

การเลือกการทดสอบเชิงถดถอยแบบปลอดภัยจากแผนภาพลำดับ



นายวรินทร์ ทัพพงษ์

จุฬาลงกรณ์มหาวิทยาลัย

CHULALONGKORN UNIVERSITY

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)

เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย

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

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

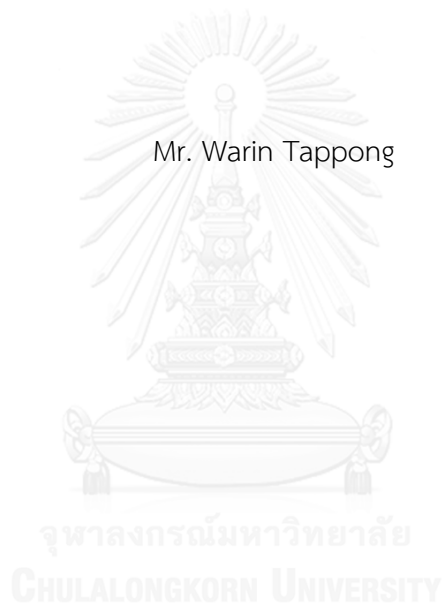
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2557

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

SAFE REGRESSION TEST SELECTION FROM SEQUENCE DIAGRAMS

Mr. Warin Tappong



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 2014

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การเลือกการทดสอบเชิงถดถอยแบบปลอดภัยจาก แผนภาพลำดับ
โดย	นายวรินทร์ ทัพพงษ์
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.ธรรมาทิพย์ สุวรรณศาสตร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้หัวข้อวิทยานิพนธ์ฉบับนี้เป็นส่วน
หนึ่งของการศึกษาตามหลักสูตรปริญญาโทบริหารธุรกิจ

.....คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.บัณฑิต เอื้ออาภรณ์)

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ
(รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.ธรรมาทิพย์ สุวรรณศาสตร์)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์)

.....กรรมการภายนอกมหาวิทยาลัย
(ผู้ช่วยศาสตราจารย์ ดร.ภัทรชัย ลลิตโรจน์วงศ์)

วรินทร์ ทัพพงษ์ : การเลือกการทดสอบเชิงถดถอยแบบปลอดภัยจากแผนภาพลำดับ (SAFE REGRESSION TEST SELECTION FROM SEQUENCE DIAGRAMS) อ.ที่ปรึกษา วิทยานิพนธ์หลัก: รศ. ดร.ธราทิพย์ สุวรรณศาสตร์, 109 หน้า.

การทดสอบเชิงถดถอยเป็นหนึ่งในกิจกรรมที่มีค่าใช้จ่ายแพงที่สุดในกระบวนการพัฒนาซอฟต์แวร์ แต่ก็ถือว่าเป็นกระบวนการทำงานหลักสำหรับการตรวจสอบคุณภาพของซอฟต์แวร์ หลังจากที่มีการพัฒนาและแก้ไขเปลี่ยนแปลง โดยงานวิจัยนี้ได้เสนอวิธีการของการเลือกกรณีทดสอบเชิงถดถอยจากแผนภาพลำดับ โดยจะสนใจกรณีทดสอบแบบบูรณาการซึ่งจะเป็นกรณีทดสอบที่อยู่ในรูปแบบของลำดับการทำงานของเมสเสจ ซึ่งกระบวนการตรวจสอบจะตรวจจับการเปลี่ยนแปลงจากแผนภาพลำดับระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับกับแผนภาพลำดับเวอร์ชันใหม่ที่มีการเปลี่ยนแปลงล่าสุด หลังจากนั้นจะใช้อัลกอริทึม Levenshtien Edit Distance ในการเปรียบเทียบเพื่อระบุการเปลี่ยนแปลง โดยผลลัพธ์ที่ได้จะเป็นรหัสที่ใช้อ้างอิงถึงลำดับของความสัมพันธ์จากแผนภาพลำดับสำหรับนำไปใช้ในการหาผลกระทบจากกรณีทดสอบที่อยู่ในฐานข้อมูลกรณีทดสอบ ซึ่งกรณีทดสอบที่ได้รับผลกระทบจะถูกคัดเลือกออกมาให้อยู่ในชุดกรณีทดสอบเชิงถดถอย รวมถึงกรณีทดสอบใหม่ที่ถูกเพิ่มขึ้นมาให้ครอบคลุมการเปลี่ยนแปลงที่มาจากแผนภาพลำดับเวอร์ชันใหม่ทั้งหมด งานวิจัยนี้ได้ประยุกต์ใช้วิธีการเลือกกรณีทดสอบแบบปลอดภัยเข้ามาใช้ร่วมกับแผนภาพลำดับ โดยกรณีทดสอบที่ถูกเลือกมาในชุดทดสอบเชิงถดถอยนั้นจะแบ่งตามประเภทของผลกระทบการเปลี่ยนแปลงและแสดงจำนวนกรณีทดสอบที่จะใช้สำหรับทดสอบซอฟต์แวร์หลังจากที่ซอฟต์แวร์มีการเปลี่ยนแปลงได้อย่างแม่นยำและมีประสิทธิภาพมากยิ่งขึ้น

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาควิชา วิศวกรรมคอมพิวเตอร์

สาขาวิชา วิศวกรรมซอฟต์แวร์

ปีการศึกษา 2557

ลายมือชื่อนิสิต

ลายมือชื่อ อ.ที่ปรึกษาหลัก

5471001021 : MAJOR SOFTWARE ENGINEERING

KEYWORDS: SEQUENCE DIAGRAM / REGRESSION TESTING / REGRESSION TEST CASE SELECTION

WARIN TAPPONG: SAFE REGRESSION TEST SELECTION FROM SEQUENCE DIAGRAMS. ADVISOR: ASSOC. PROF. TARATIP SUWANNASART, Ph.D., 109 pp.

Regression testing is one of the most expensive activities in software development life cycle. It is a primary process for ensuring software quality after it has been implemented and modified. This research has proposed an approach of safe regression test selection technique by using sequence diagrams and focused on the integration test cases which are the test cases in sequences of message calls format. The verification process could detect any changes from two sequence diagrams between an original design from the latest version of sequence diagram and a new design from the new version of sequence diagram. After that, we apply Levenshtien Edit Distance algorithm to compare and identify the impact. Then, outcome would be the code that is referred to relationship from sequence diagram which is used for finding the impacted test cases in the test case database which they would be included to the regression test set and new test cases would be generated to cover new functions. Our approach adapts traditional safe regression test selection technique based on sequence diagram which can be classified the impacted test cases and reduce the number of test cases for software testing after the design is changed.

Department: Computer Engineering Student's Signature

Field of Study: Software Engineering Advisor's Signature

Academic Year: 2014

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยความรู้และความช่วยเหลือจากรองศาสตราจารย์ ดร. ธาราทิพย์ สุวรรณศาสตร์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่เสียสละเวลาให้คำปรึกษาแนะนำแนวทางในการทำงานวิจัยตลอดจนแก้ไขข้อบกพร่องต่างๆด้วยความละเอียดถี่ถ้วนด้วยดีเสมอมา ข้าพเจ้ารู้สึกซาบซึ้งเป็นอย่างยิ่ง จึงขอกราบขอบพระคุณเป็นอย่างสูง ณ โอกาสนี้

ขอขอบพระคุณคณาจารย์ทุกท่านที่ได้แนะนำ สั่งสอนและให้ความรู้แก่ข้าพเจ้าตลอดระยะเวลาการศึกษา ขอขอบพระคุณรองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ ประธานกรรมการสอบวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร.อาทิตย์ ทองทักษ์ และผู้ช่วยศาสตราจารย์ ดร.ภัทรชัย ลลิตโรจน์วงศ์ กรรมการสอบวิทยานิพนธ์ ที่กรุณาเสียสละเวลาในการให้คำแนะนำและชี้ตติเกลาทำให้วิทยานิพนธ์เล่มนี้มีความถูกต้องและสมบูรณ์มากขึ้น

ขอขอบพระคุณ คุณพ่อ คุณแม่ พี่ชาย คุณป้า ที่ให้ความรัก ความห่วงใยและกำลังใจที่ดีเสมอมา

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ และคุณนริศรา แสนเรืองเดช สำหรับทุกคำปรึกษาและกำลังใจในการทำวิทยานิพนธ์ให้สำเร็จลุล่วงได้

คุณค่าและประโยชน์ของวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบเป็นกตัญญูทเวทิตาแด่ บพกาณี บุรพาจารย์ และผู้มีพระคุณทุกท่านทั้งในอดีตและปัจจุบัน ที่ทำให้ข้าพเจ้าเป็นผู้มีการศึกษาและประสบความสำเร็จมาจนตราบเท่าทุกวันนี้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ฅ
สารบัญรูป.....	ฉ
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของงานวิจัย.....	2
1.3 ขอบเขตของงานวิจัย	3
1.4 ขั้นตอนและวิธีการดำเนินงานวิจัย	4
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	5
1.6 ลำดับขั้นตอนในการเสนองานวิจัย.....	5
1.7 ผลงานตีพิมพ์.....	5
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	6
2.1 ทฤษฎีที่เกี่ยวข้อง.....	6
2.1.1 การเลือกกรณีทดสอบสำหรับการทดสอบเชิงถดถอย [15]	6
2.1.2 การทดสอบเชิงถดถอยแบบปลอดภัย [15].....	7
2.1.3 การวิเคราะห์ผลกระทบการเปลี่ยนแปลงของยูเอ็มแอลสำหรับการเลือกกรณี ทดสอบเชิงถดถอย [13]	8
2.1.4 อัลกอริทึม Levenshtien Edit Distance [18].....	9
2.2 งานวิจัยที่เกี่ยวข้อง	11
2.2.1 Test Sequence Generation from UML Sequence Diagram [7]	11

2.2.2 Automatic Test Case Generation from UML Models [14]	13
2.2.3 Regression Testing UML Design [4].....	15
บทที่ 3 การวิเคราะห์และออกแบบเครื่องมือ.....	17
3.1 ภาพรวมการทำงานของเครื่องมือทดสอบ	17
3.1.1 ส่วนการสร้างตารางความสัมพันธ์ (Sequence Dependency Table Generator)	18
3.1.1.1 ฐานข้อมูลความสัมพันธ์ของอ็อบเจกต์ (Sequence Related Objects Database).....	18
3.1.1.2 ตารางความสัมพันธ์เวอร์ชันต้นฉบับ (Original Sequence Dependency Table: OSDT).....	19
3.1.1.3 ตารางความสัมพันธ์เวอร์ชันใหม่ (New Sequence Dependency Table: NSDT).....	23
3.1.2 รายการลำดับการแก้ไข (List of Modification).....	25
3.1.3 ส่วนการเปรียบเทียบและหาผลกระทบการเปลี่ยนแปลง (Comparator).....	26
3.1.4 ส่วนการเลือกกรณีทดสอบ (Test Case Selector).....	31
3.1.5 ชุดกรณีทดสอบเชิงถดถอย (Regression Test Set).....	32
3.1.6 ฐานข้อมูลกรณีทดสอบ (Test Case Database).....	32
3.2 การวิเคราะห์และออกแบบเครื่องมือ.....	34
3.2.1 แผนภาพยูสเคส.....	35
3.2.1.1 แผนภาพยูสเคสในส่วนของการสร้างตารางความสัมพันธ์.....	36
3.2.1.2 แผนภาพยูสเคสในส่วนของการเปรียบเทียบการเปลี่ยนแปลง	37
3.2.1.3 แผนภาพยูสเคสในส่วนของการหาผลกระทบการเปลี่ยนแปลง.....	38
3.2.1.4 แผนภาพยูสเคสในส่วนของการเลือกกรณีทดสอบเชิงถดถอย	39
3.2.1.5 แผนภาพยูสเคสในส่วนของการสร้างกรณีทดสอบ	40

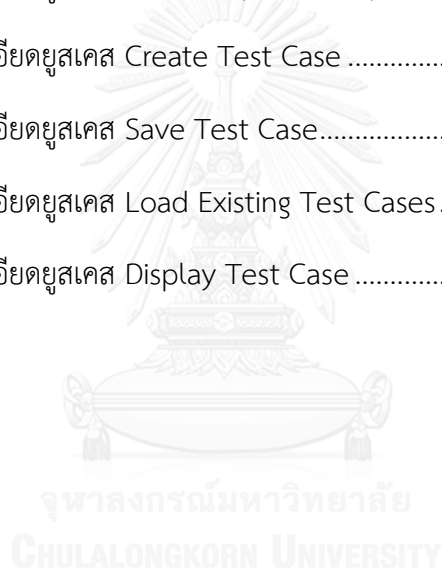
3.2.2 แผนภาพกิจกรรม.....	42
3.2.2.1 แผนภาพกิจกรรมการนำเข้าแผนภาพลำดับและการสร้างกรณีทดสอบ	43
3.2.2.2 แผนภาพกิจกรรมการเปรียบเทียบและการวิเคราะห์ผลกระทบ	44
3.2.2.3 แผนภาพกิจกรรมของการแสดงข้อมูลกรณีทดสอบ	45
3.2.3 แผนภาพคลาส	46
3.2.3.1 การสร้างตารางความสัมพันธ์จากแผนภาพลำดับ	47
3.2.3.2 การเปรียบเทียบและการวิเคราะห์หาผลกระทบการเปลี่ยนแปลง	55
3.2.3.3 การเลือกชุดทดสอบเชิงถดถอย.....	59
บทที่ 4 การพัฒนาเครื่องมือ.....	61
4.1 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ	61
4.1.1 ฮาร์ดแวร์ (Hardware).....	61
4.1.2 ซอฟต์แวร์ (Software).....	61
4.2 โครงสร้างส่วนต่อประสานของผู้ใช้งาน	62
4.2.1 เครื่องมือการสร้างตารางความสัมพันธ์และกรณีทดสอบ.....	62
4.2.2 เครื่องมือเปรียบเทียบและหาผลกระทบของการเปลี่ยนแปลง.....	64
4.2.3 เครื่องมือแสดงชุดทดสอบ	68
บทที่ 5 การทดสอบเครื่องมือ	70
5.1 สภาพแวดล้อมที่ใช้ในการทดสอบ.....	70
5.1.1 ฮาร์ดแวร์.....	70
5.1.2 ซอฟต์แวร์	70
5.2 แผนภาพลำดับที่ใช้ในการทดสอบ	70
5.2.1 แผนภาพลำดับที่ 1 ระบบเอทีเอ็ม	70
5.2.2 แผนภาพลำดับที่ 2 ระบบลงทะเบียนคนไข้.....	71

5.3 ขั้นตอนการทดสอบเครื่องมือและผลการทดสอบ	71
5.3.1 การทดสอบการสร้างตารางความสัมพันธ์	71
5.3.2 การเปรียบเทียบการเปลี่ยนแปลงระหว่างแผนภาพลำดับ	73
5.3.3 การหาผลกระทบการเปลี่ยนแปลง	75
5.3.4 การสร้างกรณีทดสอบและการเลือกชุดทดสอบเชิงถดถอย	77
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ	80
6.1 สรุปผลการวิจัย.....	80
6.2 ข้อจำกัดของเครื่องมือทดสอบ	80
6.3 ข้อเสนอแนะและแนวทางการวิจัยในอนาคต.....	81
รายการอ้างอิง	82
ภาคผนวก ก รายละเอียดยูสเคส	86
ภาคผนวก ข แผนภาพที่ใช้ในการทดสอบ	97
1. แผนภาพที่ 1 ระบบเอทีเอ็ม.....	97
1.1 แผนภาพลำดับเวอร์ชันต้นฉบับ	97
1.2 แผนภาพลำดับเวอร์ชันใหม่.....	97
2. แผนภาพที่ 2 ระบบลงทะเบียนคนไข้.....	98
2.1 แผนภาพลำดับเวอร์ชันต้นฉบับ	98
2.2 แผนภาพลำดับเวอร์ชันใหม่.....	100
ภาคผนวก ค วิธีการนำออกแผนภาพลำดับ	102
ขั้นตอนการนำออกแผนภาพลำดับ.....	102
ภาคผนวก ง คู่มือการติดตั้งเครื่องมือ	104
ขั้นตอนการติดตั้งเครื่องมือทดสอบ	104
ประวัติผู้เขียนวิทยานิพนธ์	109

สารบัญตาราง

ตารางที่ 3.1	ตารางประเภทของการเปรียบเทียบ	26
ตารางที่ 5.1	ตารางแสดงการเปลี่ยนแปลงของแผนภาพลำดับเวอร์ชันใหม่.....	72
ตารางที่ 5.2	ตารางสรุปผลการทดสอบการสร้างตารางความสัมพันธ์จากแผนภาพลำดับ	72
ตารางที่ 5.3	ตารางแสดงผลการเปรียบเทียบของแผนภาพลำดับที่ 1	73
ตารางที่ 5.4	ตารางแสดงผลการเปรียบเทียบของแผนภาพลำดับที่ 2	74
ตารางที่ 5.5	ตารางสรุปรหัสลำดับการทดสอบที่มีการเปลี่ยนแปลง.....	75
ตารางที่ 5.6	ตารางสรุปผลการทดสอบของการเปรียบเทียบการเปลี่ยนแปลง	75
ตารางที่ 5.7	ตารางแสดงผลลัพธ์การหาผลกระทบการเปลี่ยนแปลงจากแผนภาพลำดับ.....	76
ตารางที่ 5.8	ตารางสรุปรหัสที่ได้รับผลกระทบการเปลี่ยนแปลงจากแผนภาพลำดับ	77
ตารางที่ 5.9	ตารางสรุปผลการทดสอบการหาผลกระทบการเปลี่ยนแปลงจากแผนภาพลำดับ	77
ตารางที่ 5.10	ตารางการคัดเลือกชุดทดสอบเชิงถดถอยจากแผนภาพลำดับที่ 1	78
ตารางที่ 5.11	ตารางการคัดเลือกชุดทดสอบเชิงถดถอยจากแผนภาพลำดับที่ 2.....	79
ตารางที่ 5.12	ตารางสรุปผลการสร้างกรณีทดสอบและการเลือกชุดทดสอบเชิงถดถอย	79
ตารางที่ ก - 1	รายละเอียดยูสเคส Load XML.....	86
ตารางที่ ก - 2	รายละเอียดยูสเคส Generate Sequence Dependency Table	86
ตารางที่ ก - 3	รายละเอียดยูสเคส Save Sequence Dependency Table	87
ตารางที่ ก - 4	รายละเอียดยูสเคส Load Original Sequence Dependency Table	87
ตารางที่ ก - 5	รายละเอียดยูสเคส Load New Sequence Dependency Table.....	88
ตารางที่ ก - 6	รายละเอียดยูสเคส Get GID.....	88
ตารางที่ ก - 7	รายละเอียดยูสเคส Compare List of Modification.....	89
ตารางที่ ก - 8	รายละเอียดยูสเคส Generate Comparison Result	89
ตารางที่ ก - 9	รายละเอียดยูสเคส Load Comparison Result.....	90

ตารางที่ ก - 10	รายละเอียดชุดทดสอบ Find Impact.....	90
ตารางที่ ก - 11	รายละเอียดชุดทดสอบ Find Impact (ต่อ).....	91
ตารางที่ ก - 12	รายละเอียดชุดทดสอบ Generate List of Impacted.....	91
ตารางที่ ก - 13	รายละเอียดชุดทดสอบ Load Existing Test Cases.....	92
ตารางที่ ก - 14	รายละเอียดชุดทดสอบ Load List of Impacted.....	92
ตารางที่ ก - 15	รายละเอียดชุดทดสอบ Select Test Case.....	93
ตารางที่ ก - 16	รายละเอียดชุดทดสอบ Display Regression Test Set.....	93
ตารางที่ ก - 17	รายละเอียดชุดทดสอบ Load Sequence Dependency Table.....	94
ตารางที่ ก - 18	รายละเอียดชุดทดสอบ Create Test Case.....	94
ตารางที่ ก - 19	รายละเอียดชุดทดสอบ Save Test Case.....	95
ตารางที่ ก - 20	รายละเอียดชุดทดสอบ Load Existing Test Cases.....	95
ตารางที่ ก - 21	รายละเอียดชุดทดสอบ Display Test Case.....	96



สารบัญรูป

รูปที่ 2.1	กระบวนการทดสอบเชิงถดถอยแบบปลอดภัย [15].....	7
รูปที่ 2.2	รหัสเทียมของอัลกอริทึม Levenshtein Edit Distance [18]	10
รูปที่ 2.3	รูปตารางผลลัพธ์ของการคำนวณระยะแก้ไขจากข้อมูลของสตริง 2 ชุด [18].....	10
รูปที่ 2.4	กราฟลำดับความสัมพันธ์จากแผนภาพลำดับ [7]	11
รูปที่ 2.5	รหัสเทียมของอัลกอริทึม Traversal [7]	12
รูปที่ 2.6	ผลลัพธ์ของการสร้างลำดับการทดสอบจากแผนภาพลำดับ [7]	12
รูปที่ 2.7	การแปลงแผนภาพยูสเคสเป็นกราฟแผนภาพยูสเคส [14].....	13
รูปที่ 2.8	การแปลงแผนภาพลำดับเป็นกราฟแผนภาพลำดับ [14]	13
รูปที่ 2.9	ขั้นตอนการนำข้อมูลที่จำเป็นมารวมกันเพื่อใช้ในการสร้างกรณีทดสอบ [14].....	14
รูปที่ 2.10	ผลลัพธ์ของกรณีทดสอบที่ถูกสร้างขึ้นมาจากงานวิจัยนี้ [14].....	14
รูปที่ 2.11	แสดงถึงการนำข้อมูลของแผนภาพลำดับกับแผนภาพคลาสมารวมกัน [4]	15
รูปที่ 3.1	แผนภาพรวมของระบบการเลือกชุดกรณีทดสอบเชิงถดถอย.....	17
รูปที่ 3.2	ส่วนประกอบของส่วนการสร้างตารางความสัมพันธ์.....	18
รูปที่ 3.3	โครงสร้างของไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับ	19
รูปที่ 3.4	ตัวอย่างแผนภาพลำดับของระบบเอทีเอ็มในส่วนของ การตรวจสอบรหัสบัตร [14].....	20
รูปที่ 3.5	ตัวอย่างตารางความสัมพันธ์เวอร์ชันต้นฉบับจากแผนภาพลำดับของระบบเอทีเอ็ม.....	20
รูปที่ 3.6	ตัวอย่างข้อมูลของชื่อเมสเสจจากแผนภาพลำดับ	21
รูปที่ 3.7	ตัวอย่างข้อมูลของอ็อบเจกต์ต้นทางจากแผนภาพลำดับ.....	21
รูปที่ 3.8	ตัวอย่างข้อมูลของอ็อบเจกต์ปลายทางจากแผนภาพลำดับ.....	22
รูปที่ 3.9	ตัวอย่างข้อมูลของการ์ดจากแผนภาพลำดับ	22
รูปที่ 3.10	ตัวอย่างข้อมูลของการวนซ้ำจากแผนภาพลำดับ	23
รูปที่ 3.11	ตัวอย่างแผนภาพลำดับที่มีการเพิ่มเมสเสจใหม่ การลบออกและมีการสลับตำแหน่ง	24

รูปที่ 3.12 ตัวอย่างตารางความสัมพันธ์เวอร์ชันใหม่	24
รูปที่ 3.13 ตัวอย่างรายการลำดับการแก้ไขของแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่	25
รูปที่ 3.14 ขั้นตอนการเปรียบเทียบจากรายการลำดับการแก้ไข	28
รูปที่ 3.15 ตัวอย่างผลกระทบทางตรงจากการเปลี่ยนแปลงจากการแทรกและการแทนที่.....	30
รูปที่ 3.16 ขั้นตอนการเลือกกรณีทดสอบที่ได้รับผลกระทบต่อการเปลี่ยนแปลงแผนภาพลำดับ.....	31
รูปที่ 3.17 ขั้นตอนการสร้างและจัดเก็บกรณีทดสอบเข้าสู่ฐานข้อมูลกรณีทดสอบ.....	33
รูปที่ 3.18 แผนภาพรวมยูสเคสของเครื่องมือทดสอบ	35
รูปที่ 3.19 แผนภาพยูสเคสการสร้างตารางความสัมพันธ์	36
รูปที่ 3.20 แผนภาพยูสเคสการเปรียบเทียบการเปลี่ยนแปลง	37
รูปที่ 3.21 แผนภาพยูสเคสการหาผลกระทบการเปลี่ยนแปลง.....	38
รูปที่ 3.22 แผนภาพยูสเคสการเลือกกรณีทดสอบเชิงถดถอย.....	39
รูปที่ 3.23 แผนภาพยูสเคสการสร้างกรณีทดสอบ	40
รูปที่ 3.24 แผนภาพยูสเคสในส่วนของ การแสดงกรณีทดสอบ	41
รูปที่ 3.25 ภาพรวมของแผนภาพกิจกรรมของเครื่องมือทดสอบ	42
รูปที่ 3.26 แผนภาพกิจกรรมของส่วนการนำเข้าแผนภาพลำดับและการสร้างกรณีทดสอบ	43
รูปที่ 3.27 แผนภาพกิจกรรมการเปรียบเทียบและการวิเคราะห์ผลกระทบ	44
รูปที่ 3.28 แผนภาพกิจกรรมของการแสดงข้อมูลกรณีทดสอบ.....	45
รูปที่ 3.29 แผนภาพรวมของคลาสของเครื่องมือทดสอบ.....	46
รูปที่ 3.30 แผนภาพคลาสของการสร้างตารางความสัมพันธ์จากแผนภาพลำดับ	47
รูปที่ 3.31 คลาส FormMain	48
รูปที่ 3.32 คลาส SequenceDiagramService	49
รูปที่ 3.33 คลาส SequenceDependency	49
รูปที่ 3.34 คลาส TempModel	50
รูปที่ 3.35 คลาส FormDiagramPicture.....	50

รูปที่ 3.36	คลาส SequenceDiagramGenerator.....	51
รูปที่ 3.37	คลาส XmlHelper.....	51
รูปที่ 3.38	คลาส SelectionModel.....	51
รูปที่ 3.39	คลาส SafeRegressionEntities.....	52
รูปที่ 3.40	คลาส diagram_version.....	52
รูปที่ 3.41	คลาส sequence_related_object.....	53
รูปที่ 3.42	คลาส DiagramModel.....	53
รูปที่ 3.43	คลาส Constant.....	54
รูปที่ 3.44	แผนภาพรวมของคลาสของการเปรียบเทียบและการวิเคราะห์หาผลกระทบ.....	55
รูปที่ 3.45	คลาส FormSummary.....	56
รูปที่ 3.46	คลาส CompareResultModel.....	56
รูปที่ 3.47	คลาส FormCompareDetails.....	57
รูปที่ 3.48	คลาส ImpactResultModel.....	57
รูปที่ 3.49	คลาส FormResult.....	58
รูปที่ 3.50	แผนภาพคลาสของการเลือกชุดทดสอบเชิงถดถอย.....	59
รูปที่ 3.51	คลาส TestCaseService.....	59
รูปที่ 3.52	คลาส test_case.....	60
รูปที่ 4.1	หน้าจอตารางความสัมพันธ์เวอร์ชันต้นฉบับที่ถูกสร้างจากแผนภาพลำดับเวอร์ชันแรก.....	62
รูปที่ 4.2	หน้าจอแสดงรูปของแผนภาพลำดับเวอร์ชันต้นฉบับ.....	63
รูปที่ 4.3	หน้าจอการสร้างกรณีทดสอบจากแผนภาพลำดับเวอร์ชันต้นฉบับ.....	64
รูปที่ 4.4	หน้าจอของตารางความสัมพันธ์จากแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่.....	65
รูปที่ 4.5	หน้าจอชุดทดสอบเชิงถดถอยสำหรับนำไปใช้ทดสอบกับแผนภาพลำดับเวอร์ชันใหม่.....	66
รูปที่ 4.6	หน้าจอส่วนการแสดงรายละเอียดของการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลง... ..	67
รูปที่ 4.7	หน้าจอแสดงรายละเอียดของการระบุผลกระทบจากการเปลี่ยนแปลง.....	68

รูปที่ 4.8 หน้าจอแสดงข้อมูลกรณีทดสอบ	69
รูปที่ ข - 1 แผนภาพลำดับของระบบเอทีเอ็มเวอร์ชันต้นฉบับ	97
รูปที่ ข - 2 แผนภาพลำดับของระบบเอทีเอ็มเวอร์ชันใหม่	98
รูปที่ ข - 3 แผนภาพลำดับของระบบลงทะเบียนคนใช้เวอร์ชันต้นฉบับ	99
รูปที่ ข - 4 แผนภาพลำดับของระบบลงทะเบียนคนใช้เวอร์ชันใหม่	101
รูปที่ ค - 1 เลือกรายการสำหรับการนำออกแผนภาพลำดับ.....	102
รูปที่ ค - 2 หน้าจอการนำออกไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับสำหรับเครื่องมือทดสอบ.....	103
รูปที่ ง - 1 ไฟล์สำหรับใช้ในการติดตั้งเครื่องมือทดสอบ	104
รูปที่ ง - 2 หน้าจอเริ่มต้นการติดตั้งเครื่องมือทดสอบ	105
รูปที่ ง - 3 หน้าจอการเลือกโพลเดอร์ที่ต้องการจะติดตั้ง	106
รูปที่ ง - 4 หน้าจอยืนยันการติดตั้งเครื่องมือทดสอบ	107
รูปที่ ง - 5 หน้าจอแสดงการติดตั้งเสร็จสมบูรณ์.....	108
รูปที่ ง - 6 หน้าจอแสดงชอร์ตคัทของเครื่องมือติดตั้งบนเดสก์ท็อป	108

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

หนึ่งในกิจกรรมที่มีค่าใช้จ่ายมากที่สุดที่จะเกิดขึ้นจากขั้นตอนการพัฒนาซอฟต์แวร์และการบำรุงรักษานั้นก็คือ การทดสอบซอฟต์แวร์หลังการเปลี่ยนแปลง [1]

การทดสอบเชิงถดถอย (Regression Testing) นั้นถือเป็นขั้นตอนการทดสอบซอฟต์แวร์ที่มีค่าใช้จ่ายในการดำเนินการสูงที่สุดในกระบวนการทดสอบซอฟต์แวร์ ดังนั้นถ้าสามารถที่จะลดกรณีทดสอบ (Test Case) ที่ซ้ำซ้อนกันออกไปให้ได้มากที่สุดจากชุดของกรณีทดสอบที่จะถูกนำมาใช้ทำการทดสอบก็จะทำให้สามารถประหยัดต้นทุนในการทดสอบเชิงถดถอยได้ [2] อีกทั้งกระบวนการทดสอบซอฟต์แวร์นั้นมีผลต่อคุณภาพของซอฟต์แวร์และความสำเร็จของโครงการ เพื่อคุณภาพของซอฟต์แวร์ที่ถูกพัฒนาอยู่ในมาตรฐานเราจึงจำเป็นต้องทำการทดสอบในทุกๆส่วนที่ถูกแก้ไขให้ครอบคลุม เพื่อให้มั่นใจได้ว่าการเปลี่ยนแปลงนั้นจะไม่มีข้อผิดพลาดและผลกระทบต่อส่วนอื่นที่ไม่ได้รับการเปลี่ยนแปลง [3]

เมื่อมีการเปลี่ยนแปลงในซอฟต์แวร์ ไม่ว่าจะเป็นการเปลี่ยนแปลงในการออกแบบหรือรหัสต้นฉบับ (Source Code) นั้น ก็มีโอกาสเป็นไปได้ที่จะมีข้อผิดพลาดเกิดขึ้นมาใหม่ในซอฟต์แวร์ [4] หนึ่งในกลยุทธ์ของการทดสอบเชิงถดถอย คือ วิธีการทดสอบทั้งหมด (Retest-all Approach) กับซอฟต์แวร์ที่ได้ถูกเปลี่ยนแปลง แต่วิธีการนี้จำเป็นต้องใช้เวลาและทรัพยากรจำนวนมากในการดำเนินการ โดยการที่จะลดระยะเวลาในการดำเนินการสำหรับการทดสอบในส่วนที่ถูกเปลี่ยนแปลงนั้น จะต้องทำการเลือกเฉพาะกรณีทดสอบบางส่วนที่ได้รับผลกระทบจากการเปลี่ยนแปลงจากชุดทดสอบ (Test Suite) ทั้งหมดมาใช้ในการทดสอบ [5]

ยูเอ็มแอล (UML: Unified Modeling Language) เป็นที่รู้จักและถูกใช้กันอย่างกว้างขวางโดยวิศวกรซอฟต์แวร์ เพื่อใช้ในการอธิบายถึงโครงสร้างระบบของซอฟต์แวร์ ในขั้นตอนเริ่มแรกของกระบวนการออกแบบซอฟต์แวร์ วิศวกรซอฟต์แวร์จะทำการร่างยูเอ็มแอลเพื่อใช้ในการอธิบายและนำเสนอแนวความคิดเริ่มต้นก่อนที่จะนำไปใช้ในการพัฒนาต่อไป [6]

แผนภาพลำดับ (Sequence Diagram) เป็นแผนภาพหนึ่งในมาตรฐานยูเอ็มแอลที่ได้รับการยืนยันว่าเป็นแผนภาพที่ได้รับการใช้งานมากที่สุดในมาตรฐานยูเอ็มแอล [7] โดยแผนภาพลำดับจะถูกใช้กันอย่างกว้างขวางในระหว่างขั้นตอนการพัฒนาซอฟต์แวร์ โดยใช้แสดงถึงความสัมพันธ์และการติดต่อสื่อสารกันระหว่างอ็อบเจกต์ รวมทั้งแสดงถึงพฤติกรรมของระบบทั้งหมดในรูปแบบภาพที่

สามารถแสดงถึงการทำงานเป็นช่วงเวลา ลำดับชั้นโดยมีการสื่อสารกันระหว่างอ็อบเจกต์ผ่านเมสเสจที่มีทั้งแบบรับและส่ง [8]

ในระหว่างกระบวนการพัฒนาซอฟต์แวร์นั้นจะมีการเปลี่ยนแปลงอยู่เสมอ ไม่ว่าจะเป็นการเปลี่ยนแปลงที่เกิดขึ้นจากการซ่อมข้อผิดพลาด (Bug fixes) หรือการเพิ่มเติมฟังก์ชันการทำงานใหม่เข้ามา การพัฒนาการเปลี่ยนแปลงในส่วนนั้น บ่อยครั้งที่ทำให้เกิดการประเมินโครงการผิดพลาดเนื่องจากการเปลี่ยนแปลงเหล่านั้นยากเกินความคาดหมาย ซึ่งจะส่งผลกระทบต่อการประเมินระยะเวลาการพัฒนาซอฟต์แวร์ทั้งโครงการ ดังนั้นแผนภาพยูเอ็มแอลจึงเข้ามามีบทบาทในการใช้ช่วยในการวิเคราะห์ผลกระทบต่อเปลี่ยนแปลงของซอฟต์แวร์เพราะคุณสมบัติของแผนภาพยูเอ็มแอลนั้นสามารถแสดงให้เห็นขั้นตอนการพัฒนาในรูปแบบภาพทำให้ง่ายต่อการเข้าใจและการใช้ในการประเมินการพัฒนาซอฟต์แวร์ [9]

งานวิจัย [10-12] ได้นำเสนอวิธีการเลือกกรณีทดสอบเชิงถดถอยจากการหาผลกระทบการเปลี่ยนแปลงของซอฟต์แวร์ในระดับของรหัสต้นฉบับระหว่างซอฟต์แวร์เวอร์ชันปัจจุบันกับซอฟต์แวร์เวอร์ชันใหม่ แต่ข้อจำกัดของวิธีการนี้ คือ ซอฟต์แวร์เวอร์ชันใหม่จะต้องถูกพัฒนาให้เสร็จเรียบร้อยแล้วถึงจะสามารถนำมาใช้วิเคราะห์หาผลกระทบได้ [13] ทำให้ผู้ทดสอบระบบ (Tester) จำเป็นต้องรอให้ซอฟต์แวร์ถูกพัฒนาจนแล้วเสร็จจึงจะสามารถเลือกกรณีทดสอบสำหรับการทดสอบเชิงถดถอยได้

งานวิจัยนี้ได้ศึกษาวิธีการสร้างกรณีทดสอบจากแผนภาพลำดับ [14] เพื่อนำไปใช้เป็นกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอย และการเลือกกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอย [4] หลังจากที่ซอฟต์แวร์มีการเปลี่ยนแปลง โดยใช้การประยุกต์วิธีการสร้างกรณีทดสอบจากแผนภาพลำดับให้สามารถตรวจสอบถึงการเปลี่ยนแปลงที่เกิดขึ้นจากแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่ จากนั้นประยุกต์ทั้งสองวิธีเข้าด้วยกันเพื่อพัฒนาเครื่องมือที่ใช้ในการเลือกกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอยที่ได้รับผลกระทบจากการเปลี่ยนแปลงจากแผนภาพลำดับ โดยมีมุ่งหมายที่จะนำไปประยุกต์ใช้กับกรณีทดสอบสำหรับการทดสอบแบบบูรณาการ (Integration Test) เพื่อให้สามารถลดระยะเวลา ทรัพยากรและค่าใช้จ่ายในการทดสอบซอฟต์แวร์เฉพาะส่วนที่ได้รับผลกระทบจากการเปลี่ยนแปลงเท่านั้น

1.2 วัตถุประสงค์ของงานวิจัย

เพื่อออกแบบและพัฒนาเครื่องมือที่ใช้ในการเลือกกรณีทดสอบสำหรับการทดสอบเชิงถดถอยที่ได้รับผลกระทบจากการเปลี่ยนแปลงจากแผนภาพลำดับ

1.3 ขอบเขตของงานวิจัย

รายละเอียดขอบเขตของงานวิจัยมีดังนี้

- 1.3.1 เครื่องมือทดสอบจะถูกพัฒนาโดยภาษา C#
- 1.3.2 แผนภาพลำดับจะต้องออกแบบด้วยโปรแกรมวิซวลพาราไดม์ (Visual Paradigm) และรองรับการทำงานสัญลักษณ์ ดังต่อไปนี้
 - 1.3.2.1 เส้นชีวิตของอ็อบเจกต์ (Object Lifeline)
 - 1.3.2.2 จุดเริ่มต้นและสิ้นสุดของอ็อบเจกต์ระหว่างที่ยังมีชีวิตอยู่ (Activation)
 - 1.3.2.3 เส้นชีวิต (Lifeline)
 - 1.3.2.4 คำสั่งหรือฟังก์ชันที่อ็อบเจกต์หนึ่งส่งไปยังอีกอ็อบเจกต์หนึ่ง (Call)
 - 1.3.2.5 การประมวลผลและคืนค่าที่ได้ภายในอ็อบเจกต์เดียวกัน (Return)
 - 1.3.2.6 ข้อความส่งกลับ (Message Return)
- 1.3.3 สนับสนุนแผนภาพลำดับของยูเอ็มแอลเวอร์ชัน 2.0
- 1.3.4 แผนภาพลำดับที่จะใช้ในการทดสอบจะเป็นแผนภาพลำดับของระบบที่มีการแก้ไขรายละเอียดในการทำงานและนำแผนภาพลำดับทั้งสองเวอร์ชันมาใช้ในการทดสอบกับเครื่องมือทดสอบให้ครอบคลุมทุกการเปลี่ยนแปลงตามขอบเขตที่งานวิจัยนี้ได้สนับสนุน โดยเครื่องมือทดสอบสามารถตรวจสอบการเปลี่ยนแปลงจากแผนภาพลำดับได้ ดังต่อไปนี้
 - 1.3.4.1 การเพิ่มหรือลบเมสเสจ
 - 1.3.4.2 การเพิ่มหรือลบอ็อบเจกต์
 - 1.3.4.3 การเปลี่ยนแปลงรายละเอียดการทำงาน ซึ่งมีทั้งหมด 2 ประเภท คือ
 - 1) การเปลี่ยนลำดับของเมสเสจ
 - 2) การเปลี่ยนเมทอดซิกเนเจอร์
- 1.3.5 ข้อมูลที่ใช้ในการตรวจสอบการเปลี่ยนแปลง
 - 1) ชื่อของเมสเสจ
 - 2) อ็อบเจกต์ต้นทาง
 - 3) อ็อบเจกต์ปลายทาง
 - 4) การ์ด
 - 5) การวนซ้ำ

1.3.6 การสร้างและการจัดเก็บกรณีทดสอบสามารถอธิบายได้ดังต่อไปนี้

1.3.6.1 กรณีทดสอบจะอยู่ในรูปแบบของลำดับความสัมพันธ์ของเมสเสจหรือกรณีทดสอบที่ใช้สำหรับการทดสอบแบบบูรณาการ

1.3.6.2 กรณีทดสอบจะถูกจัดเก็บในฐานข้อมูลกรณีทดสอบ โดยจะจัดเก็บอยู่ในรูปแบบของตารางซึ่งฐานข้อมูลที่ใช้จะเป็นฐานข้อมูลเชิงสัมพันธ์ (Relational Database) ซึ่งจะใช้โปรแกรมไมโครซอฟท์เอสคิวแอล เซิร์ฟเวอร์ เอ็กซ์เพรส 2008 (Microsoft SQL Server Express 2008)

1.3.6.3 กรณีทดสอบใหม่จะถูกสร้างได้จากแผนภาพลำดับเวอร์ชันต้นฉบับที่ถูกนำเข้ามายังเครื่องมือทดสอบเป็นครั้งแรกและกรณีทดสอบใหม่ที่ถูกสร้างขึ้นจากการแก้ไขแผนภาพลำดับเวอร์ชันใหม่ ซึ่งจะถูกจัดเก็บไว้ในฐานข้อมูลกรณีทดสอบ

