

การตรวจจบการเรียกใช้เว็บไซต์สำหรับทดสอบดับเบิลยูเอส-บีเพล



นายณัฐพงศ์ เอื้อเพิ่มเกียรติ

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย

The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)
are the thesis authors' files submitted through the University Graduate School.

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2560

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

DETECTING WEB SERVICE INVOCATIONS FOR TESTING WS-BPEL

Mr. Nuttaphong Uaphoemkiat



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

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2017

Copyright of Chulalongkorn University

ณัฐพงศ์ เอื้อเพิ่มเกียรติ : การตรวจจับการเรียกใช้เว็บเซอร์วิสสำหรับทดสอบดับเบิลยูเอส-บีเพล (DETECTING WEB SERVICE INVOCATIONS FOR TESTING WS-BPEL) อ.ที่
 ปรึกษาวิทยานิพนธ์หลัก: รศ. ดร.ธราทิพย์ สุวรรณศาสตร์, 143 หน้า.

ดับเบิลยูเอส-บีเพล เป็นภาษาที่อนุญาตให้ผู้ออกแบบกระบวนการทางธุรกิจ นำเว็บเซอร์วิสที่มีมาทำงานร่วมกับกระบวนการทางธุรกิจขององค์กร โดยเรียกผ่านพาร์ทเนอร์ลิงก์ที่เป็นแท็กเชื่อมโยงการเรียกใช้งานเว็บเซอร์วิส เมื่อออกแบบกระบวนการทางธุรกิจโดยใช้ ดับเบิลยูเอส-บีเพลแล้ว ผู้ทดสอบควรมีการทดสอบการทำงานทุกเว็บเซอร์วิส ที่อยู่ภายใต้ดับเบิลยูเอส-บีเพล ซึ่งความท้าทายของการทดสอบเว็บเซอร์วิสคือ การตามรอยข้อความที่รับส่งระหว่างดับเบิลยูเอส-บีเพลกับเว็บเซอร์วิส และทดสอบทุกเว็บเซอร์วิสที่อยู่ภายใต้ดับเบิลยูเอส-บีเพล

วิทยานิพนธ์นี้มีจุดประสงค์เพื่อนำเสนอวิธีการสำหรับตรวจจับการเรียกใช้เว็บเซอร์วิสสำหรับทดสอบดับเบิลยูเอส-บีเพล โดยมุ่งเน้นการตรวจจับข้อความที่รับส่งระหว่างเว็บเซอร์วิส โดยใช้วิธีการแทรกห้สตันทาง และสามารถแสดงรายการเว็บเซอร์วิสที่ไม่ถูกทดสอบให้ผู้ทดสอบทราบได้ วิธีการที่นำเสนอนี้ยังครอบคลุมถึงการสร้างกรณีทดสอบเพิ่มเติมสำหรับเว็บเซอร์วิสที่ไม่ถูกทดสอบภายใต้ข้อจำกัดบางประการได้ วิทยานิพนธ์นี้ยังนำวิธีการที่เสนอมาพัฒนาเป็นเครื่องมือสำหรับตรวจจับการเรียกใช้เว็บเซอร์วิสสำหรับทดสอบดับเบิลยูเอส-บีเพล โดยเครื่องมือจะสามารถอ่านไฟล์ดับเบิลยูเอส-บีเพล เพื่อจัดเก็บเส้นทางการไหลและแทรกห้สตันทางได้ สามารถจัดเก็บและแสดงข้อความที่รับส่งระหว่างเว็บเซอร์วิสกับดับเบิลยูเอส-บีเพลได้ สามารถแสดงเว็บเซอร์วิสที่ไม่ถูกทดสอบให้ผู้ทดสอบทราบได้ นอกจากนี้ยังสามารถสร้างกรณีทดสอบเพิ่มเติมสำหรับเว็บเซอร์วิสที่ไม่ถูกทดสอบภายใต้ข้อจำกัดที่ระบุไว้ได้ เครื่องมือที่พัฒนาขึ้นนี้ได้นำไปทดสอบกับกรณีศึกษา 3 กรณีศึกษา ซึ่งผลลัพธ์ที่ได้คือ เครื่องมือสามารถจัดเก็บข้อมูลนำเข้าและข้อมูลส่งออกที่รับส่งระหว่างเว็บเซอร์วิสได้ สามารถแสดงรายการเว็บเซอร์วิสที่ไม่ถูกทดสอบ และสามารถสร้างกรณีทดสอบเพิ่มเติม ตามที่คาดหวังได้ถูกต้อง

ภาควิชา วิศวกรรมคอมพิวเตอร์

ลายมือชื่อนิสิต

สาขาวิชา วิศวกรรมซอฟต์แวร์

ลายมือชื่อ อ.ที่ปรึกษาหลัก

ปีการศึกษา 2560

5870933121 : MAJOR SOFTWARE ENGINEERING

KEYWORDS: WEB SERVICE / WS-BPEL / TESTING

NUTTAPHONG UAPHOEMKIAT: DETECTING WEB SERVICE INVOCATIONS FOR TESTING WS-BPEL. ADVISOR: ASSOC. PROF. TARATIP SUWANNASART, Ph.D., 143 pp.

WS-BPEL is a language that allows a business designer to bring existing web services integrate with a business process by using an associate tag called a partner link. After designing a business process using the WS-BPEL, a tester needs to test all web services within the WS-BPEL. The challenge in web service testing is to trace messages sending between a web service and the WS-BPEL, and test all web services within the WS-BPEL.

This thesis presents an approach for detecting web service invocations for testing WS-BPEL. The proposed approach focuses on detecting web services messages using code instrumentation technique, and is able to display untested web services to testers. The proposed approach also covers a generation of additional test cases for the untested web services under some constraints. The thesis uses the proposed approach to develop a tool for detecting web service invocations for testing WS-BPEL. Features of the tool include reading a WS-BPEL to create and store the flow graph of the WS-BPEL, instrumenting the WS-BPEL, storing and displaying messages sending between a web service and the WS-BPEL, displaying untested web services to a tester, and generating additional test cases for the untested web services within the mentioned constraints. The tool is tested by 3 case studies. The results show that the tool can store input and output messages sending between web services, display untested web services, and generate additional test cases correctly.

Department: Computer Engineering Student's Signature

Field of Study: Software Engineering Advisor's Signature

Academic Year: 2017

กิตติกรรมประกาศ

ขอขอบคุณ รองศาสตราจารย์ ดร.ธाराทิพย์ สุวรรณศาสตร์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ได้กรุณาสละเวลาให้ความช่วยเหลือ ให้คำปรึกษาแนะนำแนวทางการทำวิจัย ตลอดจนตรวจทานปรับปรุงแก้ไขข้อบกพร่องต่างๆ ในงานวิจัย เพื่อให้วิทยานิพนธ์ฉบับนี้มีความสมบูรณ์และสำเร็จลุล่วงไปด้วยดี

ขอขอบคุณ รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ ประธานกรรมการสอบ รองศาสตราจารย์ ดร.พรศิริ หมั่นไชยศรี ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์ และผู้ช่วยศาสตราจารย์ ดร.ภัทรชัย ลลิตโรจน์วงศ์ กรรมการสอบวิทยานิพนธ์ ที่ได้กรุณาสละเวลาและให้คำแนะนำเกี่ยวกับการทำวิจัย เพื่อให้วิทยานิพนธ์ฉบับนี้มีความสมบูรณ์และสำเร็จลุล่วงไปด้วยดี

ขอขอบคุณคณาจารย์ทุกท่านในภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ที่ได้มอบวิชาความรู้ที่เป็นประโยชน์ ระหว่างที่ผู้วิจัยกำลังศึกษา ตลอดจนสอบวิทยานิพนธ์สำเร็จลุล่วง

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ที่ให้คำแนะนำช่วยเหลือ และให้กำลังใจเสมอมา

ขอขอบคุณบิดา มารดา และญาติพี่น้องของผู้วิจัย ที่ให้ความสนับสนุน และให้กำลังใจแก่ผู้วิจัยเสมอมา

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ญ
สารบัญรูป.....	ฐ
บทที่ 1 บทนำ	1
1.1. ที่มาและความสำคัญของปัญหา.....	1
1.2. วัตถุประสงค์ของงานวิจัย.....	3
1.3. ขอบเขตการดำเนินงาน.....	3
1.4. ขั้นตอนการดำเนินงาน.....	4
1.5. ประโยชน์ที่คาดว่าจะได้รับ.....	4
1.6. บทความวิชาการที่ได้รับการตีพิมพ์.....	4
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	5
2.1. ทฤษฎีที่เกี่ยวข้อง.....	5
2.1.1 ดับเบิลยูเอส-พีเพิล	5
2.1.2 พาร์ทเนอร์ลิงก์.....	6
2.1.3 บีเพิลเอนจิน.....	6
2.1.4 เอกซ์เอสดี (XSD - XML Schema Definition).....	8
2.2. งานวิจัยที่เกี่ยวข้อง	9
2.2.1 งานวิจัย “Calculating BPEL Test Coverage through Instrumentation” [1]....	9

2.2.2 งานวิจัย “Automatic Test Case Selection and Generation for Regression Testing of Composite Service Based on Extensible BPEL Flow Graph” [2].....	10
2.2.3 งานวิจัย “Automated Testing of WS-BPEL Service Compositions: A Scenario-Oriented Approach” [3].....	10
บทที่ 3 วิธีการตรวจจับการเรียกใช้เว็บเซอร์วิสสำหรับทดสอบดับเบิลยูเอส-บีเพล	11
3.1 การสร้างเส้นทางการไหลของบีเพล	13
3.2 การแทรกรหัสต้นทาง	15
3.3 การดำเนินการทดสอบตามกรณีทดสอบ	18
3.4 การค้นหาเว็บเซอร์วิสที่ยังไม่ถูกทดสอบ.....	20
3.5 การสร้างกรณีทดสอบเพิ่มเติม	21
3.6 การสร้างรายงานผลการทดสอบ	23
บทที่ 4 การออกแบบและพัฒนาเครื่องมือ.....	25
4.1. การออกแบบเครื่องมือ.....	25
4.1.1. แผนภาพยูสเคส.....	25
4.1.2. แผนภาพกิจกรรม.....	26
4.1.3. แผนภาพคลาส	33
4.1.4. แผนภาพลำดับ.....	61
4.1.5. โครงสร้างของฐานข้อมูล.....	69
4.1.6. แผนภาพการติดตั้ง	70
4.2. การพัฒนาเครื่องมือสนับสนุน.....	71
4.2.1. สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ.....	71
4.2.2. โครงสร้างส่วนต่อประสานกับผู้ใช้ของเครื่องมือสนับสนุน	72
บทที่ 5 การทดสอบเครื่องมือ	76

5.1. การทดสอบเครื่องมือ	76
5.1.1. กรณีศึกษาที่ 1 ระบบออกบัตรคิวผู้ป่วยนอก	76
5.1.2. กรณีศึกษาที่ 2 ระบบค้นหาเที่ยวบิน ที่พัก และรถยนต์.....	87
5.1.3. กรณีศึกษาที่ 3 ระบบชำระเงินผู้ป่วยนอก	96
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ	106
6.1. สรุปผลการวิจัย.....	106
6.2. ข้อจำกัดของงานวิจัย	106
6.3. ข้อเสนอแนะและแนวทางการดำเนินงานต่อ	107
รายการอ้างอิง	108
ภาคผนวก.....	109
ภาคผนวก ก รายละเอียดคุณสมบัติของเครื่องมือ	110
ภาคผนวก ข ตัวอย่างการทดสอบเครื่องมือตามกรณีศึกษา	115
ภาคผนวก ค พจนานุกรมข้อมูล	135
ประวัติผู้เขียนวิทยานิพนธ์	143

สารบัญตาราง

ตารางที่ 2-1 การจำแนกโหนดของกราฟการไหลของบีเพลที่ถูกขยาย	10
ตารางที่ 4-1 CRC ของคลาส frm_readInstrument	35
ตารางที่ 4-2 CRC ของคลาส frm_viewTestDetail.....	36
ตารางที่ 4-3 CRC ของคลาส frm_BPELList	38
ตารางที่ 4-4 CRC ของคลาส frm_viewTestHistory.....	39
ตารางที่ 4-5 CRC ของคลาส frm_rptSummary.....	39
ตารางที่ 4-6 CRC ของคลาส frm_rptHistory.....	40
ตารางที่ 4-7 CRC ของคลาส facade_instrument.....	41
ตารางที่ 4-8 CRC ของคลาส facade_testcase	42
ตารางที่ 4-9 CRC ของคลาส facade_xsd	42
ตารางที่ 4-10 CRC ของคลาส ctrl_XSD.....	43
ตารางที่ 4-11 CRC ของคลาส ctrl_ReceiveNodeVarSchema	44
ตารางที่ 4-12 CRC ของคลาส ctrl_InstrumentComposite.....	44
ตารางที่ 4-13 CRC ของคลาส ctrl_InstrumentWSDL	45
ตารางที่ 4-14 CRC ของคลาส ctrl_InstrumentBPEL.....	46
ตารางที่ 4-15 CRC ของคลาส ctrl_TestCase.....	47
ตารางที่ 4-16 CRC ของคลาส ctrl_BPEL_Expression	48
ตารางที่ 4-17 CRC ของคลาส ctrl_BPEL_ORAC	49
ตารางที่ 4-18 CRC ของคลาส ctrl_BPEL_Predicate	49
ตารางที่ 4-19 CRC ของคลาส ctrl_TestResult	50
ตารางที่ 4-20 CRC ของคลาส Ctrl_BPEL.....	51

ตารางที่ 4-21 CRC ของคลาส ctrl_BPEL_Node	51
ตารางที่ 4-22 CRC ของคลาส ctrl_BPEL_DB	53
ตารางที่ 4-23 CRC ของคลาส ctrls_DB_node	54
ตารางที่ 4-24 CRC ของคลาส services_ReceiveNodeVarSchema	54
ตารางที่ 4-25 CRC ของคลาส services_TestCase	55
ตารางที่ 4-26 CRC ของคลาส services_TestResult	56
ตารางที่ 4-27 CRC ของคลาส services_BPEL_DB	56
ตารางที่ 4-28 CRC ของคลาส services_Node	58
ตารางที่ 4-29 CRC ของคลาส services_Serial	59
ตารางที่ 4-30 CRC ของคลาส Ctrls_config	59
ตารางที่ 4-31 CRC ของคลาส Ctrls_Files	60
ตารางที่ 5-1 รายการเว็บเซอร์วิสที่เกี่ยวข้องกับแผนภาพบีเพล	76
ตารางที่ 5-2 รายการเว็บเซอร์วิสที่เกี่ยวข้องกับแผนภาพบีเพล	89
ตารางที่ 5-3 รายการเว็บเซอร์วิสที่เกี่ยวข้องกับแผนภาพบีเพล	99
ตารางที่ ก-1 คำอธิบายยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเพล	110
ตารางที่ ก-2 คำอธิบายยูสเคสดูเว็บเซอร์วิสที่ไม่ถูกต้องสอบ	111
ตารางที่ ก-3 คำอธิบายยูสเคสร่างกรณีทดสอบเพิ่มเติม	112
ตารางที่ ก-4 คำอธิบายยูสเคสดูรายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส	113
ตารางที่ ก-5 คำอธิบายยูสเคสดูรายงานผลการทดสอบการเรียกใช้เว็บเซอร์วิส	114
ตารางที่ ข-1 รายการเว็บเซอร์วิสที่เกี่ยวข้องกับกรณีศึกษาที่ 1	115
ตารางที่ ข-2 สรุปผลการทดสอบกรณีศึกษาที่ 1	120
ตารางที่ ข-3 รายการเว็บเซอร์วิสที่เกี่ยวข้องกับกรณีศึกษาที่ 2	121
ตารางที่ ข-4 สรุปผลการทดสอบกรณีศึกษาที่ 2	127

ตารางที่ ข-5 รายการเว็บไซต์ที่เกี่ยวข้องกับกรณีศึกษาที่ 3	128
ตารางที่ ข-6 สรุปผลการทดสอบกรณีศึกษาที่ 3.....	133
ตารางที่ ค-1 พจนานุกรมข้อมูลในตาราง TBL_TASK.....	135
ตารางที่ ค-2 พจนานุกรมข้อมูลในตาราง TBL_NodeMaster	135
ตารางที่ ค-3 พจนานุกรมข้อมูลในตาราง TBL_NodeDetail	136
ตารางที่ ค-4 พจนานุกรมข้อมูลในตาราง TBL_ReceiveNodeVar_Schema	138
ตารางที่ ค-5 พจนานุกรมข้อมูลในตาราง TBL_XSD_Restriction	139
ตารางที่ ค-6 พจนานุกรมข้อมูลในตาราง TBL_TESTCASE	140
ตารางที่ ค-7 พจนานุกรมข้อมูลในตาราง TBL_TESTCASE_TEMPLATE	140
ตารางที่ ค-8 พจนานุกรมข้อมูลในตาราง WS_LOG.....	141
ตารางที่ ค-9 พจนานุกรมข้อมูลในตาราง SERIAL	142

สารบัญรูป

รูปที่ 2-1 ตัวอย่างไฟล์ดับเบิลยูเอส-บีเพล	5
รูปที่ 2-2 ตัวอย่างแท็กพาร์ทเนอร์ลิงก์ [8].....	6
รูปที่ 2-3 ตัวอย่างหน้าจอการออกแบบกระบวนการ	7
รูปที่ 2-4 การแสดงข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิสจากโหนด call_getPatient	7
รูปที่ 2-5 การแสดงเส้นทางการทดสอบของเครื่องมือ โดยจะแสดงเฉพาะโหนดที่ถูกทดสอบ	8
รูปที่ 2-6 ตัวอย่างแท็ก <element> สำหรับอธิบายโครงสร้างข้อมูล.....	8
รูปที่ 2-7 ตัวอย่างแท็ก <attribute> สำหรับอธิบายโครงสร้างข้อมูล.....	8
รูปที่ 2-8 ตัวอย่างแท็ก <restriction>.....	9
รูปที่ 3-1 ภาพรวมการทำงาน.....	11
รูปที่ 3-2 ภาพรวมเชิงองค์ประกอบของวิธีการ.....	12
รูปที่ 3-3 การสร้างเส้นทางการไหลของบีเพล.....	13
รูปที่ 3-4 ตัวอย่างไฟล์ดับเบิลยูเอส-บีเพล	14
รูปที่ 3-5 ตัวอย่างเส้นทางการไหลของดับเบิลยูเอส-บีเพล.....	14
รูปที่ 3-6 ตัวอย่างไฟล์ดับเบิลยูเอสดีแอล	15
รูปที่ 3-7 ตัวอย่างไฟล์เอกซ์เอสดี	15
รูปที่ 3-8 ตัวอย่างไฟล์คอมโพสิตที่ถูกแทรกรหัสต้นทาง	15
รูปที่ 3-9 ตัวอย่างไฟล์ดับเบิลยูเอสดีแอลของเว็บเซอร์วิสที่จัดเก็บข้อมูล	16
รูปที่ 3-10 การแทรกรหัสต้นทาง	16
รูปที่ 3-11 ตัวอย่างเส้นทางการไหลของบีเพลที่ถูกแทรกรหัสต้นทางแล้ว.....	18
รูปที่ 3-12 การดำเนินการทดสอบตามกรณีทดสอบ	19
รูปที่ 3-13 ตัวอย่างกรณีทดสอบ	19

รูปที่ 3-14 โหนดที่ถูกทดสอบ.....	20
รูปที่ 3-15 การค้นหาเว็บเซอร์วิชที่ยังไม่ถูกทดสอบ.....	21
รูปที่ 3-16 การสร้างกรณีทดสอบเพิ่มเติม.....	22
รูปที่ 3-17 เส้นทางการทดสอบใหม่.....	23
รูปที่ 3-18 ตัวอย่างกรณีทดสอบใหม่.....	23
รูปที่ 4-1 แผนภาพยูสเคส เครื่องมือตรวจจับการเรียกใช้เว็บเซอร์วิชสำหรับทดสอบ ดับเบิลยู เอส-บีเพล.....	26
รูปที่ 4-2 แผนภาพกิจกรรมการจับเส้นทางการไหลและแทรกรหัสต้นทาง ในขั้นตอนการ ค้นหาไฟล์.....	27
รูปที่ 4-3 แผนภาพกิจกรรมการจับเส้นทางการไหลและแทรกรหัสต้นทาง ในขั้นตอนการ จับเส้นทางการไหลและแทรกรหัสต้นทาง.....	28
รูปที่ 4-4 แผนภาพกิจกรรมการจับผลการทดสอบ.....	29
รูปที่ 4-5 แผนภาพกิจกรรมการดูผลการทดสอบ.....	30
รูปที่ 4-6 แผนภาพกิจกรรมการสร้างกรณีทดสอบเพิ่มเติม ในขั้นตอนการสร้างกรณีทดสอบ.....	31
รูปที่ 4-7 แผนภาพกิจกรรมการสร้างกรณีทดสอบเพิ่มเติม ในขั้นตอนการบันทึกกรณีทดสอบใหม่ ..	32
รูปที่ 4-8 แผนภาพกิจกรรมการดูรายงานสรุปผลการทดสอบ.....	33
รูปที่ 4-9 แผนภาพคลาส.....	34
รูปที่ 4-10 แผนภาพลำดับ ของยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเพล (ในขั้นตอนการอ่านข้อมูลดับเบิลยูเอส-บีเพล).....	62
รูปที่ 4-11 แผนภาพลำดับ ของยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเพล (ในขั้นตอนการบันทึกเส้นทางการไหลของดับเบิลยูเอส-บีเพล).....	63
รูปที่ 4-12 แผนภาพลำดับ ของยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเพล (ในขั้นตอนการสร้างโครงสร้างข้อมูลจากไฟล์เอกซ์เอสดี).....	64
รูปที่ 4-13 แผนภาพลำดับ ของยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเพล (ในขั้นตอนการบันทึกโครงสร้างข้อมูลจากไฟล์เอกซ์เอสดี).....	64

รูปที่ 4-14 แผนภาพลำดับ ของยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิ้ลยูเอส-บีเพล (ในขั้นตอนการแทรกรหัสต้นทางที่ไฟล์คอมโพสิตและคัดลอกไฟล์ดับเบิ้ลยูเอสดี).....	65
รูปที่ 4-15 แผนภาพลำดับ ของยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิ้ลยูเอส-บีเพล (ในขั้นตอนการแทรกรหัสต้นทางที่ไฟล์ดับเบิ้ลยูเอส-บีเพล)	66
รูปที่ 4-16 แผนภาพลำดับ ของยูสเคสดูเว็บเซอร์วิซที่ไม่ถูกทดสอบ	67
รูปที่ 4-17 แผนภาพลำดับ ของยูสเคสสร้างกรณีทดสอบเพิ่มเติม (ในขั้นตอนอ่านโน้ตเงื่อนไขของเว็บเซอร์วิซที่ไม่ถูกทดสอบ).....	68
รูปที่ 4-18 แผนภาพลำดับ ของยูสเคสสร้างกรณีทดสอบเพิ่มเติม(ในขั้นตอนการสร้างกรณีทดสอบ)	68
รูปที่ 4-19 แผนภาพลำดับ ของยูสเคสสร้างกรณีทดสอบเพิ่มเติม(ในขั้นตอนบันทึกและเขียนไฟล์ทดสอบเพิ่มเติม).....	69
รูปที่ 4-20 แผนภาพโครงสร้างฐานข้อมูลของเครื่องมือ	70
รูปที่ 4-21 แผนภาพการติดตั้งเครื่องมือ	71
รูปที่ 4-22 แผนภาพวินโดวส์เนวิกาชันของเครื่องมือ	72
รูปที่ 4-23 หน้าต่างอัปโหลดไฟล์และแทรกรหัสต้นทาง.....	73
รูปที่ 4-24 หน้าต่างสำหรับแสดงรายการบีเพลที่ถูกจัดเก็บในระบบ	73
รูปที่ 4-25 หน้าต่างแสดงรายละเอียดบีเพล	74
รูปที่ 4-26 หน้าต่างดูผลการทดสอบ	74
รูปที่ 4-27 หน้าต่างแสดงรายงานผลสรุปการทดสอบ.....	75
รูปที่ 4-28 หน้าต่างแสดงรายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิซ	75
รูปที่ 5-1 แผนภาพดับเบิ้ลยูเอส-บีเพลสำหรับกรณีศึกษาที่ 1 ในขั้นตอนการตรวจสอบข้อมูลผู้ป่วย	77
รูปที่ 5-2 แผนภาพดับเบิ้ลยูเอส-บีเพลสำหรับกรณีศึกษาที่ 1 ในขั้นตอนการออกบัตรคิวผู้ป่วย.....	78
รูปที่ 5-3 แผนภาพดับเบิ้ลยูเอส-บีเพล ที่ถูกแทรกหัสต้นทางของกรณีศึกษาที่ 1 ในขั้นตอนการเรียกข้อมูลผู้ป่วย	79

รูปที่ 5-4 แผนภาพดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางของกรณีศึกษาที่ 1 ในขั้นตอนการเรียกข้อมูลสิทธิการรักษา	80
รูปที่ 5-5 แผนภาพดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางของกรณีศึกษาที่ 1 ในขั้นตอนการเรียกข้อมูลความเร่งด่วน.....	81
รูปที่ 5-6 แผนภาพดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางของกรณีศึกษาที่ 1 ในขั้นตอนการเรียกข้อมูลแผนก.....	82
รูปที่ 5-7 แผนภาพดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางของกรณีศึกษาที่ 1 ในขั้นตอนการเรียกข้อมูลคิวตรวจรักษาผู้ป่วย	83
รูปที่ 5-8 รายการเว็บเซอร์วิสที่ไม่ถูกทดสอบ	84
รูปที่ 5-9 ข้อมูลนำเข้าและข้อมูลส่งออกที่รับส่งระหว่างเว็บเซอร์วิส	85
รูปที่ 5-10 รายงานสรุปผลการทดสอบสำหรับกรณีศึกษาที่ 1	85
รูปที่ 5-11 รายงานผลการทดสอบสำหรับกรณีศึกษาที่ 1	86
รูปที่ 5-12 เว็บเซอร์วิสภายใต้ดับเบิลยูเอส-บีเพล ของกรณีศึกษาที่ 1 ถูกทดสอบทั้งหมด	86
รูปที่ 5-13 รายงานสรุปผลการทดสอบหลังจากนำกรณีทดสอบใหม่ไปใช้งานกรณีศึกษาที่ 1.....	87
รูปที่ 5-14 แผนภาพดับเบิลยูเอส-บีเพลสำหรับกรณีศึกษาที่ 2.....	88
รูปที่ 5-15 ไฟล์ดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสสำหรับกรณีศึกษาที่ 2 ในขั้นตอนการเรียกข้อมูลเที่ยวบิน	90
รูปที่ 5-16 ไฟล์ดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสสำหรับกรณีศึกษาที่ 2 ในขั้นตอนการเรียกข้อมูลที่พัก.....	91
รูปที่ 5-17 ไฟล์ดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสสำหรับกรณีศึกษาที่ 2 ในขั้นตอนการเรียกข้อมูลรถยนต์.....	92
รูปที่ 5-18 เว็บเซอร์วิสที่ไม่ถูกทดสอบหลังจากนำกรณีทดสอบที่มีอยู่มาทดสอบ	93
รูปที่ 5-19 ตัวอย่างการแสดงข้อมูลส่งออกที่อยู่ในรูปแบบรายการ	93
รูปที่ 5-20 ข้อความเตือน การสร้างกรณีทดสอบสำเร็จ	94
รูปที่ 5-21 ไฟล์กรณีทดสอบใหม่ที่ได้.....	94

รูปที่ 5-22 อัฟโพลตกรณีทดสอบใหม่ที่ได้นบนเครื่องมือปีเพลเอนจิน	95
รูปที่ 5-23 เว็บเซอร์วิซภายใต้ดับเบิ้ลยูเอส-ปีเพล ของกรณีศึกษาที่ 2 ถูกทดสอบทั้งหมด	95
รูปที่ 5-24 รายงานสรุปผลการทดสอบหลังจากนำกรณีทดสอบใหม่ไปใช้งานกรณีศึกษาที่ 2.....	96
รูปที่ 5-25 แผนภาพดับเบิ้ลยูเอส-ปีเพลสำหรับกรณีศึกษาที่ 3 ในขั้นตอนตรวจสอบวิธีการชำระ เงิน	97
รูปที่ 5-26 แผนภาพดับเบิ้ลยูเอส-ปีเพลสำหรับกรณีศึกษาระบบที่ 3 ในขั้นตอนชำระเงินและรับ ยาผู้ป่วย	98
รูปที่ 5-27 ไฟล์ดับเบิ้ลยูเอส-ปีเพลที่ถูกแทรกรหัสต้นทาง สำหรับกรณีศึกษาที่ 3 ในขั้นตอนการ ตรวจสอบวิธีการชำระเงิน.....	100
รูปที่ 5-28 ไฟล์ดับเบิ้ลยูเอส-ปีเพลที่ถูกแทรกรหัสต้นทาง สำหรับกรณีศึกษาที่ 3 ในขั้นตอนการ ชำระเงินผู้ป่วย.....	101
รูปที่ 5-29 ไฟล์ดับเบิ้ลยูเอส-ปีเพลที่ถูกแทรกรหัสต้นทาง สำหรับกรณีศึกษาที่ 3 ในขั้นตอนการ ชำระเงินและพิมพ์ใบเสร็จ	102
รูปที่ 5-30 ไฟล์ดับเบิ้ลยูเอส-ปีเพลที่ถูกแทรกรหัสต้นทาง สำหรับกรณีศึกษาที่ 3 ในขั้นตอนการ รับยาผู้ป่วย.....	103
รูปที่ 5-31 ผลการทดสอบโดยใช้กรณีทดสอบใหม่ที่ถูกสร้างขึ้นจากเครื่องมือ	104
รูปที่ 5-32 ตัวอย่างข้อมูลนำเข้าและข้อมูลส่งออกในแต่ละรอบของการทำงานแบบวนซ้ำ	104
รูปที่ 5-33 ตัวอย่างรายงานสรุปผลการทดสอบ	105
รูปที่ ข-1 ขั้นตอนการอ่านและแทรกรหัสต้นทางสำเร็จ	116
รูปที่ ข-2 ขั้นตอนการระบุกรณีทดสอบผ่านเครื่องมือปีเพลเอนจิน	117
รูปที่ ข-3 ข้อมูลเว็บเซอร์วิซที่ไม่ถูกทดสอบ	117
รูปที่ ข-4 ข้อมูลนำเข้าและข้อมูลส่งออกที่รับส่งระหว่างเว็บเซอร์วิซ	118
รูปที่ ข-5 รายงานสรุปผลการทดสอบสำหรับกรณีศึกษาที่ 1	118
รูปที่ ข-6 รายงานผลการทดสอบสำหรับกรณีศึกษาที่ 1	119
รูปที่ ข-7 การนำกรณีทดสอบผู้ป่วยใหม่มาใช้ทดสอบผ่านเครื่องมือปีเพลเอนจิน.....	119

รูปที่ ข-8 เว็บไซต์วีซีกายใต้ดับเบิลยูเอส-บีเพล ของกรณีศึกษาที่ 1 ถูกทดสอบทั้งหมด	120
รูปที่ ข-9 รายงานสรุปผลการทดสอบหลังจากนำกรณีทดสอบผู้ป่วยใหม่ไปทดสอบ	120
รูปที่ ข-10 ขั้นตอนการอ่านและแทรกรหัสต้นทาง.....	122
รูปที่ ข-11 ขั้นตอนการระบุกรณีทดสอบผ่านเครื่องมือบีเพลเอนจิน	123
รูปที่ ข-12 เว็บไซต์วีซีที่ไม่ถูกทดสอบหลังจากนำกรณีทดสอบการจองเที่ยวบินและโรงแรมมาทดสอบ	123
รูปที่ ข-13 ตัวอย่างการแสดงผลข้อมูลส่งออกที่อยู่ในรูปแบบรายการ	124
รูปที่ ข-14 ตัวอย่างรายงานข้อมูลส่งออกจากกรณีทดสอบการจองเที่ยวบินและโรงแรม.....	124
รูปที่ ข-15 ข้อความเตือน การสร้างกรณีทดสอบสำเร็จ	125
รูปที่ ข-16 ไฟล์กรณีทดสอบใหม่ที่ได้.....	125
รูปที่ ข-17 อัปเดตกรณีทดสอบใหม่ที่ได้นบนเครื่องมือบีเพลเอนจิน	126
รูปที่ ข-18 ผลการทดสอบหลังจากที่นำกรณีทดสอบ ที่เครื่องมือสร้างขึ้นมาทดสอบผ่านเครื่องมือบีเพลเอนจิน.....	126
รูปที่ ข-19 รายงานสรุปผลการทดสอบ.....	127
รูปที่ ข-20 ขั้นตอนการอ่านและแทรกรหัสต้นทาง	129
รูปที่ ข-21 ข้อมูลเว็บไซต์วีซีที่ไม่ถูกทดสอบ	129
รูปที่ ข-22 ขั้นตอนการสร้างกรณีทดสอบเพิ่มเติม เครื่องมือแสดงข้อความแจ้งเตือนสร้างกรณีทดสอบเพิ่มเติมสำเร็จ	130
รูปที่ ข-23 ขั้นตอนการทดสอบผ่านเครื่องมือบีเพลเอนจิน	130
รูปที่ ข-24 ผลการทดสอบหลังจากที่นำกรณีทดสอบ ที่เครื่องมือสร้างขึ้นมาทดสอบผ่านเครื่องมือบีเพลเอนจิน.....	131
รูปที่ ข-25 ตัวอย่างข้อมูลนำเข้าและข้อมูลส่งออกที่ในแต่ละรอบของการทำงานแบบวนซ้ำ	131
รูปที่ ข-26 รายงานผลการทดสอบ ข้อมูลนำเข้าและข้อมูลส่งออก ที่รับส่งระหว่างเว็บไซต์วีซี.....	132
รูปที่ ข-27 รายงานสรุปผลการทดสอบ.....	132

บทที่ 1

บทนำ

1.1. ที่มาและความสำคัญของปัญหา

การออกแบบและพัฒนาซอฟต์แวร์เชิงบริการ (SOA - Service-Oriented Architecture) นั้นถูกนำมาใช้งานในการผนวกรวมเว็บเซอร์วิสที่มีอยู่เข้าไว้ด้วยกัน เพื่อลดเวลาและความซ้ำซ้อนในการพัฒนาเว็บเซอร์วิสตามกระบวนการทางธุรกิจขององค์กร ซึ่งในกระบวนการทางธุรกิจที่ออกแบบอาจมีการติดต่อเรียกใช้งานกับเว็บเซอร์วิสจำนวนมาก ดังนั้นเพื่อให้ผู้ออกแบบกระบวนการเห็นถึงองค์ประกอบและภาพรวมการทำงานภายในกระบวนการนั้น จำเป็นต้องใช้เครื่องมือมาช่วยสำหรับออกแบบและจำลองการทำงานของกระบวนการ โดยเครื่องมือที่สนับสนุนการทำงานดังกล่าว เช่น เครื่องมือประเภทบีเพลเอนจิน (BPEL Engine) เป็นเครื่องมือที่มีความสามารถต่างๆ เช่น การแสดงภาพรวมของกระบวนการทางธุรกิจ การออกแบบกระบวนการ การทดสอบการทำงาน เป็นต้น โดยความสามารถของเครื่องมืออาจจะมีความแตกต่างกัน ซึ่งขึ้นอยู่กับผู้พัฒนาเครื่องมือรายนั้นๆ เครื่องมือบีเพลเอนจินทำงานโดยใช้ภาษา ดับเบิลยูเอส-บีเพล (WS-BPEL – Web Services Business Process Execution Language) ซึ่งเป็นภาษาที่รองรับการออกแบบกระบวนการและเรียกใช้งานซอฟต์แวร์ในรูปแบบของเว็บเซอร์วิส เพื่อตอบสนองกระบวนการธุรกิจที่องค์กรต้องการในการเชื่อมต่อกับเว็บเซอร์วิสภายนอกดับเบิลยูเอส-บีเพลจำเป็นต้องใช้พาร์ตเนอร์ลิงก์ (Partner link) เพื่ออ้างอิงไปยังเว็บเซอร์วิสได้ โดยข้อดีของเว็บเซอร์วิสคือ องค์กรสามารถผนวกเซอร์วิสซึ่งพัฒนาโดยองค์กรอื่นเข้ามาใช้งานร่วมกับระบบงานขององค์กรได้ เพื่อลดเวลาและต้นทุนในการพัฒนาซอฟต์แวร์

เนื่องจากการนำเว็บเซอร์วิสมาใช้ในองค์กรมีบทบาทเพิ่มขึ้น จึงทำให้การทดสอบเว็บเซอร์วิสมีความสำคัญต่อการพัฒนาซอฟต์แวร์ เพื่อทำให้เกิดความมั่นใจว่าซอฟต์แวร์ในส่วนต่างๆ สามารถทำงานร่วมกันได้อย่างถูกต้อง ในการทดสอบการทำงานร่วมกันของเว็บเซอร์วิสที่ดีควรมีการทดสอบที่ครอบคลุมการเรียกใช้ทุกเว็บเซอร์วิสที่ปรากฏในกระบวนการทางธุรกิจนั้นๆ เพื่อลดความผิดพลาดที่อาจเกิดขึ้นในการนำระบบไปใช้งานจริง นอกจากการทดสอบการทำงานร่วมกันยังควรทดสอบเพื่อตรวจสอบข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิสและข้อมูลส่งออกที่ได้จากเว็บเซอร์วิสที่อยู่ในกระบวนการทางธุรกิจ เพื่อให้เกิดความมั่นใจระหว่างผู้พัฒนาเว็บเซอร์วิสและผู้เรียกใช้เว็บเซอร์วิสว่าเว็บเซอร์วิสที่เรียกใช้สามารถรับข้อมูลนำเข้าและคืนข้อมูลส่งออกได้อย่างถูกต้อง

จากการนำดับเบิลยูเอส-บีเพล และบีเพลเอนจินเข้ามามีบทบาทในธุรกิจจึงทำให้เกิดการแข่งขันเพื่อพัฒนาความสามารถของเครื่องมือบีเพลเอนจินในด้านต่างๆ เช่น การตรวจสอบประสิทธิภาพการทำงาน การออกแบบและใช้งานเครื่องมือที่ง่าย การช่วยติดตั้งไฟล์ดับเบิลยูเอส-บีเพลบนเครื่องแม่ข่าย เป็นต้น และนอกจากนี้ยังมีผู้วิจัยเข้ามาวิจัยการทดสอบการทำงานของดับเบิลยูเอส-บีเพล โดยได้นำเสนอวิธีการต่างๆ เช่น วิธีการแทรกรหัสต้นทางเพื่อตรวจจับความครอบคลุมในการทดสอบการทำงานภายในบีเพล [1] ซึ่งการแทรกรหัสต้นทางจะทำเพื่อตรวจหาเส้นทางที่ถูกทดสอบแล้ว แต่ยังไม่ได้มุ่งเน้นการตรวจหาเว็บเซอร์วิซที่ยังไม่ถูกทดสอบ ทำให้ในการตรวจสอบการทำงานร่วมกันของเว็บเซอร์วิซที่ยังไม่ถูกทดสอบ จะต้องนำข้อมูลที่ได้มาตรวจหาโหนดที่มีการเรียกใช้งานเว็บเซอร์วิซเพื่อให้ทราบได้ว่าเว็บเซอร์วิซใดบ้างที่ยังไม่ถูกทดสอบ และในงานวิจัยอื่นๆ ได้วิจัยการสร้างกรณีทดสอบเพื่อให้ครอบคลุมการทำงานของบีเพลโดยใช้วิธีการสุ่มข้อมูลและการทดสอบตามเงื่อนไขของเส้นทางไหลของบีเพล [2, 3] โดยมุ่งเน้นทุกเส้นทางไหลของบีเพลจึงทำให้ในกรณีที่ผู้ทดสอบต้องการทราบเส้นทางไหลที่ยังไม่ครอบคลุมการเรียกใช้งานเว็บเซอร์วิซ จะต้องนำข้อมูลนั้นมาวิเคราะห์เพื่อคัดกรองเฉพาะเส้นทางที่มีการเรียกใช้เว็บเซอร์วิซ เป็นต้น บีเพลเอนจิน ยังถูกพัฒนาโดยองค์กรต่างๆ เช่น เครื่องมือ ออราเคิล บีเพล โพรเซส เมเนเจอร์ (Oracle BPEL Process Manager) เป็นบีเพลเอนจินที่ถูกพัฒนาโดยบริษัท ออราเคิล (Oracle) [4] โดยมีความสามารถในการสร้างกระบวนการทางธุรกิจและสามารถทดสอบกระบวนการที่สร้างขึ้นได้ และในขั้นตอนการทดสอบเครื่องมือ ออราเคิล บีเพล โพรเซส เมเนเจอร์สามารถแสดง ข้อมูลการทดสอบในแต่ละโหนดที่ถูกทดสอบได้ โดยผู้ทดสอบสามารถดูข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิซและข้อมูลส่งออกที่ได้รับจากเว็บเซอร์วิซจากการเลือกดูโหนดที่สนใจ โดยเครื่องมือจะแสดงในรูปแบบของเอกซ์เอ็มแอล ซึ่งผู้ทดสอบจำเป็นต้องมีความรู้ด้านภาษาเอกซ์เอ็มแอล เพื่ออ่านข้อมูลดังกล่าว และนอกจากนี้ในการแสดงผลการทดสอบเครื่องมือ จะแสดงเฉพาะโหนดที่ถูกทดสอบเท่านั้น แต่จะไม่ได้แสดงโหนดที่ยังไม่ถูกทดสอบ ทำให้ผู้ทดสอบอาจไม่ทราบว่าในการทดสอบมีเว็บเซอร์วิซใดบ้างที่ยังไม่ถูกเรียกใช้งานและอาจจะส่งผลให้การสร้างกรณีทดสอบเพิ่มเติมไม่ครอบคลุมทุกเว็บเซอร์วิซ

จากปัญหาข้างต้น วิทยานิพนธ์นี้มีจุดประสงค์เพื่อนำเสนอวิธีการและพัฒนาเครื่องมือตรวจจับการเรียกใช้เว็บเซอร์วิซสำหรับทดสอบดับเบิลยูเอส-บีเพล เพื่อตรวจจับข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิซและข้อมูลส่งออกที่ได้รับจากเว็บเซอร์วิซ และตรวจสอบความครอบคลุมของการทดสอบเว็บเซอร์วิซว่าการทดสอบที่ครอบคลุมทุกเว็บเซอร์วิซหรือไม่ ในกรณีที่พบเว็บเซอร์วิซที่ยังไม่ถูกทดสอบ จะทำการสร้างกรณีทดสอบเพิ่มเติมโดยอ้างอิงจากโหนดเงื่อนไขที่ยังไม่ถูกทดสอบ

1.2. วัตถุประสงค์ของงานวิจัย

เพื่อเสนอวิธีการและพัฒนาเครื่องมือตรวจจับการเรียกใช้เว็บเซอร์วิสสำหรับทดสอบ
 ดับเบิลยูเอส-พีเพิล

1.3. ขอบเขตการดำเนินงาน

1. ออกแบบและพัฒนาเครื่องมือตรวจจับการเรียกใช้เว็บเซอร์วิสสำหรับทดสอบ
 ดับเบิลยูเอส-พีเพิล โดยมีความสามารถดังต่อไปนี้ เป็นอย่างน้อย
 - 1) สามารถอัปโหลดไฟล์ดับเบิลยูเอส-พีเพิล เพื่อแทรกห้สตันทาง สำหรับการจัดเก็บ
 ข้อมูลนำเข้าและข้อมูลส่งออกของเว็บเซอร์วิสที่ถูกเรียกใช้ได้
 - 2) สามารถอัปโหลดไฟล์ดับเบิลยูเอส-พีเพิล เพื่อจัดเก็บเส้นทางการไหลของบีเพิลลงสู่
 ฐานข้อมูลของเครื่องมือได้
 - 3) สามารถจัดเก็บผลการทดสอบเว็บเซอร์วิสได้ โดยจะจัดเก็บข้อมูลนำเข้าและข้อมูล
 ส่งออกในรูปแบบของเท็กซ์ เข้าสู่ฐานข้อมูล
 - 4) สามารถรายงานเว็บเซอร์วิสในเส้นทางการไหลของบีเพิล ที่ยังไม่ถูกทดสอบได้
 - 5) การสร้างกรณีทดสอบเพิ่มเติม จะทำได้เมื่อโหนดเงื่อนไขนั้น ใช้ตัวแปรที่เป็นข้อมูล
 นำเข้า และตัวแปรนั้นจะต้องไม่มีการเปลี่ยนแปลงค่าภายในไฟล์ดับเบิลยูเอส-
 บีเพิล
 - 6) การแก้ไขข้อมูลเพื่อสร้างกรณีทดสอบเพิ่มเติมของข้อมูลประเภทพื้นฐาน (simple
 type) จะรองรับประเภทข้อมูลที่เป็น ตัวเลขจำนวนเต็ม (Integer) ตัวเลขทศนิยม
 (Float) ตัวอักษร (String) และ บูลีน (Boolean)
 - 7) การแก้ไขข้อมูลเพื่อสร้างกรณีทดสอบเพิ่มเติมของข้อมูลประเภทซับซ้อน
 (complex type) จะรองรับข้อมูลที่ประกอบด้วยข้อมูลประเภทพื้นฐานที่เป็น
 ตัวเลขจำนวนเต็ม ตัวเลขทศนิยม ตัวอักษร และ บูลีน
2. ไฟล์ดับเบิลยูเอสดีแอล ที่นำมาใช้งานจะต้องมีการกำหนดโครงสร้างของข้อมูลนำเข้าและ
 ข้อมูลส่งออก ที่ใช้รับส่งผ่านเว็บเซอร์วิส
3. การทดลองจะทำได้โดยใช้กรณีศึกษาอย่างน้อย 3 กรณีศึกษา โดยกำหนดให้ในแต่ละกรณีศึกษา
 จะต้องมีเว็บเซอร์วิสอย่างน้อย 3 เว็บเซอร์วิส

1.4. ขั้นตอนการดำเนินงาน

1. ศึกษาการทำงานของบีเพล
2. ศึกษาการสร้างกรณีทดสอบ
3. ศึกษาเครื่องมือสำหรับการทดสอบบีเพล
4. ศึกษาการใช้งานบีเพลเอนจิน
5. กำหนดคุณลักษณะและขอบเขตความสามารถของเครื่องมือ
6. ออกแบบโครงสร้างและหน้าจอของเครื่องมือทดสอบ
7. พัฒนาเครื่องมือทดสอบตามที่ได้ออกแบบไว้
8. ทดสอบเครื่องมือทดสอบตามขอบเขตที่ได้กำหนดไว้ ประเมินผลที่ได้จากการทดสอบ
9. สรุปผลการวิจัย และข้อเสนอแนะ
10. จัดทำเอกสารวิทยานิพนธ์

1.5. ประโยชน์ที่คาดว่าจะได้รับ

1. เพื่อช่วยให้ผู้ทดสอบสามารถตรวจสอบข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิสและข้อมูลส่งออกที่ได้รับจากเว็บเซอร์วิสได้
2. เพื่อช่วยให้ผู้ทดสอบทราบถึงผลการทดสอบว่ามีการทดสอบครอบคลุมทุกเว็บเซอร์วิสหรือไม่
3. เพื่อช่วยสร้างกรณีทดสอบเพิ่มเติมเบื้องต้น ให้กับผู้ทดสอบตามโหมดเงื่อนไขของเส้นทางการไหลของบีเพล

1.6. บทความวิชาการที่ได้รับการตีพิมพ์

งานวิจัยนี้ถูกนำไปตีพิมพ์และเผยแพร่ในหัวข้อ “An Approach for Monitoring Partner Link Invocations under WS- BPEL” ณ การประชุมวิชาการ “The International MultiConference of Engineers and Computer Scientists 2017 (IMECS 2017)” ที่ถูกจัดขึ้นในช่วงวันที่ 15 - 17 มีนาคม 2560 ณ โรงแรม เดอะ รอยัล การ์เด้น เมืองเกาลูน เขตบริหารพิเศษฮ่องกงแห่ง สาธารณรัฐประชาชนจีน

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1. ทฤษฎีที่เกี่ยวข้อง

วิทยานิพนธ์นี้ได้อ้างอิงถึงทฤษฎีที่เกี่ยวข้อง เพื่อนำมาใช้ประโยชน์ในขั้นตอนการวิจัย ซึ่งประกอบด้วยทฤษฎี ดังนี้

2.1.1 ดับเบิลยูเอส-บีเพล

ดับเบิลยูเอส-บีเพล เป็นภาษาสำหรับออกแบบกระบวนการทางธุรกิจ โดยมีโครงสร้างภาษาในรูปแบบของภาษาเอกซ์เอ็มแอล ถูกพัฒนาโดยองค์กรมาตรฐานกลางโอเอซิส (OASIS) [5] เพื่ออธิบายการทำงานร่วมกันระหว่างผู้ใช้งานระบบกับเว็บเซอร์วิสที่เกี่ยวข้อง ตามกระบวนการทางธุรกิจที่ถูกกำหนดไว้ ซึ่งในการออกแบบกระบวนการผู้ใช้งานสามารถจัดเรียงลำดับขั้นตอนการทำงานของเว็บเซอร์วิสและระบุเงื่อนไขของการทำงานที่ต้องการได้ [6]

ในการทำงานของดับเบิลยูเอส-บีเพลที่มีการทำงานร่วมกับเว็บเซอร์วิสจำเป็นต้องใช้พาร์ทเนอร์ลิงก์ สำหรับอ้างอิงถึงไฟล์ดับเบิลยูเอสดีแอล (WSDL - Web Services Description Language) เพื่อให้ทราบถึงเซอร์วิสที่สามารถเรียกใช้งานได้ และคลาสที่เกี่ยวข้องกับเว็บเซอร์วิส รวมถึงที่อยู่ของเว็บเซอร์วิส โดยไฟล์ดับเบิลยูเอส-บีเพล จะมีตัวอย่าง ดังรูปที่ 2-1

```
<invoke name="call_getVitalSigns" partnerLink="SOAPReference" portType="ns2:IService1"
        operation="getViatalSigns" inputVariable="v_vitalInput"
        outputVariable="v_vitalSignOutput" bpelx:invokeAsDetail="no"/>
<if name="If_sys_moreThan_140">
  <documentation>
    <![CDATA[ABNormal BP]]>
  </documentation>
  <condition>$v_vitalSignOutput.parameters/ns2:getViatalSignsResult/ns3:sys > 140
  <assign name="set_outputResult_ABnormal">
    <copy>
      <from>concat("HN :", $v_patientOutput.parameters/ns2:getPatientResult/ns3:f
      <to expressionLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang:xpath1.0">$o
    </copy>
  </assign>
<else>
  <documentation>
    <![CDATA[Normal BP]]>
  </documentation>
  <assign name="set_outputResult_Normal">
```

รูปที่ 2-1 ตัวอย่างไฟล์ดับเบิลยูเอส-บีเพล

โครงสร้างของดับเบิลยูเอส-บีเฟลประกอบด้วยโหนดต่างๆ ดังนี้

<receive>	<reply>	<invoke>	<assign>
<wait>	<empty>	<sequence>	<if>
<forEach>	<pick>	<flow>	<scope>
<rethrow>	<validate>	<extensionActivity>	<exit>
<compensateScope>	<throw>	<while>	<compensate>
<repeatUntil>			

นอกจากนี้ ดับเบิลยูเอส-บีเฟล ยังกำหนดวิธีการจัดการกับข้อผิดพลาดที่อาจเกิดขึ้นระหว่างประมวลผลได้ [7]

2.1.2 พาร์ทเนอร์ลิงก์

พาร์ทเนอร์ลิงก์ เป็นแท็กที่ใช้อ้างอิงจากระบวนการดับเบิลยูเอส-บีเฟล ไปยังเว็บเซอร์วิสที่ทำงานร่วมกัน โดยเมื่อกระบวนการดับเบิลยูเอส-บีเฟลเรียกใช้งานเว็บเซอร์วิส พาร์ทเนอร์ลิงก์จะทำหน้าที่อ้างอิงไปยังเว็บเซอร์วิสนั้นเพื่อเรียกใช้งาน [8] โครงสร้างของแท็กพาร์ทเนอร์ลิงก์มีตัวอย่างดังรูปที่ 2-2

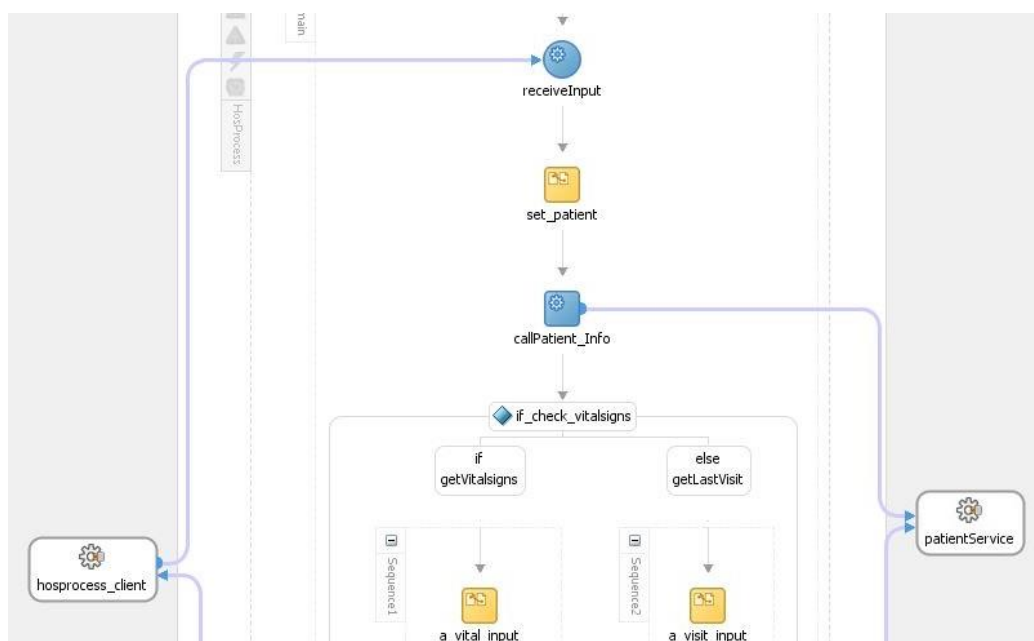
```
<partnerLinks>
  ...
  <partnerLink name="pharmacistServices"
    partnerLinkType="ns1:pharmacistServices"
    partnerRole="IPharmacistServices" />
</partnerLinks>
```

รูปที่ 2-2 ตัวอย่างแท็กพาร์ทเนอร์ลิงก์ [8]

2.1.3 บีเฟลเอนจิน

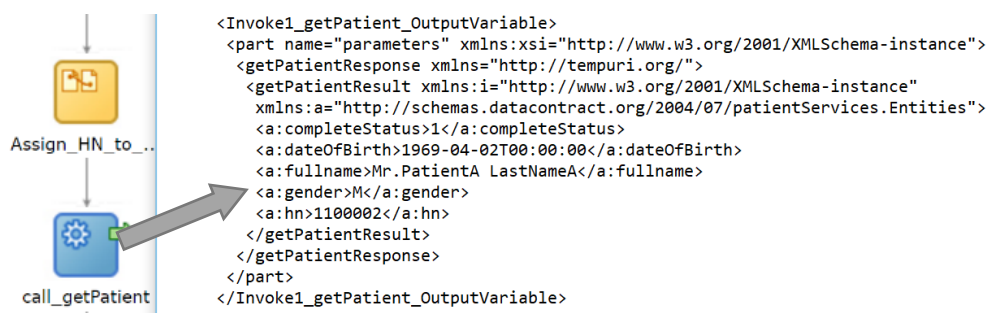
บีเฟลเอนจิน เป็นเครื่องมือสำหรับจัดการกระบวนการทางธุรกิจ โดยในปัจจุบันถูกสร้างขึ้นจากหลากหลายองค์กร โดยบีเฟลเอนจินของแต่ละองค์กรจะมีความสามารถและจุดเด่นที่แตกต่างกัน ตัวอย่างของบีเฟลเอนจิน เช่น ออราเคิล บีเฟล โพรเซส เมเนเจอร์ เป็นเครื่องมือที่ถูกพัฒนาโดยบริษัท ออราเคิล โดยเครื่องมือสามารถให้ผู้ใช้งานออกแบบกระบวนการทางธุรกิจ รวมไปถึงการนำไปติดตั้งและประมวลผลกระบวนการที่ออกแบบขึ้นได้ ซึ่งเครื่องมือรองรับการออกแบบ

การทำงานร่วมกัน ระหว่างระบบกับเว็บเซอร์วิซภายนอก และรองรับการทำงานกับระบบขององค์กรใหญ่ๆได้ [8] ในการใช้งานเครื่องมือ ผู้ใช้งานสามารถออกแบบกระบวนการได้สะดวก และสามารถนำมาออกแบบกับกระบวนการที่มีความซับซ้อนได้ [9] ในขั้นตอนการออกแบบกระบวนการทางธุรกิจ เครื่องมือจะแสดงผลดังรูปที่ 2-3



