice5000Up(T)	Price>5000	Exercised	TC01 TC02
IfPrice5000Up(F)	NOT(Price>5000)	Exercised	TC03
IfPrice10000Up(T)	Price>10000	Exercised	TC01
IfPrice10000Up(F)	NOT(Price>10000)	Exercised	TC02

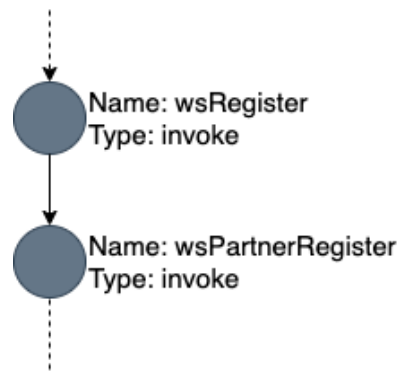
รูปที่ 5-14 รายงานการทดสอบดับเบิลยูเอส-บีเพลของกรณีศึกษาที่ 1

5.1.2 กรณีศึกษาที่ 2 ระบบร้านขนส่งพัสดุ

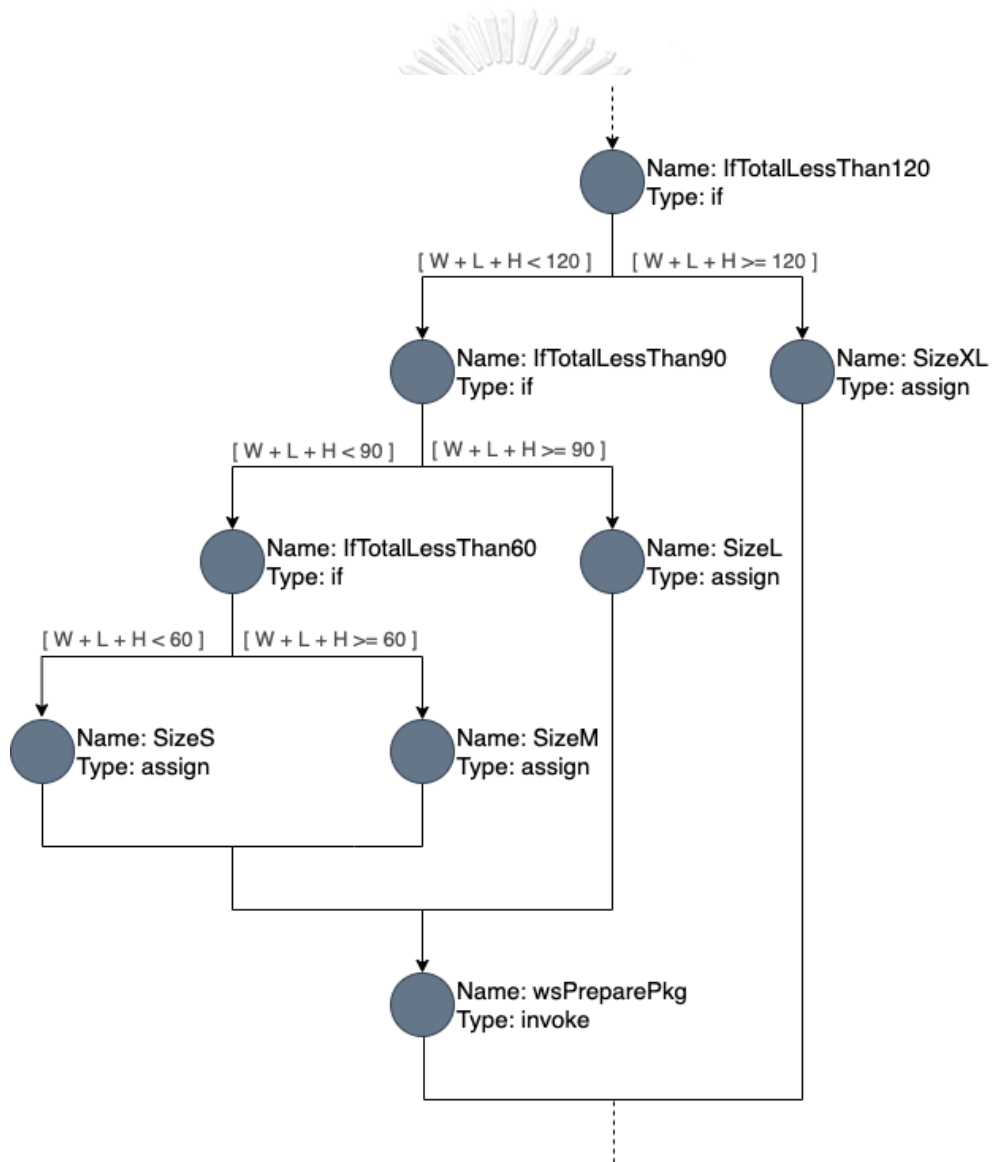
ระบบร้านขนส่งพัสดุ มีขั้นตอนเริ่มจากลูกค้าต้องการส่งสินค้าไปยังปลายทางจึงมาติดต่อที่ร้านขนส่งพัสดุ โดยลูกค้าจะต้องระบุขนาดและน้ำหนักของสินค้า พร้อมระบุว่าต้องการจัดส่งไปยังปลายทางที่อยู่ในจังหวัดเดียวกันหรือไม่ เพื่อที่จะได้คำนวณขนาดกล่องบรรจุสินค้า และค่าขนส่งที่เป็นไปตามเงื่อนไขของร้านขนส่งพัสดุ ซึ่งกระบวนการดับเบิลยูเอส-บีเพลของกรณีศึกษานี้ แสดงดังรูปที่ 5-15 ถึงรูปที่ 5-19 โดยตัวแปรนำเข้าและตัวแปรส่งออกกระบวนการดับเบิลยูเอส-บีเพลที่อยู่ในไฟล์เอกซ์เอสดี แสดงดังรูปที่ 5-20



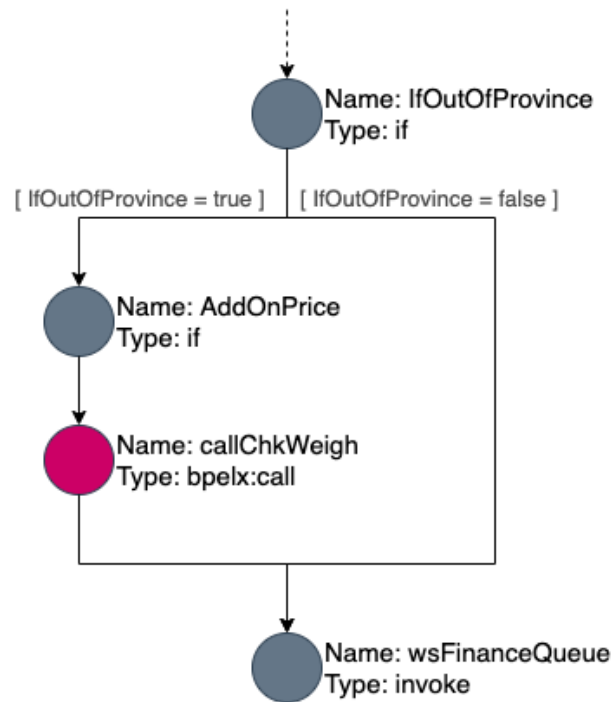
รูปที่ 5-15 กระบวนการดับเบิลยูเอส-บีเพลในไฟล์บีเพลตั้งต้น TransExpress



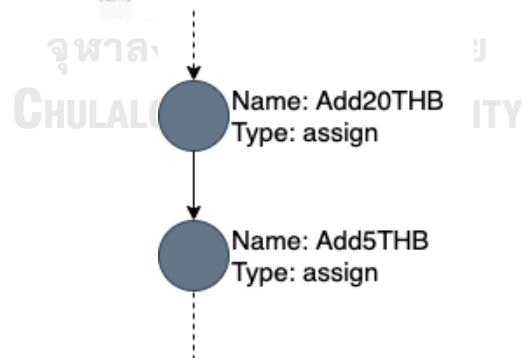
รูปที่ 5-16 กระบวนการดับเบิลยูเอส-บีเพลในไฟล์บีเพलय่อย sbpRegister