1.3.7 การวัดผลของงานวิจัยนี้จะวัดผลจากจำนวนของกรณีทดสอบที่ถูกนำมาใช้ในการทดสอบซอฟต์แวร์หลังการเปลี่ยนแปลงลดลง เมื่อเทียบกับกรณีทดสอบที่ยังสามารถใช้งานได้ทั้งหมดที่อยู่ในฐานข้อมูลกรณีทดสอบ

1.4 ขั้นตอนและวิธีการดำเนินงานวิจัย

- 1) ศึกษาและวิเคราะห์การระบุการเปลี่ยนแปลงของแผนภาพลำดับ
- 2) ศึกษาและวิเคราะห์วิธีการสร้างกรณีทดสอบจากแผนภาพลำดับ
- 3) ศึกษาและวิเคราะห์วิธีการเลือกกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอย
- 4) ออกแบบกระบวนการ วิธีการและเลือกเทคโนโลยีที่จะใช้ในการพัฒนาเครื่องมือทดสอบ
- 5) ออกแบบเครื่องมือที่ใช้ในการอ่านข้อมูลในแผนภาพลำดับ
- 6) ออกแบบเครื่องมือที่ใช้ในการตรวจสอบการเปลี่ยนแปลงของแผนภาพลำดับ
- 7) ออกแบบเครื่องมือที่ใช้ในการสร้างกรณีทดสอบ
- 8) ออกแบบเครื่องมือที่ใช้ในการเลือกกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอย
- 9) พัฒนาระบบตามทีออกแบบและเชื่อมต่อเครื่องมือที่พัฒนาเข้าด้วยกัน
- 10) จัดเตรียมแผนภาพลำดับเพื่อใช้ในการทดสอบ
- 11) ทดสอบเครื่องมือที่พัฒนาขึ้นเพื่อสนับสนุนแนวคิดในงานวิจัยชิ้นนี้
- 12) ตีพิมพ์ผลงานทางวิชาการ
- 13) สรุปผลการวิจัยและข้อเสนอแนะ
- 14) จัดทำรายงานวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1) เครื่องมือที่ใช้ในการเลือกกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอยโดยใช้แผนภาพลำดับ
- 2) ลดเวลา ทรัพยากรและค่าใช้จ่ายในการทดสอบซอฟต์แวร์เมื่อมีการเปลี่ยนแปลง
- 3) เพิ่มประสิทธิภาพในการทดสอบเชิงถดถอย ให้อยู่ในขอบเขตที่มีการเปลี่ยนแปลงเท่านั้น

1.6 ลำดับขั้นตอนในการเสนองานวิจัย

เนื้อหาของวิทยานิพนธ์ฉบับนี้แบ่งออกเป็น 6 บท ดังต่อไปนี้

บทที่ 1 บทนำ อธิบายถึงความเป็นมาและความสำคัญของปัญหาของงานวิจัย วัตถุประสงค์ ขอบเขตงานวิจัย รวมถึงขั้นตอนการออกแบบและวิธีการทำงานวิจัย รวมถึงประโยชน์ที่ผู้วิจัยคาดว่าจะได้รับและกล่าวถึงผลงานที่ได้รับการตีพิมพ์

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้องที่ได้นำมาประยุกต์และอ้างอิงในงานวิจัย

บทที่ 3 การวิเคราะห์และการออกแบบเครื่องมือทดสอบ จะนำเสนอขั้นตอนการทำงานของเครื่องมือทดสอบ การวิเคราะห์และออกแบบเครื่องมือทดสอบ

บทที่ 4 การพัฒนาเครื่องมือทดสอบ กล่าวถึงสภาพแวดล้อมทางฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในการพัฒนาเครื่องมือและหน้าจอในส่วนการต่อประสานกับผู้ใช้งานเครื่องมือทดสอบ

บทที่ 5 การทดสอบเครื่องมือ กล่าวถึงสภาพแวดล้อมทางฮาร์ดแวร์และซอฟต์แวร์ที่ใช้ในการทดสอบ ขั้นตอนการทดสอบเครื่องมือทดสอบ แผนภาพที่ใช้ในการทดสอบ และผลการทดสอบ

บทที่ 6 สรุปผลการทดสอบและผลการวิจัย ข้อจำกัดที่มีของเครื่องมือทดสอบ ข้อเสนอแนะ และแนวทางการวิจัยในอนาคต

1.7 ผลงานตีพิมพ์

ผลของงานวิจัยฉบับนี้ได้รับการพิจารณาและตีพิมพ์และเผยแพร่เป็นบทความทางวิชาการในบทความชื่อ “Safe Regression Test Selection from Sequence Diagrams” โดยผู้จัดทำคือ Warin Tappong และ Taratip Suwannasart และได้ถูกคัดเลือกให้ได้นำเสนอผลงานในงานประชุมทางวิชาการชื่อ International Conference on Manufacturing Science and Information Engineering (ICMSIE 2013) และตีพิมพ์เพื่อเผยแพร่ในเอกสารทางวิชาการชื่อ International Conference on Industrial Engineering and Management Science-2013 ซึ่งจัดทำโดย DEStech Publications, Inc. ในระหว่างวันที่ 28-29 กันยายน พ.ศ. 2556 ณ เมืองเซี่ยงไฮ้ สาธารณรัฐประชาชนจีน

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้องสำหรับงานวิจัยนี้ มีรายละเอียดดังต่อไปนี้

2.1.1 การเลือกกรณีทดสอบสำหรับการทดสอบเชิงถดถอย [15]

กลวิธีในการเลือกกรณีทดสอบสำหรับการทดสอบเชิงถดถอยเป็นการนำกรณีทดสอบที่มีอยู่แล้วกลับมาทดสอบกับโปรแกรมเวอร์ชันใหม่ โดยหนึ่งในกลยุทธ์ของการทดสอบเชิงถดถอยคือการทดสอบโปรแกรมเวอร์ชันใหม่ด้วยกรณีทดสอบทั้งหมด แต่วิธีการดังกล่าวจะต้องใช้เวลา และทรัพยากรมากในการทดสอบ [5]

งานวิจัยที่ผ่านมาได้นำเสนอวิธีการเลือกกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอยซึ่งสามารถแบ่งออกได้เป็น 5 ประเภท ดังต่อไปนี้ [15]

1) การเลือกกรณีทดสอบให้น้อยที่สุด (Minimization Techniques)

กรณีทดสอบที่ถูกเลือกจะต้องครอบคลุมถึงการเปลี่ยนแปลงจากโปรแกรมเวอร์ชันปัจจุบัน ซึ่งจะต้องเลือกกรณีทดสอบอย่างน้อย 1 กรณีทดสอบที่นำไปใช้ในการทดสอบในโปรแกรมเวอร์ชันใหม่

2) การเลือกกรณีทดสอบแบบปลอดภัย (Safe Techniques)

มุ่งเน้นไปที่การทดสอบเพื่อหาข้อผิดพลาดกับโปรแกรมเวอร์ชันใหม่ โดยการเลือกกรณีทดสอบที่อยู่ในชุดทดสอบทุกกรณีที่สามารถนำไปใช้ทดสอบกับโปรแกรมเวอร์ชันปัจจุบัน ที่จะต้องทดสอบอย่างน้อย 1 คำสั่งของการเพิ่ม ลบออก หรือถูกแก้ไขในโปรแกรมเวอร์ชันใหม่

3) การเลือกกรณีทดสอบโดยครอบคลุมการไหลของข้อมูล (Dataflow-Coverage-Based Techniques)

เลือกกรณีทดสอบทุกกรณีทดสอบที่ทดสอบการนิยามการใช้ (Definition Used) ที่หายไป หรือเปลี่ยนแปลงอย่างน้อย 1 คู่ในโปรแกรมเวอร์ชันใหม่

4) การเลือกกรณีทดสอบแบบเฉพาะหรือสุ่ม (Ad Hoc/Random Techniques)

กรณีทดสอบที่จะถูกเลือกสำหรับทดสอบกับโปรแกรมเวอร์ชันใหม่จะถูกเลือกจากประสบการณ์ของผู้ทดสอบ สามัญสำนึกหรือสุ่มเลือก

5) การเลือกกรณีแบบทดสอบทั้งหมด (Retest-All Technique)

เป็นการเลือกกรณีทดสอบที่มีอยู่ทั้งหมดกลับมาทำการทดสอบกับโปรแกรมเวอร์ชันใหม่

งานวิจัยนี้จะพิจารณาเลือกใช้การเลือกกรณีทดสอบแบบปลอดภัย เนื่องจากว่าการทดสอบแบบปลอดภัยนั้นจะทำการเลือกกรณีทดสอบสำหรับส่วนที่ได้รับผลกระทบจากการเปลี่ยนแปลงแก้ไขโปรแกรมเวอร์ชันใหม่ทั้งหมดมาใช้ในการทดสอบใหม่ ซึ่งจะรวมถึงการเพิ่มเติมฟังก์ชันงานใหม่และการตัดฟังก์ชันงานเดิมออกด้วย [16] ดังนั้นจึงมั่นใจได้ว่าการเปลี่ยนแปลงแก้ไขจากโปรแกรมเวอร์ชันใหม่นั้นจะถูกทดสอบทั้งหมด เพื่อให้มั่นใจได้ว่าการเปลี่ยนแปลงแก้ไขเหล่านั้นจะไม่ส่งผลกระทบต่อหรือสร้างข้อผิดพลาดขึ้นกับฟังก์ชันงานเดิมที่มีอยู่

2.1.2 การทดสอบเชิงถดถอยแบบปลอดภัย [15]

การทดสอบเชิงถดถอยแบบปลอดภัย คือ กระบวนการตรวจสอบซอฟต์แวร์ที่ได้ถูกแก้ไข เพื่อที่จะตรวจสอบว่ามีข้อผิดพลาดที่จะเกิดขึ้นกับซอฟต์แวร์เวอร์ชันปัจจุบันที่เคยถูกทดสอบไปแล้วหรือไม่ และเพื่อให้มั่นใจได้ว่าการแก้ไขนั้น ถูกต้องและ ไม่ทำให้เกิดผลกระทบข้างเคียงระบบงานส่วนอื่นๆในซอฟต์แวร์ที่ไม่ได้มีการแก้ไข [15]

ในกิจกรรมการบำรุงรักษาซอฟต์แวร์นั้นจะรวมถึงการเพิ่มเติมความสามารถของซอฟต์แวร์ การแก้ไขจุดบกพร่อง การปรับสมรรถภาพ และการตัดความสามารถที่ไม่ได้ถูกใช้ออกไป ทั้งหมดนี้จะถือเป็นการแก้ไขซอฟต์แวร์ซึ่งอาจจะส่งผลให้ซอฟต์แวร์ที่ได้มีการแก้ไขนั้นทำงานผิดพลาดและทำให้เกิดผลกระทบต่อซอฟต์แวร์ในส่วนอื่นๆอีกด้วย [17]

กระบวนการทดสอบเชิงถดถอยแบบปลอดภัย [15] สามารถอธิบายได้ดังรูปที่ 2.1

Let P be a procedure or program, let P' be a modified version of P , and let T be a set of tests (a *test suite*) created to test P . A typical selective retest technique proceeds as follows:

- 1) Select $T' \subseteq T$, a set of tests to execute on P' .
- 2) Test P' with T' , establishing the correctness of P' with respect to T' .
- 3) If necessary, create T'' , a set of new functional or structural tests for P' .
- 4) Test P' with T'' , establishing the correctness of P' with respect to T'' .
- 5) Create T''' , a new test suite and test history for P' , from T , T' , and T'' .

รูปที่ 2.1 กระบวนการทดสอบเชิงถดถอยแบบปลอดภัย [15]

กำหนดให้ P เป็นกระบวนการงานหรือโปรแกรม กำหนดให้ P' เป็นโปรแกรมเวอร์ชันใหม่ของโปรแกรม P และกำหนดให้ T เป็นชุดทดสอบของ P โดยที่กระบวนการทดสอบเชิงถดถอยแบบปลอดภัยจะอธิบายได้ดังต่อไปนี้

ขั้นตอนที่ 1 เลือกกรณีทดสอบที่ได้รับผลกระทบ (T') จากชุดทดสอบทั้งหมด (T) เพื่อนำมาใช้ในการทดสอบกับโปรแกรมเวอร์ชันใหม่

ขั้นตอนที่ 2 ทำการทดสอบโปรแกรมเวอร์ชันใหม่ด้วยกรณีทดสอบที่ได้รับผลกระทบ เพื่อตรวจสอบความถูกต้องของโปรแกรมเวอร์ชันใหม่

ขั้นตอนที่ 3 สร้างชุดทดสอบเชิงถดถอย (T'') ที่ใช้สำหรับทดสอบฟังก์ชันหรือโครงสร้างใหม่สำหรับโปรแกรมเวอร์ชันใหม่

ขั้นตอนที่ 4 ทดสอบโปรแกรมเวอร์ชันใหม่ด้วยชุดทดสอบเชิงถดถอย เพื่อตรวจสอบความถูกต้องของโปรแกรมเวอร์ชันใหม่

ขั้นตอนที่ 5 สร้างชุดทดสอบฉบับสมบูรณ์ (T''') สำหรับนำไปใช้ทดสอบโปรแกรมเวอร์ชันใหม่ โดยใช้ชุดทดสอบเชิงถดถอยจากชุดทดสอบทั้งหมด (T, T', T'')

2.1.3 การวิเคราะห์ผลกระทบการเปลี่ยนแปลงของยูเอ็มแอลสำหรับการเลือกกรณีทดสอบเชิงถดถอย [13]

การเลือกกรณีทดสอบเชิงถดถอยโดยทั่วไปสามารถวิเคราะห์จากการแก้ไขรหัสต้นฉบับ การไหลของตัวควบคุม (Control Flow) และการไหลของข้อมูล (Data Flow) ซึ่งวิธีที่กล่าวมานี้จะจำเป็นจะต้องถูกพัฒนาไปเรียบร้อยแล้วถึงจะสามารถวิเคราะห์ถึงผลกระทบการเปลี่ยนแปลงได้ แต่จะเป็นวิธีที่แม่นยำวิธีหนึ่ง ในเชิงของการเลือกกรณีทดสอบเชิงถดถอยแบบนี้ที่สุด การเปลี่ยนแปลงที่ระดับของรหัสต้นฉบับนั้นจะสามารถระบุถึงผลกระทบที่อยู่ในกรณีทดสอบได้

อีกทางเลือกในการพิจารณาเลือกกรณีทดสอบเชิงถดถอยนั้นจะเป็นการใช้ข้อมูลที่อยู่ในระดับของสถาปัตยกรรมหรือการออกแบบมาใช้เป็นปัจจัย ในกรณีนี้ผลกระทบที่เป็นไปได้ทั้งหมดจะถูกประเมินจากการออกแบบของเวอร์ชันใหม่ โดยการใช้วิธีเปรียบเทียบระหว่างการเปลี่ยนแปลงที่เกิดขึ้นในการออกแบบใหม่กับการออกแบบเดิม

โดยการวิเคราะห์ถึงผลกระทบที่มีต่อกรณีทดสอบเดิมนั้นจะต้องมีความสามารถในการตรวจสอบย้อนกลับ (Traceability) ระหว่างการออกแบบและกรณีทดสอบ ซึ่งจากผลกระทบของการเปลี่ยนแปลงระดับการออกแบบจะเป็นตัวกำหนดว่ากรณีทดสอบใดบ้างที่จำเป็นจะต้องถูกนำมาทดสอบอีกครั้งและกรณีทดสอบไหนบ้างที่ควรจะถูกลบออกจากชุดทดสอบเชิงถดถอย เพราะไม่สามารถใช้งานได้แล้วกับโปรแกรมเวอร์ชันใหม่ โดยที่ประโยชน์ของวิธีการนี้คือจะช่วยให้การเลือกชุดทดสอบเชิงถดถอยนั้นมีประสิทธิภาพมากยิ่งขึ้นและสามารถที่จะวางแผนสำหรับการทดสอบและการประเมินทรัพยากรที่จะใช้ในการทดสอบในเบื้องต้นได้ก่อนที่จะทำการพัฒนา

การวิเคราะห์ผลกระทบการเปลี่ยนแปลงของยูเอ็มแอลสำหรับการเลือกกรณีทดสอบเชิงถดถอย จะถูกพิจารณาประเภทของการเปลี่ยนแปลง ซึ่งจะแยกออกได้เป็น 3 ประเภทของแผนภาพ

ยูเอ็มแอล คือ แผนภาพคลาส (Class Diagram) แผนภาพลำดับและแผนภาพยูสเคส (Use Case Diagram) [13]

การเปลี่ยนแปลงที่พิจารณาได้ทั้งหมดจะสามารถอธิบายได้ดังต่อไปนี้

1) การเปลี่ยนแปลงระหว่างสองเวอร์ชันของแผนภาพคลาส จะสามารถนิยามการเปลี่ยนแปลงของแผนภาพคลาสได้ ดังต่อไปนี้

- 1) การเพิ่ม ลบและแก้ไขลักษณะประจำ (Attribute)
- 2) การเพิ่ม ลบและแก้ไขเมทอด (Method)
- 3) การเพิ่ม ลบและแก้ไขความสัมพันธ์ (Relationship)
- 4) การเพิ่ม ลบและแก้ไขคลาส

2) การเปลี่ยนแปลงระหว่างสองเวอร์ชันของแผนภาพลำดับ เนื่องจากแผนภาพยูสเคสนั้น จะถูกใช้ในการอธิบายในแผนภาพลำดับ ดังนั้นจะสามารถพิจารณาให้อยู่ในการเปลี่ยนแปลงของแผนภาพลำดับได้โดยแผนภาพยูสเคสและแผนภาพลำดับจะมีนิยามการเปลี่ยนแปลงได้ดังต่อไปนี้

- 1) การเพิ่ม ลบและแก้ไขยูสเคส
- 2) การเพิ่ม ลบและแก้ไขเมทอดในแผนภาพลำดับ ซึ่งเมทอดในที่นี้จะหมายถึงเมทอด

ซิกเนเจอร์ที่เป็นนิยามเดียวกับเมทอดที่อยู่ในแผนภาพคลาส

ซึ่งการวิเคราะห์ถึงผลกระทบการเปลี่ยนแปลงของยูเอ็มแอล ในส่วนของแผนภาพลำดับนั้น สามารถที่นำไปประยุกต์ใช้ในงานวิจัยนี้ได้ ซึ่งจะนำผลกระทบของการเปลี่ยนแปลงของแผนภาพลำดับไปใช้ในการเลือกกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอยต่อไป

2.1.4 อัลกอริทึม Levenshtien Edit Distance [18]

อัลกอริทึม Levenshtien Edit Distance เป็นอัลกอริทึมที่ใช้ในการหาระยะของการแก้ไขระหว่างจำนวนของสตริง 2 สตริง โดยผลลัพธ์ที่ได้ คือ จำนวนของระยะการแก้ไขที่น้อยที่สุด ซึ่งการแก้ไขหมายถึงการเพิ่ม การลบออกและการแทนที่ โดยใช้ในการเปรียบเทียบจากข้อมูลของสตริงทั้ง 2 ชุดไปที่ละตัว

ซึ่งการหาความแตกต่างระหว่างข้อมูลของสตริง 2 ชุด จะแสดงในรูปของรหัสเทียม (Pseudo code) ดังรูปที่ 2.2

```

int LevenshteinDistance(char s[1..m], char t[1..n])
// dL is a table with m+1 rows and n+1 columns
declare int dL[0..m, 0..n]

for i from 0 to m
    dL[i, 0] := i
for j from 1 to n
    dL[0, j] := j

for i from 1 to m
    for j from 1 to n
        if s[i] = t[j] then cost := 0
            else cost := 1
        dL[i, j] := minimum(
            dL[i-1, j] + 1, // deletion
            dL[i, j-1] + 1, // insertion
            dL[i-1, j-1] + cost // substitution
        )

return dL[m, n]

```

รูปที่ 2.2 รหัสเทียมของอัลกอริทึม Levenshtein Edit Distance [18]

จากรหัสเทียมจะแสดงให้เห็นว่าจะมีการนำข้อมูลของสตริงทั้ง 2 ชุดมาสร้างเป็นตาราง 2 มิติ โดยให้ข้อมูลชุดแรก แทนด้วย m และข้อมูลในชุดที่สอง แทนด้วย n จากนั้นจึงจะทำการนำข้อมูลของ m และ n มาทำการเปรียบเทียบเพื่อหาระยะการแก้ไข โดยในรูปที่ 2.3 จะแสดงตารางของการเปรียบเทียบระหว่างข้อมูลของสตริง 2 ชุด ที่อยู่ในรูปแบบของประโยคภาษาอังกฤษ ซึ่งประโยคแรกคือ I am old และประโยคที่สอง คือ I was cold

		I	a	m		o	l	d	
0	0	1	2	3	4	5	6	7	8
I	1	0	1	2	3	4	5	6	7
	2	1	0	1	2	3	4	5	6
w	3	2	1	1	2	3	4	5	6
a	4	3	2	1	2	3	4	5	6
s	5	4	3	2	2	3	4	5	6
	6	5	4	3	3	2	3	4	5
c	7	6	5	4	4	3	3	4	5
o	8	7	6	5	5	4	3	4	5
l	9	8	7	6	6	5	4	3	4
d	10	9	8	7	7	6	5	4	3

รูปที่ 2.3 รูปตารางผลลัพธ์ของการคำนวณระยะแก้ไขจากข้อมูลของสตริง 2 ชุด [18]

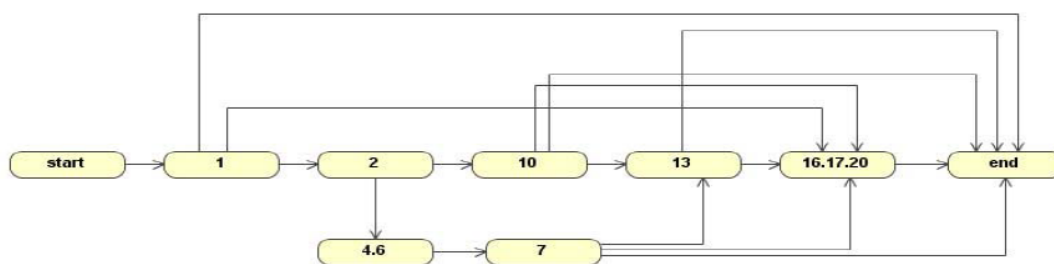
จากผลลัพธ์ของการหาระยะแก้ไขจากอัลกอริทึมนี้แสดงให้เห็นว่าระยะของการแก้ไขจะอยู่ที่จำนวน 3 ซึ่งหมายความว่า ระหว่างข้อมูลของสตริงของทั้ง 2 ประโยคนี้ มีการแก้ไขทั้งหมดจำนวน 3 จุดด้วยกัน ซึ่งก็คือ มีการเพิ่ม w มีการแทนที่ระหว่าง m กับ s และมีการตัด c ออก

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 Test Sequence Generation from UML Sequence Diagram [7]

งานวิจัยนี้นำเสนอวิธีการสร้างลำดับการทดสอบที่เป็นไปได้ทั้งหมดจากแผนภาพลำดับ โดย การสร้างลำดับการทดสอบนี้มีจุดประสงค์ที่จะนำไปใช้สำหรับการตรวจสอบความถูกต้องของการ พัฒนาระบบ โดยผลลัพธ์ที่ได้จะเป็นเส้นทางที่เป็นไปได้ทั้งหมดที่ครอบคลุมเพื่อใช้สำหรับการทดสอบ ระบบ

งานวิจัยนี้ใช้วิธีการแปลงข้อมูลที่อยู่ในแผนภาพลำดับให้เป็น กราฟลำดับความสัมพันธ์ (SDG: Sequence Dependency Graph) ดังที่แสดงในรูปที่ 2.4



รูปที่ 2.4 กราฟลำดับความสัมพันธ์จากแผนภาพลำดับ [7]

ซึ่งกราฟลำดับความสัมพันธ์จะใช้แสดงถึงความสัมพันธ์ของการสื่อสารระหว่างอ็อบเจกต์ที่อยู่ในแผนภาพลำดับ จากนั้นจะทำการใช้ อัลกอริทึม Traversal เข้ามาช่วยในการสร้างลำดับการทดสอบจากกราฟลำดับความสัมพันธ์ ดังที่แสดงในรูปที่ 2.5


```

Start=current;
Traverse (nodeArray [], current)
{
  nodeArray [] =Add (nodeArray[ ], current);
  if (current ==end)
    {Print (nodeArray [ ]);}

  else
    for each current.next
      Traverse (nodeArray [], current.next)
}

```

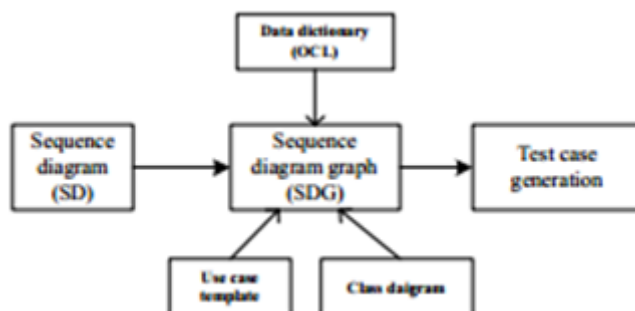
รูปที่ 2.5 รหัสเทียมของอัลกอริทึม Traversal [7]



Test sequence#1	1
Test sequence#2	1.(16.17.20)
Test sequence#3	1.2.(4.6).7
Test sequence#4	1.2. (4.6).7.13
Test sequence#5	1.2. (4.6).7.13. (16.17.20)
Test sequence#6	1.2. (4.6).7. (16.17.20)
Test sequence#7	1.2.10.13. (16.17.20)
Test sequence#8	1.2.10.13
Test sequence#9	1.2.10. (16.17.20)
Test sequence#10	1.2.10

รูปที่ 2.6 ผลลัพธ์ของการสร้างลำดับการทดสอบจากแผนภาพลำดับ [7]

รูปที่ 2.6 จะเป็นผลลัพธ์ของลำดับของการทดสอบที่เป็นไปได้ทั้งหมดในแผนภาพลำดับงานวิจัยนี้ผู้วิจัยสังเกตเห็นประโยชน์ในการสร้างลำดับการทดสอบเพื่อนำมาประยุกต์ใช้ในการสร้างกรณีทดสอบให้ครอบคลุมจากแผนภาพลำดับได้



รูปที่ 2.9 ขั้นตอนการนำข้อมูลที่จำเป็นมารวมกันเพื่อใช้ในการสร้างกรณีทดสอบ [14]

จากนั้นจะนำข้อมูลที่อยู่ในกราฟทดสอบระบบไปผ่านขั้นตอนวิธีการสร้างกรณีทดสอบ ซึ่งจะ
ทำให้ได้กรณีทดสอบที่แสดงดังรูปที่ 2.10

```

1. Test name = "ATM PIN Validation"
2. Preconditions: ATM is idle and displaying a welcome message. User inserts a card
3. Test case: Scenario 1
4. Input: Card = "Not ATM"
5. Output: Eject card
6. Postcondition: Displays welcome message
7. Test case: Scenario 2
8. Input: Card = "ATM", Status = "Stolen"
9. Output: Eject card
10. Postcondition: Back to the initial state
11. Test case: Scenario 3
12. Input: Card = "ATM", Status = "Okay", Account = "Close"
13. Output: Eject card
14. Postcondition: Displays welcome message
15. Test case: Scenario 4
16. Input: Card = "ATM", Status = "Okay", Account = "Open", PIN = "Invalid"
17. Output: Message "Invalid PIN: Try Again"
18. Postcondition: Displays welcome message
19. Input: Card = "ATM", Status = "Okay", Account = "Open", PIN = "Invalid"
20. Output: Message "Invalid PIN: Try Again"
21. Postcondition: Displays welcome message
22. Input: Card = "ATM", Status = "Okay", Account = "Open", PIN = "Invalid"
23. Output: Message "Invalid PIN: Try Again"
24. Postcondition: Displays welcome message
25. Input: Card = "ATM", Status = "Okay", Account = "Open". PIN = "Invalid", Try = <4>
26. Output: Message "Invalid PIN: Try Later"
    Eject card
27. Postcondition: Displays welcome message
28. Test case: Scenario 5
29. Input: Card = "ATM", Status = "Okay", Account = "Open", PIN = "Valid"
30. Output: Display "Hello"
31. Postcondition: Eject card
32. Postcondition: Display menu for transaction
  
```

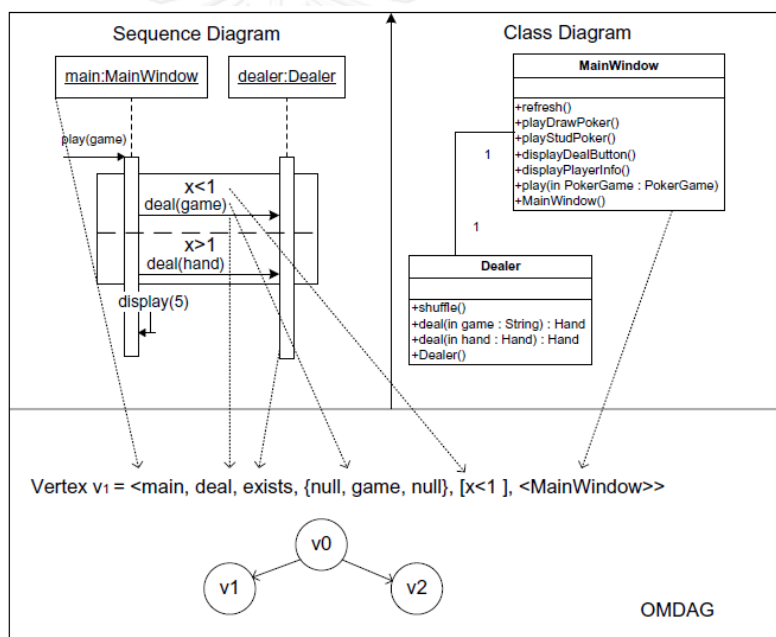
รูปที่ 2.10 ผลลัพธ์ของกรณีทดสอบที่ถูกสร้างขึ้นมาจากงานวิจัยนี้ [14]

ในขั้นตอนการแปลงแผนภาพลำดับไปเป็นกราฟแผนภาพลำดับ ผู้วิจัยได้สังเกตเห็นว่าโครงสร้าง
ในการนิยามของการแปลงแผนภาพลำดับนั้น นอกจากจะนำไปใช้ในการสร้างกรณีทดสอบแล้วยัง
สามารถที่จะนำไปประยุกต์ใช้ในการระบุการเปลี่ยนแปลงของแผนภาพลำดับเพื่อหาผลกระทบของ
การเปลี่ยนแปลงได้ด้วย โดยการประยุกต์ใช้นิยามของการแปลงแผนภาพลำดับไปเป็นกราฟแผนภาพ
ลำดับที่งานวิจัยนี้ได้นำเสนอ

2.2.3 Regression Testing UML Design [4]

งานวิจัยนี้นำเสนอวิธีการทดสอบเชิงถดถอยจากแผนภาพยูเอ็มแอล โดยจะใช้ข้อมูลจากแผนภาพคลาส แผนภาพลำดับและโอซีแอล ในการสร้างกรณีทดสอบสำหรับการตรวจสอบการตรงกันของข้อมูลในขั้นตอนการออกแบบ โดยจะมีการจัดหมวดหมู่ของการเปลี่ยนแปลงและการเลือกใช้งานกรณีทดสอบเดิมที่มีอยู่แล้วหรือการสร้างกรณีทดสอบใหม่ขึ้นมาสำหรับฟังก์ชันงานที่ถูกเพิ่มเข้ามาใหม่ เพื่อจะทำให้แน่ใจได้ว่าการเปลี่ยนแปลงนั้นจะได้รับการทดสอบอย่างครอบคลุมและครบถ้วน โดยใช้เทคนิคการเลือกกรณีทดสอบสำหรับการทดสอบเชิงถดถอยแบบปลอดภัย โดยจะมีขั้นตอนการทำงานดังนี้

1) ทำการแปลงข้อมูลของแผนภาพลำดับกับแผนภาพคลาสให้อยู่ในรูปแบบของ กราฟระบุทิศทางที่ไม่มีวัฏจักรของอ็อบเจกต์เมทอด (OMDAG: Object Method Directed Acyclic Graph) ดังรูปที่ 2.11



รูปที่ 2.11 แสดงถึงการนำข้อมูลของแผนภาพลำดับกับแผนภาพคลาสมารวมกัน [4]

- 2) ใช้การจัดลำดับแบบทอพอโลยี (Topological Sort) เพื่อให้ทราบถึงการเปลี่ยนแปลง
- 3) ทำการระบุการเปลี่ยนแปลงและจำแนกประเภทของการเปลี่ยนแปลง
- 4) ทำการเชื่อมโยงการเปลี่ยนแปลงของการออกแบบกับกราฟระบุทิศทางที่ไม่มีวัฏจักรของอ็อบเจกต์เมทอด
- 5) ทำการจัดประเภทของกรณีทดสอบ มีทั้งหมดด้วยกัน 4 ประเภทคือ
 - 1) กรณีทดสอบที่ถูกสร้างใหม่

- 2) กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว
- 3) กรณีทดสอบที่นำกลับมาทดสอบใหม่
- 4) กรณีทดสอบที่ยังใช้ได้ทั้งหมด
- 6) ทำการเลือกกรณีทดสอบจากประเภทของการเปลี่ยนแปลงที่ได้รับผลกระทบ

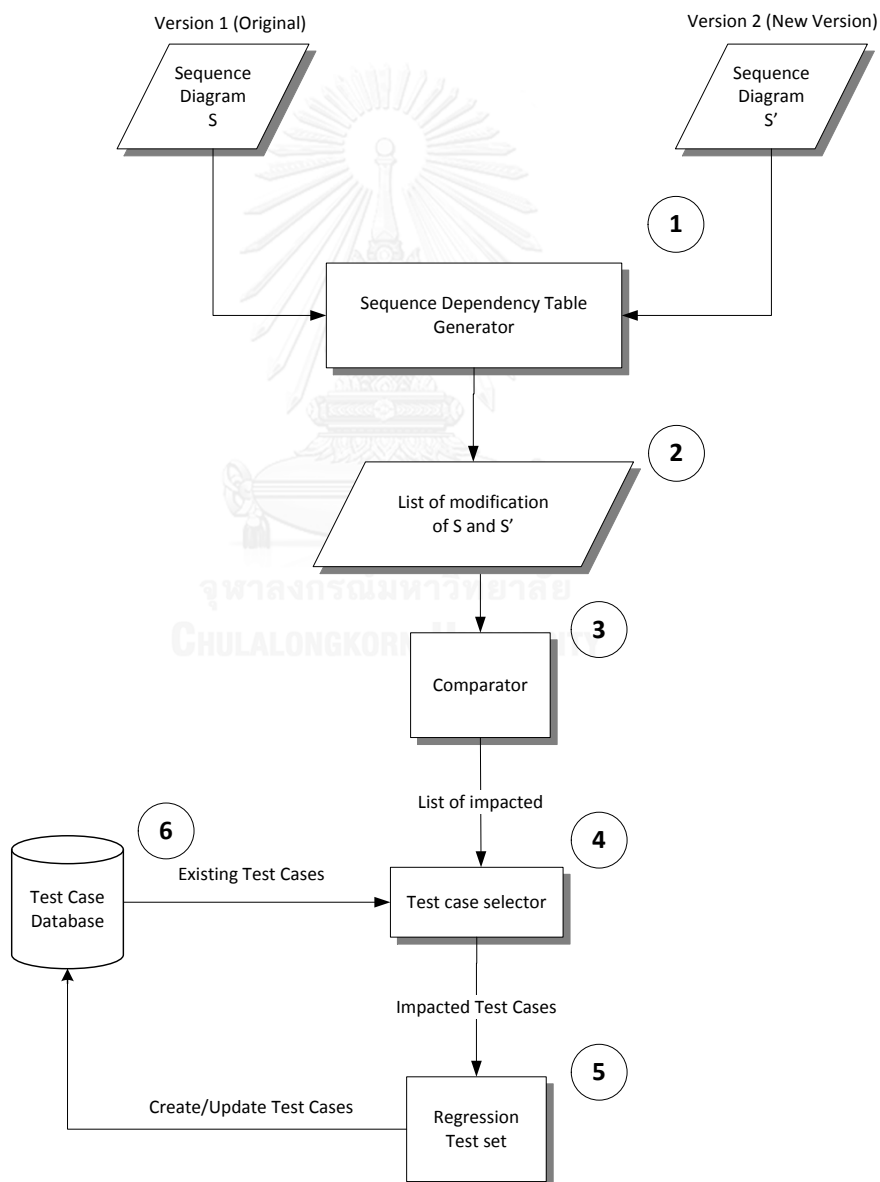
จากงานวิจัยนี้ ผู้วิจัยได้สังเกตเห็นถึงประโยชน์ของการจัดประเภทกรณีทดสอบที่มีความชัดเจน จึงจะนำประเภทของกรณีทดสอบทั้งหมด 4 ประเภทมาใช้ในการแสดงรายงานของงานวิจัยชิ้นนี้



บทที่ 3 การวิเคราะห์และออกแบบเครื่องมือ

3.1 ภาพรวมการทำงานของเครื่องมือทดสอบ

งานวิจัยนี้ต้องการนำเสนอเครื่องมือที่ใช้ในการเลือกกรณีทดสอบ สำหรับนำมาใช้เป็นชุดทดสอบในการทดสอบเชิงถดถอย โดยใช้ข้อมูลของผลกระทบของการเปลี่ยนจากแผนภาพภาพลำดับมาเป็นปัจจัยในการเลือกชุดทดสอบเชิงถดถอย ภาพรวมของเครื่องมือสามารถแสดงได้ดังรูปที่ 3.1

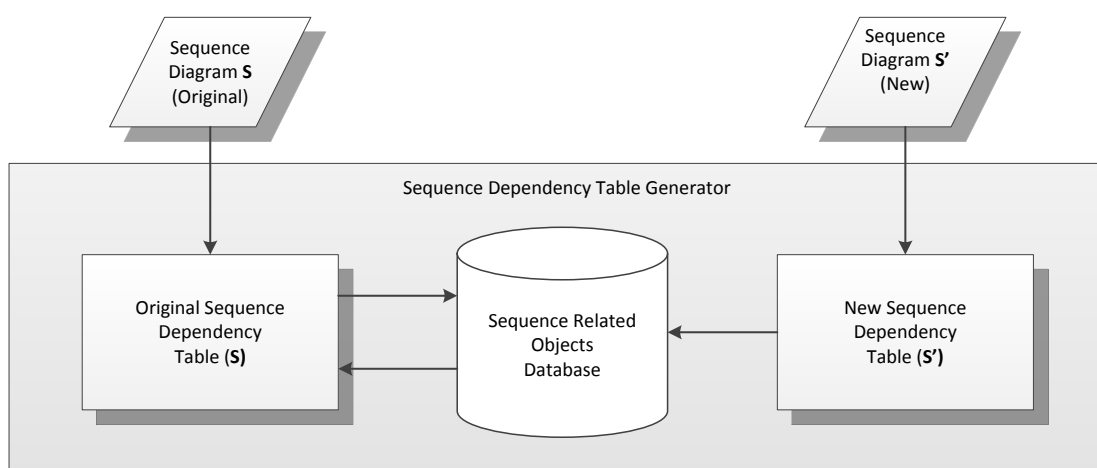


รูปที่ 3.1 ภาพรวมของระบบการเลือกชุดกรณีทดสอบเชิงถดถอย

จากรูปที่ 3.1 ระบบมีส่วนประกอบสำคัญอยู่ทั้งหมด 6 ส่วน ได้แก่

3.1.1 ส่วนการสร้างตารางความสัมพันธ์ (Sequence Dependency Table Generator)

ส่วนการสร้างตารางความสัมพันธ์มีหน้าที่ในการสร้างตารางลำดับความสัมพันธ์ของแผนภาพลำดับที่สามารถแสดงให้เห็นถึงการทำงานและความสัมพันธ์ของแผนภาพลำดับในแต่ละเวอร์ชัน โดยมีส่วนประกอบในการทำงานดังรูปที่ 3.2



รูปที่ 3.2 ส่วนประกอบของส่วนการสร้างตารางความสัมพันธ์

3.1.1.1 ฐานข้อมูลความสัมพันธ์ของอ็อบเจกต์ (Sequence Related Objects Database)

ฐานข้อมูลความสัมพันธ์ของอ็อบเจกต์มีหน้าที่ในการจัดเก็บข้อมูลของตารางลำดับความสัมพันธ์ของแผนภาพลำดับที่สามารถแสดงให้เห็นถึงการทำงานและความสัมพันธ์ของแผนภาพลำดับในแต่ละเวอร์ชัน