รูปที่ 2-3 ตัวอย่างหน้าจอการออกแบบกระบวนการ

นอกจากการสร้างกระบวนการทางธุรกิจแล้ว เครื่องมือยังสามารถติดตั้ง จัดการกระบวนการ และทดสอบการทำงานของกระบวนการได้ ซึ่งในการทดสอบการทำงานของกระบวนการทางธุรกิจที่สร้างขึ้น เครื่องมือสามารถแสดงผลการทดสอบได้ในลักษณะของเอกซ์เอ็มแอล ได้โดยผู้ทดสอบสามารถเลือกดูได้จากโหนดที่สนใจ ดังรูปที่ 2-4 นอกจากนี้เครื่องมือยังสามารถแสดงถึงเส้นทางการทดสอบที่ถูกทดสอบได้แต่เครื่องมือจะไม่ได้แสดงเว็บเซอร์วิซที่ไม่ถูกทดสอบดังรูปที่ 2-5



รูปที่ 2-4 การแสดงข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิซจากโหนด call_getPatient



รูปที่ 2-5 การแสดงเส้นทางการทดสอบของเครื่องมือ โดยจะแสดงเฉพาะโหนดที่ถูกทดสอบ

2.1.4 เอกซ์เอสดี (XSD - XML Schema Definition)

เอกซ์เอสดี เป็นเอกสารสำหรับอธิบายโครงสร้างของข้อมูลของเอกสารเอกซ์เอ็มแอล โดยไฟล์เอกซ์เอสดีจะอธิบายโครงสร้างของข้อมูลด้วยภาษาเอกซ์เอ็มแอล โดยจะแบ่งเป็นแท็กต่างๆ ดังนี้ [10]

1. <schema> เป็นแท็กที่อธิบายถึงจุดเริ่มต้นซึ่งภายในแท็กจะประกอบด้วย แท็กย่อยต่างๆ ต่อไป
2. <element> เป็นแท็กที่อธิบายโครงสร้างข้อมูลในเอกซ์เอ็มแอล โดยภายในแท็กจะอ้างถึงชื่อข้อมูล และ ประเภทของข้อมูลในเอกซ์เอ็มแอล โดยมีตัวอย่างดังรูปที่ 2-6

```
<xs:element name="lastname" type="xs:string"/>
<xs:element name="age" type="xs:integer"/>
<xs:element name="lastname" type="xs:date"/>
```

รูปที่ 2-6 ตัวอย่างแท็ก <element> สำหรับอธิบายโครงสร้างข้อมูล

3. <attribute> เป็นแท็กที่ใช้อธิบายคุณลักษณะของข้อมูลเพิ่มเติมในเอกซ์เอ็มแอล โดยภายในแท็กจะอ้างถึงชื่อข้อมูล และ ประเภทของข้อมูลในเอกซ์เอ็มแอล โดยมีตัวอย่างดังรูปที่ 2-7

```
<xs:attribute name="lang" type="xs:string"/>
```

รูปที่ 2-7 ตัวอย่างแท็ก <attribute> สำหรับอธิบายโครงสร้างข้อมูล

4. <restriction> เป็นแท็กที่ใช้อธิบายข้อจำกัดของข้อมูลในเอกซ์เอ็มแอล โดยจะอธิบายค่าที่ยอมรับได้ในแท็ก < element > และ < attribute > โดยมีตัวอย่างดังรูปที่ 2-8

```
<xs:element name="car" type="carType"/>

<xs:simpleType name="carType">
  <xs:restriction base="xs:string"/>
  <xs:enumeration value="Audi"/>
  <xs:enumeration value="Golf"/>
  <xs:enumeration value="BMW"/>
</xs:restriction>
</xs:simpleType>
```

รูปที่ 2-8 ตัวอย่างแท็ก <restriction>

2.2. งานวิจัยที่เกี่ยวข้อง

วิทยานิพนธ์นี้ได้อ้างอิงถึงงานวิจัยอื่นที่เกี่ยวข้อง เพื่อนำมาใช้ประโยชน์ในขั้นตอนการวิจัย ซึ่งประกอบด้วยงานวิจัย ดังนี้

2.2.1 งานวิจัย “Calculating BPEL Test Coverage through Instrumentation” [1]

งานวิจัยนี้ได้นำเสนอวิธีการคำนวณเพื่อหาการทดสอบที่ครอบคลุมของไฟล์บีเพล โดยใช้วิธีการแทรกรหัสต้นทางเพื่อจัดเก็บข้อมูลการทดสอบของโหนดต่างๆ ในการแทรกรหัสต้นทางมี 3 ขั้นตอนดังนี้

1. หาโหนดที่เกี่ยวข้องกับการวัดความครอบคลุมของการทดสอบ
2. วิเคราะห์โหนดและแทรกรหัสต้นทางของโหนดนั้น
3. ทำการแทนที่รหัสต้นทางที่มีการเรียกใช้เว็บเซอร์วิสเพื่อจัดเก็บความครอบคลุมของการทดสอบเว็บเซอร์วิส

จากงานวิจัยดังกล่าวสามารถนำหลักการแทรกรหัสต้นทางมาปรับใช้สำหรับจัดเก็บข้อมูลเพื่อตรวจจับการเรียกใช้เว็บเซอร์วิส

2.2.2 งานวิจัย “Automatic Test Case Selection and Generation for Regression Testing of Composite Service Based on Extensible BPEL Flow Graph” [2]

งานวิจัยนี้นำเสนอการเลือกและสร้างกรณีทดสอบสำหรับการทดสอบซ้ำโดยใช้กราฟการไหลของบีเพลที่ถูกขยาย โดยแบ่งโหนดบีเพล ออกเป็นกลุ่มต่างๆ เพื่อใช้ในการจำแนกโหนดที่จะต้องแทรกรหัสต้นทางเพื่อสร้างกรณีทดสอบ โดยมีการจำแนกโหนดดังตารางที่ 2-1

ตารางที่ 2-1 การจำแนกโหนดของกราฟการไหลของบีเพลที่ถูกขยาย

Node Type	BPEL node
Interaction Node(IN)	<invoke>, <receive>, <reply>
Normal Node(NN)	<assign>, <wait>
Exclusive Node (EN)	<if>, <pick>, <while>, <repeatUntil>

จากงานวิจัยดังกล่าวสามารถนำหลักการแบ่งประเภทกลุ่มโหนดต่างๆ มาปรับใช้ในการแทรกรหัสต้นทางได้

2.2.3 งานวิจัย “Automated Testing of WS-BPEL Service Compositions: A Scenario-Oriented Approach” [3]

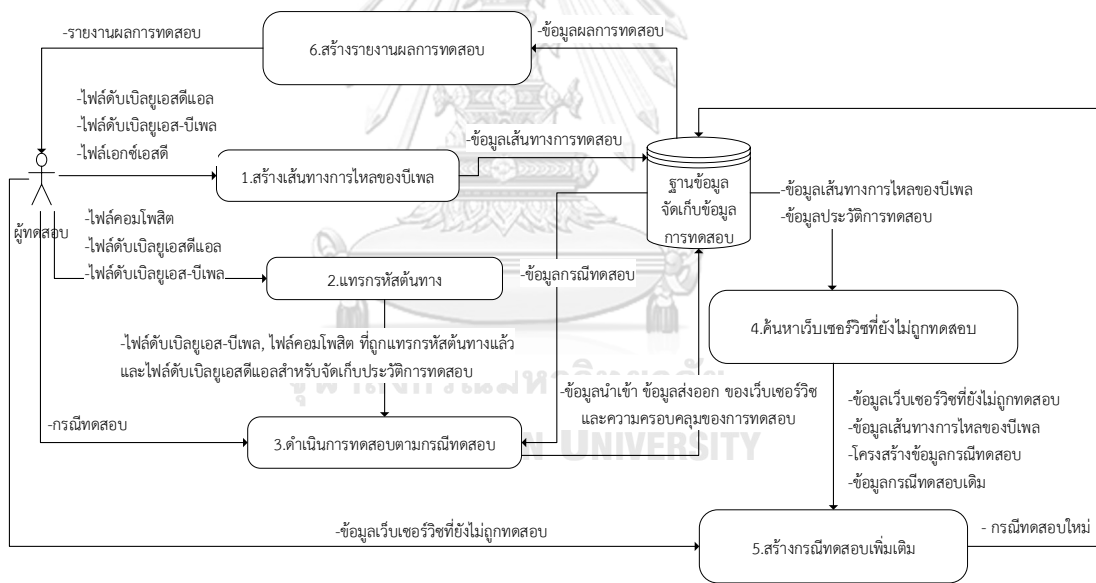
งานวิจัยนี้ได้นำเสนอวิธีการสร้างกรณีทดสอบอย่างอัตโนมัติด้วยดับเบิลยูเอส-บีเพล โดยใช้หลักการหาเงื่อนไขของเส้นทางการไหลที่ถูกระบุไว้ในเหตุการณ์ที่จะทดสอบ จากนั้นจะทำการสร้างกรณีทดสอบจากไฟล์ ดับเบิลยูเอสดีแอล ที่เกี่ยวข้องกับไฟล์ดับเบิลยูเอส-บีเพล เพื่อหาช่วงของข้อมูลนำเข้าแต่ละตัว

จากงานวิจัยดังกล่าวสามารถนำวิธีการสร้างกรณีทดสอบมาปรับใช้เพื่อสร้างกรณีทดสอบเพิ่มเติมให้ครอบคลุมการทำงานของเว็บเซอร์วิสที่เกี่ยวข้องกับไฟล์ดับเบิลยูเอส-บีเพลได้

บทที่ 3

วิธีการตรวจจัดการเรียกใช้เว็บเซอร์วิชสำหรับทดสอบดับเบิ้ลยูเอส-บีเฟล

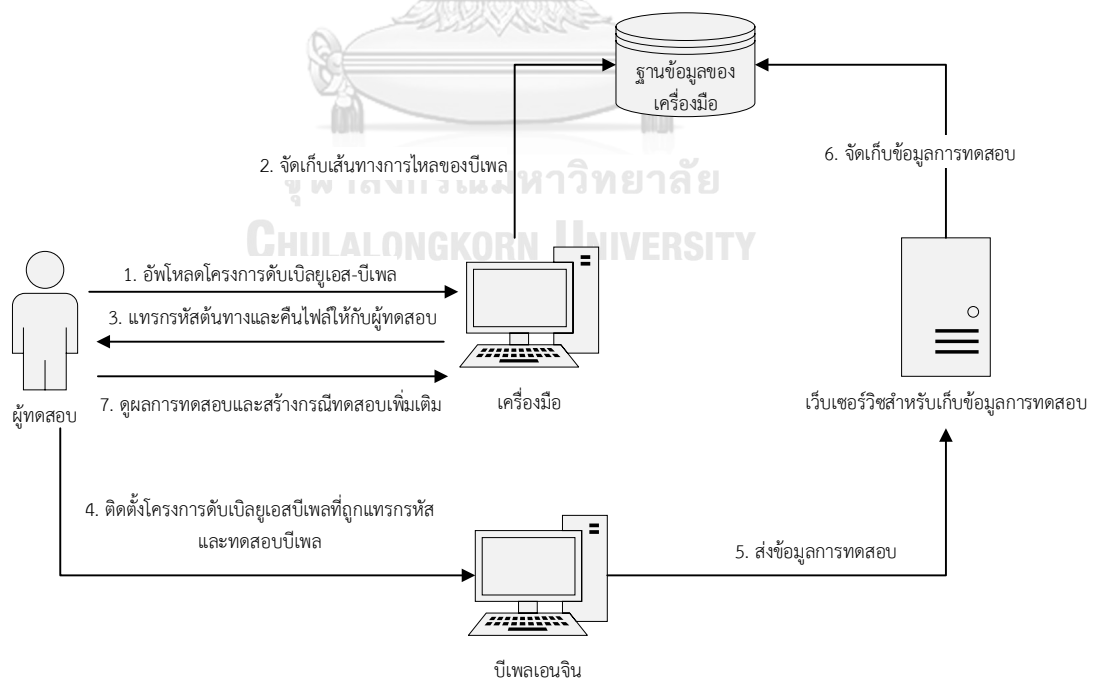
วิทยานิพนธ์นี้มีจุดประสงค์เพื่อนำเสนอวิธีการและพัฒนาเครื่องมือตรวจจัดการเรียกใช้เว็บเซอร์วิชสำหรับทดสอบดับเบิ้ลยูเอส-บีเฟล เพื่อช่วยให้ผู้ทดสอบสามารถตรวจสอบข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิช และข้อมูลส่งออกที่ได้รับจากเว็บเซอร์วิชได้ และช่วยตรวจสอบความครอบคลุมของการทดสอบเว็บเซอร์วิช เพื่อให้ผู้ทดสอบสามารถทราบได้ว่ายังมีเว็บเซอร์วิชใดที่ยังไม่ถูกทดสอบบ้าง นอกจากนี้ยังสามารถสร้างกรณีทดสอบเพิ่มเติมเพื่อครอบคลุมเว็บเซอร์วิชที่ยังไม่ถูกทดสอบซึ่งจะวิเคราะห์จากโหนดเงื่อนไขที่ไม่ถูกทดสอบและตัวแปรที่เกี่ยวข้องกับเงื่อนไขนั้น พร้อมทั้งจัดเก็บประวัติการสร้างกรณีทดสอบในฐานข้อมูล โดยวิธีการตรวจจัดการเรียกใช้เว็บเซอร์วิชสำหรับทดสอบดับเบิ้ลยูเอส-บีเฟล มีภาพรวมการทำงานดังรูปที่ 3-1



รูปที่ 3-1 ภาพรวมการทำงาน

จากรูปที่ 3-1 แสดงให้เห็นถึงภาพรวมการทำงานของวิธีการตรวจจัดการเรียกใช้เว็บเซอร์วิชสำหรับทดสอบดับเบิ้ลยูเอส-บีเฟล โดยวิธีการทำงานจะเริ่มจากการนำไฟล์ดับเบิ้ลยูเอส-บีเฟล มาสร้างเส้นทางการไหลของบีเฟลและจัดเก็บลงฐานข้อมูล จากนั้นจะแทรกรหัสต้นทางในไฟล์ดับเบิ้ลยูเอส-บีเฟล เพื่อจัดเก็บข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิชและข้อมูลส่งออกที่ได้รับจากเว็บเซอร์วิช นอกจากนี้ยังแทรกรหัสต้นทางเพื่อติดตามความครอบคลุมของการทดสอบเว็บเซอร์วิช

หลังจากแทรกรหัสต้นทางในไฟล์ดับเบิลยูเอส-บีเพลแล้ว ผู้ทดสอบจะนำไฟล์ที่ได้พร้อมทั้งไฟล์ดับเบิลยูเอสดีแอลที่เกี่ยวข้องไปทดสอบด้วยบีเพลเอนจิน โดยจะใช้กรณีทดสอบที่ผู้ทดสอบมีหรือสามารถนำกรณีทดสอบที่ถูกจัดเก็บในฐานข้อมูลมาใช้ทดสอบได้ ในขณะที่ทดสอบเมื่อมีการประมวลผลผ่านบรรทัดที่ถูกแทรกรหัสต้นทาง บีเพลเอนจิน จะส่งข้อมูลการทดสอบไปจัดเก็บในฐานข้อมูล ซึ่งข้อมูลที่ส่งไปจะประกอบด้วย ข้อมูลนำเข้าที่จะส่งไปยังเว็บเซอร์วิส ข้อมูลส่งออกที่ได้จากเว็บเซอร์วิส และข้อมูลการถูกเรียกใช้งานของโหนดนั้นๆ เพื่อวิเคราะห์ความครอบคลุมของการทดสอบ ในการเรียกดูผลการทดสอบสามารถทำได้ทั้งในขณะที่ทดสอบหรือเมื่อทดสอบเสร็จแล้ว โดยผลการทดสอบที่เรียกดูสามารถแสดง ข้อมูลนำเข้า ข้อมูลส่งออก และข้อมูลความครอบคลุมของการทดสอบเว็บเซอร์วิสได้ ในกรณีที่ตรวจพบเว็บเซอร์วิสที่ยังไม่ถูกทดสอบ วิธีการนี้ยังตรวจหาเส้นทางการไหลของบีเพลเพื่อสร้างกรณีทดสอบเพิ่มเติมสำหรับเว็บเซอร์วิสที่ไม่ถูกทดสอบได้ โดยนำเว็บเซอร์วิสที่ไม่ถูกทดสอบและโหนดเงื่อนไขที่อยู่ในเส้นทางการไหลของโหนดเว็บเซอร์วิสนั้น มาวิเคราะห์ข้อมูลในเงื่อนไขว่าตัวแปรที่ใช้เป็นข้อมูลนำเข้าที่อยู่ในกรณีทดสอบหรือไม่ ในกรณีที่ในเงื่อนไขใช้ตัวแปรที่เป็นข้อมูลนำเข้าที่อยู่ในกรณีทดสอบวิธีการนี้สามารถสร้างกรณีทดสอบเพิ่มเติมได้ โดยจะต้องใช้ไฟล์เอกซ์เอสดี และไฟล์ดับเบิลยูเอสดีแอล ที่เกี่ยวข้องกับไฟล์ดับเบิลยูเอส-บีเพลเพิ่มเติม เพื่อนำมาสร้างกรณีทดสอบชุดใหม่ที่สามารถเรียกใช้งานเว็บเซอร์วิสที่ยังไม่ถูกทดสอบพร้อมทั้งจัดเก็บประวัติการสร้างกรณีทดสอบเข้าสู่ฐานข้อมูลได้ โดยมีภาพรวมเชิงองค์ประกอบของวิธีการ ดังรูปที่ 3-2

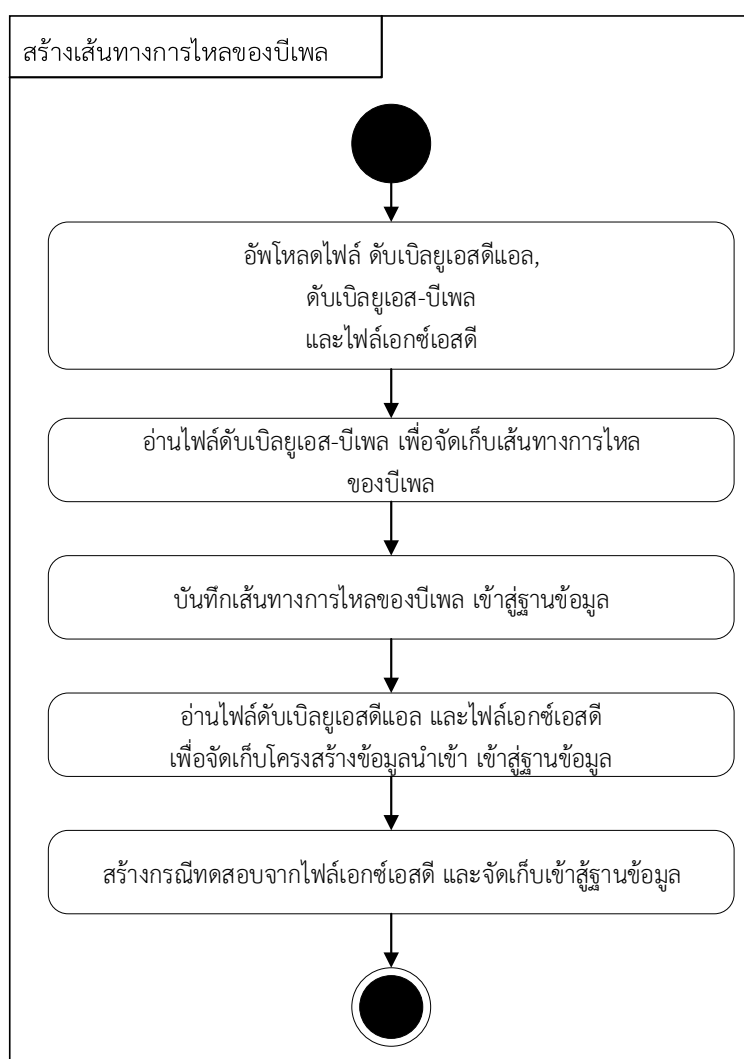


รูปที่ 3-2 ภาพรวมเชิงองค์ประกอบของวิธีการ

และมีรายละเอียดการทำงานในแต่ละขั้นตอนดังนี้

3.1 การสร้างเส้นทางการไหลของบีเพล

ในขั้นตอนการสร้างเส้นทางการไหลของบีเพล จะนำเข้าไฟล์ดับเบิลยูเอส-บีเพลจากผู้ทดสอบเพื่อหาเส้นทางการไหล โดยอ่านไฟล์ดับเบิลยูเอส-บีเพล จากนั้นจะตรวจหาโหนดเริ่มต้น และอ่านโหนดต่างๆ ที่อยู่ในไฟล์ดับเบิลยูเอส-บีเพล เพื่อสร้างความสัมพันธ์ระหว่างโหนด พร้อมทั้งบันทึกเส้นทางการไหลของดับเบิลยูเอส-บีเพลลงสู่ฐานข้อมูล นอกจากการบันทึกเส้นทางการไหลของบีเพลแล้วในขั้นตอนนี้ยังนำเข้าไฟล์ดับเบิลยูเอสดีแอล และไฟล์เอกซ์เอสดีจากผู้ทดสอบ เพื่อสร้างและจัดเก็บกรณีทดสอบของข้อมูลนำเข้าและบันทึกเข้าสู่ฐานข้อมูล โดยมีขั้นตอนการทำงานดังรูปที่ 3-3



รูปที่ 3-3 การสร้างเส้นทางการไหลของบีเพล

การทำงานในขั้นตอนสร้างเส้นทางการไหลของดับเบิลยูเอส-บีเพล เริ่มโดยผู้ทดสอบอัปโหลดไฟล์ดับเบิลยูเอสดีแอล ดับเบิลยูเอส-บีเพลและไฟล์เอกซ์เอสดี โดยมีตัวอย่างไฟล์ดับเบิลยูเอส-บีเพลดังรูปที่ 3-4

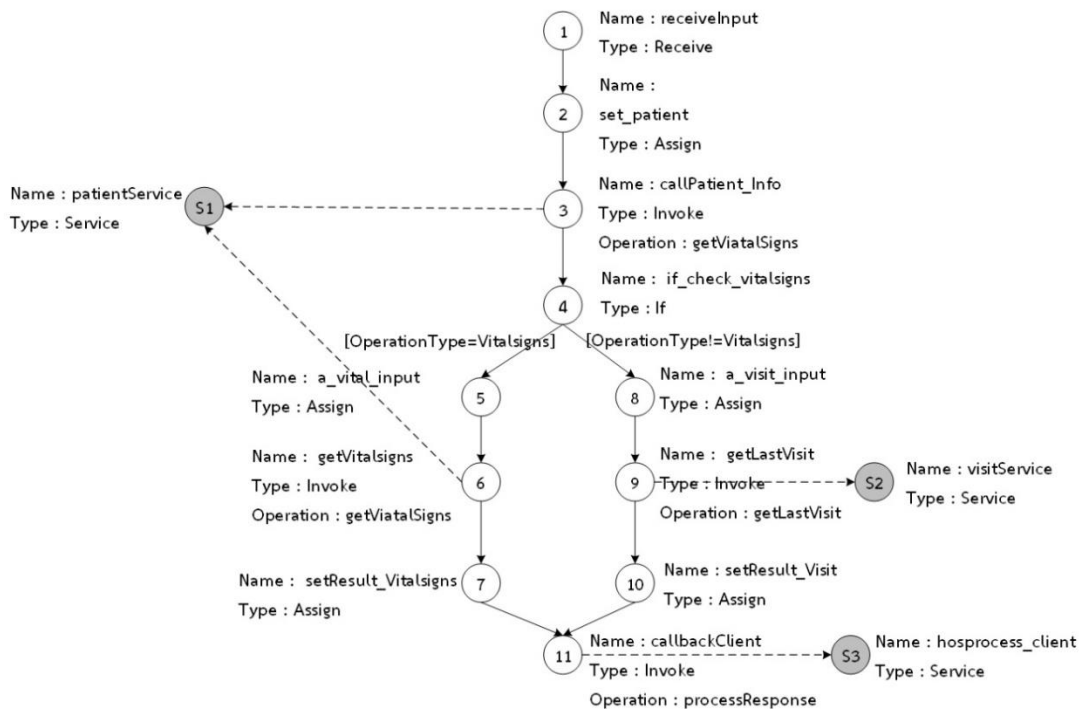
```

<sequence name="main">
  <receive name="receiveInput" partnerLink="hosprocess_client"
  portType="client:HosProcess" operation="process"
  variable="inputVariable" createInstance="yes"/>
  <assign name="set_patient">
    <copy>
      <from>$inputVariable.payload/client:hn</from>
      <to expressionLanguage="urn:oasis:names:tc:wsbpel:2.0:sublang:xpath1.0">
        $Invoke1_getPatient_InputVariable.parameters/ns1:hn</to>
    </copy>
  </assign>
  <invoke name="callPatient_Info" partnerLink="patientService" portType="ns1:IService1"
  operation="getPatient" inputVariable="Invoke1_getPatient_InputVariable"
  outputVariable="Invoke1_getPatient_OutputVariable" bpelx:invokeAsDetail="no"/>

```

รูปที่ 3-4 ตัวอย่างไฟล์ดับเบิลยูเอส-บีเพล

จากนั้นวิธีการนี้จะสร้างเส้นทางการไหลของบีเพล โดยอ่านแท็กเอกซ์เอ็มแอล เพื่อสร้างโหนดและความสัมพันธ์ระหว่างโหนด และจัดเก็บเส้นทางการไหลของดับเบิลยูเอส-บีเพลลงสู่ฐานข้อมูล โดยมีตัวอย่างเส้นทางการไหลของดับเบิลยูเอส-บีเพล ดังรูปที่ 3-5



รูปที่ 3-5 ตัวอย่างเส้นทางการไหลของดับเบิลยูเอส-บีเพล

หลังจากจัดเก็บเส้นทางการไหลของดับเบิลยูเอส-บีเพลเข้าสู่ฐานข้อมูลแล้ว จะนำไฟล์ดับเบิลยูเอสดีแอล และไฟล์เอกซ์เอสดีจากผู้ทดสอบมาสร้างและจัดเก็บข้อมูลกรณีทดสอบ โดยมีตัวอย่างไฟล์ดับเบิลยูเอสดีแอล และไฟล์เอกซ์เอสดี ดังรูปที่ 3-6 และ 3-7 ตามลำดับ


```

<wsdl:message name="HosProcessRequestMessage">
  <wsdl:part name="payload" element="client:process"/>
</wsdl:message>

<wsdl:portType name="HosProcess">
  <wsdl:operation name="process">
    <wsdl:input message="client:HosProcessRequestMessage"/>
  </wsdl:operation>
</wsdl:portType>

```

รูปที่ 3-6 ตัวอย่างไฟล์ดับเบิลยูเอสดีแอล

```

<schema attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://xmlns.oracle.com/sample_HOS/sampleHos/HosProcess"
  xmlns="http://www.w3.org/2001/XMLSchema">
  <element name="process">
    <complexType>
      <sequence>
        <element name="hn" type="string"/>
        <element name="operationType" type="string"/>
      </sequence>
    </complexType>
  </element>

```

รูปที่ 3-7 ตัวอย่างไฟล์เอกซ์เอสดี

3.2 การแทรกรหัสต้นทาง

ในขั้นตอนการแทรกรหัสต้นทาง จะนำไฟล์คอมโพสิตของโครงการดับเบิลยูเอส-บีเพล ที่ถูกอัปโหลดจากผู้ทดสอบมาแทรกรหัสต้นทาง เพื่อให้โครงการดับเบิลยูเอส-บีเพลสามารถเข้าถึงเว็บเซอร์วิสสำหรับจัดเก็บข้อมูลการทดสอบได้ โดยมีตัวอย่างไฟล์คอมโพสิตที่ถูกแทรกรหัสต้นทาง ดังรูปที่ 3-8

```

<composite name="Project_HISPayment" revision="1.0" label="2017-11-26_13-05-13_827" mode="active" state="on" xmlns="http://xmlns:
<import namespace="http://xmlns.oracle.com/Application_HISPayment/Project_HISPayment/BPELHISPayment" location="WSDLs/BPEL
<import namespace="http://tempuri.org/" location="WSDLs/LogServicesWrapper.wsdl" importType="wsdl" />
<import namespace="http://tempuri.org/" location="WSDLs/LogServices.wsdl" importType="wsdl" />

```

รูปที่ 3-8 ตัวอย่างไฟล์คอมโพสิตที่ถูกแทรกรหัสต้นทาง

หลังจากที่แทรกรหัสต้นทางที่ไฟล์คอมโพสิตแล้ววิธีการนี้จะคัดลอกไฟล์ดับเบิลยูเอสดีแอลของเว็บเซอร์วิสที่ทำหน้าที่เก็บข้อมูลการทดสอบมายังแฟ้มดับเบิลยูเอสดีแอลของโครงการเพื่อให้โครงการดับเบิลยูเอส-บีเพล สามารถอ้างอิงถึงเว็บเซอร์วิสที่เก็บข้อมูลได้ โดยมีตัวอย่างไฟล์ดับเบิลยูเอสดีแอลของเว็บเซอร์วิสที่เก็บข้อมูล ดังรูปที่ 3-9 จากนั้นจะนำไฟล์ดับเบิลยูเอส-บีเพลจากผู้ทดสอบมาแทรกรหัสต้นทางเพื่อจัดเก็บข้อมูลของการทดสอบ

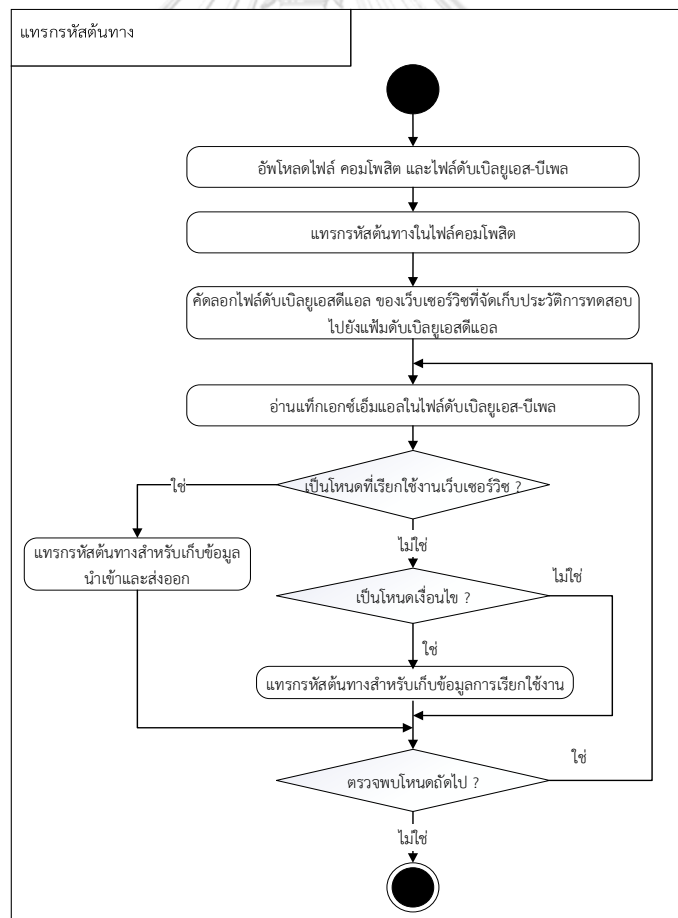
```

--
<wsdl:message name="ILogServices_setLog_InputMessage">
  <wsdl:part name="parameters" element="tns:setLog"/>
</wsdl:message>
<wsdl:message name="ILogServices_setLog_OutputMessage">
  <wsdl:part name="parameters" element="tns:setLogResponse"/>
</wsdl:message>
<wsdl:message name="ILogServices_getJobID_InputMessage">
  <wsdl:part name="parameters" element="tns:getJobID"/>
</wsdl:message>
<wsdl:message name="ILogServices_getJobID_OutputMessage">
  <wsdl:part name="parameters" element="tns:getJobIDResponse"/>
</wsdl:message>
<wsdl:portType name="ILogServices">
  <wsdl:operation name="setLog">
    <wsdl:input message="tns:ILogServices_setLog_InputMessage" wsaw:Action="http://ter
    <wsdl:output message="tns:ILogServices_setLog_OutputMessage" wsaw:Action="http://t
  </wsdl:operation>
  <wsdl:operation name="getJobID">
    <wsdl:input message="tns:ILogServices_getJobID_InputMessage" wsaw:Action="http://t
    <wsdl:output message="tns:ILogServices_getJobID_OutputMessage" wsaw:Action="http://t
  </wsdl:operation>
</wsdl:portType>

```

รูปที่ 3-9 ตัวอย่างไฟล์ดับเบิ้ลยูเอสดีแอลของเว็บเซอร์วิชที่จัดเก็บข้อมูล

เมื่อแทรกรหัสต้นทางที่ไฟล์ดับเบิ้ลยูเอส-บีเพลสำเร็จแล้ว จะคืนไฟล์ที่ถูกแทรกหัสต้นทางให้กับผู้ทดสอบเพื่อนำไปใช้งานในขั้นตอนถัดไป ซึ่งขั้นตอนนี้จะมีการทำงานดังรูปที่ 3-10



รูปที่ 3-10 การแทรกรหัสต้นทาง

ในขั้นตอนการแทรกรหัสต้นทางที่ไฟล์ดับเบิลยูเอส-บีเพล จะแทรกรหัสต้นทางโดยแทรกโหนดตั้ง
ค่า (Assign) เพื่อนำค่ารหัสของโครงการดับเบิลยูเอส-บีเพล ที่ถูกสร้างในขั้นตอนที่ 3.1 มาระบุให้กับ
โครงการดับเบิลยูเอส-บีเพล จากนั้นจะแทรกรหัสต้นทางโดยแทรกโหนดเรียกใช้ (Invoke) เพื่อร้องขอ
รหัสของการทดสอบดับเบิลยูเอส-บีเพล จากเว็บเซอร์วิสที่ทำหน้าที่เก็บผลการทดสอบ และจะแทรก
รหัสต้นทางที่โหนดอื่นๆ โดยแบ่งประเภทของโหนดออกเป็นกลุ่มต่างๆ ดังนี้

1. โหนดที่ติดต่อกับพาร์ทเนอร์ลิงก์

การแทรกรหัสต้นทางโหนดที่ติดต่อกับพาร์ทเนอร์ลิงก์จะแทรกรหัสเพื่อจัดเก็บข้อมูลนำเข้า
และข้อมูลส่งออก โดยโหนดที่อยู่ในกลุ่มติดต่อกับพาร์ทเนอร์ลิงก์ผู้วิจัยได้จำแนกโหนดและวิธีการ
แทรกรหัสต้นทางในแต่ละโหนด ดังนี้

1.1 โหนดรับค่า (Receive)

โหนดรับค่า เป็นโหนดที่รับค่าข้อมูลนำเข้าจากผู้ทดสอบ โดยการแทรกรหัสต้นทางของโหนด
รับค่า จะแทรกรหัสต้นทางด้วยโหนดตั้งค่า เพื่อจัดเก็บค่าของข้อมูลนำเข้า หลังจากเรียกใช้งานโหนด
รับค่าแล้ว จะแทรกรหัสต้นทางเพื่อนำค่าที่จัดเก็บส่งไปยังเว็บเซอร์วิส โดยแทรกรหัสด้วยโหนด
เรียกใช้

1.2 โหนดเรียกใช้

โหนดเรียกใช้ เป็นโหนดที่เรียกใช้งานเว็บเซอร์วิสภายนอก โดยก่อนเรียกใช้งานโหนดเรียกใช้
จะแทรกรหัสต้นทางด้วยโหนดตั้งค่า เพื่อจัดเก็บค่าของข้อมูลนำเข้า และแทรกรหัสต้นทางด้วยโหนด
เรียกใช้เพื่อนำค่าที่จัดเก็บส่งไปยังเว็บเซอร์วิส และหลังจากเรียกใช้งานโหนดเรียกใช้ จะแทรกรหัสต้น
ทางด้วยโหนดตั้งค่า เพื่อจัดเก็บค่าของข้อมูลส่งออก และแทรกรหัสต้นทางด้วยโหนดเรียกใช้เพื่อนำ
ค่าที่จัดเก็บส่งไปยังเว็บเซอร์วิส

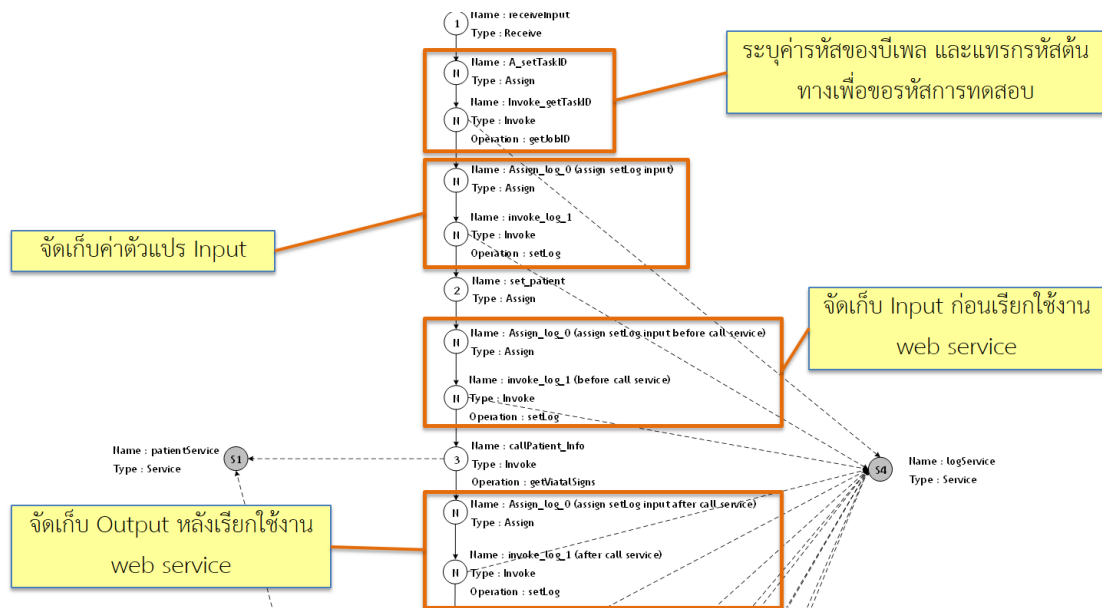
1.3 โหนดตอบกลับ (Response)

โหนดตอบกลับ เป็นโหนดที่ตอบกลับผลลัพธ์ให้กับผู้ทดสอบเมื่อจบการทำงานของ
ดับเบิลยูเอส-บีเพล โดยก่อนเรียกใช้งานโหนดตอบกลับ จะแทรกรหัสต้นทางด้วยโหนดตั้งค่า เพื่อ
จัดเก็บค่าของข้อมูลส่งออก และแทรกรหัสต้นทางด้วยโหนดเรียกใช้เพื่อนำค่าที่จัดเก็บส่งไปยังเว็บ
เซอร์วิส

2. โหนดเงื่อนไข

โหนดเงื่อนไขที่จะแทรกรหัสต้นทางโหนดต่างๆ ซึ่งประกอบด้วยโหนด “if”, “elseif”, “else”
และ “while” โดยก่อนเรียกใช้งานโหนดเงื่อนไข จะแทรกรหัสต้นทางด้วยโหนดตั้งค่า เพื่อตั้งค่า

สถานะการทดสอบที่กำลังจะเข้าทดสอบในโหมดเงื่อนไขนั้น และแทรกรหัสต้นทางด้วยโหนดเรียกใช้ เพื่อนำค่าที่จัดเก็บส่งไปยังเว็บเซอร์วิส และหลังจากเรียกใช้งานโหนดเงื่อนไข จะแทรกรหัสต้นทางด้วยโหนดตั้งค่า เพื่อตั้งค่าสถานะว่าเส้นทางของโหนดเงื่อนไขนี้ผ่านการทดสอบ จากนั้นจะแทรกรหัสต้นทางด้วยโหนดเรียกใช้เพื่อนำค่าที่จัดเก็บส่งไปยังเว็บเซอร์วิส โดยมีตัวอย่างเส้นทางการทำงานของบีเพลที่ถูกแทรกรหัสต้นทางแล้วดังรูปที่ 3-11

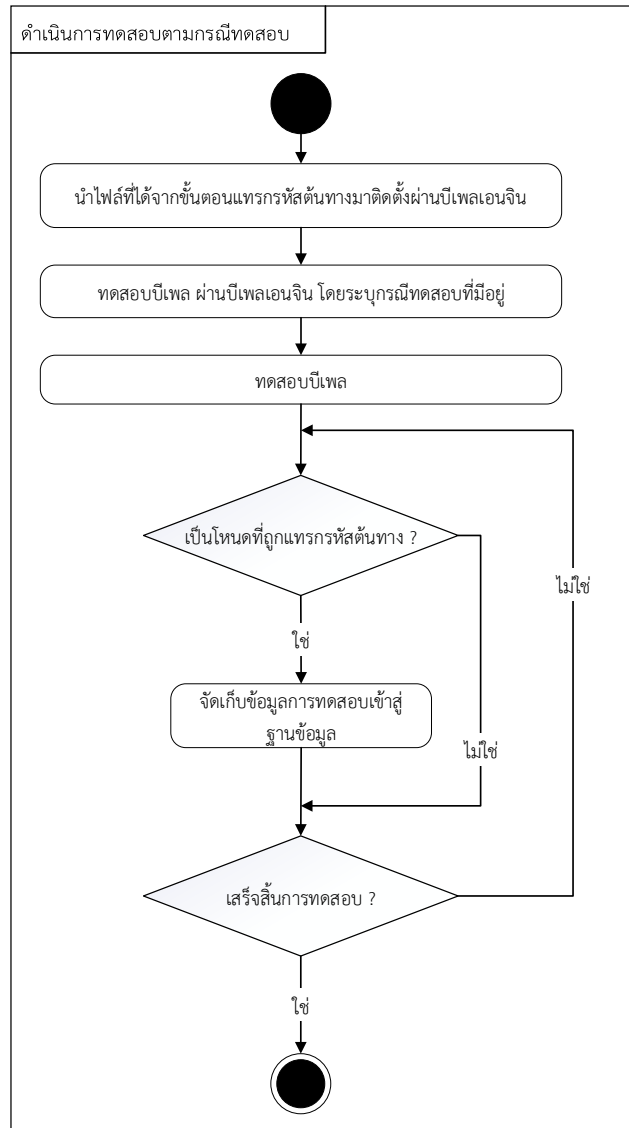


รูปที่ 3-11 ตัวอย่างเส้นทางการทำงานของบีเพลที่ถูกแทรกรหัสต้นทางแล้ว

3.3 การดำเนินการทดสอบตามกรณีทดสอบ

ในขั้นตอนการทดสอบ ผู้ทดสอบจะนำโครงการดับเบิ้ลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางแล้ว มาติดตั้งและทดสอบผ่านบีเพลเอนจิน ในการทดสอบผู้ทดสอบจะต้องระบุกรณีทดสอบที่มีอยู่ ผ่านบีเพลเอนจิน หรือผู้ทดสอบสามารถนำไฟล์กรณีทดสอบที่มีอยู่มาใช้ทดสอบได้ โดยขณะทดสอบเมื่อบีเพลเอนจินทดสอบผ่านโหนดเรียกใช้ ที่ถูกแทรกรหัสต้นทางไว้ บีเพลเอนจินจะเรียกใช้งานเว็บเซอร์วิสที่ทำหน้าที่เก็บข้อมูลการทดสอบเพื่อบันทึกข้อมูลการทดสอบเข้าสู่ฐานข้อมูล เช่น ในกรณีที่โหนดเรียกใช้นั้น ถูกแทรกรหัสเพื่อตรวจจับข้อมูลที่รับส่งระหว่างเว็บเซอร์วิส บีเพลเอนจินจะส่งข้อมูลนำเข้าและข้อมูลส่งออกของเว็บเซอร์วิส ผ่านไปยังเว็บเซอร์วิสที่เก็บข้อมูลเพื่อจัดเก็บข้อมูลนั้นลงสู่ฐานข้อมูล ซึ่งมีขั้นตอนการทำงานดังรูปที่ 3-12 เมื่อนำกรณีทดสอบดังรูปที่ 3-13 มาใช้ทดสอบผ่านบีเพลเอนจินแล้ว จะพบว่าโหนดที่เรียกใช้งานเว็บเซอร์วิสหมายเลข 3, 6 และ 11 จะถูกทดสอบดังรูปที่ 3-14 และหลังจากที่เว็บเซอร์วิสจัดเก็บข้อมูลการทดสอบแล้ว ผู้ทดสอบสามารถเรียกดูข้อมูลผลการทดสอบที่ถูกจัดเก็บได้ ซึ่งข้อมูลที่สามารถดูได้ เช่น ข้อมูลกรณีทดสอบที่นำมาทดสอบ ข้อมูล

นำเข้าที่ส่งให้กับเว็บเซอร์วิซ ข้อมูลส่งออกที่ได้รับจากเว็บเซอร์วิซ ข้อมูลผลลัพธ์ในเส้นทางการไหลของดับเบิลยูเอส-บีเฟล เป็นต้น

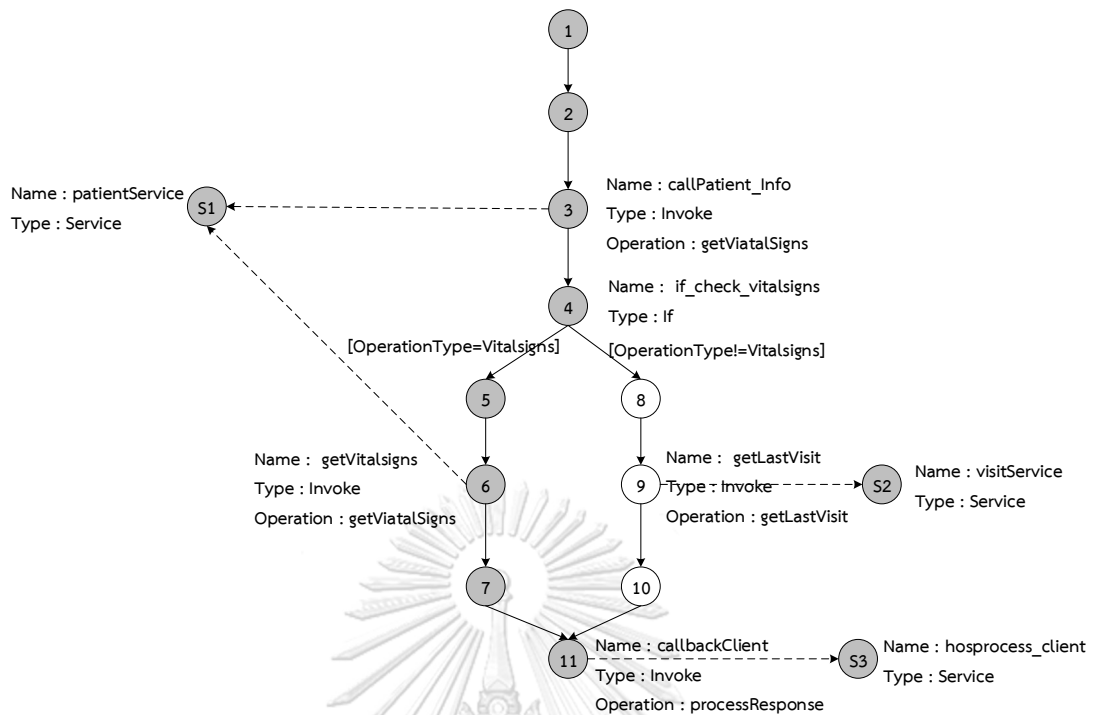


รูปที่ 3-12 การดำเนินการทดสอบตามกรณีทดสอบ

```

<inputVariable>
  <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    name="payload">
    <process
      xmlns="http://xmlns.oracle.com/sample_HOS/sampleHos/HosProcess"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <hn>5900001</hn>
      <operationType>Vitalsigns</operationType>
    </process>
  </part>
</inputVariable>
  
```

รูปที่ 3-13 ตัวอย่างกรณีทดสอบ

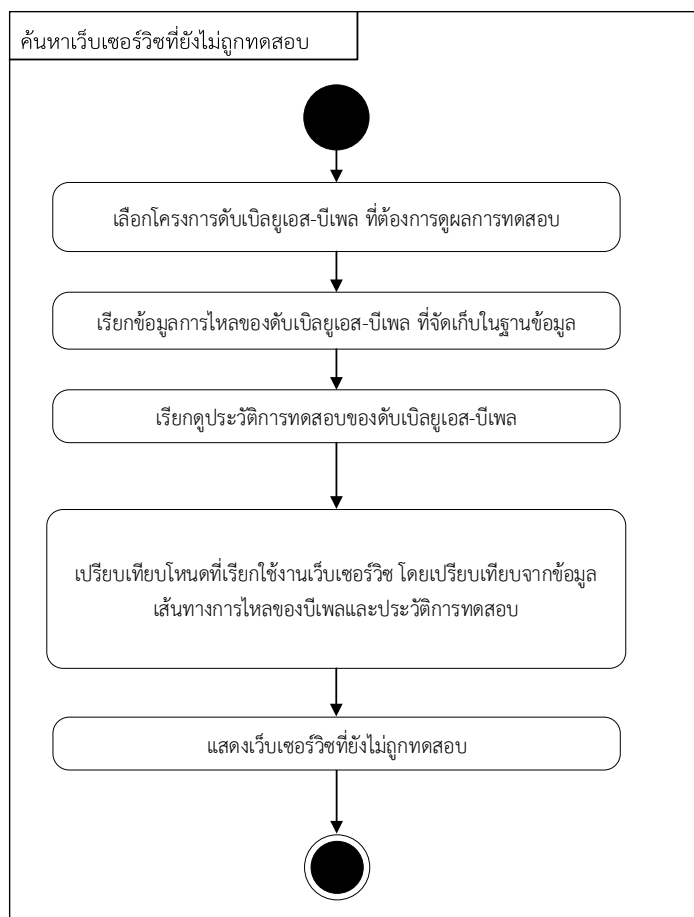


รูปที่ 3-14 โหนดที่ถูกทดสอบ

3.4 การค้นหาเว็บเซอร์วิสที่ยังไม่ถูกทดสอบ

หลังจากที่ผู้ทดสอบทำการทดสอบโครงการดับเบิลยูเอส-บีเพลผ่านบีเพลเอนจินในขั้นตอนที่ 3.3 แล้ว ในขั้นตอนนี้ผู้ทดสอบสามารถดูรายการเว็บเซอร์วิสที่ยังไม่ถูกทดสอบได้ โดยวิธีการค้นหาเว็บเซอร์วิสที่ยังไม่ถูกทดสอบจะใช้การเปรียบเทียบรายการเว็บเซอร์วิสที่อยู่ในเส้นทางการไหลของบีเพลที่ถูกจัดเก็บในขั้นตอนที่ 3.1 และรายการเว็บเซอร์วิสที่ถูกทดสอบในขั้นตอนที่ 3.3 ซึ่งการค้นหาเว็บเซอร์วิสที่ยังไม่ถูกทดสอบสามารถเรียกดูได้ทั้งในขณะที่กำลังดำเนินการทดสอบ และหลังจากดำเนินการทดสอบเสร็จสิ้นแล้ว โดยมีขั้นตอนการทำงานดังรูปที่ 3-15

จากตัวอย่างเส้นทางการไหลของดับเบิลยูเอส-บีเพลที่ถูกจัดเก็บในขั้นตอนที่ 3.1 ดังรูปที่ 3-5 เมื่อนำมาทดสอบกับเครื่องมือบีเพลเอนจิน ในขั้นตอนที่ 3.3 ผู้ทดสอบสามารถตรวจสอบเว็บเซอร์วิสที่ยังไม่ถูกทดสอบได้ โดยวิธีการนี้จะเรียกดูข้อมูลเส้นทางการไหลของบีเพลที่จัดเก็บไว้ในฐานข้อมูลขึ้นมาเปรียบเทียบกับประวัติการทดสอบ จากการเปรียบเทียบพบว่ากรณีทดสอบที่นำมาใช้จากรูปที่ 3-13 จะทดสอบผ่านเส้นทาง 1-2-3-4-5-6-7-11 ดังรูปที่ 3-14 จะพบโหนดที่มีการเรียกใช้งานเว็บเซอร์วิสแต่ยังไม่ถูกทดสอบคือโหนดที่ 9 ดังนั้นตัวอย่างนี้จะแสดงเว็บเซอร์วิสที่ยังไม่ถูกทดสอบ คือ “getLastVisit” ให้ผู้ทดสอบทราบ

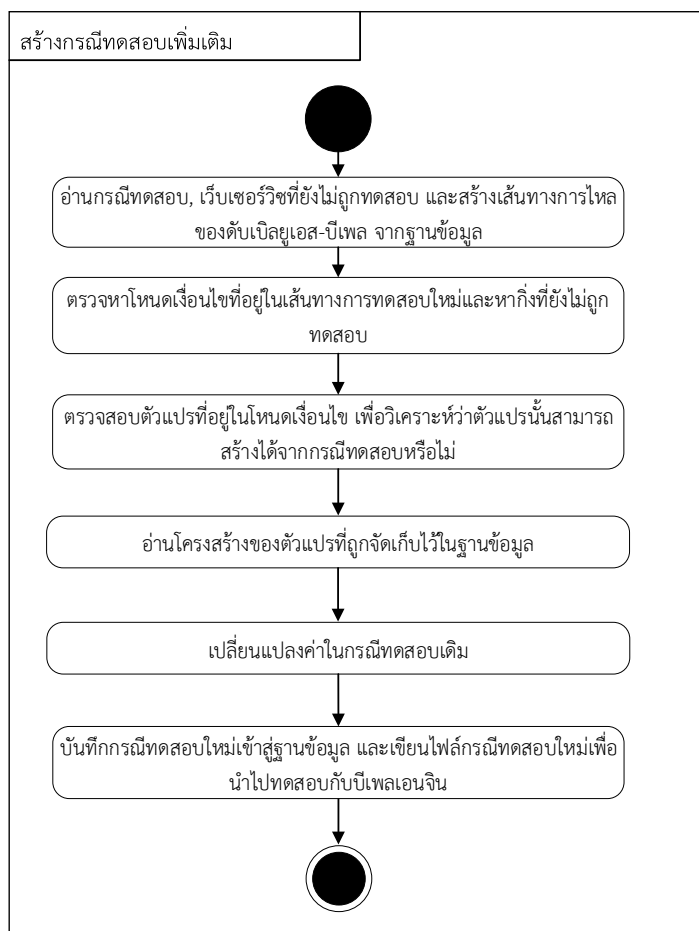


รูปที่ 3-15 การค้นหาเว็บเซอร์วิสที่ยังไม่ถูกทดสอบ

3.5 การสร้างกรณีทดสอบเพิ่มเติม

จากขั้นตอนที่ 3.4 เมื่อพบตรวจพบเว็บเซอร์วิสที่ยังไม่ถูกทดสอบ ในขั้นตอนนี้จะทำการค้นหาเส้นทางการทดสอบเพื่อนำมาสร้างกรณีทดสอบเพิ่มเติมให้ครอบคลุมเว็บเซอร์วิสที่ยังไม่ถูกทดสอบ โดยมีขั้นตอนการทำงาน ดังรูปที่ 3-16 และมีตัวอย่างการทำงานดังนี้

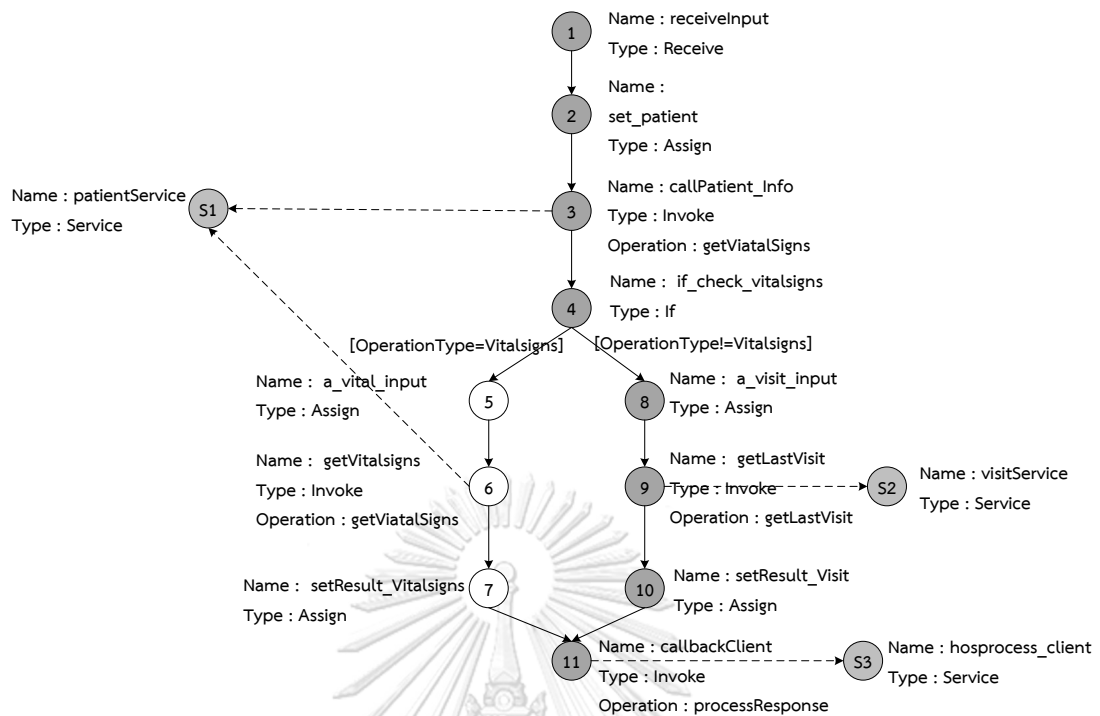
3.5.1. นำกรณีทดสอบที่ใช้ในขั้นตอนที่ 3.3 มาวิเคราะห์ข้อมูลนำเข้าของโครงการระดับเบต้าเอส-บีเฟลว่ามีข้อมูลนำเข้าใดบ้าง จากนั้นจะนำเว็บเซอร์วิสที่ยังไม่ถูกทดสอบจากขั้นตอนที่ 3.4 ซึ่งจากรูปที่ 3-14 ในขั้นตอนที่ 3.3 จะพบว่าโหนดที่เรียกเว็บเซอร์วิสที่ยังไม่ถูกทดสอบคือโหนดหมายเลข 9 โดยในขั้นตอนนี้จะเริ่มจากค้นหาโหนดเงื่อนไข ที่อยู่ในเส้นทางการไหลของโหนดที่ 9 จากตัวอย่างจะพบโหนดเงื่อนไขคือโหนดหมายเลข 4 ที่มีเงื่อนไข `OperationType!= 'Vitalsigns'` ดังรูปที่ 3-17



