

แนวทางการสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้

นายชวลิต เขี่ยมสะอาด

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2554

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)

เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ที่ส่งผ่านทางบัณฑิตวิทยาลัย

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

APPROACH TO GENERATING TEST CASES FOR USER ACCEPTANCE TEST

Mr. Chavalid Ieamsaard

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 2011

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	แนวทางการสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้
โดย	นายชวลิต เขี่ยมสะอาด
สาขาวิชา	วิศวกรรมซอฟต์แวร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	รองศาสตราจารย์ ดร.ญาใจ ลิมปิยะภรณ์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้รับวิทยานิพนธ์ฉบับนี้
เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศธีรวัฒน์)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(รองศาสตราจารย์ ดร.ญาใจ ลิมปิยะภรณ์)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.สุกรี สิ้นธุภิณโณ)

.....กรรมการภายนอกมหาวิทยาลัย
(อาจารย์ ดร.ภาสกร อภิรักษ์วรพินิต)

ชวลิต เขี่ยมสะอาด : แนวทางการสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้. (APPROACH TO GENERATING TEST CASES FOR USER ACCEPTANCE TEST) อ. ที่ปรึกษาวิทยานิพนธ์หลัก : รศ.ดร.ญาใจ ลิ้มปิยะกรณ์, 84 หน้า.

เมื่อขั้นตอนการวิเคราะห์ความต้องการได้เสร็จสิ้น ชุดทดสอบการยอมรับของผู้ใช้สามารถสร้างได้โดยพลันจากรายการความต้องการของผู้ใช้ เนื่องจากการทดสอบเป็นกิจกรรมที่สิ้นเปลืองทรัพยากร ก่อปรกัไม่สามารถหลีกเลี่ยงการเปลี่ยนแปลงความต้องการที่มักเกิดขึ้นในโครงการซอฟต์แวร์ งานวิจัยนี้ได้นำเสนอแนวทางการสร้างชุดกรณีทดสอบการยอมรับของผู้ใช้ จากคำอธิบายยูสเคสแบบอัตโนมัติ นอกจากนี้ ระบบที่พัฒนาขึ้นซึ่งเป็นส่วนหนึ่งของระบบเก็บเกี่ยวความต้องการยังสามารถส่งออกความสัมพันธ์ระหว่างยูสเคสกับกรณีทดสอบความต้องการของยูสเคสนั้นไปยังเมตริกซ์ความสามารถตามรอยความต้องการโดยอัตโนมัติ ความพยายามของงานวิจัยนี้นำไปสู่การปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ กล่าวคือ ทำให้ค่าใช้จ่ายการทดสอบลดลง และสามารถจัดการการเปลี่ยนแปลงความต้องการได้ดียิ่งขึ้น

ภาควิชา วิศวกรรมคอมพิวเตอร์ ปลายมือชื่อ นิสิต.....
 สาขาวิชา วิศวกรรมซอฟต์แวร์ ปลายมือ อ.ที่ปรึกษาวิทยานิพนธ์หลัก.....
 ปีการศึกษา 2554.....

5270746621 : MAJOR SOFTWARE ENGINEERING

KEYWORDS: USE CASE / USER ACCEPTANCE TESTS / REQUIREMENTS
MANAGEMENT / SOFTWARE PROCESS IMPROVEMENT

CHAVALID IEAMSAARD : APPROACH TO GENERATING TEST CASES FOR
USER ACCEPTANCE TEST. ADVISOR : ASSOC. PROF. YACHAI
LIMPIYAKORN, Ph.D., 84 pp.

Once the phase of requirements analysis has finished, user acceptance tests can be derived from the user requirements. Since testing is a resource consuming activity and it is inevitable for requirements changes in a software project, this research presents an automation approach to generating user acceptance tests from use case descriptions. Additionally, the implemented system, which is a component of Requirements Harvester system, can automatically export the relationships between use cases and their test cases to traceability matrix. The efforts of this research contribute to the improvement of software development process. That is, it could reduce the cost of testing and provide the ability to better manage requirements changes.

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

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

Academic Year : 2011.....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความอนุเคราะห์อย่างยิ่งของรองศาสตราจารย์ ดร.ญาใจ ลิ้มปิยะภรณ์ อาจารย์ที่ปรึกษา ซึ่งท่านได้ให้ความรู้ แนวทางการวิจัย ตรวจสอบให้คำแนะนำ และสนับสนุนเป็นอย่างดี จนทำให้การวิจัยในครั้งนี้สำเร็จออกมาด้วยดี

ขอขอบพระคุณศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล ผู้ช่วยศาสตราจารย์ ดร.สุกรี สินธุภิญโญ และอาจารย์ ดร.ภาสกร อภิรักษ์วรพินิต กรรมการสอบวิทยานิพนธ์ ที่กรุณาเสียสละเวลาให้คำแนะนำ ตรวจสอบ และแก้ไขวิทยานิพนธ์ฉบับนี้

ขอกราบขอบพระคุณบิดา มารดา และญาติพี่น้องที่ให้การสนับสนุนและเป็นกำลังใจที่ดี ให้เสมอมา และสนับสนุนด้านทุนทรัพย์ในการศึกษา รวมไปถึงทุกท่านที่มีส่วนช่วยเหลือในการทำวิทยานิพนธ์ครั้งนี้ ซึ่งมีได้กล่าวนามในที่นี้

ท้ายที่สุด ผู้วิจัยขอขอบคุณ เพื่อนทุกๆคนที่ติดตามและให้กำลังใจ รวมถึงท่านอื่นๆ ที่ได้กล่าวชื่อไว้ ณ ที่นี้ที่มีส่วนทำให้วิทยานิพนธ์สำเร็จได้ด้วยดี ผู้วิจัยหวังเป็นอย่างยิ่งว่าวิทยานิพนธ์ฉบับนี้จะเป็นประโยชน์บ้างไม่มากนักน้อยสำหรับผู้สนใจจะศึกษารายละเอียดต่อไป

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ง
บทคัดย่อภาษาอังกฤษ	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ	ช
สารบัญตาราง.....	ฌ
สารบัญภาพ	ฎ
บทที่ 1.....	1
บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย	2
1.3 ขอบเขตของการวิจัย	2
1.4 ขั้นตอนและวิธีดำเนินการวิจัย.....	2
1.5 ประโยชน์ที่ได้รับ	3
1.6 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์.....	3
1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์.....	4
บทที่ 2.....	5
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	5
2.1 ทฤษฎีที่เกี่ยวข้อง	5
2.2 งานวิจัยที่เกี่ยวข้อง	14
บทที่ 3.....	18
การออกแบบสถาปัตยกรรมระบบ.....	18
3.1 วิเคราะห์โครงสร้างระบบ.....	19
3.2 ข้อกำหนดเงื่อนไขและการทำงานของระบบ.....	25
3.3 วิเคราะห์บุคคลที่มีบทบาทหน้าที่ของผู้ที่เกี่ยวข้องกับระบบ.....	27
บทที่ 4.....	28
การออกแบบและพัฒนาระบบ	28
4.1 ความต้องการด้านหน้าที่	28
4.2 ความต้องการที่ไม่ใช่หน้าที่.....	29

4.3 การออกแบบระบบ.....	29
4.4 การพัฒนาระบบ.....	42
4.4.1 สภาพแวดล้อมและเครื่องมือที่ใช้ในการพัฒนา.....	42
4.4.2 การติดตั้งซอฟต์แวร์ในการพัฒนาระบบ.....	43
4.4.3 การพัฒนาส่วนต่อประสาน.....	43
บทที่ 5 การทดสอบระบบ.....	53
5.1 การทดสอบความถูกต้องฟังก์ชันการทำงานของระบบ.....	53
5.2 การทดสอบความถูกต้องกระบวนการทำงานของระบบ.....	58
5.3 การประเมินผลระบบสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้.....	60
5.4 สรุปผลการทดลอง.....	62
บทที่ 6.....	63
สรุปผลการวิจัยและข้อเสนอแนะ.....	63
6.1 สรุปผลการวิจัย.....	63
6.2 ข้อจำกัด.....	63
6.3 แนวทางการวิจัยต่อ.....	63
รายการอ้างอิง.....	64
ภาคผนวก.....	65
ภาคผนวก ก. พจนานุกรมข้อมูลของระบบฐานข้อมูล.....	66
ภาคผนวก ข. ตัวอย่างแสดงการทำงานของกรณีทดสอบ.....	75

สารบัญตาราง

หน้า

ตารางที่ 1 หัวข้อข้อกำหนดความต้องการซอฟต์แวร์ตามมาตรฐานไอทีริปเปิลอี 830-1998	5
ตารางที่ 2 ตัวอย่างเค้าโครงคำอธิบายยูสเคส	12
ตารางที่ 3 ตัวอย่างยูสเคสซีนาริโอที่ได้จากคำอธิบายยูสเคส	20
ตารางที่ 4 ตัวอย่างกรณีทดสอบการยอมรับของผู้ใช้	22
ตารางที่ 5 วิธีดำเนินการสร้างซีนาริโอ	23
ตารางที่ 6 วิธีดำเนินการสร้าง Extended finite state machine (EFSM)	24
ตารางที่ 7 วิธีดำเนินการสร้างกรณีทดสอบที่ครอบคลุม	25
ตารางที่ 8 ลักษณะโดยรวมของระบบ	25
ตารางที่ 9 ตัวอย่างการเขียนคำอธิบายยูสเคส	26
ตารางที่ 10 ความต้องการด้านหน้าที่	28
ตารางที่ 11 ความต้องการที่ไม่ใช่หน้าที่	29
ตารางที่ 12 คำอธิบายยูสเคส Import XML	30
ตารางที่ 13 คำอธิบายยูสเคส Manage Use Case Information	31
ตารางที่ 14 คำอธิบายยูสเคส Generates Test Case	32
ตารางที่ 15 คำอธิบายยูสเคส Manage Test Case Information	33
ตารางที่ 16 คำอธิบายยูสเคส Export XML To other System	34
ตารางที่ 17 คำอธิบายยูสเคส Create Scenario	35
ตารางที่ 18 คำอธิบายยูสเคส Create EFSM Model	36
ตารางที่ 19 คำอธิบายยูสเคส Create Test Path	37
ตารางที่ 20 ผลการทดสอบความต้องการเชิงหน้าที่	53
ตารางที่ 21 ผลการทดสอบความต้องการที่ไม่ใช่เชิงหน้าที่	57
ตารางที่ 22 การทดสอบความถูกต้องกระบวนการทำงานของระบบ	59
ตารางที่ 23 วิธีการทำการทดลองใช้ในการเปรียบเทียบจำนวนนำเข้าข้อมูลคำอธิบายยูสเคสจากเอกสารครั้งแรก	60
ตารางที่ 24 วิธีการทำการทดลองใช้ในการเปรียบเทียบจำนวนนำเข้าข้อมูลคำอธิบายยูสเคสจากเมื่อเอกสารเกิดการเปลี่ยนแปลง	61
ตารางที่ 25 รายชื่อตารางทั้งหมด	66
ตารางที่ 26 พจนานุกรมข้อมูล ตาราง TBL_PROJECT_UAT	66

ตารางที่ 27 พจนานุกรมข้อมูล ตาราง TBL_IMPORTTMP.....	67
ตารางที่ 28 พจนานุกรมข้อมูล ตาราง TBL_UC_MASTER.....	69
ตารางที่ 29 พจนานุกรมข้อมูล ตาราง TBL_NORMAL_FLOW	71
ตารางที่ 30 พจนานุกรมข้อมูล ตาราง TBL_ALTERNATE_FLOW	71
ตารางที่ 31 พจนานุกรมข้อมูล ตาราง TBL_UAT_MASTER.....	72
ตารางที่ 32 พจนานุกรมข้อมูล ตาราง TBL_UAT_DETAIL	73
ตารางที่ 33 พจนานุกรมข้อมูล ตาราง TBL_EFSM_STATE	73
ตารางที่ 34 พจนานุกรมข้อมูล ตาราง TBL_EFSM_STATE_TRANSITION	74

สารบัญภาพ

	หน้า
ภาพที่ 1 ระดับของการทดสอบ.....	9
ภาพที่ 2 ผู้ใช้งานระบบ	10
ภาพที่ 3 ยูสเคส	10
ภาพที่ 4 ความสัมพันธ์ของยูสเคสแบบต่างๆ	11
ภาพที่ 5 ตัวอย่างยูสเคสที่นำริโอของระบบ	11
ภาพที่ 6 ภาพรวมงานวิจัยการสร้างกรณีทดสอบจื๋โยจากยูเอ็มแอล	15
ภาพที่ 7 ภาพรวมงานวิจัย	16
ภาพที่ 8 นิยาม Use Case ตามแนวคิดการวิจัย	16
ภาพที่ 9 นิยาม Extended finite state machin ตามแนวคิดการวิจัย	16
ภาพที่ 10 อัลกอริทึมเพื่อสนับสนุนแนวคิดการวิจัย	17
ภาพที่ 11 ภาพรวมงานวิจัยที่น่าเสนอ.....	19
ภาพที่ 12 ตัวอย่างข้อมูลนำเข้าในรูปแบบเอ็กซ์เอ็มแอล	20
ภาพที่ 13 ตัวอย่างกราฟที่ได้จาก EFSM Model.....	21
ภาพที่ 14 แผนภาพยูสเคสระบบสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้.....	29
ภาพที่ 15 แผนภาพการออกแบบคลาสของระบบ	38
ภาพที่ 16 แผนภาพซีควเอนซ์ Import XML	39
ภาพที่ 17 แผนภาพซีควเอนซ์ Manage Use Case Info	39
ภาพที่ 18 แผนภาพซีควเอนซ์ Generates Test Case.....	40
ภาพที่ 19 แผนภาพซีควเอนซ์ Manage UAT Info	40
ภาพที่ 20 แผนภาพซีควเอนซ์ Extract XML	41
ภาพที่ 21 หน้าจอส่วนต้นไม้นำทาง	44
ภาพที่ 22 หน้าจอโปรแกรมสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้	44
ภาพที่ 23 หน้าจอเมนูไฟล์	45
ภาพที่ 24 หน้าจอเมนูเครื่องมือ	45
ภาพที่ 25 หน้าจอเมนูคลิกขวา	45
ภาพที่ 26 ข้อมูลนำเข้าคำอธิบายยูสเคสในรูปแบบเอ็กซ์เอ็มแอล	46
ภาพที่ 27 หน้าจอเลือกไฟล์เอกสารเอ็กซ์เอ็มแอล	46
ภาพที่ 28 หน้าจอ Use Case Information	47