ข้อมูลของตารางความสัมพันธ์ที่จะถูกนำมาใช้งานนั้นจะถูกสร้างโดยการอ่านข้อมูลที่อยู่ในแผนภาพลำดับที่ถูกออกแบบจากโปรแกรมสำหรับออกแบบยูเอ็มแอลชื่อโปรแกรมวิซวลพาราไดม์ และทำการนำออกมาโดยข้อมูลของแผนภาพลำดับนั้นจะถูกสร้างอยู่ในรูปแบบของไฟล์เอกซ์เอ็มแอล โดยเครื่องมือทดสอบจะเริ่มจากการอ่านข้อมูลของแผนภาพลำดับจากตำแหน่งที่อยู่ด้านบนลงปาด้านล่างและเริ่มจากทางด้านซ้ายไปทางด้านขวาและจึงทำการสร้างตารางความสัมพันธ์ของแผนภาพลำดับในแต่ละเวอร์ชันและนำไปจัดเก็บไว้ในฐานข้อมูลความสัมพันธ์ของอ็อบเจกต์

```

<?xml version="1.0" encoding="UTF-8"?>
<Project Author="warint" CommentTableSortAscending="false" CommentTableSortColumn="Date Time" Description="&lt;html&
<ProjectInfo>
<Models>
  <ModelChildren>
    <DataType Documentation_plain="" Id="90QLAsKD.AAAAQdK" Name="boolean" PmAuthor="BiG" PmCreateDateTime="2013-12-12" Documentation_plain="" />
    <DataType Documentation_plain="" Id="oMQLAsKD.AAAAQdL" Name="byte" PmAuthor="BiG" PmCreateDateTime="2013-12-12" Documentation_plain="" />
    <DataType Documentation_plain="" Id="YMQLAsKD.AAAAQdM" Name="char" PmAuthor="BiG" PmCreateDateTime="2013-12-12" Documentation_plain="" />
    <DataType Documentation_plain="" Id="EMQLAsKD.AAAAQdN" Name="double" PmAuthor="BiG" PmCreateDateTime="2013-12-12" Documentation_plain="" />
    <DataType Documentation_plain="" Id="kMQLAsKD.AAAAQdO" Name="float" PmAuthor="BiG" PmCreateDateTime="2013-12-12" Documentation_plain="" />
    <DataType Documentation_plain="" Id="UMQLAsKD.AAAAQdP" Name="int" PmAuthor="BiG" PmCreateDateTime="2013-12-31" Documentation_plain="" />
    <DataType Documentation_plain="" Id="OMQLAsKD.AAAAQdQ" Name="long" PmAuthor="BiG" PmCreateDateTime="2013-12-12" Documentation_plain="" />
    <DataType Documentation_plain="" Id="MMQLAsKD.AAAAQdR" Name="short" PmAuthor="BiG" PmCreateDateTime="2013-12-12" Documentation_plain="" />
    <DataType Documentation_plain="" Id="sMQLAsKD.AAAAQdS" Name="void" PmAuthor="BiG" PmCreateDateTime="2013-12-12" Documentation_plain="" />
    <DataType Documentation_plain="" Id="cMQLAsKD.AAAAQdT" Name="string" PmAuthor="BiG" PmCreateDateTime="2013-12-12" Documentation_plain="" />
    <Frame BaseY="40" Documentation_plain="" Id="gGKLAsKD.AAAAQk" Name="Original" PmAuthor="BiG" PmCreateDateTime="2013-12-12" Documentation_plain="" />
  </ModelChildren>
  <InteractionLifeLine Active="true" Class="false" ClassLevelStereotype="false" Documentation_plain="" />
  <InteractionLifeLine Active="true" Class="false" ClassLevelStereotype="false" Documentation_plain="" />
  <InteractionLifeLine Active="true" Class="false" ClassLevelStereotype="false" Documentation_plain="" />
  <NOTE Documentation="&lt;head&gt;&#13;&#10; &lt;style type="&quot;text/css&quot;&gt;&lt;/style&gt;&lt;/head&gt;&lt;/NOTE&gt;" />
  <NOTE Documentation="&lt;head&gt;&#13;&#10; &lt;style type="&quot;text/css&quot;&gt;&lt;/style&gt;&lt;/head&gt;&lt;/NOTE&gt;" />
  <NOTE Documentation="&lt;head&gt;&#13;&#10; &lt;style type="&quot;text/css&quot;&gt;&lt;/style&gt;&lt;/head&gt;&lt;/NOTE&gt;" />
  <InteractionLifeLine Active="true" Class="false" ClassLevelStereotype="false" Documentation_plain="" />
  <InteractionLifeLine Active="true" Class="false" ClassLevelStereotype="false" Documentation_plain="" />
  <NOTE Documentation="&lt;head&gt;&#13;&#10; &lt;style type="&quot;text/css&quot;&gt;&lt;/style&gt;&lt;/head&gt;&lt;/NOTE&gt;" />
  <NOTE Documentation="&lt;head&gt;&#13;&#10; &lt;style type="&quot;text/css&quot;&gt;&lt;/style&gt;&lt;/head&gt;&lt;/NOTE&gt;" />
  <CombinedFragment Documentation_plain="" Id="b038gsKD.AAAAQ1W" Name="CombinedFragment" OperatorKind="AND" />
  </ModelChildren>
</Frame>
<ModelRelationshipContainer Documentation_plain="" Id="FoFHAsKD.AAAAQ1f" Name="relationships" PmAuthor="BiG" />
  <ModelChildren>
  </ModelChildren>
</ModelRelationshipContainer>
</Models>
</Diagrams>
</Project>

```

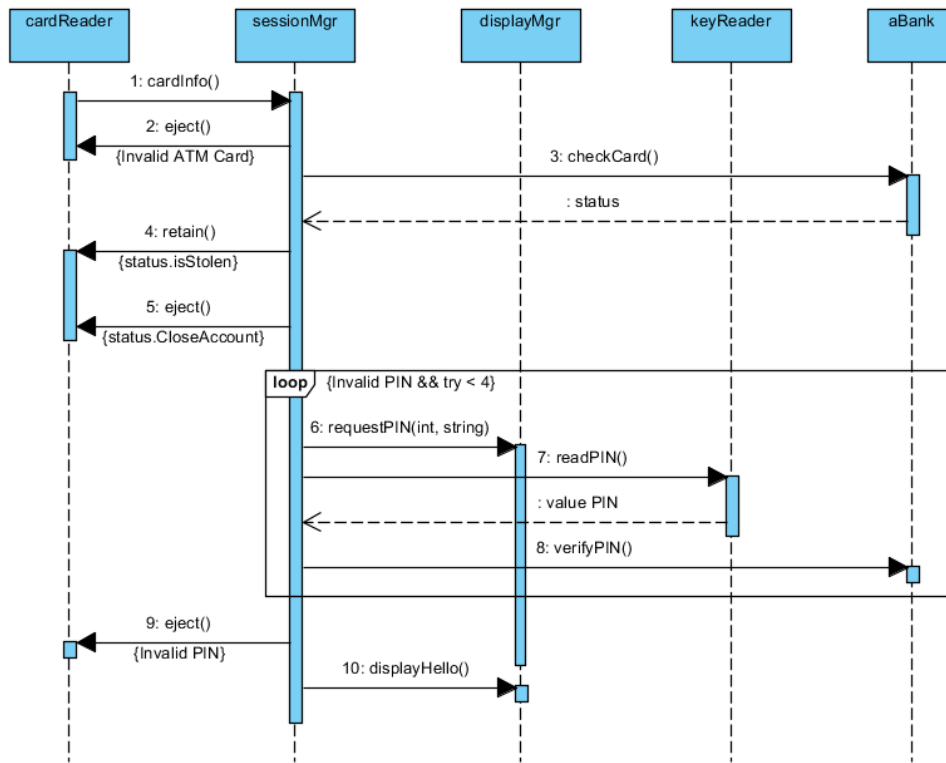
รูปที่ 3.3 โครงสร้างของไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับ

รูปที่ 3.3 คือ ตัวอย่างข้อมูลของแผนภาพลำดับที่ถูกเก็บไว้รูปแบบโครงสร้างแบบเอกซ์เอ็มแอล โดยจะมีข้อมูลความสัมพันธ์ของแผนภาพลำดับทั้งหมด ที่จะถูกนำไปใช้สำหรับสร้างตารางความสัมพันธ์ต่อไป

3.1.1.2 ตารางความสัมพันธ์เวอร์ชันต้นฉบับ (Original Sequence Dependency Table: OSDT)

ตารางความสัมพันธ์เวอร์ชันต้นฉบับมีหน้าที่จัดเก็บข้อมูลความสัมพันธ์ของแผนภาพลำดับของเวอร์ชันต้นฉบับเพื่อจะนำมาใช้เป็นข้อมูลเริ่มต้นเพื่อใช้ในการเปรียบเทียบและการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลงกับแผนภาพลำดับเวอร์ชันใหม่ที่มีการแก้ไข โดยข้อมูลในตารางนี้จะถูกสร้างจากการอ่านข้อมูลจากไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับและถูกจัดเก็บอยู่ในฐานข้อมูลความสัมพันธ์ของอ็อบเจกต์

รูปที่ 3.4 แสดงภาพตัวอย่างแผนภาพลำดับเวอร์ชันต้นฉบับของการตรวจสอบรหัสบัตรของระบบเอทีเอ็ม (ATM) และรูปที่ 3.5 แสดงตัวอย่างข้อมูลของตารางลำดับความสัมพันธ์เวอร์ชันต้นฉบับที่สร้างจากแผนภาพลำดับของการตรวจสอบรหัสบัตรของระบบเอทีเอ็ม



รูปที่ 3.4 ตัวอย่างแผนภาพลำดับของระบบเอทีเอ็มในส่วนของการตรวจสอบรหัสบัตร [14]

Original Sequence Dependency Table

	GID1	MessageName	ObjectFrom	ObjectTo	Guard	Loop
▶	1	cardInfo()	cardReader	sessionMgr		<input type="checkbox"/>
	2	eject()	sessionMgr	cardReader	Invalid ATM Card	<input type="checkbox"/>
	3	checkCard()	sessionMgr	aBank		<input type="checkbox"/>
	4	retain()	sessionMgr	cardReader	status.isStolen	<input type="checkbox"/>
	5	eject()	sessionMgr	cardReader	status.CloseAccount	<input type="checkbox"/>
	6	requestPIN(int, string)	sessionMgr	displayMgr	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
	7	readPIN()	sessionMgr	keyReader	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
	8	verifyPIN()	sessionMgr	aBank	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
	9	eject()	sessionMgr	cardReader	Invalid PIN	<input type="checkbox"/>
	10	displayHello()	sessionMgr	displayMgr		<input type="checkbox"/>

รูปที่ 3.5 ตัวอย่างตารางความสัมพันธ์เวอร์ชันต้นฉบับจากแผนภาพลำดับของระบบเอทีเอ็ม

ตารางความสัมพันธ์เวอร์ชันต้นฉบับของแผนภาพลำดับจะเก็บข้อมูลดังต่อไปนี้

1) รหัสลำดับการทดสอบ (GID1)

รหัสลำดับการทดสอบเป็นรหัสที่ใช้เพื่ออ้างอิงและยืนยันถึงลำดับของเมสเสจ ข้อมูลอ็อบเจกต์ต้นทาง อ็อบเจกต์ปลายทาง การ์ด การวนซ้ำ ที่อยู่ในแผนภาพลำดับซึ่งรหัสลำดับการทดสอบจะถูกกำหนดได้ จากลำดับของตำแหน่งของเมสเสจที่พบ โดยรหัสลำดับการทดสอบของแผนภาพลำดับเวอร์ชันต้นฉบับจะเริ่มต้นจาก รหัสเบอร์ 1 และเพิ่มขึ้นทีละ 1 สำหรับตำแหน่งต่อไป โดยข้อมูลของรหัสลำดับการทดสอบจะไม่สามารถซ้ำกันได้

2) ชื่อเมสเสจ (Message Name)

```
<?xml version="1.0" encoding="UTF-8"?>
<Project Author="warint" CommentTableSortAscending="false" CommentTableSortColumn="Date Time" Description="<html><br></html>"
type="text/css" <!--<br>
<!--<br>
</head><br>
</body><br>
</html>" DocumentationType="html" ExportedFromDiffE
ExporterVersion="8.2.0" Name="Regression" TextualAnalysisHighlightOptionCaseSensitive="false" UmlVersion="2.x" Xml_structure="simple">
  <ProjectInfo>
  <Models>
    <Frame BaseY="40" Documentation_plain="" Id="gKLasKD.AAAAQkk" Name="Original" PmAuthor="BiG" PmCreateDateTime="2013-12-30T07:04:53.377" PmLast
"2014-03-26T19:17:19.169" QualityScore="-1" Type="sd" UserIDLastNumericValue="0">
      <ModelRelationshipContainer Documentation_plain="" Id="FoFHAsKD.AAAAQlf" Name="relationships" PmAuthor="BiG" PmCreateDateTime="2013-12-30T07:23
"2014-06-24T23:24:54.447" QualityScore="-1" UserIDLastNumericValue="0">
        <ModelChildren>
          <ModelRelationshipContainer Documentation_plain="" Id="pCFHAsKD.AAAAQlg" Name="Message" PmAuthor="BiG" PmCreateDateTime="2013-12-30T07:
"2014-06-28T22:38:26.201" QualityScore="-1" UserIDLastNumericValue="0">
            <ModelChildren>
              <Message Asynchronous="false" Documentation_plain="" DurationHeight="30" EndRelationshipFromMetaModelElement=".CfbAsKD.AAAAQk0"
EndRelationshipToMetaModelElement="vwnHAsKD.AAAAQlq" FromActivation="QRXHAsKD.AAAAQl0" Id="HgXHAsKD.AAAAQl1" Name="cardInfo()" />
            </ModelChildren>
          </ModelRelationshipContainer>
        </ModelChildren>
      </ModelRelationshipContainer>
    </Frame>
  </Models>
</Project>
```

รูปที่ 3.6 ตัวอย่างข้อมูลของชื่อเมสเสจจากแผนภาพลำดับ

รูปที่ 3.6 แสดงถึงที่อยู่ของข้อมูลของชื่อเมสเสจที่ใช้ระบุถึงชื่อของเมสเสจหรือเมที่อดซิกเนเจอร์ ซึ่งจะสามารถหาได้จากข้อมูลของลักษณะประจำชื่อ Name

3) อ็อบเจกต์ต้นทาง (From Object)

```
<?xml version="1.0" encoding="UTF-8"?>
<Project Author="warint" CommentTableSortAscending="false" CommentTableSortColumn="Date Time" Description="<html><br></html>"
type="text/css" <!--<br>
<!--<br>
</head><br>
</body><br>
</html>" DocumentationType="html" ExportedFromDiffE
ExporterVersion="8.2.0" Name="Regression" TextualAnalysisHighlightOptionCaseSensitive="false" UmlVersion="2.x" Xml_structure="simple">
  <ProjectInfo>
  <Models>
    <Frame BaseY="40" Documentation_plain="" Id="gKLasKD.AAAAQkk" Name="Original" PmAuthor="BiG" PmCreateDateTime="2013-12-30T07:04:53.377" PmLast
"2014-03-26T19:17:19.169" QualityScore="-1" Type="sd" UserIDLastNumericValue="0">
      <ModelRelationshipContainer Documentation_plain="" Id="FoFHAsKD.AAAAQlf" Name="relationships" PmAuthor="BiG" PmCreateDateTime="2013-12-30T07:23
"2014-06-24T23:24:54.447" QualityScore="-1" UserIDLastNumericValue="0">
        <ModelChildren>
          <ModelRelationshipContainer Documentation_plain="" Id="pCFHAsKD.AAAAQlg" Name="Message" PmAuthor="BiG" PmCreateDateTime="2013-12-30T07:
"2014-06-28T22:38:26.201" QualityScore="-1" UserIDLastNumericValue="0">
            <ModelChildren>
              <InteractionLifeLine Active="true"
"Unspecified" Name="cardReader" Pm
            </ModelChildren>
          </ModelRelationshipContainer>
        </ModelChildren>
      </ModelRelationshipContainer>
    </Frame>
  </Models>
</Project>
```

รูปที่ 3.7 ตัวอย่างข้อมูลของอ็อบเจกต์ต้นทางจากแผนภาพลำดับ

รูปที่ 3.7 แสดงถึงที่อยู่ของข้อมูลของอ็อบเจกต์ต้นทางใช้ระบุถึงอ็อบเจกต์ต้นทางที่เป็นจุดเริ่มต้นของการใช้งานซึ่งสามารถหาได้จากข้อมูลของลักษณะประจำชื่อ Name

4) อ็อบเจกต์ปลายทาง (To Object)

```
<?xml version="1.0" encoding="UTF-8"?>
<Project Author="warint" CommentTableSortAscending=
type="text/css" style="font-size: 11px; color: #000000; font-family: C
"2014-03-26T19:17:19.169" QualityScore="-1
<ProjectInfo>
<Models>
<Frame BaseY="40" Documentation_plain="" I
"2014-03-26T19:17:19.169" QualityScore="-1
<ModelChildren>
<InteractionLifeLine Active="true"
"Unspecified" Name="cardReader" Pm
UserIDLastNumericValue="0">
<InteractionLifeLine Active="true"
"Unspecified" Name="sessionMgr" Pm
```

รูปที่ 3.8 ตัวอย่างข้อมูลของอ็อบเจกต์ปลายทางจากแผนภาพลำดับ

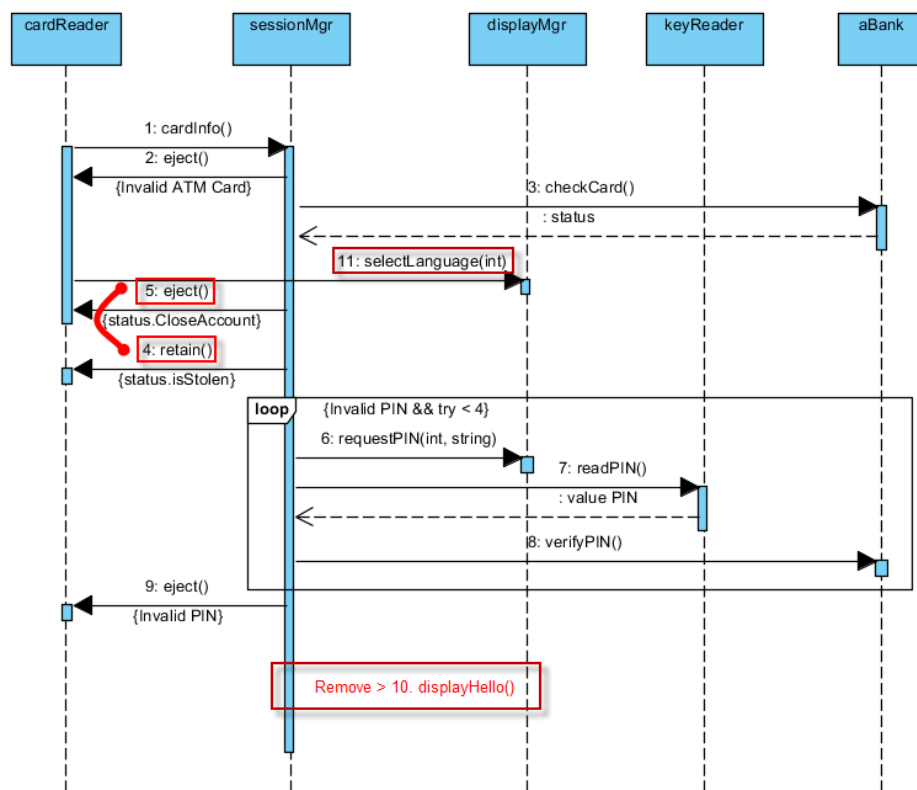
รูปที่ 3.8 แสดงถึงข้อมูลที่อยู่ของอ็อบเจกต์ปลายทางที่ใช้ระบุถึงอ็อบเจกต์ปลายทางที่เป็นปลายทางของการใช้งานซึ่งสามารถหาได้จากข้อมูลของลักษณะประจำชื่อ Name

5) การ์ด (Guard)

```
<?xml version="1.0" encoding="UTF-8"?>
<Project Author="warint" CommentTableSortAscending="false" CommentTableSortColumn="Date Time" Desc
type="text/css" style="font-size: 11px; color: #000000; font-family: C
"2014-03-26T19:17:19.169" QualityScore="-1" Type="sd" UserIDLastNumericValue="0">
<ModelChildren>
<InteractionLifeLine Active="true" Class="false" ClassLevelStereotype="false" Docu
"Unspecified" Name="cardReader" PmAuthor="BiG" PmCreateDateTime="2013-12-30T07:16:
UserIDLastNumericValue="0">
<InteractionLifeLine Active="true" Class="false" ClassLevelStereotype="false" Docu
"Unspecified" Name="sessionMgr" PmAuthor="BiG" PmCreateDateTime="2013-12-30T07:24:
UserIDLastNumericValue="0">
<InteractionLifeLine Active="true" Class="false" ClassLevelStereotype="false" Docu
"Unspecified" Name="aBank" PmAuthor="BiG" PmCreateDateTime="2013-12-30T07:41:02.2:
UserIDLastNumericValue="0">
<NOTE Documentation="
type="text/css" style="font-size: 11px; color: #000000; font-family: C
"2014-03-26T19:17:19.169" QualityScore="-1" Type="sd" UserIDLastNumericValue="0">
```

รูปที่ 3.9 ตัวอย่างข้อมูลของการ์ดจากแผนภาพลำดับ

ถ้าตรวจสอบแล้วมีข้อมูลตรงกันทั้ง 5 คอลัมน์ก็จะนำข้อมูลที่อยู่ในคอลัมน์ GID1 จากตารางความสัมพันธ์เวอร์ชันต้นฉบับมาเป็นข้อมูลในคอลัมน์ GID2 แต่ถ้าตรวจสอบแล้วมีข้อมูลไม่ตรงกัน เครื่องมือทดสอบจะทำการสร้างรหัสใหม่ให้และนำไปใช้ในคอลัมน์ GID2 ดังตัวอย่างในรูปที่ 3.12 ที่มีการสร้างรหัสใหม่ คือเบอร์ 11 ซึ่งเกิดจากมีการเพิ่มข้อมูลเข้ามาและทำการตรวจสอบกับตารางความสัมพันธ์เวอร์ชันต้นฉบับในรูปที่ 3.5 แล้วไม่พบข้อมูลที่ตรงกันจึงทำการสร้างรหัสขึ้นมาใหม่ให้อย่างอัตโนมัติ



รูปที่ 3.11 ตัวอย่างแผนภาพลำดับที่มีการเพิ่มเมสเสจใหม่ การลบออกและมีการสลับตำแหน่ง

New Sequence Dependency Table

	GID2	MessageName	ObjectFrom	ObjectTo	Guard	Loop
▶	1	cardInfo()	cardReader	sessionMgr		<input type="checkbox"/>
	2	eject()	sessionMgr	cardReader	Invalid ATM Card	<input type="checkbox"/>
	3	checkCard()	sessionMgr	aBank		<input type="checkbox"/>
	11	selectLanguage(int)	cardReader	displayMgr		<input type="checkbox"/>
	5	eject()	sessionMgr	cardReader	status.CloseAccount	<input type="checkbox"/>
	4	retain()	sessionMgr	cardReader	status.isStolen	<input type="checkbox"/>
	6	requestPIN(int, string)	sessionMgr	displayMgr	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
	7	readPIN()	sessionMgr	keyReader	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
	8	verifyPIN()	sessionMgr	aBank	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
	9	eject()	sessionMgr	cardReader	Invalid PIN	<input type="checkbox"/>

รูปที่ 3.12 ตัวอย่างตารางความสัมพันธ์เวอร์ชันใหม่

ข้อมูลในตารางความสัมพันธ์เวอร์ชันใหม่จะถูกสร้างขึ้นใหม่จากการอ่านข้อมูลจากแผนภาพลำดับเวอร์ชันใหม่ที่ถูกแก้ไขและนำเข้ามา จากนั้นจะถูกนำไปจัดเก็บเข้าฐานข้อมูลลำดับความสัมพันธ์ของอ็อบเจกต์หลังจากผ่านขั้นตอนการเปรียบเทียบแล้ว เพื่อให้สามารถนำข้อมูลกลับมาใช้ในอนาคตกับแผนภาพลำดับเวอร์ชันใหม่กว่า หลังจากที่ข้อมูลของตารางความสัมพันธ์เวอร์ชันใหม่ถูกจัดเก็บลงไปในฐานข้อมูลลำดับความสัมพันธ์ของอ็อบเจกต์แล้วก็จะถือได้ว่าข้อมูลที่อยู่ในตารางความสัมพันธ์เวอร์ชันใหม่จะกลายเป็นเวอร์ชันต้นฉบับทันที

3.1.2 รายการลำดับการแก้ไข (List of Modification)

รายการลำดับการแก้ไขจะถูกสร้างมาจากการนำข้อมูลที่อยู่ในคอลัมน์ GID1 จากตารางความสัมพันธ์เวอร์ชันต้นฉบับและคอลัมน์ GID2 จากตารางความสัมพันธ์เวอร์ชันใหม่ เพื่อที่จะนำไปใช้สำหรับขั้นตอนการเปรียบเทียบและวิเคราะห์หาผลกระทบการเปลี่ยนแปลงที่เกิดจากแผนภาพลำดับเวอร์ชันใหม่ต่อไป โดยภาพตัวอย่างของรายการลำดับการแก้ไขสามารถแสดงได้ดังรูปที่ 3.13

Compared Result List			
	GID1	GID2	Result
▶	1	1	Match
	2	2	Match
	3	3	Match
	0	11	Insertion
	4	5	Substitution
	5	4	Substitution
	6	6	Match
	7	7	Match
	8	8	Match
	9	9	Match
	10	0	Deletion

รูปที่ 3.13 ตัวอย่างรายการลำดับการแก้ไขของแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่

3.1.3 ส่วนการเปรียบเทียบและหาผลกระทบการเปลี่ยนแปลง (Comparator)

เครื่องมือทดสอบจะนำข้อมูลจากรายการลำดับการแก้ไขมาใช้เป็นข้อมูลนำเข้าในการเปรียบเทียบการเปลี่ยนแปลงระหว่างแผนภาพลำดับทั้งสองเวอร์ชัน โดยจะประยุกต์ใช้อัลกอริทึม Levenshtein Edit Distance [18] ซึ่งเป็นวิธีการใช้สำหรับการหาความแตกต่างระหว่างข้อมูล 2 ชุด โดยประเภทของการเปรียบเทียบสามารถแบ่งได้เป็น 4 ประเภทดังตารางที่ 3.1

ตารางที่ 3.1 ตารางประเภทของการเปรียบเทียบ

ประเภท	รายละเอียด
1	การลบออก (Deletion)
2	การแทรก (Insertion)
3	การแทนที่หรือสลับตำแหน่ง (Substitution)
4	ตรงกัน (Match)

โดยประเภทของการเปรียบเทียบสามารถอธิบายได้ดังต่อไปนี้

1) การลบออก (Deletion)

การที่ข้อมูลที่ถูกเปรียบเทียบของแผนภาพลำดับเวอร์ชันใหม่นั้นมีข้อมูลถูกลบออกไป โดยการเปรียบเทียบกับข้อมูลนำเข้าของความสัมพันธ์เวอร์ชันต้นฉบับและทำให้เกิดผลกระทบของการลบออกคือข้อมูลที่อยู่ติดกันทั้งบนและด้านล่าง

2) การแทรก (Insertion)

การที่ข้อมูลที่ถูกเปรียบเทียบมีการแทรกข้อมูลใหม่เข้ามาในแผนภาพลำดับเวอร์ชันใหม่และทำให้เกิดผลกระทบของการแทรกคือข้อมูลที่อยู่ติดกันทั้งด้านบนและด้านล่าง

3) การแทนที่หรือสลับตำแหน่ง (Substitution)

การที่ข้อมูลที่ถูกเปรียบเทียบถูกสลับตำแหน่งกันผลลัพธ์ของการเปรียบเทียบซึ่งจะได้มาถึงผลกระทบของการเปลี่ยนแปลงจากการแก้ไขแผนภาพลำดับเวอร์ชันใหม่นั้นสามารถนำไปใช้สำหรับการเลือกกรณีทดสอบที่อยู่ในฐานข้อมูลกรณีทดสอบ สำหรับการทดสอบเชิงถดถอยต่อไป

4) ตรงกัน (Match)

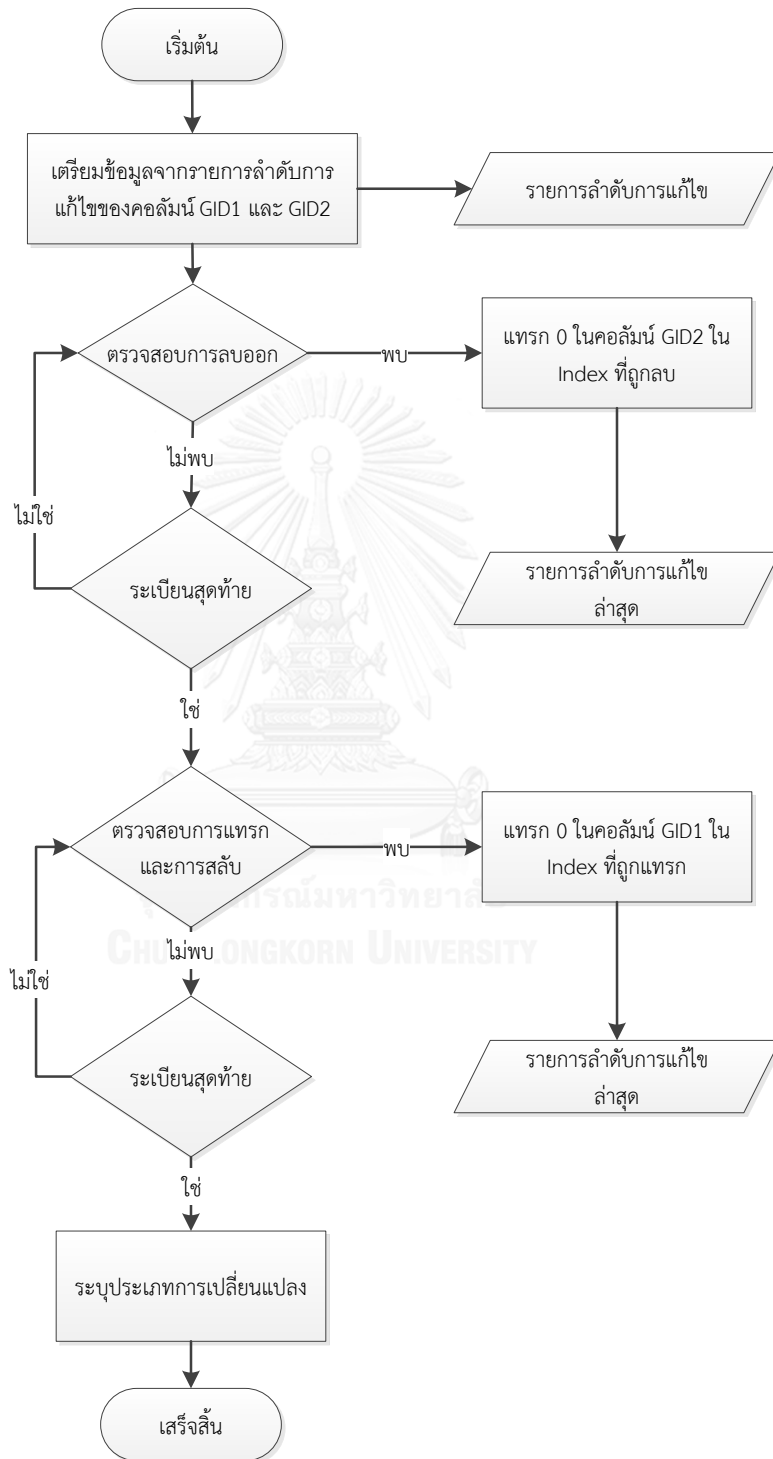
การที่ข้อมูลที่ถูกเปรียบเทียบระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับกับเวอร์ชันใหม่ไม่มีการเปลี่ยนแปลง

ประเภทของการเปลี่ยนแปลงจากการเปรียบเทียบ แบ่งได้ 3 ประเภท [4] ดังต่อไปนี้

- 1) การเพิ่มหรือลบเมสเสจ
- 2) การเพิ่มหรือลบอีอบเจกต์
- 3) การเปลี่ยนแปลงรายละเอียดการทำงานของแผนภาพลำดับ
 - 3.1) การเปลี่ยนลำดับของเมสเสจ
 - 3.2) การเปลี่ยนเมทอดซิกเนเจอร์
 - 3.2.1) การเปลี่ยนชื่อเมทอด
 - 3.2.2) การเปลี่ยนจำนวนพารามิเตอร์
 - 3.2.3) การเปลี่ยนประเภทของพารามิเตอร์



กระบวนการค้นหาผลกระทบการเปลี่ยนแปลงของงานวิจัยนี้จะประยุกต์ใช้อัลกอริทึมดังกล่าวโดยมีขั้นตอนดังรูปที่ 3.14



รูปที่ 3.14 ขั้นตอนการเปรียบเทียบจากรายการลำดับการแก้ไข

จากรูปที่ 3.14 สามารถอธิบายถึงขั้นตอนการเปรียบเทียบได้เป็นลำดับขั้นตอนดังต่อไปนี้

1. เตรียมข้อมูลจากรายการลำดับการแก้ไขของคอลัมน์ GID1 และ GID2
2. ตรวจสอบการลบบอก โดยการนำข้อมูลจากคอลัมน์ GID1 ที่ละระเบียบเข้าไปค้นหาในคอลัมน์ GID2 ถ้าพบให้ทำการค้นหาในระเบียบต่อไปถ้าไม่พบให้ทำการแทรกเลขศูนย์เข้าไปที่ตำแหน่งนั้นในคอลัมน์ GID2
3. ตรวจสอบการแทรกและการแทนที่ โดยการนำข้อมูลจากคอลัมน์ GID2 ที่ละระเบียบเข้าไปค้นหาในคอลัมน์ GID1 ถ้าพบให้ทำการค้นหาในระเบียบต่อไปถ้าไม่พบให้ทำการแทรกเลขศูนย์เข้าไปที่ตำแหน่งนั้นในคอลัมน์ GID1
4. ทำการเปรียบเทียบและระบุประเภทของการเปลี่ยนแปลงโดยนำข้อมูลของคอลัมน์ GID1 เข้าไปทำการค้นหาข้อมูลจากคอลัมน์ GID2 และทำการระบุถึงประเภทของการเปลี่ยนแปลง โดยการระบุการเปลี่ยนแปลงจะใช้ข้อมูลในคอลัมน์ GID1 และ GID2 โดยมีรายละเอียดดังต่อไปนี้
 - ข้อมูล GID1 และ GID2 ตรงกันจะระบุประเภทของการเปลี่ยนแปลงเป็นตรงกัน
 - ข้อมูล GID2 เป็นศูนย์จะระบุประเภทของการเปลี่ยนแปลงเป็นการลบบอก
 - ข้อมูล GID1 เป็นศูนย์จะระบุประเภทของการเปลี่ยนแปลงเป็นการแทรก
 - ข้อมูล GID1 และ GID2 ไม่เหมือนกันและไม่ใช่ศูนย์จะระบุประเภทของการเปลี่ยนแปลงเป็นการสลับ

ผลกระทบที่ได้จากการเปรียบเทียบจะถูกนำมาใช้ค้นหาผลกระทบจากการเปลี่ยนแปลงจากรายการความสัมพันธ์เวอร์ชันต้นฉบับสำหรับการลบบอกและเวอร์ชันใหม่สำหรับการแทรกและการแทนที่ โดยจะพิจารณาเฉพาะผลกระทบทางตรงเท่านั้น (Direct impact) ซึ่งเป็นผลกระทบที่เกิดขึ้นโดยตรงกับส่วนที่มีการแก้ไข โดยจะพิจารณาตามประเภทของการเปรียบเทียบ

New Sequence Dependency Table						
	GID2	MessageName	ObjectFrom	ObjectTo	Guard	Loop
▶	1	cardInfo()	cardReader	sessionMgr		<input type="checkbox"/>
	2	eject()	sessionMgr	cardReader	Invalid ATM Card	<input type="checkbox"/>
	3	checkCard()	sessionMgr	aBank		<input type="checkbox"/>
	11	selectLanguage(int)	cardReader	displayMgr		<input type="checkbox"/>
	5	eject()	sessionMgr	cardReader	status.CloseAccount	<input type="checkbox"/>
	4	retain()	sessionMgr	cardReader	status.isStolen	<input type="checkbox"/>
	6	requestPIN(int, string)	sessionMgr	displayMgr	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
	7	readPIN()	sessionMgr	keyReader	Invalid PIN && try < 4	<input type="checkbox"/>
	8	verifyPIN()	sessionMgr	aBank	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
	9	eject()	sessionMgr	cardReader	Invalid PIN	<input type="checkbox"/>

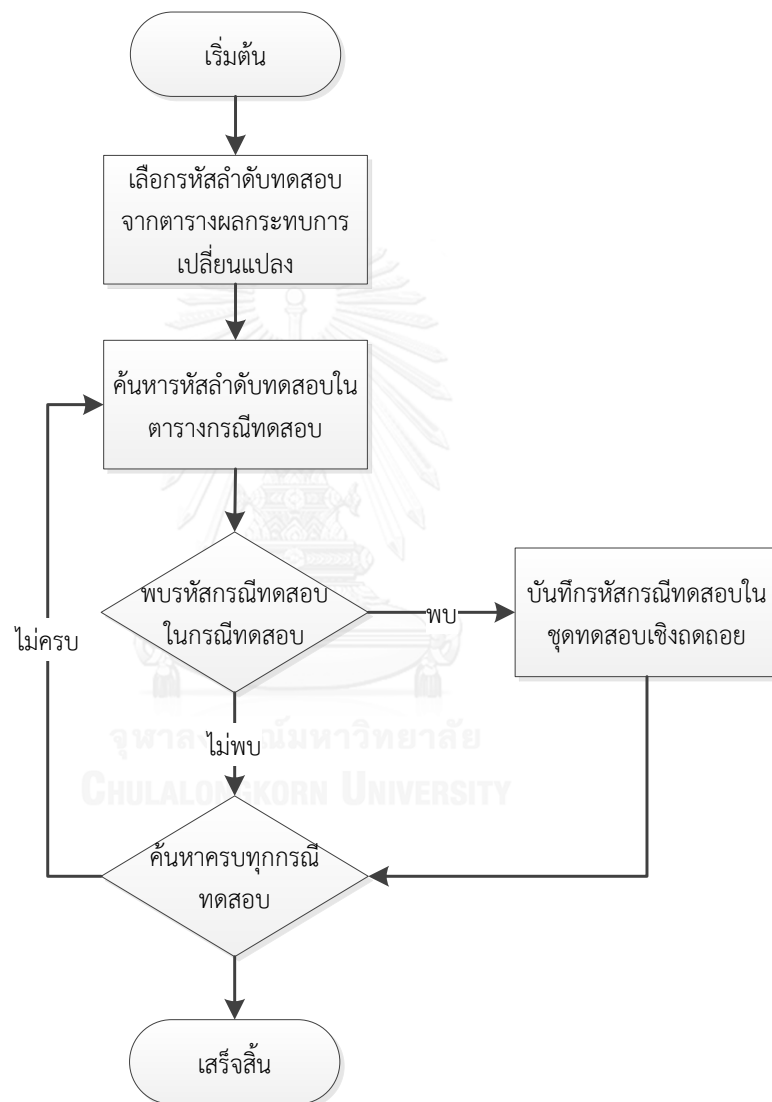
รูปที่ 3.15 ตัวอย่างผลกระทบทางตรงจากการเปลี่ยนแปลงจากการแทรกและการแทนที่

จากรูปที่ 3.15 จะแสดงให้เห็นถึงผลกระทบทางตรงของการเปลี่ยนแปลงจากการแทรกเมสเสจชื่อ selectLanguage() เข้ามาทำให้เกิดผลกระทบของการแทรกคือเมสเสจชื่อ checkCard() กับ eject() แต่เนื่องจากเมสเสจ eject() เป็นเมสเสจที่มาจากการสลับตำแหน่งกับเมสเสจ retain() ซึ่งจะได้ว่าเมสเสจ eject() กับ retain() มีการแก้ไขถูกนำไปใช้เป็นผลกระทบอยู่แล้ว

ดังนั้นจึงทำการตรวจสอบผลกระทบจากการสลับตำแหน่งของเมสเสจ eject() กับ retain() ซึ่งจะได้ผลกระทบทางตรงเพิ่มขึ้นจากการแทนที่ซึ่งก็คือเมสเสจ selectLanguage() และ requestPIN(int,string) แต่เมสเสจ selectLanguage() นั้นมาจากการแทรกอยู่แล้วดังนั้นจึงไปใช้ผลลัพธ์ของผลกระทบของเมสเสจ selectLanguage() แทน โดยผลกระทบทั้งหมดที่หาได้นั้นจะถูกนำไปใช้ในการจัดการกรณีทดสอบต่อไป

3.1.4 ส่วนการเลือกกรณีทดสอบ (Test Case Selector)

งานวิจัยนี้ได้นำวิธีการเลือกกรณีทดสอบแบบปลอดภัย [16] เข้ามาประยุกต์ใช้สำหรับการเลือกกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอย โดยจะทำการเลือกกรณีทดสอบจากทุกกรณีทดสอบอย่างน้อย 1 กรณีทดสอบที่ได้รับผลกระทบจากการเปลี่ยนแปลงทางตรง โดยการเลือกกรณีทดสอบแบบปลอดภัยนั้นสามารถอธิบายได้ดังขั้นตอนดังรูปที่ 3.16



รูปที่ 3.16 ขั้นตอนการเลือกกรณีทดสอบที่ได้รับผลกระทบต่อการเปลี่ยนแปลงแผนภาพลำดับ

3.1.5 ชุดกรณีทดสอบเชิงถดถอย (Regression Test Set)

ชุดทดสอบเชิงถดถอยจะเป็นกรณีทดสอบที่ได้มาจากผลกระทบของการเปลี่ยนแปลงจากแผนภาพลำดับเวอร์ชันใหม่และขั้นตอนการเลือกกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอยนั้นจะทำการเลือกกรณีทดสอบโดยเอารหัสลำดับของการทดสอบที่อยู่ในตารางผลกระทบการเปลี่ยนแปลงเข้าไปค้นหาในตารางกรณีทดสอบที่ละกรณีทดสอบถ้าพบรหัสลำดับของการทดสอบที่กรณีทดสอบใดก็จะนำข้อมูลของกรณีทดสอบนั้นคัดลอกมาใส่ในตารางกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอย

