เครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ

นางสาวพัชร์วรรณ อักษรศรี

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2556
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

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.

A TOOL FOR CHECKING INTEGRATION TESTING COVERAGE OF OBJECT-ORIENTED SOFTWARE

Miss Pachawan Augsornsri

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

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 2013

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	เครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ
โดย	นางสาวพัชร์วรรณ อักษรศรี
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.ธาราทิพย์ สุวรรณศาสตร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์

(ศาสตราจารย์ ดร.บัณฑิต เอื้ออาภรณ์)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ

(รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(รองศาสตราจารย์ ดร.ธาราทิพย์ สุวรรณศาสตร์)

.....กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์)

.....กรรมการภายนอกมหาวิทยาลัย

(ผู้ช่วยศาสตราจารย์ ดร.ภัทรชัย ลลิตโรจน์วงศ์)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

พัชร์วรรณ อักษรศรี : เครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ. (A TOOL FOR CHECKING INTEGRATION TESTING COVERAGE OF OBJECT-ORIENTED SOFTWARE) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: รศ. ดร.ธาราทิพย์ สุวรรณศาสตร์, 130 หน้า.

สิ่งสำคัญอย่างหนึ่งในการทดสอบซอฟต์แวร์คือ ความครอบคลุมของการทดสอบที่จำเป็นต้องทำให้ครอบคลุมเนื่องจากอาจมีข้อบกพร่องแฝงอยู่ในส่วนที่ไม่ครอบคลุมและพบเมื่อผู้ใช้ได้ใช้งานส่วนที่ไม่ครอบคลุมนั้น ในปัจจุบันนี้การพัฒนาซอฟต์แวร์เชิงวัตถุได้มีบทบาทเพิ่มขึ้นซึ่งวิธีการทดสอบซอฟต์แวร์เชิงวัตถุนั้นต่างจากทดสอบซอฟต์แวร์สัญนิยม ดังนั้นเทคนิคต่างๆ รวมทั้งเครื่องมือจึงไม่สามารถใช้ร่วมกันได้ การทดสอบที่สำคัญของซอฟต์แวร์เชิงวัตถุคือการทดสอบแบบบูรณาการซึ่งจำเป็นต้องทำให้ครอบคลุม แต่ในปัจจุบันยังไม่มีเครื่องมือที่ใช้ในการตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ รวมทั้งยังไม่ปรากฏเครื่องมือใดที่สร้างกรณีทดสอบเพิ่มเติม ในกรณีที่ตรวจสอบพบว่าการทดสอบซอฟต์แวร์ยังไม่ครอบคลุม

ผู้วิจัยจึงพัฒนาเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ โดยเครื่องมือนี้จะแทรกซอร์สโค้ดเพื่อตรวจสอบว่าซอร์สโค้ดส่วนที่ถูกแทรกว่าถูกทดสอบหรือไม่ แล้วตรวจสอบว่ามีคลาสและเมท็อดใดที่ถูกทดสอบบ้าง หลังจากนั้นจึงแสดงผลออกมาเป็นเปอร์เซ็นต์ของจำนวนคลาสที่ครอบคลุมจากการทดสอบต่อจำนวนคลาสที่ทดสอบทั้งหมด และเปอร์เซ็นต์ของเมท็อดที่ครอบคลุมจากการทดสอบต่อจำนวนเมท็อดที่ทดสอบทั้งหมด นอกจากนี้เครื่องมือจะสร้างกรณีทดสอบเพิ่มเติมจากแผนภาพลำดับและคำอธิบายยูสเคส ในกรณีที่มีคลาสหรือเมท็อดที่ไม่ครอบคลุมจากการทดสอบ เพื่อให้เพิ่มความครอบคลุมของการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ หลังจากนั้นผู้วิจัยได้ทดสอบเครื่องมือนี้กับระบบจำลองการทำงาน 13 ระบบ ซึ่งเครื่องมือสามารถตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการได้อย่างถูกต้อง ทั้งยังสามารถสร้างกรณีทดสอบได้ตามที่คาดไว้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต .....

สาขาวิชา วิศวกรรมซอฟต์แวร์ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก .....

ปีการศึกษา 2556

5470299621 : MAJOR SOFTWARE ENGINEERING
KEYWORDS: INTEGRATION TESTING / COVERAGE ANALYSIS / OBJECT-ORIENTED SOFTWARE

PACHAWAN AUGSORNSRI: A TOOL FOR CHECKING INTEGRATION TESTING COVERAGE OF OBJECT-ORIENTED SOFTWARE. ADVISOR: ASSOC. PROF. TARATIP SUWANNASART, Ph.D., 130 pp.

One of main points of software testing is test coverage which has to be covered because defects may exist in uncover parts and appear when users try to use them. Nowadays the object-oriented software is more interesting, which the test method differs from the conventional software, so some of available techniques and tools cannot be applied to object-oriented software. The important testing process of object-oriented software is integration testing which has to be covered. However, there are no tools for checking the integration testing coverage of object-oriented software. In addition, there are no tools which generate additional test cases in case of the existing test cases cannot cover the code.

We implement the tool for checking integration testing coverage of object-oriented software. The tool instruments tested source code to check whether the tested parts are executed or not and check which classes or methods are executed. After that the tool outputs the coverage result which is percentage of covered class and percentage of covered method. Moreover, the tool generates additional test cases from sequence diagrams and use case descriptions in case there are uncover classes or methods in order to increase the integration testing coverage of object-oriented software. After that, we test this tool with 13 systems. The tool can check the integration testing coverage correctly and it can generate additional test cases.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Department: Computer Engineering Student's Signature .....

Field of Study: Software Engineering Advisor's Signature .....

Academic Year: 2013

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้ด้วยความช่วยเหลืออย่างดียิ่งจาก รองศาสตราจารย์ ดร.ธาราทิพย์ สุวรรณศาสตร์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่สละเวลาช่วยเหลือ ให้คำแนะนำและคำปรึกษาเกี่ยวกับการทำวิจัย รวมถึงให้ความรู้ทั้งทางด้านวิชาการ และการดำเนินชีวิต ด้วยความห่วงใยเอาใจใส่ จนวิทยานิพนธ์นี้สำเร็จได้ด้วยดี

ขอขอบพระคุณ รองศาตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ ประธานกรรมการสอบ ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์ และ ผู้ช่วยศาสจารย์ ดร.ภัทรชัย ลลิตโรจน์วงศ์ คณะกรรมการสอบ ที่ช่วยชี้แนะและตรวจสอบวิทยานิพนธ์นี้

ขอขอบคุณอาจารย์ทุกท่านที่ได้ให้ความรู้ซึ่งจำเป็นและเป็นประโยชน์ในการทำวิจัยนี้เป็นอย่างมาก

ขอบคุณเพื่อนๆ ทุกท่านที่ให้คำแนะนำ และช่วยเหลือ และให้กำลังใจเป็นอย่างมากระหว่างการทำวิจัย

ขอบคุณคุณสานิต กาญจนอุทัย สำหรับความช่วยเหลือในทุกๆด้าน และเป็นกำลังใจที่ดีเสมอมา

สุดท้ายนี้ ขอขอบคุณบิดา มารดาของข้าพเจ้าที่สนับสนุนในด้านต่างๆ และเป็นกำลังใจสำคัญระหว่างการทำวิจัย

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญตาราง ..... ญ

สารบัญภาพ ..... ฎ

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของงานวิจัย 1

1.2 วัตถุประสงค์ของงานวิจัย 2

1.3 ขอบเขตของงานวิจัย 2

1.4 ขั้นตอนและวิธีการดำเนินงานวิจัย 3

1.5 ประโยชน์ที่คาดว่าจะได้รับ 4

1.6 ลำดับขั้นตอนในการนำเสนองานวิจัย 4

1.7 ผลงานตีพิมพ์จากงานวิจัย 4

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 6

2.1 ทฤษฎีที่เกี่ยวข้อง 6

2.1.1 การทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ 6

2.1.2 ความครอบคลุมการทดสอบ 6

2.1.3 กราฟระบุทิศทางและกราฟการไหลของการควบคุม (Direct graph and control flow graph) 7

2.1.4 เอกซ์เอ็มแอล (XML : eXtensible Markup Language) 9

2.1.5 แผนภาพลำดับ 10

2.2 งานวิจัยที่เกี่ยวข้อง 13

2.2.1 เครื่องมือ “Jcover” 13

2.2.2 งานวิจัย “Model-Based Test Case Synthesis using UML Interaction Diagrams”[12] 14

2.2.3 วิทยานิพนธ์ “วิธีการสร้างกรณีทดสอบโดยอัตโนมัติจากยูสเคส”[13] 14

หน้า

บทที่ 3 การวิเคราะห์และออกแบบเครื่องมือ 15
3.1 ภาพรวมของการทำงานของเครื่องมือ 15
3.1.1 การแทรกซอร์สโค้ด 16
3.1.2 การดำเนินการทดสอบ 18
3.1.3 การวิเคราะห์ข้อมูล 19
3.1.4 การสร้างกรณีทดสอบ 22
3.2 การวิเคราะห์และออกแบบเครื่องมือ 25
3.2.1 แผนภาพยูสเคส 25
3.2.2 แผนภาพกิจกรรม 28
3.2.3 แผนภาพคลาส 41
3.2.4 แผนภาพลำดับ 55
3.2.5 โครงสร้างฐานข้อมูล 63
บทที่ 4 การพัฒนาเครื่องมือ 64
4.1 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ 64
4.1.1 ฮาร์ดแวร์ (Hardware) 64
4.1.2 ซอฟต์แวร์ (Software) 64
4.2 โครงสร้างส่วนต่อประสานกับผู้ใช้ของเครื่องมือ 64
บทที่ 5 การทดสอบเครื่องมือ 74
5.1 สภาพแวดล้อมที่ใช้ทดสอบ 74
5.1.1 ฮาร์ดแวร์ 74
5.1.2 ซอฟต์แวร์ 74
5.2 การทดสอบเครื่องมือ 74
5.3 ผลการทดสอบ 77
5.3.1 การทดสอบการตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการ 77
5.3.2 การทดสอบการสร้างกรณีทดสอบเพิ่มเติม 79
5.4 สรุปผลการทดสอบ 80
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ 81

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

6.1 สรุปผลการวิจัย 81
6.2 ข้อจำกัดของเครื่องมือ 81
รายการอ้างอิง 83
ภาคผนวก 84
ภาคผนวก ก พจนานุกรมข้อมูล 85
ภาคผนวก ข ตัวอย่างระบบทดสอบ 86
ภาคผนวก ค วิธีการใช้เครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ 121
ประวัติผู้เขียนวิทยานิพนธ์ 130

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญตาราง

หน้า

ตารางที่ 3.1 รายละเอียดยูสเคสการแทรกซอร์สโค้ด 26
ตารางที่ 3.2 รายละเอียดยูสเคสการวิเคราะห์ความครอบคลุมของการทดสอบ 26
ตารางที่ 3.3 รายละเอียดยูสเคสการสร้างกรณีทดสอบ 27
ตารางที่ 3.5 รายละเอียดยูสเคสการสกัดข้อมูล 28
ตารางที่ 5.1 ผลการทดสอบการตรวจสอบความครอบคลุมกับระบบจำลอง 78
ตารางที่ 5.2 ผลการทดสอบสร้างกรณีทดสอบเพิ่มเติมกับระบบจำลอง 79
ตารางที่ ก-1 พจนานุกรมข้อมูลตาราง classmethoddata 85
ตารางที่ ก-2 พจนานุกรมข้อมูลตาราง sdlocate 85
ตารางที่ ก-3 พจนานุกรมข้อมูลตาราง ucdlocate 85

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญภาพ

หน้า

ภาพที่ 2-1 ตัวอย่างกราฟระบุทิศทาง 7
ภาพที่ 2-2 กราฟการไหลของการควบคุมแบบต่างๆ 8
ภาพที่ 2-3 ตัวอย่างโปรแกรมและกราฟการไหลของการควบคุมของโปรแกรม 9
ภาพที่ 2-4 แฟ้มเอกสารเอกซ์เอ็มแอลของรายละเอียดสินค้า 10
ภาพที่ 2-5 สัญลักษณ์แสดงเฟรมของแผนภาพ 10
ภาพที่ 2-6 สัญลักษณ์ตัวแทนของแอคเตอร์ 11
ภาพที่ 2-7 รูปแบบและเครื่องหมายในการแสดงอ็อบเจกต์ 11
ภาพที่ 2-8 สัญลักษณ์แสดงไลฟ์ไลน์ของอ็อบเจกต์ 11
ภาพที่ 2-9 สัญลักษณ์แสดงเมจเสจที่ใช้ในการสื่อสารระหว่างอ็อบเจกต์ 12
ภาพที่ 2-10 สัญลักษณ์แสดงคอมบายด์แฟรกเมนต์ชนิดอัลเทอร์เนทีฟ 12
ภาพที่ 2-11 สัญลักษณ์แสดงคอมบายด์แฟรกเมนต์ชนิดออปชัน 13
ภาพที่ 2-12 สัญลักษณ์แสดงคอมบายด์แฟรกเมนต์ชนิดลูป 13
ภาพที่ 3-1 ภาพรวมการทำงานของเครื่องมือ 15
ภาพที่ 3-2 ลักษณะการแทรกซอร์สโค้ดเมื่อพบการสร้างคลาส 16
ภาพที่ 3-3 ตัวอย่างการแทรกซอร์สโค้ดของการสร้างคลาส 17
ภาพที่ 3-4 ตัวอย่างการแทรกซอร์สโค้ดของการสร้างเมท็อด 17
ภาพที่ 3-5 ตัวอย่างการแทรกซอร์สโค้ดของจุดสิ้นสุดของโปรแกรม 17
ภาพที่ 3-6 ตัวอย่างไฟล์สรุปการเรียกใช้เมท็อด 19
ภาพที่ 3-7 ตัวอย่างข้อมูลที่สกัดได้จากแผนภาพลำดับ 20
ภาพที่ 3-8 ตัวอย่างคำอธิบายยูสเคส 22
ภาพที่ 3-9 ขั้นตอนการทำงานของการสร้างกรณีทดสอบ 22
ภาพที่ 3-10 ตัวอย่างแผนภาพลำดับ 23
ภาพที่ 3-11 กราฟการทำงานของแผนภาพลำดับตัวอย่างภาพที่ 3-10 24

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

ภาพที่ 3-12 แผนภาพยูสเคสของเครื่องมือ 25
ภาพที่ 3-13 แผนภาพกิจกรรมการแทรกซอร์สโค้ด 29
ภาพที่ 3-14 แผนภาพกิจกรรมการวิเคราะห์ความครอบคลุมของการทดสอบ 33
ภาพที่ 3-15 แผนภาพกิจกรรมการวิเคราะห์ข้อมูล 35
ภาพที่ 3-16 แผนภาพกิจกรรมการสร้างกราฟการทำงาน 36
ภาพที่ 3-17 แผนภาพกิจกรรมการสร้างกรณีทดสอบ 40
ภาพที่ 3-18 แผนภาพคลาสของเครื่องมือ 44
ภาพที่ 3-19 คลาส About 45
ภาพที่ 3-20 คลาส CodeInstrumentation 45
ภาพที่ 3-21 คลาส CollectedClassMethodData 46
ภาพที่ 3-22 คลาส ConditionExtractformGraph 46
ภาพที่ 3-23 คลาส ConditionMNG 46
ภาพที่ 3-24 คลาส CollectedXMLData 47
ภาพที่ 3-25 คลาส CoverageAnalysisNewProject 48
ภาพที่ 3-26 คลาส CoverageAnalysisOpenProject 48
ภาพที่ 3-27 คลาส CoverageReportGen 48
ภาพที่ 3-28 คลาส csvextract 49
ภาพที่ 3-29 คลาส DatabaseConnection 49
ภาพที่ 3-30 คลาส Main_DataExtraction 49
ภาพที่ 3-31 คลาส GraphBuilder 50
ภาพที่ 3-32 คลาส MainPage 50
ภาพที่ 3-33 คลาส NewPJFrame 51
ภาพที่ 3-34 คลาส OpenPJFrame 52
ภาพที่ 3-35 คลาส RandomData 52

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

ภาพที่ 3-36 คลาส SaveRresult 52
ภาพที่ 3-37 คลาส ScanSourcedata 53
ภาพที่ 3-38 คลาส TestCaseGen 54
ภาพที่ 3-39 คลาส UIInterfaceApp 54
ภาพที่ 3-40 คลาส UIInterfaceGenTestCase 54
ภาพที่ 3-41 คลาส XMLParsing 55
ภาพที่ 3-42 แผนภาพลำดับของการแทรกซอร์สโค้ด 57
ภาพที่ 3-43 แผนภาพลำดับของการวิเคราะห์ความครอบคลุม 58
ภาพที่ 3-44 แผนภาพลำดับของการสกัดข้อมูล 59
ภาพที่ 3-45 แผนภาพลำดับของการติดต่อกับฐานข้อมูล 60
ภาพที่ 3-46 แผนภาพลำดับของการสร้างกราฟการทำงาน 61
ภาพที่ 3-47 แผนภาพลำดับของการสร้างกรณีทดสอบ 62
ภาพที่ 3-48 โครงสร้างฐานข้อมูลของเครื่องมือ 63
ภาพที่ 4-1 แผนภาพ Window Navigation 65
ภาพที่ 4-2 WELCOME 66
ภาพที่ 4-3 หน้าต่างหลักของเครื่องมือ 66
ภาพที่ 4-4 แถบหน้าต่าง Coverage Analysis 67
ภาพที่ 4-5 แถบหน้าต่าง Test Case Generation 68
ภาพที่ 4-6 หน้าต่าง Code Instrumentation 69
ภาพที่ 4-7 Instrumentation Status 69
ภาพที่ 4-8 หน้าต่าง Coverage Analysis 70
ภาพที่ 4-9 หน้าจอ Save Result 71
ภาพที่ 4-10 หน้าต่าง Open Project 72
ภาพที่ 4-11 หน้าต่าง About 72

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

ภาพที่ 4-12 หน้าต่าง Test case generation Status 73
ภาพที่ 4-13 Coverage report status 73
ภาพที่ ข-1 แผนภาพลำดับของระบบทดสอบการคิดเงินในร้านค้า 86
ภาพที่ ข-2 ตัวอย่างกรณีทดสอบระบบคิดเงินในร้านค้า กรณีทดสอบที่ 1 87
ภาพที่ ข-3 ตัวอย่างกรณีทดสอบระบบคิดเงินในร้านค้า กรณีทดสอบที่ 2 88
ภาพที่ ข-4 ตัวอย่างกรณีทดสอบระบบคิดเงินในร้านค้า กรณีทดสอบที่ 3 88
ภาพที่ ข-5 แผนภาพลำดับของระบบทดสอบการซื้อตั๋วรถประจำทาง 89
ภาพที่ ข-6 ตัวอย่างกรณีทดสอบระบบซื้อตั๋วรถประจำทาง กรณีทดสอบที่ 1 90
ภาพที่ ข-7 ตัวอย่างกรณีทดสอบระบบซื้อตั๋วรถประจำทาง กรณีทดสอบที่ 2 90
ภาพที่ ข-8 แผนภาพลำดับของระบบทดสอบการจองห้องพักของโรงแรม 91
ภาพที่ ข-9 ตัวอย่างกรณีทดสอบระบบจองห้องพักของโรงแรม กรณีทดสอบที่ 1 94
ภาพที่ ข-10 ตัวอย่างกรณีทดสอบระบบจองห้องพักของโรงแรม กรณีทดสอบที่ 2 94
ภาพที่ ข-11 แผนภาพลำดับของระบบจำลองที่ 1 95
ภาพที่ ข-12 กรณีทดสอบของระบบจำลองที่ 1 98
ภาพที่ ข-13 แผนภาพลำดับของระบบจำลองที่ 2 99
ภาพที่ ข-14 กรณีทดสอบของระบบจำลองที่ 2 102
ภาพที่ ข-15 แผนภาพลำดับของระบบจำลองที่ 3 103
ภาพที่ ข-16 กรณีทดสอบของระบบจำลองที่ 3 105
ภาพที่ ข-17 แผนภาพลำดับของระบบจำลองที่ 4 106
ภาพที่ ข-18 กรณีทดสอบของระบบจำลองที่ 4 107
ภาพที่ ข-19 แผนภาพลำดับของระบบจำลองที่ 5 108
ภาพที่ ข-20 กรณีทดสอบของระบบจำลองที่ 5 110
ภาพที่ ข-21 แผนภาพลำดับของระบบจำลองที่ 6 111
ภาพที่ ข-22 กรณีทดสอบของระบบจำลองที่ 6 113

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หน้า

ภาพที่ ข-23 แผนภาพลำดับของระบบจำลองที่ 7 114
ภาพที่ ข-24 กรณีทดสอบของระบบจำลองที่ 7 115
ภาพที่ ข 25 แผนภาพลำดับของระบบจำลองที่ 8 116
ภาพที่ ข-25 กรณีทดสอบของระบบจำลองที่ 8 117
ภาพที่ ข-26 แผนภาพลำดับของระบบจำลองที่ 9 117
ภาพที่ ข-27 กรณีทดสอบของระบบจำลองที่ 9 118
ภาพที่ ข-28 แผนภาพลำดับของระบบจำลองที่ 10 119
ภาพที่ ข-29 กรณีทดสอบของระบบจำลองที่ 10 120
ภาพที่ ค-1 หน้าต่างการสร้างโปรเจคใหม่ 121
ภาพที่ ค-2 หน้าต่างการแทรกซอร์สโค้ด 122
ภาพที่ ค-3 หน้าต่างการแทรกซอร์ซโค้ดที่ระบุซอร์สโค้ดที่ต้องการแทรกแล้ว 122
ภาพที่ ค-4 หน้าต่างแจ้งสถานะการแทรกซอร์สโค้ดเมื่อเครื่องมือแทรกซอร์สโค้ดเสร็จแล้ว 123
ภาพที่ ค-5 หน้าต่าง Coverage Analysis 123
ภาพที่ ค-6 หน้าต่างแสดงผลการตรวจสอบความครอบคลุมของการทดสอบแบบบูรณาการ 124
ภาพที่ ค-7 หน้าต่างบันทึกผลการตรวจสอบความครอบคลุมการมดสอบ 124
ภาพที่ ค-8 หน้าต่างแจ้งสถานการณ์บันทึกผลการตรวจสอบความครอบคลุม เมื่อบันทึกผลเสร็จแล้ว 125
ภาพที่ ค-9 ตัวอย่างรายงานการตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการ 125
ภาพที่ ค-10 หน้าต่าง Open Project 126
ภาพที่ ค-11 หน้าต่าง Coverage analysis 126
ภาพที่ ค-12 หน้าต่างแสดงผลการตรวจสอบความครอบคลุมเมื่อทดสอบเพิ่มเติมกับกรณีทดสอบที่เครื่องมือสร้างขึ้น 127
ภาพที่ ค-13 แถบหน้าต่าง Test case Generation 128
ภาพที่ ค-14 แถบหน้าต่าง Test case Generation 128

หน้า

ภาพที่ ค-15 หน้าต่างแสดงสถานะเมื่อเครื่องมือสร้างกรณีทดสอบเสร็จแล้ว 129

ภาพที่ ค-16 ตัวอย่างกรณีทดสอบที่ระบบสร้างขึ้น..... 129

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของงานวิจัย

การทดสอบซอฟต์แวร์เป็นขั้นตอนสำคัญในกระบวนการพัฒนาซอฟต์แวร์ เพื่อทวนสอบความถูกต้องตรงกับความต้องการของผู้ใช้ รวมทั้งค้นหาข้อบกพร่อง (Defect) ที่อาจเกิดขึ้นในซอฟต์แวร์ก่อนที่ จะส่งมอบให้กับผู้ใช้ ซึ่งการทดสอบซอฟต์แวร์นั้นประกอบไปด้วย 3 ระดับคือ การทดสอบระดับหน่วย (Unit testing) การทดสอบแบบบูรณาการ (Integration testing) และการทดสอบระบบ (System testing) สิ่งสำคัญอย่างหนึ่งของการทดสอบซอฟต์แวร์คือ ความครอบคลุมของการทดสอบ (Test coverage) ซึ่งถ้าการทดสอบซอฟต์แวร์นั้นไม่ครอบคลุมก็จะเกิดผลเสียหายตามมาได้เนื่องจากอาจจะมีข้อบกพร่องที่แฝงอยู่ในส่วนที่ไม่ครอบคลุม และพบภายหลังจากผู้ใช้ ได้ใช้งานในส่วนที่ไม่ครอบคลุมนั้น

ในปัจจุบันการดำเนินการพัฒนาซอฟต์แวร์เชิงวัตถุได้มีบทบาทเพิ่มขึ้น ซึ่งต่างจากเดิมที่พัฒนาซอฟต์แวร์ในรูปแบบของสัญนิยม (Conventional software) ทำให้การทดสอบซอฟต์แวร์ในรูปแบบเดิมไม่สามารถใช้ร่วมกันได้ ดังนั้นเทคนิคต่างๆ รวมทั้งเครื่องมือที่ช่วยในการทดสอบซอฟต์แวร์ที่พัฒนาในรูปแบบสัญนิยม จึงไม่สามารถนำมาใช้กับการทดสอบซอฟต์แวร์เชิงวัตถุได้

จากงานวิจัย “A State-Based Approach to Integration Testing Based on UML Models”[1] ได้กล่าวถึง การทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุว่า เป็นการทดสอบที่เกี่ยวข้องกับการปฏิสัมพันธ์กัน (Interaction) ระหว่างหน่วย (Unit) ของซอฟต์แวร์เพื่อตรวจสอบว่าสามารถทำงานร่วมกันได้ตรงตามความต้องการเชิงฟังก์ชันหรือไม่ ซึ่งการทดสอบระดับนี้ถือว่ามีความจำเป็นเนื่องจากข้อบกพร่องบางอย่างไม่ได้แสดงออกมาระหว่างการทดสอบระดับหน่วย แต่จะเกิดขึ้นมาระหว่างการทำงานร่วมกันของหน่วยย่อยของซอฟต์แวร์ โดยหน่วยย่อยของซอฟต์แวร์เชิงวัตถุนั้นสามารถเป็นได้ทั้งระดับเมท็อดและระดับคลาส เมื่อได้ศึกษางานวิจัยอื่นๆเพิ่มเติมพบว่า ในขณะนี้ยังไม่มีเครื่องมือที่ใช้ในการวิเคราะห์ความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ รวมทั้งยังไม่มีเครื่องมือใดที่สร้างกรณีทดสอบเพิ่มเติม เมื่อวิเคราะห์ความครอบคลุมการทดสอบแล้วพบว่าการทดสอบนั้นยังไม่ครอบคลุม

ดังนั้นงานวิจัยนี้จึงเสนอการพัฒนาเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ โดยตรวจสอบคลาสและเมท็อดที่ถูกทดสอบ แล้วแสดงผลออกมาเป็นสัดส่วนของจำนวนคลาสที่ครอบคลุมจากการทดสอบต่อจำนวนคลาสที่ทดสอบทั้งหมด และสัดส่วนของเมท็อดที่ครอบคลุมจากการทดสอบต่อจำนวนเมท็อดที่ทดสอบทั้งหมด หลังจากนั้นสร้างกรณีทดสอบเพิ่มเติมในกรณีที่มีคลาสหรือเมท็อดที่ไม่ครอบคลุมจากการทดสอบ เพื่อให้เกิดความครอบคลุมของการทดสอบแบบบูรณาการ โดยสร้างกราฟการทำงานจากแผนภาพลำดับ (Sequence diagram) แล้วหาเส้นทางการทำงานที่มีเมท็อดที่ไม่ครอบคลุมอยู่ในเส้นทาง แล้วจึงนำข้อมูลจากคำอธิบายยูสเคสมาใช้ในการสร้างข้อมูลทดสอบ

1.2 วัตถุประสงค์ของงานวิจัย

1. เพื่อออกแบบ และพัฒนาเครื่องมือช่วยตรวจสอบความครอบคลุมของการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ

2. สร้างกรณีทดสอบสำหรับการทดสอบแบบบูรณาการเพิ่มเติมในกรณีที่พบความไม่ครอบคลุมของการทดสอบ

1.3 ขอบเขตของงานวิจัย

1. ความสามารถของเครื่องมือมีดังต่อไปนี้

1.1 เครื่องมือนี้สามารถตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุได้

1.2 เครื่องมือสามารถสร้างกรณีทดสอบเพิ่มเติมได้เมื่อตรวจสอบพบความไม่ครอบคลุมของการทดสอบแบบบูรณาการ

2. เครื่องมือสามารถนำเข้าข้อมูลได้ดังต่อไปนี้

2.1 ซอร์สโค้ดของระบบที่ต้องการนำมาทดสอบ โดยเครื่องมือสนับสนุนซอร์สโค้ดภาษาจาวา

2.2 กรณีทดสอบ

3. ผลลัพธ์ของเครื่องมือจะแสดงรายละเอียดดังต่อไปนี้เป็นอย่างน้อย

3.1 จำนวนสัดส่วนของคลาสที่ครอบคลุมจากการทดสอบต่อจำนวนคลาสทั้งหมดคิดเป็นเปอร์เซ็นต์

3.2 จำนวนสัดส่วนของเมท็อดที่ครอบคลุมจากการทดสอบต่อจำนวนเมท็อดทั้งหมดคิดเป็นเปอร์เซ็นต์

4. ในกรณีที่ต้องมีการสร้างกรณีทดสอบเพิ่มเติมนั้น จะต้องใช้ข้อมูลนำเข้าเพิ่มเติมดังต่อไปนี้

4.1 แผนภาพลำดับ โดยต้องอยู่ในรูปแบบเอกสารเอกซ์เอ็มแอลที่ถูกบันทึกด้วยโปรแกรมวิชวลพาราไดม์เวอร์ชัน 5.0

4.2 คำอธิบายยูสเคส ต้องอยู่ในรูปแบบไฟล์ซีเอสวี (CSV file : Comma Separate Value file)โดยข้อมูลที่ปรากฏในคำอธิบายยูสเคสรายละเอียดดังต่อไปนี้เป็นอย่างน้อย

4.2.1 หมายเลขแผนภาพลำดับที่เกี่ยวข้อง

4.2.2 ข้อความแสดงการทำงาน

4.2.3 เงื่อนไขที่ทำให้ลำดับการทำงานปกติของแผนภาพลำดับนั้นสามารถทำงานได้และผลลัพธ์

4.2.4 ข้อมูลนำเข้า

5. ในกรณีที่ต้องมีการสร้างกรณีทดสอบเพิ่มเติม กรณีทดสอบจะอยู่ในรูปแบบเอกสารเอชทีเอ็มแอล (HTML) ซึ่งจะแสดงรายละเอียดดังต่อไปนี้

5.1 หมายเลขกรณีทดสอบ

5.2 ชื่อกรณีทดสอบ

5.3 คำอธิบายกรณีทดสอบ

5.4 ข้อมูลเริ่มต้นของกรณีทดสอบ

5.4 ข้อมูลทดสอบ

5.5 ผลลัพธ์ที่คาด

6. คลาสที่ถือว่าครอบคลุมจากการทดสอบคือคลาสที่ทุกเมท็อดที่เป็นสมาชิกของคลาสถูกเรียกอย่างน้อย 1 ครั้ง เมื่อผ่านการทดสอบ

7. เมท็อดที่ถือว่าครอบคลุมจากการทดสอบคือเมท็อดที่ถูกเรียกอย่างน้อย 1 ครั้งเมื่อผ่านการทดสอบ

8. แผนภาพลำดับต้องอยู่ในรูปแบบที่ถูกต้องสมบูรณ์ตามข้อกำหนดยูเอ็มแอล

9. เงื่อนไขที่ปรากฏในคอมบายด์แฟรกเมนต์ที่ระบบรองรับมี 3 ชนิดคือ อัลเทอร์เนทีฟ, ออปชัน, และลูป

10. ชนิดของข้อมูลนำเข้าที่ปรากฏในคำอธิบายยูสเคสที่ระบบรองรับมี 5 ชนิดคือ ข้อมูลจำนวนเต็ม, ชนิดข้อมูลตัวเลขจำนวนจริง, ข้อมูลชุดอักขระ, ข้อมูลชนิดตรรกะ

11. กรณีทดสอบที่นำเข้าสู่ระบบต้องถูกสร้างโดยผู้ทดสอบเอง

12. เครื่องมือนี้เป็นวินโดว์แอปพลิเคชันซึ่งพัฒนาด้วยภาษาจาวา

13. เครื่องมือที่พัฒนาเสร็จแล้วจะถูกนำไปทดสอบกับซอฟต์แวร์เชิงวัตถุและกรณีทดสอบของโปรแกรมทดสอบ อย่างน้อย 3 ระบบงาน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

1.4 ขั้นตอนและวิธีการดำเนินงานวิจัย

1. ศึกษาโครงสร้างของซอฟต์แวร์เชิงวัตถุ

2. ศึกษาวิธีการตรวจสอบความครอบคลุมของการทดสอบแบบบูรณาการของซอฟต์แวร์เชิง-วัตถุ

3. ศึกษาวิธีการสร้างกรณีทดสอบจากแผนภาพลำดับ

4. กำหนดคุณลักษณะและขอบเขตความสามารถของเครื่องมือ

5. ออกแบบโครงสร้างและหน้าจอของเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ

6. พัฒนาเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิง-วัตถุ ตามที่ได้ออกแบบไว้

7. ทดสอบเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิง-วัตถุ

8. สรุปผลการวิจัยและข้อเสนอแนะ

9. จัดทำรายงานวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เครื่องมือช่วยตรวจสอบความครอบคลุมของการทดสอบแบบบูรณาการของซอฟต์แวร์เชิง-วัตถุที่สามารถใช้งานได้จริง

2. ช่วยเพิ่มประสิทธิภาพในการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ

3. เพิ่มความน่าเชื่อถือของซอฟต์แวร์เชิงวัตถุที่ผ่านการทดสอบด้วยเครื่องมือ

1.6 ลำดับขั้นตอนในการนำเสนองานวิจัย

วิทยานิพนธ์ฉบับนี้แบ่งเนื้อหาออกเป็น 6 บทดังต่อไปนี้ บทที่ 1 กล่าวถึงความเป็นมาและความสำคัญของปัญหา รวมทั้งวัตถุประสงค์ของงานวิจัย ขอบเขตของงานวิจัย ขั้นตอนและวิธีการดำเนินการวิจัย และ ประโยชน์ที่คาดว่าจะได้รับ บทที่ 2 กล่าวถึงทฤษฎีที่เกี่ยวข้องรวมถึงงานวิจัยที่เกี่ยวข้อง บทที่ 3 นำเสนอการวิเคราะห์และออกแบบเครื่องมือ รวมทั้งภาพรวมการทำงานของเครื่องมือ บทที่ 4 การพัฒนาเครื่องมือ บทที่ 5 การทดสอบและสภาพแวดล้อมการทดสอบ และบทที่ 6 บทสรุปผลการวิจัยและข้อเสนอแนะ

Chulalongkorn University

1.7 ผลงานตีพิมพ์จากงานวิจัย

งานวิจัยนี้ได้รับการตีพิมพ์เป็นบทความวิชาการดังต่อไปนี้

1. บทความวิชาการเรื่อง “Design of a Tool for Checking Integration Testing Coverage of Object-Oriented Software” โดย Pachawan Augsornsri and Taratip Suwannasart ได้รับคัดเลือกและตีพิมพ์ในการประชุมวิชาการ “2013 International Conference on Information Science and Applications (ICISA 2013)” ระหว่างวันที่ 24-26 มิถุนายน 2556 ณ โรงแรมฮิลตัน พัทยา จังหวัดชลบุรี ประเทศไทย

2. บทความวิชาการเรื่อง “An Integration Testing Coverage Tool for Object-Oriented Software” Pachawan Augsornsri and Taratip Suwannasart ได้รับคัดเลือกและตีพิมพ์ในการประชุมวิชาการ “2014 5th International Conference on Information Science and Applications (ICISA 2014)” ระหว่างวันที่ 6-9 พฤษภาคม 2557 ณ มหาวิทยาลัยสตรีอีฮวา กรุงโซล ประเทศสาธารณรัฐเกาหลี

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

การพัฒนาเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุนั้น ผู้วิจัยได้ศึกษาทฤษฎีต่างๆที่เกี่ยวข้อง อันได้แก่ การทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ ความครอบคลุมการทดสอบ กราฟระบุทิศทางและกราฟการไหลของการควบคุม เอกซ์เอ็มแอล (XML : eXtensible Markup Language) แผนภาพลำดับ ซึ่งมีรายละเอียดดังต่อไปนี้

2.1.1 การทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ

การทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ[2] เป็นการทดสอบที่เกี่ยวข้องกับการทำงานร่วมกันระหว่างหน่วยย่อยของซอฟต์แวร์ ซึ่งหน่วยย่อยต่างๆนั้นจำเป็นจะต้องถูกทดสอบระดับหน่วยเรียบร้อยแล้ว จุดที่สนใจของการทดสอบในระดับนี้คือ การทดสอบการทำงานร่วมกันระหว่างคลาสผ่านการเรียกเมท็อดหรือสัญญาณอะซิงโครนัส ตลอดจนการติดต่อกับฐานข้อมูล การทดสอบระดับนี้ถือว่ามีความจำเป็นเพราะข้อบกพร่องบางอย่างไม่ได้แสดงออกมาระหว่างการทดสอบระดับหน่วย แต่จะเกิดขึ้นมาระหว่างการทำงานร่วมกันของหน่วยย่อยของซอฟต์แวร์

ในส่วนของความครอบคลุมของการทดสอบ เนื่องจากซอฟต์แวร์ที่ถูกทดสอบแบบบูรณาการจำเป็นต้องผ่านการทดสอบระดับหน่วยแล้ว และการทดสอบนี้มุ่งเน้นไปที่การทำงานร่วมกันของแต่ละหน่วยของซอฟต์แวร์ ดังนั้น ความครอบคลุมของการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุจึงเป็นการพิจารณาการเรียกใช้ของแต่ละหน่วยย่อย

2.1.2 ความครอบคลุมการทดสอบ

ความครอบคลุมการทดสอบ[3] เป็นตัววัดการประกันคุณภาพที่พิจารณาความครอบคลุมการดำเนินการทดสอบของชุดทดสอบที่มีต่อโปรแกรม การทดสอบโดยใช้พื้นฐานความครอบคลุมนี้สามารถทำได้ในทุกระดับการทดสอบ ตั้งแต่การทดสอบระดับหน่วย การทดสอบแบบบูรณาการรวมถึงการทดสอบระบบ ผลลัพธ์ที่ได้จากการวัดความครอบคลุมจะบอกถึงจุดที่การทดสอบไม่ครอบคลุม ซึ่งสามารถนำไปใช้ในการปรับปรุงกระบวนการทวนสอบ นอกจากนี้ ยังสามารถนำไปช่วยในการทดสอบแบบถดถอย การจัดลำดับความสำคัญของกรณีทดสอบ รวมทั้งการเพิ่มหรือลดชุดทดสอบ

การวิเคราะห์ความครอบคลุมการทดสอบ[3, 4] เป็นเทคนิคการทดสอบแบบกล่องขาว สามารถแบ่งย่อยขั้นตอนได้อีก 3 ขั้นตอนคือ

1) การแทรกซอร์สโค้ด (Code instrumentation) : เป็นการใส่ซอร์ซโค้ดเพิ่มเติมเพื่อวัดผลความครอบคลุมของการทดสอบ ซึ่งสามารถแทรกซอร์ซโค้ดลงในซอร์ซโค้ดก่อนที่จะดำเนินการทดสอบ หรือแทรกซอร์สโค้ดในไบท์โค้ดในช่วงเวลารันไทม์

2) การรวบรวมข้อมูล (Data gathering) : เป็นการเก็บข้อมูลความครอบคลุมระหว่างการดำเนินการทดสอบ

3) การวิเคราะห์ความครอบคลุม (Coverage analysis) : ประกอบไปด้วยการวิเคราะห์ข้อมูลที่สามารถเก็บได้จากขั้นตอนการรวบรวมข้อมูล และหากลยุทธ์ในการทดสอบเพื่อปรับเปลี่ยน เพิ่มหรือลดชุดทดสอบ

2.1.3 กราฟระบุทิศทางและกราฟการไหลของการควบคุม (Direct graph and control flow graph)

1) กราฟระบุทิศทาง (Direct graph) [5]

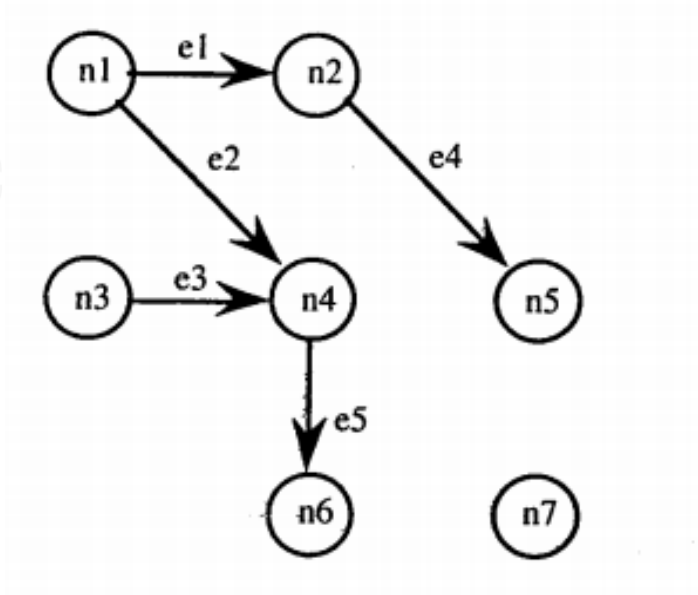
กราฟระบุทิศทาง เรียกอีกอย่างหนึ่งว่าไดกราฟ(Digraph) มีนิยามคือ D = (V,E) ประกอบไปด้วยเซตจำกัดของโหนด (Node)

$V = \{n_1,n_2,n_3,...,n_m,\}$

และเซตจำกัดของเส้นเชื่อม (Edge)

$E = \{e_1,e_2,e_3,...,e_p\}$

โดยที่แต่ละเส้น $e_k = <n_i,n_j>$ คือคู่ที่เรียงลำดับของโหนด n_i,n_j ซึ่งทั้งสองโหนดเป็นสมาชิกของเซต V ตัวอย่างของกราฟระบุทิศทาง แสดงได้ดังภาพที่ 2-1 [6]



ภาพที่ 2-1 ตัวอย่างกราฟระบุทิศทาง

2) กราฟการไหลของการควบคุม (Control flow graph) [5, 6]

กราฟการไหลของการควบคุม รู้จักกันในอีกชื่อหนึ่งว่ากราฟกระแสการไหล (Flow graph) หรือโปรแกรมกราฟ (Program graph) คือ กราฟระบุทิศทางที่มีโหนดแทนข้อความสั่ง (Statement) หรือส่วนของข้อความสั่ง (Fragment of statement) และเส้นแทนกระแสการไหลของการควบคุม กราฟการไหลของการควบคุมแบบต่างๆ แสดงได้ดังภาพที่ 2-2 โดยที่

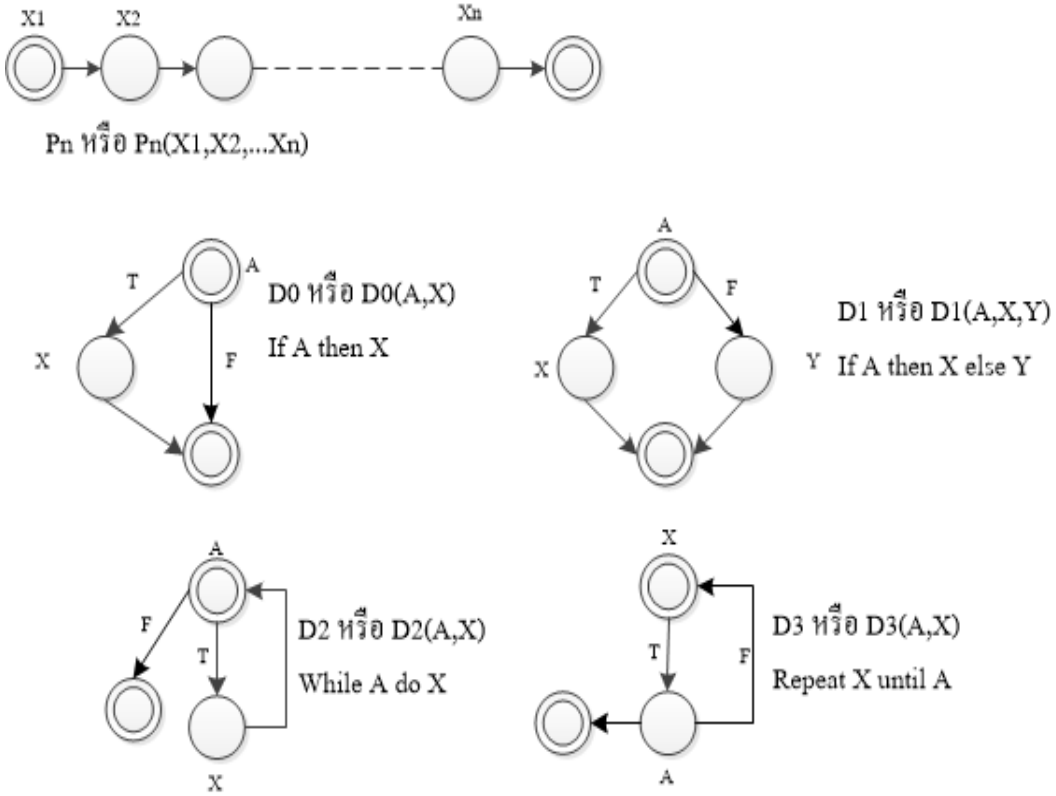
2.1) Pn หรือ Pn(X1,X2,...Xn) แสดงกราฟการไหลของการควบคุมแบบลำดับ

2.2) D0 หรือ D0(A,X) แทนคำสั่ง if A then X

2.3) D1 หรือ D1(A,X,Y) แทนคำสั่ง if A then X else Y

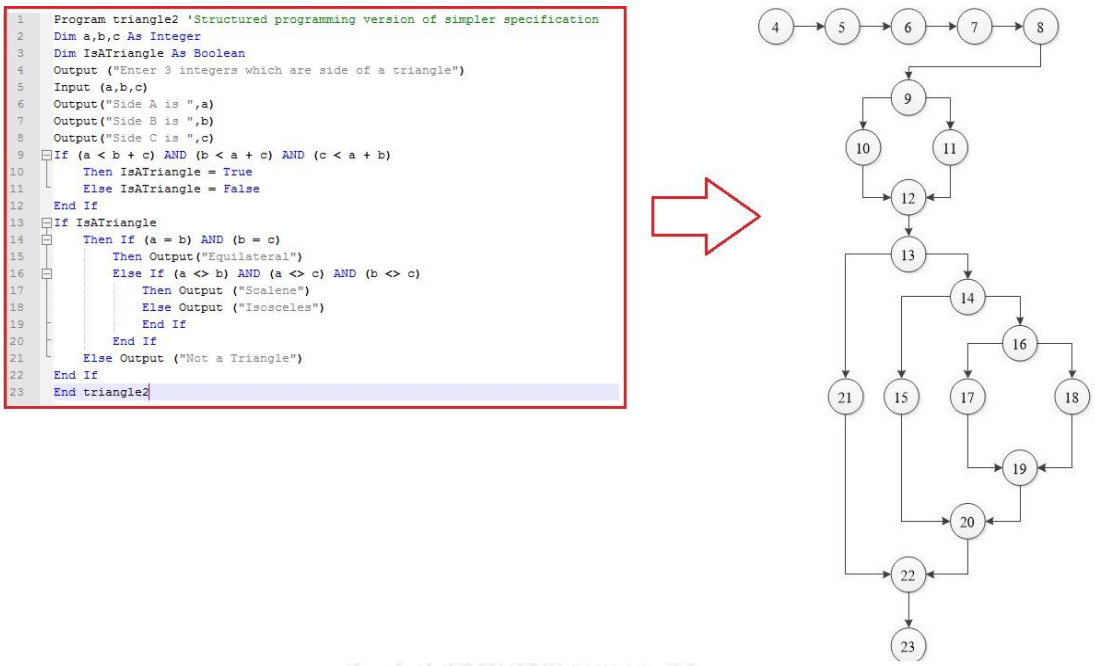
2.4) D2 หรือ D2(A,X) แทนคำสั่ง while A do X

2.5) D3 หรือ D3(A,X) แทนคำสั่ง repeat A until X



ภาพที่ 2-2 กราฟการไหลของการควบคุมแบบต่างๆ

โดยตัวอย่างของโปรแกรมและกราฟการไหลของการควบคุมของโปรแกรมสามารถแสดงได้ดังภาพที่ 2-3



ภาพที่ 2-3 ตัวอย่างโปรแกรมและกราฟการไหลของการควบคุมของโปรแกรม

2.1.4 เอกซ์เอ็มแอล (XML : eXtensible Markup Language)

เอกซ์เอ็มแอล[7, 8] ย่อมาจากคำว่า Extensible Markup Language เป็นมาตรฐานที่ได้รับการรับรองจาก W3C หรือ World Wide Web Consortium ใช้สำหรับเขียนเอกสารมาร์คอัพ เพื่อกำหนดวากยสัมพันธ์ (Syntax) ของการมาร์คอัพข้อมูลด้วยป้ายระบุ (Tags) ที่มนุษย์สามารถเข้าใจได้โดยง่ายและยืดหยุ่นเพียงพอที่จะปรับแต่งได้สำหรับใช้ในหลายๆโดเมนเช่น เว็บไซต์ หรือระบบไปรษณีย์เสียง (Voice mail) เนื่องจากเอกซ์เอ็มแอลเป็นภาษาเมตามาร์คอัพ (Metamarkup language) จึงไม่มีการกำหนดชุดของป้ายระบุที่ตายตัวเพื่อให้ทำงานได้กับทุกสายงาน ทุกคน และทุกเวลา

