การประเมินความเสี่ยงของระบบจากฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน

นางสาวกาญจนา เอี่ยมสอาด

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2556
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

System Risk Assessment based on Relational Database Schema-based Constraints Using State Machine Diagram

Miss Kanjana Eiamsaard

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Software Engineering
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2013
Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การประเมินความเสี่ยงของระบบจากฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน
โดย	นางสาวกาญจนา เอี่ยมสอาด
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ นครทิพย์ พร้อมพูล

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

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

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ
(รองศาสตราจารย์ ดร.พรศิริ หมื่นไชยศรี)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ นครทิพย์ พร้อมพูล)

.....กรรมการ
(รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ)

.....กรรมการภายนอกมหาวิทยาลัย
(ผู้ช่วยศาสตราจารย์ ดร. มฑุปายาส ทองมาก)

กาญจนา เอี่ยมสอาด : การประเมินความเสี่ยงของระบบจากฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน (System Risk Assessment based on Relational Database Schema-based Constraints Using State Machine Diagram) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. นครทิพย์ พร้อมพูล, 149 หน้า.

การประเมินความเสี่ยงสามารถนำไปใช้ได้หลายส่วนในโครงการด้านซอฟต์แวร์ ฟังก์ชันงานที่สำคัญส่วนหนึ่งของซอฟต์แวร์คือ ซอฟต์แวร์ที่เกี่ยวกับการจัดการด้านข้อมูล ซึ่งการเปลี่ยนแปลงข้อมูลอาจส่งผลกระทบต่อสถานะของซอฟต์แวร์และอาจเป็นผลให้เกิดการทำงานที่ผิดพลาดของซอฟต์แวร์ได้

งานวิจัยฉบับนี้จึงนำเสนอขั้นตอนวิธีสำหรับการประเมินความเสี่ยงที่ ซอฟต์แวร์จะทำให้ข้อมูลที่ถูกจัดเก็บลงฐานข้อมูลเชิงสัมพันธ์เกิดข้อผิดพลาด ที่มีสาเหตุจากเงื่อนไขเชิงเค้าร่างอันประกอบด้วย เงื่อนไขของค่าว่าง เงื่อนไขประเภทข้อมูล เงื่อนไขของคีย์หลัก และเงื่อนไขความสอดคล้องของข้อมูล ด้วยการพิจารณาความเสี่ยงที่เกิดขึ้นในสามระดับคือ ลำดับเหตุการณ์ของซอฟต์แวร์ องค์ประกอบของลำดับเหตุการณ์ และการเชื่อมโยงระหว่างองค์ประกอบ ด้วยการสร้างแบบจำลองความเสี่ยงที่ลำดับเหตุการณ์ของซอฟต์แวร์ทำงานผิดพลาดในสถานะความรุนแรงระดับต่างๆ จากการประยุกต์ใช้แบบจำลองของมาร์คอฟ สำหรับการประเมินความเสี่ยงในระดับองค์ประกอบนั้น ได้ประยุกต์ใช้วิธีสำหรับสร้างกรณีทดสอบตามหลักการณ์ครอบคลุมประพจน์ด้วยการสร้างต้นไม้แจกแจงทวิภาค และการสร้างเส้นทางการทดสอบแบบไพร์ม เพื่อหาความน่าจะเป็นของการทำงานที่ผิดพลาดขององค์ประกอบจากแผนภาพสเตทแมชชีน ทั้งนี้เพื่อทำให้ผู้ประเมินสามารถนำผลลัพธ์ที่ได้ไปใช้ในขั้นตอนการจัดลำดับความสำคัญในการควบคุมความเสี่ยงต่อไป เพื่อให้ง่ายต่อการประยุกต์ใช้วิธีการที่นำเสนอ ผู้วิจัยจึงได้สร้างเครื่องมือสนับสนุน นอกจากนี้ได้นำขั้นตอนวิธีนี้มาใช้กับกรณีศึกษา พบว่าผลลัพธ์ที่ได้จากการประเมินความเสี่ยงทั้งสามระดับสอดคล้องกับความเห็นของผู้เชี่ยวชาญ

ภาควิชา: วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต .....

สาขาวิชา: วิศวกรรมซอฟต์แวร์ ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลัก .....

ปีการศึกษา: 2556

5470908121: MAJOR SOFTWARE ENGINEERING
KEYWORDS: RISK ASSESSMENT / RELATIONAL DATABASE / SCHEMA-BASE CONSTRAINTS / STATE MACHINE DIAGRAM

KANJANA EIAMSAARD: SYSTEM RISK ASSESSMENT BASED ON RELATIONAL DATABASE SCHEMA-BASED CONSTRAINTS USING STATE MACHINE DIAGRAM. ADVISOR: ASST.PROF. NAKORNTHIP PROMPOON, 149 pp.

Risk assessment can be used in different parts of software projects. Software function relateก to data management is one of the major software functions. The change of data may affect to the status of software and may result in software failure.

This research proposes an algorithm for risk assessment from the potential software failure focused on the process of the information storing in the relational database resulted from the database constraint. There are four types of database constraints covered in this research: 1) entity constraint, 2) domain constraint, 3) key constraint, and 4) referential integrity constraint. Software risk concerns are in three levels: 1) scenario, 2) component of scenario, and connection among software components. The risk assessment is based on scenario risk factor model in different levels of severity by applying Markov's chain model. Risk assessment in component level is based on test cases generation from full predicate coverage by constructing binary tree and prime path testing in order to find the probability of failure from components represented by state machine diagram. The result can be used by an assessor for risk control prioritization. In addition, to facilitate the application of the proposed method, a supporting tool is also developed. Furthermore, from the application of our proposed algorithm to the case study, the risk result in all three levels conforms to the expert judgment.

Department: Computer Engineering Student's Signature:
Field of Study: Software Engineering Advisor's Signature:
Academic Year: 2013

กิตติกรรมประกาศ

ขอขอบพระคุณบิดา มารดาและสมาชิกในครอบครัวทุกคน ที่ให้การสนับสนุน ในการศึกษาเล่าเรียนจนสำเร็จหลักสูตรมหาบัณฑิต

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ นครทิพย์ พร้อมพูล อาจารย์ที่ปรึกษา ที่สละเวลาอันมีค่าเพื่ออบรม ช่วยเหลือ และแนะนำแนวทางที่พึงปฏิบัติต่อตัวผู้วิจัย และลูกศิษย์ทุกคน

ขอขอบพระคุณ รองศาสตราจารย์ ดร.พรศิริ หมื่นไชยศรี ประธานกรรมการสอบวิทยานิพนธ์ รองศาสตราจารย์ ดร.วิวัฒน์ วัฒนาวุฒิ และ ผู้ช่วยศาสตราจารย์ ดร. มฑุปายาส ทองมาก กรรมการสอบวิทยานิพนธ์ ที่ได้กรุณาสละเวลาให้คำแนะนำสำหรับโครงร่างวิทยานิพนธ์และวิทยานิพนธ์ให้มีคุณภาพยิ่งขึ้น

ขอขอบพระคุณ Mr. John Gatewood Ham สำหรับความรู้ ข้อคิด และแรงบันดาลใจในการแก้ปัญหา

ขอบคุณ คุณศยามล วัฒนทอง และเพื่อนๆทุกคนที่คอยช่วยเหลือและเป็นกำลังใจ

ท้ายที่สุด ขอบคุณผู้มีส่วนร่วมทุกคนที่ทำให้งานวิจัยฉบับนี้สำเร็จลุล่วงไปด้วยดี

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง
บทคัดย่อภาษาอังกฤษ ..... จ
กิตติกรรมประกาศ ..... ฉ
สารบัญ ..... ช
สารบัญตาราง ..... ญ
สารบัญภาพ ..... ฏ
บทที่ 1 บทนำ 1
1.1. ที่มาและความสำคัญของปัญหา 1
1.2. วัตถุประสงค์ 4
1.3. ขอบเขตงานวิจัย 4
1.4. ขั้นตอนและวิธีดำเนินงานวิจัย 4
1.5. ประโยชน์ที่คาดว่าจะได้รับ 5
1.6. บทความวิชาการที่ได้รับการตีพิมพ์ 6
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 7
2.1. ทฤษฎีและคำศัพท์ที่เกี่ยวข้อง 7
2.1.1. ความเสี่ยง (Risk) 7
2.1.2. เงื่อนไขเชิงเค้าร่างของฐานข้อมูล (Schema-based constraints) 9
2.1.3. แผนภาพสเตทแมชชีน (State machine diagram) 11
2.1.4. การสร้างเส้นทางทดสอบแบบไพร์มพาธ (Prime path testing) 14
2.1.5. วิธีการสร้างแผนภาพความสัมพันธ์ของเอนทิตี้จากแผนภาพคลาส 17
2.2. งานวิจัยที่เกี่ยวข้อง 18
2.2.1. การวิเคราะห์ความเสี่ยงในระดับสถาปัตยกรรมโดยใช้ยูเอ็มแอล 18
2.2.2. การสร้างข้อมูลทดสอบจากข้อกำหนดรายละเอียดเชิงสถานะ
(Generating test data from state-based specifications) 25
2.2.3. Test cases generation from a state chart diagram 28

หน้า

2.2.4. Risk Assessment Framework based on Goal-oriented Requirements Engineering and Object Behavioral Model 30
บทที่ 3 ขั้นตอนการประเมินความเสี่ยงของระบบจากฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่าง . 34
3.1. การวิเคราะห์และออกแบบขั้นตอนวิธีการประเมินความเสี่ยง..... 37
3.2. กำหนดขั้นตอนวิธีสำหรับการประเมินความเสี่ยง 41
3.3. การกำหนดขอบเขตและการออกแบบการทำงานของซอฟต์แวร์..... 46
3.4. การประเมินความเสี่ยงระดับองค์ประกอบ 51
3.5. ประเมินความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบของเหตุการณ์ใดๆ จากแผนภาพสเตทแมชชีน..... 55
3.6. การประเมินความเสี่ยงภายในลำดับเหตุการณ์ 56
3.6.1. การสร้างเมทริกซ์ความน่าจะเป็นของการทำงานที่ปราศจากความผิดพลาดของลำดับเหตุการณ์ 56
3.6.2. การสร้างเมทริกซ์ความน่าจะเป็นของการทำงานที่ผิดผลาดของลำดับเหตุการณ์ 56
3.6.3. จัดอันดับของลำดับเหตุการณ์ตามค่าความเสี่ยงที่คำนวณ..... 57
บทที่ 4 การประยุกต์ใช้ขั้นตอนการประเมินความเสี่ยง ของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน 59
4.1. กรณีศึกษาการประเมินความเสี่ยงฯ 59
4.2. การกำหนดขอบเขตและการออกแบบการทำงานของซอฟต์แวร์..... 62
4.3. การประเมินความเสี่ยงระดับองค์ประกอบ 91
4.3.1. การประเมินความเสี่ยงของแต่ละองค์ประกอบของลำดับเหตุการณ์จากแผนภาพสเตทแมชชีน..... 91
4.4. ประเมินความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบของเหตุการณ์ใดๆจากแผนภาพสเตทแมชชีน..... 96
4.5. การประเมินความเสี่ยงภายในลำดับเหตุการณ์ 97
4.5.1. การสร้างเมทริกซ์ความน่าจะเป็นของการทำงานที่ปราศจากความผิดพลาดของลำดับเหตุการณ์ 97

หน้า

4.5.2. การสร้างเมทริกซ์ความน่าจะเป็นของการทำงานที่ผิดผลาดของลำดับเหตุการณ์ 98
4.6. วิธีการประเมินขั้นตอนวิธี 101
บทที่ 5 วิเคราะห์ ออกแบบ พัฒนาเครื่องมือการประเมินความเสี่ยง ของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน 102
5.1. การรวบรวมและวิเคราะห์ความต้องการ 102
5.2. การออกแบบเครื่องมือการประเมินความเสี่ยง 108
5.3. การพัฒนาเครื่องมือสำหรับการประเมินความเสี่ยง 118
5.4. การทดสอบเครื่องมือสำหรับการประเมินความเสี่ยง 118
5.4.1. การตรวจสอบ (Validation) 118
5.4.2. การทวนสอบ (Verification) 119
5.4.3. การทดสอบระบบ 128
5.5. สรุปผล 128
บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ 129
6.1. สรุปผลการวิจัย 129
6.2. ข้อจำกัดของงานวิจัย 130
6.3. แนวทางการวิจัยในอนาคต 130
รายการอ้างอิง 132
ภาคผนวก 62
ภาคผนวก ก ตัวอย่างการสร้างกรณีทดสอบจากแผนภาพสเตทแมชชีน 133
ภาคผนวก ข คำอธิบายยูสเคสโดยละเอียดของเครื่องมือการประเมินความเสี่ยง 139
ประวัติผู้เขียนวิทยานิพนธ์ 149

สารบัญตาราง

หน้า

ตารางที่ 2.1 คำศัพท์ที่เกี่ยวข้อง 7
ตารางที่ 2.2 การนำเงื่อนไขเชิงเค้าร่างเมื่อนำไปใช้ร่วมกับตัวดำเนินการทางฐานข้อมูล 11
ตารางที่ 2.3 รายการประเภทของเหตุการณ์ในแผนภาพสเตทแมชชีน 12
ตารางที่ 2.4 รายการประเภทของการเปลี่ยนแปลง 13
ตารางที่ 2.5 ระดับความรุนแรงและดัชนีระดับความรุนแรงจากขั้นตอนการวิเคราะห์ภัยคุกคาม 21
ตารางที่ 2.6 ค่าความเสี่ยงของแต่ละลำดับเหตุการณ์ในระบบ 24
ตารางที่ 2.7 ค่าความจริงของนิพจน์ $A \vee B \wedge C$ 27
ตารางที่ 2.8 การเปลี่ยนสถานะของแผนภาพสเตทแมชชีนตัวอย่าง 29
ตารางที่ 2.9 ข้อกำหนดรายละเอียดประพจน์ที่ทำการขยายตัวกระตุ้นเหตุการณ์ 30
ตารางที่ 2.10 ความต้องการของกรณีทดสอบของประพจน์ที่ 1 (P1) 30
ตารางที่ 2.11 การเปลี่ยนสถานะของแผนภาพสเตทแมชชีน 31
ตารางที่ 2.12 ข้อกำหนดรายละเอียดประพจน์ที่ทำการขยายตัวกระตุ้นเหตุการณ์ 31
ตารางที่ 2.13 ความต้องการของกรณีทดสอบ 32
ตารางที่ 3.1 คำอธิบายคำศัพท์ที่ปรากฏในไวยากรณ์ภาษา 39
ตารางที่ 4.1 ความสัมพันธ์ระหว่างองค์ประกอบและตารางในฐานข้อมูลเชิงสัมพันธ์ 62
ตารางที่ 4.2 คำอธิบายแผนภาพแพคเกจของลำดับเหตุการณ์ในซอฟต์แวร์ระบบสารบรรณ 64
ตารางที่ 4.3 ที่มาและการเชื่อมโยงระหว่างแผนภาพคลาสและแผนภาพความสัมพันธ์ของเอนทิตี้ 66
ตารางที่ 4.4 คำอธิบายแผนภาพสเตทแมชชีนขององค์ประกอยการลงทะเบียนหนังสือ 72
ตารางที่ 4.5 คำอธิบายแผนภาพสเตทแมชชีนขององค์ประกอบการเสนอลงนาม 81
ตารางที่ 4.6 คำอธิบายแผนภาพสเตทแมชชีนขององค์ประกอบการส่งหนังสือถึงผู้รับผิดชอบ 86
ตารางที่ 4.7 จำนวณกรณีทดสอบขององค์ประกอบการลงทะเบียนหนังสือ 92
ตารางที่ 4.8 ระดับผลกระทบของแต่ละตารางในฐานข้อมูลและองค์ประกอบที่เกี่ยวข้อง 95
ตารางที่ 4.9 ระดับภัยคุกคามตามความรุนแรง 96
ตารางที่ 4.10 ความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบในลำดับเหตุการณ์การรับส่งหนังสือ 97
ตารางที่ 4.11 เมทริกซ์ความน่าจะเป็นเมื่อลำดับเหตุการณ์การรับส่งหนังสือทำงานได้สำเร็จ 98
ตารางที่ 4.12 เมทริกซ์ Q ของลำดับเหตุการณ์การรับส่งหนังสือ 99
ตารางที่ 4.13 เมทริกซ์ C ของลำดับเหตุการณ์การรับส่งหนังสือ 100
ตารางที่ 4.14 เมทริกซ์ Ax ของลำดับเหตุการณ์การรับส่งหนังสือ 100

หน้า

ตารางที่ 4.15 เมทริกซ์ $\mathbf{A'}$ ของลำดับเหตุการณ์การรับส่งหนังสือ 100
ตารางที่ 4.16 ปัจจัยความเสี่ยงของแต่ละลำดับเหตุการณ์ของซอฟต์แวร์ระบบสารบรรณ 101
ตารางที่ 5.1 รายการความต้องการเชิงหน้าที่ของเครื่องมือการประเมินความเสี่ยงฯ 102
ตารางที่ 5.2 รายการความต้องการเชิงคุณภาพของเครื่องมือการประเมินความเสี่ยงฯ 103
ตารางที่ 5.3 ตัวอย่างคำอธิบายยูสเคส SRR-07 106
ตารางที่ 5.4 รายการปฏิบัติตามความต้องการเชิงคุณภาพ 107
ตารางที่ 5.5 ผลลัพธ์การตรวจสอบความสมบูรณ์ระหว่างรายการความต้องเชิงหน้าที่กับเครื่องมือที่พัฒนา 119
ตารางที่ 5.6 ผลลัพธ์การทดสอบส่วนย่อยการสร้างต้นไม่แจกแจงทวิภาค 120
ตารางที่ 5.7 ผลลัพธ์การทดสอบระดับหน่วยการสร้างกรณีทดสอบด้วยหลักการณ์ครอบคลุมประพจน์ 121
ตารางที่ 5.8 ผลลัพธ์การทดสอบระดับหน่วยของการหาเส้นทางการทดสอบแบบไพร์ม 122
ตารางที่ 5.9 ผลลัพธ์การทดสอบระดับหน่วยการคำนวณหาอินเวิร์สการคูณของเมทริกซ์ 126
ตารางที่ 5.10 ตัวอย่างผลการทดสอบแบบบูรณาการณ์ 127
ตารางที่ 5.11 ตัวอย่างผลลัพธ์การทดสอบระบบ 128
ตารางที่ ก.1 ตัวอย่างการเปลี่ยนสถานะของแผนภาพสเตทแมชชีนองค์ประกอบการลงทะเบียนหนังสือ 133
ตารางที่ ก.2 ตัวอย่างการเปลี่ยนสถานะของแผนภาพสเตทแมชชีนองค์ประกอบการลงทะเบียนหนังสือ 134
ตารางที่ ก. 3 ตัวอย่างการเปลี่ยนสถานะของแผนภาพสเตทแมชชีนองค์ประกอบการลงทะเบียนหนังสือ 135
ตารางที่ ก.4 ตัวอย่างข้อกำหนดรายละเอียดประพจน์ที่ทำการขยายตัวกระตุ้นเหตุการณ์ 136
ตารางที่ ก.5 ตัวอย่างกรณีทดสอบขององค์ประกอบการลงทะเบียนหนังสือ 137
ตารางที่ ก. 6 ตัวอย่างกรณีทดสอบขององค์ประกอบการลงทะเบียนหนังสือ 138
ตารางที่ ข. 1 คำอธิบายยูสเคสการสร้างโครงการที่นำมาประเมิน 139
ตารางที่ ข. 2 คำอธิบายยูสเคสการตั้งค่าคะแนนระดับความรุนแรงตามร้อยละผลกระทบ 140
ตารางที่ ข. 3 คำอธิบายยูสเคสการนำเข้าข้อมูลลำดับเหตุการณ์ 141
ตารางที่ ข. 4 คำอธิบายยูสเคสการคำนวณความเสี่ยงระหว่างองค์ประกอบ 142
ตารางที่ ข. 5 คำอธิบายยูสเคสการตั้งค่าระดับผลกระทบจากคุณสมบัติประจำตาราง 143
ตารางที่ ข. 6 คำอธิบายยูสเคสการคำนวณระดับความรุนแรง 144

หน้า

ตารางที่ ข. 7 คำอธิบายยูสเคสการนำเข้าข้อมูลแผนภาพสเตทแมชชีนขององค์ประกอบ 145
ตารางที่ ข. 8 คำอธิบายยูสเคสการคำนวณความเสี่ยงระดับองค์ประกอบ 146
ตารางที่ ข. 9 คำอธิบายยูสเคสการรายงานผลการประเมินความเสี่ยง 147
ตารางที่ ข. 10 คำอธิบายยูสเคสการคำนวณความเสี่ยงในลำดับเหตุการณ์ 148

สารบัญรูป

หน้า

รูปที่ 2.1 โครงสร้างของกระบวนการบริหารความเสี่ยง 8
รูปที่ 2.2 ตัวอย่างแผนภาพสเตทแมชชีนแบบ External transition 14
รูปที่ 2.3 ตัวอย่างแผนภาพสเตทแมชชีนที่ใช้สำหรับสร้างเส้นทางการทดสอบ 15
รูปที่ 2.4 DTMC ของแบบจำลองพฤติกรรมการทำงานสำหรับลำดับเหตุการณ์ 22
รูปที่ 2.5 กราฟแสดงค่าปัจจัยความเสี่ยงตามระดับความรุนแรงของแต่ละลำดับเหตุการณ์ในระบบ 24
รูปที่ 2.6 ต้นไม้แจกแจงนิพจน์ $A \vee B \wedge C$.... 26
รูปที่ 2.7 กระบวนการสร้างกรณีทดสอบแบบครอบคลุมประพจน์เมื่อกำหนดให้ B และ C เป็นประโยคของการทดสอบ (เรียงจากแถวบนลงล่างตามลำดับ).... 27
รูปที่ 2.8 แผนภาพสเตทแมชชีนตัวอย่าง 28
รูปที่ 3.1 ขั้นตอนวิธีวิจัยการประเมินความเสี่ยงฯ 36
รูปที่ 3.2 ภาพรวมขั้นตอนการประเมินความเสี่ยงฯ 41
รูปที่ 3.3 กิจกรรมการกำหนดขอบเขตและการออกแบบการทำงานของซอฟต์แวร์ 42
รูปที่ 3.4 กิจกรรมการประเมินความเสี่ยงระดับองค์ประกอบ 43
รูปที่ 3.5 กิจกรรมการประเมินความเสี่ยงในลำดับเหตุการณ์.... 44
รูปที่ 3.6 อัลกอริทึมที่ใช้ในการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน 45
รูปที่ 3.7 ข้อมูลบางส่วนของเอกสารเค้าร่างฐานข้อมูลที่สร้างขึ้นด้วยภาษาเอสคิวแอล 47
รูปที่ 3.8 แผนภาพแพคเกจของคลาสที่ใช้ในการออกแบบซอฟต์แวร์.... 48
รูปที่ 3.9 แผนภาพลำดับที่ใช้ในการออกแบบซอฟต์แวร์.... 49
รูปที่ 3.10 แผนภาพสเตทแมชชีนที่ใช้ในการออกแบบองค์ประกอบการเสนอลงนาม 51
รูปที่ 4.1 ฐานข้อมูลซอฟต์แวร์ระบบสารบรรณอิเล็กทรอนิกส์.... 61
รูปที่ 4.2 แพคเกจของคลาสที่เกิดขึ้นในลำดับเหตุการณ์ของซอฟต์แวร์ระบบสารบรรณ.... 63
รูปที่ 4.3 แผนภาพลำดับของลำดับเหตุการณ์การรับส่งหนังสือภายในหน่วยงาน 70
รูปที่ 4.4 สถานะการดำเนินการกับข้อมูลฯขององค์ประกอบการลงทะเบียนหนังสือ 71
รูปที่ 4.5 สถานะการดำเนินการกับข้อมูลฯขององค์ประกอบการเสนอลงนาม.... 80
รูปที่ 4.6 สถานะการดำเนินการกับข้อมูลฯขององค์ประกอบการส่งหนังสือถึงผู้รับ 85
รูปที่ 4.7 กราฟควบคุมการทำงานของลำดับเหตุการณ์การรับส่งหนังสือ 98
รูปที่ 4.8 แบบจำลองความเสี่ยงของลำดับเหตุการณ์การรับส่งหนังสือ 99
รูปที่ 5.1 ความสามารถในการทำงานของระบบ SRR 103

หน้า

รูปที่ 5.2 โครงสร้างการทำงานของเครื่องมือการประเมินความเสี่ยงฯ 108
รูปที่ 5.3 โครงสร้างที่ใช้จัดเก็บข้อมูลชองเครื่องมือสำหรับประเมินความเสี่ยง 110
รูปที่ 5.4 ส่วนการนำเข้าข้อมูลโครงการที่นำมาประเมินความเสี่ยง 111
รูปที่ 5.5 ส่วนการตั้งค่าระดับผลกระทบของคุณลักษณะประจำตาราง 112
รูปที่ 5.6 ส่วนการนำเข้าข้อมูลองค์ประกอบ 113
รูปที่ 5.7 ส่วนการรายงานผลลัพธ์การคำนวณความเสี่ยงระดับองค์ประกอบ 115
รูปที่ 5.8 ส่วนการรายงานความเสี่ยงของลำดับเหตุการณ์ 116
รูปที่ 5.9 ส่วนการรายงานความเสี่ยงของโครงการ 117
รูปที่ 5.10 ต้นไม้แจกแจงแบบทวิภาคของนิพจน์ ~A & _A & ~B | ~C | ~D 120

บทที่ 1
บทนำ

ในบทนี้เป็นการถึงที่มาและความสำคัญของปัญหาและแนวทางการแก้ไข ที่แสดงถึงขั้นตอนวิธีที่ทำให้ได้มาซึ่งผลลัพธ์ที่ต้องการ ตลอดจนประโยชน์ของการนำเอาผลลัพธ์ที่ได้จากการวิจัยไปใช้ แสดงรายละเอียด ดังนี้

1.1. ที่มาและความสำคัญของปัญหา

ปัจจุบันซอฟต์แวร์เข้ามามีบทบาทในการบริหารจัดการข้อมูลเพื่อให้ได้เป็นสารสนเทศ (Information) ที่ตรงตามความต้องการของผู้ใช้ และคุณสมบัติหนึ่งที่ขาดไม่ได้สำหรับสารสนเทศที่เกิดขึ้นคือ “ความถูกต้อง”ซึ่งความถูกต้องของสารสนเทศนี้ ได้มาจากข้อมูลที่ปราศจากความผิดพลาดและมีความสอดคล้องกัน และการดำเนินการเพื่อให้ได้มาซึ่งความถูกต้องของข้อมูลนั้นสามารถดำเนินการได้หลายวิธี อาทิเช่น การทดสอบซอฟต์แวร์ในขั้นตอนการพัฒนารหัสโปรแกรม เป็นต้น แต่วิธีการดังกล่าวนำมาซึ่งต้นทุนการพัฒนาซอฟต์แวร์ที่มีมูลค่าสูงหรืออาจมีต้นทุนมากกว่าที่ประมาณการไว้ ดังนั้นจึงมีความจำเป็นในการใช้กระบวนการหรือขั้นตอนวิธีที่จะทำให้ล่วงรู้ถึงข้อผิดพลาดกับข้อมูลที่เกิดขึ้นจากซอฟต์แวร์ในขั้นตอนการออกแบบ และหนึ่งในวิธีดังกล่าวคือ การประเมินความเสี่ยง (Risk assessment) ซึ่งเป็นกิจกรรมที่ทำให้ทราบถึงปัจจัยความเสี่ยง ระดับความเสียหายต่อข้อมูล และรวมไปถึงกระบวนการวิธีในการจัดการกับความเสี่ยง ซึ่งผลลัพธ์จากการประเมินความเสี่ยงนี้ทำให้การพัฒนาซอฟต์แวร์ในแต่ละขั้นตอนสามารถนำเอาปัจจัยความเสี่ยงและระดับผลกระทบผนวกรวมเข้ากับกิจกรรมในแต่ละขั้นตอนการพัฒนาซอฟต์แวร์ได้ ซึ่งเป็นผลดีต่อการบริหารโครงการพัฒนาซอฟต์แวร์โดยทั่วไป

ปัญหาความผิดพลาดของข้อมูลที่จัดเก็บลงฐานข้อมูลเกิดได้จากหลายสาเหตุ และหนึ่งในสาเหตุของความผิดพลาดก็คือ การจัดเก็บข้อมูลที่ไม่เป็นไปตามเงื่อนไขเชิงเค้าร่างของฐานข้อมูลเชิงสัมพันธ์ ซึ่งราเมซ เอลมาซี (Ramez Elmasri) ได้แบ่งประเภทเงื่อนไขของฐานข้อมูลออกเป็นสามประเภทอันประกอบด้วย Inherent model-based constraints, Schema-based constraints และ Semantic constraints ซึ่งงานวิจัยฉบับนี้ได้นำเอาเงื่อนไขเชิงเค้าร่าง (Schema-based constraints) มาเป็นแกนหลักในการวิจัยเท่านั้น เนื่องจากเงื่อนไขดังกล่าวสามารถแสดงได้ด้วยภาษาที่ใช้ในการนิยามข้อมูล (Data Definition Language : DDL) ที่เกิดขึ้นในระหว่างการออกแบบฐานข้อมูล และเงื่อนไขดังกล่าวสามารถแบ่งออกเป็นสองหมวดหมู่คือ เงื่อนไขของความสัมพันธ์เชิงเดี่ยว (Single relation constraints) และเงื่อนไขของความสัมพันธ์เชิงกลุ่ม (Many relation constraint) โดยความสัมพันธ์เชิงเดี่ยวประกอบไปด้วยเงื่อนไขย่อยสามเงื่อนไขคือ เงื่อนไขประเภทและขอบเขต (Domain constraints) เงื่อนไขของคีย์หลัก (Key constraints) เงื่อนไขของค่าว่าง (Constraints on NULLs) และเงื่อนไขของความสัมพันธ์เชิงกลุ่มประกอบด้วยสองเงื่อนไขย่อยคือ

เงื่อนไขของคีย์หลักที่เกิดความสัมพันธ์ขึ้นต้องไม่เป็นค่าว่าง (Entity integrity constraints) และเงื่อนไขความสอดคล้อง (Referential integrity constraints) [3] เงื่อนไขที่ถูกกำหนดขึ้นเหล่านี้จะเป็นข้อกำหนดให้ข้อมูลที่เกิดขึ้นจากซอฟต์แวร์มีความถูกต้อง แม่นยำและสอดคล้องกัน ซึ่งเป็นไปตามคุณสมบัติอันพึงประสงค์ของซอฟต์แวร์ที่มีคุณภาพ สำหรับตัวอย่างของปัญหาที่เกี่ยวข้องกับเงื่อนไขเชิงเค้าร่างทั้งสี่ประเภท คือ

1) คุณลักษณะประจำที่ใช้เพื่อเก็บวงเงินเมื่อมีการสั่งซื้อสินค้า ซึ่งตามลักษณะทางธุรกิจแล้วคุณลักษณะประจำนี้ไม่สามารถกำหนดให้เป็นค่าว่างได้ (Entity constraint) แต่เค้าร่างฐานข้อมูลจำเป็นต้องกำหนดให้คุณลักษณะประจำตารางนี้มีค่าเป็นค่าว่างได้ (NULL) ทั้งนี้เพื่อการสร้างรายการการสั่งซื้อสินค้าในฐานข้อมูลไว้

2) ความยาวของข้อมูล หากเค้าร่างฐานข้อมูลกำหนดให้คุณลักษณะประจำมีชนิดเป็นดับเบิ้ล (Double) แต่ไม่ได้กำหนดความยาวที่จำไม่ทำให้โปรแกรมบริหารจัดการฐานข้อมูล ปรับเปลี่ยนรูปแบบการเก็บข้อมูล ก็จะทำให้โปรแกรมบริหารจัดการฐานข้อมูล จับเก็บข้อมูลในลักษณะของเลขยกกำลัง เช่น 12345678912345678909876543212399 9 แต่ข้อมูลที่ปรากฏในฐานข้อมูลเป็น 1.2345678912345678e32 ซึ่งในส่วนนี้จะทำให้การรายงานผลหรือการนำไปคำนวณต่อที่ปราศจากการปรับเปลี่ยนรูปแบบเกิดข้อผิดพลาดขึ้นได้

3) คุณสมบัติประจำตารางบางรายการต้องมีคุณสมบัติไม่ซ้ำกับข้อมูลใด ๆ แต่ไม่ได้ถูกกำหนดเป็นคีย์หลัก (Primary key) เช่น ชื่อผู้ใช้, รหัสพนักงาน เป็นต้น และคุณลักษณะประจำในลักษณะนี้เป็นอีกลักษณะหนึ่งที่อยู่นอกเหนือการควบคุมของโปรแกรมบริหารจัดการฐานข้อมูล ที่จะควบคุมให้ไม่เกิดการซ้ำกันของข้อมูล

4) โปรแกรมบริหารจัดการฐานข้อมูลบางประเภทสามารถยกเลิกความสามารถของฐานข้อมูลที่ทำหน้าที่ตรวจสอบความสอดคล้องของข้อมูลได้ อาทิเช่น MySQL MyISAM และเมื่อยกเลิกความสามารถในส่วนนี้แล้ว โอกาสที่จะนำไปสู่การอ้างอิงข้อมูลที่ผิดพลาดย่อมเกิดขึ้นได้

ดังที่กล่าวไว้ในข้างต้นว่า หากผู้พัฒนาซอฟต์แวร์สามารถทำนายถึงข้อผิดพลาดที่อาจเกิดขึ้นได้ในระหว่างการออกแบบซอฟต์แวร์จะทำให้ส่งผลดีนั้น ผู้วิจัยจึงพิจารณาแบบจำลองยูเอ็มแอล (UML modeling) ที่วัตถุประสงค์ในการนำไปใช้กับวิธีการพัฒนาซอฟต์แวร์เชิงวัตถุ ซึ่งเป็นที่แพร่หลายในปัจจุบัน และได้ข้อสรุปว่า แผนภาพสเตทแมชชีนแบบเช้จอีเว้นท์ (Change event state machine diagram) มีความเหมาะสมกับการประยุกต์ใช้ในขั้นตอนการประเมินความเสี่ยงของการเกิดข้อผิดพลาดกับข้อมูลที่ไม่เป็นไปตามเงื่อนไขเชิงเค้าร่างฯ ด้วยคุณสมบัติการแสดงถึงสถานะของสิ่งที่พิจารณาที่ขึ้นกับการกระทำและเงื่อนไขของการกระทำ

งานวิจัยฉบับนี้จึงนำเสนอขั้นตอนวิธีสำหรับการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ที่ไม่เป็นไปตามเงื่อนไขเชิงเค้าร่างด้วยการประเมินจากแผนภาพสเตทแมชชีน โดยระบุให้ความเสี่ยง

เกิดขึ้นภายในลำดับเหตุการณ์ (Scenario) ภายในองค์ประกอบ (Component) และความเสี่ยงที่เกิดขึ้นจากการเชื่อมโยงระหว่างองค์ประกอบ (Connector) เมื่อสามารถระบุปัจจัยความเสี่ยงได้แล้ว ลำดับต่อไปเป็นการประเมินค่าความเสี่ยงกับปัจจัยที่ได้ระบุไว้ในข้างต้น กล่าวคือ การประเมินความเสี่ยงในระดับองค์ประกอบ ซึ่งเป็นการคำนวณหาค่าความน่าจะเป็นของการเกิดข้อผิดพลาด ด้วยการประยุกต์ใช้วิธีการสร้างกรณีทดสอบด้วยหลักการณ์ครอบคลุมประพจน์ (Full predicate coverage) และการคำนวณหาเส้นทางการทดสอบที่เป็นไปได้แบบไพร์ม (Prime path testing) จากนั้นคำนวณหาระดับผลกระทบ ซึ่งในที่นี้จะเรียกว่าระดับความรุนแรง (Severity) เมื่อการทำงานในระดับองค์ประกอบเกิดข้อผิดพลาด ด้วยวิธีการประเมินระดับผลกระทบของคุณลักษณะประจำตารางจากฐานตามความเห็นของผู้ประเมิน ผลลัพธ์จากการคำนวณความน่าจะเป็น และระดับความรุนแรงนี้ จะทำให้ได้มาซึ่งค่าความเสี่ยงขององค์ประกอบที่ต้องการ จากนั้นพิจารณาค่าความเสี่ยงของการเชื่อมโยงระหว่างองค์ประกอบด้วยการคำนวณค่าความน่าจะเป็นที่การทำงานร่วมกันขององค์ประกอบเกิดข้อผิดพลาด และระดับความรุนแรงเมื่อการเชื่อมโยงองค์ประกอบเกิดข้อผิดพลาด ซึ่งคำนวณความน่าจะเป็นของการเกิดข้อผิดพลาดได้จากอัตรส่วนระหว่างจำนวนข้อความที่ส่งออกจากโหนดผู้ส่งไปยังโหนดผู้รับกับจำนวนข้อความทั้งหมดที่ปรากฏอยู่ในแผนภาพลำดับ และการคำนวณระดับความรุนแรงสามารถทำได้วิธีเดียวกับการหาระดับความรุนแรงขององค์ประกอบ

ขั้นตอนถัดไปเป็นการประเมินความเสี่ยงของลำดับเหตุการณ์ ด้วยการสร้างแบบจำลองความเสี่ยงของการทำงานที่ผิดพลาดของลำดับเหตุการณ์ในซอฟต์แวร์ ที่ได้จากการประยุกต์ใช้แบบจำลองของมาร์คอฟที่มีสมมุติคือ ผลลัพธ์ที่เกิดขึ้นจากเหตุการณ์ภายหลังเป็นอิสระจากเหตุการณ์ก่อนหน้า ซึ่งผลลัพธ์ที่ได้สามารถนำไปจัดอันดับความเสี่ยงตามความสำคัญและสอดคล้องกับค่าความเสี่ยงที่สามารถคำนวณได้

นอกจากนี้ผู้วิจัยได้พัฒนาเครื่องมือสำหรับประเมินความเสี่ยงตามขั้นตอนวิธีที่นำเสนอในรูปแบบเว็บเบสแอปพลิเคชัน เพื่ออำนวยความสะดวกแก่ผู้ประเมิน และสำหรับรายละเอียดของขั้นตอนวิธีการประเมินความเสี่ยง และวิธีการได้มาซึ่งขั้นตอนวิธีประเมินความเสี่ยงนั้น สามารถศึกษาได้จากส่วนประกอบต่างๆของวิทยานิพนธ์ฉบับนี้ อันประกอบด้วย บทที่หนึ่งกล่าวถึง ที่มาความสำคัญ วัตถุประสงค์ของงานวิจัย เป็นต้น บทที่สองเป็นการศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง บทที่สามเป็นการแสดงขั้นตอนวิธีการประเมินความเสี่ยง บทที่สี่เป็นการประยุกต์ใช้ขั้นตอนวิธีกับกรณีศึกษา บทที่ห้าเป็นการแสดงรายละเอียดและความเสามารถของเครื่องมือการประเมินความเสี่ยง และบทที่หกเป็นการสรุปผลงานวิจัย ข้อจำกัด และการเสนอแนวทางในการพัฒนาต่อ แสดงได้ดังรายละเอียดต่อไปนี้

1.2. วัตถุประสงค์

เพื่อนำเสนอขั้นตอนวิธีการประเมินความเสี่ยงอันเนื่องจากการเปลี่ยนแปลงข้อมูลใดๆ ภายในฐานข้อมูล ด้วยวิธีการประเมินความเสี่ยงของลำดับเหตุการณ์ องค์ประกอบ และการเชื่อมโยงระหว่างองค์ประกอบ ผ่านการคำนวณหาจำนวนกรณีทดสอบสำหรับการเปลี่ยนแปลงของสถานะภายใต้เงื่อนไขแบบเค้าร่าง การหาเส้นทางการทดสอบที่เป็นไปได้ รวมไปถึงการสร้างเมทริกซ์ความน่าจะเป็นในการทำงานที่ล้มเหลวของซอฟต์แวร์ตามระดับความรุนแรงของผลกระทบด้วยการประยุกต์แบบจำลองของมาร์คอฟ ซึ่งจะทำให้ผู้ประเมินได้ทราบถึงอันดับของลำดับเหตุการณ์ที่ถูกเรียงตามค่าความเสี่ยงที่ได้จากการประเมินความเสี่ยงในแต่ละลำดับเหตุการณ์ของซอฟต์แวร์

1.3. ขอบเขตงานวิจัย

1) การประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์เป็นการประเมินตามเงื่อนไขเชิงเค้าร่างซึ่งประกอบด้วย เงื่อนไขประเภทและขอบเขต เงื่อนไขของคีย์หลัก เงื่อนไขของค่าว่างและเงื่อนไขความสอดคล้องของข้อมูล
2) แผนภาพสเตทแมชชีนที่นำมาใช้ประเมินความเสี่ยงต้องเป็นแผนภาพสเตทแมชชีนแบบเช้จอีเวนท์ที่ประกอบด้วยเหตุการณ์กระตุ้นและเงื่อนไขของเหตุการณ์กระตุ้น
3) หลักการหากรณีทดสอบที่นำมาใช้สำหรับการประเมินความเสี่ยงคือหลักการครอบคลุมประพจน์
4) หลักการที่นำมาคำนวณหาเส้นทางการทดสอบที่เป็นไปได้คือหลักการสร้างกรณีทดสอบแบบไพร์ม
5) พัฒนาเครื่องมือที่สนับสนุนการประเมินความเสี่ยงของลำดับเหตุการณ์ที่ส่งผลให้เกิดการเปลี่ยนแปลงข้อมูลในฐานข้อมูล
6) เครื่องมือที่พัฒนารองรับระบบที่มีการออกแบบโดยใช้แผนภาพสเตทแมชชีนขององค์ประกอบที่มีจำนวนไม่ต่ำกว่า 10 สถานะ เมื่อนับรวมในทุกลำดับเหตุการณ์ที่พิจารณา
7) เครื่องมือที่พัฒนารองรับการนำเข้าข้อมูลแผนภาพลำดับและแผนภาพสเตทแมชชีนด้วยเอกสารประเภทเอ็กซ์เอ็มแอล (XML) เพื่อใช้ในการสร้างแบบจำลองพฤติกรรมการทำงานของซอฟต์แวร์และการคำนวณหาจำนวณกรณีทดสอบ
8) เครื่องมือที่พัฒนารองรับการกำหนดระดับผลกระทบของคุณสมบัติประจำตารางของฐานข้อมูลเชิงสัมพันธ์ผ่านการวิเคราะห์เอกสารเอสคิวแอล (SQL)

1.4. ขั้นตอนและวิธีดำเนินงานวิจัย

1) ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องกับเงื่อนไขเชิงเค้าร่างของฐานข้อมูลเชิงสัมพันธ์
2) ศึกษาการออกแบบข้อกำหนดความต้องการด้วยแผนภาพสเตทแมชชีน
3) ศึกษาทฤษฎีด้านการบริหารความเสี่ยงและงานวิจัยด้านการประเมินความเสี่ยงที่อาจเกิดขึ้นในขั้นตอนการออกแบบซอฟต์แวร์
4) ศึกษากฎการขยายเหตุการณ์ที่มีการกระตุ้น

5) ศึกษาการคำนวณหาค่าความน่าจะเป็นจากวิธีการสร้างกรณีทดสอบจากแผนภาพสเตทแมชชีน
6) ศึกษาการคำนวณหาเส้นทางการทดสอบที่เป็นไปได้จากแผนภาพสเตทแมชชีน
7) ศึกษากระบวนการวิเคราะห์ภัยคุกคาม
8) สร้างขั้นตอนวิธีในการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน
9) วิเคราะห์และออกแบบหน้าที่การทำงานของเครื่องมือ เพื่อสนับสนุนการประเมินความเสี่ยงฯ
10) พัฒนาเครื่องมือในการประเมินความเสี่ยงฯ
11) ทดสอบและตรวจสอบคุณภาพของเครื่องมือ
12) จัดทำบทความทางวิชาการและนำเสนอบทความ
13) สรุปผลการวิจัย และจัดทำรายงานวิทยานิพนธ์

1.5. ประโยชน์ที่คาดว่าจะได้รับ

1) ทราบถึงความเสี่ยงที่เกิดขึ้นในระดับองค์ประกอบและลำดับของเหตุการณ์เพื่อให้ทราบถึงโอกาสในการเกิดความสูญเสียจากการเปลี่ยนแปลงฐานข้อมูลตามเงื่อนไขเชิงเค้าร่าง
2) ผู้พัฒนาซอฟต์แวร์ทราบถึงโอกาสในการเกิดความสูญเสียในระหว่างขั้นตอนการออกแบบซอฟต์แวร์
3) ผู้บริหารโครงการสามารถนำเครื่องมือไปประยุกต์ใช้ในการบริหารจัดการกระบวนการพัฒนาซอฟต์แวร์ได้

1.6. บทความวิชาการที่ได้รับการตีพิมพ์

ผู้วิจัยได้เผยแพร่ผลงานวิชาการร่วมกับคณะผู้วิจัย ซึ่งเป็นบทความวิชาการระดับนานาชาติ ได้แก่

1) บทความวิชาการเรื่อง “Risk Assessment for Relational Database Schema-based Constraint Using State Machine Diagram” ซึ่งได้รับการคัดเลือกเพื่อนำเสนอและตีพิมพ์ในวารสารวิชาการ “The International MultiConference of Engineers and Computer Scientists 2013” ระหว่างวันที่13 – 15 มีนาคม 2556 ณ ประเทศฮ่องกง

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้เป็นการนำเสนอทฤษฎีและงานวิจัยที่นำมาประยุกต์ใช้กับขั้นตอนวิธีการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่าง มีรายละเอียดดังนี้

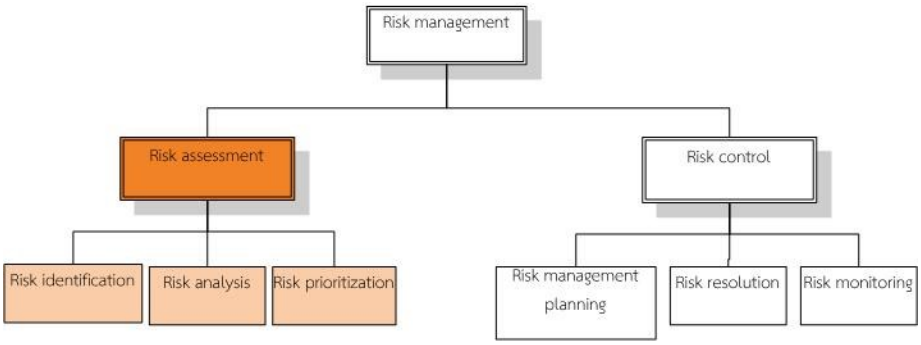
2.1. ทฤษฎีและคำศัพท์ที่เกี่ยวข้อง

ตารางที่ 2.1 คำศัพท์ที่เกี่ยวข้อง

คำศัพท์	ความหมาย
ซีนนาริโอ (Scenario)	ลำดับเหตุการณ์ของซอฟต์แวร์ที่เกิดขึ้นซึ่งทำให้เกิดการเปลี่ยนแปลงข้อมูลภายในฐานข้อมูลเชิงสัมพันธ์
คอมโพเนนท์ (Component)	องค์ประกอบของลำดับเหตุการณ์ที่มีปฏิสัมพันธ์ระหว่างกัน ซึ่งในลำดับเหตุการณ์หนึ่งๆจะประกอบไปด้วยคอมโพเนนท์อย่างน้อยหนึ่งคอมโพเนนท์
ทรานสิชัน (Transition)	การเปลี่ยนแปลงที่เกิดขึ้นแล้วทำให้สถานะของวัตถุเปลี่ยนแปลง

2.1.1. ความเสี่ยง (Risk)

หากกล่าวถึงความเสี่ยง [1] ในมุมมองของการบริหารความเสี่ยง ซึ่งประกอบไปด้วยกระบวนการที่สามารถทำให้การบริหารความเสี่ยงสำเร็จลุล่วงไปได้ แบ่งออกเป็นสองกระบวนการหลักคือ การประเมินความเสี่ยง (Risk assessment) และการควบคุมความเสี่ยง (Risk control) ซึ่งในงานวิจัยฉบับนี้ครอบคลุมในส่วนของการประเมินความเสี่ยงเท่านั้น สามารถแสดงภาพรวมของการบริหารความเสี่ยงได้ดังรูปที่ 2.1



รูปที่ 2.1 โครงสร้างของกระบวนการบริหารความเสี่ยง

ความเสี่ยงคือความน่าจะเป็นของความสูญเสีย หรือความน่าจะเป็นที่ทำให้เกิดความเสียหาย ซึ่งสามารถอธิบายในมุมมองด้านการบริหารความเสี่ยงคือ การเปิดเผยความเสี่ยง หรือบางครั้งอาจเรียกว่า “ผลกระทบความเสี่ยง” หรือ “ปัจจัยความเสี่ยง” และความเสี่ยงที่กล่าวถึงสามารถแสดงได้ดังแบบจำลอง

$$RF = P(UO) * L(UO)$$

จากแบบจำลองในข้างต้นพบว่าการนิยามความเสี่ยงดังกล่าวมีความสอดคล้องกับมาตรฐานที่องค์กรนาซา (NASA) ได้กำหนดขึ้นใจความว่า ความเสี่ยงคือ สิ่งที่เกิดจากความน่าจะเป็นที่ซอฟต์แวร์หรือโครงการได้ผลลัพธ์ที่ไม่พึงปรารถนาในหลายมิติและขึ้นกับระดับความเสียหายเมื่อเกิดผลลัพธ์ที่ไม่พึงปรารถนาขึ้น [2] มีรายละเอียดดังนี้

RF คือ ปัจจัยความเสี่ยงที่มีผลทำให้เกิดผลลัพธ์ที่ไม่พึงปรารถนา

$P(UO)$ คือ ความน่าจะเป็นของการเกิดผลลัพธ์ที่ไม่พึงปรารถนาที่ถูกพิจารณาในแต่ละมิติ

$L(UO)$ คือ ความเสียหายที่เกิดขึ้นเมื่อเกิดผลลัพธ์ที่ไม่พึงปรารถนากับผู้มีส่วนร่วมในโครงการหรือซอฟต์แวร์นั้นๆ

2.1.1.1.การประเมินความเสี่ยง (Risk assessment) [1]

สำหรับกระบวนการนี้เป็นขั้นตอนที่ทำให้ทราบถึงภัยที่เกิดขึ้นจากความเสี่ยงที่แฝงอยู่ในซอฟต์แวร์หรือโครงการที่ ต้องการประเมิน ประกอบด้วยขั้นตอนย่อย 3 ขั้นตอนคือ

1. การระบุความเสี่ยง (Risk identification) เป็นการสร้างรายการของความเสี่ยงที่แฝงอยู่ซึ่งอาจส่งผลให้โครงการไม่สำเร็จลุล่วงได้ โดยการระบุความเสี่ยงนี้อาจใช้เทคนิคของการสร้างรายการตรวจสอบ (Checklist) การตรวจสอบ การตัดสินใจ (Examination of decision drivers) หรือการเปรียบเทียบกับโครงการที่ผ่านมา

2. การวิเคราะห์ความเสี่ยง (Risk analysis) เป็นการประเมินความน่าจะเป็นที่จะเกิดความเสียหายและระดับผลกระทบที่เกิดขึ้นสำหรับความเสี่ยงที่ระบุไว้ในแต่ละรายการความเสี่ยง และสำหรับการประเมินนี้ยังครอบคลุมไปถึงปฏิสัมพันธ์ของแต่ละรายการความเสี่ยงด้วย โดยปกติแล้วเทคนิคที่ใช้ในการประเมินความเสี่ยงคือ แบบจำลองการคำนวณต้นทุน การวิเคราะห์การตัดสินใจด้วยค่าเชิงสถิติ และการวิเคราะห์ปัจจัยด้านคุณภาพ (เช่น ความน่าเชื่อถือ ความพร้อมใช้งาน และความมั่นคง)

3. การจัดลำดับความเสี่ยง (Risk prioritization) เป็นการนำเอารายการความเสี่ยงที่ผ่านการวิเคราะห์มาจัดเรียงลำดับ และมักนำเอาปัจจัยด้านการลดทอนความเสี่ยงมาร่วมพิจารณาในการจัดเรียงลำดับความเสี่ยงด้วย

2.1.1.2.ภัย (Hazard)

ภัยเป็นผลลัพธ์ของเหตุการณ์ที่แฝงด้วยความเสี่ยง ซึ่งตามมาตรฐานความปลอดภัยที่นาซาได้แบ่งระดับภัยตามความรุนแรงของผลกระทบสามารถแบ่งภัยคุกคามออกเป็น 4 ระดับ คือ

1. ระดับหายนะ (Catastrophic) เป็นระดับภัยคุกคามที่มีผลกระทบมากที่สุดคือ มีผลกระทบต่อชีวิตมนุษย์หรือทำให้ทุพพลภาพ ทำให้ระบบสูญเสียโดยสิ้นเชิง สูญเสียสาธารณูปโภคขั้นพื้นฐาน หรือการทำให้สภาพแวดล้อมถูกทำลายอย่างรุนแรง

2. ระดับวิกฤต (Critical) เป็นระดับเสียหายรุนแรงหรือเป็นอัมพาตชั่วคราว ส่วนหลักของระบบได้รับความเสียหาย

3. ระดับกลาง (Moderate) เป็นระดับที่เสียหายเล็กน้อย ส่วนย่อยของระบบได้รับความเสียหาย

4. ระดับต่ำ (Negligible) ไม่เกิดความเสียหายหรือเกิดเพียงเล็กน้อย บางส่วนของระบบมีการทำงานหนักแต่ไม่มีส่วนใดได้รับความเสียหาย [2]

2.1.2. เงื่อนไขเชิงเค้าร่างของฐานข้อมูล (Schema-based constraints)