กรณีทดสอบที่จะใช้ในการแสดงรายงานหลังจากผ่านขั้นตอนการระบุผลกระทบการเปลี่ยนแปลงได้เป็นทั้งหมด 4 ประเภท [4] ดังนี้

- 1) กรณีทดสอบที่ถูกสร้างใหม่ (New Test Cases)
เป็นกรณีทดสอบที่ถูกสร้างใหม่หลังจากมีการเพิ่มหรือแก้ไขแผนภาพลำดับ
- 2) กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว (Obsolete Test Cases)
เป็นกรณีทดสอบที่ไม่สามารถใช้งานได้กับแผนภาพลำดับเวอร์ชันใหม่
- 3) กรณีทดสอบที่นำกลับมาทดสอบใหม่ (Retestable Test Cases)
เป็นกรณีทดสอบที่ยังสามารถใช้งานได้และจำเป็นต้องนำกลับมาใช้ในการทดสอบ
- 4) กรณีทดสอบที่ยังใช้ได้ทั้งหมด (Reusable Test Cases)
เป็นกรณีทดสอบที่ยังสามารถใช้งานได้แต่ไม่จำเป็นต้องนำกลับมาทดสอบใหม่

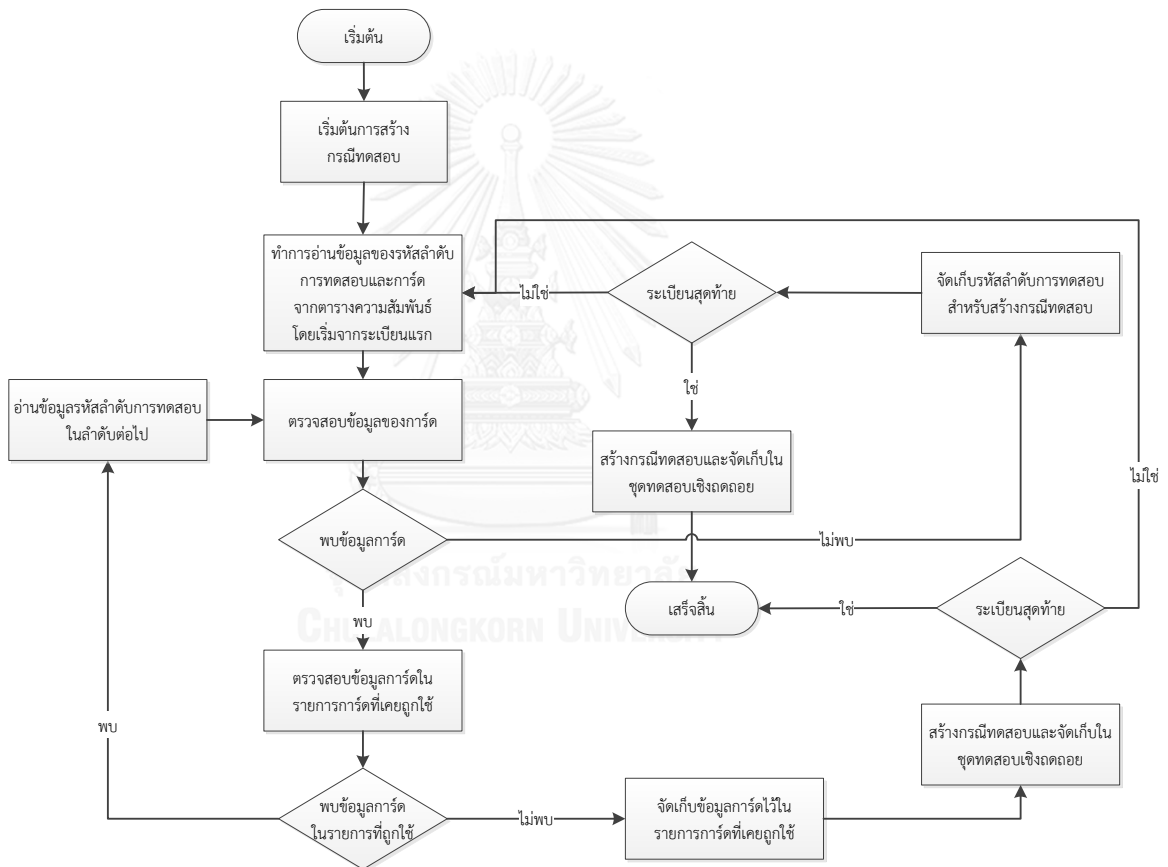
3.1.6 ฐานข้อมูลกรณีทดสอบ (Test Case Database)

ใช้สำหรับเก็บข้อมูลกรณีทดสอบทั้งหมดที่สามารถเกิดขึ้นได้ โดยข้อมูลที่อยู่ในฐานข้อมูลกรณีทดสอบจะมาจากแผนภาพลำดับซึ่งแผนภาพลำดับที่ถูกนำไปสร้างกรณีทดสอบนั้นจะถูกสร้างได้จากสองทางด้วยกันซึ่งก็คือ แผนภาพลำดับเวอร์ชันต้นฉบับหรือเวอร์ชันแรกเริ่มที่เป็นการนำแผนภาพลำดับนั้นเข้าสู่เครื่องมือทดสอบครั้งแรกและแผนภาพลำดับเวอร์ชันใหม่ที่นำมาใช้ในการเปรียบเทียบ

ซึ่งหลังจากการนำเข้าแผนภาพลำดับในครั้งแรกหรือทำการวิเคราะห์ผลกระทบการเปลี่ยนแปลงแล้วจะทำการสร้างลำดับทดสอบที่เป็นไปได้ทั้งหมดและจัดเก็บไว้ในฐานข้อมูลสำหรับใช้เป็นชุดทดสอบครั้งต่อไป โดยการแปลงแผนภาพลำดับเวอร์ชันใหม่ด้วยอัลกอริทึม Traversal จากงานวิจัยที่ [7]

กระบวนการสร้างกรณีทดสอบของงานวิจัยนี้สามารถอธิบายได้ดังต่อไปนี้

- 1) เตรียมข้อมูลที่จะใช้สำหรับสร้างกรณีทดสอบให้อยู่ในรูปแบบตารางความสัมพันธ์
- 2) อ่านข้อมูลจากตารางความสัมพันธ์ที่ละเอียดโดยใช้ข้อมูลจากคอลัมน์รหัสลำดับการทดสอบ (GID) และการ์ด (Guard) ในการสร้างกรณีทดสอบ
- 3) ในการสร้างกรณีทดสอบนั้นจะนำรหัสลำดับการทดสอบที่ถูกระบุไว้ตารางความสัมพันธ์ติดกันมาเรียงต่อกันจากบนลงล่างโดยกรณีทดสอบที่ถูกสร้างขึ้นมาแต่ละกรณีนั้นจะถูกแยกด้วยข้อมูลของการ์ด ซึ่งโดยขั้นตอนการสร้างและจัดเก็บกรณีทดสอบเข้าสู่ฐานข้อมูลกรณีทดสอบโดยละเอียดนั้น สามารถแสดงได้ดังรูปที่ 3.17



รูปที่ 3.17 ขั้นตอนการสร้างและจัดเก็บกรณีทดสอบเข้าสู่ฐานข้อมูลกรณีทดสอบ

จากรูปที่ 3.17 แสดงถึงการที่กรณีทดสอบที่ถูกสร้างขึ้นจากลำดับรหัสทดสอบที่มีความสัมพันธ์กัน โดยอ้างอิงจากรางความสัมพันธ์ โดยจะเป็นกรณีทดสอบที่ถูกสร้างขึ้นมานี้ จะใช้สำหรับสนับสนุนการทดสอบแบบบูรณาการ โดยตารางกรณีทดสอบ (Test Case Table) จะเก็บข้อมูลในฐานะข้อมูลดังต่อไปนี้

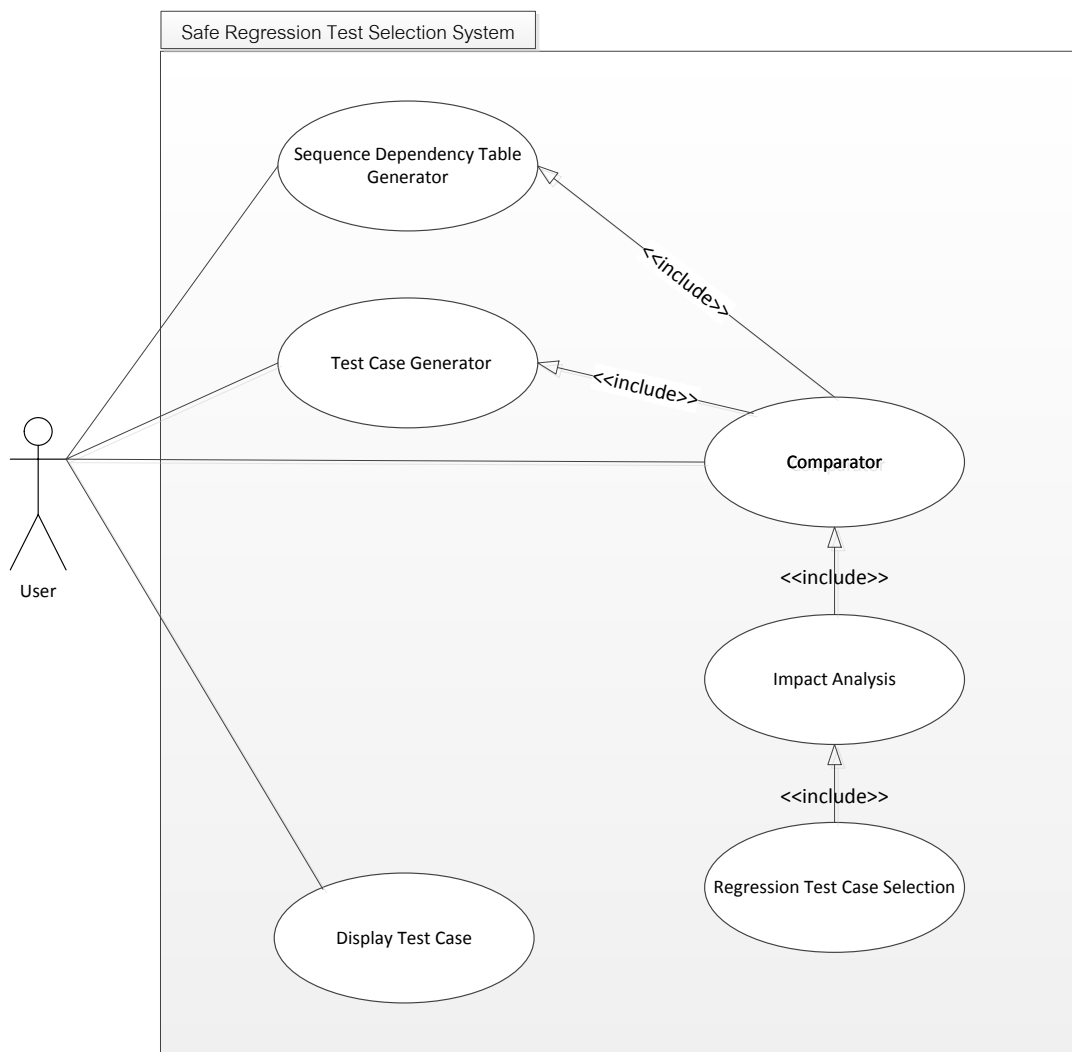
- 1) รหัสกรณีทดสอบ (Test Case No) ใช้เป็นรหัสอ้างอิงถึงกรณีทดสอบแต่ละกรณี
- 2) ลำดับการทดสอบ (Test Sequence) ใช้เก็บรหัสของความสัมพันธ์ของลำดับของการทดสอบที่มีความสัมพันธ์กันจากแผนภาพลำดับในแต่ละกรณีทดสอบที่เป็นไปได้ทั้งสำหรับใช้เป็นชุดทดสอบ

3.2 การวิเคราะห์และออกแบบเครื่องมือ

การวิเคราะห์และออกแบบเครื่องมือในงานวิจัยนี้สามารถอธิบายได้ด้วยแผนภาพยูเอ็มแอล ซึ่งเป็นแผนภาพมาตรฐานในการอธิบายและนำเสนอแนวคิดของซอฟต์แวร์ โดยจะใช้แผนภาพยูสเคส นำเสนอถึงฟังก์ชันการทำงานหลักของเครื่องมือทดสอบและปฏิสัมพันธ์ของผู้ใช้งานกับเครื่องมือทดสอบ โดยที่แผนภาพยูสเคสที่ใช้แสดงฟังก์ชันหลักของเครื่องมือทดสอบนั้นสามารถนำไปเขียนเป็นแผนภาพกิจกรรมที่เกิดขึ้นในระบบได้ต่อไป

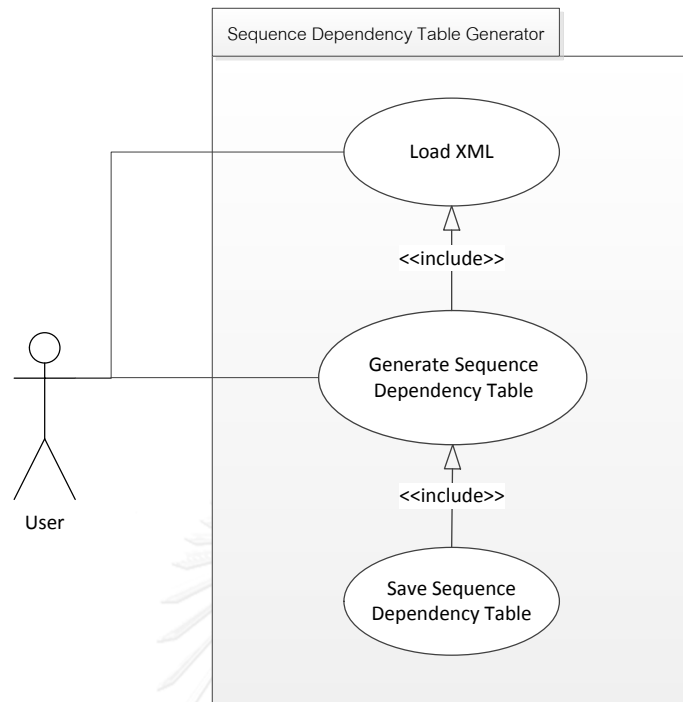
3.2.1 แผนภาพยูสเคส

แผนภาพยูสเคสใช้เพื่อแสดงฟังก์ชันการทำงานหลักทั้งหมดที่มีอยู่ในเครื่องมือทดสอบและแสดงปฏิสัมพันธ์ของผู้ใช้งานต่อฟังก์ชันงานที่มีอยู่ในเครื่องมือทดสอบ ซึ่งภาพรวมของยูสเคสสามารถแสดงรายละเอียดได้ดังรูปที่ 3.18 และรายละเอียดของแต่ละยูสเคสสามารถดูเพิ่มเติมได้จากภาคผนวก ก.



รูปที่ 3.18 แผนภาพรวมยูสเคสของเครื่องมือทดสอบ

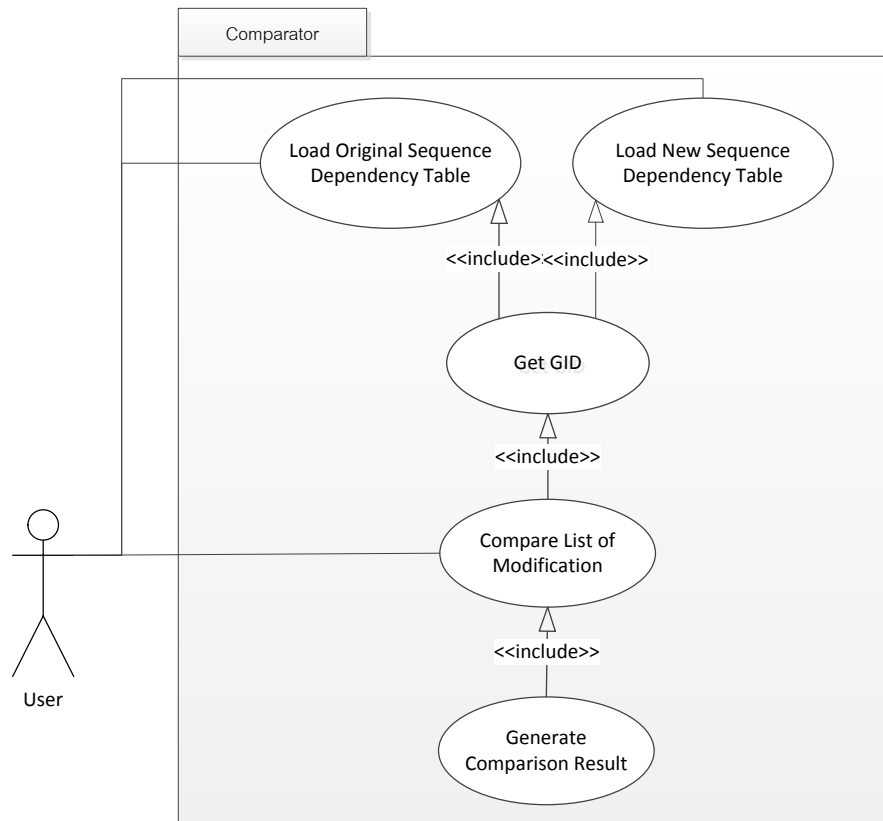
3.2.1.1 แผนภาพยูสเคสในส่วนของการสร้างตารางความสัมพันธ์



รูปที่ 3.19 แผนภาพยูสเคสการสร้างตารางความสัมพันธ์

รูปที่ 3.19 แสดงแผนภาพยูสเคสของการสร้างตารางความสัมพันธ์ ผู้ใช้งานทำการนำเข้าข้อมูลของแผนภาพลำดับที่อยู่ในรูปแบบไฟล์เอกซ์เอ็มแอล (ยูสเคส Load XML) เพื่อที่จะนำมาใช้ในการสร้างตารางความสัมพันธ์ของอีอบเจกต์ (ยูสเคส Generate Sequence Dependency Table) จากนั้นจึงทำการบันทึกข้อมูลความสัมพันธ์ (ยูสเคส Save Sequence Dependency Table)

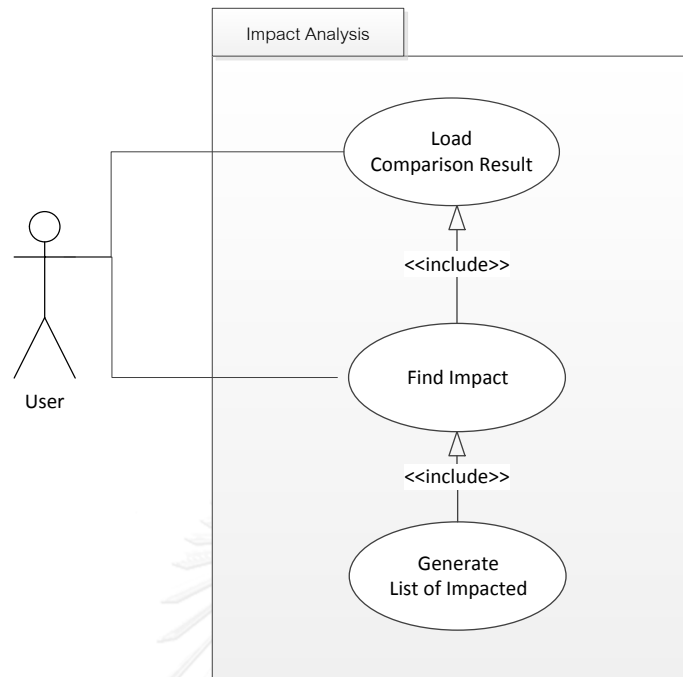
3.2.1.2 แผนภาพยูสเคสในส่วนของการเปรียบเทียบการเปลี่ยนแปลง



รูปที่ 3.20 แผนภาพยูสเคสการเปรียบเทียบการเปลี่ยนแปลง

รูปที่ 3.20 แสดงแผนภาพยูสเคสของการเปรียบเทียบของการเปลี่ยนแปลงแผนภาพลำดับของเวอร์ชันต้นฉบับกับเวอร์ชันใหม่ ผู้ใช้งานทำการนำเข้าข้อมูลตารางความสัมพันธ์ของแผนภาพลำดับเวอร์ชันต้นฉบับ (ยูสเคส Load Original Sequence Dependency Table) และเวอร์ชันใหม่ (ยูสเคส Load New Sequence Dependency Table) และทำการดึงข้อมูลของคอลัมน์ GID (ยูสเคส Get GID) โดยการเปรียบเทียบข้อมูลจากตารางความสัมพันธ์เวอร์ชันต้นฉบับและเวอร์ชันใหม่หรือสร้างข้อมูลในคอลัมน์ GID ใหม่ในกรณีที่ทำการเปรียบเทียบแล้วไม่พบข้อมูลตามเงื่อนไขที่ตรงกัน หลังจากได้ข้อมูลในคอลัมน์ GID เรียบร้อยแล้ว (ยูสเคส Compare List of Modification) ผลลัพธ์ที่ได้จะถูกเตรียมไว้สำหรับการหาผลกระทบการเปลี่ยนแปลงต่อไป (ยูสเคส Generate Comparison Result)

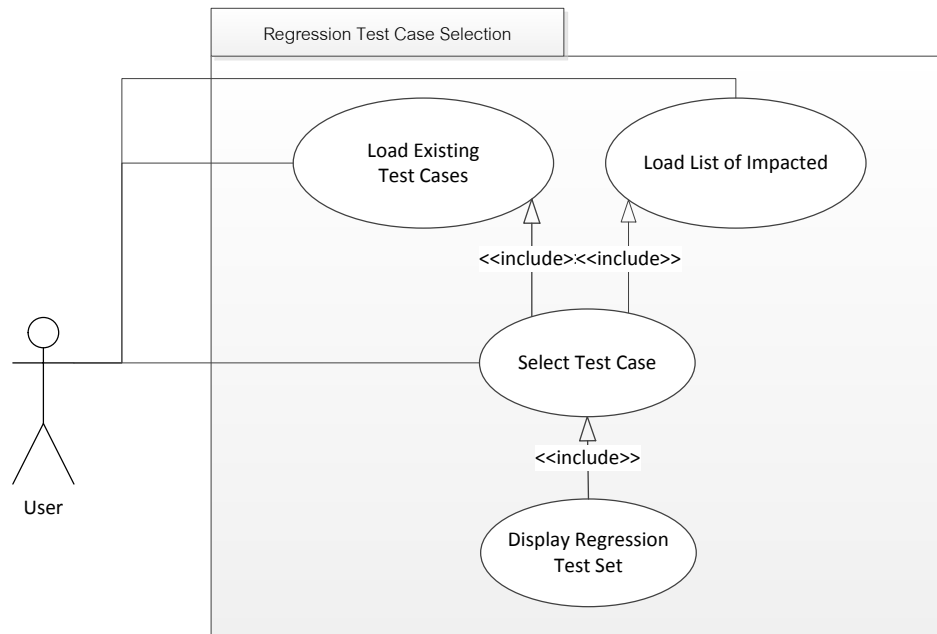
3.2.1.3 แผนภาพยูสเคสในส่วนของการหาผลกระทบการเปลี่ยนแปลง



รูปที่ 3.21 แผนภาพยูสเคสการหาผลกระทบการเปลี่ยนแปลง

รูปที่ 3.21 แสดงแผนภาพยูสเคสของการหาผลกระทบของการเปลี่ยนแปลงระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับกับเวอร์ชันใหม่ โดยการนำเข้าผลลัพธ์จากการเปรียบเทียบของผลกระทบการเปลี่ยนแปลง (ยูสเคส Load Comparison Result) ที่จะมีข้อมูลของผลลัพธ์ของการเปลี่ยนแปลง โดยจะใช้ข้อมูลของคอลัมน์ GID ของแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่มาใช้ในการหาผลกระทบ โดยการค้นหาผลกระทบการเปลี่ยนแปลงจะพิจารณาจากผลกระทบทางตรง (ยูสเคส Find Impact) เท่านั้น ซึ่งก็คือข้อมูลที่อยู่ติดกับส่วนที่มีการเปลี่ยนแปลง และผลกระทบทั้งหมดนั้นจะถูกเตรียมไว้ในรายการผลกระทบของการเปลี่ยนแปลง (ยูสเคส Generate List of Impacted)

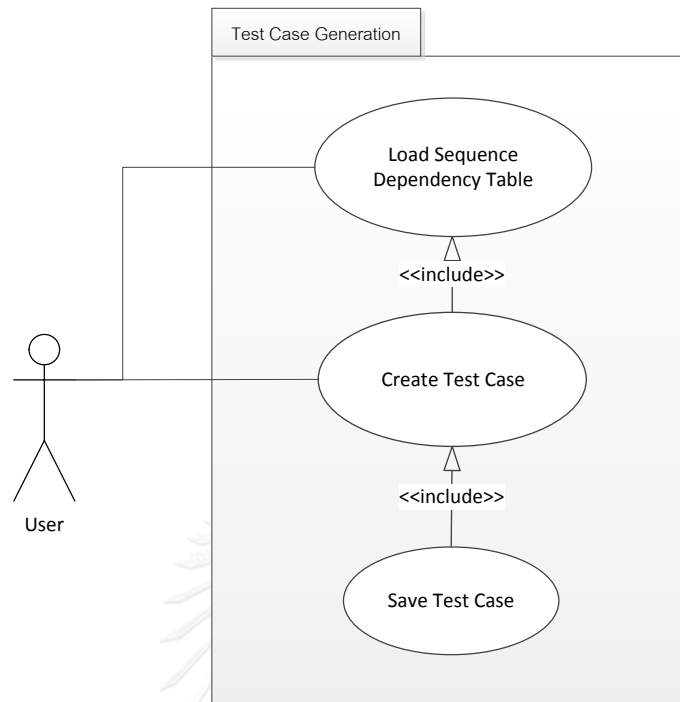
3.2.1.4 แผนภาพยูสเคสในส่วนของการเลือกกรณีทดสอบเชิงถดถอย



รูปที่ 3.22 แผนภาพยูสเคสการเลือกกรณีทดสอบเชิงถดถอย

รูปที่ 3.22 แสดงแผนภาพยูสเคสของการเลือกกรณีทดสอบเชิงถดถอยแบบปลอดภัยสำหรับการเปลี่ยนแปลงแผนภาพลำดับ ผู้ใช้งานทำการนำเข้าข้อมูลกรณีทดสอบทั้งหมดที่อยู่ในฐานข้อมูลกรณีทดสอบ (ยูสเคส Load Existing Test Cases) และนำเข้ารายการผลกระทบการเปลี่ยนแปลง (ยูสเคส Load List of Impacted) จากนั้นทำการเลือกกรณีทดสอบเชิงถดถอยจากกรณีทดสอบที่มีอยู่ทั้งหมด (ยูสเคส Select Test Case) โดยเลือกเฉพาะกรณีทดสอบที่ได้รับผลกระทบจากการเปลี่ยนแปลง จากข้อมูลที่อยู่ในรายการผลกระทบการเปลี่ยนแปลง และนำผลลัพธ์ไปจัดเก็บไว้ในชุดทดสอบเชิงถดถอยและแสดงรายการกรณีทดสอบแยกตามประเภทได้ (ยูสเคส Display Regression Test Set)

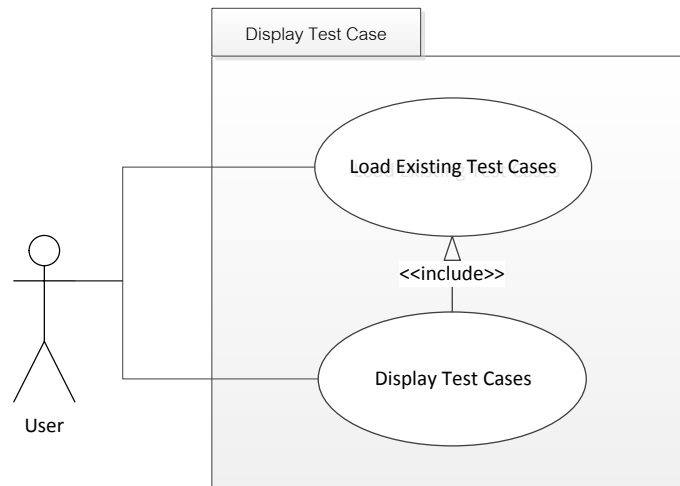
3.2.1.5 แผนภาพยูสเคสในส่วนของการสร้างกรณีทดสอบ



รูปที่ 3.23 แผนภาพยูสเคสการสร้างกรณีทดสอบ

รูปที่ 3.23 แสดงแผนภาพยูสเคสของการสร้างกรณีทดสอบ ผู้ใช้งานทำการนำเข้าข้อมูลจากตารางความสัมพันธ์ (ยูสเคส Load Sequence Dependency Table) และทำการสร้างกรณีทดสอบ (ยูสเคส Create Test Case) จากนั้นทำการบันทึกกรณีทดสอบ (ยูสเคส Save Test Case)

3.2.1.6 แผนภาพยูสเคสในส่วนของการแสดงกรณีทดสอบ

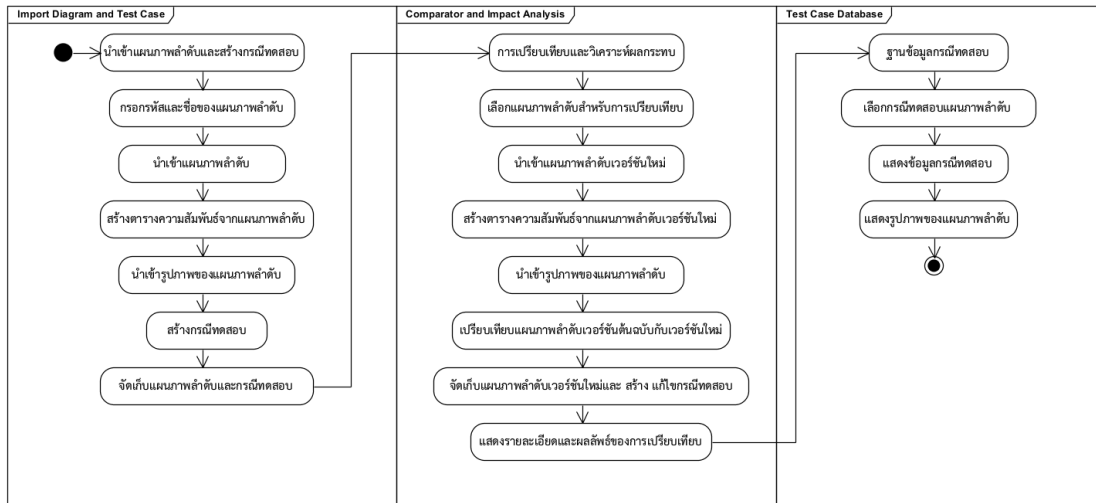


รูปที่ 3.24 แผนภาพยูสเคสในส่วนของการแสดงกรณีทดสอบ

รูปที่ 3.24 แสดงแผนภาพยูสเคสของการแสดงกรณีทดสอบที่มีอยู่ในเครื่องมือทดสอบ ผู้ใช้งานทำการเลือกข้อมูลกรณีทดสอบจากแผนภาพลำดับและเครื่องมือทดสอบจะทำการดึงข้อมูลกรณีทดสอบทั้งหมดของแผนภาพลำดับที่ถูกจากฐานข้อมูลกรณีทดสอบ (ยูสเคส Load Existing Test Cases) และทำการแสดงผลบนหน้าจอ (ยูสเคส Display Test Cases)

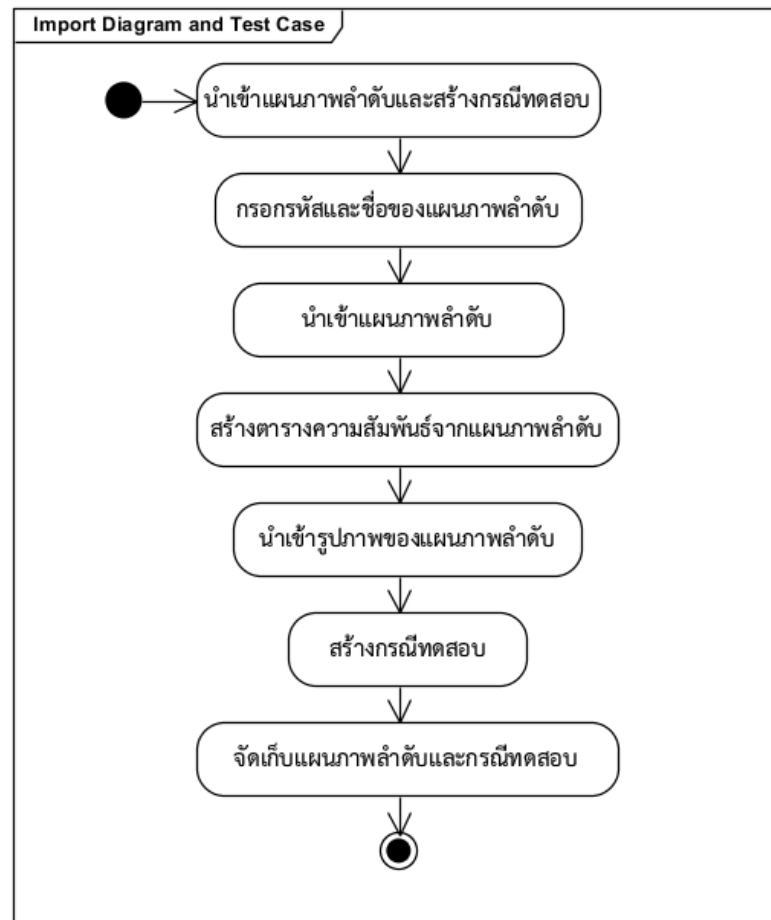
3.2.2 แผนภาพกิจกรรม

แผนภาพกิจกรรมใช้สำหรับอธิบายถึงกิจกรรมหรือขั้นตอนการทำงานในแต่ละส่วนการทำงานของเครื่องมือทดสอบ ซึ่งภาพรวมของแผนภาพกิจกรรมของเครื่องมือทดสอบจะประกอบไปด้วย 3 ส่วน ดังรูปที่ 3.25



รูปที่ 3.25 ภาพรวมของแผนภาพกิจกรรมของเครื่องมือทดสอบ

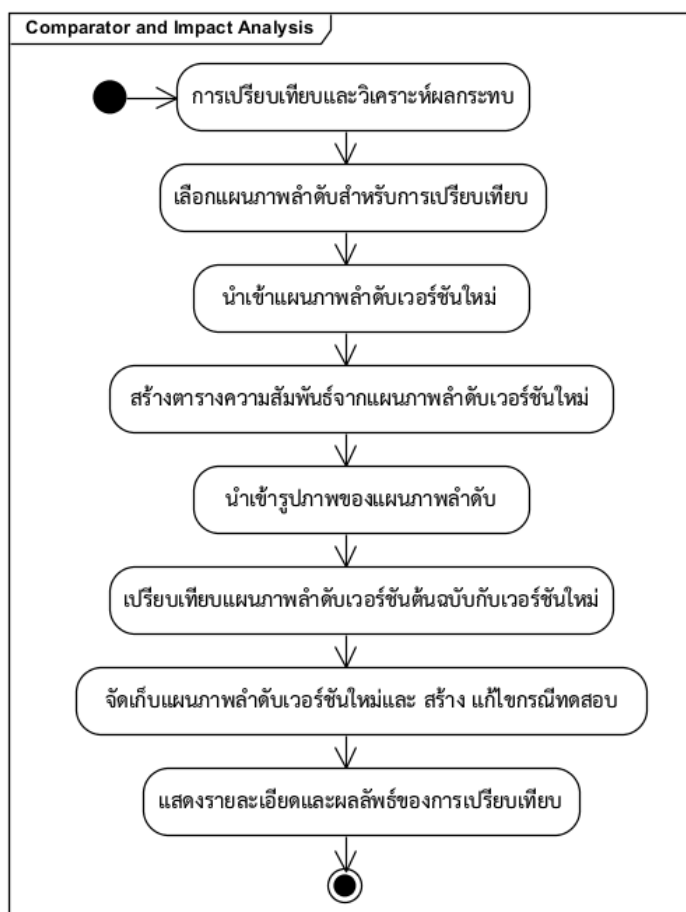
3.2.2.1 แผนภาพกิจกรรมการนำเข้าแผนภาพลำดับและการสร้างกรณีทดสอบ



รูปที่ 3.26 แผนภาพกิจกรรมของส่วนการนำเข้าแผนภาพลำดับและการสร้างกรณีทดสอบ

แผนภาพกิจกรรมจากรูป 3.26 แสดงให้เห็นถึงขั้นตอนของการนำเข้าแผนภาพลำดับและการสร้างกรณีทดสอบเข้าสู่เครื่องมือทดสอบ โดยเริ่มจากการนำเข้าแผนภาพลำดับและระบุรหัสกับชื่อของแผนภาพลำดับ ในกรณีที่มีรูปของแผนภาพลำดับก็สามารถนำเข้าได้ เพื่อที่จะนำไปใช้อ้างอิงต่อไปได้ จากนั้นทำการสร้างกรณีทดสอบและจัดเก็บเข้าสู่ฐานข้อมูลกรณีทดสอบ

3.2.2.2 แผนภาพกิจกรรมการเปรียบเทียบและการวิเคราะห์ผลกระทบ

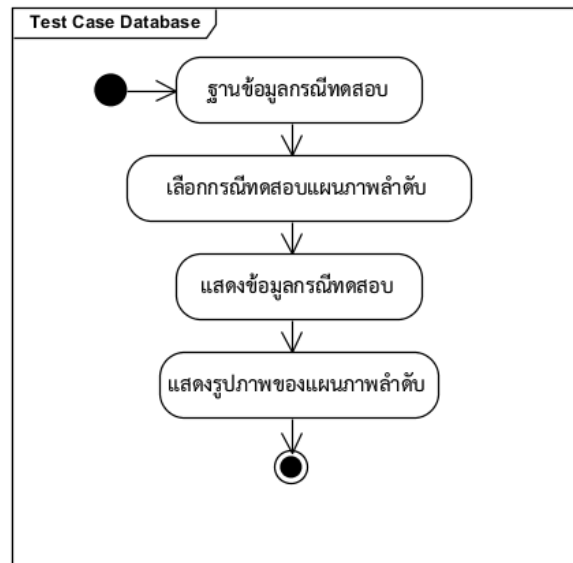


รูปที่ 3.27 แผนภาพกิจกรรมการเปรียบเทียบและการวิเคราะห์ผลกระทบ

แผนภาพกิจกรรมจากรูปที่ 3.27 แสดงให้เห็นถึงขั้นตอนของการเปรียบเทียบและการวิเคราะห์ผลกระทบจากแผนภาพลำดับเวอร์ชันต้นฉบับกับเวอร์ชันใหม่ โดยเริ่มจากการเลือกข้อมูลของแผนภาพลำดับต้นฉบับที่ถูกจัดเก็บอยู่ในระบบและทำการเลือกแผนภาพลำดับเวอร์ชันใหม่ที่จะนำไปใช้ทำการเปรียบเทียบ

ในกรณีที่มีรูปของแผนภาพลำดับเวอร์ชันใหม่ที่จะนำไปใช้อ้างอิงก็สามารถที่จะนำเข้ามาได้ จากนั้นทำการเปรียบเทียบและวิเคราะห์เพื่อหาผลกระทบของการเปลี่ยนแปลงระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับกับเวอร์ชันใหม่และจัดเก็บข้อมูลของแผนภาพลำดับเวอร์ชันใหม่และกรณีทดสอบที่ถูกสร้างใหม่หรือถูกปรับปรุงลงไปในฐานะข้อมูล ขั้นตอนสุดท้ายจะทำการแสดงรายการสรุปข้อมูลของชุดทดสอบเชิงถดถอยพร้อมทั้งแสดงรายละเอียดของผลลัพธ์ของการเปรียบเทียบ