รูปที่ 3-16 การสร้างกรณีทดสอบเพิ่มเติม

3.5.2. ตรวจสอบตัวแปรที่ใช้ภายในโหนดเงื่อนไข เพื่อวิเคราะห์ว่าตัวแปรสามารถสร้างได้จากกรณีทดสอบหรือไม่ จากขั้นตอนที่ 3.5.1 พบว่าเงื่อนไขที่อยู่ในเส้นทางการไหลของโหนดที่ 9 คือ `OperationType != 'Vitalsigns'` ในขั้นตอนนี้จะพบตัวแปร `OperationType` ซึ่งเป็นตัวแปรที่สามารถสร้างได้จากกรณีทดสอบ ในกรณีนี้จะเปลี่ยนแปลงค่าตัวแปร `OperationType` เพื่อให้การทดสอบครอบคลุมเว็บเซอร์วิสที่ยังไม่ถูกทดสอบคือโหนดหมายเลข 9 โดยใช้วิธีการสุ่ม

3.5.3. ในกรณีที่ตัวแปรในโหนดเงื่อนไขสามารถสร้างได้จากกรณีทดสอบ จะอ่านโครงสร้างข้อมูลของตัวแปรต่างๆ ที่ถูกจัดเก็บในฐานข้อมูลเพิ่มเติมเพื่อสร้างกรณีทดสอบของตัวแปรนั้นใหม่ จากนั้นจะนำกรณีทดสอบในขั้นตอนที่ 3.3 มาเปลี่ยนแปลงค่าตัวแปร `OperationType` ที่มีประเภทข้อมูลเป็นตัวอักษร เพื่อให้ได้กรณีทดสอบใหม่ดังรูปที่ 3-18



รูปที่ 3-17 เส้นทางการทดสอบใหม่

```

<inputVariable>
  <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    name="payload">
    <process
      xmlns="http://xmlns.oracle.com/sample_HOS/sampleHos/HosProcess"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <hn>5900001</hn>
      <operationType>Test</operationType>
    </process>
  </part>
</inputVariable>
  
```

รูปที่ 3-18 ตัวอย่างกรณีทดสอบใหม่

3.5.4. บันทึกกรณีทดสอบใหม่ที่ได้เข้าสู่ฐานข้อมูลพร้อมทั้งเขียนไฟล์กรณีทดสอบใหม่เพื่อนำไปทดสอบกับบีเฟลเอนจิน และเก็บประวัติการสร้างกรณีทดสอบ

3.6 การสร้างรายงานผลการทดสอบ

ผู้ใช้งานสามารถเรียกดูผลการทดสอบ ที่จัดเก็บในฐานข้อมูล และพิมพ์เป็นรายงานสรุปการทดสอบได้ โดยรายงานสรุปมีดังนี้

3.6.1. รายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส โดยแสดงข้อมูลของเว็บเซอร์วิสที่ยังไม่ถูกเรียกใช้งาน เปรียบเทียบกับเว็บเซอร์วิสที่อยู่ในไฟล์ดับเบิ้ลยูเอส-บีเพล

3.6.2. รายงานผลการทดสอบการเรียกใช้เว็บเซอร์วิส โดยจะแสดงข้อมูลการทดสอบ ซึ่งประกอบด้วยข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิส ข้อมูลส่งออกที่ได้รับจากเว็บเซอร์วิส และ กรณีทดสอบที่นำไปทดสอบ



บทที่ 4

การออกแบบและพัฒนาเครื่องมือ

จากวิธีการที่นำเสนอในบทที่ 3 ผู้วิจัยนำวิธีการนี้มาพัฒนาเป็นเครื่องมือตรวจจับการเรียกใช้เว็บไซต์สำหรับทดสอบดับเบิลยูเอส-พีเพิล โดยนำภาษายูเอ็มแอลมาใช้เพื่อออกแบบและอธิบายฟังก์ชันการทำงานของเครื่องมือ ผู้วิจัยได้ออกแบบแผนภาพต่างๆ ได้แก่ แผนภาพยูสเคส (Use Case Diagram) แผนภาพกิจกรรม (Activity Diagram) แผนภาพคลาส (Class Diagram) แผนภาพลำดับ (Sequence Diagram) และแผนภาพการติดตั้ง (Deployment Diagram) นอกจากนี้ผู้วิจัยยังออกแบบฐานข้อมูลสำหรับจัดเก็บข้อมูลการทดสอบ พร้อมทั้งกำหนดสภาพแวดล้อมของเครื่องมือ

4.1. การออกแบบเครื่องมือ

ผู้วิจัยได้นำภาษายูเอ็มแอลมาใช้งาน เพื่ออธิบายโครงสร้างและการทำงานของเครื่องมือด้วยแผนภาพต่างๆ ดังนี้

4.1.1. แผนภาพยูสเคส

เป็นแผนภาพที่แสดงถึงฟังก์ชันการทำงานของเครื่องมือ โดยผู้วิจัยออกแบบแผนภาพดังกล่าว ดังรูปที่ 4-1 และได้อธิบายรายละเอียดแต่ละยูสเคสไว้ในภาคผนวก ก โดยยูสเคสประกอบด้วยฟังก์ชันการทำงาน 4 ส่วน ดังนี้

1. การอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-พีเพิล

ผู้ทดสอบจะต้องอัปโหลดไฟล์ที่เกี่ยวข้องกับโครงการดับเบิลยูเอส-พีเพิล ได้แก่ ไฟล์คอมไพล์ ไฟล์ดับเบิลยูเอสดีแอล ไฟล์ดับเบิลยูเอส-พีเพิล และไฟล์เอกซ์เอสดี โดยเครื่องมือจะนำไฟล์ที่ได้รับจากผู้ทดสอบมาประมวลผลเพื่อจัดเก็บเส้นทางการไหล จากนั้นจะสร้างกรณีทดสอบตั้งต้น และแทรกรหัสต้นทางเพื่อจัดเก็บประวัติการรับส่งข้อมูลนำเข้า และข้อมูลส่งออกระหว่างเว็บไซต์

2. การดูข้อมูลเว็บไซต์ที่ไม่ถูกทดสอบ

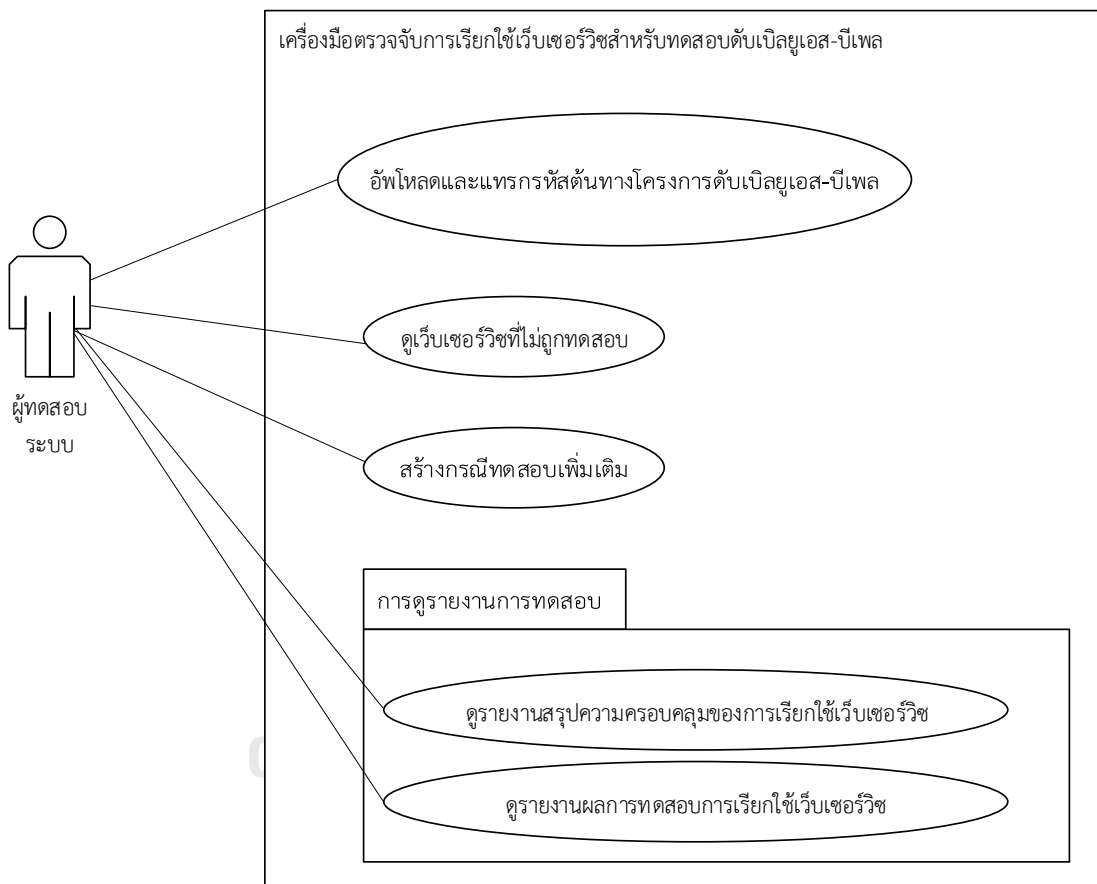
หลังจากที่ผู้ทดสอบนำไฟล์ที่ถูกแทรกรหัสต้นทางมาติดตั้งที่เครื่องแม่ข่าย และทำการทดสอบผ่านทางเครื่องมือพีเพิลเอนจินแล้ว ผู้ทดสอบสามารถดูเว็บไซต์ที่ยังไม่ถูกทดสอบผ่านทางเครื่องมือได้

3. การสร้างกรณีทดสอบเพิ่มเติม

หลังจากที่ผู้ทดสอบพบว่าเว็บเซอร์วิสที่ยังไม่ถูกทดสอบ ผู้ทดสอบสามารถเลือกเว็บเซอร์วิสนั้นๆ เพื่อสร้างกรณีทดสอบเพิ่มเติมได้

4. การดูรายงานผลการทดสอบ

ผู้ทดสอบสามารถเลือกโครงการดับเบิลยูเอส-บีเพล ที่ต้องการดูรายงานสรุปผลการทดสอบได้



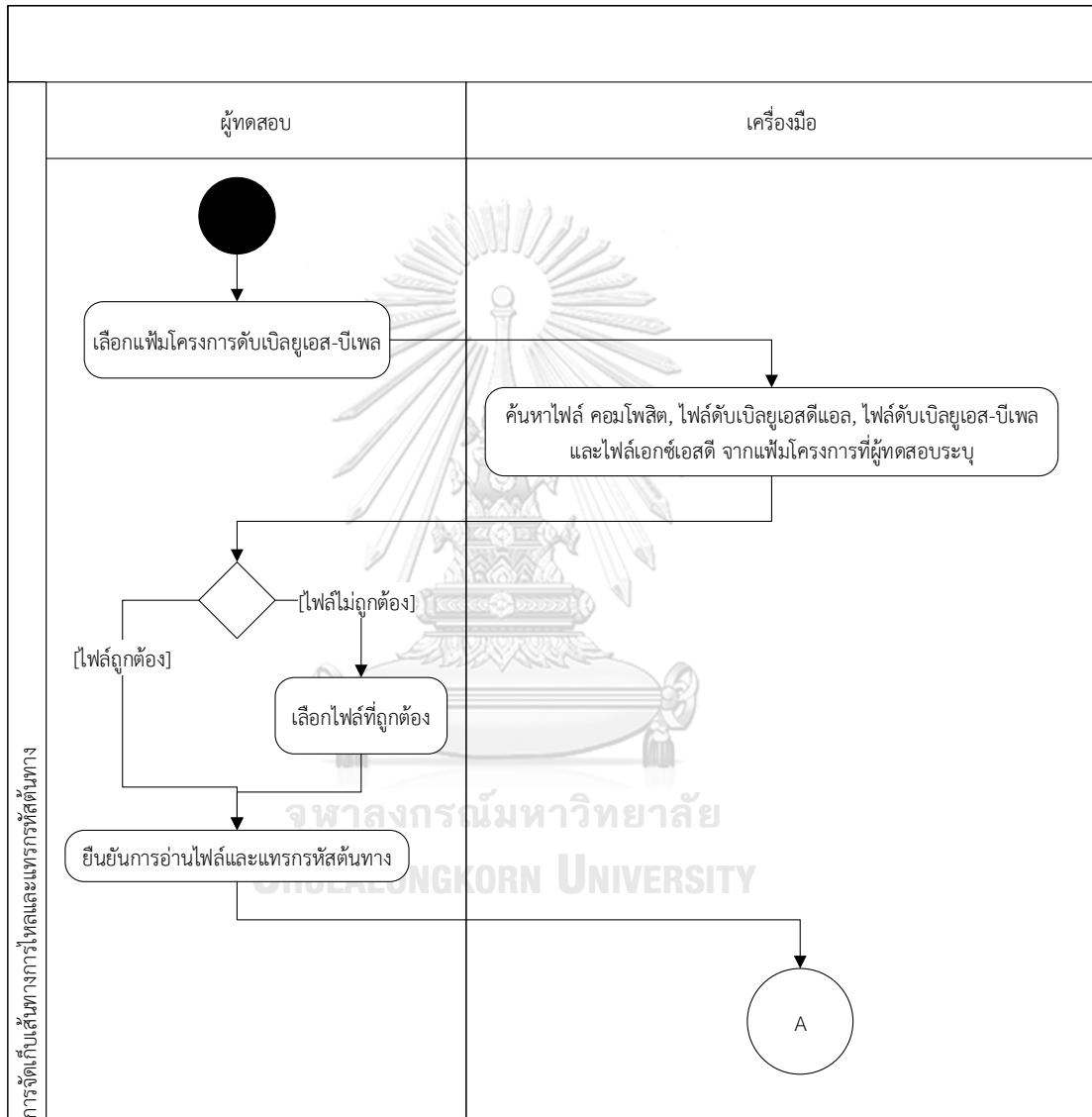
รูปที่ 4-1 แผนภาพยูสเคส เครื่องมือตรวจจับการเรียกใช้เว็บเซอร์วิสสำหรับทดสอบดับเบิลยูเอส-บีเพล

4.1.2. แผนภาพกิจกรรม

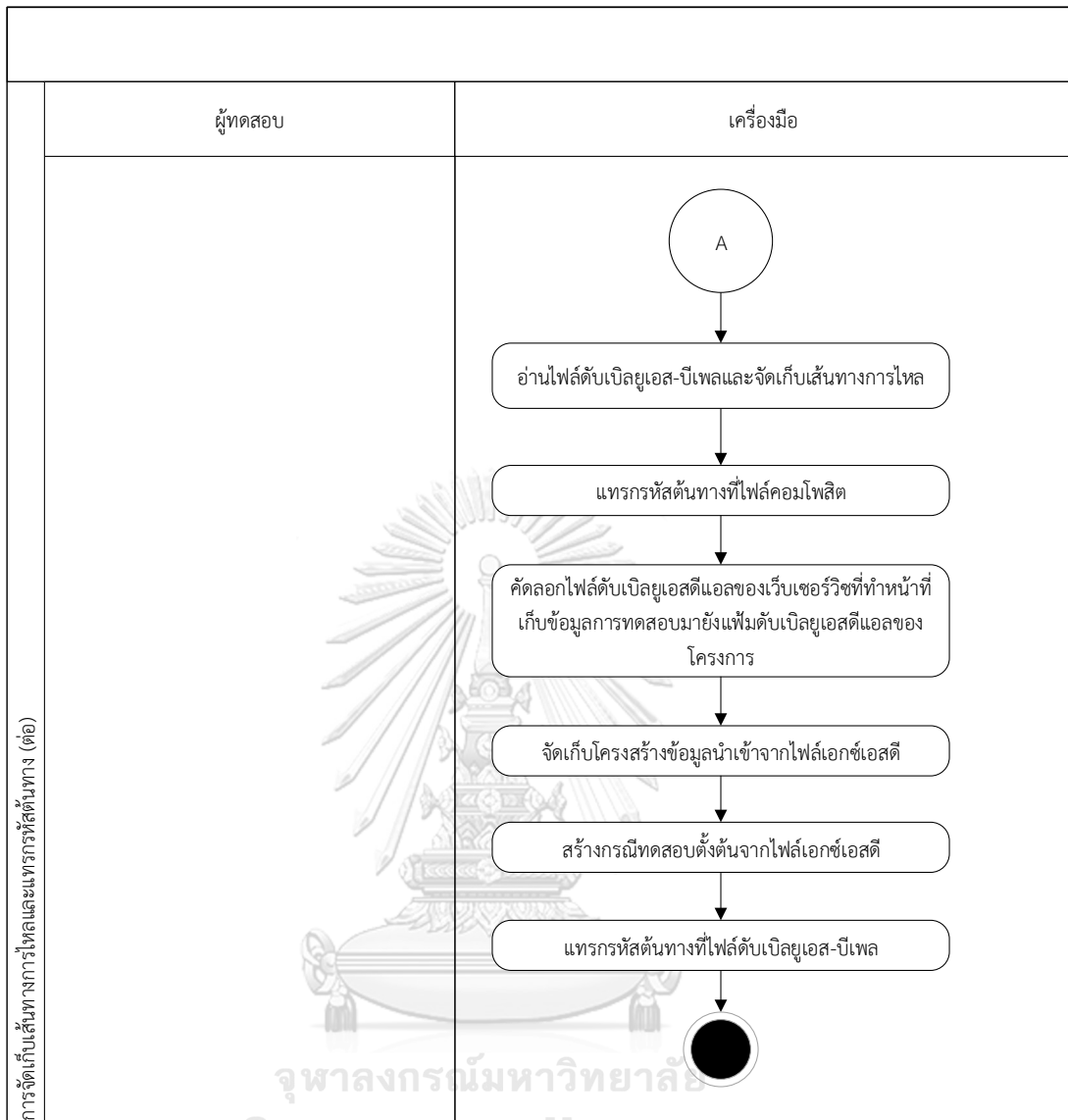
เป็นแผนภาพที่อธิบายกิจกรรมที่เกิดขึ้นระหว่างผู้ทดสอบและเครื่องมือ โดยในแผนภาพจะแสดงรายละเอียดกิจกรรมที่เกิดขึ้นภายในฟังก์ชันการทำงานต่างๆ ของเครื่องมือ ผู้วิจัยออกแบบแผนภาพกิจกรรมไว้ 4 ฟังก์ชันการทำงานดังนี้

1. แผนภาพกิจกรรมการจัดเก็บเส้นทางการไหลและแทรกรหัสต้นทาง

ผู้วิจัยออกแบบแผนภาพกิจกรรมสำหรับการจัดเก็บเส้นทางการไหลและแทรกรหัสต้นทางของดับเบิลยูเอส-บีเพล โดยมีลำดับกิจกรรมดังรูปที่ 4-2 และ 4-3



รูปที่ 4-2 แผนภาพกิจกรรมการจัดเก็บเส้นทางการไหลและแทรกรหัสต้นทาง ในขั้นตอนการค้นหาไฟล์



รูปที่ 4-3 แผนภาพกิจกรรมการจัดเก็บเส้นทางการไหลและแทรกรหัสต้นทาง ในขั้นตอนการจัดเก็บเส้นทางการไหลและแทรกรหัสต้นทาง

จากรูปที่ 4-2 และ 4-3 แสดงกิจกรรมการจัดเก็บเส้นทางการไหลและแทรกรหัสต้นทาง โดยเริ่มจากผู้ทดสอบเลือกแฟ้มโครงการดัดเบิลยูเอส-บีเพล ที่ต้องการจัดเก็บและแทรกรหัสต้นทาง จากนั้นเครื่องมือจะค้นหาไฟล์คอมโพสิต ไฟล์ดัดเบิลยูเอสดีแอล ไฟล์ดัดเบิลยูเอส-บีเพล และไฟล์เอกซ์เอสดี ที่จะนำมาจัดเก็บและแทรกรหัสต้นทาง เพื่อลดขั้นตอนการเลือกไฟล์ของผู้ทดสอบ โดยผู้ทดสอบจะต้องตรวจสอบและยืนยันการจัดไฟล์ต่างๆ จากนั้นเครื่องมืออ่านไฟล์ที่ผู้ทดสอบยืนยันแล้ว เพื่อจัดเก็บและแทรกรหัสต้นทาง โดยมีวิธีการจัดการกับไฟล์ต่างๆ ดังนี้

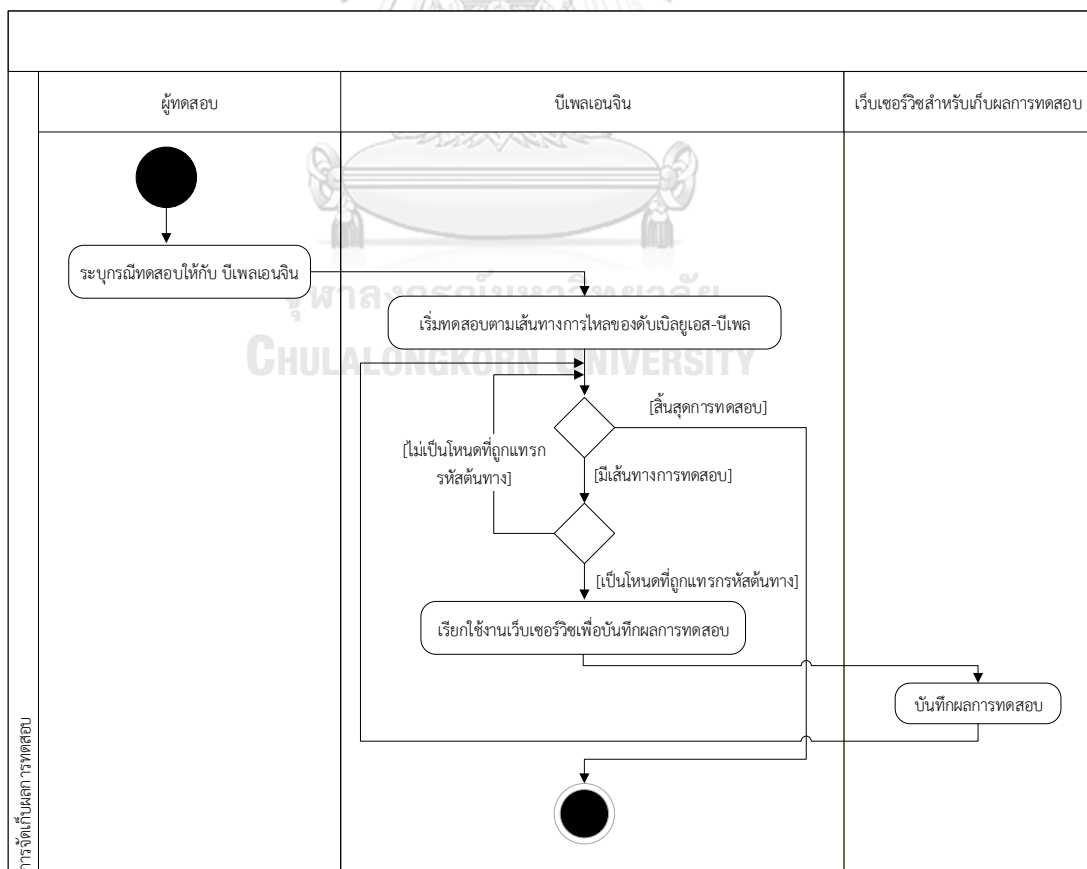
- ไฟล์ดัดเบิลยูเอส-บีเพล เครื่องมือจะจัดเก็บเส้นทางการไหลเข้าสู่ฐานข้อมูล

- ไฟล์คอมโปสิต เครื่องมือจะแทรกรหัสต้นทาง เพื่อให้โครงการดับเบิ้ลยูเอส-บีเพลสามารถเชื่อมต่อกับเว็บเซอร์วิสสำหรับจัดเก็บประวัติการทดสอบได้

หลังจากที่จัดเก็บเส้นทางการไหลและแทรกรหัสต้นทางที่ไฟล์คอมโปสิตแล้ว เครื่องมือจะคัดลอกไฟล์ดับเบิ้ลยูเอสดีแอล ของเว็บเซอร์วิสที่ทำหน้าที่เก็บข้อมูลการทดสอบมาเขียนลงในแฟ้มโครงการ เพื่อให้โครงการดับเบิ้ลยูเอส-บีเพลนี้ สามารถอ้างอิงการเรียกใช้งานเว็บเซอร์วิสที่ทำหน้าที่เก็บข้อมูลประวัติการทดสอบได้ จากนั้นจะสร้างกรณีทดสอบเบื้องต้นจากไฟล์เอกซ์เอสดี โดยอ่านไฟล์เอกซ์เอสดีจากผู้ทดสอบมาจัดเก็บลงสู่ฐานข้อมูล พร้อมทั้งนำไฟล์เอกซ์เอสดีนั้นมาสร้างกรณีทดสอบและจัดเก็บเข้าสู่ฐานข้อมูลเพื่อใช้สำหรับสร้างกรณีทดสอบภายหลัง จากนั้นเครื่องมือจะนำไฟล์ดับเบิ้ลยูเอส-บีเพลมาแทรกรหัสต้นทาง เพื่อจัดเก็บข้อมูลนำเข้าและส่งออก ระหว่างเว็บเซอร์วิส

2. แผนภาพกิจกรรมการจัดเก็บผลการทดสอบ

สำหรับการจัดเก็บผลการทดสอบที่เกิดขึ้นระหว่างการเรียกใช้งานเว็บเซอร์วิส ผู้วิจัยได้ออกแบบลำดับกิจกรรมดังรูปที่ 4-4

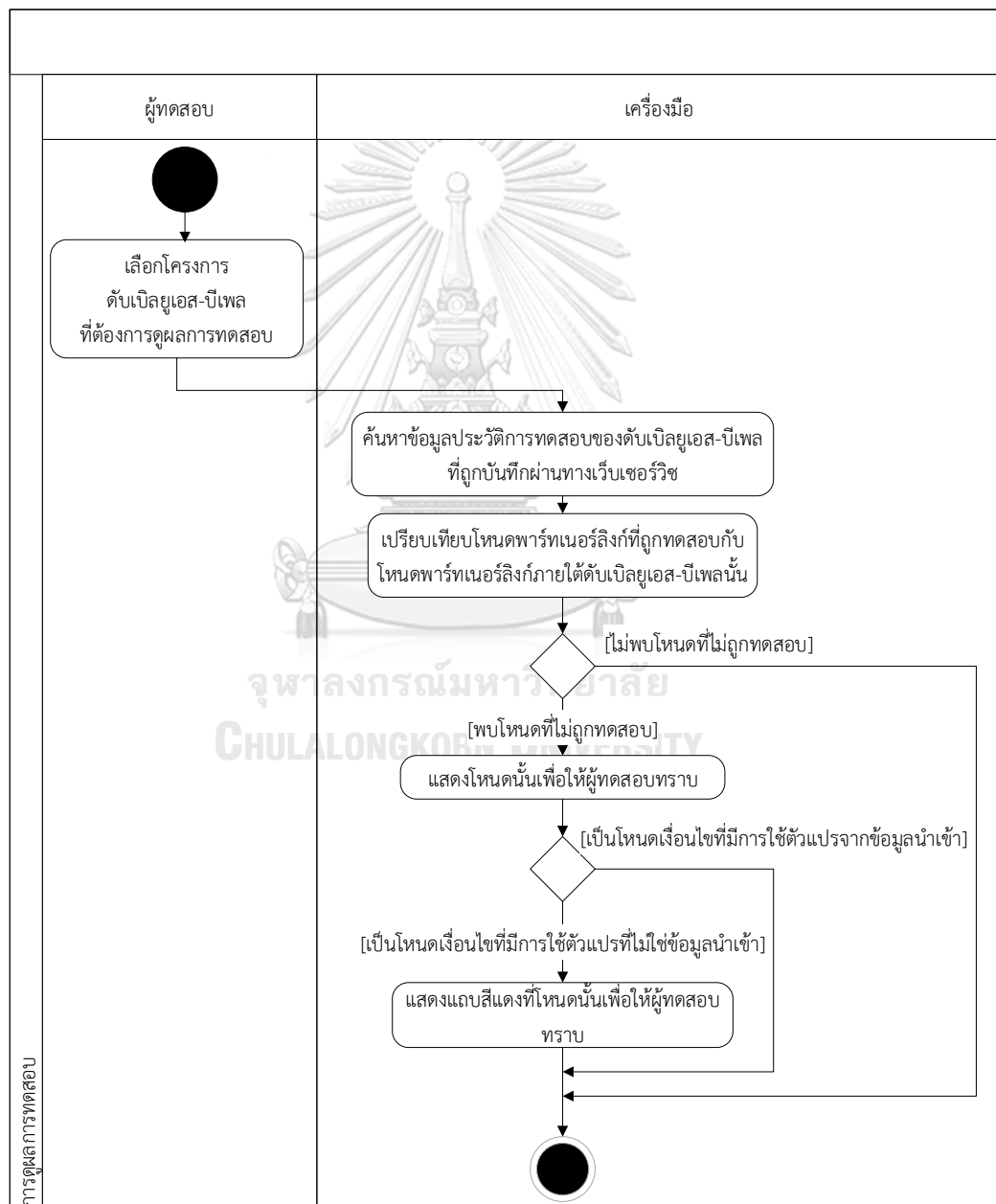


รูปที่ 4-4 แผนภาพกิจกรรมการจัดเก็บผลการทดสอบ

จากรูปที่ 4-4 แสดงกิจกรรมการจัดเก็บผลการทดสอบ โดยเริ่มจากผู้ทดสอบระบุกรณีทดสอบให้กับบีเพลเอนจิน และดำเนินการทดสอบ ซึ่งในระหว่างการทดสอบ เมื่อบีเพลเอนจินอ่านพบโหนดที่ถูกแทรกรหัสต้นทางไว้ บีเพลเอนจินจะส่งข้อมูลการทดสอบนั้นไปจัดเก็บที่ฐานข้อมูลผ่านทางเว็บเซอร์วิส

3. แผนภาพกิจกรรมการดูผลการทดสอบ

สำหรับการดูผลการทดสอบ ผู้วิจัยได้ออกแบบลำดับกิจกรรมดังรูปที่ 4-5

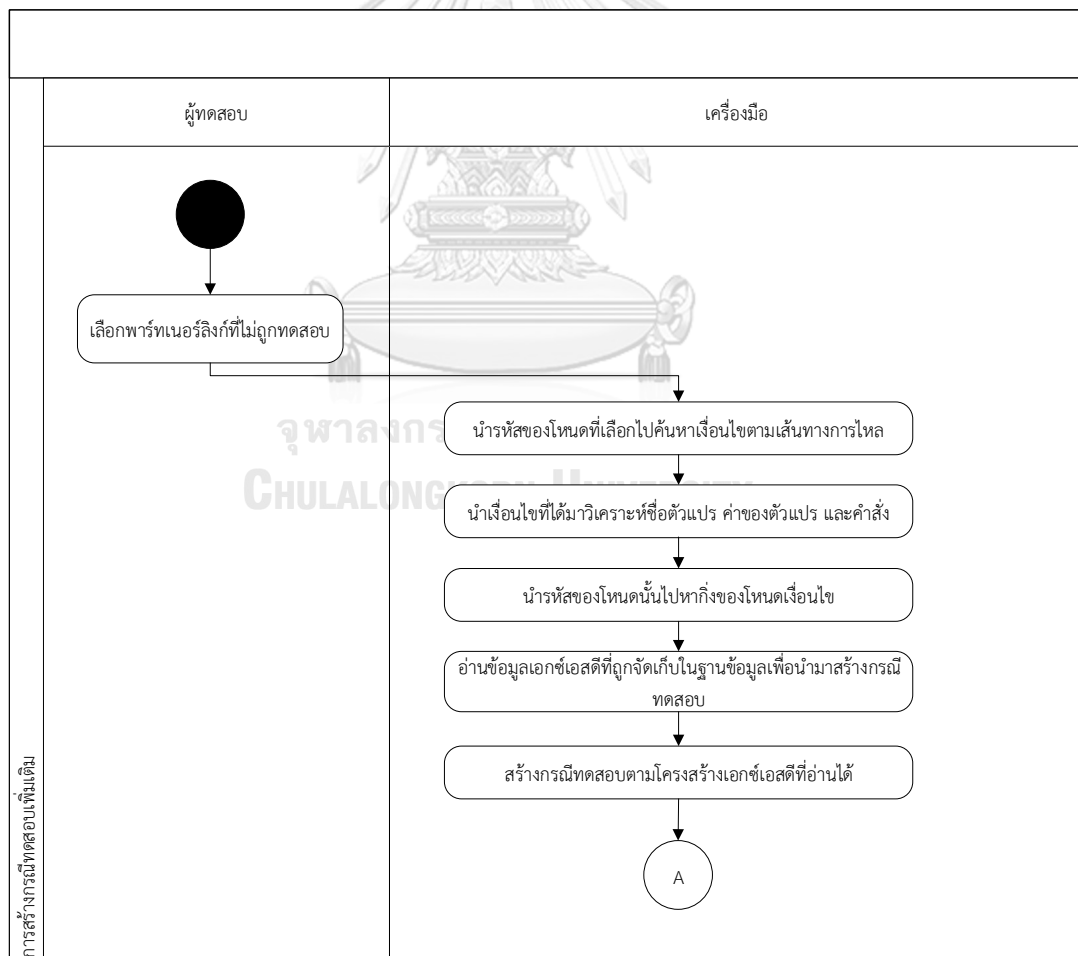


รูปที่ 4-5 แผนภาพกิจกรรมการดูผลการทดสอบ

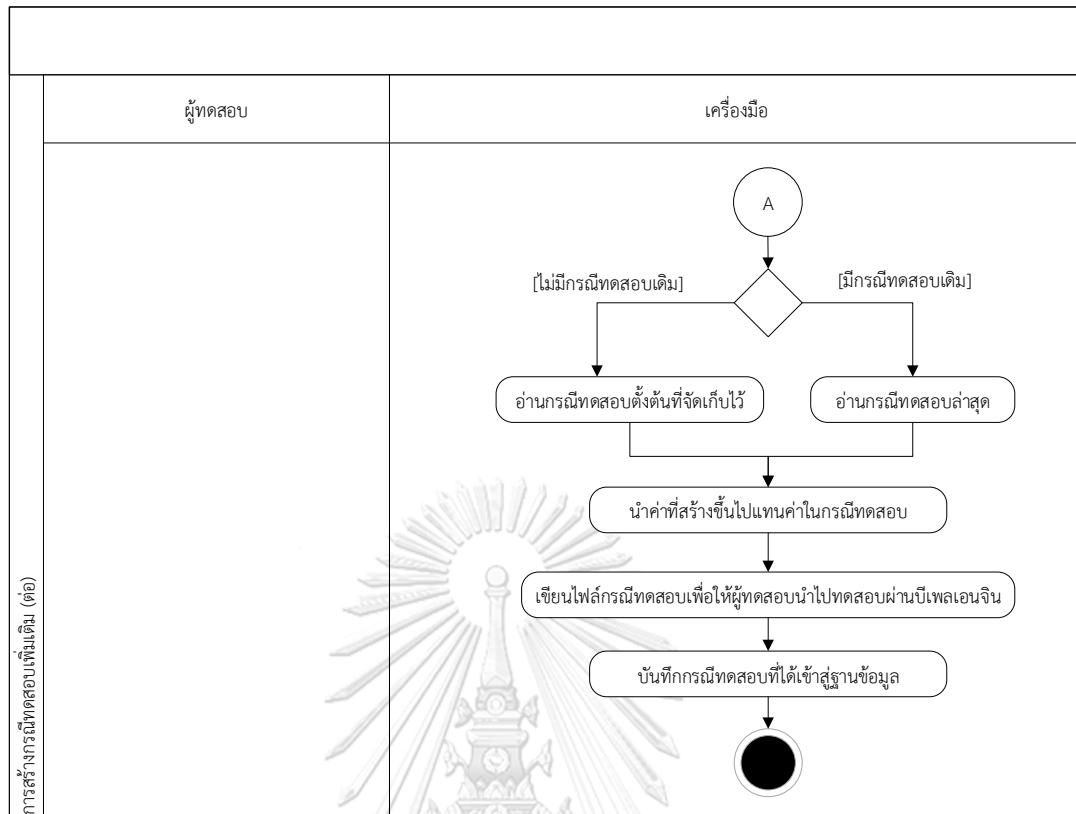
จากรูปที่ 4-5 แสดงกิจกรรมการดูแลการทดสอบ โดยเริ่มจากผู้ทดสอบเลือกโครงการที่ต้องการดูแลการทดสอบ เพื่อให้เครื่องมือค้นหาข้อมูลประวัติการทดสอบของดับเบิลยูเอส-บีเพลที่ถูกบันทึกผ่านทางเว็บเซอร์วิส โดยเครื่องมือจะเปรียบเทียบโหนดพาร์ทเนอร์ลิงก์ที่ถูกทดสอบกับโหนดพาร์ทเนอร์ลิงก์ทั้งหมดที่อยู่ภายใต้ดับเบิลยูเอส-บีเพลนั้น ในกรณีที่พบโหนดพาร์ทเนอร์ลิงก์ที่ไม่ถูกทดสอบ เครื่องมือจะแสดงโหนดนั้นให้กับผู้ทดสอบทราบ จากนั้นเครื่องมือจะค้นหาเส้นทางการไหลจากโหนดเริ่มต้นมายังโหนดของพาร์ทเนอร์ลิงก์นั้น และจะตรวจสอบโหนดเงื่อนไขที่อยู่ในเส้นทางการไหลนั้น ว่าภายในเงื่อนไขใช้ตัวแปรที่มาจากข้อมูลนำเข้าหรือไม่ ในกรณีที่ใช้ตัวแปรอื่นที่ไม่ใช่ตัวแปรนำเข้า เครื่องมือจะแสดงแถบสีแดงที่รายการนั้นเพื่อให้ผู้ทดสอบทราบว่าในกรณีนี้จะไม่รองรับการสร้างกรณีทดสอบเพิ่มเติม

4. แผนภาพกิจกรรมการสร้างกรณีทดสอบเพิ่มเติม

สำหรับการสร้างกรณีทดสอบเพิ่มเติม ผู้วิจัยได้ออกแบบลำดับกิจกรรมดังรูปที่ 4-6 และ 4-7



รูปที่ 4-6 แผนภาพกิจกรรมการสร้างกรณีทดสอบเพิ่มเติม ในขั้นตอนการสร้างกรณีทดสอบ

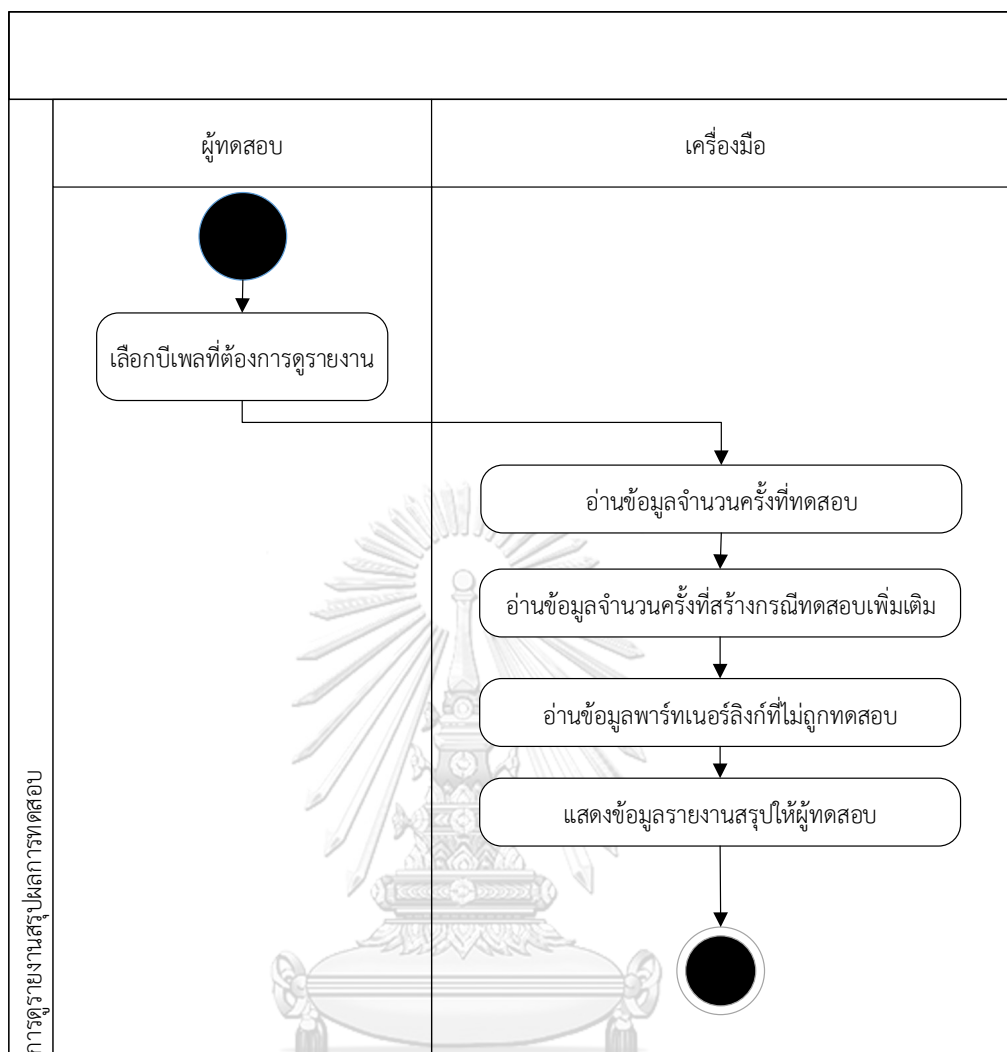


รูปที่ 4-7 แผนภาพกิจกรรมการสร้างกรณีทดสอบเพิ่มเติม ในขั้นตอนการบันทึกกรณีทดสอบใหม่

จากรูปที่ 4-6 และ 4-7 แสดงกิจกรรมการสร้างกรณีทดสอบเพิ่มเติม โดยเริ่มจากผู้ทดสอบเลือกโหนดพาร์ทเนอร์ลิงก์ที่ไม่ถูกทดสอบและต้องการสร้างกรณีทดสอบเพิ่มเติม จากนั้นเครื่องมือจะนำรหัสของโหนดนั้นไปค้นหาเงื่อนไขที่ต้องถูกทดสอบตามเส้นทางการไหล และนำเงื่อนไขที่ได้มาวิเคราะห์ชื่อตัวแปร ค่าของตัวแปร และคำสั่งการประมวลผลเพื่อนำไปใช้เป็นเงื่อนไขการสร้างค่ากรณีทดสอบใหม่ จากนั้นจะนำข้อมูลเอกซ์เอสดี ที่ถูกจัดเก็บในฐานข้อมูลมาสร้างกรณีทดสอบ ในกรณีทีโครงการดับเบิ้ลยูเอส-บีเฟลเอนจิน เคยมีประวัติการทดสอบแล้ว เครื่องมือจะอ่านค่ากรณีทดสอบล่าสุดเพื่อเปลี่ยนแปลงค่าให้เป็นไปตามกรณีทดสอบใหม่ แต่ในกรณีที่ยังไม่มีการทดสอบเครื่องมือจะอ่านค่ากรณีทดสอบดั้งเดิมที่จัดเก็บในฐานข้อมูลขึ้นมาเพื่อเปลี่ยนแปลงค่าตามกรณีทดสอบใหม่แทน หลังจากที่เครื่องมือสร้างกรณีทดสอบใหม่แล้ว เครื่องมือจะเขียนไฟล์กรณีทดสอบใหม่เพื่อให้ผู้ทดสอบนำไปทดสอบผ่านบีเฟลเอนจินได้ พร้อมทั้งจัดเก็บกรณีทดสอบใหม่เข้าสู่ฐานข้อมูลเพื่อเก็บประวัติการสร้างกรณีทดสอบนั้น

5. แผนภาพกิจกรรมการดูรายงานสรุปผลการทดสอบ

การดูรายงานการทดสอบ ผู้วิจัยได้ออกแบบลำดับกิจกรรมดังรูปที่ 4-8



รูปที่ 4-8 แผนภาพกิจกรรมการดูรายงานสรุปผลการทดสอบ

จากรูปที่ 4-8 แสดงกิจกรรมการดูรายงานสรุปผลการทดสอบ โดยเริ่มจากผู้ทดสอบเลือกปีเพลาที่ต้องการดูรายงานสรุป จากนั้นเครื่องมือจะอ่านข้อมูลจำนวนครั้งที่ทดสอบ จำนวนครั้งในการสร้างกรณีทดสอบเพิ่มเติม และอ่านข้อมูลพาร์ทเนอร์ลิงก์ที่ยังไม่ถูกทดสอบเพื่อแสดงในรายงาน

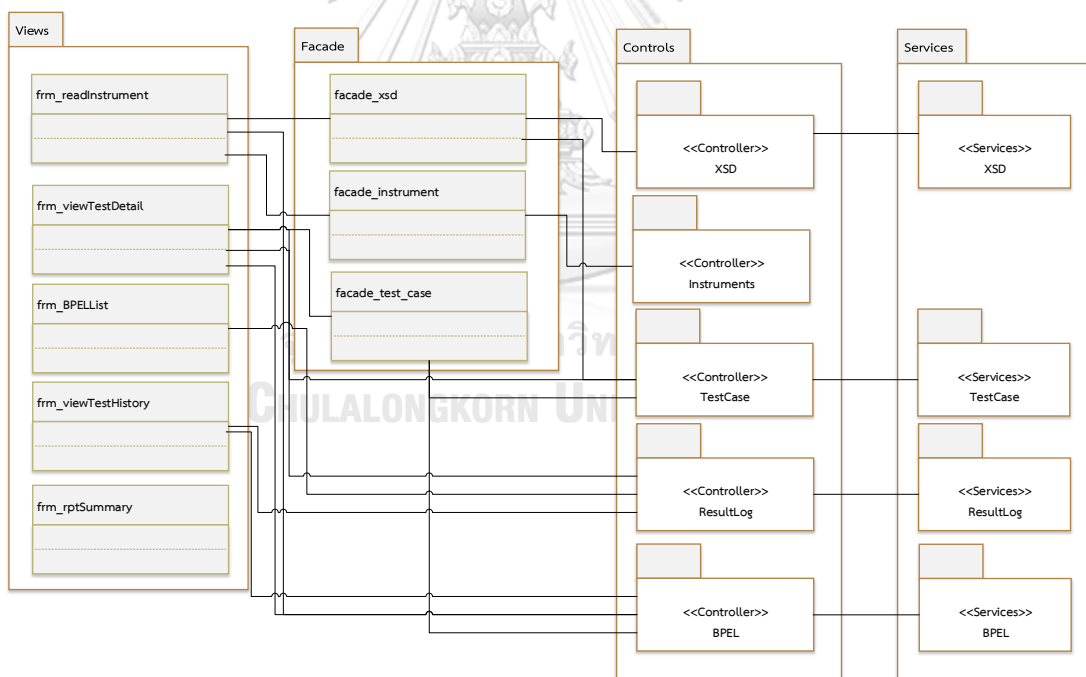
4.1.3. แผนภาพคลาส

เป็นแผนภาพที่อธิบายโครงสร้างของคลาสภายในเครื่องมือ โดยผู้วิจัยออกแบบแผนภาพคลาสไว้ ดังรูปที่ 4-9 และได้ออกแบบคลาสเป็น 4 ประเภท ได้แก่

- คลาสส่วนต่อประสาน ทำหน้าที่เป็นหน้าต่างติดต่อกับผู้ใช้งาน โดยภายในคลาสประเภทนี้จะมีการทำงานสองส่วนคือ คลาสส่วนต่อประสาน และคลาสควบคุมที่ทำงานร่วมกับส่วนต่อประสาน

- คลาสควบคุม (Controller) เป็นคลาสที่ทำหน้าที่ประมวลผล โดยนำข้อมูลจากส่วนต่อประสานมาประมวลผล และนำผลที่ได้กลับไปแสดงยังส่วนต่อประสานได้ โดยคลาสนี้จะติดต่อกับคลาสเซอร์วิสเพื่อดึงข้อมูลที่จัดเก็บอยู่ในฐานข้อมูลได้
- คลาสพะชาด (Facade) เป็นคลาสที่ทำหน้าที่ช่วยให้การเรียกใช้งานคลาสควบคุมที่มีความซับซ้อน สามารถเรียกใช้งานได้ให้ง่ายขึ้น โดยคลาสนี้จะทำหน้าที่ช่วยเรียกใช้งานคลาสควบคุมต่างๆ ตามลำดับที่ถูกกำหนดไว้
- คลาสเซอร์วิส (Service) เป็นคลาสที่ติดต่อกับฐานข้อมูล โดยคลาสนี้จะนำข้อมูลที่ผ่านการประมวลผลจากคลาสควบคุมไปบันทึกเข้าสู่ฐานข้อมูล และคลาสนี้ยังสามารถอ่านข้อมูลจากฐานข้อมูล เพื่อส่งต่อให้คลาสควบคุมนำข้อมูลไปใช้งานต่อ และส่งต่อไปแสดงที่คลาสส่วนต่อประสานได้

และนอกจากนี้ผู้วิจัยยังออกแบบคลาสเอนทิตี (Entity) ซึ่งเป็นคลาสทำหน้าที่เก็บข้อมูลสำหรับประมวลผล หรือรับส่งข้อมูลระหว่างคลาสต่างๆ ได้



รูปที่ 4-9 แผนภาพคลาส

จากรูปที่ 4-9 สามารถอธิบายความรับผิดชอบการทำงานร่วมกันของคลาส (Class Responsibilities and Collaborators Card: CRC Card) ดังตารางที่ 4-1 ถึง 4-31

ตารางที่ 4-1 CRC ของคลาส frm_readInstrument

ชื่อคลาส: frm_readInstrument	ชนิด: ส่วนต่อประสาน
คำอธิบาย: เป็นคลาสส่วนต่อประสานมีหน้าที่จัดเก็บเส้นทางการไหลของไฟล์ดับเบิลยูเอส-บีเพล และนำไฟล์ที่ผู้ทดสอบอัปโหลดมาแทรกรหัสต้นทาง	
ความรับผิดชอบ:	
+ doProcess_read_BPEL	อ่านและจัดเก็บเส้นทางการไหลของโครงการดับเบิลยูเอส-บีเพล
+ generateInputFromXSD	สร้างกรณีทดสอบจากไฟล์เอกซ์เอสดี
+ doInstrumentComposite	แทรกรหัสต้นทางที่ไฟล์คอมโพสิต
+ doInstrumentWSDL	แทรกรหัสต้นทางโดยเขียนไฟล์ดับเบิลยูเอสดีแอล สำหรับจัดเก็บข้อมูลการทดสอบเข้าสู่แฟ้มของโครงการ
+ doInstrumentBPEL	แทรกรหัสต้นทางที่ไฟล์ดับเบิลยูเอส-บีเพล
คุณสมบัติ:	
- txt_project_folder (กล่องข้อความ)	กล่องข้อความสำหรับเส้นทางของแฟ้มโครงการ
- txt_composite_path (กล่องข้อความ)	กล่องข้อความสำหรับเส้นทางของไฟล์คอมโพสิต
- txt_wsdl_path (กล่องข้อความ)	กล่องข้อความสำหรับเส้นทางของแฟ้มดับเบิลยูเอสดีแอล
- txt_MainBPEL_filePath (กล่องข้อความ)	กล่องข้อความสำหรับเส้นทางของไฟล์ดับเบิลยูเอส-บีเพล
- txt_BPEL_path (กล่องข้อความ)	กล่องข้อความสำหรับเส้นทางของแฟ้มดับเบิลยูเอส-บีเพล
- btn_project_browse (ปุ่ม)	ปุ่มค้นหาเส้นทางของแฟ้มโครงการ
- btn_compo_browse (ปุ่ม)	ปุ่มค้นหาเส้นทางของไฟล์คอมโพสิต
- btn_wsdl_browse (ปุ่ม)	ปุ่มค้นหาเส้นทางของแฟ้มดับเบิลยูเอสดีแอล
- btn_wsdl_File_browse (ปุ่ม)	ปุ่มค้นหาเส้นทางของไฟล์ดับเบิลยูเอสดีแอล
- btn_bpel_File_browse (ปุ่ม)	ปุ่มค้นหาเส้นทางของไฟล์ดับเบิลยูเอส-บีเพล
- btn_xsd_browse (ปุ่ม)	ปุ่มค้นหาเส้นทางของไฟล์เอกซ์เอสดี
- btn_clear (ปุ่ม)	ปุ่มล้างข้อมูลเส้นทาง
- btn_readAndInstruemnt (ปุ่ม)	ปุ่มอ่านและแทรกรหัสต้นทาง

ตารางที่ 4-1 CRC ของคลาส frm_readInstrument (ต่อ)

ชื่อคลาส: frm_readInstrument	ชนิด: ส่วนต่อประสาน
<p>ความสัมพันธ์:</p> <p>Composition:</p> <p style="padding-left: 40px;">BPEL.Ctrl_BPEL</p> <p style="padding-left: 40px;">facade_xsd</p> <p style="padding-left: 40px;">facade_instrument</p>	

ตารางที่ 4-2 CRC ของคลาส frm_viewTestDetail

ชื่อคลาส: frm_viewTestDetail	ชนิด: ส่วนต่อประสาน
<p>คำอธิบาย:</p> <p>เป็นคลาสส่วนต่อประสาน ทำหน้าที่แสดงรายละเอียดโครงการของดัมเบิลยูเอส-เพลที่ผู้ทดสอบเลือก</p>	
<p>ความรับผิดชอบ:</p> <ul style="list-style-type: none"> - setHeader แสดงข้อมูลโครงการบนหน้าต่าง - frm_viewTestDetail_Load อ่านข้อมูลรายละเอียดของโครงการดัมเบิลยูเอส-เพล มาแสดงโดยจะเรียกฟังก์ชันต่างๆ ต่อไป - loadMaster อ่านข้อมูลประวัติการทดสอบมาแสดงบน dtg_testMaster - loadUnExecute อ่านข้อมูลเว็บเซอร์วิชที่ไม่ถูกทดสอบ และนำมาแสดงผล - enableGenerateTestCase ตรวจสอบเว็บเซอร์วิชที่ยังไม่ถูกทดสอบ และเปิด/ปิดปุ่มสร้างกรณีทดสอบ - btn_generateTestCase_Click สร้างกรณีทดสอบเพิ่มเติมจากรายการเว็บเซอร์วิชที่ยังไม่ถูกทดสอบ - loadTestCaseHistory อ่านข้อมูลกรณีทดสอบที่นำมาใช้ล่าสุดเพื่อตรวจสอบว่าเว็บเซอร์วิชใดสามารถสร้างกรณีทดสอบเพิ่มเติมได้ โดยตรวจสอบจากชื่อตัวแปรและเงื่อนไขที่ใช้ - findChildNode ตรวจสอบหาโหนดลูกที่อยู่ภายใต้แท็กเอกซ์เอ็มแอล - dtg_unExecute_CellDoubleClick สร้างกรณีทดสอบเพิ่มเติมจากเว็บเซอร์วิชที่เลือก - doCreateAdditionalTestCase สร้างกรณีทดสอบเพิ่มเติมจากค่าตัวแปรที่ได้รับมา 	

ตารางที่ 4-2 CRC ของคลาส frm_viewTestDetail (ต่อ)