รูปที่ 5-17 กระบวนการดับเบิลยูเอส-บีเพลในไฟล์บีเพलय่อย sbpCalPkgSize



รูปที่ 5-18 กระบวนการดับเบิลยูเอส-บีเพลในไฟล์บีเพलय่อย sbpCalPrice



รูปที่ 5-19 กระบวนการดับเบิลยูเอส-บีเพลในไฟล์บีเพलय่อย sbpChkWeigh

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <schema attributeFormDefault="unqualified" elementFormDefault="qualified"
3  <element name="process">
4  <complexType>
5  <sequence>
6  <element name="Weigh">
7  <simpleType>
8  <restriction base="decimal">
9  <totalDigits value="4" />
10 <fractionDigits value="2" />
11 </restriction>
12 </simpleType>
13 </element>
14 <element name="W" type="integer" />
15 <element name="H" type="integer" />
16 <element name="L" type="integer" />
17 <element name="OutOfProvince" type="boolean" />
18 </sequence>
19 </complexType>
20 </element>
21 <element name="processResponse">
22 <complexType>
23 <sequence>
24 <element name="IsAccept" type="boolean" />
25 <element name="PackageSize" type="string" />
26 <element name="TotalPrice" type="integer" />
27 </sequence>
28 </complexType>
29 </element>
30 </schema>

```

รูปที่ 5-20 ตัวแปรนำเข้าและตัวแปรส่งออกกระบวนกรดับเบิลยูเอส-บีเพลของกรณีศึกษาที่ 2

การทดสอบกรณีศึกษาที่ 2 ผู้ทดสอบจะต้องทำตามขั้นตอนการทดสอบเครื่องมือ โดยมีขั้นตอนการทดสอบ ดังนี้

1) นำเข้าข้อมูลตั้งต้น ผู้ทดสอบนำเข้าข้อมูลตั้งต้นผ่านหน้าจอของเครื่องมือ แสดงดังรูปที่ 5-21

โดยข้อมูลตั้งต้นที่จะนำเข้าประกอบไปด้วยข้อมูล ดังนี้

- ชื่อโครงการดับเบิลยูเอส-บีเพล TransExpress
- ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล Composite.xml
- ไฟล์บีเพลตั้งต้น TransExpress.bpel
- ไฟล์ดับเบิลยูเอสดีแอล TransExpress.wsdl
- ไฟล์เอกซ์เอสดี TransExpress.xsd

เมื่อผู้ทดสอบนำเข้าข้อมูลตั้งต้นเสร็จเรียบร้อยแล้ว ผู้ทดสอบกดปุ่ม Next Step โดยเครื่องมือจะนำผู้ทดสอบไปยังหน้าจอนำเข้าชุดไฟล์บีเพลย่อย

:: Test Case Generation for WS-BPEL from a Static Call Graph :: [Home](#) [Generate Test Cases](#) [Project History](#)

1 Import Initial Data 2 Import BPEL Subprocesses 3 Generate Test Cases

Project Name
CreditCardPromotion

Composite File
\\Mac\Home\Documents\JDeveloper\Thesis\ElectronicDeptPromotion\SOA\composite.xml [Browse](#)
file extension: .xsd only

BPEL File
\\Mac\Home\Documents\JDeveloper\Thesis\ElectronicDeptPromotion\SOA\BPEL\TransExpress.bpel [Browse](#)
file extension: .bpel only

WSDL File
\\Mac\Home\Documents\JDeveloper\Thesis\ElectronicDeptPromotion\SOA\WSDLs\TransExpress.wsdl [Browse](#)
file extension: .wsdl only

XSD File
\\Mac\Home\Documents\JDeveloper\Thesis\ElectronicDeptPromotion\SOA\Schemas\TransExpress.xsd [Browse](#)
file extension: .xml only

[NEXT STEP](#)

2020 © Computer Engineering, Chulalongkorn University [Contact](#)

รูปที่ 5-21 หน้าจอนำเข้าข้อมูลตั้งต้นของกรณีศึกษาที่ 2

- 2) นำเข้าชุดไฟล์บีเพलय่อย เครื่องมือตรวจสอบพบการเรียกใช้บีเพलय่อยจากการอ่านไฟล์บีเพเลที่ตั้งต้น จึงแสดงแถบอัปโหลดไฟล์บีเพलय่อย sbpCallRegister sbpCalPkgSize และ sbpCalPrice แสดงดังรูปที่ 5-22 จากนั้นผู้ทดสอบนำเข้าไฟล์ sbpCallRegister.sbpel sbpCalPkgSize.sbpel และ sbpCalPrice.sbpel และกดปุ่ม “Next Step”

เครื่องมือตรวจสอบพบการเรียกใช้งานบีเพलय่อยเพิ่มเติมจากการอ่านไฟล์บีเพलय่อยที่ผู้ทดสอบนำเข้า จึงแสดงแถบอัปโหลดไฟล์บีเพलय่อย sbpChkWeigh ดังรูปที่ 5-23 เพื่อให้ผู้ใช้งานนำเข้าไฟล์บีเพलय่อย sbpVoucherCashback.sbpel และกดปุ่ม “Next Step” โดยเมื่อเครื่องมือไม่พบการเรียกใช้งานบีเพलय่อยเพิ่มเติมแล้ว เครื่องมือจะนำผู้ทดสอบไปยังหน้าจอสร้างกรณีทดสอบ

:: Test Case Generation for WS-BPEL from a Static Call Graph :: [Home](#) [Generate Test Cases](#) [Project History](#)

1 Import Initial Data 2 Import BPEL Subprocesses 3 Generate Test Cases

sbpRegister.sbpel [Browse](#)

sbpCalPkgSize.sbpel [Browse](#)

sbpCalPrice.sbpel [Browse](#)

[NEXT STEP](#)

2020 © Computer Engineering, Chulalongkorn University [Contact](#)

รูปที่ 5-22 หน้าจอนำเข้าไฟล์บีเพलय่อยของกรณีศึกษาที่ 2

:: Test Case Generation for WS-BPEL from a Static Call Graph :: [Home](#) [Generate Test Cases](#) [Project History](#)

1 Import Initial Data 2 Import BPEL Subprocesses 3 Generate Test Cases

sbpRegister.sbpel [Browse](#)

sbpCalPkgSize.sbpel [Browse](#)

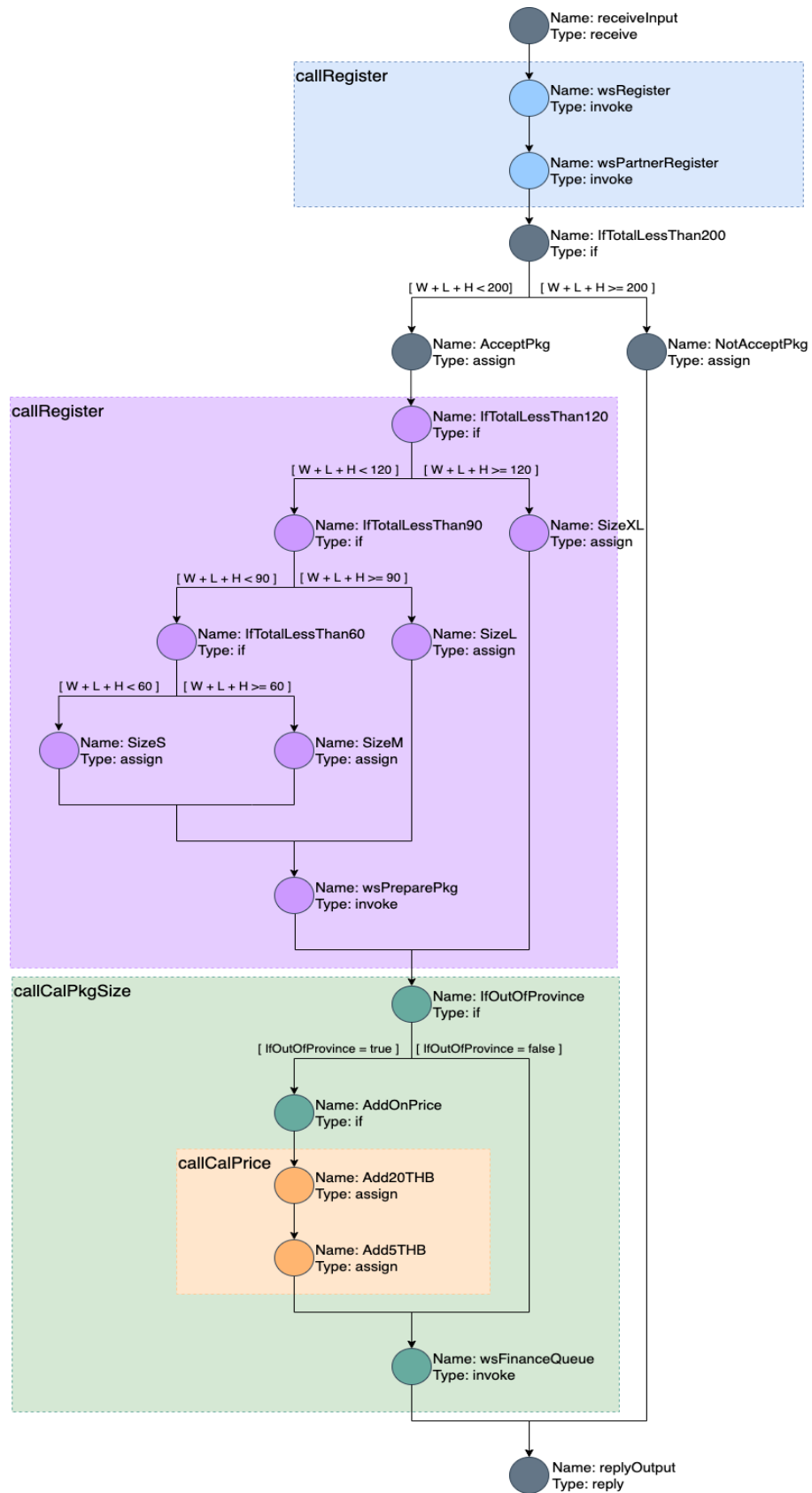
sbpCalPrice.sbpel [Browse](#)

sbpChkWeigh.sbpel [Browse](#)

[NEXT STEP](#)

2020 © Computer Engineering, Chulalongkorn University [Contact](#)

รูปที่ 5-23 หน้าจอนำเข้าไฟล์บีเพलय่อยเพิ่มเติมของกรณีศึกษาที่ 2



รูปที่ 5-24 กราฟการเรียกเชิงสถิติของกรณีศึกษาที่ 2

- 3) **สร้างกรณีทดสอบ** จากเส้นทางการไหลของบีเพลตั้งต้นและชุดไฟล์บีเพลย่อย ดังรูปที่ 5-15 ถึงรูปที่ 5-19 พบว่าไฟล์บีเพลตั้งต้นมีการเรียกใช้งานบีเพลย่อย sbpRegister sbpCalPkgSize และ sbpCallPrice โดยขณะที่ไฟล์บีเพลย่อย sbpCalPrice ก็มีการเรียกใช้งานบีเพลย่อย sbpChkWeigh อีกต่อหนึ่ง โดยจากเส้นทางการไหลของบีเพลทั้ง 5 ไฟล์และความสัมพันธ์การเรียกใช้งานกันระหว่างดับเบิลยูเอส-บีเพล สามารถสร้างเป็นกราฟการเรียกเชิงสถิติ แสดงดังรูปที่ 5-24 โดยเครื่องมือสามารถสร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติได้ทั้งหมด 9 ทางเดินทดสอบ พร้อมทั้งสร้างกรณีทดสอบในแต่ละทางเดินทดสอบ ซึ่งรายละเอียดทางเดินทดสอบและกรณีทดสอบในแต่ละทางเดินทดสอบ แสดงดังรูปที่ 5-25

จากหน้าจอรูปที่ 5-25 ผู้ทดสอบกดปุ่ม “Download” และเลือก “Instrumented Source Code” เพื่อดาวน์โหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่ง โดยผู้ทดสอบนำรหัสต้นทางที่ถูกแทรกชุดคำสั่งไปติดตั้งบนบีเพลเอนจิน จากนั้นผู้ทดสอบดาวน์โหลดกรณีทดสอบทั้งหมด โดยกดปุ่ม “Download” และเลือก “All Test Cases” ตัวอย่างกรณีทดสอบที่ดาวน์โหลดจากเครื่องมือ แสดงดังรูปที่ 5-26 และรูปที่ 5-27

- 4) **ทดสอบดับเบิลยูเอส-บีเพล** หลังจากผู้ทดสอบทำการติดตั้งรหัสต้นทางที่แทรกชุดคำสั่งบนบีเพลเอนจินแล้ว ผู้ทดสอบต้องการทดสอบทางเดินทดสอบ P6 P7 P8 และ P9 จึงนำกรณีทดสอบ TC06 TC07 TC08 และ TC09 มาทดสอบดับเบิลยูเอส-บีเพลบนบีเพลเอนจินด้วยวิธีการนำเข้ากรณีทดสอบที่แสดง ดังรูปที่ 5-28 โดยผลลัพธ์การทดสอบดับเบิลยูเอส-บีเพลด้วยกรณีทดสอบ TC06 บนบีเพลเอนจิน แสดงดังรูปที่ 5-29 โดยในการทดสอบครั้งนี้ ผู้ทดสอบจะทำการทดสอบดับเบิลยูเอส-บีเพลกับกรณีทดสอบ TC06 TC07 TC08 และ TC09

:: Test Case Generation for WS-BPEL from a Static Call Graph :: [Home](#) [Generate Test Cases](#) [Project History](#)

✓
 Import Initial Data

✓
 Import BPEL Subprocesses

3
 Generate Test Cases

[TEST REPORT](#) [DOWNLOAD](#)

Test Path	Test Path Description	Test Case ID
P1	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeS - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	TC01
P2	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeS - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	TC02
P3	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeM - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	TC03
P4	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeM - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	TC04
P5	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - SizeL - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	TC05
P6	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - SizeL - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	TC06
P7	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - SizeXL - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	TC07
P8	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - SizeXL - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	TC08
P9	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - NotAcceptPkg - replyOutput	TC09

รูปที่ 5-25 กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 2

จุฬาลงกรณ์มหาวิทยาลัย

AllTestCase.xlsx			
	A	B	C
1	Project	TransExpress	
2	Test Case ID	TC06	
3	Path Detail	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - SizeL - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	
4	Input Variable		
5	Variable Name	Data Type	Value
6	Weigh	decimal	10.81
7	W	integer	32
8	H	integer	11
9	L	integer	59
10	OutOfProvince	boolean	false
11			

TC01 TC02 TC03 TC04 TC05 TC06 TC07 TC08 TC09 +

รูปที่ 5-26 กรณีทดสอบทั้งหมดของกรณีศึกษาที่ 2 ในรูปแบบเอกซ์เซล