3.2.2.3 แผนภาพกิจกรรมของการแสดงข้อมูลกรณีทดสอบ

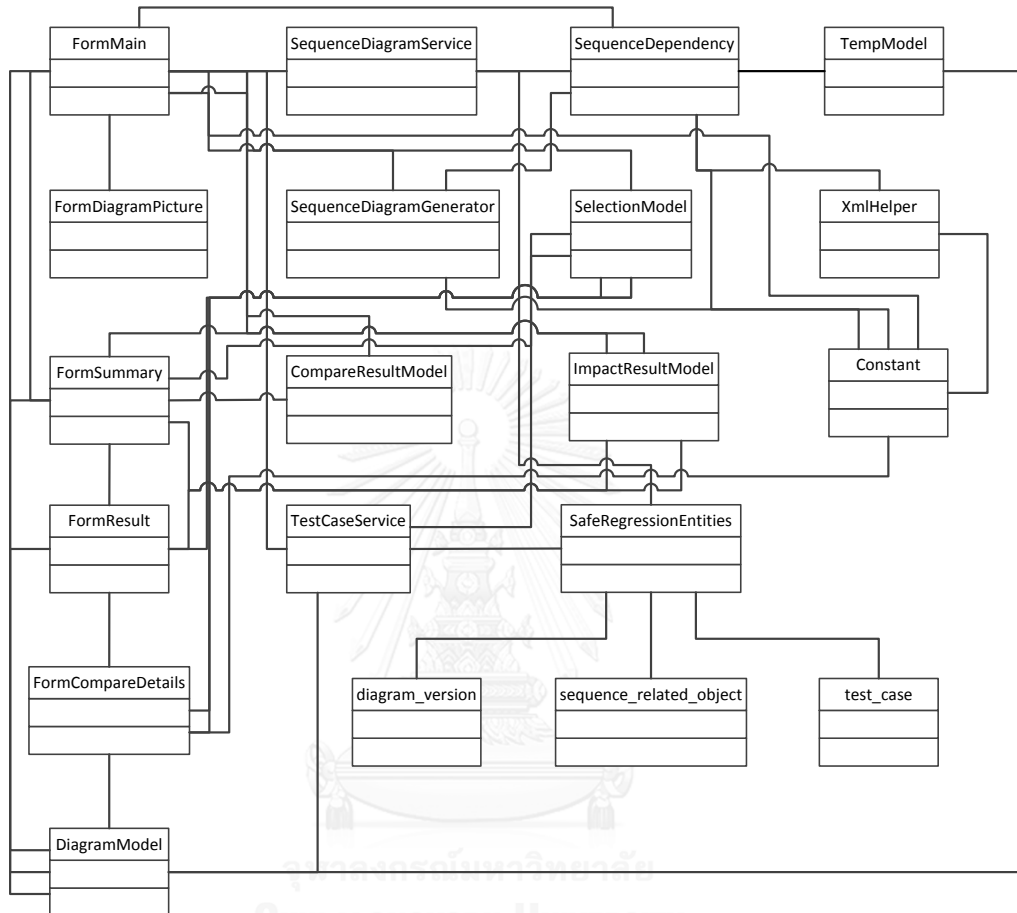


รูปที่ 3.28 แผนภาพกิจกรรมของการแสดงข้อมูลกรณีทดสอบ

แผนภาพกิจกรรมจากรูปที่ 3.28 แสดงให้เห็นถึงขั้นตอนของการแสดงข้อมูลของกรณีทดสอบ โดยจะเริ่มจากการเลือกแผนภาพลำดับที่ต้องการจะดูข้อมูล จากนั้นเครื่องมือทดสอบจะดึงข้อมูลกรณีทดสอบล่าสุดที่ถูกจัดเก็บไว้ออกมาแสดงผลบนหน้าจอ ในกรณีที่แผนภาพลำดับมีการจัดเก็บรูปของแผนภาพลำดับไว้ ก็จะสามารถแสดงรูปแผนภาพลำดับไว้สำหรับใช้อ้างอิงได้เช่นกัน

3.2.3 แผนภาพคลาส

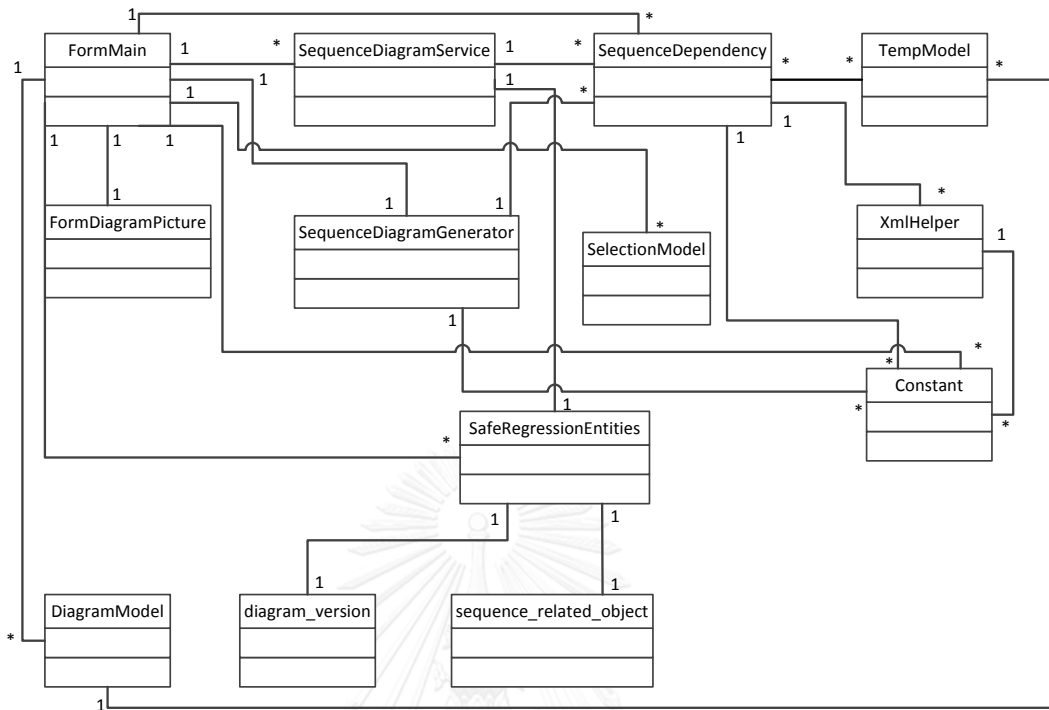
แผนภาพคลาสใช้สำหรับอธิบายโครงสร้างทั้งหมดของระบบและความสัมพันธ์ต่างๆระหว่างคลาสภายในระบบ โดยแผนภาพรวมของคลาสทั้งหมดสำหรับเครื่องมือทดสอบแสดงได้ดังรูปที่ 3.29



รูปที่ 3.29 แผนภาพรวมของคลาสของเครื่องมือทดสอบ

รูปที่ 3.29 แสดงความสัมพันธ์ของแผนภาพรวมของคลาสทั้งหมดที่ถูกออกแบบมาสำหรับใช้ในเครื่องมือการเลือกการทดสอบเชิงถดถอยแบบปลอดภัยจากแผนภาพลำดับ ซึ่งรวมไปถึงการสร้างตารางความสัมพันธ์จากแผนภาพลำดับ การเปรียบเทียบและวิเคราะห์เพื่อระบุถึงผลกระทบของการเปลี่ยนแปลงจากแผนภาพลำดับ รวมทั้งการสร้างกรณีทดสอบ และการเลือกกรณีทดสอบแบบปลอดภัยเพื่อใช้ในชุดทดสอบเชิงถดถอย

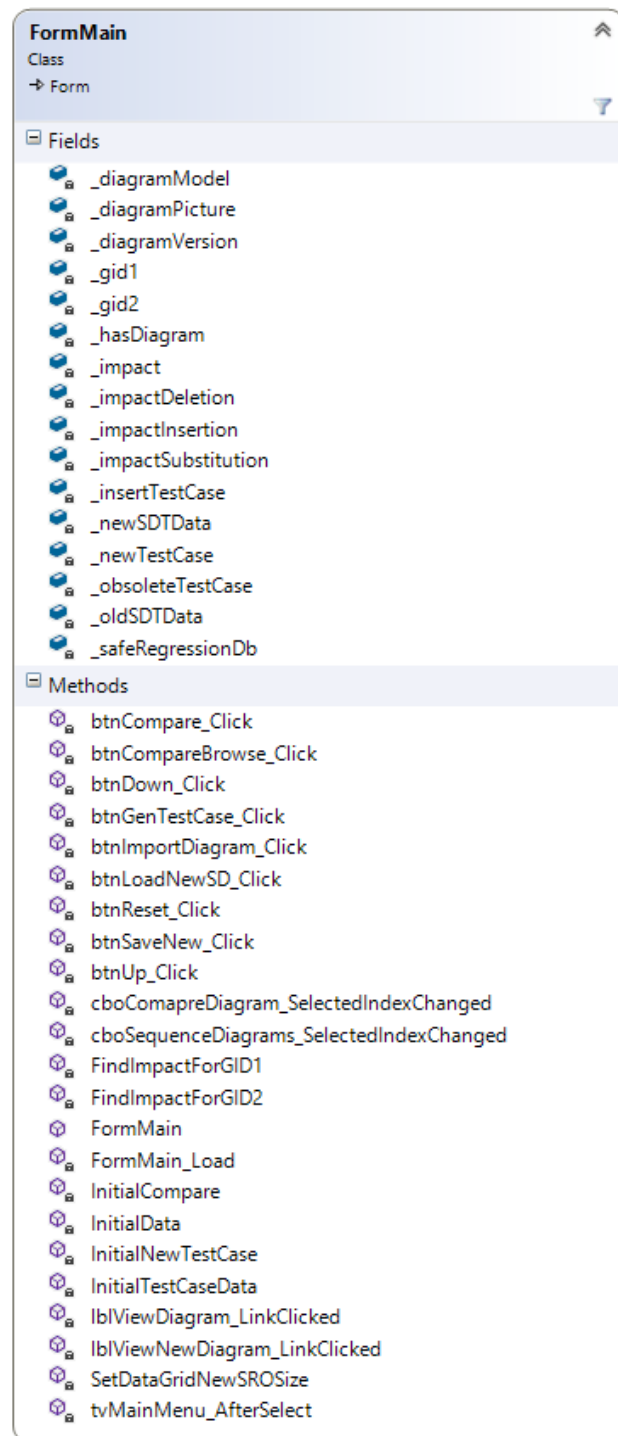
3.2.3.1 การสร้างตารางความสัมพันธ์จากแผนภาพลำดับ



รูปที่ 3.30 แผนภาพคลาสของการสร้างตารางความสัมพันธ์จากแผนภาพลำดับ

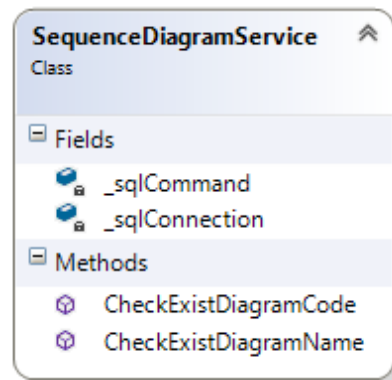
รูปที่ 3.30 แสดงความสัมพันธ์ของแผนภาพคลาสในส่วนของการสร้างตารางความสัมพันธ์จากแผนภาพลำดับ โดยจะมีรายละเอียดของแต่ละคลาส ดังต่อไปนี้

1) คลาส FormMain คือ คลาสที่ใช้สำหรับแสดงหน้าต่างของหน้าจอหลักสำหรับการจัดการใช้งานทั้งหมดที่มีในเครื่องมือ โดยหน้าจอนี้จะมีส่วนสัมพันธ์กับทุกส่วนการใช้งาน ตั้งแต่การสร้างตารางความสัมพันธ์จากแผนภาพลำดับ การเปรียบเทียบและการวิเคราะห์หาผลกระทบจากการเปลี่ยนแปลงและการเลือกกรณีทดสอบแบบปลอดภัยสำหรับชุดทดสอบเชิงถดถอย ซึ่งรายละเอียดของคลาสสามารถแสดงได้ดังรูปที่ 3.31



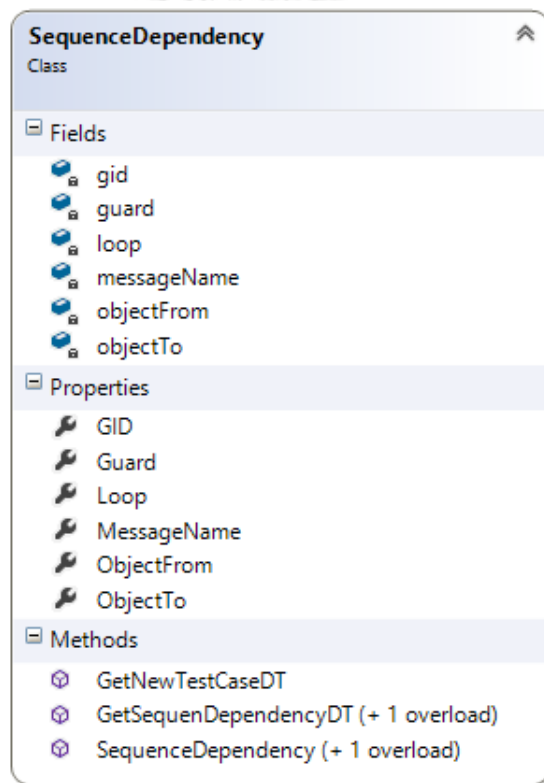
รูปที่ 3.31 คลาส FormMain

2) คลาส SequenceDiagramService คือ คลาสที่ใช้สำหรับตรวจสอบข้อมูลของชื่อและรหัสของแผนภาพลำดับ เพื่อไม่ให้มีการจัดเก็บข้อมูลของแผนภาพลำดับที่ซ้ำซ้อนกัน รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.32



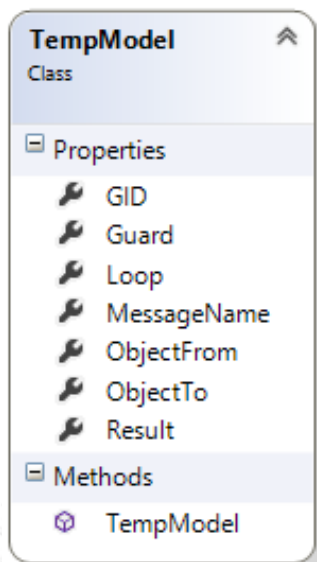
รูปที่ 3.32 คลาส SequenceDiagramService

3) คลาส SequenceDependency คือ คลาสที่ใช้เป็นแม่แบบของข้อมูลตารางความสัมพันธ์ โดยรายละเอียดของคลาสแสดงได้ดังรูปที่ 3.33



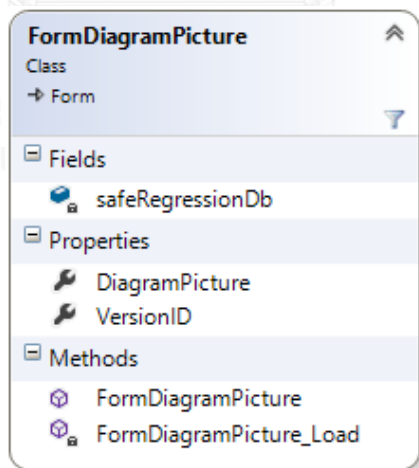
รูปที่ 3.33 คลาส SequenceDependency

4) คลาส TempModel คือ คลาสที่ใช้เป็นแม่แบบของข้อมูลที่ใช้สำหรับตารางความสัมพันธ์แบบชั่วคราว รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.34



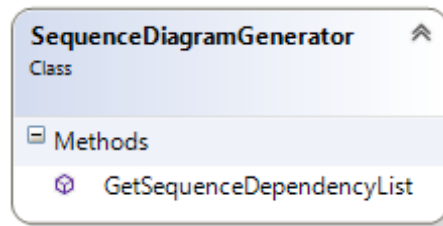
รูปที่ 3.34 คลาส TempModel

5) คลาส FormDiagramPicture คือ คลาสที่ใช้สำหรับแสดงหน้าจอของรูปภาพของแผนภาพลำดับที่ถูกรวบรวมไว้สำหรับใช้อ้างอิง รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.35



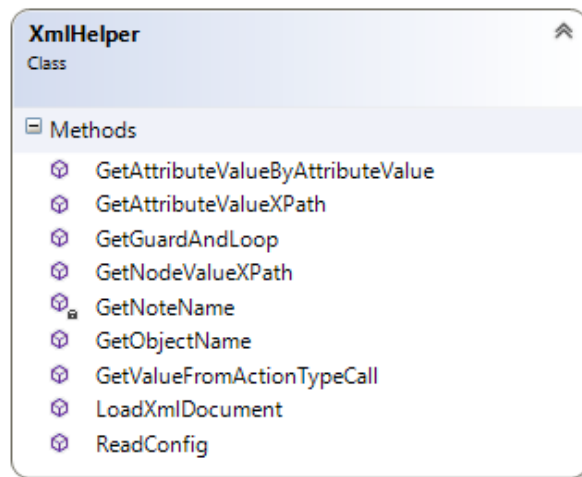
รูปที่ 3.35 คลาส FormDiagramPicture

6) คลาส SequenceDiagramGenerator คือ คลาสที่ใช้สำหรับอ่านข้อมูลจากไฟล์เอกซ์เอ็มแอลที่ถูกรวบรวมเข้ามาเพื่อใช้ในการสร้างตารางความสัมพันธ์ของแผนภาพลำดับที่ทั้งเวอร์ชันต้นฉบับและเวอร์ชันใหม่ รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.36



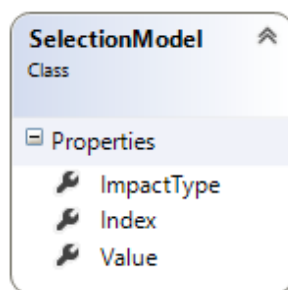
รูปที่ 3.36 คลาส SequenceDiagramGenerator

7) คลาส XmlHelper คือ คลาสที่ใช้สำหรับอ่านไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับที่ถูกรวบรวมและนำออกจากโปรแกรมวิซวลพาราไดม์ รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.37



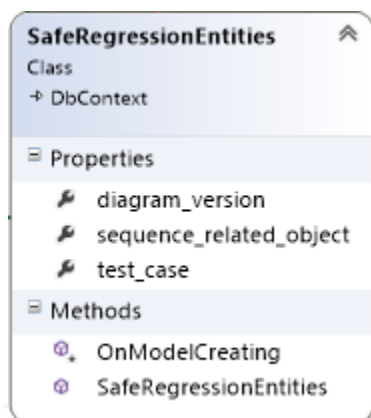
รูปที่ 3.37 คลาส XmlHelper

8) คลาส SelectionModel คือ คลาสที่ใช้สำหรับระบุตำแหน่งและข้อมูลที่ได้รับผลกระทบจากการเปลี่ยนแปลงแผนภาพลำดับ รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.38



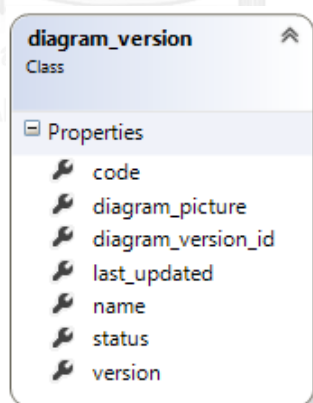
รูปที่ 3.38 คลาส SelectionModel

9) คลาส SafeRegressionEntities คือ คลาสที่ใช้ในการติดต่อฐานข้อมูลโดยจะทำการสร้างคลาสแม่แบบจากตารางที่มีอยู่ในฐานข้อมูลทั้งหมด ได้แก่ ตาราง diagram_version, sequence_related_object และ test_case โดยคลาสแม่แบบที่ถูกสร้างขึ้นมานั้นจะมีคุณสมบัติตรงกับคอลัมน์ของแต่ละตารางที่อยู่ในฐานข้อมูลและจะถูกจัดเก็บอยู่ในคลาสนี้ รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.39



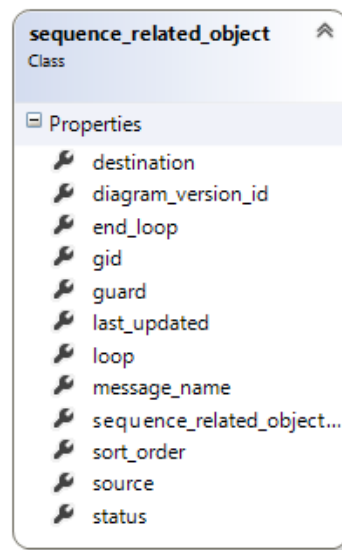
รูปที่ 3.39 คลาส SafeRegressionEntities

10) คลาส diagram_version คือ คลาสที่เป็นแม่แบบของตาราง diagram_version ที่อยู่ในฐานข้อมูล ซึ่งจะใช้สำหรับการจัดการข้อมูล เช่น ชื่อ, รหัส, สถานะ และเวอร์ชันของแผนภาพลำดับรายละเอียดของคลาสแสดงได้ดังรูปที่ 3.40



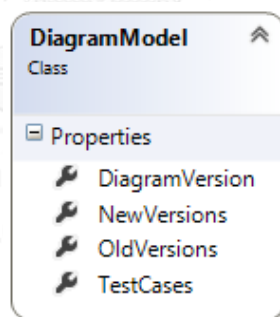
รูปที่ 3.40 คลาส diagram_version

11) คลาส sequence_related_object คือ คลาสที่เป็นแม่แบบของตาราง sequence_related_object ที่อยู่ในฐานข้อมูล ซึ่งเป็นคลาสที่ใช้สำหรับการจัดการตารางความสัมพันธ์แผนภาพลำดับ รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.41



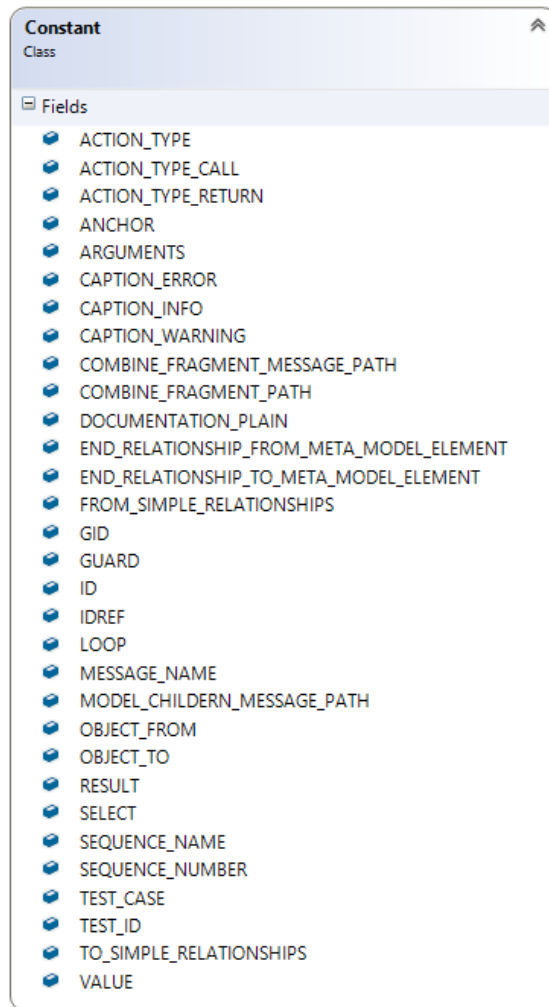
รูปที่ 3.41 คลาส sequence_related_object

12) คลาส DiagramModel คือ คลาสที่ใช้สำหรับเป็นแม่แบบของข้อมูลของตารางความสัมพันธ์เวอร์ชันต้นฉบับและเวอร์ชันใหม่ ซึ่งจะรวมถึงกรณีทดสอบสำหรับแผนภาพลำดับเวอร์ชันต้นฉบับ รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.42



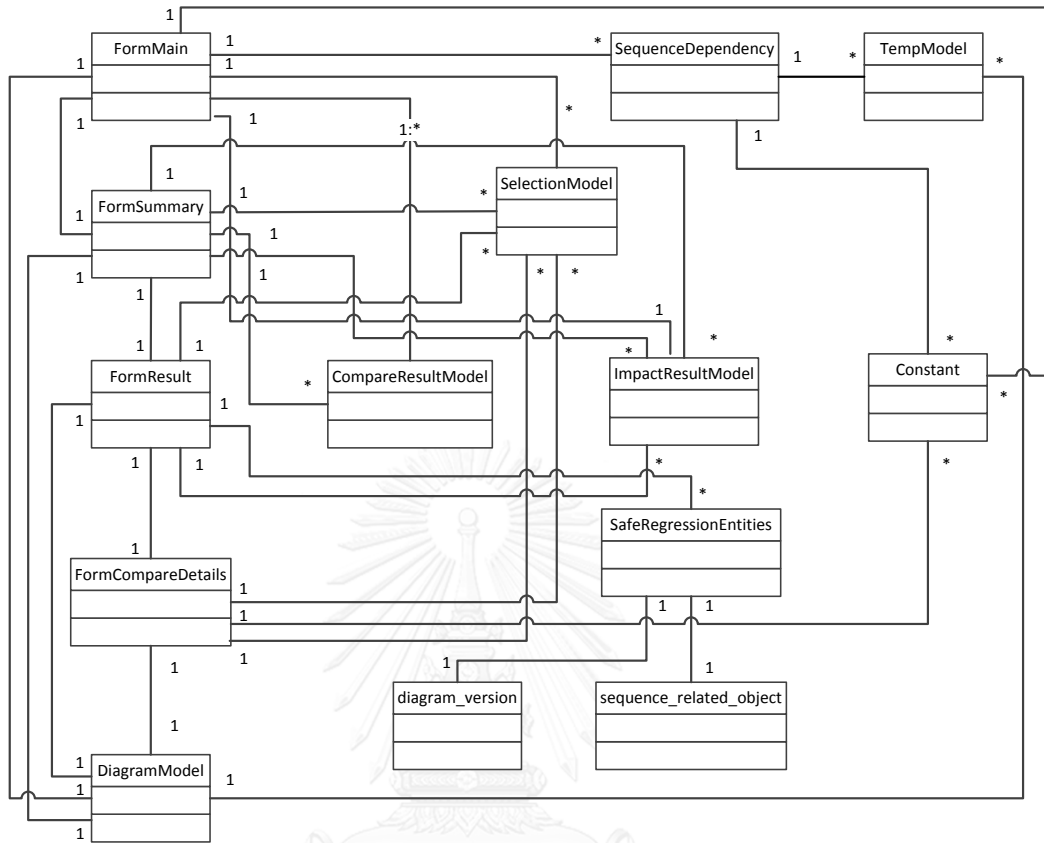
รูปที่ 3.42 คลาส DiagramModel

13) คลาส Constant คือ คลาสที่ใช้สำหรับเก็บค่าคงที่ๆจะนำไปในการทำงานในแต่ละคลาส รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.43



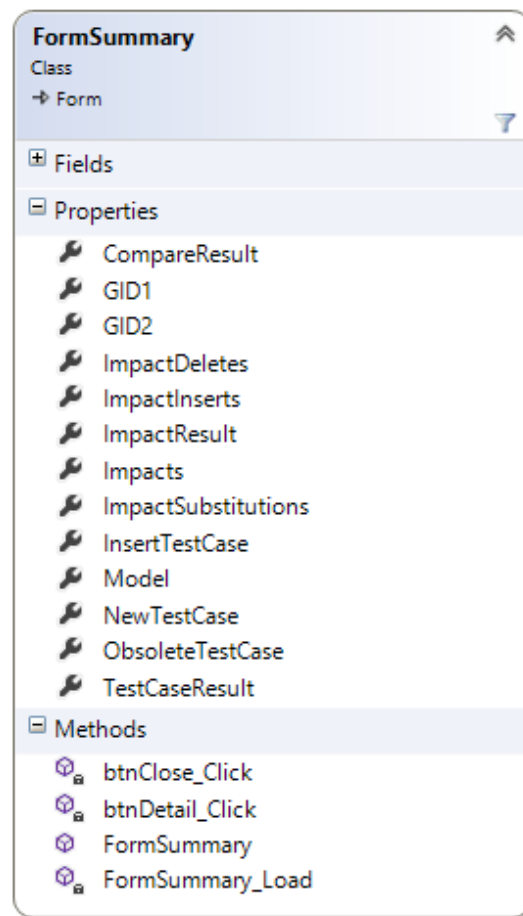
รูปที่ 3.43 คลาส Constant

3.2.3.2 การเปรียบเทียบและการวิเคราะห์หาผลกระทบการเปลี่ยนแปลง



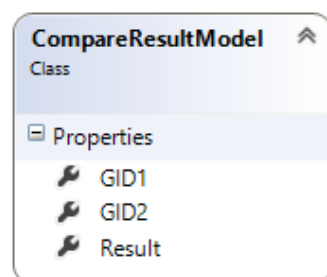
รูปที่ 3.44 แผนภาพรวมของคลาสของการเปรียบเทียบและการวิเคราะห์หาผลกระทบ

1) คลาส FormSummary คือ คลาสที่ใช้สำหรับแสดงหน้าจอของชุดทดสอบเชิงถดถอยที่สามารถนำมาใช้ในการทดสอบหลังจากที่แผนภาพลำดับเวอร์ชันต้นฉบับมีการเปลี่ยนแปลงและผ่านขั้นตอนการเปรียบเทียบและวิเคราะห์หาผลกระทบการเปลี่ยนแปลงเรียบร้อยแล้ว โดยกรณีทดสอบทั้งหมดจะถูกแสดงบนหน้าจอโดยแบ่งตามประเภทของกรณีทดสอบ ซึ่งมีด้วยกันทั้งหมด 4 ประเภท คือ กรณีทดสอบที่ถูกสร้างใหม่ กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว กรณีทดสอบที่นำกลับมาทดสอบใหม่และกรณีทดสอบที่ยังใช้ได้ทั้งหมด รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.45



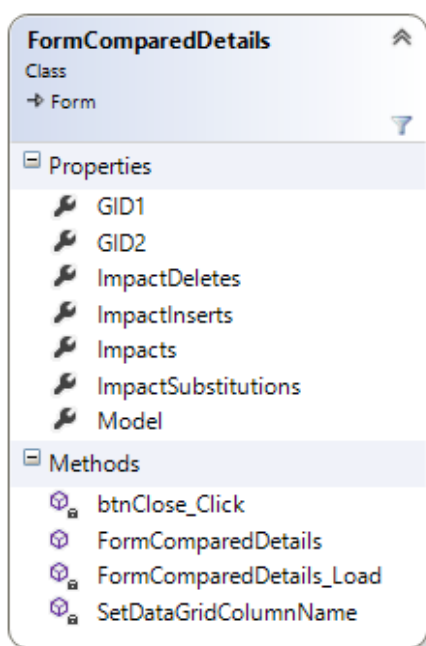
รูปที่ 3.45 คลาส FormSummary

2) คลาส CompareResultModel คือ คลาสที่ใช้สำหรับเก็บรายการของรหัสลำดับจากตารางความสัมพันธ์เวอร์ชันต้นฉบับและเวอร์ชันใหม่ รวมทั้งจะใช้เก็บและแสดงผลลัพธ์ของการเปรียบเทียบที่เกิดขึ้นจากการเปลี่ยนแปลงของแผนภาพลำดับเวอร์ชันใหม่ รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.46



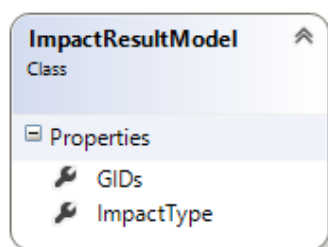
รูปที่ 3.46 คลาส CompareResultModel

3) คลาส FormCompareDetails คือ คลาสที่ใช้สำหรับแสดงหน้าจอของรายละเอียดของการเปรียบเทียบของตารางความสัมพันธ์เวอร์ชันต้นฉบับและเวอร์ชันใหม่ โดยจะแสดงรายละเอียดในรูปแบบของตารางความสัมพันธ์ซึ่งจะมีการระบุประเภทของการเปลี่ยนแปลงทั้งหมด ได้แก่ การแทรก การลบออก การแทนที่และตรงกัน รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.47



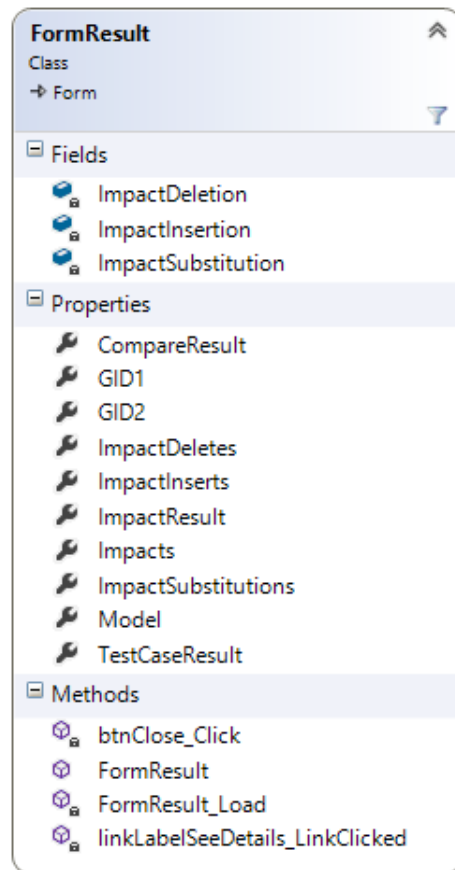
รูปที่ 3.47 คลาส FormCompareDetails

4) คลาส ImpactResultModel คือ คลาสที่ใช้ระบุถึงประเภทของผลกระทบที่ได้รับจากการเปลี่ยนแปลงของแผนภาพลำดับ โดยจะเก็บข้อมูลของรหัสลำดับในแต่ละตารางความสัมพันธ์และข้อมูลของประเภทของผลกระทบควบคู่กันไป รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.48



รูปที่ 3.48 คลาส ImpactResultModel

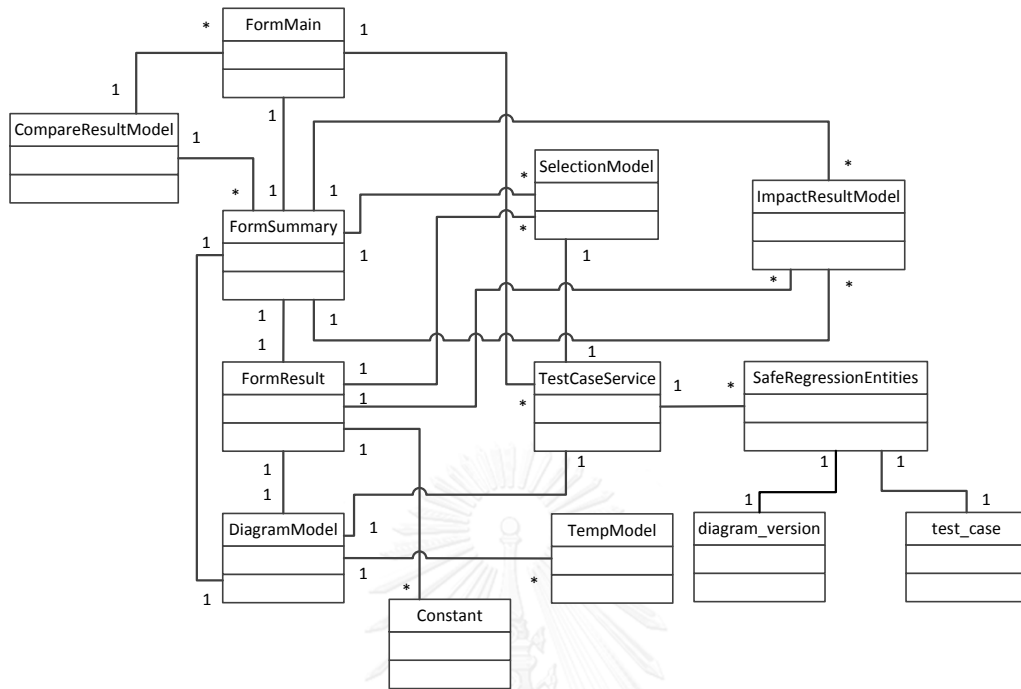
5) คลาส `FormResult` คือ คลาสที่ใช้สำหรับแสดงหน้าจอของรายละเอียดจากการเปรียบเทียบ การวิเคราะห์หาผลกระทบและกรณีทดสอบทั้งหมดที่จะใช้สำหรับการเปลี่ยนแปลงจากแผนภาพลำดับเวอร์ชันใหม่นี้ รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.49



CHULALONGKORN UNIVERSITY

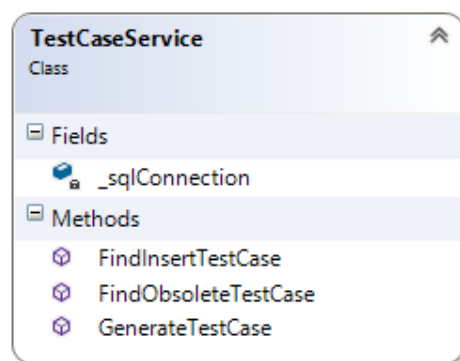
รูปที่ 3.49 คลาส `FormResult`

3.2.3.3 การเลือกชุดทดสอบเชิงถดถอย



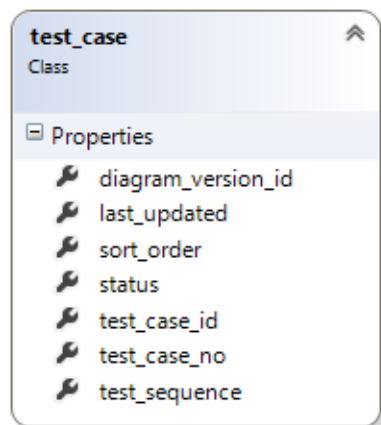
รูปที่ 3.50 แผนภาพคลาสของการเลือกชุดทดสอบเชิงถดถอย

1) คลาส TestCaseService คือ คลาสที่ใช้สำหรับการติดต่อฐานข้อมูลเกี่ยวกับกรณีทดสอบ ซึ่งจะรวมไปถึงการค้นหาข้อมูลของกรณีทดสอบที่มีอยู่ในฐานข้อมูล การสร้างกรณีทดสอบใหม่ๆ เพื่อให้ครอบคลุมในทุกส่วนของการแก้ไขเปลี่ยนแปลงและใช้สำหรับจัดเก็บข้อมูลของกรณีทดสอบเข้าไปยังฐานข้อมูลกรณีทดสอบ รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.51



รูปที่ 3.51 คลาส TestCaseService

2) คลาส test_case คือ คลาสที่ใช้สำหรับเป็นแม่แบบของข้อมูลกรณีทดสอบที่อยู่ในฐานข้อมูลกรณีทดสอบ รายละเอียดของคลาสแสดงได้ดังรูปที่ 3.52



รูปที่ 3.52 คลาส test_case



บทที่ 4

การพัฒนาเครื่องมือ

ในบทนี้จะกล่าวถึงรายละเอียดการพัฒนาเครื่องมือการเลือกกรณีทดสอบเชิงทดลองโดยแผนภาพลำดับ โดยจะกล่าวถึงการออกแบบส่วนต่อประสานของผู้ใช้งาน ขั้นตอนการพัฒนา ตัวอย่างการใช้งาน และสภาพแวดล้อมที่ใช้ในการพัฒนา ซึ่งมีรายละเอียดดังต่อไปนี้

4.1 สภาพแวดล้อมที่ใช้ในการพัฒนาเครื่องมือ

4.1.1 ฮาร์ดแวร์ (Hardware)

- เครื่องคอมพิวเตอร์ขนาดสมุดบันทึก (Notebook computer) หน่วยประมวลผลอินเทลคอร์ไอเซเว่น 2.7 กิกะเฮิร์ต (Intel Core i7 2.7 GHz)
- หน่วยความจำ (RAM) 16 กิกะไบต์ (16 GB)
- ฮาร์ดดิสก์ (Hard Disk) 250 กิกะไบต์ (250 GB)

4.1.2 ซอฟต์แวร์ (Software)

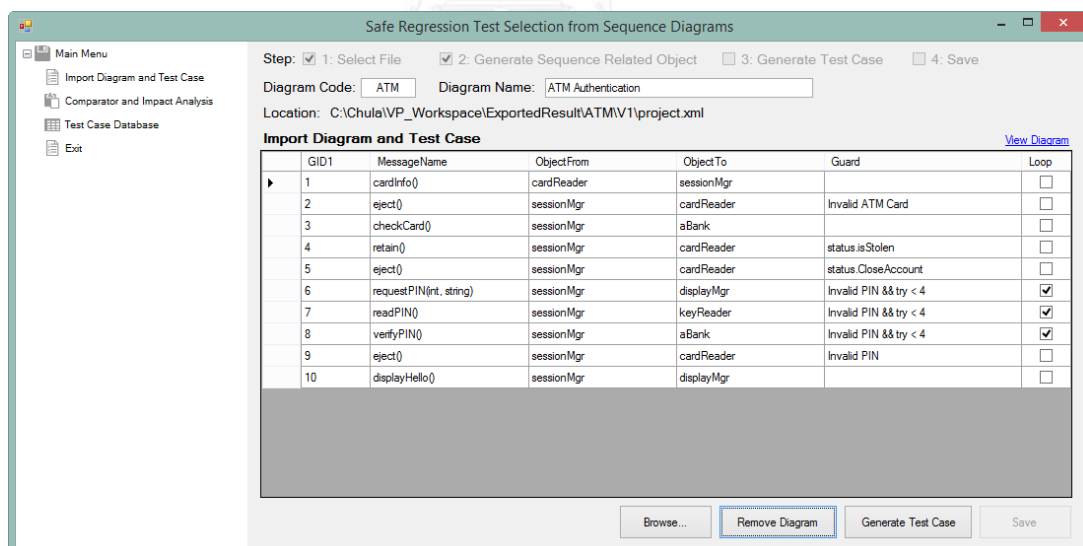
- ระบบปฏิบัติการไมโครซอฟท์วินโดวส์ 8.1 โพรเฟสชันนอล (Microsoft Windows 8.1 Professional)
- ไมโครซอฟท์วิซวลสตูดิโอ 2013 โพรเฟสชันแนล (Microsoft Visual Studio 2013 Professional) จุฬาลงกรณ์มหาวิทยาลัย
- ไมโครซอฟท์เอสคิวแอล เซิร์ฟเวอร์ เอ็กซ์เพรส 2008 (Microsoft SQL Server 2008)
- วิซวลพาราไดม์ ฟอรั ยูเอ็มแอล คอมมูนิตี้อิdition (Visual Paradigm for UML community edition)

4.2 โครงสร้างส่วนต่อประสานของผู้ใช้งาน

โครงสร้างส่วนต่อประสานของเครื่องมือ ประกอบด้วยการทำงานหลักทั้งหมด 3 ส่วนด้วยกัน ดังต่อไปนี้