ชื่อคลาส: frm_viewTestDetail	ชนิด: ส่วนต่อประสาน
<p>ความรับผิดชอบ:</p> <ul style="list-style-type: none"> - doGenerateTestCase ทำหน้าที่สร้างกรณีทดสอบ - btn_viewReport_Click เรียกใช้งานฟังก์ชันการแสดงผลงานสรุปผลการทดสอบ - openSummaryReport แสดงรายงานสรุปผลการทดสอบ - countNotConverageOper นับจำนวนเว็บเซอร์วิชที่ไม่ถูกทดสอบ - dtg_testMaster_DoubleClick แสดงข้อมูลรายละเอียดผลการทดสอบที่เลือก - btn_openTCDirectory_Click เปิดหน้าต่างเพิ่มจัดเก็บไฟล์กรณีทดสอบที่ถูกสร้างจากเครื่องมือ 	
<p>คุณสมบัติ:</p> <ul style="list-style-type: none"> - dtg_unExecute (ตารางข้อมูล) ตารางแสดงข้อมูลเว็บเซอร์วิชที่ยังไม่ถูกทดสอบ - dtg_testMaster (ตารางข้อมูล) ตารางแสดงข้อมูลประวัติการทดสอบ - btn_generateTestCase (ปุ่ม) ปุ่มสำหรับสร้างกรณีทดสอบเพิ่มเติม - btn_viewReport (ปุ่ม) ปุ่มสำหรับดูรายงานสรุป - btn_openTCDirectory (ปุ่ม) ปุ่มสำหรับดูเพิ่มจัดเก็บกรณีทดสอบที่ถูกสร้างจากเครื่องมือ 	
<p>ความสัมพันธ์:</p> <p>Composition:</p> <ul style="list-style-type: none"> ctrl_TestResult facade_testcase ctrl_TestCase ctrl_BPEL_Node frm_viewTestHistory frm_rptSummary Entities_TestCaseData Entities_DrivePathCondition 	

ตารางที่ 4-3 CRC ของคลาส frm_BPELList

ชื่อคลาส: frm_BPELList	ชนิด: ส่วนต่อประสาน
คำอธิบาย: เป็นคลาสส่วนต่อประสาน ทำหน้าที่แสดงรายการโครงการของดับเบิลยูเอส-พีเพิลที่จัดเก็บอยู่ในระบบ	
ความรับผิดชอบ:	
- btn_search_Click	ค้นหาข้อมูลโครงการดับเบิลยูเอส-พีเพิล โดยจะเรียกใช้งานฟังก์ชัน doSearch เพื่อค้นหา
- doSearch	ค้นหาข้อมูลโครงการดับเบิลยูเอส-พีเพิล
-dtg_partnetLinkName_CellDoubleClick	แสดงหน้าต่างรายละเอียดของโครงการดับเบิลยูเอส-พีเพิลที่เลือก โดยจะเรียกใช้งานคลาสส่วนต่อประสาน frm_viewTestDetail
- txt_search_jobID_KeyPress	ตรวจสอบการกดแป้นพิมพ์ 'Enter' เพื่อเรียกใช้งานฟังก์ชัน doSearch
- txt_partnerLink_name_KeyPress	ตรวจสอบการกดแป้นพิมพ์ 'Enter' เพื่อเรียกใช้งานฟังก์ชัน doSearch
คุณสมบัติ:	
- txt_search_jobID (กล่องข้อความ)	กล่องข้อความสำหรับค้นหารหัสโครงการดับเบิลยูเอส-พีเพิล
- txt_partnerLink_name (กล่องข้อความ)	กล่องข้อความสำหรับค้นหาชื่อโครงการดับเบิลยูเอส-พีเพิล
- btn_search (ปุ่ม)	ปุ่มค้นหาข้อมูลโครงการดับเบิลยูเอส-พีเพิล
- dtg_partnetLinkName (ตารางข้อมูล)	ตารางแสดงข้อมูลโครงการดับเบิลยูเอส-พีเพิล
ความสัมพันธ์:	
Composition:	
frm_viewTestDetail	
ctrl_TestResult	

ตารางที่ 4-4 CRC ของคลาส frm_viewTestHistory

ชื่อคลาส: frm_viewTestHistory	ชนิด: คลาสส่วนต่อประสาน
คำอธิบาย: เป็นคลาสสำหรับแสดงประวัติการทดสอบ	
<p>ความรับผิดชอบ:</p> <ul style="list-style-type: none"> - frm_viewTestHistory_Load อ่านข้อมูลประวัติการทดสอบจากข้อมูลที่ได้รับเข้ามา + setID ตั้งค่าข้อมูลสำหรับค้นหาประวัติ + renderMessagetext อ่านข้อมูลเอกซ์เอ็มแอลและแปลงให้อยู่ในรูปแบบข้อความ - dtg_partnetLinkName_CellClick แสดงข้อมูลตัวแปรนำเข้าและส่งออกของโหนดที่เลือก - btn_report_Click เรียกใช้งานฟังก์ชันแสดงรายงานประวัติการทดสอบ - callHistoryReport เรียกใช้งานคลาสออกรายงานประวัติการทดสอบ 	
<p>คุณสมบัติ:</p> <ul style="list-style-type: none"> - dtg_partnetLinkName (ตารางข้อมูล) ตารางแสดงข้อมูลโหนดที่ถูกทดสอบ - txt_messageBefore (กล่องข้อความ) กล่องแสดงข้อความ ข้อมูลนำเข้า - txt_messageAfter (กล่องข้อความ) กล่องแสดงข้อความ ข้อมูลส่งออก 	
<p>ความสัมพันธ์:</p> <p>Composition:</p> <ul style="list-style-type: none"> ctrl_TestResult frm_rptHistory frm_viewTestDetail 	

ตารางที่ 4-5 CRC ของคลาส frm_rptSummary

ชื่อคลาส: frm_rptSummary	ชนิด: คลาสส่วนต่อประสาน
คำอธิบาย: เป็นคลาสสำหรับแสดงรายงานสรุปการทดสอบของโครงการดับเบิลยูเอส-บีเพล	

ตารางที่ 4-5 CRC ของคลาส frm_rptSummary (ต่อ)

ชื่อคลาส: frm_rptSummary	ชนิด: คลาสส่วนต่อประสาน
<p>ความรับผิดชอบ:</p> <ul style="list-style-type: none"> - frm_rptSummary_Load นำข้อมูลที่ระบุมาแสดงบนรายงาน + setDS ตั้งค่าชุดข้อมูลที่นำมาแสดงบนรายงาน + setFormular ตั้งค่าข้อมูลสูตรที่แสดงบนรายงาน 	
<p>คุณสมบัติ:</p> <ul style="list-style-type: none"> - crystalReportViewer1 (กล่องแสดงรายงาน) กล่องแสดงรายงานสรุปการทดสอบ - rpt_summary1 (รายงาน) รายงานสรุปการทดสอบ 	
<p>ความสัมพันธ์:</p> <p>Composition:</p> <p style="text-align: center;">frm_viewTestDetail</p>	

ตารางที่ 4-6 CRC ของคลาส frm_rptHistory

ชื่อคลาส: frm_rptHistory	ชนิด: คลาสส่วนต่อประสาน
<p>คำอธิบาย:</p> <p>เป็นคลาสสำหรับแสดงรายงานประวัติการทดสอบ</p>	
<p>ความรับผิดชอบ:</p> <ul style="list-style-type: none"> - frm_rptSummary_Load นำข้อมูลที่ระบุมาแสดงบนรายงาน + setDS ตั้งค่าชุดข้อมูลที่นำมาแสดงบนรายงาน + setFormular ตั้งค่าข้อมูลสูตรที่แสดงบนรายงาน 	
<p>คุณสมบัติ:</p> <ul style="list-style-type: none"> - crystalReportViewer1 (กล่องแสดงรายงาน) กล่องแสดงรายงานประวัติการทดสอบ - rpt_testHistoryDetails1 (รายงาน) รายงานประวัติการทดสอบ 	
<p>ความสัมพันธ์:</p> <p>Composition:</p> <p style="text-align: center;">frm_viewTestHistory</p>	

ตารางที่ 4-7 CRC ของคลาส facade_instrument

ชื่อคลาส: facade_instrument	ชนิด: พะชาด
คำอธิบาย: เป็นคลาสพะชาดมีหน้าที่เรียกใช้งานคลาสควบคุมต่างๆ เพื่อแทรกรหัสต้นทาง ไฟล์ที่ผู้ทดสอบอัปโหลดเข้ามา	
ความรับผิดชอบ:	
+ instrumentComposite	แทรกรหัสต้นทางที่ไฟล์คอมไพล์
+ InstrumentCopyFile	คัดลอกไฟล์ดับเบิลยูเอสดีแอลสำหรับเก็บประวัติการทดสอบ
+ InstrumentBPEL	เรียกใช้งานฟังก์ชันแทรกรหัสต้นทาง ในแต่ละไฟล์
- instrumentLogVariable	แทรกรหัสต้นทางในส่วนของการประกาศตัวแปร
- instrumentLogPartnerlink	แทรกรหัสต้นทางในส่วนของการประกาศพาร์ทเนอร์ลิงก์
- instrumentSetTestID	แทรกรหัสต้นทางสำหรับรหัสติดตามการทดสอบ
- instrumentLogInvokeBefore AfterCallService	แทรกรหัสต้นทางในโหนดที่เรียกใช้งานพาร์ทเนอร์ลิงก์
- instrumentLogPredicate BeforeAfterCallService	แทรกรหัสต้นทางในโหนดเงื่อนไข
คุณสมบัติ:	
ความสัมพันธ์:	
Composition:	
frm_readInstrument	
Entities_instrumentInvocation	
Entities_varLogInvocation	
ctrl_InstrumentComposite	
ctrl_InstrumentWSDL	
ctrl_InstrumentBPEL	

ตารางที่ 4-8 CRC ของคลาส facade_testcase

ชื่อคลาส: facade_testcase	ชนิด: พะชาด
คำอธิบาย: เป็นคลาสพะชาดมีหน้าที่เรียกใช้งานคลาสควบคุมต่างๆ เพื่อสร้างกรณีทดสอบ	
ความรับผิดชอบ: + genTestCase	สร้างกรณีทดสอบจากโหนดที่ระบุ
คุณสมบัติ:	
ความสัมพันธ์: Composition:	
	frm_viewTestDetail
	Entities_DrivePathCondition
	Entities_ListOfTC
	ctrl_BPEL_Predicate
	Entities_ExpressionNode
	Entities_genTestCase
	ctrl_TestCase
	ctrl_BPEL_ORAC

CHULALONGKORN UNIVERSITY

ตารางที่ 4-9 CRC ของคลาส facade_xsd

ชื่อคลาส: facade_xsd	ชนิด: พะชาด
คำอธิบาย: เป็นคลาสพะชาดมีหน้าที่เรียกใช้งานคลาสควบคุมต่างๆ เพื่ออ่านไฟล์เอกซ์เอสดี และสร้างกรณีทดสอบ ตั้งต้น จากนั้นจะนำไปจัดเก็บลงสู่ฐานข้อมูล	
ความรับผิดชอบ: + generateInputFromXSD	เรียกใช้งานฟังก์ชันอ่านโครงสร้างเอกซ์เอสดีและสร้างกรณีทดสอบตั้งต้น

ตารางที่ 4-9 CRC ของคลาส facade_xsd (ต่อ)

ชื่อคลาส: facade_xsd	ชนิด: พะชาด
คุณสมบัติ:	
ความสัมพันธ์:	
Composition:	
<ul style="list-style-type: none"> frm_readInstrument ctrl_XSD ctrl_ReceiveNodeVarSchema ctrl_TestCase Entities_XSDNode 	

ตารางที่ 4-10 CRC ของคลาส ctrl_XSD

ชื่อคลาส: ctrl_XSD	ชนิด: คลาสควบคุม
คำอธิบาย:	
เป็นคลาสที่ทำหน้าที่จัดการไฟล์ และข้อมูลเอกซ์เอสดี	
ความรับผิดชอบ:	
+ readXSD	อ่านข้อมูลจากไฟล์เอกซ์เอสดีและแปลงข้อมูลให้อยู่ในรูปแบบทิตีคลาส
+ readXSDDetail	ทำหน้าที่อ่านข้อมูลแท็กเอกซ์เอ็มแอลและแปลงให้อยู่ในรูปแบบทิตีคลาส
คุณสมบัติ:	
ความสัมพันธ์:	
Composition:	
<ul style="list-style-type: none"> Entities_xsdNode Entities_node facade_xsd 	

ตารางที่ 4-11 CRC ของคลาส ctrl_ReceiveNodeVarSchema

ชื่อคลาส: ctrl_ReceiveNodeVarSchema	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสที่ทำหน้าที่จัดการไฟล์ และข้อมูลเอกซ์เอสดี	
ความรับผิดชอบ:	
+ addSchema	นำโครงสร้างข้อมูลของข้อมูลนำเข้าที่ได้จากเอกซ์เอสดีไฟล์ ไปจัดเก็บเข้าสู่ฐานข้อมูล
+ getTargetNamespace	อ่านข้อมูลเนมสเปซ ของดับเบิลยูเอส-บีเฟล
+ getVariableDataType	อ่านประเภทของตัวแปรที่ระบุ
+ generateInputTemplate	สร้างกรณิทดสอบตั้งต้น
คุณสมบัติ:	
ความสัมพันธ์:	
Composition:	
Entities_xsdNode	
Entities_node	
facade_xsd	

จุฬาลงกรณ์มหาวิทยาลัย

CHULALONGKORN UNIVERSITY

ตารางที่ 4-12 CRC ของคลาส ctrl_InstrumentComposite

ชื่อคลาส: ctrl_InstrumentComposite	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสที่ใช้สำหรับแทรกรหัสต้นทางในไฟล์คอมโพสิต	
ความรับผิดชอบ:	
+ instrumentImport	แทรกรหัสต้นทางในส่วนของการนำเข้าเว็บเซอร์วิสสำหรับเก็บผลการทดสอบ
+ instrumentComponentReference	แทรกรหัสต้นทางในส่วนของการอ้างอิงเว็บเซอร์วิสสำหรับเก็บผลการทดสอบ ที่อยู่ภายใต้โหนดคอมโพเนน

ตารางที่ 4-12 CRC ของคลาส ctrl_InstrumentComposite (ต่อ)

ชื่อคลาส: ctrl_InstrumentComposite	ชนิด: คลาสควบคุม
+ instrumentReference	แทรกรหัสต้นทางในส่วนของการอ้างอิงเว็บเซอร์วิสสำหรับเก็บผลการทดสอบ
+ instrumentWire	แทรกรหัสต้นทางในส่วนของการอ้างอิงเว็บเซอร์วิสสำหรับเก็บผลการทดสอบ
+ getNameSpace	อ่านค่าเนมสเปซของโหนดที่ระบุ
คุณสมบัติ:	
ความสัมพันธ์:	
Composition: facade_instrument	

ตารางที่ 4-13 CRC ของคลาส ctrl_InstrumentWSDL

ชื่อคลาส: ctrl_InstrumentWSDL	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสที่ใช้สำหรับคัดลอกไฟล์ดับเบิลยูเอสดีแอลของเว็บเซอร์วิสที่เก็บข้อมูลการทดสอบเข้าสู่แฟ้มดับเบิลยูเอสดีแอลของโครงการ	
ความรับผิดชอบ:	
+ copyWSDLFile	คัดลอกไฟล์ดับเบิลยูเอสดีแอลของเว็บเซอร์วิสที่เก็บข้อมูลการทดสอบเข้าสู่แฟ้มดับเบิลยูเอสดีแอลของโครงการ
คุณสมบัติ:	
ความสัมพันธ์:	
Composition: facade_instrument	

ตารางที่ 4-14 CRC ของคลาส ctrl_InstrumentBPEL

ชื่อคลาส: ctrl_InstrumentBPEL	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสที่ใช้สำหรับแทรกกรหัสต้นทางในไฟล์ดับเบิลยูเอส-บีเพล	
ความรับผิดชอบ:	
+ instrumentBPELLogVariable	แทรกกรหัสต้นทางในส่วนของการประกาศตัวแปร สำหรับเก็บประวัติ
+ instrumentBPELLogPartnerLink	แทรกกรหัสต้นทางในส่วนของการประกาศพาร์ทเนอร์ลิงก์
+ instrument_BPELInvocation	แทรกกรหัสต้นทางในส่วนของการเรียกใช้งานพาร์ทเนอร์ลิงก์
+ instrument_BPELReceive	แทรกกรหัสต้นทางในส่วนของโหนดรับข้อมูล
+ instrument_BPELReply	แทรกกรหัสต้นทางในส่วนของโหนดตอบกลับ
+ instrument_BPELPredicate	แทรกกรหัสต้นทางในส่วนของโหนดเงื่อนไข
+ instrument_BPELPredicateWhile	แทรกกรหัสต้นทางในส่วนของโหนดลูป
- PredicateNodeInstrument_If	แทรกกรหัสต้นทางในส่วนของโหนดเงื่อนไข (if)
- PredicateNodeInstrument_While	แทรกกรหัสต้นทางในส่วนของโหนดเงื่อนไขที่เป็นลูป (while)
- predicate_1_beforeassign	แทรกกรหัสต้นทางสำหรับกำหนดค่าตัวแปรที่จะส่งไปยังเว็บเซอร์วิสก่อนเรียกใช้งาน
- predicate_2_beforeinvoke	แทรกกรหัสต้นทางสำหรับเรียกเว็บเซอร์วิสสำหรับจัดเก็บข้อมูลนำเข้า ก่อนเรียกใช้งาน
- predicate_3_afterassign	แทรกกรหัสต้นทางสำหรับกำหนดค่าตัวแปรที่ได้รับจากเว็บเซอร์วิสหลังจากเรียกใช้งาน
- predicate_4_afterinvoke	แทรกกรหัสต้นทางสำหรับเรียกเว็บเซอร์วิสสำหรับจัดเก็บข้อมูลส่งออกหลังจากเรียกใช้งาน

ตารางที่ 4-14 CRC ของคลาส ctrl_InstrumentBPEL (ต่อ)

ชื่อคลาส: ctrl_InstrumentBPEL	ชนิด: คลาสควบคุม
ความรับผิดชอบ: + instrumentSetTestID แทรกรหัสต้นทางสำหรับระบุค่ารหัสการทดสอบ + instrumentNameSpaceAttrProcessTag แทรกรหัสต้นทางสำหรับระบุค่าเนมสเปซของเว็บเซอร์วิสที่เก็บข้อมูลผลการทดสอบ + getTempUriNameSpace อ่านค่าเนมสเปซของ http://tempuri.org/ + getNameSpace อ่านค่าเนมสเปซของโหนดที่ระบุ	
คุณสมบัติ:	
ความสัมพันธ์: Composition: facade_instrument	

ตารางที่ 4-15 CRC ของคลาส ctrl_TestCase

ชื่อคลาส: ctrl_TestCase	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสที่ทำหน้าที่จัดการไฟล์ และข้อมูลเอกซ์เอสดี	
ความรับผิดชอบ: + getLastestTC อ่านข้อมูลกรณีทดสอบล่าสุดที่บันทึกไว้ + insertInitTC นำข้อมูลกรณีทดสอบตั้งต้นที่สร้างขึ้นไปจัดเก็บในฐานข้อมูล + getInitTC อ่านข้อมูลกรณีทดสอบตั้งต้นที่เก็บไว้ในฐานข้อมูล + genTCToFile นำกรณีทดสอบที่จัดเก็บไว้มาเขียนเป็นไฟล์ + getInputXSDDType อ่านประเภทของตัวแปรที่เก็บในฐานข้อมูล + totalGenTC นับจำนวนครั้งที่สร้างกรณีทดสอบจากฐานข้อมูล + insertNewTC นำกรณีทดสอบใหม่ที่สร้างขึ้นไปจัดเก็บในฐานข้อมูล	
คุณสมบัติ:	

ตารางที่ 4-15 CRC ของคลาส ctrl_TestCase (ต่อ)

ชื่อคลาส: ctrl_TestCase	ชนิด: คลาสควบคุม
ความสัมพันธ์: Composition: services_TestCase facade_xsd	

ตารางที่ 4-16 CRC ของคลาส ctrl_BPEL_Expression

ชื่อคลาส: ctrl_BPEL_Expression	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสควบคุมทำหน้าที่ตรวจสอบข้อความของโหนดเงื่อนไขที่ได้รับและจำแนกประเภทของข้อความนั้น	
ความรับผิดชอบ: + checkType ตรวจสอบประเภทของข้อความที่อ่านได้จากโหนดเงื่อนไข + checkTypeCMD ตรวจสอบประเภทของคำสั่งที่อ่านได้จากข้อความภายในโหนดเงื่อนไข - isNum ตรวจสอบว่าข้อความที่รับเข้ามาเป็นตัวเลขหรือไม่	
คุณสมบัติ:	
ความสัมพันธ์: Composition: Entities_ExpressionType ctrl_BPEL_Predicate	

ตารางที่ 4-17 CRC ของคลาส ctrl_BPEL_ORAC

ชื่อคลาส: ctrl_BPEL_ORAC	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสควบคุมทำหน้าที่ตรวจสอบคำสั่งที่รับเข้ามาเพื่อนำไปสร้างกรณีทดสอบ	
ความรับผิดชอบ:	
+ isOracle	ตรวจสอบว่าคำสั่งที่รับเข้ามาเป็นคำสั่งของออราเคิลหรือไม่
+ doActionAs	ตรวจสอบคำสั่งที่รับเข้ามาเพื่อเรียกใช้งานการสร้างกรณีทดสอบของแต่ละคำสั่งต่อไป
คุณสมบัติ:	
ความสัมพันธ์:	
Composition:	
façade_testCase	
Entities_ExpressionORAC	
ctrl_BPEL_Predicate	

ตารางที่ 4-18 CRC ของคลาส ctrl_BPEL_Predicate

ชื่อคลาส: ctrl_BPEL_Predicate	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสควบคุมทำหน้าที่แปลงเงื่อนไขที่อยู่ในรูปแบบข้อความให้อยู่ในรูปแบบเอนทิตีคลาส	
ความรับผิดชอบ:	
+ lsCondition	ตรวจสอบว่าเงื่อนไขที่รับเข้ามามีเงื่อนไขย่อยหรือไม่
+ convertConditionToExp	แปลงข้อความจากหนดเงื่อนไขให้อยู่ในรูปแบบเอนทิตีคลาสเพื่อนำไปประมวลผลต่อ
- submitRecentCommand	ตรวจสอบข้อความที่รับเข้ามาเพื่อนำไปจัดเก็บในเอนทิตีคลาส
คุณสมบัติ:	

ตารางที่ 4-18 CRC ของคลาส ctrl_BPEL_Predicate (ต่อ)

ชื่อคลาส: ctrl_BPEL_Predicate	ชนิด: คลาสควบคุม
ความสัมพันธ์: Composition: Entities_ExpressionType Entities_ExpressionNode ctrl_BPEL_Expression	



ตารางที่ 4-19 CRC ของคลาส ctrl_TestResult

ชื่อคลาส: ctrl_TestResult	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสควบคุมทำหน้าที่เรียกใช้งานคลาสเซอร์วิซต่อ เพื่ออ่านข้อมูลดับเบิลยูเอส-บีเฟล และข้อมูลผลการทดสอบ	
ความรับผิดชอบ: + getJobBPEL อ่านข้อมูลโครงการดับเบิลยูเอส-บีเฟล ที่จัดเก็บในระบบ + getTestedNodeByJobID อ่านข้อมูลไหนคของดับเบิลยูเอส-บีเฟลที่ถูกทดสอบ + getTestMaster อ่านข้อมูลเวลาที่เริ่มทดสอบ + getUnExecuteInvocation อ่านข้อมูลเว็บเซอร์วิซที่ไม่ถูกทดสอบ + getReceiveInputData อ่านข้อมูลกรณีทดสอบที่นำมาใช้ + getTestHistoryByTaskJobID อ่านข้อมูลประวัติการทดสอบ	
คุณสมบัติ:	
ความสัมพันธ์: Composition: frm_BPELList services_TestResult	

ตารางที่ 4-20 CRC ของคลาส Ctrl_BPEL

ชื่อคลาส: Ctrl_BPEL	ชนิด: คลาสควบคุม
คำอธิบาย: คลาสทำหน้าที่รับเส้นทางของไฟล์ดับเบิลยูเอส-บีเพล เพื่อแปลงไฟล์ดับเบิลยูเอส-บีเพล ให้อยู่ในรูปแบบของข้อมูลเส้นทางการไหล	
ความรับผิดชอบ: + loadBPEL	อ่านไฟล์ดับเบิลยูเอส-บีเพลเพื่อแปลงเป็นเส้นทางการไหล จากนั้นจะเรียกฟังก์ชันเพื่อบันทึกเข้าสู่ฐานข้อมูล
คุณสมบัติ:	
ความสัมพันธ์: Composition: <ul style="list-style-type: none"> frm_readInstrument services_Serial Entities_node ctrl_BPEL_Node ctrl_BPELDB 	

ตารางที่ 4-21 CRC ของคลาส ctrl_BPEL_Node

ชื่อคลาส: ctrl_BPEL_Node	ชนิด: คลาสควบคุม
คำอธิบาย: คลาสทำหน้าที่อ่านข้อมูลโหนดเอกซ์เอ็มแอลของดับเบิลยูเอส-บีเพล และแปลงข้อมูลให้อยู่ในรูปแบบของเอนทิตีคลาส	
ความรับผิดชอบ: + getBPELNode + getNodeInstrumentType	อ่านโหนดเอกซ์เอ็มแอลของไฟล์ดับเบิลยูเอส-บีเพล และแปลงข้อมูลให้อยู่ในรูปแบบของเอนทิตีคลาส ตรวจสอบประเภทของโหนด โดยฟังก์ชันจะคืนค่ากลุ่มของโหนด

ตารางที่ 4-21 CRC ของคลาส ctrl_BPEL_Node (ต่อ)

ชื่อคลาส: ctrl_BPEL_Node	ชนิด: คลาสควบคุม
ความรับผิดชอบ:	
+ processInvocationNode	อ่านโนหนดเอกซ์เอ็มแอลของไฟล์ดับเบิลยูเอส-บีเพล มาแปลงให้อยู่ในรูปแบบของเอนทิตีคลาสที่เป็นโนหนดการเรียกใช้งานพาร์ทเนอร์ลิงก์
+ getInvocationNode	ทำหน้าที่แปลงโนหนดเอกซ์เอ็มแอลจากฟังก์ชัน processInvocationNode ให้อยู่ในรูปแบบของเอนทิตีคลาส
+ processPredicateNode	อ่านโนหนดเอกซ์เอ็มแอลของไฟล์ดับเบิลยูเอส-บีเพล มาแปลงให้อยู่ในรูปแบบของเอนทิตีคลาสที่เป็นโนหนดเงื่อนไข
+ getCondition	อ่านข้อมูลเงื่อนไขที่อยู่ในโนหนดเอกซ์เอ็มแอล
+ getPredicateNode	อ่านโนหนดเอกซ์เอ็มแอลที่ได้รับมาจากฟังก์ชัน processPredicateNode ให้อยู่ในรูปแบบของเอนทิตีคลาส
+ processOtherNode	อ่านโนหนดเอกซ์เอ็มแอลของไฟล์ดับเบิลยูเอส-บีเพล มาแปลงให้อยู่ในรูปแบบของเอนทิตีคลาสที่เป็นโนหนดอื่นๆ
+ processOtherPredicate	อ่านโนหนดเอกซ์เอ็มแอลของไฟล์ดับเบิลยูเอส-บีเพล มาแปลงให้อยู่ในรูปแบบของเอนทิตีคลาสที่เป็นโนหนดเงื่อนไข (elseif, else)
+ processAssignNode	อ่านโนหนดเอกซ์เอ็มแอลของไฟล์ดับเบิลยูเอส-บีเพล มาแปลงให้อยู่ในรูปแบบของเอนทิตีคลาสที่เป็นโนหนดการตั้งค่า
+ getAssignNode	อ่านโนหนดเอกซ์เอ็มแอลที่ได้รับมาจากฟังก์ชัน processAssignNode ให้อยู่ในรูปแบบของเอนทิตีคลาส
+ addToLastDownline	นำเอนทิตีโนหนดไปใส่ในเอนทิตีของเส้นทางการไหล
+ getReceiveVariableName	เรียกข้อมูลชื่อตัวแปรนำเข้าที่อยู่ภายใต้โนหนด 'Receive'
+ getListPathCondition	เรียกรายการโนหนดเงื่อนไข โดยนำรหัสของโนหนดที่ระบุมาค้นหาเงื่อนไข ที่อยู่ลำดับก่อนหน้า
+ getConditionList	อ่านข้อมูลเงื่อนไขจากรหัสโนหนดที่ระบุ

ตารางที่ 4-21 CRC ของคลาส ctrl_BPEL_Node (ต่อ)

ชื่อคลาส: ctrl_BPEL_Node	ชนิด: คลาสควบคุม
คุณสมบัติ:	
ความสัมพันธ์:	
Composition:	
Ctrl_BPEL	
facade_testcase	
Entities_node	
factory_BPELNodeDetail	
Entities_nodeDetailOther	
Entities_nodeDetailPredicate	
Entities_nodeDetailInvocation	
Entities_nodeSubConditionProcess	

ตารางที่ 4-22 CRC ของคลาส ctrl_BPEL_DB

ชื่อคลาส: ctrl_BPEL_DB	ชนิด: คลาสควบคุม
คำอธิบาย:	
เป็นคลาสที่ทำหน้าที่ติดต่อกับฐานข้อมูลที่จัดเก็บโหนดของดับเบิลยูเอส-บีเพล	
ความรับผิดชอบ:	
+ add_BPEL	ทำหน้าที่เรียกใช้งานคลาสเซอร์วิส เพื่อนำข้อมูลโหนดดับเบิลยูเอส-บีเพล ไปจัดเก็บสู่ฐานข้อมูล
+ getProjectName	ทำหน้าที่อ่านข้อมูลชื่อโครงการดับเบิลยูเอส-บีเพลจากฐานข้อมูล
คุณสมบัติ:	
ความสัมพันธ์:	
Composition:	
services_BPEL_DB	
Ctrl_BPEL	

ตารางที่ 4-23 CRC ของคลาส ctrls_DB_node

ชื่อคลาส: ctrls_DB_node	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสควบคุมทำหน้าที่อ่านข้อมูลของโหนดที่ระบุ	
ความรับผิดชอบ: + getNodeType	อ่านข้อมูลประเภทของโหนดที่ระบุ
คุณสมบัติ:	
ความสัมพันธ์: Composition: façade_testcase frm_viewTestDetail services_Node	

ตารางที่ 4-24 CRC ของคลาส services_ReceiveNodeVarSchema

ชื่อคลาส: services_ReceiveNodeVarSchema	ชนิด: เซอร์วิส
คำอธิบาย: เป็นคลาสที่ทำหน้าเข้าถึงฐานข้อมูลเพื่อจัดการข้อมูลโหนดและเส้นทางการไหลของดับเบิลยูเอส-บีเพล	
ความรับผิดชอบ: + addSchema - addSubSchemaDetail + insertNewTC + getVariableDataType + findXSDInputName + getXSDRestriction	นำโครงสร้างข้อมูลจากไฟล์เอกซ์เอสดีมาจัดเก็บเข้าสู่ฐานข้อมูล นำโครงสร้างข้อมูลที่ถูกส่งมาจากฟังก์ชัน addSchema มาจัดเก็บเข้าสู่ฐานข้อมูล นำกรณีทดสอบใหม่ที่สร้างขึ้นมาจัดเก็บเข้าฐานข้อมูล อ่านประเภทข้อมูลของตัวแปรที่รับเข้ามา ตรวจสอบว่าชื่อตัวแปรที่รับเข้ามา เป็นข้อมูลนำเข้าและมีการจัดเก็บโครงสร้างของตัวแปรนี้หรือไม่ อ่านข้อจำกัดของประเภทตัวแปรที่ระบุ
คุณสมบัติ:	

ตารางที่ 4-24 CRC ของคลาส services_ReceiveNodeVarSchema (ต่อ)

ชื่อคลาส: services_ReceiveNodeVarSchema	ชนิด: เซอร์วิซ
ความสัมพันธ์: Composition: Entities_xsdNode ctrl_ReceiveNodeVarSchema	

ตารางที่ 4-25 CRC ของคลาส services_TestCase

ชื่อคลาส: services_TestCase	ชนิด: เซอร์วิซ
คำอธิบาย: เป็นคลาสที่ทำหน้าเข้าถึงฐานข้อมูลเพื่อจัดการข้อมูลกรณีทดสอบ	
ความรับผิดชอบ: + getLastestTC อ่านข้อมูลกรณีทดสอบล่าสุด + getInitTC อ่านข้อมูลกรณีทดสอบตั้งต้น + getInputXSD อ่านโครงสร้างของตัวแปรที่จะสร้างกรณีทดสอบ + getVariableDataType อ่านประเภทข้อมูลของตัวแปรที่รับเข้ามา + getInputXSDType อ่านประเภทของตัวแปรที่จะสร้างกรณีทดสอบ + getInputXSDType_simple อ่านประเภทของตัวแปรที่จะสร้างกรณีทดสอบ สำหรับตัวแปรประเภทพื้นฐาน + insertNewTC บันทึกข้อมูลกรณีทดสอบใหม่เข้าสู่ฐานข้อมูล + totalGenTC นับจำนวนกรณีทดสอบที่ถูกสร้างขึ้น + insertInitTC บันทึกข้อมูลกรณีทดสอบตั้งต้นเข้าสู่ฐานข้อมูล	
คุณสมบัติ:	
ความสัมพันธ์: Composition: ctrl_TestCase	

ตารางที่ 4-26 CRC ของคลาส services_TestResult

ชื่อคลาส: services_TestResult	ชนิด: เซอร์วิซ																		
คำอธิบาย: เป็นคลาสเซอร์วิซทำหน้าที่เข้าถึงข้อมูล เพื่ออ่านข้อมูลดับเบิลยูเอส-บีเพล และข้อมูลผลการทดสอบ																			
ความรับผิดชอบ: <table border="0"> <tr> <td>+ getJobBPEL</td> <td>อ่านข้อมูลโครงการดับเบิลยูเอส-บีเพล ที่จัดเก็บในระบบ</td> </tr> <tr> <td>+ getTestedNodeByJobID</td> <td>อ่านข้อมูลโหนดของดับเบิลยูเอส-บีเพลที่ถูกทดสอบ</td> </tr> <tr> <td>+ getTestMaster</td> <td>อ่านข้อมูลเวลาที่เริ่มทดสอบ</td> </tr> <tr> <td>+ getUnExecuteInvocation</td> <td>อ่านข้อมูลเว็บเซอร์วิซที่ไม่ถูกทดสอบ</td> </tr> <tr> <td>+ getReceiveInputData</td> <td>อ่านข้อมูลกรณีทดสอบที่นำมาใช้</td> </tr> <tr> <td>+ getTestHistoryByTaskJobID</td> <td>อ่านข้อมูลประวัติการทดสอบ</td> </tr> <tr> <td>+ getExecuteInvocation</td> <td>อ่านข้อมูลโหนดพาร์ทเนอร์ลิงก์ที่ถูกทดสอบ</td> </tr> <tr> <td>+ getExecuteList</td> <td>อ่านข้อมูลโหนดที่ถูกทดสอบ</td> </tr> <tr> <td>+ getUnExecuteInvocationCondition</td> <td>ค้นหาข้อมูลโหนดที่ระบุพร้อมทั้งอ่านข้อมูลเงื่อนไขสุดท้ายที่จะมาถึงโหนดนั้น</td> </tr> </table>		+ getJobBPEL	อ่านข้อมูลโครงการดับเบิลยูเอส-บีเพล ที่จัดเก็บในระบบ	+ getTestedNodeByJobID	อ่านข้อมูลโหนดของดับเบิลยูเอส-บีเพลที่ถูกทดสอบ	+ getTestMaster	อ่านข้อมูลเวลาที่เริ่มทดสอบ	+ getUnExecuteInvocation	อ่านข้อมูลเว็บเซอร์วิซที่ไม่ถูกทดสอบ	+ getReceiveInputData	อ่านข้อมูลกรณีทดสอบที่นำมาใช้	+ getTestHistoryByTaskJobID	อ่านข้อมูลประวัติการทดสอบ	+ getExecuteInvocation	อ่านข้อมูลโหนดพาร์ทเนอร์ลิงก์ที่ถูกทดสอบ	+ getExecuteList	อ่านข้อมูลโหนดที่ถูกทดสอบ	+ getUnExecuteInvocationCondition	ค้นหาข้อมูลโหนดที่ระบุพร้อมทั้งอ่านข้อมูลเงื่อนไขสุดท้ายที่จะมาถึงโหนดนั้น
+ getJobBPEL	อ่านข้อมูลโครงการดับเบิลยูเอส-บีเพล ที่จัดเก็บในระบบ																		
+ getTestedNodeByJobID	อ่านข้อมูลโหนดของดับเบิลยูเอส-บีเพลที่ถูกทดสอบ																		
+ getTestMaster	อ่านข้อมูลเวลาที่เริ่มทดสอบ																		
+ getUnExecuteInvocation	อ่านข้อมูลเว็บเซอร์วิซที่ไม่ถูกทดสอบ																		
+ getReceiveInputData	อ่านข้อมูลกรณีทดสอบที่นำมาใช้																		
+ getTestHistoryByTaskJobID	อ่านข้อมูลประวัติการทดสอบ																		
+ getExecuteInvocation	อ่านข้อมูลโหนดพาร์ทเนอร์ลิงก์ที่ถูกทดสอบ																		
+ getExecuteList	อ่านข้อมูลโหนดที่ถูกทดสอบ																		
+ getUnExecuteInvocationCondition	ค้นหาข้อมูลโหนดที่ระบุพร้อมทั้งอ่านข้อมูลเงื่อนไขสุดท้ายที่จะมาถึงโหนดนั้น																		
คุณสมบัติ:																			
ความสัมพันธ์: Composition: <div style="text-align: center;">ctrl_TestResult</div>																			

ตารางที่ 4-27 CRC ของคลาส services_BPEL_DB

ชื่อคลาส: services_BPEL_DB	ชนิด: เซอร์วิซ
คำอธิบาย: เป็นคลาสที่ทำหน้าที่เข้าถึงฐานข้อมูลเพื่อจัดการข้อมูลโหนดและเส้นทางการไหลของดับเบิลยูเอส-บีเพล	

ตารางที่ 4-27 CRC ของคลาส services_BPEL_DB (ต่อ)

ชื่อคลาส: services_BPEL_DB	ชนิด: เซอร์วิส
<p>ความรับผิดชอบ:</p> <p>+ insertMasterSubProcess ทำหน้าที่เอาข้อมูลลงตาราง TBL_NodeMaster สำหรับโหนดเงื่อนไข</p> <p>+ add_BPEL ทำหน้าที่นำข้อมูลโหนดที่อยู่ในรูปแบบเอนทิตี จัดเก็บเข้าสู่ฐานข้อมูล</p> <p>- generateScriptForOther สร้างคำสั่งเอสคิวเอล เพื่อนำข้อมูลของโหนดชนิดอื่นๆ ไปจัดเก็บสู่ฐานข้อมูล</p> <p>- generateScriptForInvocation สร้างคำสั่งเอสคิวเอล เพื่อนำข้อมูลของโหนดที่เรียกใช้งานพาร์ทเนอร์ลิงก์ ไปจัดเก็บสู่ฐานข้อมูล</p> <p>- generateScriptForPredicate สร้างคำสั่งเอสคิวเอล เพื่อนำข้อมูลของโหนดที่เป็นโหนดเงื่อนไข ไปจัดเก็บสู่ฐานข้อมูล</p> <p>+ getProjectName ทำหน้าที่อ่านชื่อโครงการจากฐานข้อมูล</p>	
<p>คุณสมบัติ:</p> <p>- processOnTask เก็บรหัสของโครงการดับเบิลยูเอส-บีเพล ที่กำลังบันทึกเข้าสู่ฐานข้อมูล</p>	
<p>ความสัมพันธ์:</p> <p>Composition:</p> <p> ctrl_BPELDB</p> <p> Entities_node</p> <p> Entities_nodeDetailOther</p> <p> Entities_nodeDetailPredicate</p> <p> Entities_nodeDetailInvocation</p> <p> services_Serial</p>	

ตารางที่ 4-28 CRC ของคลาส services_Node

ชื่อคลาส: services_Node	ชนิด: คลาสเซอร์วิซ
คำอธิบาย: เป็นคลาสเซอร์วิซทำหน้าที่ติดต่อกับฐานข้อมูลเพื่ออ่านข้อมูลของโหนดดับเบิลยูเอส-บีเพล	
ความรับผิดชอบ:	
+ getNodeType	อ่านข้อมูลประเภทของโหนดที่ระบุจากฐานข้อมูล
+ getListPathCondition	อ่านข้อมูลของโหนดเงื่อนไขที่อยู่ในเส้นทางการไหลของโหนดที่ระบุ
+ getNodePathCondition	อ่านข้อมูลของโหนดเงื่อนไขที่อยู่ในเส้นทางการไหลของโหนดที่ระบุ ซึ่งจะถูกรู้จักใช้งานจากฟังก์ชัน getListPathCondition
+ getNodePathConditionFirstSequence	อ่านข้อมูลของโหนดเงื่อนไขที่อยู่ในเส้นทางการไหลโดยใช้โหนดที่ระบุและกิ่งเงื่อนไขที่ระบุ
+ getNodePrevPathCondition	ค้นหาโหนดเงื่อนไขโดยใช้รหัสโหนดที่ระบุ
+ getNodeDetailID	อ่านข้อมูลรายละเอียดของโหนดที่ระบุ
+ getReceiveVariableName	อ่านชื่อตัวแปรนำเข้าที่เก็บอยู่ในฐานข้อมูล
+ getReceiveNodeID	อ่านข้อมูลโหนดรับค่าข้อมูลนำเข้า
+ getConditionList	อ่านข้อมูลโหนดเงื่อนไขที่อยู่ภายใต้เงื่อนไขเดียวกัน
คุณสมบัติ:	
ความสัมพันธ์:	
Composition:	
ctrl_BPEL_Node	
Entities_DrivePathCondition	
Entities_nodeSubConditionProcess	
Entities_node	
Entities_nodeDetailPredicate	

ตารางที่ 4-29 CRC ของคลาส services_Serial

ชื่อคลาส: services_Serial	ชนิด: เซอร์วิซ
คำอธิบาย: เป็นคลาสที่ทำหน้าเข้าถึงฐานข้อมูลเพื่อจัดการรหัสของตารางต่างๆ	
ความรับผิดชอบ: + getSerial	หารหัสถัดไปของตารางที่ระบุ
คุณสมบัติ:	
ความสัมพันธ์: Composition: Ctrl_BPEL services_BPEL_DB	

ตารางที่ 4-30 CRC ของคลาส Ctrls_config

ชื่อคลาส: Ctrls_config	ชนิด: คลาสควบคุม
คำอธิบาย: เป็นคลาสควบคุมทำหน้าที่อ่านข้อมูลการตั้งค่าต่างๆ ที่เกี่ยวข้องกับการทำงานของเครื่องมือ	
ความรับผิดชอบ: + loadConfig + getConfig + readCfgFiles + readConfig	อ่านข้อมูลการตั้งค่า อ่านข้อมูลการตั้งค่าจากชื่อตัวแปรที่ระบุ อ่านไฟล์การตั้งค่า อ่านข้อมูลการตั้งค่า โดยจะเรียกใช้งานฟังก์ชัน readCfgFiles เพื่ออ่านไฟล์การตั้งค่า
คุณสมบัติ: - fileStream (ไฟล์สตรีม)	เก็บค่าสตรีมของไฟล์ที่อ่าน

ตารางที่ 4-30 CRC ของคลาส Ctrls_config (ต่อ)

ชื่อคลาส: Ctrls_config	ชนิด: คลาสควบคุม
คุณสมบัติ:	
- StreamWriter (StreamWriter)	เขียนข้อความที่ระบุลงไปไฟล์
- str_cfgData (string)	เก็บค่าข้อความที่อ่านได้จากไฟล์ตั้งค่า
- cfgDict (Dictionary)	เก็บค่าที่อ่านได้จากไฟล์ตั้งค่า
ความสัมพันธ์:	
Composition:	
MDI MainForm	
Entities_sysConfig	
Ctrls_Files	

ตารางที่ 4-31 CRC ของคลาส Ctrls_Files

ชื่อคลาส: Ctrls_Files	ชนิด: คลาสควบคุม
คำอธิบาย:	
เป็นคลาสควบคุมทำหน้าที่อ่านไฟล์การตั้งค่าของเครื่องมือ	
ความรับผิดชอบ:	
+ openFile	เปิดไฟล์การตั้งค่า
+ writeLog	เขียนบันทึกการทำงานของโปรแกรม
คุณสมบัติ:	
- FileStream (ไฟล์สตรีม)	เก็บค่าสตรีมของไฟล์ที่อ่าน
- StreamWriter (StreamWriter)	เขียนข้อความที่ระบุลงไปไฟล์
ความสัมพันธ์:	
Composition:	
ctrl_instrumentBPEL	

ตารางที่ 4-31 CRC ของคลาส Ctrl_files (ต่อ)

ชื่อคลาส: Ctrl_files	ชนิด: คลาสควบคุม
ความสัมพันธ์: Composition: ctrls_config Services_BPEL_DB services_ReceiveNodeVarSchema services_TestCase services_TestResult Service_Node	

4.1.4. แผนภาพลำดับ

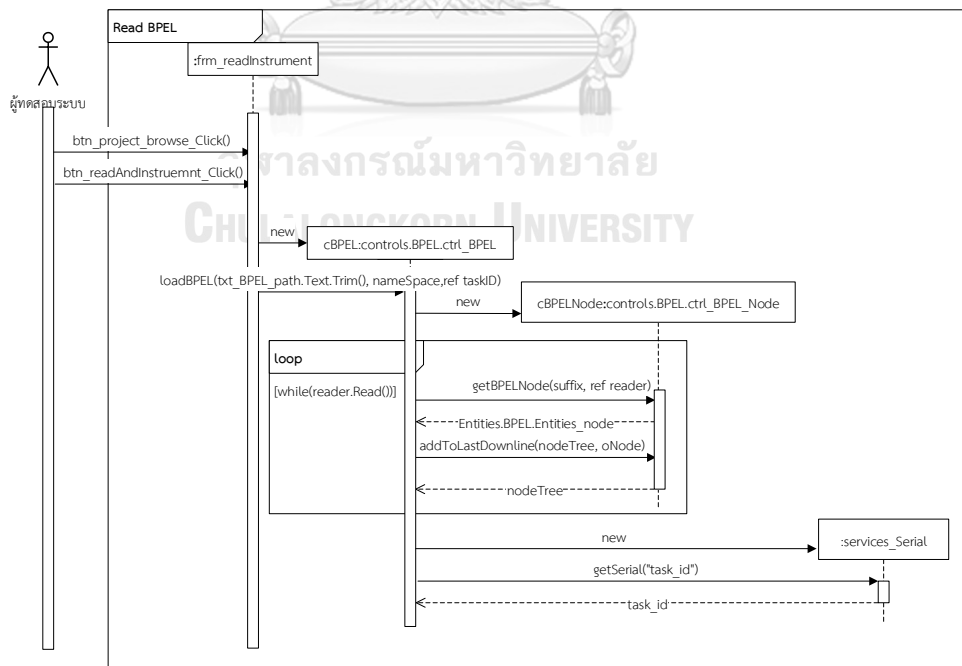
เป็นแผนภาพที่แสดงให้เห็นถึงลำดับและขั้นตอนการทำงานของเหตุการณ์ต่างๆ ที่เกิดขึ้นขณะใช้งานเครื่องมือ เช่นลำดับการเรียกใช้งานฟังก์ชัน ข้อความที่รับ-ส่งระหว่างคลาส เป็นต้น ผู้วิจัยได้ออกแบบแผนภาพลำดับโดยแบ่งตามกิจกรรมที่สำคัญไว้ดังนี้

1. การอ่านข้อมูลบีเพลเพื่อบันทึกเส้นทางการไหลและแทรกรหัสต้นทาง

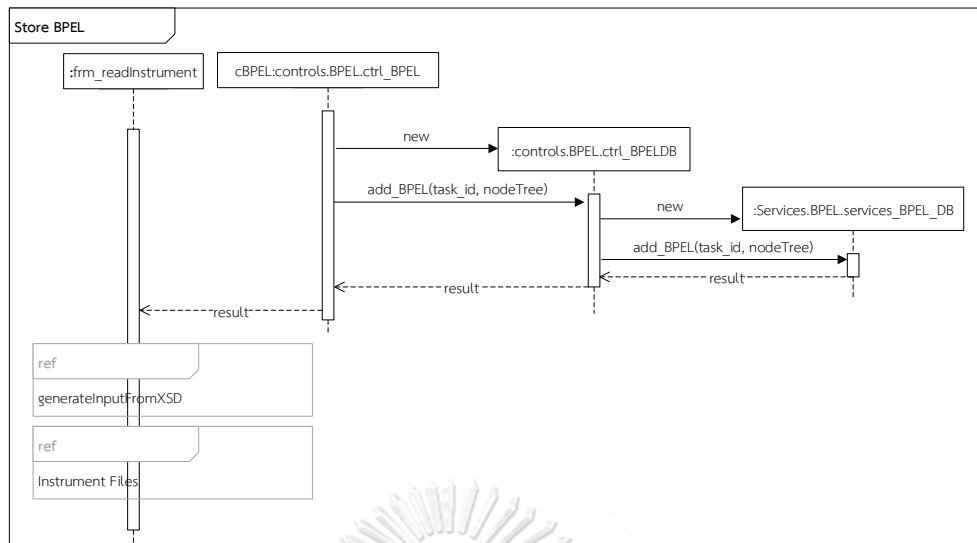
ในเหตุการณ์บันทึกเส้นทางการไหลและแทรกรหัสต้นทางของโครงการดับเบิลยูเอส-บีเพล ผู้วิจัยได้ออกแบบแผนภาพลำดับเป็น 3 ส่วน ได้แก่ ขั้นตอนการอ่านข้อมูลดับเบิลยูเอส-บีเพล เพื่อสร้างและบันทึกเส้นทางการไหล ขั้นตอนการจัดเก็บโครงสร้างข้อมูลจากไฟล์เอกซ์เอสดี และขั้นตอนการแทรกรหัสต้นทาง โดยในแต่ละขั้นตอนมีรายละเอียดการทำงานดังนี้

- ในการอ่านข้อมูลดับเบิลยูเอส-บีเพลเพื่อสร้างและบันทึกเส้นทางการไหล ผู้วิจัยได้ออกแบบแผนภาพลำดับดังรูปที่ 4-10 และ 4-11 โดยเริ่มจากผู้ทดสอบเลือกแฟ้มโครงการดับเบิลยูเอส-บีเพล จากหน้าจอโดยคลิกที่ปุ่ม “Browse” จากนั้น คลาส “frm_readInstrument” จะอ่านข้อมูลแฟ้มโครงการที่ผู้ทดสอบเลือก พร้อมทั้งค้นหาไฟล์ต่างๆ ได้แก่ ไฟล์คอมโพสิต ไฟล์ดับเบิลยูเอสดีแอล ไฟล์ดับเบิลยูเอส-บีเพล และไฟล์เอกซ์เอสดี จากนั้นผู้ทดสอบยืนยันการอ่านและแทรกรหัสต้นทางโดยคลิกที่ปุ่ม

“Read and Instrument” คลาส “frm_readInstrument” จะเรียกใช้งานฟังก์ชัน “loadBPEL” ที่อยู่ภายใต้คลาส “ctrl_BPEL” เพื่ออ่านข้อมูลดับเบิลยูเอส-บีเพลจากไฟล์ที่ผู้ใช้งานยืนยัน โดยในการอ่านข้อมูลดับเบิลยูเอส-บีเพล คลาส “ctrl_BPEL” จะเรียกใช้งานฟังก์ชัน “getBPELNode” ที่อยู่ภายใต้คลาส “controls.BPEL.ctrl_BPEL_Node” เพื่อนำโหนดดับเบิลยูเอส-บีเพลที่อ่านได้มาสร้างเข้าสู่เอนทิตีคลาส และเมื่อได้เอนทิตีคลาสของโหนดนั้นๆ แล้วคลาส “ctrl_BPEL” จะเรียกใช้งานฟังก์ชัน “addToLastDownline” เพื่อนำโหนดดับเบิลยูเอส-บีเพลที่ได้ไปจัดเก็บเป็นโครงสร้าง จากนั้นเมื่ออ่านไฟล์ดับเบิลยูเอส-บีเพลจนหมดแล้ว คลาส “ctrl_BPEL” จะร้องขอรหัสสำหรับติดตามโครงการดับเบิลยูเอส-บีเพลจากฟังก์ชัน “getSerial” ที่อยู่ภายใต้คลาส “services_Serial” และจะนำรหัสที่ได้กับโหนดดับเบิลยูเอส-บีเพลที่จัดเก็บอยู่ในรูปแบบเอนทิตีคลาส ไปจัดเก็บลงฐานข้อมูล โดยเรียกใช้ฟังก์ชัน add_BPEL ที่อยู่ภายใต้คลาส “controls.BPEL.ctrl_BPELDB” โดยที่คลาส “controls.BPEL.ctrl_BPELDB” จะเรียกใช้งานคลาส “Services.BPEL.services_BPEL_DB” เพื่อนำข้อมูลจัดเก็บเข้าสู่ฐานข้อมูล และคืนค่าผลการบันทึกข้อมูลให้กับคลาส “controls.BPEL.ctrl_BPELDB” เพื่อที่ส่งค่าต่อไปยังคลาส “ctrl_BPEL”

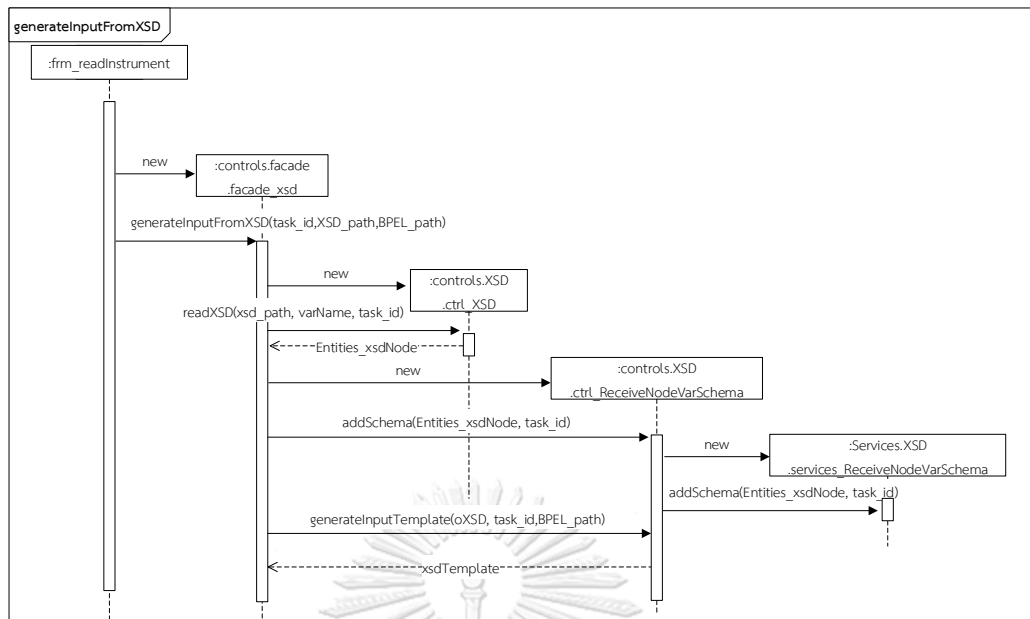


รูปที่ 4-10 แผนภาพลำดับ ของยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเพล (ในขั้นตอนการอ่านข้อมูลดับเบิลยูเอส-บีเพล)

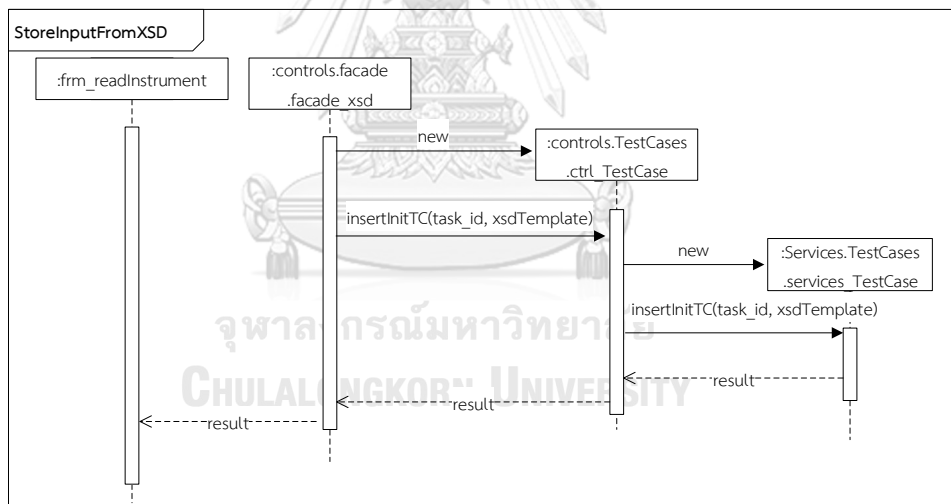


รูปที่ 4-11 แผนภาพลำดับ ของยูสเคสออฟโหลดและแทรกรหัสต้นทางโครงการดับเบิ้ลยูเอส-บีเพล (ในขั้นตอนการบันทึกเส้นทางกรโหลของดับเบิ้ลยูเอส-บีเพล)