ภาพที่ 29 หน้าจอ Data State Table.....	48
ภาพที่ 30 หน้าจอ EFSM Graph.....	49
ภาพที่ 31 หน้าจอ UAT.....	49
ภาพที่ 32 หน้าจอ Report UAT.....	50
ภาพที่ 33 หน้าจอ Log UC	51
ภาพที่ 34 หน้าจอ Log UAT	52
ภาพที่ 35 กราฟความแตกต่างของจำนวนนำเข้าข้อมูลคำอธิบายยูสเคสจากเอกสารครั้งแรก กับ จำนวนที่ระบบสร้างข้อมูลกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติ	61
ภาพที่ 36 กราฟความแตกต่างของจำนวนนำเข้าข้อมูลคำอธิบายยูสเคสเมื่อเอกสารเกิดการ เปลี่ยนแปลง กับจำนวนที่ระบบสร้างข้อมูลกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติ	62
ภาพที่ 37 ตัวอย่างข้อมูลในหัวข้อ Functional Requirements	75
ภาพที่ 38 ตัวอย่างข้อมูลในหัวข้อ Non-Functional Requirements	75
ภาพที่ 39 ตัวอย่างข้อมูลในหัวข้อ Use Case Description	76
ภาพที่ 40 ตัวอย่างโปรแกรมระบบนำเข้าข้อมูลอัตโนมัติ	76
ภาพที่ 41 เมื่อทำการคลิกขวาที่โครงการ.....	77
ภาพที่ 42 ไดอะล็อกสำหรับเลือกเอกสารเข้าสู่ระบบ	77
ภาพที่ 43 ไดอะล็อกสำหรับกำหนดค่าเอกสารการนำเข้าโดยผ่านทางข้อกำหนดของแท็ก	78
ภาพที่ 44 ไดอะล็อกรายงานผลการสกัดข้อมูลอย่างย่อ.....	79
ภาพที่ 45 หน้าจอส่วนต้นไม้นำทาง	79
ภาพที่ 46 โปรแกรมเมตริกซ์ความสามารถตามรอยความต้องการ	80
ภาพที่ 47 ตัวอย่างเอ็กซ์เอ็มแอลที่ระบบสร้างขึ้นให้ระบบภายนอก	80
ภาพที่ 48 ตัวอย่างข้อมูลในหัวข้อ User Acceptance Test	81
ภาพที่ 49 โหลดเอกสารข้อกำหนดการทดสอบซอฟต์แวร์	81
ภาพที่ 50 ไดอะล็อกสำหรับกำหนดค่าเอกสารการนำเข้าโดยผ่านทางข้อกำหนดของแท็ก ในส่วน เอกสารข้อกำหนดการทดสอบซอฟต์แวร์.....	82
ภาพที่ 51 รายงานผลการสกัดอย่างย่อ	82
ภาพที่ 52 โปรแกรมเมตริกซ์ความสามารถตามรอยความต้องการ	83
ภาพที่ 53 เลือกดูความสัมพันธ์ระหว่างแถวและหลัก	83

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ระบบซอฟต์แวร์ที่ใช้งานในปัจจุบัน มักเป็นระบบซอฟต์แวร์ขนาดใหญ่ มีความซับซ้อน และมีการเปลี่ยนแปลงอยู่เสมอ ทั้งนี้เนื่องจากการขยายที่จะให้ผู้เกี่ยวข้องทำการระบุปัญหา และความต้องการต่างๆที่เกิดขึ้นได้ทั้งหมดในช่วงแรกของการพัฒนาระบบซอฟต์แวร์ ส่งผลให้การออกแบบและพัฒนาระบบซอฟต์แวร์ไม่ถูกต้องสมบูรณ์ และเกิดปัญหาต่างๆ ด้านข้อกำหนด ความต้องการซอฟต์แวร์ (Software Requirements Specification: SRS) จึงจำเป็นจะต้องมีเอกสารข้อกำหนดความต้องการของซอฟต์แวร์ เพื่อเป็นข้อตกลงกับผู้มีส่วนได้ส่วนเสีย โดยเอกสารข้อกำหนดความต้องการของซอฟต์แวร์ จะเป็นรูปแบบเบื้องต้นของเอกสารรูปนัย (Formal Document) หรือเป็นรูปแบบภาษาธรรมชาติ (Natural Language) ซึ่งในเนื้อหาของเอกสารดังกล่าว จะบ่งบอกถึงความต้องการของผู้ใช้ระบบงาน และผู้ที่เกี่ยวข้องกับระบบ เพื่อยอมรับในข้อตกลงของความต้องการ ก่อนที่จะจะเริ่มดำเนินการพัฒนาระบบซอฟต์แวร์ ตามเอกสารข้อกำหนดความต้องการของซอฟต์แวร์

หลังจากพัฒนาระบบซอฟต์แวร์แล้วเสร็จ จะมีทีมผู้ทดสอบ ทำการทดสอบรวมระบบซอฟต์แวร์ เพื่อตรวจสอบว่าระบบซอฟต์แวร์ดังกล่าว ทำงานได้ตรงตามความต้องการตามข้อกำหนดความต้องการซอฟต์แวร์ของลูกค้า และระบบซอฟต์แวร์ ทำตามหน้าที่เฉพาะที่กำหนดขึ้น ซึ่งจากการทำงานของทีมผู้ทดสอบ พบว่ามี ค่าใช้จ่าย เวลา ทรัพยากร และแรงงานเกิดขึ้นสูง ในการที่จะทำการทดสอบระบบซอฟต์แวร์ เทียบกับเอกสารข้อกำหนดความต้องการของซอฟต์แวร์ อีกทั้งหากพบว่า ระบบซอฟต์แวร์มีข้อผิดพลาด จะทำให้เกิดการเสียค่าใช้จ่าย ทรัพยากร และแรงงาน ในการค้นหาสาเหตุของจุดบกพร่อง เพื่อที่จะแก้ไขข้อผิดพลาดที่ค้นพบมากขึ้น ในการที่จะลดโอกาสที่จะเกิดความผิดพลาด ที่อาจจะเกิดขึ้นในขั้นตอนของการพัฒนาระบบซอฟต์แวร์ จึงมีความจำเป็นต้องมีการสร้างกรณีทดสอบเอกสารข้อกำหนดความต้องการของซอฟต์แวร์ ควบคู่ไปกับพัฒนาระบบซอฟต์แวร์ตั้งแต่ต้น

จากปัญหาที่กล่าวข้างต้น งานวิจัยนี้มีจุดประสงค์เพื่อนำเสนอวิธีการอัตโนมัติในการนำเข้าข้อมูลคำอธิบายยูสเคส ซึ่งเป็นส่วนหนึ่งในเอกสารข้อกำหนดความต้องการของซอฟต์แวร์ที่เป็นภาษาไทย และอยู่ในรูปแบบเอกสารภาษาเอ็กซ์เอ็มแอล เพื่อสร้างกรณีทดสอบรูปแบบ Black-Box Testing โดยกรณีทดสอบจะครอบคลุมทุกหน้าที่การทำงานของระบบซอฟต์แวร์ เพื่อประโยชน์ในการลดความผิดพลาด ค่าใช้จ่าย เวลา ทรัพยากร และแรงงาน ที่จะเกิดขึ้นในขั้นตอน

ของการพัฒนาระบบซอฟต์แวร์ รวมทั้งสามารถค้นหาสาเหตุของจุดบกพร่อง เพื่อที่จะแก้ไขข้อผิดพลาดที่ค้นพบได้รวดเร็วยิ่งขึ้น นอกจากนี้ ความสัมพันธ์ระหว่างยูสเคส และกรณีทดสอบของความต้องการจะถูกส่งออกไปยังเมตริกซ์ความสามารถตามรอยความต้องการ

1.2 วัตถุประสงค์ของการวิจัย

1. ออกแบบแนวทางและพัฒนาระบบสนับสนุนสำหรับสร้างกรณีทดสอบแบบ Black Box จากเอกสารข้อกำหนดความต้องการซอฟต์แวร์
2. เพื่อสนับสนุนการปรับปรุงกระบวนการจัดการความต้องการและการทดสอบการยอมรับของผู้ใช้

1.3 ขอบเขตของการวิจัย

1. กรณีทดสอบที่สร้างขึ้นอยู่ในรูปแบบ Black-box Testing
2. ข้อมูลนำเข้า คือ คำอธิบายยูสเคส ซึ่งเป็นส่วนหนึ่งในเอกสารข้อกำหนดความต้องการซอฟต์แวร์ และอยู่ในรูปแบบเอ็กซ์เอ็มแอล
3. ผู้วิจัยเป็นผู้กำหนด XML Schema ให้ระบบภายนอกเพื่อสกัดรายละเอียดข้อมูลในคำอธิบายยูสเคสที่ต้องการใช้สร้างกรณีทดสอบ
4. ระบบสามารถส่งออกข้อมูล เพื่อสนับสนุนการสร้างเมตริกซ์ตามรอยความต้องการระหว่างยูสเคส กับกรณีทดสอบ
5. ผู้วิจัยเป็นผู้กำหนด XML Schema ให้ระบบภายนอกเพื่อสกัดอินสแตนซ์กับความสัมพันธ์ในไฟล์ข้อมูลที่ส่งออกเพื่อนำเข้าเมตริกซ์ความสามารถตามรอยความต้องการ
6. ระบบรองรับการทำงานกับข้อมูลภาษาไทย
7. การประเมินระบบที่พัฒนาขึ้นวัดจากความถูกต้องของกรณีทดสอบที่สร้างขึ้น และความครอบคลุมของกรณีทดสอบที่ครอบคลุมทุกฟังก์ชันหน้าที่การทำงานที่คาดหวังของระบบตามข้อกำหนดความต้องการซอฟต์แวร์

1.4 ขั้นตอนและวิธีดำเนินการวิจัย

1. ศึกษาและทำความเข้าใจทฤษฎีกรณีทดสอบรูปแบบ Black-box Testing
2. ศึกษาและทำความเข้าใจรายละเอียดของเอกสารข้อกำหนดความต้องการซอฟต์แวร์
3. ศึกษางานวิจัยที่เกี่ยวข้องกับการสร้างกรณีทดสอบ จากเอกสารข้อกำหนดความต้องการซอฟต์แวร์

4. ศึกษาและทดลองวิธีการนำเข้าสู่ข้อมูลจากเอกสารข้อกำหนดความต้องการซอฟต์แวร์ภาษาไทย ด้วยภาษา เอ็กซ์เอ็มแอล
5. ศึกษาทำความเข้าใจและกำหนดวิธีการอธิบายยูสเคสที่จำเป็นต่อกรณีทดสอบ
6. วิเคราะห์และกำหนดข้อมูลคำอธิบายยูสเคส จากเอกสารข้อกำหนดความต้องการซอฟต์แวร์
7. เลือกเครื่องมือที่เหมาะสมในการสร้างระบบซอฟต์แวร์เพื่อสนับสนุนแนวทางการวิจัย
8. ออกแบบระบบสนับสนุนเพื่อช่วยในการสร้างกรณีทดสอบ จากเอกสารข้อกำหนดความต้องการซอฟต์แวร์
9. พัฒนาระบบสนับสนุนเพื่อช่วยในการสร้างกรณีทดสอบ จากเอกสารข้อกำหนดความต้องการซอฟต์แวร์
10. ตีพิมพ์ผลงานทางวิชาการ
11. ทดสอบและประเมินผลแนวทางการวิจัย และระบบสนับสนุนเพื่อช่วยในการสร้างกรณีทดสอบ จากเอกสารข้อกำหนดความต้องการซอฟต์แวร์
12. สรุปผลการวิจัยและข้อเสนอแนะ และจัดทำวิทยานิพนธ์

1.5 ประโยชน์ที่ได้รับ

1. ได้วิธีการและระบบต้นแบบสำหรับสร้างกรณีทดสอบแบบ Black-Box Testing จากเอกสารข้อกำหนดความต้องการซอฟต์แวร์แบบอัตโนมัติ เพื่อสนับสนุนการทดสอบการยอมรับของผู้ใช้
2. ได้ข้อมูลสนับสนุนการสร้างเมตริกซ์ตามรอยความต้องการระหว่างยูสเคส กับกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้

1.6 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์

วิทยานิพนธ์นี้แบ่งเนื้อหาออกเป็น 6 บทดังต่อไปนี้ บทที่ 1 เป็นบทนำซึ่งกล่าวถึง ความเป็นมาและความสำคัญของปัญหา รวมถึงวัตถุประสงค์ของการวิจัย บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานและงานวิจัยที่เกี่ยวข้องในงานวิจัยนี้ บทที่ 3 กล่าวถึงการออกแบบสถาปัตยกรรมระบบ บทที่ 4 กล่าวถึงการออกแบบและพัฒนาระบบ บทที่ 5 กล่าวถึงวิธีการทดสอบระบบ และบทที่ 6 กล่าวถึงสรุปผลการวิจัยและข้อเสนอแนะ

1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้รับการตีพิมพ์เป็นบทความทางวิชาการในหัวข้อเรื่อง “On Integrating User Acceptance Tests Generation to Requirements Management”, C. leamsaard, and Y. Limpiyakorn. Proceedings of 2011 International Conference on Information Communication and Management 2011(ICICM 2011), Singapore, October 14-15 , 2011.