```

TC06.xml > ...
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
3 <soap:Body>
4 <ns1:process xmlns:ns1="http://xmlns.oracle.com/Thesis/ElectronicDeptPromotion/TransExpress">
5 <ns1:Weigh>10.81</ns1:Weigh>
6 <ns1:W>32</ns1:W>
7 <ns1:H>11</ns1:H>
8 <ns1:L>59</ns1:L>
9 <ns1:OutOfProvince>false</ns1:OutOfProvince>
10 </ns1:process>
11 </soap:Body>
12 </soap:Envelope>

```

รูปที่ 5-27 กรณีทดสอบ TC06 ของกรณีศึกษาที่ 2 ในรูปแบบเอกซ์เอ็มแอล

Input Arguments

Tree View ▾ Enable Validation Load Payload TC06.xml

SOAP Body

View ▾

Name	Type	Value
* payload	payload	
* Weigh	decimal	10.81
* W	integer	32
* H	integer	11
* L	integer	59
* OutOfProvince	boolean	false

Request Response

รูปที่ 5-28 การนำเข้ากรณีทดสอบ TC06 ของกรณีศึกษาที่ 2 บนปีเพลเอนจิน

จุฬาลงกรณ์มหาวิทยาลัย

Request **Response**

Test Status Request successfully received

Response Time (ms) 6096

Tree View ▾

A new flow instance was generated.

Name	Type	Value
payload	payload	
IsAccept	boolean	true
PackageSize	string	L
TotalPrice	integer	185

Request **Response**

รูปที่ 5-29 ผลลัพธ์การทดสอบด้วยกรณีทดสอบ TC06 ของกรณีศึกษาที่ 2 บนปีเพลเอนจิน

5) เรียกดูรายงานการทดสอบดับเบิลยูเอส-บีเพล

หลังจากผู้ทดสอบทำการทดสอบดับเบิลยูเอส-บีเพลด้วยกรณีทดสอบ TC06 TC07 TC08 และ TC09 บนบีเพลเอนจินเรียบร้อยแล้ว ผู้ทดสอบกลับมาที่หน้าจอ ดังรูปที่ 5-25 และกดปุ่ม “Test Report” เพื่อเรียกดูรายงานการทดสอบ โดยรายละเอียดของรายงานการทดสอบดับเบิลยูเอส-บีเพลของกรณีศึกษาที่ 2 แสดงดังรูปที่ 5-30 และพบว่าทางเดินทดสอบ P6 P7 P8 และ P9 ได้ถูกทดสอบแล้ว

:: WS-BPEL Test Report ::

Project Name: CrditCardPromotion
 Total Test: 3 Time(s)
 Printed Date: 13 October 2020 17:30

Path ID	Path Detail	Test Case	Exercised?
P1	receiveInput - Discount5Percent - IFCreditCard - CCAAllow - wsPayCreditCard - IFVisa - IFPrice2500Up - IFPrice5000Up - IFPrice10000Up - Voucher700 - wsReturnCashback - replyOutput	TC01	Exercised
P2	receiveInput - Discount5Percent - IFCreditCard - CCAAllow - wsPayCreditCard - IFVisa - IFPrice2500Up - IFPrice5000Up - IFPrice10000Up - Voucher500 - wsReturnCashback - replyOutput	TC02	Exercised
P3	receiveInput - Discount5Percent - IFCreditCard - CCAAllow - wsPayCreditCard - IFVisa - IFPrice2500Up - IFPrice5000Up - Voucher200 - wsReturnCashback - replyOutput	TC03	Exercised
P4	receiveInput - Discount5Percent - IFCreditCard - CCAAllow - wsPayCreditCard - IFVisa - IFPrice2500Up - Voucher0 - wsReturnCashback - replyOutput	TC04	Not Exercised
P5	receiveInput - Discount5Percent - IFCreditCard - CCAAllow - wsPayCreditCard - IFVisa - replyOutput	TC05	Not Exercised
P6	receiveInput - Discount5Percent - IFCreditCard - CCNotAllow - replyOutput	TC06	Not Exercised

:: Alternative node has been exercised :: (7/10)

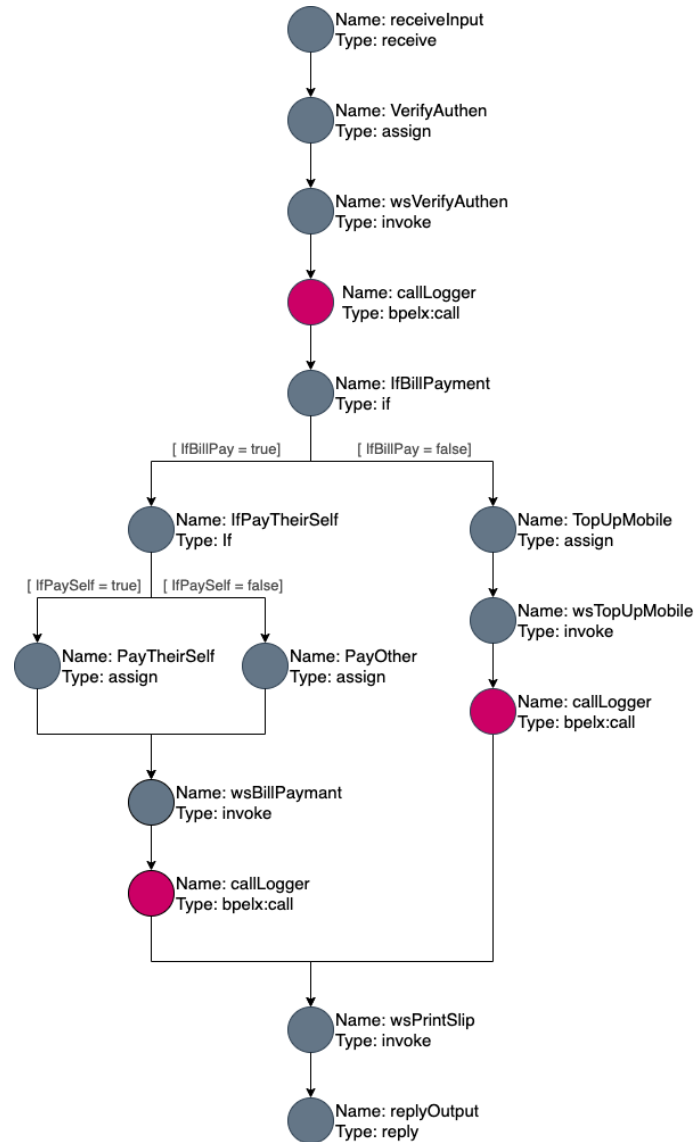
Predicate Node	Condition	Exercised?	Test Case
IFCreditCard(T)	SOF='CREDITCARD'	Exercised	TC01 TC02 TC03 TC04 TC05
IFCreditCard(F)	NOT(SOF='CREDITCARD')	Not Exercised	TC06
IFVisa(T)	CCType='VISA'	Exercised	TC01 TC02 TC03 TC04
IFVisa(F)	NOT(CCType='VISA')	Not Exercised	TC05
IFPrice2500Up(T)	Price>2500	Exercised	TC01 TC02 TC03
IFPrice2500Up(F)	NOT(Price>2500)	Not Exercised	TC04
IFPrice5000Up(T)	Price>5000	Exercised	TC01 TC02
IFPrice5000Up(F)	NOT(Price>5000)	Exercised	TC03
IFPrice10000Up(T)	Price>10000	Exercised	TC01
IFPrice10000Up(F)	NOT(Price>10000)	Exercised	TC02

รูปที่ 5-30 รายงานการทดสอบดับเบิลยูเอส-บีเพลของกรณีศึกษาที่ 2

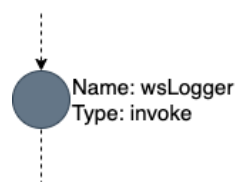
5.1.3 กรณีศึกษาที่ 3 ระบบติดตามการทำงานตู้คิออส

ตู้คิออสเป็นตู้อัตโนมัติที่บริการรับชำระและเติมเงินค่าโทรศัพท์มือถือค่ายหนึ่งในประเทศไทย โดยระบบติดตามการทำงานจะถูกติดตั้งอยู่บนตู้คิออส เพื่อติดตามการทำงานและเก็บข้อมูลการใช้บริการของผู้ใช้งาน เพื่อนำไปพัฒนาการบริการให้ดียิ่งขึ้น มีขั้นตอนเริ่มจากผู้ใช้งานเดินทางมาใช้บริการที่ตู้คิออส โดยผู้ใช้งานจะต้องเสียบบัตรประชาชนเพื่อยืนยันตัวตนผ่านตู้คิออส เลือกบริการ

พร้อมทั้งระบุจำนวนเงินที่ทำธุรกรรม ซึ่งกระบวนการดับเบิลยูเอส-บีเพลของกรณีศึกษานี้ แสดงดังรูปที่ 5-31 และรูปที่ 5-32 โดยตัวแปรนำเข้าและตัวแปรส่งออกกระบวนการดับเบิลยูเอส-บีเพลที่อยู่ในไฟล์เอกซ์เอสดี แสดงดังรูปที่ 5-33



รูปที่ 5-31 กระบวนการดับเบิลยูเอส-บีเพลในไฟล์บีเพลตั้งต้น KioskTracking



รูปที่ 5-32 กระบวนการย่อยภายใน Logger ในไฟล์บีเพลตั้งต้น KioskTracking