ส่วนประกอบในเอกสารเอกซ์เอ็มแอลมีอยู่ 2 ส่วนหลักคือ โปรล็อกอีลีเมนต์ (Prolog element) และเอกสารส่วนย่อย (Document element หรือ Root element) ในส่วนของเอกสารเอกซ์เอ็มแอลคืออีลีเมนต์เดี่ยว ซึ่งสามารถบรรจุอีลีเมนต์เพิ่มเติมในเอกสารเอกซ์เอ็มแอลได้ โดยในเอกสารเอกซ์เอ็มแอลนั้น อีลีเมนต์จะแสดงลักษณะโครงสร้างของเอกสาร และจะแสดงส่วนประกอบเนื้อหาของเอกสารอยู่ภายในสัญลักษณ์อีลีเมนต์ ประกอบไปด้วย ป้ายระบุเริ่มต้น (Start-tags) เนื้อหาภายในอีลีเมนต์ และป้ายระบุสิ้นสุด (End-tags) ส่วนเนื้อหาภายในอีลีเมนต์สามารถเป็นได้ทั้งข้อมูลหรืออีลีเมนต์อื่นๆ ที่ซ้อนอยู่ภายในหรือทั้งสองแบบ ตัวอย่างของแฟ้มเอกสารเอกซ์เอ็มแอลแสดงดังภาพที่ 2-4

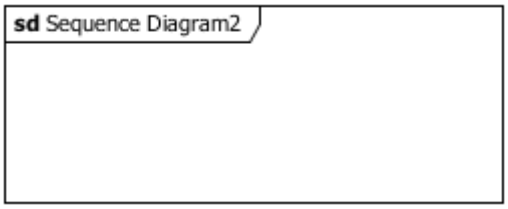
```
<?xml version="1.0"?>
<product barcode="2394287410">
  <manufacturer>Verbatim</manufacturer>
  <name>DataLife MF 2HD</name>
  <quantity>10</quantity>
  <size>3.5"</size>
  <color>black</color>
  <description>floppy disks</description>
</product>
```

ภาพที่ 2-4 แฟ้มเอกสารเอกซ์เอ็มแอลของรายละเอียดสินค้า

2.1.5 แผนภาพลำดับ

แผนภาพลำดับ [9, 10] เป็นแผนภาพชนิดหนึ่งของแผนภาพแสดงการโต้ตอบ (Interaction diagram) โดยจะแสดงอ็อบเจกต์ที่มีอยู่ในยูสเคสและแมสเสจ (Message) ที่ส่งระหว่างอ็อบเจกต์ในแต่ละช่วงเวลาและลำดับการส่งอย่างชัดเจน เนื่องจากแผนภาพลำดับนี้ให้ความสำคัญกับการเรียงลำดับของการทำกิจกรรมของแต่ละอ็อบเจกต์ จึงเป็นถูกนำมาใช้ในการวิเคราะห์ยูสเคสที่ซับซ้อนและเรียลไทม์ แผนภาพลำดับสามารถแสดงได้ทั้งในรูปแบบแผนภาพลำดับทั่วไปที่แสดงทุกซิแนริโอที่เป็นไปได้ทั้งหมดของยูสเคสลงในแผนภาพลำดับ แต่โดยปกติแล้วนักวิเคราะห์ระบบจะแสดงแผนภาพลำดับโดยใช้เป็นชุดของซิแนริโอแต่ละกรณีที่เป็นไปได้ โดยส่วนประกอบของแผนภาพลำดับประกอบด้วยหลายส่วนซึ่งสัญลักษณ์ของแผนภาพลำดับและความหมายแสดงได้ดังต่อไปนี้

1) เฟรม (Frame) เป็นสัญลักษณ์ที่ใช้กำหนดขอบเขตของภาพ โดยใช้เครื่องหมายสี่เหลี่ยมและมีลาเบลของภาพ (Diagram's label) ปรากฏอยู่ในเครื่องหมายสี่เหลี่ยมพับมุม (Dog-eared rectangle) โดยวางที่ตำแหน่งบนซ้ายสุดของเฟรม ดังภาพที่ 2-5

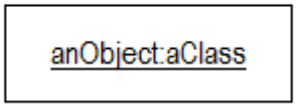


ภาพที่ 2-5 สัญลักษณ์แสดงเฟรมของแผนภาพ

2) แอคเตอร์ (Actor) คือบุคคลหรือระบบภายนอกซึ่งโต้ตอบกับระบบโดยการส่งเมสเสจแทนด้วยสัญลักษณ์รูปคน ดังภาพที่ 2-6

ภาพที่ 2-6 สัญลักษณ์ตัวแทนของแอคเตอร์

3) อ็อบเจกต์ (Object) จะถูกแสดงอยู่ในเครื่องหมายรูปสี่เหลี่ยม ซึ่งแสดงถึงอ็อบเจกต์ที่ถูกสร้างขึ้นจากคลาสที่ประกาศอยู่หลังเครื่องหมายมหัพภาค (:) ซึ่งสัญลักษณ์นี้จะถูกวางไว้ส่วนบนสุดของภาพ ดังภาพที่ 2-7



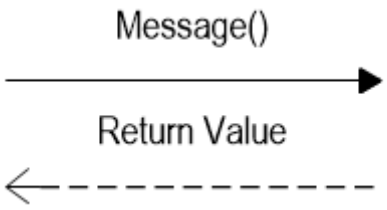
ภาพที่ 2-7 รูปแบบและเครื่องหมายในการแสดงอ็อบเจกต์

4) ไลฟ์ไลน์ (Lifeline) เป็นเครื่องหมายที่แสดงถึงช่วงระยะเวลาที่มีชีวิตอยู่ของอ็อบเจกต์ ในการรับหรือส่งเมสเสจ โดยถูกแสดงด้วยเส้นประแนวตั้ง ดังภาพที่ 2-8

ภาพที่ 2-8 สัญลักษณ์แสดงไลฟ์ไลน์ของอ็อบเจกต์

5) เมสเสจ (Message) เป็นตัวส่งข้อมูลจากอ็อบเจกต์หนึ่งไปยังอีกอ็อบเจกต์หนึ่ง โดยแบ่งออกเป็นสองลักษณะคือ การเรียกใช้และการคืนค่า(Return value) ซึ่งถูกแสดงด้วยลูกศรเส้นทึบและลูกศรเส้นประตามลำดับ ดังภาพที่ 2-9

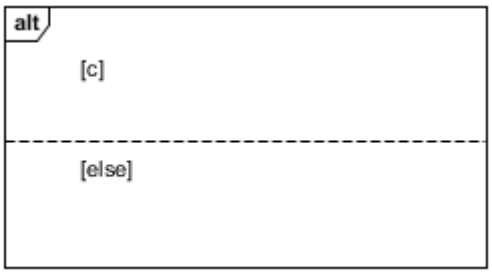
จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



ภาพที่ 2-9 สัญลักษณ์แสดงเมจเสจที่ใช้ในการสื่อสารระหว่างอ็อบเจกต์

6) คอมบายด์แฟรกเมนต์ (Combined fragment) เป็นส่วนที่ใช้ในการรวมกลุ่มของเมสเสจเข้าไว้ด้วยกัน เพื่อแสดงถึงกระแสอย่างมีเงื่อนไข (Conditional flow) เพื่อให้แบบจำลองสามารถกำหนดรูปแบบของตรรกศาสตร์ที่ต้องการให้ระบบดำเนินการได้ โดยในส่วนคอมบายด์แฟรกเมนต์ที่ได้ถูกกำหนดไว้ในยูเอ็มแอลรุ่นที่ 2.0 มีอยู่ 11 ประเภท ซึ่งในที่นี้จะอธิบายรายละเอียดเพียง 3 ประเภทเท่านั้น ได้แก่ อัลเทอร์เนทีฟ ออบชัน ลูป มีรายละเอียดดังต่อไปนี้

6.1) อัลเทอร์เนทีฟ (Alternative) เป็นคอมบายด์แฟรกเมนต์ที่ใช้เพื่อแสดงถึงทางเลือกของพฤติกรรม ในรูปแบบของ “if then else” คือกำหนดทางเลือกของเหตุการณ์ที่ไม่เกิดร่วมกัน ตัวอย่างคอมบายแฟรกเมนต์ประเภทอัลเทอร์เนทีฟ แสดงได้ดังภาพที่ 2-10



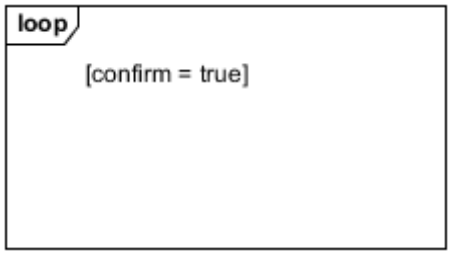
ภาพที่ 2-10 สัญลักษณ์แสดงคอมบายด์แฟรกเมนต์ชนิดอัลเทอร์เนทีฟ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

6.2) ออบชัน (Option) เป็นคอมบายด์แฟรกเมนต์ที่ใช้เพื่อแสดงถึงทางเลือกของพฤติกรรมที่อาจเกิดขึ้นหรือไม่เกิดขึ้นก็ได้ ซึ่งคล้ายกับคอมบายด์แฟรกเมนต์ประเภทอัลเทอร์เนทีฟ ต่างกันตรงที่คอมบายด์แฟรกเมนต์ประเภทออบชันจะมีเพียงหนึ่งตัวถูกดำเนินการเท่านั้น ตัวอย่างคอมบายแฟรกเมนต์ประเภทออบชัน แสดงได้ดังภาพที่ 2-11

ภาพที่ 2-11 สัญลักษณ์แสดงคอมบายด์แฟรกเมนต์ชนิดออบชัน

6.3) ลูป (Loop) เป็นคอมบายด์แฟรกเมนต์ที่ใช้แสดงถึงการวนทำซ้ำของเมจเสจภายในตัวถูกดำเนินการ การ์ดภายในลูปนอกจากเป็นเงื่อนไขบูลีนตามปกติแล้ว ยังสามารถประกอบไปด้วยจำนวนเงื่อนไขการทำซ้ำที่น้อยที่สุดและจำนวนเงื่อนไขการทำซ้ำที่มากที่สุด (minint และ maxint ตามลำดับ) เท่านั้น ตัวอย่างคอมบายแฟรกเมนต์ประเภทออบชัน แสดงได้ดังภาพที่ 2-12



ภาพที่ 2-12 สัญลักษณ์แสดงคอมบายด์แฟรกเมนต์ชนิดลูป

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 เครื่องมือ “Jcover”

เครื่องมือ Jcover [11] ถูกพัฒนาขึ้นโดยบริษัท Man Machine Systems เป็นเครื่องมือที่พัฒนาเพื่อเชิงพาณิชย์ซึ่งเป็นเครื่องมือที่ใช้ในการวิเคราะห์ความครอบคลุมของการทดสอบของโปรแกรมภาษาจาวา โดยแทรกโค้ดลงในซอร์ซโค้ดที่ต้องการทดสอบก่อนเริ่มดำเนินการทดสอบ เมื่อดำเนินการทดสอบเสร็จแล้วจะเก็บข้อมูลเพื่อสร้างรายงานเชิงสถิติด้านความครอบคลุมของการทดสอบและแจ้งส่วนของซอร์ซโค้ดที่ไม่ได้ถูกดำเนินการทดสอบ ซึ่งจะไฮไลท์ (Highlight) ส่วนของซอร์ซโค้ดในบรรทัดที่ไม่ได้ถูกดำเนินการทดสอบ โดยการวัดความครอบคลุมของการทดสอบนั้น ในเครื่องมือนี้สนับสนุนหน่วยวัดความครอบคลุม 4 ชนิดคือ ครอบคลุมทุกคำสั่ง (Statement coverage) ครอบคลุมทุกกิ่ง (Branch coverage) ครอบคลุมเมท็อด(Method coverage) และครอบคลุมคลาส (Class Coverage)

เครื่องมือนี้มีข้อดีคือ สามารถบอกบรรทัดที่ไม่ได้ดำเนินการทดสอบได้รวมทั้งสามารถบอกจำนวนสัดส่วนของ เมท็อด คลาส แพคเกจ (Package) แฟ้มข้อมูล (File) คำสั่ง (Statement) และกิ่ง

(Branch) ที่ถูกดำเนินการทดสอบต่อจำนวนที่มีอยู่ แต่อย่างไรก็ตามเครื่องมือนี้ใช้ในการทดสอบระดับหน่วย รวมทั้งเครื่องมือนี้สามารถแจ้งจุดที่ไม่ถูกดำเนินการทดสอบได้เท่านั้น ไม่สามารถสร้างกรณีทดสอบเพื่อให้เกิดความครอบคลุม งานวิจัยที่จัดทำขึ้นจึงใช้หลักการการแทรกซอร์ซโค้ดดังที่เครื่องมือนี้ใช้เพื่อหาคลาสและเมท็อดที่ไม่ถูกดำเนินการทดสอบ

2.2.2 งานวิจัย "Model-Based Test Case Synthesis using UML Interaction Diagrams"[12]

งานวิจัยนี้นำเสนอวิธีสนับสนุนกระบวนการสร้างกรณีทดสอบอย่างอัตโนมัติจากแผนภาพลำดับ โดยจะรับข้อมูลมาจากแผนภาพลำดับ เช่นอ็อบเจกต์ แมสเสจ และแฟรกเมนต์ ต่างๆ แล้วมาแปลงเป็นซิแนริโอกราฟ ซึ่งมีกลุ่มของแมสเสจเป็นโหนด หลังจากนั้นจะแปลงซิแนริโอกราฟ ให้เป็นไอทีเอ็ม (ITM: Intermediate testable Model) ซึ่งเป็นกราฟรูปแบบหนึ่งที่ลดรูปกลุ่มของโหนดที่อยู่ภายใต้เงื่อนไขต่างๆ รวมเป็นโหนดเดียว เพื่อให้ง่ายต่อการสร้างลำดับการทดสอบ เมื่อแปลงซิแนริโอกราฟให้เป็นไอทีเอ็มได้แล้วจึงลำดับการทดสอบจากไอทีเอ็มนั้น แล้วให้ผู้ใช้ระบุขอบเขตข้อมูลที่ใช้ทดสอบ

จากงานวิจัยนี้ ผู้วิจัยได้นำแนวคิดการแปลงแผนภาพลำดับให้เป็นซิแนริโอกราฟมาประยุกต์ใช้โดยในงานวิจัยที่จัดทำนี้ จะแปลงแผนภาพลำดับให้เป็นกราฟการทำงานแล้วหาเส้นทางที่ต้องการสร้างกรณีทดสอบ แล้วนำข้อมูลทดสอบจากคำอธิบายยูสเคสแทนการให้ผู้ใช้ระบุข้อมูลทดสอบ

2.2.3 วิทยานิพนธ์ "วิธีการสร้างกรณีทดสอบโดยอัตโนมัติจากยูสเคส"[13]

วิทยานิพนธ์นี้ได้ทำการออกแบบวิธีการสร้างกรณีทดสอบโดยอัตโนมัติจากยูสเคส รวมทั้งพัฒนาเครื่องมือสำหรับการสร้างกรณีทดสอบโดยอัตโนมัติจากยูสเคส โดยเครื่องมือนี้จะนำแผนภาพยูสเคสและรายละเอียดยูสเคสมาวิเคราะห์หาความสัมพันธ์ระหว่างยูสเคส ถ้ายูสเคสมีความสัมพันธ์ระหว่างกันจะรวมกันเป็นหนึ่งยูสเคสแล้วบันทึกลงฐานข้อมูล หลังจากนั้นจะสร้างกรณีทดสอบโดยการดึงข้อมูลรายละเอียดยูสเคสจากฐานข้อมูลออกมาสร้างกรณีทดสอบ

จากวิทยานิพนธ์นี้ ผู้ทำการวิจัยได้นำแนวคิดการนำเอาข้อมูลจากรายละเอียดยูสเคสมาทำการสร้างข้อมูลทดสอบในขั้นตอนการสร้างกรณีทดสอบ ซึ่งแผนภาพลำดับนั้นเป็นเรียลไลเซชันของยูสเคส จึงสามารถนำแนวคิดการนำข้อมูลรายละเอียดยูสเคสมาใช้ในการสร้างข้อมูลทดสอบได้

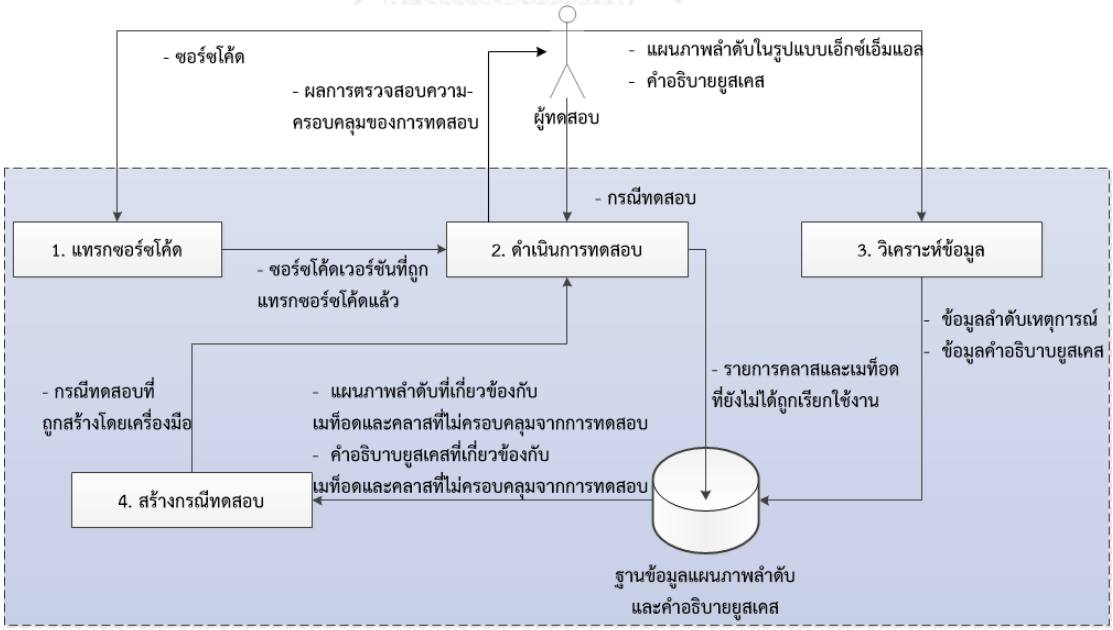
CHULALONGKORN UNIVERSITY

บทที่ 3
การวิเคราะห์และออกแบบเครื่องมือ

ในบทนี้จะอธิบายการวิเคราะห์และการออกแบบเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ โดยกล่าวถึงจะภาพรวมของเครื่องมือ และอธิบายการวิเคราะห์และออกแบบเครื่องมือด้วยแผนภาพยูสเคส (Use Case Diagram) แผนภาพคลาส (Class Diagram) และแผนภาพลำดับ (Sequence Diagram) รวมทั้งอธิบายโครงสร้างของฐานข้อมูล ซึ่งมีรายละเอียดดังต่อไปนี้

3.1 ภาพรวมของการทำงานของเครื่องมือ

วิทยานิพนธ์นี้ได้พัฒนาเครื่องมือที่ใช้ตรวจสอบความครอบคลุมของการทดสอบแบบบูรณาการ รวมทั้งสร้างกรณีทดสอบเพิ่มเติมเพื่อให้เกิดความครอบคลุม ถ้าพบว่ากรณีทดสอบที่มีอยู่ไม่สามารถทดสอบโปรแกรมแบบบูรณาการได้อย่างครอบคลุม ภาพรวมการทำงานของเครื่องมือสามารถแสดงได้ดังภาพที่ 3-1



ภาพที่ 3-1 ภาพรวมการทำงานของเครื่องมือ

ภาพที่ 3-1 แสดงถึงแผนภาพเชิงแนวคิดของเครื่องมือที่งานวิจัยนี้ได้นำเสนอ ซึ่งประกอบไปด้วย 4 ส่วนการทำงานย่อย โดยเริ่มจากผู้ทดสอบนำซอร์สโค้ดที่ต้องการหาความครอบคลุมของการทดสอบเข้าสู่ระบบ หลังจากนั้นระบบจะแทรกซอร์สโค้ดเพื่อดูว่าซอร์สโค้ดได้ถูกทดสอบที่ส่วนใดบ้าง ในขั้นตอนนี้ระบบจะส่งออกเป็นซอร์สโค้ดเวอร์ชันที่แทรกซอร์สโค้ดแล้ว เพื่อนำมาสู่ขั้นตอนการดำเนินการทดสอบ โดยขั้นตอนนี้ผู้ทดสอบจะนำกรณีทดสอบที่มีอยู่มาดำเนินการทดสอบเพื่อดูว่ากรณีทดสอบที่มีอยู่นั้น สามารถดำเนินการทดสอบแบบบูรณาการได้อย่างครอบคลุมหรือไม่ เมื่อดำเนินการทดสอบเสร็จแล้ว ระบบจะแสดงผลการตรวจสอบความครอบคลุมของการทดสอบ

นอกจากนี้ ระบบจะยังมีส่วนที่สร้างกรณีทดสอบหลังจากตรวจสอบความครอบคลุมของการทดสอบ โดยให้ผู้ทดสอบนำเข้าแผนภาพลำดับในรูปแบบเอกซ์เอ็มแอลเพื่อนำไปสร้างกราฟการทำงาน และนำเข้าคำอธิบายยูสเคสเพื่อนำไปสร้างข้อมูลทดสอบ เมื่อได้แผนภาพลำดับและคำอธิบายยูสเคสแล้ว ระบบจะวิเคราะห์ข้อมูลเพื่อแยกข้อมูลลำดับเหตุการณ์และข้อมูลคำอธิบายยูสเคส หลังจากนั้นจะบันทึกข้อมูลที่แยกออกมาได้ลงสู่ฐานข้อมูลแผนภาพลำดับและคำอธิบายยูสเคส แล้วนำชื่อคลาสและเมท็อดที่ยังไม่ครอบคลุมจากการดำเนินการทดสอบมาเปรียบเทียบกับชื่อคลาสและเมท็อดที่บรรจุอยู่ในแผนภาพลำดับในฐานข้อมูลแผนภาพลำดับและคำอธิบายยูสเคส เมื่อเปรียบเทียบแล้วพบว่าแผนภาพลำดับใดมีชื่อคลาสและเมท็อดที่ไม่ครอบคลุมจากการทดสอบด้วยกรณีทดสอบ ให้นำแผนภาพลำดับนั้นรวมทั้งคำอธิบายยูสเคสเข้าสู่ขั้นตอนการสร้างกรณีทดสอบเพื่อนำไปสร้างกรณีทดสอบต่อไป เมื่อสร้างกรณีทดสอบเสร็จแล้ว จะนำกรณีทดสอบที่สร้างได้ มาเข้าสู่ขั้นตอนการดำเนินการทดสอบอีกครั้งเพื่อตรวจสอบว่ากรณีทดสอบที่สร้างขึ้นเมื่อรวมกับกรณีทดสอบเดิมจะทำให้การทดสอบแบบบูรณาการมีความครอบคลุมทั้งหมด แล้วส่งผลการตรวจสอบความครอบคลุมให้กับผู้ทดสอบได้รับทราบ

3.1.1 การแทรกซอร์สโค้ด

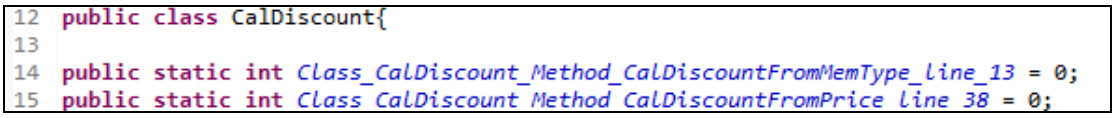
การแทรกซอร์สโค้ดนั้นเป็นการแทรกซอร์สโค้ดลงในซอร์สโค้ดที่ต้องการทดสอบโดยมีเพื่อใช้ในการตรวจสอบว่าซอร์สโค้ดส่วนนั้นได้ถูกทดสอบหรือไม่ โดยการแทรกซอร์สโค้ดนั้นระบบจะอ่านซอร์สโค้ดที่รับเข้ามาและเปรียบเทียบว่าซอร์สโค้ดที่อ่านนั้นอยู่ใน ซอร์สโค้ดที่จำเป็นต้องมีการแทรกโค้ดหรือไม่ ซึ่งมีดังต่อไปนี้

1) การสร้างคลาส เมื่อระบบอ่านซอร์สโค้ดแล้วพบว่ามีการสร้างคลาส ระบบจะแทรกซอร์สโค้ดซึ่งซอร์สโค้ดที่แทรกจะสร้างตัวแปรสำหรับนับจำนวนการเรียกใช้เมท็อดของแต่ละเมท็อด ตัวอย่างลักษณะการแทรกซอร์สโค้ดเมื่อพบการสร้างคลาส มีลักษณะดังภาพที่ 3-2

```
public static int Class_[ชื่อคลาส]_Method_[ชื่อเมท็อด]_line_[บรรทัดที่เมท็อดประกาศก่อนการแทรกซอร์สโค้ด]
```

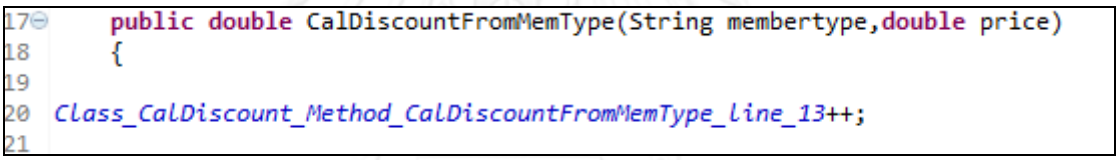
ภาพที่ 3-2 ลักษณะการแทรกซอร์สโค้ดเมื่อพบการสร้างคลาส

ตัวอย่างลักษณะการแทรกซอร์สโค้ดเมื่อพบการสร้างคลาส แสดงได้ดังแสดงได้ดังภาพที่ 3-3 ซอร์สโค้ดที่ถูกแทรกคือซอร์สโค้ดที่อยู่ในบรรทัดที่14และ15



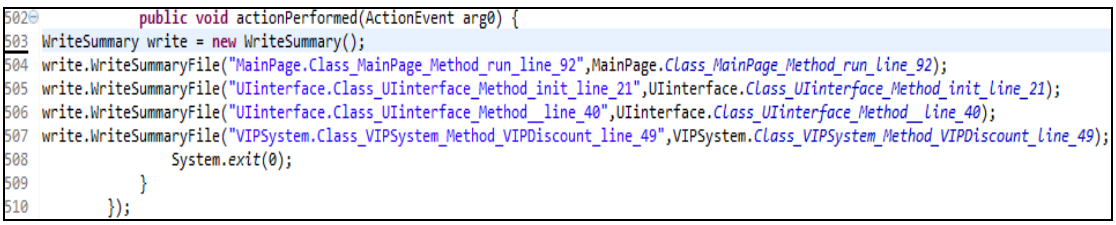
ภาพที่ 3-3 ตัวอย่างการแทรกซอร์สโค้ดของการสร้างคลาส

2) การสร้างเมท็อด เมื่อระบบอ่านซอร์สโค้ดแล้วพบว่ามีการสร้างเมท็อด ระบบจะแทรกซอร์สโค้ดซึ่งซอร์สโค้ดที่แทรกจะนับจำนวนการเรียกใช้เมท็อด โดยเพิ่มค่าของตัวแปรที่ใช้สำหรับนับจำนวนการเรียกใช้เมท็อดของแต่ละเมท็อดตัวอย่างลักษณะการแทรกซอร์สโค้ดเมื่อพบการสร้างเมท็อด สามารถแสดงได้ดังภาพที่ 3-4 ซอร์สโค้ดที่ถูกแทรกอยู่ในบรรทัดที่ 20



ภาพที่ 3-4 ตัวอย่างการแทรกซอร์สโค้ดของการสร้างเมท็อด

3) จุดสิ้นสุดการทำงานของโปรแกรม เมื่อระบบพบจุดสิ้นสุดการทดงานของโปรแกรมซึ่งแสดงได้โดยคำสั่ง “System.exit(0);” ระบบจะแทรกซอร์สโค้ดเพื่อเก็บข้อมูลการเรียกใช้เมท็อดแต่ละเมท็อด ซึ่งการเก็บข้อมูลนี้จะทำผ่านการส่งค่าของตัวแปรที่ใช้นับการเรียกใช้เมท็อดเข้าสู่คลาสหนึ่งที่ระบบได้สร้างเตรียมไว้ก่อนแล้ว คลาสนี้ใช้สำหรับเขียนค่าตัวแปรที่นับการเรียกใช้เมท็อดลงสู่ไฟล์ ซึ่งไฟล์นี้จะสรุปการเรียกใช้ของแต่ละเมท็อด ตัวอย่างการแทรกซอร์สโค้ดเมื่อพบกับจุดสิ้นสุดการทำงานของโปรแกรม แสดงได้ดัง ซอร์สโค้ดที่ถูกแทรกอยู่ในบรรทัดที่ 503-507



ภาพที่ 3-5 ตัวอย่างการแทรกซอร์สโค้ดของจุดสิ้นสุดของโปรแกรม

3.1.2 การดำเนินการทดสอบ

การดำเนินการทดสอบในขั้นตอนนี้ ผู้ทดสอบจะนำซอร์สโค้ดที่แทรกซอร์สโค้ดแล้วมาทดสอบเพื่อตรวจสอบว่าการทดสอบแบบบูรณาการด้วยกรณีทดสอบที่มีอยู่ครอบคลุมหรือไม่ ซึ่งขั้นตอนการทดสอบนี้ ผู้ทดสอบต้องดำเนินการทดสอบด้วยตัวเอง ซึ่งเมื่อกระทำการทดสอบ (Test execution) ผ่านซอร์สโค้ดที่ถูกแทรกไว้จากขั้นตอนที่ 3.1.1การแทรกซอร์สโค้ด จะเกิดผลต่างๆ ดังนี้

1) เมื่อกระทำการทดสอบผ่านการสร้างคลาส ซอร์สโค้ดที่ถูกแทรกไว้จะสร้างตัวแปรขึ้นมาตามจำนวนเมท็อดที่อยู่ในคลาสนั้นซึ่งตัวแปรเหล่านี้จะใช้ในการนับจำนวนการเรียกใช้ของแต่ละเมท็อด

2) เมื่อกระทำการทดสอบผ่านการสร้างเมท็อด ตัวแปรที่สร้างขึ้นจากการแทรกซอร์สโค้ดจะนับขึ้นเพื่อบอกจำนวนการเรียกใช้เมท็อดนั้น

3) เมื่อกระทำการทดสอบผ่านจุดสิ้นสุดของโปรแกรม ตัวแปรที่สร้างขึ้นเพื่อนับการเรียกใช้ซอร์สโค้ดรวมทั้งค่าของตัวแปรนั้นจะถูกเขียนลงในไฟล์สรุปการเรียกใช้เมท็อด ตัวอย่างไฟล์สรุปการเรียกใช้เมท็อดแสดงได้ดังภาพที่ 3-6

เมื่อสิ้นสุดการทดสอบแล้วระบบจะคำนวณความครอบคลุมของคลาส (Class coverage) และความครอบคลุมเมท็อด (Method coverage) โดยมีสมการดังต่อไปนี้

$$\text{ความครอบคลุมคลาส} = \frac{\text{จำนวนคลาสที่ครอบคลุม}}{\text{จำนวนคลาสทั้งหมดในระบบ}} \times 100 \qquad (1)$$

โดยคลาสที่ครอบคลุมคือคลาสที่เมท็อดที่เป็นสมาชิกถูกเรียกอย่างน้อยหนึ่งครั้งทุกเมท็อด

$$\text{ความครอบคลุมเมท็อด} = \frac{\text{จำนวนเมท็อดที่ครอบคลุม}}{\text{จำนวนเมท็อดทั้งหมดในระบบ}} \times 100 \qquad (2)$$

โดยเมท็อดที่ครอบคลุมคือเมท็อดถูกเรียกอย่างน้อยหนึ่งครั้ง

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

	A	B
1	CalDiscount.Class_CalDiscount_Method_CalDiscountFromMemType_line_13	0
2	CalDiscount.Class_CalDiscount_Method_CalDiscountFromPrice_line_38	0
3	CalPoint.Class_CalPoint_Method_CalpointFromMemType_line_12	1
4	CalPoint.Class_CalPoint_Method_CalPointFromPrice_line_34	1
5	CalPrice.Class_CalPrice_Method_CalTotalPrice_line_69	33
6	GetVocher.Class_GetVocher_Method_VocherSystem_line_19	1
7	GetVocher.Class_GetVocher_Method_Vocherdiscount_line_31	1
8	GetVocher.Class_GetVocher_Method_GExpiredate_line_64	1
9	GetVocher.Class_GetVocher_Method_WriteVochr_line_88	1
10	MainPage.Class_MainPage_static_Method_main_line_82	1
11	MainPage.Class_MainPage_Method_run_line_92	1
12	Uiinterface.Class_Uiinterface_Method_init_line_21	3
13	Uiinterface.Class_Uiinterface_Method__line_40	3
14	VIPSystem.Class_VIPSystem_Method_VIPDiscount_line_49	1
15		

ภาพที่ 3-6 ตัวอย่างไฟล์สรุปการเรียกใช้เมท็อด

หลังจากระบบคำนวณความครอบคลุมคลาสและความครอบคลุมเมท็อดแล้ว ระบบจะส่งผลการคำนวณ รวมทั้งชื่อคลาสและเมท็อดที่ไม่ครอบคลุมให้ผู้ทดสอบทราบ รวมทั้งบันทึกชื่อเมท็อดที่ไม่ครอบคลุมลงในไฟล์ เพื่อใช้ขั้นตอนการสร้างกรณีทดสอบต่อไป

3.1.3 การวิเคราะห์ข้อมูล

นอกจากส่วนของการตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการแล้ว ระบบยังมีส่วนสร้างกรณีทดสอบเพิ่มเติมในกรณีที่กรณีทดสอบเดิมไม่ครอบคลุม ซึ่งส่วนของการสร้างกรณีทดสอบเพิ่มเติมนั้น ผู้ทดสอบจะต้องดำเนินการในส่วนของการตรวจสอบความครอบคลุมของการทดสอบก่อน ไม่สารถใช้เครื่องมือนี้เพื่อสร้างกรณีทดสอบเพียงอย่างเดียวได้เนื่องจากระบบจำเป็นต้องใช้ชื่อเมท็อดที่ไม่ครอบคลุมซึ่งสร้างขึ้นในส่วน 3.1.2 การดำเนินการทดสอบ

ในส่วนของการสร้างกรณีทดสอบ ระบบจะให้ผู้ทดสอบนำแผนภาพลำดับที่ได้ถูกบันทึกให้เป็นเอกสารเอกซ์เอ็มแอล รวมทั้งรายละเอียดยูสเคสเข้าสู่ระบบ หลังจากนั้นระบบจะวิเคราะห์โดยการแยกข้อมูลเอกสารเอกซ์เอ็มแอลของแผนภาพลำดับ เพื่อสกัดข้อมูลต่างๆ เพื่อนำไปใช้สร้างลำดับเหตุการณ์ในขั้นตอนการสร้างกรณีทดสอบ ซึ่งมีดังต่อไปนี้

1) ชื่อแผนภาพลำดับ (Sequence diagram’s name)

2) ลำดับการเรียกใช้เมท็อด (No.)

3) ชื่อเมท็อด (Method)

4) ชื่อคลาสของเมท็อดที่ถูกเรียก (Class)

5) เงื่อนไขการเกิดของเมท็อดที่ปรากฏในคอมบายด์แฟรกเมนต์ (Combine fragment)

ในส่วนของเงื่อนไขการเกิดของเมท็อดที่ปรากฏอยู่ในคอมบายด์แฟรกเมนต์ที่ระบบนี้รองรับมี 3 ชนิด แบ่งตามคอมบายด์แฟรกเมนต์ที่ปรากฏอยู่ในแผนภาพลำดับคือ

1) อัลเทอร์เนทีฟ (Alternative) จะแสดงเงื่อนไขในรูปแบบของ "if then else"คือกำหนดทางเลือกของเหตุการณ์ที่ไม่เกิดร่วมกัน

2) ออปชัน (Option) จะแสดงเงื่อนไขที่อาจจะเกิดขึ้นหรือไม่ก็ได้ ซึ่งก็คือการแสดงเงื่อนไขในรูปแบบ "if"

3) ลูป (Loop) จะแสดงเงื่อนไขการวนซ้ำการเรียกใช้เมท็อดจนกระทั่งเงื่อนไขเป็นเท็จ ซึ่งเงื่อนไขมีทั้งเป็นเงื่อนไขบูลลีนตามปกติซึ่งในรูปแบบการเขียนโปรแกรมคือ while แล้วยังสามารถประกอบด้วยเงื่อนไขการทำซ้ำน้อยสุดและมากสุดซึ่งในรูปแบบการเขียนโปรแกรมคือ for อีกด้วย

ตัวอย่างของข้อมูลที่สกัดได้จากแผนภาพลำดับ สามารถแสดงได้ดังภาพที่ 3-7

	A	B	C	D
1	Sequence Diagram Name :	Promotion_TestData_Ver2		
2	1	UlinterfaceApp	Ulinterfac	-
3	2	init	Ulinterfac	-
4	3	CalTotalPrice	CalPrice	IDList
5	4	VIPDiscount	VIPSysten	vipsts
6	5	CalDiscountFromMemType	CalDiscou	DiscountFromMemtype&&PromotionDisCount:.:
7	6	CalDiscountFromPrice	CalDiscou	DiscountFromPrice&&PromotionDisCount:.:
8	7	CalpointFromMemType	CalPoint	PointFromMemtype&&PromotionPoint:.:
9	8	CalPointFromPrice	CalPoint	PointFromPrice&&PromotionPoint:.:
10	9	VocherSystem	GetVoche	PromotionVocher
11	10	Vocherdiscount	GetVoche	PromotionVocher
12	11	GExpiredate	GetVoche	PromotionVocher
13	12	WriteVochr	GetVoche	PromotionVocher
14				

ภาพที่ 3-7 ตัวอย่างข้อมูลที่สกัดได้จากแผนภาพลำดับ

นอกจากนี้ในขั้นตอนนี้จะวิเคราะห์ข้อมูลจากคำอธิบายยูสเคสที่ทำเข้ามาด้วยเพื่อสกัดข้อมูลที่จะนำไปใช้ในการสร้างข้อมูลทดสอบในขั้นตอนการสร้างกรณีทดสอบ ซึ่งแผ่นแบบของคำอธิบายยูสเคสนี้มีต้นแบบมาจากวิทยานิพนธ์ วิธีการสร้างกรณีทดสอบโดยอัตโนมัติจากยูสเคส[13] ดังแสดงได้ดัง โดยข้อมูลต่างๆจากคำอธิบายยูสเคสมีดังต่อไปนี้

1) Sequence number คือหมายเลขแผนภาพลำดับที่เกี่ยวข้องกับคำอธิบายยูสเคสนี้

2) Pre-condition ของยูสเคส เป็นข้อความแสดงการทำงานหรือกิจกรรม

3) Condition เริ่มต้นของ successScenario คือเงื่อนไขที่ทำให้ลำดับการทำงานปกติของแผนภาพลำดับนั้นสามารถทำงานได้ โดยระบุเงื่อนไขไว้หลังเครื่องหมาย [0]

4) Condition ของ AlternativeScenario คือเงื่อนไขซึ่งอาจเกิดขึ้นในระหว่างการเกิดลำดับเหตุการณ์ในแผนภาพลำดับ ซึ่งทำให้ผลลัพธ์การทำงานไม่ตรงกับผลลัพธ์ของเงื่อนไขการทำงานปกติ

โดยระบุเงื่อนไขไว้หลังเครื่องหมาย [1] และในกรณีที่มี Condition ของ AlternativeScenario มากกว่า 1 ให้เรียงลำดับตัวเลขโดยเริ่มจาก 1

5) Post-condition ของ successScenario เป็นผลลัพธ์หลังจากการทำงานโดยเงื่อนไขของ Condition เริ่มต้นของ successScenario ได้ โดยระบุเงื่อนไขไว้หลังเครื่องหมาย [0]

6) Post-condition ของ AlternativeScenario คือผลลัพธ์หลังจากการทำงานโดยเงื่อนไขของ Condition ของ AlternativeScenario โดยระบุเงื่อนไขไว้หลังเครื่องหมาย [1] ในกรณีที่มีเงื่อนไขมากกว่า 1 ให้ใส่หมายเลขผลลัพธ์ในตรงกับเงื่อนไขที่ทำให้เกิดผลลัพธ์นั้นระหว่างเครื่องหมาย [และ เครื่องหมาย]

7) Input data เป็นข้อมูลนำเข้า ซึ่งประกอบไปด้วย

7.1) Name เป็นชื่อของข้อมูลนำเข้า

7.2) Type เป็นชนิดของข้อมูลนำเข้า ซึ่งกำหนดให้มีทั้งหมด 4 ชนิดคือ

- Integer เป็นข้อมูลตัวเลขจำนวนเต็ม
- Float เป็นข้อมูลตัวเลขจำนวนจริง
- String เป็นข้อมูลชุดอักขระ
- Boolean เป็นข้อมูลชนิดตรรกะ ซึ่งมีค่าที่เป็นไปได้คือ ค่าจริง (True) และเท็จ (False) เท่านั้น

7.3) Size เป็นขนาดข้อมูลนำเข้า ซึ่งขนาดของข้อมูลขึ้นกับชนิดข้อมูลนำเข้าดังนี้

- Integer ให้ใส่จำนวนหลักสูงสุดที่เป็นไปได้สำหรับการสร้างข้อมูลทดสอบ ถ้าไม่กำหนด จำนวนหลักจะถูกสุ่มโดยอัตโนมัติ
- Float ให้ใส่ตัวเลขกำหนดจำนวนหลังจุดทศนิยมของข้อมูลทดสอบ ถ้าไม่กำหนดจำนวนหลังจุดทศนิยม จะถูกสุ่มโดยอัตโนมัติ
- String ให้ใส่ตัวเลขกำหนดความยาวของชุดข้อมูลอักขระ ถ้าไม่กำหนด ความยาวของชุดข้อมูลอักขระ จะถูกสุ่มโดยอัตโนมัติ
- Boolean ไม่ต้องกำหนดขนาดข้อมูลนำเข้า

7.4) Min เป็นค่าต่ำสุดที่เป็นไปได้ของข้อมูล

7.5) Max เป็นค่ามากสุดที่เป็นไปได้ของข้อมูล

7.6) Scenario no. เป็นตัวเลขที่กำหนดว่าแต่ละ input data ถูกใช้ใน successScenario หรือ AlternativeScenario หมายเลขใด

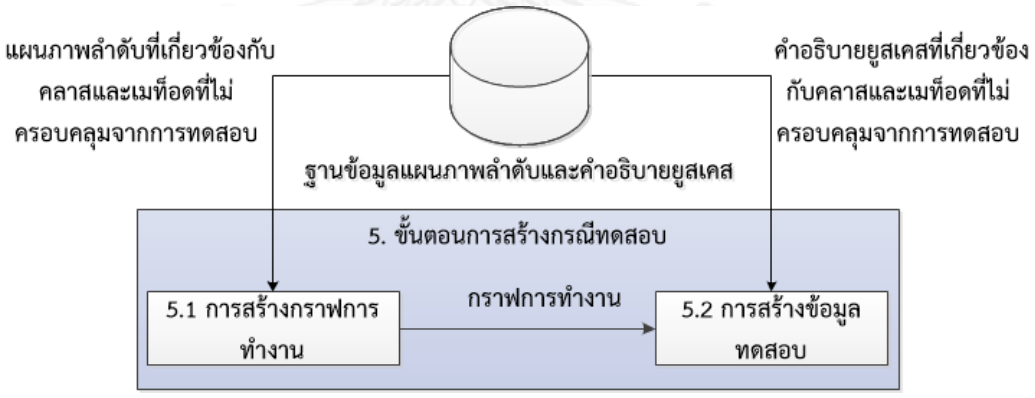
เมื่อระบบได้สกัดข้อมูลได้ตามที่ต้องการแล้วจะนำข้อมูลนั้นเข้าสู่ฐานข้อมูลแผนภาพลำดับและคำอธิบายยูสเคส

A	B	C	D	E	F
Sequence diagram number	Promotion_TestData_Ver2				
Pre-condition	Cashier calculate the merchandise price				
Success scenario condition	[0] (VIPStS == VIPYes")&&(Promotiontype == "PromotionDisCount")&&(DiscType == "DiscountFromMemtype")				
	[1] (VIPStS == VIPNo")&&(Promotiontype == "PromotionDisCount")&&(DiscType == "DiscountFromPrice")				
	[2] (VIPStS == VIPNo")&&(Promotiontype == "PromotionPoint")&&(PointType == "PointFromMemtype")				
	[3] (VIPStS == VIPNo")&&(Promotiontype == "PromotionPoint")&&(PointType == "PointFromPrice")				
Alternative scenario condition	[4] (VIPStS == VIPNo")&&(Promotiontype == "PromotionVocher")				
Post condition of success scenaric	[0] System display total price and Discount				
	[1] System display total price and Discount				
	[2] System display total price and Point				
	[3] System display total price and Point				
Post condition of alternative scen	[4] System Display totalprice , print vocher and display Vocher Already Printed " """				
input data					
Name	Type	Size	Min	Max	Scenario no.
ID	Int	4	1001	1015	[0][1][2][3][4]
VIPStS	String	-	VIPNo	VIPYes	[0][1][2][3][4]
Promotiontype	String	-	PromotionDisCou	PromotionVocher	[0][1][2][3][4]
DiscType	String	-	DiscountFromMer	DiscountFromPrice	[0][1]
PointType	String	-	PointFromMemty	PointFromPrice	[2][3]
DiscMemType	String	-	DiscountMemTyp	DiscountMemTypeC	[0]
PointMemType	String	-	PointMemtypeA	PointMemTypeC	[2]

ภาพที่ 3-8 ตัวอย่างคำอธิบายยูสเคส

3.1.4 การสร้างกรณีทดสอบ

ในหัวข้อนี้จะกล่าวถึงการสร้างกรณีทดสอบโดยเครื่องมือจะสร้างกรณีทดสอบสำหรับคลาสและเมท็อดที่ไม่ครอบคลุมจากการทดสอบ ขั้นตอนย่อยๆของการสร้างกรณีทดสอบนั้นแสดงได้ดังภาพที่ 3-9



ภาพที่ 3-9 ขั้นตอนการทำงานของการสร้างกรณีทดสอบ

ภาพที่ 3-9 แสดงให้เห็นขั้นตอนทำงานของการสร้างกรณีทดสอบ โดยขั้นตอนนี้จะรับแผนภาพลำดับที่เกี่ยวข้องกับคลาสและเมท็อดที่ไม่ครอบคลุมจากการทดสอบจากฐานข้อมูลแผนภาพลำดับและคำอธิบายยูสเคสเข้ามาเพื่อสร้างกราฟการทำงาน หลังจากนั้นจะส่งกราฟการทำงานที่สร้างได้ไปหาเส้นทางการทำงานที่มีเมท็อดที่ไม่ครอบคลุมอยู่ในเส้นทาง แล้วนำมาสร้างกรณีทดสอบโดยนำ

คำอธิบายยูสเคสที่เกี่ยวข้องกับคลาสและเมท็อดที่ไม่ครอบคลุมจากการทดสอบจากฐานข้อมูลแผนภาพลำดับและคำอธิบายยูสเคสมาสร้างข้อมูลทดสอบ โดยขั้นตอนต่างๆมีรายละเอียดดังนี้

1) การสร้างกราฟการทำงาน

การสร้างกราฟการทำงานจะเริ่มหลังจากได้รับแผนภาพลำดับแล้ว นำมาสร้างเป็นไดเร็กกราฟโดยเมท็อดที่ปรากฏในแผนภาพจะแทนด้วยโหนด และทิศทางการไหลของการควบคุม (Control flow) แทนด้วยเส้นเชื่อม (Edge) ตัวอย่างกราฟการทำงานของแผนภาพลำดับภาพที่ 3-10 สามารถแสดงได้ดังภาพที่ภาพที่ 3-11

2) การสร้างข้อมูลทดสอบ

การสร้างข้อมูลทดสอบนั้นจะเริ่มจากนำกราฟการทำงานมาหาเส้นทางที่มีเมท็อดที่ไม่ครอบคลุมจากการทดสอบ แล้วบันทึกเงื่อนไขที่ปรากฏในเส้นทางนั้นเพื่อนำไปใช้สร้างกรณีทดสอบ โดยการบันทึกเงื่อนไขมีรูปแบบดังต่อไปนี้

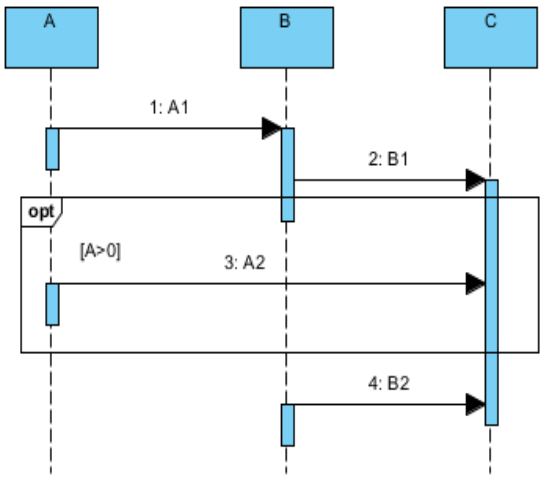
1) อัลเทอร์เนทีฟ การบันทึกเงื่อนไขที่เกิดขึ้นเป็นดังต่อไปนี้

1.1) ถ้าเมท็อดที่ต้องการอยู่ในเส้นทางที่เป็นจริง (True) ระบบจะบันทึกเงื่อนไขที่โหนดเงื่อนไข (Condition node)

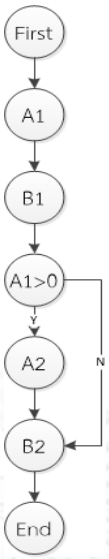
1.2) ถ้าเมท็อดที่ต้องการอยู่ในเส้นทางที่เป็นเท็จ (False) ระบบจะบันทึกเงื่อนไขเป็นนิเสธของโหนดเงื่อนไข