เงื่อนไข (Constraints) ถือเป็นหนึ่งในคุณลักษณะของฐานข้อมูลเชิงสัมพันธ์ (Relational Database) ที่มีความสำคัญ ซึ่งตามทฤษฎีที่ได้กล่าวไว้ใน [3] เงื่อนไขของฐานข้อมูลเชิงสัมพันธ์มีทั้งหมด 3 ประเภทคือ เงื่อนไขที่ได้รับการถ่ายทอดจากแบบจำลองข้อมูล (implicit constraints) เงื่อนไขที่สามารถแสดงผ่านเค้าร่างของแบบจำลองข้อมูล (Schema-based constraints หรือ explicit constraints) และสุดท้ายคือเงื่อนไขที่ไม่สามารถแสดงผ่านเค้าร่างของแบบจำลองข้อมูลได้ (application-based หรือ semantic constraints) ซึ่งในส่วนนี้จะกล่าวถึงเงื่อนไขที่สามารถแสดงผ่านเค้าร่างของแบบจำลองข้อมูลเท่านั้น

เงื่อนเชิงเค้าร่างประกอบไปด้วยประเภทของเงื่อนไขย่อยทั้งสิ้น 4 เงื่อนไขคือ เงื่อนไขของโดเมน (Domain constraints) เงื่อนไขของค่าว่าง (Entity constraints) เงื่อนไขของคีย์หลัก (Key

constraints) และเงื่อนความสอดคล้องของข้อมูล (Referential integrity constraints) มีรายละเอียดของแต่ละเงื่อนไขดังต่อไปนี้

1. เงื่อนไขของโดเมน (Domain constraints)

เงื่อนไขของโดเมนถูกกำหนดไว้เพื่อให้ข้อมูลที่ถูกจัดเก็บลงฐานข้อมูลต้องมีชนิดเดียวกันกับประเภทของแอททริบิวต์ที่ได้กำหนดไว้ เช่น แอททริบิวต์สำหรับการเก็บที่อยู่ชื่อ address ถูกสร้างขึ้นด้วยชนิดของข้อมูลแบบสายอักขระ (String) ดังนั้นข้อมูลที่ต้องการจัดเก็บต้องอยู่ในรูปแบบของสายอักขระเท่านั้น

2. เงื่อนไขของค่าว่าง (Entity constraints)

เป็นเงื่อนไขที่กำหนดไว้เพื่อระบุความต้องการของข้อมูลสำหรับแอททริบิวต์ใดแอททริบิวต์หนึ่ง ซึ่งแอททริบิวต์เหล่านี้จะถูกกำหนดว่าไม่อนุญาตให้เป็นค่าว่าง (Not null value) ซึ่งหมายความว่าการเปลี่ยนแปลงฐานข้อมูลที่เกี่ยวข้องกับแอททริบิวต์ประเภทนี้จะต้องดำเนินการภายใต้การปรากฏของข้อมูลสำหรับแอททริบิวต์นั้นๆ เช่น หากกำหนดให้แอททริบิวต์สำหรับเก็บรหัสประจำตัวประชาชนมีชื่อว่า idcard มีข้อกำหนดเป็นแบบ Not null แล้ว ข้อมูลที่ปรากฏอยู่ในฐานข้อมูลในส่วนของรหัสประจำตัวประชาชนนั้นจะไม่มีโอกาสปรากฏค่าว่างขึ้น

3. เงื่อนไขของคีย์หลัก (Key constraints)

ด้วยข้อกำหนดของฐานข้อมูลเชิงสัมพันธ์ที่กำหนดว่า ข้อมูลระหว่างสองรายการ (Records) ใดๆจะต้องไม่เหมือนกันทั้งหมด ดังนั้นคีย์จึงเป็นส่วนที่จะทำให้ข้อมูลใดๆภายในฐานข้อมูลมีความแตกต่างกัน โดยปกติแล้วแต่ละตารางนั้นจะประกอบด้วย คีย์หลัก (Primary key) ซึ่งหมายถึงสิ่งที่เป็นตัวกำหนดความแตกต่างของข้อมูลในแต่ละรายการ และสำหรับคีย์อีกประเภทหนึ่งเรียกว่าคีย์สำรอง (Candidate key) ซึ่งมีคุณสมบัติที่นำมาใช้แทนคีย์หลักได้ จึงสรุปได้ว่าการเปลี่ยนแปลงใดๆที่เกี่ยวข้องกับการเปลี่ยนแปลงค่าของคีย์หลัก จะต้องคำนึงถึงโอกาสในการเกิดคีย์หลักซ้ำซ้อนกับรายการอื่นด้วย

4. เงื่อนไขความสอดคล้องของข้อมูล (Referential integrity constraints)

เงื่อนไขรูปแบบสุดท้ายที่กล่าวถึงคือเงื่อนไขความสอดคล้องของข้อมูล ซึ่งเป็นเงื่อนไขที่ถูกกำหนดขึ้นเพื่อรักษาความสัมพันธ์ที่เกิดขึ้นระหว่างตารางหรือภายในตารางเดียวกัน โดยปกติแล้วการอ้างอิงคีย์จากตารางหนึ่งไปยังอีกตารางหนึ่ง คีย์ที่ถูกอ้างถึงจะต้องปรากฏอยู่หรือไม่เป็นค่าว่าง ความสัมพันธ์นั้นจึงจะตรงตามข้อกำหนดของเงื่อนไข

สำหรับการนำเอาเงื่อนไขประเภทต่างๆที่กล่าวไว้ในข้างต้น สามารถแบ่งลักษณะการนำไปใช้ตามประเภทของตัวดำเนินการทางฐานข้อมูลเชิงสัมพันธ์ (Relational database operations) ได้ดังตารางที่ 2.2 [3]

ตารางที่ 2.2 การนำเงื่อนไขเชิงเค้าร่างเมื่อนำไปใช้ร่วมกับตัวดำเนินการทางฐานข้อมูล

Constraint/Operation	Insert	Update	Delete
Domain constraint	✓	✓	
Entity constraint	✓	✓	
Key constraint	✓	✓	
Reference constraint	✓	✓	✓

2.1.3. แผนภาพสเตทแมชชีน (State machine diagram)

แผนภาพสเตทแมชชีนเป็นแบบจำลองที่อยู่ในรูปแบบของกราฟสถานะและการเปลี่ยนแปลงสถานะที่นำมาใช้เพื่ออธิบายพฤติกรรมเชิงพลวัตรของระบบหรือวัตถุ ตลอดวัฏจักรชีวิต โดยแต่ละระบบหรือวัตถุจะตอบสนองต่อเหตุการณ์ (Event) และเมื่อวัตถุได้รับเหตุการณ์ที่มากระทำแล้ว การตอบสนองก็จะขึ้นกับสถานะของระบบ ณ ขณะนั้นด้วย

2.1.3.1.เหตุการณ์ (Event)

เป็นสิ่งที่เกิดขึ้นโดยมีตำแหน่งและเวลาในการเกิดชัดเจน เป็นเหตุการณ์ที่เกิดขึ้น ณ เวลาขณะใดขณะหนึ่งแทนที่จะเกิดขึ้นเป็นช่วงเวลา และเมื่อเหตุการณ์เกิดขึ้นแล้วจะส่งผลให้สถานะของวัตถุที่เกี่ยวข้องกับเหตุการณ์เปลี่ยนแปลงไป โดยเหตุการณ์ที่เกิดขึ้นอาจประกอบด้วยพารามิเตอร์ที่จะส่งผลต่อลักษณะของเหตุการณ์ สามารถแบ่งประเภทของเหตุการณ์ออกเป็น 4 ประเภทดังตารางต่อไปนี้

ตารางที่ 2.3 รายการประเภทของเหตุการณ์ในแผนภาพสเตทแมชชีน

ประเภทของเหตุการณ์	คำอธิบาย	สัญลักษณ์ที่ใช้
คอลอีเวนท์ (Call event)	เหตุการณ์ที่มีการรับการร้องขออย่างชัดเจนระหว่างวัตถุที่รอรับการตอบสนอง	op (a:T)
เช้จอีเวนท์ (Change event)	เหตุการณ์ที่มีการเปลี่ยนแปลงค่าในนิพจน์บูลลีน	**when** (exp)
ซิกเนลอีเวนท์ (Signal event)	เหตุการณ์ที่มีการสื่อสารแบบไม่ประสานเวลา (Asynchronous Communication) ระหว่างวัตถุ	sname (a:T)
ไทม์อีเวนท์ (Time event)	เหตุการณ์ที่เกิดขึ้น ณ เวลาขณะใดขณะหนึ่ง	**after** (time)

สำหรับงานวิจัยฉบับนี้ได้นำประเภทของเหตุการณ์เช้จอีเวนท์ประกอบการทำวิจัย ดังนั้นจะขอกล่าวถึงประเภทของเหตุการณ์ในประเภทนี้เท่านั้น

- เช้จอีเวนท์ (Change event)

เหตุการณ์ที่มีการเปลี่ยนแปลงในรูปแบบของนิพจน์บูลลีน ซึ่งเหตุการณ์นั้นจะเกิดขึ้นเมื่อเงื่อนไขที่กำหนดเป็นจริง ซึ่งการนำเหตุการณ์ประเภทนี้มาใช้มีทั้งข้อดีและข้อเสีย ข้อดีคือเหตุการณ์จะเกิดขึ้นได้ต่อเมื่อเงื่อนไขที่กำหนดเป็นจริงเท่านั้น และจุดนี้เป็นข้อได้เปรียบกว่าการใช้แผนภาพอื่นๆ ส่วนข้อเสียคือความคลุมเครือในการหาสาเหตุและผลลัพธ์ (Cause-and-effect) ระหว่างเหตุการณ์ที่ทำให้เปลี่ยนและสิ่งที่ถูกเปลี่ยนแปลงไป อนึ่งระหว่างเหตุการณ์ที่ทำให้เปลี่ยนแปลง (Change event) และเงื่อนไขที่ระบุไว้คู่กับเหตุการณ์ (Guard condition) มีความแตกต่างกันคือเงื่อนไขจะเป็นตัวประเมินเมื่อมีเหตุการณ์กระตุ้นเมื่อเกิดการเปลี่ยนแปลงและฝั่งรับมีการรับมือกับเหตุการณ์ที่เกิดขึ้น ถ้าผลลัพธ์ออกมาผิดการเปลี่ยนแปลงจะไม่สัมฤทธิ์ผลและเงื่อนไขจะไม่ถูกประเมินใหม่ ส่วนเหตุการณ์ที่ทำให้เปลี่ยนแปลงจะถูกประเมินอย่างต่อเนื่องตราบเท่าที่เหตุการณ์นี้จะมีผลลัพธ์เป็นจริง

2.1.3.2.สถานะ (State)

สถานะเป็นการอธิบายถึงช่วงเวลาใดๆตลอดวัฏจักรชีวิตของวัตถุ พิจารณาได้ใน 3 ลักษณะ คือ กลุ่มวัตถุที่ตอบสนองต่อสิ่งที่เกิดขึ้นแบบเดียวกัน ช่วงเวลาในระหว่างที่วัตถุรอคอยเหตุการณ์ที่จะเกิดขึ้น หรือช่วงเวลาที่วัตถุรอคอยในเหตุการณ์ที่กำลังเกิดขึ้นจนแล้วเสร็จ ในแผนภาพสเตทแมชชีนนี้แต่ละสถานะถูกเชื่อมต่อเข้าหากันด้วยเส้นการเปลี่ยนสถานะ (Transitions) และแม้ว่าเส้นเชื่อม

ดังกล่าวจะใช้ในการเชื่อมต่อระหว่างสองสถานะใดๆ การเปลี่ยนแปลงก็จะถูกควบคุมโดยสถานะฝั่งต้นทางเสมอ อนึ่งทุกวัตถุที่อยู่ในสถานะเดียวกันจะมีการตอบสนองต่อเหตุการณ์ที่เป็นตัวกระตุ้นในลักษณะเดียวกัน ซึ่งแน่นอนว่าวัตถุที่อยู่คนละสถานะกันย่อมมีการตอบสนองต่อเหตุการณ์กระตุ้นที่แตกต่าง แม้ว่าเหตุการณ์ที่มากระตุ้นนั้นจะเป็นเหตุการณ์เดียวกันก็ตาม เช่น เครื่องตอบรับอัตโนมัติมีตอบสนองเมื่อมีการกดปุ่มยกเลิกรูปแบบหนึ่งเมื่อเครื่องตอบรับฯอยู่ในสถานะกำลังทำงาน แต่จะมีการตอบสนองอีกแบบหนึ่งเมื่อเครื่องตอบรับฯอยู่ในสถานะรอคอย (Idle)

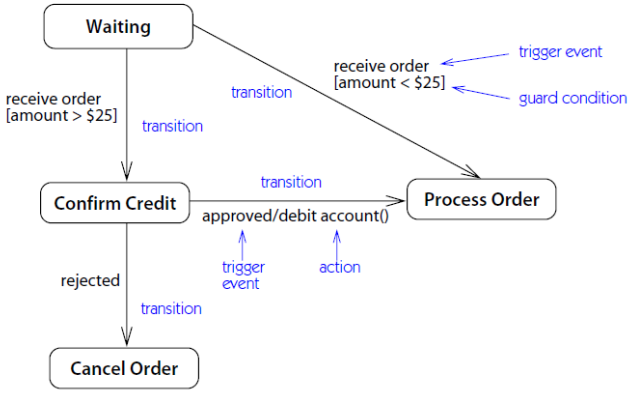
2.1.3.3.การเปลี่ยนแปลง (Transition)

การเปลี่ยนแปลงเป็นเส้นที่ชี้ออกจากสถานะที่เป็นฝ่ายกำหนดวัตถุที่อยู่ในสถานะที่จะเกิดเหตุการณ์นั้นขึ้น โดยทั่วไปแล้วการเปลี่ยนแปลงจะมีเหตุการณ์ที่เป็นตัวกระตุ้น เงื่อนไขที่ใช้ควบคุมการกระทำและสถานะเป้าหมาย โดยแบ่งประเภทของการเปลี่ยนแปลงออกเป็น 4 ประเภท ดังนี้

ตารางที่ 2.4 รายการประเภทของการเปลี่ยนแปลง

ประเภทของการเปลี่ยนแปลง	**คำอธิบาย**	**สัญลักษณ์ที่ใช้**
Entry action	เริ่มต้นทำงานเมื่อเข้าสู่สถานะ	entry/ activity
Exit action	เริ่มต้นทำงานเมื่อออกจากสถานะ	exit/ activity
External transition	การตอบสนองต่อเหตุการณ์ที่เป็นเหตุให้เปลี่ยนสถานะทั้งที่เปลี่ยนสู่สถานะอื่นและเข้าสู่สถานะเดิม	e(a:T)[guard]/activity
Internal transition	การตอบสนองต่อเหตุการณ์ที่เป็นเหตุให้การทำงานแต่ไม่เปลี่ยนสถานะ	e(a:T)[exp]/activity

จากคำอธิบายเรื่องแผนภาพสเตทแมชชีนในข้างต้นนั้น สามารถแสดงตัวอย่างแผนภาพที่จะนำไปใช้ในงานวิจัยฉบับนี้ดังรูปที่ 2 [8]



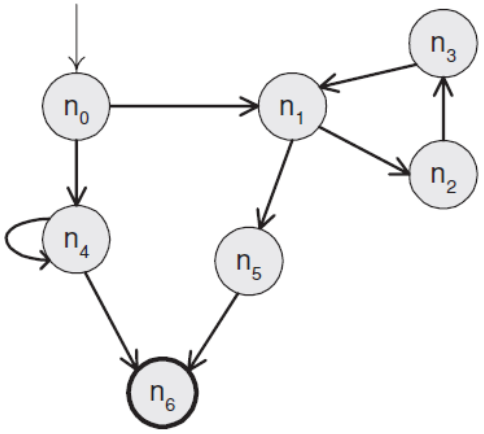
รูปที่ 2.2 ตัวอย่างแผนภาพสเตทแมชชีนแบบ External transition

2.1.4. การสร้างเส้นทางทดสอบแบบไพร์มพาธ (Prime path testing)

การหาเส้นทางการทำงานของซอฟต์แวร์ที่เป็นไปได้ (Software execution path) หรือที่เรียกว่า “เส้นทางการทดสอบ” (Test path) จากกราฟที่มีทิศทาง (Directed graphs) โดยเส้นทางที่สร้างขึ้นจะต้องเป็นเส้นทางที่เริ่มต้นจากโหนดเริ่มต้นและสิ้นสุดด้วยโหนดสิ้นสุด และกราฟที่นำมาพิจารณาต้องประกอบด้วย N N_0 และ N_f อย่างน้อยที่สุดอย่างละหนึ่งโหนดมีรายละเอียดดังนี้ [10]

- กลุ่มของ N หมายถึงจำนวนโหนดที่เป็นส่วนหนึ่งของกราฟ
- กลุ่มของ N_0 หมายถึงโหนดเริ่มต้น ซึ่ง $N_0 \subseteq N$
- กลุ่มของ N_f หมายถึงโหนดสิ้นสุด ซึ่ง $N_f \subseteq N$
- กลุ่มของ E หมายถึงเส้นระหว่างโหนด ซึ่ง E เป็นสับเซตของ $N \; x \; N$

สำหรับหลักเกณฑ์ที่ใช้เพื่อการหากรณีทดสอบจากแผนภาพในลักษณะของกราฟที่ใช้เพื่อนำเสนอตรรกะการทำงานของซอฟต์แวร์คือ Structural Coverage Criteria และหลักเกณฑ์นี้ยังประกอบไปด้วยวิธีการหลายวิธีซึ่งมีความเหมาะสมกับปัญหาในลักษณะที่แตกต่างกัน สำหรับงานวิจัยฉบับนี้ได้นำเอาวิธีการสร้างกรณีทดสอบด้วยวิธีที่เรียกว่า “หลักการครอบคลุมไพร์มพาธ” (Prime Path Coverage) ซึ่งเป็นวิธีที่เหมาะสมกับการหากรณีทดสอบจากแผนภาพสเตทแมชชีนซึ่งมีลักษณะเช่นเดียวกับ Finite State Machine (FSM) มีนิยามและขั้นตอนการทำงานที่ทำให้ได้มาซึ่งเส้นทางการทดสอบของแผนภาพสเตทแมชชีนในรูปที่ 2.3 ดังนี้



รูปที่ 2.3 ตัวอย่างแผนภาพสเตทแมชชีนที่ใช้สำหรับสร้างเส้นทางการทดสอบ

นิยาม : ไพร์มพาธคือเส้นทางที่เริ่มต้นจากโหนด n_i ถึง n_j และเส้นทางเหล่านั้นเป็น simple path และไม่ปรากฏเส้นทางที่เป็นเส้นทางรองของเส้นทางอื่น

1. สร้างเส้นทางที่มีความยาวของเส้นทางเท่ากับศูนย์ และทำสัญลักษณ์ให้กับโหนดที่เป็นโหนดสิ้นสุดด้วยเครื่องหมายอัศเจรีย์ (!) เพื่อแสดงให้เห็นว่าเส้นทางดังกล่าวไม่สามารถสร้างต่อได้อีก แสดงผลลัพธ์ได้ดังนี้

    1) [0]
    2) [1]
    3) [2]
    4) [3]
    5) [4]
    6) [5]
    7) [6]!

2. สร้างเส้นทางที่มีความยาวของเส้นทางเท่ากับหนึ่งโดยการเพิ่มโหนดจำนวนหนึ่งโหนดเข้าสู่เส้นทางที่ได้ในขั้นตอนที่ 1 และตรวจสอบว่าเส้นทางใดเป็นเส้นทางที่ไม่สามารถต่อยอดได้ หากเป็นเส้นทางที่ไม่สามารถต่อยอดได้ให้ทำเครื่องหมายอัศเจรีย์ แต่หากการเพิ่มโหนดเข้าสู่เส้นทางแล้วทำให้ซ้ำกับโหนดเดิมที่มีอยู่ในเส้นทางดังกล่าวให้ทำเครื่องหมายดอกจันทร์ (*) และในระหว่างการต่อยอดจะต้องพิจารณาโหนดถัดไปที่จะนำมาต่อยอดว่าเป็นโหนดเดียวกับตัวสุดท้ายของเส้นทางที่กำลังพิจารณาหรือไม่ แสดงผลลัพธ์ได้ดังนี้

    8) [0,1]
    9) [0,4]

10) [1,2]

11) [1,5]

12) [2,3]

13) [3,1]

14) [4,4]*

15) [4,6]!

16) [5,6]!

3. สร้างเส้นทางที่มีความยาวของเส้นทางมากขึ้นทีละหนึ่งจนไม่สามารถต่อยอดเส้นทางได้อีก แสดงผลลัพธ์ได้ดังนี้

17) [0,1,2]

18) [0,1,5]

19) [0,4,6]!

20) [1,2,3]

21) [1,5,6]!

22) [2,3,1]

23) [3,1,2]

24) [3,1,5]

25) [0,1,2,3]!

26) [0,1,5,6]!

27) [1,2,3,1]*

28) [2,3,1,2]*

29) [2,3,1,5]

30) [3,1,2,3]*

31) [3,1,5,6]!

32) [2,3,1,5,6]!

4. หลังจากการสร้างเส้นทางเสร็จสิ้น ให้นำเส้นทางทั้งหมดมาพิจารณาและตัดเส้นทางที่เป็นสับเซตของเส้นทางอื่นออก แสดงผลลัพธ์ได้ดังนี้

14) [4,4]*

19) [0,4,6]!

25) [0,1,2,3]!

26) [0,1,5,6]!

27) [1,2,3,1]*

28) [2,3,1,2]*

30) [3,1,2,3]*

32) [2,3,1,5,6]!

5. ขั้นตอนสุดท้ายคือการสร้างเส้นทางทดสอบ (Test paths) สามารถทำได้โดยการนำเส้นทางที่ยาวที่สุดในขั้นตอนการหาไพร์มพาธที่ผ่านมาและต่อเติมด้วยเส้นทางอื่นๆโดยยึดจากโหนดเริ่มต้นและท้ายสุดของเส้นทางที่นำมาพิจารณา ดังนั้นเส้นทางการทดสอบหรือเส้นทางการทำงานของซอฟต์แวร์ที่เป็นไปได้คือ

1) [0,1,2,3,1,5,6]

2) [0,1,2,3,1,2,3,1,5,6]

3) [0,1,5,6]

4) [0,4,6]

5) [0,4,4,6]

2.1.5. วิธีการสร้างแผนภาพความสัมพันธ์ของเอนทิตี้จากแผนภาพคลาส

วิธีการที่ใช้สำหรับการออกแบบแผนภาพความสัมพันธ์จากแผนภาพคลาสมีด้วยกันหลายวิธี และหนึ่งในวิธีการเหล่านั้นคือ การนำเอาคลาสหรือกลุ่มของคลาสที่มีความสัมพันธ์กัน มาสร้างเป็นตาราง คุณลักษณะประจำตาราง และความสัมพันธ์ระหว่างตารางของฐานข้อมูล ประกอบไปด้วยวิธีการทั้งสิ้นเจ็ดวิธีคือ [13]

1. นำคลาสรูปธรรม (Concrete class) มาสร้างเป็นตารางในฐานข้อมูล หรือถ้าพบว่าคลาสนามธรรม (Abstract class) มีการสืบทอดจากคลาสหลายคลาส ก็สามารถนำคลาสนามธรรมนั้นมาสร้างเป็นตารางในฐานข้อมูลได้
2. นำคุณลักษณะประจำของคลาส (Attribute) มาสร้างเป็นคุณลักษณะประจำตารางหรือคอลัมภ์ของตารางนั้นๆ
3. นำพฤติกรรมการทำงานของคลาส (Method) มาสร้างเป็นสโตโพรสิเยอร์ (Store procedure) ในฐานข้อมูล หรือฟังก์ชันงานภายในซอฟต์แวร์
4. นำคุณลักษณะประจำคลาสที่มีความสัมพันธ์แบบองค์ประกอบ (Aggregation) หรือความสัมพันธ์ทั่วไป (Association) ที่มีค่าเพียงค่าเดียว มาสร้างเป็นคีย์เชื่อมระหว่างตาราง (Forign key)
5. นำคุณลักษณะประจำคลาสที่มีค่าหลายค่า แยกไปสร้างเป็นตารางใหม่และกำหนดความสัมพันธ์แบบหนึ่งต่อหลายระหว่างตารางเดิมเข้ากับตารางที่ถูกแยกออก

6. นำคุณลักษณะประจำคลาสที่มีความสัมพันธ์แบบองค์ประกอบ (Aggregation) หรือความสัมพันธ์ทั่วไป (Association) ที่มีหลายค่าไปสร้างเป็นตารางเชื่อมโยง (Associative table) และนำคีย์หลักของตารางที่สัมพันธ์กัน มาสร้างเป็นคุณลักษณะประจำตารางในตารางเชื่อมที่ถูกสร้างขึ้นใหม่ในฐานะคีย์เชื่อมของตารางใหม่
7. สำหรับคลาสที่มีความสัมพันธ์แบบสืบทอด ให้นำคลาสที่ได้รับการสืบทอดมาสร้างเป็นตารางเพียงคลาสเดียว โดยไม่ต้องนำคลาสแม่ (Parent class) มาสร้างเป็นตารางอีก

2.2. งานวิจัยที่เกี่ยวข้อง

2.2.1. การวิเคราะห์ความเสี่ยงในระดับสถาปัตยกรรมโดยใช้ยูเอ็มแอล [5]

การประเมินความเสี่ยงเป็นส่วนสำคัญในการบริหารจัดการโครงการพัฒนาซอฟต์แวร์ หากนำเอาการประเมินความเสี่ยงมาใช้ในช่วงแรกของโครงการก็จะทำให้การตัดสินใจในการจัดสรรทรัพยากรที่ขึ้น และตามหลักการของการปรับปรุงกระบวนการและผลิตภัณฑ์นั้น การสร้างแบบจำลองเพื่อวิเคราะห์ความเสี่ยงและการรวบรวมข้อมูลในช่วงแรกของการพัฒนาซอฟต์แวร์เป็นสิ่งที่จำเป็น เพื่อที่แบบจำลองที่สร้างขึ้นจะช่วยให้สามารถระบุความเสี่ยงขององค์ประกอบและความเสี่ยงของการเชื่อมโยงระหว่างองค์ประกอบได้ ดังนั้นจึงได้นำเสนอขั้นตอนวิธีในการประเมินความเสี่ยงที่สามารถนำมาใช้ได้ตั้งแต่ช่วงแรกในวัฏจักรชีวิตการพัฒนาซอฟต์แวร์ด้วยยูเอ็มแอล

โดยขั้นแรกพิจารณาปัจจัยความเสี่ยงเชิงพลวัตร (Dynamic risk factor) ที่แฝงอยู่ในแต่ละองค์ประกอบและการเชื่อมโยงระหว่างองค์ประกอบ และประเมินระดับความรุนแรงด้วยการวิเคราะห์ระดับภัยคุกคาม (Hazard analysis)

หลักแนวคิดที่นำมาใช้คือ การประเมินความเสี่ยงตามงานวิจัยนี้ ไม่ได้พิจารณาเพียงแค่สถานะขององค์ประกอบของซอฟต์แวร์ที่พิจารณาอยู่เท่านั้น แต่ยังนำเอาความเสี่ยงที่เกิดขึ้นระหว่างการเชื่อมต่อระหว่างสถานะมาเป็นส่วนหนึ่งของการพิจารณาด้วย จากนั้นประยุกต์ใช้แบบจำลองของมาร์คอว์ฟ (Markov model) ในการพิจารณาปัจจัยความเสี่ยงของแต่ละลำดับเหตุการณ์ (Scenario) ซึ่งปัจจัยความเสี่ยงที่วิเคราะห์ได้จากลำดับของเหตุการณ์นั้นจะนำไปสู่การวิเคราะห์ความเสี่ยงในระดับของยูสเคสและระบบโดยรวม นอกจากนี้ยังได้ระบุองค์ประกอบและการเชื่อมโยงขององค์ประกอบที่มีความวิกฤตที่จะต้องได้รับการวิเคราะห์ ออกแบบ การพัฒนาและการทดสอบที่ระมัดระวัง สำหรับขั้นตอนวิธีในการวิเคราะห์ความเสี่ยงที่ได้กล่าวไว้ในข้างต้น มีรายละเอียดการดำเนินการดังต่อไปนี้

2.2.1.1.กระบวนการวิเคราะห์ความเสี่ยง (Risk Analysis Process)

กระบวนการวิเคราะห์ความเสี่ยงในที่นี้เป็นการอธิบายถึงการแบ่งลำดับขั้นตอนที่ทำให้ได้มาซึ่งผลลัพธ์สุดท้ายของขั้นตอนวิธี กล่าวคือ โครงสร้างของแผนภาพยูสเคสจะประกอบไปด้วยยูสเคสที่มี

ความเป็นอิสระต่อกันประกอบอยู่และในแต่ละยูสเคส และแต่ละยูสเคสจะประกอบด้วยลำดับเหตุการณ์อย่างน้อยหนึ่งลำดับ และในแต่ละลำดับเหตุการณ์จะสามารถแสดงได้ด้วยแผนภาพลำดับ และด้วยแผนภาพลำดับนี้เองจะแสดงให้เห็นถึงจำนวนองค์ประกอบและข้อความที่ถูกส่งระหว่างองค์ประกอบเหล่านั้น ซึ่งขั้นตอนวิธีดังกล่าวได้แบ่งส่วนพิจารณาภายส่วนประกอบย่อยของแต่ละยูสเคส ลำดับเหตุการณ์ องค์ประกอบและการเชื่อมโยงขององค์ประกอบตามลำดับ สำหรับแต่ละลำดับเหตุการณ์นั้นปัจจัยความเสี่ยงในแต่ละองค์ประกอบถูกพิจารณาออกมาในรูปของค่าความซับซ้อนเชิงพลวัต (Dynamic complexity) ส่วนปัจจัยความเสี่ยงของการปฏิสัมพันธ์ระหว่างองค์ประกอบถูกพิจารณาออกมาให้รูปของค่าคลัปปลิงเชิงพลวัต (Dynamic coupling) ส่วนระดับความรุนแรงที่เกิดขึ้นนั้นถูกกำหนดโดยผู้เชี่ยวชาญในสาขานั้นๆโดยอาศัยหลักการวิเคราะห์ภัยคุกคาม

หลังจากที่ได้ค่าปัจจัยความเสี่ยงของแต่ละองค์ประกอบและปฏิสัมพันธ์แล้ว ลำดับต่อไปจะเป็นการพิจารณาในส่วนของลำดับเหตุการณ์ด้วยแบบจำลองมาร์คอว์ฟ จากนั้นค่าความเสี่ยงของลำดับขั้นตอนที่ได้จะนำไปใช้เพื่อคำนวณความเสี่ยงโดยรวมของระบบต่อไป ในท้ายที่สุดผลลัพธ์ของขั้นตอนทั้งสิ้นที่ระบุไว้ในข้างต้นคือ รายการของยูสเคส รายการของลำดับเหตุการณ์ และรายการขององค์ประกอบและปฏิสัมพันธ์ที่มีความเสี่ยงในระดับวิกฤต

1) การประเมินปัจจัยความเสี่ยงขององค์ประกอบและการเชื่อมโยง (Assessment of the Component/Connector Risk Factors)

ในแต่ละลำดับเหตุการณ์สามารถคำนวณค่าปัจจัยความเสี่ยงเชิงพลวัตสำหรับองค์ประกอบและปฏิสัมพันธ์ได้ด้วยการคำนวณค่าความซับซ้อนเชิงพลวัตและค่าคลัปปลิงเชิงพลวัตตามลำดับโดยนำมาพิจารณาร่วมกับระดับความรุนแรงที่จะเกิดขึ้นซึ่งมีความแตกต่างกันไปในแต่ละเหตุการณ์ มีรายละเอียดดังนี้

- ปัจจัยความเสี่ยงขององค์ประกอบ (rf_i^x) สามารถคำนวณได้จาก

$$rf_i^x = DOC_i^x \cdot svt_i^x$$

โดยกำหนดให้ DOC_i^x คือค่าบรรทัดฐานความซับซ้อนเชิงพลวัต (Normalized dynamic complexity) ที่มีค่าอยู่ในช่วง $0 \leq DOC_i^x \leq 1$ และ svt_i^x คือระดับความรุนแรงขององค์ประกอบลำดับที่ i ในลำดับเหตุการณ์ที่ S_{x}

- ปัจจัยความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบ (rf_{ij}^x) ลำดับที่ i และ j ในลำดับเหตุการณ์ที่ S_{x} สามารถคำนวณได้จาก

$$rf_{ij}^x = EOC_{ij}^x \cdot svt_{ij}^x$$

โดยกำหนดให้ EOC_{ij}^x คือค่าบรรทัดฐานคลัปปลิงเชิงพลวัต (Normalized dynamic coupling) มีค่าอยู่ในช่วง $0 \leq EOC_{ij}^x \leq 1$ และ svt_{ij}^x คือระดับความรุนแรงของปฏิสัมพันธ์ระหว่างองค์ประกอบลำดับที่ i ถึงองค์ประกอบลำดับที่ j ในลำดับเหตุการณ์ที่ S_{x}

2) มาตรวัดข้อกำหนดรายละเอียดเชิงพลวัตด้วยยูเอ็มแอล (Dynamic Specification Metrics Using UML)

เป็นที่ทราบกันดีว่าจำนวนข้อบกพร่องที่พบในซอฟต์แวร์มีความเกี่ยวข้องกับระดับความซับซ้อนของซอฟต์แวร์ ดังนั้นการคำนวณหาค่าความซับซ้อนเชิงพลวัตขององค์ประกอบสามารถประยุกต์ได้จากสูตรการคำนวณค่าความซับซ้อนของแมคเคป (McCabe's cyclomatic complexity) คือ $CC = e - n + 2$ แต่ทว่าสูตรดังกล่าวเป็นการคำนวณหาค่าความซับซ้อนจากกราฟควบคุมการไหล (Control flow graph) ซึ่งโดยปกติแล้วกราฟประเภทนี้จะสามารถทำได้เมื่อมีการพัฒนาจนถึงระดับซอร์สโค้ด จึงต้องประยุกต์สูตรดังกล่าวให้สามารถใช้ควบคู่กับแผนภาพสเตทแมชชีนที่พัฒนาขึ้น กล่าวคือ แผนภาพสเตทแมชชีนขององค์ประกอบลำดับที่ i มีจำนวนของสถานะและการเปลี่ยนแปลงระหว่างสถานะซึ่งเป็นการอธิบายพฤติกรรมเชิงพลวัตรขององค์ประกอบหนึ่งๆในลำดับเหตุการณ์ที่ S_x

กำหนดให้ C_i^x แทนจำนวนสถานะภายในองค์ประกอบลำดับที่ i และ T_i^x แทนจำนวนการเปลี่ยนแปลงระหว่างสถานะภายในองค์ประกอบลำดับที่ i ในลำดับเหตุการณ์ที่ S_x โดยที่จำนวนทั้งสองจำนวนถูกปรับให้สามารถนำมาใช้กับการคำนวณหาค่าความซับซ้อนของแมคเคปคือ $c_i^x = |C_i^x|$ เปรียบได้กับจำนวนโหนด และ $t_i^x = |T_i^x|$ เปรียบได้กับจำนวนเส้นภายในกราฟควบคุมการไหล ดังนั้นสามารถกำหนดได้ว่าการคำนวนหาค่าความซับซ้อนขององค์ประกอบ doc_i^x ในลำดับที่ i ในลำดับเหตุการณ์ที่ S_x ดังนี้

$$doc_i^x = t_i^x - c_i^x + 2$$

และสามารถคำนวณค่าบรรทัดฐานความซับซ้อนขององค์ประกอบในลำดับที่ i ในลำดับเหตุการณ์ที่ S_x ได้จาก

$$DOC_i^x = \frac{doc_i^x}{\sum_{k \epsilon S_x} doc_k^x}$$

ในลำดับต่อไปเป็นการคำนวณหาค่าบรรทัดฐานคลัปปลิงของปฏิสัมพันธ์ระหว่างองค์ประกอบซึ่งผลลัพธ์ที่ได้จะแสดงออกในรูปของเมทริกซ์ (Matrix) โดยแถวของเมทริกซ์หมายถึงองค์ประกอบในฝั่งผู้ส่ง ในขณะที่หลักหมายถึงองค์ประกอบฝั่งผู้รับ มีรายละเอียดดังนี้

กำหนดให้ MT_{ij}^x แทนกลุ่มของข้อความที่ถูกส่งจากองค์ประกอบ i ถึง j และ MT^x แทนข้อความทั้งหมดที่มีการแลกเปลี่ยนกันในทุกๆองค์ประกอบในลำดับเหตุการณ์ที่ S_x ซึ่งสามารถคำนวณค่า EOC_{ij}^x ได้ดังนี้

$$EOC_{ij}^x = \frac{|MT_{ij}^x|_{i.j \epsilon S_x, i \neq j}}{|MT^x|}$$

3) การวิเคราะห์ระดับความรุนแรง

นอกเหนือจากความซับซ้อนและคลัปปลิงเชิงพลวัตที่คำนึงถึงในการพิจารณาความเสี่ยงที่เกิดขึ้นแล้ว ระดับความรุนแรงถือได้ว่าเป็นอีกปัจจัยหนึ่งที่ผลต่อการคำนวณความเสี่ยงที่อาจเกิดขึ้น ยกตัวอย่างเช่น องค์ประกอบของซอฟต์แวร์หนึ่งอาจมีค่าความซับซ้อนต่ำแต่หากมีความเสียหายเกิดขึ้นนั้นจะทำให้ส่งผลกระทบที่รุนแรงต่อซอฟต์แวร์ได้ ดังนั้นการวิเคราะห์ระดับความรุนแรงจึงเป็นสิ่งที่สำคัญ

การวิเคราะห์ภัย (Hazard analysis) เป็นวิธีการวิเคราะห์ระดับความรุนแรงวิธีหนึ่งที่ใช้ในการแบ่งระดับความรุนแรงมีรายละเอียดดังนี้

ตารางที่ 2.5 ระดับความรุนแรงและดัชนีระดับความรุนแรงจากขั้นตอนการวิเคราะห์ภัยคุกคาม

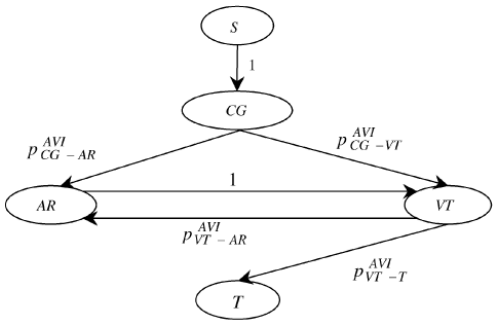
ระดับความรุนแรง	คำอธิบาย	ดัชนีความรุนแรง
Catastrophic	ความเสียหายที่อาจทำให้เกิดความตายหรือระบบทั้งหมดเสียหาย	0.95
Critical	ความเสียหายที่ทำให้บาดเจ็บรุนแรง ส่วนหลักของระบบเสียหาย	0.75
Marginal	ความเสียหายที่ทำให้บาดเจ็บเล็กน้อย ส่วนรองของระบบเสียหายหรือล่าช้า	0.50
Minor	ความเสียหายที่ไม่ร้ายแรงเพียงพอต่อการทำให้ระบบเสียหายหรือถูกทำลาย แต่ทำให้ต้องวางแผนในการปรับปรุงในภายหลัง	0.25

2.2.1.2.ความเสี่ยงของลำดับเหตุการณ์ (Scenarios Risk Factors)

ในขั้นตอนนี้เป็นการสร้างแบบจำลองโดยใช้พื้นฐานของสถานะเพื่อประเมินความน่าเชื่อถือซึ่งพิจารณาทั้งในส่วนข้อบกพร่องขององค์ประกอบและปฏิสัมพันธ์ ประกอบด้วย 2 ขั้นตอนคือ

1) สร้างแบบจำลองแสดงพฤติกรรมการทำงานของซอฟต์แวร์

เป็นการสร้างแบบจำลองที่มีลักษณะคล้ายกราฟควบคุมการไหลซึ่งมีโหนดเริ่มต้น (S) และโหนดสุดท้าย (T) เพียงโหนดเดียว แต่ทว่าแบบจำลองที่สร้างขึ้นในขั้นตอนนี้ไม่ได้กำหนดให้มีโหนดเริ่มต้นและโหนดสุดท้ายเพียงโหนดเดียวแต่สามารถกำหนดให้โหนดดังกล่าวปรากฏขึ้นในแบบจำลองได้มากกว่าหนึ่งโหนด (Multientry multiexit graphs) ดังนั้นสถานะที่ปรากฏในแบบจำลองที่มีลักษณะคล้ายกราฟควบคุมการไหลนี้เป็นตัวแทนขององค์ประกอบ ส่วนเส้นระหว่างโหนดเป็นตัวแทนของการควบคุมการเปลี่ยนแปลงระหว่างองค์ประกอบ แสดงได้ดังรูปที่ 2.4



รูปที่ 2.4 DTMC ของแบบจำลองพฤติกรรมการทำงานสำหรับลำดับเหตุการณ์

นอกจากนี้ยังสรุปได้ว่าการควบคุมการเปลี่ยนแปลงระหว่างองค์ประกอบมีคุณสมบัติเป็นไปตามกฎของมาร์คอร์ฟ และข้อสรุปนี้เองทำให้สามารถสร้างแบบจำลองพฤติกรรมการทำงานของแต่ละลำดับเหตุการณ์ได้ด้วย Absorbing discrete time Markov chain (DTMC) ที่อยู่ในรูปของเมทริกซ์ความน่าจะเป็นในการเปลี่ยนแปลงสถานะ ดังแบบจำลอง $P^x = [p^x_{ij}]$

โดยกำหนดให้ p^x_{ij} แทนความน่าจะเป็นที่โปรแกรมจะทำงานจากในองค์ประกอบลำดับที่ i สำเร็จและเปลี่ยนไปทำงานในองค์ประกอบลำดับที่ j ในลำดับเหตุการณ์ที่ S_x และความน่าจะเป็นดังกล่าวสามารถคำนวณได้จากจำนวนข้อความที่ส่งจากองค์ประกอบลำดับที่ i ถึงลำดับที่ j เทียบกับจำนวนข้อความทั้งหมดที่ถูกส่งจากองค์ประกอบลำดับที่ i ถึงทุกองค์ประกอบในลำดับเหตุการณ์ S_x ดังแบบจำลอง

$$p^x_{ij} = \frac{n^x_{ij}}{\sum_j n^x_{ij}}$$

ผลลัพธ์ที่ได้จากแบบจำลองในข้างต้นสามารถแสดงได้ด้วยเมทริกซ์ความน่าจะเป็นของการเปลี่ยนสถานะดังต่อไปนี้

$$P^x = \begin{array}{c} \\ S \\ CG \\ AR \\ VT \\ T \end{array} \begin{array}{c} \begin{array}{ccccc} S & CG & AR & VT & T \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0.5 & 0 & 0.5 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

(1) สร้างแบบจำลองความเสี่ยงของลำดับเหตุการณ์

ขั้นตอนที่สองคือการสร้างแบบจำลองความเสี่ยงในระดับเหตุการณ์เพื่อพิจารณาปัจจัยความเสี่ยงขององค์ประกอบและปฏิสัมพันธ์ ซึ่งความล้มเหลวอาจเกิดขึ้นในระหว่างที่ระบบทำงานและสมมติให้ความล้มเหลวเหล่านั้นเป็นอิสระต่อกัน และด้วยเหตุผลดังกล่าวสามารถแบบจำลองเสี่ยงในระดับเหตุการณ์ได้โดยนำเอาแบบจำลองพฤติกรรมการทำงานที่ได้ในขั้นตอนแรกมาเพิ่มเติมด้วยโหนดของสถานะที่ผิดพลาด (F) เนื่องระดับความรุนแรงนั้นมีผลต่อการคำนวณหาค่าความเสี่ยง และผลจากการวิเคราะห์ความเสี่ยงในขั้นตอนที่สองนั้นทำให้สามารถระบุโหนดความผิดพลาดได้ทั้งสิ้นสี่โหนดคือ F_{minor}, $F_{marginal}$,$F_{critical}$ และ $F_{catastrophic}$ และเมื่อนำกฎของมาร์คอว์ฟมาประยุกต์ใช้กับแบบจำลองนี้

จะได้ว่าจำนวนของโหนดของสถานะชั่วคราว (Transient node) มีค่าเท่ากับ n+1 (เมื่อกำหนดให้ n เท่ากับจำนวนองค์ประกอบรวมกับโหนดเริ่มต้น S) และจำนวนโหนดดูดกลืน (Absorbing node) เท่ากับ m+1 (เมื่อกำหนดให้ m เท่ากับจำนวนโหนดของสถานะที่ผิดพลาดรวมกับโหนดสุดท้าย T)

จากนั้นปรับเมทริกซ์ความน่าจะเป็น P^x เป็น $\overline{P^x}$ ซึ่งทำให้ค่า p_{ij}^x ถูกเปลี่ยนเป็น $(1-rf_i^x)\cdot p_{ij}^x \cdot (1-rf_{ij}^x)$ แบบจำลองนี้แสดงถึงกรณีที่องค์ประกอบลำดับที่ i ทำงานโดยปราศจากความล้มเหลว และสามารถเปลี่ยนสถานะของการทำงานไปยังองค์ประกอบลำดับที่ j โดยปราศจากความล้มเหลวเช่นเดียวกัน ความล้มเหลวขององค์ประกอบลำดับที่ i ใดๆถูกพิจารณาโดยการสร้างเส้นเชื่อมไปยังโหนดสถานะล้มเหลวที่มีระดับความรุนแรงที่เหมาะสมด้วยความน่าจะเป็น rf_i^x และเช่นเดียวกันความล้มเหลวของปฏิสัมพันธ์ระหว่างองค์ประกอบลำดับที่ iกับองค์ประกอบลำดับที่ j ถูกพิจารณาด้วยการสร้างเส้นเชื่อมไปยังโหนดสถานะล้มเหลวด้วยความน่าจะเป็น $(1-rf_i^x)\cdot p_{ij}^x \cdot rf_{ij}^x$ สำหรับเมทริกซ์ความน่าจะเป็นของการเปลี่ยนแปลงแบบ DTMC ซึ่งก็คือ $\overline{P^x}$ มีรูปแบบดังนี้

$$\overline{P^x} = \begin{bmatrix} Q^x & C^x \\ 0 & I \end{bmatrix}$$

กำหนดให้

Q^x เป็นเมทริกซ์ขนาด (n+1)×(n+1) ที่แสดงถึงความน่าจะเป็นในการเปลี่ยนสถานะระหว่างโหนดสถานะชั่วคราว

I คือเมทริกซ์เอกลักษณ์ขนาด (m+1)×(m+1)

C^xแทนเมทริกซ์ขนาด (m+1)×(m+1) ที่แสดงความน่าจะเป็นในการเปลี่ยนสถานะระหว่างโหนดสถานะชั่วคราวถึงโหนดสถานะความล้มเหลว

A^x แทนเมทริกซ์ความน่าจะเป็นที่ DTMC เริ่มต้นจากโหนดสถานะชั่วคราวเปลี่ยนแปลงสถานะไปยังโหนดความล้มเหลว สามารถคำนวณได้ดังแบบจำลอง

$$A^x = (I - Q^x)^{-1} C^x$$

ดังที่กล่าวไว้ในข้างต้นเมทริกซ์ความน่าจะเป็น Q^x, C^x และ A^x แสดงตัวอย่างได้ดังต่อไปนี้

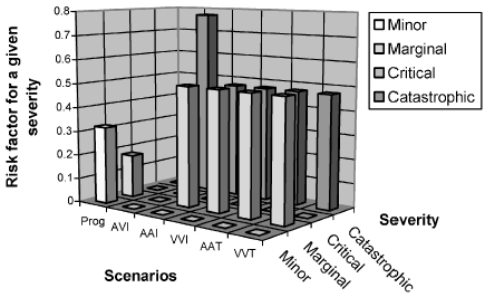
$$Q^{AVI} = \begin{array}{c} \\ S \\ CG \\ AR \\ VT \end{array} \begin{array}{c} \begin{array}{cccc} S & CG & AR & VT \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0.4998 & 0.4998 \\ 0 & 0 & 0 & 0.3619 \\ 0 & 0 & 0.0472 & 0 \end{pmatrix} \end{array}$$

$$C^{AVI} = \begin{array}{c} \\ S \\ CG \\ AR \\ VT \end{array} \begin{array}{c} \begin{array}{ccccc} T & F_{\text{minor}} & F_{\text{marginal}} & F_{\text{critical}} & F_{\text{catastrophic}} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0004 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.6381 \\ 0.3258 & 0 & 0 & 0 & 0.6270 \end{pmatrix} \end{array}$$

$$A^{AVI} = (I - Q^{AVI})^{-1} C^{AVI} =$$

$$\begin{array}{c} \\ S \\ CG \\ AR \\ VT \end{array} \begin{array}{c} \begin{array}{ccccc} T & F_{\text{minor}} & F_{\text{marginal}} & F_{\text{critical}} & F_{\text{catastrophic}} \end{array} \\ \begin{pmatrix} 0.2256 & 0 & 0.0004 & 0 & 0.7740 \\ 0.2256 & 0 & 0.0004 & 0 & 0.7740 \\ 0.1200 & 0 & 0 & 0 & 0.8800 \\ 0.3315 & 0 & 0 & 0 & 0.6685 \end{pmatrix} \end{array}$$

ขั้นตอนที่ได้กล่าวมาทั้งหมดจะทำให้สามารถคำนวณหาปัจจัยความเสี่ยงของลำดับเหตุการณ์หนึ่งๆในระบบได้ จากนั้นนำปัจจัยความเสี่ยงที่ได้ในแต่ละลำดับเหตุการณ์มาสร้างกราฟสามมิติที่แกน z แทนค่าปัจจัยความเสี่ยงตามระดับความรุนแรงที่กำหนดไว้ แสดงได้ดังรูปที่ 4



รูปที่ 2.5 กราฟแสดงค่าปัจจัยความเสี่ยงตามระดับความรุนแรงของแต่ละลำดับเหตุการณ์ในระบบ

(2) การระบุองค์ประกอบวิกฤต (Identifying Critical Components)

หลังจากที่ได้ค่าความเสี่ยงในระดับองค์ประกอบของระบบตามความต้องการแล้ว ขั้นตอนสุดท้ายคือการระบุระดับความวิกฤติของลำดับเหตุการณ์ ซึ่งขั้นตอนนี้เป็นขั้นตอนที่สำคัญต่อกระบวนการพัฒนาซอฟต์แวร์ เพื่อทำให้การบริหารโครงการสามารถระบุทรัพยากรให้เหมาะสมกับลักษณะของงานในโครงการได้มีประสิทธิภาพมากยิ่งขึ้น แสดงได้ดังตารางที่ 2.6

ตารางที่ 2.6 ค่าความเสี่ยงของแต่ละลำดับเหตุการณ์ในระบบ

	Programming	AVI	AAI	VVI	AAT	VVT
Minor	0.3169	0	0	0	0	0
Marginal	0.1782	0.0004	0.5002	0.5002	0.5001	0.5001
Critical	0	0	0	0	0	0
Catastrophic	0	0.7740	0.4743	0.4743	0.4747	0.4747
Scenario risk factor	0.4951	0.7744	0.9745	0.9745	0.9748	0.9748

2.2.2. การสร้างข้อมูลทดสอบจากข้อกำหนดรายละเอียดเชิงสถานะ (Generating test data from state-based specifications) [7]

งานวิจัยนี้นำเสนอหลักการทั่วไปในการสร้างกรณีทดสอบจากข้อกำหนดรายละเอียดเชิงสถานะ ซึ่งประกอบไปด้วยวิธีการทดสอบในหลายระดับซึ่งรวมไปถึงหลักการครอบคลุม (Coverage criteria) โดยขั้นตอนการสร้างกรณีทดสอบมีด้วยกันหลายขั้นตอนสำหรับการเปลี่ยนรายละเอียดความต้องการเป็นกรณีทดสอบ

โดยปกติแล้วแผนภาพสเตทแมชชีนได้กำหนดเงื่อนไขก่อนหน้าบนการเปลี่ยนแปลงไปยังสถานะใดๆ ซึ่งเป็นค่าที่ถูกกำหนดไว้สำหรับการเปลี่ยนสถานะซึ่งค่าเหล่านี้จะต้องเป็นไปตามข้อกำหนดจึงจะทำให้การเปลี่ยนแปลงสถานะนั้นเกิดขึ้นได้ ส่วนเหตุการณ์กระตุ้นเป็นเหตุการณ์ที่ทำให้ค่าที่อยู่ในตัวแปรมีเปลี่ยนแปลงซึ่งเป็นสาเหตุให้มีการปรับเปลี่ยนสถานะนั่นเอง สำหรับค่าที่เกิดขึ้นก่อนเหตุการณ์กระตุ้นเรียกว่า “before-value” และค่าที่เกิดขึ้นหลังเหตุการณ์กระตุ้นเรียกว่า “after-values” ส่วนสถานะที่เกิดการเปลี่ยนแปลงเรียกว่า “pre-state” และสถานะที่เกิดขึ้นหลังการเปลี่ยนแปลงเรียกว่า “pre-state”