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 แนวปฏิบัติสำหรับข้อกำหนดความต้องการของซอฟต์แวร์ (IEEE std. 830-1998 Recommended Practice for Software Requirements Specifications)

มาตรฐานไอทีริปเปิลอี 830-1998 [1] บรรจุนิยามปฏิบัติที่แนะนำสำหรับการกำหนดความต้องการของซอฟต์แวร์ มีส่วนประกอบสำคัญของข้อกำหนดความต้องการของซอฟต์แวร์ ซึ่งสามารถแบ่งเป็นหัวข้อและส่วนประกอบตามตารางที่ 1 มีข้อชี้แนะในการจัดทำในหัวข้อต่างๆ เอาไว้ด้วย

ตารางที่ 1 หัวข้อข้อกำหนดความต้องการซอฟต์แวร์ตามมาตรฐานไอทีริปเปิลอี 830-1998

Chapter / Section	Topic
1	บทนำ (Introduction)
1.1	จุดประสงค์ (Purpose)
1.2	ขอบเขต (Scope)
1.3	คำจำกัดความ (Definitions)
1.4	รหัสนิยม (Acronyms)
1.5	คำย่อ (Abbreviations)
1.6	เอกสารอ้างอิง (References)
1.7	ภาพรวม (Overview)
2	คำอธิบายโดยรวม (Overall description)
2.1	แนวคิดหรือภาพโดยรวมของซอฟต์แวร์ (Product perspective)
2.1.1	การต่อประสานกับผู้ใช้ (User interface)
2.1.2	การต่อประสานกับฮาร์ดแวร์ (Hardware interfaces)
2.1.3	การต่อประสานกับซอฟต์แวร์ (Software interfaces)
2.1.4	การต่อประสานกับระบบสื่อสาร (Communications interfaces)
2.1.5	ข้อจำกัดของหน่วยความจำ (Memory constraints)
2.1.6	การดำเนินการ (Operations)
2.1.7	ความต้องการด้านการดัดแปลง (Site adaptation requirements)
2.2	หน้าที่ผลิตภัณฑ์ (Product functions)
2.3	ลักษณะของผู้ใช้ (User characteristics)
2.4	ข้อจำกัด (Constraints)

2.5	ข้อสันนิษฐานและการขึ้นต่อกัน (Assumptions and dependencies)
2.6	คุณสมบัติในการเตรียมการ (Qualification provisions)
2.7	ความสามารถในการตามรอยความต้องการ (Requirements Traceability)
2.8	การแบ่งสัดส่วนความต้องการ (Apportioning of requirements)
3	ความต้องการที่เฉพาะเจาะจง (Specific requirements)
3.1	การต่อประสานภายนอก (External interfaces)
3.2	ความต้องการด้านหน้าที่ (Functional requirements)
3.3	ความต้องการด้านประสิทธิภาพ (Performance requirements)
3.4	ข้อจำกัดในการออกแบบ (Design constraints)
3.4.1	มาตรฐานที่มีใช้ในองค์กร (Standards compliance)
3.4.2	ข้อจำกัดในการใช้งานซอฟต์แวร์ (Implementation constraints)
3.5	ลักษณะประจำ (Software system attributes)
3.5.1	ความเชื่อถือได้ (Reliability)
3.5.2	สภาพพร้อมใช้งาน (Availability)
3.5.3	ความมั่นคง (Security)
3.5.4	ความสามารถในการบำรุงรักษา (Maintainability)
3.5.5	ความสามารถในการใช้ได้หลายระบบ (Portability)
3.6	ความต้องการอื่น ๆ (Other requirements)
3.6.1	ความต้องการด้านฐานข้อมูลเชิงตรรกะ (Logical database requirements)
3.6.2	ความต้องการด้านการดำเนินการ (Operational requirements)
3.6.3	ความต้องการด้านการใช้ทรัพยากรฮาร์ดแวร์คอมพิวเตอร์ให้เกิดประโยชน์สูงสุด (Computer hardware resource utilization requirements)
3.6.4	ความต้องการด้านระบบสื่อสารคอมพิวเตอร์ (Computer communication requirements)
3.6.5	ปัจจัยที่มีผลต่อคุณภาพของซอฟต์แวร์ (Software quality factors)
3.6.6	ความต้องการด้านการเตรียมขั้นตอนต่าง ๆ ที่ใช้ในการพัฒนาซอฟต์แวร์ (Logistics-related requirements)
3.6.7	ความต้องการด้านความปลอดภัย (Safety requirements)
3.6.8	ความต้องการด้านการทวนสอบ (Verification requirements)
3.6.9	ความต้องการด้านการยอมรับ (Acceptance requirements)
3.6.10	ความต้องการด้านเอกสาร (Documentation requirements)
3.6.11	ความต้องการด้านบรรจุภัณฑ์ซอฟต์แวร์ (Packaging requirements)
3.6.12	ความต้องการด้านสภาพแวดล้อม (Environment requirements)
3.6.13	ความต้องการด้านการฝึกอบรมบุคคลที่เกี่ยวข้อง (Training-related requirements)

4	ข้อมูลสนับสนุน (Supporting information)
4.1	บันทึก (Notes)
4.2	ภาคผนวก (Appendixes)
4.3	ดัชนี (Index)

2.1.2 แนวทางในการออกแบบกรณีทดสอบ (Test Cases Design Approaches) [2]

การทดสอบเป็นกระบวนการของการประมวลผลรหัสคำสั่ง หรือ ประเมินระบบด้วยคนหรือเครื่องมือ ว่าเป็นไปตามความต้องการที่กำหนดหรือไม่ จากนั้นก็ให้ค่าของความแตกต่างของค่าที่คาดหวังกับค่าที่เกิดขึ้นจริง

กรณีทดสอบ (test case) ที่เราจัดทำขึ้นเพื่อไว้สำหรับใช้ในการทดสอบระบบซอฟต์แวร์ ประกอบด้วย

- 1) ข้อมูลนำเข้า (input) มีสองประเภทคือ เงื่อนไขก่อนการทำงาน (precondition) และ คำนำเข้าจริง (Actual Input)
- 2) ผลลัพธ์ที่คาดหวัง (expected output) มีสองประเภทคือ เงื่อนไขหลังการทำงาน (postcondition) และค่านำออกจริง (actual output)

กรณีทดสอบ จะต้องประกอบไปด้วย

- รหัสกรณีทดสอบ (Test Case ID) ปกติแล้วจะตั้งเป็นลำดับตัวเลข เช่น UAT001, UAT002 เป็นต้น
- ชื่อกรณีทดสอบ (Test Case Name) โดยปกติแล้วจะตั้งชื่อให้สื่อ เช่น การ login เข้าระบบ
- คำอธิบาย (Purpose) เป็นการอธิบายกรณีต่างๆ ว่าต้องการ Test กรณีใดบ้าง
- เงื่อนไขก่อนการทำงาน (Precondition) เป็นการอธิบายเงื่อนไขก่อนการทดสอบระบบ ว่าต้องทำกรณีใดก่อน ถึงจะดำเนินการทดสอบได้
- ข้อมูลนำเข้า (Input) เป็นข้อมูลที่จะใช้ทดสอบในกรณีต่างๆ
- ผลลัพธ์ที่คาดหวัง (Expected Result) เป็นผลลัพธ์ที่คาดหวังว่าจะได้ออกมา ซึ่งจะต้องตรงกับข้อกำหนดความต้องการซอฟต์แวร์
- วิธีในการทดสอบ
- ผลลัพธ์ที่ได้จริง (Actual Result) เป็นผลลัพธ์ที่ได้จากกรณีทดสอบ

- ประวัติของกรณีทดสอบ (Execution History) เพื่อเก็บเป็นประวัติของกรณีทดสอบ โดยจะประกอบด้วย
 - วันที่ทดสอบ (Date) เพื่อระบุวันที่ดำเนินการทดสอบ
 - ผลลัพธ์ของการทดสอบ (Test Result) เพื่อระบุว่ากรณีทดสอบข้อนี้ผ่าน หรือไม่ผ่าน
 - เวอร์ชัน (Version) เพื่อระบุเวอร์ชันของกรณีทดสอบ
 - ทดสอบโดย (Run By) เพื่อระบุของผู้ทดสอบระบบ

2.1.2.1 Functional (Black-Box) Testing

การทดสอบกรณีนี้ เป็นการทดสอบโดยเปรียบเสมือนว่าโปรแกรมหรือระบบซอฟต์แวร์ที่จะทำการทดสอบเป็นกล่องดำ ที่ไม่รู้เลยว่าข้างใน ซอฟต์แวร์นั้นมีอะไรบ้าง คือ ไม่สนใจโครงสร้างคำสั่งภายในรหัสต้นฉบับภายใน มองเพียงว่า อะไรเป็นข้อมูลนำเข้าที่ใส่เข้ามา เพื่อที่ทำให้ระบบซอฟต์แวร์นั้นถูกทำงาน และมีอะไรบ้างเป็นผลลัพธ์ที่สอดคล้องกัน สนใจข้อกำหนดความต้องการซอฟต์แวร์ การออกแบบกรณีทดสอบจะอยู่บนพื้นฐานของข้อกำหนดความต้องการซอฟต์แวร์

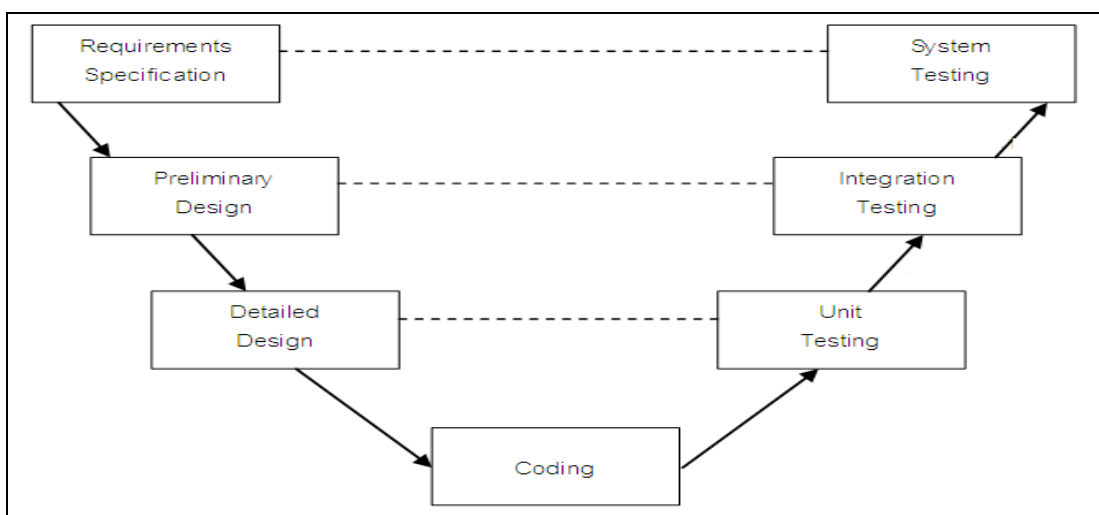
2.1.2.2 Structural (White-Box, Glass-Box, Clear-Box) Testing

การทดสอบกรณีนี้เพื่อดูโครงสร้างของโปรแกรม หรือทางเดินในโปรแกรม ซึ่งจำเป็นต้องรู้โครงสร้างภายในของระบบซอฟต์แวร์ ดังนั้น การออกแบบกรณีทดสอบจะขึ้นกับรหัสต้นฉบับ (Source code) และนำรหัสต้นฉบับมาแปลงเป็นกราฟ จากนั้นทำการสร้างกรณีทดสอบเพื่อที่จะให้ รหัสต้นฉบับนั้น ถูกทดสอบ ดังนั้นการออกแบบกรณีทดสอบจะอยู่บนพื้นฐานของ Coverage criteria (Statement Coverage, Branch Coverage, Path Coverage)

2.1.3 ระดับของการทดสอบ (Level Of Testing)

เป็นการอธิบายถึง ระดับของการทดสอบ ลักษณะคล้ายกับแบบจำลองน้ำตก (Waterfall model) โดยการนำเอา วัฏจักรการพัฒนาซอฟต์แวร์ (Software Development Life Cycle) และวัฏจักรการทดสอบซอฟต์แวร์ (Testing Life Cycle) มารวมเชื่อมความสัมพันธ์กัน เพื่อเป็นแนวทางว่าควรจะใช้การทดสอบแบบใดให้เหมาะสมกับระบบซอฟต์แวร์ และความต้องการของผู้ใช้งาน ในแต่ละขั้นตอนดังแสดงในภาพที่ 1

งานวิจัยนี้อยู่ในขั้นตอนของข้อกำหนดความต้องการซอฟต์แวร์ ซึ่งสามารถสร้างกรณีทดสอบสำหรับการทดสอบระบบ (System Testing) หรือการทดสอบการยอมรับระบบของผู้ใช้ (User Acceptance Test: UAT) เพื่อทดสอบหน้าที่การทำงานรวมทั้งหมดของระบบซอฟต์แวร์ (ซึ่งกระทำโดยผู้ใช้มีส่วนร่วมในการทดสอบระบบ)



ภาพที่ 1 ระดับของการทดสอบ

2.1.4 XMLเอ็กซ์เอ็มแอล (XML: Extensible Markup Language) [3]

เป็นภาษาที่ถูกออกแบบมาเพื่อใช้สำหรับอธิบายข้อมูล การจัดเก็บ และจัดส่งข้อมูลบน เวิลด์ไวด์เว็บ ซึ่งมีพื้นฐานที่พัฒนามาจากภาษาเอสจีเอ็มแอล เอกสารเอ็กซ์เอ็มแอลจะมีการ อธิบายข้อมูลในส่วนรูปแบบที่เป็นลำดับชั้น (Hierarchy) เหมือนต้นไม้ (Tree) ทำให้สามารถสร้าง โปรแกรมประยุกต์เพื่อทำการประมวลผลเอกสารและต่อประสาน (Binding) กับเอกสารเอ็กซ์เอ็ม แอลในการแสดงผลบนเว็บเบราว์เซอร์ได้อย่างอิสระ

โดยทั่วไปเอ็กซ์เอ็มแอลจะประกอบด้วยส่วนหลักคือ

- ตัวเอกสารเอ็กซ์เอ็มแอลซึ่งเป็นโครงสร้างทางตรรกะ (Logical Structure) อธิบายคุณลักษณะต่างๆ ของข้อมูลที่บรรจุอยู่ในเอกสารในรูปแบบโครงสร้าง ลำดับชั้น
- คำอธิบายชนิดของเอกสารหรือที่เรียกว่า ดีทีดี (DTD: Document Type Declaration) ทำหน้าที่ในการกำหนดไวยากรณ์ของเอกสารเอ็กซ์เอ็มแอลที่มี รูปแบบถูกต้อง (Well-formed document) และสามารถทำให้เป็นเอกสารที่ ถูกต้องสมบูรณ์ (Valid XML Document) ซึ่งเอกสารทั้งสองแบบนี้จะมีข้อบังคับ ที่แตกต่างกัน

2.1.5 ยูเอ็มแอล (Unified Modeling Language)[4]

ยูเอ็มแอลเป็นภาษาที่ใช้ในการอธิบายโครงสร้าง และการทำงานของระบบซอฟต์แวร์ ซึ่ง จะสามารถอธิบายแนวคิดเชิงวัตถุได้อย่างมีประสิทธิภาพ แผนภาพของยูเอ็มแอล ประกอบด้วย 9 แผนภาพหลัก แต่จะขออธิบายเฉพาะแผนภาพที่ใช้ในงานวิจัยนี้ ได้แก่

2.1.5.1 แผนภาพยูสเคส (Use Case Diagram)

ยูสเคส เป็นสิ่งที่แสดงถึงกิจกรรมที่ผู้ใช้สามารถทำงานร่วมกับระบบได้ โดยยูสเคสจัดเป็นแบบจำลองเชิงตรรกะ ที่อธิบายกิจกรรมของระบบ ซึ่งยังไม่มีกระบวนการว่ากิจกรรมต่างๆ ของระบบนั้นจะถูกพัฒนาขึ้นอย่างไร แผนภาพยูสเคส จะแสดงให้เห็นถึงฟังก์ชันการทำงานหลักของระบบ รวมถึงผู้ใช้ และผู้ที่มีความเกี่ยวข้องกับระบบ โดยยูสเคสแสดงหน้าที่การทำงานของระบบ ส่วนประกอบของแผนภาพยูสเคสมีดังนี้