2) ออปชัน ระบบจะบันทึกเงื่อนไขที่โหนดเงื่อนไข

3) ลูป ระบบจะบันทึกเงื่อนไขในโหนดเงื่อนไขโดยไม่คำนึงถึงจำนวนรอบของการวนซ้ำ เนื่องจากงานวิจัยนี้มุ่งเน้นที่การเรียกใช้เมท็อด



ภาพที่ 3-10 ตัวอย่างแผนภาพลำดับ



ภาพที่ 3-11 กราฟการทำงานของแผนภาพลำดับตัวอย่างภาพที่ 3-10

หลังจากบันทึกเงื่อนไขที่ปรากฏในเส้นทางได้แล้วได้แล้ว ระบบจะนำข้อมูลจากคำอธิบายยูสเคสนั้นมาสร้างข้อมูลทดสอบ ซึ่งแผ่นแบบของกรณีทดสอบนี้มีต้นแบบมาจากวิทยานิพนธ์ วิธีการสร้างกรณีทดสอบโดยอัตโนมัติจากยูสเคส[13] โดยแผ่นแบบของกรณีทดสอบมีรายละเอียดดังนี้

1) หมายเลขกรณีทดสอบ (Test case ID)

2) ชื่อกรณีทดสอบ (Test case name)

3) ข้อความอธิบายกรณีทดสอบ (Test case description)

4) ข้อมูลเริ่มต้นของกรณีทดสอบ (Initial value)

5) ข้อมูลทดสอบ (Input value)

6) ผลลัพธ์ที่คาด (Expected output)

โดยวิธีการสร้างข้อมูลทดสอบคือ นำเงื่อนไขที่บันทึกได้มาหาว่าตรงหรือเป็นส่วนหนึ่งของ Condition เริ่มต้นหมายเลขใด แล้วบันทึก Condition เริ่มต้นนั้นไว้ เพื่อใช้เป็น ข้อความอธิบายกรณีทดสอบ (ข้อ 3) ข้อความอธิบายกรณีทดสอบ (Test case description) ในแผ่นแบบกรณีทดสอบ) ในกรณีที่ไม่มีเงื่อนไขปรากฏในเส้นทาง จะใช้ Condition เริ่มต้นของ successScenario แล้วบันทึกหมายเลขนั้นไว้ เพื่อนำไปหาผลลัพธ์ที่คาด (ข้อ 6) ผลลัพธ์ที่คาด (Expected output) ในแผ่นแบบกรณีทดสอบ) โดยนำตัวเลขนั้นไปหาว่าตรงกับ Post-condition หมายเลขใด รวมทั้งนำไปหาข้อมูลทดสอบ (ข้อ 5) ข้อมูลทดสอบ (Input value) ในแผ่นแบบกรณีทดสอบ) โดยนำตัวเลขไปเทียบในช่อง Scenario no. ในส่วนของ Input data ว่าต้องใช้ข้อมูลทดสอบชื่อใดบ้าง หลังจากระบบทราบว่าต้องใช้ Input data ชื่ออะไรบ้าง ก็จะเก็บข้อมูลว่า Input data ชื่อนั้นมีขนาดเท่าใด ช่วงค่าต่ำสุดและสูงสุดเท่าใด แล้วสุ่มข้อมูลให้สอดคล้องกับขนาดหรือช่วงค่าต่ำสุดและสูงสุดนั้น

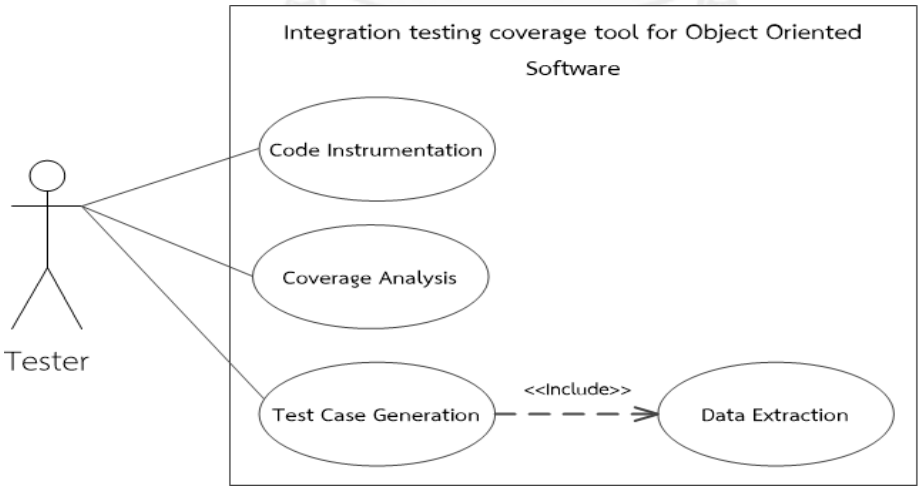
หลังจากนั้นระบบจะกำหนดข้อมูลที่ใช้ในกรณีทดสอบที่เหลือคือ หมายเลขกรณีทดสอบจะใช้หมายเลขเรียงตามลำดับการสร้างกรณีทดสอบ ชื่อกรณีทดสอบจะกำหนดโดยใช้ใช้ชื่อแผนภาพลำดับต่อกับหมายเลขที่ได้จากการหา Condition และ ข้อมูลเริ่มต้นของกรณีทดสอบ จะใช้ Precondition ที่กำหนดไว้ในคำอธิบายยูสเคส

3.2 การวิเคราะห์และออกแบบเครื่องมือ

การวิเคราะห์และออกแบบเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการสามารถแสดงได้ด้วยแผนภาพยูสเคส (Use Case Diagram) แผนภาพคลาส (Class Diagram) และแผนภาพลำดับ (Sequence Diagram) ซึ่งมีรายละเอียดดังต่อไปนี้

3.2.1 แผนภาพยูสเคส

แผนภาพยูสเคสเป็นแผนภาพที่แสดงให้เห็นถึงฟังก์ชันการทำงานหลักของระบบ โดยแผนภาพยูสเคสของเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุประกอบไปด้วย 4 ยูสเคสคือ ยูสเคสการแทรกซอร์สโค้ด (Code instrumentation) ยูสเคสการวิเคราะห์ความครอบคลุมของการทดสอบ (Coverage Analysis) ยูสเคสการสร้างกรณีทดสอบ (Coverage Analysis) และยูสเคสการสกัดข้อมูล (Data Extraction) ซึ่งสามารถแสดงได้ดังภาพที่ 3-12 โดยรายละเอียดยูสเคสของการแทรกซอร์สโค้ดสามารถแสดงได้ดังตารางที่ 3.1 รายละเอียดยูสเคสการวิเคราะห์ความครอบคลุมของการทดสอบสามารถแสดงได้ดังตารางที่ 3.2 รายละเอียดยูสเคสการสร้างกรณีทดสอบสามารถแสดงได้ดังตารางที่ 3.3 และรายละเอียดยูสเคสการสกัดข้อมูลสามารถแสดงได้ดังตารางที่ 3.4



ภาพที่ 3-12 แผนภาพยูสเคสของเครื่องมือ

ตารางที่ 3.1 รายละเอียดยูสเคสการแทรกซอร์สโค้ด

ยูสเคส	Code Instrumentation
แอคเตอร์	Tester
รายละเอียดยูสเคส	เพื่อแทรกซอร์สโค้ดที่ผู้ทดสอบนำเข้าสู่ระบบ
ยูสเคสที่สัมพันธ์	-
เงื่อนไขก่อนหน้า	-
ขั้นตอน	1. ระบบสร้างหน้าต่างให้ผู้ทดสอบนำซอร์สโค้ดเข้าสู่เครื่องมือ รวมทั้งให้ผู้ทดสอบเลือกโฟลเดอร์ที่ต้องการให้ระบบจัดเก็บซอร์สโค้ดที่แทรกแล้ว 2. ผู้ทดสอบกดปุ่มเลือกซอร์สโค้ดที่ต้องการทดสอบ 3. ผู้ทดสอบกดปุ่มเลือกโฟลเดอร์ที่ต้องการให้ระบบจัดเก็บซอร์สโค้ดที่แทรกแล้ว 4.ระบบแทรกซอร์สโค้ด 5.ระบบบันทึกซอร์สโค้ดที่แทรกแล้วลงในโฟล์เดอร์ที่ผู้ทดสอบระบุ
เงื่อนไขภายหลัง	ซอร์สโค้ดที่ต้องการนำมาทดสอบถูกแทรกซอร์สโค้ด

ตารางที่ 3.2 รายละเอียดยูสเคสการวิเคราะห์ความครอบคลุมของการทดสอบ

ยูสเคส	Coverage Analysis
แอคเตอร์	Tester
รายละเอียดยูสเคส	เพื่อวิเคราะห์ความครอบคลุมการทดสอบ
ยูสเคสที่สัมพันธ์	-
เงื่อนไขก่อนหน้า	ซอร์สโค้ดที่ถูกแทรกต้องถูกนำไปทดสอบ
ขั้นตอน	1. ระบบสร้างหน้าต่างให้ผู้ทดสอบระบุโฟลเดอร์ที่เก็บซอร์สโค้ดที่ถูกแทรกแล้ว 2. ระบบวิเคราะห์ความครอบคลุมการทดสอบ 3. ระบบแสดงผลการวิเคราะห์ความครอบคลุมการทดสอบ และให้ผู้ทดสอบระบุโฟลเดอร์ที่ต้องการเก็บผลการวิเคราะห์ 4.ระบบบันทึกชื่อเมท็อดที่ไม่ครอบคลุมลงสู่ไฟล์เพื่อใช้ในการสร้างกรณีทดสอบ

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ 3.2 รายละเอียดยูสเคสการวิเคราะห์ความครอบคลุมของการทดสอบ(ต่อ)

เงื่อนไขภายหลัง	ระบบแสดงผลการวิเคราะห์ความครอบคลุมการทดสอบและเก็บผลการวิเคราะห์ไว้ในโฟลเดอร์ที่ผู้ทดสอบระบุ

ตารางที่ 3.3 รายละเอียดยูสเคสการสร้างกรณีทดสอบ

ยูสเคส	Test Case Generation
แอคเตอร์	Tester
รายละเอียดยูสเคส	เพื่อสร้างกรณีทดสอบ
ยูสเคสที่สัมพันธ์	Include 1. Data Extraction
เงื่อนไขก่อนหน้า	-
ขั้นตอน	1. ระบบสร้างหน้าต่างให้ผู้ทดสอบระบุโฟลเดอร์ที่เก็บแผนภาพลำดับ และโฟลเดอร์ที่เก็บคำอธิบายยูสเคส 2. ผู้ทดสอบเลือกโฟลเดอร์ที่ใช้เก็บแผนภาพลำดับ 3. ผู้ใช้เลือกโฟลเดอร์ที่ใช้เก็บคำอธิบายยูสเคส 4. ผู้ใช้เลือกโฟลเดอร์ที่ต้องการใช้เก็บกรณีทดสอบที่สร้างได้ 5. ระบบนำแผนภาพลำดับเข้าสู่ระบบและสกัดข้อมูลโดยใช้ยูสเคส Data Extraction 6.ระบบบันทึกข้อมูลที่สกัด ที่อยู่ของไฟล์ข้อมูลที่สกัดได้จากแผนภาพลำดับ ที่อยู่ของคำอธิบายยูสเคสลงสู่ฐานข้อมูล 7. ระบบนำไฟล์ที่บันทึกชื่อเมท็อดที่ไม่ครอบคลุมมาใช้ในการหาไฟล์ข้อมูลที่สกัดจากแผนภาพลำดับ และคำอธิบายยูสเคสที่เกี่ยวข้องจากฐานข้อมูล 8. ระบบสร้างกราฟการทำงานจากแผนภาพลำดับที่เกี่ยวข้อง 9.ระบบใช้ข้อมูลจากคำอธิบายยูสเคสที่เกี่ยวข้องมาสร้างข้อมูลทดสอบของกรณีทดสอบ 10. ระบบบันทึกกรณีทดสอบลงสู่โฟลเดอร์ที่ผู้ทดสอบกำหนด.
เงื่อนไขภายหลัง	ระบบสร้างกรณีทดสอบและบันทึกลงส่งโฟลเดอร์ที่กำหนด

ตารางที่ 3.4 รายละเอียดยูสเคสการสกัดข้อมูล

ยูสเคส	Data Extracrtion
แอคเตอร์	-
รายละเอียดยูสเคส	เพื่อสกัดข้อมูลจากแผนภาพลำดับที่เข้าสู่ระบบ
ยูสเคสที่สัมพันธ์	-
เงื่อนไขก่อนหน้า	-
ขั้นตอน	1.แผนภาพลำดับถูกนำเข้าสู่ระบบ 2. เครื่องมือสกัดข้อมูลที่ต้องการจากแผนภาพลำดับ ได้แก่ ชื่อเมท็อด ลำดับการเรียกใช้เมท็อด คลาสของเมท็อด เงื่อนไขที่ปรากฏอยู่ในคอมบายแฟรกเมนต์ ชนิดของคอมบายแฟรกเมนท์ที่ปรา- กฏในแผนภาพลำดับ และ ลำดับการซ้อนกันของเงื่อนไขในแต่ละคอมบายแฟรกเมนต์ 3. เครื่องมือบันทึกข้อมูลที่สกัดได้ลงในไฟล์
เงื่อนไขภายหลัง	-

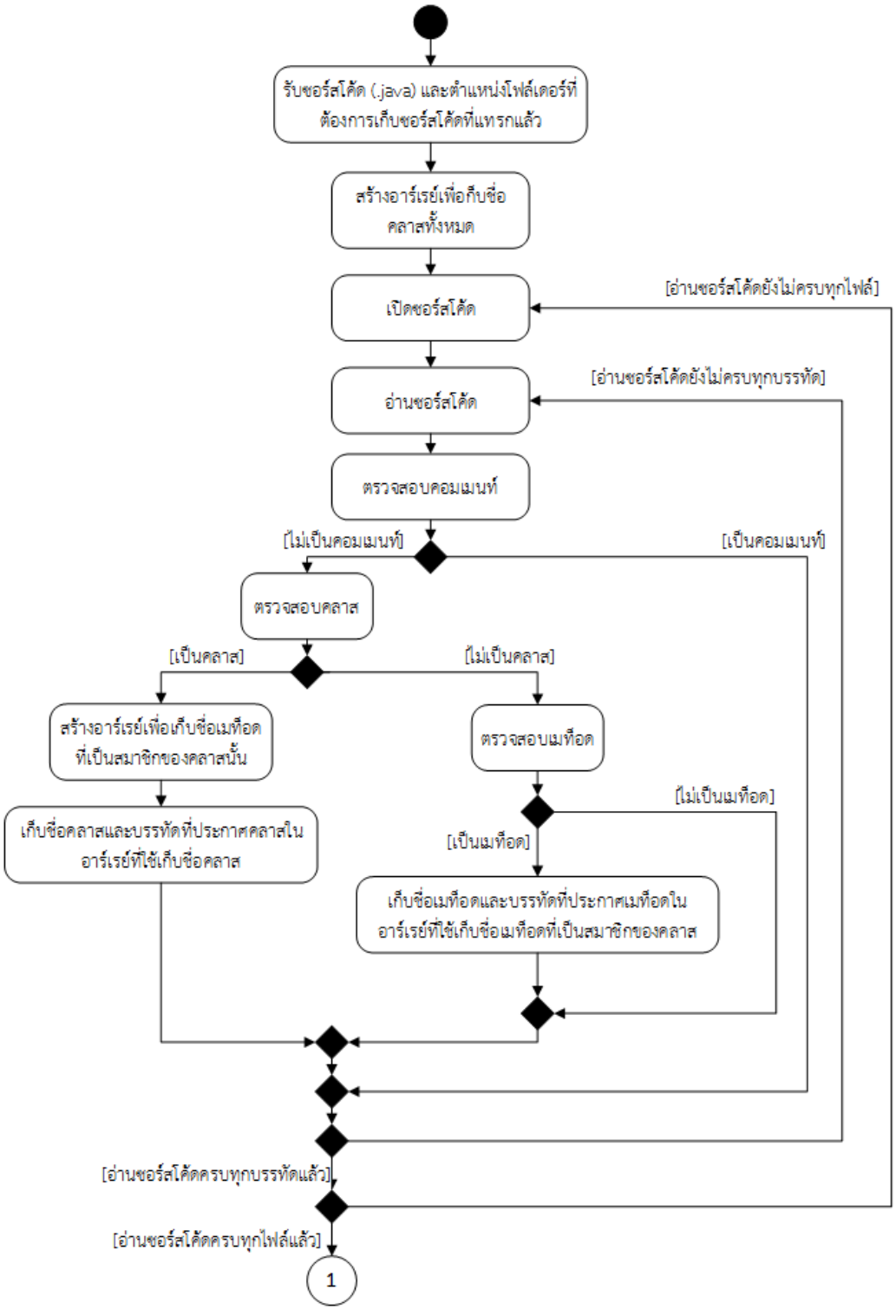
3.2.2 แผนภาพกิจกรรม

แผนภาพกิจกรรมเป็นแผนภาพที่แสดงกิจกรรมเบื้องต้นและความสัมพันธ์ระหว่างกิจกรรมของระบบ แผนภาพกิจกรรมของเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการประกอบไปด้วย 5 แผนภาพคือ แผนภาพกิจกรรมการแทรกซอร์สโค้ด ดังแสดงได้ดัง ภาพที่ 3-13 แผนภาพกิจกรรมการวิเคราะห์ความครอบคลุมของการทดสอบ ดังแสดงได้ดัง ภาพที่ 3-14 แผนภาพกิจกรรมการวิเคราะห์ข้อมูล ดังแสดงได้ดัง ภาพที่ 3-15 แผนภาพกิจกรรมการสร้างกราฟการทำงาน ดังแสดงได้ดัง ภาพที่ 3-16 และแผนภาพกิจกรรมการสร้างกรณีทดสอบ ดังแสดงได้ดัง ภาพที่ 3-17

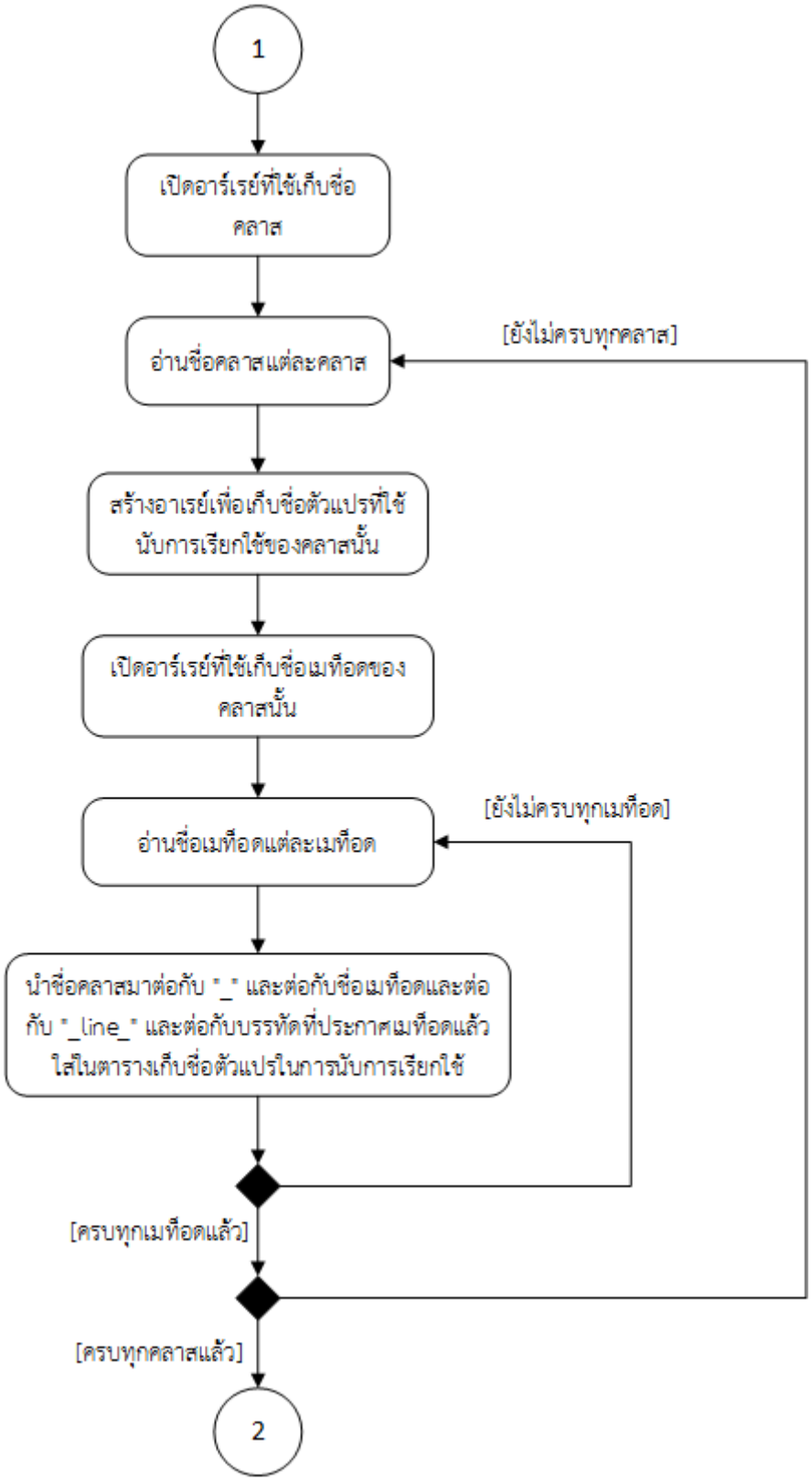
1) แผนภาพกิจกรรมการแทรกซอร์สโค้ด

แผนภาพกิจกรรมการแทรกซอร์สโค้ดในภาพที่ 3-13 แสดงให้เห็นถึงขั้นตอนการแทรกซอร์สโค้ด เริ่มจากเครื่องมือจะอ่านซอร์สโค้ดครั้งละบรรทัดเป็นจำนวน 2 รอบ รอบแรกเครื่องมือจะอ่านซอร์สโค้ดเพื่อหาว่าซอร์สโค้ดที่นำมาทดสอบนั้นมีกี่คลาส มีคลาสอะไรบ้าง แล้วเก็บชื่อคลาสไว้ในอาร์เรย์ที่ใช้ในการเก็บชื่อคลาสทั้งหมดในระบบ หลังจากนั้น จะเก็บชื่อเมท็อดของแต่ละคลาสว่ามีกี่เมท็อด อะไรบ้าง แล้วเก็บชื่อเมท็อดไว้ในอาร์เรย์ที่ใช้เก็บชื่อเมท็อด เมื่อเครื่องมืออ่านซอร์สโค้ดรอบแรกจนครบแล้ว เครื่องมือจะนำชื่อคลาสและชื่อเมท็อดที่เก็บได้จากการอ่านซอร์สโค้ดมาสร้างเป็นตัว

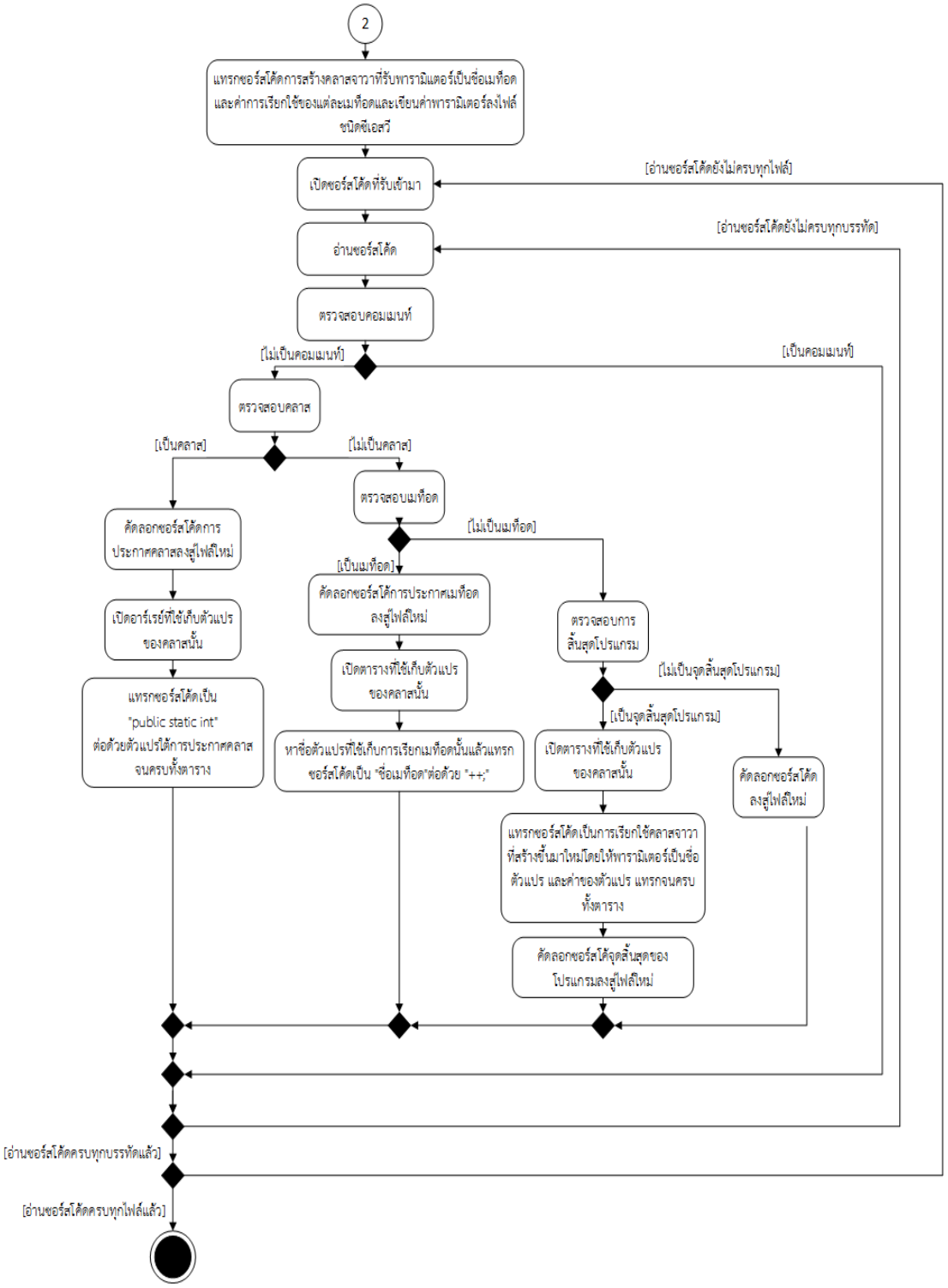
แปร แล้วระบบจะอ่านซอร์สโค้ดรอบที่สอง เพื่อนำตัวแปรที่สร้างขึ้นแทรกลงไปในซอร์สโค้ดที่ต้องการนำมาทดสอบ



ภาพที่ 3-13 แผนภาพกิจกรรมการแทรกซอร์สโค้ด



ภาพที่ 3-13 แผนภาพกิจกรรมของการแทรกซอร์สโค้ด(ต่อ)



ภาพที่ 3-13 แผนภาพกิจกรรมของการแทรกซอร์สโค้ด(ต่อ)

2) แผนภาพกิจกรรมการวิเคราะห์ความครอบคลุมของการทดสอบ

แผนภาพกิจกรรมการวิเคราะห์ความครอบคลุมของการทดสอบในภาพที่ 3-14 แสดงให้เห็นวิธีการวิเคราะห์ความครอบคลุมการทดสอบ โดยเครื่องมือจะเปิดไฟล์สรุปการเรียกใช้เมท็อด หลังจากนั้นจะอ่านชื่อตัวแปรและจำนวนการเรียกใช้แล้วนำชื่อตัวแปรมาหาว่าเป็นตัวแปรของคลาสใดและเมท็อดใด หลังจากนั้นเครื่องมือจะบันทึกชื่อคลาสและเมท็อดทั้งหมดใส่ในอาร์เรย์ แล้วบันทึกชื่อคลาสและเมท็อดที่มีการเรียกใช้เป็น 0 โดยแยกเป็นอาร์เรย์อีกชุดหนึ่ง เมื่อบันทึกชื่อคลาสและเมท็อดเสร็จแล้ว เครื่องมือจะคำนวณความครอบคลุมคลาสและความครอบคลุมเมท็อด หลังจากนั้นจึงส่งผลการคำนวณและชื่อคลาสรวมทั้งชื่อเมท็อดที่ไม่ครอบคลุมให้ผู้ทดสอบได้รับทราบ

3) แผนภาพกิจกรรมการวิเคราะห์ข้อมูล

แผนภาพกิจกรรมการวิเคราะห์ข้อมูลในภาพที่ 3-15 แสดงให้เห็นวิธีการวิเคราะห์ข้อมูล โดยเครื่องมือจะอ่านแผนภาพลำดับ หลังจากนั้นเครื่องมือจะเก็บชื่อเมท็อดและคลาสรวมทั้งลำดับการเรียกในแผนภาพลำดับลงในอาร์เรย์ เมื่อเก็บชื่อเมท็อดและคลาสครบทั้งหมดแล้วจะนำอาร์เรย์ที่เก็บชื่อเมท็อดนั้นมาหาว่าเมท็อดแต่และเมท็อดมีเงื่อนไขการเกิดอย่างไรบ้าง และเงื่อนไขการเรียกเมท็อดเป็นชนิดใด หลังจากนั้นจึงนำข้อมูลที่อยู่ในอาร์เรย์คือ ชื่อเมท็อด ชื่อคลาส เงื่อนไขการเรียกเมท็อด รวมทั้ง ลำดับการเรียกเมท็อด ชื่อแผนภาพลำดับใส่ลงในไฟล์สรุปข้อมูลที่สกัดได้ และนำชนิดของเงื่อนไขการเกิดใส่ลงในไฟล์ชนิดของเงื่อนไข

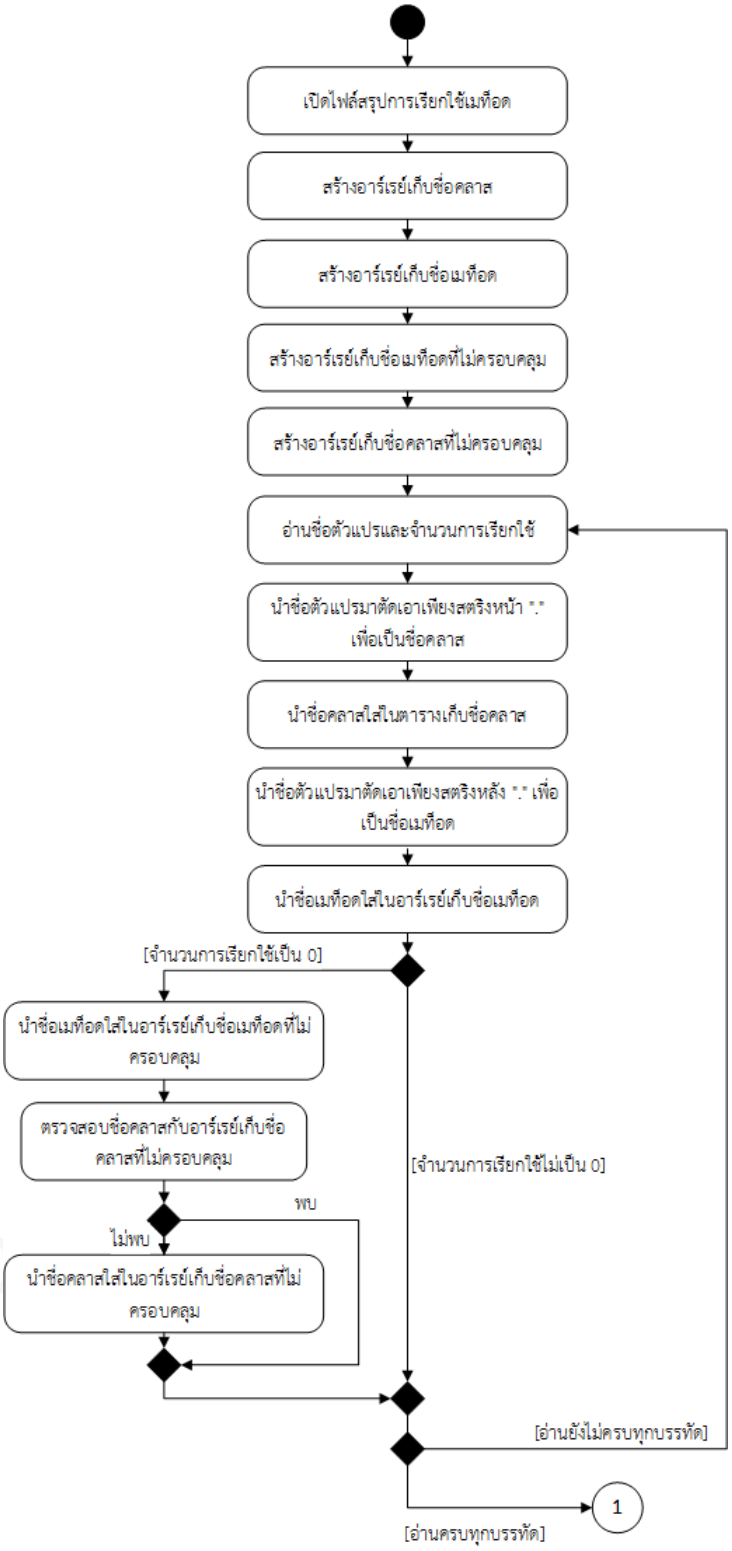
4) แผนภาพกิจกรรมการสร้างกราฟการทำงาน

แผนภาพกิจกรรมการสร้างกราฟการทำงานในภาพที่ 3-16 แสดงให้เห็นถึงวิธีการสร้างกราฟการทำงาน โดยเครื่องมือจะสร้างกราฟหลัก ในขั้นต้น กราฟหลักจะมี 2 โหนดคือ โหนดตั้งต้น และโหนดสิ้นสุด หลังจากนั้นเครื่องมือจะอ่านเมท็อดและเงื่อนไขการเกิดเมท็อดในไฟล์สรุปข้อมูลที่สกัดได้แล้วนำเมท็อดและเงื่อนไขการเกิดใส่ลงไปในสแตกจนครบ หลังจากนั้นเครื่องมือจะนำอีลิเมนท์ที่อยู่แสตกออกมาแล้วตรวจสอบว่าเป็นอีลิเมนท์ชนิดใด (เมท็อด เงื่อนไขชนิดอัลเทอร์เนทีฟ เงื่อนไขชนิดลูป และเงื่อนไขชนิดออปชัน) แล้วสร้างกราฟย่อยตามชนิดของอีลิเมนท์นั้นๆ หลังจากนั้นจึงนำกราฟย่อยมาต่อกับกราฟหลัก ซึ่งจะทำไปจนครบทุกเมท็อด

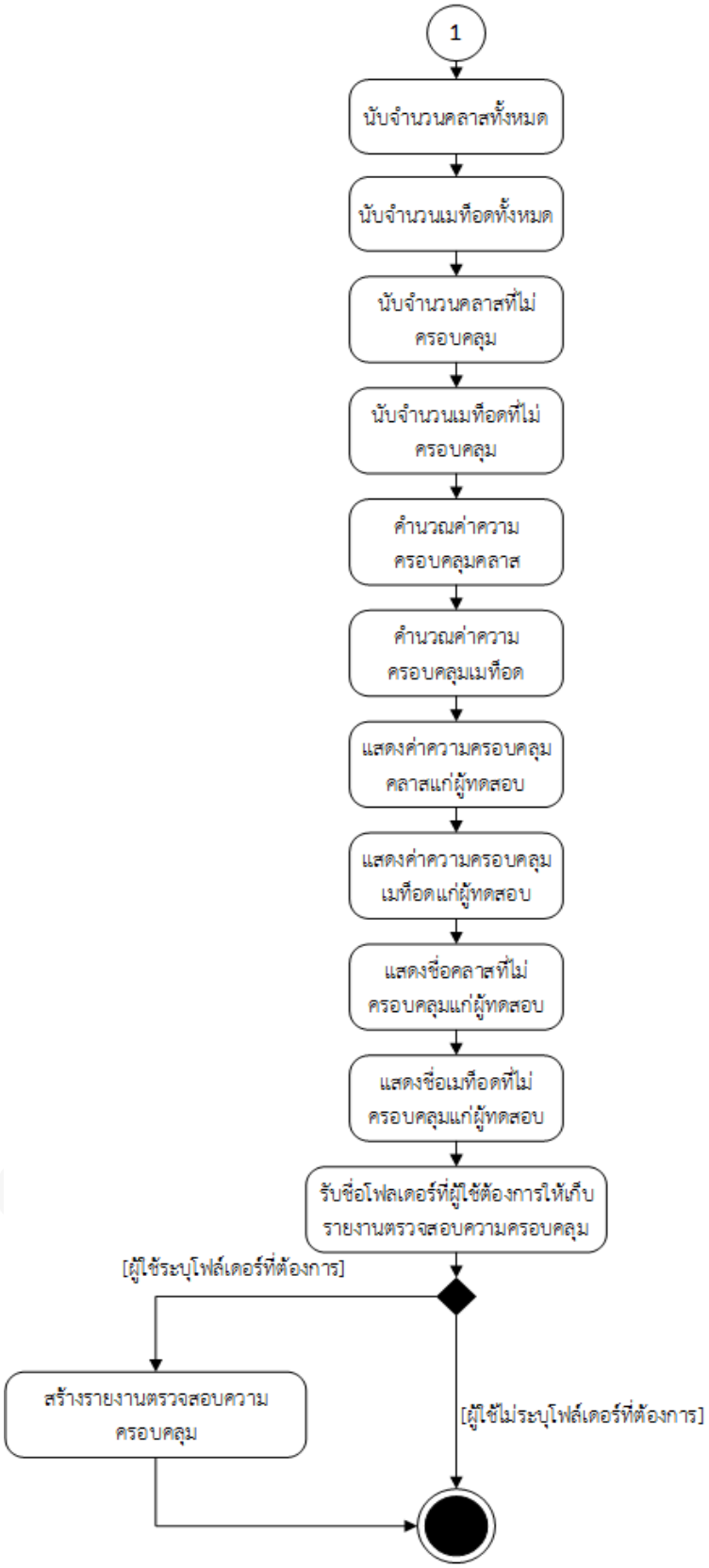
5) แผนภาพกิจกรรมการสร้างกรณีทดสอบ

แผนภาพกิจกรรมการสร้างกรณีทดสอบในภาพที่ 3-17 แสดงให้เห็นถึงวิธีการสร้างกรณีทดสอบ โดยเครื่องมือจะนำชื่อเมท็อดที่ไม่ครอบคลุมมาเส้นทางการเรียกจากกราฟการทำงานที่สร้างขึ้น แล้วบันทึกเงื่อนไขการเรียกเมท็อด หลังจากนั้นจะนำเงื่อนไขไปหาในช่อง Condition เริ่มต้นของ SuccessScenario และ AlternativeScenario ว่าอยู่ในหมายลขใด แล้วบันทึก Condition เริ่มต้นนั้นเพื่อใช้เป็นข้อความอธิบายกรณีทดสอบ และบันทึกเลขหน้า Condition เริ่มต้นนั้นด้วย หลังจากนั้นนำตัวเลขไปหาในช่อง Post-condition ที่ตัวเลขเดียวกันเพื่อใช้เป็นผลลัพธ์ที่คาด และนำไปหาข้อมูลทดสอบในช่อง Input value หลังจากนั้นจึงสุ่มค่าข้อมูลทดสอบ กำหนดหมายเลขกรณีทดสอบ กำหนดชื่อกรณีทดสอบ และสุดท้ายจึงสร้างกรณีทดสอบในรูปแบบไฟล์เอชทีเอ็มแอล

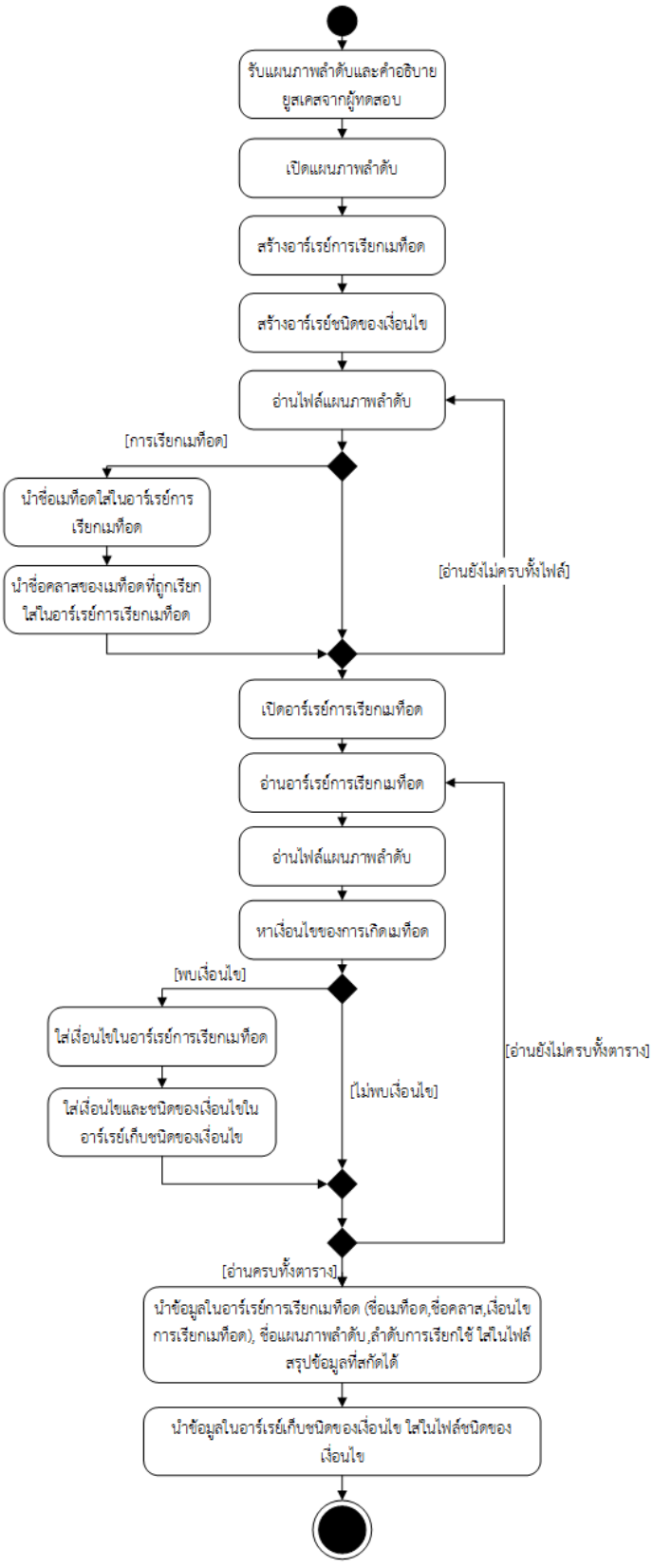
Chulalongkorn University



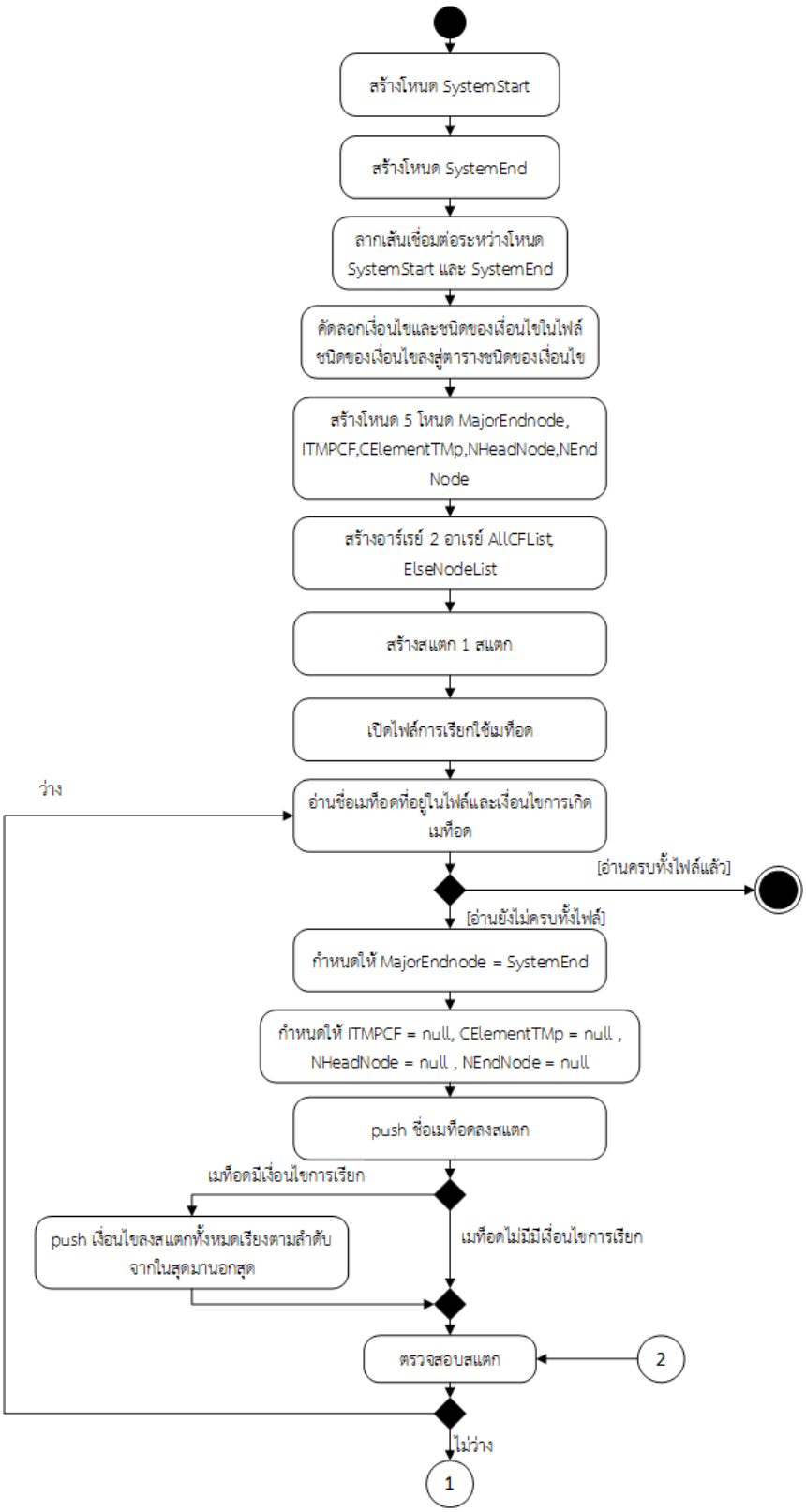
ภาพที่ 3-14 แผนภาพกิจกรรมการวิเคราะห์ความครอบคลุมของการทดสอบ



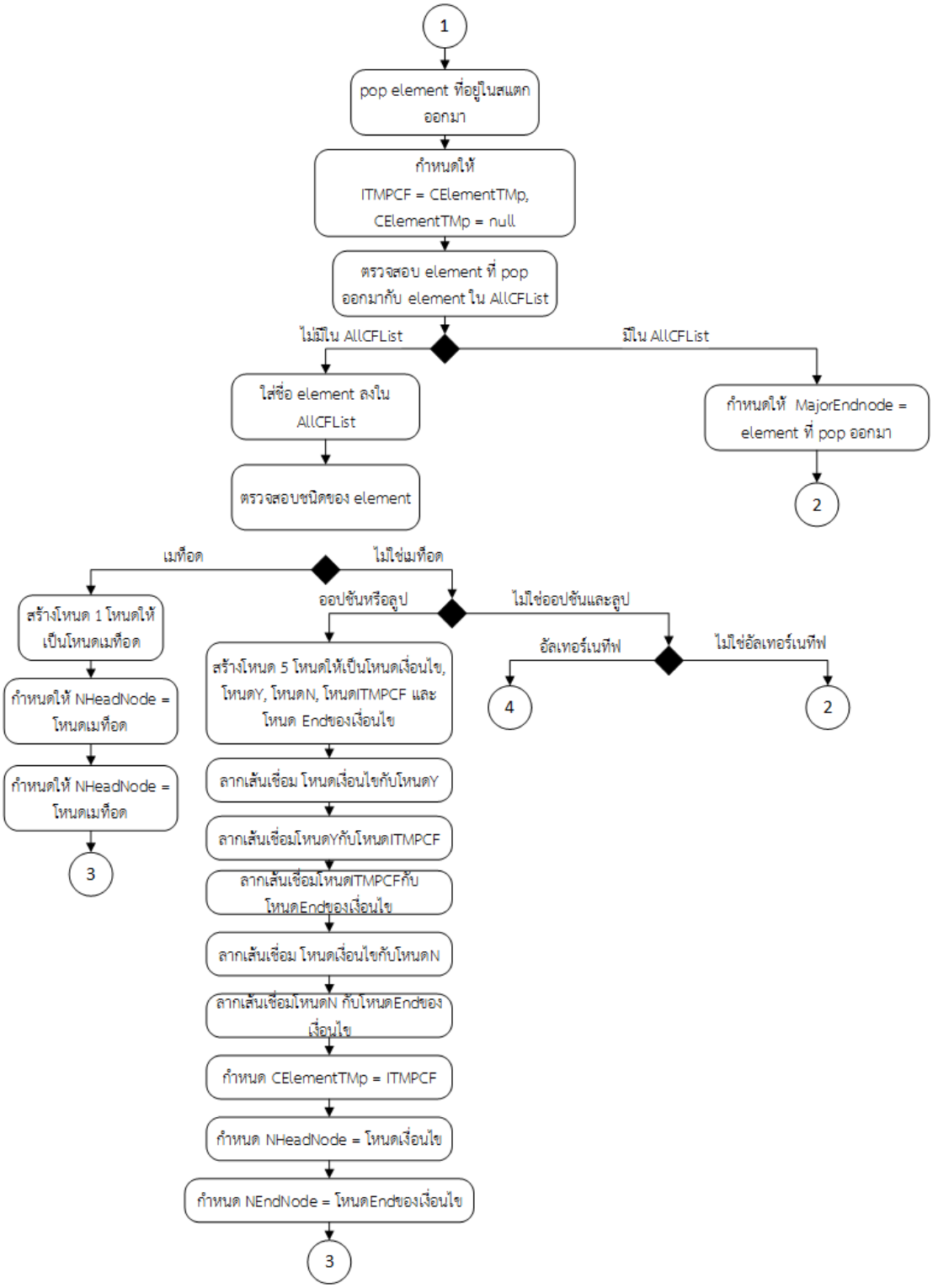
ภาพที่ 3-14 แผนภาพกิจกรรมการวิเคราะห์ความครอบคลุมของการทดสอบ(ต่อ)