- ในการจัดเก็บโครงสร้างข้อมูลจากไฟล์เอกซ์เอสดี ผู้วิจัยได้ออกแบบแผนภาพลำดับ ดังรูปที่ 4-12 และ 4-13 โดยหลังจากที่เครื่องมือได้จัดเก็บข้อมูลเส้นทางกรโหลของดับเบิ้ลยูเอส-บีเพลแล้ว เครื่องมือจะเรียกใช้งานฟังก์ชัน “generateInputFromXSD” ที่อยู่ภายใต้คลาส “facade.facade_xsd” โดยจะระบุหมายเลขติดตามโครงการดับเบิ้ลยูเอส-บีเพล ที่อยู่ของไฟล์เอกซ์เอสดี และที่อยู่ของไฟล์ดับเบิ้ลยูเอส-บีเพล โดยคลาส “facade.facade_xsd” จะเรียกใช้งานฟังก์ชัน “readXSD” ที่อยู่ภายใต้คลาส “XSD.ctrl_XSD” เพื่ออ่านไฟล์เอกซ์เอสดีและเก็บข้อมูลเอกซ์เอสดีนั้นเข้าสู่คลาส “Entities_xsdNode” จากนั้นคลาส “facade.facade_xsd” จะนำข้อมูลเอกซ์เอสดีที่ได้ และหมายเลขสำหรับติดตามโครงการดับเบิ้ลยูเอส-บีเพลไปจัดเก็บในฐานข้อมูลโดยผ่านฟังก์ชัน “addSchema” ที่อยู่ภายใต้คลาส “ctrl_XSD” จากนั้นคลาส “ctrl_XSD” จะนำข้อมูลไปจัดเก็บในฐานข้อมูลโดยเรียกใช้งานคลาส “services_ReceiveNodeVarSchema” เมื่อจัดเก็บข้อมูลเอกซ์เอสดีแล้ว คลาส “facade.facade_xsd” จะเรียกใช้งานฟังก์ชัน “generateInputTemplate” ที่อยู่ภายใต้คลาส “ctrl_ReceiveNodeVarSchema” เพื่อสร้างกรณีทดสอบตั้งต้น จากนั้นจะเรียกใช้ฟังก์ชัน “insertInitTC” ที่อยู่ภายใต้คลาส “ctrl_TestCase” เพื่อนำกรณีทดสอบตั้งต้นที่ได้ไปจัดเก็บในฐานข้อมูลโดยเรียกใช้งานฟังก์ชัน “insertInitTC” ที่อยู่ภายใต้คลาส “services_TestCase”



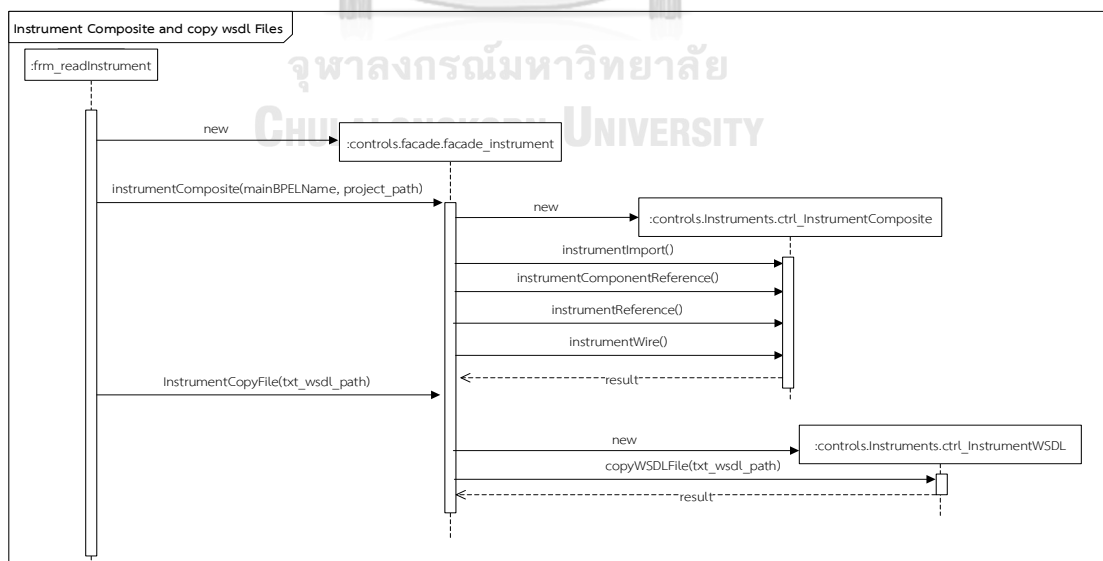
รูปที่ 4-12 แผนภาพลำดับ ของยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเพล (ในขั้นตอนการสร้างโครงสร้างข้อมูลจากไฟล์เอกซ์เอสดี)



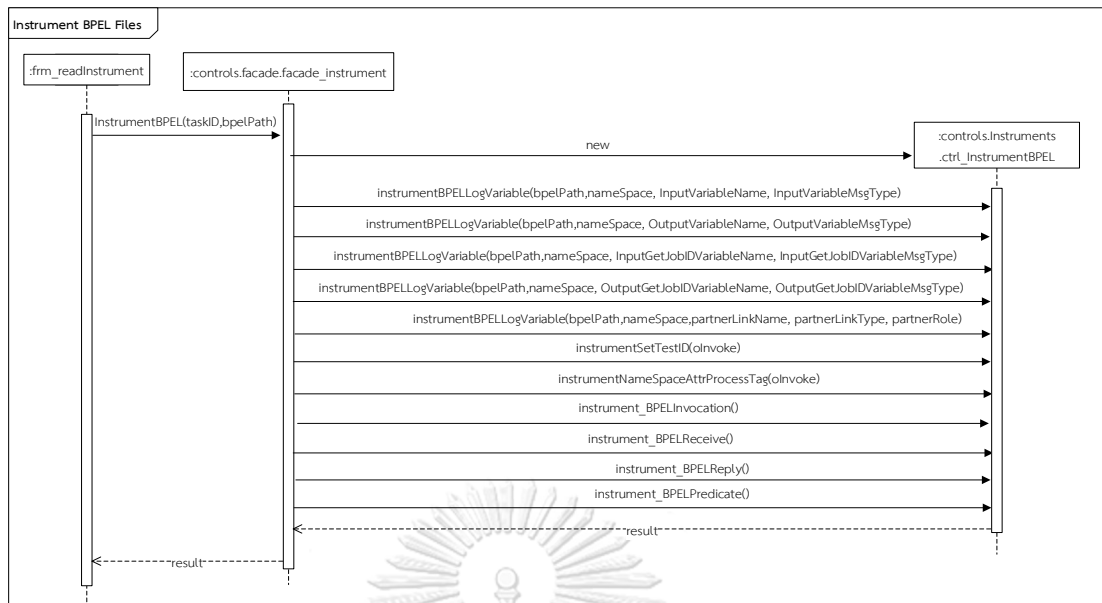
รูปที่ 4-13 แผนภาพลำดับ ของยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเพล (ในขั้นตอนการบันทึกโครงสร้างข้อมูลจากไฟล์เอกซ์เอสดี)

- ในขั้นตอนการแทรกรหัสต้นทาง จะแบ่งเป็น 3 ขั้นตอน ได้แก่ การแทรกรหัสต้นทางของไฟล์คอมโพสิต เพื่อให้โครงการดับเบิลยูเอส-บีเพล สามารถอ้างอิงไปยังเว็บเซอร์วิสที่ทำหน้าที่จัดเก็บข้อมูลการทดสอบได้ การนำไฟล์ดับเบิลยูเอสดีแอลของเว็บเซอร์วิสที่มีหน้าที่เก็บข้อมูลการทดสอบมาวางไว้ในแฟ้มดับเบิลยูเอสดีแอล ของโครงการเพื่อให้โครงการสามารถอ้างอิงมายังไฟล์ดับเบิลยูเอสดีแอลได้ และการแทรกรหัสต้นทางของไฟล์

ดับเบิลยูเอส-บีเพลเพื่อเรียกใช้งานเว็บเซอร์วิสสำหรับจัดเก็บข้อมูล เมื่อพบโหนดที่มีการเรียกใช้งานพาร์ทเนอร์ลิงก์ ผู้วิจัยได้ออกแบบแผนภาพลำดับตั้งรูปที่ 4-14 และ 4-15 ซึ่งหลังจากที่เครื่องมือจัดเก็บข้อมูลเอกซ์เอสดีและสร้างกรณีทดสอบตั้งต้นแล้ว คลาส “frm_readInstrument” จะทำการแทรกรหัสต้นทางของไฟล์คอมโพสิต โดยเรียกใช้งานฟังก์ชัน “instrumentComposite” ที่อยู่ภายใต้คลาส “facade_instrument” จากนั้น คลาส “facade_instrument” จะเรียกใช้งานคลาส “ctrl_InstrumentComposite” เพื่อแทรกรหัสต้นทางที่ไฟล์คอมโพสิตให้สามารถอ้างอิงไฟล์เว็บเซอร์วิสที่เก็บข้อมูลได้ หลังจากแทรกรหัสต้นทางของไฟล์คอมโพสิตแล้ว เครื่องมือจะคัดลอกไฟล์ดับเบิลยูเอสดีแอล โดยที่ คลาส “facade_instrument” จะเรียกใช้งานฟังก์ชัน “copyWSDLFile” ที่อยู่ภายใต้คลาส “ctrl_InstrumentWSDL” เพื่อคัดลอกไฟล์ดับเบิลยูเอสดีแอล จากนั้นจะทำการแทรกรหัสต้นทางที่ไฟล์ดับเบิลยูเอส-บีเพล ซึ่งคลาส “frm_readInstrument” จะเรียกใช้งานฟังก์ชัน “InstrumentBPEL” ที่อยู่ภายใต้คลาส “facade_instrument” โดยในการแทรกรหัสต้นทางของไฟล์ดับเบิลยูเอส-บีเพลจะมีลำดับดังนี้ การประกาศตัวแปรสำหรับข้อมูลนำเข้าและข้อมูลส่งออกของเว็บเซอร์วิสที่ทำหน้าที่จัดเก็บข้อมูล การประกาศตัวแปรนำเข้าและตัวแปรส่งออกที่มีหน้าที่รับค่าหมายเลขการทดสอบ การร้องขอรหัสทดสอบขณะที่ทำการทดสอบ การประกาศขอบเขตของเว็บเซอร์วิสที่จัดเก็บข้อมูล การแทรกโหนดจัดเก็บข้อมูลของโหนดที่มีการติดต่อกับพาร์ทเนอร์ลิงก์ ได้แก่ โหนด “Invoke” “Receive” “Reply” และโหนดเงื่อนไข เช่น “If” “else”



รูปที่ 4-14 แผนภาพลำดับ ของยูสเคสฮัพโพลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเพล (ในขั้นตอนการแทรกรหัสต้นทางที่ไฟล์คอมโพสิตและคัดลอกไฟล์ดับเบิลยูเอสดี)

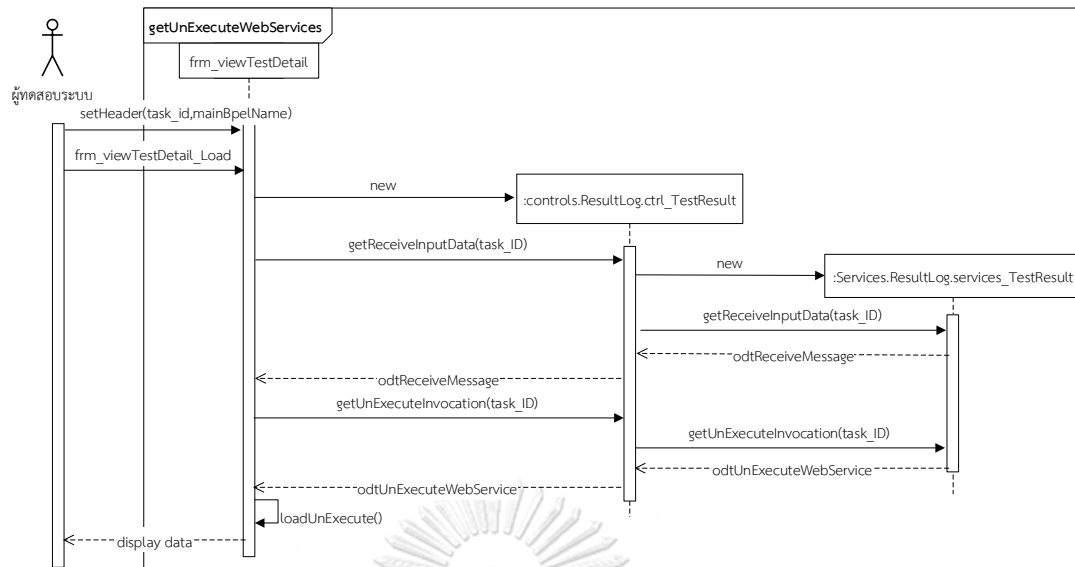


รูปที่ 4-15 แผนภาพลำดับ ของยูสเคสอ็อปโพลด์และแทรกรหัสต้นทางโครงการดับเบิ้ลยูเอส-บีเพล (ในขั้นตอนการแทรกรหัสต้นทางที่ไฟล์ดับเบิ้ลยูเอส-บีเพล)

2. การค้นหาเว็บเซอร์วิซที่ไม่ถูกทดสอบ

ในการค้นหาเว็บเซอร์วิซที่ไม่ถูกทดสอบผู้วิจัย ได้ออกแบบแผนภาพลำดับดังรูปที่ 4-16 โดยมีรายละเอียดการทำงานดังนี้

- ผู้ทดสอบระบบเลือกโครงการดับเบิ้ลยูเอส-บีเพลที่ต้องการดูข้อมูลเว็บเซอร์วิซที่ไม่ถูกทดสอบ โดยเลือกหมายเลขรหัสติดตามหรือชื่อโครงการ เพื่อเป็นการตั้งคำรหัสและชื่อโครงการให้กับคลาส “frm_viewTestDetail” จากนั้นจะเรียกใช้งานคลาส “frm_viewTestDetail” เพื่อเปิดหน้าต่างส่วนต่อประสานและอ่านข้อมูลการทดสอบเพื่อตรวจสอบข้อมูลนำเข้าของโครงการ โดยเรียกใช้ฟังก์ชัน “getReceiveInputData” ที่อยู่ภายใต้คลาส “ctrl_TestResult” จากนั้นคลาส “ctrl_TestResult” จะเรียกใช้งานคลาส “services_TestResult” เพื่ออ่านข้อมูลนำเข้าที่ถูกจัดเก็บในฐานข้อมูล และคืนค่าให้กับคลาส “frm_viewTestDetail” จากนั้นคลาส “frm_viewTestDetail” จะเรียกใช้งานฟังก์ชัน “getUnExecuteInvocation” ที่อยู่ภายใต้คลาส “ctrl_TestResult” เพื่อดึงข้อมูลเว็บเซอร์วิซที่ไม่ถูกทดสอบ ซึ่งคลาส “ctrl_TestResult” จะเรียกใช้งานคลาส “services_TestResult” เพื่อเข้าถึงข้อมูลที่ถูกจัดเก็บในฐานข้อมูล หลังจากนั้นคลาส “frm_viewTestDetail” จะนำข้อมูลที่ได้อามาแสดงผลโดยเรียกใช้ฟังก์ชัน “loadUnExecute”



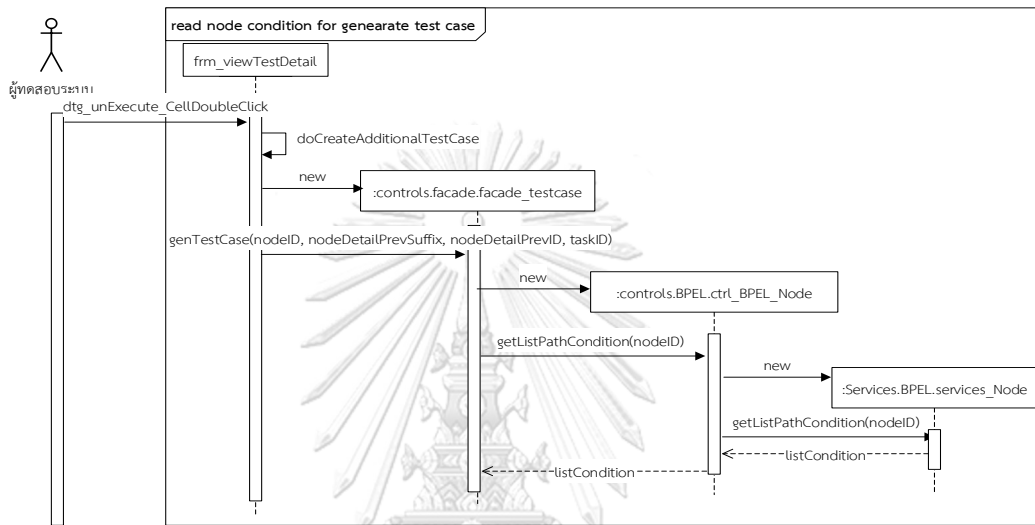
รูปที่ 4-16 แผนภาพลำดับ ของยูสเคสดูเว็บเซอร์วิสที่ไม่ถูกทดสอบ

3. การสร้างกรณีทดสอบเพิ่มเติม

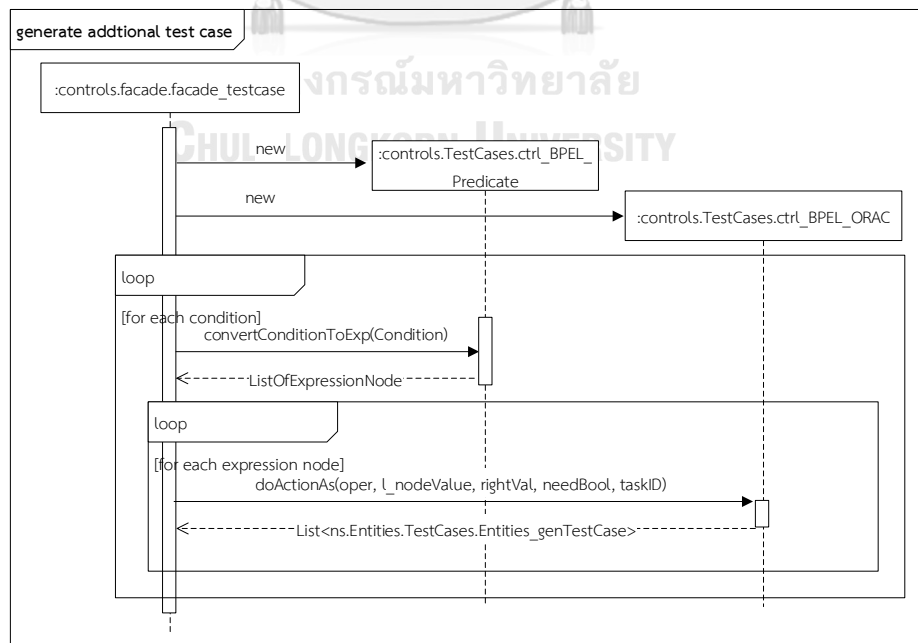
ผู้วิจัยได้ออกแบบแผนภาพลำดับสำหรับการสร้างกรณีทดสอบเพิ่มเติมดังรูปที่ 4-17 ถึง 4-19 โดยมีรายละเอียดการทำงานดังนี้

- ผู้ทดสอบระบบเลือกโหนดพาร์ทเนอร์ลิงก์ที่ยังไม่ถูกทดสอบ เพื่อสร้างกรณีทดสอบใหม่ จากนั้นคลาส “frm_viewTestDetail” จะเรียกใช้งานฟังก์ชัน “genTestCase” ที่อยู่ในไค้คลาส “facade_testcase” เพื่อเริ่มกระบวนการสร้างกรณีทดสอบเพิ่มเติม คลาส “facade_testcase” จะเรียกใช้งานฟังก์ชัน “getListPathCondition” ที่อยู่ในไค้คลาส “ctrl_BPEL_Node” เพื่ออ่านข้อมูลโหนดเงื่อนไขจากโหนดเริ่มต้นจนถึงโหนดเงื่อนไขที่ต้องการทดสอบ โดยคลาส “ctrl_BPEL_Node” จะเรียกใช้งานคลาส “services_Node” เพื่อติดต่อกับฐานข้อมูล เมื่อได้โหนดเงื่อนไขแล้วเครื่องมือจะนำโหนดเงื่อนไขที่ได้แปลงเป็นโหนดประมวลผล โดยเรียกใช้งานฟังก์ชัน “convertConditionToExp” ที่อยู่ในไค้คลาส “ctrl_BPEL_Predicate” โดยจะนำแต่ละโหนดเงื่อนไขที่ได้มาประมวลผล จากนั้นเครื่องมือจะเรียกใช้งานคลาส “ctrl_BPEL_ORAC” เพื่อนำโหนดประมวลผลที่ได้ส่งไปให้ฟังก์ชัน “doActionAs” ซึ่งในฟังก์ชันจะทำการเปลี่ยนแปลงค่าให้เป็นไปตามเงื่อนไขของเส้นทางที่ต้องการ เมื่อได้ค่าที่ควรจะเป็นของแต่ละเงื่อนไขแล้วเครื่องมือจะอ่านค่ากรณีทดสอบล่าสุด โดยเรียกใช้งานฟังก์ชัน “getLastestTC” ที่อยู่ในไค้คลาส “ctrl_TestCase” และคลาส “ctrl_TestCase” จะเรียกใช้งานคลาส “services_TestCase” เพื่อเข้าถึงข้อมูลใน

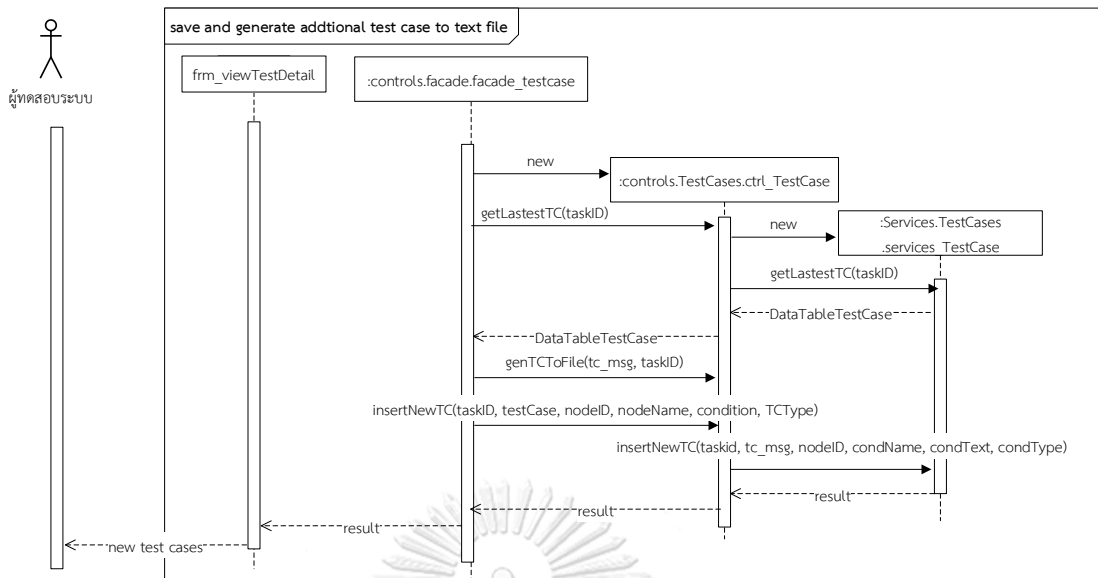
ฐานข้อมูล เมื่อได้กรณืทดสอบแล้วเครื่องมือจะนำค่าที่ควรจะเป็นมาแทนค่าตามตัวแปรที่อยู่ในกรณืทดสอบและ บันทึกข้อมูลในรูปแบบไฟล์ข้อความ ผ่านฟังก์ชัน “genTCToFile” ที่อยู่ภายใต้คลาส “ctrl_TestCase” และนอกจากนี้ยังนำกรณืทดสอบใหม่ที่ได้ไปบันทึกในฐานข้อมูลเพื่อเป็นประวัติการสร้างกรณืทดสอบ โดยจะเรียกใช้งานฟังก์ชัน “insertNewTC” ที่อยู่ภายใต้คลาส “ctrl_TestCase” และเรียกใช้งานคลาส “services_TestCase” เพื่อนำข้อมูลไปจัดเก็บในฐานข้อมูล



รูปที่ 4-17 แผนภาพลำดับ ของยูสเคสสร้างกรณืทดสอบเพิ่มเติม (ในขั้นตอนอ่านโหนดเงื่อนไขของเว็บเซอร์วิชที่ไม่ถูกทดสอบ)



รูปที่ 4-18 แผนภาพลำดับ ของยูสเคสสร้างกรณืทดสอบเพิ่มเติม(ในขั้นตอนการสร้างกรณืทดสอบ)



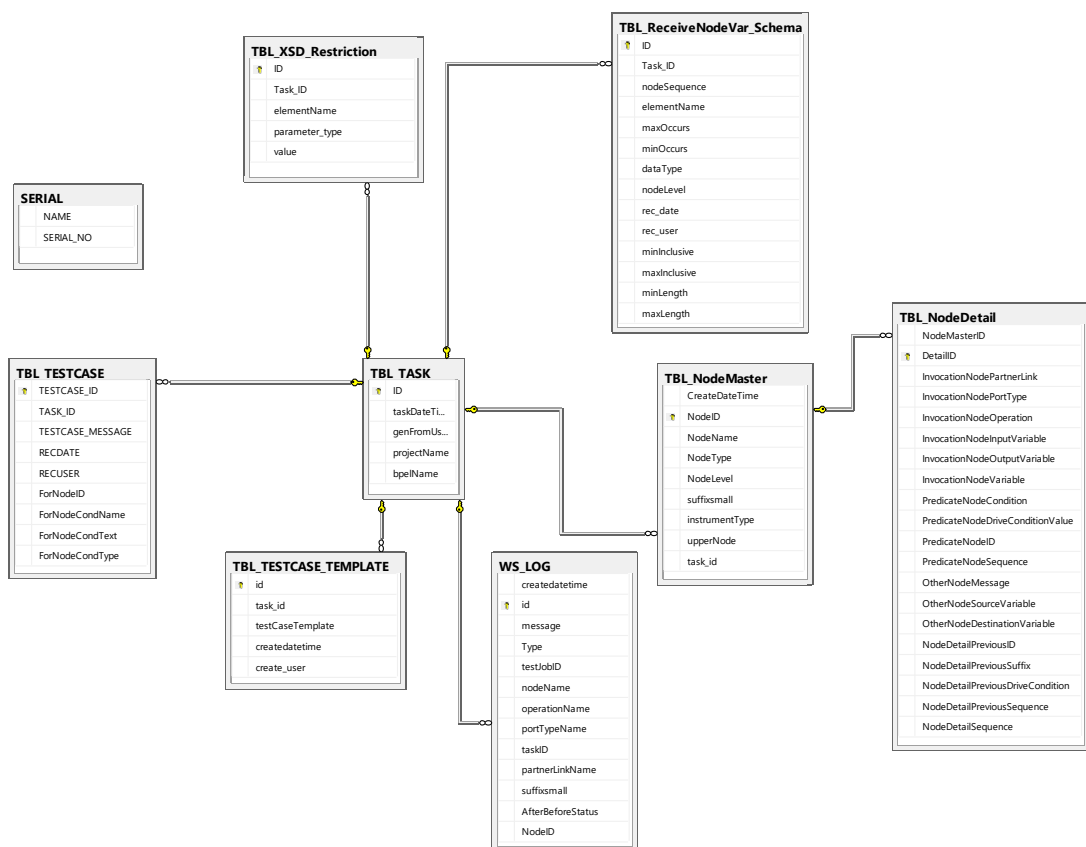
รูปที่ 4-19 แผนภาพลำดับ ของยูสเคสสร้างกรณีทดสอบเพิ่มเติม(ในขั้นตอนบันทึกและเขียนไฟล์ทดสอบเพิ่มเติม)

4.1.5. โครงสร้างของฐานข้อมูล

จากการออกแบบเครื่องมือ ผู้วิจัยได้ออกแบบฐานข้อมูลสำหรับจัดเก็บข้อมูลต่างๆ ดังรูปที่ 4-20 โดยมีรายละเอียดแต่ละตารางดังนี้ และอธิบายพจนานุกรมข้อมูลไว้ใน ภาคผนวก ค

- 1) ตาราง TBL_TASK เป็นตารางจัดเก็บข้อมูลรหัสการติดตามของโครงการระดับเบิลยูเอส-บีเฟล
- 2) ตาราง TBL_NodeMaster เป็นตารางจัดเก็บข้อมูลโหนดของดับเบิลยูเอส-บีเฟล ประกอบด้วยข้อมูลต่างๆ เช่น รหัสโหนด รหัสติดตามข้อมูลโหนด ชื่อโหนด ประเภทโหนด ลำดับของโหนด เป็นต้น
- 3) ตาราง TBL_NodeDetail เป็นตารางจัดเก็บข้อมูลรายละเอียดของโหนด ประกอบด้วยข้อมูลต่างๆ เช่น รหัสโหนด รหัสรายละเอียดโหนด ข้อมูลเงื่อนไขของโหนด ข้อมูลการเรียกใช้งานเว็บเซอร์วิส เป็นต้น
- 4) ตาราง TBL_ReceiveNodeVar_Schema เป็นตารางจัดเก็บโครงสร้างข้อมูลนำเข้าของโครงการดับเบิลยูเอส-บีเฟล ที่อ่านมาจากไฟล์เอกซ์เอสดี
- 5) ตาราง TBL_XSD_Restriction เป็นตารางจัดเก็บข้อจำกัดของข้อมูลนำเข้าที่อ่านมาจากไฟล์เอกซ์เอสดี

- 6) ตาราง TBL_TESTCASE เป็นตารางจัดเก็บข้อมูลกรณีทดสอบเพิ่มเติมที่ถูกสร้างจากเครื่องมือ
- 7) ตาราง TBL_TESTCASE_TEMPLATE เป็นตารางจัดเก็บข้อมูลกรณีทดสอบตั้งต้นที่ถูกสร้างจากเครื่องมือ
- 8) ตาราง WS_LOG เป็นตารางจัดเก็บข้อมูลผลการทดสอบที่บันทึกโดยเว็บเซอร์วิสสำหรับเก็บผลการทดสอบ
- 9) ตาราง SERIAL เป็นตารางจัดเก็บข้อมูลรหัสที่ถูกสร้างขึ้นมาเพื่อนำไปใช้ในตารางต่างๆ



รูปที่ 4-20 แผนภาพโครงสร้างฐานข้อมูลของเครื่องมือ

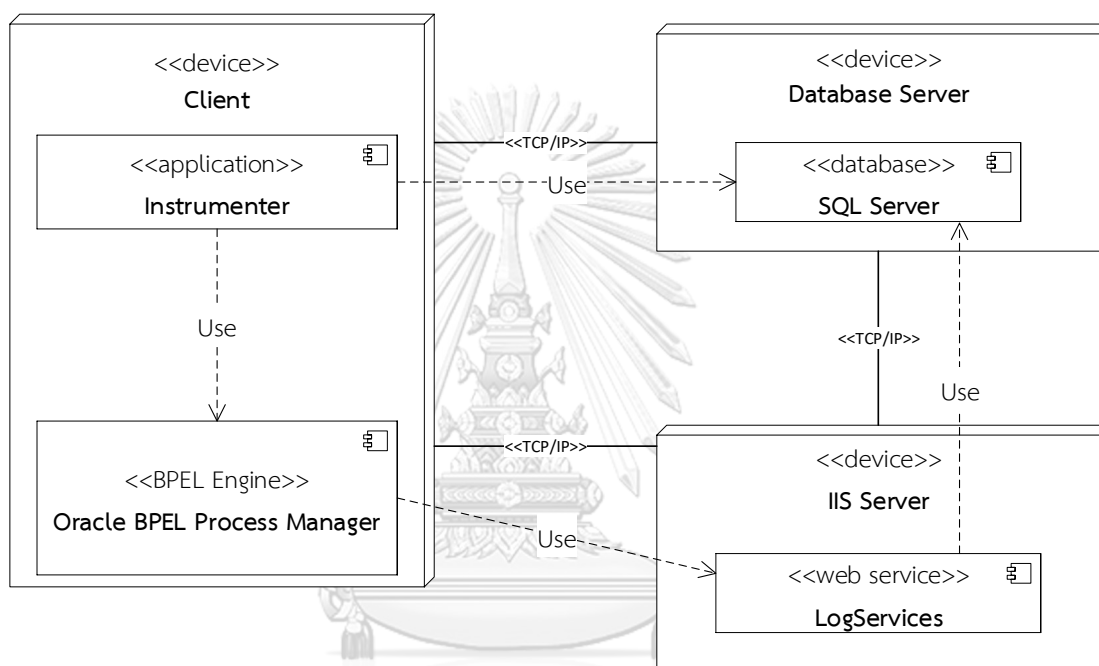
4.1.6. แผนภาพการติดตั้ง

เป็นแผนภาพที่อธิบายถึงสถาปัตยกรรมของซอฟต์แวร์และการติดตั้งซอฟต์แวร์บนอุปกรณ์ต่างๆ โดยผู้วิจัยได้ออกแบบการแผนภาพการติดตั้ง ดังรูปที่ 4-21 ซึ่งแบ่งเป็น 3 อุปกรณ์ได้แก่

- เครื่องลูกข่าย จะติดตั้งเครื่องมือตรวจจัดการเรียกใช้เว็บเซอร์วิสสำหรับทดสอบดับเบิลยูเอส-บีเพล โดยเครื่องมือจะติดต่อกับเครื่องฐานข้อมูลแม่ข่าย เพื่อบันทึกและเรียกดู

ข้อมูลต่างๆ และยังติดตั้งเครื่องมือบีเพลเอนจินเพื่อประมวลผลและทดสอบโครงการ
ดับเบิลยูเอส-บีเพล ที่ถูกแทรกรหัสต้นทางแล้ว

- เครื่องฐานข้อมูลแม่ข่าย จะติดตั้งฐานข้อมูลเอสคิวแอล เซิร์ฟเวอร์ เพื่อจัดเก็บข้อมูลที่ส่งมาจากเครื่องมือและเว็บเซอร์วิสที่จัดเก็บข้อมูลการทดสอบ
- เครื่องไอไอเอส เซิร์ฟเวอร์ จะติดตั้งเว็บเซอร์วิสสำหรับจัดเก็บข้อมูลการทดสอบที่ได้รับมาจากดับเบิลยูเอส-บีเพล และจะจัดเก็บข้อมูลนั้นไปยังฐานข้อมูล



รูปที่ 4-21 แผนภาพการติดตั้งเครื่องมือ

4.2. การพัฒนาเครื่องมือสนับสนุน

จากการออกแบบเครื่องมือข้างต้น ผู้วิจัยได้กำหนด สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ
โครงสร้างฐานข้อมูล และโครงสร้างส่วนต่อประสานของเครื่องมือดังนี้

4.2.1. สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ

ผู้วิจัยกำหนดสภาพแวดล้อมที่จะนำมาพัฒนาเครื่องมือดังนี้

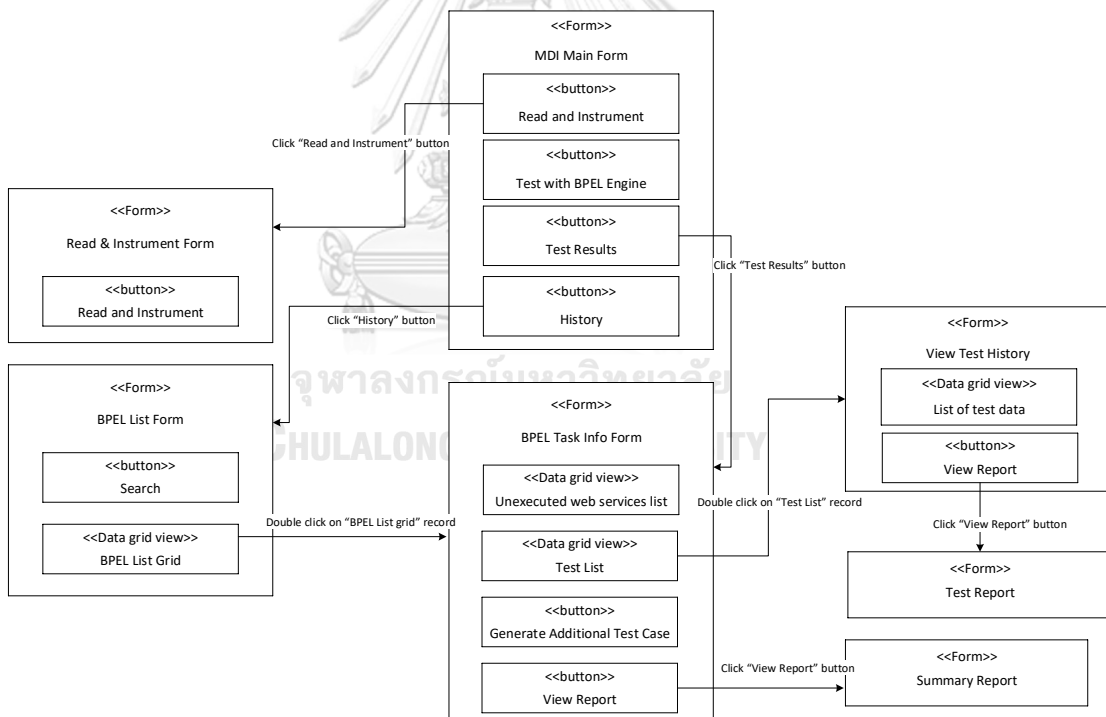
ฮาร์ดแวร์ (Hardware)

- เครื่องคอมพิวเตอร์ชนิดพกพา หน่วยประมวลผลอินเทลคอร์ไอไฟร์ท 2.5 กิกะเฮิร์ตซ์ (Intel Core i5 2.5 GHz)
- หน่วยความจำของคอมพิวเตอร์ (RAM) 20 กิกะไบต์ (20 GB)

- ฮาร์ดดิสก์ (Harddisk) 256 กิกะไบต์ (256 GB)
- ซอฟต์แวร์ (Software)
- ระบบปฏิบัติการไมโครซอฟต์วินโดวส์ 10 (64 บิต)
 - ฐานข้อมูลเอสคิวแอล เซิร์ฟเวอร์ 2014 (Microsoft SQL Server 2014)
 - ไอไอเอส เซิร์ฟเวอร์ 10 (IIS 10)
 - วิซวลสตูดิโอ 2015 (Visual Studio 2015)

4.2.2. โครงสร้างส่วนต่อประสานกับผู้ใช้ของเครื่องมือสนับสนุน

ผู้วิจัยได้นำแผนภาพวินโดวส์เนวิเกชัน (Window Navigation) ซึ่งเป็นแผนภาพที่อธิบายความสัมพันธ์ระหว่างส่วนต่อประสานและผู้ใช้งานระบบ มาอธิบายความสัมพันธ์ของเครื่องมือ ดังรูปที่ 4-22 ประกอบด้วยหน้าจอการทำงาน 5 หน้าจอ โดยมีรายละเอียดแต่ละหน้าจอดังนี้



รูปที่ 4-22 แผนภาพวินโดวส์เนวิเกชันของเครื่องมือ

1. หน้าต่างอัปโหลดไฟล์และแทรกรหัสต้นทาง

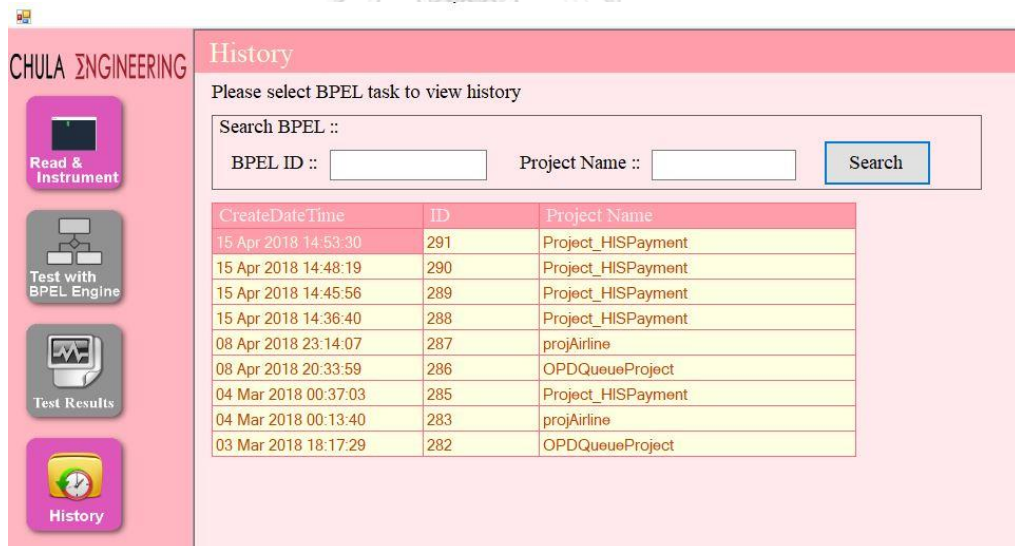
เป็นหน้าต่างสำหรับผู้ทดสอบนำแฟ้มโครงการดับเบิลยูเอส-บีเพล มาอัปโหลด ผ่านเครื่องมือเพื่อบันทึกเส้นทางการไหลเข้าสู่ฐานข้อมูลของเครื่องมือ และแทรกรหัสต้นทางที่ไฟล์ต่างๆ ของโครงการ โดยมีหน้าจอ ดังรูปที่ 4-23



รูปที่ 4-23 หน้าต่างอัปโหลดไฟล์และแทรกรหัสต้นทาง

2. หน้าต่างดูประวัติการทดสอบโครงการดับเบิลยูเอส-บีเพลที่ถูกจัดเก็บในระบบ

เป็นหน้าต่างสำหรับแสดงรายการของโครงการดับเบิลยูเอส-บีเพลที่ถูกจัดเก็บในระบบ ที่ผู้ทดสอบสามารถเรียกดูประวัติการทดสอบของโครงการได้ โดยมีตัวอย่างหน้าต่าง ดังรูปที่ 4-24

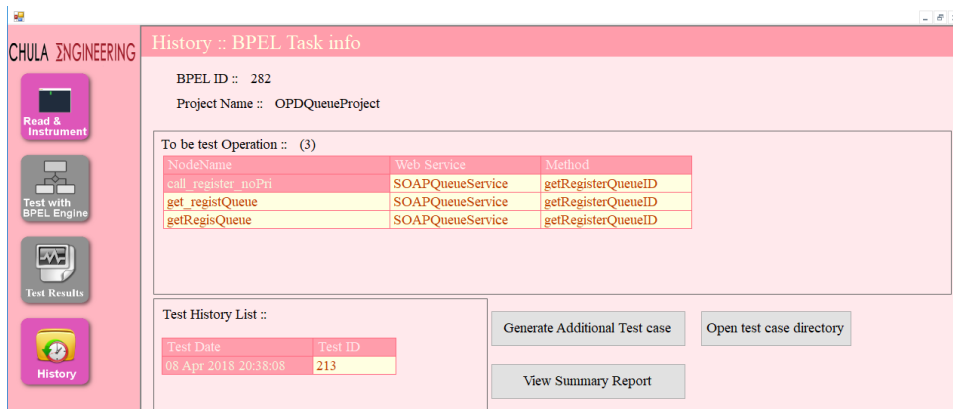


รูปที่ 4-24 หน้าต่างสำหรับแสดงรายการบีเพลที่ถูกจัดเก็บในระบบ

3. หน้าต่างแสดงรายละเอียดการทดสอบ โครงการดับเบิลยูเอส-บีเพล

เป็นหน้าต่างแสดงรายละเอียดการทดสอบของโครงการดับเบิลยูเอส-บีเพล ที่ผู้ทดสอบระบุ โดยภายในหน้าจอจะแสดงรายการเว็บเซอร์วิสที่ไม่ถูกทดสอบ ซึ่งจะแสดงข้อมูลต่างๆ

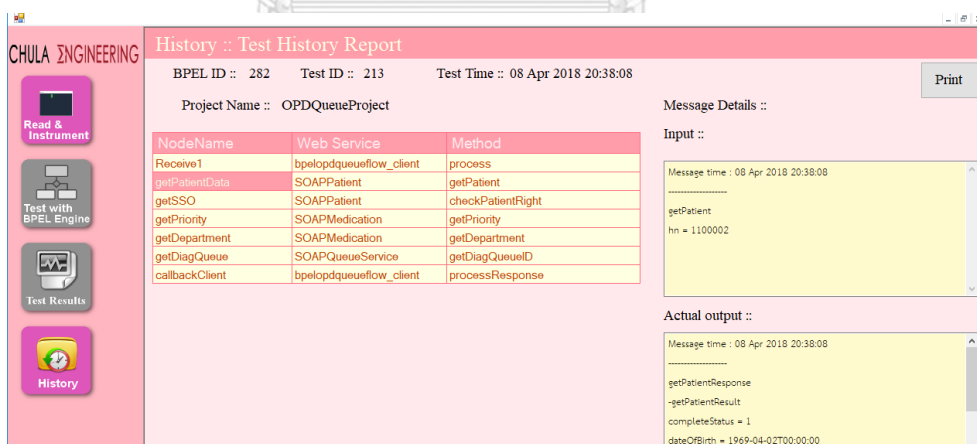
เช่น ชื่อของโหนด เว็บเซอร์วิส ฟังก์ชันของเว็บเซอร์วิส ซึ่งผู้ทดสอบยังสามารถสร้างกรณีทดสอบเพิ่มเติมได้โดยดับเบิลคลิกเลือกที่รายการเว็บเซอร์วิสที่ไม่ถูกทดสอบ นอกจากนี้เครื่องมือยังแสดงประวัติการทดสอบที่ด้านล่างซ้ายของหน้าต่างเพื่อให้ผู้ทดสอบดับเบิลคลิกเพื่อเข้าไปดูรายละเอียดการทดสอบในแต่ละครั้งได้ ดังรูปที่ 4-25



รูปที่ 4-25 หน้าต่างแสดงรายละเอียดบีเพล

4. หน้าต่างดูผลการทดสอบ

เป็นหน้าต่างแสดงรายละเอียดผลการทดสอบที่เลือก โดยแสดงรายการเว็บเซอร์วิสและฟังก์ชันของเว็บเซอร์วิสที่ถูกทดสอบ พร้อมทั้งแสดงข้อมูลนำเข้าและข้อมูลส่งออกที่รับส่งระหว่างเว็บเซอร์วิส ดังรูปที่ 4-26



รูปที่ 4-26 หน้าต่างดูผลการทดสอบ

5. หน้าต่างแสดงรายงานผลการทดสอบการเรียกใช้เว็บเซอร์วิส

เป็นหน้าต่างสำหรับแสดงรายงานผลการทดสอบการเรียกใช้เว็บเซอร์วิสของโครงการดับเบิลยูเอส-บีเพล โดยจะแสดงรายละเอียดของการทดสอบในครั้งนั้นๆ เช่น ข้อมูลนำเข้า ข้อมูลส่งออก ของแต่ละเว็บเซอร์วิสที่ถูกทดสอบ ดังรูปที่ 4-27

Test History Report

Test Details Report

Print Date Time : 22May2018 20:40:31

BPEL ID : 282 Project Name : OPDQueueProject Test Time : 08 Apr 2018 20:38:08

Node Name	Web Service	Method	Input	Output
Receive1	bpelopdqueueflow_client	process	Message time : 08 Apr 2018 20:38:08 ----- process hn = 1100002 sso_id = SSO department_id = 1 priority_id = 1	Message time :
getPatientData	SOAPPatient	getPatient	Message time : 08 Apr 2018 20:38:08 ----- getPatient hn = 1100002	Message time : 08 Apr 2018 20:38:08 ----- getPatientResponse -getPatientResult completeStatus = 1 dateOfBirth = 1969-04-02T00:00:00 fullname = Mr.PatientA LastNameA gender = M hn = 1100002
getSSO	SOAPPatient	checkPatientRight	Message time : 08 Apr 2018 20:38:09 ----- checkPatientRight	Message time : 08 Apr 2018 20:38:09 ----- checkPatientRightResponse

Current Page No: 1 Total Page No: 1 Zoom Factor: 200%

รูปที่ 4-27 หน้าต่างแสดงรายงานผลสรุปการทดสอบ

6. รายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส

เป็นหน้าต่างสำหรับแสดงรายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิสของโครงการดับเบิลยูเอส-บีเพล โดยจะแสดงรายการเว็บเซอร์วิสที่ไม่ถูกทดสอบ จำนวนครั้งที่ทดสอบ และจำนวนครั้งที่สร้างกรณีทดสอบเพิ่มเติม ดังรูปที่ 4-28

Test Summary Report

Print Date Time : 22May2018 20:38:24

BPEL ID : 282 Project Name : OPDQueueProject

Total Test : 1 Times

Total Generate Test Case: 10 Times

Total Not Coverage Invocation : 1

Not Coverage Invocation List

Node Name	Method	Condition	Branch
call_register_noPri	getRegisterQueueID	\$inputVariable.payload/client.priority_id != ""	False
get_regisQueue	getRegisterQueueID	\$inputVariable.payload/client.sso_id != ""	False
getRegisQueue	getRegisterQueueID	\$inputVariable.payload/client.hn != ""	False

Current Page No: 1 Total Page No: 1 Zoom Factor: 200%

รูปที่ 4-28 หน้าต่างแสดงรายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส

บทที่ 5

การทดสอบเครื่องมือ

5.1. การทดสอบเครื่องมือ

ผู้วิจัยได้สร้างกรณีศึกษาสำหรับทดสอบการทำงานของเครื่องมือ ประกอบด้วย 3 กรณีศึกษา ซึ่งในแต่ละกรณีศึกษาจะมีการเรียกใช้งานเว็บเซอร์วิสจำนวน 3 เว็บเซอร์วิส โดยผู้วิจัยอธิบายรายละเอียดของการทดสอบในแต่ละกรณีศึกษาไว้ในภาคผนวก ข. ซึ่งกรณีศึกษาที่นำมาใช้ทดสอบการทำงานของเครื่องมือมีดังนี้

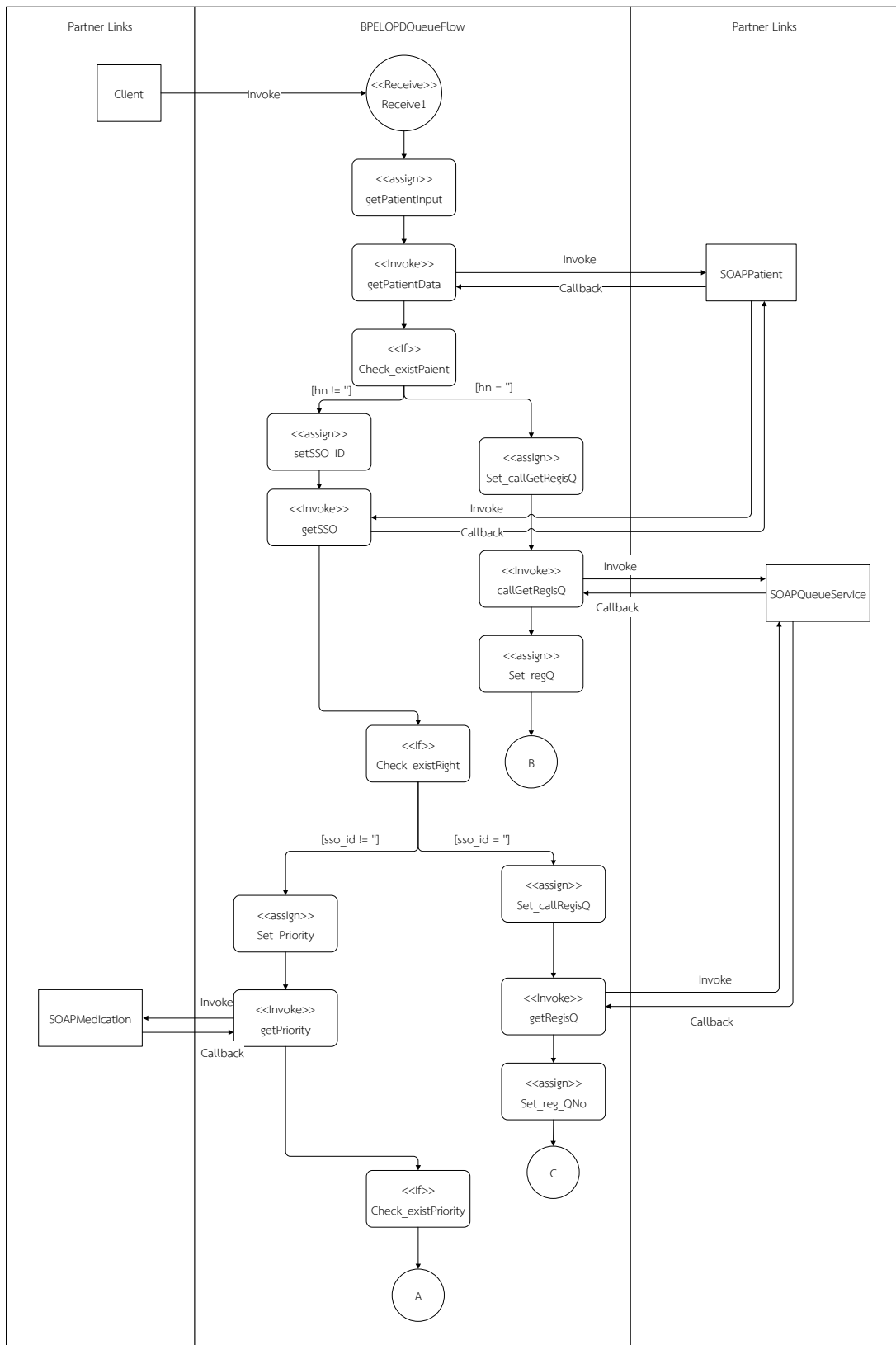
5.1.1. กรณีศึกษาที่ 1 ระบบออกบัตรคิวผู้ป่วยนอก

ระบบออกบัตรคิวผู้ป่วยนอก มีขั้นตอนการทำงานเริ่มจากผู้ป่วยเข้ามาใช้บริการที่แผนกผู้ป่วยนอก โดยในการรับบัตรคิวผู้ป่วยจะต้องแจ้งรหัสประจำตัวผู้ป่วย (HN) หรือบัตรประจำตัวประชาชน พร้อมทั้งแจ้งสิทธิการรักษา ให้กับเจ้าหน้าที่ จากนั้นเจ้าหน้าที่จะคัดกรองผู้ป่วยและออกบัตรคิวผู้ป่วยไปยังคลินิกต่างๆ นอกจากนี้เจ้าหน้าที่ยังสามารถระบุความเร่งด่วนของผู้ป่วยเพื่อออกบัตรคิวพิเศษได้ และในกรณีที่ผู้ป่วยยังไม่มีหมายเลขประจำตัว หรือไม่พบสิทธิการรักษา เจ้าหน้าที่จะต้องส่งผู้ป่วยไปยังห้องเวชระเบียนเพื่อทำบัตรประจำตัวผู้ป่วย หรือตรวจสอบสิทธิการรักษา โดยเจ้าหน้าที่จะต้องออกบัตรคิวห้องเวชระเบียนให้แก่ผู้ป่วยเพื่อติดต่อเจ้าหน้าที่ห้องเวชระเบียน

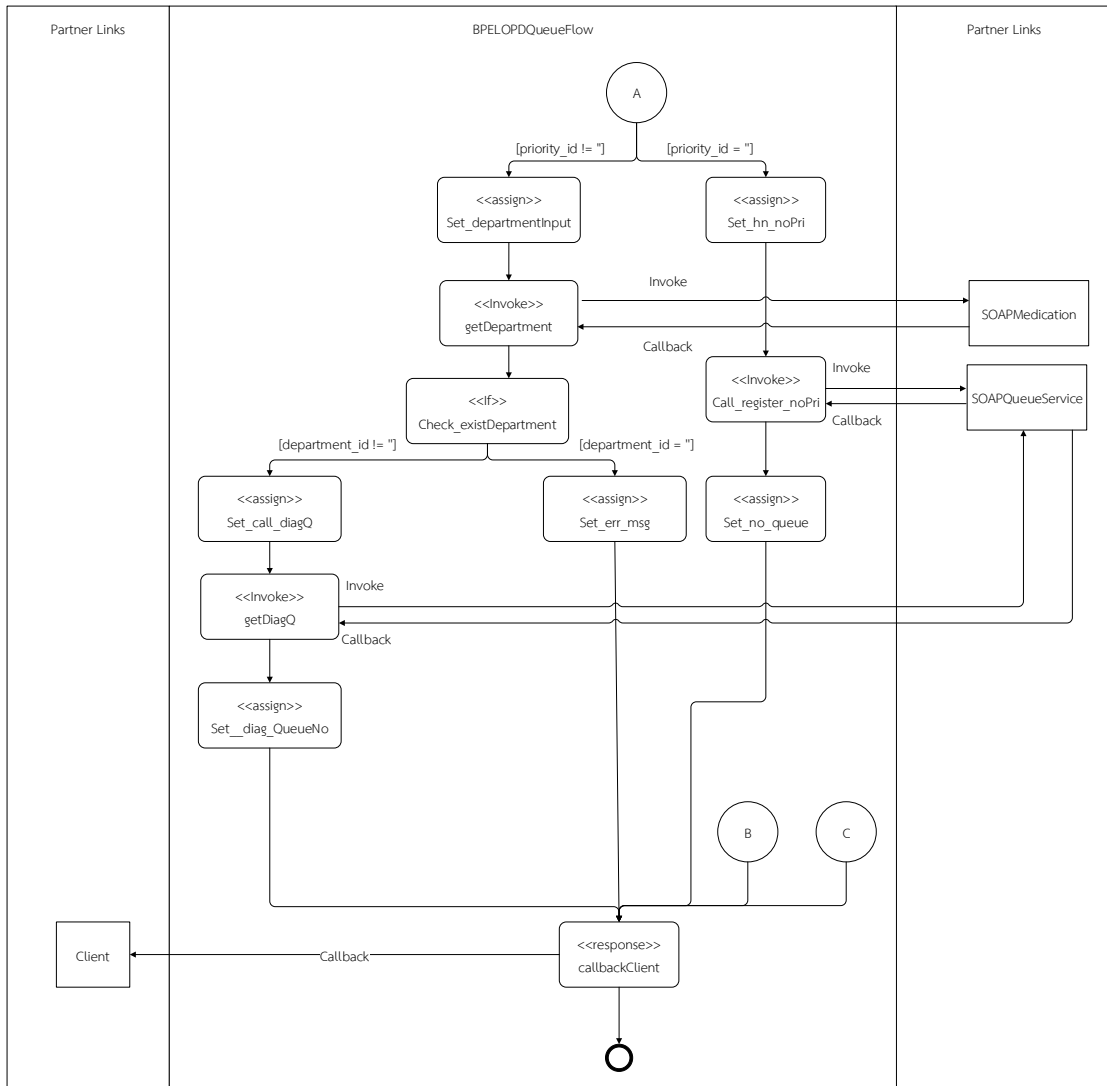
ในกรณีศึกษานี้ผู้วิจัยต้องการทดสอบการทำงานของเครื่องมือว่าสามารถแสดงข้อมูลนำเข้าและข้อมูลส่งออกที่รับส่งระหว่างเว็บเซอร์วิสได้หรือไม่ โดยผู้วิจัยได้นำกรณีศึกษาดังกล่าวมาสร้างเป็นแผนภาพดับเบิลยูเอส-บีเพล โดยมีรายละเอียดดังรูปที่ 5-1 และ 5-2 และมีการติดต่อกับเว็บเซอร์วิสต่างๆ ดังตารางที่ 5-1