4.2.1 เครื่องมือการสร้างตารางความสัมพันธ์และกรณีทดสอบ

หน้าจอหลักของเครื่องมือการสร้างตารางความสัมพันธ์จะเป็นหน้าจอที่จะถูกใช้เป็นขั้นตอนแรกของการทำงานทั้งหมด เพราะข้อมูลของแผนภาพลำดับจะต้องถูกนำเข้ามาสร้างตารางความสัมพันธ์และกรณีทดสอบก่อนที่จะเริ่มมีการแก้ไขข้อมูลของแผนภาพลำดับเวอร์ชันถัดไป โดยวิธีการนำเข้าข้อมูลของแผนภาพลำดับและการสร้างกรณีทดสอบนั้นจะอยู่ในรายการเลือก Import Diagram and Test Case โดยจะเป็นการนำเข้าข้อมูลของแผนภาพลำดับที่ถูกออกแบบมาแล้วจากโปรแกรมวิซวลพาราไดม์และถูกนำออกมาในรูปแบบเอกซ์เอ็มแอลไฟล์ โดยขั้นตอนการนำเข้าแผนภาพลำดับทำได้โดยการกดปุ่ม Browse เพื่อทำการเลือกไฟล์ของแผนภาพลำดับ จากนั้นข้อมูลของตารางความสัมพันธ์เวอร์ชันต้นฉบับของแผนภาพลำดับจะถูกสร้างและแสดงในตารางบนหน้าจอ โดยผู้ใช้งานจำเป็นต้องกรอกรหัสและชื่อของแผนภาพลำดับ ที่ไม่ซ้ำกับที่เคยมีอยู่ในฐานข้อมูล เพื่อใช้ระบุถึงรายละเอียดของแผนภาพลำดับนั้นๆ ได้อย่างถูกต้อง โดยตัวอย่างหน้าจอแสดงได้ดังรูปที่ 4.1

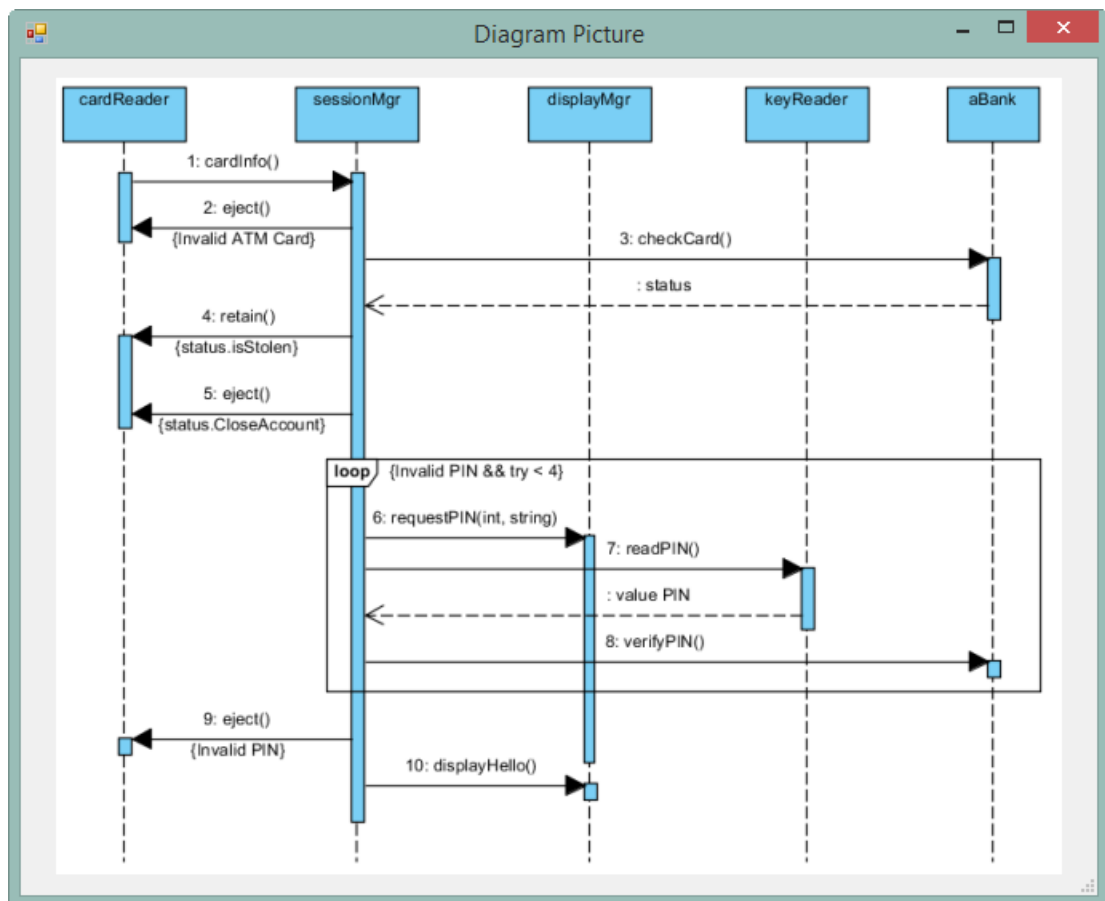


รูปที่ 4.1 หน้าจอตารางความสัมพันธ์เวอร์ชันต้นฉบับที่ถูกสร้างจากแผนภาพลำดับเวอร์ชันแรก

ในขั้นตอนการนำเข้าแผนภาพลำดับและการสร้างตารางความสัมพันธ์นี้ ผู้ใช้งานสามารถที่จะจัดเก็บรูปของแผนภาพลำดับที่ถูกออกแบบและนำออกมาเพื่อใช้ในการอ้างอิงเข้าไปในฐานข้อมูลเพื่อที่จะสามารถใช้ในการเรียกดูภายหลังได้ด้วยเช่นกัน โดยการกดปุ่ม Import Diagram และทำการ

เลือกรูปภาพของแผนภาพลำดับ โดยจะสนับสนุนการจัดเก็บไฟล์ภาพเฉพาะนามสกุล jpg, jpeg และ png เท่านั้น

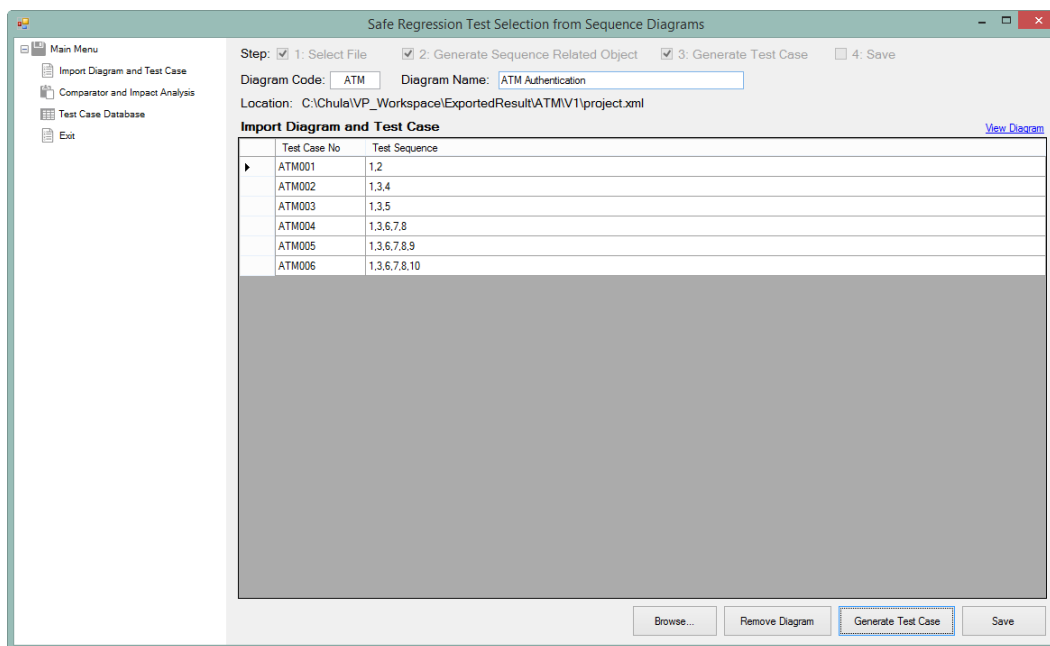
โดยรูปของแผนภาพลำดับจะถูกนำไปจัดเก็บในฐานข้อมูลพร้อมกับตารางความสัมพันธ์และกรณีทดสอบ โดยตัวอย่างหน้าจอของรูปของแผนภาพลำดับแสดงได้ดังรูปที่ 4.2



รูปที่ 4.2 หน้าจอแสดงรูปของแผนภาพลำดับเวอร์ชันต้นฉบับ

หลังจากขั้นตอนการสร้างตารางความสัมพันธ์และการนำเข้ารูปภาพของแผนภาพลำดับที่จะนำไปใช้สำหรับอ้างอิงเรียบร้อยแล้ว ขั้นตอนต่อไปคือการสร้างกรณีทดสอบจากแผนภาพลำดับ โดยใช้ข้อมูลจากตารางความสัมพันธ์ที่แสดงอยู่ในตารางความสัมพันธ์บนหน้าจอของเครื่องมือทดสอบ โดยขั้นตอนของการสร้างกรณีทดสอบนั้นสามารถทำได้โดยการกดปุ่ม Generate Test Case

กรณีทดสอบจะถูกสร้างขึ้นและแสดงผลอยู่ในตารางบนหน้าจอของเครื่องมือ ดังที่แสดงในรูป 4.3 โดยจะแสดงข้อมูลของรหัสกรณีทดสอบและลำดับการทดสอบ ขั้นตอนต่อไป คือ การจัดเก็บข้อมูลของตารางความสัมพันธ์และกรณีทดสอบที่ถูกสร้างขึ้นทั้งหมดลงไปพื้นฐานข้อมูล โดยการกดปุ่ม Save เมื่อเสร็จสิ้นขั้นตอนจัดเก็บข้อมูลก็จะสามารถนำข้อมูลของตารางความสัมพันธ์และกรณีทดสอบไปใช้ในการใช้งานต่อไปได้



รูปที่ 4.3 หน้าจอการสร้างกรณีทดสอบจากแผนภาพลำดับเวอร์ชันต้นฉบับ

4.2.2 เครื่องมือเปรียบเทียบและหาผลกระทบของการเปลี่ยนแปลง

หน้าจอหลักของเครื่องมือเปรียบเทียบและการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลง จะอยู่ในรายการเลือก Comparator and Impact Analysis ซึ่งหน้าจอนี้จะประกอบด้วย 3 ส่วน ได้แก่ ส่วนการสร้างตารางความสัมพันธ์จากแผนภาพลำดับเวอร์ชันใหม่ ส่วนการแสดงรายการของชุดทดสอบเชิงถดถอยและส่วนการแสดงรายละเอียดของการเปรียบเทียบและการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลงของแผนภาพลำดับระหว่างเวอร์ชันต้นฉบับกับเวอร์ชันใหม่

โดยขั้นตอนการทำงานจะเริ่มจาก ผู้ใช้งานทำการแก้ไขแผนภาพลำดับจากแผนภาพลำดับเวอร์ชันต้นฉบับให้รองรับตามความต้องการทางธุรกิจจากโปรแกรมวิซวลพาราไดม์และทำการนำออกมาในรูปแบบไฟล์เอกซ์เอ็มแอลเพื่อใช้ในการสร้างตารางความสัมพันธ์เวอร์ชันใหม่ในเครื่องมือทดสอบต่อไป

ในขั้นตอนการเปรียบเทียบและการวิเคราะห์หาผลกระทบของการแก้ไขนั้น ผู้ใช้งานจะต้องทำการเลือกข้อมูลของแผนภาพลำดับเวอร์ชันต้นฉบับก่อน โดยการเลือกจากรายชื่อของแผนภาพลำดับที่อยู่ในคอมโบบ็อกซ์เพื่อที่จะนำมาใช้เป็นข้อมูลสำหรับการเปรียบเทียบและวิเคราะห์หาผลกระทบจากการเปลี่ยนแปลงจากแผนภาพลำดับเวอร์ชันใหม่ โดยการนำเข้าแผนภาพลำดับเวอร์ชันใหม่นั้นสามารถทำได้โดยการกดปุ่ม Browse และทำการเลือกไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับเวอร์ชันใหม่ จากนั้นเครื่องมือทดสอบจะทำการสร้างตารางความสัมพันธ์จากแผนภาพลำดับเวอร์ชันใหม่ที่ถูกเลือก โดยหน้าจอการสร้างตารางความสัมพันธ์เวอร์ชันใหม่แสดงได้ดังรูปที่ 4.4

Original Sequence Dependency Table	New Sequence Dependency Table				
GID1	MessageName	ObjectFrom	ObjectTo	Guard	Loop
1	cardInfo()	cardReader	sessionMgr		<input type="checkbox"/>
2	eject()	sessionMgr	cardReader	Invalid ATM Card	<input type="checkbox"/>
3	checkCard()	sessionMgr	sBank		<input type="checkbox"/>
4	retain()	sessionMgr	cardReader	status.isStolen	<input type="checkbox"/>
5	eject()	sessionMgr	cardReader	status.CloseAccount	<input type="checkbox"/>
6	requestPIN(int, string)	sessionMgr	displayMgr	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
7	readPIN()	sessionMgr	keyReader	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
8	verifyPIN()	sessionMgr	sBank	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
9	eject()	sessionMgr	cardReader	Invalid PIN	<input type="checkbox"/>
10	displayHello()	sessionMgr	displayMgr		<input type="checkbox"/>

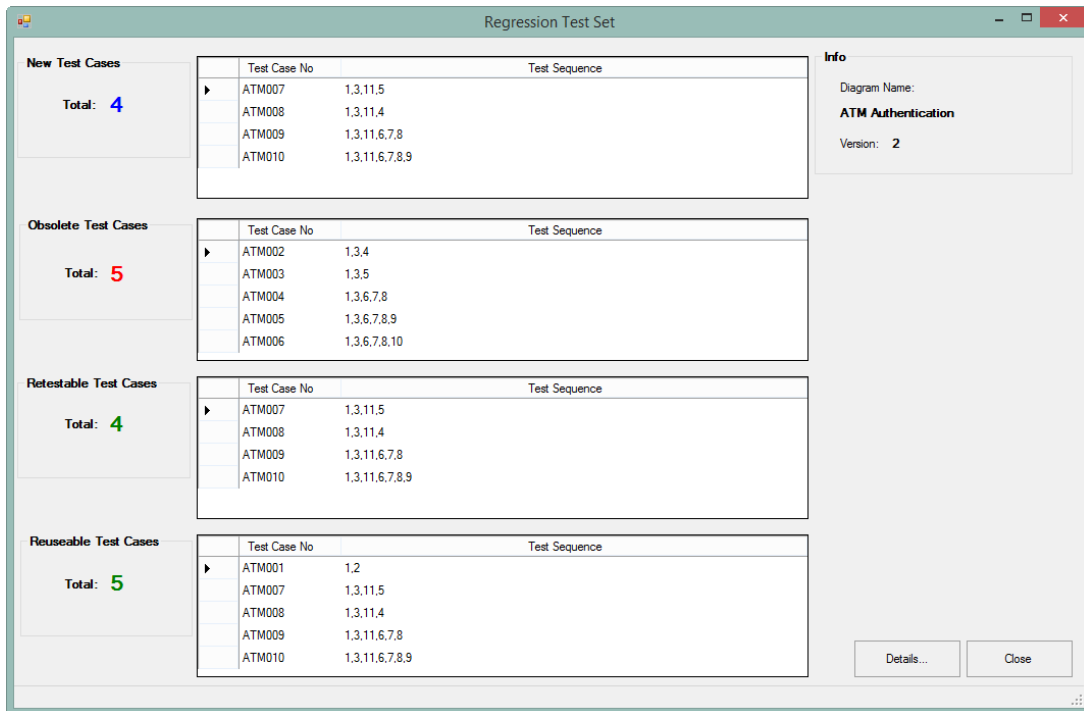
New Sequence Dependency Table					
GID2	MessageName	ObjectFrom	ObjectTo	Guard	Loop
1	cardInfo()	cardReader	sessionMgr		<input type="checkbox"/>
2	eject()	sessionMgr	cardReader	Invalid ATM Card	<input type="checkbox"/>
3	checkCard()	sessionMgr	sBank		<input type="checkbox"/>
11	selectLanguage(int)	cardReader	displayMgr		<input type="checkbox"/>
5	eject()	sessionMgr	cardReader	status.CloseAccount	<input type="checkbox"/>
4	retain()	sessionMgr	cardReader	status.isStolen	<input type="checkbox"/>
6	requestPIN(int, string)	sessionMgr	displayMgr	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
7	readPIN()	sessionMgr	keyReader	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
8	verifyPIN()	sessionMgr	sBank	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
9	eject()	sessionMgr	cardReader	Invalid PIN	<input type="checkbox"/>

รูปที่ 4.4 หน้าจอของตารางความสัมพันธ์จากแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่

ผู้ใช้งานสามารถที่จะจัดเก็บรูปของแผนภาพลำดับเวอร์ชันใหม่ไว้ใช้สำหรับอ้างอิงได้ โดยรูปของแผนภาพลำดับเวอร์ชันใหม่จะถูกจัดเก็บแทนที่รูปของแผนภาพลำดับเวอร์ชันต้นฉบับซึ่งจะสามารถใช้ในการเรียกดูภายหลังได้ โดยการกดปุ่ม Import Diagram และทำการเลือกรูปภาพของแผนภาพลำดับ โดยจะสนับสนุนการจัดเก็บไฟล์ภาพเฉพาะนามสกุล jpg, jpeg และ png เท่านั้น

หลังจากขั้นตอนการนำเข้าข้อมูลของแผนภาพลำดับเวอร์ชันใหม่และสร้างตารางความสัมพันธ์เรียบร้อยแล้ว ผู้ใช้งานสามารถทำการเปรียบเทียบและการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลงได้โดยการกดปุ่ม Compare จากนั้นเครื่องมือทดสอบจะทำการเปรียบเทียบและการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลงระหว่างตารางความสัมพันธ์เวอร์ชันต้นฉบับและเวอร์ชันใหม่ รวมถึงการจัดการกรณีทดสอบสำหรับกรณีทดสอบที่ได้รับผลกระทบจากการแก้ไขและทำการค้นหาและเลือกกรณีทดสอบแบบปลอดภัยสำหรับนำไปใช้ในชุดทดสอบเชิงถดถอยที่จะนำมาใช้ในการทดสอบต่อไป

หน้าจอชุดทดสอบเชิงถดถอยจะแบ่งและแสดงประเภทของกรณีทดสอบทั้งหมด 4 ประเภทด้วยกัน ได้แก่ กรณีทดสอบที่ถูกสร้างใหม่ กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว กรณีทดสอบที่นำกลับมาทดสอบใหม่และกรณีทดสอบที่ยังใช้ได้ทั้งหมด โดยชุดทดสอบเชิงถดถอยจะแสดงได้ดังรูปที่ 4.5



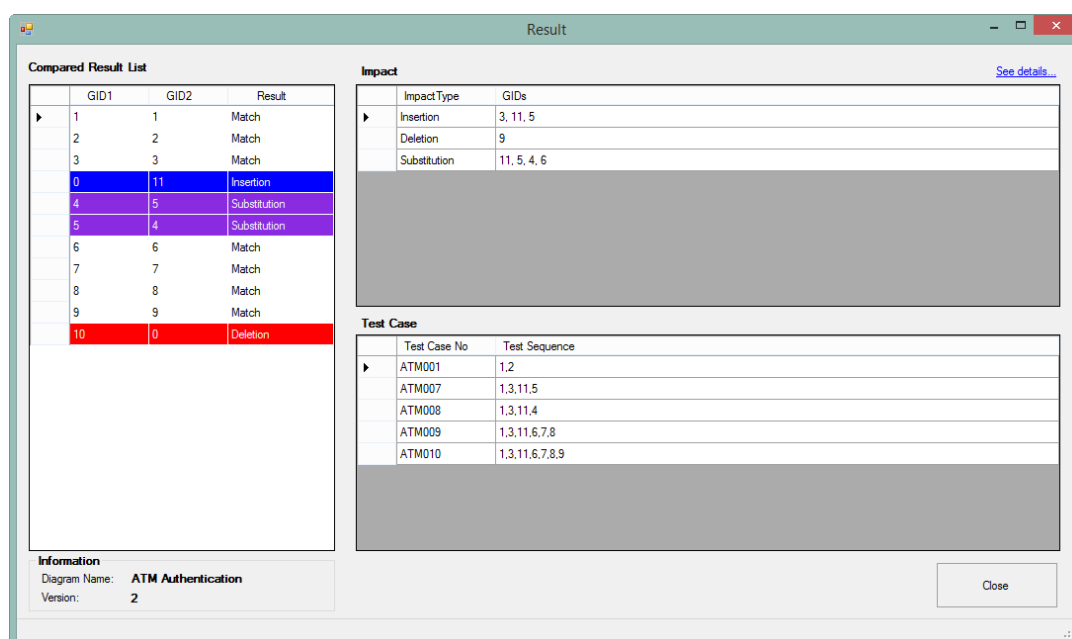
รูปที่ 4.5 หน้าจอชุดทดสอบเชิงถดถอยสำหรับนำไปใช้ทดสอบกับแผนภาพลำดับเวอร์ชันใหม่

ในกรณีที่ผู้ใช้งานต้องการที่จะตรวจสอบรายละเอียดของการเปรียบเทียบ การวิเคราะห์หาผลกระทบของการเปลี่ยนแปลงและการจัดการกรณีทดสอบนั้น ผู้ใช้งานสามารถกดปุ่ม Details เพื่อให้เครื่องมือทดสอบทำการเปิดหน้าจอสำหรับแสดงรายละเอียดของการเปรียบเทียบและการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลงระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่ได้

หน้าจอส่วนการแสดงผลรายละเอียดของการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลงของแผนภาพลำดับระหว่างเวอร์ชันต้นฉบับกับเวอร์ชันใหม่นั้นจะแสดงผลรายละเอียดของการเปรียบเทียบและระบุถึงข้อมูลในแผนภาพลำดับที่ได้รับผลกระทบต่อการเปลี่ยนแปลงจากแผนภาพลำดับเวอร์ชันใหม่ โดยตารางชื่อ Compared Result List จะแสดงรหัสจากตารางความสัมพันธ์ที่มีการเปลี่ยนแปลงและระบุสีในพื้นที่หลังของแต่ละระเบียน โดยผลลัพธ์ของการเปรียบเทียบและการระบุสีในพื้นที่หลังจะแบ่งออกเป็น 4 ประเภท ได้แก่ การแทรกจะใช้พื้นที่หลังเป็นสีน้ำเงิน, การลบออกจะใช้พื้นที่หลังเป็นสีแดง, การแทนที่จะใช้พื้นที่หลังเป็นสีม่วงและตรงกันจะไม่มีสีระบุ

ผลลัพธ์ของการเปรียบเทียบนั้นจะทำให้เกิดผลกระทบทางตรงต่อข้อมูลที่อยู่ในตารางความสัมพันธ์แผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่ โดยผลกระทบทั้งหมดจะถูกนำไปค้นหาและนำมาแสดงอยู่ในตารางชื่อ Impact ซึ่งจะแสดงประเภทของผลกระทบทั้งหมด 3 ประเภท ได้แก่ การเพิ่ม การลบออกและการแทนที่

หลังจากผ่านขั้นตอนการเปรียบเทียบและการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลงเรียบร้อยแล้ว เครื่องมือทดสอบจะทำการคัดเลือกกรณีทดสอบแบบปลอดภัย เพื่อที่จะนำไปใช้สำหรับชุดทดสอบเชิงถดถอย โดยกรณีทดสอบทั้งหมดที่เกี่ยวข้องกับแผนภาพลำดับเวอร์ชันใหม่นี้จะถูกแสดงอยู่ในตารางชื่อ Test Case ซึ่งรายละเอียดของกรณีทดสอบจะแสดงได้ดังรูปที่ 4.6



รูปที่ 4.6 หน้าจอส่วนการแสดงผลรายละเอียดของการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลง

หน้าจอแสดงรายละเอียดของการระบุผลกระทบจากการเปลี่ยนแปลงจะแสดงก็ต่อเมื่อผู้ใช้งานกดปุ่มไฮเปอร์ลิงค์ See details จากมุมมองของหน้าจอส่วนการแสดงผลรายละเอียดของการวิเคราะห์หาผลกระทบของการเปลี่ยนแปลง โดยตัวอย่างของหน้าจอจะแสดงได้ดังรูปที่ 4.7 โดยข้อมูลที่จะแสดงจะเป็นข้อมูลของตารางความสัมพันธ์แผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่ที่ได้รับผลกระทบจากการเปลี่ยนแปลงจากแผนภาพลำดับเวอร์ชันใหม่ โดยตารางความสัมพันธ์เวอร์ชันต้นฉบับนั้นจะใช้ในการแสดงถึงผลกระทบจากการลบออกเท่านั้น โดยข้อมูลที่ถูกลบออกไปนั้นจะแสดงพื้นหลังเป็นสีแดงส่วนข้อมูลที่จะได้รับผลกระทบจากการลบออกนั้นจะแสดงพื้นหลังเป็นสีส้ม

ส่วนตารางความสัมพันธ์เวอร์ชันใหม่นั้นจะใช้ในการแสดงถึงผลกระทบจากการแทรกและการแทนที่ โดยข้อมูลที่ถูกแทรกจะแสดงพื้นหลังเป็นสีน้ำเงินและข้อมูลที่มีการแทนที่ที่แสดงพื้นหลังเป็นสีม่วง ส่วนผลกระทบที่ได้รับจากการแทรกและแทนที่นั้นจะแสดงพื้นหลังเป็นสีส้มเช่นเดียวกับการลบออก โดยจะสามารถดูรายละเอียดของสีได้จากคำอธิบายที่อยู่ในกรอบ Legend

Original Sequence Dependency Table

GID1	MessageName	ObjectFrom	ObjectTo	Guard	Loop
1	cardInfo()	cardReader	sessionMgr		<input type="checkbox"/>
2	eject()	sessionMgr	cardReader	Invalid ATM Card	<input type="checkbox"/>
3	checkCard()	sessionMgr	aBank		<input type="checkbox"/>
4	retain()	sessionMgr	cardReader	status.isStolen	<input type="checkbox"/>
5	eject()	sessionMgr	cardReader	status.CloseAccount	<input type="checkbox"/>
6	requestPIN(int, string)	sessionMgr	displayMgr	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
7	readPIN()	sessionMgr	keyReader	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
8	verifyPIN()	sessionMgr	aBank	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
9	eject()	sessionMgr	cardReader	Invalid PIN	<input type="checkbox"/>
10	displayHello()	sessionMgr	displayMgr		<input type="checkbox"/>

New Sequence Dependency Table

GID2	MessageName	ObjectFrom	ObjectTo	Guard	Loop
1	cardInfo()	cardReader	sessionMgr		<input type="checkbox"/>
2	eject()	sessionMgr	cardReader	Invalid ATM Card	<input type="checkbox"/>
3	checkCard()	sessionMgr	aBank		<input type="checkbox"/>
11	selectLanguage(int)	cardReader	displayMgr		<input type="checkbox"/>
5	eject()	sessionMgr	cardReader	status.CloseAccount	<input type="checkbox"/>
4	retain()	sessionMgr	cardReader	status.isStolen	<input type="checkbox"/>
6	requestPIN(int, string)	sessionMgr	displayMgr	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
7	readPIN()	sessionMgr	keyReader	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
8	verifyPIN()	sessionMgr	aBank	Invalid PIN && try < 4	<input checked="" type="checkbox"/>
9	eject()	sessionMgr	cardReader	Invalid PIN	<input type="checkbox"/>

Legend
 Insertion = Blue Deletion = Red Substitution = Blue Violet Impacted = Orange

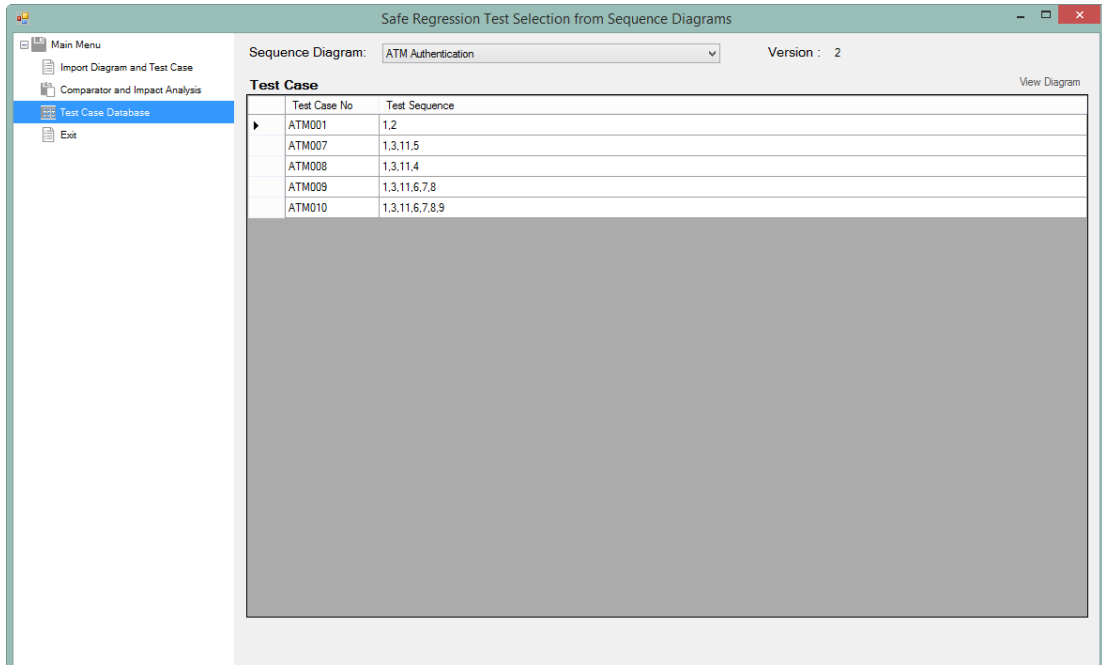
รูปที่ 4.7 หน้าจอแสดงรายละเอียดของการระบุผลกระทบจากการเปลี่ยนแปลง

4.2.3 เครื่องมือแสดงชุดทดสอบ

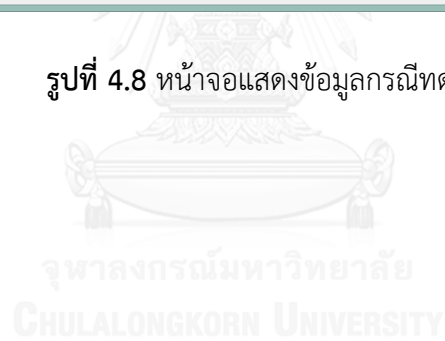
ผู้ใช้งานสามารถเรียกดูกรณีทดสอบสำหรับแต่ละแผนภาพลำดับได้จากหน้าจอฐานข้อมูลกรณีทดสอบ โดยจะอยู่ในรายการเลือก Test Case Database ซึ่งผู้ใช้งานจะต้องทำการเลือกชื่อของแผนภาพลำดับที่ต้องการจะดูรายละเอียดจากคอมโบบ็อกซ์ โดยถัดไปจากคอมโบบ็อกซ์จะมีการแสดงเลขเวอร์ชันของแผนภาพลำดับที่ถูกเลือกอยู่ในช่องของกล่องข้อความชื่อ Version

หลังจากทำการเลือกข้อมูลของแผนภาพลำดับเรียบร้อยแล้ว เครื่องมือทดสอบจะทำการดึงข้อมูลกรณีทดสอบที่มีอยู่ในฐานข้อมูลขึ้นมาแสดงในตาราง Test Case อีกทั้งผู้ใช้งานยังสามารถที่จะเรียกดูข้อมูลของรูปของแผนภาพลำดับเพื่อใช้อ้างอิงได้ โดยการกดปุ่มไฮเปอร์ลิงค์ View Diagram ในกรณีที่ผู้ใช้งานไม่ได้ทำการนำเข้รูปของแผนภาพลำดับจากขั้นตอนการสร้างตารางความสัมพันธ์

เวอร์ชันต้นฉบับหรือขั้นตอนของการเปรียบเทียบและการวิเคราะห์หาผลกระทบการเปลี่ยนแปลงนั้น
 ปุ่มไฮเปอร์ลิงก์จะไม่สามารถกดได้ ตัวอย่างของหน้าจอฐานข้อมูลแสดงได้ดังรูปที่ 4.8



รูปที่ 4.8 หน้าจอแสดงข้อมูลกรณีทดสอบ



บทที่ 5

การทดสอบเครื่องมือ

5.1 สภาพแวดล้อมที่ใช้ในการทดสอบ

5.1.1 ฮาร์ดแวร์

- เครื่องคอมพิวเตอร์ขนาดสมุดบันทึก หน่วยประมวลผลอินเทลคอร์ไอเซเว่น 2.7 กิกะเฮิร์ต
- หน่วยความจำ 16 กิกะไบต์
- ฮาร์ดดิสก์ 250 กิกะไบต์

5.1.2 ซอฟต์แวร์

- ระบบปฏิบัติการไมโครซอฟท์วินโดวส์ 8.1 โพรเฟสชัน
- ไมโครซอฟท์วิซวลสตูดิโอ 2013 โพรเฟสชันแนล
- ไมโครซอฟท์เอสคิวแอล เซิร์ฟเวอร์ เอ็กซ์เพรส 2008
- วิซวลพาราไดม์ ฟอว์ ยูเอ็มแอล คอมมูนิตี้อีดิชัน

5.2 แผนภาพลำดับที่ใช้ในการทดสอบ

แผนภาพลำดับที่ใช้ในการทดสอบความถูกต้องของเครื่องมือ นั้น ผู้วิจัยได้ทำการเลือกแผนภาพลำดับที่มีการใช้งานจริงมาทำการแก้ไขและทดสอบ ซึ่งมีทั้งหมด 2 แผนภาพลำดับ ซึ่งตัวอย่างของแผนภาพลำดับสามารถดูได้จาก ภาคผนวก ข.

5.2.1 แผนภาพลำดับที่ 1 ระบบเอทีเอ็ม

แผนภาพลำดับที่ 1 จะเป็นแผนภาพของระบบเอทีเอ็มในส่วนของการตรวจสอบรหัสบัตร ซึ่งจะถูกใช้เป็นแผนภาพเวอร์ชันต้นฉบับและมีการแก้ไขแผนภาพให้มีการรองรับการแสดงผลแบบหลายภาษาโดยทำการเพิ่มเมสเสจใหม่เข้าไปสำหรับรับค่าของภาษาที่ต้องการให้แสดงผลจากผู้ใช้งานและมีการสลับตำแหน่งของการทำงานให้มีการตรวจสอบสถานะของบัญชีผู้ใช้งานก่อนการตรวจสอบสถานะของบัตรเอทีเอ็มและมีการลบบอกของเมสเสจการแสดงความพร้อมรับบนหน้าจอ ซึ่งการแก้ไขแผนภาพตามความต้องการนี้จะถูกใช้เป็นแผนภาพเวอร์ชันใหม่

5.2.2 แผนภาพลำดับที่ 2 ระบบลงทะเบียนคนไข้

แผนภาพลำดับที่ 2 จะเป็นแผนภาพของระบบการลงทะเบียนคนไข้ในโรงพยาบาลโดยเริ่มจากการลงทะเบียนแบบธรรมดา ซึ่งจะใช้เป็นแผนภาพเวอร์ชันต้นฉบับและมีการแก้ไขให้รองรับการลงทะเบียนคนไข้แบบผู้ป่วยนอก (OPD) ซึ่งหมายถึง ผู้ป่วยที่ได้รับการรักษาที่คลินิกหรือโรงพยาบาล โดยไม่ต้องนอนพักรักษาตัวในโรงพยาบาล โดยทำการเพิ่มอ็อบเจกต์ของประเภทของคนไข้และมีการเพิ่มเมสเสจสำหรับทำการลงทะเบียนคนไข้แบบผู้ป่วยนอกและมีการลบเมสเสจออกในส่วนของการตรวจสอบข้อมูลรายละเอียดของคนไข้บางส่วน

5.3 ขั้นตอนการทดสอบเครื่องมือและผลการทดสอบ

ขั้นตอนการทดสอบเครื่องมือเป็นขั้นตอนที่ใช้ในการตรวจสอบความถูกต้องของการทำงานของเครื่องมือที่ได้ถูกออกแบบและพัฒนาขึ้นมาตามหลักทฤษฎีในงานวิจัย โดยแบ่งการทดสอบออกเป็น 4 ส่วน ดังต่อไปนี้

5.3.1 การทดสอบการสร้างตารางความสัมพันธ์

ขั้นตอนการทดสอบการสร้างตารางความสัมพันธ์ทำได้โดยการนำเข้าไฟล์ข้อมูลของแผนภาพลำดับที่อยู่ในรูปแบบเอกซ์เอ็มแอลเข้ามาในเครื่องมือทดสอบผ่านทางรายการเลือก Import Diagram and Test Case โดยที่เครื่องมือทดสอบจะต้องสามารถทำการอ่านข้อมูลของแผนภาพลำดับและสร้างตารางความสัมพันธ์ตามจำนวนของเมสเสจที่มีอยู่ในแผนภาพลำดับได้อย่างถูกต้อง

โดยขั้นตอนการทดสอบจะทำการแก้ไขแผนภาพลำดับที่ 1 และ แผนภาพลำดับที่ 2 ให้มีการเปลี่ยนแปลงครบทุกเงื่อนไขที่เครื่องมือทดสอบสามารถทำการตรวจสอบได้ตามขอบเขตของงานวิจัย โดยการแก้ไขแผนภาพลำดับเวอร์ชันใหม่สามารถแสดงได้ตามประเภทของการเปลี่ยนแปลง ดังตารางที่ 5.1

ตารางที่ 5.1 ตารางแสดงการเปลี่ยนแปลงของแผนภาพลำดับเวอร์ชันใหม่

การเปลี่ยนแปลง	แผนภาพลำดับที่ 1	แผนภาพลำดับที่ 2
การแทรกเมสเสจ	<ul style="list-style-type: none"> ● selectLanguage() 	<ul style="list-style-type: none"> ● HasVisitOrRelatedObjectsChanged() ● SetPatientManager(PatientControlsManager) ● SaveOPD(IPatientVisitTx, ref PatientVisit) ● OPDVisitUpdatableObjects(PatientVisit) ● UpdateOutPatientVisit(IPatientVisitTx) ● Populate(Guid, bool) ● LoadProfile(Guid, bool)
การลบเมสเสจออก	<ul style="list-style-type: none"> ● displayHello() 	<ul style="list-style-type: none"> ● RemoveConsentReviewIfHidden() ● InsertPatient(IPatientTx, ref Patient, ref PatientReferralCollection, ref PolicySubscriptionCollection, ref DocumentCollection)
การแทนที่ของเมสเสจ	<ul style="list-style-type: none"> ● eject() ● retain() 	-
การแทรกอ็อบเจกต์	-	<ul style="list-style-type: none"> ● VisitManager ● OPDVisitUpdatableObjects ● PatientVisitTx
การลบอ็อบเจกต์	-	<ul style="list-style-type: none"> ● PatientTx
การเปลี่ยนเมทอดซิกเนเจอร์	-	<ul style="list-style-type: none"> ● ValidateData(PatientVisit)

สรุปผลการทดสอบของการสร้างตารางความสัมพันธ์จากแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่ของแผนภาพลำดับที่ 2 แผนภาพ แสดงสรุปได้ดังตารางที่ 5.2

ตารางที่ 5.2 ตารางสรุปผลการทดสอบการสร้างตารางความสัมพันธ์จากแผนภาพลำดับ

การทดสอบ	แผนภาพลำดับที่ 1		แผนภาพลำดับที่ 2	
	คาดหวัง	ผลลัพธ์	คาดหวัง	ผลลัพธ์
จำนวนระเบียบของข้อมูลจากแผนภาพลำดับเวอร์ชันต้นฉบับ	10 ระเบียบ	10 ระเบียบ	16 ระเบียบ	16 ระเบียบ
จำนวนระเบียบของข้อมูลจากแผนภาพลำดับเวอร์ชันใหม่	9 ระเบียบ	9 ระเบียบ	21 ระเบียบ	21 ระเบียบ

5.3.2 การเปรียบเทียบการเปลี่ยนแปลงระหว่างแผนภาพลำดับ

ขั้นตอนการทดสอบการเปรียบเทียบการเปลี่ยนแปลงระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับกับเวอร์ชันใหม่นั้นทำได้โดยการเปรียบเทียบของข้อมูลจากตารางความสัมพันธ์เวอร์ชันต้นฉบับและเวอร์ชันใหม่โดยทำการทดสอบได้จากรายการเลือก Comparator and Impact Analysis และทำการเลือกแผนภาพลำดับเวอร์ชันต้นฉบับที่ต้องการจะทำการเปรียบเทียบจากนั้นทำการนำเข้าไฟล์ข้อมูลของแผนภาพลำดับเวอร์ชันใหม่ที่ได้ทำการออกแบบใหม่และนำออกมาอยู่ในรูปแบบไฟล์เอกซ์เอ็มแอลแล้วเข้ามาในเครื่องมือทดสอบ

การทดสอบแผนภาพของระบบที่ 1 จะทำการทดสอบโดยการแก้ไขแผนภาพลำดับเวอร์ชันใหม่ให้มีการแทรก การลบออกและการแทนที่ของเมสเสจ

การทดสอบแผนภาพของระบบที่ 2 จะทำการทดสอบโดยการแก้ไขแผนภาพลำดับเวอร์ชันใหม่ให้มีการเพิ่มอ็อบเจกต์ การแทรกและการลบออกของเมสเสจ