```

3      <element name="process">
4          <complexType>
5              <sequence>
6                  <element name="IDCard">
7                      <simpleType>
8                          <restriction base="integer">
9                              <totalDigits value="13" />
10                             <minInclusive value="1000000000000" />
11                             <maxInclusive value="9999999999999" />
12                         </restriction>
13                     </simpleType>
14                 </element>
15                 <element name="MobileNo">
16                     <simpleType>
17                         <restriction base="integer">
18                             <totalDigits value="10" />
19                             <minInclusive value="1000000000" />
20                             <maxInclusive value="9999999999" />
21                         </restriction>
22                     </simpleType>
23                 </element>
24                 <element name="Price">
25                     <simpleType>
26                         <restriction base="decimal">
27                             <totalDigits value="7" />
28                             <fractionDigits value="2" />
29                         </restriction>
30                     </simpleType>
31                 </element>
32             </sequence>
33         </complexType>
34     </element>
35     <element name="processResponse">
36         <complexType>
37             <sequence>
38                 <element name="logStatus" type="string" />
39                 <element name="result" type="boolean" />
40             </sequence>
41         </complexType>
42     </element>

```

รูปที่ 5-33 ตัวแปรนำเข้าและตัวแปรส่งออกกระบวนการดับเบิลยูเอส-บีเพลของกรณีศึกษาที่ 3

การทดสอบกรณีศึกษาที่ 3 ผู้ทดสอบจะต้องทำตามขั้นตอนการทดสอบเครื่องมือ โดยมีขั้นตอนการทดสอบ ดังนี้

- 1) **นำเข้าข้อมูลตั้งต้น** ผู้ทดสอบนำเข้าข้อมูลตั้งต้นผ่านหน้าจอของเครื่องมือ แสดงดังรูปที่ 5-34 โดยข้อมูลตั้งต้นที่จะนำเข้าประกอบไปด้วยข้อมูล ดังนี้
 - ชื่อโครงการดับเบิลยูเอส-บีเพล KioskTracking
 - ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล Composite.xml

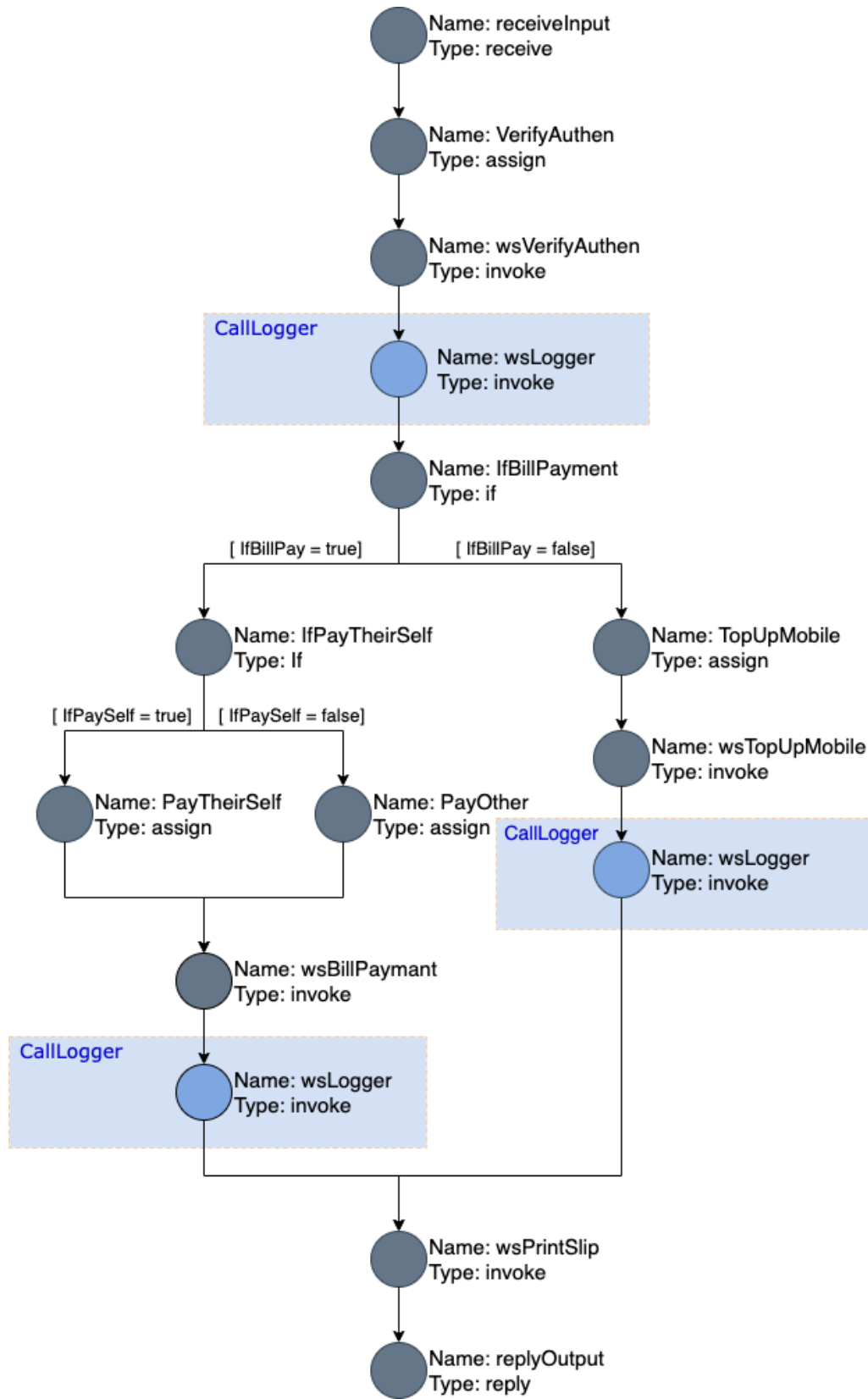
- ไฟล์บีเพลดั้งเดิม KioskTracking.bpel
- ไฟล์ดับเบิลยูเอสดีแอล KioskTracking.wsdl
- ไฟล์เอกซ์เอสดี KioskTracking.xsd

เมื่อผู้ทดสอบนำข้อมูลตั้งต้นเสร็จเรียบร้อยแล้ว ผู้ทดสอบกดปุ่ม “Next Step” โดยเครื่องมือจะนำผู้ทดสอบไปยังหน้าจอแนะนำเข้าสู่ชุดไฟล์บีเพลด้อย

The screenshot displays a web application interface for generating test cases. At the top, there is a navigation bar with the title "Test Case Generation for WS-BPEL from a Static Call Graph" and links for "Home", "Generate Test Cases", and "Project History". Below the navigation bar is a progress indicator with three steps: "1 Import Initial Data", "2 Import BPEL Subprocesses", and "3 Generate Test Cases". The first step is highlighted. The main content area contains several input fields with "Browse" buttons for selecting files: "Project Name" (filled with "CreditCardPromotion"), "Composite File" (filled with a local path), "BPEL File" (filled with a local path), "WSDL File" (filled with a local path), and "XSD File" (filled with a local path). A "NEXT STEP" button is positioned at the bottom right of the form. The footer of the page includes the copyright notice "2020 © Computer Engineering, Chulalongkorn University" and a "Contact" link.

รูปที่ 5-34 หน้าจอแนะนำเข้าสู่ข้อมูลตั้งต้นของกรณีศึกษาที่ 3

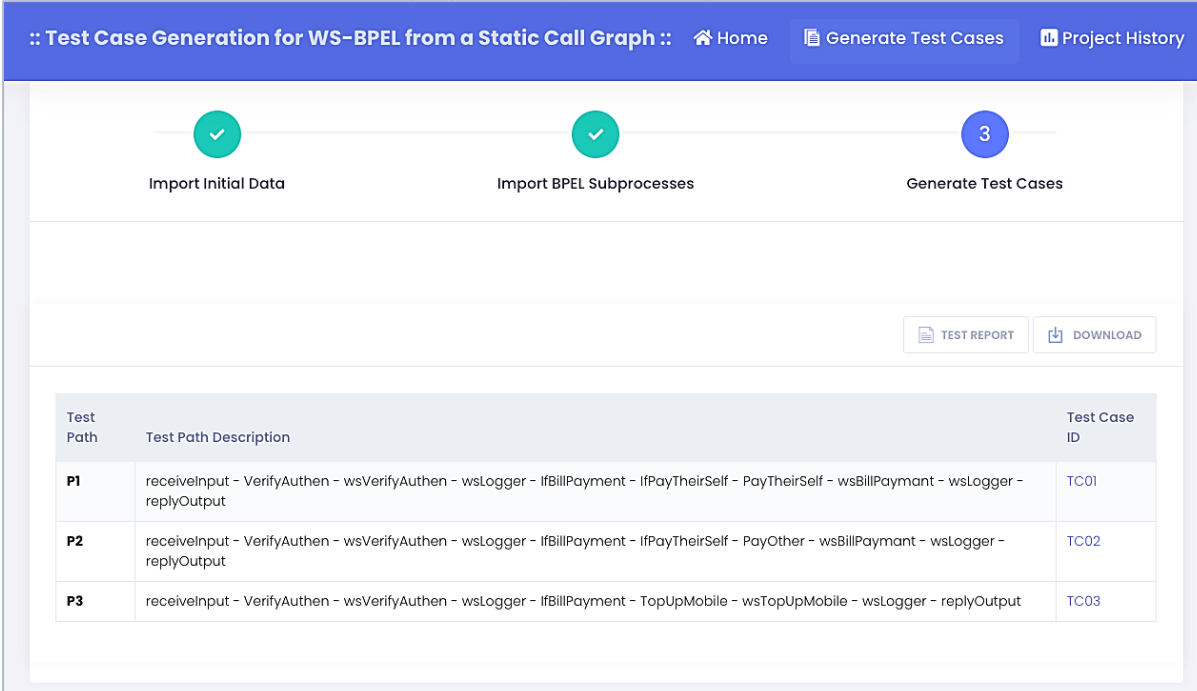
- นำเข้าสู่ชุดไฟล์บีเพลด้อย เครื่องมือตรวจสอบพบการเรียกใช้บีเพลด้อยที่เป็นกระบวนการย่อยภายใน และปรากฏกระบวนการย่อยภายในในไฟล์บีเพลดั้งเดิมแล้ว เครื่องมือจึงนำผู้ทดสอบไปยังหน้าจอสร้างกรณีทดสอบ



รูปที่ 5-35 กราฟการเรียกเชิงสถิติของกรณีศึกษาที่ 3

3) **สร้างกรณีทดสอบ** จากเส้นทางการไหลของไฟล์บีเพลที่ตั้งต้นและกระบวนการย่อยภายใน ดังรูปที่ 5-31 และรูปที่ 5-32 พบว่าไฟล์บีเพลที่ตั้งต้นมีการเรียกใช้กระบวนการย่อยภายใน Logger โดยจากเส้นทางการไหลของบีเพลและความสัมพันธ์การเรียกใช้งานระหว่างดับเบิลยูเอส-บีเพลสามารถสร้างเป็นกราฟการเรียกเชิงสถิติ แสดงดังรูปที่ 5-35 โดยเครื่องมือสามารถสร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติได้ทั้งหมด 3 ทางเดินทดสอบ พร้อมทั้งสร้างกรณีทดสอบในแต่ละทางเดินทดสอบ ซึ่งรายละเอียดทางเดินทดสอบและกรณีทดสอบในแต่ละทางเดินทดสอบ แสดงดังรูปที่ 5-36

จากหน้าจอ ดังรูปที่ 5-36 ผู้ทดสอบกดปุ่ม “Download” และเลือก “Instrumented Files” เพื่อดาวน์โหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่ง โดยผู้ทดสอบนำรหัสต้นทางที่ถูกแทรกชุดคำสั่งไปติดตั้งบนบีเพลออนไลน์ จากนั้นผู้ทดสอบดาวน์โหลดกรณีทดสอบทั้งหมด โดยกดปุ่ม “Download” และเลือก “All Test Cases” ตัวอย่างกรณีทดสอบที่ผู้ทดสอบดาวน์โหลดจากเครื่องมือ แสดงดังรูปที่ 5-37 และรูปที่ 5-38



:: Test Case Generation for WS-BPEL from a Static Call Graph :: Home Generate Test Cases Project History

Import Initial Data Import BPEL Subprocesses **3** Generate Test Cases

TEST REPORT DOWNLOAD

Test Path	Test Path Description	Test Case ID
P1	receiveInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - IfPayTheirSelf - PayTheirSelf - wsBillPaymant - wsLogger - replyOutput	TC01
P2	receiveInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - IfPayTheirSelf - PayOther - wsBillPaymant - wsLogger - replyOutput	TC02
P3	receiveInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - TopUpMobile - wsTopUpMobile - wsLogger - replyOutput	TC03

รูปที่ 5-36 กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 3

	A	B	C
1	Project	KioskTracking	
2	Test Case ID	TC01	
3	Path Detail	receiveInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - IfPayTheirSelf - PayTheirSelf - wsBillPayment - wsLogger - replyOutput	
4	Input Variable		
5	Variable Name	Data Type	Value
6	IDCard	integer	5678940987638
7	MobileNo	integer	7343206754
8	Price	decimal	1054.85
9			

รูปที่ 5-37 กรณีทดสอบทั้งหมดของกรณีศึกษาที่ 3 ในรูปแบบเอกซ์เซล