สำหรับซอฟต์แวร์ที่มีการทำงานขึ้นกับสถานะนั้น กรณีทดสอบมักถูกสร้างจากเหตุการณ์ที่เป็นตัวกระตุ้นและเงื่อนไขก่อนหน้าสำหรับการเปลี่ยนแปลงแต่ละครั้ง ซึ่งนำเสนอหลักการทดสอบไว้สี่ระดับคือ หลักการครอบคลุมทรานสิชัน (Transition coverage) หลักการครอบคลุมประพจน์ (Full predicate coverage) หลักการครอบคลุมคู่ทรานสิชัน (Transition-pair coverage) และหลักการแบบลำดับเหตุการณ์ที่สมบูรณ์ (Complete sequence) ซึ่งสามารถนำเอาแผนภาพสเตทแมชชีนมาประยุกต์ใช้ผ่านกราฟที่มีทิศทาง (Directed graph) ซึ่งกำหนดให้โหนดแทนสถานะและเส้นกราฟแทนการเปลี่ยนแปลงระหว่างสถานะที่เป็นไปได้ อนึ่งหลักการในการทดสอบทั้งสี่ประเภทที่กล่าวมาสามารถนำมาประยุกต์ใช้กับข้อกำหนดความต้องการเชิงสถานะได้ซึ่งการเลือกใช้ต้องขึ้นอยู่กับงบประมาณในการทดสอบซอฟต์แวร์นั้นๆด้วย และสำหรับหลักการครอบคลุมทรานสิชันเป็นวิธีการทดสอบที่ทำให้ได้จำนวนกรณีทดสอบน้อยกว่าหลักการครอบคลุมประพจน์แต่การนำหลักการครอบคลุมประพจน์ไปใช้จะครอบคลุมหลักการครอบคลุมทรานสิชันด้วย ซึ่งการนำไปใช้ควรเลือกเพียงหลักการใดหลักการหนึ่งเท่านั้น ส่วนหลักการครอบคลุมคู่ทรานสิชันถูกนำมาใช้เพื่อตรวจสอบส่วนต่อประสานระหว่างสถานะ ซึ่งตามข้อเท็จจริงแล้วการทดสอบประเภทนี้ถือเป็นส่วนหนึ่งของหลักการครอบคลุมทรานสิชันเพียงแต่นำมาใช้ในการทดสอบต่างจุดประสงค์กัน และสุดท้ายหลักการแบบลำดับเหตุการณ์ที่สมบูรณ์ ถูกนำมาใช้เพื่อทดสอบความสมบูรณ์ของการทำงานในแต่ละเส้นทาง

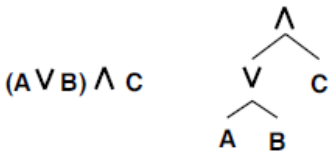
2.2.2.1.หลักการครอบคลุมประพจน์ (Full predicate coverage criterion)

หลักการนี้กำหนดให้แต่ละเงื่อนไขในแต่ละประพจน์บนการเปลี่ยนแปลงถูกทดสอบอย่างเป็นอิสระจากกัน ซึ่งนิพจน์ที่นำมาใช้คือนิพจน์บูลลีนที่ประกอบไปด้วยตัวดำเนินการ AND (∧) OR (∨)

และ NOT (~) ซึ่งหากกำหนดให้ประพจน์ P ใดๆมีนิพจน์แบบบูลลีนแล้ว ค่า P มีเพียงสองค่าที่เป็นไปได้คือ ถูกและผิดเท่านั้น

1) Satisfying full predicate coverage

มีวิธีมากมายที่จะทำให้สามารถสร้างกรณีทดสอบที่เป็นไปตามหลักการครอบคลุมประพจน์ หนึ่งในวิธีเหล่านั้นที่งานวิจัยนี้นำเสนอคือการนำต้นไม้แจกแจงนิพจน์ (Expression parse tree) ที่มีลักษณะเป็นต้นไม้แบบไบนาลี (Binary tree) ที่มี **AND (∧) OR (∨) และ NOT (~)** เป็นตัวดำเนินการสำหรับโหนดภายใน และมีตัวแปรหรือค่าคงที่เป็นลีฟโหนด (Leaf nodes) มาประยุกต์ใช้ในการหากรณีทดสอบที่เป็นไปตามหลักการครอบคลุมประพจน์ แสดงตัวอย่างของต้นไม้แจกแจงนิพจน์สำหรับ $(A \vee B) \wedge C$ ดังตัวอย่างต่อไปนี้

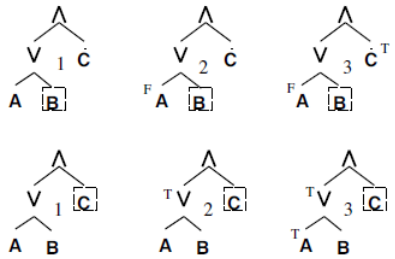


รูปที่ 2.6 ต้นไม้แจกแจงนิพจน์ $(A \vee B) \wedge C$

จากต้นไม้ในข้างต้นสามารถสร้างกรณีทดสอบตามหลักการครอบคลุมประพจน์ตามขั้นตอนดังนี้

1. เลือกประโยคของการทดสอบ (test clause)
2. ต้นไม้ถูกเดินทางไปยังส่วนบนสุด (root) จากประโยคที่ถูกเลือก และเดินทางจากส่วนบนสุดไปยังแต่ละประโยค
3. ในขณะที่มีการเดินจากแต่ละประโยคสู่ส่วนบนสุดของต้นไม้ มีข้อกำหนดดังนี้
   3.1. ถ้าโหนดแม่ของประโยคที่ถูกเลือกเป็น or แล้ว โหนดพี่น้องจะต้องมีค่าเป็นเท็จ
   3.2. ถ้าโหนดแม่ของประโยคที่ถูกเลือกเป็น and แล้ว โหนดพี่น้องจะต้องมีค่าเป็นจริง
   3.3. ถ้าโหนดที่ถูกเลือกเป็นตัวดำเนินการ not แล้ว โหนดแม่จะถูกกำหนดให้มีค่ากลับกันกับโหนดลูก
4. เมื่อเดินทางจากส่วนบนสุดสู่ประโยค มีข้อกำหนดดังนี้
   4.1. ถ้าโหนด and มีค่าเป็นจริงแล้ว โหนดลูกทั้งสองโหนดจะต้องมีค่าเป็นจริง แต่ถ้าโหนด and มีค่าเป็นเท็จแล้วโหนดอย่างน้อยหนึ่งโหนดจะต้องมีค่าเป็นเท็จ
   4.2. ถ้าโหนด or มีค่าเป็นเท็จแล้ว โหนดลูกทั้งสองโหนดจะต้องมีค่าเป็นเท็จด้วย แต่ถ้าโหนด or มีค่าเป็นจริงแล้ว โหนดลูกอย่างน้องหนึ่งตัวต้องมีค่าเป็นจริง
   4.3. ถ้าเป็นโหนด not แล้ว โหนดลูกจะต้องมีค่ากลับกันกับโหนดแม่

ขั้นตอนในข้างต้นสามารถอธิบายกระบวนการที่เกิดขึ้นเมื่อกำหนดให้ B และ C เป็นประโยคของการทดสอบ แสดงได้ดังรูปที่ 6



รูปที่ 2.7 กระบวนการสร้างกรณีทดสอบแบบครอบคลุมประพจน์เมื่อกำหนดให้ B และ C เป็นประโยคของการทดสอบ (เรียงจากแถวบนลงล่างตามลำดับ)

จากขั้นตอนการกระจายนิพจน์ของประโยคที่ต้องการทดสอบนั้น สามารถแสดงผลให้อยู่ในรูปของตารางค่าความจริง (Truth table) ได้ดังนี้

ตารางที่ 2.7 ค่าความจริงของนิพจน์ $(A \vee B) \wedge C$

ลำดับที่	$(A \vee B) \wedge C$		
1	T	F	T
2	F	F	T
3	F	T	T
4	F	F	T
5	T	T	T
6	T	T	T

2.2.2.2.การจัดการเหตุการณ์กระตุ้น (Handling triggering events)

ดังที่กล่าวไว้ว่าเหตุการณ์ที่เป็นตัวกระตุ้นเป็นการทำให้ค่าที่มีอยู่ในตัวแปรนั้นๆเปลี่ยนแปลงไป ส่วนนิพจน์เป็นสาเหตุให้ซอฟต์แวร์มีการเปลี่ยนแปลงจากสถานะหนึ่งไปยังอีกสถานะหนึ่ง โดยปกติแล้วเหตุการณ์ที่เป็นตัวกระตุ้นจะถูกกำหนดให้มีสองค่าคือ ค่าที่เกิดขึ้นก่อนเหตุการณ์กระตุ้นเรียกว่า (before-value) และค่าที่เกิดขึ้นหลังเหตุการณ์กระตุ้น (after-value) ดังนั้นในงานวิจัยนี้จึงได้กำหนดรูปแบบของเหตุการณ์กระตุ้นออกเป็นสองแบบคือค่าที่เกิดขึ้นและหลังเหตุการณ์กระตุ้น ใช้สัญลักษณ์ A และ A' ตามลำดับ

1) กฎการขยายเหตุการณ์ที่มีการกระตุ้น (Triggering event expansion rule)

เนื่องจากการเปลี่ยนแปลงสถานะของวัตถุภายในแผนภาพสเตทแมชชีนนั้นจะเกิดขึ้นเมื่อมีเหตุการณ์เข้ากระตุ้นวัตถุที่สถานะใดสถานะหนึ่ง โดยปกติแล้วการแสดงเหตุการณ์เหล่านั้นจะแสดงได้

ด้วยนิพจน์ทางคณิตศาสตร์ ซึ่งสามารถนำเอาหลักการกระจายประพจน์มาใช้เพื่อขยายเหตุการณ์เหล่านั้นได้ดังนี้

1. $@T(X) = \sim X \wedge X'$
2. $@T(X \wedge Y) = \sim(X \wedge Y) \wedge (X' \wedge Y') = (\sim X \wedge \sim Y) \wedge (X' \wedge Y')$
3. $@T(X \vee Y) = \sim(X \vee Y) \wedge (X' \vee Y') = (\sim X \wedge \sim Y) \wedge (X' \vee Y')$
4. $@F(X) = X \wedge \sim X'$
5. $@F(X \wedge Y) = (X \wedge Y) \wedge \sim(X' \wedge Y') = (X \wedge Y) \wedge (\sim X' \vee \sim Y')$
6. $@F(X \vee Y) = (X \vee Y) \wedge \sim(X' \vee Y') = (X \vee Y) \wedge (\sim X' \wedge \sim Y')$

   กำหนดให้

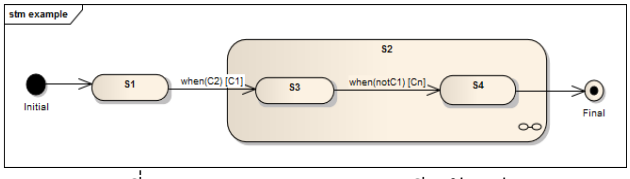
   - $@T$ หรือ $@F$ แทนเหตุการณ์กระตุ้นซึ่งทำให้ค่า X เปลี่ยนแปลงเมื่อเกิดการกระตุ้นให้มีการเปลี่ยนแปลงสถานะ
   - $@T(X)$ คือค่า X ต้องเปลี่ยนจากเท็จ เป็น จริง จึงจะทำให้เกิดการเปลี่ยนสถานะ
   - $@F(X)$ คือค่า X ต้องเปลี่ยนจากจริง เป็น เท็จ จึงทำให้เกิดการเปลี่ยนสถานะ
   - X และ X' แทนค่าของเหตุการณ์ก่อนการกระตุ้นและค่าของเหตุการณ์หลังการกระตุ้นซึ่งจะทำให้เกิดการเปลี่ยนสถานะ

2.2.3. Test cases generation from a state chart diagram [8]

งานวิจัยนี้เป็นการนำเสนอวิธีการในการสร้างกรณีทดสอบที่สอดคล้องกับหลักการครอบคลุมประพจน์จากแผนภาพสเตทแมชชีนที่มีสถานะย่อยภายใน ซึ่งกรณีทดสอบที่ได้จะใช้ในการติดตามพฤติกรรมของคลาสในระบบ มีขั้นตอนการสร้างกรณีทดสอบจากแผนภาพสเตทแมชชีนดังนี้

1) พิจารณาเงื่อนไขของทรานสิชันจากข้อกำหนดรายละเอียดยูเอ็มแอล

เป็นขั้นตอนการพิจารณาเงื่อนไขของการเปลี่ยนแปลงว่ามีประพจน์ใดภายใต้เงื่อนไขอะไรบ้างในแต่ละการเปลี่ยนแปลงของสถานะ หากกำหนดให้แผนภาพสเตทแมชชีนมีรายละเอียดดังรูปที่ 2.8 ดังนั้นการพิจารณาเงื่อนไขตามขั้นตอนนี้สามารถแสดงผลได้ดังตารางที่ 8



รูปที่ 2.8 แผนภาพสเตทแมชชีนตัวอย่าง

ตารางที่ 2.8 การเปลี่ยนสถานะของแผนภาพสเตทแมชชีนตัวอย่าง

สถานะแม่	สถานะปัจจุบัน	เงื่อนไขเหตุการณ์				สถานะเป้าหมาย
		C1	C2	.....	Cn	
-	S1	t	@T	-	-	S3
S2	S3	@F	-	-	F	S4

จากตารางที่ 8 สามารถอธิบายความหมายในแต่ละสดมภ์ (Column) ได้ดังต่อไปนี้

- สดมภ์แรก คือ สถานะแม่ (Parent state) แทนสถานะที่มีสถานะย่อยภายโดยสถานะย่อยหมายถึงสถานะปัจจุบันที่กำลังพิจารณาอยู่ กรณีไม่มีสถานะย่อยภายในจะใช้สัญลักษณ์ “-”
- สดมภ์ที่สอง คือ สถานะปัจจุบัน (Current state) เป็นสถานะที่กำลังพิจารณาในแต่ละทรานสิชันที่มีเหตุการณ์เข้ามากระตุ้น
- สดมภ์ที่สาม คือ เงื่อนไขเหตุการณ์ (Conditioned event) ซึ่งประกอบไปด้วยเหตุการณ์กระตุ้นและเงื่อนไขการ์ด
- สดมภ์ที่สี่ คือ สถานะเป้าหมาย (Target state) คือ สถานะที่คาดว่าจะเกิดขึ้นเมื่อมีเหตุการณ์เข้ามากระตุ้นซึ่งทำให้เปลี่ยนสถานะหรือยังคงอยู่สถานะเดิม

ในส่วนของแต่ละแถวของตารางเป็นการกำหนดค่าเงื่อนไขของเหตุการณ์ที่ทำหน้าที่กระตุ้นซึ่งทำให้เกิดการเปลี่ยนแปลงจากสถานะปัจจุบันไปยังสถานะเป้าหมาย โดยแถวที่มีการใส่สัญลักษณ์ @T(C) หรือ @F(C) ที่อยู่ภายใต้สดมภ์ของเงื่อนไขเหตุการณ์ จะเป็นเหตุการณ์ที่มีการกระตุ้น และแถวที่มีการใส่ค่า t หรือ f แทนค่าของเงื่อนไขการ์ด ดังนั้นข้อมูลที่แสดงในตารางที่ 2.8 มีความหมายคือ

ประพจน์ที่ 1 (P1) มีเหตุการณ์กระตุ้น C2 และมีเงื่อนไขการ์ดคือ C1 เมื่อใช้กฎการกระจายจะได้ว่า

$$\boldsymbol{@T(C2)} \wedge \boldsymbol{C1} = \sim C2 \wedge C2' \wedge C1$$

ประพจน์ที่ 2 (P2) มีเหตุการณ์กระตุ้น C1 และมีเงื่อนไขการ์ดคือ Cn เมื่อใช้กฎการกระจายจะได้ว่า

$$\boldsymbol{@T(C1)} \wedge \boldsymbol{Cn} = \sim C1 \wedge C1' \wedge Cn$$

เมื่อทำการขยายกฎครบทุกประพจน์แล้ว นำค่าที่ได้ไปสร้างความต้องการของกรณีทดสอบดังตารางที่ 2.9

ตารางที่ 2.9 ข้อกำหนดรายละเอียดประพจน์ที่ทำการขยายตัวกระตุ้นเหตุการณ์

หมายเลขประพจน์	สถานะแม่	สถานะปัจจุบัน	เงื่อนไขเหตุการณ์	สถานะเป้าหมาย
P1	-	S1	$\sim C2 \wedge C2' \wedge C1$	S3
P2	S2	S3	$\sim C1 \wedge C1' \wedge Cn$	S4

1) พิจารณาความต้องการของกรณีทดสอบในแต่ละประพจน์เพื่อนำไปสร้างกรณีทดสอบ

ขั้นตอนนี้เป็นการสร้างตารางค่าความจริงโดยอาศัยหลักการครอบคลุมประพจน์ที่ได้กล่าวไว้แล้วในข้างต้น โดยนำเอารายละเอียดประพจน์ที่ได้ในขั้นแรกมาสร้างเป็นตารางค่าความจริง ดังตารางที่ 2.10

ตารางที่ 2.10 ความต้องการของกรณีทดสอบของประพจน์ที่ 1 (P1)

หมายเลขประพจน์	สถานะแม่	สถานะปัจจุบัน	C1	C2	.....	Cn	เหตุการณ์หลังการกระตุ้น	สถานะเป้าหมาย
P1	-	S1	t	F			$C2'$=True	S3
		S1	t	T			$C2'$=True	S1
		S1	f	F			$C2'$=True	S1
		S1	t	F			$C2'$=False	S1

2.2.4. Risk Assessment Framework based on Goal-oriented Requirements Engineering and Object Behavioral Model [6]

งานวิจัยนี้นำเสนอกรอบงานสำหรับการประเมินความเสี่ยงในระหว่างขั้นตอนการวิเคราะห์และออกแบบซอฟต์แวร์ โดยระบุความเสี่ยงจากพฤติกรรมและปฏิสัมพันธ์ของวัตถุ ผลลัพธ์ที่ได้คือความเสี่ยงที่เกิดขึ้นกับแต่ละวัตถุ ความเสี่ยงในแต่ละลำดับเหตุการณ์ตลอดจนความเสี่ยงในระดับฟังก์ชันงานของซอฟต์แวร์

อนึ่งงานวิจัยฉบับนี้ได้นำแนวคิดจากงานวิจัย [5] มาเป็นแนวทางในการปฏิบัติ ดังนั้นในส่วนนี้จะกล่าวถึงแนวคิดที่ปรับปรุงจากเดิมเท่านั้น

2.2.4.1.การระบุปัจจัยความเสี่ยง

การระบุปัจจัยความเสี่ยงแบ่งออกเป็น 2 ขั้นตอนคือ

1) การระบุปัจจัยความเสี่ยงพฤติกรรมของวัตถุจากแผนภาพสเตทแมชชีน กล่าวคือ ปัจจัยความเสี่ยง (rf_i^x) ของวัตถุ i ในลำดับเหตุการณ์ที่ S_x มีค่าเท่ากับ

$$rf_i^x = PUOOS_i \cdot svt_i$$

โดยกำหนดให้ $PUOOS_i$ ที่มีค่าอยู่ในช่วง $0 \leq PUOOS_i \leq 1$ คือ ผลลัพธ์ที่ไม่พึงปรารถนาของวัตถุในลำดับเหตุการณ์ที่ S_x และ svt_i คือ ระดับความรุนแรงของวัตถุ i ในลำดับเหตุการณ์ที่ S_x ซึ่งค่าความน่าจะเป็นของผลลัพธ์ที่ไม่พึงประสงค์ของวัตถุในลำดับเหตุการณ์ถูกพิจารณาจากจำนวนกรณีทดสอบที่ถูกสร้างขึ้นจากเงื่อนไขในแต่ละการเปลี่ยนแปลงของสถานะในแผนภาพสเตทแมชชีน โดยที่กรณีทดสอบที่ถูกสร้างขึ้นได้นำหลักการครอบคลุมประพจน์ [9] ซึ่งทำให้มั่นใจได้ว่าจำนวนของกรณีทดสอบของแต่ละวัตถุนั้นไม่เท่ากัน ทั้งนี้ขึ้นกับความซับซ้อนของเงื่อนไขและจำนวนประโยคบนแต่ละประพจน์ของการเปลี่ยนแปลงสถานะ ดังนั้นความน่าจะเป็นของผลลัพธ์ที่พึงปรารถนาของวัตถุ i PSO_i (Probability of Satisfactory Outcome) ของการเปลี่ยนสถานะจาก p ไป q คำนวณได้จาก

$$PSO_{pq} = \frac{|TrueTestCases_{pq}|}{|TotalTestCases_{pq}|}$$

โดยกำหนดให้ $|TrueTestCases_{pq}|$ คือจำนวนกรณีทดสอบที่ถูกสร้างจากการเปลี่ยนแปลงจากสถานะ p ไป q และให้ผลลัพธ์ออกมาถูกต้อง ส่วน $|TotalTestCases_{pq}|$ คือจำนวนกรณีทดสอบทั้งหมดที่ถูกสร้างขึ้น

และเมื่อการเปลี่ยนสถานะของแผนภาพสเตทแมชชีน ข้อกำหนดรายละเอียดประพจน์ที่ทำการขยายตัวกระตุ้นเหตุการณ์ และความต้องการของกรณีทดสอบเป็นไปตามตารางที่ 2.11 – 2.13 ตามลำดับ จะสามารถคำนวณความน่าจะเป็นของผลลัพธ์ที่พึงปรารถนาได้เท่ากับ ¼ = 0.25

ตารางที่ 2.11 การเปลี่ยนสถานะของแผนภาพสเตทแมชชีน

Previous node	productInfo	validItem	updateComplete	Items=shipmentOrder	New node
Idle	@T	-	-	-	Validating Product
Validating Product	-	@T	-	-	Updating Inventory
Updating Inventory	-	-	@T	t	Updating FCSHost Server

ตารางที่ 2.12 ข้อกำหนดรายละเอียดประพจน์ที่ทำการขยายตัวกระตุ้นเหตุการณ์

Predicate Id	Pre-state	Expression	Post-state
P1	Idle	$\neg productInfo \wedge productInfo'$	Validating Product
P2	Validating Product	$\neg validItem \wedge validItem'$	Updating Inventory
P3	Updating Inventory	$\neg updateComplete \wedge updateComplete' \wedge (items = shipmentOrder)$	Updating FCSHost Server

ตารางที่ 2.13 ความต้องการของกรณีทดสอบ

Predicate Id	Pre-state	*updateComplete*	*items < shipmentOrder*	*updateComplete'*	Post-state
P3-1	Updating Inventory	F	T	T	Updating FCSHost Server
P3-2	Updating Inventory	T	T	T	Updating Inventory
P3-3	Updating Inventory	F	F	T	Updating Inventory
P3-4	Updating Inventory	F	T	F	Updating Inventory

เมื่อคำนวณความน่าจะเป็นของผลลัพธ์ที่พึงปรารถนาของแต่ละการเปลี่ยนแปลงของสถานะในแผนภาพสเตทแมชชีนทั้งหมดแล้ว ขั้นตอนต่อไปคือการคำนวณหาค่าความน่าจะเป็นของเส้นทางที่พึงปรารถนาในแต่ละเส้นทางการทำงานภายในแผนภาพสเตทแมชชีน ซึ่งคำนวณได้จาก

$$PSO_{initial-final} = \prod_{i=1}^{n-1} PSO_{P_i P_{i+1}}$$

ดังนั้นค่าของความน่าจะเป็นของเส้นทางที่ไม่พึงปรารถนาจะได้มีค่าเท่ากับ

$$PUO_{initial-final} = 1 - PSO_{initial-final}$$

ขั้นตอนถัดไปคือการหาค่าความน่าจะเป็นของวัตถุ (Probability of Unsatisfactory Outcome: PUOO) ซึ่งคำนวณได้จากผลรวมของผลลัพธ์ที่ไม่พึงปรารถนาของเส้นทางการทำงานทุกเส้นทางในแผนภาพสเตทแมชชีน โดยมีการใช้ค่าน้ำหนักของแต่ละเส้นทางด้วยจำนวนเส้นทางที่เป็นไปได้ทั้งหมด ดังนี้

$$PUOO_{\mathrm{i}} = \sum_{k=1}^{m} \left[\left[\frac{PP_k}{\sum_{l=1}^{m} PP_l} \right] \cdot \left(PUO_{initial-final} \right)_k \right]$$

กำหนดให้

$PUOO_{\mathrm{i}}$ คือความน่าจะเป็นของผลลัพธ์ที่ไม่พึงปรารถนาของวัตถุ ที่มีจำนวนเส้นทางการทำงานในแผนภาพสเตทแมชชีนที่มีจำนวนเส้นทาง m เส้นทาง

PP_k คือจำนวนเส้นทางที่เป็นไปได้ทั้งหมดในเส้นทางการทำงาน (Possible Path: PP) ที่ k ในแผนภาพสเตทแมชชีน ซึ่งก็คือผลคูณของจำนวนค่าความจริงที่เป็นไปได้ทั้งหมดของแต่ละการเปลี่ยนแปลงของสถานะที่อยู่ในเส้นทางนั้น

$\sum_{l=1}^{m} PP_l$ คือผลรวมของจำนวนเส้นทางที่เป็นไปได้ทั้งหมดจำนวน m เส้นทางในแผนภาพสเตทแมชชีน

$\left(PUO_{initial-final} \right)_k$ คือความน่าจะเป็นของผลลัพธ์ที่ไม่พึงปรารถนาของเส้นทางการทำงานที่ k ในแผนภาพสเตทแมชชีน

ขั้นตอนสุดท้ายคือการปรับค่าความน่าจะเป็นของผลลัพธ์ที่ไม่พึงปรารถนาของวัตถุให้เข้ากับลำดับเหตุการณ์ที่ S_{x} ใดๆ (Probability of Unsatisfactory Outcome of an Object in a Scenario: $PUOOS_i$) ดังสูตร

$$PUOOS_{\mathrm{i}} = \frac{\mathrm{PPO_i}}{\sum_{\mathrm{j=1}}^{\mathrm{n}} \mathrm{PPO_j}} \cdot PUOO_{\mathrm{i}}$$ ซึ่ง *i* เป็นสมาชิกในลำดับเหตุการณ์ที่ S_{x}

กำหนดให้

$\mathrm{PPO_i}$ คือจำนวนเส้นทางที่เป็นไปได้ทั้งหมดของวัตถุ i ซึ่งเกิดจากผลรวมของจำนวนเส้นทางที่เป็นไปได้ทั้งหมดในแต่ละเส้นทางการทำงานของแผนภาพสเตทแมชชีนของวัตถุ i

$\sum_{\mathrm{j=1}}^{\mathrm{n}} \mathrm{PPO_j}$ คือผลรวมของจำนวนเส้นทางที่เป็นไปได้ทั้งหมดของ n วัตถุในลำดับเหตุการณ์ที่ S_{x}

2) การพิจารณาความเสี่ยงของปฏิสัมพันธ์ระหว่างวัตถุ

ในส่วนของการคำนวณความเสี่ยงของปฏิสัมพันธ์ระหว่างวัตถุนี้ได้นำเอาแนวคิดจากงานวิจัย [5] มาคำนวณหาปัจจัยความเสี่ยงซึ่งได้กล่าวไว้แล้วในหัวข้อที่ 3.1 การวิเคราะห์ความเสี่ยงในระดับสถาปัตยกรรมโดยใช้ยูเอ็มแอล เรื่องการคำนวณหาค่าบรรทัดฐานคลัปปลิงของปฏิสัมพันธ์ระหว่างองค์ประกอบ

2.2.4.2.การวิเคราะห์ความเสี่ยงของลำดับเหตุการณ์

หลังจากการระบุความเสี่ยงให้กับพฤติกรรมและปฏิสัมพันธ์ของวัตถุแล้ว ขั้นตอนต่อไปคือการคำนวณหาค่าความเสี่ยงในระดับของเหตุการณ์ ซึ่งงานวิจัยนี้ได้นำเอาวิธีการสร้างแบบจำลองความเสี่ยงของพฤติกรรมการทำงานที่ได้กล่าวไว้ในหัวข้อที่ 3.1 การวิเคราะห์ความเสี่ยงในระดับสถาปัตยกรรมโดยใช้ยูเอ็มแอล เรื่องความเสี่ยงของลำดับเหตุการณ์

บทที่ 3

ขั้นตอนการประเมินความเสี่ยงของระบบจากฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่าง

งานวิจัยฉบับนี้เป็นการนำเสนอขั้นตอนวิธีสำหรับการประเมินความเสี่ยงของเหตุการณ์ที่มีความเกี่ยวข้องกับการเปลี่ยนแปลงข้อมูลในฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่าง โดยต่อไปจะใช้คำว่า การประเมินความเสี่ยงฯ ซึ่งได้นำเอาแนวคิดในการพิจารณาความเสี่ยงของลำดับเหตุการณ์ในซอฟต์แวร์ที่พิจารณาความเสี่ยงที่เกิดขึ้นกับองค์ประกอบและความเสี่ยงที่เกิดขึ้นกับปฏิสัมพันธ์ขององค์ประกอบเหล่านั้นด้วย [5] โดยนำเอาแบบจำลองสำหรับการประเมินความเสี่ยงที่ถูกนำเสนอไว้มาใช้กับการประเมินความเสี่ยงที่อาจเกิดจากการเปลี่ยนแปลงข้อมูลในฐานข้อมูล และงานวิจัยนี้แบ่งขั้นตอนการดำเนินการออกเป็น 9 ขั้นตอน มีรายละเอียดดังนี้

1. **กำหนดขอบเขตและปัญหาที่เกี่ยวเนื่องกับความผิดพลาดของข้อมูลในฐานข้อมูลที่สอดคล้องกับเงื่อนไขเชิงเค้าร่าง** เป็นขั้นตอนที่ผู้วิจัยได้หยิบยกปัญหาที่เกิดขึ้นในอุตสาหกรรมการพัฒนาซอฟต์แวร์มาสร้างรายการความต้องการ กำหนดสมมุติฐานและขอบเขตการแก้ปัญหาให้เกิดความชัดเจนในระหว่างการทำวิจัย และได้กล่าวถึงรายละเอียดดังกล่าวในบทที่หนึ่ง

2. **ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง** หลังจากการกำหนดปัญหาที่ชัดเจนแล้วขั้นตอนต่อไปคือการศึกษาทฤษฎีและงานวิจัยที่สามารถนำมาใช้ประโยชน์ ทั้งนี้เพื่อให้ผู้ทำวิจัยมีความเข้าใจในคุณสมบัติของสิ่งที่นำมาใช้ นอกจากนี้การนำเอาความรู้จากงานวิจัยที่มีความเกี่ยวข้องมาใช้นั้น ยังเป็นการยกระดับความเชื่อมั่นของผลลัพธ์ที่ได้จากงานวิจัยฉบับนี้อีกด้วย ซึ่งรายละเอียดต่างๆได้กล่าวไว้ในบทที่สอง

3. **การวิเคราะห์และออกแบบขั้นตอนวิธีการประเมินความเสี่ยง** ขั้นตอนนี้เป็นการประยุกต์องค์ความรู้จากการศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง ซึ่งเมื่อผู้วิจัยมีความรู้ความเข้าใจเกี่ยวกับสิ่งที่พิจารณาแล้ว จึงสามารถผสมผสานองค์ความรู้ที่ได้และปรับปรุงรายละเอียดการทำงานให้เกิดประสิทธิภาพสูงสุด และสามารถปรับปรุงแนวคิดที่ทำการศึกษาบนพื้นฐานของความถูกต้องและเหมาะสม

4. **กำหนดขั้นตอนวิธีสำหรับการประเมินความเสี่ยง** ขั้นตอนนี้ถือเป็นหัวใจสำคัญของงานวิจัยซึ่งเป็นส่วนที่แสดงถึงขั้นตอนการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีนที่ผู้วิจัยได้นำเสนอ โดยกล่าวไว้ในบทที่สามนี้

5. **ตรวจสอบขั้นตอนวิธีสำหรับการประเมินความเสี่ยง** ขั้นตอนนี้เป็นการตรวจสอบแบบจำลองหรือทฤษฎีที่นำมาใช้ในแต่ละขั้นตอนการประเมินความเสี่ยงในด้านความถูกต้องและข้อจำกัดของแบบจำลองและทฤษฎีที่นำมาใช้

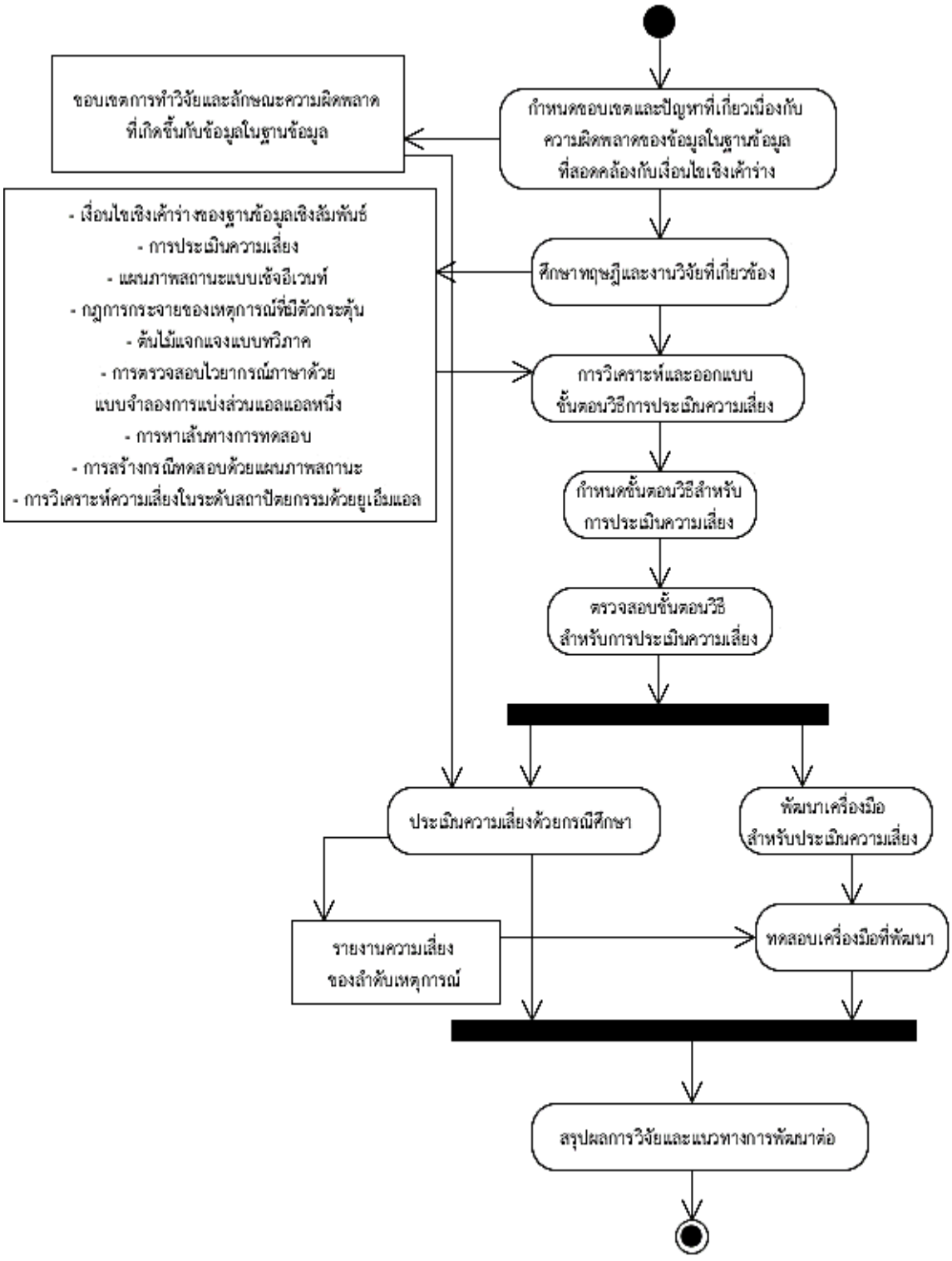
6. **ประเมินความเสี่ยงด้วยกรณีศึกษา** ในส่วนนี้เป็นการประเมินความเสี่ยงฯของกรณีศึกษาด้วยขั้นตอนวิธีที่นำเสนอ เพื่อพิจารณาความสอดคล้องระหว่างผลลัพธ์ที่ได้จากขั้นตอนวิธีและความเห็นของผู้เชี่ยวชาญในกรณีศึกษา รายละเอียดดังปรากฏไว้ในบทที่สี่

7. **พัฒนาเครื่องมือสำหรับประเมินความเสี่ยง** เมื่อขั้นตอนวิธีการประเมินความเสี่ยงฯได้ผ่านการตรวจสอบจากผู้เชี่ยวชาญแล้ว ผู้วิจัยจึงมีแนวคิดในการพัฒนาเครื่องมือเพื่ออำนวยความสะดวกแก่ผู้ที่ต้องการประเมินความเสี่ยงกับซอฟต์แวร์ที่กำลังพัฒนาในช่วงการออกแบบ นอกจากนี้ผู้วิจัยได้จัดทำเอกสารการออกแบบการพัฒนาเครื่องมือดังกล่าวด้วยแผนภาพยูเอ็มแอลอันประกอบด้วยแผนภาพยูสเคส คำอธิบายยูสเคส แผนภาพคลาส และแผนภาพองค์ประกอบ และเนื่องจากเครื่องมือที่พัฒนาได้จัดทำในลักษณะของเว็บเบสแอปพลิเคชัน ดังนั้นผู้วิจัยจึงนำเสนอแผนภาพความสัมพันธ์ของเอ็นทิตี้ (Entity Relationship diagram) เพื่อเป็นประโยชน์สำหรับผู้ที่ต้องการนำเอาเครื่องมือไปพัฒนาต่อยอด ซึ่งรายละเอียดต่างจะกล่าวไว้ในบทที่ห้า

8. **ทดสอบเครื่องมือที่พิจารณา** ในระหว่างการพัฒนาเครื่องมือตลอดจนการพัฒนาเครื่องมือแล้วเสร็จ ผู้วิจัยมีกระบวนการในการตรวจสอบการทำงานของเครื่องมือเปรียบเทียบกับผลลัพธ์ที่ได้จากกรณีศึกษาการประเมินความเสี่ยงด้วยการทดสอบระดับในหน่วยย่อย (Unit testing) และการทดสอบการทำงานร่วมกัน (Integration testing)

9. **สรุปผลการวิจัยและแนวทางการพัฒนาต่อ** ขั้นตอนนี้เป็นขั้นตอนสุดท้ายในกระบวนการวิจัยการประเมินความเสี่ยงฯ โดยเป็นการกล่าวถึงบทสรุปของงานวิจัย และผลลัพธ์ที่ได้เมื่อเปรียบเทียบกับสมมุติฐานที่กำหนดขึ้น นอกจากนี้ยังนำเสนอในส่วนของข้อจำกัดและแนวทางการพัฒนาขั้นตอนวิธีการประเมินความเสี่ยง รายละเอียดดังกล่าวแสดงไว้ในบทที่หก

จากการอธิบายกระบวนที่ใช้สำหรับพัฒนาขั้นตอนวิธีการประเมินความเสี่ยงในข้างต้นสามารถแสดงภาพรวมการทำวิจัยด้วยแผนภาพกิจกรรม ดังรูปที่ 3.1



รูปที่ 3.1 ขั้นตอนวิธีวิจัยการประเมินความเสี่ยงฯ

3.1. การวิเคราะห์และออกแบบขั้นตอนวิธีการประเมินความเสี่ยง

หลังจากขั้นตอนการศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องนั้น ผู้วิจัยได้สกัดเอาองค์ความรู้ที่จะสามารถนำมาใช้ประโยชน์ในการแก้ปัญหาให้เป็นไปตามสมมุติฐานที่กำหนดขึ้น และสิ่งที่ดำเนินการศึกษาประกอบด้วย เงื่อนไขเชิงเค้าร่างของฐานข้อมูลเชิงสัมพันธ์ การประเมินความเสี่ยง แผนภาพสเตทแมชชีนแบบเช้จอีเวนท์ กฎการกระจายของเหตุการณ์ที่มีตัวกระตุ้น ต้นไม้แจกแจงแบบทวิภาค การตรวจสอบไวยากรณ์ภาษาด้วยแบบจำลองการแบ่งส่วนแอลแอลหนึ่ง การหาเส้นทางการทดสอบ การสร้างกรณีทดสอบด้วยแผนภาพสเตทแมชชีน การวิเคราะห์ความเสี่ยงในระดับสถาปัตยกรรมด้วยยูเอ็มแอล มีรายละเอียดดังต่อไปนี้

1) เงื่อนไขเชิงเค้าร่างของฐานข้อมูลเชิงสัมพันธ์ (Relational database constraints)

องค์ความรู้ที่ได้จากการศึกษาและวิเคราะห์เงื่อนไขเชิงเค้าร่างนี้ ถูกนำมาใช้เพื่อกำหนดลักษณะและสาเหตุของข้อผิดพลาดที่ทำให้การเปลี่ยนแปลงของข้อมูลเกิดข้อผิดพลาดขึ้นได้ โดยแบ่งเงื่อนไขออกเป็นสี่เงื่อนไขคือ เงื่อนไขของค่าว่าง เงื่อนไขของโดเมน เงื่อนไขของคีย์หลัก และเงื่อนไขความสอดคล้องของข้อมูล

2) การประเมินความเสี่ยง (Risk assessment)

การศึกษาการประเมินความเสี่ยงที่เป็นส่วนหนึ่งในการบริหารจัดการความเสี่ยงนั้น ทำให้ผู้วิจัยได้รู้ถึงวัตถุประสงค์หลักในการประเมินความเสี่ยงซึ่งก็คือ การทำให้ผู้ประเมินทราบถึงภัยหรือความผิดพลาดที่แฝงอยู่ในสิ่งที่ทำการประเมินและทราบถึงผลกระทบที่เกิดขึ้นจากความผิดพลาดนั้นๆ ซึ่งผลลัพธ์ที่ได้จากการประเมินความเสี่ยงนี้จะทำให้ผู้ประเมินสามารถนำเอาผลลัพธ์นั้นไปจัดลำดับความสำคัญในขั้นตอนการควบคุมความเสี่ยงต่อไปได้ ดังนั้นผู้วิจัยจึงสรุปได้ว่าแนวทางในการแก้ปัญหาที่สอดคล้องกับรายการความต้องการคือ การประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน ซึ่งจะทำให้ผู้ใช้ขั้นตอนวิธีที่นำเสนอทราบถึงภัยหรือเหตุผิดพลาดที่แฝงอยู่ในซอฟต์แวร์ที่มีความเกี่ยวข้องกับความผิดพลาดของข้อมูลที่ทำการจัดเก็บลงฐานข้อมูลเชิงสัมพันธ์ได้ในระหว่างการออกแบบซอฟต์แวร์เพื่อใช้ในขั้นตอนการบริหารความเสี่ยงต่อไปได้

3) แผนภาพสเตทแมชชีนแบบเช้จอีเวนท์ (Change event state machine)

โดยปกติแล้วแผนภาพสเตทแมชชีนมีวัตถุประสงค์ในการนำเสนอสถานะของระบบหรือวัตถุที่พิจารณา ซึ่งมีการเปลี่ยนแปลงไปตามเหตุการณ์ที่มากระตุ้น และแผนภาพสเตทแมชชีนยังถูกแบ่งประเภทของแผนภาพออกเป็นหลายแบบ อาทิเช่น คอลอีเวนท์ ไทม์อีเวนท์ เช้จอีเวนท์ เป็นต้น แต่สำหรับงานวิจัยฉบับนี้นำเอาแผนภาพแบบเช้จอีเวนท์เป็นข้อมูลเข้าสำหรับการประเมิน ด้วยคุณสมบัติที่สถานะมีการตอบสนองต่อเหตุการณ์ที่เป็นตัวกระตุ้นพร้อมกับพิจารณาเงื่อนไขซึ่งเป็นส่วนที่ทำให้เกิดการเปลี่ยนแปลงในนิพจน์บูลลีน และการศึกษาคุณสมบัติของแผนภาพสเตทแมชชีนนี้เอง

ทำให้ผู้วิจัยนำเอาแผนภาพสเตทแมชชีนมาใช้กับการแสดงผลสถานะของข้อมูลที่ต้องการจัดเก็บลงฐานข้อมูลเชิงสัมพันธ์ให้เป็นไปตามเงื่อนไขเชิงเค้าร่าง

4) กฎการกระจายของเหตุการณ์ที่มีตัวกระตุ้น (Triggering event expansion rule)

เมื่อพบว่าแผนภาพสเตทแมชชีนเป็นแผนภาพที่เหมาะสมกับการประเมินความเสี่ยงของข้อมูลที่ถูกจัดเก็บลงฐานข้อมูลนั้น สิ่งถัดไปที่ต้องพิจารณาคือเงื่อนไขของการเปลี่ยนสถานะหรือที่เรียกว่า “ทรานซิชัน” ที่มีลักษณะเป็นนิพจน์บูลลีน และกฎการกระจายของเหตุการณ์ที่มีตัวกระตุ้นนี้สามารถทำได้โดยการกระจายประพจน์

5) ต้นไม้แจกแจงแบบทวิภาค (Parsing binary tree)

หลังจากการกระจายประพจน์ของทรานซิชันเรียบร้อยแล้ว การพิจารณาในลำดับต่อไปคือการหาค่าความจริงของนิพจน์ดังกล่าวด้วยการนำนิพจน์เหล่านั้นมาสร้างเป็นต้นไม้ทวิภาคเพื่อนำไปใช้ในขั้นตอนของการสร้างกรณีทดสอบ

6) การตรวจสอบไวยากรณ์ภาษาด้วยแบบจำลองการแบ่งส่วนแอลแอลหนึ่ง (LL(1) Parsing)

ในขั้นตอนการสร้างต้นไม้ทวิภาคในข้างต้นเป็นการสร้างต้นไม้จากนิพจน์บูลลีนซึ่งมีคุณสมบัติความไม่เท่ากันของลำดับความสำคัญของตัวดำเนินการ (Operator precedence) และมีการเรียงลำดับจากซ้ายไปขวา (Left Associativity) สำหรับตัวดำเนินการบูลลีนมีการเรียงลำดับความสำคัญคือ NOT, AND และ OR ซึ่งแบบจำลองการแบ่งส่วนนี้เองที่เป็นเครื่องมือที่ช่วยให้ได้ต้นไม้แบบทวิภาคที่ถูกต้องตามลำดับความสำคัญ และงานวิจัยฉบับนี้ ได้กำหนดไวยากรณ์ภาษาของนิพจน์บูลลีนมีรายละเอียดดังต่อไปนี้

G = (NT, T, P, S)

กำหนดให้

- G คือ ไวยากรณ์ภาษาในลักษณะที่ไม่พิจารณาองค์ประกอบร่วม (Context Free Grammar)
- NT คือ กลุ่มของสัญลักษณ์นอลเทอมินัล
- T คือ กลุ่มของสัญลักษณ์เทอมินัล
- P คือ กลุ่มของกฎการสร้างผลิตภัณฑ์
- S คือ สัญลักษณ์เริ่มต้นของไวยากรณ์

NT= { expression, termtail, term, factortail, factor, andop, orop }

T= { OPERAND, NOT, AND, OR, (,) }

P= {

product --> expression EOL

expression --> term termtail

termtail --> orop term termtail

termtail --> EPSILON

term --> factor factortail

factortail --> andop term tail

factortail --> EPSILON

factor --> OPERAND

factor --> NOT factor

factor --> (expression)

andop --> AND

orop --> OR

}

S = expression

จากการกำหนดไวยากรณ์ภาษาในข้างต้นมีการนำเอาคำศัพท์ทั้งแบบที่เป็นที่รู้จักและแบบที่ไม่คุ้นเคย ซึ่งความหมายของคำศัพท์ต่างๆ แสดงรายละเอียดไว้ดังตารางที่ 3.1

ตารางที่ 3.1 คำอธิบายคำศัพท์ที่ปรากฏในไวยากรณ์ภาษา

ลำดับที่	คำศัพท์	ความหมาย
1	expression	นิพจน์
2	EOL	อักขระสุดท้ายของบรรทัด
3	EPSILON	เครื่องหมาย **ε** ใช้แทนการเลือกกฎระหว่างกฎที่มีตัวดำเนินการหรือกฎที่เป็นค่าว่าง
4	NOT	ตัวดำเนินการนิเสธ (~)
5	AND	ตัวดำเนินการและ (&)
6	OR	ตัวดำเนินการหรือ (|)
7	OPERAND	ตัวถูกดำเนินการ

7) การสร้างกรณีทดสอบด้วยแผนภาพสเตทแมชชีน

ในส่วนนี้เป็นการศึกษาถึงความเป็นไปได้ในการสร้างกรณีทดสอบจากแผนภาพสเตทแมชชีน ซึ่งเป็นการศึกษาจากงานวิจัยต้นแบบที่แสดงถึงขั้นตอนวิธีและประสิทธิผลของกรณีทดสอบที่สร้างขึ้น ดังนั้นผู้วิจัยจึงสรุปได้ว่าความเป็นไปได้ในการนำเอาขั้นตอนวิธีดังกล่าวมาเป็นส่วนหนึ่งในการหาความน่าจะเป็นเพื่อนำไปสู่ผลลัพธ์ที่ต้องการซึ่งก็คือความเสี่ยง

8) กรอบงานสำหรับการประเมินความเสี่ยงในระหว่างขั้นตอนการวิเคราะห์และออกแบบซอฟต์แวร์

กรอบงานที่ได้ทำการศึกษานี้ผู้วิจัยนำมาใช้เป็นแนวทางหลักในการดำเนินการวิจัย และได้ปรับปรุงวิธีการบางขั้นตอนด้วยแนวคิดหรือเครื่องมือที่ถูกนำเสนอขึ้นภายหลังการเผยแพร่งานวิจัยฉบับดังกล่าว อาทิเช่น การปรับปรุงขั้นตอนการหาเส้นทางการทำงานของซอฟต์แวร์ด้วยการนำวิธีการหาเส้นทางการทดสอบด้วยเทคนิคการหาเส้นทางการทดสอบแบบไพร์มพาธ เป็นต้น

9) การวิเคราะห์ความเสี่ยงในระดับสถาปัตยกรรมด้วยยูเอ็มแอล

งานวิจัยที่ได้ศึกษาฉบับนี้เป็นอีกฉบับที่มีความสำคัญเนื่องจากเป็นงานวิจัยที่นำเสนอการวิเคราะห์ความเสี่ยงของซอฟต์แวร์ในระดับสถาปัตยกรรม ซึ่งเป็นการวิเคราะห์ซอฟต์แวร์ในระดับบนและมีการนำเสนอการวิเคราะห์ความเสี่ยงจากแผนภาพสเตทแมชชีนที่มีวัตุประสงค์ในการแสดงค่าความซับซ้อนของซอฟต์แวร์ที่นำมาทดสอบ นอกจากนี้ยังพิจารณาความน่าจะเป็นของความผิดพลาดด้วยทฤษฎีห่วงโซ่ของมาร์คอฟ (Markov chain model) และสรุปเป็นเมทริกซ์ของความน่าจะเป็นของโอกาสที่ซอฟต์แวร์จะเกิดข้อผิดพลาดขึ้น [5]

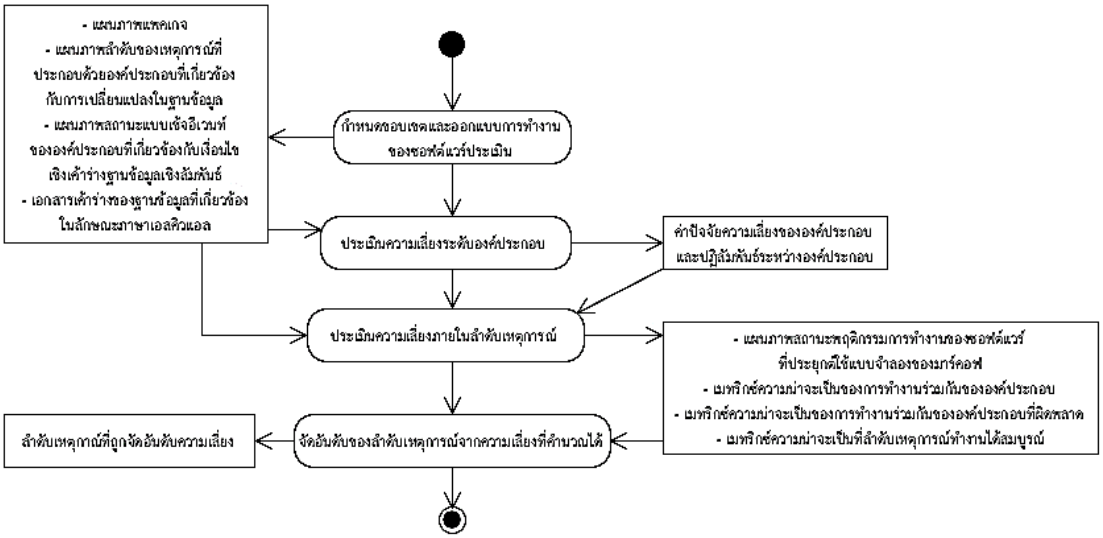
ดังนั้นผู้วิจัยจึงเล็งเห็นส่วนประโยชน์ที่นำเอาแนวคิดดังกล่าวมาปรับปรุงใช้กับการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน และประยุกต์ขั้นตอนบางขั้นตอนที่งานวิจัยที่ได้ศึกษานี้บางส่วน ให้มีความสอดคล้องกับปัญหาที่ได้กำหนดขึ้น

10) การหาเส้นทางการทดสอบ (Path testing)

จากงานวิจัยที่ได้ศึกษาในขั้นต้น ผู้ทำวิจัยพบว่าขั้นตอนหนึ่งในการคำนวณหาความน่าจะเป็นโดยการนำเอาเส้นทางการทำงานของซอฟต์แวร์ที่เป็นไปได้มาพิจารณานั้น ไม่มีการกล่าวถึงขั้นตอนวิธีที่ทำให้ได้มาซึ่งเส้นทางที่เป็นไปได้อย่างชัดเจน ดังนั้นผู้วิจัยจึงค้นคว้าวิธีการสร้างเส้นทางการทดสอบซึ่งก็หมายถึงเส้นทางการทำงานของซอฟต์แวร์ที่เป็นไปได้ด้วยแผนภาพสเตทแมชชีน ซึ่งพบว่ามีหลักการณ์ของการสร้างกรณีทดสอบจากกราฟ (Graph testing) ด้วยเทคนิคการสร้างกรณีทดสอบแบบไพร์ม (Prime path testing) ที่มีความเหมาะสมในการใช้งานร่วมกับแผนภาพสเตทแมชชีนที่มีคุณลักษณะแบบกราฟที่มีทิศทาง