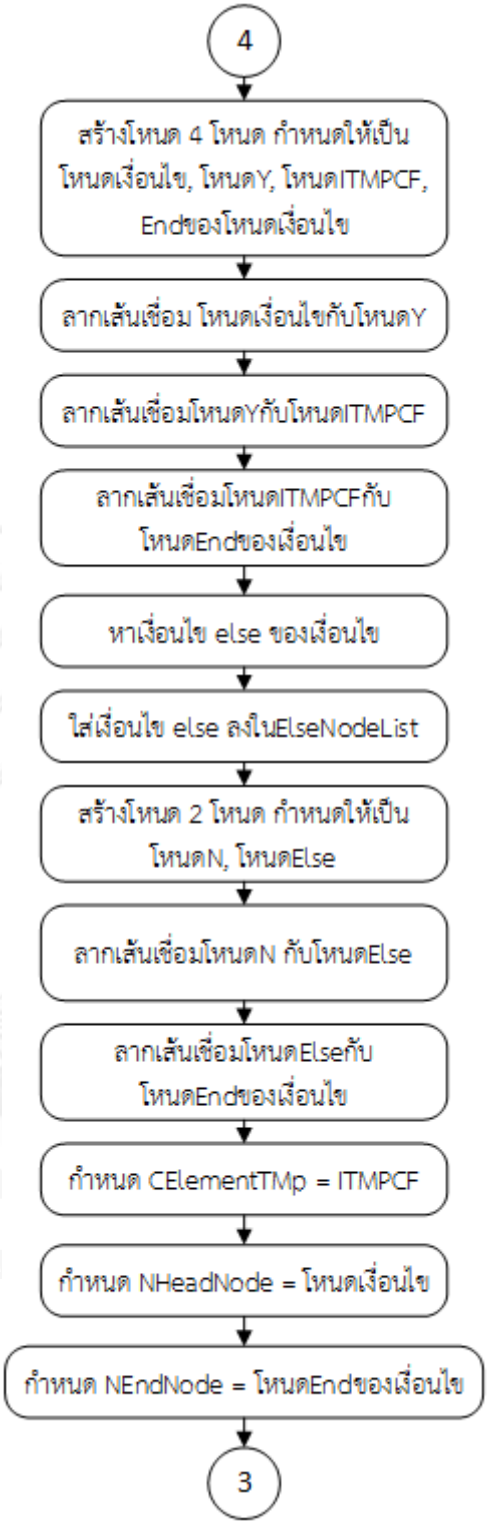
ภาพที่ 3-15 แผนภาพกิจกรรมการวิเคราะห์ข้อมูล



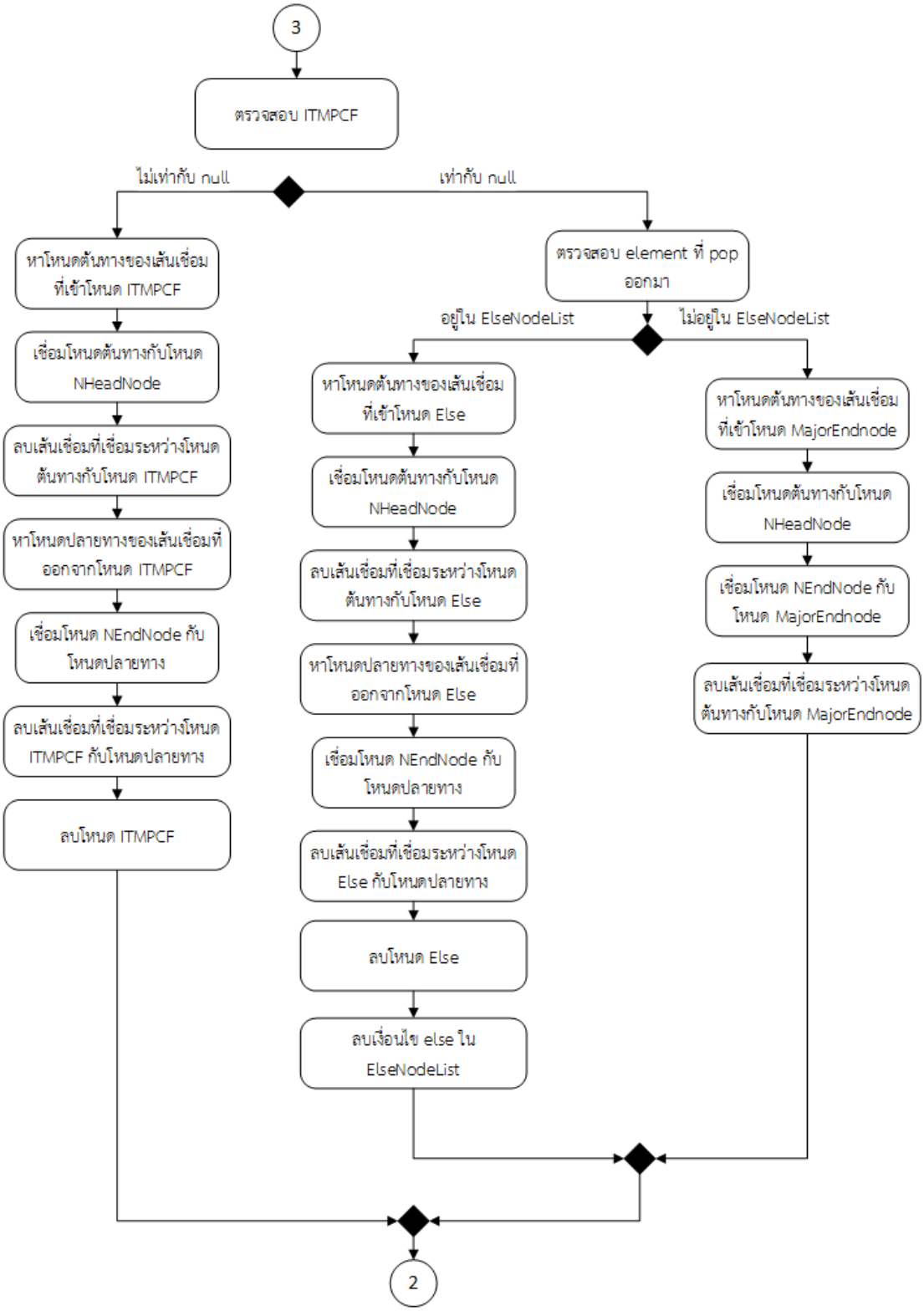
ภาพที่ 3-16 แผนภาพกิจกรรมการสร้างกราฟการทำงาน



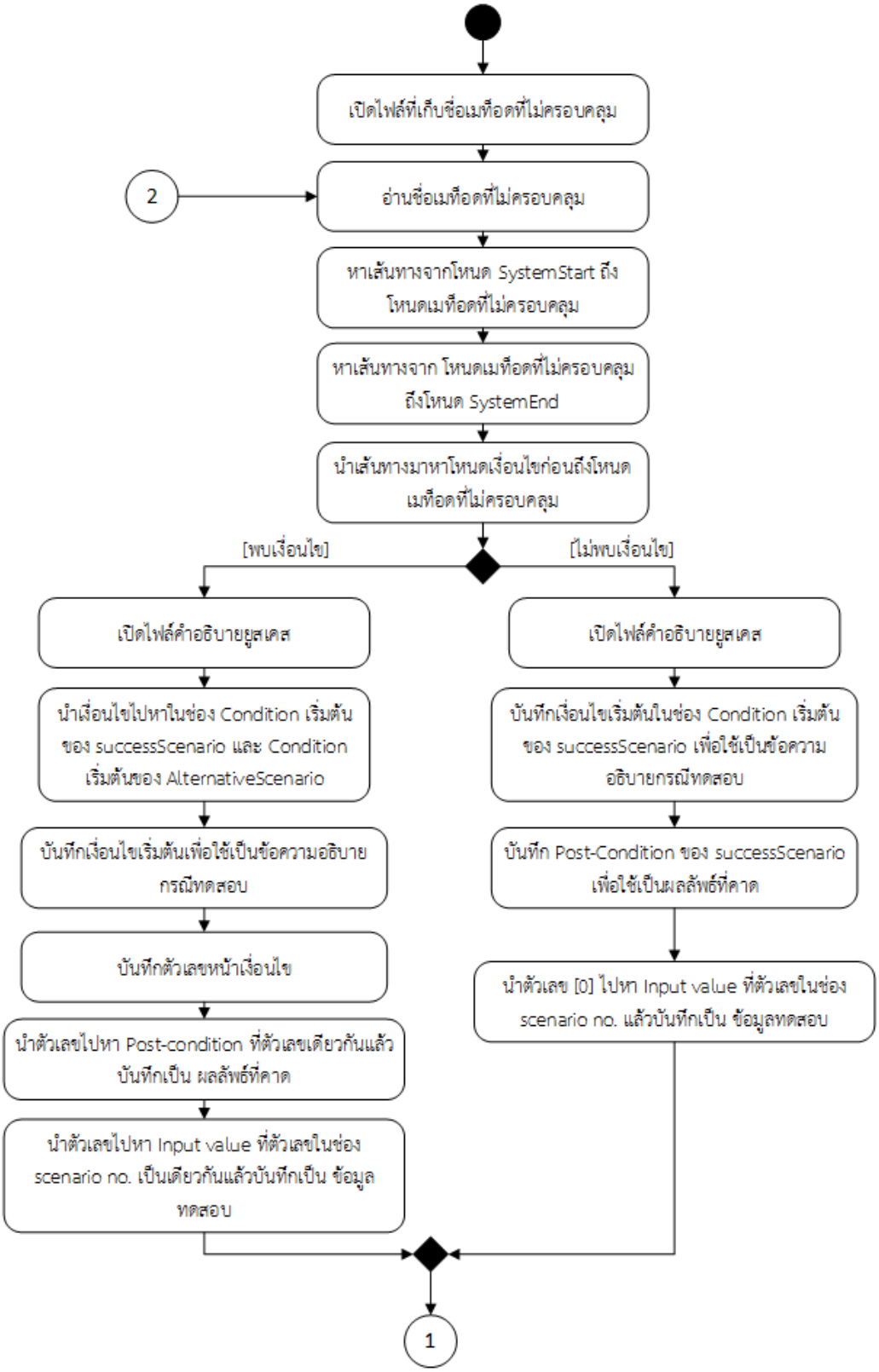
ภาพที่ 3-16 แผนภาพกิจกรรมการสร้างกราฟการทำงาน(ต่อ)



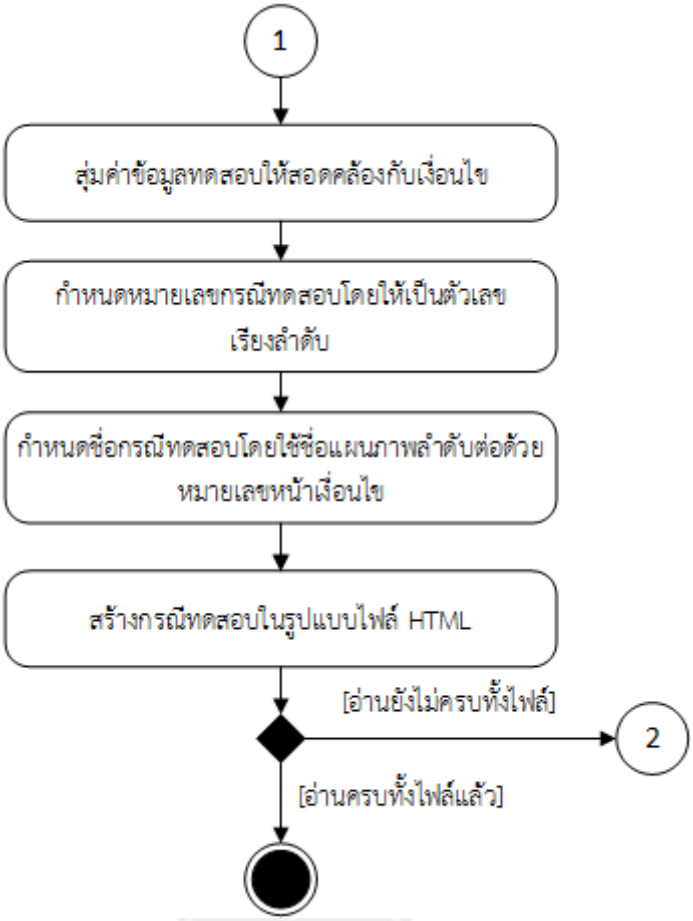
ภาพที่ 3-16 แผนภาพกิจกรรมการสร้างกราฟการทำงาน(ต่อ)



ภาพที่ 3-16 แผนภาพกิจกรรมการสร้างกราฟการทำงาน(ต่อ)



ภาพที่ 3-17 แผนภาพกิจกรรมการสร้างกรณีทดสอบ



ภาพที่ 3-17 แผนภาพกิจกรรมการสร้างกรณีทดสอบ(ต่อ)

3.2.3 แผนภาพคลาส

แผนภาพคลาสเป็นแผนภาพที่แสดงรายละเอียดของคลาสและความสัมพันธ์ระหว่างคลาส ซึ่งแผนภาพคลาสของเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการแสดงได้ดัง ภาพที่ 3-18 ซึ่งมีรายละเอียดดังต่อไปนี้

1) คลาส About คือคลาสที่ทำหน้าที่ติดต่อกับผู้ทดสอบในส่วนของการแสดงข้อมูลเกี่ยวกับเครื่องมือและผู้พัฒนา รายละเอียดของ About แสดงดังภาพที่ 3-19

2) คลาส CodeInstrumentation คือคลาสที่ทำหน้าที่แทรกซอร์สโค้ดลงในซอร์สโค้ดที่ต้องการดำเนินการทดสอบ รายละเอียดของคลาส CodeInstrumentation แสดงได้ดังภาพที่ 3-20

3) คลาส CollectedClassMethodData คือคลาสที่ทำหน้าที่เตรียมข้อมูลที่สกัดได้จากแผนภาพลำดับและคำอธิบายยูสเคสเพื่อบันทึกลงสู่ฐานข้อมูล รายละเอียดของคลาส CollectedClassMethodData แสดงได้ดังภาพที่ 3-21

4) คลาส ConditionExtractformGraph คือคลาสที่ทำหน้าที่บันทึกเงื่อนไขที่ปรากฏอยู่ในเส้นทางที่มีเมท็อดที่ไม่ครอบคลุมจากการทดสอบหลังจากสร้างกราฟการทำงานได้แล้ว รายละเอียดของคลาส ConditionExtractformGraph แสดงได้ดังภาพที่ 3-22

5) คลาส ConditionMNG คือคลาสที่ทำหน้าที่สกัดข้อมูลที่เป็นเงื่อนไขคำอธิบายยูสเคส เพื่อนำมาใช้ในการสร้างข้อมูลทดสอบ รายละเอียดของคลาส ConditionMNG แสดงได้ดังภาพที่ 3-23

6) คลาส CollectedXMLData คือคลาสที่ทำหน้าที่สกัดข้อมูลที่จำเป็นต้องใช้ในการสร้างกรณีทดสอบจากแผนภาพลำดับ รายละเอียดของคลาส CollectedClassMethodData แสดงได้ดังภาพที่ 3-24

7) คลาส CoverageAnalysisNewProject เป็นคลาสที่วิเคราะห์ความครอบคลุมของการทดสอบ ในกรณีที่วิเคราะห์ความครอบคลุมการทดสอบเป็นครั้งแรกทำหน้าที่คำนวณความครอบคลุมคลาสและความครอบคลุมเมท็อด รวมทั้งระบุคลาสและเมท็อดที่ไม่ครอบคลุม รายละเอียดของคลาส CoverageAnalysisNewProject แสดงได้ดัง ภาพที่ 3-25

8) คลาส CoverageAnalysisOpenProject เป็นคลาสที่วิเคราะห์ความครอบคลุมของการทดสอบในกรณีที่วิเคราะห์ความครอบคลุมการทดสอบครั้งต่อมาหลังจากทดสอบเพิ่มเติมด้วยกรณีทดสอบที่เครื่องมือสร้างขึ้น ทำหน้าที่คำนวณความครอบคลุมคลาสและความครอบคลุมเมท็อด รวมทั้งระบุคลาสและเมท็อดที่ไม่ครอบคลุมรายละเอียดของคลาส CoverageAnalysisOpenProject แสดงได้ดังภาพที่ 3-26

9) คลาส CoverageReportGen เป็นคลาสที่ทำหน้าที่สร้างรายงานความครอบคลุมของการทดสอบในรูปแบบ html รายละเอียดของคลาส CoverageReportGen แสดงได้ดังภาพที่ 3-27

10) คลาส csvextract เป็นคลาสที่ทำหน้าที่เป็นตัวกลางในการจัดการสร้างกรณีทดสอบตั้งแต่การสกัดข้อมูลจากคำอธิบายยูสเคส หาเงื่อนไขที่ตรงกับเงื่อนไขที่ได้จากเส้นทางของกราฟการทำงานและผลของเงื่อไข รวมทั้งบันทึกกรณีทดสอบในรูปแบบ html รายละเอียดของคลาส csvextract แสดงได้ดัง ภาพที่ 3-28

11) คลาส DatabaseConnection เป็นคลาสที่ทำหน้าที่จัดการเกี่ยวกับฐานข้อมูลทั้งหมดตั้งแต่การนำข้อมูลแผนภาพลำดับและคำอธิบายยูสเคสเข้าสู่ฐานข้อมูล รวมถึงการค้นหาแผนภาพลำดับและคำอธิบายยูสเคสที่เกี่ยวข้อง รายละเอียดของคลาส DatabaseConnection แสดงได้ดังภาพที่ 3-29

12) คลาส Main_DataExtraction เป็นคลาสที่ทำหน้าที่เป็นตัวกลางในการแปลงข้อมูลจากแผนภาพลำดับที่สกัดได้เข้าสู่คลาส GraphBuilder เพื่อสร้างกราฟการทำงานจากแผนภาพลำดับ หาเส้นทางที่มีเมท็อดที่ไม่ครอบคลุมจากการทดสอบปรากฏอยู่ และเป็นตัวกลางในการส่งเส้นทางที่พบไปหาเงื่อนไขที่ปรากฏในเส้นทาง รวมทั้งส่งเงื่อนไขที่หาได้เพื่อนำไปสร้างกรณีทดสอบต่อไป รายละเอียดของคลาส Main_DataExtraction แสดงได้ดัง ภาพที่ 3-30

13) คลาส GraphBuilder เป็นคลาสที่ทำหน้าที่สร้างกราฟการทำงานจากแผนภาพลำดับ รายละเอียดของคลาส GraphBuilder แสดงได้ดัง ภาพที่ 3-31

14) คลาส MainPage เป็นคลาสที่ทำหน้าที่ติดต่อผู้ทดสอบ โดยคลาสนี้เป็นคลาสหลักของเครื่องมือเพื่อในไปสู่การใช้งานทั้งการสร้างโปรเจคใหม่ หรือเปิดโปรเจคเดิมเพื่อตรวจสอบความครอบคลุมหลังจากได้ทดสอบด้วยกรณีทดสอบที่เครื่องมือสร้างขึ้น รายละเอียดของคลาส MainPage แสดงได้ดังภาพที่ 3-32

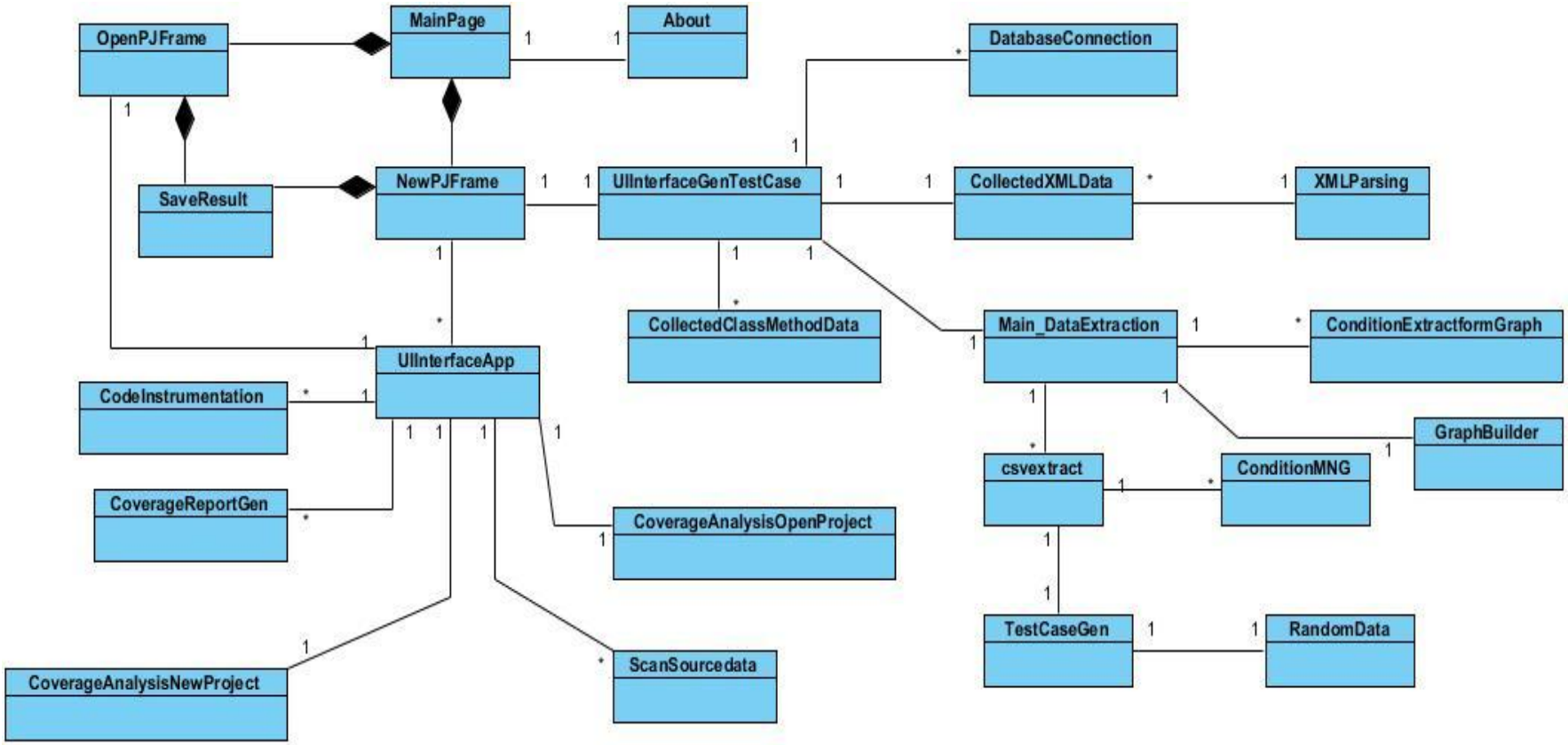
15) คลาส NewPJFrame เป็นคลาสที่ติดต่อผู้ทดสอบในการสร้างโปรเจคใหม่เพื่อให้ผู้ทดสอบนำ ซอร์สโค้ดเข้าสู่ระบบเพื่อแทรกซอร์สโค้ด แสดงผลตรวจสอบความครอบคลุมการทดสอบ รวมทั้งให้ผู้ทดสอบใส่แผนภาพลำดับและคำอธิบายยูสเคสเพื่อสร้างกรณีทดสอบ รายละเอียดของคลาส NewPJFrame แสดงได้ดังภาพที่ 3-33

16) คลาส OpenPJFrame เป็นคลาสที่ติดต่อผู้ทดสอบในการเปิดโปรเจคเดิมเพื่อแสดงผลการตรวจสอบความครอบคลุมการทดสอบหลังจากที่ผู้ทดสอบได้ใช้กรณีทดสอบที่เครื่องมือสร้างขึ้นไปทดสอบ ทดสอบ รายละเอียดของคลาส OpenPJFrame แสดงได้ดังภาพที่ 3-34

17) คลาส RandomData เป็นคลาสที่ทำหน้าที่สุ่มค่าเพื่อใช้เป็นค่าของข้อมูลทดสอบ รายละเอียดของคลาส RandomData แสดงได้ดังภาพที่ 3-35

18) คลาส SaveResult เป็นคลาสที่ทำหน้าที่ติดต่อกับผู้ทดสอบเพื่อให้ผู้ทดสอบระบุโฟลเดอร์ที่ต้องการให้ระบบบันทึกรายงานความครอบคลุมการทดสอบในรูปแบบ html รายละเอียดของคลาส SaveResult แสดงได้ดังภาพที่ 3-36

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



ภาพที่ 3-18 แผนภาพคลาสของเครื่องมือ

About
+run() : void +main() : void +About() : void

ภาพที่ 3-19 คลาส About

CodeInstrumentation
+chk_class : Boolean +chk_comment : Boolean +chk_method : Boolean +chk_constructort : Boolean +ChkMethodInstr : Boolean +ChkClassInstr : Boolean +writesumfile : Boolean +line_count : integer +tmp_classname : String +tmp_copy_line : String +tmp_copy_stringline : String
+ReadInsedSource (String, String, String, ArrayList<String>) : void +CreateTmpDirectory(String) : String +WriteFileSuumary(String) : void +CheckClass(String, String, ArrayList<String>, String) : void +InsertClass(String, ArrayList<String>, String, String) : void +MethodInstrumented(String, String, ArrayList<String>, String) : void +WritecounterMethodcall(String, String, String) : void +WriteImportLib(String, String) : void +WriteInstrumentFile(String, String, String) : void +ChkComment (String) : Boolean +ExitInstrumented(String, String, ArrayList<String>, String) : void +CounterCallInstrumented(String, String, ArrayList<String>) : void

ภาพที่ 3-20 คลาส CodeInstrumentation

CHULALONGKORN UNIVERSITY

CollectedClassMethodData
+ClassName : ArrayList<String> +MethodName : ArrayList<String>
+SDNameExtract(String, String) : String +MeNametoMeID(String, String) : String +ChooseUCDFile(String, String) : String +UCDNameExtract (String, String) : String + ClassandMethodExtract (String, String) : void +UncoverMethodExtract (String) : ArrayList<String>

ภาพที่ 3-21 คลาส CollectedClassMethodData

ConditionExtractformGraph
+Conditioneleex : ArrayList<String> +YesConditioneleex: ArrayList<String>
+conditionextract (String) : void

ภาพที่ 3-22 คลาส ConditionExtractformGraph

ConditionMNG
+SuccessConditionExtract(String) : String +AlternativeConditionExtract(String) : String + GetConditionExisting (String, ArrayList<String>) : boolean +GetScenarioFromUCD(String ,String) : String

ภาพที่ 3-23 คลาส ConditionMNG

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาพที่ 3-24 คลาส CollectedXMLData

CoverageAnalysisNewProject
+VariableName : ArrayList<String> +VariableCounting : ArrayList<String> +UncoverMethodName : ArrayList<String> +UncoverClassName : ArrayList<String> +AllClassName : ArrayList<String> +MethodCoverage : float +ClassCoverage : float
+ExtractCoverageData (String) : void +MethodCoverageAna (ArrayList<String>, ArrayList<String>) : void +UncoverClassExtract(ArrayList<String>, ArrayList<String> , ArrayList<String>) : void +ClassCoverageAna (ArrayList<String> , ArrayList<String>) : void

ภาพที่ 3-25 คลาส CoverageAnalysisNewProject

CoverageAnalysisOpenProject
+VariableName : ArrayList<String> +VariableCounting : ArrayList<String> +UncoverMethodName : ArrayList<String> +UncoverClassName : ArrayList<String> +AllClassName : ArrayList<String> +VariableNameDu : ArrayList<String> +VariableCountingDu : ArrayList<String> +MethodCoverage : float +ClassCoverage : float
+ExtractCoverageData (String) : void +MethodCoverageAna (ArrayList<String>, ArrayList<String>) : void +UncoverClassExtract(ArrayList<String>, ArrayList<String> , ArrayList<String>) : void +ClassCoverageAna (ArrayList<String> , ArrayList<String>) : void

ภาพที่ 3-26 คลาส CoverageAnalysisOpenProject

CoverageReportGen
+WriteCoverageReport(String,float, float, ArrayList<String>, ArrayList<String>) : void +WriteUnCoverMethod (String , ArrayList<String>) : void

ภาพที่ 3-27 คลาส CoverageReportGen

csvextract
+GenedCondition : ArrayList<String> +ConditionSplit : ArrayList<String> +InputVar : ArrayList<String> +InputValue : ArrayList<String> +GentDataVar : ArrayList<String> +GentDataVaLue : ArrayList<String> +OutputTestCaseName : String +ScenarioOutput : String +ConditionName : String +OutputInitialVal : String
+ UCDCSVExtract(String ,ArrayList<String>, String, String) : void

ภาพที่ 3-28 คลาส csvextract

DatabaseConnection
+connect : Connection
+getconnection : Connection +selectlocation (String) : String +inserttable (String, String ,String, String) : void +insertSDLocation (String, String) : void +insertUCDLocation (String , String) : void

ภาพที่ 3-29 คลาส DatabaseConnection

Main_DataExtraction
+ConditionUsage : ArrayList<String>
+GraphDataExtraction(String ,String ,String) : void

ภาพที่ 3-30 คลาส Main_DataExtraction

CHULALONGKORN UNIVERSITY

GraphBuilder
+directedGraph : DirectedGraph<String, DefaultEdge> +OperatingCF : ArrayList<String> +AllCFName : ArrayList<String> +OperatingCondition : ArrayList<String> +ElseCounting : ArrayList<String> +OperatingGuardtype : ArrayList<String> +MSSID : ArrayList<String> +MSSName : ArrayList<String> +AlrConnectednodeList : ArrayList<String> +ElseNodeList : ArrayList<String> +PopElement : String +iTmp : String +CiTmp : String +NHeadNode : String +NEndNode : String +MajEndNode :String +MSSCFStack : Stack<String>
+initialvariable(String) : void +searchAlrElseNode(String) : String +searchAllCF(String) :boolean +CopyCFData(ArrayList<String>, ArrayList<String>) : void +GetAllCF(String) : String +GetCondition(String) : String +GetGuardType(String) :String +GetElse(String) : String +ConnectedNode(String, String) : void +CopyMSSIDandMSSName(String) : void +ConvertMSSIDtoMSSName(String) :String +ExtractMSSData (String) : void +PushMSSandCFtoStack(String, String) : void +PushMSSandCFtoStack(String , String[]) :void +PushMSStoStack(String) :void +PopStack(Stack<String>) : void +MSSCreateSubGraph(String) : void +OPTCreateSubGraph(String) : void +ALTCreateSubGraph(String) : void

ภาพที่ 3-31 คลาส GraphBuilder

MainPage
-contentPane : Jpanel -desktopPane : JDesktopPane
+run() : void +main() : void +MainPagee() : void

ภาพที่ 3-32 คลาส MainPage

NewPJFrame
-NewPJFrame : Jpanel -ClassCoverageOutput : JTextField -MethodCoverage : JTextField -SDfolderTxtField : JTextField -UCDfoldertextField : JTextField -TCFoldertextField : JTextField -FolderNameTxf : JTextField -instrFoldertxt : JTextField -CoverageAna : JPanel -TestCaseGen : JPanel -CoverageAnaPanel : JPanel -FillFilePanel : JPanel -CodeInsPanel : JPanel +fc : JFileChooser +folderchoose: JFileChooser +choosertitle : String +files : File[] +Outputfolder : String +SDFolder : String +UCDFolder : String +TestcaseFolder : String
+run() : void +main() : void +NewPJFrame() : void

ภาพที่ 3-33 คลาส NewPJFrame

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

OpenPJFrame
-NewPJFrame : Jpanel -model : DefaultTableModel -ClassCoverageOutput : JTextField -MethodCoverage : JTextField - instrFoldertxt : JTextField -tabbedPane : JTabbedPane +fc : JFileChooser +folderchoose : JFileChooser +choosertitle : String +files : File[] +Outputfolder : String
+run() : void +main() : void +OpenPJFrame() : void

ภาพที่ 3-34 คลาส OpenPJFrame

RandomData
+RandomString(String) : String +RandomStringUnFixsize() : String +RandomIntFromSize(String) : integer +RandomIntUnFixSize() : integer +RandomIntFromMinMax(String, String) : integer +RandomFloatFromSize (String) : float +RandomFloatUnFixSize () : float +RandomFloatFromMinMax (String, Srtring) : float +RandomBool () : boolean

ภาพที่ 3-35 คลาส RandomData

SaveResult
-contentPane : Jpanel -textField : JTextField
+run() : void +main() : void +SaveResult() : void

ภาพที่ 3-36 คลาส SaveRresult

19) คลาส ScanSourcedata เป็นคลาสที่ทำหน้าที่เก็บข้อมูลของซอร์สโค้ดที่นำมาทดสอบว่ามีคลาสใดบ้าง และแต่ละคลาสประกอบไปด้วยเมท็อดอะไรบ้าง เพื่อใช้ในการสร้างตัวแปรที่ใช้นับการเรียกใช้เมท็อด รายละเอียดของคลาส ScanSourcedata แสดงได้ดังภาพที่ 3-37

20) คลาส TestCaseGen เป็นคลาสที่ทำหน้าที่บันทึกกรณีทดสอบในรูปแบบของ html รายละเอียดของคลาส TestCaseGen แสดงได้ดังภาพที่ 3-38

21) คลาส UIInterfaceApp เป็นคลาสตัวกลางที่เชื่อมระหว่างคลาสที่เป็นส่วนต่อประสานของผู้ทดสอบในการแทรกซอร์สโค้ดและตรวจสอบความครอบคลุม กับส่วนที่ทำงานในการแทรกซอร์สโค้ดและตรวจสอบความครอบคลุม รายละเอียดของคลาส UIInterfaceApp แสดงได้ดังภาพที่ 3-39

22) คลาส UIInterfaceGenTestCase เป็นคลาสที่ทำหน้าที่เป็นตัวกลางเชื่อมระหว่างคลาสที่เป็นส่วนต่อประสานของผู้ทดสอบในการสร้างกรณีทดสอบ กับส่วนที่ทำงานในการสร้างกรณีทดสอบ รายละเอียดของคลาส UIInterfaceApp แสดงได้ดังภาพที่ 3-40

23) คลาส XMLParsing เป็นคลาสที่ทำหน้าที่เป็นตัวกลางในการจัดการการสกัดข้อมูลจากแผนภาพลำดับ รายละเอียดของคลาส XMLParsing สามารถแสดงได้ดังภาพที่ 3-41

ScanSourcedata
-comment_status : boolean **-tmp_classname : String** **-tmp_methodname : String** **-setclass : boolean** **-line_count : integer** **+ClassData : ArrayList<String>** **+AllClassName : ArrayList<String>** **+AllObjectName : ArrayList<String>**
+KeepSourceData (String) : void **-ChkComment (String) : boolean** **-ChkEvntHdl(String) :integer** **+ChkClass(String, Boolean) : void** **+ChkMethod(String, Boolean) : void** **+MngCallStatic(ArrayList<String>, ArrayList<String>) : ArrayList<String>**

ภาพที่ 3-37 คลาส ScanSourcedata

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

TestCaseGen
+GenWHCondition : ArrayList<String> +Gentestdata : ArrayList<String> +GentestVar : ArrayList<String>
+ValidateTestdata (ArrayList<String>,ArrayList<String> , ArrayList<String>) : void +GenTestdataWOCondition (ArrayList<String> ,String) : String +GenTestdataInCondition(String,ArrayList<String> ,String) : String +WriteTestCase(String , String, String, String, String, String, ArrayList<String>, ArrayList<String>,String) : void +TestCaseExistCK(String ,String) : boolean +TestCaseSummary(String ,String) : void

ภาพที่ 3-38 คลาส TestCaseGen

UIInterfaceApp
+ClassandMethodName : ArrayList<String> +AllClassNameCopy : ArrayList<String> +StaticVariableCT : ArrayList<String> +UncoverClass : ArrayList<String> +UncoverMethod : ArrayList<String> +UncoverClassREA : ArrayList<String> +UncoverMethodREA : ArrayList<String> +ClassCoverage : float +MethodCoverage : float +ClassCoverageREA : float +MethodCoverageREA : float
+UIinterface(File[] , String) : void +delete(File) : void +CoverageAnalysisUIApp (String) : void +CoverageReportGen(String) : void +CoverageAnalysisOpProjectUIApp (String) : void

ภาพที่ 3-39 คลาส UIInterfaceApp

UIInterfaceGenTestCase
+ClassName : ArrayList<String> +MethodName : ArrayList<String> +UnCoverMethod : ArrayList<String> +UnCoverMethodSet : ArrayList<String> +UnCoverLocationSet : ArrayList<String>
+InterfaceGenTestCase (String, String, String ,String) : void

ภาพที่ 3-40 คลาส UIInterfaceGenTestCase

XMLParsing
+XMLParingInterface (String , String , String) : void

ภาพที่ 3-41 คลาส XMLParsing

3.2.4 แผนภาพลำดับ

แผนภาพลำดับ เป็นแผนภาพที่ใช้แสดงปฏิสัมพันธ์ระหว่างวัตถุต่างๆในระบบ แผนภาพลำดับของเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการมี 6 แผนภาพ ดังนี้

1) แผนภาพลำดับของการแทรกซอร์สโค้ด ดังภาพที่ 3-42 การทำงานเริ่มจากผู้ทดสอบเลือกซอร์สโค้ดที่ต้องการทดสอบเข้าสู่ระบบ ระบบจะเขียนโค้ดเพื่อใช้ในการบันทึกค่าของตัวแปรที่ใช้นับการเรียกใช้เมท็อดลงสู่ไฟล์ในรูปแบบ .csv ด้วยเมท็อด writeFilesummay หลังจากนั้นระบบจะเก็บข้อมูลของซอร์สโค้ดว่ามีกี่คลาส แต่ละคลาสมีเมท็อดอะไรบ้างด้วยเมท็อดหมายเลข 4. KeepSourceData จนกระทั่งถึงเมท็อดหมายเลข 8. ChkEvntHdl หลังจากนั้น ระบบจะสรุปว่าต้องสร้างตัวแปรขึ้นมากี่ตัว อะไรบ้างด้วยเมท็อดหมายเลข 9. MngCallStatic และหมายเลข 10. ReadInsedSource หลังจากนั้นจึงทำการแทรกซอร์สโค้ดด้วยเมท็อดหมายเลข 11. WriteImportLib ถึงเมท็อดหมาย 18. ExitInstrumented แล้วรายงานผลการแทรกซอร์สโค้ดให้ผู้ทดสอบทราบ

2) แผนภาพลำดับของการวิเคราะห์ความครอบคลุม ดังแสดงในภาพที่ 3-43 เริ่มจากผู้ระบุโฟล์เดอร์ที่ใช้เก็บซอร์สโค้ดที่แทรกแล้ว ซึ่งระบบได้แทรกซอร์สโค้ดให้เขียนไฟล์สรุปการเรียกใช้เมท็อดไว้ที่โฟล์เดอร์นั้น ระบบจะเปิดไฟล์สรุปและคำนวณความครอบคลุมคลาส ความครอบคลุมเมท็อดรวมทั้งระบุคลาสและเมท็อดที่ไม่ครอบคลุมด้วยเมท็อดหมายเลข 4. MethodCoverageAna 5. UncoverClassExtract, และเมท็อดหมายเลข 6. ClassCoverageAna หลังจากนั้นจะส่งผลให้ผู้ทดสอบทราบและให้ผู้ทดสอบระบุโฟล์เดอร์ที่ต้องการเก็บรายงานการวิเคราะห์ความครอบคลุม หลังจากนั้นระบบจะบันทึกรายงานความครอบคลุมในรูปแบบ html ด้วยเมท็อดหมายเลข 10. CoverageReportGen และเมท็อดหมายเลข 11. WriteCoverageReport และบันทึกรายการเมท็อดที่ไม่ถูกเรียกใช้ด้วยเมท็อด 12. WriteUnCoverMethod

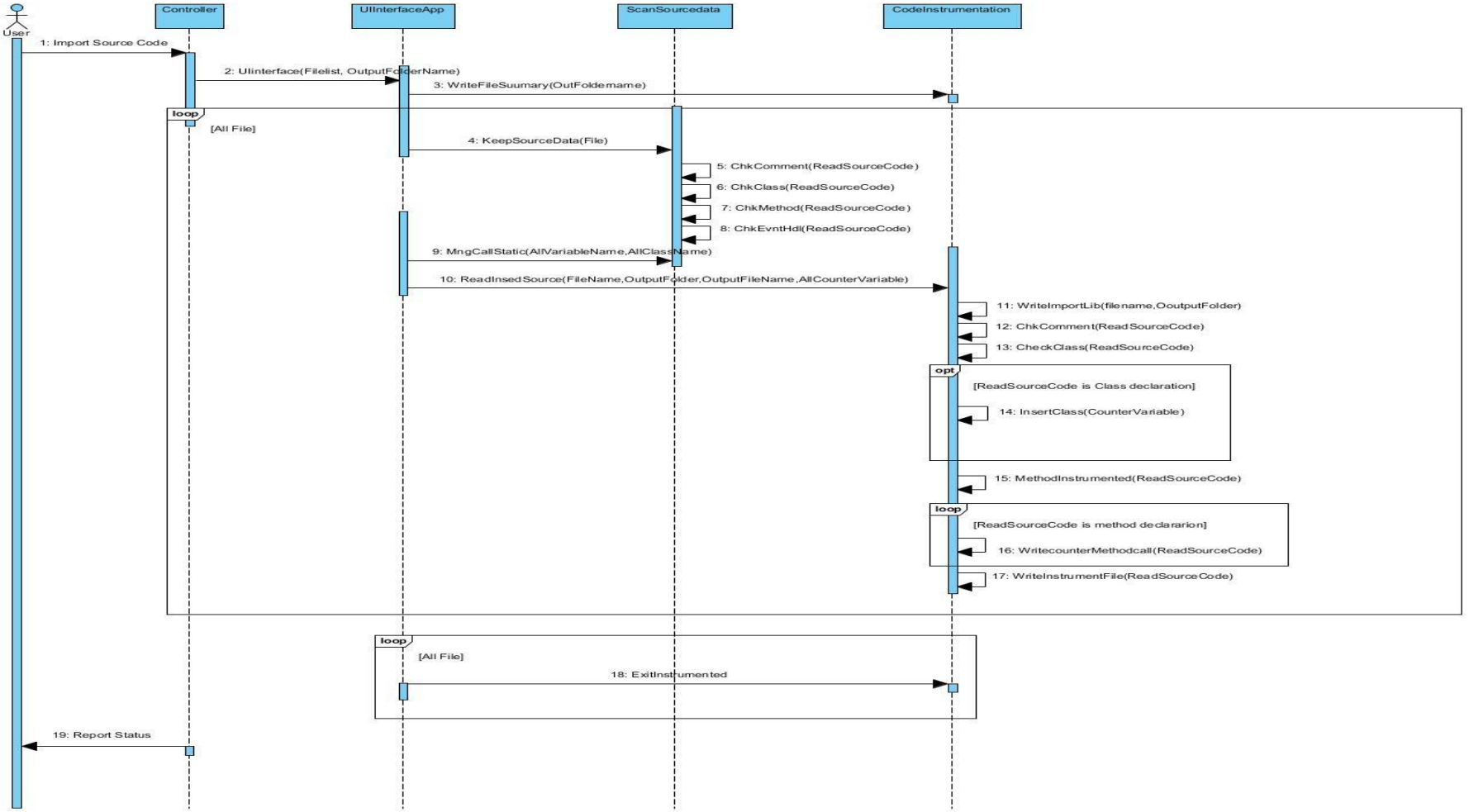
3) แผนภาพลำดับของการสกัดข้อมูล ดังแสดงใน ภาพที่ 3-44 เริ่มจากผู้ทดสอบระบุโฟล์เดอร์ที่ใช้เก็บแผนภาพลำดับ หลังจากนั้นระบบจะสกัดข้อมูลที่จำเป็นในการสร้างกราฟการทำงาน ดังนี้ ชื่อเมท็อดรวมทั้งลำดับการเรียกใช้ ชื่อคลาส และเงื่อนไขการเกิดเมท็อด ซึ่งการสกัดข้อมูลนี้จะกระทำด้วยเมท็อดหมายเลข 4. ParsingSeqDiagramName จนถึงเมท็อดหมายเลข 16. ParserCombineFragment และรวบรวมข้อมูลสรุปลงในไฟล์เพื่อนำไปใช้ในการสร้างกราฟการทำงานด้วยเมท็อดหมายเลข 17. SaveCombineFrament 18. CollectAllCF และ เมท็อดหมายเลข 19. SaveMethodCalledList

4) แผนภาพลำดับของการติดต่อกับฐานข้อมูล ดังแสดงในภาพที่ 3-45 หลังจากระบบได้สกัดข้อมูลที่ต้องการจากแผนภาพลำดับที่ผู้ใช้เลือกแล้ว (ดังแสดงในแผนภาพลำดับของการสกัดข้อมูล) ระบบจะนำข้อมูลของแผนภาพลำดับ โฟลเดอร์ที่ใช้เก็บไฟล์ข้อมูลที่สกัดได้จากแผนภาพลำดับ รวมมทั้งโฟลเดอร์ที่ใช้เก็บคำอธิบายยูสเคลที่เกี่ยวข้อง เข้าสู่ฐานข้อมูล โดย เมท็อด หมายเลข 2. SDNameExtract และ หมายเลข 3. ChooseUCDFile เป็นการเลือกคำอธิบายยูสเคสให้ตรงกับแผนภาพลำดับ แล้วนำข้อมูลของแผนภาพลำดับ โฟลเดอร์ที่ใช้เก็บไฟล์ข้อมูลที่สกัดได้จากแผนภาพลำดับ โฟลเดอร์ที่ใช้เก็บคำอธิบายยูสเคลที่เกี่ยวข้อง ใส่ลงไปในฐานข้อมูลด้วยเมท็อด 4. insertSDLocation 5. insertUCDLocation และ 6. Inserttable หลังจากนั้น ระบบจะนำรายการเมท็อดที่ไม่ถูกเรียกใช้มาหาแผนภาพลำดับและคำอธิบายยูสเคสที่เกี่ยวข้องด้วยเมท็อดหมายเลข 8. Selectlocation และ 9. selectUCDlocation

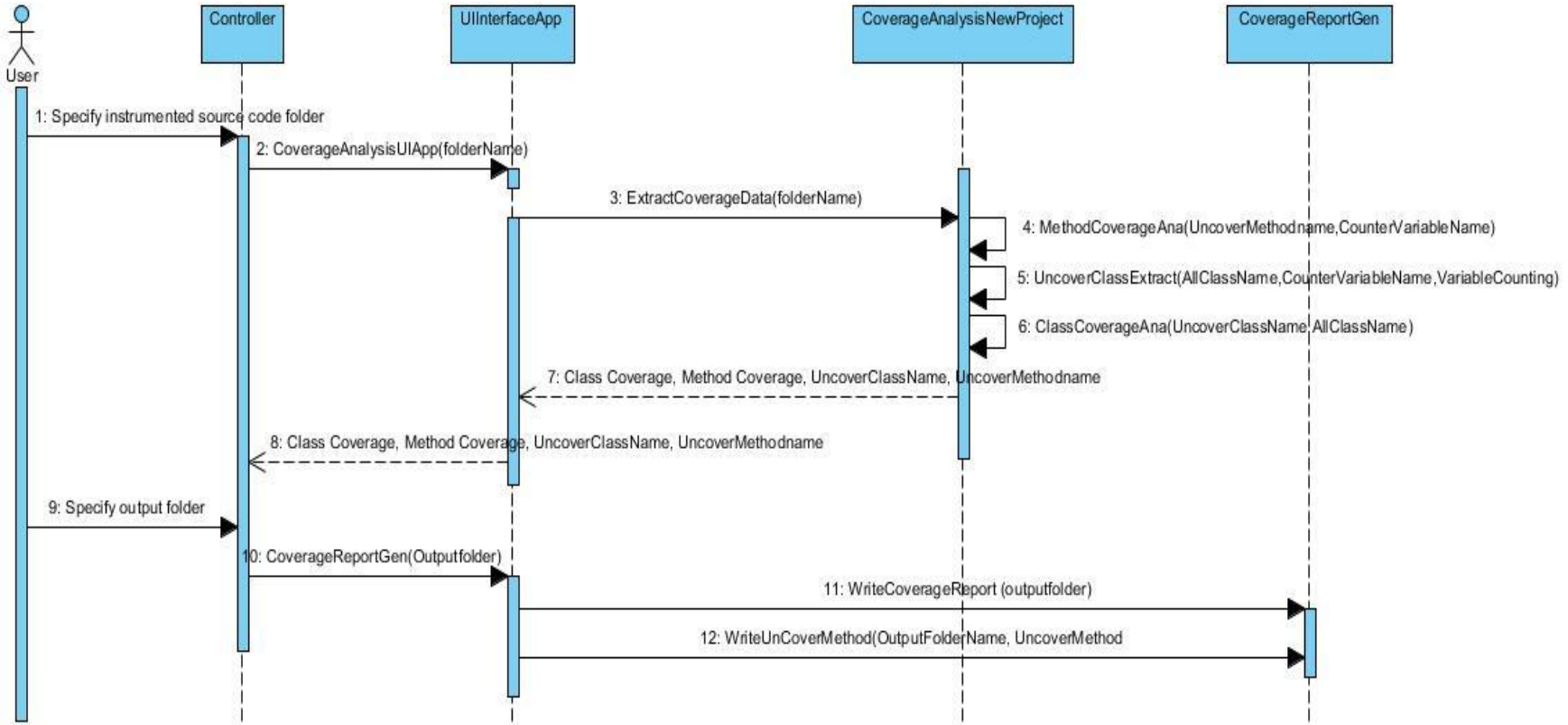
5) แผนภาพลำดับของการสร้างกราฟการทำงาน ดังแสดงในภาพที่ 3-46 เมื่อระบบได้รับไฟล์ข้อมูลที่สกัดได้ของแผนภาพลำดับที่เกี่ยวข้องกับเมท็อดที่ไม่ถูกเรียกใช้แล้ว (ดังแสดงในแผนภาพลำดับของการติดต่อกับฐานข้อมูล) ระบบจะนำข้อมูลในไฟล์ไฟล์ข้อมูลที่สกัดได้ มาสร้างกราฟการทำงานด้วยเมท็อดหมายเลข 3. ConnectedNode จนถึงเมท็อดหมาย 14. MSSCreateSubGraph