```

TC01.xml > ...
1 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
2   <soap:Body>
3     <ns1:process xmlns:ns1="http://xmlns.oracle.com/Thesis/ElectronicDeptPromotion/KioskTracking">
4       <ns1:IDCard>5678940987638</ns1:IDCard>
5       <ns1:MobileNo>7343206754</ns1:MobileNo>
6       <ns1:Price>1054.85</ns1:Price>
7     </ns1:process>
8   </soap:Body>
9 </soap:Envelope>

```

รูปที่ 5-38 กรณีทดสอบ TC01 ของกรณีศึกษาที่ 3 ในรูปแบบเอกซ์เอ็มแอล

- 4) ทดสอบดับเบิลยูเอส-บีเพล หลังจากผู้ทดสอบทำการติดตั้งรหัสต้นทางที่แทรกชุดคำสั่งบนบีเพลเอนจินแล้ว ผู้ทดสอบต้องการทดสอบทางเดินทดสอบ P1 P2 และ P3 จึงนำกรณีทดสอบ TC01 TC02 และ TC03 มาทดสอบดับเบิลยูเอส-บีเพลบนบีเพลเอนจินด้วยวิธีการนำเข้ากรณีทดสอบที่แสดง ดังรูปที่ 5-38 โดยผลลัพธ์การทดสอบดับเบิลยูเอส-บีเพลด้วยกรณีทดสอบ TC01 บนบีเพลเอนจิน แสดงดังรูปที่ 5-39 โดยในการทดสอบครั้งนี้ ผู้ทดสอบจะทำการทดสอบดับเบิลยูเอส-บีเพลกับกรณีทดสอบ TC01 TC02 และ TC03

Input Arguments

Tree View ▾ Enable Validation Load Payload **TC01.xml** **Update...** **Save Payload**

SOAP Body

View ▾ **Detach**

Name	Type	Value
▲ * payload	payload	
* IDCard	integer	5678940987638
* MobileNo	integer	7343206754
* Price	decimal	1054.85

Request Response

รูปที่ 5-39 การนำเข้ากรณีทดสอบ TC01 ของกรณีศึกษาที่ 3 บนพีเพิลเอนจิน

Request **Response**

Test Status Request successfully received.

Response Time (ms) 3492

Tree View ▾

A new flow instance was generated. **Launch Flow Trace**

Name	Type	Value
▲ payload	payload	
logStatus	string	MyBillPay
result	boolean	true

Request **Response**

รูปที่ 5-40 ผลลัพธ์การทดสอบด้วยกรณีทดสอบ TC01 ของกรณีศึกษาที่ 3 บนพีเพิลเอนจิน

- 5) **เรียกดูรายงานการทดสอบดับเบิลยูเอส-พีเพิล** หลังจากผู้ทดสอบทำการทดสอบดับเบิลยูเอส-พีเพิลด้วยกรณีทดสอบ TC01 TC02 และ TC03 บนพีเพิลเอนจินเรียบร้อยแล้ว ผู้ทดสอบกลับมาที่หน้าจอ ดังรูปที่ 5-36 และกดปุ่ม “Test Report” เพื่อเรียกดูรายงานการทดสอบ โดยรายละเอียดรายงานการทดสอบดับเบิลยูเอส-พีเพิลของกรณีศึกษาที่ 3 แสดงดังรูปที่ 5-41 และพบว่าทางเดินทดสอบ P1 P2 และ P3 ได้ถูกทดสอบแล้ว

:: WS-BPEL Test Report ::

Project Name: KioskTracking
 Total Test: 3 Time(s)
 Printed Date: 17 October 2020 04:29

Path ID	Path Detail	Test Case	Exercised?
P1	receiveInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - IfPayTheirSelf - PayTheirSelf - wsBillPayment - wsLogger - replyOutput	TC01	Exercised
P2	receiveInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - IfPayTheirSelf - PayOther - wsBillPayment - wsLogger - replyOutput	TC02	Exercised
P3	receiveInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - TopUpMobile - wsTopUpMobile - wsLogger - replyOutput	TC03	Exercised

:: Alternative node has been exercised :: (3/3)

Predicate Node	Condition	Exercised?	Test Case
IfBillPayment(T)	SOF='CREDITCARD'	Exercised	TC01 TC02
IfBillPayment(F)	NOT(SOF='CREDITCARD')	Exercised	TC03
IfPayTheirSelf(T)	IfPaySelf=true	Exercised	TC01
IfPayTheirSelf(F)	NOT(IfPaySelf=true)	Exercised	TC02

รูปที่ 5-41 รายงานการทดสอบดับเบิลยูเอส-บีเพลของกรณีศึกษาที่ 3

5.4 ผลการทดสอบเครื่องมือ

ผลการทดสอบเครื่องมือสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ แสดงดังตารางที่ 5-1 ถึงตารางที่ 5-6 โดยรายละเอียดผลการทดสอบจะแสดงให้เห็นถึงความครอบคลุมของกรณีทดสอบที่สามารถครอบคลุมทางเดินทดสอบในระดับกิ่ง คือครอบคลุมทุกโหนดเงื่อนไขในกราฟการเรียกเชิงสถิติ ซึ่งรายละเอียดของกรณีทดสอบสามารถดูเพิ่มเติมได้ที่ภาคผนวก ข

- 1) ผลการทดสอบกรณีศึกษาที่ 1 ระบบโปรแกรมรับเครดิตเงินคืนของแผนกเครื่องใช้ไฟฟ้า เครื่องมือมีการสร้างกรณีทดสอบจำนวน 6 กรณี รายละเอียดกรณีทดสอบที่ครอบคลุมทางเดินทดสอบและโหนดเงื่อนไขในกราฟการเรียกเชิงสถิติ แสดงดังตารางที่ 5-1 และตารางที่ 5-2

ตารางที่ 5-1 กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 1

ทางเดินทดสอบ	รายละเอียดทางเดินทดสอบ	กรณีทดสอบ
P1	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - IfPrice5000Up - IfPrice10000Up - Voucher700 - wsReturnCashback - replyOutput	TC01
P2	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - IfPrice5000Up - IfPrice10000Up - Voucher500 - wsReturnCashback - replyOutput	TC02
P3	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - IfPrice5000Up - Voucher200 - wsReturnCashback - replyOutput	TC03
P4	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - Voucher0 - wsReturnCashback - replyOutput	TC04
P5	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IfVisa - replyOutput	TC05
P6	receiveInput - Discount5Percent - IfCreditCard - CCNotAllow - replyOutput	TC06

ตารางที่ 5-2 กรณีทดสอบที่ครอบคลุมเงื่อนไขในกราฟการเรียกเชิงสถิติของกรณีศึกษาที่ 1

ลำดับที่	เงื่อนไขในกราฟการเรียกเชิงสถิติ	กรณีที่เป็นจริง/เท็จ	กรณีทดสอบที่ครอบคลุม
1	IfCreditCard	True	TC01, TC02, TC03, TC04, TC05
		False	TC06
2	IfVisa	True	TC01, TC02, TC03, TC04
		False	TC05

ตารางที่ 5-2 กรณีทดสอบที่ครอบคลุมเงื่อนไขในกราฟการเรียกเชิงสถิติของกรณีศึกษาที่ 1 (ต่อ)

ลำดับที่	เงื่อนไขในกราฟการเรียกเชิงสถิติ	กรณีที่เป็นจริง/เท็จ	กรณีทดสอบที่ครอบคลุม
3	IfPrice2500Up	True	TC01, TC02, TC03
		False	TC04
4	IfPrice5000Up	True	TC01, TC02
		False	TC03
5	IfPrice10000Up	True	TC01
		False	TC02

2) ผลการทดสอบกรณีศึกษาที่ 2 ระบบร้านขนส่งพัสดุ เครื่องมือมีการสร้างกรณีทดสอบจำนวน 9 กรณี โดยรายละเอียดกรณีทดสอบที่ครอบคลุมทางเดินทดสอบและเงื่อนไขในกราฟการเรียกเชิงสถิติ แสดงดังตารางที่ 5-3 และตารางที่ 5-4

ตารางที่ 5-3 กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 2

ทางเดินทดสอบ	รายละเอียดทางเดินทดสอบ	กรณีทดสอบ
P1	receivelInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeS - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	TC01
P2	receivelInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeS - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	TC02

ตารางที่ 5-3 กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 2 (ต่อ)

ทางเดินทดสอบ	รายละเอียดทางเดินทดสอบ	กรณีทดสอบ
P3	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeM - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	TC03
P4	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeM - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	TC04
P5	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - SizeL - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	TC05
P6	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - SizeL - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	TC06
P7	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - SizeXL - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	TC07
P8	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - SizeXL - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	TC08

ตารางที่ 5-3 กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 2 (ต่อ)

ทางเดินทดสอบ	รายละเอียดทางเดินทดสอบ	กรณีทดสอบ
P9	receiveInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - NotAcceptPkg - replyOutput	TC09

ตารางที่ 5-4 กรณีทดสอบที่ครอบคลุมเงื่อนไขในกราฟการเรียกเชิงสถิติของกรณีศึกษาที่ 2

ลำดับที่	เงื่อนไขในกราฟการเรียกเชิงสถิติ	กรณีที่เป็นจริง/เท็จ	กรณีทดสอบที่ครอบคลุม
1	IfTotalLessThan200	True	TC01, TC02, TC03, TC04, TC05, TC06, TC07, TC08
		False	TC09
2	IfTotalLessThan120	True	TC01, TC02, TC03, TC04, TC05, TC06
		False	TC07, TC08
3	IfTotalLessThan90	True	TC01, TC02, TC03, TC04
		False	TC05, TC06
4	IfTotalLessThan60	True	TC01, TC02
		False	TC03, TC04
5	IfOutOfProvince	True	TC01, TC03, TC05, TC07