ผู้กระทำ หรือระบบที่เกี่ยวข้องกับระบบ (Actor) ซึ่งจะมีการระบุบทบาทของผู้กระทำ หรือระบบที่เกี่ยวข้องกับระบบ ภาพที่ 2 แสดงสัญลักษณ์ที่ใช้แทนผู้ใช้งานระบบ



Actor

ภาพที่ 2 ผู้ใช้งานระบบ

ยูสเคส (Use Case) แสดงหน้าที่การทำงานของระบบ และระบุฟังก์ชันการทำงานที่เกิดขึ้นในระบบ โดยอาจเป็นหน้าที่การทำงานย่อย หรือคลาส ใช้วงรีเป็นสัญลักษณ์แทนยูสเคส ดังแสดงในภาพที่ 3



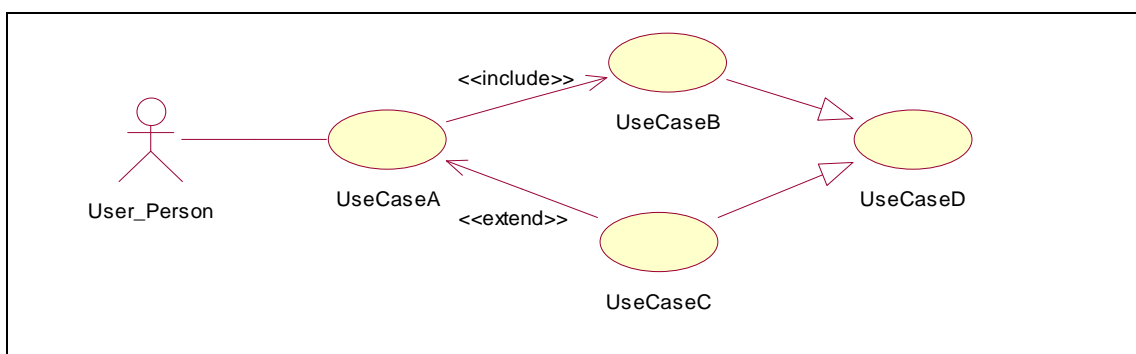
Use Case

ภาพที่ 3 ยูสเคส

ความสัมพันธ์ของยูสเคส (Use Case Relationships) ความสัมพันธ์ของยูสเคสเกิดขึ้นเมื่อยูสเคสมีการเรียกหรือถูกเรียกใช้ โดยมากแล้วจะเกิดความสัมพันธ์แบบระหว่างผู้กระทำ หรือระบบที่เกี่ยวข้องกับระบบ กับยูสเคส และความสัมพันธ์แบบรวม รูปแบบของความสัมพันธ์ แสดงในภาพที่ 4

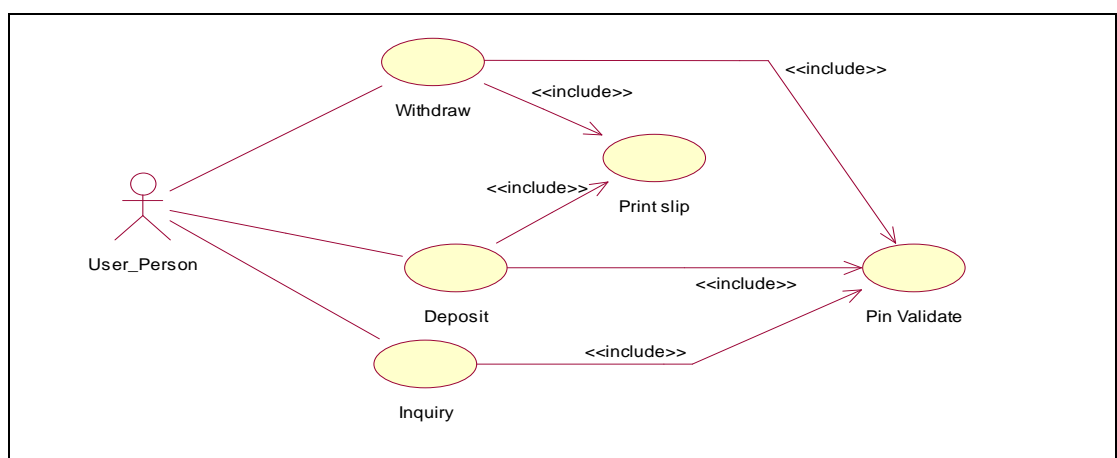
- ความสัมพันธ์ (Association) แสดงความสัมพันธ์ระหว่างผู้กระทำ หรือระบบที่เกี่ยวข้องกับระบบ กับยูสเคส แผนภาพยูสเคสใช้เส้นตรง
- แบบรวม (Include) แสดงความสัมพันธ์ระหว่างยูสเคส ถ้ามีเหตุการณ์เกิดขึ้นกับยูสเคสหนึ่งเรียกใช้งานอีกยูสเคสหนึ่ง ใช้เส้นตรงที่มีลูกศรชี้ไปยังยูสเคสที่ถูกเรียกใช้และมีคำว่า <<include>> กำกับบนเส้นลูกศร ตัวอย่างเช่นยูสเคสบี ถูกเรียกใช้โดยยูสเคสเอ

- แบบขยาย (Extend) แสดงความสัมพันธ์ระหว่างยูสเคส ถ้ามีเหตุการณ์เกิดขึ้นกับยูสเคสหนึ่งไม่สามารถทำงานได้ตามปกติจึงมียูสเคสอีกอันทำงานแทนเพื่อเป็นทางเลือก แผนภาพยูสเคสใช้เส้นตรงที่มีหัวลูกศรชี้จากยูสเคสที่ถูกขยายและมีคำว่า <<extends>> กำกับบนเส้นลูกศร ตัวอย่างเช่นยูสเคสเอ มีความสัมพันธ์แบบขยายยูสเคสซี
- สืบทอด (Generalization) แสดงความสัมพันธ์ระหว่างยูสเคส ที่มีความสัมพันธ์แบบรับทอดคุณสมบัติ ใช้เส้นตรงที่มีหัวลูกศรเป็นรูปสามเหลี่ยมโป่ง ชี้ไปยังยูสเคสต้นแบบ ตัวอย่างเช่นยูสเคสบีและยูสเคสซี สืบทอดจากยูสเคสดี



ภาพที่ 4 ความสัมพันธ์ของยูสเคสแบบต่างๆ

ในงานวิจัยนี้จะใช้แผนภาพยูสเคสซึ่งใช้ในการอธิบายหน้าที่การทำงานของระบบ มาแสดงหน้าที่ทั้งหมดของระบบ เพื่อแสดงถึงตัวดำเนินการของระบบ และความสัมพันธ์ระหว่างหน้าที่ จะถูกนำมาใช้สร้างเป็นยูสเคสซีนารีโอ (Use Case Scenario) ของระบบ ดังแสดงตัวอย่างในภาพที่ 5 เป็นเค้าโครงการทำงานของการถามยอดบัญชี (Inquiry) ซึ่งจะใช้ยูสเคสสอบถามยอดเงิน (Inquiry) โดยมีความสัมพันธ์แบบรวมกับยูสเคสตรวจสอบรหัส (Pin Validate)



ภาพที่ 5 ตัวอย่างยูสเคสซีนารีโอของระบบ

2.1.5.2 คำอธิบายยูสเคส (Use Case Description)

คำอธิบายยูสเคส จะอธิบายฟังก์ชันการทำงานพื้นฐานของระบบซึ่งจะอธิบายว่าผู้ใช้จะสามารถทำอะไรได้บ้าง และระบบจะมีการตอบสนองการทำงานอย่างไร โดยคำอธิบายยูสเคสจะประกอบไปด้วยโครงสร้างข้อมูลต่างๆ ซึ่งสามารถแสดงได้ดังตารางที่ 2

ตารางที่ 2 ตัวอย่างเค้าโครงคำอธิบายยูสเคส

ID:	Use case type:	Importance level:
Use Case Name:		
Primary actor:		
Stakeholders and interests:		
Brief description:		
Trigger:		
Type:		
Relationships:		
Association:		
Include:		
Extend:		
Generalization:		
Normal flow of Events:		
Subflows:		
Alternate/Exceptional flows:		

รายละเอียดยูสเคสรูปแบบตารางหนึ่งสดมภ์ดังแสดงในตารางที่ 2 สามารถอธิบายได้ดังนี้

- Use Case Name: ชื่อยูสเคส โดยชื่อของยูสเคสควรเป็นกลุ่มคำกริยา เช่น ตรวจสอบสิทธิการเข้าใช้ระบบ
- ID: รหัสยูสเคส การกำหนดรหัสยูสเคส จะเป็นการกำหนดลักษณะเฉพาะของยูสเคส ที่ง่ายต่อการอ้างอิง รวมถึงสามารถตามรอยกลับไปยังความต้องการที่เกี่ยวข้องกันได้
- Importance level: ระดับความสำคัญของยูสเคส เมื่อเปรียบเทียบกับยูสเคสอื่นในระบบเดียวกัน มีไว้เพื่อช่วยในการจัดลำดับความสำคัญเร่งด่วนของงานที่ต้องทำ เช่น สูง/ปานกลาง/ต่ำ

- Primary actor: เป็นข้อความแสดงชื่อบุคคล หรือระบบที่จะก่อให้เกิดการดำเนินงานตามยูสเคส
- Use case type: ประเภทของยูสเคส โดยระบุว่า “ภาพรวม” หรือ “เชิงละเอียด” (Overview/Detail) และ “พื้นฐาน” หรือ “จริง” (Essential/Real)
- Stakeholders and interests: เป็นข้อความแสดงชื่อบุคคล หรือระบบที่ได้รับผลประโยชน์จากการทำงานของยูสเคส ได้แก่ ผู้กระทำหลัก หรือ อาจเป็นผู้ที่เกี่ยวข้องกับยูสเคส
- Brief description: เป็นข้อความสาระสำคัญของยูสเคส ซึ่งโดยทั่วไปจะเป็นการอธิบายสั้นๆ เพื่อให้เข้าใจว่ายูสเคสนี้ให้บริการใด แก่ผู้ใดบ้าง และจะมีการใช้งานเมื่อใด
- Trigger: เป็นข้อความแสดงเหตุการณ์ที่ก่อให้เกิดการดำเนินงานตามยูสเคส สามารถจำแนกได้เป็น สองประเภท
 1. External Trigger : คือ สิ่งกระตุ้นภายนอก เช่น ลูกค้ำมีความต้องการสั่งซื้อสินค้า
 2. Temporal Trigger : คือ สิ่งกระตุ้นตามเวลา เช่น เมื่อเลยกำหนดระยะเวลาการยืมหนังสือ ระบบจะต้องจัดทำระบบเตือนแก่ผู้ยืม และเจ้าหน้าที่ห้องสมุด เป็นต้น
- Relationships: แสดงความสัมพันธ์ ว่ายูสเคสมีความเกี่ยวข้องกับผู้กระทำหลัก ผู้ใช้ และยูสเคสอื่นๆอย่างไร แบ่งออกเป็น 4 รูปแบบ คือ
 1. Association: เป็นการระบุความสัมพันธ์ระหว่างยูสเคส และผู้กระทำแบบเกี่ยวเนื่อง ในที่นี้จะหมายถึงบทบาทของผู้ใช้เมื่อใช้ระบบ
 2. Include: เป็นการระบุความสัมพันธ์แบบรวม ระหว่างยูสเคส ว่ายูสเคสใดมีฟังก์ชันการทำงานที่จะถูกรวมเข้าไว้กับยูสเคสนั้นๆ เสมอ
 3. Extend: เป็นการระบุความสัมพันธ์แบบขยายระหว่างยูสเคส ว่ายูสเคสใดมีฟังก์ชันการทำงานที่สามารถนำมารวมเข้าไว้เป็นทางเลือก (Optional) กับยูสเคสนั้นๆ
 4. Generalization: เป็นการระบุความสัมพันธ์แบบเอนเนอรัลไลเซชัน คุณสมบัติ ความสัมพันธ์ระหว่างยูสเคส กับยูสเคสที่มีการรับทอดคุณสมบัติ (Inheritance) จากยูสเคสนั้น

- Normal flow of events: เป็นข้อความแสดงลำดับขั้นตอนการทำงานของแต่ละยูสเคส สามารถจำแนกได้เป็น 3 ประเภท ดังนี้
 1. Normal Flow : เป็นข้อความแสดงขั้นตอนการทำงานปกติของ ยูสเคส
 2. Subflows: เป็นข้อความแสดงขั้นตอนการทำงานที่แตกแยกย่อยลงมา จากขั้นตอนการทำงานปกติของ ยูสเคส เพื่อไม่ให้ส่วนของขั้นตอนปกติของยูสเคสซับซ้อนเกินไป
 3. Alternate/Exceptional flows: เป็นข้อความแสดงเงื่อนไขที่ทำให้เกิดเหตุการณ์อื่น ที่อาจเกิดขึ้นได้ในบางกรณี

โดยงานวิจัยนี้จะใช้คำอธิบายยูสเคส ในระดับยูสเคสจริง (Real Use Case) เนื่องจากให้รายละเอียดการทำงานของผู้ใช้ได้ละเอียดกว่าแบบอื่นๆ และมีการเพิ่มข้อมูลบางส่วนลงไปเพื่อความสามารถปรับใช้งานให้เหมาะสมกับการสร้างกรณีทดสอบ เช่น Precondition (เงื่อนไขก่อนหน้า) เพื่อใช้บอกเหตุการณ์ที่ต้องเกิดขึ้นก่อนจะเกิดเหตุการณ์ในยูสเคสจะสามารถเกิดขึ้นได้ Postcondition (เงื่อนไขตามหลัง) เพื่อใช้บอกเหตุการณ์ที่เกิดขึ้นหลังจากทำยูสเคสสำเร็จ