6) แผนภาพลำดับของการสร้างกรณีทดสอบ ดังแสดงในภาพที่ 3-47 เมื่อระบบสร้างกราฟการทำงานเสร็จแล้ว ระบบจะนำชื่อเมท็อดที่ไม่ถูกเรียกใช้มาหาเส้นทางและเงื่อนไขการเรียกใช้เมท็อด ด้วยเมท็อดหมายเลข 1. Conditionextract หลังจากนั้นจะนำเงื่อนไขที่ได้ไปหาข้อมูลที่จำเป็นต้องใช้ในการสร้างกรณีทดสอบจากคำอธิบายยูสเคสด้วยเมท็อดหมายเลข 2. UCDCSVExtract จนถึงเมท็อดหมายเลข 6. GetScenarioFromUCD แล้วระบบจะตรวจสอบว่าเงื่อนไขการเรียกใช้เมท็อดนี้ ถูกสร้างกรณีทดสอบแล้วหรือไม่ด้วยเมท็อดหมายเลข 7. TestCaseExistCK ถ้าพบว่าเงื่อนไขนั้นยังไม่ถูกสร้างกรณีทดสอบก็จะสร้างกรณีทดสอบด้วยเมท็อดหมายเลข 8. WriteTestCase

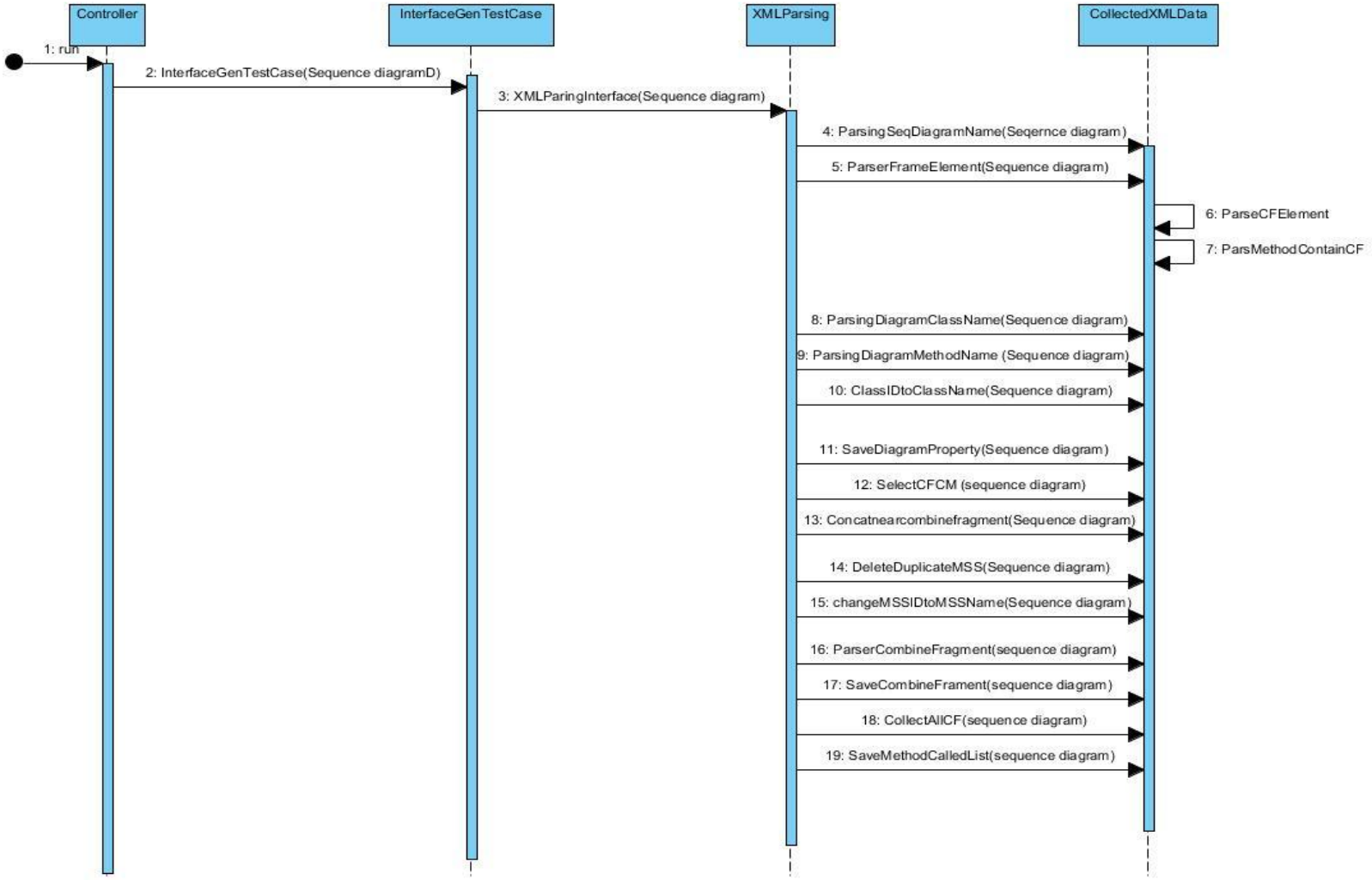
จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



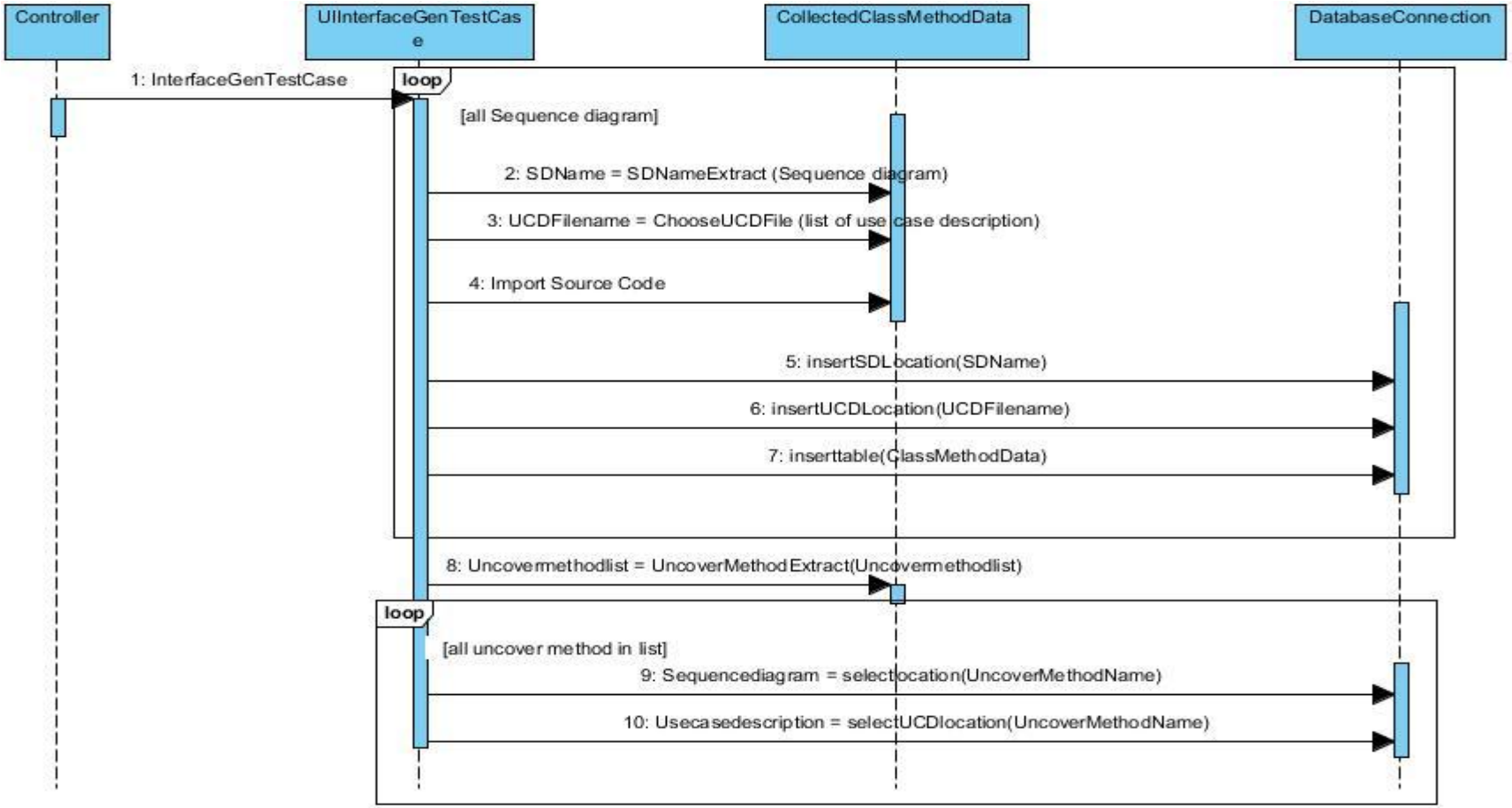
ภาพที่ 3-42 แผนภาพลำดับของการแทรกซอร์สโค้ด



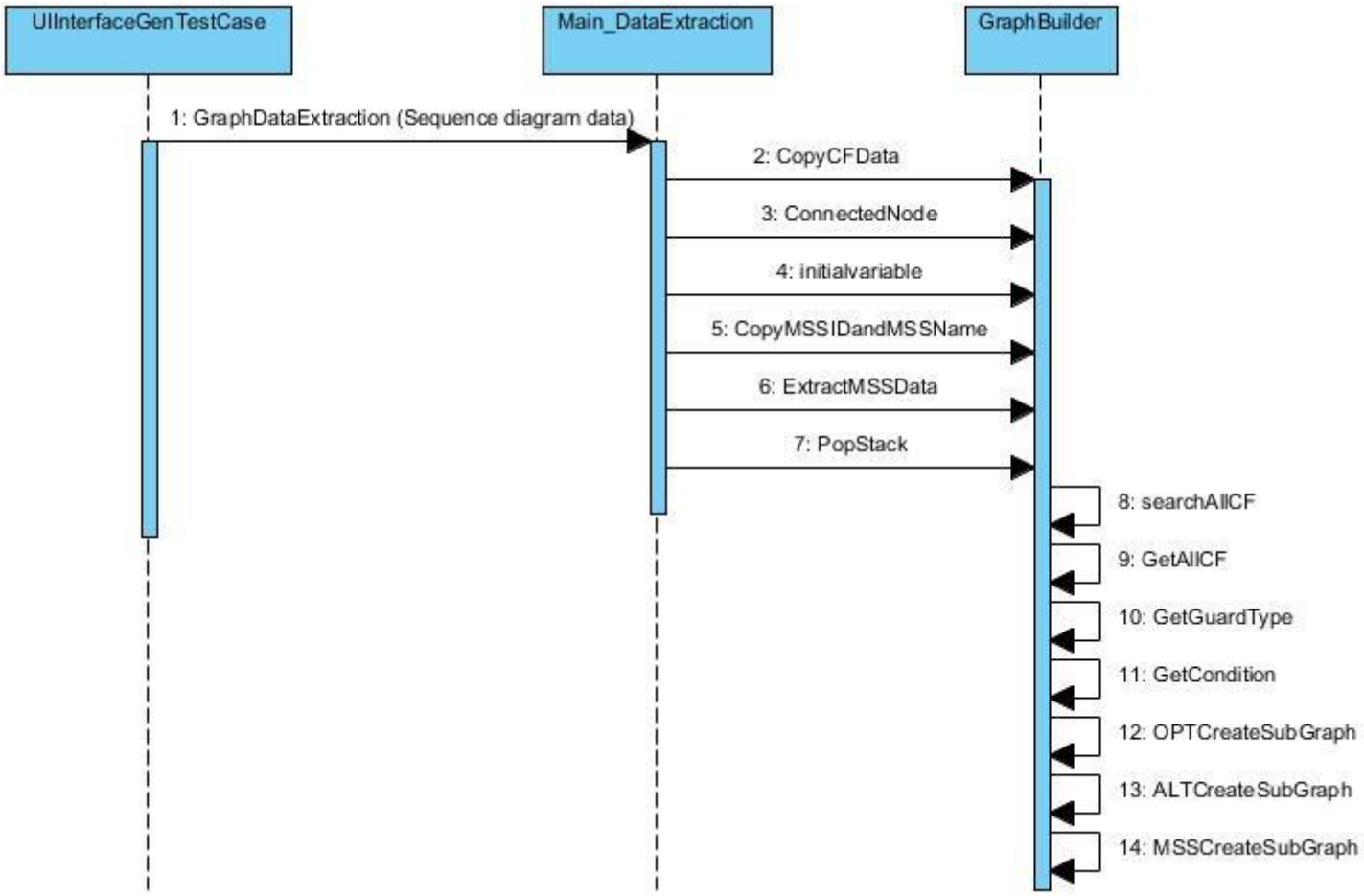
ภาพที่ 3-43 แผนภาพลำดับของการวิเคราะห์ความครอบคลุม



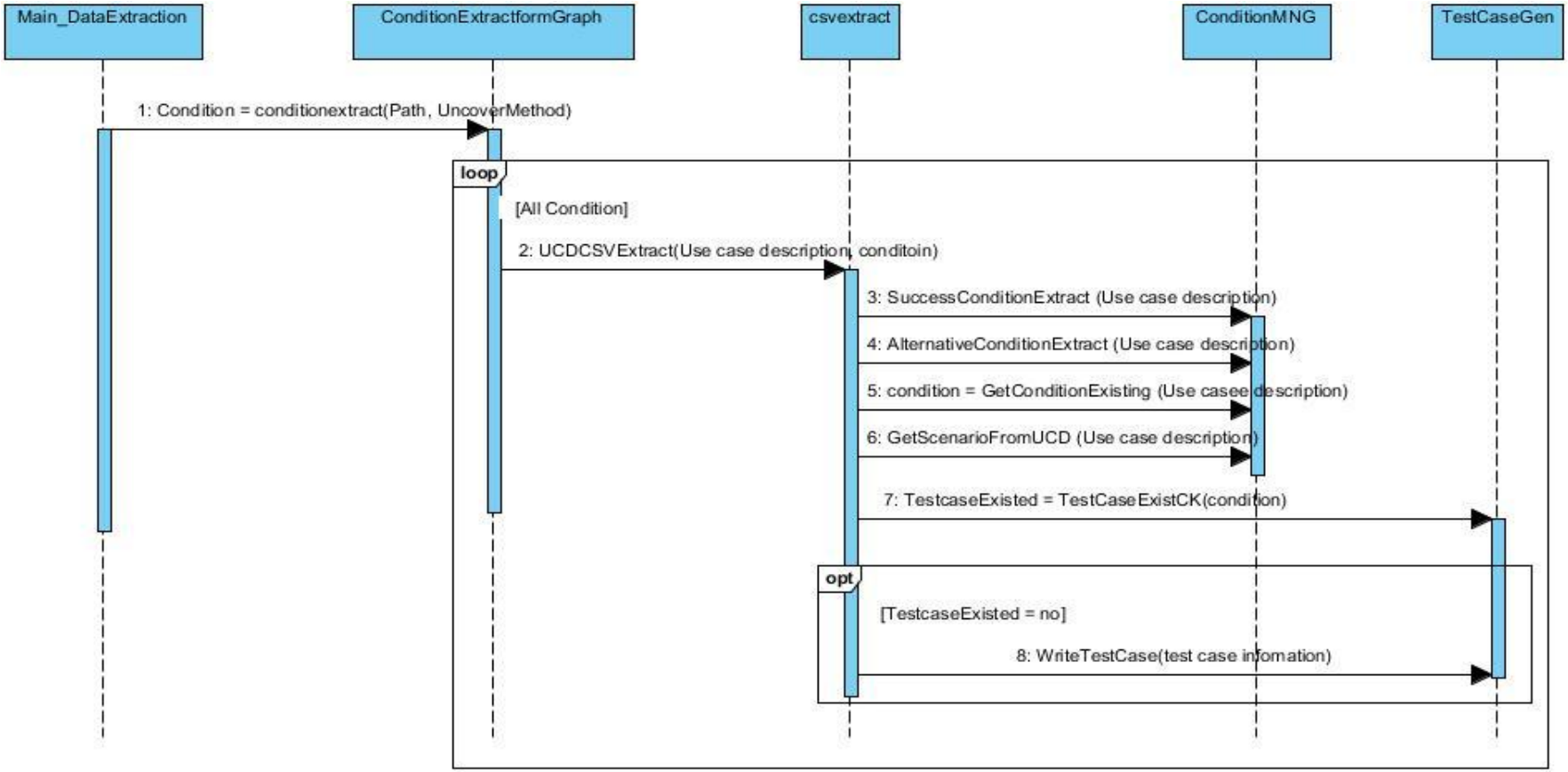
ภาพที่ 3-44 แผนภาพลำดับของการสกัดข้อมูล



ภาพที่ 3-45 แผนภาพลำดับของการติดต่อกับฐานข้อมูล



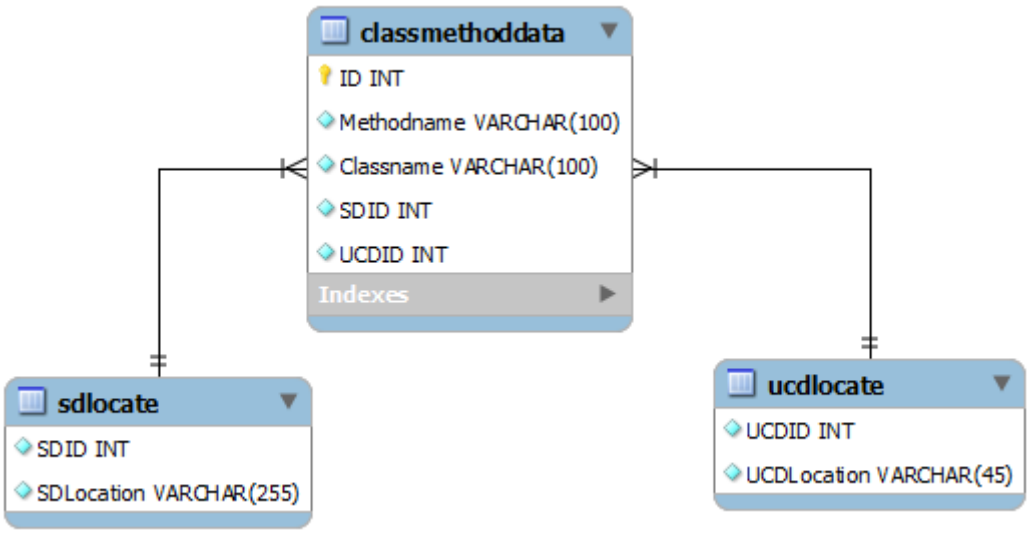
ภาพที่ 3-46 แผนภาพลำดับของการสร้างกราฟการทำงาน



ภาพที่ 3-47 แผนภาพลำดับของการสร้างกรณีทดสอบ

3.2.5 โครงสร้างฐานข้อมูล

โครงสร้างฐานข้อมูลของเครื่องมือ สามารถแสดงได้ดังแผนภาพอีอาร์ภาพที่ 3-48 โดยรายละเอียดของโครงสร้างฐานข้อมูลเป็นดังนี้



ภาพที่ 3-48 โครงสร้างฐานข้อมูลของเครื่องมือ

1) ตาราง classmethoddata เป็นตารางที่ใช้เก็บชื่อคลาสและเมท็อดในแต่ละแผนภาพลำดับและคำอธิบายยูสเคสที่เกี่ยวข้อง

2) ตาราง sdlocate เป็นตารางที่ใช้เก็บชื่อโฟลเดอร์ที่เก็บแผนภาพลำดับแต่ละแผนภาพ

3) ตาราง ucdlocate เป็นตารางที่ใช้เก็บชื่อโฟลเดอร์ที่เก็บคำอธิบายยูสเคส

พจนานุกรมข้อมูลของแต่ละตารางสามารถดูได้ที่ภาคผนวก ก

บทที่ 4
การพัฒนาเครื่องมือ

ในบทการพัฒนาเครื่องมือนี้จะกล่าวถึงสภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ รวมทั้งโครงสร้างส่วนต่อประสานกับผู้ทดสอบของเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ ซึ่งมีรายละเอียดดังต่อไปนี้

4.1 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ

สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือมีรายละเอียดดังต่อไปนี้

4.1.1 ฮาร์ดแวร์ (Hardware)

1) เครื่องคอมพิวเตอร์แบบโน้ตบุ๊ก (Notebook) หน่วยประมวลผลอินเทลคอร์ไอไฟว์ 2.50 กิกะเฮิร์ท (Intel Core i5 GHz)

2) หน่วยความจำสำรอง 8.0 กิกะไบต์ (8.0 GB)

3) ฮาร์ดดิสก์ (Harddisk) 600 กิกะไบต์ (600 GB)

4.1.2 ซอฟต์แวร์ (Software)

1) ระบบปฏิบัติการ (Operating System) ไมโครซอฟท์วินโดวส์เอท (Microsoft Windows 8)

2) อีคลิปไอดีอี เวอร์ชัน เคปเลอร์ (EclipseIDE Kepler)

3) จาวาเจดีเค เวอร์ชัน 7.0 (Java JDK 7.0)

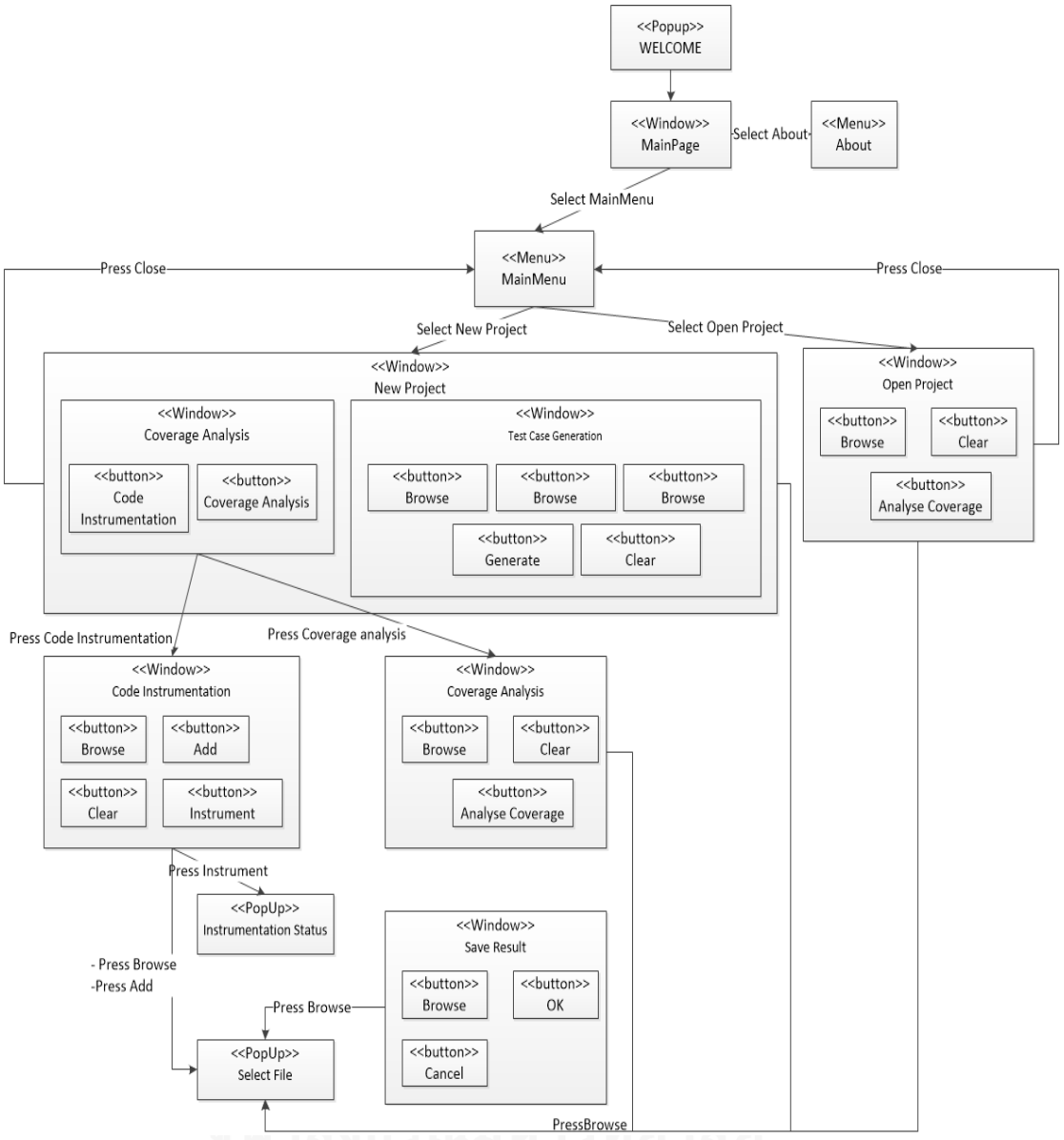
4) วินโดวส์บิวเดอร์โปร เวอร์ชัน 4.2 (Window Builder Pro 4.2)

5) ฐานข้อมูลมายเอสคิวแอล เวอร์ชัน 5.0.51b (MySQL 5.0.51b)

6) เจกราฟทีไลบาร์รี เวอร์ชัน 0.9 (jgrapht 0.9)

4.2 โครงสร้างส่วนต่อประสานกับผู้ใช้ของเครื่องมือ

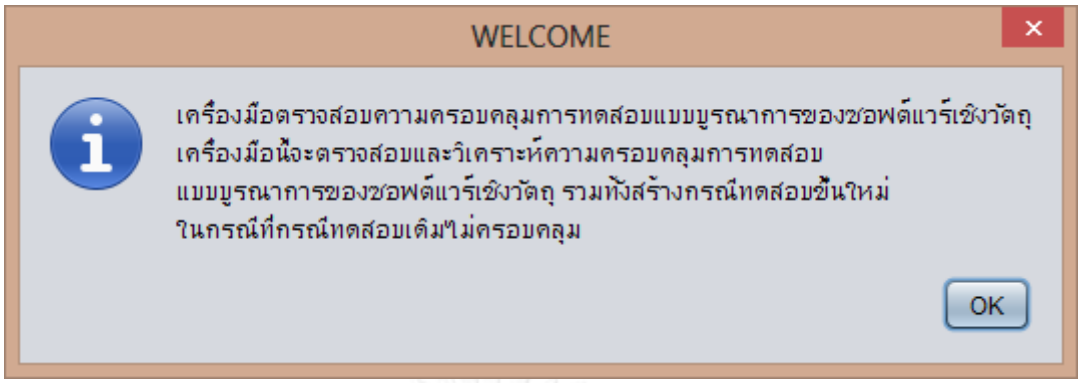
โครงสร้างส่วนต่อประสานของเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ จะถูกอธิบายด้วยแผนภาพ Window Navigation ซึ่งใช้อธิบายถึงความสัมพันธ์ต่างๆของส่วนต่อประสานทั้งหมดในโปรแกรมแบ่งตามหน้าที่ต่างๆ ดังภาพ 4.1



ภาพที่ 4-1 แผนภาพ Window Navigation

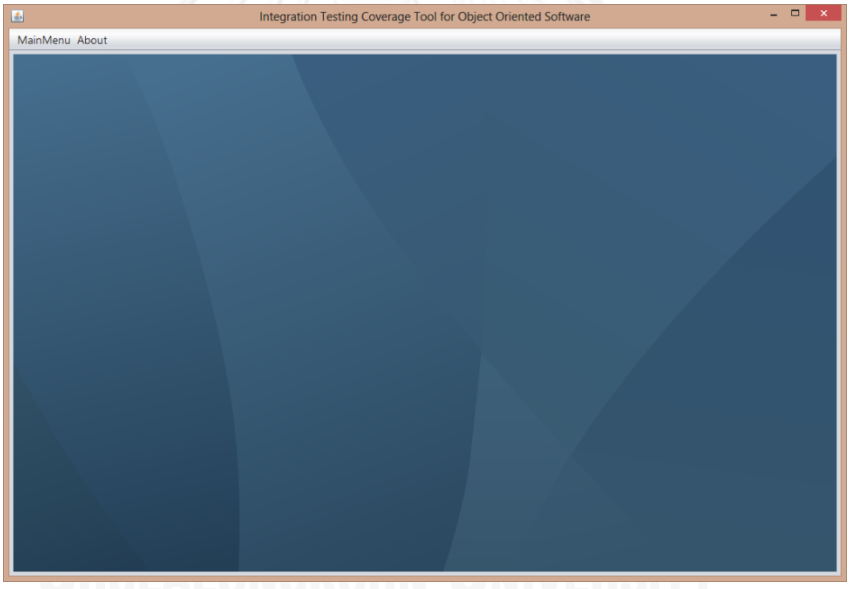
จากภาพที่ 4-1 แผนภาพ Window Navigation แผนภาพ Window Navigation นี้แสดงส่วนประกอบส่วนต่อประสานของเครื่องมือ ซึ่งจะประกอบไปด้วย หน้าต่าง เมนู แต่ละส่วนประกอบมีรายละเอียดดังต่อไปนี้

1) WELCOME เป็นหน้าต่างแรกที่แสดงขึ้นมาเมื่อเปิดเครื่องมือ แสดงรายละเอียดของเครื่องมือ ดังภาพที่ 4-2



ภาพที่ 4-2 WELCOME

2) หน้าต่าง MainPage เป็นหน้าต่างหลักของเครื่องมือ ซึ่งประกอบไปด้วย 2 เมนู ได้แก่ เมนู MainMenu และเมนู หน้าต่าง MainPage สามารถแสดงได้ดัง



ภาพที่ 4-3 หน้าต่างหลักของเครื่องมือ

2.1) เมนู MainMenu เป็นเมนูหลักที่ประกอบไปด้วยเมนูย่อย 3 เมนูคือ

2.1.1) เมนู New Project เป็นเมนูที่ใช้สร้างโปรเจคใหม่

2.1.2) เมนู Open Project เป็นเมนูที่ใช้เปิดโปรเจคเดิมที่เคยสร้างไว้

2.1.3) เมนู Exit เป็นเมนูเพื่อออกจากเครื่องมือนี้

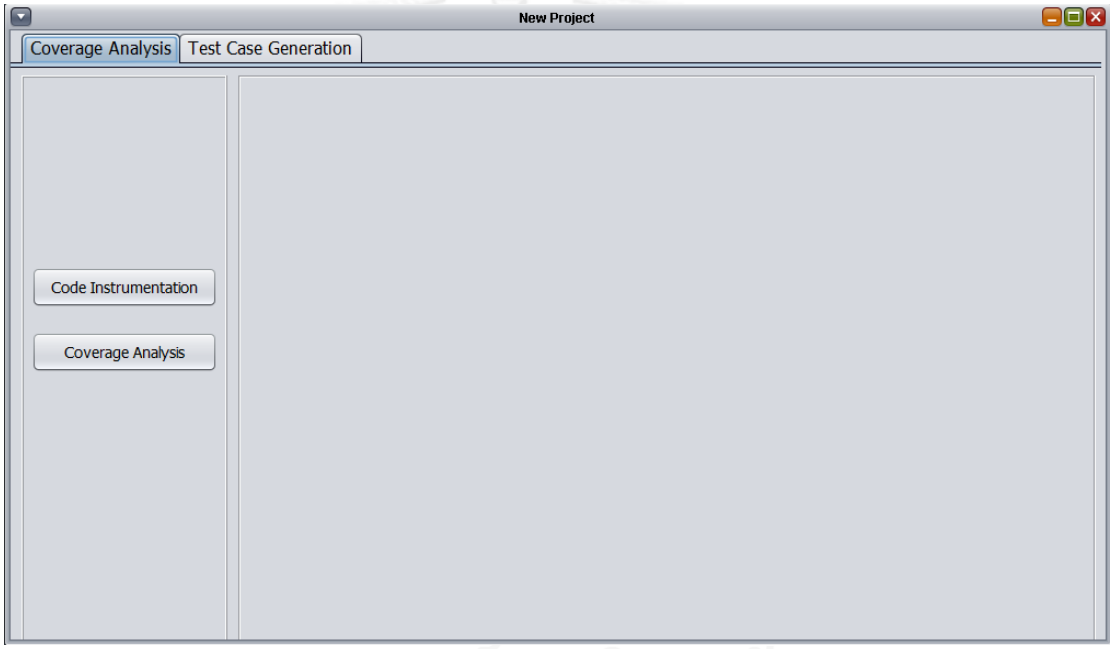
3) เมนู About เป็นเมนูที่ใช้บอกรายละเอียดเกี่ยวกับเครื่องมือนี้

4) หน้าต่าง New Project หน้าต่างนี้เป็นหน้าต่างสร้างโปรเจคใหม่ โดยหน้าต่างนี้ประกอบไปด้วย แถบหน้าต่าง 2 แถบ ได้แก่

4.1) แถบหน้าต่าง Coverage Analysis เป็นส่วนที่ใช้ในการวิเคราะห์ความครอบคลุมของการทดสอบ สามารถแสดงได้ดังภาพที่ 4-4 แถบหน้าต่างนี้ประกอบไปด้วย 2 ปุ่ม คือ

4.1.1) Code Instrumentation เป็นปุ่มที่ใช้เปิดหน้าต่าง Code Instrumentation เพื่อใช้ในการแทรกซอร์สโค้ด

4.1.2) Coverage Analysis เป็นปุ่มที่ใช้เปิดหน้าต่าง Coverage Analysis เพื่อใช้ในการวิเคราะห์ความครอบคลุมการทดสอบ



ภาพที่ 4-4 แถบหน้าต่าง Coverage Analysis

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.2) แถบหน้าต่าง Test Case Generation เป็นส่วนที่ใช้ในการสร้างกรณีทดสอบเพิ่มเติมในกรณีที่กรณีทดสอบเดิมไม่ครอบคลุม ดังแสดงในภาพที่ 4-5 แถบหน้าต่างนี้ ประกอบด้วยปุ่ม 5 ปุ่มดังนี้

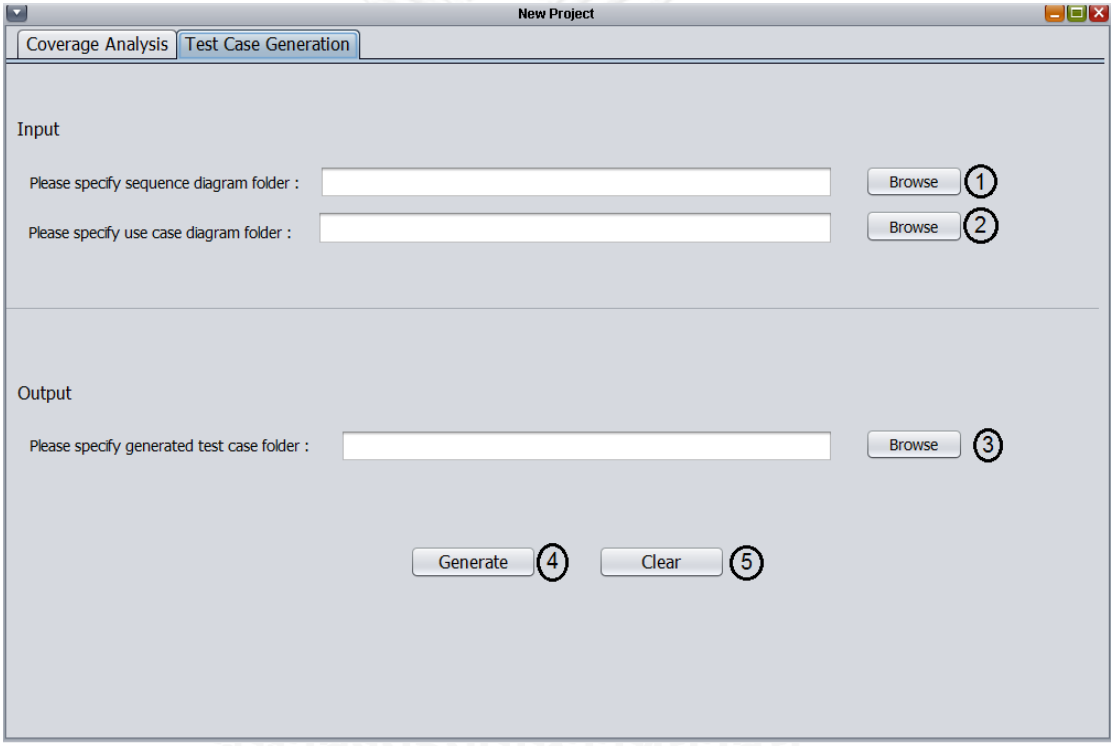
4.2.1) ปุ่ม Browse หมายเลข 1 เมื่อกดปุ่มนี้จะแสดงหน้าต่างป๊อบอัพเพื่อให้ผู้ใช้เลือกโฟลเดอร์ที่ใช้เก็บแผนภาพลำดับ

4.2.2) ปุ่ม Browse หมายเลข 2 เมื่อกดปุ่มนี้จะแสดงหน้าต่างป๊อบอัพเพื่อให้ผู้ใช้เลือกโฟลเดอร์ที่ใช้เก็บคำอธิบายยูสเคส

4.2.3) ปุ่ม Browse หมายเลข 3 เมื่อกดปุ่มนี้จะแสดงหน้าต่างป๊อบอัพเพื่อให้ผู้ใช้เลือกโฟลเดอร์ที่ต้องการให้เก็บกรณีทดสอบที่เครื่องมือสร้าง

4.2.4) ปุ่ม Generate หมายเลข 4 เมื่อกดปุ่มนี้ เครื่องมือจะสร้างกรณีทดสอบจากแผนภาพลำดับและคำอธิบายยูสเคสที่เก็บไว้ในโฟลเดอร์ที่กำหนด และบันทึกกรณีทดสอบลงในโฟลเดอร์ที่กำหนด

4.2.5) ปุ่ม Clear หมายเลข 5 เมื่อกดปุ่มนี้ จะเป็นการยกเลิกตำแหน่งโฟลเดอร์ที่เก็บแผนภาพลำดับ คำอธิบายยูสเคส และโฟลเดอร์ที่ต้องการเก็บกรณีทดสอบที่ระบุไว้



ภาพที่ 4-5 แถบหน้าต่าง Test Case Generation

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

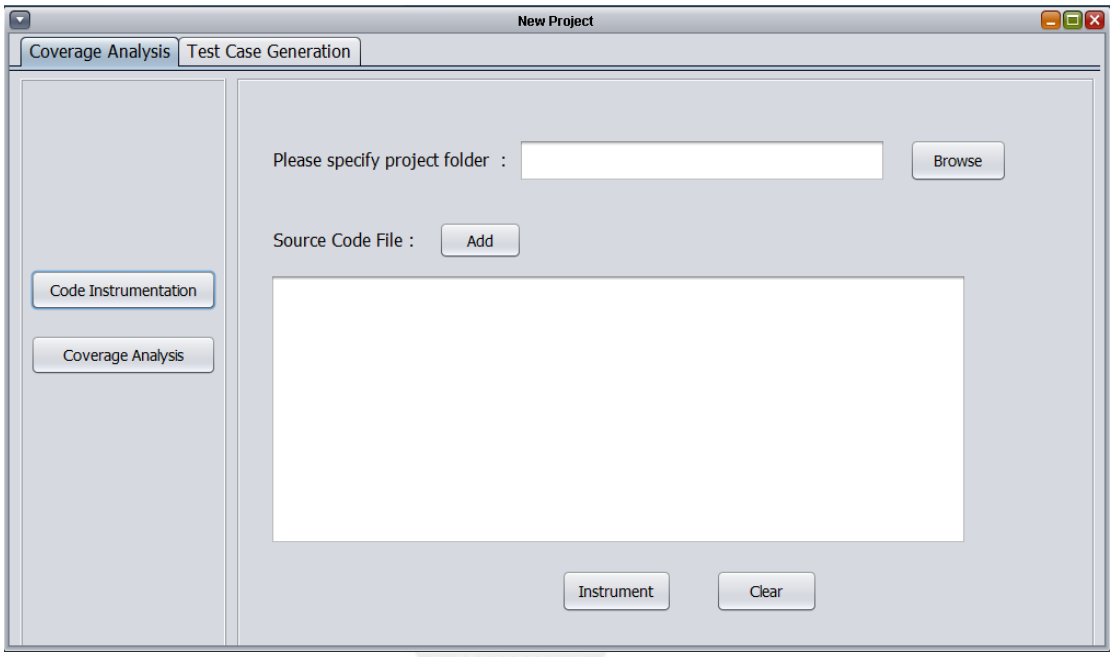
5) หน้าต่าง Code Instrumentation เป็นหน้าต่างที่ให้ผู้ใช้นำซอร์สโค้ดที่ต้องการทดสอบเข้าสู่เครื่องมือเพื่อนำมาแทรกซอร์สโค้ด โดยหน้าต่าง Code Instrumentation สามารถแสดงได้ดังภาพที่ 4-6 หน้าต่างนี้ประกอบไปด้วย 4 ปุ่มดังนี้

5.1) ปุ่ม Browse เมื่อกดปุ่มนี้แล้ว จะแสดงหน้าต่างป๊อบอัพเพื่อให้ผู้ใช้เลือกโฟลเดอร์ที่ต้องการให้เก็บซอร์สโค้ดที่ถูกแทรก

5.2) ปุ่ม Add เมื่อปุ่มนี้แล้ว จะแสดงหน้าต่างป๊อบอัพเพื่อให้ผู้ใช้เลือกไฟล์ซอร์สโค้ดที่ต้องการทดสอบ

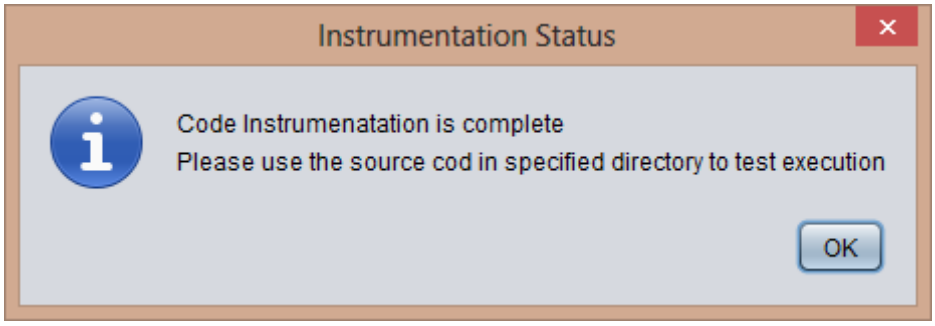
5.3) ปุ่ม Instrument เมื่อกดปุ่มนี้แล้ว เครื่องมือจะดำเนินการแทรกซอร์สโค้ดและบันทึกลงในโฟลเดอร์ที่ระบุไว้

5.4) ปุ่ม clear เมื่อกดปุ่มนี้ จะเป็นการยกเลิกตำแหน่งโฟลเดอร์ที่ต้องการเก็บซอร์สโค้ดที่ถูกแทรกรวมทั้งยกเลิกการนำซอร์สโค้ดที่ระบุเข้าสู่ระบบ



ภาพที่ 4-6 หน้าต่าง Code Instrumentation

6) Instrumentation Status จะแสดงเมื่อผู้ใช้กดปุ่ม Instrument ในหน้าต่าง Code Instrumentation และระบบแทรกซอร์สโค้ดเสร็จแล้ว หน้าต่างInstrumentation Status สามารถแสดงได้ดังภาพที่ 4-7



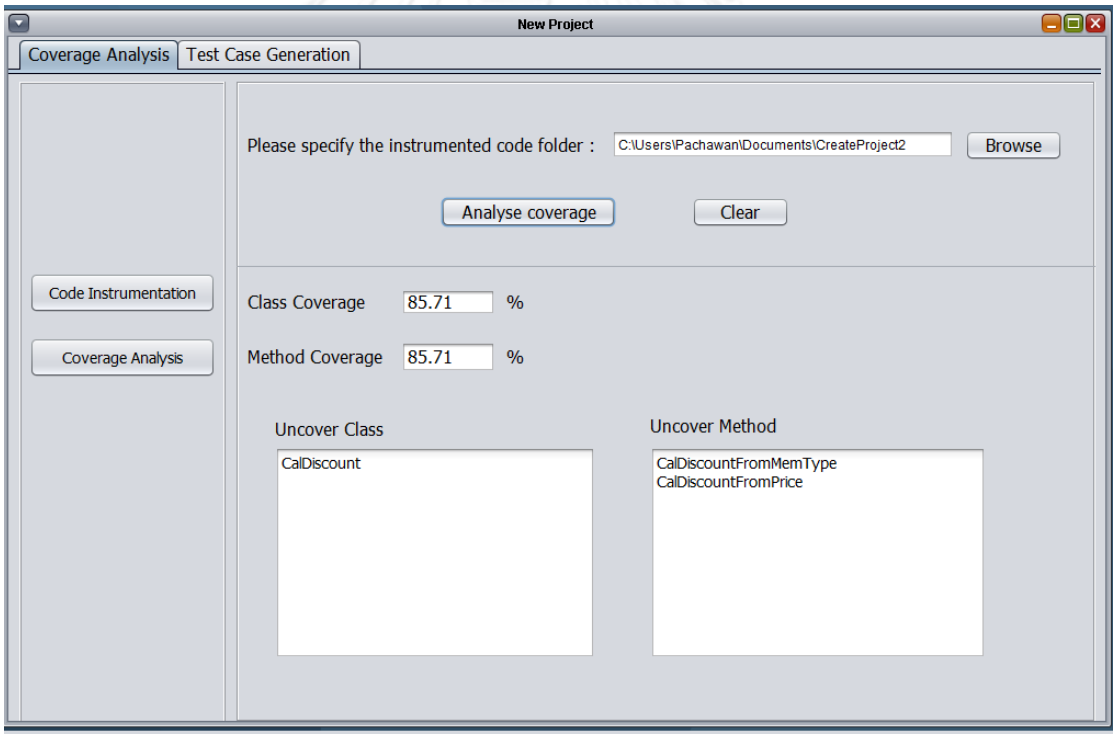
ภาพที่ 4-7 Instrumentation Status

7) หน้าต่าง Coverage Analysis เป็นหน้าต่างที่แสดงผลการวิเคราะห์ความครอบคลุมของการทดสอบแบบบูรณาการโดยให้ผู้ใช้ระบุตำแหน่งที่เก็บซอร์สโค้ดที่แทรกซอร์สโค้ดแล้ว และระบบจะแสดงผลการวิเคราะห์รวมทั้งแสดงคลาสและเมท็อดที่ไม่ครอบคลุม ดังแสดงในภาพที่ 4-8 หน้าต่างนี้ประกอบด้วย 3 ปุ่ม ดังนี้

7.1) ปุ่ม Browse เมื่อกดปุ่มนี้แล้ว จะแสดงหน้าต่างป๊อบอัพเพื่อให้ผู้ใช้เลือกโฟลเดอร์ที่เก็บซอร์สโค้ดที่ถูกแทรก

7.2) ปุ่ม Analyse Coverage เมื่อกดปุ่มนี้แล้ว เครื่องมือจะแสดงผลการวิเคราะห์ความครอบคลุม แบ่งเป็นจำนวนคลาสที่ครอบคลุมคิดเป็นร้อยละ โดยจะแสดงในช่อง Class coverage และจำนวนเมท็อดที่ครอบคลุมคิดเป็นร้อยละ โดยจะแสดงในช่อง Method coverage รวมทั้งชื่อคลาสและชื่อเมท็อดที่ไม่ครอบคลุม

7.3) ปุ่ม Clear เมื่อกดปุ่มนี้แล้ว ระบบจะยกเลิกการใช้โฟลเดอร์ที่เก็บซอร์สโค้ดที่ถูกแทรกนั้น



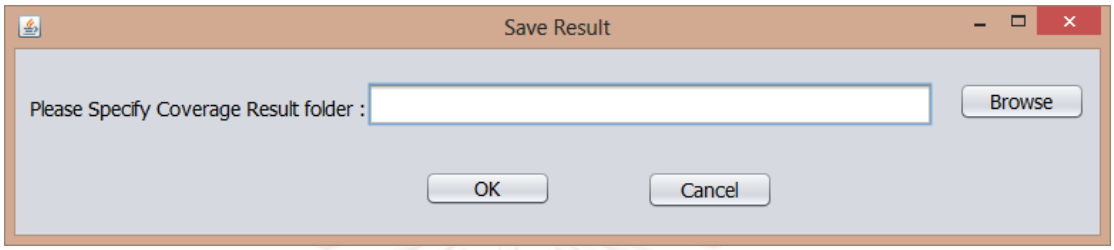
ภาพที่ 4-8 หน้าต่าง Coverage Analysis

8) หน้าต่าง Save Result เป็นหน้าต่างที่ใช้บันทึกผลการวิเคราะห์ความครอบคลุม ดังภาพที่ 4-9 ประกอบไปด้วย 3 ปุ่มดังนี้

8.1) ปุ่ม Browse เมื่อกดปุ่มนี้แล้ว จะแสดงหน้าต่างป๊อบอัพเพื่อให้ผู้ใช้เลือกโฟลเดอร์ที่ต้องการเก็บผลการวิเคราะห์ความครอบคลุม