โดยผลลัพธ์ที่คาดหวังจะเป็นรหัสลำดับการทดสอบ (GID) ที่ใช้แสดงถึงข้อมูลของเมสเสจในตารางความสัมพันธ์และแบ่งรหัสลำดับการทดสอบที่มีการเปลี่ยนแปลงตามประเภทของการเปรียบเทียบตามตารางที่ 3.1 โดยผลลัพธ์ของการทดสอบการเปรียบเทียบที่เกิดจากการเปลี่ยนแปลงตามตารางที่ 5.1 ระหว่างแผนภาพลำดับที่ 1 และ แผนภาพลำดับที่ 2 นั้นสามารถแสดงได้ ดังตารางที่ 5.3 และตารางที่ 5.4

ตารางที่ 5.3 ตารางแสดงผลการเปรียบเทียบของแผนภาพลำดับที่ 1

รหัส GID1	รหัส GID2	ประเภทการเปรียบเทียบ
1	1	ตรงกัน
2	2	ตรงกัน
3	3	ตรงกัน
-	11	การแทรก
4	5	การแทนที่
5	4	การแทนที่
6	6	ตรงกัน
7	7	ตรงกัน
8	8	ตรงกัน
9	9	ตรงกัน
10	-	การลบออก

ตารางที่ 5.4 ตารางแสดงผลการเปรียบเทียบของแผนภาพลำดับที่ 2

รหัส GID1	รหัส GID2	ประเภทการเปรียบเทียบ
1	1	ตรงกัน
2	2	ตรงกัน
3	3	ตรงกัน
-	4	ตรงกัน
4	5	ตรงกัน
5	4	ตรงกัน
6	6	ตรงกัน
7	7	ตรงกัน
8	8	ตรงกัน
9	9	ตรงกัน
10	10	ตรงกัน
11	11	ตรงกัน
12	12	ตรงกัน
13	13	ตรงกัน
14	-	การลบบอก
15	-	การลบบอก
16	-	การลบบอก
-	17	การแทรก
-	18	การแทรก
-	19	การแทรก
-	20	การแทรก
-	21	การแทรก
-	22	การแทรก
-	23	การแทรก
-	24	การแทรก
-	25	การแทรก

หลังจากขั้นตอนการเปรียบเทียบจะทำให้ได้รหัสลำดับการทดสอบที่มีการเปลี่ยนแปลงซึ่งจะ
ถูกนำไปใช้สำหรับการหาผลกระทบการเปลี่ยนแปลงในขั้นตอนต่อไป ซึ่งรหัสลำดับการทดสอบที่มี
การเปลี่ยนแปลงจากแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่ สามารถสรุปและแสดงได้ดัง
ตารางที่ 5.5

ตารางที่ 5.5 ตารางสรุปรหัสลำดับการทดสอบที่มีการเปลี่ยนแปลง

การทดสอบ	แผนภาพลำดับที่ 1		แผนภาพลำดับที่ 2	
	รหัส GID 1	รหัส GID 2	รหัส GID 1	รหัส GID 2
ผลลัพธ์ของการเปรียบเทียบของแผนภาพลำดับ	<ul style="list-style-type: none"> ● 10 	<ul style="list-style-type: none"> ● 11 ● 5 ● 4 	<ul style="list-style-type: none"> ● 14 ● 15 ● 16 	<ul style="list-style-type: none"> ● 17 ● 18 ● 19 ● 20 ● 21 ● 22 ● 23 ● 24

สรุปผลการทดสอบของการเปรียบเทียบการเปลี่ยนแปลงระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่ของแผนภาพลำดับทั้ง 2 แผนภาพ แสดงสรุปได้ดังตารางที่ 5.6

ตารางที่ 5.6 ตารางสรุปผลการทดสอบของการเปรียบเทียบการเปลี่ยนแปลง

การทดสอบ	แผนภาพลำดับที่ 1			แผนภาพลำดับที่ 2		
	จำนวนการแก้ไข	คาดหวัง	ผลลัพธ์	จำนวนการแก้ไข	คาดหวัง	ผลลัพธ์
ตรงกัน	-	7 รหัส	7 รหัส	-	13 รหัส	13 รหัส
การแทรก	1 รหัส	1 รหัส	1 รหัส	8 รหัส	8 รหัส	8 รหัส
การลบออก	1 รหัส	1 รหัส	1 รหัส	3 รหัส	3 รหัส	3 รหัส
การแทนที่	2 รหัส	2 รหัส	2 รหัส	0 รหัส	0 รหัส	0 รหัส

5.3.3 การหาผลกระทบการเปลี่ยนแปลง

ขั้นตอนการทดสอบการหาผลกระทบการเปลี่ยนแปลงระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับกับเวอร์ชันใหม่นั้นจะทำได้หลังจากขั้นตอนการเปรียบเทียบการเปลี่ยนแปลงระหว่างแผนภาพลำดับเพราะในขั้นตอนการผลกระทบการเปลี่ยนแปลงนั้นจำเป็นต้องใช้ผลลัพธ์ของรหัสลำดับการทดสอบจากการเปรียบเทียบการเปลี่ยนแปลง

หลังจากที่ได้รหัสลำดับการทดสอบมาจากขั้นตอนการเปรียบเทียบการเปลี่ยนแปลงแล้วเครื่องมือทดสอบจะนำรหัสลำดับการทดสอบที่ได้มาไปทำการหาผลกระทบทางตรงในตารางความสัมพันธ์เวอร์ชันต้นฉบับสำหรับการลบออกและในตารางความสัมพันธ์เวอร์ชันใหม่สำหรับการแทรกและแทนที่

โดยผลลัพธ์ที่คาดหวังจากขั้นตอนการหาผลกระทบการเปลี่ยนแปลงก็คือ รหัสลำดับการทดสอบที่มีการเปลี่ยนแปลงจากการแทรก การลบออกและการแทนที่ รวมถึงรหัสลำดับการทดสอบที่อยู่ในตารางความสัมพันธ์และที่รับผลกระทบทางตรงจากการเปลี่ยนแปลง

เครื่องมือวิจัยนำผลลัพธ์ของการเปรียบเทียบจากตารางที่ 5.5 มาทำการหาผลกระทบของการเปลี่ยนแปลงจากแผนภาพลำดับที่ 1 และแผนภาพลำดับที่ 2 ซึ่งผลกระทบจากการเปลี่ยนแปลงทางตรงที่ได้ทั้งหมด สามารถแสดงได้ดังตารางที่ 5.7

ตารางที่ 5.7 ตารางแสดงผลลัพธ์การหาผลกระทบการเปลี่ยนแปลงจากแผนภาพลำดับ

การทดสอบ	แผนภาพลำดับที่ 1				แผนภาพลำดับที่ 2			
	GID 1	ผลกระทบ	GID 2	ผลกระทบ	GID 1	ผลกระทบ	GID 2	ผลกระทบ
การแทรก	-	-	● 11	● 3 ● 5	-	-	● 17 ● 18 ● 19 ● 20 ● 21 ● 22 ● 23 ● 24	● 13 ● 17 ● 18 ● 19 ● 20 ● 21 ● 22 ● 23 ● 24
การลบออก	● 10	● 9	-	-	● 14 ● 15 ● 16	● 13	-	-
การแทนที่	-	-	● 5 ● 4	● 11 ● 5 ● 4 ● 6	-	-	-	-

หลังจากทำการหาผลกระทบการเปลี่ยนแปลงที่เกิดขึ้นจากการเปรียบเทียบระหว่างแผนภาพลำดับที่ 1 และแผนภาพลำดับที่ 2 ทำให้สามารถสรุปผลกระทบการเปลี่ยนแปลงระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับกับเวอร์ชันใหม่ ได้ดังตารางที่ 5.8

ตารางที่ 5.8 ตารางสรุปรหัสที่ได้รับผลกระทบจากการเปลี่ยนแปลงจากแผนภาพลำดับ

การทดสอบ	แผนภาพลำดับที่ 1	แผนภาพลำดับที่ 2
การแทรก	3, 11, 5	13, 17, 18, 19, 20, 21, 22, 23, 24
การลบออก	9	13
การแทนที่	11, 5, 4, 6	-

สรุปผลการทดสอบของการหาผลกระทบจากการเปลี่ยนแปลงจากแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่ของแผนภาพลำดับทั้ง 2 แผนภาพ แสดงสรุปได้ดังตารางที่ 5.9

ตารางที่ 5.9 ตารางสรุปผลการทดสอบการหาผลกระทบจากการเปลี่ยนแปลงจากแผนภาพลำดับ

การทดสอบ	แผนภาพลำดับที่ 1			แผนภาพลำดับที่ 2		
	จำนวนรหัสลำดับจากการเปรียบเทียบ	คาดหวัง	ผลลัพธ์	จำนวนรหัสลำดับจากการเปรียบเทียบ	คาดหวัง	ผลลัพธ์
การแทรก	1 รหัส	3 รหัส	3 รหัส	8 รหัส	9 รหัส	9 รหัส
การลบออก	1 รหัส	1 รหัส	1 รหัส	3 รหัส	1 รหัส	1 รหัส
การแทนที่	2 รหัส	4 รหัส	4 รหัส	0 รหัส	0 รหัส	0 รหัส

5.3.4 การสร้างกรณีทดสอบและการเลือกชุดทดสอบเชิงถดถอย

ขั้นตอนการทดสอบการสร้างกรณีทดสอบและการเลือกชุดทดสอบเชิงถดถอยนั้นสามารถทำได้หลังจากขั้นตอนการหาผลกระทบจากการเปลี่ยนแปลงเพราะในขั้นตอนนี้จำเป็นต้องใช้ข้อมูลของรหัสลำดับการทดสอบที่ได้รับผลกระทบจากการแก้ไขแผนภาพลำดับของเวอร์ชันใหม่

หลังจากได้ข้อมูลของรหัสลำดับการทดสอบที่ได้รับผลกระทบจากการเปลี่ยนแปลงทั้งหมดมาแล้วนั้น เครื่องมือทดสอบจะทำการนำรหัสลำดับการทดสอบเข้าไปค้นหาในฐานข้อมูลกรณีทดสอบของแผนภาพลำดับที่ 1 และ 2

โดยผลลัพธ์ที่คาดหวังในขั้นตอนนี้มีรายละเอียดดังต่อไปนี้

- กรณีทดสอบจะถูกคัดเลือกเข้าไปในชุดทดสอบเชิงถดถอย ถ้าข้อมูลของกรณีทดสอบนั้นมีรหัสลำดับการทดสอบที่ได้รับผลกระทบอยู่ด้วย
- กรณีทดสอบจะถูกสร้างขึ้นใหม่ ถ้ารหัสลำดับการทดสอบที่ได้รับผลกระทบไม่เคยมีอยู่ในฐานข้อมูลกรณีทดสอบมาก่อนหรือรหัสลำดับการทดสอบมีการเปลี่ยนแปลงลำดับของข้อมูล

- กรณีทดสอบจะถูกถอดออกจากฐานข้อมูล ในกรณีที่กรณีทดสอบที่มีอยู่ในฐานข้อมูลนั้นมีข้อมูลของรหัสลำดับการทดสอบที่เป็นผลกระทบประเภทการลบบอกอยู่
- กรณีทดสอบที่ยังสามารถใช้งานได้ทั้งหมดในฐานข้อมูลสำหรับแผนภาพลำดับ

จากขั้นตอนการหาผลกระทบของการเปลี่ยนแปลงระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับ และเวอร์ชันใหม่จากตารางที่ 5.8 สามารถนำผลลัพธ์ในตารางไปใช้ในการคัดเลือกกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอยของแผนภาพลำดับที่ 1 และแผนภาพลำดับที่ 2 ซึ่งสามารถแสดงรายละเอียดได้ดังตารางที่ 5.10 และตารางที่ 5.11

ตารางที่ 5.10 ตารางการคัดเลือกชุดทดสอบเชิงถดถอยจากแผนภาพลำดับที่ 1

รหัสกรณีทดสอบ	รหัสลำดับกรณีทดสอบ	รหัสที่ได้รับผลกระทบ	ประเภทกรณีทดสอบ	ชุดกรณีทดสอบเชิงถดถอย	กรณีทดสอบที่ยังใช้งานได้
ATM001	1, 2	-	กรณีทดสอบที่ยังใช้ได้	×	✓
ATM002	1, 3, 4	3, 4	กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว	×	×
ATM003	1, 3, 5	3, 5	กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว	×	×
ATM004	1, 3, 6, 7, 8	3, 6	กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว	×	×
ATM005	1, 3, 6, 8, 9	3, 6, 9	กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว	×	×
ATM006	1, 3, 6, 7, 8, 10	3, 6	กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว	×	×
ATM007	1, 3, 11, 5	3, 11, 5	กรณีทดสอบที่ถูกสร้างใหม่	✓	✓
ATM008	1, 3, 11, 4	3, 11, 4	กรณีทดสอบที่ถูกสร้างใหม่	✓	✓
ATM009	1, 3, 11, 6, 7, 8	3, 11, 6	กรณีทดสอบที่ถูกสร้างใหม่	✓	✓
ATM010	1, 3, 11, 6, 7, 8, 9	3, 11, 6, 9	กรณีทดสอบที่ถูกสร้างใหม่	✓	✓

ตารางที่ 5.11 ตารางการคัดเลือกชุดทดสอบเชิงถดถอยจากแผนภาพลำดับที่ 2

รหัสกรณีทดสอบ	รหัสลำดับกรณีทดสอบ	รหัสที่ได้รับผลกระทบ	ประเภทกรณีทดสอบ	ชุดกรณีทดสอบเชิงถดถอย	กรณีทดสอบที่ยังใช้งานได้
REG001	1, 2, 3, 4, 5	-	กรณีทดสอบที่ยังใช้ได้	×	✓
REG002	1, 2, 3, 4, 6, 7, 8, 9, 10, 11	-	กรณีทดสอบที่ยังใช้ได้	×	✓
REG003	1, 2, 3, 4, 6, 7, 8, 9, 10, 12, 13	13	กรณีทดสอบที่ยังใช้ได้	✓	✓
REG004	1, 2, 3, 4, 6, 7, 8, 9, 10, 12, 14, 15, 16	14, 15, 16	กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว	×	×
REG005	1, 2, 3, 4, 6, 7, 8, 9, 10, 12, 17, 18, 19, 20, 21, 22, 23	17, 18, 19, 20, 21, 22, 23	กรณีทดสอบที่ถูกสร้างใหม่	✓	✓
REG006	1, 2, 3, 4, 6, 7, 8, 9, 10, 12, 17, 18, 19, 20, 21, 22, 24	17, 18, 19, 20, 21, 22, 24	กรณีทดสอบที่ถูกสร้างใหม่	✓	✓

ผลการทดสอบของการสร้างกรณีทดสอบและการเลือกชุดทดสอบเชิงถดถอยจากแผนภาพลำดับ แผนภาพลำดับที่ 2 แผนภาพที่นำมาทดสอบ แสดงสรุปได้ดังตารางที่ 5.12

ตารางที่ 5.12 ตารางสรุปผลการสร้างกรณีทดสอบและการเลือกชุดทดสอบเชิงถดถอย

การทดสอบการจัดการกรณีทดสอบ	แผนภาพลำดับที่ 1		แผนภาพลำดับที่ 2	
	คาดหวัง	ผลลัพธ์	คาดหวัง	ผลลัพธ์
กรณีทดสอบที่ถูกสร้างใหม่	4 กรณี	4 กรณี	2 กรณี	2 กรณี
กรณีทดสอบที่ไม่สามารถใช้งานได้แล้ว	5 กรณี	5 กรณี	1 กรณี	1 กรณี
กรณีทดสอบที่นำกลับมาทดสอบใหม่	4 กรณี	4 กรณี	3 กรณี	3 กรณี
กรณีทดสอบที่ยังใช้ได้ทั้งหมด	5 กรณี	5 กรณี	5 กรณี	5 กรณี

จากผลการทดสอบ เครื่องมือทดสอบสามารถช่วยลดกรณีทดสอบของแผนภาพลำดับที่ 1 ได้จำนวน 1 กรณีทดสอบจากทั้งหมด 5 กรณีทดสอบหรือคิดเป็น 20 เปอร์เซ็นต์ และช่วยลดกรณีทดสอบของแผนภาพลำดับที่ 2 ได้จำนวน 2 กรณีทดสอบจากทั้งหมด 5 กรณีทดสอบหรือคิดเป็น 40 เปอร์เซ็นต์

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

6.1 สรุปผลการวิจัย

จากการศึกษา ออกแบบ พัฒนาและทดสอบเครื่องมือสำหรับการเลือกกรณีทดสอบเชิงถดถอยแบบปลอดภัยจากแผนภาพลำดับ เพื่อตอบสนองต่อวัตถุประสงค์ของงานวิจัยตามขอบเขตและสืบเนื่องไปถึงการทดสอบเครื่องมือที่พัฒนาขึ้นนั้น สามารถสรุปผลการวิจัยได้ว่าชุดข้อมูลกรณีทดสอบเชิงถดถอยแบบปลอดภัยที่ได้จากเครื่องมือทดสอบที่พัฒนาขึ้นนั้นสามารถช่วยให้ผู้ใช้งานสามารถทำการเลือกกรณีทดสอบเชิงถดถอยแบบปลอดภัยในกรณีที่มีการเปลี่ยนแปลงหน้าที่การทำงานของซอฟต์แวร์ตั้งแต่ระดับของการออกแบบได้อย่างมีประสิทธิภาพและแม่นยำโดยใช้ข้อมูลจากแผนภาพลำดับเป็นข้อมูลในการช่วยคัดเลือกเฉพาะกรณีทดสอบที่ได้รับผลกระทบเท่านั้น เพื่อหลีกเลี่ยงการทดสอบซอฟต์แวร์ใหม่ทั้งหมดในกรณีที่มีการเปลี่ยนแปลงแก้ไขต่างๆ ทำให้ช่วยลดระยะเวลา ทรัพยากรและค่าใช้จ่ายในการคัดเลือกกรณีทดสอบเชิงถดถอยได้อย่างมีประสิทธิภาพ

6.2 ข้อจำกัดของเครื่องมือทดสอบ

- 1) แผนภาพลำดับจะต้องถูกออกแบบและนำออกจากโปรแกรมวิซวลพาราไดม์ในรูปแบบของไฟล์เอกซ์เอ็มแอลและทำงานบนระบบปฏิบัติการวินโดวส์เท่านั้น
- 2) เครื่องมือไม่สนับสนุนการสร้างตารางความสัมพันธ์และการสร้างกรณีทดสอบจากแผนภาพลำดับแบบมีเงื่อนไข (if-else condition)
- 3) รหัสลำดับการทดสอบ (GID) ของแผนภาพลำดับเวอร์ชันใหม่จะต้องถูกสร้างจากเครื่องมือทดสอบเท่านั้นและรหัสลำดับการทดสอบจะไม่ถูกใช้ซ้ำ
- 4) เครื่องมือไม่สนับสนุนการปรับปรุงรหัสลำดับการทดสอบที่ถูกสร้างขึ้นมาจากตารางความสัมพันธ์กลับไปแผนภาพลำดับที่อยู่ในโปรแกรมวิซวลพาราไดม์ได้แบบอัตโนมัติ ผู้ใช้งานจะต้องทำการนำรหัสลำดับการทดสอบที่ถูกสร้างจากเครื่องมือทดสอบไปปรับปรุงที่แผนภาพลำดับด้วยตัวเอง
- 5) เครื่องมือไม่สนับสนุนการวิเคราะห์การวนซ้ำที่มีการใช้สัญลักษณ์ดอกจัน (*) กำกับไว้ที่เมสเสจ
- 6) เครื่องมือไม่สนับสนุนการดูประวัติของการเปลี่ยนแปลงของแผนภาพลำดับและกรณีทดสอบในแต่ละเวอร์ชันย้อนหลังได้

6.3 ข้อเสนอแนะและแนวทางการวิจัยในอนาคต

- 1) พัฒนาเครื่องมือให้สามารถปรับปรุงรหัสลำดับกลับไปทีแผนภาพลำดับได้อย่างอัตโนมัติ
- 2) พัฒนาเครื่องมือให้สามารถรองรับการทำงานแบบมีเงื่อนไขได้
- 3) พัฒนาเครื่องมือให้สนับสนุนการนำเข้าข้อมูลของแผนภาพลำดับจากซอฟต์แวร์อื่นที่ใช้สำหรับการออกแบบแผนภาพลำดับได้
- 4) พัฒนาเครื่องมือให้สามารถแสดงไฮไลต์บนรูปของแผนภาพลำดับสำหรับกรณีทดสอบ
- 5) พัฒนาเครื่องมือให้รองรับการออกแบบแผนภาพลำดับและจัดเก็บอยู่ในเครื่องมือได้
- 6) พัฒนาเครื่องมือให้สามารถเรียกดูรายงานของกรณีทดสอบเชิงถดถอยในแต่ละเวอร์ชันที่มีการเปลี่ยนแปลงได้



รายการอ้างอิง

1. Harrold, M.J., *Reduce, reuse, recycle, recover: Techniques for improved regression testing*, in *Software Maintenance, ICSM 2009. IEEE International Conference on*. 2009, ICSM.
2. Liang You, Y.L., *A Genetic Algorithm For The Time-Aware Regression Testing Reduction Problem*, in *Natural Computation (ICNC), 2012 Eighth International International Conference on*. 2012, ICNC. p. 596-599.
3. Anoj Kumar, S.T., K.K. Mishra and A.K.Misra,, *Generation of efficient test data using path selection strategy with elitist GA in regression testing*, in *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*. 2010, Computer Science and Information Technology (ICCSIT). p. 389-393.
4. Orest Pilskalns and Gunay Uyan, A.A., *Regression Testing UML Designs*, in *Software Maintenance, 2006. ICSM '06. 22nd IEEE International Conference on*. 2006, ICSM. p. 254-264.
5. Gregg Rothermel, M.J.H., *Analyzing Regression Test Selection Techniques*. *Software Engineering, IEEE Transactions on*, 1996. 22(8): p. 529-551.
6. E. Lank, J.T., S. Chen, and D. Blostein,. *On-line recognition of UML diagrams*. in *Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on*. 2001. IEEE.
7. Philip Samuel, A.T.J., *Test Sequence Generation from UML Sequence Diagrams*, in *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2008. SNPD '08. Ninth ACIS International Conference on*. 2008, SNPD. p. 879-887.
8. Lunjin Lu and Dae-Kyoo Kim, *Required Behavior of Sequence Diagrams: Semantics and Refinement*, in *Engineering of Complex Computer Systems (ICECCS), 2011 16th IEEE International Conference on*. 2011, IEEE. p. 127-136.

9. Christopher Ackermann and Mikael Lindvall, *Understanding Change Requests to Predict Software Impact*, in *Software Engineering Workshop*, 2006. SEW '06. 30th Annual IEEE/NASA. 2006, IEEE. p. 66-75.
10. H. M. Kao. *Regression Testing on Object-Oriented Programs*. in *Proceedings 10th International Symposium on Software Reliability Engineering (Cat. No.PR00433)*. 1999.
11. T. Koju and S. Takada. *Regression Test Selection based on Intermediate Code for Virtual Machines*. in *Software Maintenance, 2003. ICSM 2003. Proceedings. International Conference on 2003*. IEEE.
12. W. El-hamid and S. El-etriby, *Regression Test Selection Technique for Multi-Programming Language*, in *Informatics and Systems (INFOS), 2010 The 7th International Conference on*. 2010, IEEE. p. 1-5.
13. L. C. Briand, Y.L.a.G.S. *Automating Impact Analysis and Regression Test Selection Base on UML Designs*. in *Software Maintenance, 2002. Proceedings. International Conference on*. 2002. Montreal (Canada),: IEEE Computer Society.
14. Monalisa Sarma, R.M., *Automatic Test Case Generation from UML Models*, in *Information Technology, (ICIT 2007)*. 10th International Conference on 2007, IEEE. p. 196-201.
15. G.Rothermel and M.J. Harrold, *Empirical studies of a safe regression test selection technique*. *Software Engineering, IEEE Transactions on*, 1998. 24(6): p. 401-419.
16. G. Rothermel and M.J. Harrold. *A Safe, Efficient Algorithm for Regression Test Selection*. in *Software Maintenance ,1993. CSM-93, Proceedings., Conference on 1993*. IEEE.
17. G.Duggal, B.S., *Understanding Regression Testing Techniques*, in *COIT*, 2008, RMIT Institute of Engineering and Technology. 2008, COIT: Committee on Information Technology, San Francisco City, USA.

18. Lyras, D.P.S., K.N. ; Fakotakis, N.D.,, *Using the Levenshtein Edit Distance for Automatic Lemmatization: A Case Study for Modern Greek and English*, in *Tools with Artificial Intelligence*, 2007. ICTAI 2007. 19th IEEE International Conference on. 2007, IEEE. p. 428-435.





ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาคผนวก ก
รายละเอียดยูสเคส

ตารางที่ ก - 1 รายละเอียดยูสเคส Load XML

รหัสยูสเคส:	1
ชื่อยูสเคส:	Load XML
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	นำเข้าข้อมูลแผนภาพลำดับจากไฟล์เอกซ์เอ็มแอล
ความสัมพันธ์	-
เงื่อนไขก่อนหน้า:	นำออกข้อมูลแผนภาพลำดับจากโปรแกรมวิซวลพาราไดม์
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1) เครื่องมือทดสอบแสดงหน้าจอการนำเข้าไฟล์เอกซ์เอ็มแอล 2) ผู้ใช้งานทำการเลือกไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับที่ต้องการจะใช้งาน
เงื่อนไขภายหลัง:	ไฟล์เอกซ์เอ็มแอลถูกนำเข้าเครื่องมือทดสอบ

ตารางที่ ก - 2 รายละเอียดยูสเคส Generate Sequence Dependency Table

รหัสยูสเคส:	2
ชื่อยูสเคส:	Generate Sequence Dependency Table
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	สร้างตารางความสัมพันธ์แผนภาพลำดับเวอร์ชันต้นฉบับ
ความสัมพันธ์	Include: Load XML
เงื่อนไขก่อนหน้า:	นำเข้าไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับ
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Load XML เพื่อนำข้อมูลแผนภาพลำดับเข้ามาในเครื่องมือทดสอบ 2. เครื่องมือทดสอบทำการตรวจสอบโครงสร้างของไฟล์ 3. เครื่องมือทดสอบทำการสร้างตารางความสัมพันธ์ โดยการอ่านข้อมูลจากไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับเวอร์ชันต้นฉบับ 4. แสดงข้อมูลตารางความสัมพันธ์บนหน้าจอ
เงื่อนไขภายหลัง:	ตารางความสัมพันธ์จากแผนภาพลำดับเวอร์ชันต้นฉบับถูกสร้างขึ้น

ตารางที่ ก - 3 รายละเอียดยูสเคส Save Sequence Dependency Table

รหัสยูสเคส:	3
ชื่อยูสเคส:	Save Sequence Dependency Table
ผู้กระทำหลัก (Actor):	-
รายละเอียดยูสเคส:	บันทึกข้อมูลตารางความสัมพันธ์แผนภาพลำดับเวอร์ชันต้นฉบับ
ความสัมพันธ์	Include: Generate Sequence Dependency Table
เงื่อนไขก่อนหน้า:	ตารางความสัมพันธ์ถูกสร้างในรูปแบบตาราง
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Generate Sequence Dependency Table 2. เครื่องมือทดสอบแสดงข้อมูลตารางความสัมพันธ์เวอร์ชันต้นฉบับ 3. ผู้ใช้งานทำการกดปุ่มบันทึกข้อมูล 4. เครื่องมือทดสอบบันทึกข้อมูล
เงื่อนไขภายหลัง:	ข้อมูลตารางความสัมพันธ์เวอร์ชันต้นฉบับถูกจัดเก็บลงฐานข้อมูล

ตารางที่ ก - 4 รายละเอียดยูสเคส Load Original Sequence Dependency Table

รหัสยูสเคส:	4
ชื่อยูสเคส:	Load Original Sequence Dependency Table
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	สร้างตารางความสัมพันธ์แผนภาพลำดับเวอร์ชันต้นฉบับ
ความสัมพันธ์	-
เงื่อนไขก่อนหน้า:	ข้อมูลของตารางความสัมพันธ์ของแผนภาพลำดับเวอร์ชันต้นฉบับ
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. ผู้ใช้งานเลือกแผนภาพลำดับที่ต้องการ 2. เครื่องมือทดสอบแสดงข้อมูลตารางความสัมพันธ์เวอร์ชันต้นฉบับบนหน้าจอ
เงื่อนไขภายหลัง:	แสดงตารางความสัมพันธ์จากแผนภาพลำดับเวอร์ชันต้นฉบับ

ตารางที่ ก - 5 รายละเอียดยูสเคส Load New Sequence Dependency Table

รหัสยูสเคส:	5
ชื่อยูสเคส:	Load New Sequence Dependency Table
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	สร้างตารางความสัมพันธ์แผนภาพลำดับเวอร์ชันต้นฉบับ
ความสัมพันธ์	-
เงื่อนไขก่อนหน้า:	นำเข้าไปไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับเวอร์ชันใหม่
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Load XML เพื่อนำข้อมูลแผนภาพลำดับเวอร์ชันใหม่เข้ามาในเครื่องมือทดสอบ 2. เรียกใช้งานยูสเคส Generate Sequence Dependency Table สำหรับสร้างตารางความสัมพันธ์ของแผนภาพลำดับเวอร์ชันใหม่ 3. แสดงข้อมูลตารางความสัมพันธ์ของแผนภาพลำดับเวอร์ชันใหม่บนหน้าจอ
เงื่อนไขภายหลัง:	ตารางความสัมพันธ์จากแผนภาพลำดับเวอร์ชันใหม่ถูกสร้างขึ้น

ตารางที่ ก - 6 รายละเอียดยูสเคส Get GID

รหัสยูสเคส:	6
ชื่อยูสเคส:	Get GID
ผู้กระทำหลัก (Actor):	-
รายละเอียดยูสเคส:	ดึงข้อมูลรหัส GID จากแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่
ความสัมพันธ์	Include: Load Original Sequence Dependency Table, Load New Sequence Dependency Table
เงื่อนไขก่อนหน้า:	ข้อมูลตารางความสัมพันธ์แผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Load Original Sequence Dependency Table และ Load New Sequence Dependency Table 2. เครื่องมือทดสอบดึงข้อมูลรหัส GID จากตารางความสัมพันธ์
เงื่อนไขภายหลัง:	รายการลำดับการแก้ไขถูกสร้างขึ้น

ตารางที่ ก - 7 รายละเอียดยูสเคส Compare List of Modification

รหัสยูสเคส:	7
ชื่อยูสเคส:	Compare List of Modification
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	ทำการเปรียบเทียบและวิเคราะห์หาผลกระทบการเปลี่ยนแปลงจากแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่
ความสัมพันธ์	Include: Get GID
เงื่อนไขก่อนหน้า:	จัดเตรียมข้อมูลรายการลำดับการแก้ไข
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Get GID เพื่อดึงข้อมูลรายการลำดับการแก้ไข 2. ทำการเปรียบเทียบข้อมูลจากรายการลำดับการแก้ไขระหว่างรายการลำดับการแก้ไขจากแผนภาพลำดับเวอร์ชันต้นฉบับกับเวอร์ชันใหม่
เงื่อนไขภายหลัง:	-

ตารางที่ ก - 8 รายละเอียดยูสเคส Generate Comparison Result

รหัสยูสเคส:	8
ชื่อยูสเคส:	Generate Comparison Result
ผู้กระทำหลัก (Actor):	-
รายละเอียดยูสเคส:	แสดงผลลัพธ์ของการเปรียบเทียบและระบุประเภทของการเปลี่ยนแปลงของการแก้ไขของแผนภาพลำดับ
ความสัมพันธ์	Include: Compare List of Modification
เงื่อนไขก่อนหน้า:	ข้อมูลการเปรียบเทียบของรายการลำดับการแก้ไข
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Compare List of Modification เพื่อนำเข้าข้อมูลผลลัพธ์ของการเปรียบเทียบ 2. เครื่องมือทดสอบทำการอ่านข้อมูลของผลลัพธ์ของการเปรียบเทียบ 3. เครื่องมือทดสอบทำการระบุประเภทของการเปรียบเทียบ 4. แสดงข้อมูลผลลัพธ์ของการเปรียบเทียบบนหน้าจอ
เงื่อนไขภายหลัง:	ข้อมูลผลลัพธ์ของการเปรียบเทียบที่เกิดจากการเปลี่ยนแปลงระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับและเวอร์ชันใหม่ถูกสร้างขึ้น

ตารางที่ ก - 9 รายละเอียดยูสเคส Load Comparison Result

รหัสยูสเคส:	9
ชื่อยูสเคส:	Load Comparison Result
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	นำเข้าข้อมูลผลลัพธ์ของการเปรียบเทียบสำหรับใช้ในการค้นหาผลกระทบจากการเปลี่ยนแปลงของแผนภาพลำดับ
ความสัมพันธ์	-
เงื่อนไขก่อนหน้า:	ข้อมูลผลลัพธ์การเปรียบเทียบระหว่างแผนภาพลำดับเวอร์ชันต้นฉบับกับเวอร์ชันใหม่
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. นำเข้าข้อมูลผลลัพธ์ของการเปรียบเทียบ 2. จัดเตรียมรายการของรหัสทดสอบที่มีประเภทของการเปรียบเทียบเป็นการแทรก การลบออกและการแทนที่
เงื่อนไขภายหลัง:	รายการรหัสลำดับการทดสอบที่มีการเปลี่ยนแปลงจากแผนภาพลำดับ

ตารางที่ ก - 10 รายละเอียดยูสเคส Find Impact

รหัสยูสเคส:	10
ชื่อยูสเคส:	Find Impact
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	ค้นหารหัสลำดับการทดสอบที่ได้รับผลกระทบจากการเปรียบเทียบ
ความสัมพันธ์	Include: Load Comparison Result
เงื่อนไขก่อนหน้า:	นำเข้าข้อมูลลำดับการทดสอบที่มีการเปลี่ยนแปลงจากแผนภาพลำดับ
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Load Comparison Result เพื่อนำรายการรหัสลำดับการทดสอบเข้าสู่เครื่องมือทดสอบ 2. นำข้อมูลที่อยู่ในรายการรหัสลำดับการทดสอบที่มีประเภทเป็นการลบออกไปค้นหาตำแหน่งในตารางความสัมพันธ์เวอร์ชันต้นฉบับ 3. เมื่อพบตำแหน่งแล้วจะนำข้อมูลรหัสลำดับการทดสอบที่อยู่ด้านบนและด้านล่างไปจัดเก็บไว้ในรายการผลกระทบจากการเปลี่ยนแปลง 4. นำข้อมูลที่อยู่ในรายการรหัสลำดับการทดสอบที่มีประเภทเป็นการแทรกและลบไปค้นหาตำแหน่งในตารางความสัมพันธ์เวอร์ชันใหม่

ตารางที่ ก - 11 รายละเอียดยูสเคส Find Impact (ต่อ)

	5. เมื่อพบตำแหน่งแล้วจะนำข้อมูลรหัสลำดับการทดสอบที่อยู่ด้านบนและด้านล่างไปจัดเก็บไว้ในรายการผลกระทบจากการเปลี่ยนแปลง
เงื่อนไขภายหลัง:	รายการรหัสลำดับการทดสอบที่ได้รับผลกระทบจากการเปลี่ยนแปลงจากแผนภาพลำดับเวอร์ชันใหม่

ตารางที่ ก - 12 รายละเอียดยูสเคส Generate List of Impacted

รหัสยูสเคส:	11
ชื่อยูสเคส:	Generate List of Impacted
ผู้กระทำหลัก (Actor):	-
รายละเอียดยูสเคส:	สร้างและแบ่งประเภทของรายการรหัสลำดับการทดสอบที่ได้รับผลกระทบจากการเปลี่ยนแปลงจากแผนภาพลำดับเวอร์ชันใหม่
ความสัมพันธ์	Include: Find Impact
เงื่อนไขก่อนหน้า:	รายการรหัสลำดับการทดสอบที่ได้รับผลกระทบจากการเปลี่ยนแปลง
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Find Impact เพื่อนำเข้าข้อมูลรายการรหัสลำดับการทดสอบที่ได้รับผลกระทบ 2. เครื่องมือทดสอบทำการแบ่งประเภทของรหัสลำดับการทดสอบตามประเภทของการเปรียบเทียบ 3. แสดงข้อมูลรายการรหัสการทดสอบที่ได้รับผลกระทบและแบ่งตามประเภทของการเปรียบเทียบบนหน้าจอ
เงื่อนไขภายหลัง:	รายการรหัสการทดสอบที่ได้รับผลกระทบจากการเปลี่ยนแปลงที่แบ่งตามประเภทของการเปรียบเทียบ

ตารางที่ ก - 13 รายละเอียดชุดทดสอบ Load Existing Test Cases

รหัสชุดทดสอบ:	12
ชื่อชุดทดสอบ:	Load Existing Test Cases
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดชุดทดสอบ:	นำเข้ากรณีทดสอบในฐานข้อมูลทดสอบ
ความสัมพันธ์	-
เงื่อนไขก่อนหน้า:	สร้างกรณีทดสอบและจัดเก็บอยู่ในฐานข้อมูล
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เครื่องมือทดสอบดึงข้อมูลกรณีทดสอบของแผนภาพลำดับที่มีอยู่ในฐานข้อมูลขึ้นมา 2. เครื่องมือทดสอบจัดเตรียมข้อมูลกรณีทดสอบให้พร้อม
เงื่อนไขภายหลัง:	-

ตารางที่ ก - 14 รายละเอียดชุดทดสอบ Load List of Impacted

รหัสชุดทดสอบ:	13
ชื่อชุดทดสอบ:	Load List of Impacted
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดชุดทดสอบ:	นำเข้ารายการรหัสการทดสอบที่ได้รับผลกระทบจากการเปลี่ยนแปลง
ความสัมพันธ์	-
เงื่อนไขก่อนหน้า:	รายการรหัสการทดสอบที่ได้รับผลกระทบจากการเปลี่ยนแปลงที่แบ่งตามประเภทของการเปรียบเทียบ
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เครื่องมือทดสอบนำเข้าข้อมูลรายการรหัสการทดสอบที่ได้รับผลกระทบจากการเปลี่ยนแปลง
เงื่อนไขภายหลัง:	-

ตารางที่ ก - 15 รายละเอียดยูสเคส Select Test Case

รหัสยูสเคส:	14
ชื่อยูสเคส:	Select Test Case
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	เลือกกรณีทดสอบจากผลกระทบการเปลี่ยนแปลง
ความสัมพันธ์	Include: Load Existing Test Cases, Load List of Impacted
เงื่อนไขก่อนหน้า:	รายการกรณีทดสอบและรายการรหัสลำดับการทดสอบ
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Load Existing Test Cases และ Load List Impacted เพื่อใช้ในการเลือกกรณีทดสอบต่อไป 2. เครื่องมือทดสอบทำการอ่านข้อมูลจากรายการรหัสลำดับการทดสอบและนำไปค้นหาในรายการกรณีทดสอบที่ละรายการ 3. เมื่อพบรหัสลำดับการทดสอบจากกรณีทดสอบให้นำรหัสกรณีทดสอบไปจัดเก็บไว้ในรายการกรณีทดสอบที่ได้รับผลกระทบ
เงื่อนไขภายหลัง:	รายการกรณีทดสอบทั้งหมดที่ได้รับผลกระทบ

ตารางที่ ก - 16 รายละเอียดยูสเคส Display Regression Test Set

รหัสยูสเคส:	15
ชื่อยูสเคส:	Display Regression Test Set
ผู้กระทำหลัก (Actor):	-
รายละเอียดยูสเคส:	แสดงรายการกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอย
ความสัมพันธ์	Include: Select Test Case
เงื่อนไขก่อนหน้า:	รายการกรณีทดสอบทั้งหมดที่ได้รับผลกระทบ
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Select Test Case เพื่อแสดงรายการกรณีทดสอบทั้งหมดที่ได้รับผลกระทบ 2. เครื่องมือทดสอบทำการอ่านข้อมูลจากรายการกรณีทดสอบ 3. เครื่องมือทดสอบทำการแยกประเภทกรณีทดสอบแบบตามประเภทของกรณีทดสอบ 4. แสดงรายการกรณีทดสอบสำหรับชุดทดสอบเชิงถดถอยบนหน้าจอ
เงื่อนไขภายหลัง:	-

ตารางที่ ก - 17 รายละเอียดยูสเคส Load Sequence Dependency Table

รหัสยูสเคส:	16
ชื่อยูสเคส:	Load Sequence Dependency Table
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	นำเข้าตารางความสัมพันธ์แผนภาพลำดับ
ความสัมพันธ์	-
เงื่อนไขก่อนหน้า:	นำเข้าไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับ
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. ผู้ใช้งานทำการเลือกไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับ 2. เครื่องมือทดสอบทำการสร้างตารางความสัมพันธ์ 3. เครื่องมือทดสอบแสดงข้อมูลตารางความสัมพันธ์บนหน้าจอ
เงื่อนไขภายหลัง:	ตารางความสัมพันธ์จากแผนภาพลำดับถูกสร้างขึ้น

ตารางที่ ก - 18 รายละเอียดยูสเคส Create Test Case

รหัสยูสเคส:	17
ชื่อยูสเคส:	Create Test Case
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	สร้างกรณีทดสอบจากแผนภาพลำดับ
ความสัมพันธ์	Include: Load Sequence Dependency Table
เงื่อนไขก่อนหน้า:	ข้อมูลตารางความสัมพันธ์แผนภาพลำดับ
ขั้นตอนการทำงาน:	<ol style="list-style-type: none"> 1. เรียกใช้งานยูสเคส Load Sequence Dependency Table เพื่อนำเข้าข้อมูลตารางความสัมพันธ์แผนภาพลำดับ 2. เครื่องมือทดสอบอ่านข้อมูลจากตารางความสัมพันธ์แผนภาพลำดับ 3. เครื่องมือทดสอบทำการสร้างกรณีทดสอบจากตารางความสัมพันธ์
เงื่อนไขภายหลัง:	กรณีทดสอบจากแผนภาพลำดับ