2.2 งานวิจัยที่เกี่ยวข้อง

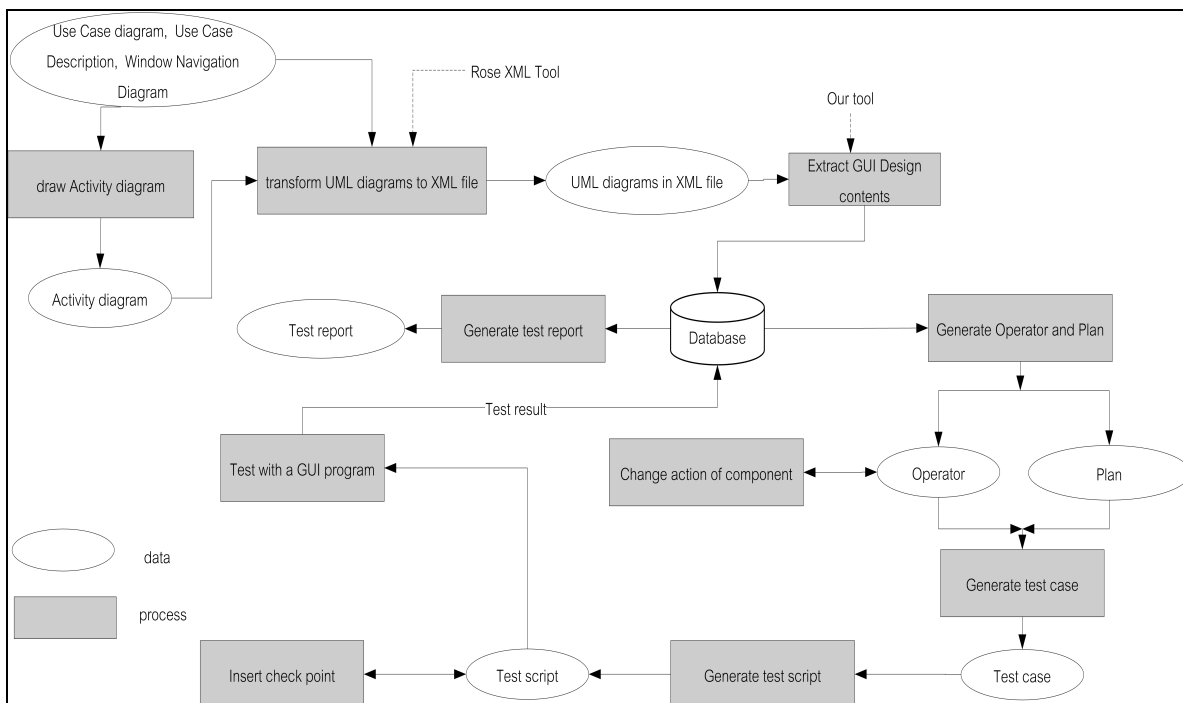
2.2.1 การสร้างกรณีทดสอบจิวไอจากยูเอ็มแอล [5]

บทความวิจัยนี้ได้นำเสนอแนวทางการสร้างกรณีทดสอบ ในระดับของงานซึ่งมีแนวคิดคล้ายคลึงกับแผนงานปัญญาประดิษฐ์ โดยชุดกรณีทดสอบจะครอบคลุมทุกฟังก์ชันหน้าที่การทำงานของระบบซอฟต์แวร์ ดังแสดงในเอกสารการออกแบบและมีปริมาณกรณีทดสอบไม่มากจนเกินไป นอกจากนี้ยังได้พัฒนาวิธีการและเครื่องมือสำหรับสร้างกรณีทดสอบส่วนต่อประสานกราฟิกกับผู้ใช้จากแผนภาพยูเอ็มแอล โดยนำแผนภาพยูสเคส คำอธิบายยูสเคส และแผนภาพกิจกรรมมาประยุกต์ให้สามารถสร้างกรณีทดสอบ ของส่วนต่อประสานกราฟิกกับผู้ใช้ และนำข้อกำหนดของซอฟต์แวร์ (Software Specification) เป็นตัวบ่งชี้ความครบถ้วนถูกต้องของหน้าที่การทำงานระบบ

เพื่อให้สามารถสร้างกรณีทดสอบได้ทันทีหลังจากทราบข้อกำหนดซอฟต์แวร์ วิธีการดังกล่าวจะช่วยประหยัดทรัพยากรที่ใช้ในขั้นตอนการทดสอบ รวมทั้งสนับสนุนการนำกรณีทดสอบกลับมาใช้งานได้ใหม่ โดยภาพรวมของบทความวิจัยแสดงดังภาพที่ 6

งานวิจัยดังกล่าว ได้แสดงให้เห็นถึงความสามารถในการนำเอาแผนภาพยูสเคส แผนภาพกิจกรรม และคำอธิบายยูสเคสมาประยุกต์ใช้ในการสร้างกรณีทดสอบส่วนต่อประสานกราฟิกกับ

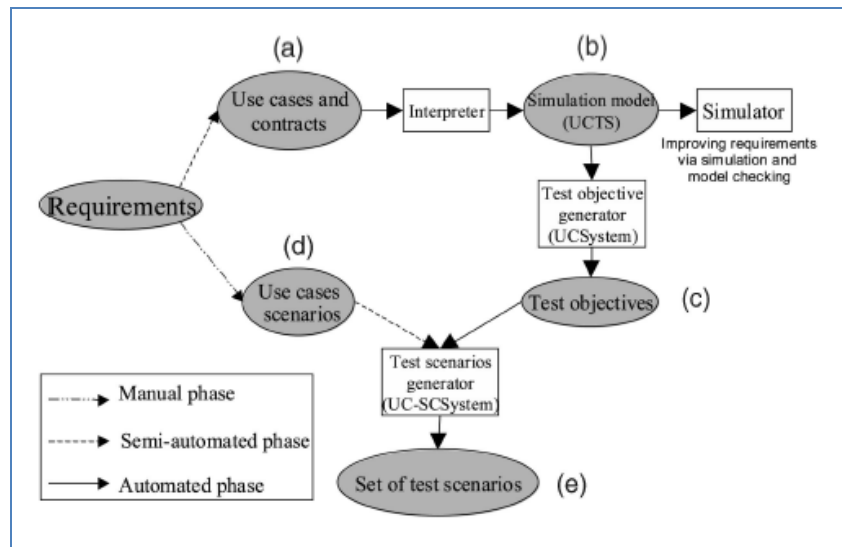
ผู้ใช้ ซึ่งผู้วิจัยได้ใช้เป็นแนวทางในการนำเสนอวิธีการสร้างกรณีทดสอบ จากเอกสารข้อกำหนดความต้องการซอฟต์แวร์ โดยใช้ยูสเคส



ภาพที่ 6 ภาพรวมงานวิจัยการสร้างกรณีทดสอบจึยไ้จากยูเอ็มแอล

2.2.2 Automatic Test Generation: A Use Case Driven Approach[6]

บทความวิจัยนี้ได้นำเสนอแนวคิด ของความสำคัญของยูสเคส ที่มีต่อการทดสอบรวมระบบซอฟต์แวร์เข้าด้วยกันและการทำการทดสอบระบบซอฟต์แวร์โดยอัตโนมัติ จึงได้มีแนวทางที่จะสร้างสินาริโอของกรณีทดสอบรวมระบบซอฟต์แวร์ (System Test Scenarios) เข้าด้วยกันโดยอัตโนมัติ เพื่อให้ใช้ในทดสอบระบบซอฟต์แวร์ฝังตัว โดยเริ่มจากข้อกำหนดความต้องการซอฟต์แวร์ที่อยู่ในภาษารูปนัย บนพื้นฐานของยูสเคส และทำการสร้าง สังเคราะห์กรณีทดสอบโดยอัตโนมัติ โดยเป้าหมายของการสร้างกรณีทดสอบจะครอบคลุม Statement Coverage ที่มีอยู่ในข้อกำหนดความต้องการซอฟต์แวร์ โดยภาพรวมของบทความวิจัยแสดงดังภาพที่ 7 จากผลสรุปของกรณีศึกษาและข้อสังเกตดังกล่าว งานวิจัยนี้จึงให้ความสนใจไปที่การแปลงลำดับของยูสเคสไปยังกลุ่มของลำดับการทดสอบ โดยผ่านการสร้างลำดับกรณีทดสอบอัตโนมัติ



ภาพที่ 7 ภาพรวมงานวิจัย

2.2.3 On Generating EFSM Models from Use Cases [7]

บทความวิจัยนี้ได้นำเสนอแนวคิดในการนำเข้ากลุ่มข้อมูลของยูสเคส เพื่อทำการสร้างแบบจำลองให้อยู่ในรูปแบบของ Extended finite state machine (EFSM) แนวคิดดังกล่าวได้รวมถึงรายละเอียดที่เฉพาะเจาะจงของยูสเคส ได้แก่ Included use cases, Extension points, Conditional statements แสดงดังภาพที่ 8, 9, 10

Definition (Use Case) A use case $u \in U$ is a tuple $(a, \Sigma_{Main}, E, X, P)$ where $a \in \alpha$ is the actor, Σ_{Main} is a statement block describing the actor system interactions, E is a set of exceptions, X is a set of extension points and P is the pre-condition for the use case

ภาพที่ 8 นิยาม Use Case ตามแนวคิดการวิจัย

Definition An extended finite state machine (EFSM) can be uniquely defined by the tuple $M = \{Q, I, O, D, T\}$, where Q is a set of control states, I is a set of inputs, O is a set of outputs, D is a n dimensional space $D_1 \times \dots \times D_n$, T is a transition relation, $T: Q \times D \times I \rightarrow Q \times D \times O$.

ภาพที่ 9 นิยาม Extended finite state machin ตามแนวคิดการวิจัย

Algorithm: Normalizer. **Input:** $\mu = \{\Delta, \beta\}$; **Output:** $N = \{I, O, D, T\}$.

Steps :

Initialize I, O, T and D to $\{\}$

1. From Δ populate a set τ defining the types of elements in Δ .
2. For each use case u in β populate a set of guarded statement blocks.
3. Flatten each guarded statement block derive a set of guarded scenarios $R = \{(g, \Sigma)\}$.
4. For each guarded statement block in R
 - * Derive $\mathbf{x} \subset \delta$ such that $g(\mathbf{x}) = \text{True}$
 - * For each statement in Σ determine the statement type T
 - * If T is **input** populate set of input parameters i .
 - * else if T is **output** populate set of output parameters \mathbf{o}
 - * else if T is **update** compute $\mathbf{F} = \mathbf{F}.f$, where f is the update.
 - * $I = I \cup i, O = O \cup \mathbf{o}, T = T \cup ((\mathbf{x}, i), (\mathbf{F}(\mathbf{x}), \mathbf{o})), D = D \cup (\mathbf{x}, \mathbf{F}(\mathbf{x}))$

ภาพที่ 10 อัลกอริทึมเพื่อสนับสนุนแนวคิดการวิจัย

บทที่ 3

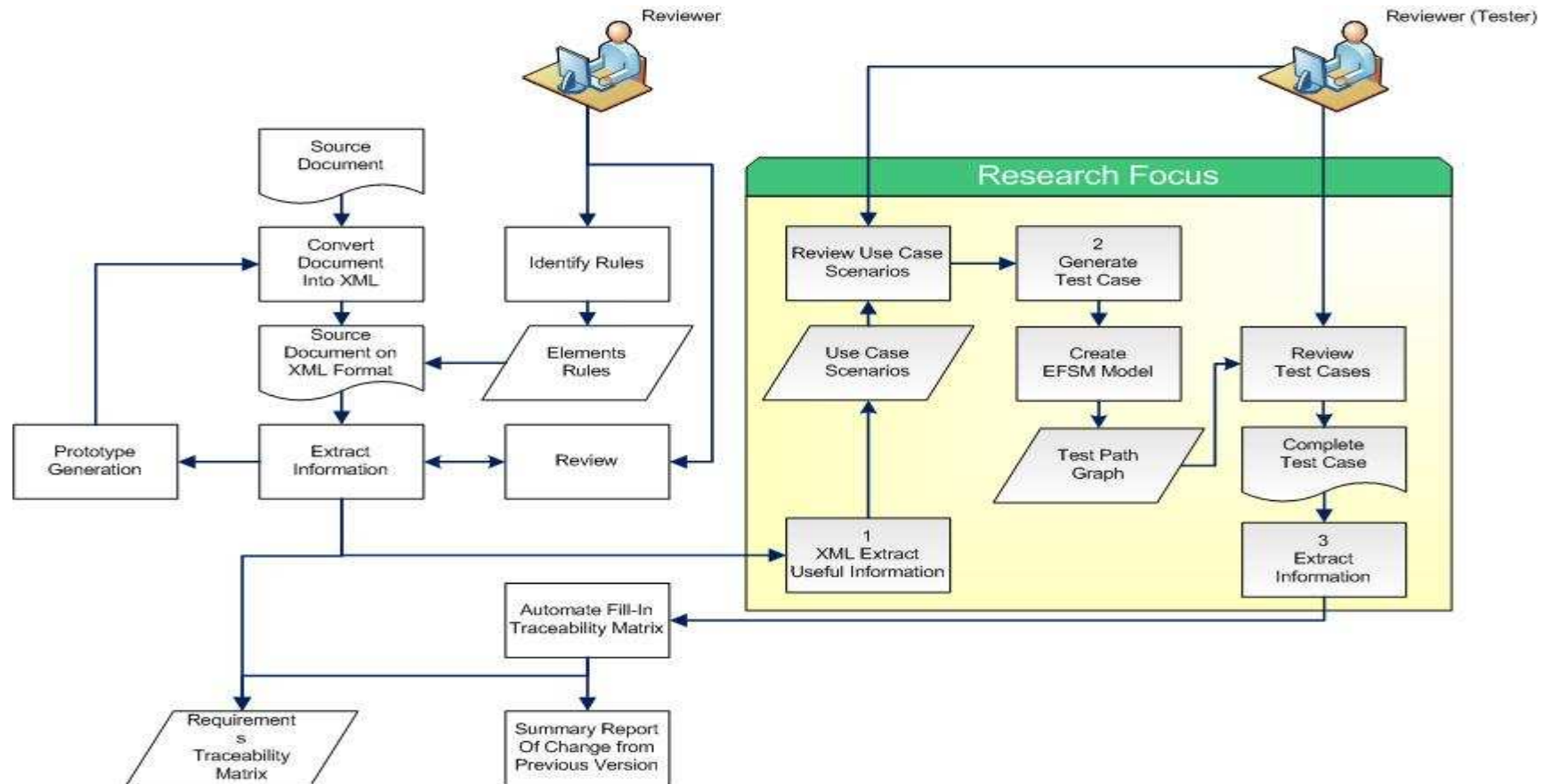
การออกแบบสถาปัตยกรรมระบบ

งานวิจัยนี้เป็นระบบย่อยของระบบ Requirements Harvester— RH ซึ่งเป็นระบบที่บูรณาการงานควบคุมคุณภาพ (Quality Control) กับความสามารถจัดการความต้องการเข้าด้วยกัน โดยงานควบคุมคุณภาพ หมายถึง การทวนสอบ (Verification) และการยืนยันสอบใช้ได้จริง (Validation) ส่วนความสามารถจัดการความต้องการ หมายถึง เครื่องมือคือเมตริกซ์ความสามารถตามรอยความต้องการ

ปัจจุบัน RH ประกอบด้วยระบบย่อย 4 ระบบ กล่าวคือ 1) เมตริกซ์ความสามารถตามรอยความต้องการ [8] 2) ระบบนำเข้าข้อมูลอัตโนมัติ [9] 3) ระบบสร้างกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติจากเอกสารข้อกำหนดความต้องการ (งานวิจัยนี้) และ 4) การสร้างระบบต้นแบบ (prototype) อัตโนมัติจากเอกสารข้อกำหนดความต้องการ (หัวข้อวิทยานิพนธ์)