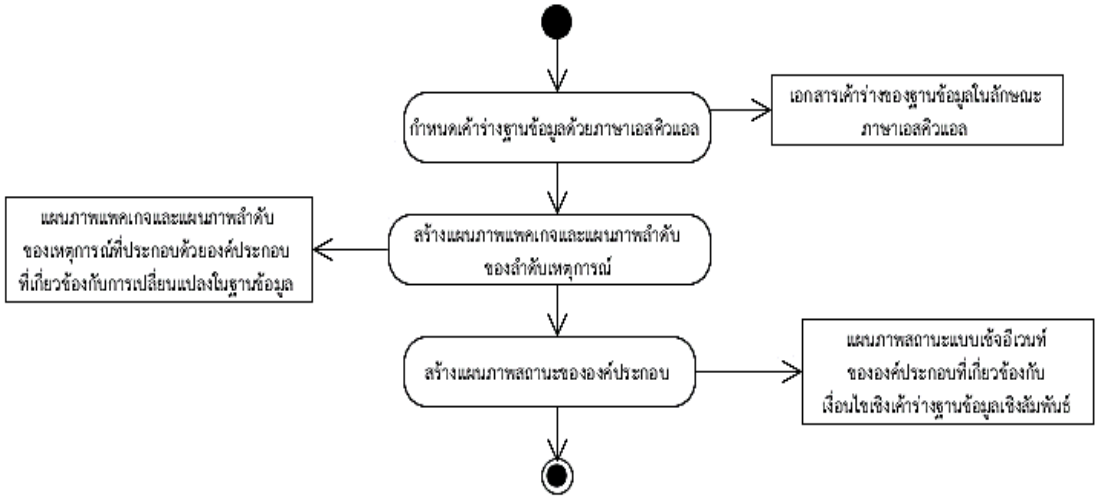
3.2. กำหนดขั้นตอนวิธีสำหรับการประเมินความเสี่ยง

ขั้นตอนวิธีสำหรับการประเมินความเสี่ยงที่ได้นำเสนอถูกแบ่งขั้นตอนการทำงานหลักออกเป็นสามขั้นตอนด้วยกันคือ การกำหนดขอบเขตและการทำงานของซอฟต์แวร์ในการประเมินความเสี่ยง การประเมินความเสี่ยงระดับองค์ประกอบ การประเมินความเสี่ยงภายในลำดับเหตุการณ์ สามารถแสดงภาพรวมการทำงานด้วยแผนภาพกิจกรรมดังรูปที่ 3.2

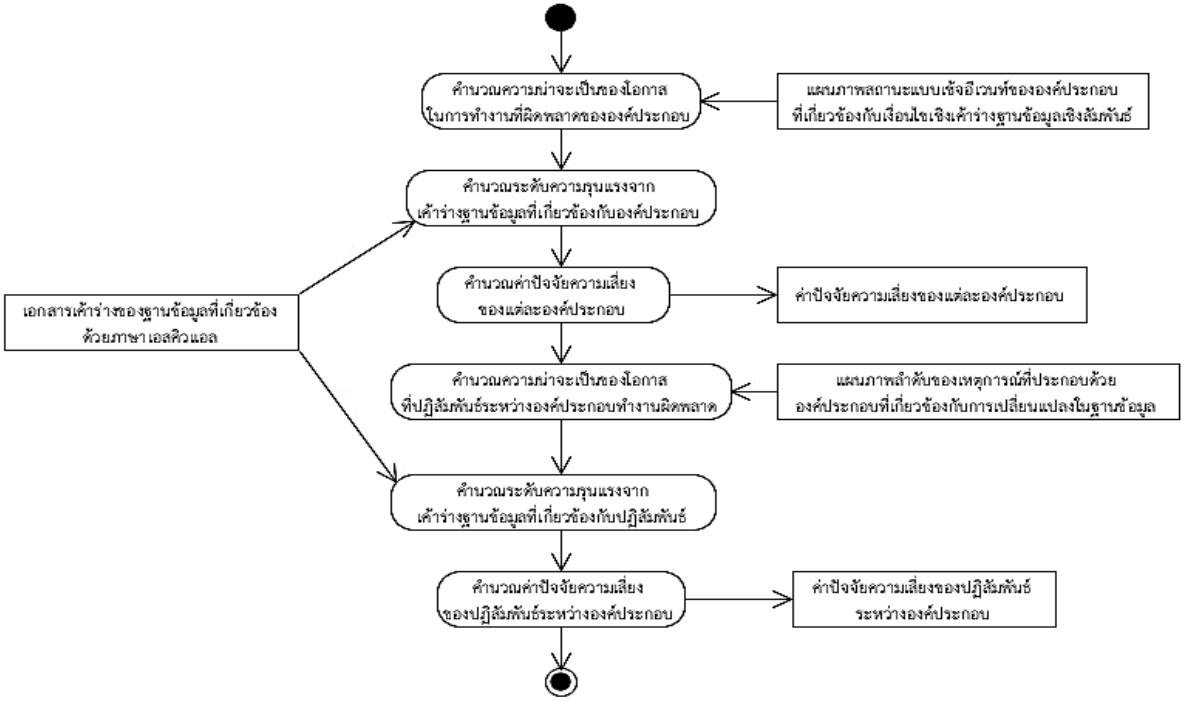


รูปที่ 3.2 ภาพรวมขั้นตอนการประเมินความเสี่ยงฯ

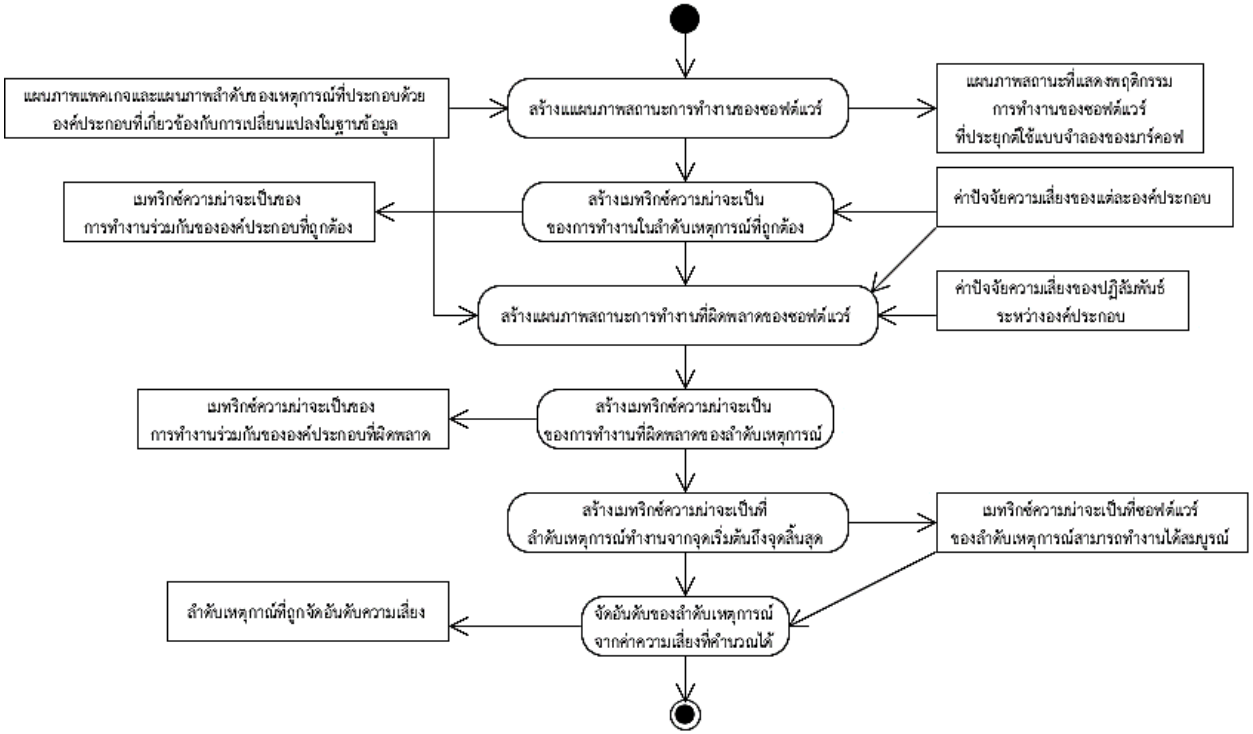
กิจกรรมที่ปรากฏในรูปที่ 3.2 แต่ละกิจกรรม ประกอบไปด้วยขั้นตอนย่อยที่ทำหน้าที่จัดการข้อมูลนำเข้าจากขั้นตอนลำดับก่อนหน้า และส่งต่อผลลัพธ์ให้กับลำดับกิจกรรมถัดไป สามารถแสดงรายละเอียดของขั้นตอนต่างๆได้ดังรูปที่ 3.3 - 3.5



รูปที่ 3.3 กิจกรรมการกำหนดขอบเขตและการออกแบบการทำงานของซอฟต์แวร์



รูปที่ 3.4 กิจกรรมการประเมินความเสี่ยงระดับองค์ประกอบ



รูปที่ 3.5 กิจกรรมการประเมินความเสี่ยงในลำดับเหตุการณ์

จากแผนภาพกิจกรรมที่แสดงถึงขั้นตอนวิธีการประเมินความเสี่ยงนั้น สามารถนำเสนอในรูปแบบของอัลกอริธึม (Algorithm) ได้ดังรูปที่ 3.6

ขั้นตอนวิธี การประเมินความเสี่ยงของแบบจำลองเชิงพฤติกรรม

ข้อมูลนำเข้า : แผนภาพลำดับ แผนภาพสถานะและแผนภาพความสัมพันธ์ของฐานข้อมูลในรูปแบบแฟ้มเอ็กซ์เอ็มแอล

ข้อมูลนำออก : รายงานระดับความเสี่ยงในระดับองค์ประกอบ การเชื่อมโยงระหว่างองค์ประกอบ และปัจจัยความเสี่ยงรวมในแต่ละลำดับเหตุการณ์

ข้อกำหนด : แผนภาพสถานะต้องเป็นแบบเช้จอีเวนท์ และแผนภาพความสัมพันธ์ของเอ็นทิตี้

เงื่อนไขก่อน : แผนภาพที่ออกแบบต้องมีความถูกต้องตามวากยสัมพันธ์

สำหรับแต่ละลำดับเหตุการณ์

สำหรับแต่ละองค์ประกอบ

คำนวณความน่าจะเป็นที่ไม่พึงปรารถนาของแต่ละองค์ประกอบ

คำนวณระดับความรุนแรงจากระดับผลกระทบในแต่ละตารางในฐานข้อมูล

คำนวณค่าปัจจัยความเสี่ยงขององค์ประกอบ

สำหรับแต่ละปฏิสัมพันธ์ขององค์ประกอบ

คำนวณความน่าจะเป็นที่ไม่พึงปรารถนาของแต่ละการเชื่อมโยงขององค์ประกอบ

คำนวณระดับความรุนแรงจากจำนวนความสัมพันธ์ระหว่างตารางในฐานข้อมูลที่เกี่ยวข้องกับปฏิสัมพันธ์ขององค์ประกอบ

คำนวณค่าปัจจัยความเสี่ยงปฏิสัมพันธ์ขององค์ประกอบ

สร้างแบบจำลองแสดงพฤติกรรมของซอฟต์แวร์และสร้างแบบจำลองความเสี่ยงของลำดับเหตุการณ์

คำนวณค่าปัจจัยความเสี่ยงของลำดับเหตุการณ์

จัดเรียงลำดับเหตุการณ์ด้วยค่าปัจจัยความเสี่ยงของลำดับเหตุการณ์

เงื่อนไขหลัง : ผลลัพธ์ที่ได้สามารถนำไปสนับสนุนการบริหารความเสี่ยงในระดับของลำดับเหตุการณ์ได้

รูปที่ 3.6 อัลกอริทึมที่ใช้ในการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน

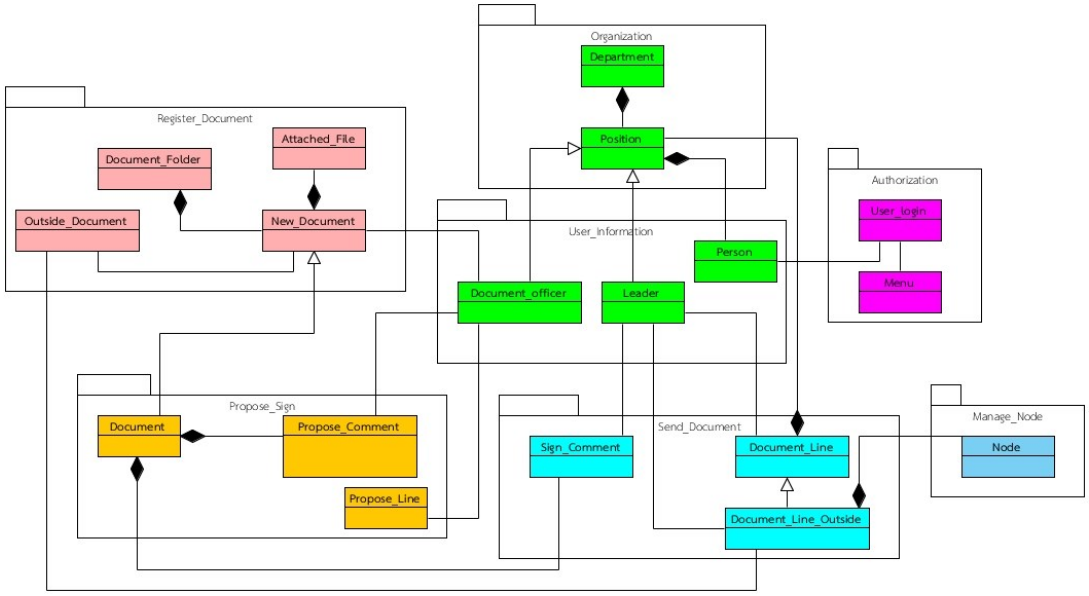
3.3. การกำหนดขอบเขตและการออกแบบการทำงานของซอฟต์แวร์

ขั้นตอนนี้เป็นการเตรียมข้อมูลนำเข้าที่ใช้ในกระบวนการประเมินความเสี่ยงฯ ที่ประกอบด้วยเอกสารเค้าร่างของฐานข้อมูลที่ถูกสร้างขึ้นด้วยภาษาเอสคิวแอล แผนภาพแพคเกจของคลาสที่ใช้ในการออกแบบซอฟต์แวร์ แผนภาพลำดับที่แสดงถึงลำดับการทำงานและปฏิสัมพันธ์ภายใน และสุดท้ายคือแผนภาพสเตทแมชชีนแบบเช้จอีเวนท์ที่แสดงถึงการเปลี่ยนแปลงสถานะขององค์ประกอบที่ขึ้นกับเหตุการณ์ที่เป็นตัวกระตุ้น (Trigger event) และเงื่อนไข (Condition) โดยแผนภาพยูเอ็มแอลเหล่านี้มีความสัมพันธ์และเกี่ยวเนื่องกันคือ แผนภาพแพคเกจเป็นการรวมกันของคลาสที่ใช้ในการออกแบบการทำงานของซอฟต์แวร์ที่แสดงถึงกลุ่มก้อนการทำงานหนึ่งๆ ในที่นี้ถูกเรียกว่า *“องค์ประกอบ”* (Component) และองค์ประกอบเหล่านี้ถูกนำมาใช้ในการสร้างแผนภาพลำดับในฐานะเส้นชีวิต (Lifeline) โดยมีจุดประสงค์ในการแสดงลำดับการทำงานและปฏิสัมพันธ์ของแพคเกจที่เป็นองค์ประกอบการทำงานภายในหนึ่งเหตุการณ์ของซอฟต์แวร์ ในที่นี้ถูกเรียกว่า *“ลำดับเหตุการณ์”* (Scenario) เมื่อสร้างแผนภาพลำดับเรียบร้อยแล้วจะนำเอาองค์ประกอบต่างมาพิจารณาสร้างแผนภาพสเตทแมชชีนแบบเช้จอีเวนท์ที่แสดงถึงกลไกการทำงานภายในที่เกี่ยวข้องกับการเปลี่ยนแปลงข้อมูลในฐานข้อมูลภายใต้เงื่อนไขเชิงเค้าร่างของฐานข้อมูลเชิงสัมพันธ์ สำหรับข้อมูลนำเข้าในข้างต้นแสดงตัวอย่างได้ดังรูปที่ 3.7 – 3.10 ตามลำดับ

```
CREATE TABLE IF NOT EXISTS `document` (
  `docID` bigint(19) NOT NULL AUTO_INCREMENT,
  `docCode` varchar(50) NOT NULL,
  `laststatus` tinyint(4) NOT NULL,
  `createDate` date NOT NULL,
  `subject` varchar(255) NOT NULL,
  `from` varchar(255) DEFAULT NULL,
  `to` bigint(19) DEFAULT NULL,
  `shortDesc` varchar(255) DEFAULT NULL,
  `deptOwner` varchar(255) DEFAULT NULL,
  `secretID` int(10) NOT NULL,
  `typeID` int(10) NOT NULL,
  `speedID` int(10) NOT NULL,
  `recSendTypeID` int(10) NOT NULL,
  `refdocID` bigint(19) NOT NULL,
  PRIMARY KEY (`docID`),
  UNIQUE KEY `docCode` (`docCode`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8
AUTO_INCREMENT=1 ;
```

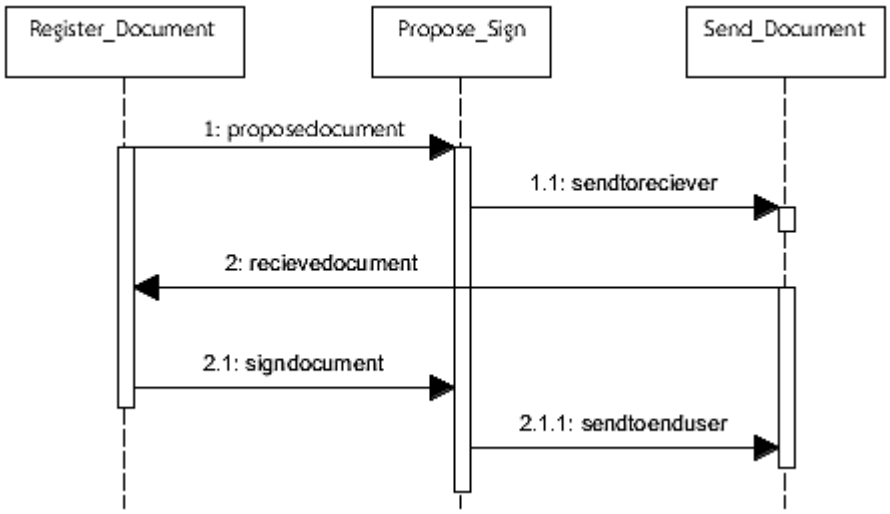
รูปที่ 3.7 ข้อมูลบางส่วนของเอกสารเค้าร่างฐานข้อมูลที่สร้างขึ้นด้วยภาษาเอสคิวแอล

ตัวอย่างแผนภาพแพคเกจในรูปที่ 3.8 เป็นการแสดงถึงการทำงานร่วมกันของคลาสที่ถูกออกแบบเพื่อการทำงานในซอฟต์แวร์ จากนั้นนำมาแบ่งกลุ่มเพื่อตอบสนองหน้าที่การทำงานเฉพาะด้าน โดยกลุ่มของคลาสเหล่านี้มีการเรียกใช้งานระหว่างกัน จนก่อเกิดเป็นลำดับเหตุการณ์ของซอฟต์แวร์ขึ้น



รูปที่ 3.8 แผนภาพแพคเกจของคลาสที่ใช้ในการออกแบบซอฟต์แวร์

จากรูปที่ 3.8 ที่แสดงถึงการเรียกใช้งานระหว่างองค์ประกอบของคลาสนั้น สามารถนำเอาการเรียกใช้งานระหว่างองค์ประกอบต่างๆ มาสร้างเป็นแผนภาพลำดับการทำงานของเหตุการณ์ที่เกิดขึ้นในซอฟต์แวร์ ดังตัวอย่างในรูปที่ 3.8 เป็นแผนภาพที่แสดงลำดับการทำงานของลำดับเหตุการณ์การส่งหนังสือถงผู้รับผิดชอบในซอฟต์แวร์ระบบสารบรรณ ด้วยการนำเอาองค์ประกอบที่ปรากฏในแผนภาพแพคเกจมาสร้างเป็นเส้นชีวิตในแผนภาพลำดับ และการเรียกใช้งานระหว่างองค์ประกอบเหล่านี้ในลำดับต่อไปผู้วิจัยจะเรียกว่า “การเชื่อมโยงระหว่างองค์ประกอบ”



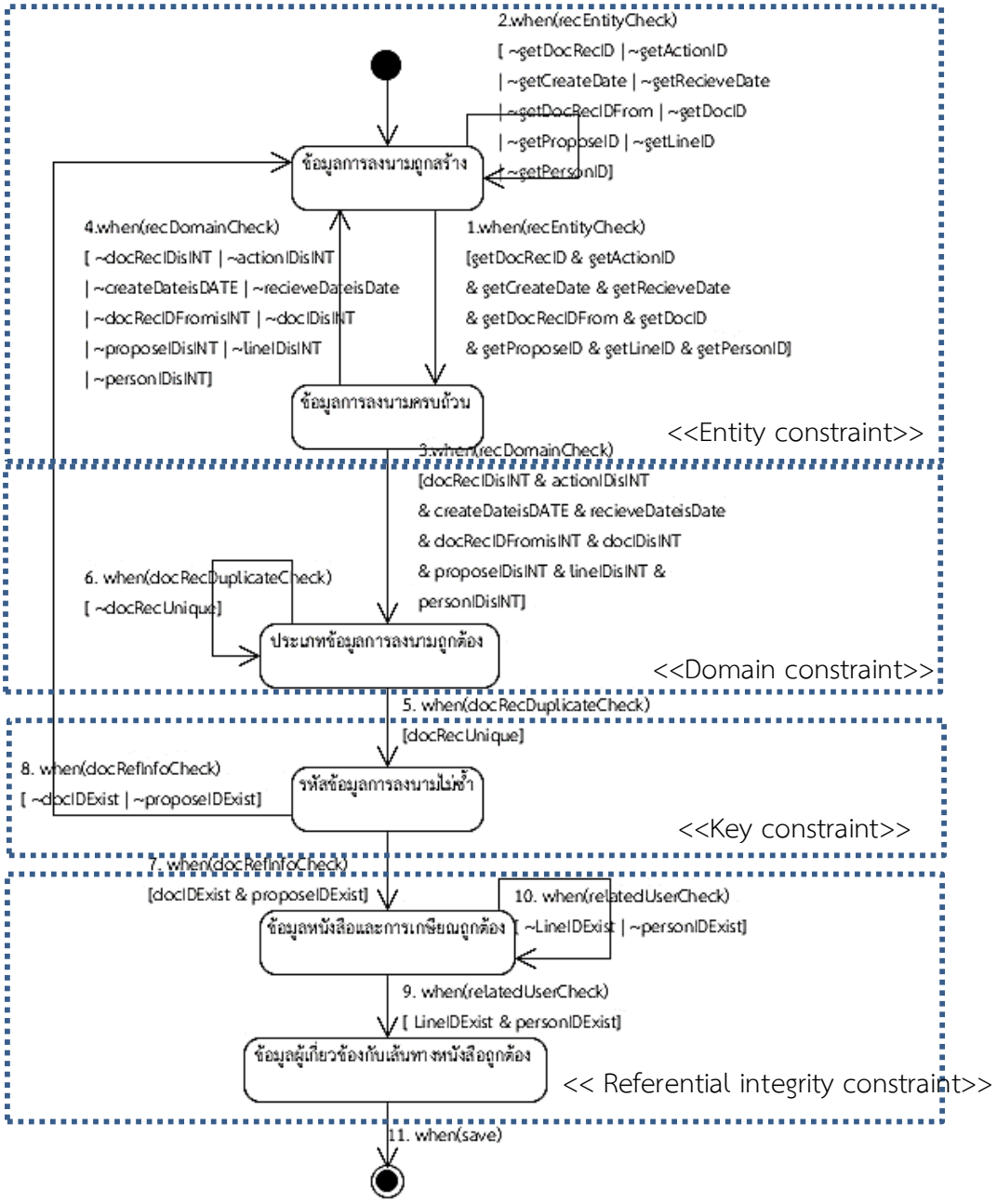
รูปที่ 3.9 แผนภาพลำดับที่ใช้ในการออกแบบซอฟต์แวร์

เมื่อพิจารณาลำดับเหตุการณ์ได้ดังรูปที่ 3.9 แล้ว จะเห็นได้ว่าการทำงานในระดับองค์ประกอบเต็มไปด้วยการทำงานที่ซับซ้อน จนอาจเป็นสาเหตุให้ข้อมูลที่ใช้ในการจัดเก็บลงฐานข้อมูลเกิดข้อผิดพลาด และไม่เป็นไปตามเงื่อนไขเชิงเค้าร่างของฐานข้อมูลเชิงสัมพันธ์ ดังนั้นในรูปที่ 3.10 เป็นการนำเสนอองค์ประกอบที่มีผลในการเปลี่ยนแปลงข้อมูลใดๆ ในฐานข้อมูลด้วยแผนภาพสเตทแมชชีน สำหรับวิธีการที่ใช้ในการออกแบบแผนภาพสเตทแมทชีนในขั้นตอนวิธีการประเมินความเสี่ยงที่นำเสนอ ดังนี้

1. แผนภาพสเตทแมทชีน เป็นแผนภาพที่นำเสนอกลไกการทำงานที่เกี่ยวข้องกับการบริหารจัดการข้อมูลตามเงื่อนไขเชิงเค้าร่าง ขององค์ประกอบที่ได้กำหนดขึ้น
2. สถานะ (State) ถูกสร้างขึ้นเพื่อนำเสนอสถานะของข้อมูลที่เป็นไปตามเงื่อนไขเชิงเค้าร่าง
3. การเปลี่ยนแปลงระหว่างสถานะ (Transition) นำเหตุการณ์สำหรับการตรวจสอบข้อมูลตามเงื่อนไขเชิงเค้าร่างมาใช้เป็นเหตุการณ์กระตุ้น และนำคุณลักษณะประจำตารางที่พิจารณาร่วมกับประเภทของเงื่อนไขเชิงเค้าร่าง เพื่อใช้กำหนดเงื่อนไขของเหตุการณ์กระตุ้น และจากตัวอย่างในรูปที่ 3.10 สามารถอธิบายความหมายได้พอสังเขป ดังนี้

- เงื่อนไขเชิงเค้าร่าง (Entity constraint) ในส่วนของเหตุการณ์กระตุ้นนำเอาการเหตุการณ์ที่เป็นการตรวจสอบค่าว่างของคุณลักษณะประจำตารางที่ไม่สามารถเป็นค่าว่างได้ ภายใต้ชื่อเหตุการณ์ recEntityCheck และเงื่อนไขประกอบเหตุการณ์กระตุ้นนี้คือ DocRecID ที่หมายถึงหมายเลขเอกสารที่ไม่ขาดคุณสมบัติประจำตารางนี้ได้ ภายใต้คำว่า getDocRecID เป็นต้น

- เงื่อนไขประเภทข้อมูล (Domain constraint) นำเอาเหตุการณ์ที่ชื่อ recDomainCheck มาเป็นเหตุการณ์กระตุ้นในการเปลี่ยนสถานะ และพิจารณาร่วมกับเงื่อนไขของเหตุการณ์กระตุ้น อาทิเช่น createDateisDATE ซึ่งหมายถึงการตรวจสอบว่าข้อมูลของวันที่ลงทะเบียนหนังสือมีรูปแบบเป็นวันที่ที่ถูกต้อง
- เงื่อนไขของคีย์หลัก (Primary Key) ในส่วนของเหตุการณ์กระตุ้นได้นำเอาเหตุการณ์ชื่อ docRecUnique มาเป็นเหตุการณ์กระตุ้นที่ทำให้เปลี่ยนจากสถานะ และพิจารณาร่วมกับเงื่อนไขชื่อ docIDunique ซึ่งหมายถึงค่าของ docID ต้องไม่ซ้ำกับค่าอื่นในตาราง
- เงื่อนไขความสอดคล้องของข้อมูล (Reference integrity constraint) ในส่วนของเหตุการณ์ที่กระตุ้นให้เกิดการเปลี่ยนแปลงของสถานะคือ relatedUserCheck ซึ่งเป็นเหตุการณ์ที่ตรวจสอบข้อมูลของผู้ใช้ที่อ้างอิงถึง ว่ามีปรากฏอยู่ในตารางอื่นหรือไม่ และพิจารณาร่วมกับเงื่อนไขชื่อ อาทิเช่น personIDExist ซึ่งหมายถึงค่าของ personID ที่อ้างอิง ต้องมีปรากฏอยู่ในตารางอื่น



รูปที่ 3.10 แผนภาพสเตทแมชชีนที่ใช้ในการออกแบบองค์ประกอบการเสนอลงนาม

3.4. การประเมินความเสี่ยงระดับองค์ประกอบ

หลังจากขั้นตอนการออกแบบแผนภาพที่ใช้ในการประเมินความเสี่ยงเรียบร้อยแล้ว ขั้นตอนต่อไปเป็นการประเมินความเสี่ยงระดับองค์ประกอบที่พิจารณาในสองส่วนคือ ความเสี่ยงขององค์ประกอบ ซึ่งหมายถึง ความน่าจะเป็นที่การทำงานภายในองค์ประกอบเกิดข้อผิดพลาดและพิจารณาร่วมกับระดับความรุนแรงที่จะส่งผลกระทบเมื่อองค์ประกอบนั้นทำงานผิดพลาด อีกส่วนหนึ่งคือความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบ ซึ่งหมายถึงความน่าจะเป็นที่การเชื่อมโยงกัน

ระหว่างองค์ประกอบเกิดข้อผิดพลาดและพิจารณาร่วมกับระดับความรุนแรงที่จะส่งผลกระทบเมื่อการเชื่อมโยงระหว่างองค์ประกอบเกิดข้อผิดพลาด ในขั้นตอนนี้สามารถแบ่งออกเป็นสองขั้นตอนด้วยกันคือ

1) การประเมินความเสี่ยงของแต่ละองค์ประกอบของลำดับเหตุการณ์จากแผนภาพสเตทแมชชีน

เมื่อกำหนดให้ $\boldsymbol{rf_i^x}$ คือปัจจัยความเสี่ยงขององค์ประกอบใดๆ ในลำดับเหตุการณ์ที่ $\boldsymbol{S_x}$ แล้วสามารถคำนวณค่าปัจจัยความเสี่ยงขององค์ประกอบได้จากแบบจำลอง

$$rf_i^x = PUOCS_i \cdot svt_i \quad (1)$$

กำหนดให้

- $PUOCS_i$ คือ ความน่าจะเป็นของผลลัพธ์ที่ไม่พึงปรารถนาขององค์ประกอบ
- svt_i คือระดับความรุนแรงที่เกิดขึ้นเมื่อเกิดผลลัพธ์ที่ไม่พึงปรารถนา และจากแบบจำลองในข้างต้นมีขั้นตอนในการคำนวณค่า $PUOCS_i$และค่า svt_i ดังนี้
- $\boldsymbol{x}$ คือลำดับเหตุการณ์ใดๆในซอฟต์แวร์

2) คำนวณความน่าจะเป็นของผลลัพธ์ที่พึงปรารถนาขององค์ประกอบ (Probability of satisfactory component: PSC) ของการเปลี่ยนสถานะจาก $\boldsymbol{p}$ ไป $\boldsymbol{q}$ ผ่านหลักการสร้างกรณีทดสอบแบบครอบคลุมประพจน์ที่จะทำให้ได้มาซึ่งจำนวนของกรณีทดสอบ ด้วยแบบจำลอง

$$PSC_{pq} = \frac{|TrueTestCases_{pq}|}{|TotalTestCases_{pq}|} \quad (2)$$

กำหนดให้

- $|TrueTestCases_{pq}|$ คือจำนวนกรณีทดสอบที่ถูกสร้างจากการเปลี่ยนแปลงจากสถานะ $\boldsymbol{p}$ ไป $\boldsymbol{q}$ และให้ผลลัพธ์ออกมาถูกต้อง
- $|TotalTestCases_{pq}|$ คือจำนวนกรณีทดสอบทั้งหมดที่ถูกสร้างขึ้นตามหลักการครอบคลุมประพจน์

3) คำนวณหาค่าความน่าจะเป็นของเส้นทางที่พึงปรารถนาในแต่ละเส้นทางการทำงานภายในแผนภาพสเตทแมชชีน ด้วยแบบจำลอง

$$PSC_{initial-final} = \prod_{i=1}^{n-1} PSC_{P_iP_{i+1}} \quad (3)$$

กำหนดให้

- $PSC_{initial-final}$ คือความน่าจะเป็นของเส้นทางที่พึงปรารถนาที่เกิดขึ้นภายในแผนภาพสเตทแมชชีนจากจุดเริ่มต้นของการทำงานไปยังจุดสิ้นสุดการทำงาน

– $PSC_{P_iP_{i+1}}$ คือความน่าจะเป็นของผลลัพธ์ที่พึงปรารถนาในแต่ละสถานะย่อย ซึ่งในที่นี้จะนำเอาความน่าจะเป็นจากสถานะต้นทาง (*i*) คูณกับความน่าจะเป็นของสถานะปลายทาง (*j*)

4) คำนวณหาค่าของความน่าจะเป็นของเส้นทางที่ไม่พึงปรารถนา ด้วยแบบจำลอง

$$PUC_{initial-final} = 1 - PSC_{initial-final} \quad (4)$$

กำหนดให้

– $PUC_{initial-final}$ คือค่าของความน่าจะเป็นของเส้นทางที่ไม่พึงปรารถนาที่มีความน่าจะเป็นสูงสุดเท่ากับ 1

5) คำนวณความน่าจะเป็นที่ไม่พึงปราถนาขององค์ประกอบ (Probability of Unsatisfactory Outcome: PUOC) ด้วยแบบจำลอง

$$PUOC_{\mathrm{i}} = \sum_{k=1}^{m} \left[\left[\frac{PP_k}{\sum_{l=1}^{m} PP_l} \right] \cdot \left(PUC_{initial-final} \right)_k \right] \quad (5)$$

กำหนดให้

– $PUOC_{\mathrm{i}}$ คือความน่าจะเป็นขององค์ประกอบแต่ละองค์ประกอบในลำดับเหตุการณ์

– m คือจำนวนเส้นทางที่เป็นไปได้ในแผนภาพสเตทแมชชีน

– k คือเส้นทางการทำงานในแผนภาพสเตทแมชชีน

– PP_k คือผลคูณของกรณีทดสอบที่เป็นไปได้ของแต่ละสถานะภายในของเส้นทางในแผนภาพ

– $\sum_{l=1}^{m} PP_l$ คือผลรวมของผลคูณของกรณีทดสอบที่เป็นไปได้ในแต่ละเส้นทางของแผนภาพ

– $\left(PUC_{initial-final} \right)_k$ คือความน่าจะเป็นของเส้นทางที่ไม่พึงปรารถนาในเส้นทางที่ k

6) ปรับค่าความน่าจะเป็นของผลลัพธ์ที่ไม่พึงปรารถนาขององค์ประกอบให้เข้ากับลำดับเหตุการณ์ที่ S_{x}

$$PUOCS_{\mathrm{i}} = \frac{\mathrm{PPC_i}}{\sum_{\mathrm{j=1}}^{\mathrm{n}} \mathrm{PPC_j}} \cdot PUOC_{\mathrm{i}} \quad (6)$$

กำหนดให้

– PPC_i คือจำนวนกรณีทดสอบทั้งหมดขององค์ประกอบลำดับที่ *i* ซึ่งเปรียบได้กับค่าความซับซ้อนขององค์ประกอบนั้นๆ

– $\sum_{\mathrm{j=1}}^{\mathrm{n}} \mathrm{PPC_j}$ คือผลรวมของกรณีทดสอบสำหรับทุกๆองค์ประกอบภายใต้ลำดับเหตุการณ์ที่ S_{x}

7) คำนวณระดับความรุนแรงจากจำนวนความสัมพันธ์ระหว่างตารางในฐานข้อมูลเชิงสัมพันธ์

ในส่วนนี้เป็นการอธิบายถึงวิธีในการคำนวณค่าระดับความรุนแรงขององค์ประกอบที่มีการทำงานเกี่ยวข้องกับตารางในฐานข้อมูล ซึ่งความรุนแรงที่เกิดขึ้นในแต่ละองค์ประกอบนั้นย่อมมีความแตกต่างกันอันเนื่องมาจากระดับความยากง่ายต่อการเกิดข้อผิดพลาดหรือความสัมพันธ์ระหว่างตารางในฐานข้อมูล มีรายละเอียดของการทำงานดังนี้

1. สกัดเอาคุณลักษณะประจำตาราง (Attribute) และชนิดของคุณลักษณะเพื่อใช้เป็นข้อมูลในการระบุระดับผลกระทบในการเกิดข้อผิดพลาดของผู้เชี่ยวชาญ ซึ่งมีแนวทางสำหรับการกำหนดระดับความรุนแรงนั้นให้พิจารณาผลกระทบของความผิดพลาดด้วยปัจจัยสามปัจจัยคือ ฟังก์ชันงาน จำนวนความสัมพันธ์ระหว่างตาราง และประเภทของเงื่อนไขเชิงเค้าร่าง ตามลำดับ ยกตัวอย่างเช่น คุณลักษณะประจำชื่อ docID ภายในตาราง Document ของระบบสารบรรณ ซึ่งเป็นคีย์หลักของตาราง หากนำไปเปรียบเทียบกับคุณลักษณะประจำ subject ภายในตารางเดียวกันนั้น ย่อมมีระดับผลกระทบที่รุนแรงกว่า อันเนื่องมาจากประเภทของเงื่อนไขเชิงเค้าร่าง แต่หากนำคีย์หลักจากตารางอื่นมาเปรียบเทียบ อาทิเช่น deptID ที่เป็นคีย์หลักของตาราง Department แล้ว docID ย่อมมีระดับผลกระทบที่สูงกว่า อันเนื่องมาจากตาราง Document เป็นตารางที่ใช้เก็บข้อมูลตั้งต้นของฟังก์ชันงานหลายฟังก์ชันงานภายในระบบสารบรรณ และจำนวนตารางที่มีความสัมพันธ์กับตาราง Document มีจำนวนมากกว่าจำนวนตารางที่เกี่ยวข้องกับตาราง Document และระดับความผลกระทบเหล่านี้ถูกแบ่งออกเป็นสามระดับคือ

- ระดับสูง (High)
- ระดับกลาง (Medium)
- ระดับต่ำ (Low)

2. คำนวณหาระดับของผลกระทบของตารางที่เกี่ยวข้องกับองค์ประกอบที่พิจารณาด้วยแบบจำลอง

$$svt = \frac{\sum_{i=1}^{N} t_i}{T_{DB}} \quad (7)$$

กำหนดให้

svt คือ ค่าของระดับผลกระทบที่เกิดขึ้นกับตารางที่เกี่ยวข้องกับองค์ประกอบที่พิจารณา

t_i คือ ค่าระดับผลกระทบของตารางที่เกี่ยวข้องกับองค์ประกอบในลำดับที่ i

N คือ จำนวนตารางที่เกี่ยวข้องกับองค์ประกอบที่พิจารณา

T_{DB} คือ ค่าระดับผลกระทบที่เกิดขึ้นกับทุกๆตารางในฐานข้อมูล

3. คำนวณระดับผลกระทบที่เกิดขึ้นในแต่ละตาราง ด้วยระดับช่องโหว่ของการเกิดข้อผิดพลาดในแต่ละคุณลักษณะประจำตารางจากการประเมินของผู้เชี่ยวชาญ ด้วยแบบจำลอง

$$t_i = \sum_{j=1}^{n} W_j \quad (8)$$

กำหนดให้

t_i คือ ค่าระดับผลกระทบของตารางที่เกี่ยวข้องกับองค์ประกอบในลำดับที่ i

W_j คือ ระดับผลกระทบในแต่ละคุณลักษณะประจำของตารางที่ได้จากการประเมินของผู้เชี่ยวชาญ

n คือ จำนวนคุณลักษณะประจำของตารางลำดับที่ i

4. คำนวณระดับผลกระทบที่เกิดขึ้นกับฐานข้อมูลด้วยแบบจำลอง

$$T_{DB} = \sum_{k=1}^{N} T_k \quad (9)$$

กำหนดให้

T_{DB} คือ ค่าของภัยคุกคามระดับผลกระทบที่เกิดขึ้นกับทุกๆตารางในฐานข้อมูล

T_k คือ ค่าของระดับผลกระทบที่เกิดขึ้นกับตารางลำดับที่ k ในฐานข้อมูล DB

k คือ ลำดับของตารางภายใต้ฐานข้อมูล DB

N คือ จำนวนตารางภายใต้ฐานข้อมูล DB

3.5. ประเมินความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบของเหตุการณ์ใดๆ จากแผนภาพสเตทแมชชีน

ค่าความเสี่ยงของปฏิสัมพันธ์ฯ (rf_{ij}^{x}) ในส่วนนี้ได้อ้างอิงการคำนวณจากงานวิจัย [5] ซึ่งพิจารณาได้จากแบบจำลอง $rf_{ij}^{x} = EOC_{ij}^{x} \cdot svt_{ij}^{x}$ โดยที่ค่าEOC_{ij}^{x} คำนวณได้จากแบบจำลอง

$$EOC_{ij}^{x} = \frac{|MT_{ij}^{x}|_{i,j \epsilon S_x, i \neq j}}{|MT^{x}|} \quad (10)$$

กำหนดให้

EOC_{ij}^{x} คือ ความน่าจะเป็นของล้มเหลวที่เกิดขึ้นจากปฏิสัมพันธ์ระหว่างองค์ประกอบในลำดับเหตุการณ์ที่ $\boldsymbol{x}$

$|MT_{ij}^{x}|$ คือ จำนวนข้อความที่ส่งจากองค์ประกอบ i ถึงองค์ประกอบ j ในแผนภาพลำดับ

$|MT^{x}|$ คือ จำนวนข้อความทั้งหมดในแผนภาพลำดับ

สำหรับการคำนวณระดับความรุนแรงของปฏิสัมพันธ์ระหว่างองค์ประกอบ (svt_{ij}^{x}) กำหนดให้ใช้วิธีเดียวกันกับการคำนวณระดับความรุนแรงขององค์ประกอบ

3.6. การประเมินความเสี่ยงภายในลำดับเหตุการณ์

ขั้นตอนนี้เป็นขั้นตอนการประเมินความเสี่ยงในแต่ละลำดับเหตุการณ์ซึ่งนำเอาผลลัพธ์จากการประเมินความเสี่ยงในระดับองค์ประกอบและปฏิสัมพันธ์ระหว่างองค์ประกอบมาสร้างเป็นเมทริกซ์ความน่าจะเป็นประเภทต่างๆ และนำเมทริกซ์เหล่านั้นมาผ่านการคำนวณให้ได้มาซึ่งเมทริกซ์ความน่าจะเป็นที่ซอฟต์แวร์เริ่มต้นการทำงานและจบลงที่สถานะสิ้นสุดและ จบลงที่สถานะความล้มเหลวประเภทต่างๆด้วยค่าความเสี่ยงต่างกัน ผู้วิจัยแบ่งขั้นตอนการประเมินออกเป็น 2 ขั้นตอนคือ

3.6.1. การสร้างเมทริกซ์ความน่าจะเป็นของการทำงานที่ปราศจากความผิดพลาดของลำดับเหตุการณ์

ขั้นตอนนี้เริ่มต้นจากการสร้างกราฟควบคุมการทำงาน (Control flow graph) ที่ได้จากแผนภาพลำดับที่แสดงปฏิสัมพันธ์ระหว่างองค์ประกอบในลำดับเหตุการณ์ ซึ่งแสดงถึงพฤติกรรมการทำงานของซอฟต์แวร์ โดยกราฟที่สร้างขึ้นนั้นกำหนดให้มีโหนดเริ่มต้นและโหนดสิ้นสุดการทำงานเพียงจุดเดียว (Single entry and single exist node) โดยโหนดที่ปรากฏในกราฟดังกล่าวเป็นตัวแทนขององค์ประกอบที่มีการทำงานร่วมกันในลำดับเหตุการณ์ และเส้นเชื่อมระหว่างโหนดเป็นการแสดงถึงการเปลี่ยนแปลงการทำงานระหว่างองค์ประกอบ ซึ่งการเปลี่ยนแปลงการทำงานขององค์ประกอบเหล่านี้มีการเปลี่ยนแปลงในลักษณะที่อิสระต่อกันซึ่งเป็นไปตามกฎห่วงโซ่ของมาร์คอฟ และด้วยลักษณะดังกล่าวจึงสามารถสร้างเมทริกซ์ความน่าจะเป็นที่ซอฟต์แวร์ทำงานได้สำเร็จด้วยแบบจำลอง [5]

$$P^x = [p_{ij}^x] \tag{11}$$

กำหนดให้

$[p_{ij}^x]$ คือความน่าจะเป็นที่ซอฟต์แวร์ทำงานจากองค์ประกอบ i และเข้าสู่การทำงานในองค์ประกอบj และความน่าจะเป็นดังกล่าวสามารถคำนวณได้จากแบบจำลอง

$$p_{ij}^x = \frac{n_{ij}^x}{n_i^x} \tag{12}$$

กำหนดให้

- n_{ij}^x คือจำนวนเส้นเชื่อมระหว่างองค์ประกอบ i ไปยังองค์ประกอบ j
- n_i^x คือจำนวนเส้นเชื่อมจากองค์ประกอบ i ทั้งหมดในแผนภาพ

3.6.2. การสร้างเมทริกซ์ความน่าจะเป็นของการทำงานที่ผิดผลาดของลำดับเหตุการณ์

ขั้นตอนนี้เป็นการสร้างแบบจำลองความเสี่ยงของลำดับเหตุการณ์ซึ่งเป็นการพิจารณาค่าความเสี่ยงขององค์ประกอบและการเชื่อมโยงระหว่างองค์ประกอบ โดยความผิดพลาดสามารถเกิดได้ในระหว่างการทำงานภายในองค์ประกอบหรือการเปลี่ยนแปลงระหว่างองค์ประกอบ ซึ่งความผิดพลาดเหล่านี้เกิดขึ้นอย่างเป็นอิสระต่อกัน และเนื่องจากระดับความรุนแรงของความผิดพลาดเป็น

ปัจจัยหนึ่งที่สำคัญต่อการประเมินค่าความเสี่ยง ดังนั้นงานวิจัยฉบับนี้จึงเพิ่มเติมสถานะของความผิดพลาดซึ่งเทียบเท่ากับความรุนแรงในแต่ละระดับเข้าสู่แบบจำลองความเสี่ยงของลำดับเหตุการณ์ ดังนั้นแบบจำลองนี้ประกอบไปด้วยโหนดชั่วคราวจำนวน $(n+1)$ โหนด (โหนดที่เป็นตัวแทนองค์ประกอบและโหนดเริ่มต้น) อีกส่วนหนึ่งคือโหนดดูดกลืนจำนวน $(m+1)$ โหนด (โหนดสถานะความผิดพลาดและโหนดสิ้นสุด)

จากนั้นทำการเปลี่ยนแปลงเมทริกซ์ความน่าจะเป็น P^x เป็น $\overline{P}^x$ ซึ่งทำให้ p_{ij}^x เปลี่ยนเป็น $(1-rf_i^x)\cdot p_{ij}^x \cdot (1-rf_{ij}^x)$ ซึ่งเมทริกซ์ความน่าจะเป็น $\overline{P}^x$ นี้เป็นการแสดงความน่าจะเป็นเมื่อองค์ประกอบ i ไม่มีการทำงานที่ผิดพลาดและการเปลี่ยนแปลงระหว่างองค์ประกอบ $\boldsymbol{i}$ ไปสู่องค์ประกอบ j ไม่เกิดความผิดพลาดเช่นกัน สำหรับความผิดพลาดขององค์ประกอบลำดับที่ $\boldsymbol{i}$ นั้นพิจารณาได้จากการสร้างเส้นเชื่อมไปยังสถานะความผิดพลาดด้วยความน่าจะเป็น rf_i^x และเช่นเดียวกับความผิดพลาดของการเชื่อมโยงระหว่างองค์ประกอบ สามารถพิจารณาได้โดยการสร้างเส้นเชื่อมโยงไปยังสถานะความผิดพลาดด้วยค่าความน่าจะเป็นเท่ากับ $(1-rf_i^x)\cdot p_{ij}^x \cdot (rf_{ij}^x)$ ซึ่งระดับความผิดพลาดดังกล่าวเป็นระดับความผิดพลาดที่ได้จากการประเมินความเสี่ยงในระดับองค์ประกอบที่ $\boldsymbol{i}$ ขั้นตอนต่อไปเป็นการพิจารณาความน่าจะเป็นของการเปลี่ยนแปลงจากสถานะปกติเข้าสู่สถานะความผิดพลาดด้วยแบบจำลอง

$$A^x = (I - Q^x)^{-1} C^x \tag{13}$$

กำหนดให้

- A^x คือเมทริกซ์ความน่าจะเป็นของการทำงานที่เริ่มต้นจากสถานะชั่วคราวเข้าสู่สถานะดูดกลืน ขนาด $(m+1)\,x\,(m+1)$
- I คือเมทริกซ์เอกลักษณ์ $(m+1)\,x\,(m+1)$
- Q^x คือเมทริกซ์ความน่าจะเป็นของการเปลี่ยนแปลงจากสถานะสถานะชั่วคราวเข้าสู่สถานะสถานะชั่วคราวขนาด $(n+1)\,x\,(n+1)$
- C^x คือเมทริกซ์ความน่าจะเป็นของการเปลี่ยนแปลงจากสถานะปกติเข้าสู่สถานะดูดกลืนขนาด $(n+1)\,x\,(m+1)$

3.6.3. จัดอันดับของลำดับเหตุการณ์ตามค่าความเสี่ยงที่คำนวณ

เมื่อคำนวณหาเมทริกซ์ A^x เรียบร้อยแล้ว ขั้นตอนต่อไปคือการนำสมาชิกภายในแถวเดียวกันมาปรับค่าให้เข้ากับความน่าจะเป็นด้วยแบบจำลอง

$$A^{x\prime} = [a_{ij}^{x\prime}] \tag{14}$$

โดยที่ $a_{ij}^{x}{}'$ คำนวณได้จากแบบจำลอง

$$a_{ij}^{x}{}' = \frac{a_{ij}}{\sum_{i=1}^{n} a_{ij}} \quad (15)$$

จากนั้นระบุค่าความเสี่ยงของลำดับเหตุการณ์โดยการหาผลรวมของค่าความเสี่ยงในแต่ละระดับความผิดพลาดของลำดับเหตุการณ์นั้นๆ และนำผลลัพธ์ดังกล่าวมาเปรียบเทียบระหว่างลำดับเหตุการณ์ในซอฟต์แวร์ที่ทำการประเมิน ซึ่งผลลัพธ์สุดท้ายจะทำให้ผู้ประเมินทราบถึงลำดับเหตุการณ์ที่ส่งผลกระทบต่อโครงการพัฒนาซอฟต์แวร์ที่มีความเสี่ยงแฝงมากที่สุด ซึ่งหมายถึงหากลำดับเหตุการณ์ดังกล่าวเกิดข้อผิดพลาดขึ้น จะทำให้เกิดความเสียหายต่อซอฟต์แวร์มากที่สุด

บทที่ 4

การประยุกต์ใช้ขั้นตอนการประเมินความเสี่ยง ของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน

จากขั้นตอนการประเมินความเสี่ยงที่ได้กล่าวไว้ในบทที่สามนั้น จะเห็นได้ว่าการประเมินความเสี่ยงมีความซับซ้อนและประกอบด้วยหลายขั้นตอนที่จะทำให้ได้มาซึ่งค่าความเสี่ยงที่ต้องการ ดังนั้นรายละเอียดที่ปรากฏในบทนี้ เป็นการแสดงขั้นตอนการประเมินความเสี่ยงที่แฝงอยู่ในซอฟต์แวร์ที่นำมาใช้ประเมินอย่างละเอียด ดังต่อไปนี้

4.1. กรณีศึกษาการประเมินความเสี่ยงฯ

ขั้นตอนการประเมินฯที่ผู้วิจัยนำเสนอเป็นขั้นตอนสำหรับการประเมินความเสี่ยงที่แฝงอยู่ภายในซอฟต์แวร์ที่สามารถคำนวณค่าความเสี่ยงได้ในระหว่างการออกแบบ และซอฟต์แวร์ที่นำมาเป็นกรณีศึกษาคือ “ซอฟต์แวร์ระบบสารบรรณอิเล็กทรอนิกส์” ซึ่งเป็นซอฟต์แวร์สำหรับการรับส่งเอกสารภายในองค์กรที่มีลักษณะการส่งเอกสารที่คำนึงถึงระดับชั้นขององค์กร โดยลำดับเหตุการณ์ที่นำมาพิจารณาในการประเมินความเสี่ยงประกอบด้วยลำดับเหตุกาณ์ของการรับส่งหนังสือภายในหน่วยงาน ลำดับเหตุการณ์ของการนำเข้าบุคคลสู่ระบบ และลำดับเหตุการณ์ของการรับส่งหนังสือข้ามหน่วยงาน มีรายละเอียดดังนี้

1) ลำดับเหตุการณ์ของการรับส่งหนังสือภายในหน่วยงาน

เหตุการณ์นี้เริ่มต้นจากผู้ใช้ในฐานะเจ้าหน้าที่สารบรรณทำการสร้างเอกสารผ่านเมนูลงทะเบียนหนังสือและได้ผลลัพธ์คือเอกสารสำหรับใช้ส่งเพื่อแจ้งข่าวสารในองค์กรที่มีคุณลักษณะที่เป็นไปตามที่หน่วยงานกำหนด และขั้นตอนต่อไปเป็นการเสนอเอกสารผ่านเมนูการเสนอลงนามหนังสือ ซึ่งหมายถึงผู้ใช้ในฐานะเจ้าหน้าที่สารบรรณเกษียณข้อความเพื่อแจ้งให้ผู้ใช้ในระดับหัวหน้างานรับทราบ จากนั้นส่งต่อให้กับผู้ใช้ระดับหัวหน้างานทำหน้าที่ลงนามและส่งเอกสารถึงผู้รับ ขั้นตอนสุดท้ายของการส่งเอกสารคือ การส่งเอกสารฉบับดังกล่าวถึงผู้รับ ซึ่งเป็นการดำเนินการผ่านเมนูการส่งหนังสือถึงผู้รับผิดชอบ ในส่วนนี้ผู้ใช้ระดับหัวหน้างานพิจารณาข้อความที่ปรากฏในเอกสารและเกษียณข้อความลงนามรับรองเอกสารฉบับดังกล่าวและทำการส่งเอกสารถึงผู้รับต่อไป