ตารางที่ 5-1 รายการเว็บเซอร์วิสที่เกี่ยวข้องกับแผนภาพบีเพล

ชื่อพาร์ทเนอร์ลิงก์	ชื่อฟังก์ชันเว็บเซอร์วิส
SOAPPatient	getPatient
SOAPPatient	checkPatientRight
SOAPMedicationServices	getPriority
SOAPMedicationServices	getDepartment
SOAPQueueService	getDiagQueueID
SOAPQueueService	getRegisterQueueID



รูปที่ 5-1 แผนภาพดับเบิลยูเอส-บีเฟลสำหรับกรณีศึกษาที่ 1 ในขั้นตอนการตรวจสอบข้อมูลผู้ป่วย

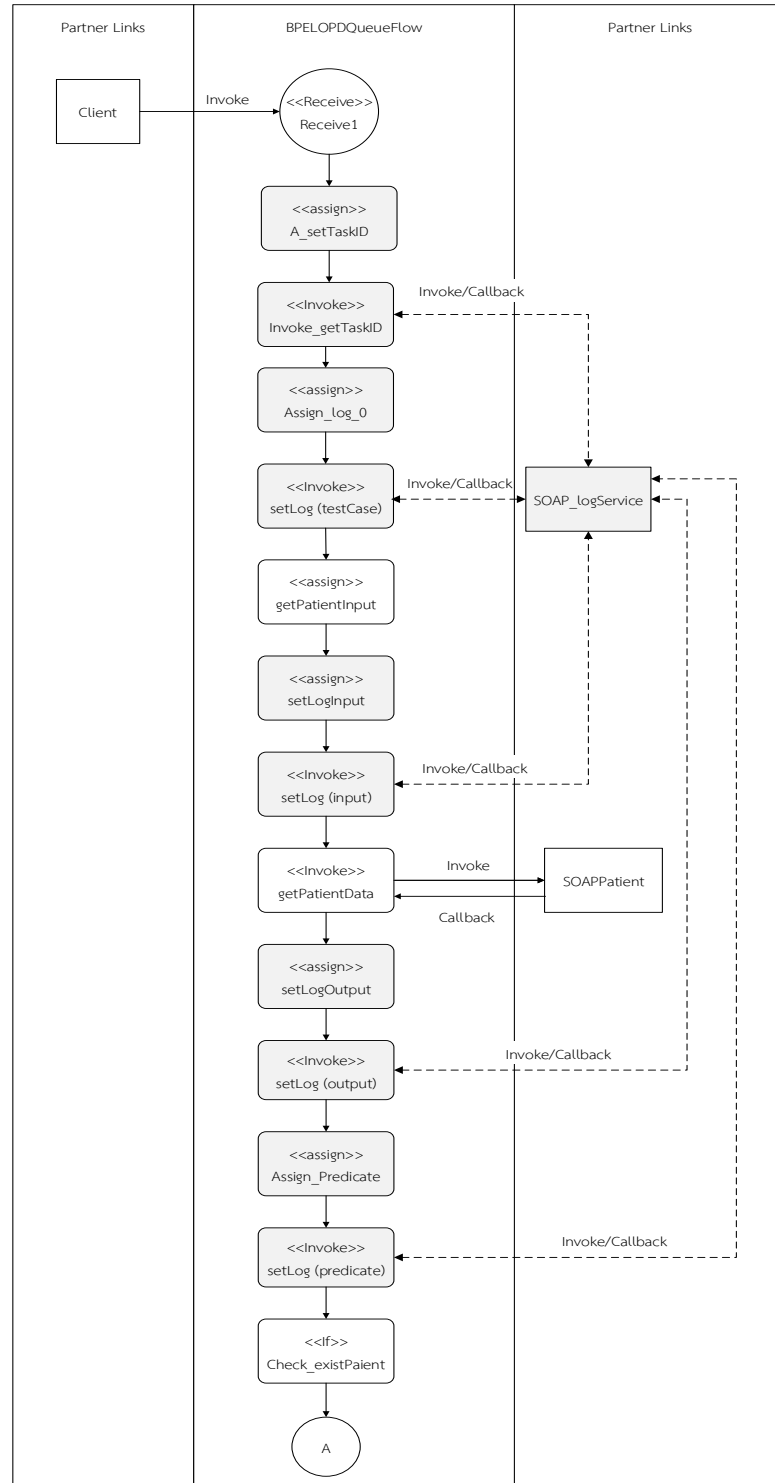


รูปที่ 5-2 แผนภาพดับเบิลยูเอส-บีเพลสำหรับกรณีศึกษาที่ 1 ในขั้นตอนการออกบัตรคิวผู้ป่วย

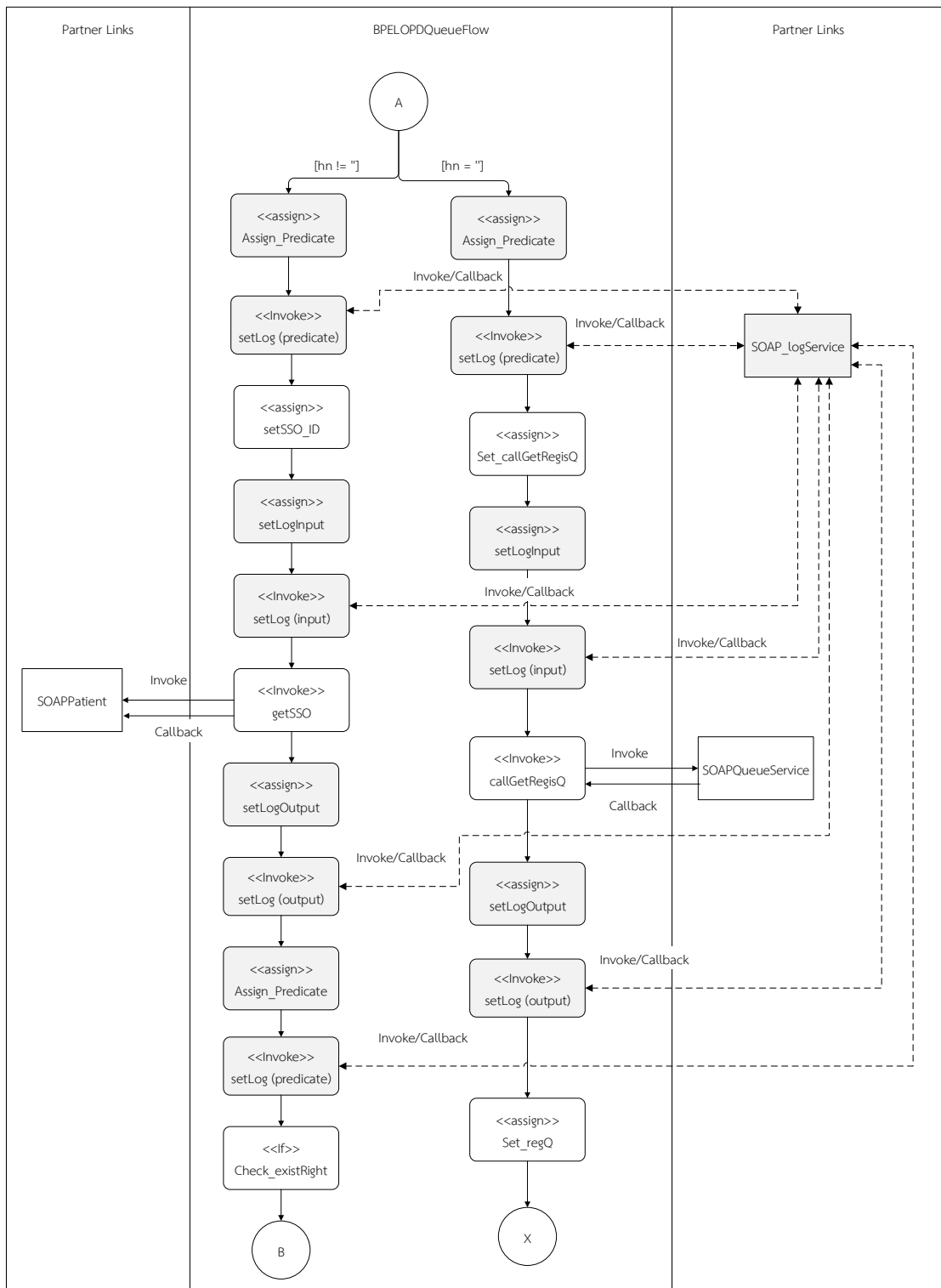
จากตารางที่ 5-1 พบว่าในโครงการดับเบิลยูเอส-บีเพลนี้มีการเรียกใช้งานฟังก์ชันเว็บเซอร์วิสทั้งหมด 6 ฟังก์ชัน ซึ่งในการทดสอบผู้ทดสอบระบบควรทดสอบการทำงานให้ครอบคลุมฟังก์ชันเว็บเซอร์วิสทั้ง 6 ฟังก์ชันนี้ ในการทดสอบโครงการดับเบิลยูเอส-บีเพล ของกรณีศึกษานี้ ผู้วิจัยจะใช้การทดสอบโดยนำโครงการมาแทรกรหัสต้นทาง จากนั้นจะนำโครงการมาติดตั้งและทดสอบผ่านบีเพลเอนจิน พร้อมทั้งกลับมาตรวจสอบผลการทดสอบที่เครื่องมือ

การทดสอบเครื่องมือเริ่มจากผู้ทดสอบเลือกแฟ้มโครงการดับเบิลยูเอส-บีเพล จากนั้นเครื่องมือจะค้นหาไฟล์ที่จะต้องอัปโหลดให้อัตโนมติ ซึ่งประกอบด้วยไฟล์ดับเบิลยูเอส-บีเพล ไฟล์เอกซ์เอสดี ไฟล์คอมโพสิต และไฟล์ดับเบิลยูเอสดีแอล โดยผู้ทดสอบจะต้องตรวจสอบไฟล์ต่างๆ อีกครั้งเพื่อยืนยันการบันทึกเส้นทางการไหลและแทรกรหัสต้นทาง เมื่อผู้ทดสอบยืนยันการอ่าน

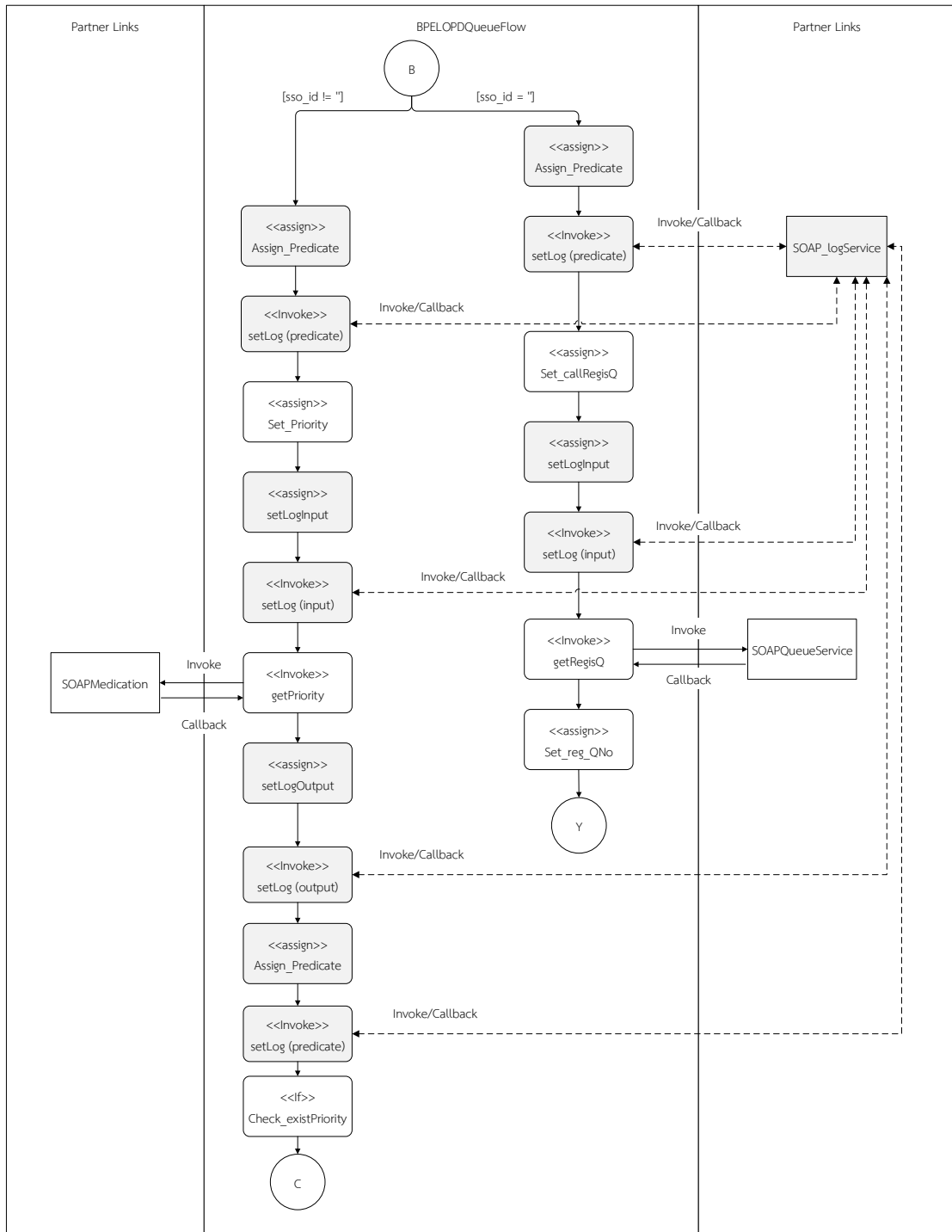
และแทรกรหัสต้นทางแล้ว เครื่องมือจะแทรกรหัสต้นทางและได้แผนภาพดับเบิลยูเอส-บีเพล ดังรูปที่ 5-3 ถึง 5-7



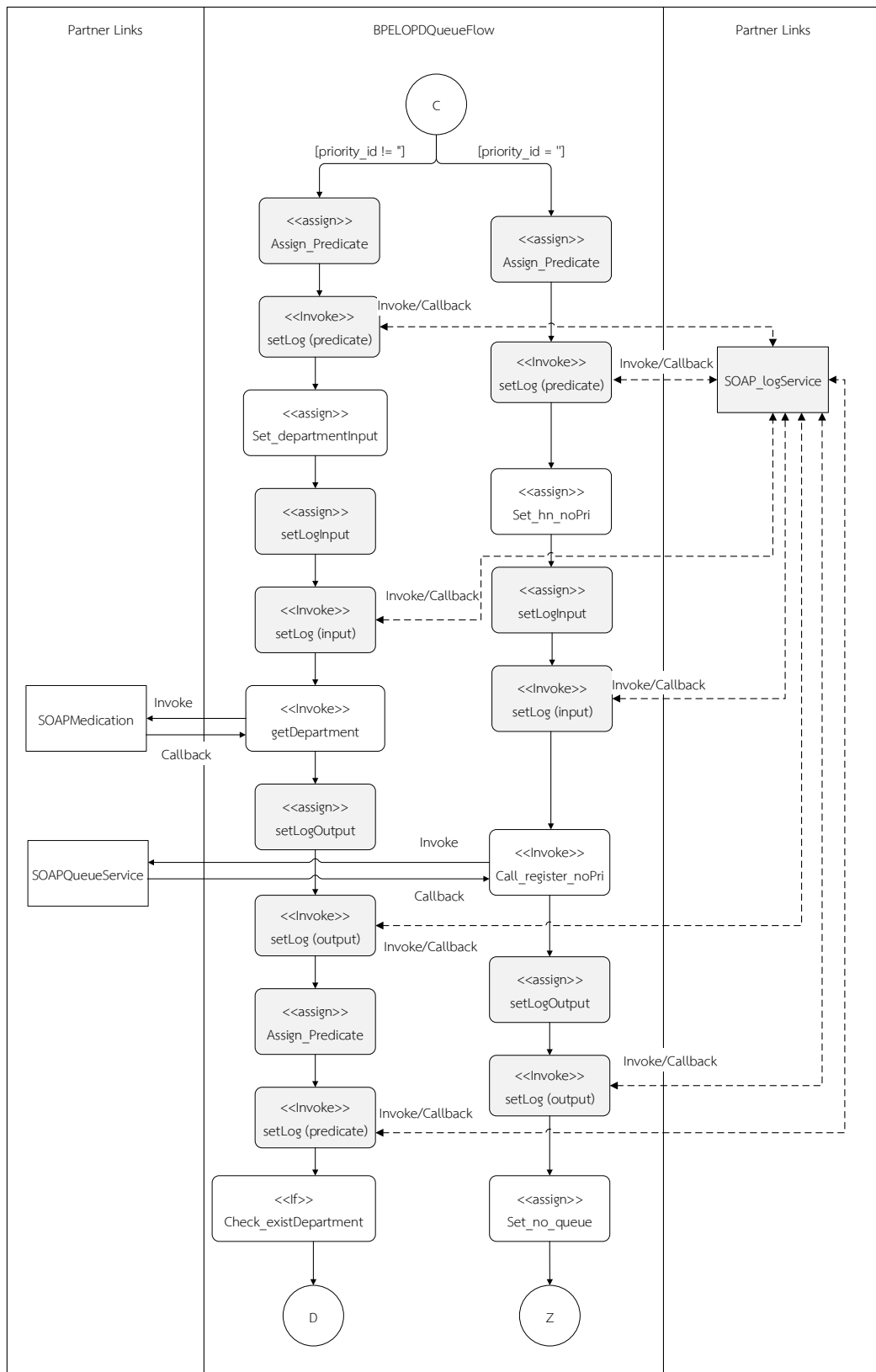
รูปที่ 5-3 แผนภาพดับเบิลยูเอส-บีเพล ที่ถูกแทรกรหัสต้นทางของกรณีศึกษาที่ 1 ในขั้นตอนการเรียกข้อมูลผู้ป่วย



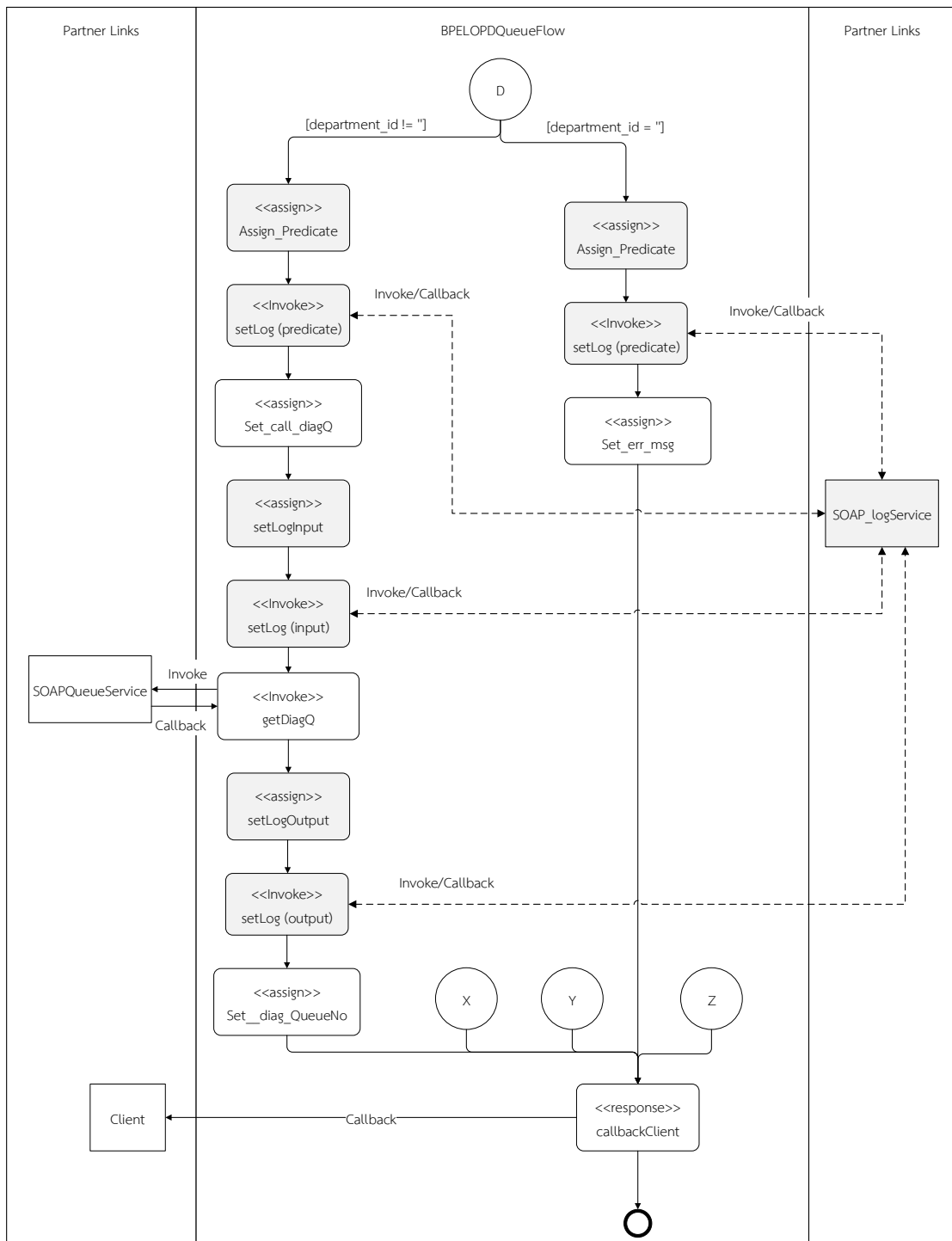
รูปที่ 5-4 แผนภาพดับเบิลยูเอส-บีเฟลที่ถูกแทรกรหัสต้นทางของกรณีศึกษาที่ 1 ในขั้นตอนการเรียกข้อมูลสิทธิการรักษา



รูปที่ 5-5 แผนภาพดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางของกรณีศึกษาที่ 1 ในขั้นตอนการเรียกข้อมูลความเร่งด่วน



รูปที่ 5-6 แผนภาพดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางของกรณีศึกษาที่ 1 ในขั้นตอนการเรียกข้อมูลแผนก



รูปที่ 5-7 แผนภาพดับเบิลยูเอส-บีเพลที่ถูกแทรกห้สต้นทางของกรณีศึกษาที่ 1 ในขั้นตอนการเรียกข้อมูลคิวตรวจรักษาผู้ป่วย

เมื่อแทรกรหัสต้นทางในโครงการดับเบิลยูเอส-บีเพลแล้ว ผู้ทดสอบสามารถนำโครงการดับเบิลยูเอส-บีเพล ที่ถูกแทรกรหัสต้นทางแล้วมาติดตั้งและทดสอบผ่านเครื่องมือบีเพลเอนจินได้ โดยในขณะที่ทำการทดสอบเมื่อเครื่องมือบีเพลเอนจินเรียกใช้งานโหนดเว็บเซอร์วิสสำหรับเก็บประวัติการทดสอบ ที่ถูกแทรกจากเครื่องมือ บีเพลเอนจินจะส่งข้อมูลการทดสอบไปยังเว็บเซอร์วิส ที่จัดเก็บประวัติการทดสอบนั้น โดยผู้ทดสอบสามารถเรียกดูเว็บเซอร์วิสที่ไม่ถูกทดสอบได้จากเครื่องมือ ดังรูปที่ 5-8

History :: BPEL Task info

BPEL ID :: 286
Project Name :: OPDQueueProject

To be test Operation :: (3)

NodeName	Web Service	Method
call_register_noPri	SOAPQueueService	getRegisterQueueID
get_registQueue	SOAPQueueService	getRegisterQueueID
getRegisQueue	SOAPQueueService	getRegisterQueueID

Test History List ::

Test Date	Test ID
08 Apr 2018 20:39:31	214

Generate Additional Test case Open test case directory

View Summary Report

รูปที่ 5-8 รายการเว็บเซอร์วิสที่ไม่ถูกทดสอบ

จากรูปที่ 5-8 เครื่องมือจะแสดงรายการเว็บเซอร์วิสที่ไม่ถูกทดสอบคือ ฟังก์ชันเว็บเซอร์วิส “getRegisterQueueID” โดยฟังก์ชันนี้จะถูกเรียกใช้งานจากโหนดต่างๆ 3 โหนด ได้แก่ โหนด “call_register_no_Pri” โหนด “get_registQueue” และ โหนด “getRegisQueue” นอกจากนี้ผู้ใช้งานยังสามารถเรียกดูประวัติข้อมูลนำเข้าและข้อมูลส่งออก ที่ถูกจัดเก็บได้ ดังรูปที่ 5-9 และสามารถดูรายงานสรุปผลการทดสอบได้โดยคลิกที่ปุ่ม “View Summary Report” ดังรูปที่ 5-10

History :: Test History Report

BPEL ID :: 286 Test ID :: 214 Test Time :: 08 Apr 2018 20:39:31 Print

Project Name :: OPDQueueProject

NodeName	Web Service	Method
Receive1	bpelopdqueueflow_client	process
getPatientData	SOAPPatient	getPatient
getSSO	SOAPPatient	checkPatientRight
getPriority	SOAPMedication	getPriority
getDepartment	SOAPMedication	getDepartment
getDiagQueue	SOAPQueueService	getDiagQueueID
callbackClient	bpelopdqueueflow_client	processResponse

Message Details ::

Input ::

Message time : 08 Apr 2018 20:39:31

getPatient
hn = 1100002

Actual output ::

Message time : 08 Apr 2018 20:39:32

getPatientResponse
-getPatientResult
completeStatus = 1
dateOfBirth = 1969-04-02T00:00:00

รูปที่ 5-9 ข้อมูลนำเข้าและข้อมูลส่งออกที่รับส่งระหว่างเว็บเซอร์วิส

Test Summary Report Print Date Time : 22May2018 20:50:39

BPEL ID : 286 Project Name : OPDQueueProject

Total Test : 1 Times
Total Generate Test Case: 0 Times
Total Not Coverage Invocation : 1

Not Coverage Invocation List

Node Name	Method	Condition	Branch
call_register_noPri	getRegisterQueueID	\$inputVariable.payload/client:priority_id != ""	False
get_registQueue	getRegisterQueueID	\$inputVariable.payload/client:sso_id != ""	False
getRegisQueue	getRegisterQueueID	\$inputVariable.payload/client:hn != ""	False

รูปที่ 5-10 รายงานสรุปผลการทดสอบสำหรับกรณีศึกษาที่ 1

CHULALONGKORN UNIVERSITY

จากรูปที่ 5-9 ผู้ทดสอบสามารถดูรายการข้อมูลการทดสอบได้โดยคลิกปุ่ม “Print” เครื่องมือจะแสดงรายละเอียดข้อมูลนำเข้าและข้อมูลส่งออกที่รับส่งระหว่างเว็บเซอร์วิสให้กับผู้ทดสอบ ในรูปแบบของรายงาน ดังรูปที่ 5-11

Test Details Report				
BPEL ID : 286		Project Name : OPDQueueProject		Print Date Time : 22May2018 20:51:27
Node Name	Web Service	Method	Input	Output
Receive1	bpelopdqueueflow_client	process	Message time : 08 Apr 2018 20:39:31 ----- process hn = 1100002 sso_id = SSO department_id = 1 priority_id = 1	Message time :
getPatientData	SOAPPatient	getPatient	Message time : 08 Apr 2018 20:39:31 ----- getPatient hn = 1100002	Message time : 08 Apr 2018 20:39:32 ----- getPatientResponse -getPatientResult completeStatus = 1 dateOfBirth = 1969-04-02T00:00:00 fullname = Mr.PatientA LastNameA gender = M hn = 1100002
getSSO	SOAPPatient	checkPatientRight	Message time : 08 Apr 2018 20:39:32 ----- checkPatientRight	Message time : 08 Apr 2018 20:39:32 ----- checkPatientRightResponse

รูปที่ 5-11 รายงานผลการทดสอบสำหรับกรณีศึกษาที่ 1

จากผลการทดสอบข้างต้น เมื่อผู้ทดสอบนำกรณีทดสอบที่เป็นผู้ป่วยใหม่มาใช้ทดสอบผ่านเครื่องมือปีเพลเอนจิน และกลับมาตรวจสอบผลการทดสอบที่เครื่องมือ จะพบว่าฟังก์ชันเว็บเซอร์วิส “getRegisterQueueID” ได้ถูกทดสอบ ซึ่งทำให้เว็บเซอร์วิสที่อยู่ภายใต้ดับเบิ้ลยูเอส-ปีเพลนี้ ถูกทดสอบทั้งหมด ทำให้ไม่พบรายการเว็บเซอร์วิสที่รอทดสอบในตาราง “To be test Operation” ดังรูปที่ 5-12 และผู้ทดสอบสามารถดูรายงานสรุปผลการทดสอบของเครื่องมือได้ดังรูปที่ 5-13 ซึ่งจากรายงานนี้ จะสามารถบอกข้อมูลต่างๆ ดังนี้ ชื่อโครงการดับเบิ้ลยูเอส-ปีเพล จำนวนครั้งที่มีการทดสอบ จำนวนครั้งที่สร้างกรณีทดสอบเพิ่มเติมจากเครื่องมือ จำนวนเว็บเซอร์วิสที่ไม่ครอบคลุม และรายละเอียดของเว็บเซอร์วิสที่ไม่ครอบคลุม ของดับเบิ้ลยูเอส-ปีเพล นี้

History :: BPEL Task info		
BPEL ID :: 286		
Project Name :: OPDQueueProject		
To be test Operation :: (0)		
NodeName	Web Service	Method

รูปที่ 5-12 เว็บเซอร์วิสภายใต้ดับเบิ้ลยูเอส-ปีเพล ของกรณีศึกษาที่ 1 ถูกทดสอบทั้งหมด

Test Summary Report		Print Date Time : 22May2018 20:53.11	
BPEL ID : 286	Project Name : OPDQueueProject		
Total Test : 2 Times			
Total Generate Test Case: 0 Times			
Total Not Coverage Invocation : 0			
Not Coverage Invocation List			
Node Name	Method	Condition	Branch

รูปที่ 5-13 รายงานสรุปผลการทดสอบหลังจากนำกรณีทดสอบใหม่ไปใช้งานกรณีศึกษาที่ 1

5.1.2. กรณีศึกษาที่ 2 ระบบค้นหาเที่ยวบิน ที่พัก และรถยนต์

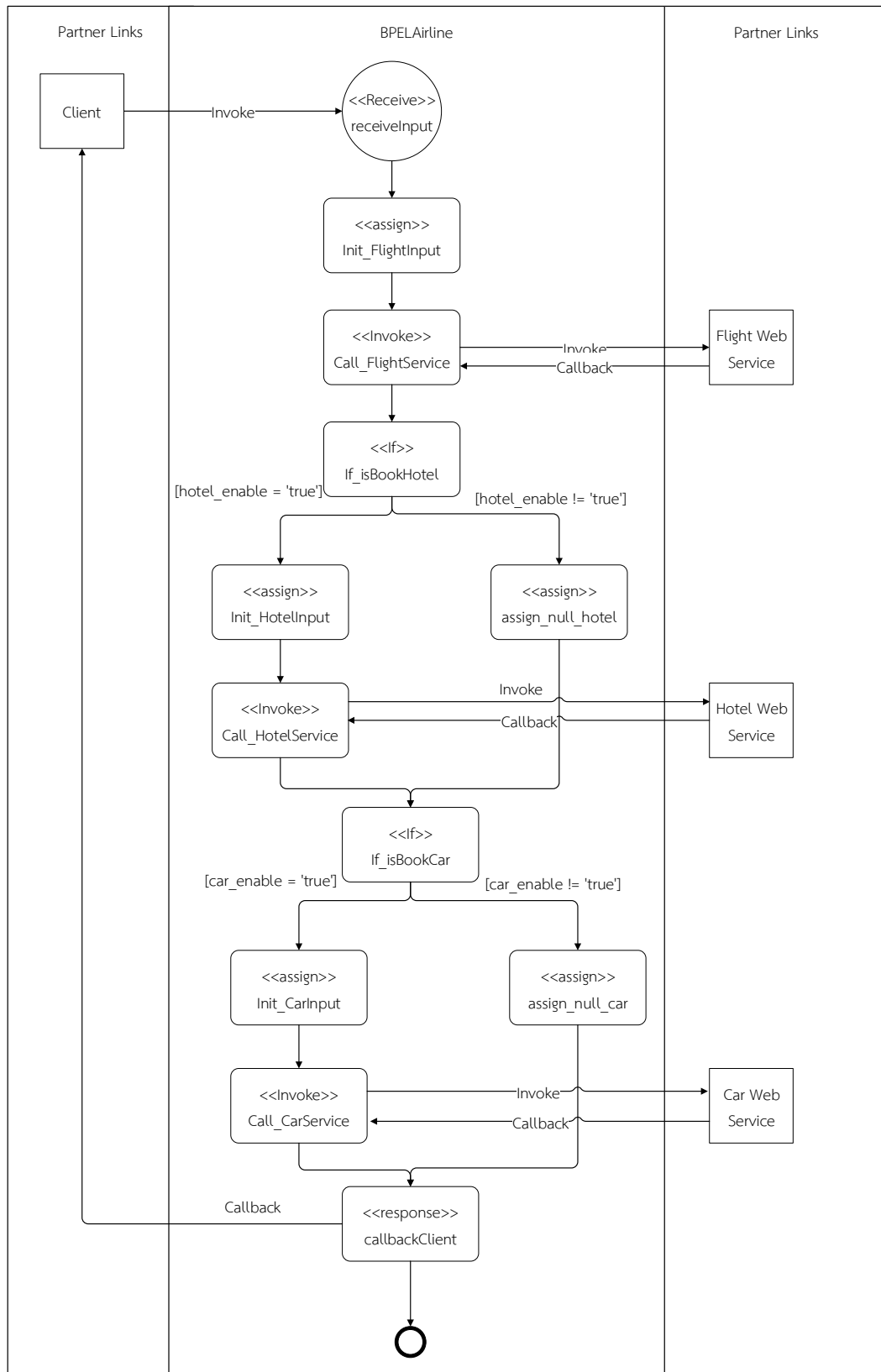
ระบบค้นหาเที่ยวบิน ที่พัก และรถยนต์ มีขั้นตอนการทำงานเริ่มจากลูกค้าระบุข้อมูลเที่ยวบิน ข้อมูลที่พัก และข้อมูลการจองรถยนต์ โดยรายละเอียดของข้อมูลที่ต้องระบุมีดังนี้

- ข้อมูลเที่ยวบิน ประกอบด้วย สถานที่ขึ้นเครื่อง สถานที่ปลายทาง วันที่เดินทาง วันที่กลับ และจำนวนผู้โดยสาร
- ข้อมูลการจองโรงแรม ประกอบด้วย ละแวกของโรงแรม วันที่เข้าพัก วันที่ออก และจำนวนห้องที่ต้องการจอง
- ข้อมูลการจองรถยนต์ ประกอบด้วย วันที่เริ่มเช่ารถ วันที่สิ้นสุดการเช่า และจำนวนรถที่ต้องการเช่า

โดยระบบจะค้นหาข้อมูลเที่ยวบิน ข้อมูลโรงแรม และข้อมูลรถยนต์ และนำข้อมูลที่ค้นหาได้มาแสดงให้กับลูกค้า

ในกรณีศึกษาที่ผู้วิจัยต้องการทดสอบการทำงานของเครื่องมือ ว่าสามารถสร้างกรณีทดสอบได้ครอบคลุมทุกการเรียกใช้งานเว็บเซอร์วิสหรือไม่ และสามารถรับและแสดงข้อมูลส่งออกที่อยู่ในรูปแบบรายการได้หรือไม่ โดยภายในกรณีศึกษาจะประกอบด้วยโหนดเงื่อนไข และโหนดการเรียกใช้งานเว็บเซอร์วิสต่างๆ

ผู้วิจัยได้นำกรณีศึกษาดังกล่าวมาสร้างเป็นแผนภาพดับเบิลยูเอส-พีเพิล โดยมีรายละเอียดดังรูปที่ 5-14 และมีการติดต่อกับเว็บเซอร์วิสต่างๆ ดังตารางที่ 5-2



รูปที่ 5-14 แผนภาพดับเบิลยูเอส-บีเพลสำหรับกรณีศึกษาที่ 2

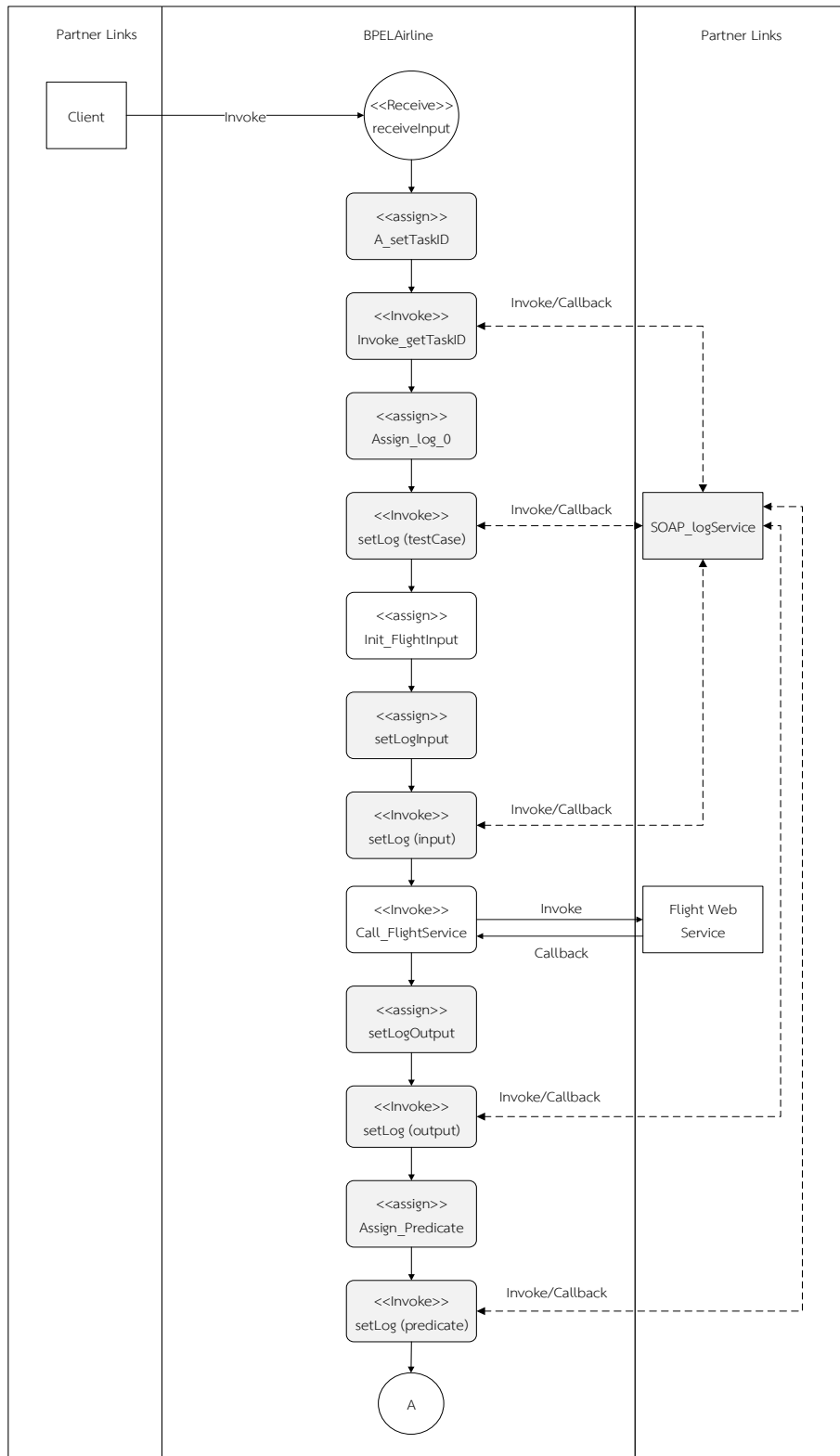
ตารางที่ 5-2 รายการเว็บเซอร์วิสที่เกี่ยวข้องกับแผนภาพบีเพล

ชื่อพาร์เทนอร์ลิงก์	ชื่อฟังก์ชันเว็บเซอร์วิส
FlightService	getFlight
HotelService	getHotel
CarService	getCar

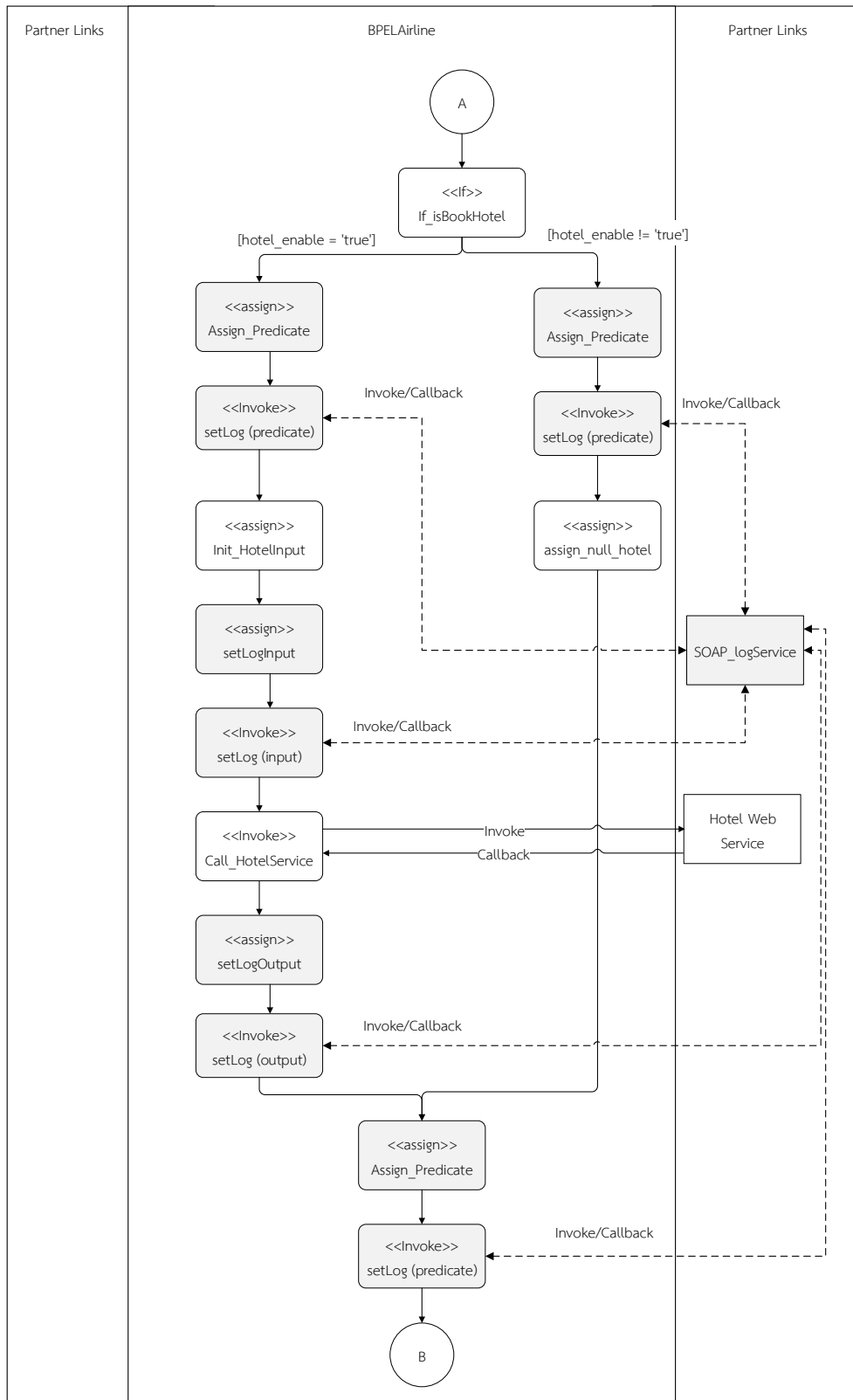
จากตารางที่ 5-2 พบว่าในระดับเบิลยูเอส-บีเพลนี้มีการเรียกใช้งานฟังก์ชันเว็บเซอร์วิสทั้งหมด 3 ฟังก์ชัน ซึ่งในการทดสอบผู้ทดสอบจำเป็นต้องทดสอบการทำงานให้ครอบคลุมฟังก์ชันเว็บเซอร์วิสทั้ง 3 ฟังก์ชันนี้ โดยวิธีในการทดสอบโครงการดับเบิลยูเอส-บีเพล ในกรณีศึกษาที่ผู้วิจัยจะใช้การทดสอบโดยนำโครงการมาแทรกรหัสต้นทาง จากนั้นจะนำโครงการมาติดตั้งผ่านบีเพลแอนจิน และนำกรณีทดสอบที่มีอยู่ซึ่งไม่ครอบคลุมทุกการเรียกใช้งานเว็บเซอร์วิสมาทดสอบผ่านเครื่องมือบีเพล-แอนจิน และจะทำการสร้างกรณีทดสอบเพิ่มเติมโดยใช้เครื่องมือเพื่อให้ครอบคลุมทุกเว็บเซอร์วิส จากนั้นจะนำกรณีทดสอบใหม่ที่ได้ไปทดสอบผ่านเครื่องมือบีเพลแอนจิน และกลับมาดูผลการทดสอบที่เครื่องมือ

เมื่อผู้ทดสอบนำโครงการดับเบิลยูเอส-บีเพล มาแทรกรหัสต้นทางผ่านเครื่องมือแล้ว ผู้ทดสอบจะได้แผนภาพดับเบิลยูเอส-บีเพล ใหม่ที่ถูกแทรกหัสต้นทาง ดังรูปที่ 5-15 ถึง 5-17

หลังจากที่ผู้ทดสอบทำการแทรกหัสต้นทางที่โครงการดับเบิลยูเอส-บีเพลสำเร็จแล้ว จะนำโครงการที่ถูกแทรกหัสต้นทางนั้นไปติดตั้งผ่านเครื่องมือบีเพลแอนจิน และนำกรณีทดสอบที่มีอยู่มาใช้ทดสอบ ซึ่งในกรณีทดสอบที่นำมาใช้จะต้องเรียกใช้งานฟังก์ชันเว็บเซอร์วิสจำนวน 2 ฟังก์ชัน คือ “getFlight” และ “getHotel” โดยจากการทดสอบ กรณีทดสอบดังกล่าวพบฟังก์ชันเว็บเซอร์วิสทั้ง 2 ฟังก์ชันถูกทดสอบ และพบว่ายังมีฟังก์ชันเว็บเซอร์วิส ที่ไม่ถูกทดสอบอีก 1 ฟังก์ชัน คือ “getCar” ดังรูปที่ 5-18 และผู้ทดสอบสามารถดูผลการทดสอบจากกรณีทดสอบที่นำมาทดสอบได้ดังรูปที่ 5-19 โดยเครื่องมือจะรองรับการรับส่งข้อมูลที่อยู่ในรูปแบบรายการของรายการโรงแรมที่ได้รับจากเว็บเซอร์วิส

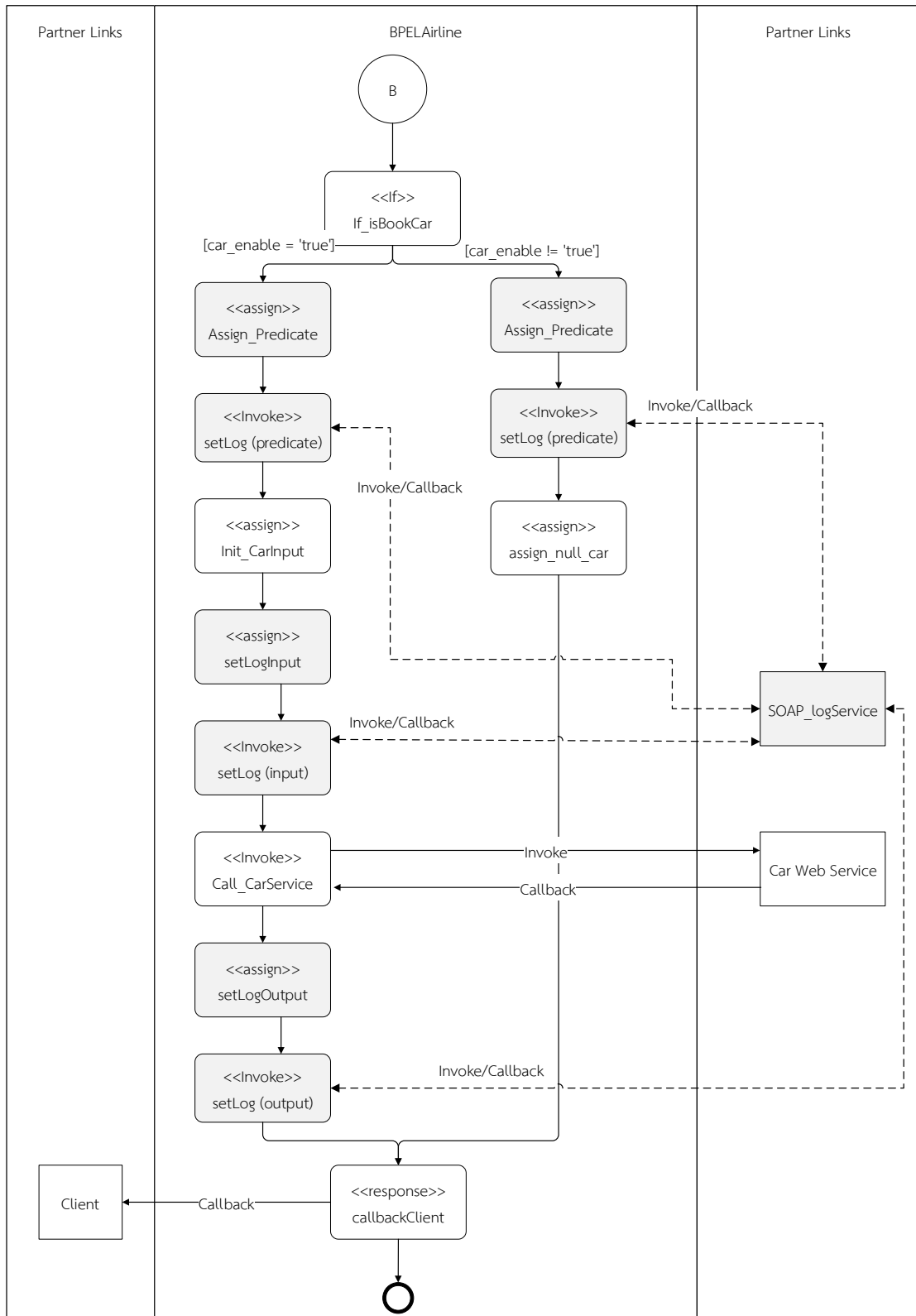


รูปที่ 5-15 ไฟล์ดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสสำหรับกรณีศึกษาที่ 2 ในขั้นตอนการเรียกข้อมูล
เที่ยวบิน



รูปที่ 5-16 ไฟล์ดับเบิลยูเอส-บีเพลที่ถูกแทรกสำหรับกรณีศึกษาที่ 2 ในขั้นตอนการเรียกข้อมูลที่

พัก



รูปที่ 5-17 ไฟล์ดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสสำหรับกรณีศึกษาที่ 2 ในขั้นตอนการเรียกข้อมูลรถยนต์

History :: BPEL Task info

BPEL ID :: 287
Project Name :: projAirline

To be test Operation :: (1)

NodeName	Web Service	Method
Call_Car_service	CarService	getCar

Test History List ::

Test Date	Test ID
08 Apr 2018 23:25:59	218

Generate Additional Test case Open test case directory

View Summary Report

รูปที่ 5-18 เว็บไซต์วิชที่ไม่ถูกทดสอบหลังจากนำกรณีทดสอบที่มีอยู่มาทดสอบ

History :: Test History Report

BPEL ID :: 287 Test ID :: 218 Test Time :: 08 Apr 2018 23:25:59 Print

Project Name :: projAirline

NodeName	Web Service	Method
receiveInput	bpelairline_client	process
Call_FlightService	FlightService	getFlight
Call_HotelService	HotelService	getHotel
callbackClient	bpelairline_client	processResponse

Message Details ::

Input ::

Message time : 08 Apr 2018 23:25:59

```
-----
getHotel
-oHotel
hotelArea = Seoul
hotel_leaveDate = 2018-04-18
```

Actual output ::

```
hotel_Name = Hotel D
hotel_address = Address Hotel D
hotel_leaveDate = 2018-04-18
hotel_price = 5122.5
hotel_stayDate = 2018-04-08
total_room = 1
```

รูปที่ 5-19 ตัวอย่างการแสดงผลข้อมูลส่งออกที่อยู่ในรูปแบบรายการ

จากนั้นจะทดสอบการสร้างกรณีทดสอบเพิ่มเติม โดยให้ผู้ทดสอบทำการคลิกที่ปุ่ม “Generate Additional Test case” เมื่อเครื่องมือสร้างกรณีทดสอบสำเร็จจะมีข้อความแจ้งเตือน ดังรูปที่ 5-20 ซึ่งเครื่องมือจะสร้างกรณีทดสอบเพิ่มเติมในรูปแบบไฟล์ข้อความ ดังรูปที่ 5-21 เพื่อนำไปทดสอบกับเครื่องมือบีเพลเอนจิน ผ่านวิธีการอัปโหลดไฟล์กรณีทดสอบต่อไป ดังรูปที่ 5-22

History :: BPEL Task info

BPEL ID :: 287
Project Name :: projAirline

To be test Operation :: (1)

NodeName	Web Service	Method
Call_Car_service	CarService	getCar

Create testCase Complete

OK

Test History List ::

Test Date	Test ID
08 Apr 2018 23:25:59	218

Generate Additional Test case

Open test case directory

View Summary Report

รูปที่ 5-20 ข้อความเตือน การสร้างกรณีทดสอบสำเร็จ

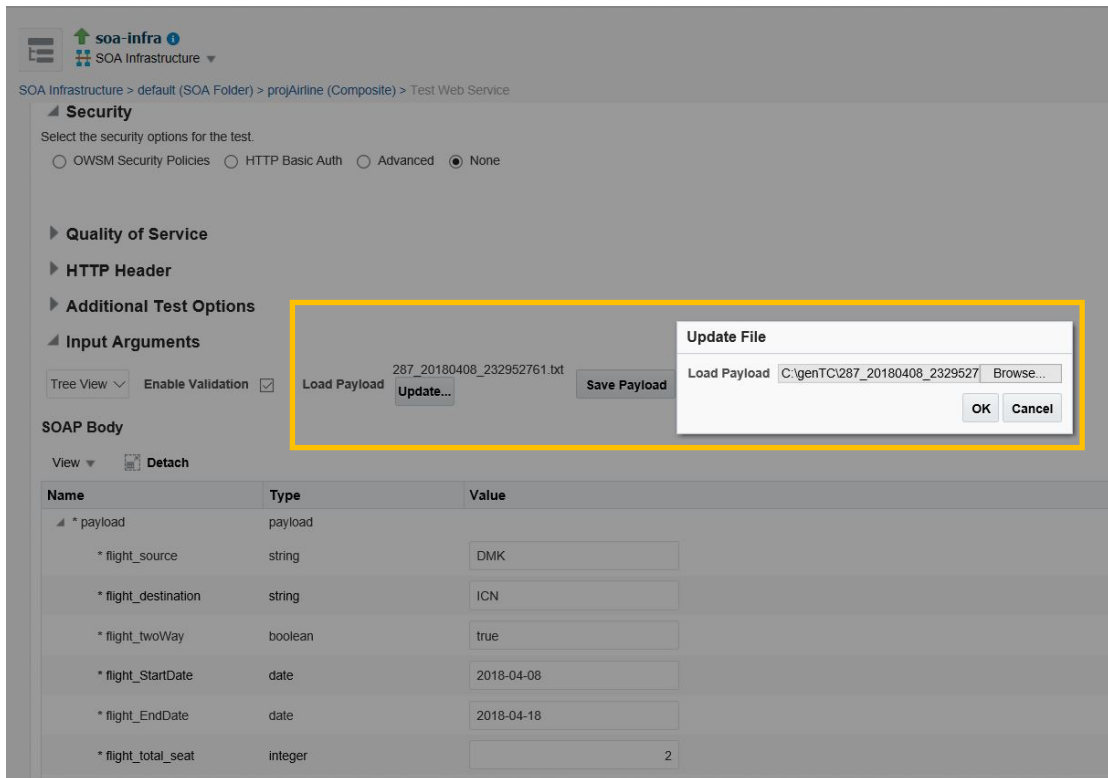
287_20180408_232952761 4/8/2018 11:29 PM Text Document 1 KB