งานวิจัยนี้มุ่งเน้นการปรับปรุงกระบวนการซอฟต์แวร์ (Software Process Improvement) เพื่อเพิ่มประสิทธิภาพ ลดค่าใช้จ่าย ทรัพยากร เวลา และความพยายามของนักวิเคราะห์ระบบรวมทั้งผู้ทดสอบระบบ โดยงานวิจัยนี้ แบ่งออกเป็น 3 ส่วนหลัก ได้แก่ ส่วนที่ 1) นำเข้าข้อมูลจากเอกสารข้อกำหนดความต้องการซอฟต์แวร์ที่เป็นไฟล์ในรูปแบบเอ็กซ์เอ็มแอล ส่วนที่ 2) สร้างกรณีทดสอบ ตรวจสอบกรณีทดสอบจากคำอธิบายยูสเคส โดยใช้ Extend Finite State Machine และ ส่วนที่ 3) ส่งออกข้อมูลไปเมตริกซ์ความสามารถตามรอยในรูปแบบเอ็กซ์เอ็มแอล โดยมีลำดับขั้นตอนของการดำเนินงานแสดงดังภาพที่ 11 ดังนี้

3.1 วิเคราะห์โครงสร้างระบบ



ภาพที่ 11 ภาพรวมงานวิจัยที่น่าเสนอ

รายละเอียดขั้นตอนการทำงานในภาพที่ 11 สามารถอธิบายได้ดังนี้

1. นำเข้าข้อมูลจากเอกสารข้อกำหนดความต้องการซอฟต์แวร์ที่เป็นไฟล์ในรูปแบบเอกซ์เอ็มแอลแสดงดังภาพที่ 12 และทำการสร้างยูสเคสขึ้นมาวิธีโอทั้งหมดจากคำอธิบายยูสเคสที่ได้รับ โดยการแปลงข้อมูลคำอธิบายยูสเคสให้อยู่ในรูปแบบขึ้นมาวิธีโอ วิธีดำเนินการสร้างขึ้นมาวิธีโอแสดงดังตารางที่ 3 โดยผลลัพธ์ที่ได้จะได้ตารางยูสเคสขึ้นมาวิธีโอทั้งหมด เพื่อใช้เป็นข้อมูลในการสร้างกรณีทดสอบการยอมรับของผู้ใช้ แสดงดังตารางที่ 3