เมื่อสิ้นสุดการส่งหนังสือถึงผู้รับผิดชอบเรียบร้อยแล้ว ขั้นตอนต่อไปเป็นการเริ่มต้นการรับหนังสือของผู้รับ โดยเริ่มต้นจากผู้ใช้ในฐานะเจ้าหน้าที่สารบรรณของหน่วยงานผู้รับเอกสารทำการตรวจสอบเอกสารที่ได้รับว่าผู้ส่งได้ส่งเอกสารถูกต้อง จากนั้นทำการลงทะเบียนรับเอกสารดังกล่าวและเสนอให้ผู้ใช้ในระดับหัวหน้าฝ่ายของตนลงนามหนังสือ และขั้นตอนสุดท้ายของลำดับเหตุการณ์นี้คือ การส่งเอกสารถึงผู้รับในระดับสุดท้ายของเส้นทางเอกสาร ดังนั้นจึงสรุปได้ว่าลำดับเหตุการณ์นี้ประกอบไปด้วยการทำงานร่วมกันขององค์ประกอบ “การลงทะเบียนหนังสือ (Register_Document)” องค์ประกอบ “การเสนอลงนาม (Propose_Sign)” และองค์ประกอบการส่งหนังสือ (Send_Document) และแสดงการทำงานร่วมกันขององค์ประกอบดังรูปที่ 4.2

หากพิจารณาถึงความเกี่ยวเนื่องกับฐานข้อมูล สามารถสรุปได้ว่าลำดับเหตุการณ์นี้มีความเกี่ยวข้องกับตารางชื่อ “Document”, “DocRecieveSend”, “RecieveSendType”, “DocLineConfig”, “DocSecretLevel”, “DocSpeedLevel”, “DocCommentSign” และ “DocCommentPropose” ดังปรากฏความสัมพันธ์ของตารางดังกล่าวได้ดังรูปที่ 4.1

2) ลำดับเหตุการณ์ของการนำเข้าผู้ใช้ลงสู่ระบบ

เหตุการณ์นี้เริ่มต้นด้วยการสร้างโครงสร้างองค์กรขึ้นภายในระบบซึ่งทำให้เกิดรหัสของหน่วยงานที่ถูกสร้างขึ้นภายใต้โครงสร้างองค์กรเหล่านั้น ขั้นตอนถัดไปเป็นการตั้งค่าข้อมูลพื้นฐานที่มีความเกี่ยวข้องกับรายละเอียดต่อผู้ใช้ที่นำเข้าสู่ระบบ อาทิเช่น คำนำหน้าชื่อ รหัสประเภทบุคลากร เป็นต้น เมื่อสร้างข้อมูลพื้นฐานเรียบร้อยแล้ว ขั้นตอนถัดไปเป็นการนำเข้าข้อมูลผู้ใช้จากเอกสารแนบ โดยการประมวลผลข้อมูลที่ปรากฏในเอกสารและจัดเก็บลงฐานข้อมูลและขั้นตอนดังกล่าวทำให้ได้ผลลัพธ์คือ รายชื่อผู้ใช้พร้อมด้วยรายละเอียดของผู้ใช้และหน่วยงานที่ผู้ใช้สังกัดอยู่ลงในระบบ และขั้นตอนสุดท้ายคือ ระบบดำเนินการสร้างชื่อผู้ใช้และรหัสผ่านพร้อมทั้งจัดเก็บข้อมูลเข้าสู่ส่วนการบริหารจัดการผู้ใช้ (User Management System) เพื่อรองรับการเข้าใช้งานของผู้ใช้ต่อไป

จากรายละเอียดในข้างต้น สามารถแบ่งองค์ประกอบที่ทำงานร่วมกันในลำดับเหตุการณ์ออกเป็นสามองค์ประกอบ คือ องค์ประกอบการสร้างโครงสร้างองค์กร (Organization) องค์ประกอบการสร้างข้อมูลผู้ใช้และการนำเข้าข้อมูลผู้ใช้ (User_Information) และองค์ประกอบการบริหารจัดการสิทธิ์ผู้ใช้ (Authorization) และสามารถแสดงการทำงานร่วมกันขององค์ประกอบได้ดังรูปที่ 4.2

และเมื่อพิจารณาถึงความเกี่ยวข้องกับฐานข้อมูล ลำดับเหตุการณ์นี้มีความเกี่ยวข้องกับตาราง Department, Person, Position และ UMS ดังแสดงไว้ในรูปที่ 4.1

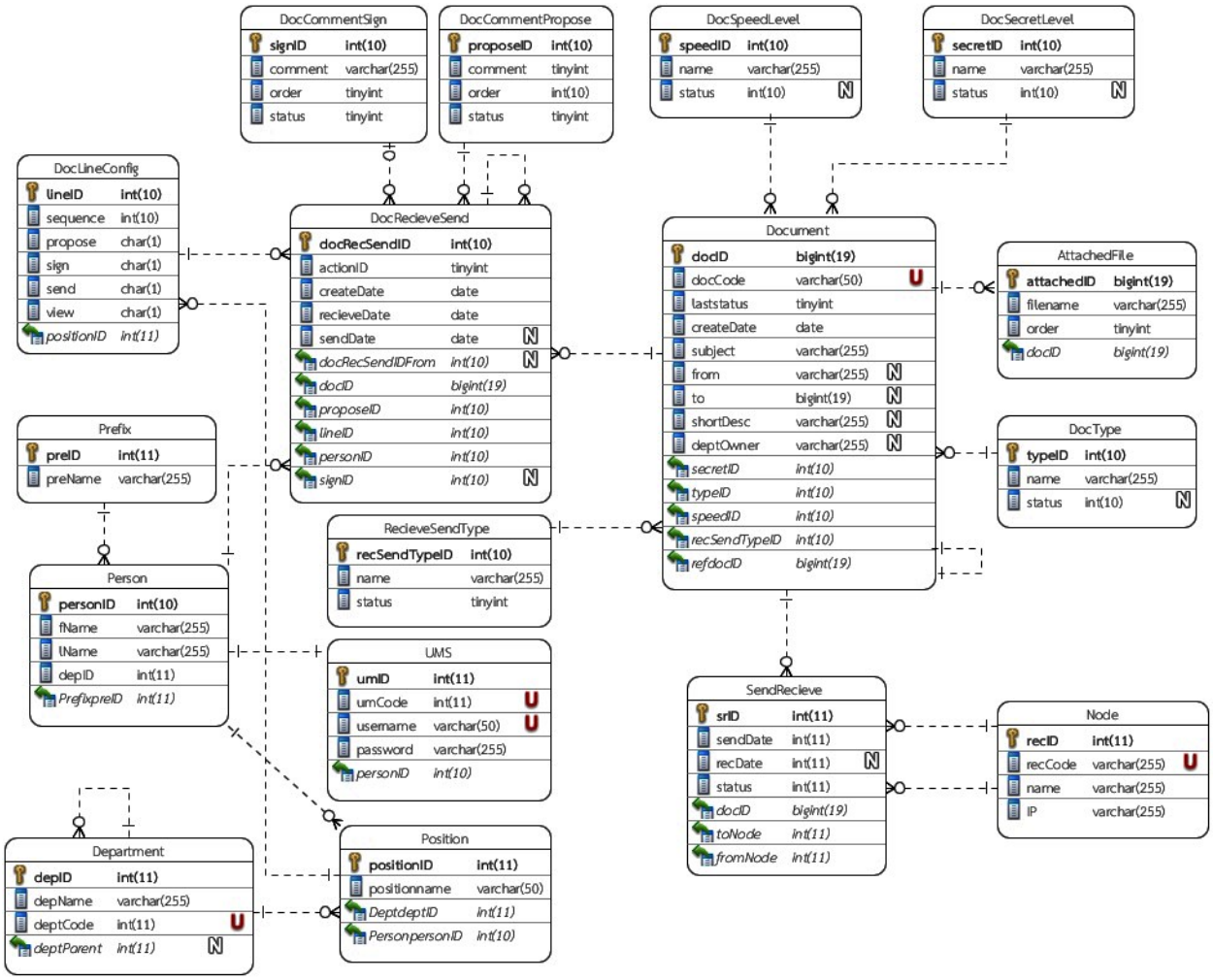
3) ลำดับเหตุการณ์ของการรับส่งหนังสือข้ามหน่วยงาน

เหตุการณ์นี้เริ่มต้นจากการตั้งค่าเครื่องแม่ข่ายของผู้รับที่ต้องการส่งหนังสือภายนอกหน่วยงาน จากนั้นผู้ส่งเลือกหน่วยงานที่ต้องการส่งหนังสือออก และเมื่อผู้รับได้รับหนังสือฉบับดังกล่าวเป็นที่เรียบร้อยแล้ว ระบบจะทำการอัพเดทสถานะของหนังสือเพื่อทำให้ผู้ใช้ทราบว่าหนังสือที่ส่งข้ามหน่วยงานได้ส่งถึงผู้รับจริง

การทำงานในส่วนนี้สามารถแบ่งองค์ประกอบที่ทำงานร่วมกันได้จำนวนสามองค์ประกอบด้วยกันคือ องค์ประกอบการตั้งค่าเครื่องแม่ข่ายปลายทาง (Manage_Node) องค์ประกอบการส่งหนังสือ (Send_Document) และองค์ประกอบการลงทะเบียนรับหนังสือ (Register_Document) และสามารถแสดงการทำงานร่วมกันขององค์ประกอบได้ดังรูปที่ 4.2

และเมื่อพิจารณาถึงความเกี่ยวข้องกับฐานข้อมูล ลำดับเหตุการณ์นี้มีความเกี่ยวข้องกับตาราง Node, SendRecieve, Document และตาราง DocRecieveSend ดังแสดงไว้ในรูปที่ 4.1

กรณีศึกษาซอฟต์แวร์ระบบสารบรรณอิเล็กทรอนิกส์ในข้างต้น ได้รับการออกแบบฐานข้อมูลเชิงสัมพันธ์เพื่อเก็บข้อมูลที่เกิดขึ้นจากการใช้งาน สามารถแสดงรายละเอียดความสัมพันธ์ของฐานข้อมูลได้ด้วยแผนภาพความสัมพันธ์ของเอนทิตี้ดังรูปที่ 4.1



รูปที่ 4.1 ฐานข้อมูลซอฟต์แวร์ระบบสารบรรณอิเล็กทรอนิกส์

จากลำดับเหตุการณ์ทั้งสามเหตุการณ์ที่ได้กล่าวมาแล้วนั้น เมื่อนำมาพิจารณาร่วมกับแผนภาพความสัมพันธ์ของเอนทิตี้ดังรูปที่ 4.1 สามารถสรุปความสัมพันธ์ระหว่างองค์ประกอบและตารางของฐานข้อมูลได้ดังตารางที่ 4.1 สำหรับคำอธิบายตารางได้แสดงไว้ในภาคผนวก ก.

ตารางที่ 4.1 ความสัมพันธ์ระหว่างองค์ประกอบและตารางในฐานข้อมูลเชิงสัมพันธ์

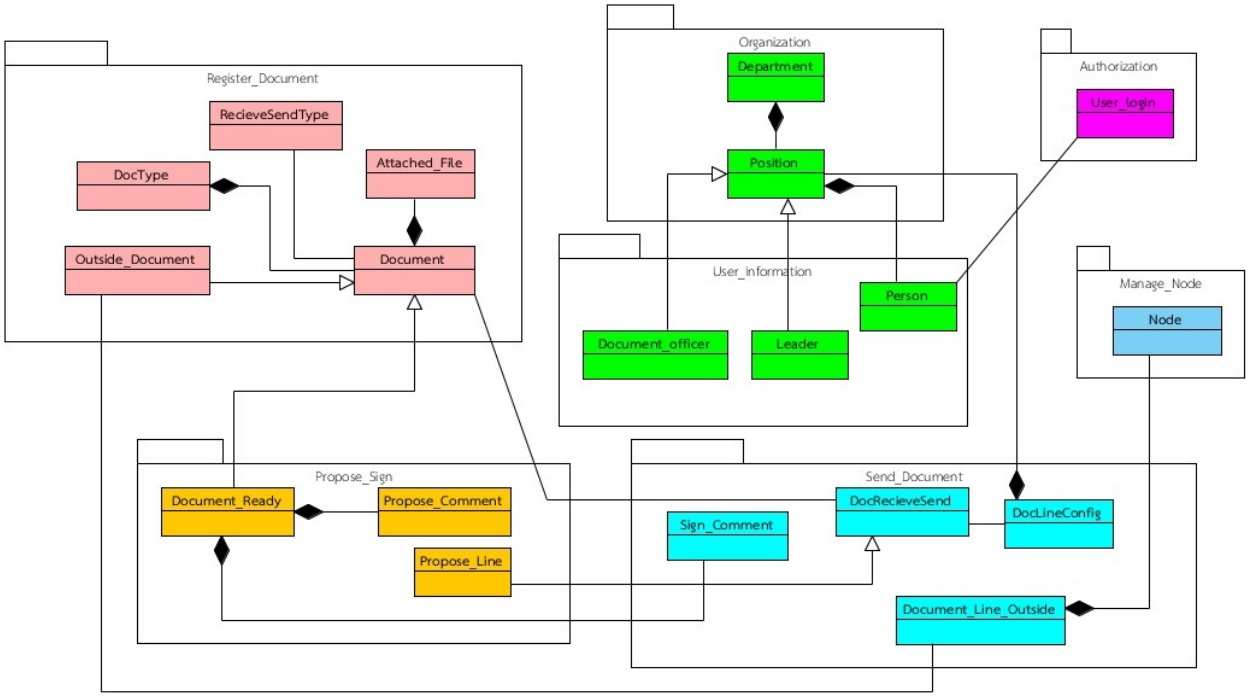
ลำดับเหตุการณ์	องค์ประกอบ	ชื่อตาราง
การรับส่งหนังสือภายในหน่วยงาน	การลงทะเบียนหนังสือ	Attachedfile, Document, DocType DocRecieveSend, DocSecretLevel, DocSpeedLevel
	การเสนอลงนาม	DocCommentPropose, DocLineConfig, DocRecieveSend
	การส่งหนังสือถึงผู้รับผิดชอบ	DocCommentSign, DocLineConfig, DocRecieveSend
การนำเข้าผู้ใช้ลงสู่ระบบ	การสร้างโครงสร้างองค์กร	Department, Position
	การสร้างข้อมูลผู้ใช้และการนำเข้าข้อมูลผู้ใช้	Person, Prefix
	การบริหารจัดการสิทธิ์ผู้ใช้	Person, UMS
การรับส่งหนังสือข้ามหน่วยงาน	การตั้งค่าเครื่องแม่ข่ายปลายทาง	Node
	การส่งหนังสือภายนอก	SendRecieve, Node
	กาลงทะเบียนรับหนังสือ	DocRecieveSend, Document

4.2. การกำหนดขอบเขตและการออกแบบการทำงานของซอฟต์แวร์

ขั้นตอนนี้เป็นขั้นตอนที่ผู้ประเมินต้องกำหนดลำดับเหตุการณ์เพื่อนำมาประเมินหาค่าความเสี่ยง ซึ่งหลักเกณฑ์ในการกำหนดคือลำดับเหตุการณ์ที่เกี่ยวข้องกับการเปลี่ยนแปลงข้อมูลภายในฐานข้อมูล ที่เป็นส่วนหนึ่งของยูสเคสในแผนภาพยูสเคส ซึ่งปกติแล้วในหนึ่งยูสเคสจะประกอบด้วยลำดับเหตุการณ์อย่างน้อยหนึ่งลำดับเหตุการณ์ เมื่อผู้ประเมินสามารถกำหนดลำดับเหตุการณ์ภายในซอฟต์แวร์ที่ต้องการประเมินแล้ว ส่วนต่อไปเป็นการแยกองค์ประกอบที่ประกอบอยู่ในลำดับเหตุการณ์หนึ่งๆ ซึ่งผู้วิจัยได้นำเอา “ลำดับเหตุการณ์ของการรับส่งหนังสือภายในหน่วยงาน” ใช้ในการแสดงรายละเอียดของการประเมินตั้งแต่ขั้นตอนแรกจนถึงขั้นตอนสุดท้าย มีรายละเอียดดังนี้

การทำงานภายในลำดับเหตุการณ์ของการรับส่งหนังสือภายในหน่วยงานประกอบด้วยองค์ประกอบจำนวนสามองค์ประกอบคือ องค์ประกอบของการลงทะเบียนหนังสือ องค์ประกอบของการเสนอลงนาม และองค์ประกอบของการส่งหนังสือถึงผู้รับผิดชอบ ซึ่งองค์ประกอบดังกล่าวเกิดจากการทำงานร่วมกันของคลาสที่

เกิดขึ้นในขั้นตอนการวิเคราะห์ระบบ จากนั้นนำคลาสที่มีการทำงานร่วมกันเพื่อตอบสนองงานกลุ่มหนึ่งๆ แสดงได้ดังแผนภาพแพคเกจดังรูปที่ 4.2



รูปที่ 4.2 แพคเกจของคลาสที่เกิดขึ้นในลำดับเหตุการณ์ของซอฟต์แวร์ระบบสารบรรณ

จากรูปที่ 4.2 สามารถอธิบายการทำงานระหว่างองค์ประกอบได้ดังตารางที่ 4.2

ตารางที่ 4.2 คำอธิบายแผนภาพแพคเกจของลำดับเหตุการณ์ในซอฟต์แวร์ระบบสารบรรณ

ลำดับที่	ชื่อแพคเกจ	ชื่อคลาส	คุณสมบัติและการทำงาน
1	Register_Document	DocType	มีคุณลักษณะสำหรับการจัดเก็บหนังสือภายใต้ประเภทเดียวกัน และสามารถแสดงรายการของเอกสารที่จัดเก็บภายใต้แฟ้มเอกสารประเภทเดียวกันได้
2		Document	มีคุณลักษณะสำหรับการจัดเก็บข้อมูลภายในหนังสือทางสารบรรณและมีข้อมูลเอกสารแนบเป็นส่วนประกอบ สามารถบริหารจัดการ รวมถึงการสร้างหนังสือขึ้นใหม่ได้
3		Attached_File	มีคุณลักษณะสำหรับการจัดเก็บเอกสารแนบ และสามารถจัดเก็บเอกสารแนบลงในระบบได้
4		Outside_Document	มีคุณลักษณะสำหรับการจัดเก็บหนังสือที่ได้รับจากภายนอก และสามารถลงทะเบียนหนังสือที่ถูกส่งมาจากภายนอกได้
5	Propose_Sign	Document_Ready	เป็นคลาสที่ได้รับการสืบทอดจากคลาส New_Document ซึ่งมีคุณคุณลักษณะสที่แตกต่างกันคือ คลาสนี้มีข้อความเกษียณเพื่อรอลงนามและข้อความเกษียณการลงนามเป็นส่วนประกอบ ส่วนความสามารถเป็นไปตามคลาสที่ทำการสืบทอดมา
6		Propose_Comment	มีคุณลักษณะสำหรับการจัดเก็บข้อความเกษียณเพื่อรอลงนาม และสามารถบริหารจัดการข้อความเกษียณเพื่อรอลงนามได้
7		Propose_Line	มีคุณลักษณะสำหรับการจัดเก็บเส้นทางของการเสนอลงนาม มีความสามารถในการกำหนดเส้นทางการเสนอลงนามหนังสือได้
8	Organization	Department	มีคุณลักษณะสำหรับการจัดเก็บโครงสร้างส่วนงานและมีส่วนประกอบเป็นตำแหน่งภายในหน่วยงาน มีความสามารถบริหารจัดการโครงสร้างหน่วยงานและกำหนดตำแหน่งในส่วนงานได้

ลำดับที่	ชื่อแพคเกจ	ชื่อคลาส	คุณสมบัติและการทำงาน
9	Organization	Position	มีคุณลักษณะสำหรับการจัดเก็บตำแหน่งและโดยมีข้อมูลบุคคลเป็นส่วนประกอบ สามารถบริหารจัดการข้อมูลตำแหน่งและกำหนดบุคคลในแต่ละตำแหน่งได้
10	User_Information	Person	มีคุณลักษณะสำหรับจัดเก็บข้อมูลบุคคล และสามารถจัดการข้อมูลของบุคคลได้
11		Document_Officer	คลาสนี้ได้รับการสืบทอดจากคลาส Position โดยสิ่งที่แตกต่างออกไปคือ ความสามารถในการเสนอลงนามและกำหนดเส้นทางการลงนาม
12		Leader	คลาสนี้ได้รับการสืบทอดจากคลาส Position โดยสิ่งที่แตกต่างออกไปคือ ความสามารถในส่งหนังสือและกำหนดเส้นทางการส่งหนังสือทั้งภายในและภายนอก
13	Send_Document	Sign_Comment	มีคุณลักษณะสำหรับการจัดเก็บข้อมูลการลงนาม และสามารถจัดการข้อมูลการลงนามได้
14		DocRecieveSend	มีคุณลักษณะสำหรับจัดเก็บเส้นทางการส่งหนังสือภายใน และสามารถจัดการ
15		Document_Line_Outside	มีคุณลักษณะสำหรับจัดเก็บเส้นทางการส่งหนังสือภายนอกและมีข้อมูลโหนดผู้รับเป็นองค์ประกอบ มีความสามารถในการกำหนดเส้นทางการส่งหนังสือภายนอกได้
		DocLineConfig	มีคุณลักษณะในการจัดเก็บลำดับทางสารบรรณ และสามารถกำหนดเส้นทางการส่งหนังสือได้
16	Authorization	User_Login	มีคุณลักษณะสำหรับจัดเก็บข้อมูลการพิสูจน์ตัวตน มีความสามารถจัดการข้อมูลในการพิสูจน์ตัวตนได้
18	Mange_Node	Node	มีคุณลักษณะสำหรับจัดเก็บข้อมูลเครื่องแม่ข่ายของผู้รับ และสามารถจัดการข้อมูลเครื่องแม่ข่ายได้

จะเห็นได้ว่าการออกแบบคลาสในระดับบน (High level conceptual model) ดังรูปที่ 4.2 นั้นนำไปสู่การออกแบบแผนภาพความสัมพันธ์ของเอนทิตี้ผ่านวิธีการเชื่อมโยงที่ปรากฏใน [13] ทำให้ได้มาซึ่งตารางในฐานข้อมูลทั้งสิ้น 17 ตาราง ดังรูปที่ 4.1 และสามารถแสดงที่มาและการเชื่อมโยงระหว่างแผนภาพทั้งสองได้ดังตารางที่ 4.3

ตารางที่ 4.3 ที่มาและการเชื่อมโยงระหว่างแผนภาพคลาสและแผนภาพความสัมพันธ์ของเอนทิตี้

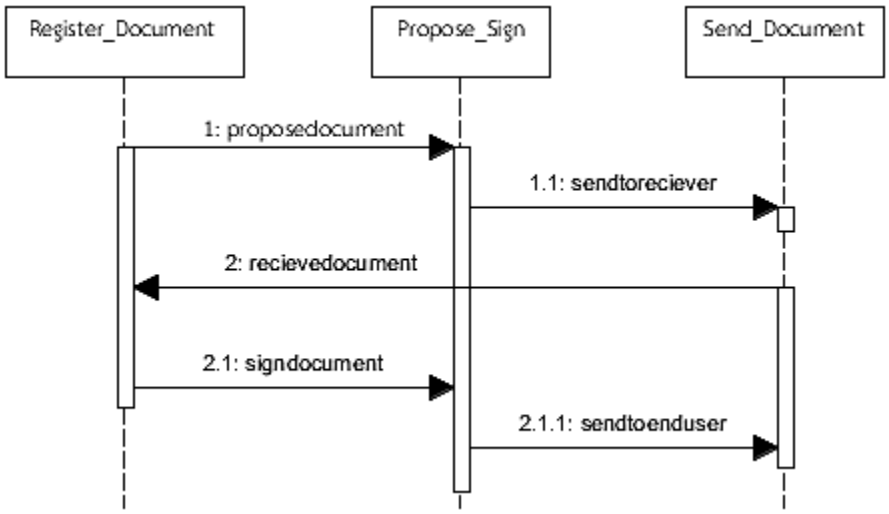
ลำดับที่	ชื่อคลาส	ชื่อตาราง	เหตุผลประกอบ
1	Document, Document_Ready และ Outside_Document	Document, DocSpeedLevel และ DocSecretLevel	คลาส Document_Ready และคลาส Outside_Document ได้รับการถ่ายทอดมาจากคลาส Document ซึ่งสามารถนำคุณลักษณะของคลาสลูก (Subclass) มาสร้างเป็นตารางได้โดยไม่ต้องสร้างตารางให้กับตารางแม่ (Parent Class) สำหรับตาราง DocSpeedLevel และ DocSecretLevel เกิดจากคุณลักษณะประจำหนึ่งของคลาส Document ที่มีค่าหลายค่า จึงนำเอาคุณสมบัตินั้นแยกออกเป็นตารางเฉพาะ
2	DocType	docType	คลาส DocType มีคุณสมบัติในการนำไปสร้างเป็นวัตถุ (Concrete class) ซึ่งสามารถนำมาสร้างเป็นตารางในฐานข้อมูลได้ ส่วนในด้านความสัมพันธ์ระหว่างคลาสมีความสัมพันธ์แบบองค์ประกอบ (Composition) กับตาราง Document และเมื่อนำมาสร้างเป็นตารางในฐานข้อมูลแล้วทำให้ตารางทั้งสองมีความสัมพันธ์แบบหนึ่งต่อหลายกับตาราง Document ซึ่งทำให้คีย์หลักในตาราง DocType ถูกนำไปใช้เป็นคีย์เชื่อมในตาราง Document
3	Attached_File	AttachedFile	คลาส DocType มีคุณสมบัติในการนำไปสร้างเป็นวัตถุ ซึ่งสามารถนำมาสร้างเป็นตารางในฐานข้อมูลได้ ส่วนในด้านความสัมพันธ์ระหว่างคลาสมีความสัมพันธ์แบบองค์ประกอบกับตาราง Document และเมื่อนำมาสร้างเป็นตารางในฐานข้อมูล

ลำดับที่	ชื่อคลาส	ชื่อตาราง	เหตุผลประกอบ
			แล้วทำให้ตารางทั้งสองมีความสัมพันธ์ระหว่างตาราง Document และตาราง AttachedFile เป็นแบบหนึ่งต่อหลายซึ่งทำให้คีย์หลักในตาราง Document ถูกนำไปใช้เป็นคีย์เชื่อมในตาราง AttachedFile
4	DocRecieveSend, Propose_Line	DocRecieveSend	คลาส Propose_Line ได้รับการสืบทอดจากคลาส DocRecieveSend และมีความสัมพันธ์กับคลาส Document ซึ่งสามารถนำมาสร้างเป็นตารางชื่อ DocRecieveSend ในฐานข้อมูลและทำให้ความสัมพันธ์ระหว่างตาราง Document และ DocRecieveSend เป็นแบบหนึ่งต่อหลาย
5	Propose_comment	DocCommentPropose	คลาส Propose_comment มีที่สามารถนำมาสร้างเป็นวัตถุได้ และมีความสัมพันธ์แบบองค์ประกอบกับคลาส Document จึงนำมาสร้างเป็นตารางในฐานข้อมูลและมีความสัมพันธ์ระหว่างตาราง DocCommentPropose และ Document เป็นแบบหนึ่งต่อหลาย
6	Sign_comment	DocCommentSign	คลาส Sign_comment มีลักษณะเป็นคลาสที่สามารถนำมาสร้างเป็นวัตถุได้ และมีความสัมพันธ์แบบองค์ประกอบกับคลาส Document จึงนำมาสร้างเป็นตารางในฐานข้อมูลและมีความสัมพันธ์ระหว่างตาราง DocCommentSign และ Document เป็นแบบหนึ่งต่อหลาย
7	Document_line_outside	SendRecieve	คลาส Document_line_outside เป็นคลาสที่สามารถนำมาสร้างเป็นวัตถุได้ และมีความสัมพันธ์กับคลาส Outside_Document ที่สืบทอดมากจากตาราง Document ดังนั้นทำให้เกิดเป็นตารางชื่อ SendRecieve ที่มีความสัมพันธ์แบบหนึ่งต่อหลายจากตาราง Document

ลำดับที่	ชื่อคลาส	ชื่อตาราง	เหตุผลประกอบ
8	Position, Document_officer และ Leader	Position	คลาส Document_office และ Leader ได้รับการถ่ายทอดมาจากคลาส Position และมีความสัมพันธ์กับคลาส Department แบบองค์ประกอบ ซึ่งการสร้างตารางสามารถนำเอาคุณลักษณะประจำของคลาสทั้งหมดมาสร้างเป็นคุณลักษณะประจำตาราง และมีความสัมพันธ์ระหว่างตาราง Department และตาราง Position แบบหนึ่งต่อหลาย
9	Department	Department	คลาส Department มีลักษณะเป็นคลาสที่สามารถนำมาสร้างเป็นวัตถุได้ จึงทำให้เกิดตาราง Department ในฐานข้อมูล
10	Person	Person และ Prefix	คลาส Person เป็นคลาสที่นำมาสร้างเป็นวัตถุได้และมีคำนำหน้าชื่อเป็นลักษณะประจำซึ่งมีค่าหลายค่าซ้ำๆกัน ดังนั้นจึงออกแบบให้มีตาราง Prefix ในฐานข้อมูล
11	User_login	UMS	คลาส User_login มีลักษณะเป็นคลาสที่สามารถนำมาสร้างเป็นวัตถุได้และมีความสัมพันธ์กับคลาส Person จึงทำให้เกิดตารางขึ้นในฐานข้อมูลภายใต้ชื่อ UMS และมีความสัมพันธ์กับตาราง Person แบบหนึ่งต่อหนึ่ง
12	DocLineConfig	DocLineConfig	คลาส DocLineConfig มีลักษณะเป็นคลาสที่สามารถนำมาสร้างเป็นวัตถุได้ โดยมีความสัมพันธ์กับคลาส DocRecieveSend และความสัมพันธ์แบบองค์ประกอบกับคลาส Position ซึ่งทำให้เกิดตาราง DocLineConfig ที่มีความสัมพันธ์กับตาราง DocRecieveSend แบบหนึ่งต่อหลาย และมีความสัมพันธ์ระหว่างตาราง Position กับตาราง DocLineConfig แบบหนึ่งต่อหลาย

ลำดับที่	ชื่อคลาส	ชื่อตาราง	เหตุผลประกอบ
13	Node	Node	คลาส Node มีลักษณะเป็นคลาสที่สามารถนำมาสร้างเป็นวัตถุได้ และมีความสัมพันธ์แบบองค์ประกอบกับคลาส Document_line_outside จึงทำให้เกิดตาราง Node ที่มีความสัมพันธ์แบบหนึ่งต่อหลายกับตาราง SendRecieve
14	RecieveSendType	RecieveSendType	คลาส Node มีลักษณะเป็นคลาสที่สามารถนำมาสร้างเป็นวัตถุได้ และมีความสัมพันธ์กับตาราง Document ซึ่งทำให้เกิดตาราง RecieveSendType ที่มีความสัมพันธ์กับตาราง Document แบบหนึ่งต่อหลาย

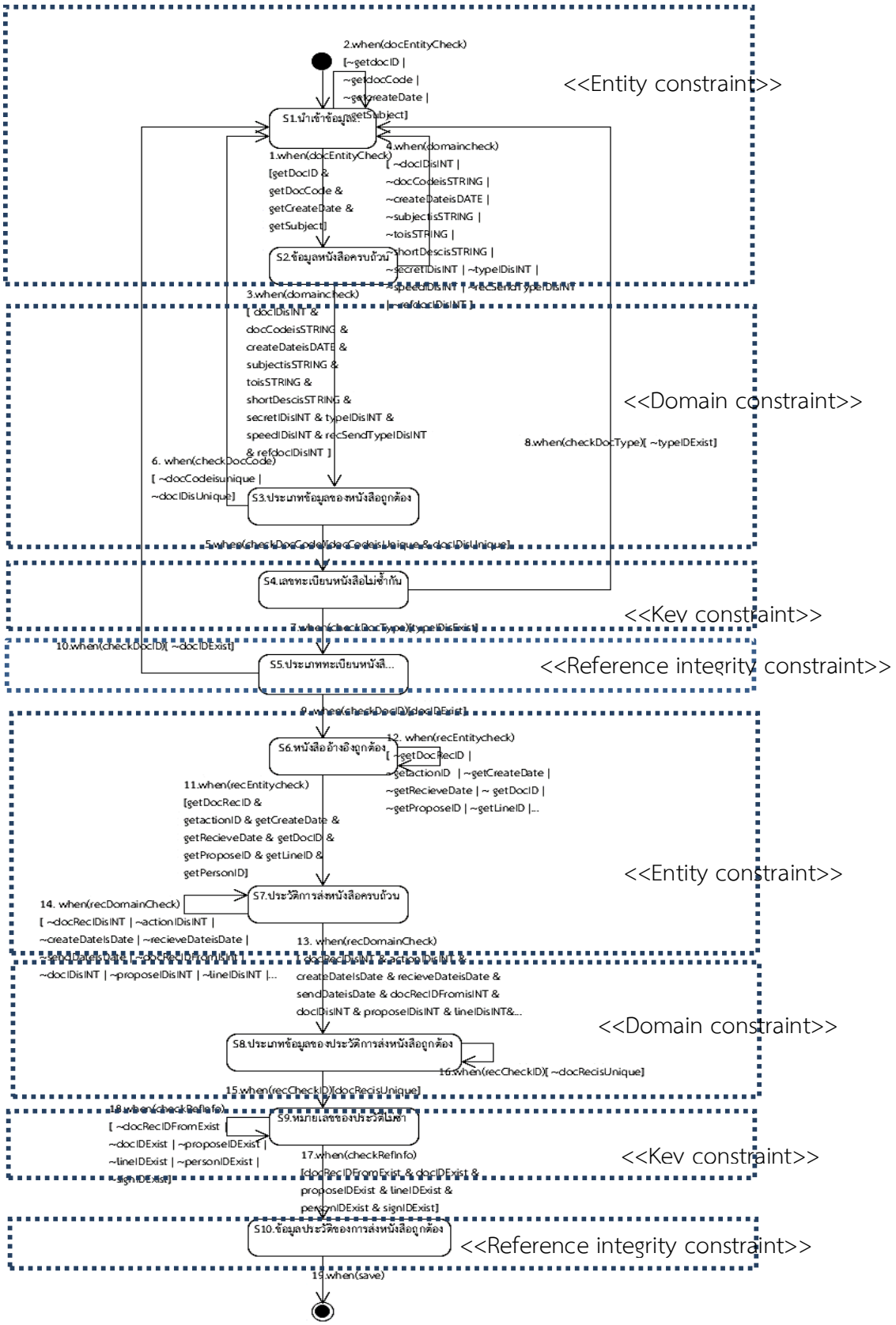
เมื่อผู้ประเมินสามารถกำหนดองค์ประกอบที่ทำงานร่วมกันภายในหนึ่งลำดับเหตุการณ์ได้แล้ว ขั้นตอนถัดไปเป็นการพิจารณาถึงปฏิสัมพันธ์ระหว่างองค์ประกอบผ่านแผนภาพลำดับ สามารถแสดงแผนภาพลำดับของลำดับเหตุการณ์ได้ดังรูปที่ 4.3



รูปที่ 4.3 แผนภาพลำดับของลำดับเหตุการณ์การรับส่งหนังสือภายในหน่วยงาน

จะเห็นได้ว่าการสร้างแผนภาพลำดับมีการนำเอาองค์ประกอบที่ปรากฏในแผนภาพแพคเกจมาใช้เป็นเส้นชีวิต (Lifeline) จากนั้นระบุปฏิสัมพันธ์ระหว่างองค์ประกอบผ่านข้อความที่ใช้สื่อสารระหว่างองค์ประกอบ นอกจากนี้แผนภาพลำดับยังถูกนำไปใช้ในขั้นตอนการประเมินความเสี่ยงของลำดับเหตุการณ์โดยการสร้างกราฟควบคุมการทำงานบนพื้นฐานของแผนภาพลำดับนี้

หลังจากการกำหนดองค์ประกอบภายในลำดับเหตุการณ์เรียบร้อยแล้ว ขั้นต่อไปเป็นการพิจารณาสถานะการทำงานภายในองค์ประกอบด้วยแผนภาพสเตทแมชชีน โดยมีวัตถุประสงค์เพื่อแสดงสถานะของการดำเนินการกับข้อมูลที่ถูกจัดเก็บลงฐานข้อมูลเชิงสัมพันธ์ภายใต้เงื่อนไขเชิงเค้าร่าง สำหรับองค์ประกอบทั้งสามองค์ประกอบในข้างต้น สามารถแสดงสถานะของการดำเนินการกับข้อมูลฯได้ดังรูปที่ 4.4 - 4.6 ตามลำดับ



รูปที่ 4.4 สถานะการดำเนินการกับข้อมูลฯขององค์ประกอบการลงทะเบียนหนังสือ

จากรูปที่ 4.4 สามารถอธิบายเหตุการณ์อันเป็นตัวกระตุ้นที่ทำให้เกิดการเปลี่ยนแปลงสถานะและเงื่อนไขที่เกี่ยวข้องได้ดังตารางที่ 4.4

ตารางที่ 4.4 คำอธิบายแผนภาพสเตทแมชชีนขององค์ประกอยการลงทะเบียนหนังสือ

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
P0	เริ่มต้น - นำเข้าข้อมูลหนังสือ	-	เป็นค่าปริยายที่การเริ่มต้นการทำงานขององค์ประกอบต้องเข้าสู่สถานะการนำเข้าข้อมูลหนังสือ
P1	นำเข้าข้อมูลหนังสือ - ข้อมูลหนังสือครบถ้วน	when(docEntityCheck)[getDocID & getDocCode & getCreateDate & getSubject]	การเปลี่ยนสถานะในส่วนนี้เกิดจากการตรวจสอบเงื่อนไขค่าว่า (EntityCheck) โดยมีเงื่อนไขคือ ค่าของ DocID, DocCode, CreateDate และ Subject ที่เป็นคุณลักษณะประจำตาราง Document จะต้องไม่เป็นค่าว่าง
P2	นำเข้าข้อมูลหนังสือ - นำเข้าข้อมูลหนังสือ	when(docEntityCheck)[~getdocID | ~getdocCode | ~getcreateDate | ~getSubject]	การเปลี่ยนสถานะจากการนำเข้าหนังสือไม่สามารถเปลี่ยนสถานะไปยังสถานะอื่นได้เมื่อการตรวจสอบเงื่อนไขค่าว่างพบ DocID, DocCode, CreateDate และ Subject ค่าใดค่าหนึ่งเป็นค่าว่าง

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
P3	ข้อมูลหนังสือครบถ้วน - ประเภทข้อมูลของหนังสือถูกต้อง	when(domainCheck)[docIDisINT & docCodeisSTRING & createDateisDATE & subjectisSTRING & toisSTRING & shortDescisSTRING & secretIDisINT & typeIDisINT & speedIDisINT & recSendTypeIDisINT & refdocIDisINT]	การเปลี่ยนสถานะในส่วนนี้เกิดจากการตรวจสอบเงื่อนไขประเภทข้อมูล (domainCheck) โดยมีเงื่อนไขคือ ค่าของ docID เป็นจำนวนเต็ม ค่าของ docCode เป็นสายอักขระ ค่าของ createDate มีรูปแบบเป็นวันที่ ค่าของ subject เป็นสายอักขระ ค่าของ to เป็นสายอักขระ ค่าของ shortDesc เป็นสายอักขระ ค่าของ secretID เป็นจำนวนเต็ม ค่าของ typeID เป็นจำนวนเต็ม ค่าของ speedID เป็นจำนวนเต็ม ค่าของ recSendTypeID เป็นจำนวนเต็ม ค่าของ refdocID เป็นจำนวนเต็ม ซึ่งประเภทของข้อมูลเหล่านี้ต้องเป็นไปตามเงื่อนไขประเภทข้อมูลที่ได้กำหนดไว้ในตาราง Document
P4	ข้อมูลหนังสือครบถ้วน - นำเข้าข้อมูลหนังสือ	when(domaincheck)[~docIDisINT |~docCodeisSTRING | ~createDateisDATE | ~subjectisSTRING | ~toisSTRING |	การเปลี่ยนสถานะจากสถานะข้อมูลหนังสือครบถ้วนกลับไปสู่สถานะการนำเข้าข้อมูลหนังสือ เกิดขึ้นเมื่อดำเนินการตรวจสอบ

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
		~shortDescisSTRING | ~secretIDisINT | ~typeIDisINT | ~speedIDisINT | ~recSendTypeIDisINT | ~refdocIDisINT]	เงื่อนไประเภทข้อมูลและพบว่า ค่าใดค่าหนึ่งในภายในเงื่อนไขไม่เป็นไปตามเงื่อนไขที่กำหนดไว้ในตาราง Document
P5	ประเภทข้อมูลของหนังสือถูกต้อง - เลขทะเบียนหนังสือไม่ซ้ำกัน	when(checkDocCode)[docCodeisUnique & docIDisUnique]	การเปลี่ยนสถานะจากประเภทข้อมูลของหนังสือถูกต้องไปยังสถานะทะเบียนหนังสือไม่ซ้ำกันได้สำเร็จ เกิดการเหตุการณ์การตรวจสอบ docID และ docCode ตามเงื่อนไขของคีย์หลัก โดยจะต้องไม่พบข้อมูลที่มีค่าเหมือนกันอยู่ในฐานข้อมูล
P6	ประเภทข้อมูลของหนังสือถูกต้อง - นำเข้าข้อมูลหนังสือ	when(checkDocCode)[~docCodeisunique | ~docIDisUnique]	การเปลี่ยนสถานะจากประเภทข้อมูลกลับสู่สถานะนำเข้าข้อมูลหนังสือเกิดขึ้นเนื่องจากการตรวจสอบค่า docID หรือ docCode มีข้อมูลที่มีค่าเหมือนกันอยู่ก่อนแล้วในฐานข้อมูล
P7	เลขทะเบียนหนังสือไม่ซ้ำกัน - ประเภททะเบียนหนังสือถูกต้อง	when(checkDocType)[typeIDisExist]	การเปลี่ยนสถานะจากสถานะเลขทะเบียนหนังซื้อไม่ซ้ำกันไปยังสถานะของประเภททะเบียนหนังสือเกิดจากการตรวจสอบประเภททะเบียนหนังสือ (checkDocType) ตามเงื่อนไขความสอดคล้องของข้อมูลและ

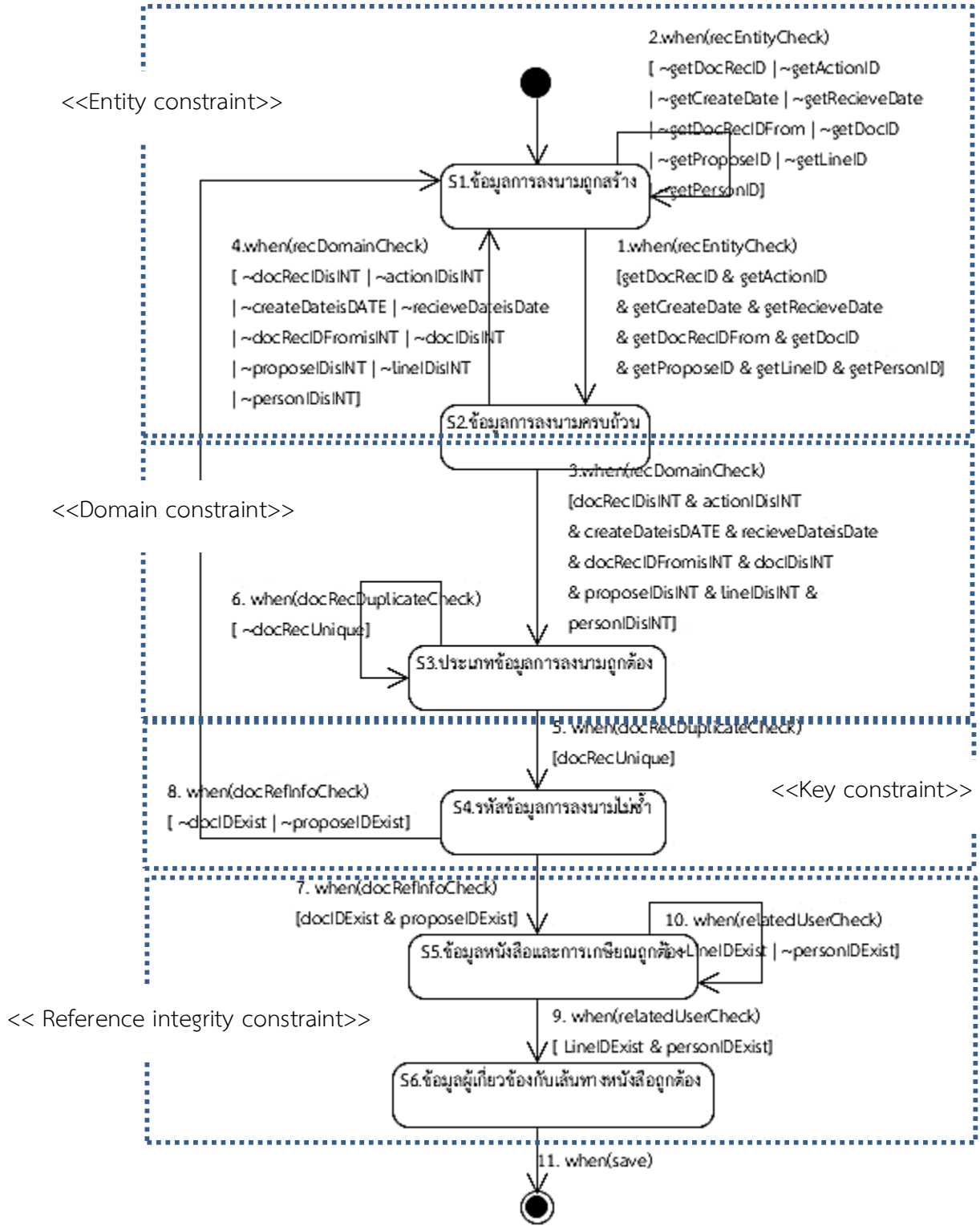
ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
			พบว่าค่า typeID มีค่าปรากฏอยู่ในตาราง DocType
P8	เลขทะเบียนหนังสือไม่ซ้ำกัน - นำเข้าข้อมูลหนังสือ	when(checkDocType)[~typeIDExist]	การเปลี่ยนสถานะจากสถานะเลขทะเบียนหนังสือไม่ซ้ำกันกลับสู่สถานะนำเข้าข้อมูลเนื่องจากผลลัพธ์ของการตรวจสอบค่า typeID ไม่ปรากฏในตาราง DocType
P9	ประเภททะเบียนหนังสือถูกต้อง - หนังสืออ้างอิงถูกต้อง	when(checkDocID)[docIDExist]	การเปลี่ยนสถานะจากประเภททะเบียนหนังสือถูกต้องไปยังสถานะหนังสืออ้างอิงถูกต้องเกิดขึ้นเนื่องจากการตรวจสอบหาเอกสารอ้างอิง (checkDocID) ซึ่งค่าของ docID มีปรากฏอยู่ในตาราง Document
P10	ประเภททะเบียนหนังสือถูกต้อง - นำเข้าข้อมูลหนังสือ	when(checkDocID)[~docIDExist]	การเปลี่ยนสถานะจากประเภททะเบียนหนังสือถูกต้องกลับสู่สถานะนำเข้าข้อมูลหนังสือเนื่องจากการตรวจสอบเอกสารอ้างอิงไม่พบค่าของ docID ปรากฏอยู่ในตาราง Document
P11	หนังสืออ้างอิงถูกต้อง - ประวัติการส่งหนังสือครบถ้วน	when(recEntitycheck)[getDocRecID & getactionID & getCreateDate &	การเปลี่ยนสถานะจากหนังสืออ้างอิงถูกต้องไปยังสถานะประวัติการส่งหนังสือครบถ้วนเกิดจากการตรวจสอบเงื่อนไขค่าว่างของ

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
		getRecieveDate & getDocID & getProposeID & getLineID &getPersonID]	ประวัติการส่งหนังสือ (recEntityCheck) มีค่า docRecID, getActionID, getCreateDate, getRecieveDate, docID, proposeID, lineID และ personID ครบถ้วน
P12	หนังสืออ้างอิงถูกต้อง - หนังสืออ้างอิงถูกต้อง	when(recEntitycheck)[~getDocRecID | ~getactionID | ~getCreateDate | ~getRecieveDate | ~ getDocID | ~getProposeID | ~getLineID | ~getPersonID]	การเปลี่ยนแปลงจากสถานะหนังสืออ้างอิงถูกต้องไม่สามารถเปลี่ยนไปยังสถานะอื่นได้ เนื่องจากการตรวจสอบเงื่อนไขค่าว่างของประวัติการส่งหนังสือไม่พบค่าใดค่าหนึ่งของ docRecID, getActionID, getCreateDate, getRecieveDate, docID, proposeID, lineID และ personID
P13	ประวัติการส่งหนังสือครบถ้วน - ประเภทข้อมูลของประวัติการส่งหนังสือถูกต้อง	when(recDomainCheck)[docRecIDisINT & actionIDisINT & createDateIsDate & recieveDateisDate & sendDateisDate & docRecIDFromisINT & docIDisINT & proposeIDisINT & lineIDisINT& personIDisINT & signIDisINT]	การเปลี่ยนสถานะจากประวัติการส่งหนังสือครบถ้วนไปยังสถานะประเภทข้อมูลของประวัติการส่งหนังสือถูกต้อง เกิดจากการตรวจสอบประเภทข้อมูลตามเงื่อนไขประเภทข้อมูลโดยกำหนดให้ ค่า recID เป็นจำนวนเต็ม ค่า actionID เป็นจำนวนเต็ม

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
			ค่า createDate เป็นข้อมูลวันที่ ค่า recieveDate เป็นข้อมูลวันที่ ค่า sendDate เป็นข้อมูลวันที่ ค่า docRecIDFrom เป็นจำนวนเต็ม ค่า docID เป็นจำนวนเต็ม ค่า proposeID เป็นจำนวนเต็ม ค่า lineID เป็นจำนวนเต็ม ค่า personID เป็นจำนวนเต็ม ค่า signID เป็นจำนวนเต็ม
P14	ประวัติการส่งหนังสือครบถ้วน - ประวัติการส่งหนังสือครบถ้วน	when(recDomainCheck)[~docRecIDisINT | ~actionIDisINT | ~createDateIsDate | ~recieveDateisDate | ~sendDateisDate | ~docRecIDFromIsInt | ~docIDisINT | ~proposeIDisINT | ~lineIDisINT | ~personIDisINT | ~signIDisINT]	การเปลี่ยนสถานะจากประวัติการส่งหนังสือครบถ้วนไม่สามารถเปลี่ยนไปยังสถานะอื่นได้ เนื่องจากการตรวจสอบประเภทข้อมูลพบค่าใดค่าหนึ่งของเงื่อนไขไม่เป็นไปตามเค่าร่างฐานข้อมูล
P15	ประเภทข้อมูลของประวัติการส่งหนังสือถูกต้อง - หมายเลขของประวัติไม่ซ้ำ	when(recCheckID)[docRecisUnique]	การเปลี่ยนแปลงสถานะประเภทข้อมูลของประวัติการส่งหนังสือถูกต้องไปยังสถานะหมายเลขของประวัติไม่ซ้ำ เนื่องจากการสอบหมายเลขประจำของประวัติการส่ง

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
			หนังสือตามเงื่อนไขของคีย์หลักไม่พบค่า docRec ซ้ำกับค่าอื่นที่ปรากฏในตาราง DocRecieveSend
P16	ประเภทข้อมูลของประวัติการส่งหนังสือถูกต้อง - ประเภทข้อมูลของประวัติการส่งหนังสือถูกต้อง	when(recCheckID)[~docRecisUnique]	การเปลี่ยนแปลงสถานะประเภทข้อมูลของประวัติไม่สามารถเปลี่ยนแปลงไปยังสถานะอื่นได้ เนื่องจากการตรวจสอบหมายเลขประจำของประวัติการส่งหนังสือตามเงื่อนไขของคีย์หลัก พบค่า docRec ซ้ำกับค่าอื่นที่ปรากฏในตาราง DocRecieveSend
P17	หมายเลขของประวัติไม่ซ้ำ - ข้อมูลประวัติของการส่งหนังสือถูกต้อง	when(checkRefInfo)[docRecIDFromExist & docIDExist &proposeIDExist & lineIDExist &personIDExist & signIDExist]	การเปลี่ยนแปลงสถานะหมายเลขของประวัติไม่ซ้ำไปยังสถานะข้อมูลประวัติของการส่งหนังสือถูกต้อง เกิดจากการตรวจสอบข้อมูลอ้างอิงประกอบด้วย docID, proposeID, lineID, personID และ signID ซึ่งเป็นไปตามเงื่อนไขความสอดคล้องของข้อมูล และได้ผลลัพธ์คือข้อมูลดังกล่าวมีปรากฏอยู่ในตารางที่อ้างถึงจริง
P18	หมายเลขของประวัติไม่ซ้ำ - หมายเลขของประวัติไม่ซ้ำ	when(checkRefInfo)[~docRecIDFromExist | ~docIDExist |	การเปลี่ยนแปลงสถานะจากหมายเลขของประวัติไม่ซ้ำไม่สามารถเปลี่ยนแปลงสถานะ