287_20180408_232952761 - Notepad

File Edit Format View Help

```
<ns1:process xmlns:ns1="http://xmlns.oracle.com/BPELAirline/projAirline/BPELAirline"
xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <ns1:flight_source>DMK</ns1:flight_source>
  <ns1:flight_destination>ICN</ns1:flight_destination>
  <ns1:flight_twoWay>true</ns1:flight_twoWay>
  <ns1:flight_StartDate>2018-04-08</ns1:flight_StartDate>
  <ns1:flight_EndDate>2018-04-18</ns1:flight_EndDate>
  <ns1:flight_total_seat>2</ns1:flight_total_seat>
  <ns1:hotel_enable>true</ns1:hotel_enable>
  <ns1:hotel_area>Seoul</ns1:hotel_area>
  <ns1:hotel_stayDate>2018-04-08</ns1:hotel_stayDate>
  <ns1:hotel_leaveDate>2018-04-18</ns1:hotel_leaveDate>
  <ns1:total_totalRoom>1</ns1:total_totalRoom>
  <ns1:car_enable>true</ns1:car_enable>
  <ns1:car_startDate>2018-04-08</ns1:car_startDate>
  <ns1:car_endDate>2018-04-08</ns1:car_endDate>
  <ns1:total_car>0</ns1:total_car>
</ns1:process>
```

รูปที่ 5-21 ไฟล์กรณีทดสอบใหม่ที่ได้



รูปที่ 5-22 อัปเดตกรณีทดสอบใหม่ที่ได้นับเครื่องมือปีเพเลนจีน

หลังจากนำกรณีทดสอบใหม่ที่ได้นำมาทดสอบผ่านเครื่องมือปีเพเลนจีน จะทำให้ฟังก์ชันเว็บเซอร์วิส “getCar” ถูกทดสอบ ซึ่งทำให้เว็บเซอร์วิสที่อยู่ภายใต้ดับเบิลยูเอส-ปีเพเลนนี้ ถูกทดสอบทั้งหมด ทำให้ไม่พบรายการเว็บเซอร์วิสที่รอทดสอบในตาราง “To be test Operation” ดังรูปที่ 5-23 และผู้ทดสอบสามารถดูรายงานสรุปผลการทดสอบของเครื่องมือได้ ดังรูปที่ 5-24



รูปที่ 5-23 เว็บเซอร์วิสภายใต้ดับเบิลยูเอส-ปีเพเลน ของกรณีศึกษาที่ 2 ถูกทดสอบทั้งหมด

Test Summary Report		Print Date Time : 22May2018 21:02:13	
BPEL ID : 287	Project Name : projAirline		
Total Test : 2 Times			
Total Generate Test Case: 1 Times			
Total Not Coverage Invocation : 0			
Not Coverage Invocation List			
Node Name	Method	Condition	Branch

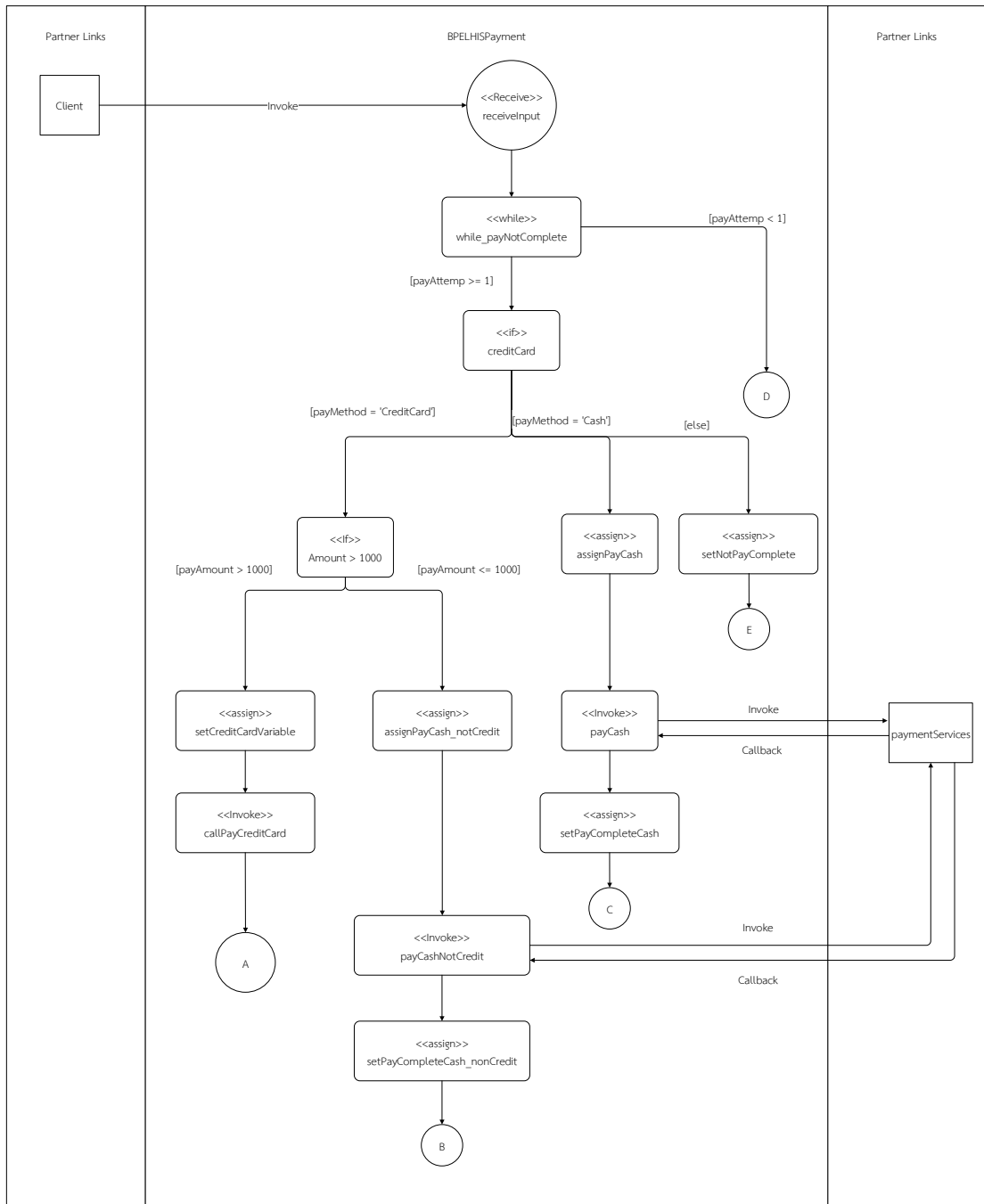
รูปที่ 5-24 รายงานสรุปผลการทดสอบหลังจากนำกรณีทดสอบใหม่ไปใช้งานกรณีศึกษาที่ 2

จากกรณีทดสอบนี้พบว่า เครื่องมือสามารถสร้างกรณีทดสอบเพิ่มเติมเพื่อให้ครอบคลุมการเรียกใช้งานเว็บเซอร์วิสที่อยู่ภายในกรณีศึกษาได้ นอกจากนี้ยังสามารถรับข้อมูลส่งออกที่อยู่ในรูปแบบรายการและแสดงค่าข้อมูลส่งออกได้อย่างถูกต้อง

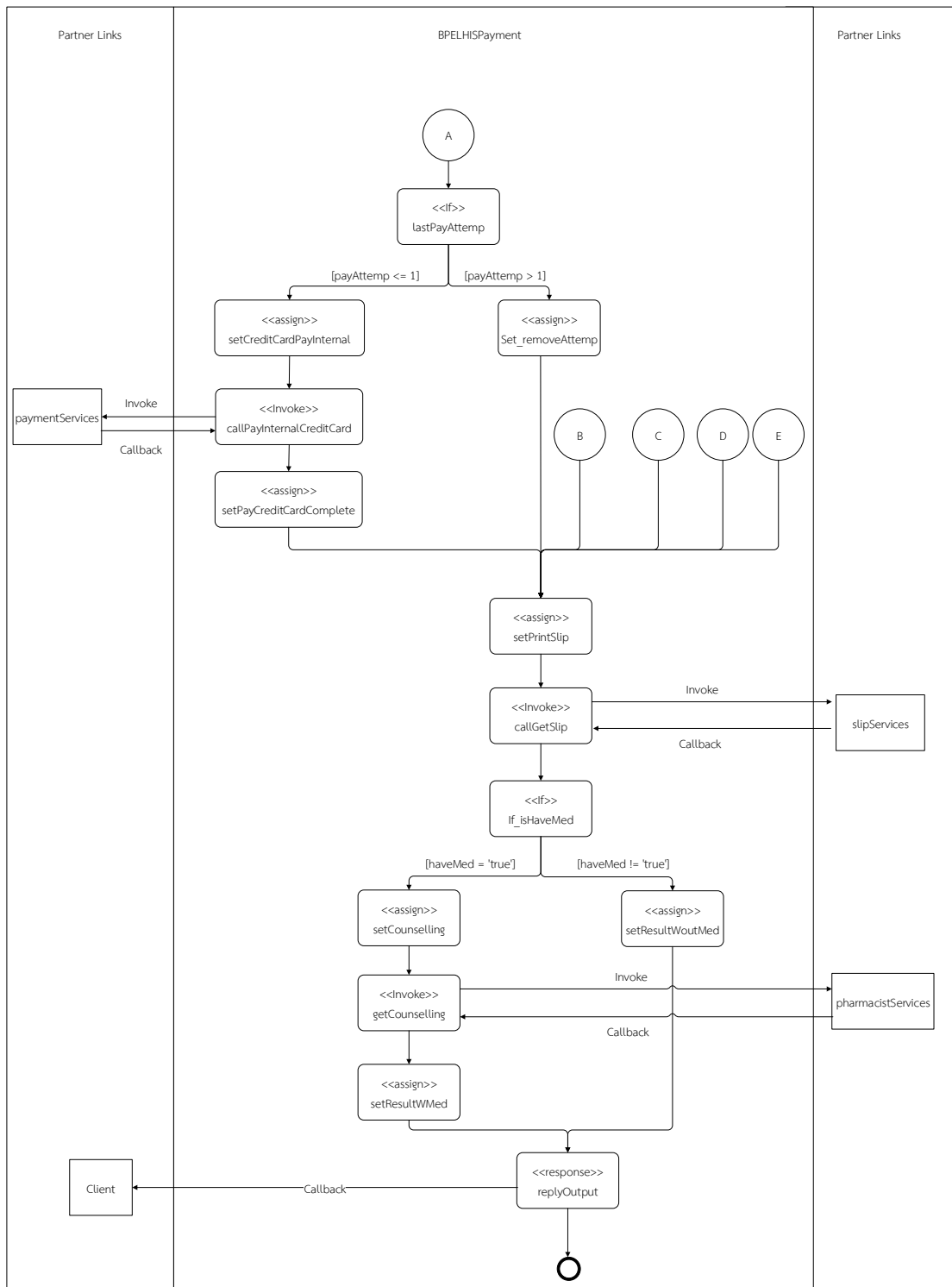
5.1.3. กรณีศึกษาที่ 3 ระบบชำระเงินผู้ป่วยนอก

ผู้วิจัยได้นำตัวอย่างกรณีศึกษาระบบชำระเงินผู้ป่วยนอกมาเพื่อทดสอบการตรวจจัดการเรียกใช้เว็บเซอร์วิสที่มีการเรียกใช้งานเชิงวนซ้ำ โดยกรณีศึกษานี้มีขั้นตอนการทำงานเริ่มจากผู้ป่วยกำลังชำระเงินแก่การเงินในโรงพยาบาล โดยในการชำระเงินผู้ป่วยสามารถเลือกช่องทางการชำระเงินด้วยเงินสด หรือบัตรเครดิตได้ ในการชำระเงินด้วยบัตรเครดิตนั้นจะต้องมียอดค่าใช้จ่ายมากกว่า 1000 บาท โดยในการชำระค่าบริการด้วยบัตรเครดิตระบบจะวนตัดบัตรเครดิตจนกว่าจะครบจำนวนรอบที่ระบุไว้ จากนั้นเมื่อชำระเงินเสร็จสิ้น เจ้าหน้าที่จะพิมพ์ใบเสร็จให้กับผู้ป่วย และในกรณีที่ผู้ป่วยต้องรับยาที่ห้องจ่ายยา จะต้องไปยังจุดรับยาที่ทางโรงพยาบาลระบุไว้ โดยเภสัชกรจะพิมพ์ใบคำแนะนำการใช้ยาและจ่ายยาให้กับผู้ป่วย

จากขั้นตอนการทำงานข้างต้นสามารถออกแบบเป็นการทำงานของดับเบิลยูเอส-บีเพลได้ดังรูปที่ 5-25 และ 5-26 และมีตารางติดต่อกับเว็บเซอร์วิสดังตารางที่ 5-3



รูปที่ 5-25 แผนภาพดับเบิลยูเอส-บีเพลสำหรับกรณีศึกษาที่ 3 ในขั้นตอนตรวจสอบวิธีการชำระเงิน



รูปที่ 5-26 แผนภาพดับเบิลยูเอส-บีเพลสำหรับกรณีศึกษาระบบที่ 3 ในขั้นตอนชำระเงินและรับยาผู้ป่วย

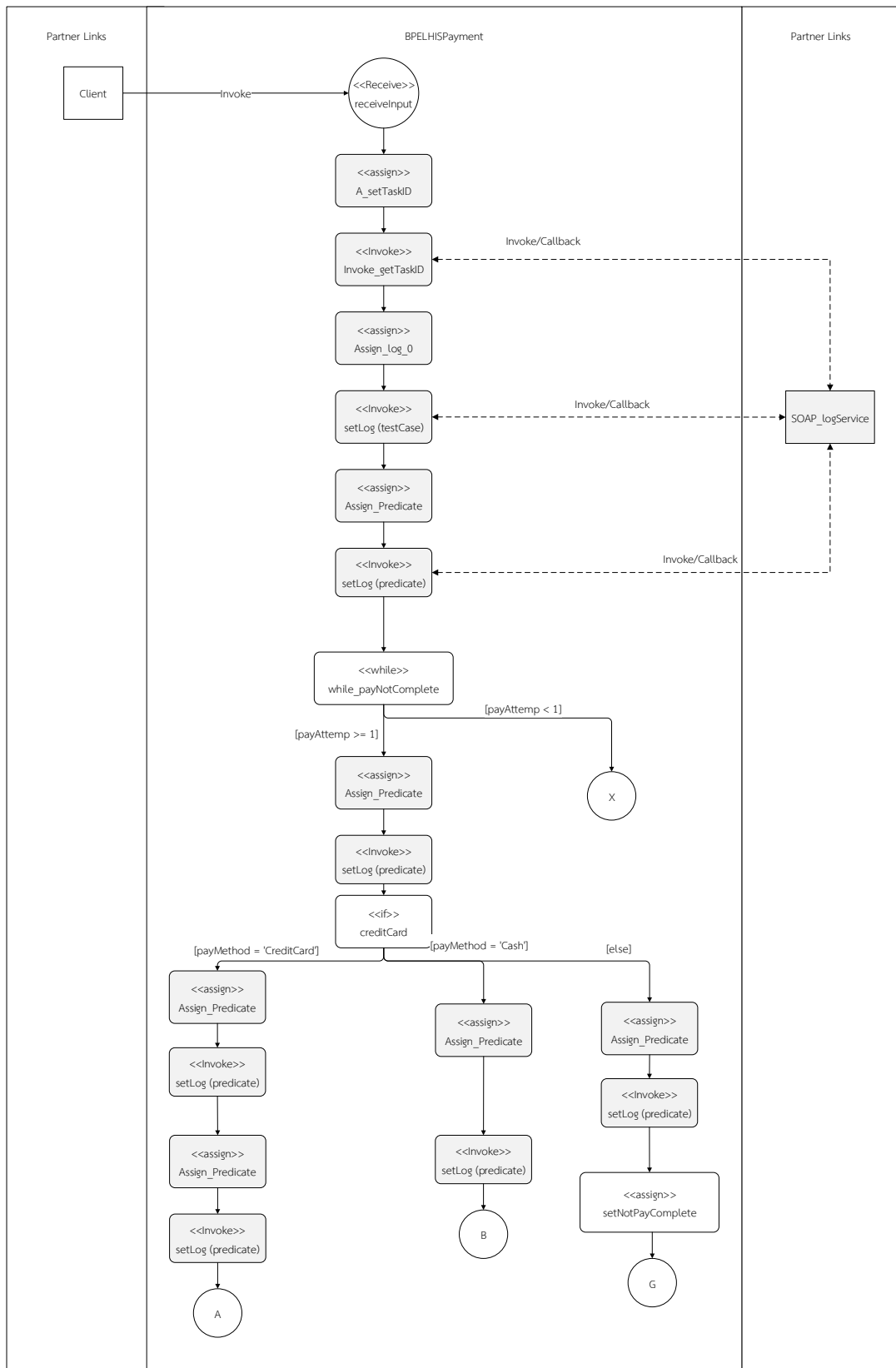
ตารางที่ 5-3 รายการเว็บเซอร์วิสที่เกี่ยวข้องกับแผนภาพบีเพล

ชื่อพาร์ทเนอร์ลิงก์	ชื่อฟังก์ชันเว็บเซอร์วิส
externalCreditCardServices	payCreditCard
paymentServices	submitPayCreditCard
paymentServices	submitPayCash
slipServices	getPrintData
pharmacistServices	getCounsellingSheet

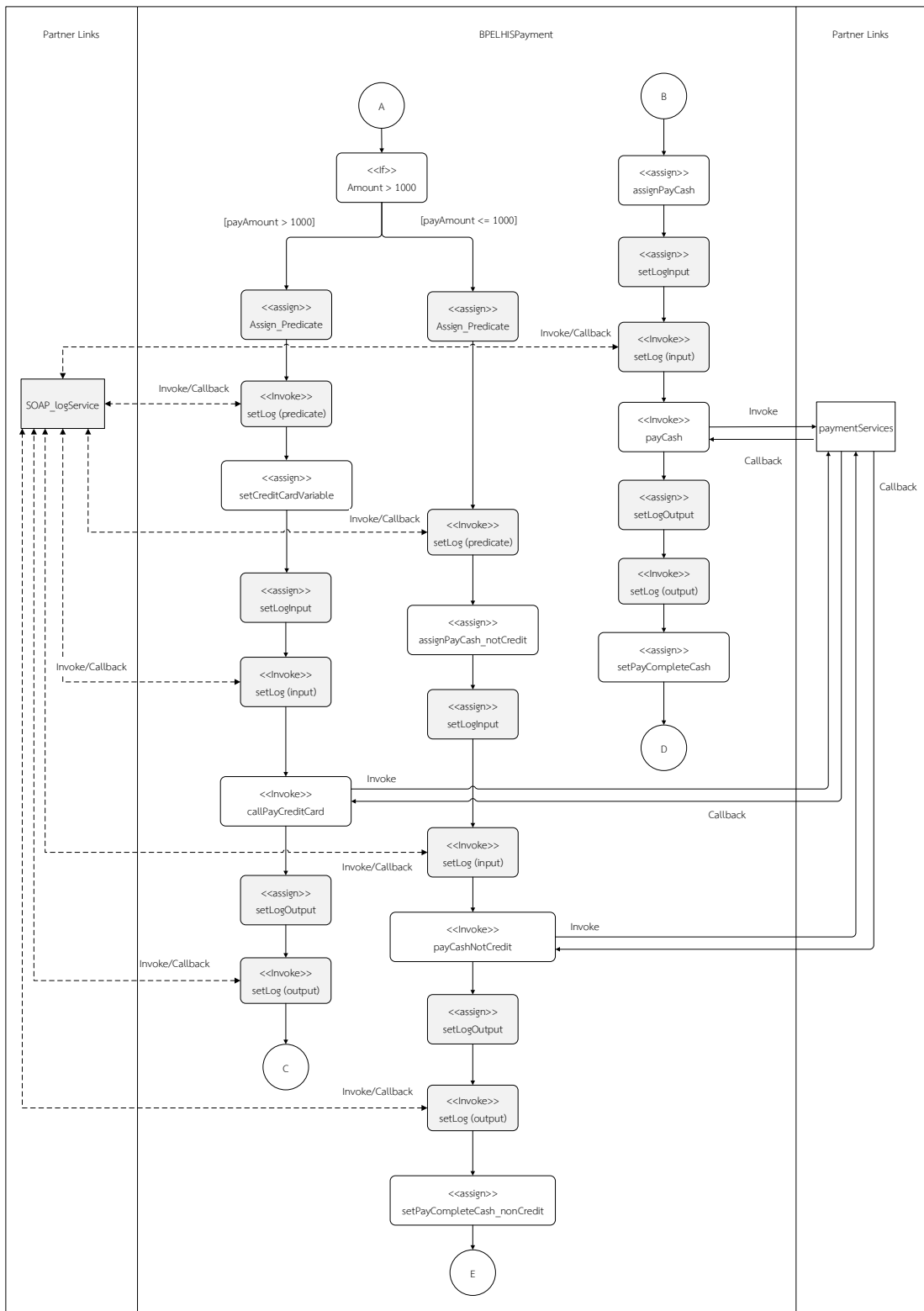
จากตารางที่ 5-3 พบว่าในระดับเบิลยูเอส-บีเพลนี้มีโหนดที่การเรียกใช้งานฟังก์ชันเว็บเซอร์วิสทั้งหมด 5 ฟังก์ชัน ซึ่งในการทดสอบผู้ทดสอบจำเป็นต้องทดสอบการทำงานให้ครอบคลุมเว็บเซอร์วิสทั้ง 5 ฟังก์ชันนี้ โดยวิธีในการทดสอบโครงการดับเบิลยูเอส-บีเพล ในกรณีศึกษาที่ผู้วิจัยจะทำการทดสอบโดยนำโครงการมาแทรกรหัสต้นทาง จากนั้นจะนำโครงการมาติดตั้งผ่านบีเพลเอนจิน และทำการสร้างกรณีทดสอบเพิ่มเติมโดยใช้เครื่องมือเพื่อให้ครอบคลุมทุกกรณีทดสอบ จากนั้นนำกรณีทดสอบใหม่ทั้งหมดที่ได้ไปทดสอบผ่านบีเพลเอนจิน และกลับมาดูผลการทดสอบที่เครื่องมือ

จากการทดสอบพบว่าเมื่อผู้ทดสอบนำโครงการดับเบิลยูเอส-บีเพล มาแทรกรหัสต้นทาง เครื่องมือสามารถแทรกรหัสต้นทางเพื่อจัดเก็บประวัติได้อย่างถูกต้อง ดังรูปที่ 5-27 ถึง 5-30

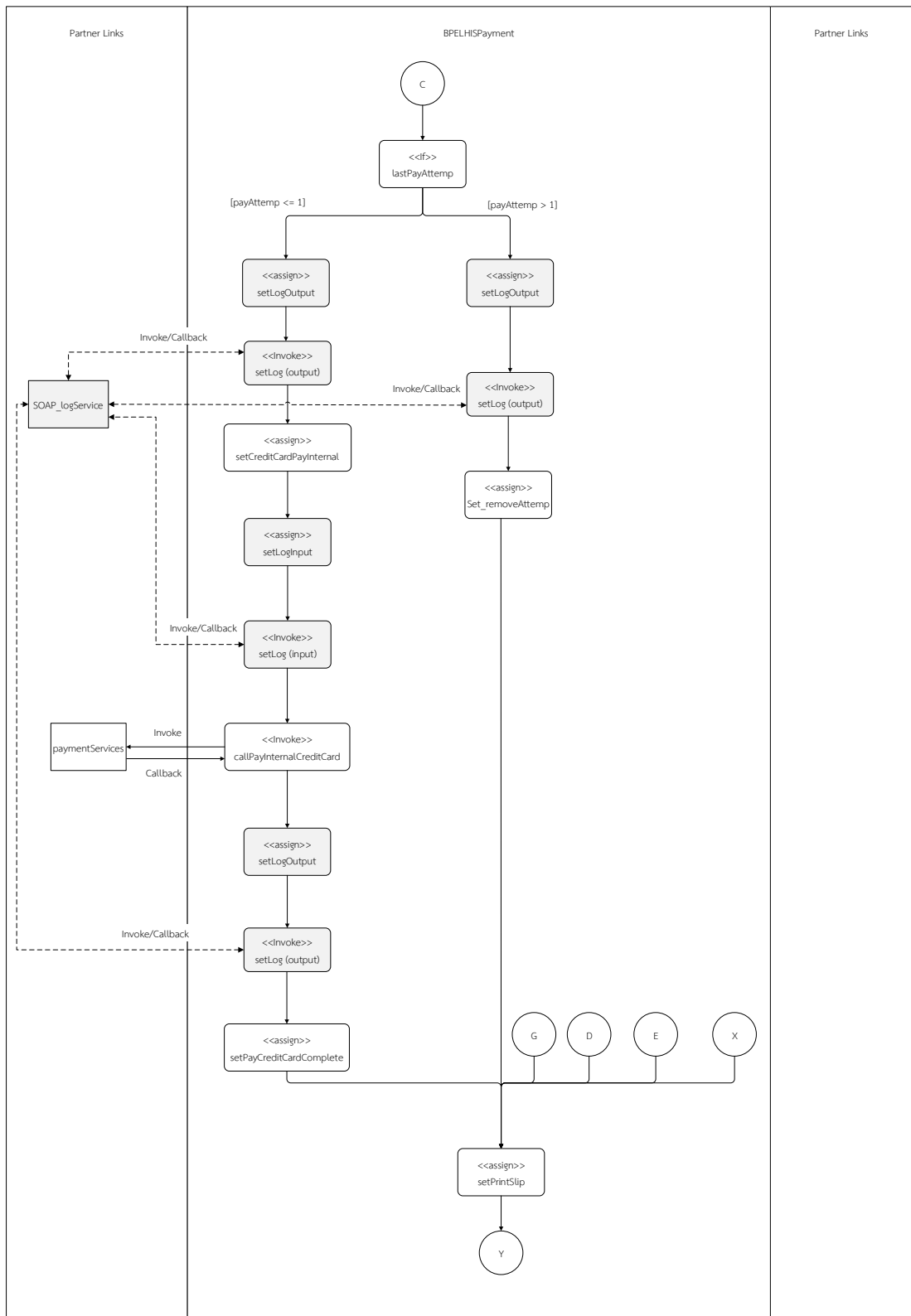
เมื่อผู้ทดสอบทำการสร้างกรณีทดสอบเพิ่มเติมโดยใช้เครื่องมือแล้ว จากนั้นจะนำโครงการดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางแล้ว ไปติดตั้งและทดสอบผ่านเครื่องมือบีเพลเอนจิน โดยใช้กรณีทดสอบที่ถูกสร้างจากเครื่องมือ พบว่ากรณีทดสอบที่นำมาใช้ครอบคลุมการเรียกใช้งานของเว็บเซอร์วิสทั้ง 3 เว็บเซอร์วิสที่อยู่ภายใต้โครงการดับเบิลยูเอส-บีเพล ดังรูปที่ 5-31 และผู้ทดสอบสามารถดูข้อมูลนำเข้า ข้อมูลส่งออกในแต่ละรอบของการทำงานแบบวนซ้ำ และสามารถดูรายงานสรุปการทดสอบได้ดังรูปที่ 5-32 และ 5-33 ตามลำดับ



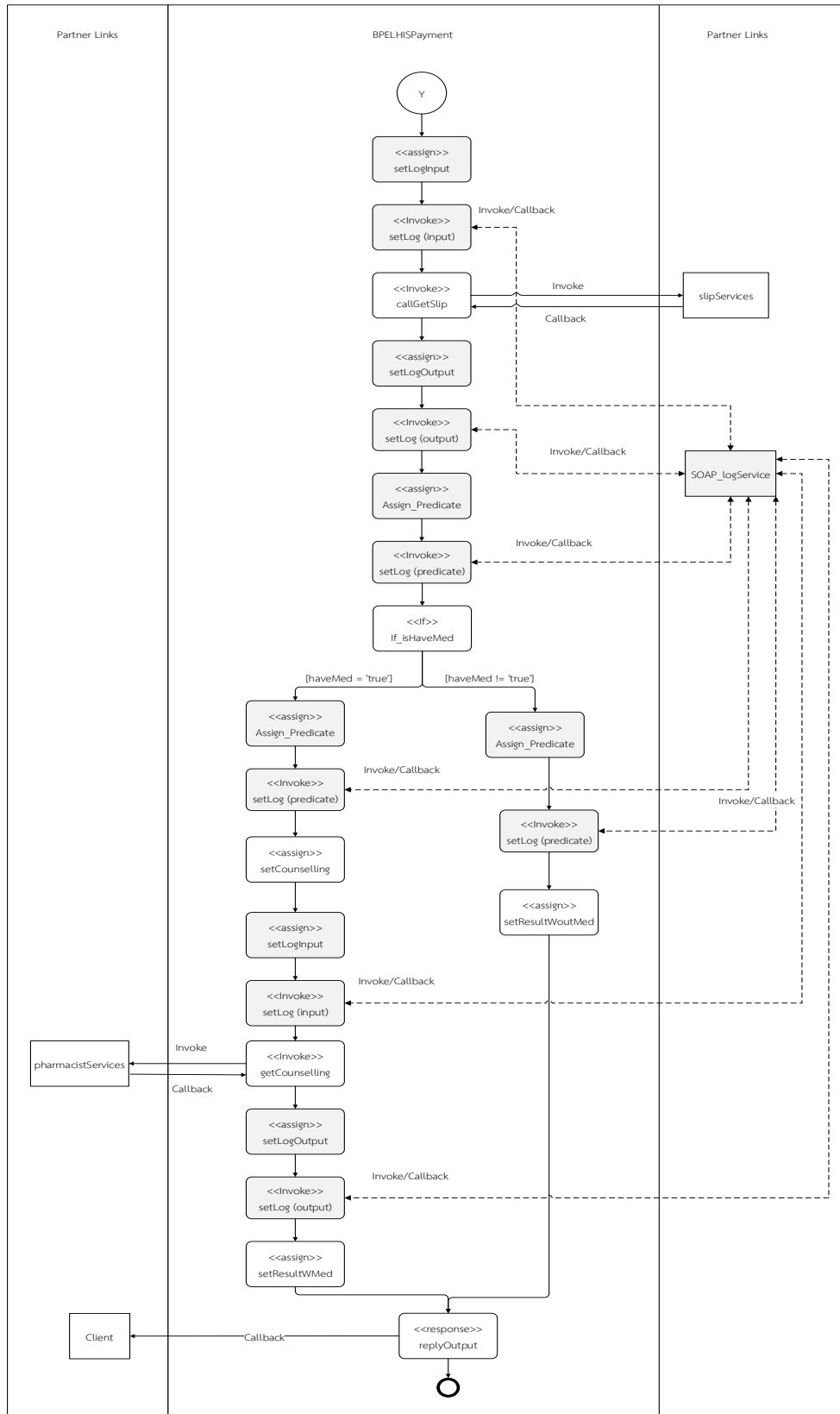
รูปที่ 5-27 ไฟล์ดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทาง สำหรับกรณีศึกษาที่ 3 ในขั้นตอนการตรวจสอบวิธีการชำระเงิน



รูปที่ 5-28 ไฟล์ดับเบิลยูเอส-บีเพลที่ถูกแทรกห้ต้นทาง สำหรับกรณีศึกษาที่ 3 ในขั้นตอนการชำระ
เงินผู้ป่วย



รูปที่ 5-29 ไฟล์ดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทาง สำหรับกรณีศึกษาที่ 3 ในขั้นตอนการชำระ
เงินและพิมพ์ใบเสร็จ



รูปที่ 5-30 ไฟล์ดับเบิ้ลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทาง สำหรับกรณีศึกษาที่ 3 ในขั้นตอนการรับยา

ผู้ป่วย

History :: BPEL Task info

BPEL ID :: 285
Project Name :: Project_HISPayment

To be test Operation :: (0)

NodeName	Web Service	Method

Test History List ::

Test Date	Test ID
04 Mar 2018 00:45:06	207
04 Mar 2018 00:45:23	208
04 Mar 2018 00:45:37	209
04 Mar 2018 00:45:59	210
04 Mar 2018 00:46:11	211
04 Mar 2018 00:47:44	212

Generate Additional Test case Open test case directory

View Summary Report

รูปที่ 5-31 ผลการทดสอบโดยใช้กรณีทดสอบใหม่ที่ถูกสร้างขึ้นจากเครื่องมือ

History :: Test History Report

BPEL ID :: 285 Test ID :: 212 Test Time :: 04 Mar 2018 00:47:44 Print

Project Name :: Project_HISPayment

NodeName	Web Service	Method
receiveInput	bpelhispayment_client	process
callPayCreditCard	externalCreditCardServices	payCreditCard
callPayCreditCard	externalCreditCardServices	payCreditCard
callPayCreditCard	externalCreditCardServices	payCreditCard
callPayInternalCreditCard	paymentServices	submitPayCreditCard
callGetSlip	slipServices	getPrintData
replyOutput	bpelhispayment_client	process

Message Details ::

Input ::

Message time : 04 Mar 2018 00:47:45

payCreditCard
transactionID = T12345
amount = 920116.0

Actual output ::

Message time : 04 Mar 2018 00:47:45

payCreditCardResponse
payCreditCardResult = true

รูปที่ 5-32 ตัวอย่างข้อมูลนำเข้าและข้อมูลส่งออกที่ในแต่ละรอบของการทำงานแบบวนซ้ำ

จากกรณีทดสอบนี้พบว่า เครื่องมือสามารถสร้างกรณีทดสอบให้ครอบคลุมการเรียกใช้งานเว็บเซอร์วิสได้อย่างครอบคลุม และสามารถดูประวัติการรับส่งข้อมูลนำเข้าและส่งออกของแต่ละรอบในการเรียกใช้งานได้

Test Summary Report		Print Date Time : 22May2018 21:09.44	
BPEL ID : 285		Project Name : Project_HISPayment	
Total Test : 6 Times			
Total Generate Test Case: 6 Times			
Total Not Coverage Invocation : 0			
Not Coverage Invocation List			
Node Name	Method	Condition	Branch

รูปที่ 5-33 ตัวอย่างรายงานสรุปผลการทดสอบ



บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

6.1. สรุปผลการวิจัย

วิทยานิพนธ์นี้ได้เสนอวิธีการและพัฒนาเครื่องมือตรวจจับการเรียกใช้เว็บเซอร์วิสสำหรับทดสอบระดับเบิยูเอส-พีเพิล เพื่อตรวจจับข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิสและข้อมูลส่งออกที่ได้รับจากเว็บเซอร์วิส และตรวจสอบความครอบคลุมของการทดสอบเว็บเซอร์วิสได้ โดยเครื่องมือสามารถสามารถให้ผู้ทดสอบอัปโหลดไฟล์ระดับเบิยูเอส-พีเพิล ไฟล์เอกซ์เอสดี ไฟล์คอมโพสิต ไฟล์ระดับเบิยูเอสดีแอล เพื่อนำมาสร้างเส้นทางการไหล สร้างกรณีทดสอบ และแทรกรหัสต้นทางได้ เมื่อผู้ทดสอบนำโครงการระดับเบิยูเอส-พีเพิล ที่ถูกแทรกรหัสต้นทางมาติดตั้งบนเครื่องแม่ข่ายแล้ว ผู้ทดสอบสามารถทดสอบระดับเบิยูเอส-พีเพิล นั้นได้ตามปกติโดยในขณะที่พีเพิลเอนจินทดสอบผ่านโหนดที่ถูกแทรกรหัสต้นทางพีเพิลเอนจิน จะส่งข้อมูลนำเข้าและข้อมูลส่งออกไปจัดเก็บในฐานข้อมูลของเครื่องมือผ่านทางเว็บเซอร์วิสที่เครื่องมือระบุไว้ได้ นอกจากนี้ในขณะที่ทดสอบผู้ทดสอบสามารถตรวจสอบการเรียกใช้งานเว็บเซอร์วิสผ่านทางเครื่องมือ และสามารถสร้างกรณีทดสอบเพิ่มเติมสำหรับเว็บเซอร์วิสที่ไม่ถูกทดสอบภายใต้ข้อจำกัดและขอบเขตที่ระบุไว้ได้ ซึ่งจากการทดสอบเครื่องมือโดยใช้กรณีศึกษาข้างต้นพบว่าเครื่องมือสามารถทำงานได้อย่างถูกต้อง โดยสามารถให้ช่วยอำนวยความสะดวกให้แก่ผู้ทดสอบดังนี้

1. ผู้ทดสอบสามารถติดตามข้อมูลนำเข้าที่ส่งไปยังเว็บเซอร์วิสและข้อมูลส่งออกที่ได้รับจากเว็บเซอร์วิสได้
2. ผู้ทดสอบทราบถึงผลการทดสอบว่ามีการทดสอบครอบคลุมทุกเว็บเซอร์วิสหรือไม่
3. ช่วยสร้างกรณีทดสอบเพิ่มเติมเบื้องต้น ให้กับผู้ทดสอบตามโหนดเงื่อนไขของเส้นทางการไหลของพีเพิล

6.2. ข้อจำกัดของงานวิจัย

ข้อจำกัดของเครื่องมือมีดังนี้

1. เครื่องมือรองรับการทำงานร่วมกับเครื่องมือพีเพิลเอนจิน ออราเคิล บีเพิล โพรเซสเมเนเจอร์ 12c เท่านั้น
2. เครื่องมือรองรับการทำงานร่วมกับโหนดระดับเบิยูเอส-พีเพิล ดังนี้

Receive, Response, Invoke, Assign, Sequence, If, Elseif, Else และ While

3. การสร้างกรณีทดสอบเพิ่มเติม สามารถสร้างได้ต่อเมื่อค่าตัวแปรที่อยู่ในโหนดเงื่อนไขเป็นค่าที่ไม่มีมีการเปลี่ยนแปลงภายในโครงการดับเบิลยูเอส-บีเฟล และต้องเป็นตัวแปรนำเข้าที่รับมาจากผู้ทดสอบ
4. ในการสร้างกรณีทดสอบ เครื่องมือรองรับคำสั่งในโหนดเงื่อนไข เฉพาะคำสั่งอย่างง่ายซึ่งประกอบด้วย $>$, $>=$, $<$, $<=$, $=$ และ $!=$
5. การแก้ไขข้อมูลเพื่อสร้างกรณีทดสอบเพิ่มเติมของข้อมูลประเภทพื้นฐานและประเภทซับซ้อน จะรองรับข้อมูลที่ประกอบด้วยข้อมูลประเภทพื้นฐานที่เป็น ตัวเลข จำนวนเต็ม ตัวเลขทศนิยม ตัวอักษร และ บูลีน
6. เครื่องมือจะเริ่มจัดเก็บผลการทดสอบและตรวจจบการเรียกใช้เว็บเซอร์วิสได้หลังจากนำโครงการดับเบิลยูเอส-บีเฟลที่ถูกแทรกหีสันทางผ่านเครื่องมือ มาติดตั้งและทดสอบผ่านบีเฟลเอนจิน
7. เครื่องมือรองรับโครงการดับเบิลยูเอส-บีเฟล ที่ภายในโครงการมีไฟล์ดับเบิลยูเอส-บีเฟลไฟล์เดียวเท่านั้น

6.3. ข้อเสนอแนะและแนวทางการดำเนินงานต่อ

1. ควรพัฒนาเครื่องมือให้รองรับกับบีเฟลเอนจิน อื่นๆ เพิ่มเติมเพื่อไม่ยึดติดกับเครื่องมือใดเครื่องมือหนึ่ง
2. ควรพัฒนาการทำงานร่วมกับโหนดอื่นๆ ของดับเบิลยูเอส-บีเฟล
3. ควรพัฒนาให้เครื่องมือรองรับคำสั่งในโหนดเงื่อนไขเพิ่มขึ้น
4. ควรพัฒนาเครื่องมือให้รองรับการสร้างกรณีทดสอบ ในข้อมูลประเภทอื่นๆ เพิ่มเติม
5. ควรพัฒนาเครื่องมือให้ติดต่อกับบีเฟลเอนจินได้ โดยไม่ได้สลับการทำงานระหว่างเครื่องมือและบีเฟลเอนจิน
6. ควรพัฒนาเครื่องมือให้รองรับการสร้างกรณีทดสอบเพิ่มเติม ให้ครอบคลุมทุกกิ่งของเส้นทางการไหลในไฟล์ดับเบิลยูเอส-บีเฟล

รายการอ้างอิง

- [1] D. Lübke, L. Singer, and A. Salnikow, "Calculating BPEL test coverage through instrumentation," presented at the Proceedings of the 2009 ICSE Workshop on Automation of Software Test, AST 2009, 2009.
- [2] B. Li, D. Qiu, S. Ji, and D. Wang, "Automatic test case selection and generation for regression testing of composite service based on extensible BPEL flow graph," presented at the IEEE International Conference on Software Maintenance, ICSM, 2010.
- [3] C.-A. Sun, Y. Zhao, L. Pan, H. Liu, and T. Chen, "Automated Testing of WS-BPEL Service Compositions: A Scenario-Oriented Approach," presented at the IEEE Transactions on Services Computing, 2015.
- [4] (2016-11-13). *Oracle Co. Ltd*, . Available: <https://www.oracle.com/index.html>
- [5] (2016- 11- 26). *OASIS Web Services Business Process Execution Language (WSBPEL) TC | OASIS*. Available: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel
- [6] D. Ings, A. Endpoints, V. Mehta, R. Mueller, O. Corporation, R. Rangaswamy, *et al.*, "WS-BPEL Extension for People Committee Specification," pp. 1-57, 2010.
- [7] O. W. S. B. P. E. L. (WSBPEL), D. Jordan, and A. Alves, "Web Services Business Process Execution Language Version 2.0," vol. 11, ed, 2007, pp. 1-264.
- [8] M. B. Juric and D. Weerasiri, *WS-BPEL 2.0 Beginner's Guide*, 2014.
- [9] O. D. Sheet, K. E. Y. Features, A. B. Engine, B. Designer, and B. Runtime, "ORACLE BPEL PROCESS MANAGER," ed, pp. 1-4.
- [10] (2016- 12- 19) . *XML Schema Tutorial*. Available: http://www.w3schools.com/XML/schema_intro.asp



ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาคผนวก ก
รายละเอียดยูสเคสของเครื่องมือ

ในภาคผนวก ก จะแสดงรายละเอียดยูสเคสแต่ละรายการ ดังต่อไปนี้

1. อัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-พีเพิล

ตารางที่ ก-1 คำอธิบายยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-พีเพิล

ชื่อยูสเคส	อัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-พีเพิล	
เหตุการณ์	ผู้ทดสอบระบบอัปโหลดไฟล์เพื่อแทรกรหัสต้นทาง	
สิ่งกระตุ้นเหตุการณ์	ผู้ทดสอบระบบต้องการแทรกรหัสต้นทาง เพื่อจัดเก็บประวัติการทดสอบ	
รายละเอียด	ผู้ทดสอบระบบนำโครงการดับเบิลยูเอส-พีเพิลที่ต้องการทดสอบและจัดเก็บประวัติการทดสอบ มาอัปโหลดผ่านเครื่องมือเพื่อแทรกรหัสต้นทางสำหรับจัดเก็บข้อมูล	
ผู้กระทำ	ผู้ทดสอบระบบ	
ยูสเคสที่เกี่ยวข้อง	-	
เงื่อนไข	ผู้ทดสอบจะต้องมีโครงการดับเบิลยูเอส-พีเพิลที่ต้องการนำมาจัดเก็บประวัติ	
เงื่อนไขหลัง	โครงการดับเบิลยูเอส-พีเพิล นำอัปโหลดจะถูกแทรกรหัสต้นทางเพื่อจัดเก็บประวัติการทดสอบ และเครื่องมือจะจัดเก็บเส้นทางการไหลของดับเบิลยูเอส-พีเพิล เข้าสู่ฐานข้อมูล	
ลำดับการทำงาน	ผู้กระทำ	ระบบ
	1. ผู้ทดสอบเลือกแฟ้มโครงการดับเบิลยูเอส-พีเพิล	1.1. ระบบค้นหาไฟล์ที่ต้องการจากแฟ้มโครงการที่ผู้ทดสอบระบุ
	2. ผู้ทดสอบยืนยันการอัปโหลดโครงการดับเบิลยูเอส-พีเพิล	2.1. ระบบอ่านไฟล์ดับเบิลยูเอส-พีเพิล และจัดเก็บเส้นทางการไหลเข้าสู่ฐานข้อมูล 2.2. ระบบแทรกรหัสต้นทางที่ไฟล์ดับเบิลยูเอส-พีเพิล

ตารางที่ ก-1 คำอธิบายยูสเคสอัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเฟล (ต่อ)

ชื่อยูสเคส	อัปโหลดและแทรกรหัสต้นทางโครงการดับเบิลยูเอส-บีเฟล	
ลำดับการทำงาน	ผู้กระทำ	ระบบ
	3. ผู้ทดสอบตรวจสอบเพิ่มโครงการดับเบิลยูเอส-บีเฟลที่ถูกแทรกหัสแล้ว	2.3. ระบบแสดงข้อความการแทรกหัสต้นทางสำเร็จ
เงื่อนไขข้อผิดพลาด	2. กรณีที่ผู้ทดสอบอัปโหลดไฟล์ไม่ครบตามที่ระบบต้องการ หรือไม่ได้เลือกเพิ่มโครงการระบบจะแจ้งเตือนผู้ทดสอบ 2.1. ในกรณีที่ระบบตรวจพบว่าเพิ่มโครงการนี้ถูกแทรกหัสต้นทางแล้วระบบจะแจ้งเตือนข้อความให้กับผู้ทดสอบระบบ	

2. ดูข้อมูลเว็บเซอร์วิชที่ไม่ถูกทดสอบ

ตารางที่ ก-2 คำอธิบายยูสเคสดูเว็บเซอร์วิชที่ไม่ถูกทดสอบ

ชื่อยูสเคส	ดูเว็บเซอร์วิชที่ไม่ถูกทดสอบ
เหตุการณ์	ผู้ทดสอบระบบต้องการดูเว็บเซอร์วิชที่ไม่ถูกทดสอบ
สิ่งกระตุ้นเหตุการณ์	ผู้ทดสอบระบบเลือกโครงการดับเบิลยูเอส-บีเฟลที่ต้องการดูเว็บเซอร์วิชที่ไม่ถูกทดสอบ
รายละเอียด	หลังจากที่โครงการดับเบิลยูเอส-บีเฟล ถูกจัดเก็บเส้นทางการไหลและถูกแทรกหัสต้นทางแล้ว เมื่อผู้ทดสอบระบบนำโครงการนี้ไปติดตั้งและทดสอบ ผู้ทดสอบสามารถดูผลการทดสอบและเว็บเซอร์วิชที่ยังไม่ถูกทดสอบได้
ผู้กระทำ	ผู้ทดสอบระบบ
ยูสเคสที่เกี่ยวข้อง	-
เงื่อนไข	ผู้ทดสอบระบบนำโครงการดับเบิลยูเอส-บีเฟล มาแทรกหัสต้นทางแล้ว
เงื่อนไขหลัง	ระบบแสดงข้อมูลเว็บเซอร์วิชที่ไม่ถูกทดสอบ

ตารางที่ ก-2 คำอธิบายยูสเคสดูเว็บเซอร์วิชที่ไม่ถูกทดสอบ (ต่อ)

ชื่อยูสเคส	ดูเว็บเซอร์วิชที่ไม่ถูกทดสอบ	
ลำดับการทำงาน	ผู้กระทำ	ระบบ
	1. เลือกโครงการดับเบิลยูเอส-บีเพลที่ต้องการดูผล	1.1. ค้นหาข้อมูลและแสดงเว็บเซอร์วิชที่ไม่ถูกทดสอบ
เงื่อนไขข้อผิดพลาด	-	

3. สร้างกรณีทดสอบเพิ่มเติม

ตารางที่ ก-3 คำอธิบายยูสเคสสร้างกรณีทดสอบเพิ่มเติม

ชื่อยูสเคส	สร้างกรณีทดสอบเพิ่มเติม	
เหตุการณ์	ผู้ทดสอบระบบต้องการสร้างกรณีทดสอบเพิ่มเติม เพื่อให้ครอบคลุมเว็บเซอร์วิชที่เลือก	
สิ่งกระตุ้นเหตุการณ์	ผู้ทดสอบระบบต้องการสร้างกรณีทดสอบเพิ่มเติม เพื่อให้ครอบคลุมเว็บเซอร์วิชที่เลือก	
รายละเอียด	หลังจากที่ผู้ทดสอบระบบพบเว็บเซอร์วิชที่ยังไม่ถูกทดสอบ ผู้ทดสอบระบบสามารถเลือกเว็บเซอร์วิชนั้นเพื่อสร้างกรณีทดสอบเพิ่มเติมได้	
ผู้กระทำ	ผู้ทดสอบระบบ	
ยูสเคสที่เกี่ยวข้อง	-	
เงื่อนไข	ผู้ทดสอบระบบนำโครงการดับเบิลยูเอส-บีเพล มาแทรกรหัสต้นทางแล้ว	
เงื่อนไขหลัง	ระบบสร้างกรณีทดสอบ	
ลำดับการทำงาน	ผู้กระทำ	ระบบ
	1. เลือกเว็บเซอร์วิชที่ไม่ถูกทดสอบ	1.1. ตรวจสอบว่าสามารถสร้างกรณีทดสอบให้ครอบคลุมได้หรือไม่ 1.2. สร้างกรณีทดสอบเพิ่มเติม 1.3. แจ้งเตือนผู้ทดสอบระบบเมื่อสร้างกรณีทดสอบสำเร็จ
	2. ผู้ทดสอบตรวจสอบไฟล์กรณีทดสอบ	

ตารางที่ ก-3 คำอธิบายยูสเคสสร้างกรณีทดสอบเพิ่มเติม (ต่อ)

ชื่อยูสเคส	สร้างกรณีทดสอบเพิ่มเติม
เงื่อนไขข้อผิดพลาด	1 กรณีที่เป็นเว็บเซอร์วิสที่ไม่สามารถสร้างกรณีทดสอบได้ ระบบจะแจ้งเตือนผู้ทดสอบระบบโดยการคาดแถบสีที่รายการนั้น

4. รายงานผลการทดสอบ

ตารางที่ ก-4 คำอธิบายยูสเคสรายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส

ชื่อยูสเคส	รายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส	
เหตุการณ์	ผู้ทดสอบระบบเรียกดูรายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส	
สิ่งกระตุ้นเหตุการณ์	เมื่อผู้ทดสอบระบบต้องการดูรายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส	
รายละเอียด	ผู้ทดสอบเรียกดูรายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส	
ผู้กระทำ	ผู้ทดสอบระบบ	
ยูสเคสที่เกี่ยวข้อง	-	
เงื่อนไข	โครงการดับเบิลยูเอส-บีเพล ที่สามารถเรียกดูรายงานได้จะต้องเป็นโครงการที่ถูกจัดเก็บเส้นทางการไหลเข้าสู่ฐานข้อมูลแล้ว	
เงื่อนไขหลัง	รายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส	
ลำดับการทำงาน	ผู้กระทำ	ระบบ
	1. เลือกโครงการดับเบิลยูเอส-บีเพลที่ต้องการดูรายงานสรุป จากนั้นคลิกปุ่มดูรายงาน	1.1. ระบบอ่านข้อมูลการทดสอบ 1.2. ระบบแสดงรายงานให้แก่ผู้ทดสอบระบบ
เงื่อนไขข้อผิดพลาด	-	

ตารางที่ ก-5 คำอธิบายยูสเคสดูรายงานผลการทดสอบการเรียกใช้เว็บเซอร์วิส

ชื่อยูสเคส	ดูรายงานผลการทดสอบการเรียกใช้เว็บเซอร์วิส	
เหตุการณ์	ผู้ทดสอบระบบเรียกดูรายงานผลการทดสอบการเรียกใช้เว็บเซอร์วิส	
สิ่งกระตุ้นเหตุการณ์	เมื่อผู้ทดสอบระบบต้องการดูรายงานผลการทดสอบการเรียกใช้เว็บเซอร์วิส	
รายละเอียด	ผู้ทดสอบเรียกดูรายงานสรุปความครอบคลุมของการเรียกใช้เว็บเซอร์วิส	
ผู้กระทำ	ผู้ทดสอบระบบ	
ยูสเคสที่เกี่ยวข้อง	-	
เงื่อนไข	<ul style="list-style-type: none"> - โครงการดับเบิลยูเอส-บีเพล ที่สามารถเรียกดูรายงานได้จะต้องเป็นโครงการที่ถูกจัดเก็บเส้นทางการไหลเข้าสู่ฐานข้อมูลแล้ว - โครงการดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัส จะต้องถูกนำไปติดตั้งและทดสอบผ่านเครื่องมือบีเพลเอนจิน 	
เงื่อนไขหลัง	รายงานผลการทดสอบการเรียกใช้เว็บเซอร์วิส	
ลำดับการทำงาน	ผู้กระทำ	ระบบ
	1. เลือกโครงการดับเบิลยูเอส-บีเพลที่ต้องการดูรายงาน	1.1. ระบบอ่านข้อมูลการทดสอบ 1.2. ระบบแสดงรายการการทดสอบ
	2. เลือกการทดสอบที่ต้องการดูรายงาน	2.1. ระบบค้นหาและแสดงข้อมูลการทดสอบ
	3. คลิกที่ปุ่ม Print เพื่อดูรายงาน	3.1. ระบบแสดงรายงานให้แก่ผู้ทดสอบระบบ
เงื่อนไขข้อผิดพลาด	-	

ภาคผนวก ข

ตัวอย่างการทดสอบเครื่องมือตามกรณีศึกษา

ในภาคผนวก ข จะแสดงรายละเอียดและขั้นตอนการทดสอบเครื่องมือตามกรณีศึกษาที่นำมาใช้
ดังนี้

1. กรณีศึกษาระบบออกบัตรคิวผู้ป่วยนอก

ในกรณีศึกษานี้ผู้วิจัยต้องการทดสอบการทำงานของเครื่องมือว่าสามารถแสดงข้อมูลนำเข้า
และข้อมูลส่งออกที่รับส่งระหว่างเว็บเซอร์วิชได้หรือไม่ โดยในกรณีศึกษานี้จะมีฟังก์ชัน
เว็บเซอร์วิซจำนวน 6 ฟังก์ชัน ดังตารางที่ ข-1

ตารางที่ ข-1 รายการเว็บเซอร์วิซที่เกี่ยวข้องกับกรณีศึกษาที่ 1

ชื่อพาร์ทเนอร์ลิงก์	ชื่อฟังก์ชันเว็บเซอร์วิซ
SOAPPatient	getPatient
SOAPPatient	checkPatientRight
SOAPMedicationServices	getPriority
SOAPMedicationServices	getDepartment
SOAPQueueService	getDiagQueueID
SOAPQueueService	getRegisterQueueID

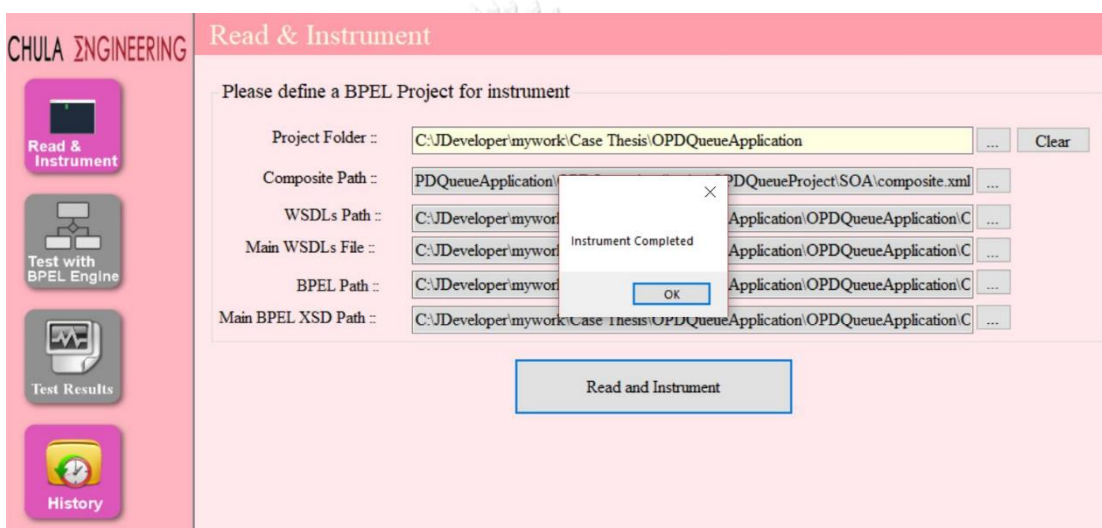
ซึ่งจะทดสอบโดยใช้กรณีทดสอบ 2 กรณีทดสอบคือ