8.2) ปุ่ม OK สำหรับกดตกลง

8.3) ปุ่ม Cancel สำหรับกดยกเลิก



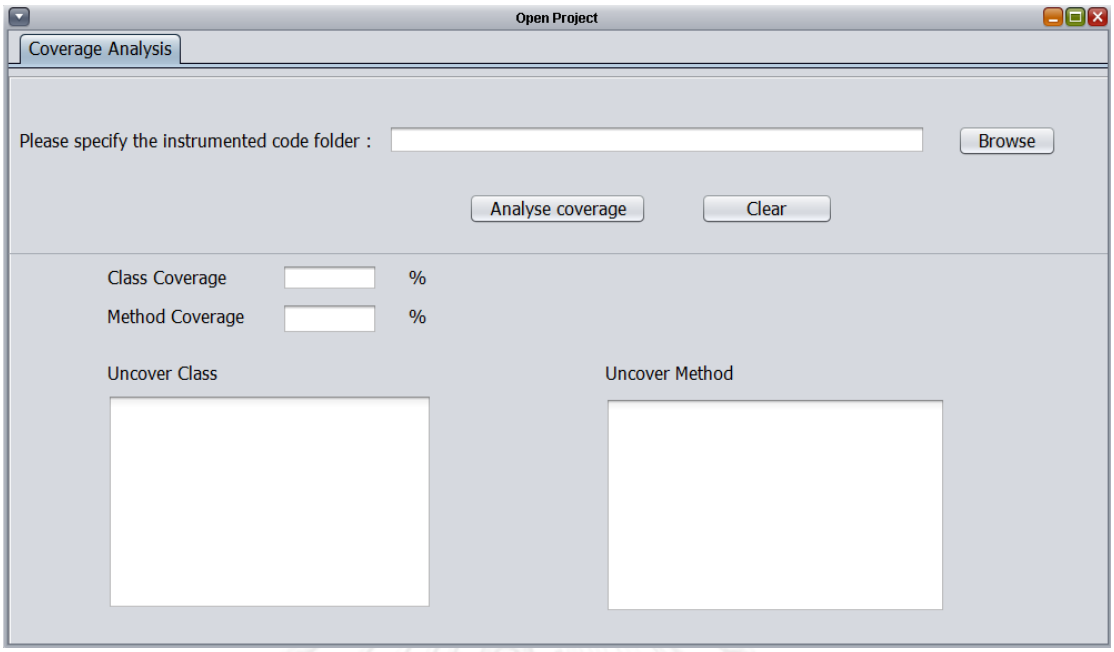
ภาพที่ 4-9 หน้าจอ Save Result

9) หน้าต่าง Open Project หน้าต่างนี้เป็นหน้าต่างที่ใช้วิเคราะห์ความครอบคลุมการทดสอบกรณีที่ได้มีการวิเคราะห์ไปแล้วครั้งหนึ่ง แล้วได้มีการทดสอบเพิ่มเติมจากกรณีทดสอบที่เครื่องมือได้สร้างขึ้น หน้าต่าง Open Project สามารถแสดงได้ดังภาพที่ 4-10 โดยหน้าต่างนี้จะประกอบไปด้วย

9.1) ปุ่ม Browse เมื่อกดปุ่มนี้แล้ว จะแสดงหน้าต่างป๊อบอัพเพื่อให้ผู้ใช้เลือกโฟลเดอร์ที่เก็บซอร์สโค้ดที่ถูกแทรก

9.2) ปุ่ม Analyse Coverage เมื่อกดปุ่มนี้แล้ว เครื่องมือจะแสดงผลการวิเคราะห์ความครอบคลุม แบ่งเป็นจำนวนคลาสที่ครอบคลุมคิดเป็นร้อยละ โดยจะแสดงในช่อง Class coverage และจำนวนเมท็อดที่ครอบคลุมคิดเป็นร้อยละ โดยจะแสดงในช่อง Method coverage รวมทั้งชื่อคลาสและชื่อเมท็อดที่ไม่ครอบคลุม

9.3) ปุ่ม Clear เมื่อกดปุ่มนี้แล้ว ระบบจะยกเลิกการใช้โฟลเดอร์ที่เก็บซอร์สโค้ดที่ถูกแทรกนั้น

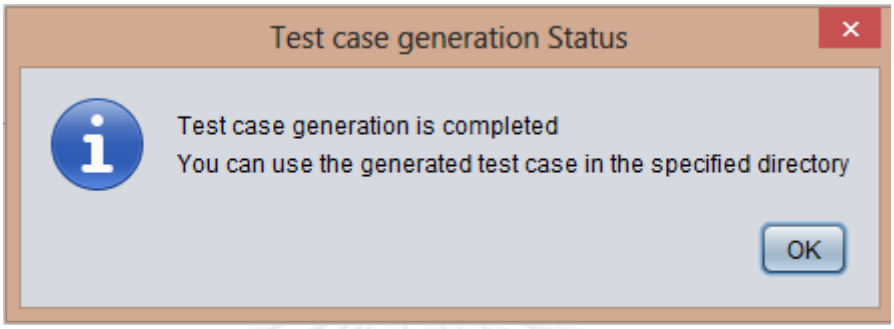


ภาพที่ 4-10 หน้าต่าง Open Project

10) หน้าต่าง About เป็นหน้าต่างที่แสดงรายละเอียดเกี่ยวกับเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ หน้าต่าง About สามารถแสดงได้ดังภาพที่ 4-11

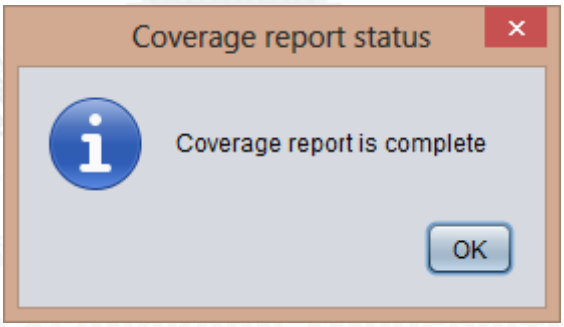
ภาพที่ 4-11 หน้าต่าง About

11) Test case generation Status จะแสดงเมื่อผู้ใช้กดปุ่ม Generate ในแถบหน้าต่าง Test Case Generation และระบบสร้างกรณีทดสอบเสร็จแล้ว หน้าต่าง Test case generation Status สามารถแสดงได้ดังภาพที่ 4-12



ภาพที่ 4-12 หน้าต่าง Test case generation Status

12) Coverage Report Status จะแสดงเมื่อผู้ใช้กดปุ่ม OK ในหน้าต่าง Save Result และระบบบันทึกผลการวิเคราะห์ความครอบคลุมแล้ว หน้าต่าง Coverage Report Status สามารถแสดงได้ดังภาพที่ 4-13



ภาพที่ 4-13 Coverage report status

บทที่ 5
การทดสอบเครื่องมือ

หัวข้อนี้กล่าวถึงการทดสอบเครื่องมือเพื่อตรวจสอบความถูกต้อง โดยจะกล่าวถึงสภาพแวดล้อมที่ใช้ในการทดสอบ รายละเอียดของการทดสอบเครื่องมือ และผลการทดสอบเครื่องมือ โดยมีรายละเอียดดังต่อไปนี้

5.1 สภาพแวดล้อมที่ใช้ทดสอบ

สภาพแวดล้อมที่ใช้ทดสอบเครื่องมือ มีรายละเอียดดังต่อไปนี้

5.1.1 ฮาร์ดแวร์

1) เครื่องคอมพิวเตอร์แบบโน้ตบุ๊ก (Notebook) หน่วยประมวลผลอินเทลคอร์ไอไฟว์ 2.50 กิกะเฮิร์ท (Intel Core i5 GHz)

2) หน่วยความจำสำรอง 8.0 กิกะไบต์ (8.0 GB)

3) ฮาร์ดดิสก์ (Harddisk) 600 กิกะไบต์ (600 GB)

5.1.2 ซอฟต์แวร์

1) ระบบปฏิบัติการ (Operating System) ไมโครซอฟท์วินโดวส์เอท (Microsoft Windows 8)

2) อีคลิปไอดีอี เวอร์ชัน เคปเลอร์ (EclipseIDE Kepler)

3) จาวาเจดีเค เวอร์ชัน 7.0 (Java JDK 7.0)

4) วินโดวส์บิวเดอร์โปร เวอร์ชัน 4.2 (Window Builder Pro 4.2)

5) ฐานข้อมูลมายเอสคิวแอล เวอร์ชัน 5.0.51b (MySQL 5.0.51b)

6) เจกราฟทีไลบาร์รี เวอร์ชัน 0.9 (jgrapht 0.9)

5.2 การทดสอบเครื่องมือ

การทดสอบเครื่องมือนี้ เพื่อตรวจสอบว่าเครื่องมือสามารถตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ รวมทั้งสามารถสร้างกรณีทดสอบเพิ่มเติมได้หรือไม่ โดยการทดสอบนี้จะกระทำกับระบบจำลองการทำงานที่สร้างขึ้นเพื่อทดสอบเครื่องมือนี้โดยเฉพาะ ซึ่งระบบจำลองนั้นประกอบไปด้วย 3 ระบบคือ ระบบคิดเงินร้านค้า ระบบซื้อตั๋วรถประจำทาง และระบบจองห้องพักของโรงแรม แต่ละระบบมีรายละเอียดดังนี้

1) ระบบคิดเงินในร้านค้า ระบบนี้จะจำลองระบบคิดเงินในร้านค้า โดยระบบนี้จะคิดราคาสินค้า และมีโปรโมชันให้ลูกค้าเลือกว่าต้องการได้รับโปรโมชันแบบได้ส่วนลด ได้คะแนนสะสม หรือได้รับเป็นบัตรกำนัล รวมทั้งมีระบบโปรโมชันของลูกค้าวีไอพีด้วย ระบบนี้ประกอบไปด้วย 7 คลาส 14

เมท็อด รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบคิดเงินในร้านค้า รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

2) ระบบซื้อตั๋วรถประจำทาง ระบบนี้จะจำลองระบบคิดเงินค่าตั๋วรถประจำทาง โดยให้ผู้ใช้เลือกเส้นทางการเดินรถ จำนวนตั๋วที่ต้องการซื้อ รวมทั้งมีโปรโมชันลดราคาให้กับนักเรียน ระบบนี้ประกอบไปด้วย 3 คลาส 9 เมท็อด รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบซื้อตั๋วรถประจำทาง รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

3) ระบบจองห้องพักของโรงแรม ระบบนี้จะจำลองระบบการคิดเงินของจองห้องพักของโรงแรม โดยให้ผู้ใช้เลือกชนิดของห้องพัก จำนวนวันที่เข้าพัก จำนวนห้องที่ต้องการจอง สิ่งอำนวยความสะดวกที่ต้องการใช้เช่น สปา บริการรถรับส่ง รวมถึงอาหารเช้า ระบบนี้ประกอบไปด้วย 4 คลาส 10 เมท็อด รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจองห้องพักของโรงแรม รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข

นอกจากนี้ผู้วิจัยได้สร้างระบบทดสอบจำลองขึ้นมาอีก 8 ระบบโดยมีวัตถุประสงค์หลักเพื่อทดสอบการสร้างกราฟและการสร้างข้อมูลทดสอบในส่วนของสร้างกรณีทดสอบ ดังต่อไปนี้

1) ระบบจำลองที่ 1 ประกอบไปด้วย 4 คลาส 13 เมท็อด สร้างขึ้นเพื่อทดสอบการสร้างกราฟการทำงานเมื่อแผนภาพลำดับประกอบไปด้วยการซ้อนกันของคอมบายด์แฟรกเมนท์ชนิดออปชันซ้อนอยู่ในคอมบายด์แฟรกเมนท์ชนิดอัลเทอร์เนทีฟ ว่าสามารถสร้างกราฟการทำงานได้อย่างถูกต้องหรือไม่ รวมทั้งทดสอบการสร้างข้อมูลทดสอบ เมื่อมีการดำเนินการของชนิดของข้อมูลทดสอบดังต่อไปนี้

1.1) ข้อมูลทดสอบชนิดจำนวนจริงที่มีการดำเนินการด้วยเครื่องหมาย >= และ=

1.2) ข้อมูลทดสอบชนิดจำนวนเต็มที่มีการดำเนินการด้วยเครื่องหมาย >=

1.3) ข้อมูลทดสอบชนิดข้อมูลอักขระที่มีการดำเนินการด้วยเครื่องหมาย !=

รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจำลองที่ 1 รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

2) ระบบจำลองที่ 2 ประกอบไปด้วย 5 คลาส 13 เมท็อด สร้างขึ้นเพื่อทดสอบการสร้างกราฟการทำงานเมื่อแผนภาพลำดับประกอบไปด้วยการซ้อนกันของคอมบายด์แฟรกเมนท์ชนิดออป-ชันซ้อนอยู่ในคอมบายด์แฟรกเมนท์ชนิดอัลเทอร์เนทีฟ และคอมบายด์แฟรกเมนท์ชนิดอัลเทอร์เนทีฟซ้อนอยู่ในคอมบายด์แฟรกเมนท์ชนิดอัลเทอร์เนทีฟ ว่าสามารถสร้างกราฟการทำงานได้อย่างถูกต้องหรือไม่ รวมทั้งทดสอบการสร้างข้อมูลทดสอบ เมื่อมีการดำเนินการของชนิดของข้อมูลทดสอบดังต่อไปนี้

2.1) ข้อมูลทดสอบชนิดจำนวนจริงที่มีการดำเนินการด้วยเครื่องหมาย <= และ <

2.2) ข้อมูลทดสอบชนิดจำนวนเต็มที่มีการดำเนินการด้วยเครื่องหมาย <= และ <

2.3) ข้อมูลทดสอบชนิดข้อมูลอักขระที่มีการดำเนินการด้วยเครื่องหมาย !=

2.4) ข้อมูลทดสอบชนิดตรรกะที่มีการดำเนินการด้วยเครื่องหมาย =

รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจำลองที่ 2 รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

3) ระบบจำลองที่ 3 ประกอบไปด้วย 4 คลาส 11 เมท็อด สร้างขึ้นเพื่อทดสอบการสร้างกรณีทดสอบเมื่อแผนภาพลำดับประกอบไปด้วยคอมบายด์แฟรกเมนท์ชนิดออปชันและอัลเทอร์เนทีฟ รวมทั้งมีการดำเนินการของชนิดของข้อมูลทดสอบดังต่อไปนี้

3.1) ข้อมูลทดสอบชนิดจำนวนจริงที่มีการดำเนินการด้วยเครื่องหมาย > และ !=

3.2) ข้อมูลทดสอบชนิดจำนวนเต็มที่มีการดำเนินการด้วยเครื่องหมาย > != และ =

รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจำลองที่ 3 รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

4) ระบบจำลองที่ 4 ประกอบไปด้วย 4 คลาส 8 เมท็อด สร้างขึ้นเพื่อทดสอบการสร้างกรณีทดสอบเมื่อมีการดำเนินการของชนิดของข้อมูลทดสอบดังต่อไปนี้

4.1) ข้อมูลทดสอบชนิดจำนวนจริงที่มีการดำเนินการด้วยเครื่องหมาย >=

4.2) ข้อมูลทดสอบชนิดจำนวนเต็มที่มีการดำเนินการด้วยเครื่องหมาย <= และ >=

4.3) ข้อมูลทดสอบชนิดตรรกะที่มีการดำเนินการด้วยเครื่องหมาย !=

รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจำลองที่ 4 รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

5) ระบบจำลองที่ 5 ประกอบไปด้วย 4 คลาส 10 เมท็อด สร้างขึ้นเพื่อทดสอบการสร้างกรณีทดสอบมีการดำเนินการของชนิดของข้อมูลทดสอบดังต่อไปนี้

5.1) ข้อมูลทดสอบชนิดจำนวนจริงที่มีการดำเนินการด้วยเครื่องหมาย <=

5.2) ข้อมูลทดสอบชนิดจำนวนเต็มที่มีการดำเนินการด้วยเครื่องหมาย >และ =

5.3) ข้อมูลทดสอบชนิดตรรกะที่มีการดำเนินการด้วยเครื่องหมาย =

รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจำลองที่ 5 รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

6) ระบบจำลองที่ 6 ประกอบไปด้วย 4 คลาส 9 เมท็อด สร้างขึ้นเพื่อทดสอบการสร้างกรณีทดสอบเมื่อมีการดำเนินการของชนิดของข้อมูลทดสอบดังต่อไปนี้

6.1) ข้อมูลทดสอบชนิดจำนวนจริงที่มีการดำเนินการด้วยเครื่องหมาย <

6.2) ข้อมูลทดสอบชนิดจำนวนเต็มที่มีการดำเนินการด้วยเครื่องหมาย >

6.3) ข้อมูลทดสอบชนิดข้อมูลอักขระที่มีการดำเนินการด้วยเครื่องหมาย =

6.4) ข้อมูลทดสอบชนิดตรรกะที่มีการดำเนินการด้วยเครื่องหมาย =

รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจำลองที่ 6 รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

7) ระบบจำลองที่ 7 ประกอบไปด้วย 4 คลาส 10 เมท็อด สร้างขึ้นเพื่อทดสอบการสร้างกรณีทดสอบเมื่อมีการดำเนินการของชนิดของข้อมูลทดสอบดังต่อไปนี้

7.1) ข้อมูลทดสอบชนิดจำนวนจริงที่มีการดำเนินการด้วยเครื่องหมาย >, != และ =

7.2) ข้อมูลทดสอบชนิดจำนวนเต็มที่มีการดำเนินการด้วยเครื่องหมาย = และ <

7.3) ข้อมูลทดสอบชนิดข้อมูลอักขระที่มีการดำเนินการด้วยเครื่องหมาย = และ !=

7.4) ข้อมูลทดสอบชนิดตรรกะที่มีการดำเนินการด้วยเครื่องหมาย =

รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจำลองที่ 7 รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

8) ระบบจำลองที่ 8 ประกอบไปด้วย 3 คลาส 3 เมท็อด สร้างขึ้นเพื่อทดสอบการสร้างกรณีทดสอบเมื่อมีเงื่อนไขการเรียกเมท็อดที่มากกว่า 1 วงเล็บ และเชื่อมกันด้วยเครื่องหมาย && รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจำลองที่ 8 รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

นอกจากนี้ผู้วิจัยได้สร้างระบบทดสอบจำลองขึ้นมาอีก 2 ระบบโดยมีวัตถุประสงค์หลักเพื่อทดสอบการตรวจสอบความครอบคลุมการทดสอบในกรณีที่มีเมท็อดหรือคลาสที่มีอยู่ในซอร์สโค้ดแต่ไม่มีในแผนภาพลำดับและคำอธิบายยูสเคสดังนี้

1) ระบบจำลองที่ 9 ประกอบไปด้วย 5 คลาส 14 เมท็อด แต่ปรากฏในแผนภาพลำดับเพียง 4 คลาส 10 เมท็อด รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจำลองที่ 9 รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

2) ระบบจำลองที่ 10 ประกอบไปด้วย 4 คลาส 10 เมท็อด แต่ปรากฏในแผนภาพลำดับเพียง 4 คลาส 9 เมท็อด รายละเอียดของคลาส แผนภาพลำดับ คำอธิบายยูสเคสของระบบจำลองที่ 10 รวมทั้งกรณีทดสอบจะอธิบายไว้ในภาคผนวก ข.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

5.3 ผลการทดสอบ

5.3.1 การทดสอบการตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการ

เมื่อได้ทดสอบเครื่องมือโดยใช้กรณีทดสอบที่สร้างขึ้นกับระบบทดสอบ ได้รับผลการทดสอบดังตารางที่ 5.1

ตารางที่ 5.1 ผลการทดสอบการตรวจสอบความครอบคลุมกับระบบจำลอง

ชื่อระบบ	จำนวนกรณีทดสอบ	จำนวนคลาสทั้งหมด	จำนวนคลาสที่ไม่ครอบคลุม	จำนวนเมท็อดทั้งหมด	จำนวนเมท็อดที่ไม่ครอบคลุม	ความครอบคลุมคลาส (ร้อยละ)	ความครอบคลุมเมท็อด (ร้อยละ)
ระบบคิดเงินในร้านค้า	3	6	1	12	2	83.33%	83.33%
ระบบซื้อตั๋วรถประจำทาง	2	3	1	9	1	66.67%	88.85%
ระบบจองห้องพักของโรงแรม	2	4	2	10	2	50%	80%
ระบบจำลองที่ 1	1	4	3	13	9	25%	30.76%
ระบบจำลองที่ 2	1	5	4	13	10	20%	23.8%
ระบบจำลองที่ 3	1	4	3	11	8	25%	27.27%
ระบบจำลองที่ 4	1	4	3	8	6	25%	25%
ระบบจำลองที่ 5	1	4	3	10	7	25%	30%
ระบบจำลองที่ 6	1	4	3	9	8	25%	11.11%
ระบบจำลองที่ 7	1	4	3	10	9	25%	10%
ระบบจำลองที่ 8	1	3	1	3	1	66.66%	66.66%
ระบบจำลองที่ 9	1	5	4	14	13	20%	7.14%

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 5.1 ผลการทดสอบการตรวจสอบความครอบคลุมกับระบบจำลอง(ต่อ)

ระบบจำลองที่ 10	1	4	3	10	9	25%	10%

5.3.2 การทดสอบการสร้างกรณีทดสอบเพิ่มเติม

หลังจากเครื่องมือได้ตรวจสอบความครอบคลุมการทดสอบแล้ว เครื่องมือได้สร้างกรณีทดสอบเพิ่มเติม ได้รับผลการทดสอบดังตารางที่ 5.2

ตารางที่ 5.2 ผลการทดสอบสร้างกรณีทดสอบเพิ่มเติมกับระบบจำลอง

ชื่อระบบ	จำนวนกรณีทดสอบที่สร้างเพิ่มเติม	ค่าความครอบคลุมคลาส หลังจากทดสอบด้วยกรณีทดสอบที่สร้างขึ้น	ค่าความครอบคลุมเม-ท็อด หลังจากทดสอบด้วยกรณีทดสอบที่สร้างขึ้น
ระบบคิดเงินในร้านค้า	2	100%	100%
ระบบซื้อตั๋วรถประจำทาง	1	100%	100%
ระบบจองห้องพักของโรงแรม	2	100%	100%
ระบบจำลองที่ 1	9	100%	100%
ระบบจำลองที่ 2	7	100%	100%
ระบบจำลองที่ 3	5	100%	100%
ระบบจำลองที่ 4	4	100%	100%
ระบบจำลองที่ 5	6	100%	100%
ระบบจำลองที่ 6	5	100%	100%
ระบบจำลองที่ 7	4	100%	100%
ระบบจำลองที่ 8	2	100%	100%
ระบบจำลองที่ 9	3	80%	71.43%
ระบบจำลองที่ 10	4	75%	90%

5.4 สรุปผลการทดสอบ

จากการทดสอบเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการในส่วนของการตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการ และการสร้างกรณีทดสอบเพิ่มเติมได้ผลดังตารางที่ 5.1 และตารางที่ 5.2 สามารถสรุปผลการทดลองได้ว่า เครื่องมือสามารถตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการและสร้างสามารถสร้างกรณีทดสอบเพิ่มเติมในกรณีที่การทดสอบไม่ครอบคลุมได้

จากตารางที่ 5.2 พบว่าระบบจำลองที่ 9 และ 10 นั้น ค่าความครอบคลุมคลาส หลังจากทดสอบด้วยกรณีทดสอบที่สร้างขึ้น และค่าความครอบคลุมเมท็อด หลังจากทดสอบด้วยกรณีทดสอบที่สร้างขึ้น จะไม่เท่ากับ 100% เนื่องจากในระบบจำลองที่ 9 นั้นมีคลาสที่ไม่ปรากฏในแผนภาพลำดับ 1 คลาสและมีเมท็อดที่ไม่ปรากฏในแผนภาพลำดับ 4 เมท็อด และระบบจำลองที่ 10 นั้น มีเมท็อดที่ไม่ปรากฏในแผนภาพลำดับ 1 เมท็อด ซึ่งเครื่องมือสามารถระบุชื่อคลาสและเมท็อดที่ไม่ปรากฏในแผนภาพลำดับได้อย่างถูกต้องในการตรวจสอบความครอบคลุมการทดสอบหลังจากทดสอบด้วยกรณีทดสอบที่สร้างขึ้นแล้ว

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 6
สรุปผลการวิจัยและข้อเสนอแนะ

จากการศึกษา วิจัย และพัฒนาเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการ สามารถสรุปผลการวิจัย ข้อจำกัดของเครื่องมือ และแนวทางในการพัฒนาต่อ โดยมีรายละเอียดดังนี้

6.1 สรุปผลการวิจัย

งานวิจัยนี้นำเสนอเครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ ซึ่งเครื่องมือนี้สามารถตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุได้ โดยแสดงผลออกมาในรูปแบบสัดส่วนของจำนวนคลาสที่ครอบคลุมต่อจำนวนคลาสทั้งหมดคิดเป็นร้อยละ และสัดส่วนของจำนวนเมท็อดที่ครอบคลุมต่อจำนวนเมท็อดทั้งหมดคิดเป็นร้อยละ ทั้งยังสามารถระบุคลาสและเมท็อดที่ไม่ครอบคลุมและออกรายงานการตรวจสอบความครอบคลุมได้อีกด้วย ทำให้ผู้ใช้ทราบถึงความครอบคลุมของการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ เพิ่มความน่าเชื่อถือในการทดสอบ

นอกจากนี้ เครื่องมือยังสามารถสร้างกรณีทดสอบเพิ่มเติมในกรณีที่พบว่าการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุนั้นยังไม่ครอบคลุม ทำให้ช่วยลดเวลาในการสร้างกรณีทดสอบเพิ่มเติม และเพิ่มความครอบคลุมในการทดสอบแบบบูรณาการอีกด้วย

6.2 ข้อจำกัดของเครื่องมือ

เครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ ยังมีข้อจำกัดดังต่อไปนี้

1) เครื่องมือสามารถทำงานกับโปรแกรมที่พัฒนาด้วยภาษาจาวาเท่านั้น

2) การดำเนินการทดสอบนั้น ผู้ทดสอบจะต้องเป็นผู้ดำเนินการด้วยตนเอง

3) เครื่องมือยังไม่รองรับการพัฒนาโปรแกรมที่ประกอบด้วยคุณสมบัติการสืบทอดคุณสมบัติ (Inheritance) และ การพ้องรูป (Polymorphism)

4) กรณีทดสอบที่สร้างขึ้นด้วยข้อมูลทดสอบที่เป็นสตริง จะเป็นข้อมูลที่ถูกสุ่มขึ้นมา ดังนั้นในบางกรณีจะไม่สามารถใช้กรณีทดสอบที่สร้างขึ้นมาได้ เช่นข้อมูลทดสอบที่เป็นชื่อจังหวัด หรือชื่อเฉพาะต่างๆ เป็นต้น

6.3 แนวทางการพัฒนาต่อ

1) พัฒนาเครื่องมือให้รองรับซอฟต์แวร์เชิงวัตถุที่พัฒนาด้วยภาษาอื่นๆ

2) พัฒนาเครื่องมือให้ดำเนินการทดสอบได้อย่างอัตโนมัติ

3) พัฒนาเครื่องมือให้รองรับการพัฒนาโปรแกรมประกอบด้วยคุณสมบัติการสืบทอดคุณสมบัติ และ การพ้องรูป โดยมีแนวคิดดังต่อไปนี้

3.1) การสืบทอดคุณสมบัติ: สามารถพัฒนาเครื่องมือให้รองรับได้ในส่วนของการแทรกซอร์สโค้ด โดยเมื่อเครื่องมืออ่านซอร์สโค้ดแล้วพบการประกาศคลาสลูก ให้เครื่องมือบันทึกชื่อคลาสหน้าคำหลัก (keyword) คำว่า extend เท่านั้น

3.2) การพ้องรูป: สามารถพัฒนาเครื่องมือให้รองรับได้ในส่วนของการแทรกซอร์สโค้ด โดยเมื่อเครื่องมืออ่านซอร์สโค้ดแล้วพบว่ามีเมท็อดที่มีชื่อซ้ำกันแต่มีอาร์กิวเมนต์ที่ต่างกัน ให้บันทึกชื่อเมท็อดนั้นให้เป็นเมท็อดที่อยู่ในกลุ่มเมท็อดที่มีการพ้องรูป ซึ่งเมท็อดกลุ่มนี้จะมีตัวแปรที่ใช้นับการเรียกใช้ที่ต่างจากเมท็อดทั่วไปเช่น มีการเติมชนิดและชื่อของอาร์กิวเมนต์ต่อท้ายชื่อตัวแปร

4) พัฒนาเครื่องมือให้รองรับการแสดงผลความครอบคลุมในรูปแบบอื่น เช่นความครอบคลุมเส้นทางการเรียกใช้ (path coverage)

5) พัฒนาเครื่องมือให้สามารถสร้างข้อมูลทดสอบที่เป็นสตริงที่มีความหมายได้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

รายการอ้างอิง

[1] Shaukat Ali, L.C.B., Muhammad Jaffar-ur Rehman, Hajra Asghar, Muhammad Zohaib Z. Iqbal, Aamer Nadeem., *A state-based approach to integration testing based on UML models.* Information and Software Technology, 2007. **49**(11–12): p. 1087–1106.

[2] E.S.F. Najumudheen, R.M., Debasis Samanta., *A Dependence Graph-based Test Coverage Analysis Technique for Object-Oriented Programs.*, in *2009 Sixth International Conference on Information Technology: New Generations*. 2009 p. 763 – 768.

[3] Muhammad Shahid, S.I., *An Evaluation of Test Coverage Tools in Software Testing.*, in *2011 International Conference on Telecommunication Technology and Applications.* 2011 Singapore. p. 216-222.

[4] Patton., R., *Software testing.* 2005, Indiana: Sams Indianapolis

[5] Jorgensen., P.C., *Software Testing A Craftsman's Approach Third Edition.* . 2008, New York: Auerbach Publications.

[6] พรศิริ หมื่นไชยศรี, *Software Metric* มาตรวัดซอฟต์แวร์. 2548: เอกสารคำสอนวิชา 2110721 ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย.

[7] Elliotte Rusty Harold, W.S.M., *XML in A Nutshell 3rd Edition.* 2004, California: O'Reilly Media.

[8] สุธินี ภูมิสนธิ์, เครื่องมือทดสอบลำดับเว็บเพจของโปรแกรมประยุกต์บนเว็บโดยใช้แผนภาพซีเควนซ์ ของยูเอ็มแอล, in ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ 2552, จุฬาลงกรณ์มหาวิทยาลัย.

[9] Alan Dennis, B.H.W., David Tagarden., *Systems Analysis and Design with UML An Object-Oriented Approach.* . 2010, New Jersey: John Wiley & Sons

[10] ฉัฐธร ทองระอา, การสร้างกรณีทดสอบจากแผนภาพซีเควนซ์. , in ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี 2550, จุฬาลงกรณ์มหาวิทยาลัย.

[11] Man Machine Systems. *Jcover*. 2009; Available from: www.mmsindia.com/JCover.html

[12] Ashalatha Nayak, D.S., *Model-Based Test Cases Synthesis using UML Interaction Diagrams.* ACM SIGSOFT Software Engineering Notes, 2009. **34**(2): p. 1-10.

[13] เศรษฐพงศ์ ลีฬหรัตนรักษ์, วิธีการสร้างกรณีทดสอบโดยอัตโนมัติจากยูสเคส., in ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ 2547, จุฬาลงกรณ์มหาวิทยาลัย.

ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก ก

พจนานุกรมข้อมูล

ตารางที่ ก-1 พจนานุกรมข้อมูลตาราง classmethoddata

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง
ID	หมายเลขเมท็อด	Integer	PK	ไม่ว่าง
Methodname	ชื่อเมท็อด	Varchar(100)		ไม่ว่าง
Classname	ชื่อคลาสที่เมท็อดเป็นสมาชิก	Varchar(100)		ไม่ว่าง
SDID	ลำดับแผนภาพลำดับ	Integer	FK	ไม่ว่าง
UCDID	ลำดับคำอธิบายยูสเคส	Integer	FK	ไม่ว่าง

ตารางที่ ก-2 พจนานุกรมข้อมูลตาราง sdlocate

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง
SDID	ลำดับแผนภาพลำดับ	Integer		ไม่ว่าง
SDLocation	ชื่อโฟลเดอร์ที่ใช้เก็บแผนภาพลำดับ	Varchar(255)		ไม่ว่าง

ตารางที่ ก-3 พจนานุกรมข้อมูลตาราง ucdlocate

ชื่อ	คำอธิบาย	ชนิดข้อมูล	คีย์	ค่าว่าง
UCDID	ลำดับคำอธิบายยูสเคส	Integer		ไม่ว่าง
UCDLocation	ชื่อโฟลเดอร์ที่ใช้เก็บคำอธิบายยูสเคส	Varchar(255)		ไม่ว่าง

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

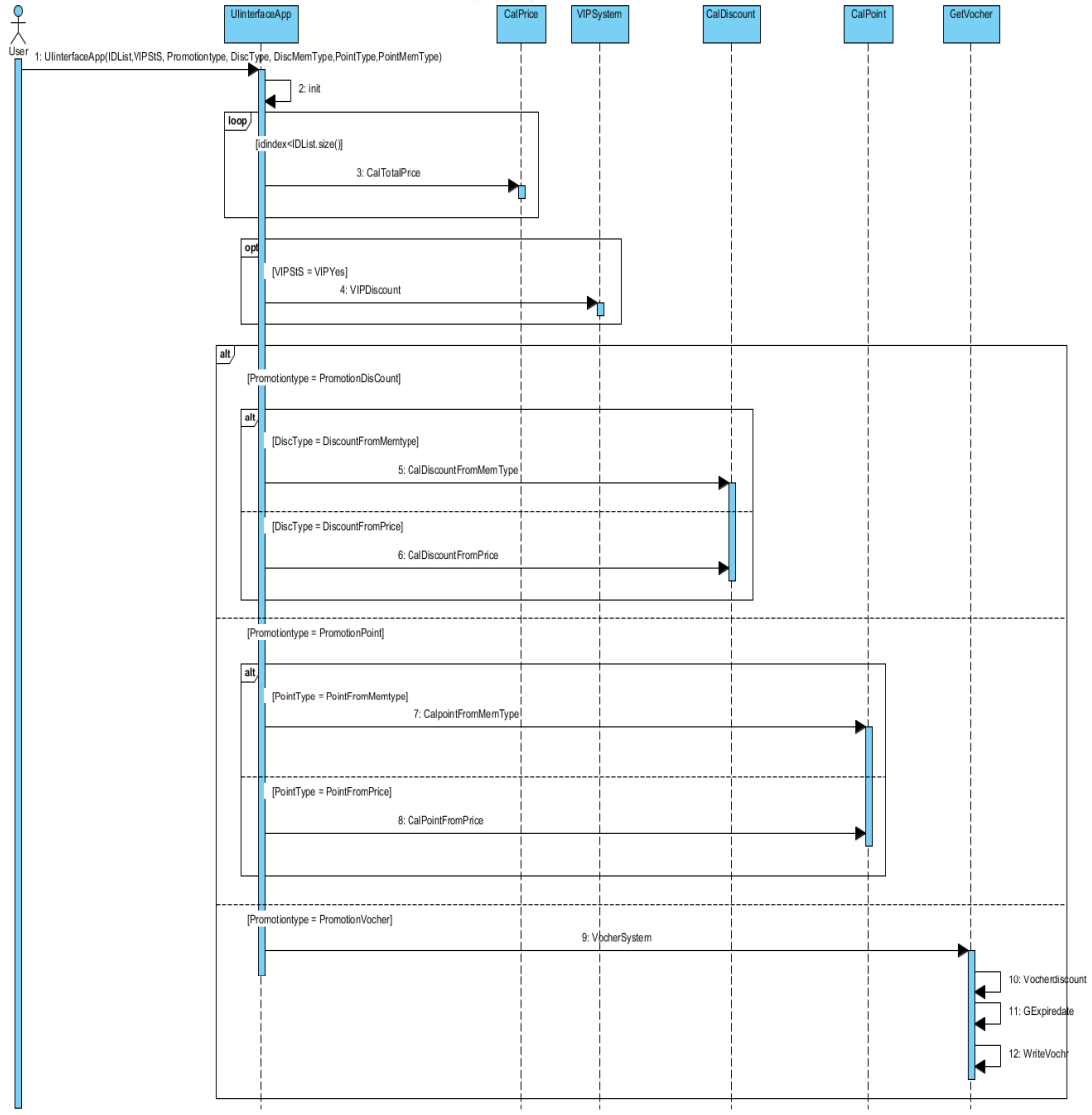
ภาคผนวก ข

ตัวอย่างระบบทดสอบ

1. ระบบคิดเงินในร้านค้า

การทดสอบด้วยระบบคิดเงินในร้านค้าสามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

1.1) แผนภาพลำดับ



ภาพที่ ข-1 แผนภาพลำดับของระบบทดสอบการคิดเงินในร้านค้า

1.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

```
Sequence diagram number,Promotion_TestData_Ver2,,,,
Pre-condition,Cashier calculate the merchandise price,,,,
Success scenario condition,"[0] (VIPStS ==  "VIPYes" )&&(Promotiontype == "PromotionDisCount")&&(DiscType ==
"DiscountFromMemtype")""""""",,,,
Alternative scenario condition,"[1] (VIPStS ==  "VIPNo")&&(Promotiontype == "PromotionDisCount")&&(DiscType ==
"DiscountFromPrice")""",,,,
"[2] (VIPStS ==  "VIPNo")&&(Promotiontype == "PromotionPoint")&&(PointType == "PointFromMemtype")""",,,,,
"[3] (VIPStS ==  "VIPNo")&&(Promotiontype == "PromotionPoint")&&(PointType == "PointFromPrice")""",,,,,
"[4] (VIPStS ==  "VIPNo")&&(Promotiontype == "PromotionVocher")""""""",,,,,
Post condition of success scenario,[0] System display total price and Discount,,,,
Post condition of alternative scenario,"[1] System display total price and Discount
[2] System display total price and Point
[3] System display total price and Point
[4] System Display totalprice , print vocher and display   Vocher Already Printed ""  """""",,,,
input data,,,,,
Name,Type,Size ,Min ,Max,Scenario no.
ID,Int,4,1001,1015,[0][1][2][3][4]
VIPStS,String,-,VIPNo,VIPYes,[0][1][2][3][4]
Promotiontype,String,-,PromotionDisCount,PromotionVocher,[0][1][2][3][4]
DiscType,String,-,DiscountFromMemtype,DiscountFromPrice,[0][1]
PointType,String,-,PointFromMemtype,PointFromPrice,[2][3]
DiscMemType,String,-,DiscountMemTypeA,DiscountMemTypeC,[0]
PointMemType,String,-,PointMemtypeA,PointMemTypeC,[2]
```

1.3) ตัวอย่างกรณีทดสอบ

Test Case

Test case ID	**1**
Test case name	**Promotion_TestData_Ver2[0]**
Test case description	**(VIPStS == "VIPYes")&&(Promotiontype == "PromotionVocher")**
Initial Value	**Cashier calculate the merchandise price**
Input Value	
VIPStS	**"VIPYes"**
Promotiontype	**"PromotionVocher"**
ID	**1003**
Expected Output	**System Display totalprice , print vocher and display "Vocher Already Printed"**

ภาพที่ ข-2 ตัวอย่างกรณีทดสอบระบบคิดเงินในร้านค้า กรณีทดสอบที่ 1

Test Case

Test case ID	2
Test case name	Promotion_TestData_Ver2[1]
Test case description	(VIPStS == "VIPNo")&&(Promotiontype == "PromotionDisCount")&&(DiscType == "DiscountFromPrice")""
Initial Value	Cashier calculate the merchandise price
Input Value	
VIPStS	"VIPNo"
Promotiontype	"PromotionDisCount"
DiscType	"DiscountFromPrice"
ID	1011
Expected Output	System display total price and Discount

ภาพที่ ข-3 ตัวอย่างกรณีทดสอบระบบคิดเงินในร้านค้า กรณีทดสอบที่ 2

Test Case

Test case ID	3
Test case name	Promotion_TestData_Ver2[0]
Test case description	(VIPStS == "VIPYes")&&(Promotiontype == "PromotionDisCount")&&(DiscType == "DiscountFromMemtype")"""""
Initial Value	Cashier calculate the merchandise price
Input Value	
VIPStS	"VIPYes"
Promotiontype	"PromotionDisCount"
DiscType	"DiscountFromMemtype"
ID	1003
DiscMemType	DiscountMemTypeA
Expected Output	System display total price and Discount

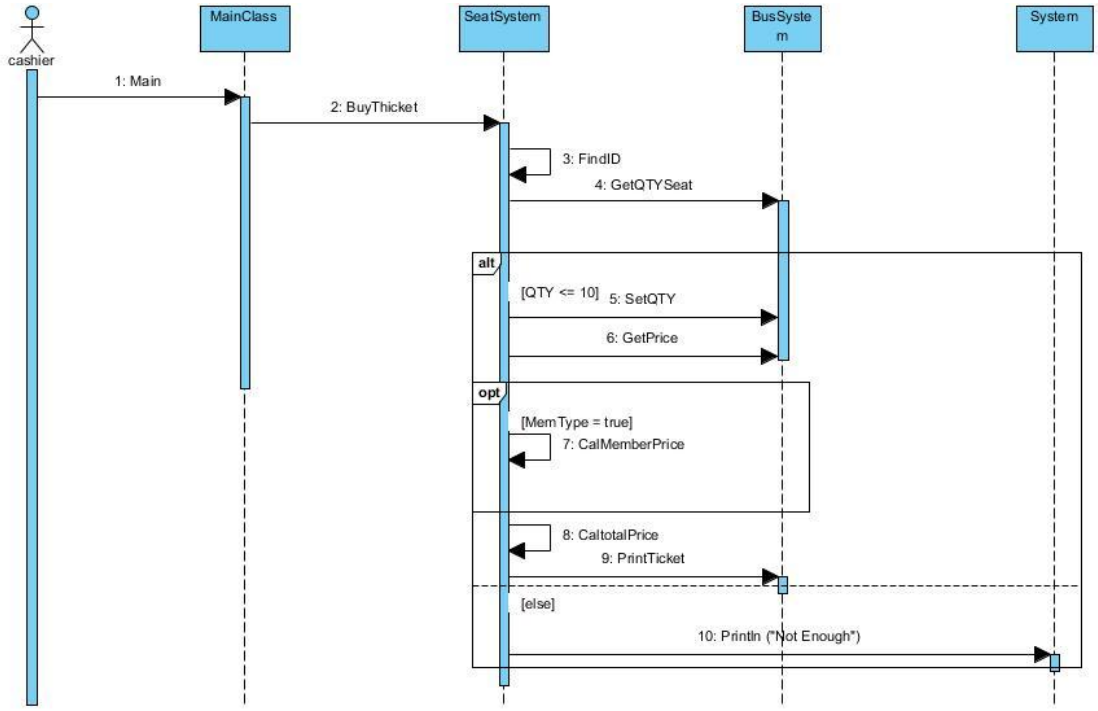
ภาพที่ ข-4 ตัวอย่างกรณีทดสอบระบบคิดเงินในร้านค้า กรณีทดสอบที่ 3

2. ระบบซื้อตั๋วรถประจำทาง

การทดสอบด้วยระบบซื้อตั๋วรถประจำทางสามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูส-เคส และกรณีทดสอบดังนี้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.1) แผนภาพลำดับ



ภาพที่ ข-5 แผนภาพลำดับของระบบทดสอบการซื้อตั๋วรถประจำทาง

2.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,BuyTicket_22062014,,,,
Pre-condition,Cashier calculate the Thicket Price,,,,
Success scenario condition,"[0] (SeatQTY >= QTY)&&(MemType == true)",,,,
Alternative scenario condition,"[1] (SeatQTY >= QTY)&&(MemType == false)
[2] (SeatQTY < QTY)",,,,
Post condition of success scenario,[0] System show total price with 10% discount,,,,
Post condition of alternative scenario,"[1] System show total price without discount
[2] System show ""Not Enough seat""",,,,
input data,,,,,
Name,Type,Size ,Min ,Max,Scenario no.
Way,int,-,1,9,[0][1][2]
QTY,int,-,1,10,[0][1][2]
Mtype,boolean,-,-,-,[0][1]

2.3) กรณีทดสอบ

Test Case

Test case ID	1
Test case name	BuyTicket_22062014[1]
Test case description	(QTY<=10)&&(MemType == false)
Initial Value	Cashier calculate the Thicket Price
Input Value	
Way	2
QTY	2
Mtype	false
Expected Output	System show total price without discount

ภาพที่ ข-6 ตัวอย่างกรณีทดสอบระบบซื้อตั๋วรถประจำทาง กรณีทดสอบที่ 1

Test Case

Test case ID	2
Test case name	BuyTicket_22062014[2]
Test case description	(QTY > 10)
Initial Value	Cashier calculate the Thicket Price
Input Value	
Way	1
QTY	7
Expected Output	System show "Not Enough seat"

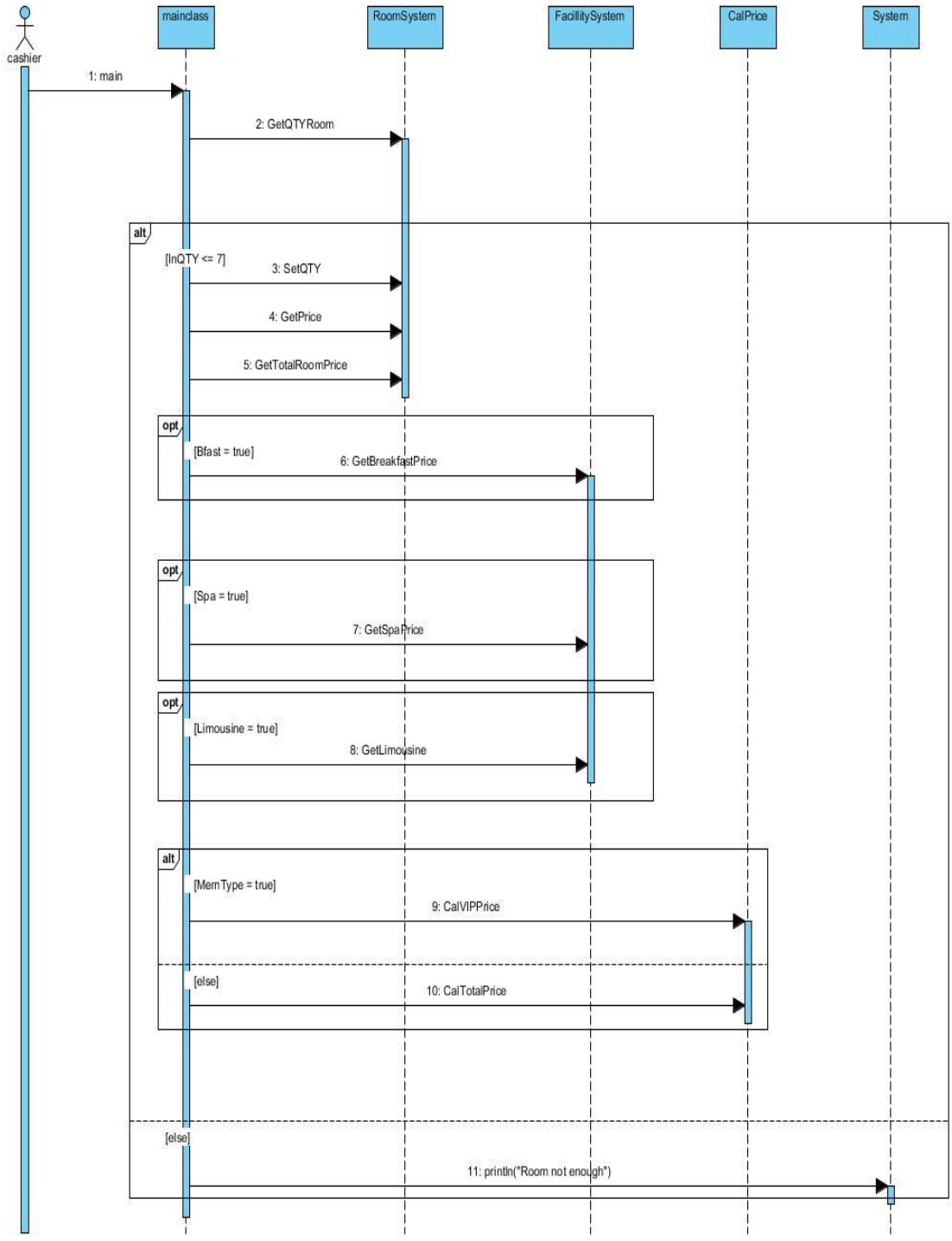
ภาพที่ ข-7 ตัวอย่างกรณีทดสอบระบบซื้อตั๋วรถประจำทาง กรณีทดสอบที่ 2

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3. ระบบจองห้องพักของโรงแรม

การทดสอบด้วยระบบจองห้องพักของโรงแรมสามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

3.1) แผนภาพลำดับ



ภาพที่ ข-8 แผนภาพลำดับของระบบทดสอบการจองห้องพักของโรงแรม

3.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,Hotel,,,,

Pre-condition,Cashier reserve the hotel room ,,,,

Success scenario condition,[0] (RemainRoom >= InQTY)&&(Bfast == true)&&(Spa == true)&&(Limousine == true)&&(MemType == true),,,,

Alternative scenario condition,"[1] (RemainRoom >= InQTY)&&(Bfast == true)&&(Spa == true)&&(Limousine == false)&&(MemType == false)

[2] (RemainRoom >= InQTY)&&(Bfast == true)&&(Spa == false)&&(Limousine == true)&&(MemType == false)

[3] (RemainRoom >= InQTY)&&(Bfast == true)&&(Spa == false)&&(Limousine == false)&&(MemType == false)

[4] (RemainRoom >= InQTY)&&(Bfast == false)&&(Spa == true)&&(Limousine == true)&&(MemType == false)

[5] (RemainRoom >= InQTY)&&(Bfast == false)&&(Spa == true)&&(Limousine == false)&&(MemType == false)

[6] (RemainRoom >= InQTY)&&(Bfast == false)&&(Spa == false)&&(Limousine == true)&&(MemType == false)

[7] (RemainRoom >= InQTY)&&(Bfast == false)&&(Spa == false)&&(Limousine == false)&&(MemType == false)

[8] (RemainRoom >= InQTY)&&(Bfast == true)&&(Spa == true)&&(Limousine == true)&&(MemType == true)

[9] (RemainRoom >= InQTY)&&(Bfast == true)&&(Spa == false)&&(Limousine == true)&&(MemType == true)

[10] (RemainRoom >= InQTY)&&(Bfast == true)&&(Spa == false)&&(Limousine == false)&&(MemType == true)

[11] (RemainRoom >= InQTY)&&(Bfast == false)&&(Spa == true)&&(Limousine == true)&&(MemType == true)

[12] (RemainRoom >= InQTY)&&(Bfast == false)&&(Spa == true)&&(Limousine == false)&&(MemType == true)

[13] (RemainRoom >= InQTY)&&(Bfast == false)&&(Spa == false)&&(Limousine == true)&&(MemType == true)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

[14] (RemainRoom >= InQTY)&&(Bfast == false)&&(Spa == false)&&(Limousine == false)&&(MemType == true)

[15] (RemainRoom >= InQTY)&&(Bfast == true)&&(Spa == true)&&(Limousine == false)&&(MemType == true)

[16] (RemainRoom < InQTY)",,,,

Post condition of success scenario,"[0] System show total room price with breakfast price, spa price, Limousine price : with 10 %discount",,,,

Post condition of alternative scenario,"[1] System show total room price with breakfast price, spa price

[2] System show total room price with breakfast price, Limousine price

[3] System show total room price with breakfast price

[4] System show total room price with spa price, Limousine price

[5] System show total room price with spa price

[6] System show total room price with Limousine price

[7] System show total room price

[8] System show total room price with breakfast price, spa price, Limousine price : with 10 %discount

[9] System show total room price with breakfast price, Limousine price : with 10 %discount

[10] System show total room price with breakfast price : with 10 %discount

[11] System show total room price with spa price, Limousine price : with 10 %discount

[12] System show total room price with spa price : with 10 %discount

[13] System show total room price with Limousine price : with 10 %discount

[14] System show total room price : with 10 %discount

[15] System show total room price with breakfast price, spa price : with 10 %discount

[16] System show ""Not Enough Room""",,,,

input data,,,,,

Name,Type,Size ,Min ,Max,Scenario no.