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
		~proposeIDExist | ~lineIDExist | ~personIDExist | ~signIDExist]	ได้เนื่องจากผลจากการตรวจสอบข้อมูลอ้างอิงตัวใดตัวหนึ่งไม่ปรากฏค่าในตารางที่เกี่ยวข้อง
P19	หมายเลขของประวัติไม่ซ้ำ – จบการทำงาน	when(save)	การเปลี่ยนแปลงจากสถานะหมายเลขของประวัติไม่ซ้ำเข้าสู่สถานะสิ้นสุดการทำงานเกิดจากการบันทึกข้อมูลที่ตรวจสอบมาทั้งหมดลงฐานข้อมูล



รูปที่ 4.5 สถานะการดำเนินการกับข้อมูลฯขององค์ประกอบการเสนอลงนาม

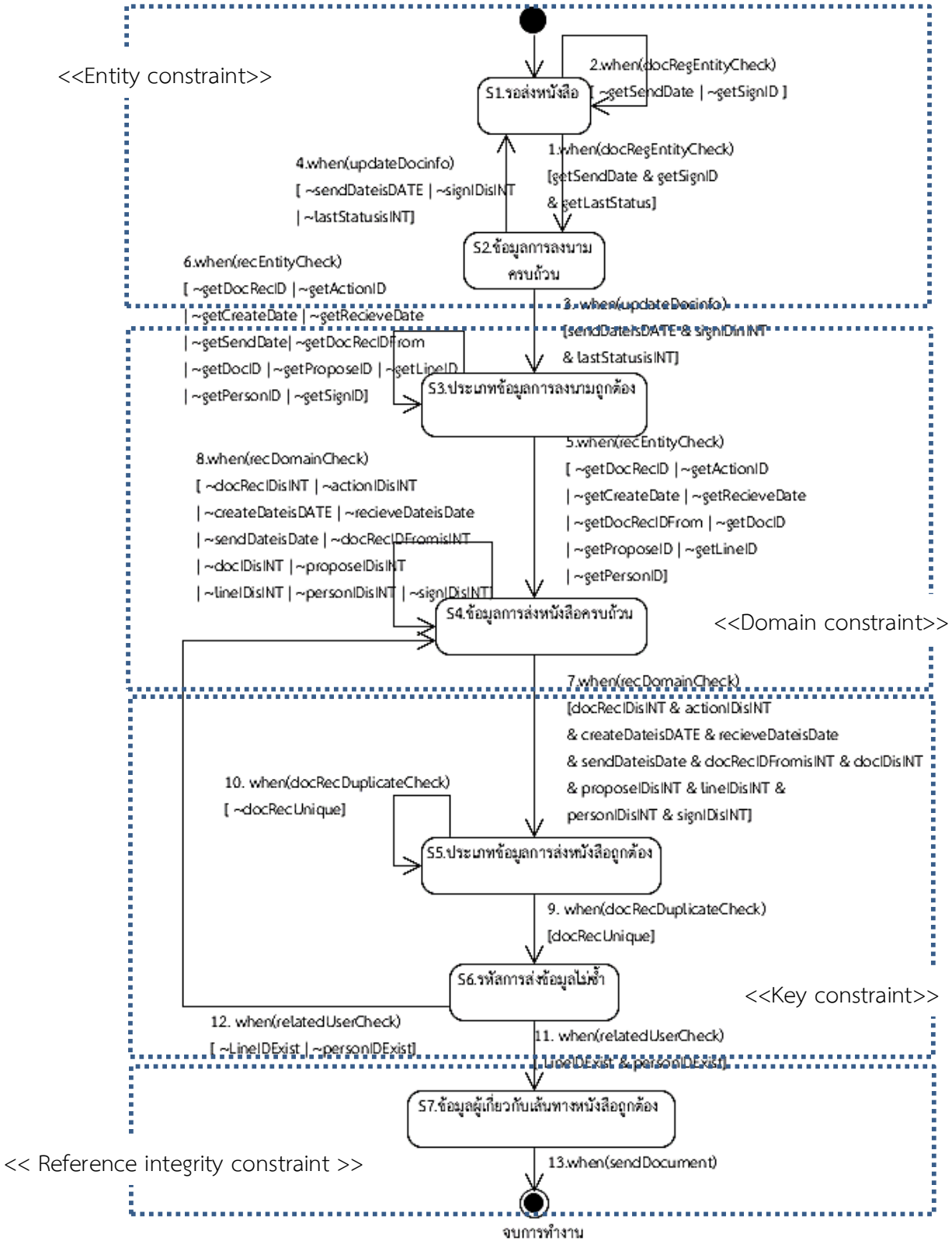
ตารางที่ 4.5 คำอธิบายแผนภาพสเตทแมชชีนขององค์ประกอบการเสนอลงนาม

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
P0	เริ่มต้น - ข้อมูลการลงนามถูกสร้าง	-	เป็นค่าปริยายที่การเริ่มต้นการทำงานขององค์ประกอบต้องเข้าสู่สถานะการนำเข้าข้อมูลหนังสือ
P1	ข้อมูลการลงนามถูกสร้าง - ข้อมูลการลงนามครบถ้วน	1.when(recEntityCheck)[getDocRecID & getActionID & getCreateDate & getRecieveDate & getDocRecIDFrom & getDocID & getProposeID & getLineID & getPersonID]	การเปลี่ยนสถานะในส่วนนี้เกิดจากการตรวจสอบเงื่อนไขค่าว่าง (recEntityCheck) และพบค่าของ docRecID, actionID, createDate, recieveDate, docRecIDFrom, docID, proposeID, lineID และ personID
P2	ข้อมูลการลงนามถูกสร้าง - ข้อมูลการลงนามถูกสร้าง	2.when(recEntityCheck)[~getDocRecID | ~getActionID | ~getCreateDate | ~getRecieveDate | ~getDocRecIDFrom | ~getDocID | ~getProposeID | ~getLineID | ~getPersonID]	การเปลี่ยนสถานะในส่วนนี้ไม่สามารถเปลี่ยนไปยังสถานะอื่นได้ เนื่องจากการตรวจสอบค่าว่าง และพบว่าค่าใดค่าหนึ่งภายในเงื่อนไข ปรากฏเป็นค่าว่าง
P3	ข้อมูลการลงนามครบถ้วน - ประเภทข้อมูลการลงนามถูกต้อง	3.when(recDomainCheck)[docRecIDisINT & actionIDisINT & createDateisDATE & recieveDateisDate & docRecIDFromisINT &	การเปลี่ยนสถานะในส่วนนี้เกิดจากการตรวจสอบเงื่อนไขประเภทข้อมูล (recDomainCheck) และพบว่าค่าของ docRecID, actionID, docRecIDFrom,

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
		docIDisINT & proposeIDisINT & lineIDisINT & personIDisINT]	docID, proposeID, lineID และ personID เป็นจำนวนเต็ม และค่า createDate และ recieveDate อยู่ในรูปแบบของวันที่
P4	ข้อมูลการลงนามครบถ้วน - ข้อมูลการลงนามถูกสร้าง	4.when(recDomainCheck)[~docRecIDisINT | ~actionIDisINT | ~createDateisDATE | ~recieveDateisDate | ~docRecIDFromisINT | ~docIDisINT | ~proposeIDisINT | ~lineIDisINT | ~personIDisINT]	การเปลี่ยนแปลงสถานะในส่วนนี้เป็นการย้อนกลับไปสู่สถานะข้อมูลการลงนามถูกสร้าง เนื่องจากการตรวจสอบประเภทข้อมูล (recDomainCheck) พบว่าค่าใดค่าหนึ่งของ docRecID, actionID, docRecIDFrom, docID, proposeID, lineID และ personID ไม่ใช่จำนวนเต็ม หรือ createDate และ recieveDate ไม่อยู่ในรูปแบบของวันที่
P5	ประเภทข้อมูลการลงนามถูกต้อง - รหัสข้อมูลการลงนามไม่ซ้ำ	5. when(docRecDuplicateCheck)[docRecUnique]	การเปลี่ยนแปลงสถานะในส่วนนี้เกิดจากการตรวจสอบเงื่อนไขของคีย์หลัก ซึ่งผลลัพธ์จะต้องไม่พบค่า docRec มีค่าซ้ำกัน
P6	ประเภทข้อมูลการลงนามถูกต้อง - ประเภทข้อมูลการลงนามถูกต้อง	6. when(docRecDuplicateCheck)[~docRecUnique]	การเปลี่ยนแปลงสถานะในส่วนนี้ไม่สามารถเปลี่ยนไปยังสถานะอื่นได้

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
			เนื่องจากการตรวจสอบเงื่อนไขของคีย์หลัก และพบค่า docRec ซ้ำกับค่าที่มีอยู่เดิมในฐานข้อมูล
P7	รหัสข้อมูลการลงนามไม่ซ้ำ - ข้อมูลหนังสือและการเกษียณถูกต้อง	7. when(docRefInfoCheck)[docIDExist & proposeIDExist]	การเปลี่ยนสถานะในส่วนนี้เกิดจากกาตรวจสอบข้อมูลที่เกี่ยวข้องกับการตรวจสอบเงื่อนไขความสอดคล้องของข้อมูล และพบว่าค่าของ docID และ proposeID มีปรากฏอยู่ก่อนการอ้างถึง
P8	รหัสข้อมูลการลงนามไม่ซ้ำ - ข้อมูลการลงนามถูกสร้าง	8. when(docRefInfoCheck)[~docIDExist | ~proposeIDExist]	การเปลี่ยนแปลงสถานะในส่วนนี้เป็นการย้อนไปสู่สถานะข้อมูลการลงนามถูกสร้าง ซึ่งเกิดจากการตรวจสอบความสอดคล้องของข้อมูล และพบว่าข้อมูลของ docID และ proposeID ที่อ้างถึงไม่มีปรากฏอยู่ในฐานข้อมูล
P9	ข้อมูลหนังสือและการเกษียณถูกต้อง - ข้อมูลผู้เกี่ยวข้องกับเส้นทางหนังสือถูกต้อง	9. when(relatedUserCheck)[LineIDExist & personIDExist]	การเปลี่ยนแปลงสถานะไปยังข้อมูลผู้เกี่ยวข้องกับเส้นทางหนังสือถูกต้อง เกิดจากการตรวจสอบข้อมูลบุคคลที่เกี่ยวข้อง ซึ่งประกอบด้วย lineID และ personID

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
			และปรากฏว่ามีข้อมูลที่อ้างถึงปรากฏอยู่ในฐานข้อมูล
P10	ข้อมูลหนังสือและการเกษียณถูกต้อง - ข้อมูลหนังสือและการเกษียณถูกต้อง	10. when(relatedUserCheck)[~LineIDExist | ~personIDExist]	การเปลี่ยนแปลงสถานะในส่วนนี้ไม่สามารถเปลี่ยนแปลงได้ เนื่องจากผลจากการตรวจสอบข้อมูลบุคคลที่เกี่ยวข้อง ซึ่งประกอบด้วย lineID และ personID แต่ไม่ปรากฏข้อมูลที่อ้างถึง
P11	ข้อมูลผู้เกี่ยวข้องกับเส้นทางหนังสือถูกต้อง – จบการทำงาน	11. when(save)	การเปลี่ยนแปลงจากสถานะหมายเลขของประวัติไม่ซ้ำเข้าสู่สถานะสิ้นสุดการทำงาน เกิดจากการบันทึกข้อมูลที่ตรวจสอบมาทั้งหมดลงฐานข้อมูล



รูปที่ 4.6 สถานะการดำเนินการกับข้อมูลฯขององค์ประกอบการส่งหนังสือถึงผู้รับ

ตารางที่ 4.6 คำอธิบายแผนภาพสเตทแมชชีนขององค์ประกอบการส่งหนังสือถึงผู้รับผิดชอบ

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
P0	เริ่มต้น - รอส่งหนังสือ	-	เป็นค่าปริยายที่การเริ่มต้นการทำงานขององค์ประกอบต้องเข้าสู่สถานะการนำเข้าข้อมูลหนังสือ
P1	รอส่งหนังสือ - ข้อมูลการลงนามครบถ้วน	1.when(docRegEntityCheck)[getSendDate & getSignID & getLastStatus]	การเปลี่ยนแปลงสถานะในส่วนนี้เกิดจากการตรวจสอบเงื่อนไขของค่าว่าง (docRegEntityCheck) และพบค่าของ signID และ lastStatus
P2	รอส่งหนังสือ - รอส่งหนังสือ	2.when(docRegEntityCheck)[~getSendDate | ~getSignID]	การเปลี่ยนแปลงสถานะในส่วนนี้ไม่สามารถเปลี่ยนแปลงไปยังสถานะอื่นได้ เนื่องจากการตรวจสอบเงื่อนไขของค่าว่าง (docRegEntityCheck) และพบค่าของ signID และ lastStatus
P3	ข้อมูลการลงนามครบถ้วน - ประเภทข้อมูลการลงนามถูกต้อง	3. when(updateDocinfo)[sendDateisDATE & signIDisINT & lastStatusisINT]	การเปลี่ยนแปลงจากสถานะในส่วนนี้ เกิดจากการอัพเดทข้อมูลของเอกสารและข้อมูลเหล่านั้นเป็นไปตามเงื่อนไขเชิงเค้าร่างคือ sendDate มีชนิดข้อมูลแบบวันที่

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
			ส่วน signID และ lastStatus มีข้อมูลเป็นจำนวนเต็ม
P4	ข้อมูลการลงนามครบถ้วน - รอส่งหนังสือ	4.when(updateDocinfo)[~sendDateisDATE | ~signIDisINT | ~lastStatusisINT]	การเปลี่ยนแปลงจากสถานะในส่วนนี้ ไม่สามารถเกิดขึ้นได้ เนื่องจากการอัพเดทข้อมูลของเอกสารและข้อมูลใดข้อมูลหนึ่งไม่เป็นไปตามเงื่อนไขเชิงเค้าร่างคือ sendDate ไม่ใช่ข้อมูลแบบวันที่หรือ signID และ lastStatus ไม่ใช่จำนวนเต็ม
P5	ประเภทข้อมูลการลงนามถูกต้อง - ข้อมูลการส่งหนังสือถูกสร้าง	5.when(recEntityCheck)[getDocRecID & getActionID & getCreateDate & getRecieveDate & getDocRecIDFrom & getDocID & getProposeID & getLineID & getPersonID]	การเปลี่ยนแปลงสถานะเกิดขึ้นจากการตรวจสอบเงื่อนไขของค่าว่าง (recEntityCheck) และพบค่า docID, actionID, createDate, recieveDate, docRecIDFrom, docID, proposeID, lineID และ personID ไม่เป็นค่าว่าง
P6	ประเภทข้อมูลการลงนามถูกต้อง - ประเภทข้อมูลการลงนามถูกต้อง	6.when(recEntityCheck)[~getDocRecID | ~getActionID | ~getCreateDate | ~getRecieveDate | ~getSendDate| ~getDocRecIDFrom | ~getDocID |	การเปลี่ยนแปลงสถานะในส่วนนี้ไม่สามารถเกิดขึ้นได้ เนื่องจากการตรวจสอบค่าว่าง ซึ่งไม่พบค่าของ docID, actionID, createDate, recieveDate, docRecIDFrom, docID, proposeID,

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
		~getProposeID | ~getLineID | ~getPersonID | ~getSignID]	lineID และ personID ตัวใดตัวหนึ่งเป็นค่าว่าง
P7	ข้อมูลการส่งหนังสือครบถ้วน - ประเภทข้อมูลการส่งหนังสือถูกต้อง	7.when(recDomainCheck)[docRecIDisINT & actionIDisINT & createDateisDATE & recieveDateisDate & sendDateisDate & docRecIDFromisINT & docIDisINT & proposeIDisINT & lineIDisINT & personIDisINT & signIDisINT]	การเปลี่ยนแปลงสถานะในส่วนนี้ เกิดขึ้นจากการตรวจสอบเงื่อนไขประเภทข้อมูล (recDomainCheck) ซึ่งพบว่า ค่าของ docRecID, actionID, docRecIdFrom, docID, proposeID, lineID, personID และ signID มีค่าเป็นจำนวนเต็ม ส่วน createDate, recieveDate, sendDate มีค่าเป็นตัวเลข
P8	ข้อมูลการส่งหนังสือครบถ้วน - ข้อมูลการส่งหนังสือครบถ้วน	8.when(recDomainCheck)[~docRecIDisINT | ~actionIDisINT | ~createDateisDATE | ~recieveDateisDate | ~sendDateisDate | ~docRecIDFromisINT | ~docIDisINT | ~proposeIDisINT | ~lineIDisINT | ~personIDisINT | ~signIDisINT]	การเปลี่ยนแปลงสถานะในส่วนนี้ เป็นการเปลี่ยนสถานะกลับไปยังสถานะข้อมูลการส่งหนังสือครบถ้วน ซึ่งเกิดจากผลลัพธ์ของการตรวจสอบตามเงื่อนไขประเภทข้อมูล และพบว่าค่าใดค่าหนึ่งของ docRecID, actionID, docRecIdFrom, docID, proposeID, lineID, personID และ signID ค่าใดค่าหนึ่งไม่ใช่จำนวนเต็ม หรือค่าใดค่าหนึ่งของ createDate,

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
			recieveDate และ sendDate ไม่ใช่ข้อมูลรูปแบบวันที่
P9	ประเภทข้อมูลการส่งหนังสือถูกต้อง - รหัสการส่งข้อมูลไม่ซ้ำ	9. when(docRecDuplicateCheck)[docRecUnique]	การเปลี่ยนแปลงสถานะในส่วนนี้ เป็นการเปลี่ยนแปลงไปยังสถานะรหัสการส่งข้อมูลไม่ซ้ำ เนื่องจากค่าของ docRecUnique ไม่ซ้ำกับข้อมูลเดิมในตาราง
P10	ประเภทข้อมูลการส่งหนังสือถูกต้อง - ประเภทข้อมูลการส่งหนังสือถูกต้อง	10. when(docRecDuplicateCheck)[~docRecUnique]	การเปลี่ยนแปลงในสถานะนี้ไม่สามารถเปลี่ยนแปลงไปสถานะอื่นได้ เนื่องจากค่าของ docRec มีปรากฏอยู่แล้วในฐานข้อมูล
P11	รหัสการส่งข้อมูลไม่ซ้ำ - ข้อมูลผู้เกี่ยวกับเส้นทางหนังสือถูกต้อง	11. when(relatedUserCheck)[LineIDExist & personIDExist]	การเปลี่ยนแปลงสถานะในส่วนนี้ เกิดจากการตรวจสอบข้อมูลบุคคลที่เกี่ยวข้อง และพบว่าข้อมูลที่อ้างถึงมีปรากฏอยู่ในฐานข้อมูลแล้ว
P12	รหัสการส่งข้อมูลไม่ซ้ำ ข้อมูลการส่งหนังสือครบถ้วน	12. when(relatedUserCheck)[~LineIDExist | ~personIDExist]	การเปลี่ยนแปลงสถานะในส่วนนี้ เป็นการย้อนกลับไปยังสถานะข้อมูลการส่งหนังสือครบถ้วน ซึ่งไม่พบข้อมูล lineID หรือ personID ปรากฏอยู่ในฐานข้อมูล

ลำดับที่	สถานะเริ่มต้น – สถานะปลายทาง	เหตุการณ์และเงื่อนไข	คำอธิบาย
P13	ข้อมูลการส่งหนังสือครบถ้วน – จบการทำงาน	13.when(sendDocument)	การเปลี่ยนแปลงจากสถานะหมายเลขของประวัติไม่ซ้ำเข้าสู่สถานะสิ้นสุดการทำงาน เกิดจากการบันทึกข้อมูลที่ตรวจสอบมาทั้งหมดลงฐานข้อมูล

4.3. การประเมินความเสี่ยงระดับองค์ประกอบ

หลังจากการรวบรวมข้อมูลเข้าสำหรับการประเมินความเสี่ยงเรียบร้อยแล้ว ขั้นตอนตอนต่อไปเป็นการประเมินความเสี่ยงระดับองค์ประกอบถูกแบ่งออกเป็นสองขั้นตอนย่อยคือ การประเมินความเสี่ยงของแต่ละองค์ประกอบของลำดับเหตุการณ์จากแผนภาพสเตทแมชชีน และการประเมินความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบ มีรายละเอียดดังนี้

4.3.1. การประเมินความเสี่ยงของแต่ละองค์ประกอบของลำดับเหตุการณ์จากแผนภาพสเตทแมชชีน

ขั้นตอนนี้สามารถคำนวณค่าปัจจัยความเสี่ยงขององค์ประกอบได้จากแบบจำลอง $rf_i^x = PUOCS_i \cdot svt_i$ และด้วยแบบจำลองนี้นำมาซึ่งขั้นตอนการคำนวณความน่าจะเป็นของโอกาสที่องค์ประกอบเกิดข้อผิดพลาด และระดับความรุนแรงเมื่อองค์ประกอบเกิดข้อผิดพลาดขึ้น รวมทั้งสิ้นหกขั้นตอน ดังนี้

1) คำนวณความน่าจะเป็นของผลลัพธ์ที่พึงปรารถนาขององค์ประกอบ (Probability of Satisfactory Component: PSC) ของการเปลี่ยนสถานะจาก p ไป q ผ่านหลักการสร้างกรณีทดสอบแบบครอบคลุมประพจน์ที่จะทำให้ได้มาซึ่งจำนวนของกรณีทดสอบ ด้วยแบบจำลอง

$$PSC_{pq} = \frac{|TrueTestCases_{pq}|}{|TotalTestCases_{pq}|}$$

ในส่วนนี้เป็นการนำเอาประพจน์ที่เป็นเงื่อนไขในการเปลี่ยนแปลงสถานะมากกระจายประพจน์ด้วยกฎการกระจาย เพื่อให้ได้นิพจน์หลังเกิดเหตุการณ์กระตุ้น สามารถทำได้โดยการสร้างตารางการเปลี่ยนแปลงสถานะ จากนั้นนำการเปลี่ยนแปลงดังกล่าวมาสร้างเป็นตารางข้อกำหนดรายละเอียดประพจน์ที่ทำการขยายตัวกระตุ้นเหตุการณ์ เมื่อได้ประพจน์ที่ผ่านการขยายแล้วจะนำเอาประพจน์ดังกล่าวมาสร้างความต้องการของกรณีทดสอบ สามารถแสดงรายละเอียดได้ดังตารางที่ ก.1 – ก.3 ตามลำดับ และผลลัพธ์การคำนวณหาจำนวนของกรณีทดสอบจากตารางที่ ก.3 สามารถสรุปได้ดังตารางที่ 4.7

ตารางที่ 4.7 จำนวณกรณีทดสอบขององค์ประกอบการลงทะเบียนหนังสือ

หมายเลขประพจน์	$\|TrueTestCases_{pq}\|$	$\|TotalTestCases_{pq}\|$	PSC_{pq}
P1	7	1	0.1428
P2	8	4	0.500
P3	14	1	0.0714
P4	15	11	0.7333
P5	5	1	0.2000
P6	6	2	0.3333
P7	4	1	0.2500
P8	4	1	0.2500
P9	4	1	0.2500
P10	4	1	0.2500
P11	11	1	0.0909
P12	12	8	0.7272
P13	14	1	0.0714
P14	15	11	0.7333
P15	4	1	0.2500
P16	4	1	0.2500
P17	9	1	0.1111
P18	10	6	0.6000
P19	3	1	0.3333

2) คำนวณหาค่าความน่าจะเป็นของเส้นทางที่พึงปรารถนาในแต่ละเส้นทางการทำงานภายในแผนภาพสเตทแมชชีน ด้วยแบบจำลอง

$$PSC_{initial-final} = \prod_{i=1}^{n-1} PSC_{P_iP_{i+1}}$$

การคำนวณในขั้นตอนนี้เริ่มต้นด้วยการหาเส้นทางที่พึงปราถนาจากแผนภาพสเตทแมชชีนขององค์ประกอบการลงทะเบียนหนังสือด้วยวิธีการหาเส้นทางการทดสอบแบบไพร์ม ซึ่งทำให้ได้เส้นทางการทดสอบเป็นจำนวนสองเส้นทางคือ

S1 – S2 – S3 – S4 – S5 – S6 – S7 – S8 – S9 – S10 และเมื่อคำนวณหาค่าความน่าจะเป็นของเส้นทางที่พึงปราถนาเส้นทางที่หนึ่งจะได้เท่ากับ 0.000000007669

S1 – S2 – S3 – S4 – S5 - S1 – S2 – S3 – S4 – S5 – S6 – S7 – S8 – S9 – S10 และเมื่อคำนวณหาค่าความน่าจะเป็นของเส้นทางที่พึงปราถนาเส้นทางที่หนึ่งจะได้เท่ากับ 0.000000000000978

3) คำนวณหาค่าของความน่าจะเป็นของเส้นทางที่ไม่พึงปรารถนา ด้วยแบบจำลอง

$$PUC_{initial-final} = 1 - PSC_{initial-final}$$

ขั้นตอนนี้เป็นการหาค่าความน่าจะเป็นของเส้นทางที่ไม่พึงปราถนา ซึ่งเป็นการนำเอาค่าสูงสุดของความน่าจะเป็นลบออกด้วยความน่าจะเป็นของเส้นทางที่พึงปราถนา ที่จะทำให้ได้มาซึ่งความน่าจะเป็นของเส้นทางที่ไม่พึงปรารถนาเท่ากับ 0.999999992331 และ 0.999999999999022 ตามลำดับ

4) คำนวณความน่าจะเป็นที่ไม่พึงปราถนาขององค์ประกอบ (Probability of Unsatisfactory Outcome: PUOC) ด้วยแบบจำลอง

$$PUOC_{\mathrm{i}} = \sum_{k=1}^{m} \left[\left[\frac{PP_k}{\sum_{l=1}^{m} PP_l} \right] \cdot \left(PUC_{initial-final} \right)_k \right]$$

ขั้นตอนนี้เป็นการคำนวณค่าความน่าจะเป็นขององค์ประกอบ ซึ่งเป็นการคำนวณโดยพิจารณาจากจำนวนของกรณีทดสอบของทุกเส้นทางที่เป็นไปได้ และจากขั้นตอนในข้างต้นสามารถระบุค่าให้กับปัจจัยทุกๆปัจจัยในแบบจำลองดังนี้

- m คือจำนวนเส้นทางที่เป็นไปได้ในแผนภาพสเตทแมชชีน ซึ่งองค์ประกอบของการลงทะเบียนหนังสือมีค่าเท่ากับ 2
- PP_1 คือผลคูณของกรณีทดสอบที่เป็นไปได้ของแต่ละสถานะภายในของเส้นทางที่เป็นไปได้ที่เส้นทางที่หนึ่งคือ (7 x 14 x 5 x 4 x 4 x 11 x 14 x 4 x 9 x 3) มีค่าเท่ากับ 130,394,880
- PP_2 คือผลคูณของกรณีทดสอบที่เป็นไปได้ของแต่ละสถานะภายในของเส้นทางที่เป็นไปได้ที่เส้นทางที่สองคือ (7 x 14 x 5 x 4 x 4 x 7 x 14 x 5 x 4 x 4 x 11 x 14 x 4 x 9 x 3) มีค่าเท่ากับ 1,022,295,859,200
- $\sum_{l=1}^{m} PP_l$ คือผลรวมของผลคูณของกรณีทดสอบที่เป็นไปได้ในทุกๆเส้นทาง มีค่าเท่ากับ 1,022,426,254,080

ดังนั้นความน่าจะเป็นขององค์ประกอบการลงทะเบียนหนังสือมีค่าเท่ากับ

$$\left[\left(\left(\frac{130,394,880}{1022426254080} \right) \cdot 0.999999992331 \right) + \left(\left(\frac{1022295859200}{1022426254080} \right) \cdot 0.999999999999022 \right) \right]$$

ซึ่งมีค่าเท่ากับ 0.999999999998044

5) ปรับค่าความน่าจะเป็นของผลลัพธ์ที่ไม่พึงปรารถนาขององค์ประกอบให้เข้ากับลำดับเหตุการณ์ที่ S_{x} ด้วยแบบจำลอง

$$PUOCS_{\mathrm{i}} = \frac{\mathrm{PPC_i}}{\sum_{\mathrm{j=1}}^{\mathrm{n}} \mathrm{PPC_j}} \cdot PUOC_{\mathrm{i}}$$

ลำดับเหตุการณ์ของการรับส่งหนังสือถึงผู้รับผิดชอบประกอบไปด้วยองค์ประกอบทั้งสิ้นสามองค์ประกอบด้วยกัน ซึ่งขั้นตอนในข้างต้นเป็นการคำนวณหาความน่าจะเป็นที่ไม่พึงปราถนาขององค์ประกอบการลงทะเบียนหนังสือ และเนื่องจากขั้นตอนนี้เป็นการพิจารณาทุกๆองค์ประกอบของลำดับเหตุการณ์ ดังนั้นผู้ประเมินต้องทำการประเมินความน่าจะเป็นที่ไม่พึงปราถนาขององค์ประกอบที่เหลือสององค์ประกอบตามขั้นตอนที่ได้แสดงไว้ในขั้นต้น ซึ่งผลลัพธ์ของการคำนวนความน่าจะเป็นที่ไม่พึงปรารถนาขององค์ประกอบการเสนอลงนามมีค่าเท่ากับ 0.99999997991193 และความน่าจะเป็นที่ไม่พึงปราถนาขององค์ประกอบการส่งหนึงสือถึงผู้รับผิดชอบมีค่าเท่ากับ 0.99999978027387 และเมื่อเข้าสู่กระบวนการปรับความน่าจะเป็นขององค์ประกอบให้เข้ากับลำดับเหตุการณ์การรับส่งหนังสือถึงผู้รับผิดชอบจะทำให้ ความน่าจะเป็นขององค์ประกอบทั้งสามเท่ากับ 0.4526627219 0.2455621252 และ 0.3017750816 ตามลำดับ

6) คำนวณระดับความรุนแรงจากจำนวนความสัมพันธ์ระหว่างตารางในฐานข้อมูลเชิงสัมพันธ์

ขั้นตอนนี้ผู้ประเมินต้องเป็นผู้กำหนดตารางในฐานข้อมูลที่มีความเกี่ยวข้องกับองค์ประกอบที่พิจารณา และตารางที่เกี่ยวข้องกับองค์ประกอบการลงทะเบียนหนังสือประกอบด้วย 6 ตาราง จากทั้งหมด 17 ตาราง คือ AttachedFile, DocRecieveSend, DocsecretLevel, DocSpeedLevel, DocType และ Document จากนั้นผู้ประเมินมีหน้ากำหนดระดับผลกระทบเมื่อเกิดข้อผิดผลาด โดยแบ่งระดับผลกระทบออกเป็นสามระดับคือ สูง ปานกลาง และต่ำ โดยกำหนดคะแนนของระดับแต่ละระดับคือ ห้า สาม หนึ่ง ตามลำดับ เมื่อกำหนดระดับผลกระทบให้กับคุณลักษณะประจำแล้ว จึงสามารถคำนวณระดับความรุนแรงได้จากแบบจำลอง

$$svt = \frac{\sum_{i=1}^{N} t_i}{T_{DB}}$$

สำหรับรายละเอียดของระดับผลกระทบในแต่ละตารางภายในฐานข้อมูล สามารถแสดงได้ดังตารางที่ 4.8

ตารางที่ 4.8 ระดับผลกระทบของแต่ละตารางในฐานข้อมูลและองค์ประกอบที่เกี่ยวข้อง

ลำดับที่	ชื่อตาราง	ระดับผลกระทบ (t_k)
1	AttachedFile	10
2	Department	10
3	DoccommentPropose	6
4	DocCommentSign	6
5	DocLineConfig	15
6	DocRecieveSend	25
7	DocSecretLevel	5
8	DocSpeedLevel	5
9	DocType	5
10	Document	24
11	Person	12
12	RecieveSendType	5
13	Node	12
14	Position	7
15	Prefix	4
16	SendRecieve	21
17	UMS	15

และข้อมูลจากตารางที่ 4.2 ทำให้สามารถคำนวณระดับความรุนแรงของทั้งสามองค์ประกอบเท่ากับ 0.395721925 0.245989305 และ 0.245989305 ตามลำดับ จากนั้นนำระดับความรุนแรงที่ได้ไปเปรียบเทียบกับค่าที่ได้กำหนดไว้ในขั้นตอนการวิเคราะห์ระดับความรุนแรงซึ่งขึ้นอยู่กับการกำหนดของผู้ประเมิน สำหรับกรณีศึกษานี้ผู้ประเมินได้วิเคราะห์ระดับความรุนแรงไว้ดังตารางที่ 4.9

ตารางที่ 4.9 ระดับภัยคุกคามตามความรุนแรง

ระดับความรุนแรง	ร้อยละของผลกระทบ	ระดับภัย
ระดับหายนะ	> 60%	1
ระดับวิกฤต	40% < x <= 60%	0.75
ระดับกลาง	20% < x <= 40%	0.5
ระดับต่ำ	< 20%	0.25

เมื่อนำระดับความรุนแรงที่ได้จากการคำนวณมาพิจารณาควบคู่กับการกำหนดระดับความรุนแรงให้เข้ากับมาตรฐานของภัยคุกคามที่กำหนดโดยนาซา (NASA) [2] แล้วจะทำให้ระดับภัยหรือผลกระทบขององค์ประกอบทั้งสามคือ 0.5 นั่นหมายถึงองค์ประกอบทั้งสามจะมีค่าความเสี่ยงอยู่ในระดับกลาง ซึ่งใช้เป็นเครื่องกำหนดสถานะความผิดพลาดในขั้นตอนการประเมินความเสี่ยงของลำดับเหตุการณ์ต่อไป

ขั้นตอนที่ผ่านมานำมาซึ่งความน่าจะเป็นที่ไม่พึงปราถนาขององค์ประกอบและระดับความเสียหายที่เกิดขึ้นกับความผิดพลาดที่อาจเกิดขึ้นขององค์ประกอบแต่ละองค์ประกอบ ซึ่งถูกนำเสนอในรูปแบบของค่าปัจจัยความเสี่ยง แสดงรายการได้ดังต่อไปนี้

1. ปัจจัยความเสี่ยงขององค์ประกอบการลงทะเบียนหนังสือ (rf_1^x) สามารถคำนวณได้จาก 0.4526627219 x 0.395721925 ได้ผลลัพธ์เท่ากับ 0.1791
2. ปัจจัยความเสี่ยงขององค์ประกอบการเสนอลงนาม (rf_2^x) สามารถคำนวณได้จาก 0.2455621252 x 0.245989305 ได้ผลลัพธ์เท่ากับ 0.0604
3. ปัจจัยความเสี่ยงขององค์ประกอบการส่งหนังสือถึงผู้รับผิดชอบ (rf_3^x) สามารถคำนวณได้จาก 0.3017750816 x 0.245989305 ได้ผลลัพธ์เท่ากับ 0.0742

4.4. ประเมินความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบของเหตุการณ์ใดๆจากแผนภาพสเตทแมชชีน

ความเสี่ยงของปฏิสัมพันธ์นี้พิจารณาจากความน่าจะเป็นที่องค์ประกอบเกิดปฏิสัมพันธ์ระหว่างกัน ซึ่งดูได้จากข้อความที่ส่งระหว่างองค์ประกอบภายในแผนภาพลำดับ และลำดับเหตุการณ์ที่นำเสนอได้ดังรูปที่ 4.3 สามารถหาค่าความน่าจะเป็นที่ปฏิสัมพันธ์เกิดข้อผิดพลาดขึ้นด้วยแบบจำลอง

$$EOC_{ij}^{x} = \frac{|MT_{ij}^{x}|_{i.j \epsilon S_x, i \neq j}}{|MT^{x}|}$$

และผลลัพธ์จากการคำนวณแสดงได้ดังตารางที่ 4.10

ตารางที่ 4.10 ความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบในลำดับเหตุการณ์การรับส่งหนังสือ

โหนดรับ / โหนดส่ง	การลงทะเบียนหนังสือ	การเสนอลงนาม	การส่งหนังสือถึงผู้รับผิดชอบ
การลงทะเบียนหนังสือ	0	0.4	0
การเสนอลงนาม	0	0	0.4
การส่งหนังสือถึงผู้รับผิดชอบ	0.2	0	0

ค่า EOC_{ij}^{x} ที่คำนวณได้ในข้างต้นเปรียบเทียบได้กับค่าความน่าจะเป็นที่เกิดความเสียหายระหว่างองค์ประกอบขึ้น ดังนั้นการที่จะได้มาซึ่งค่าความเสี่ยงนั้น ต้องพิจารณาถึงระดับความรุนแรงด้วย และวิธีการคำนวณระดับความรุนแรงกำหนดให้ใช้วิธีเดียวกันกับการคำนวณระดับความรุนแรงขององค์ประกอบ ดังนั้นค่าความเสี่ยงของปฏิสัมพันธ์ฯ มีรายละเอียดดังนี้

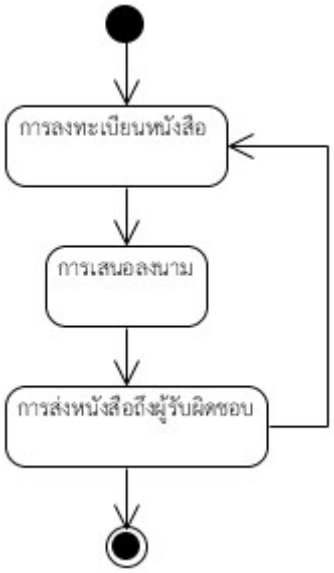
1. ค่าความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบการลงทะเบียนหนังสือและการเสนอลงนามมีค่าเท่ากับ 0.2032
2. ค่าความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบการเสนอลงนามและการส่งหนังสือถึงผู้รับผิดชอบมีค่าเท่ากับ 0.1112
3. ค่าความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบการส่งหนังสือถึงผู้รับผิดชอบและการลงทะเบียนหนังสือมีค่าเท่ากับ 0.1016

4.5. การประเมินความเสี่ยงภายในลำดับเหตุการณ์

ขั้นตอนการประเมินความเสี่ยงภายในลำดับเหตุการณ์ เป็นการประเมินความเสี่ยงหลังจากการประเมินความเสี่ยงในระดับองค์ประกอบ โดยแบ่งขั้นตอนการประเมินออกเป็นสองขั้นตอน ดังนี้

4.5.1. การสร้างเมทริกซ์ความน่าจะเป็นของการทำงานที่ปราศจากความผิดพลาดของลำดับเหตุการณ์

ขั้นตอนนี้เริ่มต้นจากการวิเคราะห์การทำงานร่วมกันของลำดับเหตุการณ์ที่แสดงด้วยแผนภาพลำดับ จากนั้นนำมาสร้างเป็นกราฟควบคุมการทำงานของซอฟต์แวร์ ดังนั้นหากพิจารณาแผนภาพลำดับในรูปที่ 4.3 แล้ว กราฟควบคุมการทำงานจึงมีลักษณะดังรูปที่ 4.7



รูปที่ 4.7 แบบจำลองพฤติกรรมการทำงานของลำดับเหตุการณ์การรับส่งหนังสือ

ขั้นตอนถัดไปเป็นการคำนวณหาเมทริกซ์ความน่าจะเป็น $P^x = [p_{ij}^x]$ ซึ่งเป็นเมทริกซ์ที่แสดงถึงความน่าจะเป็นที่ซอฟต์แวร์ทำงานได้สำเร็จ และเพื่อความสะดวกในการทำความเข้าใจ จึงแสดงผลเมทริกซ์ดังกล่าวในรูปแบบของตาราง ดังปรากฏในตารางที่ 4.11

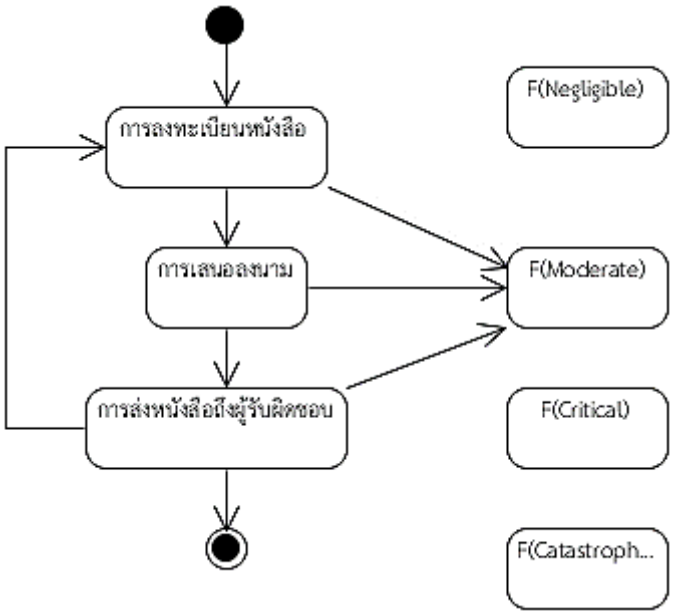
ตารางที่ 4.11 เมทริกซ์ความน่าจะเป็นเมื่อลำดับเหตุการณ์การรับส่งหนังสือทำงานได้สำเร็จ

โหนดรับ / โหนดส่ง	เริ่มต้น	การลงทะเบียนหนังสือ	การเสนอลงนาม	การส่งหนังสือถึงผู้รับผิดชอบ	สิ้นสุด
เริ่มต้น	0	1	0	0	0
การลงทะเบียนหนังสือ	0	0	1	0	0
การเสนอลงนาม	0	0	0	1	0
การส่งหนังสือถึงผู้รับผิดชอบ	0	0.5	0	0	0.5
สิ้นสุด	0	0	0	0	1

4.5.2. การสร้างเมทริกซ์ความน่าจะเป็นของการทำงานที่ผิดผลาดของลำดับเหตุการณ์

ขั้นตอนนี้เป็นการสร้างแบบจำลองความเสี่ยงของลำดับเหตุการณ์ซึ่งเป็นการพิจารณาค่าความเสี่ยงขององค์ประกอบและการเชื่อมโยงระหว่างองค์ประกอบ สามารถทำได้โดยการเพิ่มสถานะของความผิดพลาดเข้าสู่แบบจำลองการทำงานของซอฟต์แวร์ จากนั้นสร้างเส้นเชื่อมโยงระหว่างสถานะปกติเข้าสู่สถานะความผิดพลาดในระดับต่างๆ ซึ่งขึ้นอยู่กับระดับผลกระทบที่ได้จากขั้นตอน

การประเมินความเสี่ยงขององค์ประกอบ สามารถแสดงแบบจำลองความเสี่ยงของลำดับเหตุการณ์การรับส่งหนังสือมีลักษณะดังรูปที่ 4.8



รูปที่ 4.8 แบบจำลองความเสี่ยงของลำดับเหตุการณ์การรับส่งหนังสือ

ลำดับต่อไปเป็นการคำนวณหาค่าเมทริกซ์ความน่าจะเป็นของการเปลี่ยนแปลงจากสถานะของลจะเห็นว่าการได้มาซึ่งเมทริกซ์ที่ต้องการนั้นจะต้องผ่านขั้นตอนการคำนวณหาค่าเมทริกซ์ความน่าจะเป็นของการเปลี่ยนแปลงจากสถานะสถานะชั่วคราวเข้าสู่สถานะสถานะชั่วคราว (Q^x) และคำนวณเมทริกซ์ความน่าจะเป็นของการเปลี่ยนแปลงจากสถานะปกติเข้าสู่สถานะดูดกลืน (C^x) และผลลัพธ์ของการคำนวณแสดงได้ดังตารางที่ 4.12 – 4.15

ตารางที่ 4.12 เมทริกซ์ Q^x ของลำดับเหตุการณ์การรับส่งหนังสือ

โหนดรับ / โหนดส่ง	เริ่มต้น	การลงทะเบียนหนังสือ	การเสนอลงนาม	การส่งหนังสือถึงผู้รับผิดชอบ
เริ่มต้น	0	1	0	0
การลงทะเบียนหนังสือ	0	0	0.6540	0
การเสนอลงนาม	0	0	0	0.8350
การส่งหนังสือถึงผู้รับผิดชอบ	0	0.4158	0	0

ตารางที่ 4.13 เมทริกซ์ C^x ของลำดับเหตุการณ์การรับส่งหนังสือ

โหนดรับ / โหนดส่ง	สิ้นสุด	F(minor)	F(marginal)	F(critical)	F(catastophic)
เริ่มต้น	0	0	0	0	0
การลงทะเบียนหนังสือ	0	0	0.3459	0	0
การเสนอลงนาม	0	0	0.1649	0	0
การส่งหนังสือถึงผู้รับผิดชอบ	0.4628	0	0.1212	0	0

ตารางที่ 4.14 เมทริกซ์ A^x ของลำดับเหตุการณ์การรับส่งหนังสือ

โหนดรับ / โหนดส่ง	สิ้นสุด	F(Negligible)	F(Moderate)	F(Critical)	F(Catastrophic)
เริ่มต้น	0.3271	0	0.6729	0	0
การลงทะเบียนหนังสือ	0.3271	0	0.6729	0	0
การเสนอลงนาม	0.5001	0	0.4999	0	0
การส่งหนังสือถึงผู้รับผิดชอบ	0.5989	0	0.4011	0	0

ตารางที่ 4.15 เมทริกซ์ $A^{x\prime}$ ของลำดับเหตุการณ์การรับส่งหนังสือ

โหนดรับ / โหนดส่ง	สิ้นสุด	F(Negligible)	F(Moderate)	F(Critical)	F(Catastrophic)
เริ่มต้น	0.3271	0	0.6729	0	0
การลงทะเบียนหนังสือ	0.3271	0	0.6729	0	0
การเสนอลงนาม	0.5001	0	0.4999	0	0
การส่งหนังสือถึงผู้รับผิดชอบ	0.5989	0	0.4011	0	0

จากเมทริกซ์ความน่าจะเป็นในตารางที่ 4.15 สามารถนำมาสรุปผลแสดงปัจจัยความเสี่ยงของลำเหตุการณ์แต่ละลำดับเหตุการณ์ได้ดังตารางที่ 4.16

ตารางที่ 4.16 ปัจจัยความเสี่ยงของแต่ละลำดับเหตุการณ์ของซอฟต์แวร์ระบบสารบรรณ

ลำดับเหตุการณ์ / ระดับความรุนแรง	การรับส่งหนังสือ	การนำเข้าผู้ใช้ลงสู่ระบบ	การส่งหนังสือภายนอก
ระดับต่ำ (Negligible)	0	0.3107	0.3200
ระดับกลาง (Moderate)	0.6729	0	0.1186
ระดับวิกฤต (Critical)	0	0	0
ระดับหายนะ (Catastrophic)	0	0	0
ปัจจัยความเสี่ยงของลำดับเหตุการณ์	0.6729	0.3107	0.4386

ผลลัพธ์สุดท้ายที่ปรากฏในตารางที่ 4.16 แสดงให้เห็นว่าลำดับเหตุการณ์การรับส่งหนังสือมีความเสี่ยงที่จะทำให้เกิดข้อผิดพลาดกับข้อมูลที่จัดเก็บลงฐานข้อมูลมากที่สุด ซึ่งมีค่าความเสี่ยงในการเกิดข้อผิดพลาดรวมถึง 0.6729 และรองลงมาคือลำดับเหตุการณ์การส่งหนังสือภายนอกหน่วยงาน โดยมีค่าความเสี่ยงในการเกิดข้อผิดพลาดเท่ากับ 0.4386 และอันดับสุดท้ายคือลำดับเหตุการณ์การนำเข้าผู้ใช้ลงสู่ระบบมีค่าเท่ากับ 0.3107 และในลำดับเหตุการณ์การส่งหนังสือภายนอกหน่วยงานนั้นจะเห็นว่าค่าปัจจัยความเสี่ยงของลำดับเหตุการณ์เกิดขึ้นในสถานะความผิดพลาดทั้งในระดับต่ำและระดับกลาง ซึ่งเป็นการแสดงให้เห็นว่า องค์ประกอบของลำดับเหตุการณ์นี้ ทำให้เกิดความเสี่ยงในแต่ละดับไม่เท่ากัน อันเนื่องจากค่าปัจจัยความเสี่ยงขององค์ประกอบภายใน

4.6. วิธีการประเมินขั้นตอนวิธี

ในส่วนนี้ผู้วิจัยใช้วิธีการทวนสอบกับผู้เชี่ยวชาญในซอฟต์แวร์ระบบสารบรรณ ซึ่งเป็นผู้ทำหน้าที่ในการออกแบบโครงสร้างการทำงาน โครงสร้างฐานข้อมูล และพัฒนาซอฟต์แวร์ดังกล่าว ให้ความเห็นว่าลำดับเหตุการณ์การรับส่งหนังสือมีความเสี่ยงที่ทำให้ข้อมูลเกิดความผิดพลาดมากที่สุด รองลงมาเป็นการส่งหนังสือภายนอกหน่วยงาน และการนำเข้าผู้ใช้ลงสู่ระบบ ตามลำดับ ดังนั้นจึงสรุปได้ว่าขั้นตอนวิธีการประเมินความเสี่ยงที่นำเสนอ สามารถจัดอันดับความเสี่ยงของลำดับเหตุการณ์ภายในซอฟต์แวร์ได้ถูกต้อง

บทที่ 5

วิเคราะห์ ออกแบบ พัฒนาเครื่องมือการประเมินความเสี่ยง ของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน

ขั้นตอนวิธีสำหรับการประเมินความเสี่ยงที่ได้กล่าวมาแล้ว จะเห็นว่าการได้มาซึ่งปัจจัยความเสี่ยงของลำดับเหตุการณ์ในซอฟต์แวร์ ต้องผ่านการดำเนินงานในหลายขั้นตอนและมีความซับซ้อน ดังนั้นผู้วิจัยจึงพัฒนาเครื่องมือสำหรับการประเมินความเสี่ยงฯ และนำเสนอขั้นตอนการพัฒนาเครื่องมือดังกล่าว ซึ่งประกอบด้วยการรวบรวมและวิเคราะห์ความต้องการ การออกแบบ การพัฒนา และการทดสอบเครื่องมือ มีรายละเอียดดังนี้

5.1. การรวบรวมและวิเคราะห์ความต้องการ

ในส่วนการรวบรวมความต้องการสามารถทำได้โดยการศึกษาขั้นตอนวิธีการประเมินความเสี่ยงที่ได้นำเสนอ และสามารถแบ่งประเภทความต้องการเป็นสองส่วนคือความต้องการเชิงหน้าที่และความต้องการเชิงคุณภาพ ดังแสดงรายละเอียดของความต้องการในตารางที่ 5.1 – 5.2

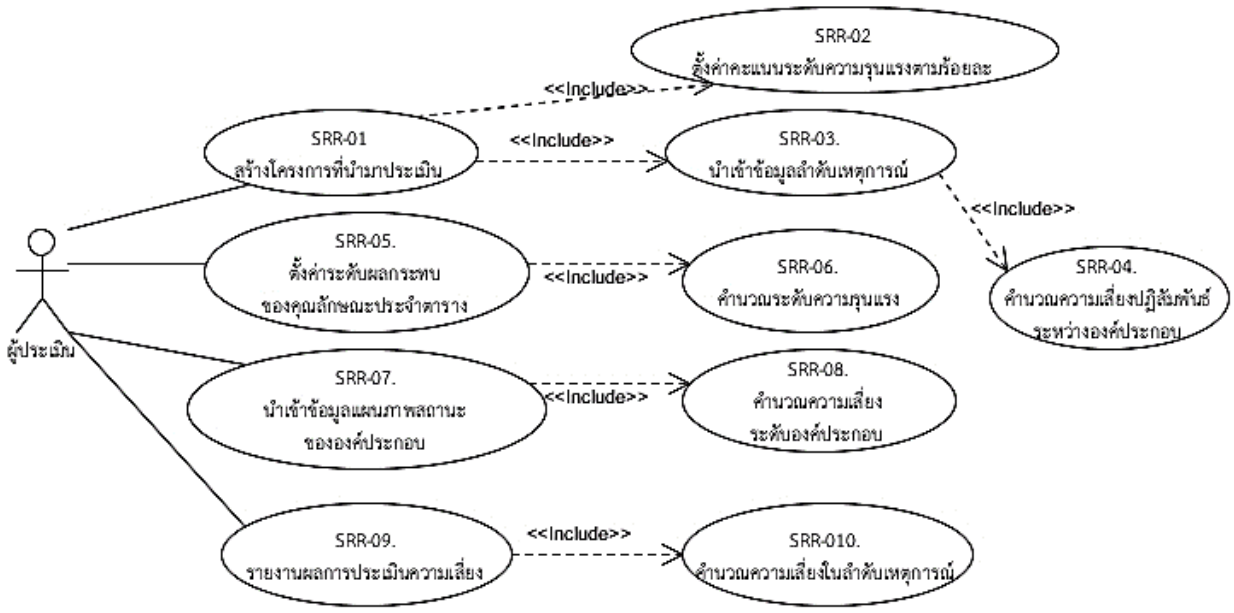
ตารางที่ 5.1 รายการความต้องการเชิงหน้าที่ของเครื่องมือการประเมินความเสี่ยงฯ

หมายเลขความต้องการ	รายการความต้องการ
FR1	รองรับการสร้างโครงการสำหรับประเมินความเสี่ยง ซึ่งต้องประกอบไปด้วยลำดับเหตุการณ์ และองค์ประกอบของลำดับเหตุการณ์
FR2	รองรับการตั้งค่าระดับความรุนแรงในแต่ละระดับความผิดพลาดเป็นรายโครงการ
FR3	รองรับการตั้งค่าระดับผลกระทบของแต่ละคุณลักษณะประจำตาราง โดยใช้เอกสารเอสคิวแอลเป็นข้อมูลนำเข้า
FR4	รองรับการนำเข้าแผนภาพลำดับของลำดับเหตุการณ์ในรูปแบบเอกสารเอ็กซ์เอ็มแอล
FR5	รองรับการนำเข้าแผนภาพสเตทแมชชีนขององค์ประกอบในรูปแบบเอกสารเอ็กซ์เอ็มแอล
FR6	รองรับขั้นตอนการประเมินความเสี่ยงในระดับองค์ประกอบทุกตามขั้นตอนวิธีที่นำเสนอไว้ในบทที่สาม
FR7	รองรับขั้นตอนการประเมินความเสี่ยงภายในลำดับเหตุการณ์
FR8	รองรับการรายงานผลปัจจัยความเสี่ยงของลำดับเหตุการณ์

ตารางที่ 5.2 รายการความต้องการเชิงคุณภาพของเครื่องมือการประเมินความเสี่ยงฯ

หมายเลขความต้องการ	รายการความต้องการ
NFR1	เครื่องมือที่พัฒนามีการออกแบบให้ง่ายต่อการใช้งาน (Usability)
NFR2	เครื่องมือที่พัฒนาต้องออกแบบให้ง่ายต่อการเข้าถึง (Availability)
NFR3	เครื่องมือที่พัฒนาต้องออกแบบให้ง่ายต่อการบำรุงรักษา (Maintainability)

จากรายการความต้องการเชิงหน้าที่ในข้างต้น สามารถวิเคราะห์เป็นความสามารถของเครื่องมือสำหรับการประเมินความเสี่ยงฯ ได้ดังแผนภาพยูสเคสดังปรากฏในรูปที่ 5.1 และเครื่องมือดังกล่าวถูกพัฒนาภายใต้ชื่อ “Scenario Risk Ranker” ซึ่งลำดับต่อไปผู้วิจัยจะเรียกว่า SRR มีรายละเอียดดังนี้



รูปที่ 5.1 ความสามารถในการทำงานของระบบ SRR

จากยูสเคสที่ปรากฏในข้างต้นสามารถอธิบายยูสเคสแบบพอสังเขปดังนี้

1. SRR-01 **สร้างโครงการที่นำมาประเมิน** ยูสเคสนี้ทำหน้าที่เป็นจุดเริ่มต้นของการทำงานของเครื่องมือประเมินความเสี่ยง ซึ่งเป็นการกำหนดชื่อโครงการที่นำมาประเมินหาค่าความเสี่ยง
2. SRR-02 **ตั้งค่าคะแนนระดับความรุนแรงของร้อยละ** ยูสเคสนี้ทำหน้าที่ในส่วนการตั้งค่าคะแนนของระดับความรุนแรงที่ถูกกำหนดไว้สี่ระดับ โดยผู้ใช้มีหน้าที่กำหนดช่วงร้อยละของระดับความรุนแรง และค่าความรุนแรงประจำช่วงต่างๆ เพื่อนำค่าความรุนแรงนี้ไปใช้ในขั้นตอนการประเมินความเสี่ยงควบคู่กับความน่าจะเป็นขององค์ประกอบ
3. SRR-03 **นำเข้าข้อมูลลำดับเหตุการณ์** ยูสเคสนี้ทำหน้าที่ในการนำเข้าข้อมูลของลำดับเหตุการณ์ ซึ่งประกอบด้วยชื่อของลำดับเหตุการณ์ และแผนภาพลำดับในรูปแบบของเอกสารเอ็กซ์เอ็มแอลของแต่ละลำดับเหตุการณ์ และการนำเข้าข้อมูลลำดับเหตุการณ์ในแต่ละครั้ง ผู้ใช้สามารถนำเข้าได้ครั้งละมากกว่าหนึ่งลำดับ
4. SRR-04 **คำนวณความเสี่ยงปฏิสัมพันธ์ระหว่างองค์ประกอบ** ยูสเคสนี้ทำหน้าที่คำนวณความเสี่ยงระหว่างองค์ประกอบที่ปรากฏในแผนภาพลำดับ โดยระบบจะเริ่มต้นการคำนวณให้อัตโนมัติหลังจากการนำเข้าแผนภาพลำดับของลำดับเหตุการณ์
5. SRR-05 **ตั้งค่าระดับผลกระทบจากคุณลักษณะประจำตาราง** ยูสเคสนี้ทำหน้าที่ในการรับข้อมูลการประเมินหรือการตั้งค่าระดับผลกระทบที่เกิดขึ้นจากคุณลักษณะประจำตารางภายในฐานข้อมูลเชิงสัมพันธ์ ซึ่งเครื่องมือรองรับระดับผลการประเมินคือสูง กลาง และต่ำ ซึ่งการทำงานในส่วนนี้จะเริ่มขึ้นต่อเมื่อผู้ใช้ได้นำเข้าเค้าร่างฐานข้อมูลในลักษณะเอกสารเอสคิวแอลเข้าสู่ระบบเป็นที่เรียบร้อยแล้ว
6. SRR-06 **คำนวณระดับความรุนแรง** ยูสเคสนี้ทำหน้าที่ในการคำนวณหาระดับความรุนแรงที่เกิดขึ้นในแต่ละตารางในฐานข้อมูลตามขั้นตอนที่ได้นำเสนอในบทที่สาม และการทำงานในส่วนนี้จะเกิดขึ้นหลังจากผู้ใช้ทำการตั้งค่าระดับผลกระทบในแต่ละคุณลักษณะประจำตารางแล้วเท่านั้น
7. SRR-07 **นำเข้าข้อมูลแผนภาพสเตทแมชชีนขององค์ประกอบ** ยูสเคสนี้ทำหน้าที่ในการนำเข้าข้อมูลขององค์ประกอบ ซึ่งประกอบด้วยชื่อขององค์ประกอบและแผนภาพสเตทแมชชีนในรูปแบบเอกสารเอ็กซ์เอ็มแอล และการนำเข้าองค์ประกอบนี้ผู้ใช้ต้องทำการเลือกลำดับเหตุการณ์ขององค์ประกอบที่ต้องการนำเข้าเสียก่อน จึงจะทำให้ระบบทำงานได้สำเร็จ
8. SRR-08 **คำนวณความเสี่ยงระดับองค์ประกอบ** ยูสเคสนี้ทำหน้าที่ในการคำนวณความเสี่ยงขององค์ประกอบตามขั้นตอนวิธีที่ได้นำเสนอในบทที่สาม ซึ่งการทำงานในส่วนนี้เริ่มขึ้นหลังจากผู้ใช้นำเข้าข้อมูลขององค์ประกอบเรียบร้อยแล้ว

9. SRR-09 รายงานผลการประเมินความเสี่ยง ยูสเคสนี้ทำหน้าที่ในการรายงานผลการประเมินความเสี่ยงหลังจากการดำเนินการตามขั้นตอนวิธีที่ได้นำเสนอ ซึ่งรายงานดังกล่าวสามารถรายงานค่าความเสี่ยงในระดับขององค์ประกอบ และในระดับของลำดับเหตุการณ์ได้

10. SRR-10 คำนวณความเสี่ยงในลำดับเหตุการณ์ ยูสเคสนี้ทำหน้าที่ในการสร้างเมทริกซ์ความน่าจะเป็นของการทำงานของซอฟต์แวร์ ตลอดจนคำนวณเมทริกซ์ความน่าจะเป็นที่ซอฟต์แวร์เริ่มต้นการทำงานและสิ้นสุดการทำงานด้วยความถูกต้อง ซึ่งเป็นไปตามขั้นตอนวิธีที่ได้นำเสนอในบทที่สาม

สำหรับคำอธิบายยูสเคสโดยละเอียดได้ยกตัวอย่างไว้ดังตารางที่ 5.3 และคำอธิบายยูสเคสทั้งหมดจะปรากฏได้ดังตารางที่ ข.1 – ข.10 ในภาคผนวก ข.