1. กรณีทดสอบผู้ป่วยที่มีประวัติแล้วเข้ามารับการรักษ
2. กรณีทดสอบที่เป็นผู้ป่วยใหม่เข้ามาใช้บริการรักษา โดยจะต้องรับคิวจากห้องเวชระเบียนเพื่อ
ทำบัตรผู้ป่วย

ขั้นตอนการทดสอบมีดังนี้

1.1. ขั้นตอนการอ่านและแทรกรหัสต้นทาง

ในขั้นตอนนี้จะนำโครงการดับเบิลยูเอส-บีเพลมาแทรกรหัสต้นทาง โดยเลือกแฟ้มโครงการดับเบิลยูเอส-บีเพลที่ต้องการ เครื่องมือจะค้นหาไฟล์ที่ใช้ในการอ่านและแทรกรหัสต้นทางจากแฟ้มโครงการดับเบิลยูเอส-บีเพลให้ ซึ่งประกอบด้วย ไฟล์คอมโพสิต ไฟล์ดับเบิลยูเอสดีแอล และไฟล์ดับเบิลยูเอส-บีเพล เมื่อเครื่องมือค้นหาไฟล์เรียบร้อยแล้ว ผู้ทดสอบจะต้องทำการตรวจสอบไฟล์อีกครั้ง จากนั้นจึงคลิกที่ปุ่ม “Read & Instrument” เพื่อให้เครื่องมืออ่านและจัดเก็บเส้นทางการไหลและแทรกรหัสต้นทาง เมื่อแทรกรหัสต้นทางสำเร็จระบบจะแสดงข้อความแจ้งเตือน ดังรูปที่ ข-1



รูปที่ ข-1 ขั้นตอนการอ่านและแทรกรหัสต้นทางสำเร็จ

1.2. ขั้นตอนการนำโครงการดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสแล้ว ไปทดสอบผ่านเครื่องมือบีเพลเอนจิน

ในขั้นตอนนี้ผู้ทดสอบจะนำโครงการดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางแล้วมาติดตั้งและทดสอบผ่านเครื่องมือบีเพลเอนจิน โดยในการทดสอบผ่านเครื่องมือบีเพลเอนจิน ผู้ทดสอบสามารถระบุกรณีทดสอบที่ต้องการทดสอบได้ โดยตัวอย่างจะระบุกรณีทดสอบที่ผู้ป่วยที่เคยมีประวัติการตรวจแล้ว เข้ามารับบริการ ดังรูปที่ ข-2

Tree View ▾ Enable Validation Load Payload Browse... Save Payload

SOAP Body

View ▾ Detach

Name	Type	Value
* payload	payload	
hn	string	1100002
sso_id	string	SSO
department_id	string	1
priority_id	string	1

Request Response

รูปที่ ข-2 ขั้นตอนการระบุกรณีทดสอบผ่านเครื่องมือบีเพลเอนจิน

1.3. ขั้นตอนการดูผลการทดสอบ

หลังจากที่ผู้ทดสอบนำกรณีทดสอบไปทดสอบผ่านเครื่องมือบีเพลเอนจินแล้ว ผู้ทดสอบสามารถดูผลการทดสอบผ่านทางเครื่องมือได้ ดังรูปที่ ข-3 ซึ่งจะพบว่าฟังก์ชันของเว็บเซอร์วิสที่ไม่ถูกทดสอบคือฟังก์ชัน “getRegisterQueueID” โดยฟังก์ชันนี้จะถูกเรียกใช้งานจากโหนดต่างๆ 3 โหนด ได้แก่ โหนด “call_register_no_Pri” โหนด “get_registQueue” และ โหนด “getRegistQueue” นอกจากนี้ผู้ใช้งานยังสามารถเรียกดูประวัติข้อมูลนำเข้าและข้อมูลส่งออกที่ถูกจัดเก็บได้ ดังรูปที่ ข-4 และสามารถดูรายงานสรุปผลการทดสอบได้โดยคลิกที่ปุ่ม “View Summary Report” ดังรูปที่ ข-5

History :: BPEL Task info

BPEL ID :: 286
Project Name :: OPDQueueProject

To be test Operation :: (3)

NodeName	Web Service	Method
call_register_noPri	SOAPQueueService	getRegisterQueueID
get_registQueue	SOAPQueueService	getRegisterQueueID
getRegisQueue	SOAPQueueService	getRegisterQueueID

Test History List ::

Test Date	Test ID
08 Apr 2018 20:39:31	214

Generate Additional Test case Open test case directory

View Summary Report

รูปที่ ข-3 ข้อมูลเว็บเซอร์วิสที่ไม่ถูกทดสอบ

History :: Test History Report

BPEL ID :: 286 Test ID :: 214 Test Time :: 08 Apr 2018 20:39:31 Print

Project Name :: OPDQueueProject

NodeName	Web Service	Method
Receive1	bpelopdqueueflow_client	process
getPatientData	SOAPPatient	getPatient
getSSO	SOAPPatient	checkPatientRight
getPriority	SOAPMedication	getPriority
getDepartment	SOAPMedication	getDepartment
getDiagQueue	SOAPQueueService	getDiagQueueID
callbackClient	bpelopdqueueflow_client	processResponse

Message Details ::

Input ::

Message time : 08 Apr 2018 20:39:31

getPatient
hn = 1100002

Actual output ::

Message time : 08 Apr 2018 20:39:32

getPatientResponse
-getPatientResult
completeStatus = 1
dateOfBirth = 1969-04-02T00:00:00

รูปที่ ข-4 ข้อมูลนำเข้าและข้อมูลส่งออกที่รับส่งระหว่างเว็บเซอร์วิส

Test Summary Report Print Date Time : 22May2018 20:50:39

BPEL ID : 286 Project Name : OPDQueueProject

Total Test : 1 Times
Total Generate Test Case: 0 Times
Total Not Coverage Invocation : 1

Not Coverage Invocation List

Node Name	Method	Condition	Branch
call_register_noPri	getRegisterQueueID	\$inputVariable.payload/client:priority_id != ""	False
get_registQueue	getRegisterQueueID	\$inputVariable.payload/client:sso_id != ""	False
getRegisQueue	getRegisterQueueID	\$inputVariable.payload/client:hn != ""	False

รูปที่ ข-5 รายงานสรุปผลการทดสอบสำหรับกรณีศึกษาที่ 1


จากรูปที่ ข-4 ผู้ทดสอบสามารถดูรายการข้อมูลการทดสอบได้โดยคลิกปุ่ม “Print” เครื่องมือจะแสดงรายละเอียดข้อมูลนำเข้าและข้อมูลส่งออกที่รับส่งระหว่างเว็บเซอร์วิสให้กับผู้ทดสอบ ในรูปแบบของรายงาน ดังรูปที่ ข-6

Node Name	Web Service	Method	Input	Output
Receive1	bpelopdqueueflow_client	process	Message time : 08 Apr 2018 20:39:31 ----- process hn = 1100002 sso_id = SSO department_id = 1 priority_id = 1	Message time :
getPatientData	SOAPPatient	getPatient	Message time : 08 Apr 2018 20:39:31 ----- getPatient hn = 1100002	Message time : 08 Apr 2018 20:39:32 ----- getPatientResponse -getPatientResult completeStatus = 1 dateOfBirth = 1969-04-02T00:00:00 fullname = Mr.PatientA LastNameA gender = M hn = 1100002
getSSO	SOAPPatient	checkPatientRight	Message time : 08 Apr 2018 20:39:32 ----- checkPatientRight	Message time : 08 Apr 2018 20:39:32 ----- checkPatientRightResponse

รูปที่ ข-6 รายงานผลการทดสอบสำหรับกรณีศึกษาที่ 1

1.4. ขั้นตอนนำกรณีทดสอบที่เป็นผู้ป่วยใหม่มาทดสอบ

ในขั้นตอนนี้ผู้ทดสอบจะนำกรณีทดสอบที่เป็นผู้ป่วยใหม่มาใช้ทดสอบกับเครื่องมือบีเพลเอนจิน ดังรูปที่ ข-7 ซึ่งหลังจากการทดสอบเมื่อกลับมาตรวจสอบผลการทดสอบที่เครื่องมือพบว่า ฟังก์ชันของเว็บเซอร์วิสทั้งหมดที่อยู่ภายใต้ดับเบิ้ลยูเอส-บีเพลในกรณีศึกษาที่ 1 ถูกทดสอบทั้งหมด ดังรูปที่ ข-8

View  Detach

Name	Type	Value
* payload	payload	
hn	string	<input type="text"/>
sso_id	string	<input type="text"/>
department_id	string	<input type="text"/>
priority_id	string	<input type="text"/>

รูปที่ ข-7 การนำกรณีทดสอบผู้ป่วยใหม่มาใช้ทดสอบผ่านเครื่องมือบีเพลเอนจิน

History :: BPEL Task info

BPEL ID :: 286

Project Name :: OPDQueueProject

To be test Operation :: (0)

NodeName	Web Service	Method
----------	-------------	--------

รูปที่ ข-8 เว็บไซต์วิชภายใต้ดับเบิลยูเอส-บีเพล ของกรณีศึกษาที่ 1 ถูกทดสอบทั้งหมด
1.5. ขั้นตอนดูรายงานสรุปผลการทดสอบ

เมื่อผู้ทดสอบดูรายงานสรุปผลการทดสอบ หลังจากทำการทดสอบทั้ง 2 กรณีทดสอบเรียบร้อยแล้ว จะพบว่าไม่มีเว็บไซต์วิชใดในกรณีศึกษาที่ 1 ที่ยังไม่ถูกทดสอบ ดังรูปที่ ข-9

Test Summary Report		Print Date Time : 22May2018 20:53.11	
BPEL ID : 286	Project Name : OPDQueueProject		
Total Test : 2 Times			
Total Generate Test Case: 0 Times			
Total Not Coverage Invocation : 0			
Not Coverage Invocation List			
Node Name	Method	Condition	Branch

รูปที่ ข-9 รายงานสรุปผลการทดสอบหลังจากนำกรณีทดสอบผู้ช่วยใหม่ไปทดสอบจากการทดสอบในกรณีทดสอบนี้สามารถสรุปผลการทดสอบได้ดังตารางที่ ข-2

ตารางที่ ข-2 สรุปผลการทดสอบกรณีศึกษาที่ 1

ลำดับ	การทดสอบ	ผลที่คาดหวัง	ผลการทดสอบ
1	การอ่านและจัดเก็บเส้นทางการไหลของดับเบิลยูเอส-บีเพล	สามารถจัดเก็บเส้นทางการไหลของดับเบิลยูเอส-บีเพลเข้าสู่ฐานข้อมูลได้	ผ่าน
2	การแทรกรหัสต้นทางสำหรับจัดเก็บผลการทดสอบ	สามารถแทรกรหัสต้นทางเพื่อจัดเก็บผลการทดสอบได้	ผ่าน

ตารางที่ ข-2 สรุปผลการทดสอบกรณีศึกษาที่ 1 (ต่อ)

ลำดับ	การทดสอบ	ผลที่คาดหวัง	ผลการทดสอบ
3	การนำโครงการดับเบิลยูเอส-ปีเพลที่ถูกแทรกรหัสต้นทางไปติดตั้งและทดสอบผ่านเครื่องมือปีเพลเอนจิน	สามารถติดตั้งและทดสอบผ่านเครื่องมือปีเพลเอนจินได้	ผ่าน
4	การดูรายการเว็บเซอร์วิสที่ไม่ถูกทดสอบ	สามารถดูรายการเว็บเซอร์วิสที่ไม่ถูกทดสอบได้	ผ่าน
5	การดูผลการทดสอบ ข้อมูลนำเข้า และข้อมูลส่งออกจากเว็บเซอร์วิส	สามารถดูข้อมูลผลการทดสอบ ข้อมูลนำเข้าและข้อมูลส่งออกจากเว็บเซอร์วิสได้	ผ่าน
6	การดูรายงานสรุปผลการทดสอบและดูรายงานผลการทดสอบ	สามารถดูรายงานสรุปผลการทดสอบและดูรายงานผลการทดสอบ	ผ่าน

2. กรณีศึกษาระบบค้นหาเที่ยวบิน ที่พัก และรถยนต์

ในกรณีศึกษานี้ผู้วิจัยต้องการทดสอบการทำงานของเครื่องมือ ว่าสามารถสร้างกรณีทดสอบได้ครอบคลุมทุกการเรียกใช้งานเว็บเซอร์วิสหรือไม่ และสามารถรับและแสดงข้อมูลส่งออกที่อยู่ในรูปแบบรายการได้หรือไม่ โดยในกรณีศึกษานี้จะมีฟังก์ชัน เว็บเซอร์วิสจำนวน 3 ฟังก์ชัน ดังตารางที่ ข-3

ตารางที่ ข-3 รายการเว็บเซอร์วิสที่เกี่ยวข้องกับกรณีศึกษาที่ 2

ชื่อพาร์เทนอร์ลิงก์	ชื่อฟังก์ชันเว็บเซอร์วิส
FlightService	getFlight
HotelService	getHotel
CarService	getCar

ซึ่งจะทดสอบโดยใช้กรณีทดสอบ 2 กรณีทดสอบคือ

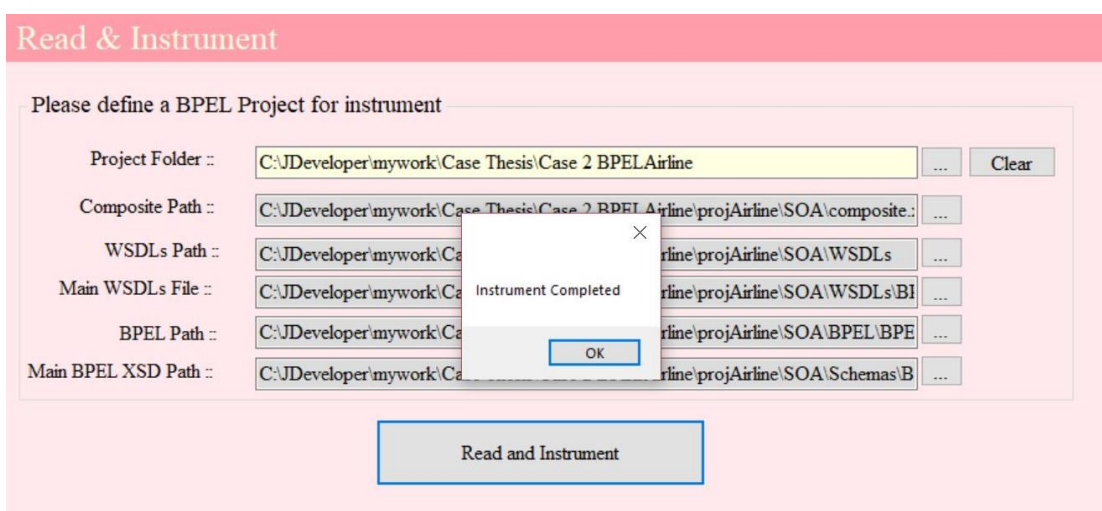
1. กรณีทดสอบการจองเที่ยวบินและโรงแรม

2. กรณีทดสอบที่สร้างจากเครื่องมือ

ในการทดสอบผู้วิจัยมีลำดับการทดสอบเครื่องมือดังนี้

2.1. ขั้นตอนการอ่านและแทรกรหัสต้นทาง

ในขั้นตอนนี้จะนำโครงการดับเบิลยูเอส-บีเพลมาแทรกรหัสต้นทาง โดยเลือกเพิ่มโครงการดับเบิลยูเอส-บีเพล จากนั้นคลิกที่ปุ่ม “Read & Instrument” ดังรูปที่ ข-10



รูปที่ ข-10 ขั้นตอนการอ่านและแทรกรหัสต้นทาง

2.2. ขั้นตอนการนำกรณีทดสอบการจองเที่ยวบินและโรงแรม มาทดสอบผ่านเครื่องมือบีเพลเอนจิน

ในขั้นตอนนี้ผู้ทดสอบจะนำโครงการดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางแล้วมาติดตั้งและทดสอบผ่านเครื่องมือบีเพลเอนจิน โดยในการทดสอบผ่านเครื่องมือบีเพลเอนจิน ผู้ทดสอบสามารถระบุกรณีทดสอบที่ต้องการทดสอบได้ โดยตัวอย่างจะระบุกรณีทดสอบการจองเที่ยวบินและโรงแรม ดังรูปที่ ข-11

* payload	payload	
* flight_source	string	DMK
* flight_destination	string	ICN
* flight_twoWay	boolean	true
* flight_StartDate	date	2018-04-08
* flight_EndDate	date	2018-04-18
* flight_total_seat	integer	2
* hotel_enable	boolean	true
* hotel_area	string	Seoul
* hotel_stayDate	date	2018-04-08
* hotel_leaveDate	date	2018-04-18
* total_totalRoom	integer	1
* car_enable	boolean	false
* car_startDate	date	2018-04-08
* car_endDate	date	2018-04-08
* total_car	integer	0

รูปที่ ข-11 ขั้นตอนการระบุกรณีทดสอบผ่านเครื่องมือปีเพเลนจิน

2.3. ขั้นตอนการดูผลการทดสอบ

หลังจากที่ผู้ทดสอบนำกรณีทดสอบไปทดสอบผ่านเครื่องมือปีเพเลนจินแล้ว ผู้ทดสอบสามารถดูผลการทดสอบผ่านทางเครื่องมือ โดยพบว่าฟังก์ชันเว็บเซอร์วิสจำนวน 2 ฟังก์ชัน คือ “getFlight” และ “getHotel” ได้ถูกทดสอบแล้ว และพบว่ายังมีฟังก์ชันเว็บเซอร์วิส ที่ไม่ถูกทดสอบอีก 1 ฟังก์ชัน คือ “getCar” ดังรูปที่ ข-12 และผู้ทดสอบสามารถดูผลการทดสอบจากกรณีทดสอบที่นำมาทดสอบได้ดังรูปที่ ข-13 โดยเครื่องมือจะรองรับการรับส่งข้อมูลที่อยู่ในรูปแบบรายการของ รายการโรงแรมที่ได้รับจากเว็บเซอร์วิส ผู้ทดสอบสามารถพิมพ์ผลการทดสอบโดยกดปุ่ม “Print” ได้ซึ่งจะได้รายงานดังรูปที่ ข-14

History :: BPEL Task info		
BPEL ID :: 287		
Project Name :: projAirline		
To be test Operation :: (1)		
NodeName	Web Service	Method
Call_Car_service	CarService	getCar
Test History List ::		
Test Date	Test ID	
08 Apr 2018 23:25:59	218	
Generate Additional Test case		Open test case directory

รูปที่ ข-12 เว็บเซอร์วิสที่ไม่ถูกทดสอบหลังจากนำกรณีทดสอบการจองเที่ยวบินและโรงแรมมาทดสอบ

History :: Test History Report

BPEL ID :: 287 Test ID :: 218 Test Time :: 08 Apr 2018 23:25:59 Print

Project Name :: projAirline

NodeName	Web Service	Method
receiveInput	bpelairline_client	process
Call_FlightService	FlightService	getFlight
Call_HotelService	HotelService	getHotel
callbackClient	bpelairline_client	processResponse

Message Details ::

Input ::

```

Message time : 08 Apr 2018 23:25:59
-----
getHotel
-oHotel
hotelArea = Seoul
hotel_LeaveDate = 2018-04-18

```

Actual output ::

```

hotel_Name = Hotel D
hotel_address = Address Hotel D
hotel_LeaveDate = 2018-04-18
hotel_price = 5122.5
hotel_stayDate = 2018-04-08
total_room = 1

```

รูปที่ ข-13 ตัวอย่างการแสดงผลข้อมูลส่งออกที่อยู่ในรูปแบบรายการ

Test Details Report			Print Date Time : 22May2018 21:29:37	
BPEL ID : 287		Project Name : projAirline		Test Time : 08 Apr 2018 23:25:59
Node Name	Web Service	Method	Input	Output
receiveInput	bpelairline_client	process	Message time : 08 Apr 2018 23:25:59 ----- process flight_source = DMK flight_destination = ICN flight_twoWay = true flight_StartDate = 2018-04-08 flight_EndDate = 2018-04-18 flight_total_seat = 2 hotel_enable = true hotel_area = Seoul hotel_stayDate = 2018-04-08 hotel_LeaveDate = 2018-04-18 total_totalRoom = 1 car_enable = false car_startDate = 2018-04-08 car_endDate = 2018-04-08 total_car = 0	Message time :
Call_FlightService	FlightService	getFlight	Message time : 08 Apr 2018 23:25:59 ----- getFlight -oFlight destination = ICN flight_EndDate = 2018-04-18 flight_StartDate = 2018-04-08 flight_total_seat = 2 flight_twoWay = true source = DMK	Message time : 08 Apr 2018 23:25:59 ----- getFlightResponse -getFlightResult -Entities_flight destination = ICN flight_EndDate = 2018-04-18 flight_ResultEnd_date = 2018-04-18 16:56:00 flight_ResultStart_date = 2018-04-08 10:21:00 flight_StartDate = 2018-04-08

รูปที่ ข-14 ตัวอย่างรายงานข้อมูลส่งออกจากการทดสอบการจองเที่ยวบินและโรงแรม

2.4. ขั้นตอนการสร้างกรณีทดสอบเพิ่มเติม

จากรูปที่ ข-12 ผู้ทดสอบสามารถสร้างกรณีทดสอบเพิ่มเติมได้ โดยคลิกที่ปุ่ม “Generate Additional Test case” เมื่อเครื่องมือสร้างกรณีทดสอบสำเร็จจะมีข้อความแจ้งเตือน ดังรูปที่ ข-15 ซึ่งเครื่องมือจะสร้างกรณีทดสอบเพิ่มเติมในรูปแบบไฟล์ข้อความ ดังรูปที่ ข-16

History :: BPEL Task info

BPEL ID :: 287
Project Name :: projAirline

To be test Operation :: (1)

NodeName	Web Service	Method
Call_Car_service	CarService	getCar

Test History List ::

Test Date	Test ID
08 Apr 2018 23:25:59	218

Buttons: Generate Additional Test case, Open test case directory, View Summary Report

Dialog: Create testCase Complete (OK)

รูปที่ ข-15 ข้อความเตือน การสร้างกรณีทดสอบสำเร็จ

287_20180408_232952761 4/8/2018 11:29 PM Text Document 1 KB

287_20180408_232952761 - Notepad

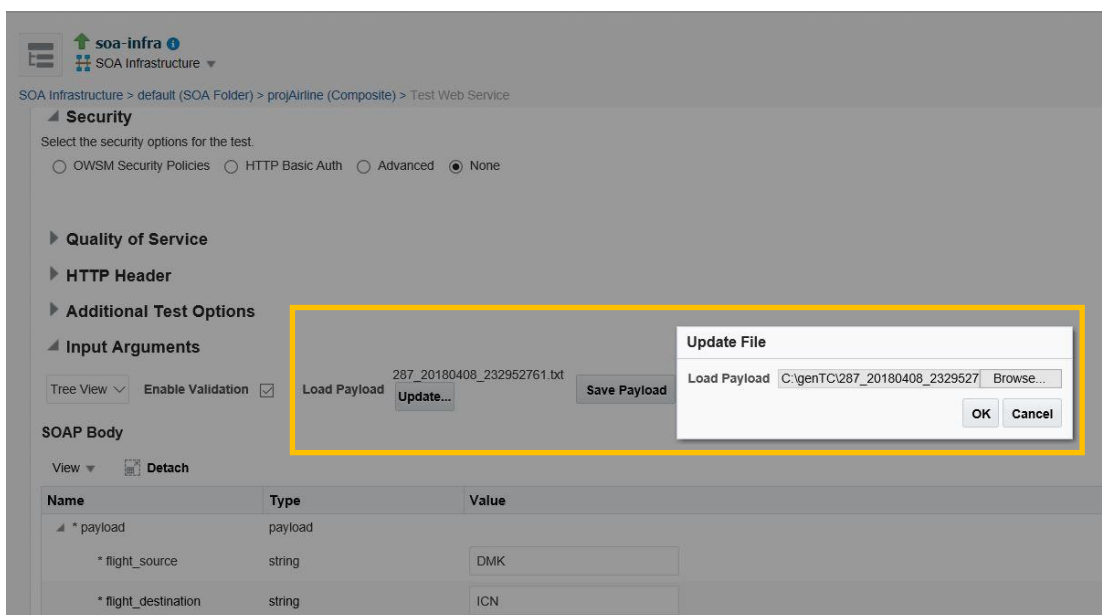
File Edit Format View Help

```
<ns1:process xmlns:ns1="http://xmlns.oracle.com/BPELAirline/projAirline/BPELAirline"
xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <ns1:flight_source>DMK</ns1:flight_source>
  <ns1:flight_destination>ICN</ns1:flight_destination>
  <ns1:flight_twoWay>true</ns1:flight_twoWay>
  <ns1:flight_StartDate>2018-04-08</ns1:flight_StartDate>
  <ns1:flight_EndDate>2018-04-18</ns1:flight_EndDate>
  <ns1:flight_total_seat>2</ns1:flight_total_seat>
  <ns1:hotel_enable>true</ns1:hotel_enable>
  <ns1:hotel_area>Seoul</ns1:hotel_area>
  <ns1:hotel_stayDate>2018-04-08</ns1:hotel_stayDate>
  <ns1:hotel_leaveDate>2018-04-18</ns1:hotel_leaveDate>
  <ns1:total_totalRoom>1</ns1:total_totalRoom>
  <ns1:car_enable>true</ns1:car_enable>
  <ns1:car_startDate>2018-04-08</ns1:car_startDate>
  <ns1:car_endDate>2018-04-08</ns1:car_endDate>
  <ns1:total_car>0</ns1:total_car>
</ns1:process>
```

รูปที่ ข-16 ไฟล์กรณีทดสอบใหม่ที่ได้

2.5. ขั้นตอนการสร้างกรณีทดสอบเพิ่มเติม

หลังจากที่ได้ไฟล์กรณีทดสอบใหม่ดังรูปที่ ข-16 แล้วผู้ทดสอบสามารถนำไฟล์กรณีทดสอบใหม่ที่ได้ ไปทดสอบกับเครื่องมือบีเพลออนจิน ผ่านวิธีการอัปโหลดไฟล์กรณีทดสอบได้ ดังรูปที่ ข-17



รูปที่ ข-17 อัปเดตกรณีทดสอบใหม่ที่ได้นับเครื่องมือปีเพลเอนจิน
เมื่อนำกรณีทดสอบที่เครื่องมือสร้างขึ้นมาทดสอบผ่านปีเพลเอนจินพบว่า เว็บเซอร์วิสทั้งหมด
ถูกทดสอบ ดังรูปที่ ข-18



รูปที่ ข-18 ผลการทดสอบหลังจากที่นำกรณีทดสอบ ที่เครื่องมือสร้างขึ้นมาทดสอบผ่านเครื่องมือ
ปีเพลเอนจิน

นอกจากนี้ผู้ทดสอบยังสามารถดูรายงานสรุปผลการทดสอบของโครงการดับเบิลยูเอส-ปีเพล
ได้ โดยคลิกที่ปุ่ม “View Summary Report” ดังรูปที่ ข-19

Test Summary Report		Print Date Time : 22May2018 21:02.13	
BPEL ID : 287	Project Name : projAirline		
Total Test : 2 Times			
Total Generate Test Case: 1 Times			
Total Not Coverage Invocation : 0			
Not Coverage Invocation List			
Node Name	Method	Condition	Branch

รูปที่ ข-19 รายงานสรุปผลการทดสอบ

จากการทดสอบในกรณีทดสอบนี้สามารถสรุปผลการทดสอบได้ดังตารางที่ ข-4

ตารางที่ ข-4 สรุปผลการทดสอบกรณีศึกษาที่ 2

ลำดับ	การทดสอบ	ผลที่คาดหวัง	ผลการทดสอบ
1	การอ่านและจัดเก็บเส้นทางการไหลของดับเบิลยูเอส-บีเพล	สามารถจัดเก็บเส้นทางการไหลของดับเบิลยูเอส-บีเพลเข้าสู่ฐานข้อมูลได้	ผ่าน
2	การแทรกรหัสต้นทางสำหรับจัดเก็บผลการทดสอบ	สามารถแทรกรหัสต้นทางเพื่อจัดเก็บผลการทดสอบได้	ผ่าน
3	การนำโครงการดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางไปติดตั้งผ่านเครื่องมือบีเพลเอนจิน	สามารถติดตั้งโครงการผ่านผ่านเครื่องมือบีเพลเอนจินได้	ผ่าน
4	การสร้างกรณีทดสอบเพิ่มเติมเพื่อให้ครอบคลุมการเรียกใช้งานเว็บเซอร์วิส	สามารถสร้างกรณีทดสอบเพิ่มเติม สำหรับข้อมูลนำเข้าที่เป็นรูปแบบข้อความ บูลีน และตัวเลขจำนวนเต็ม ได้ และเมื่อนำกรณีทดสอบที่ถูกสร้างขึ้นไปทดสอบผ่านเครื่องมือบีเพลเอนจิน พบว่าเว็บเซอร์วิสถูกทดสอบอย่างครอบคลุม	ผ่าน

ตารางที่ ข-4 สรุปผลการทดสอบกรณีศึกษาที่ 2 (ต่อ)

ลำดับ	การทดสอบ	ผลที่คาดหวัง	ผลการทดสอบ
5	การดูผลการทดสอบ ข้อมูลนำเข้า และข้อมูลส่งออกจากเว็บเซอร์วิส	สามารถดูข้อมูลผลการทดสอบ ข้อมูลนำเข้าและข้อมูลส่งออกที่อยู่ในรูปแบบรายการจากเว็บเซอร์วิสได้	ผ่าน
6	การดูรายงานสรุปผลการทดสอบและดูรายงานผลการทดสอบ	สามารถดูรายงานสรุปผลการทดสอบและดูรายงานผลการทดสอบ	ผ่าน

3. กรณีศึกษาระบบชำระเงินผู้ป่วยนอก

ในกรณีศึกษาที่ผู้วิจัยต้องการทดสอบการทำงานของเครื่องมือ ว่าสามารถสร้างกรณีทดสอบได้ครอบคลุมทุกการเรียกใช้งานเว็บเซอร์วิสหรือไม่ และสามารถรับและแสดงข้อมูลส่งออกของพาร์ทเนอร์ลิงก์ที่มีการเรียกใช้งานเชิงวนซ้ำ โดยในกรณีศึกษาจะมีฟังก์ชัน เว็บเซอร์วิสจำนวน 5 ฟังก์ชัน ดังตารางที่ ข-5

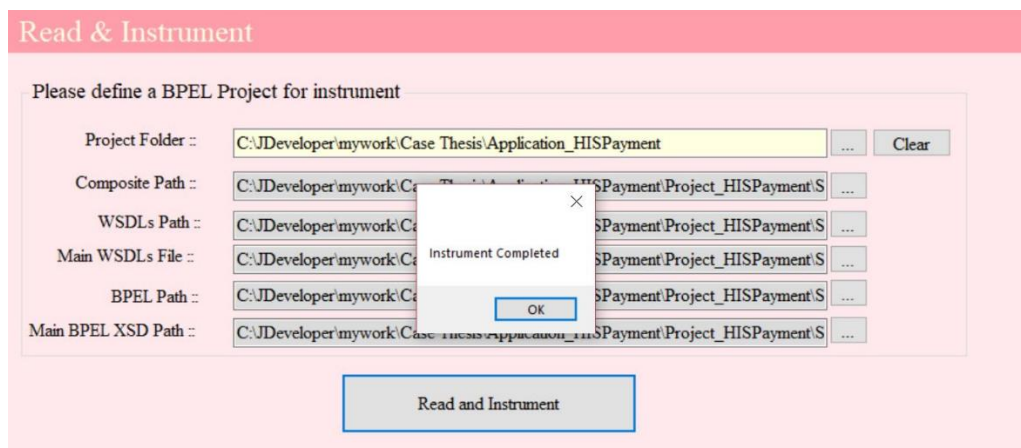
ตารางที่ ข-5 รายการเว็บเซอร์วิสที่เกี่ยวข้องกับกรณีศึกษาที่ 3

ชื่อพาร์ทเนอร์ลิงก์	ชื่อฟังก์ชันเว็บเซอร์วิส
externalCreditCardServices	payCreditCard
paymentServices	submitPayCreditCard
paymentServices	submitPayCash
slipServices	getPrintData
pharmacistServices	getCounsellingSheet

ในการทดสอบของกรณีศึกษาจะใช้กรณีทดสอบที่ถูกสร้างขึ้นจากเครื่องมือทั้งหมด โดยผู้วิจัยมีลำดับการทดสอบเครื่องมือดังนี้

3.1. ขั้นตอนการอ่านและแทรกรหัสต้นทาง

ในขั้นตอนนี้จะนำโครงการดับเบิลยูเอส-บีเพลมาแทรกรหัสต้นทาง โดยเลือกเพิ่มโครงการดับเบิลยูเอส-บีเพล จากนั้นคลิกที่ปุ่ม “Read & Instrument” ดังรูปที่ ข-20



รูปที่ ข-20 ขั้นตอนการอ่านและแทรกรหัสต้นทาง

3.2. ขั้นตอนการสร้างกรณีทดสอบเพิ่มเติม

ในขั้นตอนนี้ผู้ทดสอบสามารถสร้างกรณีทดสอบเพิ่มเติมให้ ได้โดยเข้าเมนู “Test Results” ซึ่งจะพบว่าในโครงการดับเบิลยูเอส-บีเพลนี้มีโหนดที่เรียกใช้งานเว็บเซอร์วิสและรอกทดสอบจำนวน 6 โหนด ดังรูปที่ ข-21 ซึ่งผู้ทดสอบสามารถสร้างกรณีทดสอบให้ครอบคลุมทั้ง 6 โหนดได้เพื่อนำไปทดสอบผ่านบีเพลเอนจิน โดยคลิกที่ปุ่ม “Generate Additional Test Case” ดังรูปที่ ข-22

History :: BPEL Task info

BPEL ID :: 285
Project Name :: Project_HISPayment

To be test Operation :: (6)

NodeName	Web Service	Method
callPayCreditCard	externalCreditCardServ...	payCreditCard
callPayInternalCreditCard	paymentServices	submitPayCreditCard
payCashNotCredit	paymentServices	submitPayCash
payCash	paymentServices	submitPayCash
callGetSlip	slipServices	getPrintData
getCounselling	pharmacistServices	getCounsellingSheet

Test History List ::

Test Date	Test ID

Generate Additional Test case Open test case directory

View Summary Report

รูปที่ ข-21 ข้อมูลเว็บเซอร์วิสที่ไม่ถูกทดสอบ

History :: BPEL Task info

BPEL ID :: 285

Project Name :: Project_HISPayment

To be test Operation :: (6)

NodeName	Web Service	Method
callPayCreditCard	externalCreditCardServ...	payCreditCard
callPayInternalCreditCard	paymentServices	submitPayCreditCard
payCashNotCredit		submitPayCash
payCash		submitPayCash
callGetSlip		getPrintData
getCounselling		getCounsellingSheet

Test History List ::

Test Date	Test ID
-----------	---------

Buttons: Generate Additional Test case, Open test case directory, View Summary Report

Dialog: Create testCase Complete

รูปที่ ข-22 ขั้นตอนการสร้างกรณีทดสอบเพิ่มเติม เครื่องมือแสดงข้อความแจ้งเตือนสร้างกรณีทดสอบเพิ่มเติมสำเร็จ

3.3. ขั้นตอนการทดสอบผ่านเครื่องมือปีเพลเอนจิน

หลังจากที่ผู้ทดสอบระบบสร้างกรณีทดสอบผ่านเครื่องมือแล้ว ผู้ทดสอบสามารถนำกรณีทดสอบที่สร้างขึ้นมาทดสอบผ่านเครื่องมือปีเพลเอนจิน ได้ดังรูปที่ ข-23

soa-infra SOA Infrastructure

SOA Infrastructure > default (SOA Folder) > Project_HISPayment (Composite) > Test Web Service

Input Arguments

Tree View Enable Validation Load Payload 285_20180304_003806844.txt Update... Save Payload

SOAP Body

View Detach

Name	Type	Value
* payload	payload	
* hn	string	S
* vn	string	S
* payComplete	boolean	false
* payMethod	string	CreditCard
* payAmount	double	920116.0
* numberOfSlip	integer	1
* payAttemp	integer	1

Update File

Load Payload C:\genTC\285_20180304_003806844.txt Browse... OK Cancel

รูปที่ ข-23 ขั้นตอนการทดสอบผ่านเครื่องมือปีเพลเอนจิน

เมื่อนำกรณีทดสอบที่เครื่องมือสร้างขึ้นมาทดสอบผ่านปีเพลเอนจินพบว่า เว็บเซอร์วิสถูกทดสอบอย่างครอบคลุม ดังรูปที่ ข-24

History :: BPEL Task info

BPEL ID :: 285
Project Name :: Project_HISPayment

To be test Operation :: (0)

NodeName	Web Service	Method

Test History List ::

Test Date	Test ID
04 Mar 2018 00:45:06	207
04 Mar 2018 00:45:23	208
04 Mar 2018 00:45:37	209
04 Mar 2018 00:45:59	210
04 Mar 2018 00:46:11	211
04 Mar 2018 00:47:44	212

Generate Additional Test case Open test case directory

View Summary Report

รูปที่ ข-24 ผลการทดสอบหลังจากที่นำกรณีทดสอบ ที่เครื่องมือสร้างขึ้นมาทดสอบผ่านเครื่องมือ
บีเพลเอนจิน

ผู้ทดสอบยังสามารถดูรายงานการทดสอบในแต่ละครั้งได้ โดยดับเบิลคลิกเลือกรายการ
ทดสอบเพื่อดูข้อมูลในส่วนของ “Test History List” ของรูปที่ ข-24 เมื่อดับเบิลคลิกรายการ
การทดสอบแล้วเครื่องมือจะแสดงหน้าต่างผลการทดสอบดังรูปที่ ข-25 จะพบว่าเครื่องมือ
รองรับการเก็บข้อมูลการทดสอบที่อยู่ในรูปแบบวนซ้ำ และผู้ทดสอบยังสามารถพิมพ์รายงาน
ผลการทดสอบได้ดังรูปที่ ข-26

History :: Test History Report

BPEL ID :: 285 Test ID :: 212 Test Time :: 04 Mar 2018 00:47:44 Print

Project Name :: Project_HISPayment

NodeName	Web Service	Method
receiveInput	bpelhispayment_client	process
callPayCreditCard	externalCreditCardServices	payCreditCard
callPayCreditCard	externalCreditCardServices	payCreditCard
callPayCreditCard	externalCreditCardServices	payCreditCard
callPayInternalCreditCard	paymentServices	submitPayCreditCard
callGetSlip	slipServices	getPrintData
replyOutput	bpelhispayment_client	process

Message Details ::

Input ::

Message time : 04 Mar 2018 00:47:45

payCreditCard
transactionID = T12345
amount = 920116.0

Actual output ::

Message time : 04 Mar 2018 00:47:45

payCreditCardResponse
payCreditCardResult = true

รูปที่ ข-25 ตัวอย่างข้อมูลนำเข้าและข้อมูลส่งออกที่ในแต่ละรอบของการทำงานแบบวนซ้ำ

Test Details Report		Print Date Time : 22May2018 21:41.18		
BPTEL ID : 285		Project Name : Project_HISPayment		Test Time : 04 Mar 2018 00:47:44
Node Name	Web Service	Method	Input	Output
receiveInput	bpelhispayment_client	process	Message time : 04 Mar 2018 00:47:44 ----- process hn = S vn = S payComplete = false payMethod = CreditCard payAmount = 920116.0 numberOfSilp = 1 payAttemp = 3 haveMed = false	Message time :
callPayCreditCard	externalCreditCardServices	payCreditCard	Message time : 04 Mar 2018 00:47:44 ----- payCreditCard transactionID = T12345 amount = 920116.0	Message time : 04 Mar 2018 00:47:44 ----- payCreditCardResponse payCreditCardResult = true
callPayCreditCard	externalCreditCardServices	payCreditCard	Message time : 04 Mar 2018 00:47:45 ----- payCreditCard transactionID = T12345 amount = 920116.0	Message time : 04 Mar 2018 00:47:45 ----- payCreditCardResponse payCreditCardResult = true
callPayCreditCard	externalCreditCardServices	payCreditCard	Message time : 04 Mar 2018 00:47:45 ----- payCreditCard transactionID = T12345 amount = 920116.0	Message time : 04 Mar 2018 00:47:45 ----- payCreditCardResponse payCreditCardResult = true
callPayInternalCreditCard	paymentServices	submitPayCreditCard	Message time : 04 Mar 2018 00:47:45 -----	Message time : 04 Mar 2018 00:47:45 -----

รูปที่ ข-26 รายงานผลการทดสอบ ข้อมูลนำเข้าและข้อมูลส่งออก ที่รับส่งระหว่างเว็บเซอร์วิส
นอกจากนี้ผู้ทดสอบยังสามารถดูรายงานสรุปผลการทดสอบของโครงการดับเบิลยูเอส-
บีเพลได้ โดยคลิกที่ปุ่ม “View Summary Report” ดังรูปที่ ข-27

Test Summary Report		Print Date Time : 22May2018 21:09.44	
BPTEL ID : 285		Project Name : Project_HISPayment	
Total Test : 6 Times			
Total Generate Test Case: 6 Times			
Total Not Coverage Invocation : 0			
Not Coverage Invocation List			
Node Name	Method	Condition	Branch

รูปที่ ข-27 รายงานสรุปผลการทดสอบ
จากการทดสอบในกรณีทดสอบนี้สามารถสรุปผลการทดสอบได้ดังตารางที่ ข-6

ตารางที่ ข-6 สรุปผลการทดสอบกรณีศึกษาที่ 3

ลำดับ	การทดสอบ	ผลที่คาดหวัง	ผลการทดสอบ
1	การอ่านและจัดเก็บเส้นทางการไหลของดับเบิลยูเอส-บีเพล	สามารถจัดเก็บเส้นทางการไหลของดับเบิลยูเอส-บีเพลเข้าสู่ฐานข้อมูลได้	ผ่าน
2	การแทรกรหัสต้นทางสำหรับจัดเก็บผลการทดสอบ	สามารถแทรกรหัสต้นทางเพื่อจัดเก็บผลการทดสอบได้	ผ่าน
3	การนำโครงการดับเบิลยูเอส-บีเพลที่ถูกแทรกรหัสต้นทางไปติดตั้งผ่านเครื่องมือบีเพลเอนจิน	สามารถติดตั้งโครงการผ่านผ่านเครื่องมือบีเพลเอนจินได้	ผ่าน
4	การสร้างกรณีทดสอบเพิ่มเติมเพื่อให้ครอบคลุมการเรียกใช้งานเว็บเซอร์วิส	สามารถสร้างกรณีทดสอบเพิ่มเติม สำหรับข้อมูลนำเข้าที่เป็นรูปแบบข้อความ บูลีน ตัวเลขจำนวนเต็ม และตัวเลขทศนิยมได้ และเมื่อนำกรณีทดสอบที่ถูกสร้างขึ้นไปทดสอบผ่านเครื่องมือบีเพลเอนจิน พบว่าเว็บเซอร์วิสถูกทดสอบอย่างครอบคลุม	ผ่าน
5	การดูผลการทดสอบ ข้อมูลนำเข้า และข้อมูลส่งออกจากเว็บเซอร์วิส และสามารถดูการเรียกใช้งานเว็บเซอร์วิสที่อยู่ภายใต้การทำงานแบบวนซ้ำ	สามารถดูข้อมูลผลการทดสอบ ข้อมูลนำเข้าและข้อมูลส่งออกจากเว็บเซอร์วิสได้ นอกจากนี้ยังแสดงข้อมูลการทดสอบ การเรียกใช้งานเว็บเซอร์วิสที่อยู่ภายใต้การทำงานแบบวนซ้ำได้	ผ่าน

ตารางที่ ข-6 สรุปผลการทดสอบกรณีศึกษาที่ 3 (ต่อ)

ลำดับ	การทดสอบ	ผลที่คาดหวัง	ผลการทดสอบ
6	การดูรายงานสรุปผลการทดสอบและดูรายงานผลการทดสอบ	สามารถดูรายงานสรุปผลการทดสอบและดูรายงานผลการทดสอบ	ผ่าน



ภาคผนวก ค
พจนานุกรมข้อมูล

ในภาคผนวก ค จะอธิบายพจนานุกรมข้อมูลในแต่ละตาราง โดยมีรายละเอียดดังนี้

ตารางที่ ค-1 พจนานุกรมข้อมูลในตาราง TBL_TASK

ชื่อตาราง	TBL_TASK			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
ID	รหัสโครงการบีเพล	int	ใช่	ใช่
taskDateTime	วันที่จัดเก็บข้อมูล	datetime	ไม่	ใช่
genFromUser	ผู้จัดเก็บ	varchar(50)	ไม่	ไม่
projectName	ชื่อโครงการ	varchar(250)	ไม่	ไม่
bpelName	ชื่อบีเพล	varchar(250)	ไม่	ไม่

ตารางที่ ค-2 พจนานุกรมข้อมูลในตาราง TBL_NodeMaster

ชื่อตาราง	TBL_NodeMaster			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
CreateDateTime	วันเวลาที่จัดเก็บข้อมูลโหนด	datetime	ไม่	ไม่
NodeID	รหัสโหนดหลัก	int	ใช่	ใช่
NodeName	ชื่อโหนด	varchar(250)	ไม่	ไม่
NodeType	ประเภทโหนด	varchar(250)	ไม่	ไม่
NodeLevel	ลำดับชั้นของโหนด	int	ไม่	ไม่
suffixsmall	ลำดับของโหนด	int	ไม่	ไม่

ตารางที่ ค-2 พจนานุกรมข้อมูลในตาราง TBL_NodeMaster (ต่อ)

ชื่อตาราง	TBL_NodeMaster			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
instrumentType	ประเภทการแทรกรหัสต้นทางของ โหนด	varchar(3)	ไม่	ไม่
upperNode	รหัสโหนดหลักก่อนหน้า	int	ไม่	ไม่
task_id	หมายเลขโครงการปีเพล	int	ไม่	ไม่

ตารางที่ ค-3 พจนานุกรมข้อมูลในตาราง TBL_NodeDetail

ชื่อตาราง	TBL_NodeDetail			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
NodeMasterID	รหัสของโหนดหลัก	int	ไม่	ไม่
DetailID	รหัสของรายละเอียดโหนด	int	ใช่	ใช่
InvocationNode PartnerLink	ข้อมูลพาร์ตเนอร์ลิงก์ที่ใช้	varchar(250)	ไม่	ไม่
InvocationNode PortType	ข้อมูลพอร์ตไทป์	varchar(250)	ไม่	ไม่
InvocationNode Operation	ข้อมูลเมธอดที่เรียกใช้ในเว็บเซอร์วิส	varchar(250)	ไม่	ไม่
InvocationNode InputVariable	ชื่อตัวแปรนำเข้าของพาร์ตเนอร์ลิงก์	varchar(250)	ไม่	ไม่
InvocationNode OutputVariable	ชื่อตัวแปรส่งออกของ พาร์ตเนอร์ลิงก์	varchar(250)	ไม่	ไม่

ตารางที่ ค-3 พจนานุกรมข้อมูลในตาราง TBL_NodeDetail (ต่อ)

ชื่อตาราง	TBL_NodeDetail			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
InvocationNode Variable	ชื่อตัวแปรที่ใช้งานในโหนด Receive, Response	varchar(250)	ไม่	ไม่
PredicateNode Condition	เงื่อนไขของโหนด	text	ไม่	ไม่
PredicateNode DriveCondition Value	เงื่อนไขของกิ่งที่โหนดอาศัยอยู่	varchar(250)	ไม่	ไม่
PredicateNodeID	รหัสของโหนดเงื่อนไขที่โหนดอาศัยอยู่	varchar(250)	ไม่	ไม่
PredicateNode Sequence	หมายเลขกิ่งของเงื่อนไขที่โหนดอาศัยอยู่	varchar(250)	ไม่	ไม่
OtherNode Message	เก็บข้อความโหนดอื่นๆ	text	ไม่	ไม่
OtherNode SourceVariable	เก็บข้อมูลการคัดลอกค่าต้นทางในกรณีเป็นโหนด assign	text	ไม่	ไม่
OtherNode Destination Variable	เก็บข้อมูลการคัดลอกค่าปลายทางในกรณีเป็นโหนด assign	text	ไม่	ไม่

ตารางที่ ค-3 พจนานุกรมข้อมูลในตาราง TBL_NodeDetail (ต่อ)

ชื่อตาราง	TBL_NodeDetail			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
NodeDetail PreviousID	เก็บข้อมูลโหนดเงื่อนไขก่อนหน้า	int	ไม่	ไม่
NodeDetail PreviousSuffix	เก็บข้อมูลกิ่งของโหนดเงื่อนไขก่อนหน้า	int	ไม่	ไม่
NodeDetail PreviousDrive Condition	เก็บข้อมูลเงื่อนไขของโหนดเงื่อนไขก่อนหน้า	varchar(250)	ไม่	ไม่
NodeDetail Previous Sequence	เก็บลำดับของโหนดที่อยู่ภายใต้กิ่งของโหนดเงื่อนไข โดยเริ่มนับจากโหนดที่เริ่มทำงาน	int	ไม่	ไม่
NodeDetail Sequence	เก็บลำดับของโหนดที่อยู่ภายใต้กิ่งของโหนดเงื่อนไข โดยเริ่มนับจากโหนดเงื่อนไข	int	ไม่	ไม่

ตารางที่ ค-4 พจนานุกรมข้อมูลในตาราง TBL_ReceiveNodeVar_Schema

ชื่อตาราง	TBL_ReceiveNodeVar_Schema			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
ID	รหัสของโครงสร้างข้อมูล	int	ใช่	ใช่
Task_ID	รหัสของโครงการบีเพล	int	ไม่	ใช่
nodeSequence	ลำดับของข้อมูล	int	ไม่	ไม่
elementName	ชื่อข้อมูล	varchar(50)	ไม่	ไม่

ตารางที่ ค-4 พจนานุกรมข้อมูลในตาราง TBL_ReceiveNodeVar_Schema (ต่อ)

ชื่อตาราง	TBL_ReceiveNodeVar_Schema			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
maxOccurs	จำนวนที่เกิดได้มากที่สุด	varchar(50)	ไม่	ไม่
minOccurs	จำนวนที่เกิดขึ้นได้น้อยสุด	varchar(50)	ไม่	ไม่
dataType	ประเภทข้อมูล	varchar(50)	ไม่	ไม่
nodeLevel	ลำดับชั้นของข้อมูล	int	ไม่	ไม่
rec_date	วันที่จัดเก็บ	datetime	ไม่	ไม่
rec_user	ผู้จัดเก็บ	varchar(50)	ไม่	ไม่
minInclusive	ค่าต่ำสุด	varchar(20)	ไม่	ไม่
maxInclusive	ค่าสูงสุด	varchar(20)	ไม่	ไม่
minLength	ขนาดต่ำสุด	varchar(20)	ไม่	ไม่
maxLength	ขนาดสูงสุด	varchar(20)	ไม่	ไม่

ตารางที่ ค-5 พจนานุกรมข้อมูลในตาราง TBL_XSD_Restriction

ชื่อตาราง	TBL_XSD_Restriction			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
ID	รหัสกรณีทดสอบ	int	ใช่	ใช่
Task_ID	รหัสโครงการบีเพล	int	ไม่	ไม่
elementName	ชื่อข้อมูล	varchar(50)	ไม่	ไม่
parameter_type	ประเภทของข้อจำกัดข้อมูล	varchar(50)	ไม่	ไม่
value	ค่าของข้อมูลที่เกิดขึ้นได้	varchar(50)	ไม่	ไม่

ตารางที่ ค-6 พจนานุกรมข้อมูลในตาราง TBL_TESTCASE

ชื่อตาราง	TBL_TESTCASE			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
TESTCASE_ID	รหัสกรณีทดสอบ	int	ใช่	ใช่
TASK_ID	รหัสโครงการบีเพล	int	ไม่	ไม่
TESTCASE_MESSAGE	ข้อความกรณีทดสอบ	text	ไม่	ไม่
RECDATE	วันที่สร้างกรณีทดสอบ	datetime	ไม่	ไม่
RECUSER	ชื่อผู้สร้างกรณีทดสอบ	varchar(50)	ไม่	ไม่
ForNodeID	รหัสโหนดที่สร้างกรณีทดสอบ	int	ไม่	ไม่
ForNodeCond Name	โหนดเงื่อนไขที่สร้างกรณีทดสอบ	varchar(50)	ไม่	ไม่
ForNodeCond Text	ข้อความเงื่อนไขของโหนดเงื่อนไขที่สร้างกรณีทดสอบ	text	ไม่	ไม่
ForNodeCond Type	ประเภทของโหนดเงื่อนไขที่สร้างกรณีทดสอบ	varchar(50)	ไม่	ไม่

ตารางที่ ค-7 พจนานุกรมข้อมูลในตาราง TBL_TESTCASE_TEMPLATE

ชื่อตาราง	TBL_TESTCASE_TEMPLATE			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
id	รหัสกรณีทดสอบ	int	ใช่	ใช่
task_id	รหัสโครงการบีเพล	int	ไม่	ไม่

ตารางที่ ค-7 พจนานุกรมข้อมูลในตาราง TBL_TESTCASE_TEMPLATE (ต่อ)

ชื่อตาราง	TBL_TESTCASE_TEMPLATE			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
testcase Template	ข้อความกรณีทดสอบ	text	ไม่	ไม่
createdatetime	วันที่สร้างกรณีทดสอบ	datetime	ไม่	ไม่
create_user	ชื่อผู้สร้างกรณีทดสอบ	varchar(50)	ไม่	ไม่

ตารางที่ ค-8 พจนานุกรมข้อมูลในตาราง WS_LOG

ชื่อตาราง	WS_LOG			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
Create datetime	วันเวลาที่บันทึกข้อมูล	datetime	ไม่	ไม่
id	รหัสประวัติการทดสอบ	int	ใช่	ใช่
message	ข้อความที่บันทึก	text	ไม่	ไม่
Type	ประเภทของโหนดที่บันทึก (P:Predicate, I:Invoke)	varchar(50)	ไม่	ไม่
testJobID	รหัสการทดสอบ	int	ไม่	ไม่
nodeName	ชื่อโหนดที่บันทึกข้อมูล	varchar(150)	ไม่	ไม่
Operation Name	ชื่อเมธอดที่ถูกทดสอบ	varchar(150)	ไม่	ไม่
portType Name	ชื่อพอร์ตไทป์ที่ถูกทดสอบ	varchar(150)	ไม่	ไม่

ตารางที่ ค-8 พจนานุกรมข้อมูลในตาราง WS_LOG (ต่อ)

ชื่อตาราง	WS_LOG			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
taskID	รหัสโครงการบีเพล	int	ไม่	ไม่
partnerLink Name	ชื่อพาร์ทเนอร์ลิงก์ที่ถูกทดสอบ	varchar(150)	ไม่	ไม่
suffixsmall	ลำดับการทดสอบ	int	ไม่	ไม่
AfterBefore Status	สถานะก่อน-หลังการทดสอบ (A:After, B:Before)	varchar(3)	ไม่	ไม่
NodeID	รหัสโหนดที่ถูกทดสอบ	varchar(50)	ไม่	ไม่

ตารางที่ ค-9 พจนานุกรมข้อมูลในตาราง SERIAL

ชื่อตาราง	SERIAL			
ชื่อฟิลด์	รายละเอียด	ประเภทข้อมูล	คีย์	จำเป็น
NAME	ชื่อรหัส	varchar(50)	ไม่	ใช่
SERIAL_NO	หมายเลขรหัส	int	ไม่	ใช่

ประวัติผู้เขียนวิทยานิพนธ์

นายณัฐพงศ์ เอื้อเพิ่มเกียรติ เกิดเมื่อวันที่ 2 พฤศจิกายน 2530 สำเร็จการศึกษาระดับปริญญาตรี หลักสูตรวิทยาศาสตร์บัณฑิต (วท.บ.) สาขาวิทยาการคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ปีการศึกษา 2552

ประสบการณ์การทำงาน ได้เข้าทำงานที่บริษัท สุปริม ไฮทีร์่า จำกัด ในตำแหน่งโปรแกรมเมอร์อาวุโส เป็นเวลา 7 ปี และปัจจุบันทำงานที่บริษัท ไอทีวัน จำกัด ในตำแหน่งที่ปรึกษา

ได้เข้าศึกษาในระดับปริญญาโทมหาบัณฑิต หลักสูตรวิทยาศาสตรมหาบัณฑิต (วท.ม.) สาขาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2558