		False	TC02, TC04, TC06, TC08

- 3) ผลการทดสอบกรณีศึกษาที่ 3 ระบบติดตามการทำงานของตู้คืออส เครื่องมือมีการสร้างกรณีทดสอบจำนวน 3 กรณี โดยรายละเอียดกรณีทดสอบที่ครอบคลุมทางเดินทดสอบและเงื่อนไขในกราฟการเรียกเชิงสถิติ แสดงดังตารางที่ 5-5 และตารางที่ 5-6

ตารางที่ 5-5 กรณีทดสอบที่ครอบคลุมทางเดินทดสอบของกรณีศึกษาที่ 3

ทางเดินทดสอบ	รายละเอียดทางเดินทดสอบ	กรณีทดสอบ
P1	receiveInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - IfPayTheirSelf - PayTheirSelf - wsBillPaymant - wsLogger - replyOutput	TC01
P2	receiveInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - IfPayTheirSelf - PayOther - wsBillPaymant - wsLogger - replyOutput	TC02
P3	receiveInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - TopUpMobile - wsTopUpMobile - wsLogger - replyOutput	TC03

ตารางที่ 5-6 กรณีทดสอบที่ครอบคลุมเงื่อนไขในกราฟการเรียกเชิงสถิติของกรณีศึกษาที่ 3

ลำดับที่	เงื่อนไขในกราฟการเรียกเชิงสถิติ	กรณีที่เป็นจริง/เท็จ	กรณีทดสอบที่ครอบคลุม
1	IfBillPayment	True	TC01, TC02
		False	TC03
2	IfPayTheirSelf	True	TC01
		False	TC02

5.5 สรุปผลการทดสอบเครื่องมือ

จากผลการทดสอบเครื่องมือสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพล จากกราฟการเรียกเชิงสถิติ ดังแสดงในตารางที่ 5-1 ถึง 5-6 พบว่าเครื่องมือสามารถสร้างกรณีทดสอบได้ครอบคลุมทางเดินทดสอบในระดับกึ่ง คือครอบคลุมทุกเงื่อนไขของกราฟการเรียกเชิงสถิติ โดยในกรณีศึกษาที่ 1 และ 2 เป็นกรณีศึกษาที่มีการเรียกใช้งานกันระหว่างไฟล์ดับเบิลยูเอส-บีเพลในรูปแบบกระบวนการย่อยภายนอกที่มีการเรียกใช้กระบวนการทางธุรกิจต่อกัน ซึ่งกรณีศึกษาที่ 1 เป็นกรณีศึกษาที่มีไฟล์ดับเบิลยูเอส-บีเพลเรียกใช้งานบีเพลย่อยแค่ครั้งละ 1 ไฟล์เท่านั้น ส่วนใน

กรณีศึกษาที่ 2 เป็นกรณีศึกษาที่มีไฟล์ดับเบิลยูเอส-พีเพิลเรียกใช้งานปีเพลด้อยครั้งละหลายไฟล์ และ
ในกรณีศึกษาที่ 3 เป็นกรณีศึกษาที่มีการเรียกใช้งานกันระหว่างดับเบิลยูเอส-พีเพิลในรูปแบบ
กระบวนการย่อยภายใน ซึ่งเครื่องมือยังคงสามารถสร้างกรณีทดสอบได้ครบถ้วนและครอบคลุม
ทางเดินทดสอบในระดับกึ่ง ซึ่งหมายความว่าครอบคลุมทุกโหนดเงื่อนไขของกราฟการเรียกเชิงสถิติ
ทั้งกรณีที่มีการเรียกใช้งานกันระหว่างดับเบิลยูเอส-พีเพิลในรูปแบบกระบวนการย่อยภายนอกที่มีการ
เรียกใช้งานปีเพลด้อยครั้งละ 1 ไฟล์ หรือหลายไฟล์ และกรณีที่มีการเรียกใช้งานระหว่างดับเบิลยูเอส-
พีเพิลในรูปแบบกระบวนการย่อยภายใน



บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

จากการศึกษา วิเคราะห์ วิจัย และพัฒนาเครื่องมือสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ สามารถสรุปผลการวิจัย ข้อจำกัดของงานวิจัย และแนวทางการพัฒนางานวิจัยต่อไปในอนาคต โดยมีรายละเอียด ดังนี้

6.1 สรุปผลการวิจัย

งานวิจัยนี้ได้นำเสนอวิธีการและพัฒนาเครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ เพื่อรองรับกรณีที่มีการเรียกใช้งานกันระหว่างดับเบิลยูเอส-บีเพล โดยผู้ทดสอบสามารถนำเข้าชุดไฟล์ข้อมูลโครงการดับเบิลยูเอส-บีเพล ที่ประกอบไปด้วย ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล ไฟล์บีเพลตั้งต้น ไฟล์ดับเบิลยูเอสดีแอล ไฟล์เอกซ์เอสดี และชุดไฟล์บีเพลย่อยที่ถูกเรียกใช้งาน โดยเครื่องมือจะอ่านข้อมูลในแต่ละไฟล์นำเข้า เพื่อจัดเก็บเส้นทางการไหลของบีเพล แทรกชุดคำสั่งในรหัสต้นทาง ตรวจสอบการเรียกใช้งานกันระหว่างไฟล์ดับเบิลยูเอส-บีเพลเพื่อสร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติ วิเคราะห์หาตัวแปรและข้อจำกัดของตัวแปร จากนั้นเครื่องมือจะนำข้อมูลที่วิเคราะห์ได้มารวบรวมและสร้างเป็นกรณีทดสอบในแต่ละทางเดินทดสอบ เมื่อผู้ทดสอบนำรหัสต้นทางที่แทรกชุดคำสั่งไปติดตั้งบนบีเพลเอนจิน และทำการทดสอบดับเบิลยูเอส-บีเพลบนบีเพลเอนจินด้วยกรณีทดสอบที่เครื่องมือสร้าง บีเพลเอนจินจะส่งข้อมูลการทดสอบกลับมายังเครื่องมือผ่านเว็บเซอร์วิสตามที่ได้แทรกชุดคำสั่งไว้ สุดท้ายผู้ใช้งานสามารถเรียกดูรายงานการทดสอบเพื่อตรวจสอบข้อมูลการทดสอบดับเบิลยูเอส-บีเพลได้ ซึ่งจากการทดสอบเครื่องมือโดยใช้กรณีศึกษา 3 กรณีศึกษาดัง หัวข้อ 5.3 พบว่าเครื่องมือสามารถทำงานได้ถูกต้อง

6.2 ข้อจำกัดงานวิจัย

เครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ มีข้อจำกัดดังต่อไปนี้

- 1) เครื่องมือได้ทดสอบดับเบิลยูเอส-บีเพลและรองรับการทำงานร่วมกับบีเพลเอนจินของ Oracle JDeveloper เวอร์ชัน 12c เท่านั้น

- 2) เครื่องมือรองรับการทำงานร่วมกับโหนดดับเบิลยูเอส-บีเพล ดังนี้ <receive>, <reply>, <invoke>, <assign>, <sequence>, <extensionActivity>, <bpelx:call>, <if> และ <else>
- 3) เครื่องมือรองรับเงื่อนไข เฉพาะเงื่อนไขพื้นฐานที่ประกอบไปด้วย >, >=, <, <=, =, !=, true และ false
- 4) เครื่องมือรองรับการวิเคราะห์ตัวแปรเฉพาะโอเพอร์เรเตอร์ที่อยู่ในฟังก์ชันคณิตศาสตร์ +, -, * และ div
- 5) เครื่องมือรองรับเฉพาะข้อจำกัดของตัวแปรนำเข้ากระบวนการดับเบิลยูเอส-บีเพล ดังนี้ enumeration, fractionDigits, length, maxExclusive, maxInclusive, maxLength, minExclusive, minInclusive, minLength และ totalDigits
- 6) เครื่องมือรองรับการสร้างกรณีทดสอบ ที่ค่าตัวแปรในโหนดเงื่อนไข <if> ต้องเป็นค่าตัวแปรนำเข้าที่ไม่มีการเปลี่ยนแปลงภายในดับเบิลยูเอส-บีเพล
- 7) เครื่องมือรองรับตัวแปรที่ไม่ซ้ำกันในแต่ละไฟล์ดับเบิลยูเอส-บีเพลที่มีการเรียกใช้กัน

6.3 แนวทางการพัฒนางานวิจัย

เครื่องมือการสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ สามารถนำไปพัฒนาต่อยอดเพื่อให้ครอบคลุมการทำงานมากยิ่งขึ้น ดังต่อไปนี้

- 1) พัฒนาเครื่องมือให้สามารถรองรับและทดสอบร่วมกับบีเพลเอนจินอื่น ๆ ได้ โดยที่ไม่ยึดติดกับเครื่องมือ
- 2) พัฒนาเครื่องมือให้สามารถทำงานร่วมกับบีเพลเอนจินได้ โดยไม่ต้องสลับการทำงานระหว่างเครื่องมือและบีเพลเอนจิน
- 3) พัฒนาพัฒนาเครื่องมือให้รองรับฟังก์ชันการทำงานประเภทอื่น ๆ ที่ซับซ้อนขึ้น เช่น ฟังก์ชันการใช้งานขั้นสูง (Advance Functions) หรือฟังก์ชันทางด้านตรรกะ (Logical Functions) หรือฟังก์ชันอักขระ (String Functions) เป็นต้น

บรรณานุกรม

- 1 P. Brown, J. A. Estefan, K. Laskey, F. G. McCabe, and D. Thornton. *Reference Architecture Foundation for Service Oriented Architecture Version 1.0* [Online]. <http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/cs01/soa-ra-v1.0-cs01.html> (accessed 10 October 2020).
- 2 A. Alves et al. *Web Services Business Process Execution Language Version 2.0* [Online]. <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html> (accessed 10 October 2020).
- 3 Oracle Corporation. *Oracle SOA Suite Templates and Reusable Subprocesses* [Online]. <https://docs.oracle.com/middleware/1221/soasuite/develop/GUID-FAA53226-470D-4120-95ED-1EAB6D96AA3F.htm#SOASE88048> (accessed 10 October 2020).
- 4 E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana. *Web Services Description Language (WSDL) Version 1.1* [Online]. <https://www.w3.org/TR/2001/NOTE-wsdl-20010315> (accessed 13 October 2020).
- 5 W3C. *The World Wide Web Consortium* [Online]. <https://www.w3.org> (accessed 13 October 2020).
- 6 S. Gao, C. M. Sperberg-McQueen, and H. S. Thompson. *W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures* [Online]. <https://www.w3.org/TR/xmlschema11-1/> (accessed 13 October 2020).
- 7 M. B. Juric. *A Hands-on Introduction to BPEL* [Online]. <http://www.oracle.com/technetwork/articles/matjaz-bpel1-090575.html> (accessed 10 October 2020).
- 8 B. Holland. *Call Graph Construction Algorithms Explained* [Online]. <https://ben-holland.com/call-graph-construction-algorithms-explained/> (accessed 13 October 2020).
- 9 P. Nakngern and T. Suwannasart, *A Design of WS-BPEL Test Case Generation Tool Based on Path Conditions*, presented at the *The International MultiConference of Engineers and Computer Scientists (IMECS)*, 2017.