InRoomType,int ,-,1,4,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16]

Chulalongkorn University

InQTY,int ,-,1,7,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16]

IntDay,int ,-,1,9,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16]

Bfast,boolean,-,-,-,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15]

Spa,boolean,-,-,-,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15]

Dspa,int ,-,1,9,[0][1][4][5][8][11][12][15]

Limousine,boolean,-,-,-,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16]

RLimousine,int ,-,1,2,[0][2][4][6][8][9][11][13]

MemType,boolean,-,-,-,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15]

3.3) กรณีทดสอบ

Test Case

Test case ID	**1**
Test case name	**Hotel[3]**
Test case description	**(InQTY <= 7)&&(Bfast == true)&&(Spa == false)&&(Limousine == false)&&(MemType == false)**
Initial Value	**Cashier reserve the hotel room**
Input Value	
InRoomType	**1**
InQTY	**2**
IntDay	**3**
Bfast	**true**
Spa	**false**
Limousine	**false**
MemType	**false**
Expected Output	**System show total room price with breakfast price**

ภาพที่ ข-9 ตัวอย่างกรณีทดสอบระบบจองห้องพักของโรงแรม กรณีทดสอบที่ 1

Test Case

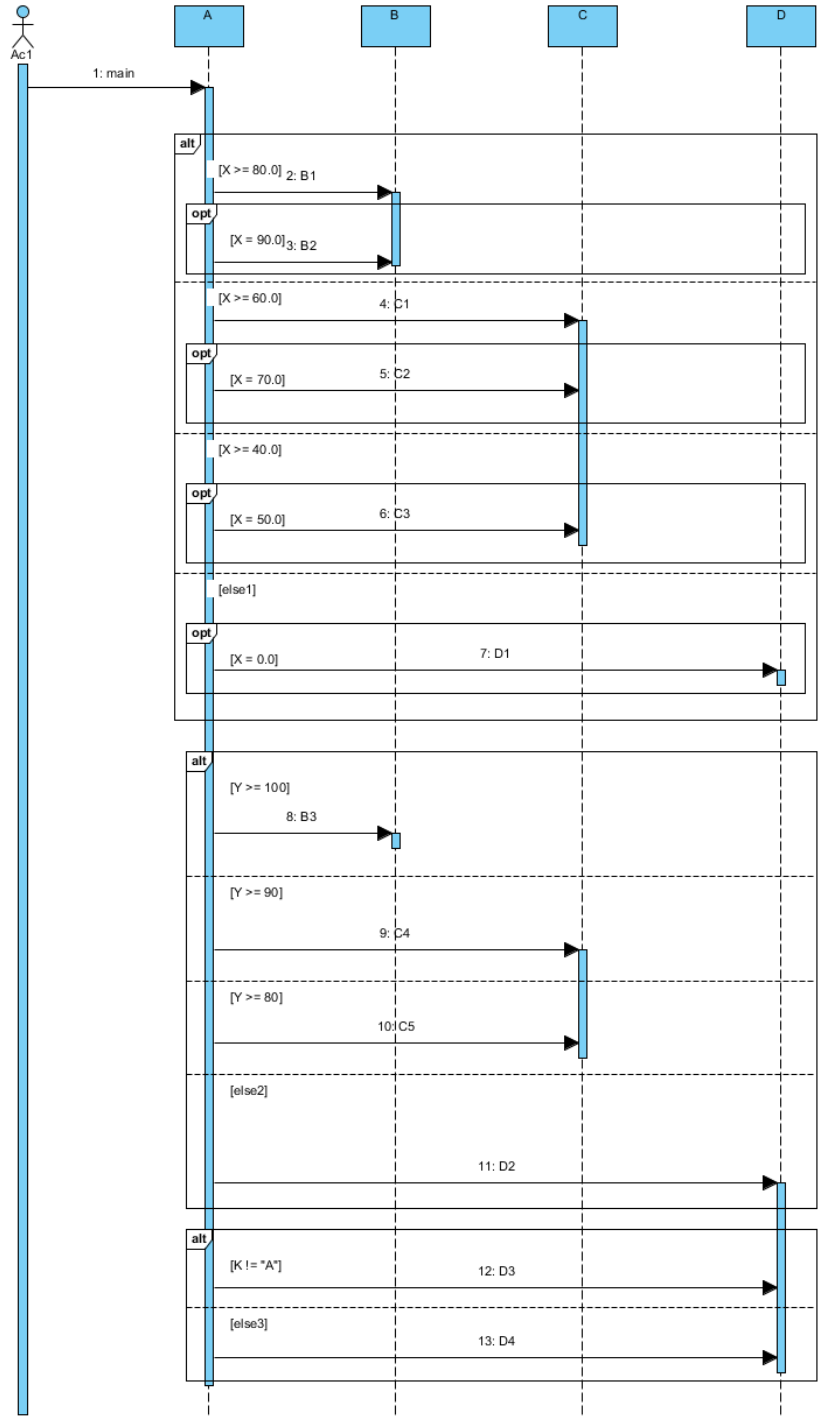
Test case ID	**2**
Test case name	**Hotel[6]**
Test case description	**(InQTY <= 7)&&(Bfast == false)&&(Spa == false)&&(Limousine == true)&&(MemType == false)**
Initial Value	**Cashier reserve the hotel room**
Input Value	
InRoomType	**2**
InQTY	**2**
IntDay	**3**
Bfast	**false**
Spa	**false**
Limousine	**true**
RLimousine	**1**
MemType	**false**
Expected Output	**System show total room price with Limousine price**

ภาพที่ ข-10 ตัวอย่างกรณีทดสอบระบบจองห้องพักของโรงแรม กรณีทดสอบที่ 2

4. ระบบจำลองที่ 1

การทดสอบด้วยระบบจำลองที่ 1 สามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

4.1) แผนภาพลำดับ



ภาพที่ ข-11 แผนภาพลำดับของระบบจำลองที่ 1

4.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,TestMockup2,,,,

Pre-condition,Test Simulation ,,,,

Success scenario condition,[0] (X <= 50.0)&&(StringEQ1 == "Triangle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 20)&&(W < 30.0) ,,,,

Alternative scenario condition,"[1] (X <= 50.0)&&(StringEQ1 == "Triangle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 20)&&(W < 30.0)

[2] (X <= 50.0)&&(StringEQ1 == "Triangle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[3] (X <= 50.0)&&(StringEQ1 != "Triangle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 50)&&(W < 60.0)

[4] (X <= 50.0)&&(StringEQ1 != "Triangle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[5] (X <= 50.0)&&(StringEQ1 != "Triangle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[6] (X <= 80.0)&&(StringEQ2 == "Regtangle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 50)&&(W < 60.0)

[7] (X <= 80.0)&&(StringEQ2 == "Regtangle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[8] (X <= 80.0)&&(StringEQ2 == "Regtangle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[9] (X <= 80.0)&&(StringEQ2 != "Regtangle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 50)&&(W < 60.0)

[10] (X <= 80.0)&&(StringEQ2 != "Regtangle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[11] (X <= 80.0)&&(StringEQ2 != "Regtangle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[12] (X <= 120.0)&&(StringEQ3 == "Circle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 50)&&(W < 60.0)

[13] (X <= 120.0)&&(StringEQ3 == "Circle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[14] (X <= 120.0)&&(StringEQ3 == "Circle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[15] (X <= 120.0)&&(StringEQ3 != "Circle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z <

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

50)&&(W < 60.0)

[16] (X <= 120.0)&&(StringEQ3 != "Circle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[17] (X <= 120.0)&&(StringEQ3 != "Circle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[18] (X > 50.0)&&(X > 80.0)&&(X > 120.0)&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 50)&&(W < 60.0)

[19] (X > 50.0)&&(X > 80.0)&&(X > 120.0)&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[20] (X > 50.0)&&(X > 80.0)&&(X > 120.0)&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)
",,,,

Post condition of success scenario,"[0] System show B1,D1,E1,E2",,,,

Post condition of alternative scenario,"[1] System show B1,D2,E1,E2

[2] System show B1,D3,E1,E2

[3] System show B2,D21,E1,E2

[4] System show B2,D2,E1,E2

[5] System show B2,D3,E1,E2

[6] System show C1,D1,E1,E2

[7] System show C1,D2,E1,E2

[8] System show C1,D3,E1,E2

[9] System show C2,D1,E1,E2

[10] System show C2,D2,E1,E2

[11] System show C2,D3,E1,E2

[12] System show C3,D1,E1,E2

[13] System show C3,D2,E1,E2

[14] System show C3,D3,E1,E2

[15] System show C4,D1,E1,E2

[16] System show C4,D2,E1,E2

[17] System show C4,D3,E1,E2

[18] System show C5,D1,E1,E2

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

[19] System show C5,D2,E1,E2

[20] System show C5,D3,E1,E2",,,,

input data,,,,,

Name,Type,Size ,Min ,Max,Scenario no.

X,float,-,0,200,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16][17][18][19][20]

StringEQ1,String,12,-,-,[0][1][2][3][4][5]

StringEQ2,String,12,-,-,[6][7][8][9][10][11]

StringEQ3,String,12,-,-,[12][13][14][15][16][17]

Y,int,-,0,150,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16][17][18][19][20]

booleanEQ1,boolean,-,-,-,[0][3][6][9][12][15][18]

booleanEQ2,boolean,-,-,-,[1][4][7][10][13][16][19]

Z,int,-,0,100,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16][17][18][19][20]

W,float,-,0.0,120.0,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16][17][18][19][20]

4.3) กรณีทดสอบ

Test Case

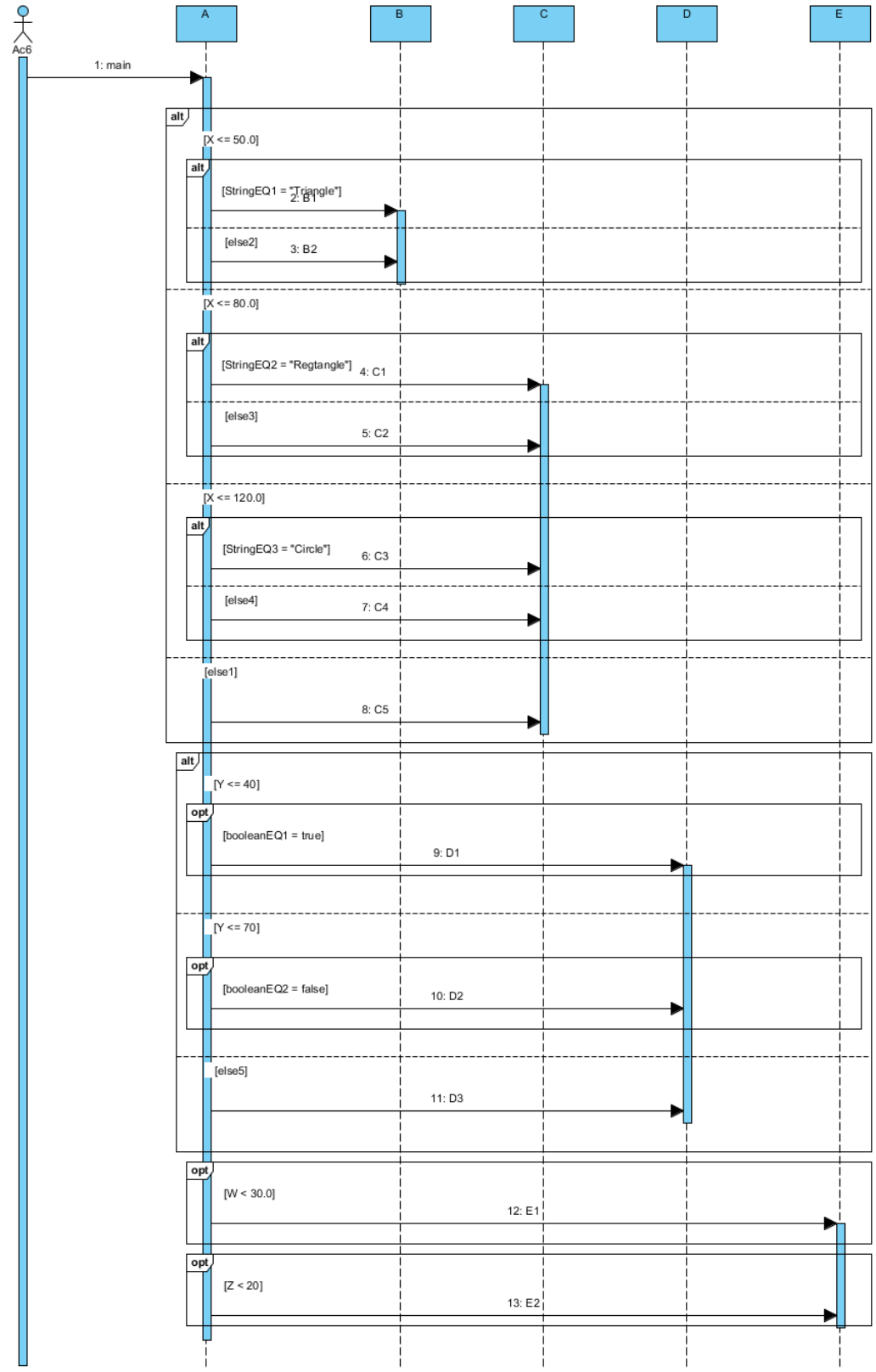
Test case ID	1
Test case name	TestMockup1
Test case description	(X = 0.0)&&(Y < 80)&&(K != "A")
Initial Value	Testing simulation
Input Value	
X	0
Y	79
K	"JERJEAQ"
Expected Output	System show D1,D2,D4

ภาพที่ ข-12 กรณีทดสอบของระบบจำลองที่ 1

5. ระบบจำลองที่ 2

การทดสอบด้วยระบบจำลองที่ 2 สามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

5.1) แผนภาพลำดับ



ภาพที่ ข-13 แผนภาพลำดับของระบบจำลองที่ 2

5.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,TestMockup2,,,,

Pre-condition,Test Simulation ,,,,

Success scenario condition,[0] (X <= 50.0)&&(StringEQ1 == "Triangle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 20)&&(W < 30.0) ,,,,

Alternative scenario condition,"[1] (X <= 50.0)&&(StringEQ1 == "Triangle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 20)&&(W < 30.0)

[2] (X <= 50.0)&&(StringEQ1 == "Triangle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[3] (X <= 50.0)&&(StringEQ1 != "Triangle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 50)&&(W < 60.0)

[4] (X <= 50.0)&&(StringEQ1 != "Triangle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[5] (X <= 50.0)&&(StringEQ1 != "Triangle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[6] (X <= 80.0)&&(StringEQ2 == "Regtangle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 50)&&(W < 60.0)

[7] (X <= 80.0)&&(StringEQ2 == "Regtangle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[8] (X <= 80.0)&&(StringEQ2 == "Regtangle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[9] (X <= 80.0)&&(StringEQ2 != "Regtangle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 50)&&(W < 60.0)

[10] (X <= 80.0)&&(StringEQ2 != "Regtangle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[11] (X <= 80.0)&&(StringEQ2 != "Regtangle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[12] (X <= 120.0)&&(StringEQ3 == "Circle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 50)&&(W < 60.0)

[13] (X <= 120.0)&&(StringEQ3 == "Circle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[14] (X <= 120.0)&&(StringEQ3 == "Circle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[15] (X <= 120.0)&&(StringEQ3 != "Circle")&&(Y <= 40)&&(booleanEQ1 == true)&&(Z <

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

50)&&(W < 60.0)

[16] (X <= 120.0)&&(StringEQ3 != "Circle")&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[17] (X <= 120.0)&&(StringEQ3 != "Circle")&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)

[18] (X > 50.0)&&(X > 80.0)&&(X > 120.0)&&(Y <= 40)&&(booleanEQ1 == true)&&(Z < 50)&&(W < 60.0)

[19] (X > 50.0)&&(X > 80.0)&&(X > 120.0)&&(Y <= 70)&&(booleanEQ2 == false)&&(Z < 50)&&(W < 60.0)

[20] (X > 50.0)&&(X > 80.0)&&(X > 120.0)&&(Y > 40)&&(Y > 70)&&(Z < 50)&&(W < 60.0)
",,,,

Post condition of success scenario,"[0] System show B1,D1,E1,E2",,,,

Post condition of alternative scenario,"[1] System show B1,D2,E1,E2

[2] System show B1,D3,E1,E2

[3] System show B2,D21,E1,E2

[4] System show B2,D2,E1,E2

[5] System show B2,D3,E1,E2

[6] System show C1,D1,E1,E2

[7] System show C1,D2,E1,E2

[8] System show C1,D3,E1,E2

[9] System show C2,D1,E1,E2

[10] System show C2,D2,E1,E2

[11] System show C2,D3,E1,E2

[12] System show C3,D1,E1,E2

[13] System show C3,D2,E1,E2

[14] System show C3,D3,E1,E2

[15] System show C4,D1,E1,E2

[16] System show C4,D2,E1,E2

[17] System show C4,D3,E1,E2

[18] System show C5,D1,E1,E2

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

[19] System show C5,D2,E1,E2

[20] System show C5,D3,E1,E2",,,,

input data,,,,,

Name,Type,Size ,Min ,Max,Scenario no.

X,float,-,0,200,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16][17][18][19][20]

StringEQ1,String,12,-,-,[0][1][2][3][4][5]

StringEQ2,String,12,-,-,[6][7][8][9][10][11]

StringEQ3,String,12,-,-,[12][13][14][15][16][17]

Y,int,-,0,150,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16][17][18][19][20]

booleanEQ1,boolean,-,-,-,[0][3][6][9][12][15][18]

booleanEQ2,boolean,-,-,-,[1][4][7][10][13][16][19]

Z,int,-,0,100,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16][17][18][19][20]

W,float,-,0.0,120.0,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15][16][17][18][19][20]

5.3) กรณีทดสอบ

Test Case

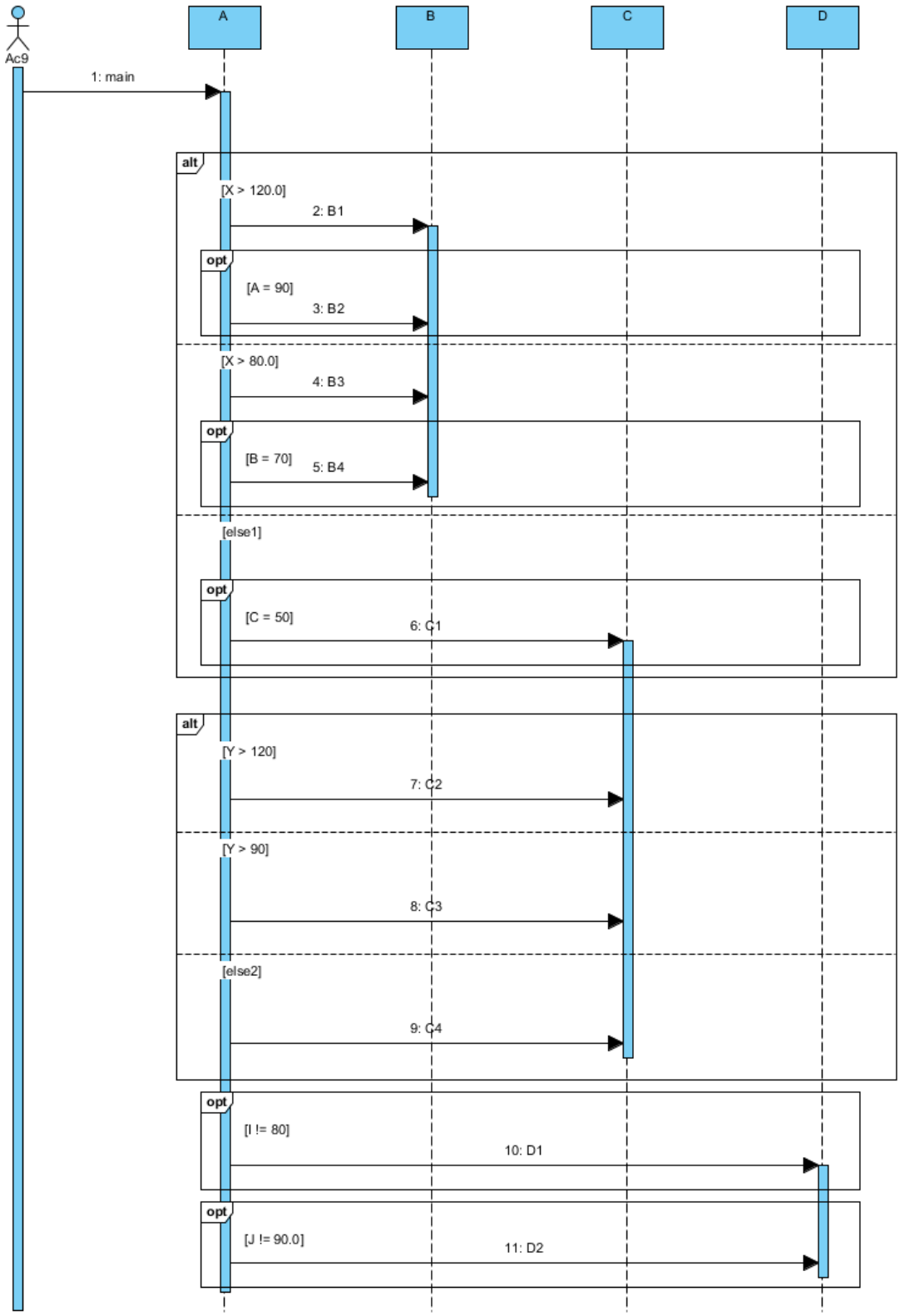
Test case ID	**2**
Test case name	**TestMockup2**
Test case description	**(X > 120.0)&&(Y > 70)&&(Z > 20)&&(W > 30.0)**
Initial Value	**Test Simulation**
Input Value	
X	**121**
Y	**71**
Z	**21**
W	**31.0**
Expected Output	**System show C5,D3**

ภาพที่ ข-14 กรณีทดสอบของระบบจำลองที่ 2

6. ระบบจำลองที่ 3

การทดสอบด้วยระบบจำลองที่ 3 สามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

6.1) แผนภาพลำดับ



ภาพที่ ข-15 แผนภาพลำดับของระบบจำลองที่ 3

6.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,TestMockup3,,,,

Pre-condition,Test simulation ,,,,

Success scenario condition,[0] (X > 120.0)&&(Y > 120)&&(I != 80)&&(J != 90.0),,,,

Alternative scenario condition,"[1] (X > 120.0)&&(Y > 90)&&(I != 80)&&(J != 90.0)

[2] (X > 120.0)&&(Y <= 120)&&(Y <= 90)&&(I != 80)&&(J != 90.0)

[3] (X > 120.0)&&(A == 90)&&(Y > 120)&&(I != 80)&&(J != 90.0)

[4] (X > 120.0)&&(A == 90)&&(Y > 90)&&(I != 80)&&(J != 90.0)

[5] (X > 120.0)&&(A == 90)&&(Y <= 120)&&(Y <= 90)&&(I != 80)&&(J != 90.0)

[6] (X > 80.0)&&(Y > 120)&&(I != 80)&&(J != 90.0)

[7] (X > 80.0)&&(Y > 90)&&(I != 80)&&(J != 90.0)

[8] (X > 80.0)&&(Y <= 120)&&(Y <= 90)&&(I != 80)&&(J != 90.0)

[9] (X > 80.0)&&(B == 70)&&(Y > 120)&&(I != 80)&&(J != 90.0)

[10] (X > 80.0)&&(B == 70)&&(Y > 90)&&(I != 80)&&(J != 90.0)

[11] (X > 80.0)&&(B == 70)&&(Y <= 120)&&(Y <= 90)&&(I != 80)&&(J != 90.0)

[12] (X <= 120.0)&&(X <= 80.0)&&(C == 50)&&(Y > 120)&&(I != 80)&&(J != 90.0)

[13] (X <= 120.0)&&(X <= 80.0)&&(C == 50)&&(Y > 90)&&(I != 80)&&(J != 90.0)

[14] (X <= 120.0)&&(X <= 80.0)&&(C == 50)&&(Y <= 120)&&(Y <= 90)&&(I != 80)&&(J != 90.0)",,,,

Post condition of success scenario,"[0] System show B1,C2,D1,D2",,,,

Post condition of alternative scenario,"[1] System show B1,C3,D1,D2

[2] System show B1,C4,D1,D2

[3] System show B2,C2,D1,D2

[4] System show B2,C3,D1,D2

[5] System show B2,C4,D1,D2

[6] System show B3,C2,D1,D2

[7] System show B3,C3,D1,D2

[8] System show B3,C4,D1,D2

[9] System show B4,C2,D1,D2

[10] System show B4,C3,D1,D2

[11] System show B4,C4,D1,D2

[12] System show C1,C2,D1,D2

[13] System show C1,C3,D1,D2

[14] System show C1,C4,D1,D2",,,,

input data,,,,,

Name,Type,Size ,Min ,Max,Scenario no.

X,float,-,0,150.0,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14]

Y,int,-,0,150,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14]

I,int,-,0,100,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14]

J,float,-,0,100.0,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14]

A,int,-,0,100,[3][4][5]

B,int,-,0,100,[9][10][11]

C,int,-,10,90,[12][13][14]

6.3) กรณีทดสอบ

Test Case

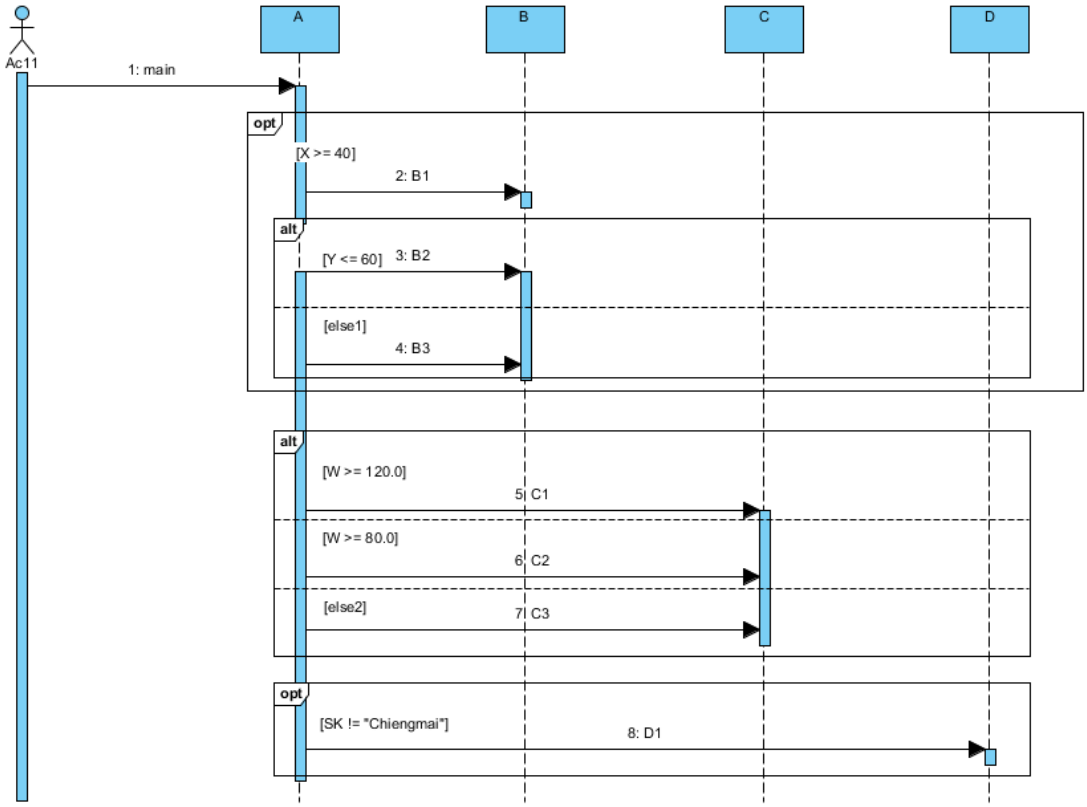
Test case ID	**1**
Test case name	**TestMockup3**
Test case description	**(X < 80.0)&&(C == 50)&&(Y < 90)&&(I == 80)&&(J == 90.0)**
Initial Value	**Test simulation**
Input Value	
X	**78.05**
Y	**89**
I	**80**
J	**90.0**
C	**50**
Expected Output	**System show C4,C1**

ภาพที่ ข-16 กรณีทดสอบของระบบจำลองที่ 3

7. ระบบจำลองที่ 4

การทดสอบด้วยระบบจำลองที่ 4 สามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

7.1) แผนภาพลำดับ



ภาพที่ ข-17 แผนภาพลำดับของระบบจำลองที่ 4

7.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,TestMockup4,,,,
Pre-condition,Test Simulation ,,,,
Success scenario condition,[0] (X >= 40)&&(W >= 120.0)&&(SK != "Chiengmai"),,,,
Alternative scenario condition,"[1] (X >= 40)&&(W >= 80.0)&&(SK != "Chiengmai")
[2] (X >= 40)&&(W < 120.0)&&(W < 80.0)&&(SK != "Chiengmai")
[3] (X >= 40)&&(Y <= 60)&&(W >= 120.0)&&(SK != "Chiengmai")
[4] (X >= 40)&&(Y <= 60)&&(W >= 80.0)&&(SK != "Chiengmai")
[5] (X >= 40)&&(Y <= 60)&&(W < 120.0)&&(W < 80.0)&&(SK != "Chiengmai")

[6] (X >= 40)&&(Y > 60)&&(W >= 120.0)&&(SK != "Chiengmai")

[7] (X >= 40)&&(Y > 60)&&(W >= 80.0)&&(SK != "Chiengmai")

[8] (X >= 40)&&(Y > 60)&&(W < 120.0)&&(W < 80.0)&&(SK != "Chiengmai")",,,,

Post condition of success scenario,"[0] System show B1,C1,D1",,,,

Post condition of alternative scenario,"[1] System show B1,C2,D1

[2] System show B1,C3,D1

[3] System show B2,C1,D1

[4] System show B2,C2,D1

[5] System show B2,C3,D1

[6] System show B3,C1,D1

[7] System show B3,C2,D1

[8] System show B3,C3,D1",,,,

input data,,,,,

Name,Type,Size ,Min ,Max,Scenario no.

X,int,-,0,100,[0][1][2][3][4][5][6][7][8]

W,float,-,0.0,120.0,[0][1][2][3][4][5][6][7][8]

SK,String,10,-,-,[0][1][2][3][4][5][6][7][8]

Y,int,-,20,120,[3][4][5][6][7][8]

7.3) กรณีทดสอบ

Test Case

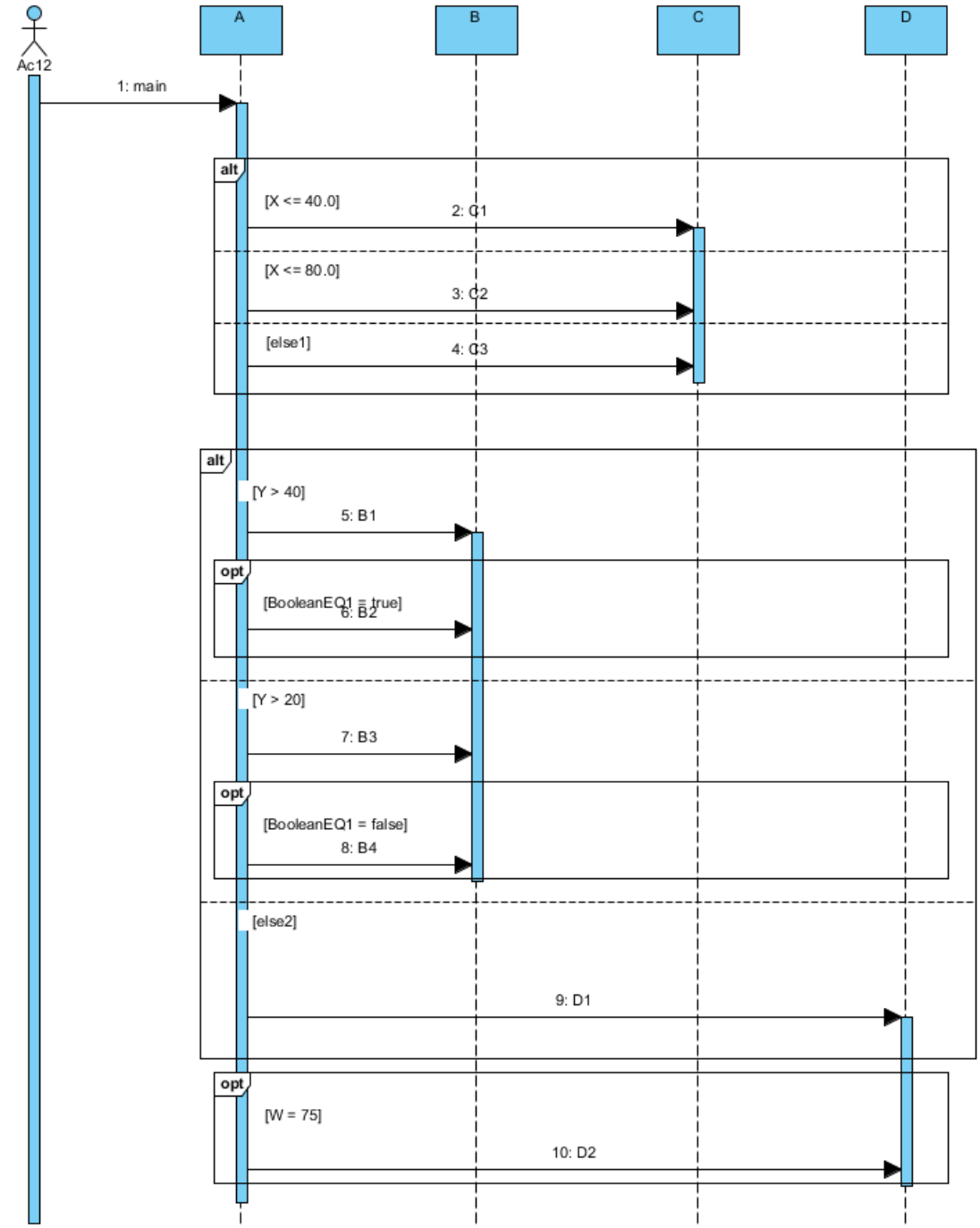
Test case ID	**1**
Test case name	**TestMockup4**
Test case description	**(X < 40)&&(W < 80.0)&&(SK == "Chiengmai")**
Initial Value	**Test Simulation**
Input Value	
X	**40**
W	**80.0**
SK	**"Chiengmai"**
Expected Output	**System show C3**

ภาพที่ ข-18 กรณีทดสอบของระบบจำลองที่ 4

8. ระบบจำลองที่ 5

การทดสอบด้วยระบบจำลองที่ ถ สามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

8.1) แผนภาพลำดับ



ภาพที่ ข-19 แผนภาพลำดับของระบบจำลองที่ 5

8.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,TestMockup5,,,,

Pre-condition,Cashier reserve the hotel room ,,,,

Success scenario condition,[0] (X <= 40.0)&&(Y > 40)&&(W == 75),,,,

Alternative scenario condition,"[1] (X <= 40.0)&&(Y > 40)&&(BooleanEQ1 == true)&&(W == 75)

[2] (X <= 40.0)&&(Y > 20)&&(W == 75)

[3] (X <= 40.0)&&(Y > 20)&&(BooleanEQ1 == false)&&(W == 75)

[4] (X <= 40.0)&&(Y <= 40)&&(Y <= 20)&&(W == 75)

[5] (X <= 80.0)&&(Y > 40)&&(W == 75)

[6] (X <= 80.0)&&(Y > 40)&&(BooleanEQ1 == true)&&(W == 75)

[7] (X <= 80.0)&&(Y > 20)&&(W == 75)

[8] (X <= 80.0)&&(Y > 20)&&(BooleanEQ1 == false)&&(W == 75)

[9] (X <= 80.0)&&(Y <= 40)&&(Y <= 20)&&(W == 75)

[10] (X > 40.0)&&(X > 80.0)&&(Y > 40)&&(W == 75)

[11] (X > 40.0)&&(X > 80.0)&&(Y > 40)&&(BooleanEQ1 == true)&&(W == 75)

[12] (X > 40.0)&&(X > 80.0)&&(Y > 20)&&(W == 75)

[13] (X > 40.0)&&(X > 80.0)&&(Y > 20)&&(BooleanEQ1 == false)&&(W == 75)

[14] (X > 40.0)&&(X > 80.0)&&(Y <= 40)&&(Y <= 20)&&(W == 75)",,,,

Post condition of success scenario,"[0] System show C1,B1,D2,,,,

Post condition of alternative scenario,"[1] System show C1,B2,D2

[2] System show C1,B3,D2

[3] System show C1,B4,D2

[4] System show B1,D1,D2

[5] System show C2,B1,D2

[6] System show C2,B2,D2

[7] System show C2,B3,D2

[8] System show C2,B4,D2

[9] System show C2,D1,D2

[10] System show C3,B1,D2

[11] System show C3,B2,D2

[12] System show C3,B3,D2

[13] System show C3,B4,D2

[14] System show C3,D1,D2",,,,

input data,,,,,

Name,Type,Size ,Min ,Max,Scenario no.

X,float,-,0,150.0,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14]

Y,int,-,0,100,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14]

W,int,-,0,100,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14]

BooleanEQ1,boolean,-,-,-,[1][3][6][8][11][13]

8.3) กรณีทดสอบ

Test Case

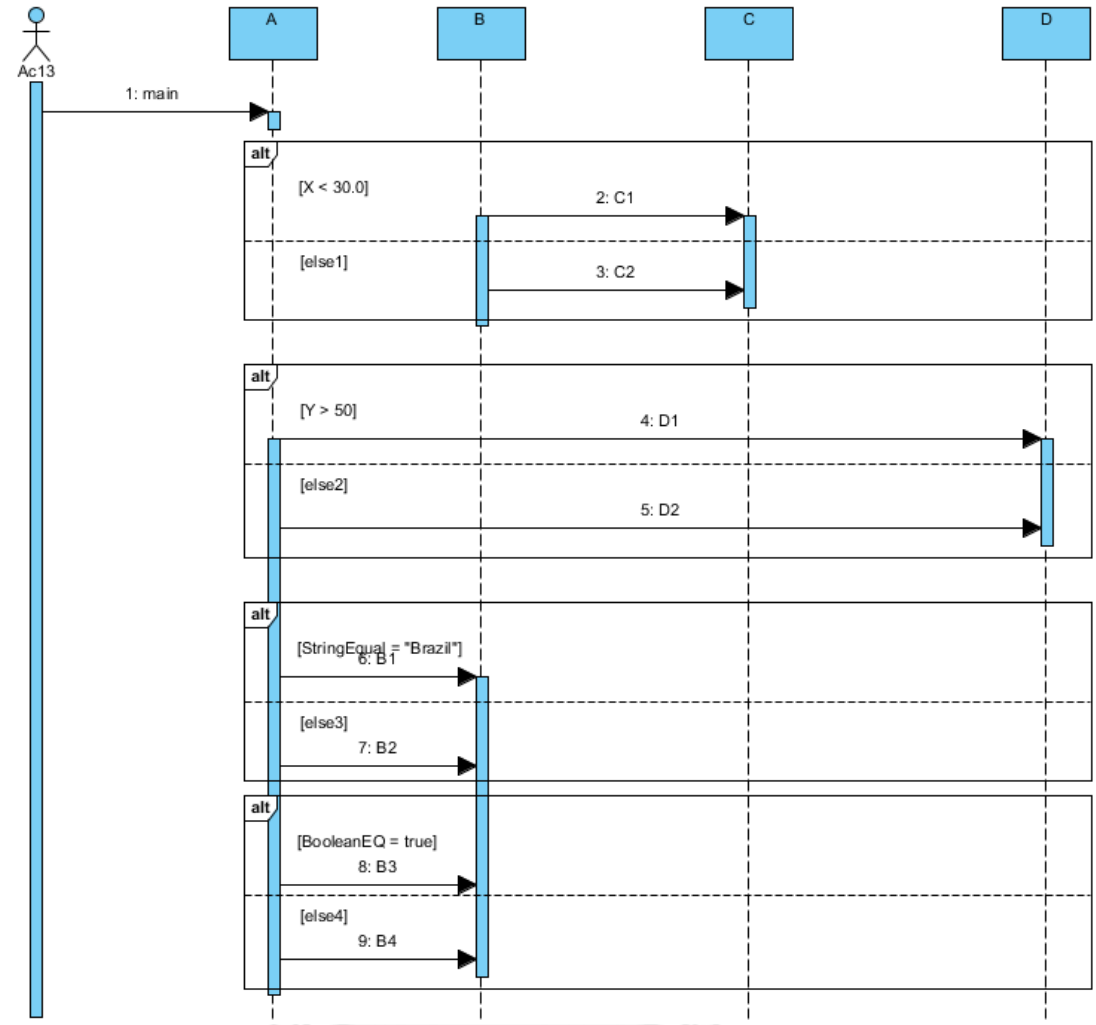
Test case ID	**1**
Test case name	**TestMockup5**
Test case description	**(X > 80.0)&&(Y < 20)&&(W != 75)**
Initial Value	**Cashier reserve the hotel room**
Input Value	
X	**100**
Y	**19**
W	**71**
Expected Output	**System show C3,D1**

ภาพที่ ข-20 กรณีทดสอบของระบบจำลองที่ 5

9. ระบบจำลองที่ 6

การทดสอบด้วยระบบจำลองที่ 6 สามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

9.1) แผนภาพลำดับ



ภาพที่ ข-21 แผนภาพลำดับของระบบจำลองที่ 6

9.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,TestMockup6,,,,
Pre-condition,Test Simulation ,,,,
Success scenario condition,[0] (X < 30.0)&&(Y > 50)&&(StringEqual == "Brazil")&&(BooleanEQ == true),,,,
Alternative scenario condition,"[1] (X < 30.0)&&(Y > 50)&&(StringEqual == "Brazil")&&(BooleanEQ == false)
[2] (X < 30.0)&&(Y > 50)&&(StringEqual != "Brazil")&&&&(BooleanEQ == true)

[3] (X < 30.0)&&(Y > 50)&&(StringEqual != "Brazil")&&(BooleanEQ == false)

[4] (X < 30.0)&&(Y <= 50)&&(StringEqual == "Brazil")&&(BooleanEQ == true)

[5] (X < 30.0)&&(Y <= 50)&&(StringEqual == "Brazil")&&(BooleanEQ == false)

[6] (X < 30.0)&&(Y <= 50)&&(StringEqual != "Brazil")&&&&(BooleanEQ == true)

[7] (X < 30.0)&&(Y <= 50)&&(StringEqual != "Brazil")&&(BooleanEQ == false)

[8] (X >= 30.0)&&(Y > 50)&&(StringEqual == "Brazil")&&(BooleanEQ == true)

[9] (X >= 30.0)&&(Y > 50)&&(StringEqual == "Brazil")&&(BooleanEQ == false)

[10] (X >= 30.0)&&(Y > 50)&&(StringEqual != "Brazil")&&&&(BooleanEQ == true)

[11] (X >= 30.0)&&(Y > 50)&&(StringEqual != "Brazil")&&(BooleanEQ == false)

[12] (X >= 30.0)&&(Y <= 50)&&(StringEqual == "Brazil")&&(BooleanEQ == true)

[13] (X >= 30.0)&&(Y <= 50)&&(StringEqual == "Brazil")&&(BooleanEQ == false)

[14] (X >= 30.0)&&(Y <= 50)&&(StringEqual != "Brazil")&&&&(BooleanEQ == true)

[15] ((X >= 30.0)&&(Y <= 50)&&(StringEqual != "Brazil")&&(BooleanEQ == false)",,,,

Post condition of success scenario,"[0] System show C1,D1,B1,B3,,,,

Post condition of alternative scenario,"[1] System show total C1,D1,B1,B4

[2] System show C1,D1,B2,B3

[3] System show C1,D1,B2,B4

[4] System show C1,D2,B1,B3

[5] System show C1,D2,B1,B4

[6] System show C1,D2,B2,B3

[7] System show C1,D2,B2,B4

[8] System show C2,D1,B1,B3

[9] System show C2,D1,B1,B4

[10] System show C2,D1,B2,B3

[11] System show C2,D1,B2,B4

[12] System show C2,D2,B1,B3

[13] System show C2,D2,B1,B4

[14] System show C2,D2,B2,B3

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

[15] System show C2,D2,B2,B4",,,,

input data,,,,,

Name,Type,Size ,Min ,Max,Scenario no.

X,float,-,0.0,100.0,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15]

Y,int,-,0,150,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15]

StringEqual,String,10,-,-,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15]

BooleanEQ,boolean,-,-,-,[0][1][2][3][4][5][6][7][8][9][10][11][12][13][14][15]

9.3) กรณีทดสอบ

Test Case

Test case ID	**1**
Test case name	**TestMockup6**
Test case description	**(X < 30.0)&&(Y < 50)&&(StringEqual != "Brazil")&&(BooleanEQ == false)**
Initial Value	**Test Simulation**
Input Value	
X	**29.00**
Y	**33**
StringEqual	**"Br"**
BooleanEQ	**false**
Expected Output	**System show C2,D2,B2,B4**

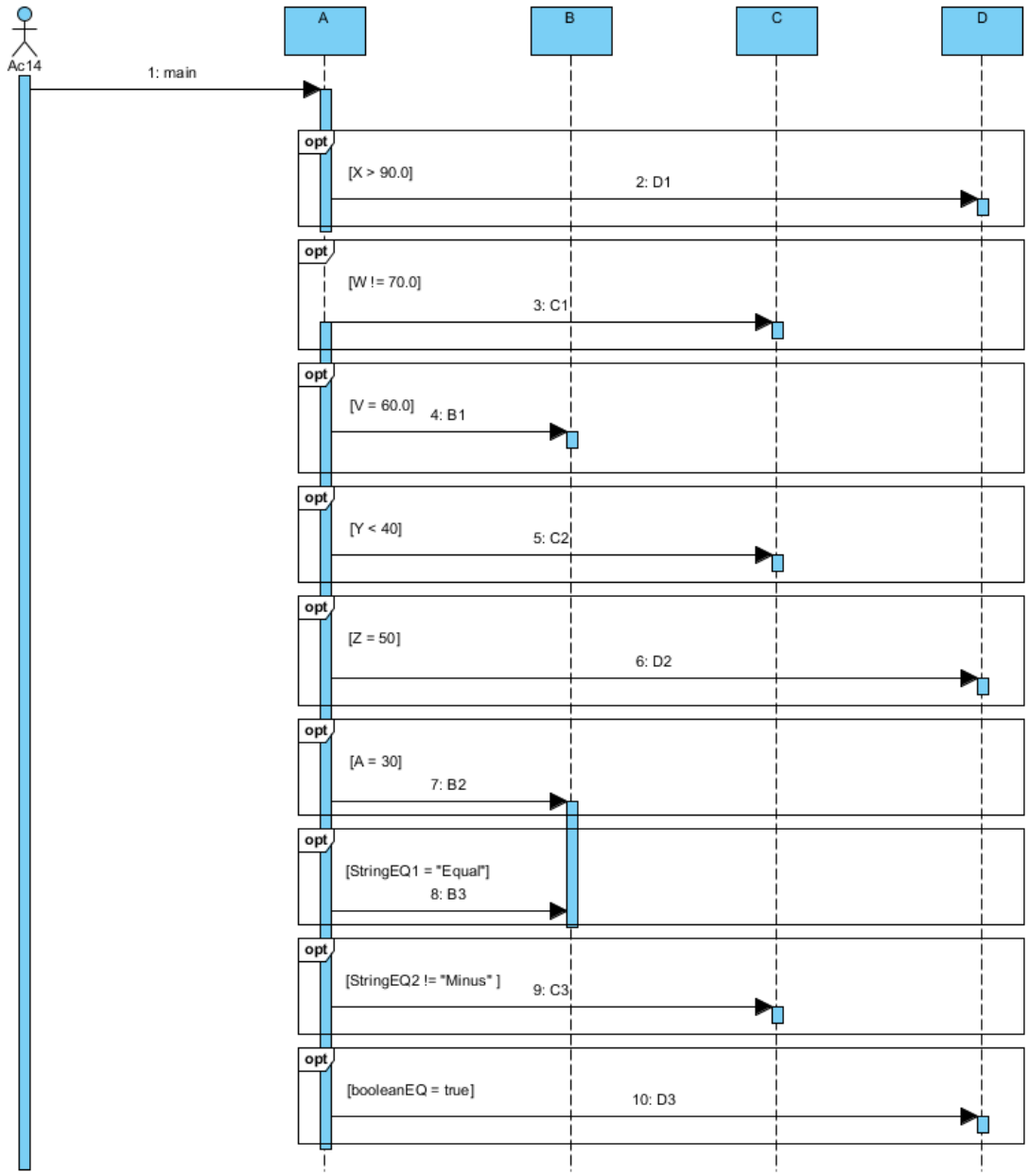
ภาพที่ ข-22 กรณีทดสอบของระบบจำลองที่ 6

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

10. ระบบจำลองที่ 7

การทดสอบด้วยระบบจำลองที่ 7 สามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

10.1) แผนภาพลำดับ



ภาพที่ ข-23 แผนภาพลำดับของระบบจำลองที่ 7

10.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,TestMockup7,,,,
Pre-condition,Test Simulation,,,,
Success scenario condition,[0] (X > 90.0)&&(BooleanEQ == true),,,,
Alternative scenario condition,"[1] (W != 70.0)&&(StringEQ2 != "Minus")

[2] (V == 60.0)&&(A == 30)

[3] (Y < 40)&&(StringEQ == "Equal")&&(Z == 50)",,,,

Post condition of success scenario,"[0] System show D1,D3,,,,

Post condition of alternative scenario,"[1] System show C1,C3

[2] System show B1,B2

[3] System show C2,B3,D2",,,,

input data,,,,,

Name,Type,Size ,Min ,Max,Scenario no.