นอกเหนือจากการพิจารณาข้อกำหนดความต้องการเชิงหน้าที่แล้ว ความต้องการเชิงคุณภาพก็เป็นสิ่งที่พึงพิจารณาในขั้นตอนการออกแบบเช่นกัน ดังนั้นรายการปฏิบัติเพื่อให้สอดคล้องกับความต้องการเชิงคุณภาพจึงแสดงไว้ดังตารางที่ 5.4

ตารางที่ 5.3 ตัวอย่างคำอธิบายยูสเคส SRR-07

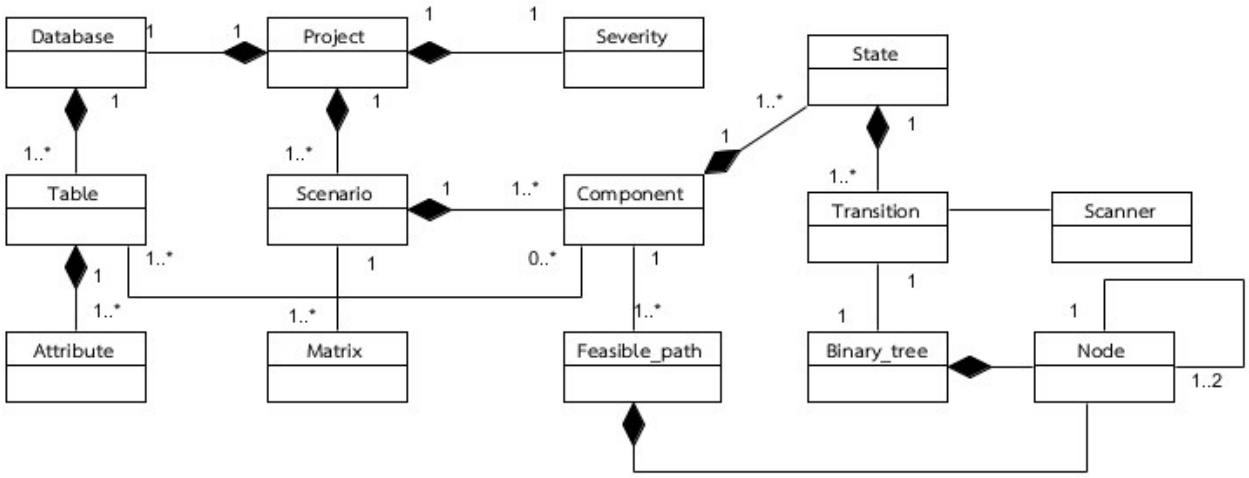
ชื่อยูสเคส: นำเข้าข้อมูลแผนภาพสเตทแมชชีนขององค์ประกอบ	**รหัส:** SRR-007	**ระดับความสำคัญ:** ปานกลาง
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงวิธีการนำเข้าข้อมูลแผนภาพสเตทแมชชีนขององค์ประกอบ		
สิ่งกระตุ้น : ผู้ใช้เลือกเมนูนำเข้าข้อมูลขององค์ประกอบ **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายนอก (External Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: คำนวณความเสี่ยงระดับองค์ประกอบ การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 1. หลังจากการตั้งค่าระดับความรุนแรงของคุณลักษณะประจำตารางแล้ว ระบบแสดงหน้าจอสำหรับนำเข้าข้อมูลองค์ประกอบ 2. ผู้ใช้เลือกลำดับเหตุการณ์ขององค์ประกอบที่ต้องการนำเข้า โดยที่ระบบแสดงรายการของลำดับเหตุการณ์ที่ถูกบันทึกไว้ในขั้นตอนการสร้างลำดับเหตุการณ์ 3. ผู้ใช้ระบุชื่อขององค์ประกอบ และเอกสารแผนภาพสเตทแมชชีนขององค์ประกอบในรูปแบบเอกสารเอ็กซ์เอ็มแอล 4. ระบบบันทึกข้อมูลขององค์ประกอบที่ได้นำเข้า		
ขั้นตอนการทำงานย่อย:		
ขั้นตอนการทำงานทางเลือก/พิเศษ: 2.1 ผู้ใช้สามารถนำเข้าข้อมูลขององค์ประกอบเข้าสู่ระบบได้มากกว่าหนึ่งองค์ประกอบ โดยการกดปุ่มเพิ่มจำนวนองค์ประกอบ โดยปราศจากการประมวลผลไปยังหน้าการทำงานอื่น		

ตารางที่ 5.4 รายการปฏิบัติตามความต้องการเชิงคุณภาพ

ลำดับ	รายการความต้องการ	รายการปฏิบัติ
1	NFR1	การออกแบบมีการหลีกเลี่ยงการนำเข้าข้อมูลจากผู้ใช้ให้มากที่สุด ซึ่งจะเห็นได้จากขั้นตอนการตั้งค่าระดับผลกระทบของคุณลักษณะประจำตาราง ที่มีการสกัดเอาข้อมูลที่ใช้ในการตั้งค่า จึงทำให้หน้าที่ของผู้ใช้เหลือเพียงการเลือกระดับผลกระทบเท่านั้น
2	NFR2	เครื่องมือสำหรับประเมินความเสี่ยงนี้ ถูกพัฒนาให้อยู่ในลักษณะของเว็บเบสแอปพลิเคชัน จึงทำให้การประเมินความเสี่ยงสามารถทำได้ทุกที่ที่มีการเชื่อมต่อผ่านเครือข่าย
3	NFR3	การพัฒนาเครื่องมือนี้ถูกพัฒนาด้วยภาษาพีเอชพี (PHP) ซึ่งเป็นตัวแปลภาษาแบบเปิดเผยรหัส (Open source) ที่มีการพัฒนาไปอย่างรวดเร็ว ดังนั้นจึงกำหนดให้การพัฒนาเครื่องมือใช้ตัวแปลภาษาเวอร์ชัน 5.3.26 ซึ่งเป็นรุ่นที่มีเสถียรภาพและจัดอยู่ในรุ่นที่ผู้ผลิตยังคงบำรุงรักษาต่อไป นอกจากนี้ยังได้กำหนดเงื่อนไขในการพัฒนารหัสโปรแกรม เช่น การตั้งชื่อคลาสเป็นตัวพิมพ์ใหญ่ การตั้งชื่อฟังก์ชันด้วยตัวพิมพ์เล็ก การเขียนคอมเม้นท์สำหรับส่วนการทำงานที่ทำความเข้าใจได้ยาก เป็นต้น

5.2. การออกแบบเครื่องมือการประเมินความเสี่ยง

หลังจากขั้นตอนการวิเคราะห์ความต้องการจนได้เป็นรายการความสามารถของเครื่องมือที่นำเสนอผ่านแผนภาพยูสเคสนั้น ลำดับต่อไปเป็นการออกแบบโครงสร้างการทำงานของเครื่องมือการประเมินความเสี่ยงผ่านการนำเสนอด้วยแผนภาพคลาส แสดงได้ดังรูปที่ 5.2



รูปที่ 5.2 โครงสร้างการทำงานของเครื่องมือการประเมินความเสี่ยงฯ

แผนภาพคลาสในข้างต้นประกอบไปด้วยการทำงานร่วมกันของคลาสจำนวน 14 คลาส ซึ่งแต่ละคลาสมีคุณสมบัติประจำและหน้าที่ดังต่อไปนี้

1) คลาส Project มีคุณสมบัติในการจัดเก็บข้อมูลโครงการที่นำมาประเมิน และทำหน้าที่บริหารจัดการข้อมูลที่เกี่ยวข้องกับโครงการ ซึ่งคุณสมบัติบางรายการมีคุณสมบัติเป็นวัตถุของคลาส Database คลาส Scenario และ คลาส Severity

2) คลาส Database มีคุณสมบัติในการจัดเก็บข้อมูลที่เกี่ยวข้องกับฐานข้อมูล และทำหน้าที่บริหารจัดการข้อมูลและการคำนวณระดับความรุนแรงของฐานข้อมูล

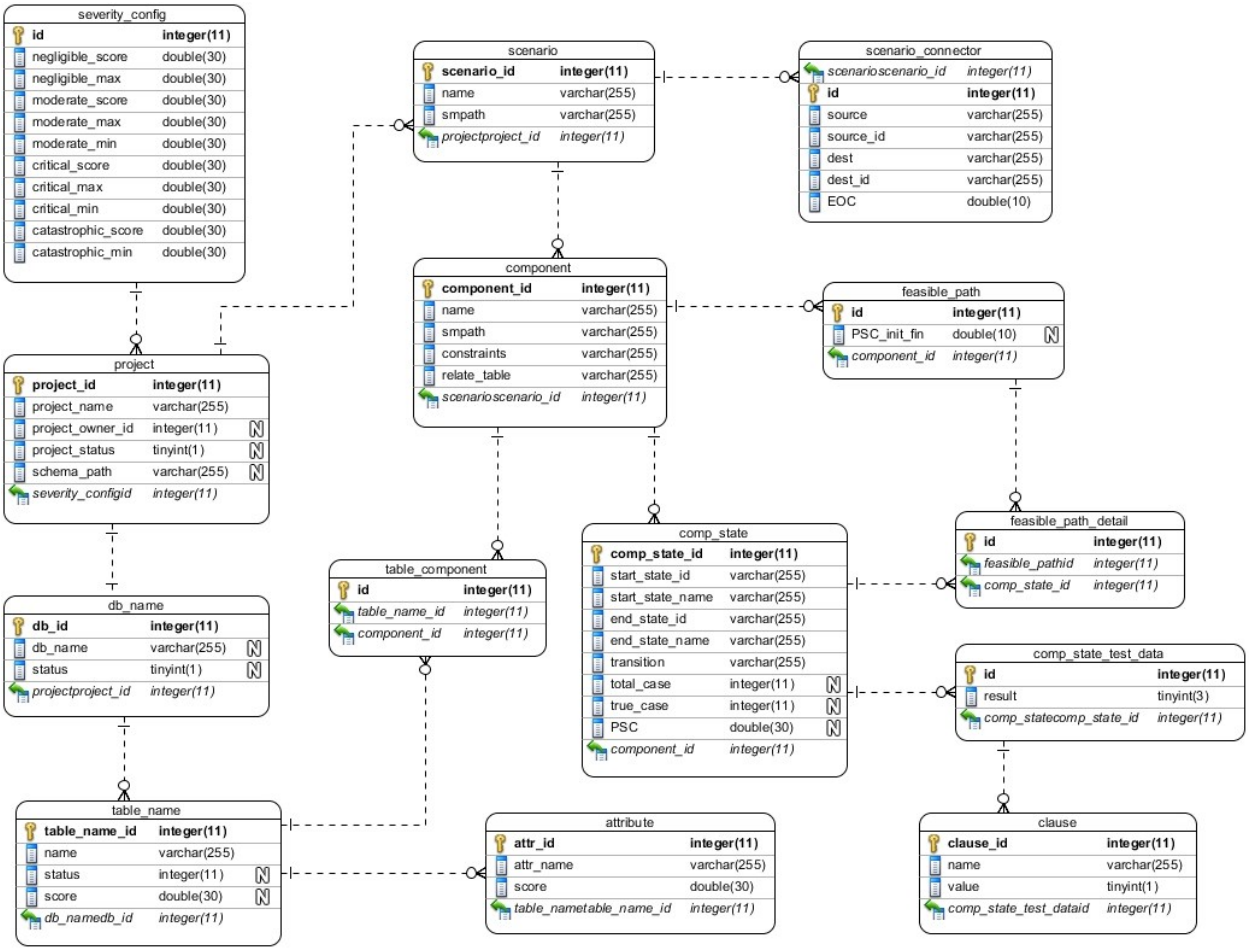
3) คลาส Table มีคุณสมบัติในการจัดเก็บข้อมูลที่เกี่ยวข้องกับตาราง และทำหน้าที่บริหารจัดการข้อมูลรวมถึงการคำนวณระดับความรุนแรงของตาราง

4) คลาส Attribute มีคุณสมบัติในการจัดเก็บข้อมูลของคุณสมบัติประจำตาราง และทำหน้าที่บริหารจัดการข้อมูล

5) คลาส Severity มีคุณสมบัติในการจัดเก็บช่วงของร้อยละและคะแนนของระดับความรุนแรง และทำหน้าที่บริหารจัดการข้อมูล

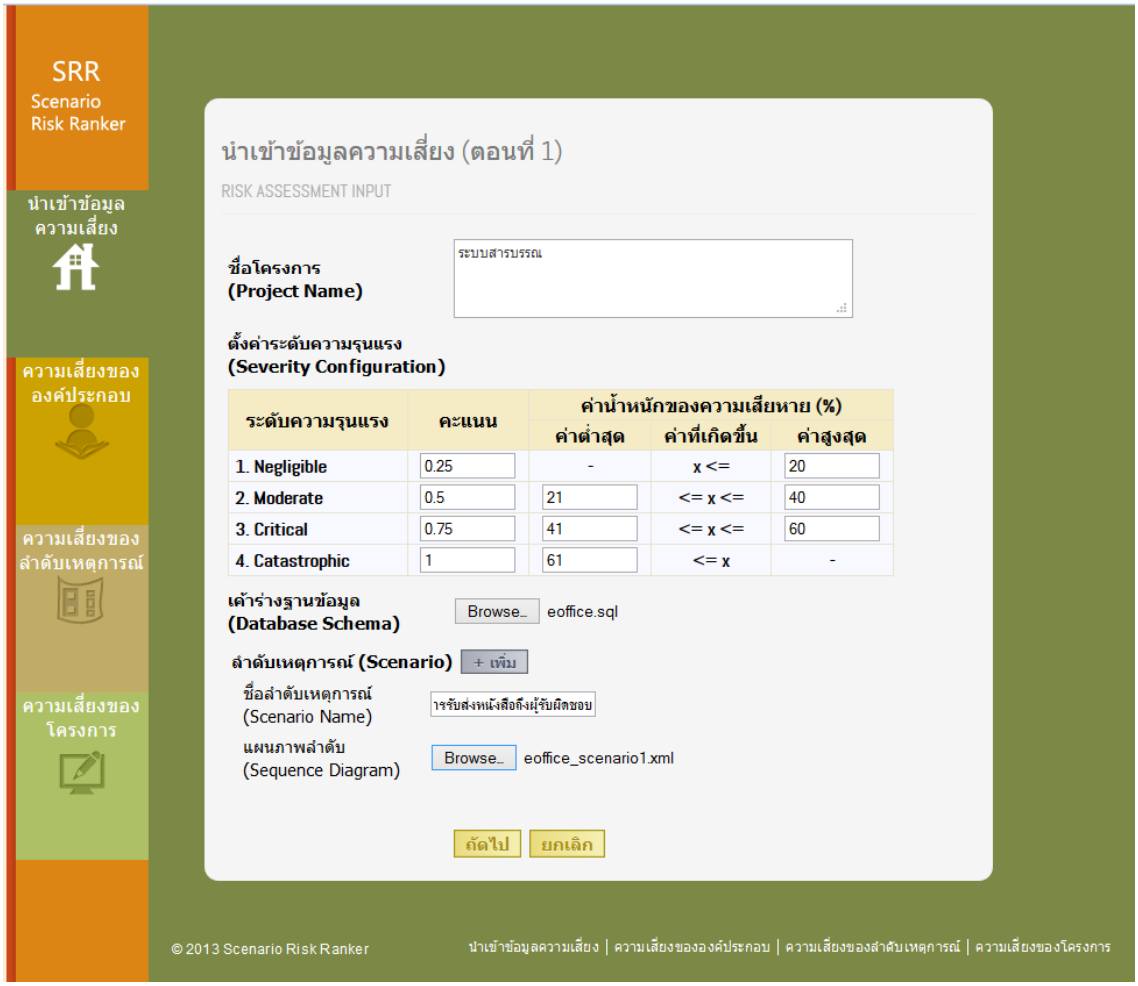
6) คลาส Scenario มีคุณสมบัติในการจัดเก็บข้อมูลของลำดับเหตุการณ์ และมีหน้าที่สร้างแบบจำลองการทำงานของซอฟต์แวร์ รวมถึงการสร้างแบบจำลองความเสี่ยงของลำดับเหตุการณ์
7) คลาส Matrix มีคุณสมบัติในการจัดเก็บข้อมูลใดๆในลักษณะของเมทริกซ์ และมีหน้าที่ดำเนินการกับเมทริกซ์ด้วยตัวดำเนินการ บวก ลบ คูณ การหาอินเวิร์สการคูณ และการหาเมทริกซ์เอกลักษณะ
8) คลาส Component มีคุณสมบัติในการจัดเก็บข้อมูลขององค์ประกอบ และมีหน้าที่ในการบริหารจัดการ รวมถึงการสกัดเอาเงื่อนไขบนการเปลี่ยนแปลงระหว่างสถานะเพื่อเข้าสู่กฎการกระจายหลังการกระตุ้น
9) คลาส Feasible_path มีคุณสมบัติในการจัดเก็บเส้นทางการทดสอบที่เป็นไปได้ และมีหน้าที่ในการคำนวณหาเส้นทางการทดสอบที่เป็นไปได้ด้วยวิธีแบบไพร์ม
10) คลาส State มีคุณสมบัติในการจัดเก็บข้อมูลสถานะขององค์ประกอบในลำดับเหตุการณ์ และมีหน้าที่ในการคำนวณหาจำนวนกรณีทดสอบทั้งหมดและกรณีทดสอบที่มีค่าเป็นจริง
11) คลาส Transition มีคุณสมบัติในการจัดเก็บกรณีทดสอบทุกกรณีที่เกิดขึ้นจากการประยุกต์ใช้หลักการณ์แบบครอบคลุมประพจน์ และมีหน้าที่บริหารจัดการข้อมูลของกรณีทดสอบ
12) คลาส Binary_tree มีคุณสมบัติในการจัดเก็บรากของต้นไม้แบบทวิภาค และทำหน้าที่ในการต่อยอดโหนด รวมถึงการค้นคืนต้นไม้แบบล่างขึ้นบน (Depth-First Search)
13) คลาส Node มีคุณสมบัติในการจัดเก็บค่าของประโยคและตัวชี้ด้านซ้ายและขวา และมีหน้าที่ในการสร้างโหนด
14) คลาส Scanner มีคุณสมบัติในการจัดเก็บข้อมูลในลักษณะโทเคน (Token) และมีหน้าที่สร้างรายการโทเคน (Token list) จากประพจน์หลังการกระจาย

และเนื่องจากเครื่องมือที่พัฒนาได้พัฒนาในรูปแบบของเว็บเบสแอปพลิเคชัน ดังนั้นจึงนำเสนอแผนภาพความสัมพันธ์ของเอ็นทิตี้เพื่อแสดงถึงโครงสร้างการจัดเก็บข้อมูลของเครื่องมือสำหรับการประเมินความเสี่ยง ดังรูปที่ 5.3



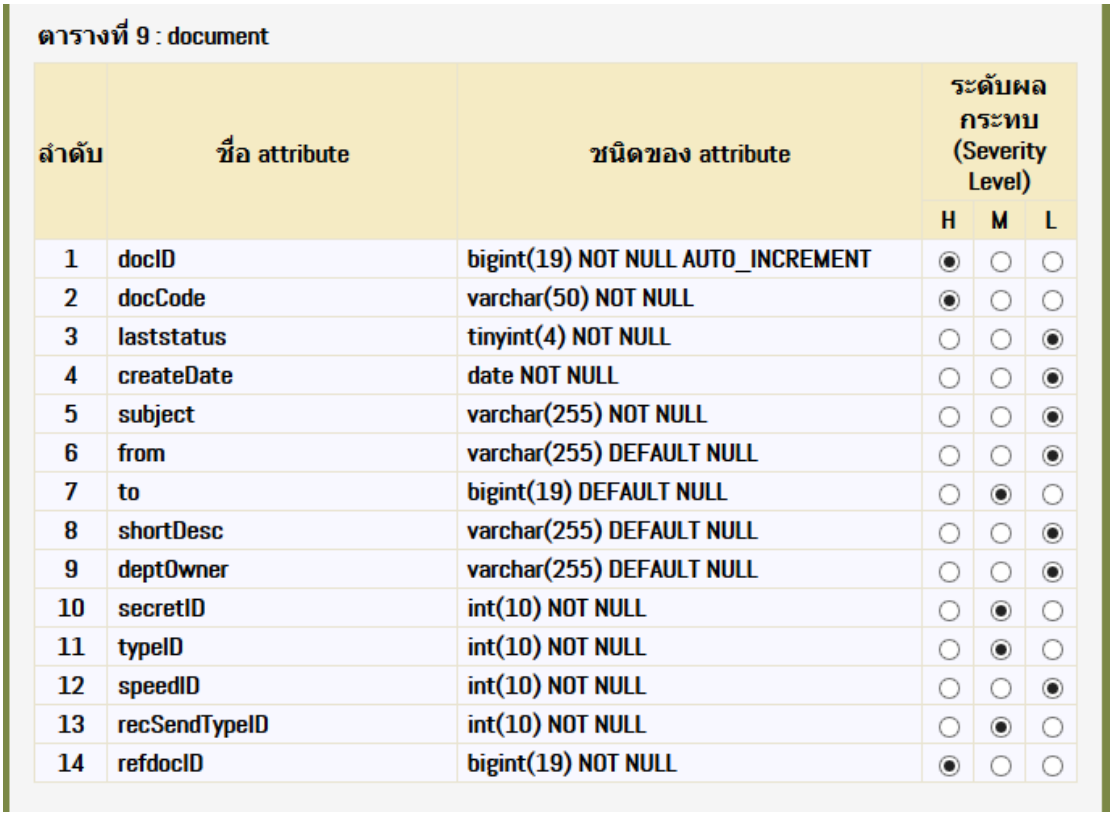
รูปที่ 5.3 โครงสร้างที่ใช้จัดเก็บข้อมูลของเครื่องมือสำหรับประเมินความเสี่ยง

นอกจากการออกแบบโครงสร้างการทำงานและการจัดเก็บข้อมูลของเครื่องมือสำหรับการประเมินแล้ว ในบทนี้ยังได้นำเสนอการออกแบบส่วนต่อประสานโดยมีวัตถุประสงค์เพื่อให้ผู้ใช้งานสามารถประเมินความเสี่ยงได้สะดวกมากยิ่งขึ้น มีรายละเอียดดังรูปที่ 5.4 – 5.9



รูปที่ 5.4 ส่วนการนำเข้าข้อมูลโครงการที่นำมาประเมินความเสี่ยง

การนำเข้าข้อมูลความเสี่ยงในรูป 5.4 เป็นขั้นตอนแรกของการประเมินความเสี่ยงของโครงการขั้นตอนแรกด้วยการใช้เครื่องมือสำหรับประเมินความเสี่ยง ซึ่งประกอบด้วยการนำเข้าชื่อโครงการที่นำมาประเมินความเสี่ยง การตั้งค่าระดับความรุนแรง ส่วนการนำเข้าเค้าร่างฐานข้อมูล และส่วนการนำเข้าลำดับเหตุการณ์ที่ออกแบบให้ผู้ใช้สามารถเพิ่มลำดับเหตุการณ์ได้โดยปราศจากการประมวลผลไปยังหน้าการทำงานอื่น ซึ่งจะทำให้ผู้ใช้สามารถเพิ่มลำดับเหตุการณ์ภายในโครงการได้สะดวกยิ่งขึ้น



ตารางที่ 9 : document

ลำดับ	ชื่อ attribute	ชนิดของ attribute	ระดับผลกระทบ (Severity Level)		
			H	M	L
1	docID	bigint(19) NOT NULL AUTO_INCREMENT	◉	○	○
2	docCode	varchar(50) NOT NULL	◉	○	○
3	laststatus	tinyint(4) NOT NULL	○	○	◉
4	createDate	date NOT NULL	○	○	◉
5	subject	varchar(255) NOT NULL	○	○	◉
6	from	varchar(255) DEFAULT NULL	○	○	◉
7	to	bigint(19) DEFAULT NULL	○	◉	○
8	shortDesc	varchar(255) DEFAULT NULL	○	○	◉
9	deptOwner	varchar(255) DEFAULT NULL	○	○	◉
10	secretID	int(10) NOT NULL	○	◉	○
11	typeID	int(10) NOT NULL	○	◉	○
12	speedID	int(10) NOT NULL	○	○	◉
13	recSendTypeID	int(10) NOT NULL	○	◉	○
14	refdocID	bigint(19) NOT NULL	◉	○	○

รูปที่ 5.5 ส่วนการตั้งค่าระดับผลกระทบของคุณลักษณะประจำตาราง

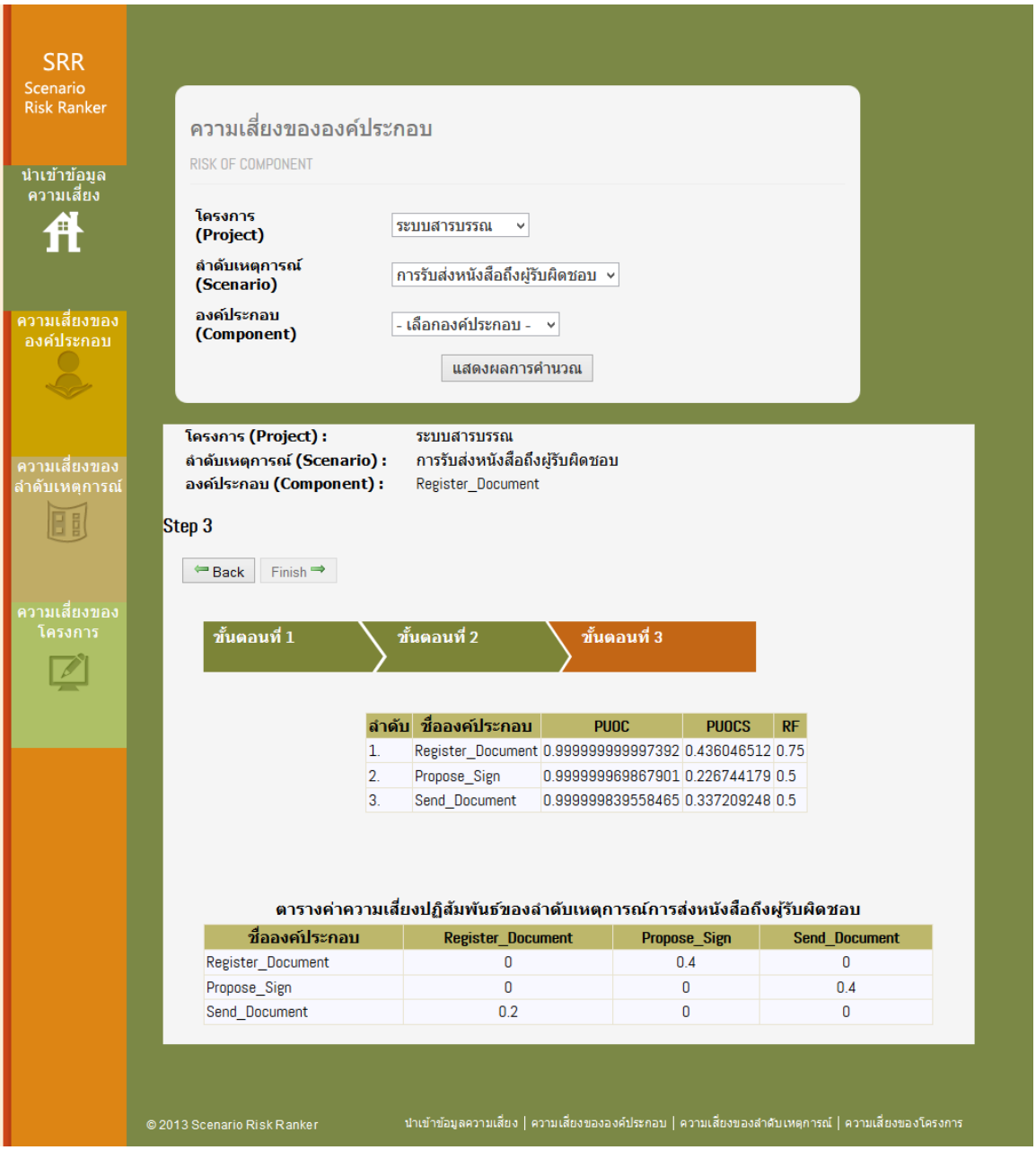
การตั้งค่าระดับผลกระทบของคุณลักษณะประจำตารางในรูปที่ 5.4 เป็นการออกแบบหน้าจอเพื่อรองรับการกำหนดค่าระดับผลกระทบของคุณลักษณะประจำตาราง ซึ่งคำนึงถึงความง่ายต่อการใช้งานและการควบคุมข้อมูลนำเข้าให้มีความถูกต้องมากที่สุด กล่าวคือ ชื่อฐานข้อมูล ชื่อตาราง และรายการคุณลักษณะประจำที่ปรากฏในส่วนการตั้งค่านี้ได้จากการใช้โปรแกรมสกัดเอาข้อมูลดังกล่าวจากเอกสารเชิงเค้าร่างฐานข้อมูลที่ได้นำเข้าจากส่วนการนำเข้าข้อมูลโครงการประเมินความเสี่ยงในขั้นตอนแรก นอกจากนี้ยังออกแบบลักษณะการนำเข้าข้อมูลแบบตัวเลือกเดียวเพื่อทำให้ได้มาซึ่งผลลัพธ์การคำนวณระดับความรุนแรงที่ถูกต้อง

รูปที่ 5.6 ส่วนการนำเข้าข้อมูลองค์ประกอบ

การนำเข้าข้อมูลองค์ประกอบดังปรากฏในรูปที่ 5.5 นั้นเป็นขั้นตอนที่เกิดขึ้นหลังจากการกำหนดระดับผลกระทบของฐานข้อมูล ซึ่งประกอบด้วย การเลือกลำดับเหตุการณ์และองค์ประกอบที่ได้จากการประมวลผล

ขั้นตอนการนำเข้าลำดับเหตุการณ์ และเมื่อทำการเลือกข้อมูลทั้งสองส่วนแล้ว ขั้นต่อไปเป็นการนำเข้าเอกสารแผนภาพสเตทแมชชีนในรูปแบบเอ็กซ์เอ็มแอลของแต่ละองค์ประกอบเข้าสู่ระบบ และขั้นตอนสุดท้ายเป็นเลือกตารางของฐานข้อมูลที่เกี่ยวข้องกับองค์ประกอบ

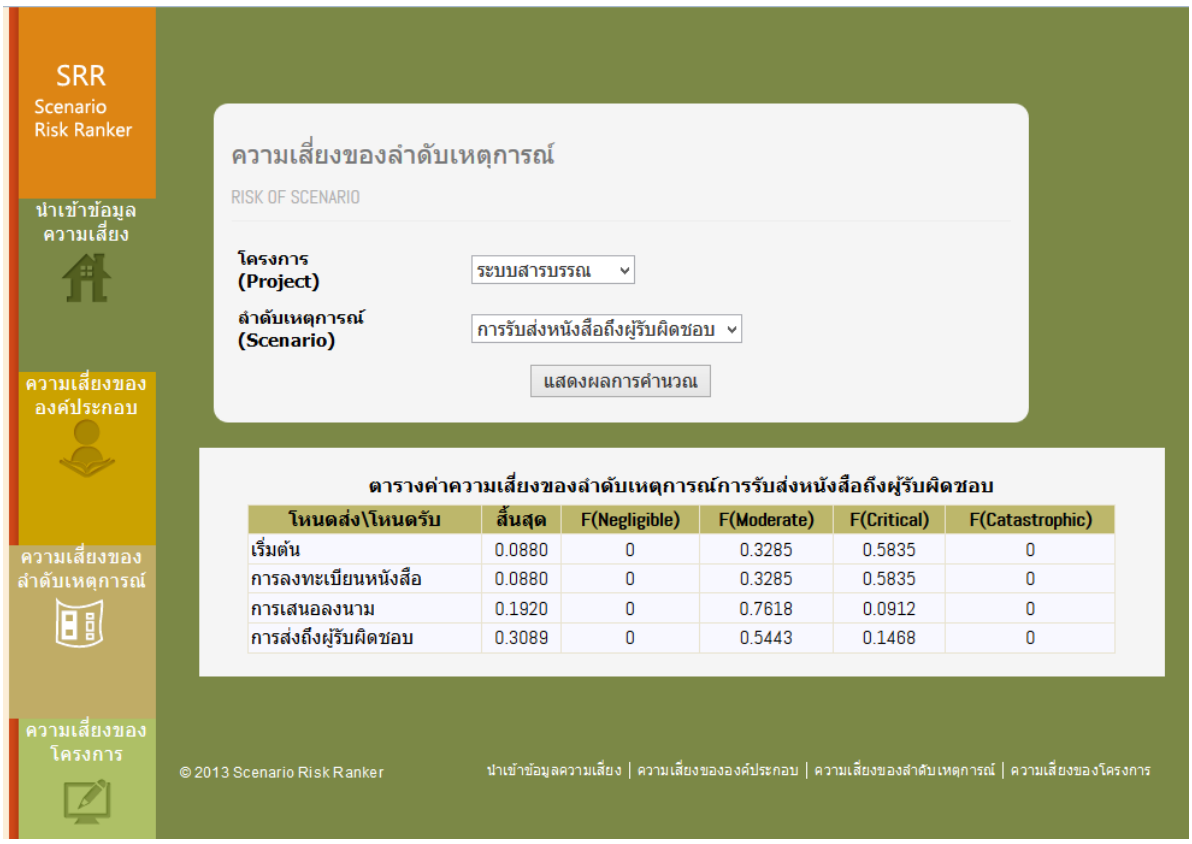
จากการออกแบบการทำงานในส่วนนี้จะเห็นได้ว่าผู้พัฒนาได้คำนึงถึงการควบคุมการนำเข้าข้อมูลที่ถูกต้อง เช่น การเลือกรายการองค์ประกอบที่เกิดจากการสกัดเอาองค์ประกอบที่ปรากฏอยู่จริงในแผนภาพลำดับของลำดับเหตุการณ์ และการเลือกตารางของฐานข้อมูลที่เกี่ยวข้องจากรายการตารางที่ปรากฏจริงในเอกสารเชิงเค้าร่างฐานข้อมูล นอกจากนี้ยังออกแบบให้ผู้ใช้สามารถเพิ่มรายการอองค์ประกอบของลำดับเหตุการณ์หนึ่งๆในโครงการได้โดยปราศจากการประมวลผลไปยังหน้าการทำงานอื่น ซึ่งเป็นลักษระโดยทั่วไปของโครงสร้างในโครงการที่นำมาประเมินความเสี่ยง



รูปที่ 5.7 ส่วนการรายงานผลลัพธ์การคำนวณความเสี่ยงระดับองค์ประกอบ

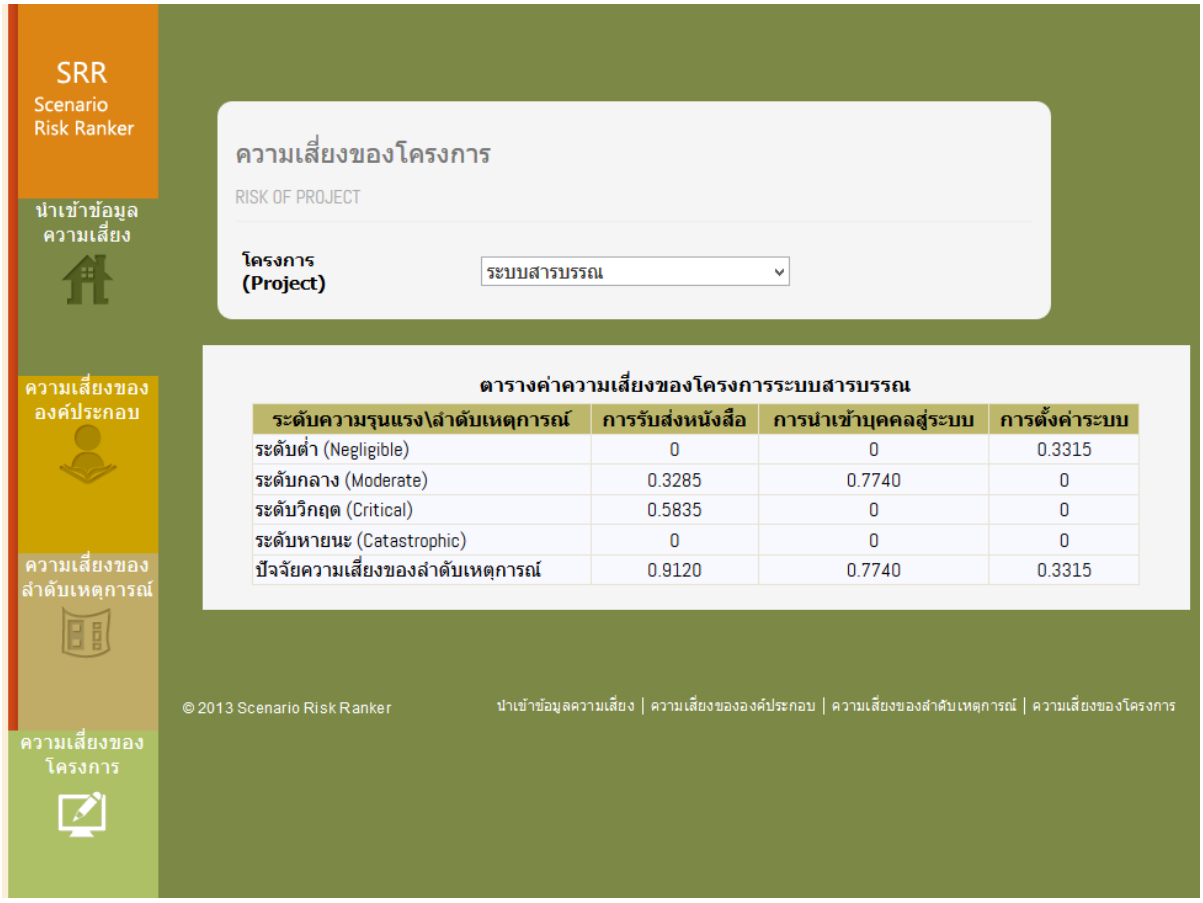
การรายงานผลค่าความเสี่ยงระดับองค์ประกอบในรูปที่ 5.6 ได้ออกแบบให้สามารถแสดงผลลัพธ์และที่มาของผลลัพธ์ซึ่งเป็นผลลัพธ์ย่อยที่เกิดจากขั้นตอนวิธีการประเมินความเสี่ยงระดับองค์ประกอบที่นำเสนอไว้ในบทที่สาม โดยแบ่งการแสดงผลออกเป็นสามขั้นตอนซึ่งประกอบด้วย ขั้นตอนที่หนึ่ง เป็นการแสดงจำนวนกรณีทดสอบทั้งหมดที่ได้จากการประยุกต์ใช้หลักการณ์คลอบคลุมประพจน์ ขั้นตอนที่สองเป็นการแสดงเส้นทางการทำงานของ

ซอฟต์แวร์ที่เป็นไปได้ซึ่งรวมไปถึงความน่าจะเป็นของเส้นทางที่พึงปรารถนาและไม่ปราถนา และขั้นตอนสุดท้ายเป็นการแสดงผลลัพธ์ของความน่าจะเป็นขององค์ประกอบ (PUOC) ความน่าจะเป็นขององค์ประกอบที่ปรับเข้ากับลำดับเหตุการณ์ (PUOCS) และค่าปัจจัยความเสี่ยง (RF) ที่เกิดขึ้นกับองค์ประกอบแต่ละองค์ประกอบ



รูปที่ 5.8 ส่วนการรายงานความเสี่ยงของลำดับเหตุการณ์

การรายงานความเสี่ยงของลำดับเหตุการณ์ที่แสดงได้ดังรูปที่ 5.7 เครื่องมือการประเมินความเสี่ยงถูกออกแบบให้แสดงผลลัพธ์สุดท้ายในขั้นตอนการประเมินความเสี่ยงของลำดับเหตุการณ์ ซึ่งหมายถึงการรวมค่าความเสี่ยงของสถานะความผิดพลาดจากเมทริกซ์ความน่าจะเป็นของการเปลี่ยนแปลงจากสถานะปกติเข้าสู่สถานะความผิดพลาด (A^x)



รูปที่ 5.9 ส่วนการรายงานความเสี่ยงของโครงการ

ส่วนการรายงานความเสี่ยงของโครงการในรูปที่ 5.8 เป็นการแสดงรายงานขั้นสุดท้ายที่ได้จากการคำนวณความเสี่ยงของโครงการที่นำมาประเมิน ซึ่งเป็นการค่าปัจจัยความเสี่ยงในแต่ละระดับของลำดับเหตุการณ์ภายใต้โครงการที่นำมาประเมิน

5.3. การพัฒนาเครื่องมือสำหรับการประเมินความเสี่ยง

เครื่องมือสำหรับการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีนได้ถูกพัฒนาในรูปแบบของเว็บเบสแอปพลิเคชัน ซึ่งทำให้ผู้ใช้งานสามารถเข้าถึงได้สะดวกและไม่จำเป็นต้องติดตั้งโปรแกรมใดๆเพื่อใช้งานเครื่องมือการประเมินความเสี่ยง สำหรับอุปกรณ์และเครื่องมือที่นำมาพัฒนาเครื่องมือการประเมินความเสี่ยงมีรายละเอียดดังนี้

1) รายละเอียดด้านฮาร์ดแวร์

- หน่วยประมวลผลกลางอินเทล คอร์ ไอห้า (Intel Core i5) รุ่นที่-3 รหัส 3210M ความเร็วสัญญาณนาฬิกา 2.50 กิกะเฮิรตซ์ (GHz)
- หน่วยความจำหลักขนาด 8 กิกะไบต์
- จานบันทึกแบบแข็งขนาด 300 กิกะไบต์

2) รายละเอียดด้านซอฟต์แวร์

- ระบบปฏิบัติการไมโครซอฟท์วินโดวส์เวอร์ชัน 8 สำหรับอุปกรณ์ 64 บิต
- ตัวแปรภาษาพีเอชพี (PHP) เวอร์ชัน 5.3.26
- โปรแกรมเว็บเซิฟเวอร์อาปาเช่ (Apache) เวอร์ชัน 2.2
- โปรแกรมบริหารจัดการฐานข้อมูลมายเอสคิวแอล (MySQL) เวอร์ชัน 5.5.32
- โปรแกรมวิชวลพาราดามสำหรับยูเอ็มแอล เวอร์ชัน 10.0

5.4. การทดสอบเครื่องมือสำหรับการประเมินความเสี่ยง

การทดสอบเครื่องมือที่พัฒนาแบ่งออกเป็นสองส่วนด้วยกันคือการตรวจสอบและการทวนสอบ (Validation and Verification) โดยการตรวจสอบผู้วัยเลือกใช้วิธีการสร้างรายการตรวจสอบแบบฮิวริสติก (Heuristic Checklist) ส่วนการทวนสอบนั้นสามารถแบ่งกระบวนการทวนสอบออกเป็นสามวิธีได้แก่ การทดสอบระดับหน่วย (Unit testing) การทดสอบแบบบูรณาการณ์ (Integration testing) และการทดสอบระบบ (System testing) มีรายละเอียดดังต่อไปนี้

5.4.1. การตรวจสอบ (Validation)

ขั้นตอนนี้เป็นการตรวจสอบความสามารถที่ปรากฏในเครื่องมือกับรายการความต้องการที่ได้รวบรวมไว้ในขั้นต้น แสดงได้ดังตารางที่ 5.5

ตารางที่ 5.5 ผลลัพธ์การตรวจสอบความสมบูรณ์ระหว่างรายการความต้องเชิงหน้าที่กับเครื่องมือที่พัฒนา

รายการความต้องการ	ผลการตรวจสอบ [ปรากฏ/ไม่ปรากฏ]	แหล่งอ้างอิง
FR1-2, FR4	ปรากฏ	เมนูนำเข้าข้อมูลความเสี่ยง
FR3	ปรากฏ	ขั้นตอนที่เกิดขึ้นหลังการนำเข้าข้อมูลความเสี่ยง
FR5	ปรากฏ	ขั้นตอนที่สองของการนำเข้าข้อมูลความเสี่ยง
FR6-7	ปรากฏ	ขั้นตอนที่เกิดขึ้นหลังนำเข้าข้อมูลความเสี่ยงเรียบร้อยแล้ว
FR8	ปรากฏ	เมนูความเสี่ยงขององค์ประกอบ ลำดับเหตุการณ์ และโครงการ

5.4.2. การทวนสอบ (Verification)

ขั้นตอนนี้เป็นการตรวจสอบสิ่งที่พัฒนาขึ้นตามรายการความต้องการว่าสิ่งต่างๆเหล่านั้นมีการทำงานที่ถูกต้องตามหลักเกณฑ์ที่ได้กำหนดขึ้นหรือไม่ ซึ่งประกอบไปด้วยประเภทการทดสอบสามประเภท ดังนี้

1) การทดสอบระดับหน่วย

ในส่วนนี้เป็นการแบ่งส่วนการทำงานที่สำคัญและมีความซับซ้อน เพื่อนำมาทดสอบผลลัพธ์ก่อนการนำเอาส่วนการทำงานต่างๆมาทำงานร่วมกันประกอบด้วย การสร้างต้นไม้แจกแจงแบบทวิภาค การสร้างกรณีทดสอบด้วยหลักการณ์ครอบคลุมประพจน์ การหาเส้นทางการทดสอบแบบไพร์ม การคำนวณหาอินเวิร์สการคูณของเมทริกซ์ด้วยวิธีการดำเนินการแบบแถว (Row operation)

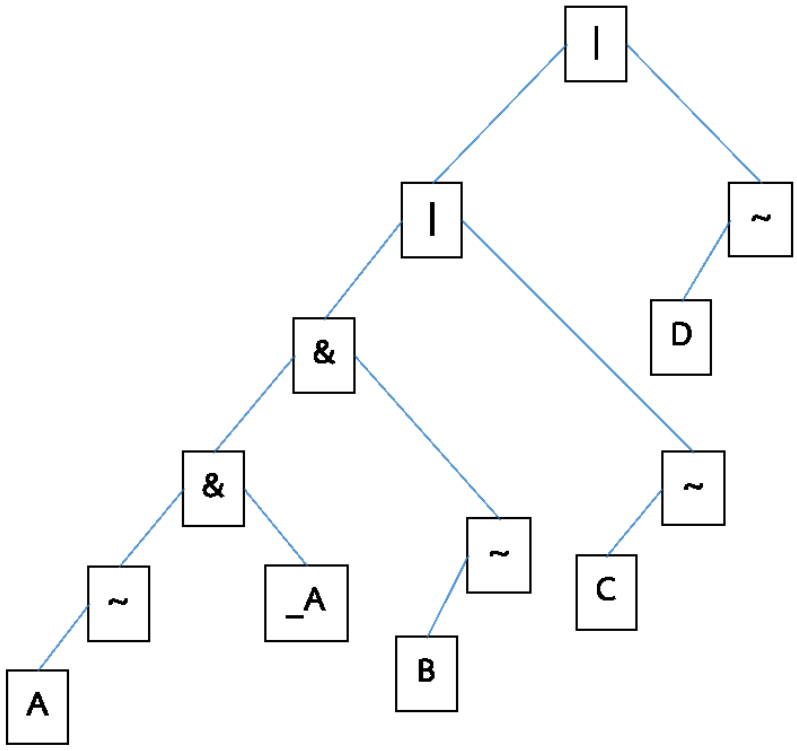
(1) การสร้างต้นไม้แจงแจงแบบทวิภาค

การทดสอบในส่วนนี้ผู้วิจัยใช้วิธีการจำลองข้อมูลนำเข้าหลายรูปแบบ สำหรับนำเข้าเพื่อทดสอบความถูกต้องของการพัฒนาโปรแกรม แสดงตัวอย่างการทดสอบได้ดังตารางที่ 5.1