- 10 E. Shamsoddin-Motlagh, *Automatic Test Case Generation for Orchestration Languages at Service Oriented Architecture*, *International Journal of Computer Applications*, pp. 35-44, 2013, doi: 10.5120/13876-1756.
- 11 S. Laokok and T. Suwannasart, *An Approach for Test Case Generation from a Static Call Graph for Object-Oriented Programming*, presented at the *The International MultiConference of Engineers and Computer Scientists (IMECS)*, 2017.





ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาคผนวก ก

รายละเอียดยูสเคสของเครื่องมือ

ในภาคผนวก ก เป็นการแสดงรายละเอียดเกี่ยวกับยูสเคสของเครื่องมือการสร้างกรณีทดสอบ สำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-บีเพลจากกราฟการเรียกเชิงสถิติ รายละเอียดยูสเคสแสดงดังตารางที่ ก-1 ถึงตารางที่ ก-8

ตารางที่ ก-1 รายละเอียดของยูสเคสนำเข้าข้อมูล

รหัสยูสเคส	UC001
ชื่อยูสเคส	นำเข้าข้อมูล
แอกเตอร์	เครื่องมือ
รายละเอียดยูสเคส	เพื่อนำข้อมูลเข้าสู่เครื่องมือ ประกอบไปด้วยข้อมูลตั้งต้นและชุดไฟล์บีเพลย่อย โดยข้อมูลตั้งต้น ได้แก่ ชื่อโครงการ ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล ไฟล์บีเพลตั้งต้น ไฟล์ดับเบิลยูเอสดีแอล และไฟล์เอกซ์เอสดี
ยูสเคสที่เกี่ยวข้อง	Include: แทรกชุดคำสั่งในรหัสต้นทาง
เงื่อนไขก่อนหน้า	<ol style="list-style-type: none">1) ไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล อยู่ในรูปแบบ .xml2) ไฟล์บีเพลตั้งต้น อยู่ในรูปแบบ .bpel3) ไฟล์ดับเบิลยูเอสดีแอล อยู่ในรูปแบบ .wsdl4) ไฟล์เอกซ์เอสดี อยู่ในรูปแบบ .xsd5) ไฟล์บีเพลย่อยอยู่ในรูปแบบ .sbpel
ขั้นตอน	<ol style="list-style-type: none">1) ผู้ทดสอบระบุชื่อโครงการดับเบิลยูเอส-บีเพล2) ผู้ทดสอบเลือกไฟล์คอมโพสิตของดับเบิลยูเอส-บีเพล3) ผู้ทดสอบเลือกไฟล์บีเพลตั้งต้น4) ผู้ทดสอบเลือกไฟล์ดับเบิลยูเอสดีแอล5) ผู้ทดสอบเลือกไฟล์เอกซ์เอสดี6) ผู้ทดสอบกดปุ่ม Next เพื่อให้เครื่องมือดำเนินการนำเข้าไฟล์ตั้งต้น7) เครื่องมือค้นหาการเรียกใช้งานบีเพลย่อยจากไฟล์บีเพลตั้งต้น8) เครื่องมือแสดงแถบอัปโหลดไฟล์ตามจำนวนบีเพลย่อยที่มีการเรียกใช้งาน9) ผู้ทดสอบเลือกไฟล์บีเพลย่อยให้ถูกต้องและครบถ้วน10) ผู้ทดสอบกดปุ่ม Next เพื่อให้เครื่องมือนำเข้าชุดไฟล์บีเพลย่อย

ตารางที่ ก-1 รายละเอียดของยูสเคสนำเข้าข้อมูล (ต่อ)

ขั้นตอน	11) เครื่องมือจัดเก็บชุดไฟล์บีเพลย่อย และค้นหาการเรียกใช้งานบีเพลย่อย จากชุดไฟล์บีเพลย่อยที่ผู้ทดสอบนำเข้า 12) เครื่องมือทำซ้ำขั้นตอนที่ 8 - 11 จนกว่าจะไม่พบการเรียกใช้งานบีเพลย่อยเพิ่มเติม 13) เรียกใช้ยูสเคสแทรกชุดคำสั่งในรหัสต้นทาง
เงื่อนไขภายหลัง	เครื่องมือจัดเก็บไฟล์บีเพลตั้งต้น ไฟล์ดับเบิลยูเอสดีแอล ไฟล์เอกซ์เอสดี และชุดไฟล์บีเพลย่อยไว้บนเว็บเซิร์ฟเวอร์ พร้อมทั้งบันทึกข้อมูลในฐานข้อมูล

ตารางที่ ก-2 รายละเอียดของยูสเคสแทรกชุดคำสั่งในรหัสต้นทาง

รหัสยูสเคส	UC002
ชื่อยูสเคส	แทรกชุดคำสั่งในรหัสต้นทาง
แอกเตอร์	เครื่องมือ
รายละเอียดยูสเคส	เพื่อแทรกชุดคำสั่งในรหัสต้นทางของไฟล์บีเพลตั้งต้นและชุดไฟล์บีเพลย่อย
ยูสเคสที่เกี่ยวข้อง	-
เงื่อนไขก่อนหน้า	ไฟล์บีเพลตั้งต้นและชุดไฟล์บีเพลย่อยได้รับการจัดเก็บอยู่บนเว็บเซิร์ฟเวอร์
ขั้นตอน	1) เครื่องมืออ่านไฟล์บีเพลตั้งต้นและชุดไฟล์บีเพลย่อย 2) เครื่องมือแทรกชุดคำสั่งในรหัสต้นทาง
เงื่อนไขภายหลัง	เครื่องมือจัดเก็บรหัสต้นทางที่ถูกแทรกชุดคำสั่งลงในเว็บเซิร์ฟเวอร์

ตารางที่ ก-3 รายละเอียดของยูสเคสสร้างกรณีทดสอบ

รหัสยูสเคส	UC003
ชื่อยูสเคส	สร้างกรณีทดสอบ
แอกเตอร์	ผู้ทดสอบ
รายละเอียดยูสเคส	เพื่อสร้างกรณีทดสอบตามทางเดินทดสอบ
ยูสเคสที่เกี่ยวข้อง	Include: นำเข้าข้อมูล

ตารางที่ ก-3 รายละเอียดของยูสเคสสร้างกรณีทดสอบ (ต่อ)

เงื่อนไขก่อนหน้า	ไฟล์บีเพลดั้งเดิม ชุดไฟล์บีเพลด้อย ไฟล์ดับเบิลยูเอสดีแอล และไฟล์เอกซ์เอสดี ได้รับการจัดเก็บอยู่บนเว็บเซิร์ฟเวอร์
ขั้นตอน	<ol style="list-style-type: none"> 1) เรียกใช้ยูสเคสนำเข้าข้อมูล 2) เครื่องมืออ่านไฟล์บีเพลดั้งเดิม และชุดไฟล์บีเพลด้อย 3) เครื่องมือจัดเก็บเส้นทางการไหลของบีเพลด 4) เครื่องมือสร้างกราฟการเรียกเชิงสถิติ โดยวิเคราะห์ความสัมพันธ์การเรียกใช้งานกันระหว่างดับเบิลยูเอส-บีเพลด 5) เครื่องมือสร้างทางเดินทดสอบจากกราฟการเรียกเชิงสถิติ 6) เครื่องมือค้นหารายการตัวแปร และข้อจำกัดของตัวแปรที่อยู่บนทางเดินทดสอบ 7) เครื่องมือสุ่มค่าข้อมูลข้อมูลนำเข้ากระบวนการดับเบิลยูเอส-บีเพลดตามข้อจำกัดของตัวแปร 8) เครื่องมือสร้างกรณีทดสอบ
เงื่อนไขภายหลัง	เครื่องมือบันทึกกรณีทดสอบลงในฐานข้อมูล

ตารางที่ ก-4 รายละเอียดของยูสเคสดาวนโหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่ง

รหัสยูสเคส	UC004
ชื่อยูสเคส	ดาวนโหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่ง
แอกเตอร์	ผู้ทดสอบ
รายละเอียดยูสเคส	เพื่อเรียกดู/ดาวนโหลดรหัสต้นทางที่ถูกแทรกชุดคำสั่ง
ยูสเคสที่เกี่ยวข้อง	-
เงื่อนไขก่อนหน้า	รหัสต้นทางที่ถูกแทรกชุดคำสั่งได้รับการบันทึกอยู่บนเว็บเซิร์ฟเวอร์
ขั้นตอน	<ol style="list-style-type: none"> 1) ผู้ทดสอบกดปุ่ม Download 2) ผู้ทดสอบเลือก Instrumented Files 3) เครื่องมือส่งออกไฟล์รหัสต้นทางที่ถูกแทรกชุดคำสั่งไปยังผู้ทดสอบ
เงื่อนไขภายหลัง	เครื่องมือส่งออกรหัสต้นทางที่ถูกแทรกชุดคำสั่งไปยังผู้ทดสอบ

ตารางที่ ก-5 รายละเอียดของยูสเคสดาวนโหลดกรณีทดสอบทั้งหมด

รหัสยูสเคส	UC005
ชื่อยูสเคส	ดาวนโหลดกรณีทดสอบทั้งหมด
แอกเตอร์	ผู้ทดสอบ
รายละเอียดยูสเคส	เพื่อเรียกดู/ดาวนโหลดกรณีทดสอบทั้งหมด
ยูสเคสที่เกี่ยวข้อง	-
เงื่อนไขก่อนหน้า	กรณีทดสอบทั้งหมดได้รับการบันทึกอยู่บนเว็บเซิร์ฟเวอร์
ขั้นตอน	1) ผู้ทดสอบกดปุ่ม Download 2) ผู้ทดสอบเลือก All Test Cases 3) เครื่องมือส่งออกกรณีทดสอบในรูปแบบไฟล์เอกซ์เซลและเอกซ์เอ็มแอลไปยังผู้ทดสอบ
เงื่อนไขภายหลัง	เครื่องมือส่งออกกรณีทดสอบในรูปแบบไฟล์เอกซ์เซลและเอกซ์เอ็มแอลไปยังผู้ทดสอบ

ตารางที่ ก-6 รายละเอียดของยูสเคสเรียกดูรายละเอียดกรณีทดสอบ

รหัสยูสเคส	UC006
ชื่อยูสเคส	เรียกดูรายละเอียดกรณีทดสอบ
แอกเตอร์	ผู้ทดสอบ
รายละเอียดยูสเคส	เพื่อเรียกดูกรณีทดสอบที่สนใจ
ยูสเคสที่เกี่ยวข้อง	-
เงื่อนไขก่อนหน้า	กรณีทดสอบของแต่ละทางเดินทดสอบถูกบันทึกอยู่ในฐานข้อมูล
ขั้นตอน	ผู้ทดสอบกดหมายเลขกรณีทดสอบของทางเดินทดสอบที่สนใจที่แสดงอยู่ในตาราง
เงื่อนไขภายหลัง	เครื่องมือแสดงรายละเอียดกรณีทดสอบที่ผู้ทดสอบเลือก

ตารางที่ ก-7 รายละเอียดของยูสเคสเรียกดูรายงานการทดสอบ

รหัสยูสเคส	UC007
ชื่อยูสเคส	เรียกดูรายงานการทดสอบ
แอกเตอร์	ผู้ทดสอบ
รายละเอียดยูสเคส	เพื่อเรียกดูรายงานการทดสอบ
ยูสเคสที่เกี่ยวข้อง	-
เงื่อนไขก่อนหน้า	ข้อมูลทางเดินทดสอบได้รับการบันทึกอยู่ในฐานข้อมูล
ขั้นตอน	ผู้ทดสอบกดปุ่ม Test Report
เงื่อนไขภายหลัง	เครื่องมือแสดงรายงานการทดสอบให้แก่ผู้ทดสอบ

ตารางที่ ก-8 รายละเอียดของยูสเคสเรียกดูประวัติการทดสอบดับเบิลยูเอส-บีเฟล

รหัสยูสเคส	UC008
ชื่อยูสเคส	เรียกดูประวัติการทดสอบดับเบิลยูเอส-บีเฟล
แอกเตอร์	ผู้ทดสอบ
รายละเอียดยูสเคส	เพื่อเรียกดูประวัติการทดสอบดับเบิลยูเอส-บีเฟล
ยูสเคสที่เกี่ยวข้อง	-
เงื่อนไขก่อนหน้า	โครงการดับเบิลยูเอส-บีเฟลที่สามารถเรียกดูประวัติการทดสอบได้ จะต้องเป็นโครงการที่ข้อมูลทางเดินทดสอบได้รับการบันทึกอยู่ในฐานข้อมูล
ขั้นตอน	<ol style="list-style-type: none"> 1) เครื่องมือแสดงโครงการดับเบิลยูเอส-บีเฟลที่อยู่ในระบบ 2) ผู้ใช้งานกดเลือกโครงการดับเบิลยูเอส-บีเฟลที่ต้องการ 3) เครื่องมือแสดงประวัติการทดสอบดับเบิลยูเอส-บีเฟล
เงื่อนไขภายหลัง	เครื่องมือแสดงประวัติการทดสอบดับเบิลยูเอส-บีเฟล

ภาคผนวก ข

รายละเอียดกรณีทดสอบของแต่ละกรณีศึกษา

ในภาคผนวก ข เป็นการแสดงรายละเอียดเกี่ยวกับกรณีทดสอบที่ได้จากการทดสอบกรณีศึกษาทั้ง 3 กรณีศึกษากับเครื่องมือสร้างกรณีทดสอบสำหรับเว็บเซอร์วิสแบบดับเบิลยูเอส-พีแอล จากกราฟการเรียกเชิงสถิติ โดยรายละเอียดของกรณีทดสอบที่จำแนกตามกรณีศึกษา แสดงดังตารางที่ ข-1 ถึงตารางที่ ข-18

- 1) กรณีศึกษาที่ 1 ระบบโปรโมชั่นรับเครดิตเงินคืนของแผนกเครื่องใช้ไฟฟ้า

ตารางที่ ข-1 กรณีทดสอบ TC01 ของกรณีศึกษาที่ 1

โครงการ	CreditCardPromotion	
รหัสกรณีทดสอบ	TC01	
ทางเดินทดสอบ	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - IfPrice5000Up - IfPrice10000Up - Voucher700 - wsReturnCashback - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Price	decimal	34690.42