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
- <project id="25550108190823940" name="NPA SRS V3.0">
  <package id="0" name="User Requirements" />
  <package id="1" name="Product Feature" />
+ <package id="2" name="Functional Requirement">
+ <package id="3" name="Non-Functional Requirement">
- <package id="4" name="Use Case Description">
  <data id_relate="25550108190842740000" name="Alternate flows" table_num="1"></data>
  <data id_relate="255501081908426280002" name="Brief description" table_num="1">เป็นการกำหนด Security ในการเข้าถึงข้อมูลให้กับผู้ปฏิบัติงาน</data>
  <data id_relate="255501081908433420006" name="Importance level" table_num="1">High</data>
  <data id_relate="255501081908448170008" name="Normal flow of Events" table_num="1">1. ระบบแสดงรายชื่อ User\n2. User Admin เลือกชื่อ User ที่จะกำหนดสิทธิ์\n3. User Admin กำหนดสิทธิ์การเข้าถึงข้อมูลของผู้ปฏิบัติงาน โดยแบ่งกลุ่มดังนี้\n3.1 กลุ่มผู้ใช้ทั่วไปสามารถบันทึกและแก้ไขข้อมูลของตนเองได้ และสามารถดูข้อมูลทั้งหมดได้\n3.2 กลุ่มผู้บริหารสามารถเรียกดูข้อมูลได้ทั้งหมด และสามารถแก้ไขข้อมูลที่อยู่ภายใต้ความดูแลของตนเองได้ถึงหมด\n4. ระบบบันทึกข้อมูล</data>
  <data id_relate="255501081908441850009" name="Post-Condition" table_num="1">User สามารถกำหนด Security ในการเข้าถึงข้อมูลให้กับผู้ปฏิบัติงานได้สำเร็จ</data>
  <data id_relate="2555010819084422200010" name="Pre-Condition" table_num="1">มีรายชื่อ User ที่จะกำหนดสิทธิ์ในระบบ</data>
  <data id_relate="2555010819084411800011" name="Primary actor" table_num="1">AC02,AC03</data>
  <data id_relate="2555010819084493800012" name="Related Requirement" table_num="1">FR09</data>
  <data id_relate="2555010819084432300015" name="Use Case Name" table_num="1">UC1: Setup Security</data>
  <data id_relate="255501081908466100016" name="Use case type" table_num="1">Real Use Case</data>
</package>
<package id="5" name="User Acceptance Test" />
<package id="6" name="User" />
</project>
```

ภาพที่ 12 ตัวอย่างข้อมูลนำเข้าในรูปแบบเอกซ์เอ็มแอล

ตารางที่ 3 ตัวอย่างยูสเคสขึ้นมาวิธีโอที่ได้จากคำอธิบายยูสเคส

State No	Scenario No	Scenario Descriptions	A-Cond No	A-Cond descriptions	Scenario type
1	1.	User พิมพ์ User ID และ Password			
2	2	ระบบตรวจสอบความถูกต้องของ User ID และ Password			
3	3	ระบบจะให้ Roles แก่ผู้ใช้ระบบ ตามที่กำหนดไว้			
4	4	ผู้ใช้ User สามารถเข้าระบบได้			
5	2a1	ระบบไม่อนุญาตให้เข้าระบบ	2a.	ผู้ใช้ใส่ User ID ,Password ไม่ถูกต้อง	Alternate flows
6	2a2	ระบบจะให้ใส่อีกครั้ง และกลับไปทำข้อที่ 2	2a.	ผู้ใช้ใส่ User ID ,Password ไม่ถูกต้อง	Alternate flows

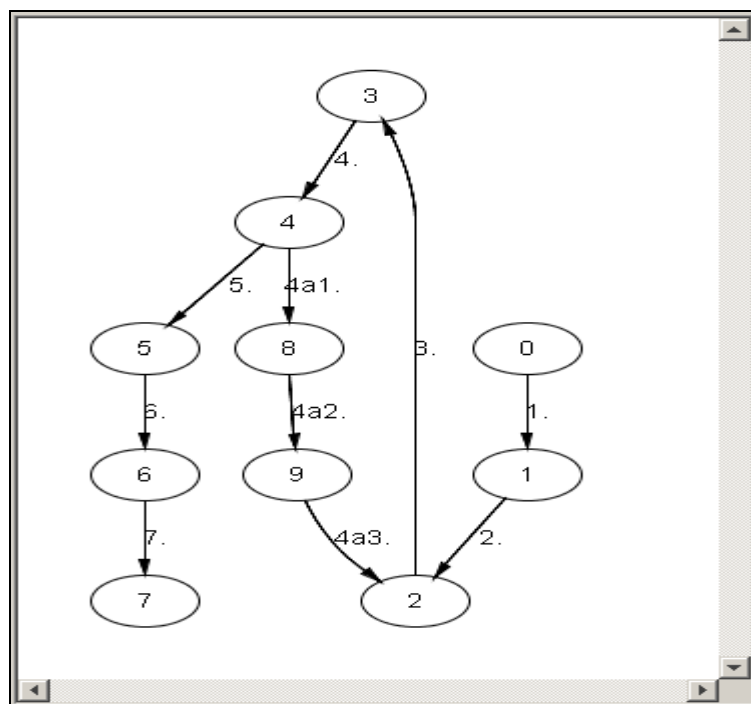
2.สร้างกรณีทดสอบการยอมรับของผู้ใช้จากยูสเคสซึ่งนารีโอที่ได้จากกระบวนการแรก โดยมีกระบวนการย่อย 2 ขั้นตอน ดังนี้

2.1 สร้าง EFSM Model เป็นขั้นตอนการนำข้อมูลยูสเคสซึ่งนารีโอมาทำการประมวลผลให้อยู่ในรูปแบบ Extended Finite State Machine(EFSM) เพื่อให้ข้อมูลในยูสเคสซึ่งนารีโออยู่ในรูปแบบ EFSM Model วิธีดำเนินการสร้าง Extended finite state machine (EFSM) แสดงดังตารางที่ 6

2.2 สร้างกรณีทดสอบการยอมรับของผู้ใช้ เป็นขั้นตอนการนำ EFSM Model มาประมวลผลในรูปแบบของกราฟ โดยวิธีดำเนินการสร้างกรณีทดสอบที่ครอบคลุม แสดงดังตารางที่ 7 และแสดงเส้นทางของกราฟ ซึ่งครอบคลุมทุกกรณีทดสอบทดสอบการยอมรับของผู้ใช้ แสดงดังภาพที่ 13

เมื่อสิ้นสุดกระบวนการย่อย 2 ขั้นตอนดังกล่าวแล้ว ผู้ที่ทำหน้าที่ในการทบทวนกรณีทดสอบการยอมรับของผู้ใช้ สามารถเข้ามาทำการแก้ไข / ปรับปรุงกรณีทดสอบให้ครบถ้วนสมบูรณ์มากยิ่งขึ้น

3.ส่งออกข้อมูลความสัมพันธ์ระหว่างกรณีทดสอบการยอมรับของผู้ใช้และยูสเคส เป็นกระบวนการในการส่งออกข้อมูลของกรณีทดสอบ ที่มีความสัมพันธ์กับคำอธิบายยูสเคสไปให้โปรแกรมสร้างความสามารถตามรอยความต้องการ (Requirements Traceability Matrix)



ภาพที่ 13 ตัวอย่างกราฟที่ได้จาก EFSM Model

ตารางที่ 4 ตัวอย่างกรณีทดสอบการยอมรับของผู้ใช้

UAT No	Pre-Condition	Test Scenario	Expected Result	Related Use-Case	Related Requirements
UAT05	User ที่จะเข้าใช้ระบบจะต้องมีรายชื่ออยู่ในกลุ่มผู้ใช้ระบบ	<ol style="list-style-type: none"> 1. ผู้ใช้ระบบพิมพ์ User ID และ Password 2. ระบบตรวจสอบความถูกต้องของ User ID และ Password 3. ระบบจะให้ Roles แก่ผู้ใช้ระบบตามที่กำหนดไว้ 4. ผู้ใช้ระบบ สามารถเข้าระบบได้ 	ผู้ใช้ระบบสามารถเข้าใช้ระบบได้	UC3	FR10
UAT06	User ที่จะเข้าใช้ระบบจะต้องมีรายชื่ออยู่ในกลุ่มผู้ใช้ระบบ	<ol style="list-style-type: none"> 1. ผู้ใช้ระบบพิมพ์ User ID และ Password 2. ระบบตรวจสอบความถูกต้องของ User ID และ Password 2a. ผู้ใช้ระบบใส่ User ID ,Password ไม่ถูกต้อง 	ระบบไม่อนุญาตให้เข้าระบบ	UC3	FR10
UAT07	User ที่จะเข้าใช้ระบบจะต้องมีรายชื่ออยู่ในกลุ่มผู้ใช้ระบบ	<ol style="list-style-type: none"> 1. ผู้ใช้ระบบพิมพ์ User ID และ Password 2. ระบบตรวจสอบความถูกต้องของ User ID และ Password 2a. ผู้ใช้ระบบใส่ User ID ,Password ไม่ถูกต้อง 2a2.ระบบจะให้ใส่อีกครั้ง และกลับไปทำข้อที่ 2 4. ผู้ใช้ระบบ สามารถเข้าระบบได้ 	ผู้ใช้ระบบสามารถเข้าใช้ระบบได้	UC3	FR10

ตารางที่ 5 วิธีดำเนินการสร้างซีนารีโอ

Algorithm	สร้างยูสเคสซีนารีโอ
Input	เอกสารข้อกำหนดความต้องการซอฟต์แวร์ในส่วนของ Use Case Description(UC) ที่เป็นไฟล์ในรูปแบบเอ็กซ์เอ็มแอลที่สร้างมาจากระบบนำเข้าข้อมูลอัตโนมัติ(D) แทนด้วยเซต D
Output	นำเข้าข้อมูลยูสเคสซีนารีโอไปยังฐานข้อมูล
Pre-processing	<ol style="list-style-type: none"> นำเข้าข้อมูล D แยกออกมาอ่านไฟล์เอ็กซ์เอ็มแอล กำหนดนิพจน์ปรกติ (regular expression) สำหรับ Normal flow of Events REGEX = [0-9]+[.] กำหนดนิพจน์ปรกติ (regular expression) สำหรับ Alternate flows และ Exceptional flows REGEX1 = ^[0-9]+[a-zA-Z]+[:] REGEX2 = [0-9]+[a-zA-Z]+[0-9]+[.]
Main Step	<ol style="list-style-type: none"> จัดเก็บข้อมูลเอกสารข้อกำหนดความต้องการซอฟต์แวร์ในส่วนของคำอธิบายยูสเคส (UC) ลงในตารางชั่วคราว สร้างขั้นตอนการทำงานของยูสเคส ตามเงื่อนไขกำหนดนิพจน์ปรกติ สำหรับ Normal flow of Events โดยตรวจสอบจากฟิลด์ Normal flow of Events ในไฟล์เอ็กซ์เอ็มแอล โดยที่เงื่อนไขต้องอยู่ในรูปแบบ REGEX = [0-9]+[.] ตัวอย่างเช่น 1. สร้างเงื่อนไขที่ทำให้เกิดเหตุการณ์อื่น (Alternate flows / Exceptional flows) แทนด้วย A-Condition ตามเงื่อนไขกำหนดนิพจน์ปรกติ (regular expression) สำหรับ Alternate flows และ Exceptional flows โดยตรวจสอบจากฟิลด์ Alternate flows / Exceptional flows ในไฟล์เอ็กซ์เอ็มแอล โดยที่เงื่อนไขต้องอยู่ในรูปแบบ ^[0-9]+[a-zA-Z]+[:] ตัวอย่างเช่น 1a: สร้างขั้นตอนการทำงานที่ทำให้เกิดเหตุการณ์อื่นของยูสเคส โดยตรวจสอบจากฟิลด์ Alternate flows / Exceptional flows ในไฟล์เอ็กซ์เอ็มแอล ตามเงื่อนไขกำหนดนิพจน์ปรกติ (regular expression) สำหรับ Alternate flows และ Exceptional flows โดยที่เงื่อนไขต้องอยู่ในรูปแบบ [0-9]+[a-zA-Z]+[0-9]+[.] ตัวอย่างเช่น 1a01. จัดเก็บข้อมูลลงในฐานข้อมูล

Remark	
--------	--

ตารางที่ 6 วิธีดำเนินการสร้าง Extended finite state machine (EFSM)

Procedure	สร้างแบบจำลอง EFSM
Input	ยูสเคสซีนารีโอ (UC) แทนด้วยเซต UC
Output	แบบจำลอง EFSM
Pre-processing	1. ดึงข้อมูลยูสเคสซีนารีโอจากฐานข้อมูล
Main Step	<ol style="list-style-type: none"> 1. กำหนดข้อมูล Start state ตั้งต้น $\rightarrow S_0$ 2. กำหนดข้อมูล transitionSet เก็บไว้ในตัวแปรตั้งต้น 3. กำหนดข้อมูล VariableSet เก็บไว้ในตัวแปรตั้งต้น 4. กำหนดข้อมูล inputSet เก็บไว้ในตัวแปรตั้งต้น 5. กำหนดข้อมูล outputSet เก็บไว้ในตัวแปรตั้งต้น 6. ดึงข้อมูลขั้นตอนการทำงานของยูสเคสซีนารีโอจากฐานข้อมูล 7. สร้างโหนดการทำงานของยูสเคส 8. ตรวจสอบเงื่อนไขของยูสเคสซีนารีโอ <ol style="list-style-type: none"> 8.1 มี Input หรือไม่ 8.2 มี Output หรือไม่ 8.2 มี A-condition หรือไม่ 9. สร้าง Transition ของยูสเคสจากโหนดเริ่มต้นไปยังโหนดสิ้นสุด 10. ทำซ้ำข้อที่ 6 จนกว่าขั้นตอนการทำงานปกติของยูสเคสครบถ้วน
Remark	VariableSet คือเซตของ A-Condition

ตารางที่ 7 วิธีดำเนินการสร้างกรณีทดสอบที่ครอบคลุม

Procedure	สร้างกรณีทดสอบการยอมรับของผู้ใช้
Input	แบบจำลอง EFSM
Output	กรณีทดสอบการยอมรับของผู้ใช้ที่ครอบคลุม
Pre-processing	1.ดึงข้อมูลแบบจำลอง EFSM จากฐานข้อมูล
Main Step	1.ดึงข้อมูล Transition State ของแบบจำลอง EFSM จากฐานข้อมูล 2.แสดงเส้นทางทั้งหมดของ Transition State ของแบบจำลอง EFSM จากฐานข้อมูล โดยใช้วิธีการค้นหาแนวลึก 3.ทำซ้ำข้อที่ 1 จนกว่าจะครบถ้วนทุกเส้นทาง
Remark	-

3.2 ข้อกำหนดเงื่อนไขและการทำงานของระบบ

ในส่วนนี้อธิบายถึงกระบวนการโดยรวมของระบบ ประกอบไปด้วย อินพุตของระบบ, เอาต์พุตของระบบ, เงื่อนไขของระบบ, ผู้ที่เกี่ยวข้อง และการประเมินระบบการสร้างกรณีทดสอบการยอมรับของผู้ใช้โดยใช้กรณีศึกษาคือ ข้อกำหนดความต้องการซอฟต์แวร์จากบริษัทแห่งหนึ่ง โดยจะสรุปไว้ในตารางที่ 8

ตารางที่ 8 ลักษณะโดยรวมของระบบ

ลักษณะ	คำอธิบาย
อินพุตของระบบ	เอกสารข้อกำหนดความต้องการซอฟต์แวร์ในส่วนของ Use Case Description(UC) ที่เป็นไฟล์ในรูปแบบเอกซ์เอ็มแอลที่สร้างมาจากระบบ นำเข้าข้อมูลอัตโนมัติ
เงื่อนไขของระบบ	1.ต้องมีสิ่งที่ต้องเพิ่มเติมไปยังข้อกำหนดความต้องการซอฟต์แวร์ Use Case Description(UCD) ข้อมูลแถว Alternate flows / Exceptional flows หลัก จะต้องเป็นตัวเลขและตัวอักษร และขึ้นด้วย ":" เช่น 02a:ผู้ใช้รหัสผู้ใช้ระบบและ รหัสผ่านไม่ถูกต้อง ข้อมูลแถว Alternate flows / Exceptional flows ย่อย จะต้องเป็นตัวเลขและตัวอักษร และขึ้นด้วย "." เช่น 02a01.ระบบไม่อนุญาตให้เข้าระบบ ดังตัวอย่างการเขียนคำอธิบายยูสเคสในตารางที่ 9 2.UCD ไฟล์เอกซ์เอ็มแอลต้องมีข้อมูลที่สามารถนำไปสร้าง UAT ได้ 3.ไฟล์เอกซ์เอ็มแอลที่ระบบสร้าง ระบบเป็นผู้กำหนดเค้าร่างเอกซ์เอ็มแอล

ลักษณะ	คำอธิบาย
เอาต์พุตของระบบ	<p>1.ได้ข้อมูลแบ่งออกตามประเภท UC, UAT ไปยังระบบความสามารถตาม รอยความต้องการจากระบบ</p> <p>2.Test Specification โดยใช้ Test Case Specification สามารถตามรอย ความต้องการได้แก่ UC ระหว่างUAT</p> <p>3.ส่งออกข้อมูล UAT ในรูปแบบไฟล์ xml ตามรูปแบบ xml schema ที่ งานวิจัยนี้กำหนด</p>
ผู้ที่เกี่ยวข้องกับ ระบบ	ผู้ทบทวน
การประเมินระบบ	วัดจากการสร้างกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติและการส่งออก ข้อมูลสำหรับความสามารถตามรอยความต้องการซอฟต์แวร์ได้อย่าง ถูกต้องแม่นยำและ ครบถ้วน ตรงตามเอกสารนำเข้่าต้นทาง

ตารางที่ 9 ตัวอย่างการเขียนคำอธิบายยูสเคส

Use Case Name:	UC003: Login
Use case type:	Real Use Case
Importance level:	High
Related Requirement:	FR010
Primary actor:	AC001
Brief description:	จะทำการตรวจสอบรหัสผู้ใช้ระบบและ รหัสผ่านก่อนเข้าระบบ
Trigger:	-
Type:	-
Association:	-
Include:	-
Extend:	-
Generalization:	-
Normal flow of Events:	<p>1.ผู้ใช้ระบบกรอกรหัสผู้ใช้ระบบและ รหัสผ่าน</p> <p>2. ระบบตรวจสอบความถูกต้องของรหัสผู้ใช้ระบบและ รหัสผ่าน</p> <p>3. ระบบจะให้สิทธิ์แก่ผู้ใช้ระบบตามที่กำหนดไว้</p> <p>4. ผู้ใช้ระบบสามารถเข้าสู่ระบบได้</p>