ตารางที่ 5.6 ผลลัพธ์การทดสอบส่วนย่อยการสร้างต้นไม่แจกแจงทวิภาค

ลำดับ	ข้อมูลทดสอบ	ผลลัพธ์ที่คาดหวัง	ผลลัพธ์ที่เกิดขึ้นจริง	ผลการทดสอบ
1	when(A)	~A & _A	~A & _A	ผ่าน
2	when(A)[B & C & ~D]	~A & _A & B & C & ~D	~A & _A & B & C & ~D	ผ่าน
3	when(A)[B | C | ~D]	~A & _A & B | C | ~D	~A & _A & B | C | ~D	ผ่าน
4	when(A)[~B | ~C | ~D]	~A & _A & ~B | ~C | ~D	~A & _A & ~B | ~C | ~D	ผ่าน
5	when(A)[B & C & D & E & F]	~A & _A & B & C & D & E & F	~A & _A & B & C & D & E & F	ผ่าน

กรณีทดสอบในลำดับที่สี่สามารถนำนิพจน์ที่ได้มาสร้างเป็นต้นไม้แจกแจงแบบทวิภาคได้ดังรูปที่ 5.10



รูปที่ 5.10 ต้นไม้แจกแจงแบบทวิภาคของนิพจน์ ~A & _A & ~B | ~C | ~D

(2) การสร้างกรณีทดสอบด้วยหลักการณ์ครอบคลุมประพจน์

การทดสอบในส่วนนี้มีวัตถุประสงค์ในการตรวจสอบจำนวนกรณีทดสอบที่ถูกสร้างขึ้นด้วยหลักการณ์ครอบคลุมประพจน์ ซึ่งกระบวนการทดสอบนี้ใช้วิธีการจำลองข้อมูลนำเข้าเพื่อตรวจสอบผลลัพธ์ที่ได้จากโปรแกรมที่พัฒนาเทียบกับผลลัพธ์ที่คาดหวัง แสดงได้ดังตารางที่ 5.7

ตารางที่ 5.7 ผลลัพธ์การทดสอบระดับหน่วยการสร้างกรณีทดสอบด้วยหลักการณ์ครอบคลุมประพจน์

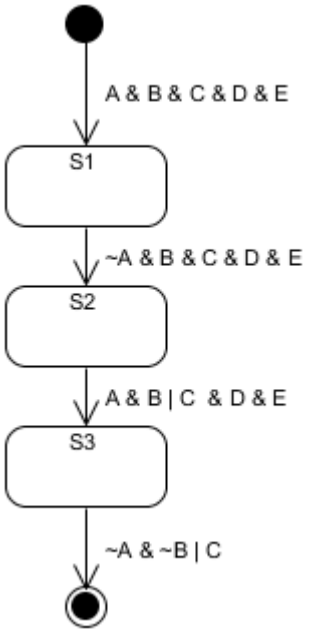
ลำดับ	ข้อมูลทดสอบ	กรณีทดสอบที่คาดหวัง	กรณีทดสอบที่เกิดขึ้นจริง	ผลการทดสอบ
1	~A & _A	3	3	ผ่าน
2	~A & _A & B & C & ~D	6	6	ผ่าน
3	~A & _A & B | C | ~D	7	7	ผ่าน
4	~A & _A & ~B | ~C | ~D	7	7	ผ่าน
5	~A & _A & B & C & D & E & F	8	8	ผ่าน

(3) การหาเส้นทางการทดสอบแบบไพร์ม

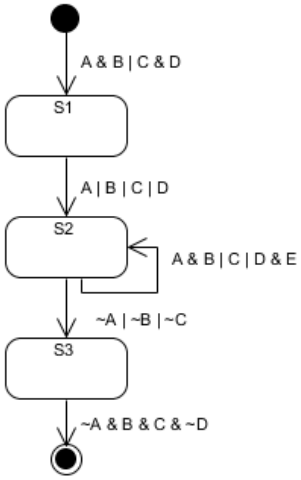
การทดสอบในส่วนนี้เป็นการหาเส้นทางการทดสอบแบบไพร์ม ซึ่งในกระบวนการทดสอบได้จำลองแผนภาพสเตทแมชชีนดังรูปที่ 5.11 – 5.16 เพื่อใช้ในขั้นตอนการหาเส้นทางการทดสอบ ดังสรุปได้ในตารางที่ 5.8

ตารางที่ 5.8 ผลลัพธ์การทดสอบระดับหน่วยของการหาเส้นทางการทดสอบแบบไพร์ม

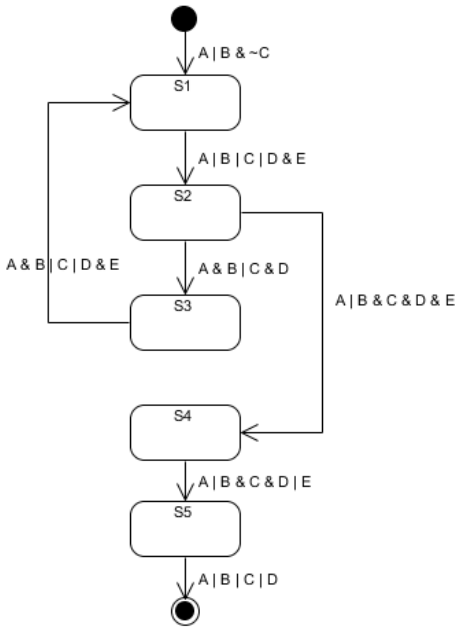
ลำดับ	ข้อมูลทดสอบ	กรณีทดสอบที่คาดหวัง	กรณีทดสอบที่เกิดขึ้นจริง	ผลการทดสอบ
1	แผนภาพสเตทแมชชีนตัวอย่างที่หนึ่ง	1. start-S1-S2-S3-S4	1. start-S1-S2-S3-S4	ผ่าน
2	แผนภาพสเตทแมชชีนตัวอย่างที่สอง	1. start-S1-S2-S3-S4 2. start-S1-S2-S2-S3-S4	1. start-S1-S2-S3-S4 2. start-S1-S2-S2-S3-S4	ผ่าน
3	แผนภาพสเตทแมชชีนตัวอย่างที่สาม	1. start-S1-S2-S3-S1-S2-S3-S1-S2-S4-S5-S6 2. start-S1-S2-S4-S5-S6 3. start-S1-S2-S3-S1-S2-S4-S5-S6	1. start-S1-S2-S3-S1-S2-S3-S1-S2-S4-S5-S6 2. start-S1-S2-S4-S5-S6 3. start-S1-S2-S3-S1-S2-S4-S5-S6	ผ่าน
4	แผนภาพสเตทแมชชีนตัวอย่างที่สี่	1. start-S1-S2-S3-S4-S5-S6 2. start-S1-S2-S3-S4-S5-S1-S2-S3-S4-S5-S6	1. start-S1-S2-S3-S4-S5-S6 2. start-S1-S2-S3-S4-S5-S1-S2-S3-S4-S5-S6	ผ่าน
5	แผนภาพสเตทแมชชีนตัวอย่างที่ห้า	1. start-S1-S2-S4-S5-S6 2. start-S1-S2-S3-S6	1. start-S1-S2-S4-S5-S6 2. start-S1-S2-S3-S6	ผ่าน
6	แผนภาพสเตทแมชชีนตัวอย่างที่หก	1. start-S1-S2-S3-S1-S2-S3-S1-S5-S6 2. start-S1-S5-S6 3. start-S1-S2-S3-S1-S5-S6 4. start-S4-S6 5. start-S4-S4-S6	1. start-S1-S2-S3-S1-S2-S3-S1-S5-S6 2. start-S1-S5-S6 3. start-S1-S2-S3-S1-S5-S6 4. start-S4-S6 5. start-S4-S4-S6	ผ่าน



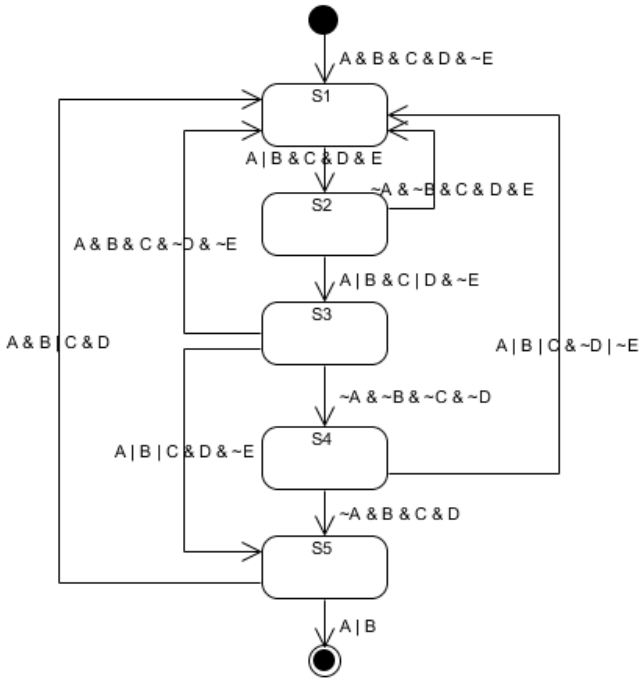
รูปที่ 5.1 แผนภาพสเตทแมชชีนตัวอย่างที่หนึ่ง



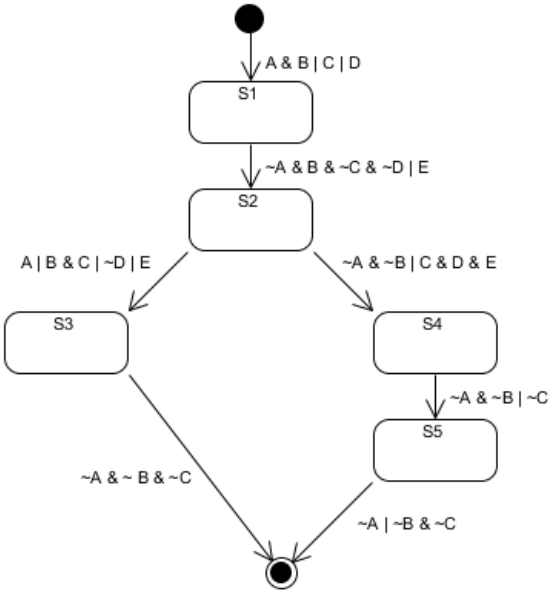
รูปที่ 5.2 แผนภาพสเตทแมชชีนตัวอย่างที่สอง



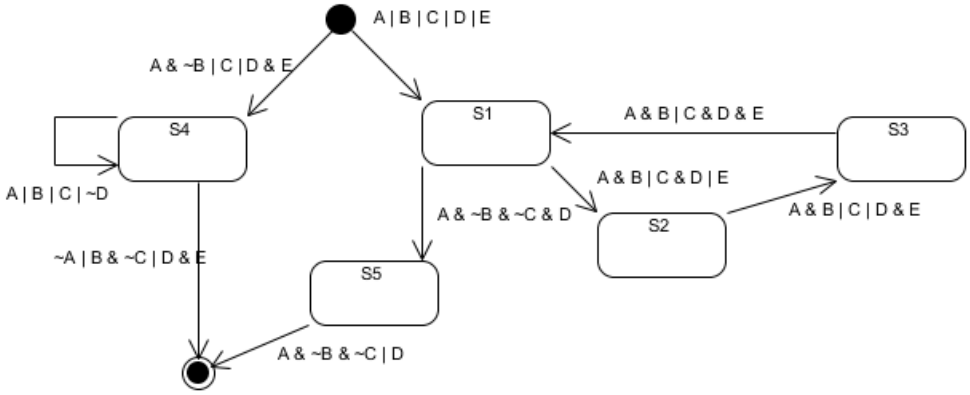
รูปที่ 5.3 แผนภาพสเตทแมชชีนตัวอย่างที่สาม



รูปที่ 5.4 แผนภาพสเตทแมชชีนตัวอย่างที่สี่



รูปที่ 5.5 แผนภาพสเตทแมชชีนตัวอย่างที่ห้า



รูปที่ 5.6 แผนภาพสเตทแมชชีนตัวอย่างที่หก

(4) การคำนวณหาอินเวิร์สการคูณของเมทริกซ์ด้วยวิธีการดำเนินการแบบแถว

การทดสอบระดับหน่วยของการคำนวณหาอินเวิร์สการคูณของเมทริกซ์มีด้วยกันหลายวิธี แต่วิธีที่สามารถรองรับข้อมูลเข้าในลักษณะของเมทริกซ์ขนาดต่างๆ โดยไม่มีข้อจำกัดในเรื่องขนาดของเมทริกซ์นั้น ได้แก่ การหาอินเวิร์สของเมทริกซ์ด้วยวิธีการดำเนินการแบบแถว ซึ่งกระบวนการทดสอบระดับหน่วยนี้ ผู้วิจัยใช้วิธีการสุ่มตัวเลขเพื่อทดสอบหาอินเวิร์สการคูณของเมทริกซ์นั้นๆ แสดงรายละเอียดแสดงได้ดังตารางที่ 5.9

ตารางที่ 5.9 ผลลัพธ์การทดสอบระดับหน่วยการคำนวณหาอินเวิร์สการคูณของเมทริกซ์

ลำดับ	ข้อมูลทดสอบ	กรณีทดสอบที่คาดหวัง	กรณีทดสอบที่เกิดขึ้นจริง	ผลการทดสอบ
1	$\begin{bmatrix}4 & 3\\3 & 2\end{bmatrix}$	$\begin{bmatrix}-2 & 3\\3 & -4\end{bmatrix}$	$\begin{bmatrix}-2 & 3\\3 & -4\end{bmatrix}$	ผ่าน
2	$\begin{bmatrix}1 & 2 & 3\\0 & 1 & 4\\5 & 6 & 0\end{bmatrix}$	$\begin{bmatrix}-24 & 18 & 5\\20 & -15 & -4\\-5 & 4 & 1\end{bmatrix}$	$\begin{bmatrix}-24 & 18 & 5\\20 & -15 & -4\\-5 & 4 & 1\end{bmatrix}$	ผ่าน
3	$\begin{bmatrix}1 & 0 & 0 & 1\\0 & 2 & 1 & 2\\2 & 1 & 0 & 1\\2 & 0 & 1 & 4\end{bmatrix}$	$\begin{bmatrix}-2 & -0.5 & 1 & 0.5\\1 & 0.5 & 0 & -0.5\\-8 & -1 & 2 & 2\\3 & 0.5 & -1 & -0.5\end{bmatrix}$	$\begin{bmatrix}-2 & -0.5 & 1 & 0.5\\1 & 0.5 & 0 & -0.5\\-8 & -1 & 2 & 2\\3 & 0.5 & -1 & -0.5\end{bmatrix}$	ผ่าน
4	$\begin{bmatrix}1 & 8 & -9 & 7 & 5\\0 & 1 & 0 & 4 & 4\\0 & 0 & 1 & 2 & 5\\0 & 0 & 0 & 1 & -5\\0 & 0 & 0 & 0 & 1\end{bmatrix}$	$\begin{bmatrix}1 & -8 & 9 & 7 & 175\\0 & 1 & 0 & -4 & -24\\0 & 0 & 1 & -2 & -15\\0 & 0 & 0 & 1 & 5\\0 & 0 & 0 & 0 & 1\end{bmatrix}$	$\begin{bmatrix}1 & -8 & 9 & 7 & 175\\0 & 1 & 0 & -4 & -24\\0 & 0 & 1 & -2 & -15\\0 & 0 & 0 & 1 & 5\\0 & 0 & 0 & 0 & 1\end{bmatrix}$	ผ่าน

2) การทดสอบแบบบูรณาการณ์ (Integration testing)

ขั้นตอนนี้เป็นการนำเสนอตัวอย่างของการทดสอบแบบบูรณาการณ์ โดยเป็นการนำเอางานระดับหน่วยย่อยที่มีการทำงานต่อเนื่องกันมาทดสอบความถูกต้อง จึงยกตัวอย่างการทดสอบแบบบูรณาการณ์ของการสร้างต้นไม้แจงแจงทวิภาคและการสร้างกรณีทดสอบแบบครอบคลุมประพจน์ เป็นตัวอย่างสำหรับขั้นตอนการทดสอบแบบบูรณาการณ์นี้

ในขั้นต้นได้กล่าวไว้แล้วว่าการสร้างต้นไม้แจกแจงแบบทวิภาคเป็นการนำเอานิพจน์อันเป็นเงื่อนไขบนการเปลี่ยนแปลงระหว่างสถานะ และการสร้างกรณีทดสอบนั้นเป็นการสร้างกรณีทดสอบจากต้นไม้แจกแจงทวิภาค จากนั้นทำการท่องต้นไม้ (Traverse tree) แบบล่างขึ้นบน (Depth-First search) จึงจะได้มาซึ่งจำนวณของกรณีทดสอบทั้งหมดและกรณีทดสอบที่ให้ค่าเป็นจริง ดังนั้นจะเห็นได้ว่าการทำงานของส่วนงานระดับย่อยทั้งสองมีความเกี่ยงเนื่องกันอย่างชัดเจน และสามารถแสดงผลลัพธ์ของการทดสอบได้ดังตารางที่ 5.10

ตารางที่ 5.10 ตัวอย่างผลการทดสอบแบบบูรณาการณ์

ลำดับ	ข้อมูลทดสอบ	ค่าคาดหวังอัตราส่วนระหว่างกรณีทดสอบที่เป็นจริงต่อกรณีทดสอบทั้งหมด	อัตราส่วนระหว่างกรณีทดสอบที่เป็นจริงต่อกรณีทดสอบทั้งหมดที่เกิดขึ้น	ผลการทดสอบ
1	when(A)	0.3333	0.3333	ผ่าน
2	when(A)[B & C & ~D]	0.1667	0.1667	ผ่าน
3	when(A)[B | C | ~D]	0.4285	0.4285	ผ่าน
4	when(A)[~B | ~C | ~D]	0.4285	0.4285	ผ่าน
5	when(A)[B & C & D & E & F]	0.125	0.125	ผ่าน

5.4.3. การทดสอบระบบ

ขั้นตอนนี้เป็นขั้นตอนสุดท้ายในกระบวนการทวนสอบซอฟต์แวร์และเป็นขั้นตอนการทดสอบครั้งสุดท้ายก่อนการนำเครื่องมือไปใช้งาน สามารถทำได้โดยการทดสอบซอฟต์แวร์ที่พัฒนาร่วมกับซอฟต์แวร์อื่นๆที่มีหน้าที่ในการให้บริการ แสดงการทดสอบได้ดังตารางที่ 5.11

ตารางที่ 5.11 ตัวอย่างผลลัพธ์การทดสอบระบบ

ลำดับ	รายการทดสอบ	ค่าคาดหวัง	ผลลัพธ์ที่เกิดขึ้น	ผลการทดสอบ
1	ติดตั้งเครื่องมือโดยปราศจากการทำงานของโปรแกรมบริหารจัดการฐานข้อมูล	ระบบแจ้งข้อผิดพลาดด้านการเชื่อมต่อกับฐานข้อมูล	ระบบแจ้งข้อผิดพลาดด้านการเชื่อมต่อกับฐานข้อมูล	ผ่าน
2	เรียกใช้เครื่องมือผ่านเว็บเบราซ์เซอร์ไฟร์ฟอกซ์ (FireFox)	สามารถเรียกใช้งานได้	สามารถเรียกใช้งานได้	ผ่าน

5.5. สรุปผล

การรวบรวมความต้องการ การวิเคราะห์ ออกแบบ การพัฒนา และการทดสอบที่ได้กล่าวไว้ในข้างต้น เป็นเครื่องมือที่สามารถยืนยันได้ว่า เครื่องมือสำหรับการประเมินความเสี่ยงให้ผลลัพธ์ที่มีฟังก์ชันการทำงานที่ครบถ้วนตามข้อกำหนดความต้องการและเป็นไปตามขั้นตอนวิธีที่ได้นำเสนอไว้ในบทที่สาม และขั้นตอนการทวนสอบเป็นการช่วยแสดงความแม่นยำของผลลัพธ์ที่ได้จากการใช้งานเครื่องมือการประเมินความเสี่ยงนี้ ซึ่งจะช่วยลดข้อผิดพลาดจากการประเมินความเสี่ยงโดยปราศจากการใช้เครื่องมืออีกด้วย

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

ในบทนี้เป็นการเสนอประเด็นสำคัญและบทสรุปของงานวิจัยฉบับนี้ โดยแบ่งออกเป็น สรุปผลการวิจัย ข้อจำกัดของงานวิจัย และแนวทางการวิจัยในอนาคต แสดงได้ดังต่อไปนี้

6.1. สรุปผลการวิจัย

งานวิจัยฉบับนี้นำเสนอขั้นตอนวิธีการประเมินความเสี่ยงที่อาจเกิดขึ้นกับฐานข้อมูลเชิงสัมพันธ์ในด้านของข้อมูลที่ไม่เป็นไปตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน ที่มีสาเหตุความผิดพลาดจากการทำงานที่ล้มเหลวของซอฟต์แวร์ และข้อผิดพลาดที่อยู่นอกเหนือการควบคุมของโปรแกรมบริหารจัดการฐานข้อมูล ซึ่งเป็นปัจจัยสำคัญที่ทำให้เกิดการคิดค้นขั้นตอนวิธีการประเมินความเสี่ยงนี้ขึ้น กล่าวคือ หากการพัฒนาซอฟต์แวร์ที่พึ่งพิงความสามารถในการควบคุมข้อผิดพลาดตามเงื่อนไขเชิงเค้าร่างหรือการทดสอบซอฟต์แวร์หลังขั้นตอนการพัฒนาซอฟต์แวร์เพียงอย่างเดียวย่อมส่งผลให้เกิดข้อผิดพลาดและความสูญเสียขึ้นไม่มากก็น้อย ซึ่งขึ้นอยู่กับประสบการณ์ของผู้พัฒนาซอฟต์แวร์ แต่หากมีขั้นตอนวิธีที่จะทำให้ผู้พัฒนาซอฟต์แวร์ล่วงรู้ความเสี่ยงของการเกิดข้อผิดพลาดขึ้นกับข้อมูลในฐานข้อมูลได้ในระหว่างขั้นตอนการออกแบบซอฟต์แวร์แล้ว ก็จะทำให้ผู้พัฒนาซอฟต์แวร์สามารถบริหารทรัพยากรที่ใช้ในการพัฒนาได้แม่นยำ และเป็นผลดีต่อโครงการมากยิ่งขึ้น งานวิจัยฉบับนี้ยังได้นำเสนอเครื่องมือที่ช่วยให้ผู้พัฒนาซอฟต์แวร์ล่วงรู้ถึงความเสี่ยงที่แฝงอยู่ในโครงการเป็นจำนวนสองชิ้นด้วยกัน คือ

1) ขั้นตอนวิธีการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน

ขั้นตอนวิธีที่นำเสนอแบ่งการพิจารณาออกเป็นสองส่วนคือ ลำดับเหตุการณ์และองค์ประกอบของลำดับเหตุการณ์ ที่พิจารณาทั้งในส่วนของความเสี่ยงที่เกิดขึ้นกับองค์ประกอบ และการเชื่อมโยงระหว่างองค์ประกอบ โดยนำเอาหลักการณ์ครอบคลุมประพจน์และวิธีการคำนวณหาเส้นทางการทดสอบแบบไพร์มเพื่อทำให้ทราบถึงความน่าจะเป็นขององค์ประกอบ โดยมีแผนภาพสเตทแมชชีนเป็นข้อมูลเข้าสำหรับการคำนวณ สำหรับการคำนวณความน่าจะเป็นในการทำงานที่ล้มเหลวของการเชื่อมโยงระหว่างองค์ประกอบนั้น ได้นำเอาการเรียกใช้ระหว่างองค์ประกอบในแผนภาพลำดับเป็นข้อมูลเข้าในการพิจารณา และที่ขาดไม่ได้สำหรับการคำนวณความเสี่ยงก็คือ การคำนวณหาระดับผลกระทบเมื่อเกิดข้อผิดพลาดขึ้น ซึ่งทำได้โดยการคำนวณระดับความรุนแรงจากเอกสารเค้าร่างฐานข้อมูลที่เกี่ยวข้องกับองค์ประกอบและการเชื่อมโยงระหว่างองค์ประกอบ ซึ่งผลลัพธ์จากการคำนวณในระดับองค์ประกอบสามารถนำมาประยุกต์เข้ากับแบบจำลองของมาร์คอฟที่มีสมมุติฐานเกี่ยวกับผลลัพธ์ของเหตุการณ์หลังเป็นอิสระจากเหตุการณ์ก่อนหน้า ซึ่งจะนำไปสู่การสร้างแบบจำลองความเสี่ยงของซอฟต์แวร์ในแต่ละระดับความผิดพลาดที่ได้กำหนดขึ้น

2) เครื่องมือการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่างด้วยแผนภาพสเตทแมชชีน

หากพิจารณาขั้นตอนวิธีทั้งหมดที่นำเสนอเพื่อให้ได้มาซึ่งความเสี่ยงที่แฝงอยู่ในโครงการพบว่ามีความซับซ้อนและประกอบไปด้วยขั้นตอนหลายขั้นตอน ดังนั้นผู้วิจัยจึงพัฒนาเครื่องมือการประเมินความเสี่ยงนี้ในรูปแบบของเว็บเบสแอปพลิเคชันให้กับผู้ที่ต้องการประเมินความเสี่ยง สามารถทราบถึงความเสี่ยงของโครงการโดยปราศจากการนำขั้นตอนวิธีไปใช้อย่างไม่ถูกต้องตามเงื่อนไข โดยเครื่องมือที่พัฒนาขึ้นนี้เป็นไปตามรายการความต้องการเชิงหน้าที่และคุณภาพตามที่กำหนดไว้ในขั้นตอนวิธีการประเมินความเสี่ยง อีกทั้งยังผ่านกระบวนการทดสอบซอฟต์แวร์ในระดับหน่วย (Unit test) ระดับบูรณาการณ์ (Integration test) และการทดสอบระบบ (System test) ซึ่งเป็นการเพิ่มระดับความเชื่อมั่นให้กับผู้ประเมินในการนำเอาเครื่องมือที่พัฒนานี้ไปใช้เพื่อการประเมินความเสี่ยงของฐานข้อมูลเชิงสัมพันธ์ตามเงื่อนไขเชิงเค้าร่าง

6.2. ข้อจำกัดของงานวิจัย

ถึงแม้ว่าผลลัพธ์ของการประเมินจะทำให้ทราบว่าขั้นตอนวิธีในการประเมินความเสี่ยงฯ สามารถจัดอันอับความเสี่ยงที่เกิดขึ้นกับข้อมูลในฐานข้อมูลได้นั้น แต่อย่างไรก็ตามผู้วิจัยเล็งเห็นถึงข้อจำกัดของขั้นตอนวิธีที่นำเสนอไว้ ดังนี้

1) ขั้นตอนวิธีการประเมินความเสี่ยงที่นำเสนอสามารถใช้ได้กับซอฟต์แวร์ที่มีการใช้งานฐานข้อมูลเพียงฐานข้อมูลเดียว เนื่องจากเงื่อนไขเชิงเค้าร่างทั้งสี่เงื่อนไขที่ได้กำหนดไว้ในขอบเขตของการวิจัยเป็นเงื่อนไขของตารางที่อยู่ในฐานข้อมูลเดียวกันเท่านั้น อีกทั้งขั้นตอนวิธีในการประเมินไม่มีส่วนใดเกี่ยวข้องกับการระบุความเสี่ยงของการใช้ฐานข้อมูลมากกว่าหนึ่งฐานข้อมูล
2) ขั้นตอนวิธีที่ใช้ในการคำนวณหาระดับความรุนแรงยังใช้วิธีที่ทำให้ผลลัพธ์ขึ้นอยู่กับผู้ประเมิน (Subjective)
3) เครื่องมือสำหรับประเมินความเสี่ยงถูกจำกัดให้ใช้เอกสารเค้าร่างฐานข้อมูลสำหรับโปรแกรมบริหารจัดการฐานข้อมูลมายเอสคิวแอล (MySQL) เท่านั้น

6.3. แนวทางการวิจัยในอนาคต

จากผลลัพธ์ที่ได้จากการนำเอาขั้นตอนการประเมินความเสี่ยงมาใช้เพื่อการประเมินความเสี่ยงของฐานข้อมูลกับซอฟต์แวร์ระบบสารบรรณอิเล็กทรอนิกส์ที่กล่าวไว้ในบทที่สี่แล้วนั้น งานวิจัยฉบับนี้สามารถนำไปศึกษาเพิ่มเติมเพื่อปรับปรุงขั้นตอนวิธีที่ปราศจากข้อจำกัดในข้างต้น และเพิ่มเติมขีดความสามารถของขั้นตอนวิธีได้ดังนี้

1) การพิจารณาระดับความรุนแรงใช้วิธีการประเมินในระดับตารางของฐานข้อมูล ซึ่งสามารถปรับปรุงโดยให้ค่าน้ำหนักกับคุณสมบัติประจำตาราง ซึ่งจะทำให้ผลลัพธ์ที่ได้มีความใกล้เคียงความเป็นจริงมากกว่าการประเมินในระดับตาราง เพราะการคำนวณระดับความรุนแรงในลักษณะนี้จะเป็นผลลัพธ์ที่ปราศจากค่ารบกวน (Noise) อาทิเช่น ตารางที่ใช้จัดเก็บผลการเรียนของนักศึกษาประกอบด้วยคุณลักษณะประจำตารางเป็นจำนวณ 30 คุณลักษณะ แต่มีเพียงสองคุณลักษณะเท่านั้นที่มีผลต่อระดับความรุนแรงที่แท้จริง เป็นต้น

2) สามารถปรับปรุงขั้นตอนการประเมินความเสี่ยงในระดับองค์ประกอบให้สามารถระบุถึงค่าความเสี่ยงของประเภทเงื่อนไขเชิงเค้าร่างแต่ละประเภทได้ เนื่องจากขั้นตอนวิธีที่ได้นำเสนอไว้สามารถประเมินความเสี่ยงได้ละเอียดที่สุดในระดับการเปลี่ยนแปลงขององค์ประกอบ ซึ่งทำให้ผู้ประเมินรับรู้เพียงค่าของความเสี่ยงที่เกิดขึ้น แต่จะไม่ทราบถึงความเสี่ยงประเภทใดที่ทำให้เกิดผลกระทบได้มากที่สุด

รายการอ้างอิง

[1] B.W. Boehm, Software risk management: principle and practices, IEEE Software, vol. 08, no.1, pp. 32-41, Jan 1991.

[2] NASA Technical Std. NASA-STD-8719.13A, Software Safety, March 2004,pp. 26-27.

[3] Ramez Elmasri, Shamkant B. Navathe, Fundamentals of database systems sixth edition, pp 67-74.

[4] James Rumbaugh, Ivar Jacobson, Grady Booch The Unified Modeling Language Reference Manual Second Edition USA, Addison-Wesley, 2005, pp 79-89.

[5] Katerina Goseva-Popstojanova, Ahmed Hassan, Ajith Guedem Architectural-Level Risk Analysis Using UML, IEEE transactions on software, vol. 29, no. 10, Oct 2003

[6] Akekachai Tangsuksant and Nakornthip Prompoon, Risk Assessment Framework based on Goal-oriented Requirements Engineering and Object Behavioral Model

[7] Jeff Offutt, Shaoying Liu, Aynur Abdurazik and Paul Ammann. Generating test data from state-based specifications, Springer-Verlag Berlin Heidelberg, February 2003

[8] Karunee Bowornprasirtkul and Nakornthip Prompoon, Test cases generation from a state chart diagram, 2004

[9] Hongyu Zhang, Hee Beng Kuan Tanc, Lu Zhangd, Xi Lina,b, Xiaoyin Wangd, Chun Zhangd and Hong Meid. Checking enforcement of integrity constraints in database applications based on code patterns, Journal [Systems and Software], December 2011 vol. 84(12), pp 2253-2264

[10]Paul Ammann, Jeff Offutt, Introduction to Software Testing, pp 33-42

[11]Björn Franke, Lexical Analysis – Constructing a Scanner from Regular Expression University of Edinburgh School of Informatics, 2013

[12]Björn Franke, Top-down parsing; recursive descent and LL(1), University of Edinburgh School of Informatics, 2013

[13]Alan Dennis, Barbara Haley Wixom and David Tegarden, Systems Analysis and Design with UML Version 2.0 second edition, pp 343-345

ภาคผนวก

ภาคผนวก ก

ตัวอย่างการสร้างกรณีทดสอบจากแผนภาพสเตทแมชชีน

ตารางที่ ก.1 ตัวอย่างการเปลี่ยนสถานะของแผนภาพสเตทแมชชีนองค์ประกอบการลงทะเบียนหนังสือ

ลำดับ	สถานะปัจจุบัน	เงื่อนไขเหตุการณ์																
		docEntityCheck	docDomainCheck	checkDocCode	checkDocType	checkDocID	recEntityCheck	recDomainCheck	recCheckID	checkRefInfo	save	getDocID	getDocCode	getCreateDate	getSubject	docIDisINT	docCodeIsSTRING	createDateisDate	subjectisSTRING
1	นำเข้าข้อมูลหนังสือ	@T	-	-	-	-	-	-	-	-	-	t	t	t	t	-	-	-	-
2	นำเข้าข้อมูลหนังสือ	@T	-	-	-	-	-	-	-	-	-	f	f	f	f	-	-	-	-
3	ข้อมูลหนังสือครบถ้วน	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	t	t	t	t

ตารางที่ ก.2 ตัวอย่างการเปลี่ยนสถานะของแผนภาพสเตทแมชชีนองค์ประกอบการลงทะเบียนหนังสือ

ลำดับ	สถานะ ปัจจุบัน	เงื่อนไขเหตุการณ์																
		toisSTRING	shortDescisSTRING	secretIDisINT	typeIDisINT	speedIDisINT	recSendTypeisINT	refDocIDisINT	docCodeisUnique	docIDisUnique	typeIDExist	docIDExist	getDocRecID	getactionID	getCreateDate	getRecieveDate	getDocID	getProposeID	getLineID
1	นำเข้าข้อมูลหนังสือ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	นำเข้าข้อมูลหนังสือ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
3	ข้อมูลหนังสือ ครบถ้วน	t	t	t	t	t	t	t	-	-	-	-	-	-	-	-	-	-	-
4	ข้อมูลหนังสือ ครบถ้วน	f	f	f	f	f	f	f	-	-	-	-	-	-	-	-	-	-	-
5	ประเภทข้อมูลของ หนังสือถูกต้อง	-	-	-	-	-	-	-	t	t	-	-	-	-	-	-	-	-	-
6	ประเภทข้อมูลของ หนังสือถูกต้อง	-	-	-	-	-	-	-	f	f	-	-	-	-	-	-	-	-	-

ตารางที่ ก. 3 ตัวอย่างการเปลี่ยนสถานะของแผนภาพสเตทแมชชีนองค์ประกอบการลงทะเบียนหนังสือ

ลำดับ	สถานะ ปัจจุบัน	เงื่อนไขเหตุการณ์	สถานะเป้าหมาย
		signIDExist	
1	นำเข้าข้อมูลหนังสือ	-	ข้อมูลหนังสือครบถ้วน
2	นำเข้าข้อมูลหนังสือ	-	นำเข้าข้อมูลหนังสือ
3	ข้อมูลหนังสือครบถ้วน	-	ประเภทข้อมูลของ หนังสือถูกต้อง
4	ข้อมูลหนังสือครบถ้วน	-	นำเข้าข้อมูลหนังสือ
5	ประเภทข้อมูลของ หนังสือถูกต้อง	-	เลขทะเบียนหนังสือไม่ซ้ำ
6	ประเภทข้อมูลของ หนังสือถูกต้อง	-	นำเข้าข้อมูลหนังสือ

ตารางที่ ก.4 ตัวอย่างข้อกำหนดรายละเอียดประพจน์ที่ทำการขยายตัวกระตุ้นเหตุการณ์

หมายเลขประพจน์	สถานะปัจจุบัน	เงื่อนไขเหตุการณ์		สถานะเป้าหมาย
		ก่อนกระจาย	หลังกระจาย	
P1	นำเข้าข้อมูลหนังสือ	@T(docEntityCheck) & getDocID & getDocCode & getCreateDate & ~getSubject	~docEntityCheck & _docEntityCheck & getDocID & getDocCode & getCreateDate & getSubject	ข้อมูลหนังสือครบถ้วน
P2	นำเข้าข้อมูลหนังสือ	@T(docEntityCheck) & ~getdocID | ~getdocCode | ~getcreateDate | ~getSubject	~docEntityCheck & _docEntityCheck & ~getdocID | ~getdocCode | ~getcreateDate | ~getSubject	นำเข้าข้อมูลหนังสือ
P3	ข้อมูลหนังสือครบถ้วน	@T(domaincheck) & docIDisINT & docCodeisSTRING & createDateisDATE & subjectisSTRING & toisSTRING & shortDescisSTRING & secretIDisINT & typeIDisINT & speedIDisINT & recSendTypeIDisINT & refdocIDisINT	~domaincheck & _domaincheck & docIDisINT & docCodeisSTRING & createDateisDATE & subjectisSTRING & toisSTRING & shortDescisSTRING & secretIDisINT & typeIDisINT & speedIDisINT & recSendTypeIDisINT & refdocIDisINT	ประเภทข้อมูลของหนังสือถูกต้อง

ตารางที่ ก.5 ตัวอย่างกรณีทดสอบขององค์ประกอบการลงทะเบียนหนังสือ

หมายเลขประพจน์	สถานะปัจจุบัน	หลังกระจาย	เงื่อนไขเหตุการณ์				
			docEntityCheck	docDomainCheck	checkDocCode	checkDocType	checkDocID
P1	นำเข้าข้อมูลหนังสือ	~docEntityCheck & _docEntityCheck & getDocID & getDocCode & getCreateDate & getSubject	T	-	-	-	-
			F	-	-	-	-
			F	-	-	-	-
			F	-	-	-	-
			F	-	-	-	-
			F	-	-	-	-
			F	-	-	-	-

ตารางที่ ก. 6 ตัวอย่างกรณีทดสอบขององค์ประกอบการลงทะเบียนหนังสือ

หมายเลขประพจน์	สถานะปัจจุบัน	เงื่อนไขเหตุการณ์													
		recEntityCheck	recDomainCheck	recCheckID	checkRefInfo	save	getDocID	getDocCode	getCreateDate	getSubject	docIDisINT	docCodeIsSTRING	createDateisDate	subjectisSTRING	toisSTRING
P1	นำเข้าข้อมูลหนังสือ	-	-	-	-	-	T	T	T	T	-	-	-	-	-
		-	-	-	-	-	T	T	T	T	-	-	-	-	-
		-	-	-	-	-	T	T	T	T	-	-	-	-	-
		-	-	-	-	-	F	T	T	T	-	-	-	-	-
		-	-	-	-	-	T	F	T	T	-	-	-	-	-
		-	-	-	-	-	T	T	F	F	-	-	-	-	-
		-	-	-	-	-	T	T	T	F	-	-	-	-	-

ภาคผนวก ข

คำอธิบายยูสเคสโดยละเอียดของเครื่องมือการประเมินความเสี่ยง

ตารางที่ ข. 1 คำอธิบายยูสเคสการสร้างโครงการที่นำมาประเมิน

ชื่อยูสเคส: สร้างโครงการที่นำมาประเมิน	**รหัส:** SRR-001	**ระดับความสำคัญ:** น้อย
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงวิธีการสร้างโครงการพัฒนาซอฟต์แวร์ที่นำมาประเมินความเสี่ยง		
สิ่งกระตุ้น : ผู้ใช้เลือกเมนูนำเข้าข้อมูลความเสี่ยง **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายนอก (External Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: ตั้งค่าคะแนนระดับความรุนแรงตามร้อยละ นำเข้าข้อมูลลำดับเหตุการณ์ การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 1. ผู้ใช้เข้าสู่หน้าจอนำเข้าข้อมูลความเสี่ยง 2. ระบบแสดงแบบฟอร์มการนำเข้าข้อมูลซึ่งประกอบด้วยชื่อโครงการที่นำมาประเมินความเสี่ยง 3. ระบบบันทึกข้อมูลของโครงการ		
ขั้นตอนการทำงานย่อย: -		
ขั้นตอนการทำงานทางเลือก/พิเศษ: -		

ตารางที่ ข. 2 คำอธิบายยูสเคสการตั้งค่าคะแนนระดับความรุนแรงตามร้อยละผลกระทบ

ชื่อยูสเคส: ตั้งค่าคะแนนระดับความรุนแรงตามร้อยละผลกระทบ	**รหัส:** SRR-002	**ระดับความสำคัญ:** ปานกลาง
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงวิธีการตั้งค่าคะแนนระดับความรุนแรงตามร้อยละของแต่ละโครงการ		
สิ่งกระตุ้น : ผู้ใช้เลือกเมนูนำเข้าข้อมูลความเสี่ยง **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายนอก (External Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: - การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 1. ผู้ใช้เข้าสู่หน้าจอนำเข้าข้อมูลความเสี่ยง 2. ระบบแสดงแบบฟอร์มการตั้งค่าระดับความรุนแรง ในลักษณะการให้คะแนนในแต่ละช่วงร้อยละของผลกระทบ 3. ระบบบันทึกข้อมูลคะแนนและร้อยละตามช่วงของระดับความรุนแรงที่กำหนดไว้		
ขั้นตอนการทำงานย่อย: -		
ขั้นตอนการทำงานทางเลือก/พิเศษ: -		

ตารางที่ ข. 3 คำอธิบายยูสเคสการนำเข้าข้อมูลลำดับเหตุการณ์

ชื่อยูสเคส: นำเข้าข้อมูลลำดับเหตุการณ์	**รหัส:** SRR-003	**ระดับความสำคัญ:** ปานกลาง
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงถึงขั้นตอนการนำเข้าข้อมูลลำดับเหตุการณ์		
สิ่งกระตุ้น : ผู้ใช้เลือกเมนูนำเข้าข้อมูลความเสี่ยง **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายนอก (External Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: คำนวณความเสี่ยงระหว่างองค์ประกอบ การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 1. ผู้ใช้เข้าสู่หน้าจอนำเข้าข้อมูลความเสี่ยง 2. ระบบแสดงแบบฟอร์มการนำเข้าลำดับเหตุการณ์ ซึ่งประกอบไปด้วยชื่อลำดับเหตุการณ์และเอกสารแผนภาพลำดับในรูปแบบเอ็กซ์เอ็มแอล 3. ระบบบันทึก		
ขั้นตอนการทำงานย่อย:		
ขั้นตอนการทำงานทางเลือก/พิเศษ: 2.1 หากผู้ใช้ต้องการสร้างลำดับเหตุการณ์มากกว่าหนึ่งลำดับเหตุการณ์สามารถเพิ่มเติมลำดับเหตุการณ์ได้ผ่านการกดปุ่มเพิ่มได้ทันที โดยปราศจากประมวลผลไปยังหน้าการทำงานอื่น		

ตารางที่ ข. 4 คำอธิบายยูสเคสการคำนวณความเสี่ยงระหว่างองค์ประกอบ

ชื่อยูสเคส: คำนวณความเสี่ยงระหว่างองค์ประกอบ	**รหัส:** SRR-004	**ระดับความสำคัญ:** ปานกลาง
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงถึงขั้นตอนการประเมินความเสี่ยงระหว่างองค์ประกอบ		
สิ่งกระตุ้น : ระบบนำเข้าแผนภาพลำดับในแต่ละลำดับเหตุการณ์สำเร็จ **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายใน(Internal Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: - การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 1. ระบบสกัดเอาองค์ประกอบที่ปรากฏในแผนภาพลำดับในรูปแบบเอกสารเอ็กซ์เอ็มแอล และพิจารณาจำนวนข้อความระหว่างองค์ประกอบในแผนภาพ 2. ระบบนำข้อมูลที่ได้ในข้อที่หนึ่งมาคำนวณหาค่าความเสี่ยงของปฏิสัมพันธ์ระหว่างองค์ประกอบตามที่นำเสนอไว้ในบทที่สาม 3. ระบบบันทึกผลการคำนวณค่าความเสี่ยงระหว่างองค์ประกอบ		
ขั้นตอนการทำงานย่อย:		
ขั้นตอนการทำงานทางเลือก/พิเศษ:		

ตารางที่ ข. 5 คำอธิบายยูสเคสการตั้งค่าระดับผลกระทบจากคุณสมบัติประจำตาราง

ชื่อยูสเคส: ตั้งค่าระดับผลกระทบจากคุณสมบัติประจำตาราง	**รหัส:** SRR-005	**ระดับความสำคัญ:** ปานกลาง
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงวิธีการตั้งค่าระดับผลกระทบจากคุณสมบัติประจำตาราง		
สิ่งกระตุ้น : ระบบประมวลผลเอกสารเอสคิวแอลเรียบร้อย **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายใน (Internal Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: คำนวณระดับความรุนแรง การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 1. ระบบแสดงหน้าจอสำหรับการตั้งค่า ซึ่งเป็นการสกัดเอาคุณลักษณะประจำตารางจากแฟ้มเอกสารเอสคิวแอล 2. ผู้ใช้ตั้งค่าให้กับคุณลักษณะประจำตารางที่ปรากฏในหน้าจอด้วยระดับผลกระทบสูง กลาง หรือต่ำ 3. ระบบทำการบันทึกข้อมูล จากนั้นคำนวณระดับผลกระทบในแต่ละตารางและฐานข้อมูล		
ขั้นตอนการทำงานย่อย:		
ขั้นตอนการทำงานทางเลือก/พิเศษ:		

ตารางที่ ข. 6 คำอธิบายยูสเคสการคำนวณระดับความรุนแรง

ชื่อยูสเคส: คำนวณระดับความรุนแรง	**รหัส:** SRR-006	**ระดับความสำคัญ:** ปานกลาง
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงวิธีการคำนวณระดับความรุนแรงของตารางและฐานข้อมูล		
สิ่งกระตุ้น : ผู้ใช้กำหนดระดับผลกระทบในแต่ละคุณลักษณะประจำเรียบร้อย **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายนอก (External Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: - การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 1. ระบบนำเอาการตั้งค่าระดับความรุนแรงในแต่ละคุณลักษณะประจำมาประมวลผลระดับผลกระทบในแต่ละตาราง 2. ระบบทำการประมวลผลระดับผลกระทบในส่วนของฐานข้อมูล 3. ระบบนำผลลัพธ์ที่ได้จากข้อที่หนึ่งและสองมาคำนวณหาระดับความรุนแรงในแต่ละตาราง ตามขั้นตอนการคำนวณหาระดับความรุนแรงที่นำเสนอไว้ใน 4. ระบบบันทึกผลลัพธ์จากความรุนแรงที่ได้จากการคำนวณของแต่ละตาราง		
ขั้นตอนการทำงานย่อย:		
ขั้นตอนการทำงานทางเลือก/พิเศษ:		

ตารางที่ ข. 7 คำอธิบายยูสเคสการนำเข้าข้อมูลแผนภาพสเตทแมชชีนขององค์ประกอบ

ชื่อยูสเคส: นำเข้าข้อมูลแผนภาพสเตทแมชชีนขององค์ประกอบ	**รหัส:** SRR-007	**ระดับความสำคัญ:** ปานกลาง
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงวิธีการนำเข้าข้อมูลแผนภาพสเตทแมชชีนขององค์ประกอบ		
สิ่งกระตุ้น : ผู้ใช้เลือกเมนูนำเข้าข้อมูลขององค์ประกอบ **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายนอก (External Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: คำนวณความเสี่ยงระดับองค์ประกอบ การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 5. หลังจากการตั้งค่าระดับความรุนแรงของคุณลักษณะประจำตารางแล้ว ระบบแสดงหน้าจอสำหรับนำเข้าข้อมูลองค์ประกอบ 6. ผู้ใช้เลือกลำดับเหตุการณ์ขององค์ประกอบที่ต้องการนำเข้า โดยที่ระบบแสดงรายการของลำดับเหตุการณ์ที่ถูกบันทึกไว้ในขั้นตอนการสร้างลำดับเหตุการณ์ 7. ผู้ใช้ระบุชื่อขององค์ประกอบ และเอกสารแผนภาพสเตทแมชชีนขององค์ประกอบในรูปแบบเอกสารเอ็กซ์เอ็มแอล 8. ระบบบันทึกข้อมูลขององค์ประกอบที่ได้นำเข้า		
ขั้นตอนการทำงานย่อย:		
ขั้นตอนการทำงานทางเลือก/พิเศษ: 2.1 ผู้ใช้สามารถนำเข้าข้อมูลขององค์ประกอบเข้าสู่ระบบได้มากกว่าหนึ่งองค์ประกอบ โดยการกดปุ่มเพิ่มจำนวนองค์ประกอบ โดยปราศจากการประมวลผลไปยังหน้าการทำงานอื่น		

ตารางที่ ข. 8 คำอธิบายยูสเคสการคำนวณความเสี่ยงระดับองค์ประกอบ

ชื่อยูสเคส: คำนวณความเสี่ยงระดับองค์ประกอบ	**รหัส:** SRR-008	**ระดับความสำคัญ:** มาก
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงวิธีการคำนวณความเสี่ยงระดับองค์ประกอบ		
สิ่งกระตุ้น : ระบบบันทึกข้อมูลองค์ประกอบที่นำเข้าเรียบร้อยแล้ว **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายใน (Internal Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: - การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 1. ระบบนำเอาประพจน์ที่ปรากฏบนการเปลี่ยนแปลงสถานะภายในแผนภาพสเตทแมชชีน 2. ระบบแสดงแบบฟอร์มการนำเข้าข้อมูลความเสี่ยงอันประกอบด้วย ชื่อโครงการ การตั้งค่าระดับความรุนแรง ส่วนนำเข้าเอกสารเอสคิวแอล ชื่อลำดับเหตุการณ์และแผนภาพลำดับในรูปแบบเอกสารเอ็กซ์เอ็มแอล 3. ในส่วนการนำเข้าลำดับเหตุการณ์ ระบบอำนวยความสะดวกในการสร้างลำดับเหตุการณ์ได้มากกว่าหนึ่งลำดับเหตุการณ์ 4. ระบบบันทึก		
ขั้นตอนการทำงานย่อย:		
ขั้นตอนการทำงานทางเลือก/พิเศษ:		

ตารางที่ ข. 9 คำอธิบายยูสเคสการรายงานผลการประเมินความเสี่ยง

ชื่อยูสเคส: รายงานผลการประเมินความเสี่ยง	**รหัส:** SRR-009	**ระดับความสำคัญ:** ปานกลาง
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงวิธีการสร้างโครงการพัฒนาซอฟต์แวร์ที่นำมาประเมินความเสี่ยง		
สิ่งกระตุ้น : ผู้ใช้เลือกเมนูรายงานผลการประเมินความเสี่ยง **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายนอก (External Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: คำนวณความเสี่ยงในลำดับเหตุการณ์ การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 1. ผู้ใช้เลือกโครงการที่ต้องการทราบรายงานการประเมินความเสี่ยง 2. ระบบแสดงผลลัพธ์ของการประเมินความเสี่ยงในโครงการตามลำดับเหตุการณ์ที่ปรากฏในโครงการนั้นๆ		
ขั้นตอนการทำงานย่อย:		
ขั้นตอนการทำงานทางเลือก/พิเศษ:		

ตารางที่ ข. 10 คำอธิบายยูสเคสการคำนวณความเสี่ยงในลำดับเหตุการณ์

ชื่อยูสเคส: คำนวณความเสี่ยงในลำดับเหตุการณ์	**รหัส:** SRR-010	**ระดับความสำคัญ:** มาก
ผู้กระทำหลัก: ผู้ประเมิน		**ประเภทยูสเคส:** เชิงละเอียด
ผู้มีส่วนเกี่ยวข้องและการใช้ประโยชน์: ผู้มีความประสงค์ในการประเมินความเสี่ยงของโครงการพัฒนาซอฟต์แวร์ในขั้นตอนการออกแบบ		
คำอธิบาย: เป็นยูสเคสที่ใช้แสดงวิธีการประเมินความเสี่ยงของโครงการในส่วนของลำดับเหตุการณ์		
สิ่งกระตุ้น : ระบบนำเข้าข้อมูลขององค์ประกอบครบถ้วนทุกลำดับเหตุการณ์ **ประเภทของสิ่งกระตุ้น:** สิ่งกระตุ้นภายใน (Internal Trigger)		
ความสัมพันธ์: ความเกี่ยวเนื่อง: - การรวม: - การขยาย: - การรับทอดคุณสมบัติ: -		
ขั้นตอนการทำงานปกติ: 1. ระบบนำผลลัพธ์ที่ได้จากการคำนวณความเสี่ยงในระดับขององค์ประกอบ และปฏิสัมพันธ์ระหว่างองค์ประกอบ มาคำนวณหาความเสี่ยงในลำดับเหตุการณ์ตามขั้นตอนวิธีที่ได้นำเสนอไว้ในบทที่สาม		
ขั้นตอนการทำงานย่อย:		
ขั้นตอนการทำงานทางเลือก/พิเศษ:		

ประวัติผู้เขียนวิทยานิพนธ์

นางสาวกาญจนา เอี่ยมสอาด เกิดเมื่อวันที่ 24 มกราคม พ.ศ. 2529 ที่จังหวัดชลบุรี สำเร็จการศึกษาระดับปริญญาตรี วิทยาศาสตรบัณฑิต จากภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยบูรพา ในปีการศึกษา 2550 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2554