SOF	string	CREDITCARD
CCNo	string	5059232372440152
CreditCardType	string	VISA

ตารางที่ ข-2 กรณีทดสอบ TC02 ของกรณีศึกษาที่ 1

โครงการ	CreditCardPromotion	
รหัสกรณีทดสอบ	TC02	
ทางเดินทดสอบ	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - IfPrice5000Up - IfPrice10000Up - Voucher500 - wsReturnCashback - replyOutput	

ตารางที่ ข-2 กรณีทดสอบ TC02 ของกรณีศึกษาที่ 1 (ต่อ)

ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Price	decimal	7218.54
SOF	string	CREDITCARD
CCNo	string	9857600384756128
CreditCardType	string	VISA

ตารางที่ ข-3 กรณีทดสอบ TC03 ของกรณีศึกษาที่ 1

โครงการ	CreditCardPromotion	
รหัสกรณืทดสอบ	TC03	
ทางเดินทดสอบ	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - IfPrice5000Up - Voucher200 - wsReturnCashback - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Price	decimal	1753.41
SOF	string	CREDITCARD
CCNo	string	1873246702042983
CreditCardType	string	VISA

ตารางที่ ข-4 กรณีทดสอบ TC04 ของกรณีศึกษาที่ 1

โครงการ	CreditCardPromotion	
รหัสกรณืทดสอบ	TC04	
ทางเดินทดสอบ	receiveInput - Discount5Percent - IfCreditCard - CCAAllow - wsPayCreditCard - IfVisa - IfPrice2500Up - Voucher0 - wsReturnCashback - replyOutput	

ตารางที่ ข-4 กรณีทดสอบ TC04 ของกรณีศึกษาที่ 1 (ต่อ)

ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Price	decimal	1753.41
SOF	string	CREDITCARD
CCNo	string	9467582939419930
CreditCardType	string	VISA

ตารางที่ ข-5 กรณีทดสอบ TC05 ของกรณีศึกษาที่ 1

โครงการ	CreditCardPromotion	
รหัสกรณืทดสอบ	TC05	
ทางเดินทดสอบ	receiveInput - Discount5Percent - IfCreditCard - CCAllow - wsPayCreditCard - IfVisa - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Price	decimal	682439.29
SOF	string	CREDITCARD
CCNo	string	6573035782841912
CreditCardType	string	JCB

CHULALONGKORN UNIVERSITY

ตารางที่ ข-6 กรณีทดสอบ TC06 ของกรณีศึกษาที่ 1

โครงการ	CreditCardPromotion	
รหัสกรณืทดสอบ	TC06	
ทางเดินทดสอบ	receiveInput - Discount5Percent - IfCreditCard - CCNotAllow - replyOutput	

ตารางที่ ข-6 กรณีทดสอบ TC06 ของกรณีศึกษาที่ 1 (ต่อ)

ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Price	decimal	5913.41
SOF	string	CASH
CCNo	string	7942395889063418
CreditCardType	string	MASTER

2) กรณีศึกษาที่ 2 ระบบร้านขนส่งพัสดุ

ตารางที่ ข-7 กรณีทดสอบ TC01 ของกรณีศึกษาที่ 2

โครงการ	TranExpress	
รหัสกรณีทดสอบ	TC01	
ทางเดินทดสอบ	receivelInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeS - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Weigh	decimal	10.81
W	integer	15
H	integer	9
L	integer	30
OutOfProvince	boolean	true

ตารางที่ ข-8 กรณีทดสอบ TC02 ของกรณีศึกษาที่ 2

โครงการ	TranExpress	
รหัสกรณีทดสอบ	TC02	
ทางเดินทดสอบ	receivelInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeS - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Weigh	decimal	40.56
W	integer	20
H	integer	11
L	integer	14
OutOfProvince	boolean	false

ตารางที่ ข-9 กรณีทดสอบ TC03 ของกรณีศึกษาที่ 2

โครงการ	TranExpress	
รหัสกรณีทดสอบ	TC03	
ทางเดินทดสอบ	receivelInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeM - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	

ตารางที่ ข-9 กรณีทดสอบ TC03 ของกรณีศึกษาที่ 2 (ต่อ)

ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Weigh	decimal	5.83
W	integer	17
H	integer	23
L	integer	49
OutOfProvince	boolean	true

ตารางที่ ข-10 กรณีทดสอบ TC04 ของกรณีศึกษาที่ 2

โครงการ	TranExpress	
รหัสกรณีสอบ	TC04	
ทางเดินทดสอบ	receivInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - IfTotalLessThan60 - SizeM - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Weigh	decimal	8.21
W	integer	32
H	integer	19
L	integer	21
OutOfProvince	boolean	false

ตารางที่ ข-11 กรณีทดสอบ TC05 ของกรณีศึกษาที่ 2

โครงการ	TranExpress	
รหัสกรณีทดสอบ	TC05	
ทางเดินทดสอบ	receivInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - SizeL - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Weigh	decimal	49.11
W	integer	51
H	integer	28
L	integer	36
OutOfProvince	boolean	true

ตารางที่ ข-12 กรณีทดสอบ TC06 ของกรณีศึกษาที่ 2

โครงการ	TranExpress	
รหัสกรณีทดสอบ	TC06	
ทางเดินทดสอบ	receivInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - IfTotalLessThan90 - SizeL - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Weigh	decimal	10.81
W	integer	32
H	integer	11
L	integer	59
OutOfProvince	boolean	false

ตารางที่ ข-13 กรณีทดสอบ TC07 ของกรณีศึกษาที่ 2

โครงการ	TranExpress	
รหัสกรณีทดสอบ	TC07	
ทางเดินทดสอบ	receivInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - SizeXL - wsPreparePkg - IfOutOfProvince - AddOnPrice - Add20THB - Add5THB - wsFinanceQueue - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Weigh	decimal	55.02
W	integer	47
H	integer	63
L	integer	51
OutOfProvince	boolean	true

ตารางที่ ข-14 กรณีทดสอบ TC08 ของกรณีศึกษาที่ 2

โครงการ	TranExpress	
รหัสกรณีทดสอบ	TC08	
ทางเดินทดสอบ	receivInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - AcceptPkg - IfTotalLessThan120 - SizeXL - wsPreparePkg - IfOutOfProvince - wsFinanceQueue - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Weigh	decimal	69.38
W	integer	66
H	integer	81
L	integer	45
OutOfProvince	boolean	false

ตารางที่ ข-15 กรณีทดสอบ TC09 ของกรณีศึกษาที่ 2

โครงการ	TranExpress	
รหัสกรณีทดสอบ	TC09	
ทางเดินทดสอบ	receivInput - wsRegister - wsPartnerRegister - IfTotalLessThan200 - NotAcceptPkg - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
Weigh	decimal	1.39
W	integer	132
H	integer	41
L	integer	87
OutOfProvince	boolean	false

3) กรณีศึกษาที่ 3 ระบบติดตามการทำงานตู้คีออส

ตารางที่ ข-16 กรณีทดสอบ TC01 ของกรณีศึกษาที่ 3

โครงการ	KioskTracking	
รหัสกรณีทดสอบ	TC01	
ทางเดินทดสอบ	receivInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - IfPayTheirSelf - PayTheirSelf - wsBillPaymant - wsLogger - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
IDCard	integer	5678940987638
MobileNo	integer	0734326754
Price	decimal	1054.85

ตารางที่ ข-17 กรณีทดสอบ TC02 ของกรณีศึกษาที่ 3

โครงการ	KioskTracking	
รหัสกรณืทดสอบ	TC02	
ทางเดินทดสอบ	receivelInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - IfPayTheirSelf - PayTheirSelf - wsBillPaymant - wsLogger - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
IDCard	integer	8484962395873
MobileNo	integer	5424737865
Price	decimal	23458.01

ตารางที่ ข-18 กรณีทดสอบ TC03 ของกรณีศึกษาที่ 3

โครงการ	KioskTracking	
รหัสกรณืทดสอบ	TC03	
ทางเดินทดสอบ	receivelInput - VerifyAuthen - wsVerifyAuthen - wsLogger - IfBillPayment - TopUpMobile - wsTopUpMobile - wsLogger - replyOutput	
ตัวแปรนำเข้า		
ตัวแปร	ชนิดข้อมูล	ค่าข้อมูล
IDCard	integer	3905749387205
MobileNo	integer	6390927845
Price	decimal	4771.03

ประวัติผู้เขียน

ชื่อ-สกุล	วาริรัตน์ บัวเสนาะ
วัน เดือน ปี เกิด	21 ธันวาคม 2529
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษา	1) วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมซอฟต์แวร์ จุฬาลงกรณ์มหาวิทยาลัย 2) วิทยาศาสตรบัณฑิต สาขาวิชาสถิติประยุกต์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพมหานคร
ที่อยู่ปัจจุบัน	
ผลงานตีพิมพ์	Wareerat Bousanoh and Taratip Suwannasart. 2018. Test Case Generation for WS-BPEL from a Static Call Graph. In Proceedings of The 11th International Conference on Computer and Electrical Engineering (ICCEE 2018), Tokyo, Japan.