X,float,-,50,200,[0]

V,float,-,50,70,[2]

A,int,-,20,50,[2]

W,float,-,30,100,[1]

StringEQ2,String,10,-,-,[1]

Y,int,-,10,90,[3]

BooleanEQ,boolean,-,-,-,[0]

StringEQ,String,10,-,-,[3]

Z,int,-,0,100,[3]

10.3) กรณีทดสอบ

Test Case

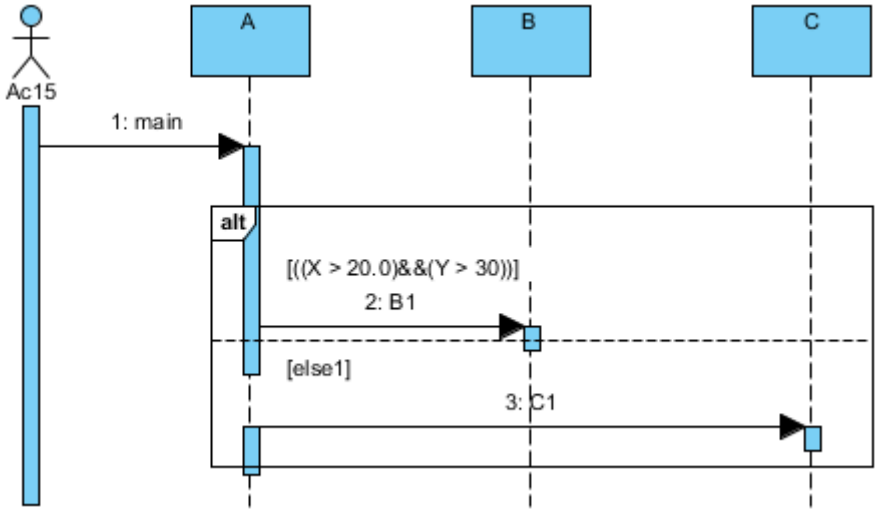
Test case ID	**1**
Test case name	**TestMockup7**
Test case description	**(X < 90.0)&&(W == 70.0)&&(V == 60.0)&&(A != 30)&&(Y < 40)&&(StringEQ1 != "Equal")&&(StringEQ2 == "Minus")&&(BooleanEQ == "true")&&(Z != 50)**
Initial Value	**Test Simulation**
Input Value	
X	**39**
Y	**39**
StringEQ	**"Eq"**
StringEQ2	**"Minus"**
BooleanEQ	**true**
A	**50**
V	**60.0**
W	**70**
Z	**50**
Expected Output	**System show**

ภาพที่ ข-24 กรณีทดสอบของระบบจำลองที่ 7

11. ระบบจำลองที่ 8

การทดสอบด้วยระบบจำลองที่ 3 สามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

11.1) แผนภาพลำดับ



ภาพที่ ข-25 แผนภาพลำดับของระบบจำลองที่ 8

11.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,TestMockup8,,,,
Pre-condition,Test Simulation,,,,
Success scenario condition,[0] (((X > 20.0)&&(Y > 30))),,,,
Alternative scenario condition,[1] (((X <= 20.0)&&(Y <= 30))),,,,
Post condition of success scenario,"[0] System show B1,,,,
Post condition of alternative scenario,"[1] System show C1",,,,
input data,,,,,
Name,Type,Size ,Min ,Max,Scenario no.
X,float,-,0.0,50.0,[0][1]
Y,int,-,15,45,[0][1]

11.3) กรณีทดสอบ

Test Case

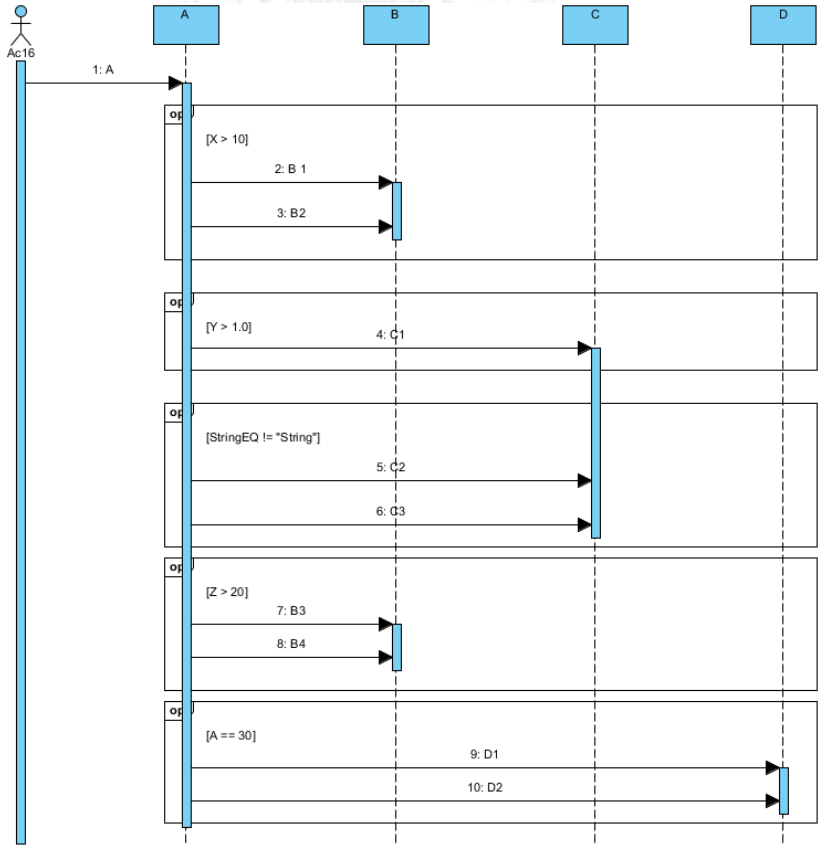
Test case ID	**1**
Test case name	**TestMockup8**
Test case description	**(((X <= 20.0)&&(Y <= 30)))**
Initial Value	**Test Simulation**
Input Value	
X	**7.5652003**
Y	**25**
Expected Output	**System show C1**

ภาพที่ ข-26 กรณีทดสอบของระบบจำลองที่ 8

12. ระบบจำลองที่ 9

การทดสอบด้วยระบบจำลองที่ 9 สามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

12.1) แผนภาพลำดับ



ภาพที่ ข-27 แผนภาพลำดับของระบบจำลองที่ 9

12.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

```
Sequence diagram number,TestMockup9,,,,
Pre-condition,Test Simulation,,,,
Success scenario condition,[0] (X > 10),,,,
Alternative scenario condition,"[1] (Y > 1.0)
[2] (Z > 20)&&(A == 30)
[3] (StringEQ != "String"),,,,
Post condition of success scenario,"[0] System show B1,B2,,,,
Post condition of alternative scenario,"[1] System show C1
[2] System show B3,B4,D1,D2
[3] System show C2,C3",,,,
input data,,,,,
Name,Type,Size ,Min ,Max,Scenario no.
X,int,-,0,20,[0]
Y,float,-,0.0,10.0,[1]
A,int,-,20,50,[2]
Z,int,-,0,100,[2]
StringEQ,String,10,-,-,[3]
```

12.3) กรณีทดสอบ

Test Case

Test case ID	**1**
Test case name	**TestMockup9**
Test case description	**(X < 10)&&(Y < 1.0)&&(StringEQ == "String")&&(Z < 20)&&(A != 30)**
Initial Value	**Test Simulation**
Input Value	
A	**20**
Z	**18**
X	**2**
Y	**0.0**
StringEQ	**"String"**
Expected Output	**System show**

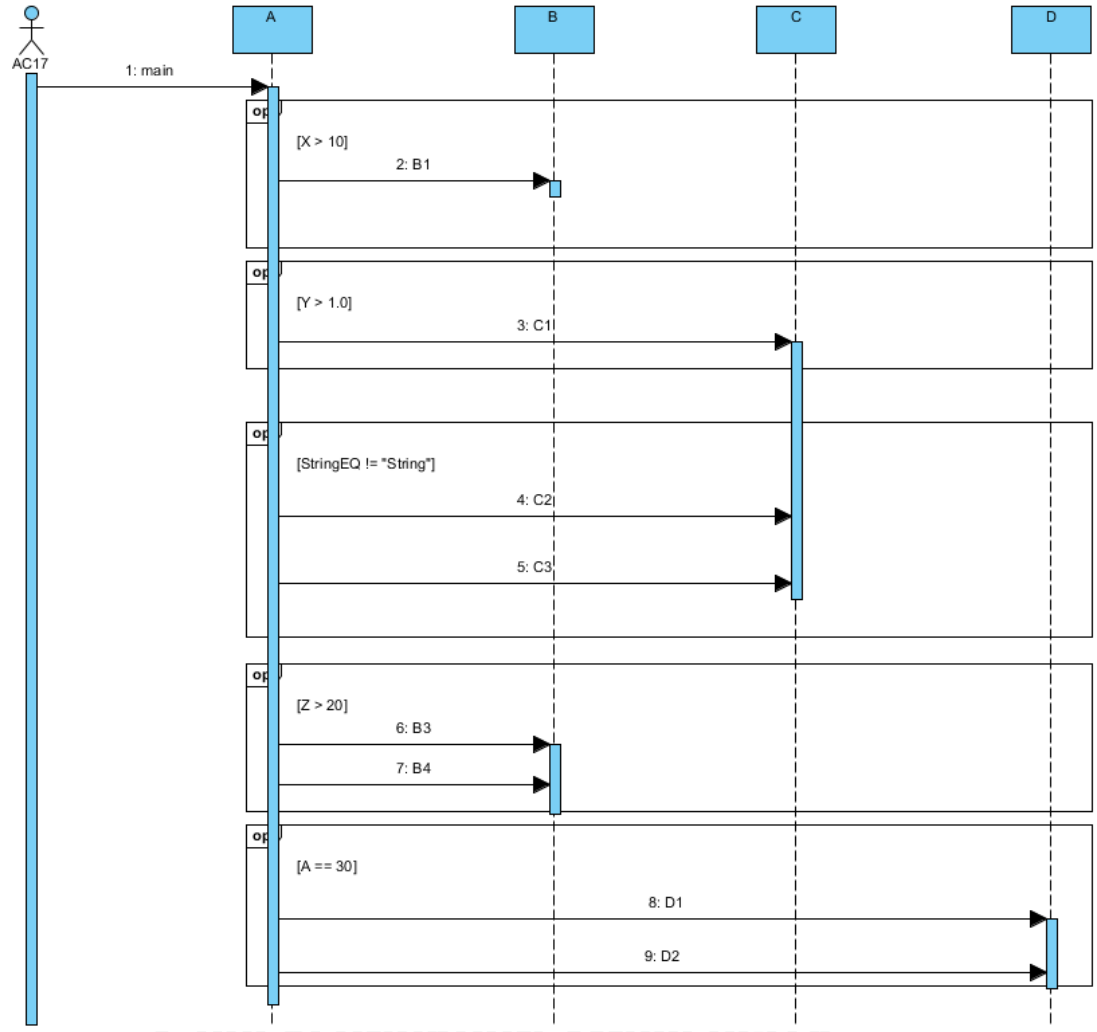
ภาพที่ ข-28 กรณีทดสอบของระบบจำลองที่ 9

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

13. ระบบจำลองที่ 10

การทดสอบด้วยระบบจำลองที่ 10 สามารถแสดงด้วย แผนภาพลำดับ คำอธิบายยูสเคส และกรณีทดสอบดังนี้

13.1) แผนภาพลำดับ



CHULALONGKORN UNIVERSITY

ภาพที่ ข-29 แผนภาพลำดับของระบบจำลองที่ 10

13.2) คำอธิบายยูสเคสในรูปแบบไฟล์ซีเอสวี

Sequence diagram number,TestMockup10,,,,
Pre-condition,Test Simulation,,,,
Success scenario condition,[0] (X > 10),,,,
Alternative scenario condition,"[1] (Y > 1.0)

[2] (Z > 20)&&(A == 30)

[3] (StringEQ != "String"),,,,

Post condition of success scenario,"[0] System show B1,,,,

Post condition of alternative scenario,"[1] System show C1

[2] System show B3,B4,D1,D2

[3] System show C2,C3",,,,

input data,,,,,

Name,Type,Size ,Min ,Max,Scenario no.

X,int,-,0,20,[0]

Y,float,-,0.0,10.0,[1]

A,int,-,20,50,[2]

Z,int,-,0,100,[2]

StringEQ,String,10,-,-,[3]

13.3) กรณีทดสอบ

Test Case

Test case ID	**1**
Test case name	**TestMockup10**
Test case description	**(X < 10)&&(Y < 1.0)&&(StringEQ == "String")&&(Z < 20)&&(A != 30)**
Initial Value	**Test Simulation**
Input Value	
A	**20**
Z	**18**
X	**2**
Y	**0.0**
StringEQ	**"String"**
Expected Output	**System show**

ภาพที่ ข-30 กรณีทดสอบของระบบจำลองที่ 10

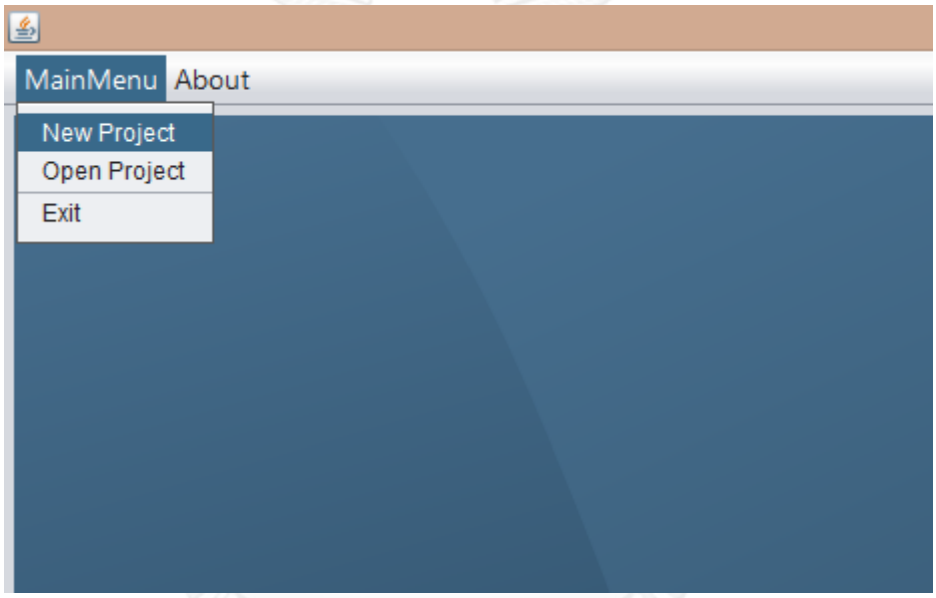
ภาคผนวก ค

วิธีการใช้เครื่องมือตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการของซอฟต์แวร์เชิงวัตถุ

เครื่องมือการตรวจสอบความครอบคุลมการทดสอบแบบบูรณาการสามารถแบ่งได้เป็น 2 ส่วน คือ ส่วนการตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการและส่วนการสร้างกรณีทดสอบ ซึ่งมีรายละเอียดดังต่อไปนี้

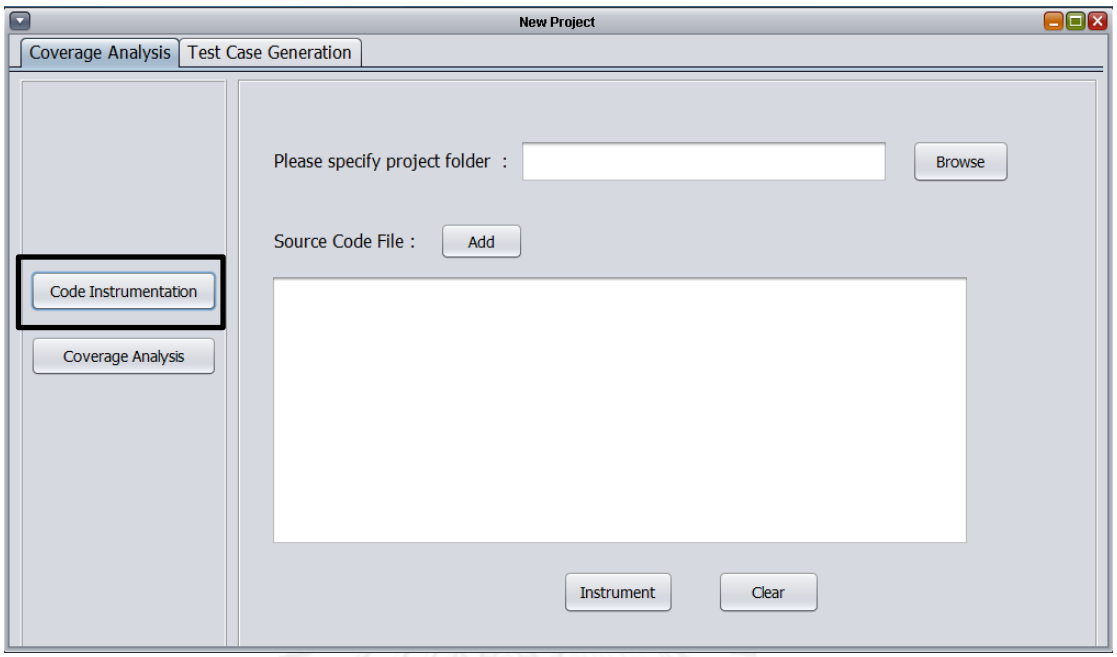
1. ส่วนของการตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการ

1.1 เลือกเมนู MainMenu และเลือก New Project ดังภาพ ค-1 เพื่อสร้างโปรเจคใหม่



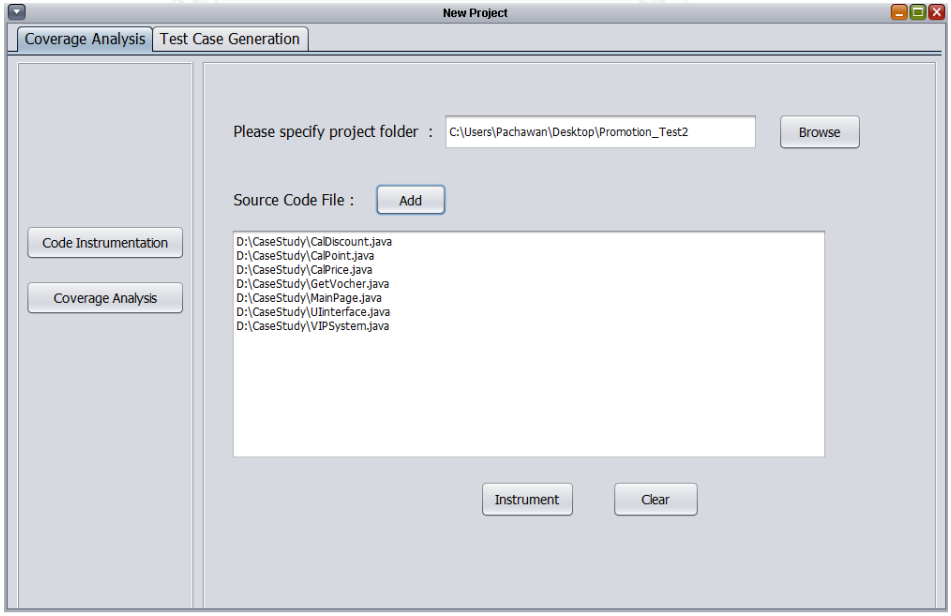
ภาพที่ ค-1 หน้าต่างการสร้างโปรเจคใหม่

1.2 กดปุ่ม Code instrumentation ระบบจะแสดงหน้าต่างให้ผู้ใช้กำหนดโฟลเดอร์ที่ต้องการให้ที่ต้องการเก็บซอร์สโค้ดที่ถูกแทรกรวมทั้งระบุซอร์สโค้ดที่ต้องการทดสอบดังภาพ ค-2



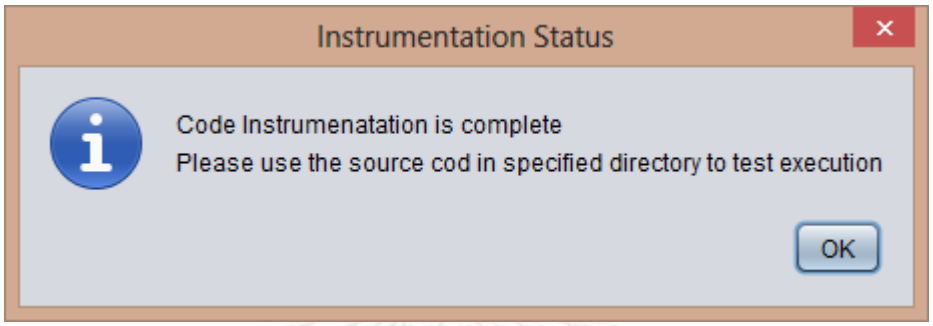
ภาพที่ ค-2 หน้าต่างการแทรกซอร์สโค้ด

1.3 กดปุ่ม Browse เพื่อระบุโฟล์เดอร์ที่ต้องการใช้เก็บซอร์สโค้ดที่แทรกแล้ว และกดปุ่ม Add เพื่อเลือกซอร์สโค้ดที่ต้องการแทรก หลังจากนั้นเครื่องมือจะแสดงชื่อโฟลเดอร์ที่ใช้เก็บซอร์สโค้ดที่ต้องการแทรกและชื่อซอร์สโค้ดที่ต้องการแทรกดังภาพที่ ค-3



ภาพที่ ค-3 หน้าต่างการแทรกซอร์ซโค้ดที่ระบุซอร์สโค้ดที่ต้องการแทรกแล้ว

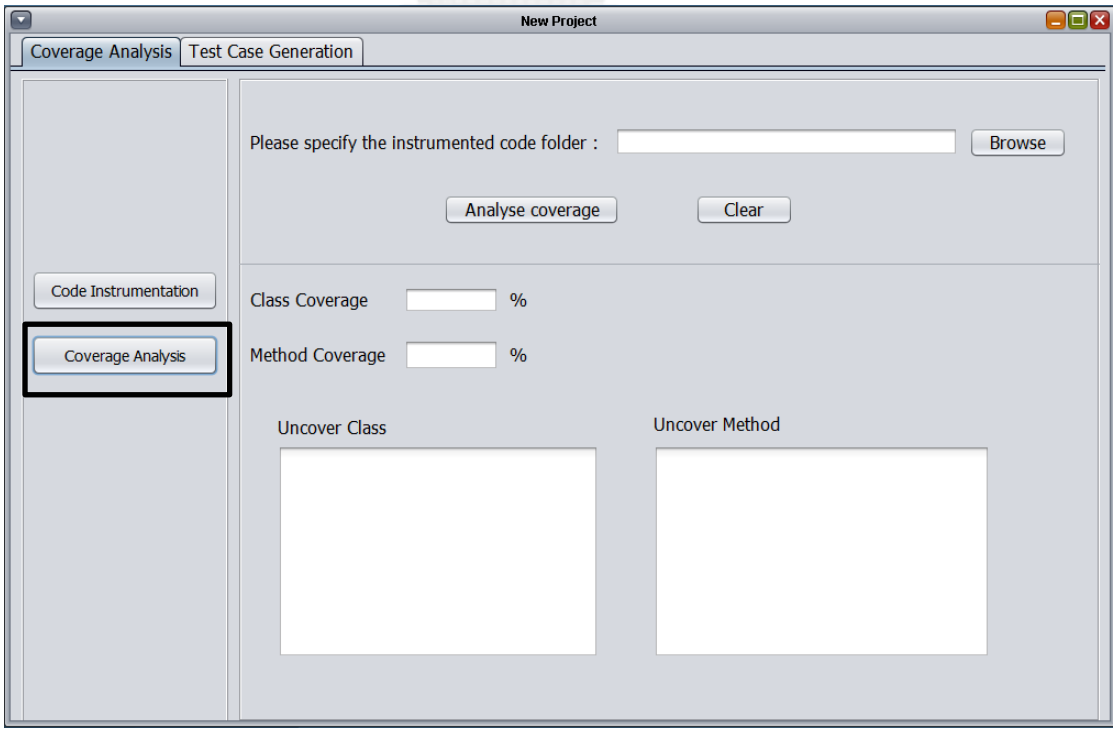
1.4 กดปุ่ม Instrument ระบบจะแทรกซอร์สโค้ดและเก็บไว้ในโฟลเดอร์ที่ระบุ เมื่อเครื่องมือแทรกซอร์สโค้ดเสร็จแล้ว เครื่องมือจะแจ้งให้ผู้ทดสอบทราบ ดังภาพที่ ค-4



ภาพที่ ค-4 หน้าต่างแจ้งสถานะการแทรกซอร์สโค้ดเมื่อเครื่องมือแทรกซอร์สโค้ดเสร็จแล้ว

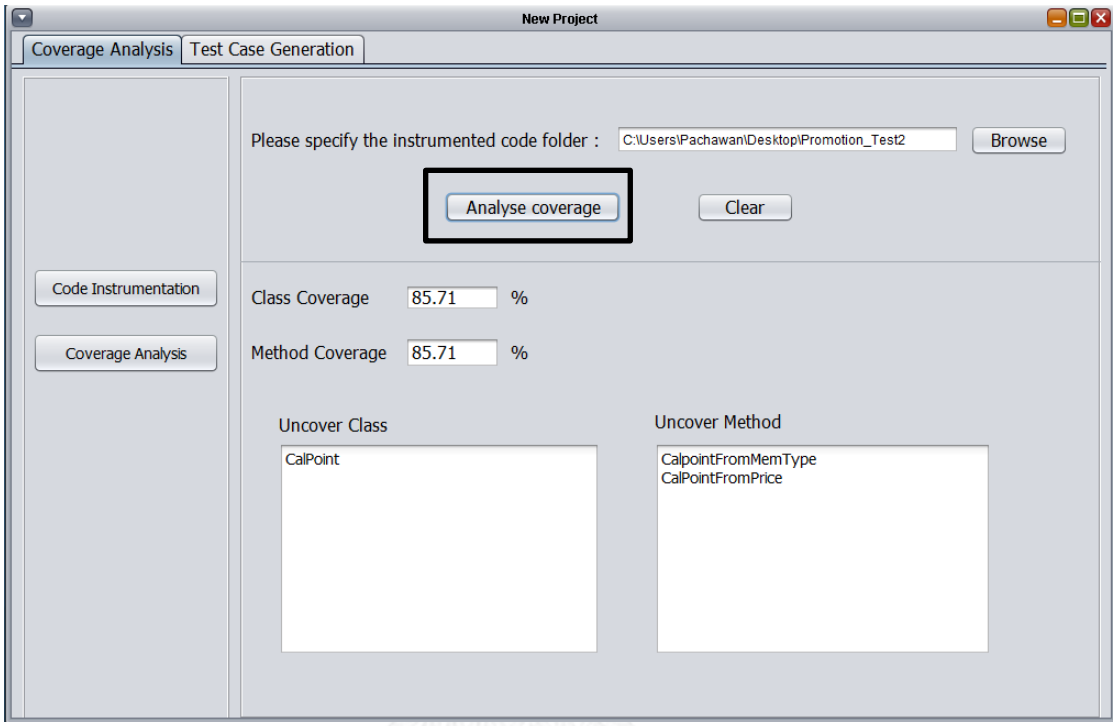
1.5 ผู้ทดสอบนำซอร์สโค้ดที่แทรกซอร์สโค้ดแล้วไปทดสอบแบบบูรณาการ

1.6 ผู้ทดสอบกดปุ่ม Coverage analysis เพื่อดูผลความครอบคลุมการทดสอบแบบบูรณาการ เครื่องมือจะแสดงหน้าต่าง Coverage analysis ดังภาพ ค-5



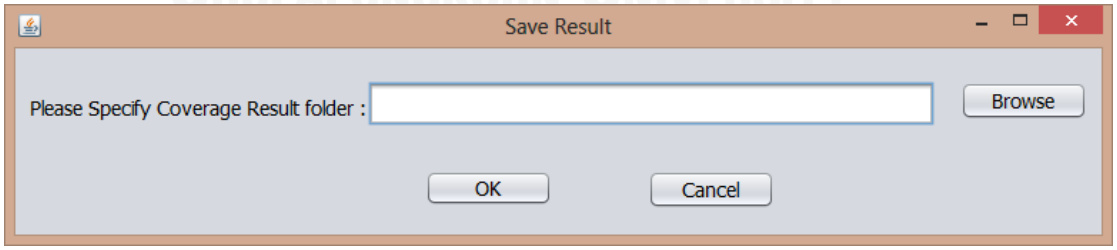
ภาพที่ ค-5 หน้าต่าง Coverage Analysis

1.7 ผู้ทดสอบเลือกโฟลเดอร์ที่ใช้เก็บซอร์สโค้ดที่ถูกแทรกและกด Analyse Coverage ระบบจะแสดงผลการตรวจสอบความครอบคลุมดังภาพ ค-6



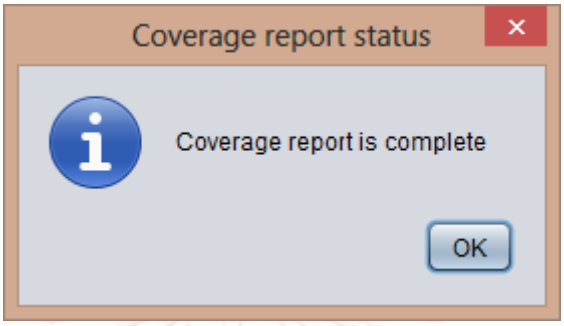
ภาพที่ ค-6 หน้าต่างแสดงผลการตรวจสอบความครอบคลุมของการทดสอบแบบบูรณาการ

1.8 เครื่องมือจะแสดงหน้าต่างให้ผู้ใช้เลือกโฟลเดอร์ที่ใช้เก็บรายงานความครอบคลุมการทดสอบดังภาพที่ ค-7



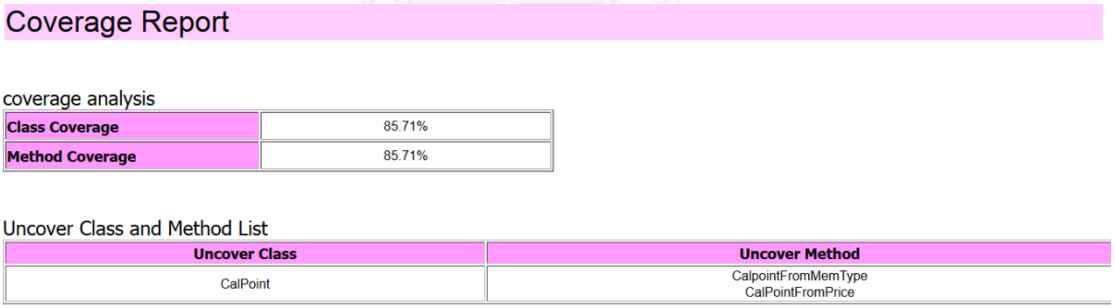
ภาพที่ ค-7 หน้าต่างบันทึกผลการตรวจสอบความครอบคลุมการมดสอบ

1.9 กด Browse และเลือกโฟลเดอร์ที่ต้องการบันทึกรายงานการตรวจสอบความครอบคลุมการทดสอบ หลังจากนั้นเครื่องมือจะแสดงหน้าต่างแจ้งสถานะเมื่อบันทึกรายงานการตรวจสอบความครอบคลุมเสร็จแล้ว ดังภาพ ค-8



ภาพที่ ค-8 หน้าต่างแจ้งสถานการณ์บันทึกผลการตรวจสอบความครอบคลุม เมื่อบันทึกผลเสร็จแล้ว

1.10 ตัวอย่างของรายงานการตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการสามารถแสดงได้ดังภาพที่ ค-9


Coverage Report

coverage analysis

Class Coverage	85.71%
Method Coverage	85.71%

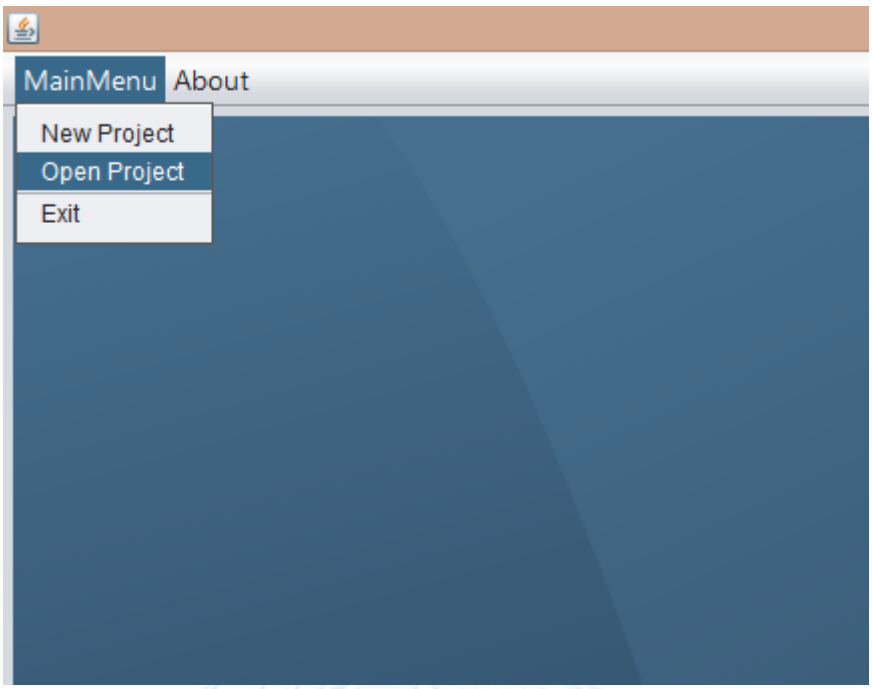
Uncover Class and Method List

Uncover Class	Uncover Method
CalPoint	CalpointFromMemType CalPointFromPrice

ภาพที่ ค-9 ตัวอย่างรายงานการตรวจสอบความครอบคลุมการทดสอบแบบบูรณาการ

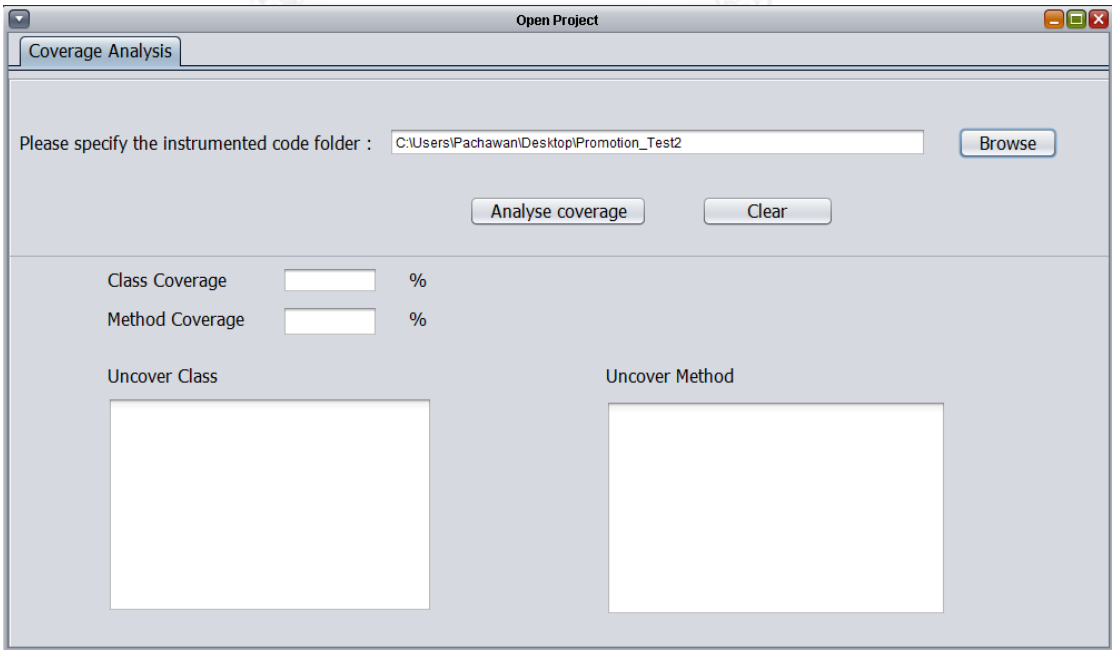
1.11 เมื่อเครื่องมือสร้างกรณีทดสอบเสร็จแล้วและผู้ทดสอบได้ใช้กรณีทดสอบที่สร้างขึ้นทดสอบซอฟต์แวร์อีกครั้งแล้ว ผู้ทดสอบสามารถตรวจสอบความครอบคลุมที่เพิ่มขึ้นดังนี้

1.11.1 ผู้ทดสอบเปิดเครื่องมือ แล้วเลือกหน้า Open Project ดังภาพที่ ค-10 เพื่อเปิดโปรเจคเดิมที่สร้างไว้



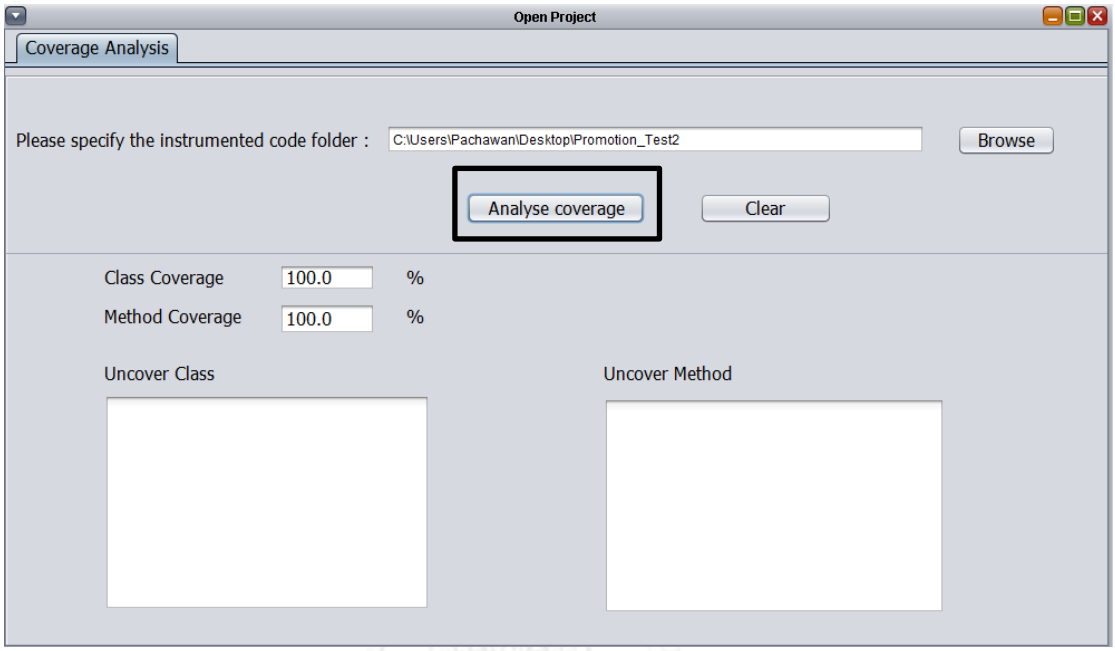
ภาพที่ ค-10 หน้าต่าง Open Project

1.11.2 เครื่องมือจะแสดงหน้าจอเพื่อให้ผู้ใช้ระบุโฟลเดอร์ที่ใช้เก็บซอร์สโค้ดที่ได้แทรกซอร์สโค้ดไว้ ดังภาพที่ ค-11



ภาพที่ ค-11 หน้าต่าง Coverage analysis

1.11.3 เมื่อผู้ใช้ระบุโฟลเดอร์ที่ใช้เก็บซอร์สโค้ดที่แทรกแล้ว และกดปุ่ม Analyse Coverage เครื่องมือจะแสดงผลการตรวจสอบความครอบคลุมหลังจากที่ได้ทดสอบเพิ่มเติมด้วยกรณีทดสอบที่เครื่องมือสร้างขึ้นดังภาพที่ ค-12

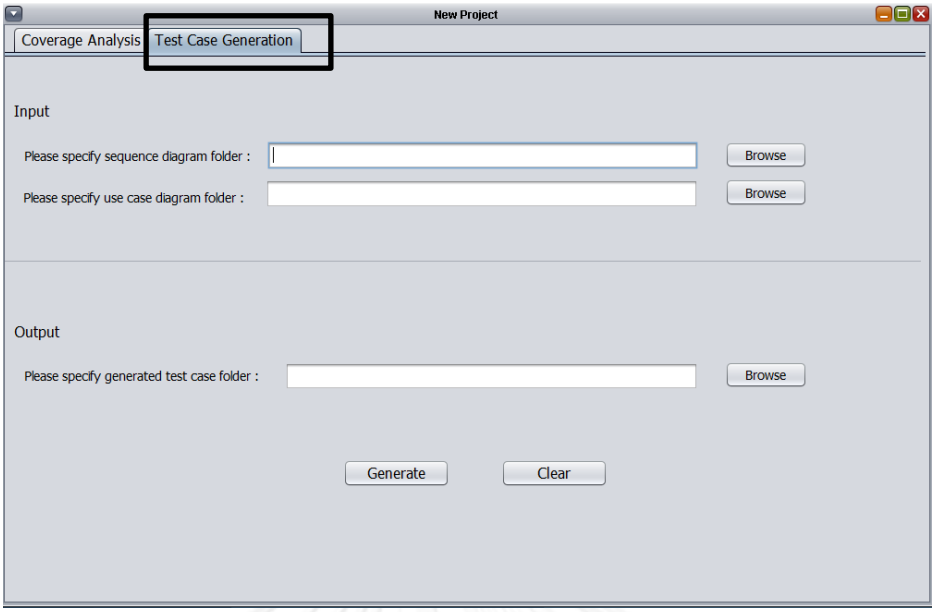


ภาพที่ ค-12 หน้าต่างแสดงผลการตรวจสอบความครอบคลุมเมื่อทดสอบเพิ่มเติมกับกรณีทดสอบที่เครื่องมือสร้างขึ้น

2. การสร้างกรณีทดสอบ

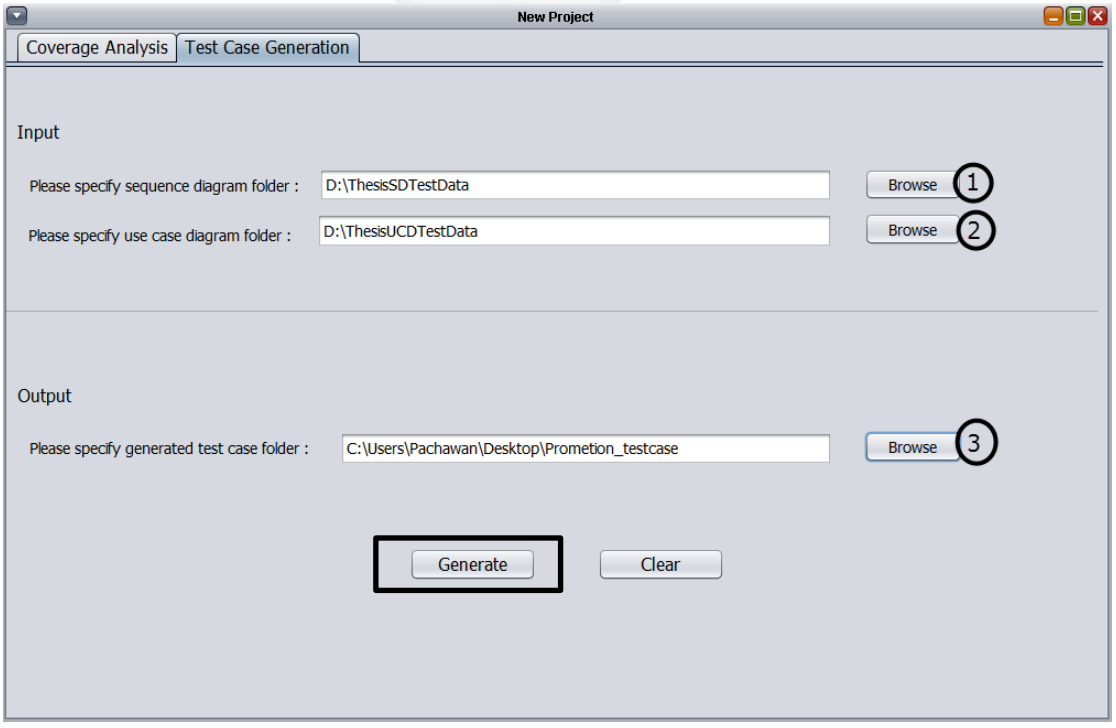
เมื่อผู้ทดสอบตรวจสอบความครอบคลุมของการทดสอบแล้วพบว่าการทดสอบยังไม่ครอบคลุมและต้องการสร้างกรณีทดสอบเพิ่มเติม สามารถทำได้ดังนี้

2.1 เลือกแถบ Test case generation เครื่องมือจะแสดงหน้าต่างการสร้างกรณีทดสอบดังภาพที่ ค-13



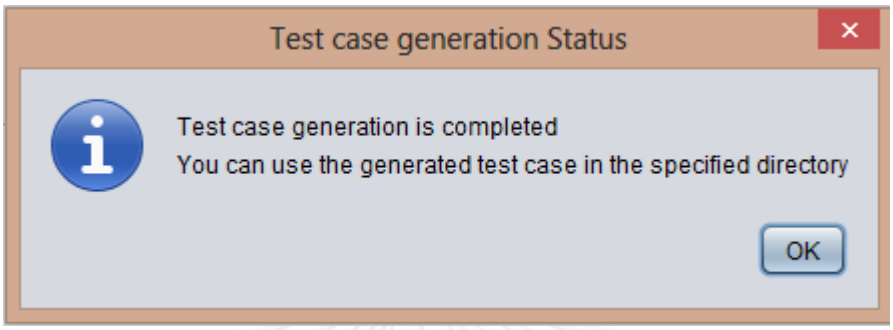
ภาพที่ ค-13 แถบหน้าต่าง Test case Generation

2.2 จากภาพที่ ค-14 ผู้ทดสอบกดปุ่ม Browse หมายเลข 1 เพื่อเลือโฟลเดอร์ที่ใช้เก็บแผนภาพลำดับ กดปุ่ม Browse หมายเลข 2 เพื่อเลือกโฟลเดอร์ที่ใช้เก็บคำอธิบายยูสเคส และกดปุ่ม Browse หมายเลข 3 เพื่อเลือกโฟลเดอร์ที่ต้องการใช้เก็บกรณีทดสอบ และกดปุ่ม Generate



ภาพที่ ค-14 แถบหน้าต่าง Test case Generation

2.3 เมื่อผู้ทดสอบกดปุ่ม Generate และเครื่องมือสร้างกรณีทดสอบเสร็จแล้ว จะแสดงหน้าต่างเพื่อแจ้งผู้ใช้ ดังภาพที่ ค-15



ภาพที่ ค-15 หน้าต่างแสดงสถานะเมื่อเครื่องมือสร้างกรณีทดสอบเสร็จแล้ว

2.4 ตัวอย่างกรณีทดสอบที่สร้างได้สามารถแสดงได้ดังภาพที่ ค-16

Test Case

Test case ID	**2**
Test case name	**Promotion_TestData_Ver2[3]**
Test case description	**(VIPStS == "VIPNo")&&(Promotiontype == "PromotionPoint")&&(PointType == "PointFromPrice")"**
Initial Value	**Cashier calculate the merchandise price**
Input Value	
VIPStS	**"VIPNo"**
Promotiontype	**"PromotionPoint"**
PointType	**"PointFromPrice"**
ID	**1007**
Expected Output	**System display total price and Point**

ภาพที่ ค-16 ตัวอย่างกรณีทดสอบที่ระบบสร้างขึ้น

ประวัติผู้เขียนวิทยานิพนธ์

น.ส.พัชร์วรรณ อักษรศรี เกิดวันที่ 9 ธันวาคม 2528 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาในหลักสูตรวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ เมื่อปีการศึกษา 2550 หลังจากนั้นทำงานที่บริษัท โตโยต้า ทูโช อิเล็กทรอนิกส์ (ไทยแลนด์) และเข้าศึกษาในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ จุฬาลงกรณ์มหาวิทยาลัยในปีการศึกษา 2554

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University