ตารางที่ ก - 19 รายละเอียดยูนิตทดสอบ Save Test Case

รหัสยูนิตทดสอบ:	18
ชื่อยูนิตทดสอบ:	Save Test Case
ผู้กระทำหลัก (Actor):	-
รายละเอียดยูนิตทดสอบ:	จัดเก็บกรณีทดสอบเข้าสู่ฐานข้อมูล
ความสัมพันธ์	Include: Create Test Case
เงื่อนไขก่อนหน้า:	นำเข้าไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับ
ขั้นตอนการทำงาน:	1. เรียกใช้งานยูนิตทดสอบ Create Test Case เพื่อนำข้อมูลกรณีทดสอบที่ ถูกสร้างมาใช้งาน
	2. เครื่องมือทดสอบทำการจัดเก็บข้อมูลเข้าสู่ฐานข้อมูล
เงื่อนไขภายหลัง:	-

ตารางที่ ก - 20 รายละเอียดยูนิตทดสอบ Load Existing Test Cases

รหัสยูนิตทดสอบ:	19
ชื่อยูนิตทดสอบ:	Load Existing Test Cases
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูนิตทดสอบ:	เรียกดูข้อมูลกรณีทดสอบในฐานข้อมูลทดสอบ
ความสัมพันธ์	-
เงื่อนไขก่อนหน้า:	สร้างกรณีทดสอบและจัดเก็บอยู่ในฐานข้อมูล
ขั้นตอนการทำงาน:	1. ผู้ใช้ทำการเลือกแผนภาพลำดับที่ต้องการ 2. เครื่องมือทดสอบดึงข้อมูลกรณีทดสอบของแผนภาพลำดับที่มีอยู่ใน ฐานข้อมูลขึ้นมา 3. เครื่องมือทดสอบจัดเตรียมข้อมูลกรณีทดสอบให้พร้อม
เงื่อนไขภายหลัง:	-

ตารางที่ ก - 21 รายละเอียดทดสอบ Display Test Case

รหัสยูสเคส:	20
ชื่อยูสเคส:	Display Test Cases
ผู้กระทำหลัก (Actor):	ผู้ใช้งาน
รายละเอียดยูสเคส:	แสดงข้อมูลกรณีทดสอบบนหน้าจอ
ความสัมพันธ์	Include: Load Existing Test Case
เงื่อนไขก่อนหน้า:	ดึงข้อมูลกรณีทดสอบของแผนภาพลำดับจากฐานข้อมูล
ขั้นตอนการทำงาน:	1. เครื่องมือทดสอบแสดงข้อมูลกรณีทดสอบบนหน้าจอ
เงื่อนไขภายหลัง:	-

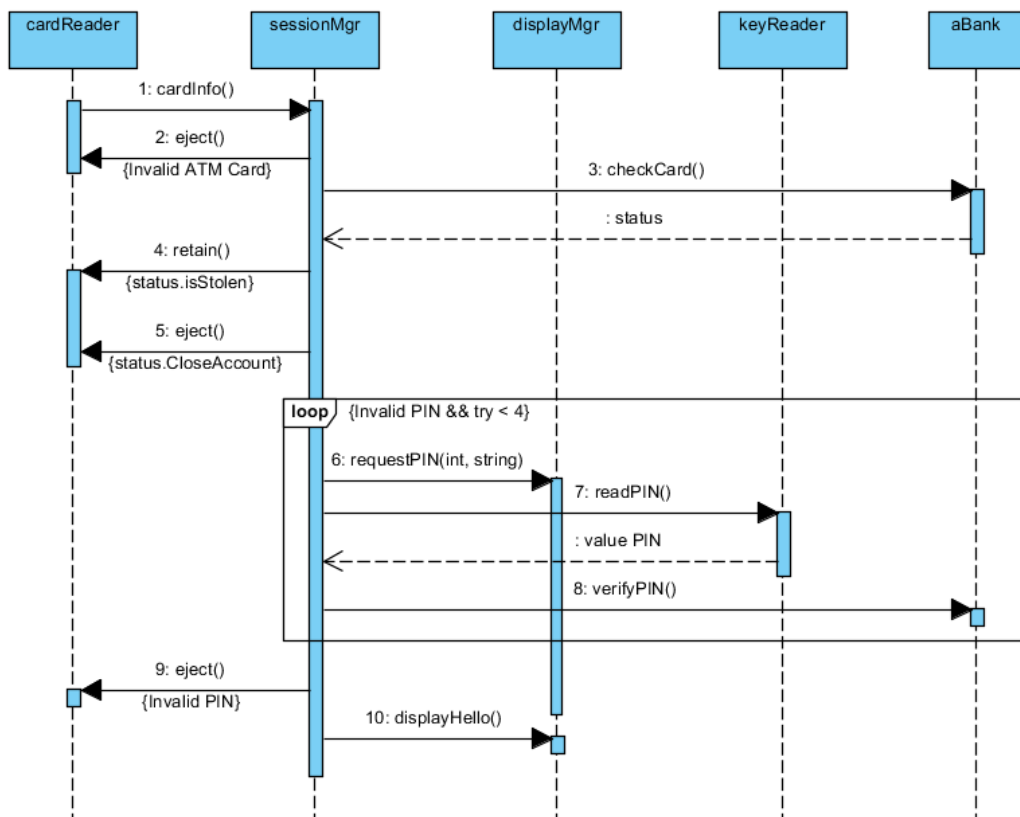


ภาคผนวก ข แผนภาพที่ใช้ในการทดสอบ

1. แผนภาพที่ 1 ระบบเอทีเอ็ม

1.1 แผนภาพลำดับเวอร์ชันต้นฉบับ

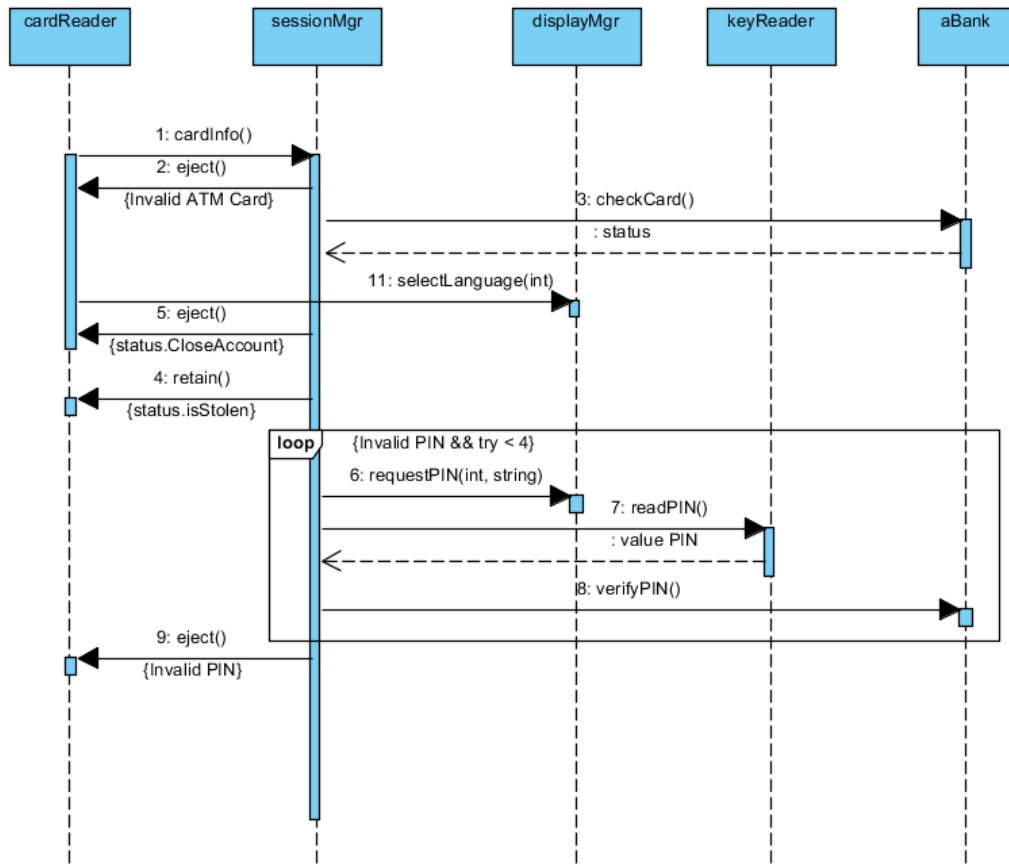
แผนภาพลำดับของระบบเอทีเอ็มในส่วนของการตรวจสอบรหัสบัตรที่ถูกใช้ในงานวิจัย [14] ซึ่งจะถูกใช้เป็นแผนภาพลำดับสำหรับเวอร์ชันต้นฉบับ ซึ่งในแผนภาพจะประกอบด้วยความสัมพันธ์ของการทำงานของระบบเอทีเอ็มที่แสดงได้ดังรูปที่ ข-1



รูปที่ ข - 1 แผนภาพลำดับของระบบเอทีเอ็มเวอร์ชันต้นฉบับ

1.2 แผนภาพลำดับเวอร์ชันใหม่

แผนภาพลำดับของระบบเอทีเอ็มในส่วนของการตรวจสอบรหัสบัตรที่มีการแก้ไขซึ่งจะถูกใช้เป็นแผนภาพลำดับเวอร์ชันใหม่ ซึ่งรายละเอียดของการเปลี่ยนแปลงจะมีการแทรกเมสเสจ การลบออกและการแทนที่ ซึ่งสามารถแสดงได้ดังรูปที่ ข-2

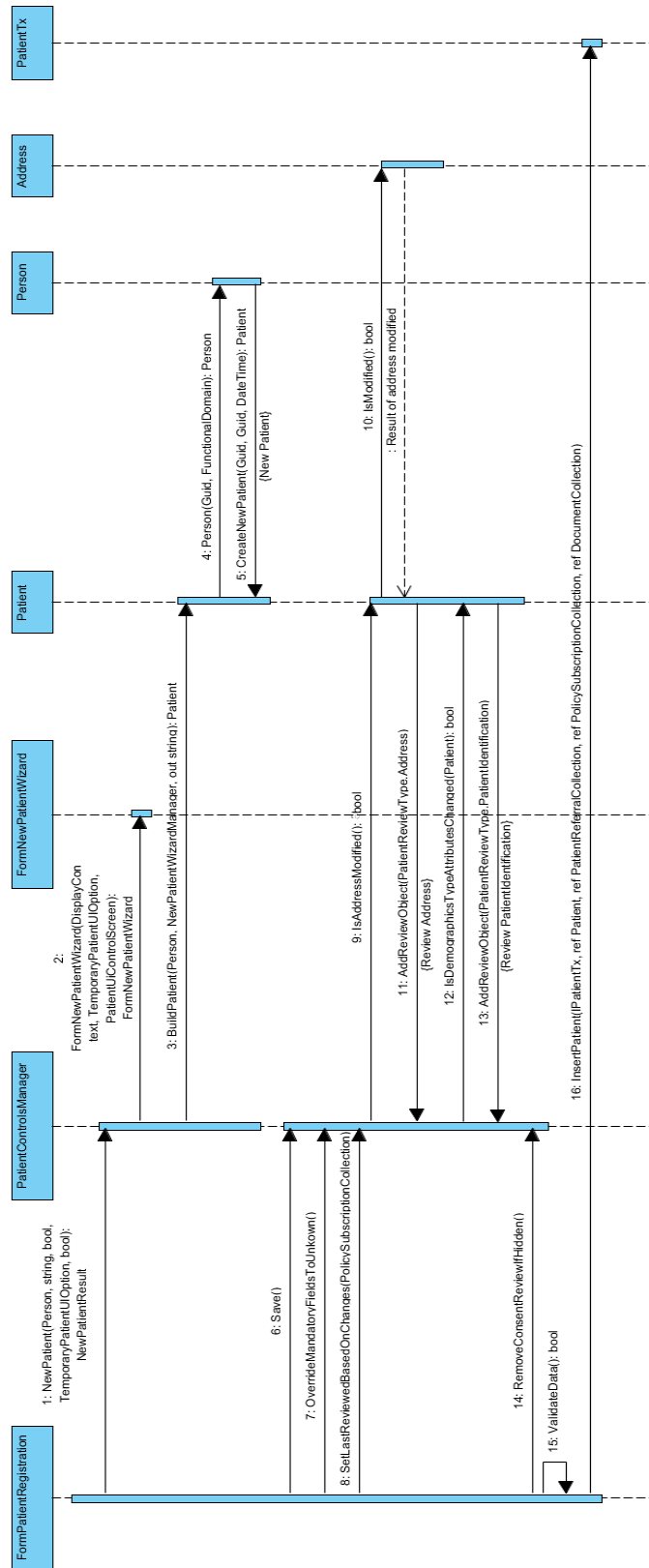


รูปที่ ข - 2 แผนภาพลำดับของระบบเอทีเอ็มเวอร์ชันใหม่

2. แผนภาพที่ 2 ระบบลงทะเบียนคนใช้

2.1 แผนภาพลำดับเวอร์ชันต้นฉบับ

แผนภาพลำดับของระบบลงทะเบียนคนใช้ ซึ่งจะถูกใช้เป็นแผนภาพลำดับสำหรับเวอร์ชันต้นฉบับ ซึ่งในแผนภาพจะประกอบด้วยความสัมพันธ์ของการทำงานของระบบลงทะเบียนคนใช้ที่แสดงได้ดังรูปที่ ข-3

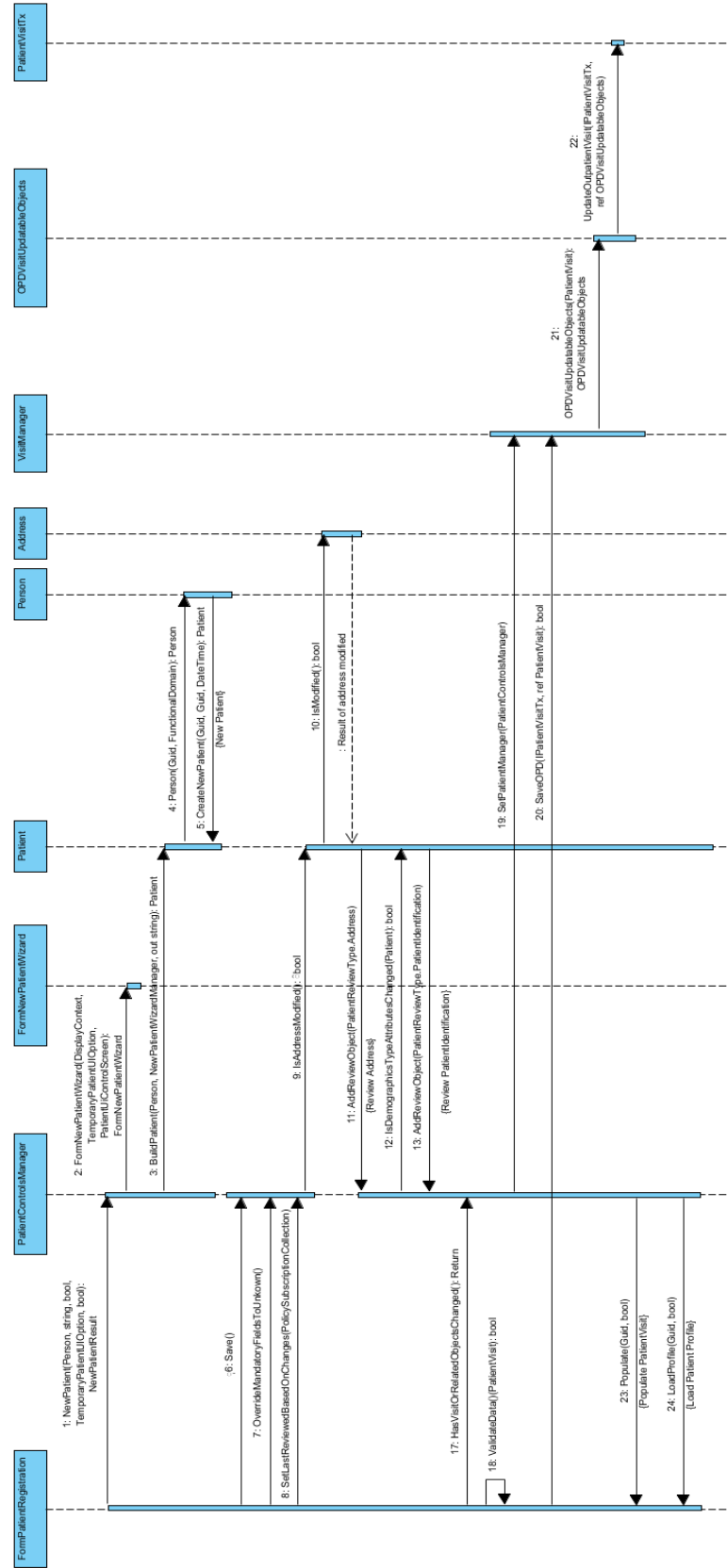


รูปที่ ข - 3 แผนภาพลำดับของระบบลงทะเบียนคนไข้เวอร์ชันต้นฉบับ

2.2 แผนภาพลำดับเวอร์ชันใหม่

แผนภาพลำดับของระบบลงทะเบียนคนไข้ที่มีการแก้ไขให้รองรับการลงทะเบียนคนไข้แบบผู้ป่วยนอก ซึ่งจะถูกใช้เป็นแผนภาพลำดับเวอร์ชันใหม่ ซึ่งรายละเอียดของการเปลี่ยนแปลงสามารถแสดงได้ดังรูปที่ ข-4





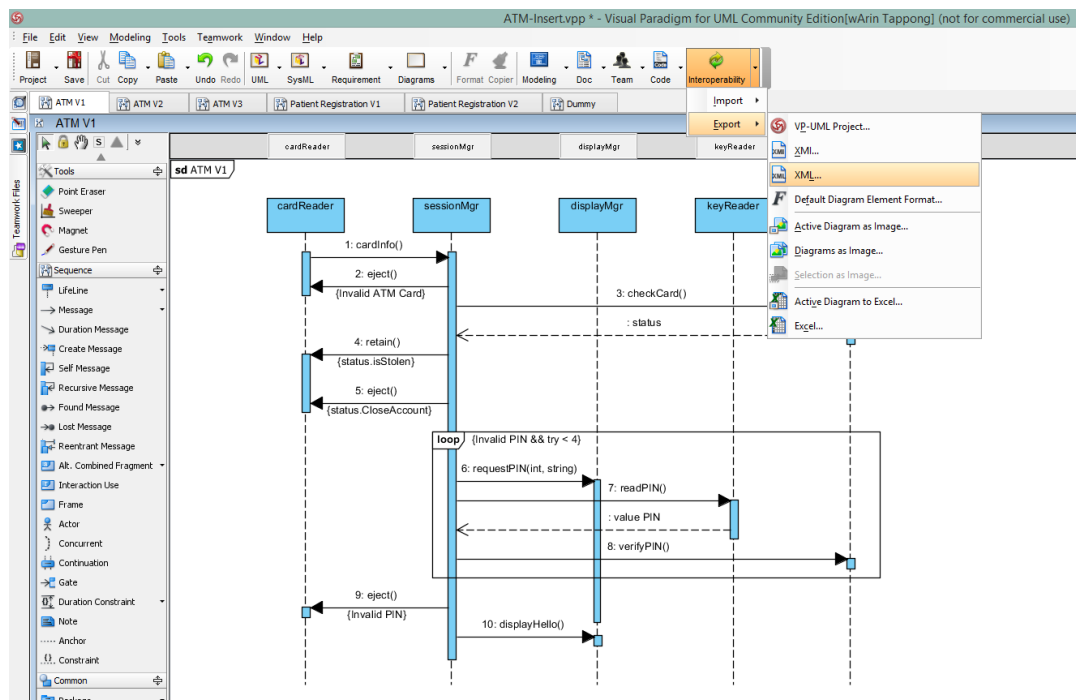
รูปที่ ข - 4 แผนภาพลำดับของระบบลงทะเบียนคนไข้เวอร์ชันใหม่

ภาคผนวก ค วิธีการนำออกแผนภาพลำดับ

ขั้นตอนการนำออกแผนภาพลำดับ

แผนภาพลำดับสามารถนำออกได้ด้วยขั้นตอนดังต่อไปนี้

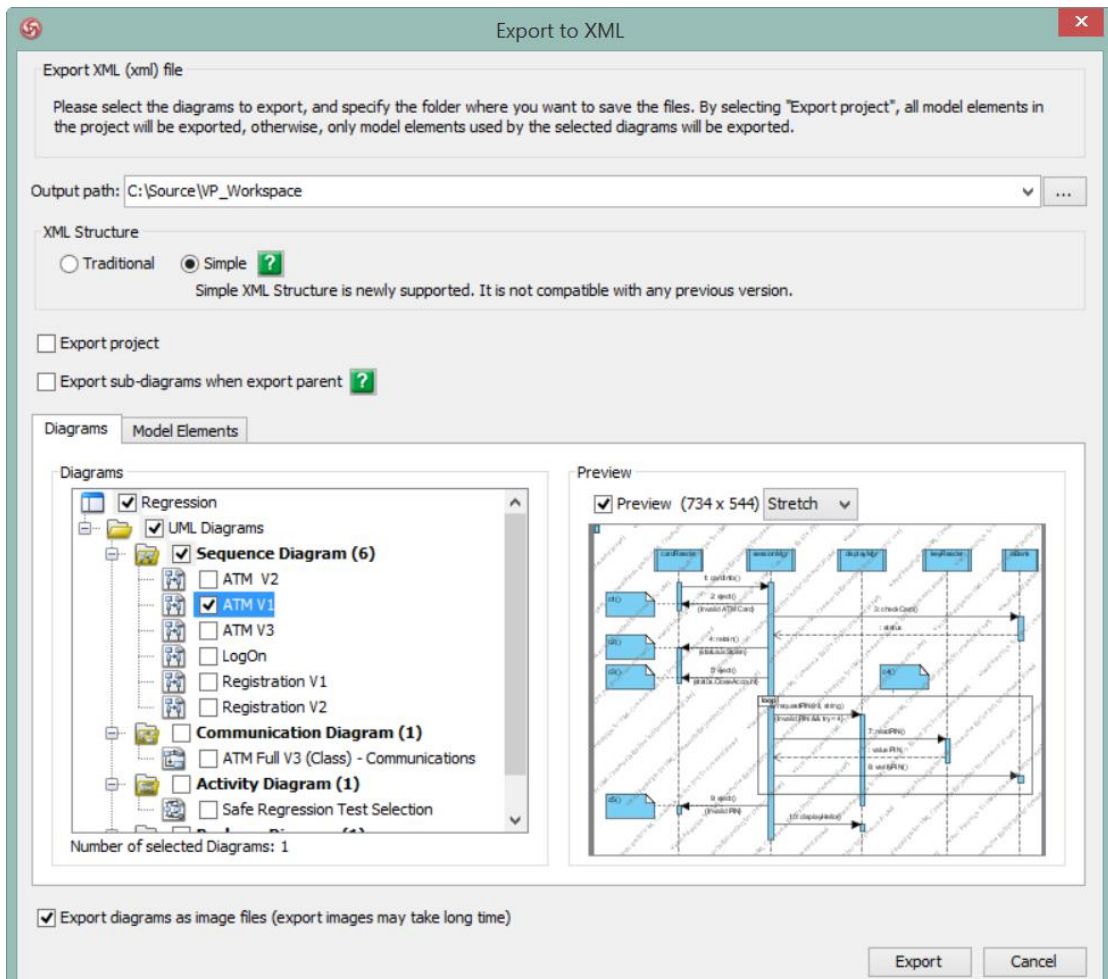
1. ทำการออกแบบแผนภาพลำดับตามต้องการ
2. เลือกรายการ Interoperability > Export > เลือกรายการ XML... ดังรูปที่ ค-1



รูปที่ ค - 1 เลือกรายการสำหรับการนำออกแผนภาพลำดับ

3. เลือกโฟลเดอร์ (Folder) ที่ต้องการจะนำไฟล์ออกหรือระบุตำแหน่ง
4. เลือกประเภทของ XML Structure เป็น Simple
5. เลือกแผนภาพลำดับที่ต้องการจะนำออกจากรายการทั้งหมดที่ถูกแสดงบนหน้าจอ
6. ในกรณีที่ต้องการนำออกไฟล์ที่เป็นรูปภาพของแผนภาพลำดับจะต้องทำการเลือกรายการ Export diagrams as image files (export image may take long time)
7. เลือกรายการ Preview ในกรณีที่ต้องการจะดูรูปตัวอย่างของแผนภาพลำดับก่อนการนำออก
8. กดปุ่ม Export เพื่อทำการนำออกแผนภาพลำดับ

โดยขั้นตอนที่ 3-8 สามารถแสดงหน้าจอตัวอย่างได้ดังรูปที่ ค-2



รูปที่ ค - 2 หน้าจอการนำออกไฟล์เอกซ์เอ็มแอลของแผนภาพลำดับสำหรับเครื่องมือทดสอบ

ภาคผนวก ง

คู่มือการติดตั้งเครื่องมือ

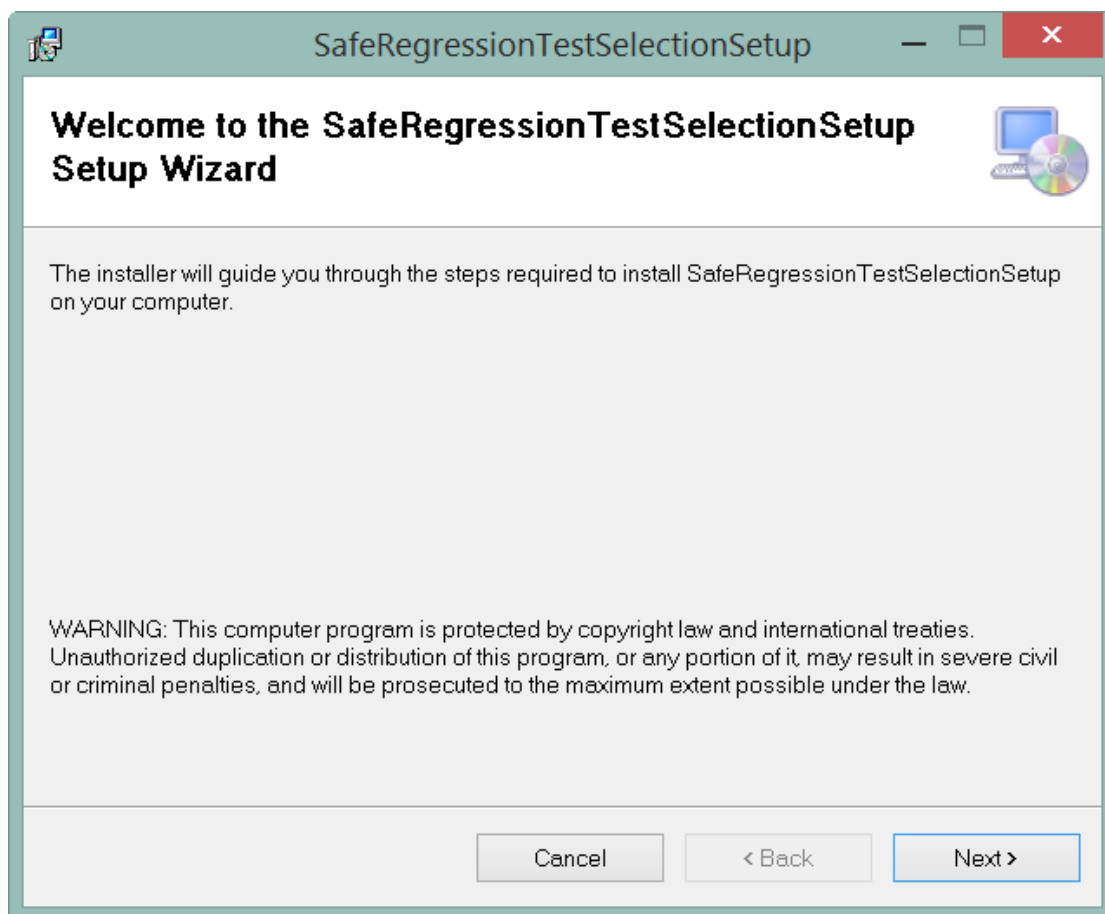
ขั้นตอนการติดตั้งเครื่องมือทดสอบ

1. ใส่แผ่นซีดีรอม (CD-Rom) เพื่อติดตั้งเครื่องมือทดสอบ
2. เลือกไฟล์ที่ใช้ในการติดตั้งที่มีชื่อว่า SafeRegressionSetup.exe



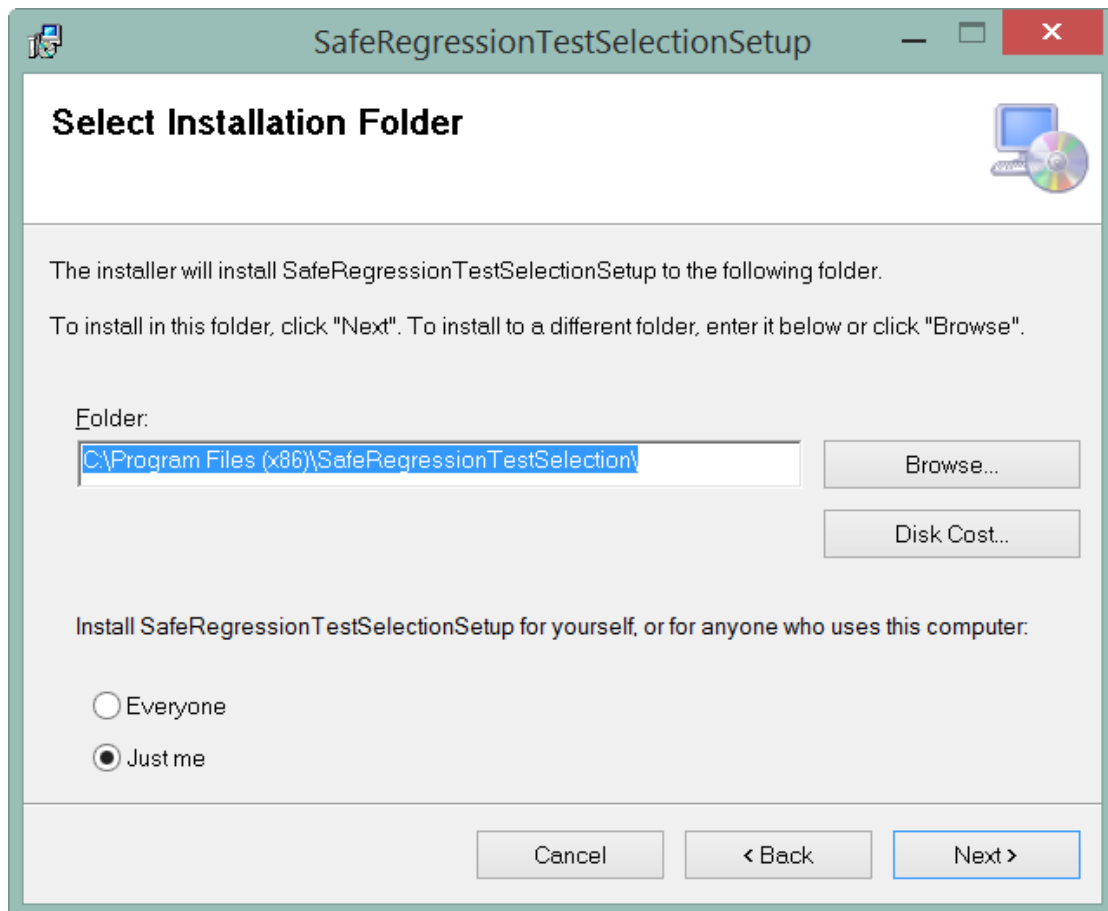
รูปที่ ง - 1 ไฟล์สำหรับใช้ในการติดตั้งเครื่องมือทดสอบ

3. ดับเบิลคลิกที่ไฟล์ เพื่อเริ่มต้นการติดตั้ง
4. หน้าจอเริ่มต้นการติดตั้งจะถูกแสดงขึ้นมา ดังรูปที่ ง-2



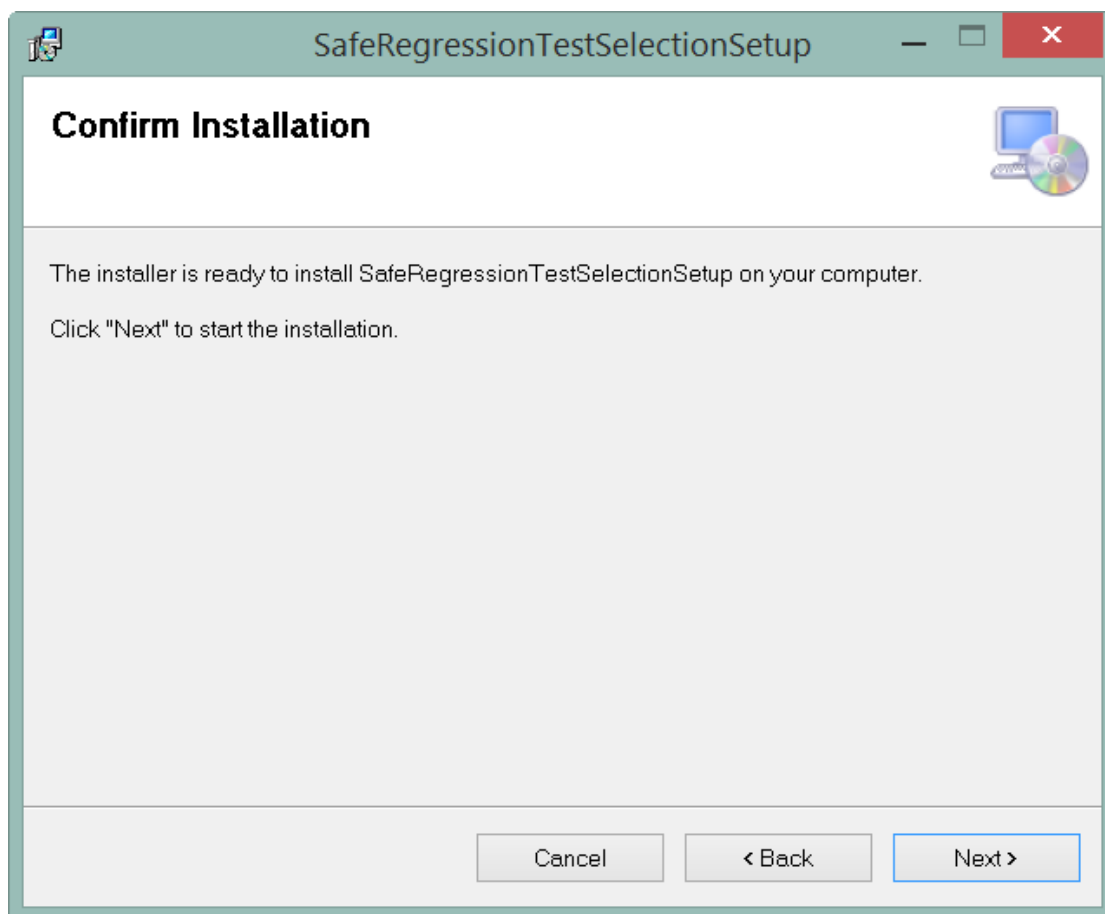
รูปที่ ง - 2 หน้าจอเริ่มต้นการติดตั้งเครื่องมือทดสอบ

5. คลิกปุ่ม Next เพื่อดำเนินการขั้นตอนต่อไป
6. ทำการระบุโฟลเดอร์หรือเลือกโฟลเดอร์ที่ต้องการจะติดตั้งเครื่องมือทดสอบโดยการกดปุ่ม Browse ดังรูปที่ ง-3



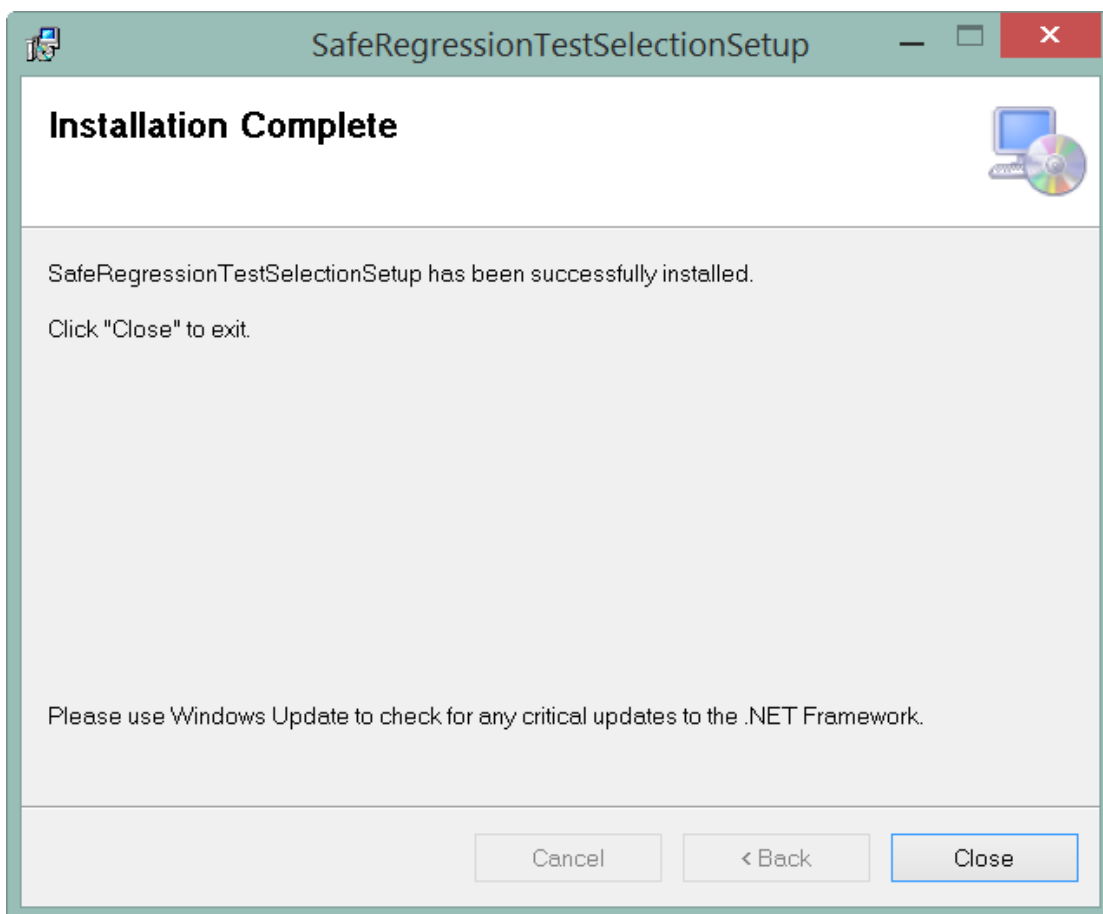
รูปที่ ง - 3 หน้าจอการเลือกโฟลเดอร์ที่ต้องการจะติดตั้ง

7. กดปุ่ม Next เพื่อดำเนินการขั้นตอนต่อไป
8. ทำการยืนยันการติดตั้งเครื่องมือทดสอบ โดยการกดปุ่ม Next ดังรูปที่ ง-4



รูปที่ ง - 4 หน้าจอยืนยันการติดตั้งเครื่องมือทดสอบ

9. เมื่อติดตั้งเครื่องมือทดสอบเสร็จเรียบร้อยแล้วจะแสดงหน้าจอการติดตั้งสำเร็จ
10. กดปุ่ม Close เพื่อจบขั้นตอนการติดตั้งเครื่องมือทดสอบ ดังรูปที่ ง-5



รูปที่ ง - 5 หน้าจอแสดงการติดตั้งเสร็จสมบูรณ์

11. เครื่องมือทดสอบจะสามารถใช้งานได้จากชอร์ตคัท (Shortcut) บนหน้าเดสก์ท็อป (Desktop) ดังรูปที่ ง-6



รูปที่ ง - 6 หน้าจอแสดงชอร์ตคัทของเครื่องมือติดตั้งบนเดสก์ท็อป

ประวัติผู้เขียนวิทยานิพนธ์

นายวรินทร์ ทัพพงษ์ เกิดเมื่อวันที่ 13 กรกฎาคม พ.ศ. 2526 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาระดับปริญญาบัณฑิต หลักสูตรบริหารธุรกิจบัณฑิต สาขาเทคโนโลยีสารสนเทศ ธุรกิจ จากมหาวิทยาลัยธุรกิจบัณฑิต เมื่อ พ.ศ. 2549 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ณ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อปี พ.ศ. 2554

ประวัติการทำงาน หลังจากจบการศึกษาระดับปริญญาบัณฑิตได้เริ่มทำงานเป็น นักพัฒนาซอฟต์แวร์ที่ บริษัท ไอเดนทิฟาย จำกัด จากนั้นได้ย้ายไปทำงานที่ บริษัท แกรนด์โพร์ โซลูชัน จำกัด (มหาชน) เมืองกัวลาลัมเปอร์ ประเทศมาเลเซีย และหลังจากนั้นได้เปลี่ยนสายงาน ไปเป็นนักทดสอบซอฟต์แวร์ที่ บริษัท ไมโครซอฟท์ (ประเทศไทย) จำกัด และปัจจุบันได้ย้ายไป ทำงานที่ บริษัท โอโรอน เฮลท์ (ประเทศไทย) จำกัด