Alternate flows / Exceptional flows:	02a: ผู้ใช้รหัสผู้ใช้ระบบและ รหัสผ่านไม่ถูกต้อง 02a01. ระบบไม่อนุญาตให้เข้าระบบ 02a02. ไปยังขั้นตอนที่ 2
Pre-condition	ผู้ใช้ระบบที่จะเข้าใช้ระบบจะต้องมีรายชื่ออยู่ในกลุ่มผู้ใช้ระบบ
Post-Condition	ผู้ใช้ระบบสามารถเข้าใช้ระบบได้

3.3 วิเคราะห์บุคคลที่มีบทบาทหน้าที่ของผู้ที่เกี่ยวข้องกับระบบ

บทบาทหน้าที่ของผู้ที่เกี่ยวข้องกับระบบสามารถจำแนกได้ดังนี้

3.2.1 ผู้ทบทวน (Reviewer)

- นำเข้าเอกสารข้อกำหนดความต้องการซอฟต์แวร์ที่เป็นไฟล์ในรูปแบบเอ็กซ์เอ็มแอลเข้าสู่ระบบ
- ทบทวนข้อมูลที่น่าเข้าระบบ

บทที่ 4 การออกแบบและพัฒนาระบบ

จากการศึกษากระบวนการสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้และวิเคราะห์ความสัมพันธ์ระหว่างกระบวนการสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้และกระบวนการอื่นๆ ตามที่ได้นำเสนอไปในบทที่ 3 แล้วนั้น ผู้เสนอดิถยานิพนธ์ได้ทำการสรุปความต้องการด้านหน้าที่ ความต้องการที่ไม่ใช่หน้าที่ เพื่อทำการพัฒนาระบบสนับสนุนขั้นตอนการดำเนินงานวิจัยดังกล่าว รวมถึงใช้การออกแบบในลักษณะแนวคิดเชิงวัตถุ ใช้ภาษายูเอ็มแอลในการวิเคราะห์และออกแบบระบบงาน

แผนภาพที่ใช้ในออกแบบประกอบด้วย แผนภาพยูสเคส เพื่อแสดงภาพรวมของระบบงาน คำอธิบายยูสเคส เพื่อแสดงรายละเอียดของแต่ละยูสเคส แผนภาพซีควเอนซ์ เพื่อแสดงการทำงานต่างๆ ในระบบทั้งหมด และแผนภาพคลาส เพื่อแสดงความสัมพันธ์ของข้อมูลรวมถึงข้อมูลต่างๆ ที่จะต้องทำการจัดเก็บ และนำข้อมูลที่ได้เข้าสู่ฐานข้อมูลต่อไป

4.1 ความต้องการด้านหน้าที่

ความต้องการด้านหน้าที่ (Functional Requirement) ของสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้แสดงดังตารางที่ 10

ตารางที่ 10 ความต้องการด้านหน้าที่

รหัส	รายละเอียด
FR01	ระบบสามารถนำเข้าข้อมูลจากเอกสารข้อกำหนดความต้องการซอฟต์แวร์ที่เป็นไฟล์ในรูปแบบเอ็กซ์เอ็มแอลได้
FR02	ผู้ใช้งานระบบสามารถบริหารจัดการ แก้ไข เปลี่ยนแปลงข้อมูลคำอธิบายยูสเคสได้
FR03	ระบบสามารถสร้าง EFSM Model จากยูสเคสซีนารีโอได้
FR04	ระบบสามารถวาดกราฟจาก EFSM Model ได้
FR05	ระบบสามารถสร้างเส้นทางกรณีทดสอบจากกราฟ EFSM model ได้ (Generated Test Path)
FR06	ผู้ใช้งานระบบสามารถบริหารจัดการ แก้ไข เปลี่ยนแปลงกรณีทดสอบได้
FR07	ระบบสามารถส่งออกข้อมูลของกรณีทดสอบที่มีความสัมพันธ์กับคำอธิบายยูสเคสไปให้โปรแกรมสร้างความสามารถตามรอยความต้องการในรูปแบบเอ็กซ์เอ็มแอลได้ตามรูปแบบ XML Schema ที่กำหนดได้

4.2 ความต้องการที่ไม่ใช่หน้าที่

ความต้องการที่ไม่ใช่หน้าที่ (Non Functional Requirement) ของระบบสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้แสดงดังตารางที่ 11

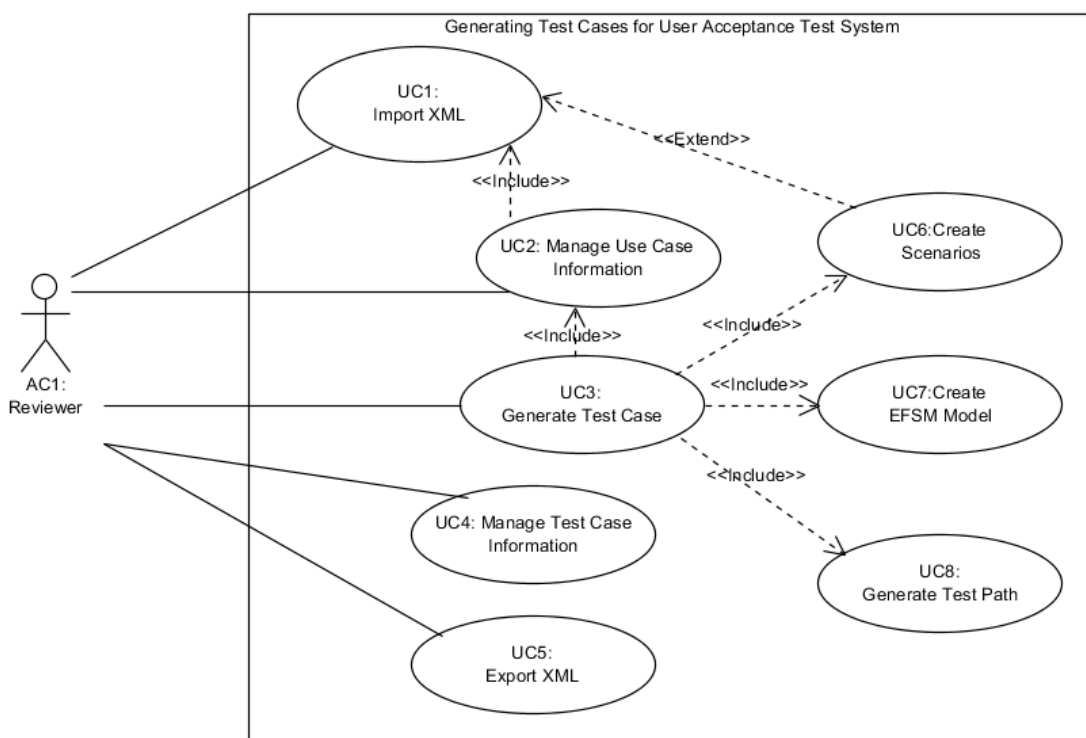
ตารางที่ 11 ความต้องการที่ไม่ใช่หน้าที่

รหัส	รายละเอียด
NFR01	ระบบสามารถเก็บ Log การแก้ไข เปลี่ยนแปลงกรณีทดสอบได้
NFR02	ระบบสามารถเก็บ Log การแก้ไข เปลี่ยนแปลงข้อมูลคำอธิบายยูสเคสได้

4.3 การออกแบบระบบ

4.3.1 แผนภาพยูสเคส (Use Case Diagram)

แผนภาพยูสเคสนำมาใช้อธิบายหน้าที่การทำงานหลักของระบบดังแสดงในภาพที่ 14



ภาพที่ 14 แผนภาพยูสเคสระบบสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้

4.3.2 คำอธิบายยูสเคส (Use Case Description)

คำอธิบายยูสเคสเป็นการใส่รายละเอียดลงไปว่าแต่ละยูสเคสหรือแต่ละฟังก์ชันของระบบหรือซอฟต์แวร์ มีการเริ่มต้นโดยผู้กระทำ หรือระบบที่เกี่ยวข้องกับระบบอย่างไร มีการทำงาน

เกิดขึ้นอย่างไร รวมถึงจุดสิ้นสุดการทำงานลงอย่างไร โดยคำอธิบายยูสเคสของแต่ละยูสเคสดังแสดงในภาพที่ 14 มีดังนี้ แสดงในตารางที่ที่ 12 ถึง ตารางที่ที่ 19

ตารางที่ 12 คำอธิบายยูสเคส Import XML

หมายเลขยูสเคส : UC01	ชื่อยูสเคส : Import XML
ผู้เกี่ยวข้องหลัก : Reviewer	
รายละเอียด : เป็นยูสเคสที่อธิบายถึงขั้นตอนในการนำเข้าข้อมูลเอกสารรูปแบบเอ็กซ์เอ็มแอลจากระบบ [X] เพื่อเป็นข้อมูลตั้งต้นในการสร้างกรณีทดสอบ	
ความสัมพันธ์ : Association : Reviewer Use : - Extend : - Generalization : -	
กระแสเหตุการณ์ปกติ: 1. Reviewer เข้าระบบไปที่เมนู Import XML 2. Reviewer นำเข้าข้อมูลเอกสารข้อกำหนดความต้องการซอฟต์แวร์ที่เป็นไฟล์ในรูปแบบเอ็กซ์เอ็มแอล 3. ระบบตรวจสอบโครงสร้างเอ็กซ์เอ็มแอล 4. ระบบจัดเก็บข้อมูลลงเพิ่มข้อมูลชั่วคราว 5. ระบบแสดงข้อความทำรายการสำเร็จ 6. Reviewer ยืนยันการบันทึกข้อมูล 7. ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log	
กระแสเหตุการณ์ทางเลือก:- 3a: โครงสร้างเอ็กซ์เอ็มแอลไม่ถูกต้อง 3a1.ระบบแสดงข้อความเตือนให้นำเข้าเอกสารใหม่อีกครั้ง 3a2.กลับไปทำขั้นตอนที่ 2 อีกครั้ง 6a:ไม่ยืนยันการบันทึกข้อมูล 6a1.ระบบสิ้นสุดการทำงาน และบันทึก log	

ตารางที่ 13 คำอธิบายยูสเคส Manage Use Case Information

หมายเลขยูสเคส : UC02	ชื่อยูสเคส : Manage Use Case Information
ผู้เกี่ยวข้องหลัก : Reviewer	
รายละเอียด : เป็นยูสเคสที่อธิบายถึงขั้นตอนในการปรับปรุง/แก้ไข คำอธิบายยูสเคสที่ได้ทำการนำเข้าข้อมูลก่อนที่จะดำเนินการสร้างกรณีทดสอบ	
ความสัมพันธ์ : Association : Reviewer Use : UC01 : Import XML Extend : - Generalization : -	
กระแสเหตุการณ์ปกติ: <ol style="list-style-type: none"> 1. Reviewer เข้าระบบไปที่เมนู Use Case Information 2. Reviewer ระบุยูสเคสที่ต้องการปรับปรุงแก้ไข 3. ระบบแสดงยูสเคสที่ Reviewer ต้องการ 4. Reviewer คลิกที่ปุ่ม Edit เพื่อทำการปรับปรุงแก้ไข 5. ระบบเปิดฟิลด์ตามที่กำหนดเพื่อให้แก้ไขยูสเคส 6. Reviewer คลิกที่ปุ่ม Save เพื่อบันทึกข้อมูลปรับปรุงแก้ไข 7. ระบบแสดงข้อความเตือนเพื่อบันทึกข้อมูล 8. ระบบตรวจสอบข้อมูลที่บันทึก 9. ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log 	
กระแสเหตุการณ์ทางเลือก:- 7a:ไม่ยืนยันการบันทึกข้อมูล 7a1.ระบบสิ้นสุดการทำงาน และบันทึก log	

ตารางที่ 14 คำอธิบายยูสเคส Generates Test Case

หมายเลขยูสเคส : UC03	ชื่อยูสเคส : Generates Test Case
ผู้เกี่ยวข้องหลัก : Reviewer	
รายละเอียด : เป็นยูสเคสที่อธิบายถึงขั้นตอนในการสร้างกรณีทดสอบการยอมรับของผู้ใช้	
ความสัมพันธ์ : Association : Reviewer Use : UC02: Manage Use Case Information Extend : - Generalization : -	
กระแสเหตุการณ์ปกติ: 1. Reviewer ไปที่ต้นไม้นำทาง 2. Reviewer คลิกขวาที่ยูสเคสหลักที่ต้องการสร้างกรณีทดสอบ (สร้างกรณีทดสอบทุกยูสเคส) 3. ระบบทำการสร้างยูสเคสขึ้นมาวีโอ 4. ระบบสร้าง EFSM Model 5. ระบบสร้างเส้นทางกรณีทดสอบจาก EFSM 6. ระบบแสดงข้อความทำรายการสำเร็จ 7. ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log	
กระแสเหตุการณ์ทางเลือก: 2a: Reviewer คลิกขวาที่ยูสเคสย่อยที่ต้องการสร้างกรณีทดสอบทดสอบ (สร้างกรณียูสเคสหนึ่งๆ) 2a1.ระบบทำการสร้างยูสเคสขึ้นมาวีโอ 2a2.ระบบสร้าง EFSM Model 2a3.ระบบสร้างเส้นทางกรณีทดสอบจาก EFSM 2a4.ระบบแสดงข้อความทำรายการสำเร็จ 2a5.ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล 6a:ระบบทำรายการไม่สำเร็จ 6a1.ระบบแสดงข้อความเตือน 6a2.สิ้นสุดการทำงาน และบันทึก log	

ตารางที่ 15 คำอธิบายยูสเคส Manage Test Case Information

หมายเลขยูสเคส : UC04	ชื่อยูสเคส : Manage Test Case Information
ผู้เกี่ยวข้องหลัก : Reviewer	
รายละเอียด : เป็นยูสเคสที่อธิบายถึงขั้นตอนในการปรับปรุง/แก้ไข กรณีทดสอบการยอมรับของผู้ใช้ที่ได้มาจากการสร้างกรณีทดสอบจากขั้นตอนก่อนหน้า	
ความสัมพันธ์ : Association : Reviewer Use : UC03:Generates Test Case Extend : - Generalization : -	
กระแสเหตุการณ์ปกติ: <ol style="list-style-type: none"> 1. Reviewer เข้าระบบไปที่เมนู Test Case Information 2. Reviewer ระบุกรณีทดสอบที่ต้องการปรับปรุงแก้ไข 3. ระบบแสดงกรณีทดสอบที่ Reviewer ต้องการ 4. Reviewer คลิกที่ปุ่ม Edit เพื่อทำการปรับปรุงแก้ไข 5. ระบบเปิดฟิลด์ตามที่กำหนดเพื่อให้แก้ไขกรณีทดสอบ 6. Reviewer คลิกที่ปุ่ม Save เพื่อบันทึกข้อมูลปรับปรุงแก้ไข 7. ระบบแสดงข้อความเตือนเพื่อบันทึกข้อมูล 8. ระบบตรวจสอบข้อมูลที่บันทึก 9. ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log 	
กระแสเหตุการณ์ทางเลือก: <p>7a:ไม่ยืนยันการบันทึกข้อมูล</p> <p>7a1.ระบบสิ้นสุดการทำงาน และบันทึก log</p>	

ตารางที่ 16 คำอธิบายยูสเคส Export XML To other System

หมายเลขยูสเคส : UC05	ชื่อยูสเคส : Export XML To other System
ผู้เกี่ยวข้องหลัก : Reviewer	
รายละเอียด : เป็นยูสเคสที่อธิบายถึงขั้นตอนในการส่งออกข้อมูลความสัมพันธ์ระหว่างยูสเคสกับกรณีทดสอบความต้องการของยูสเคสนั้นไปยังเมตริกซ์ความสามารถตามรอยความต้องการโดยอัตโนมัติในรูปแบบเอกสารเอ็กซ์เอ็มแอล และ .docx	
ความสัมพันธ์ : Association : Reviewer Use : UC04:Manage Test Case Information Extend : - Generalization : -	
กระแสเหตุการณ์ปกติ: <ol style="list-style-type: none"> 1. Reviewer เข้าระบบไปที่เมนู Report 2. Reviewer คลิกที่ปุ่ม Export XML 3. ระบบตรวจสอบข้อมูลที่จำเป็น 4. ระบบสร้างกรณีทดสอบในรูปแบบเอ็กซ์เอ็มแอล 5. ระบบส่งออกข้อมูลให้ Reviewer ตามเงื่อนไขที่ระบุ 6. ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log 	
กระแสเหตุการณ์ทางเลือก:- <ol style="list-style-type: none"> 4a: ไม่สามารถสร้างกรณีทดสอบในรูปแบบเอ็กซ์เอ็มแอล 4a1.ระบบแสดงข้อความเตือน 4a2.ระบบสิ้นสุดการทำงาน และบันทึก log 	

ตารางที่ 17 คำอธิบายยูสเคส Create Scenario

หมายเลขยูสเคส : UC6	ชื่อยูสเคส : Create Scenario
ผู้เกี่ยวข้องหลัก :-	
รายละเอียด : เป็นยูสเคสที่อธิบายถึงขั้นตอนในการสร้างยูสเคสขึ้นมาวิธีใด	
ความสัมพันธ์ : Association : UC03:Generates Test Case Use : - Extend : - Generalization : -	
กระแสเหตุการณ์ปกติ: 1. ระบบดึงข้อมูลยูสเคสจากฐานข้อมูลมาประมวลผล 2. ระบบทำการสร้างขั้นตอนการทำงานปกติของยูสเคส 3. ระบบตรวจสอบเงื่อนไขที่ทำให้เกิดเหตุการณ์อื่น ของขั้นตอนการทำงานปกติของยูสเคส 4. ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log	
กระแสเหตุการณ์ทางเลือก:-	

ตารางที่ 18 คำอธิบายยูสเคส Create EFSM Model

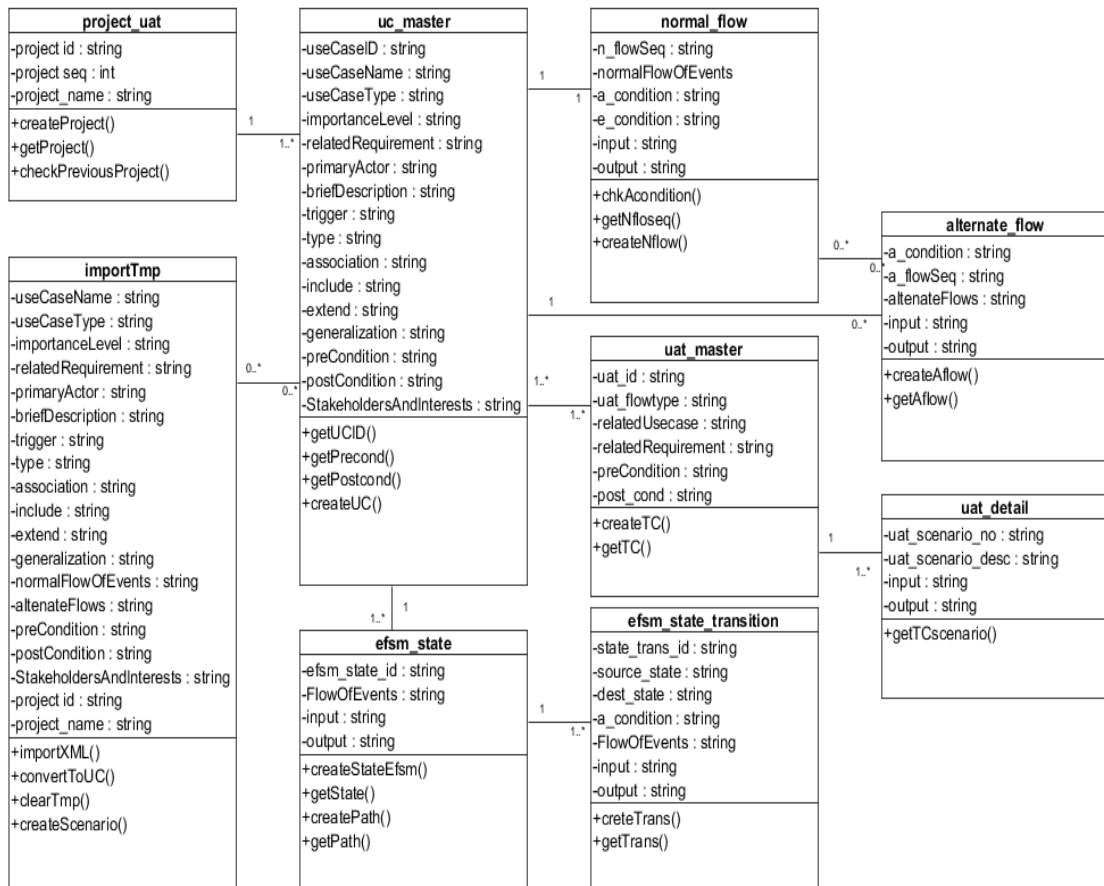
หมายเลขยูสเคส : UC7	ชื่อยูสเคส : Create EFSM Model
ผู้เกี่ยวข้องหลัก :-	
รายละเอียด : เป็นยูสเคสที่อธิบายถึงขั้นตอนในการสร้าง EFSM Model	
ความสัมพันธ์ : Association : UC03:Generates Test Case Use : - Extend : - Generalization : -	
กระแสเหตุการณ์ปกติ: <ol style="list-style-type: none"> 1. ระบบสร้างโหนดเริ่มต้น 2. ระบบดึงข้อมูลขั้นตอนการทำงานของยูสเคสจากฐานข้อมูล 3. สร้างโหนดของยูสเคส 4. ตรวจสอบเงื่อนไขที่ทำให้เกิดเหตุการณ์อื่น 5. สร้างเงื่อนไขที่ทำให้เกิดเหตุการณ์อื่น 6. สร้าง Transition ของยูสเคส 7. ทำซ้ำข้อที่ 2 จนกว่าขั้นตอนการทำงานปกติ และขั้นตอนที่ทำให้เกิดเหตุการณ์อื่นของยูสเคสครบถ้วน 8. ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log 	
กระแสเหตุการณ์ทางเลือก:-	

ตารางที่ 19 คำอธิบายยูสเคส Create Test Path

หมายเลขยูสเคส : UC8	ชื่อยูสเคส : Create Test Path
ผู้เกี่ยวข้องหลัก :-	
รายละเอียด : เป็นยูสเคสที่อธิบายถึงขั้นตอนในการสร้างเส้นทางกรณีทดสอบ	
ความสัมพันธ์ : Association : UC03:Generates Test Case Use : - Extend : - Generalization : -	
กระแสเหตุการณ์ปกติ: 1. ระบบดึงข้อมูล EFSM Model จากฐานข้อมูล 2. ระบบสร้างเส้นทางของแต่ละโหนดใน EFSM Model 3. ระบบสร้างข้อมูล เงื่อนไขก่อนหน้า และเงื่อนไขตามหลัง(ผลลัพธ์ที่คาดหวัง) ในแต่ละเส้นทางที่สร้างได้ 4. ระบบสร้างข้อมูลนำเข้า และข้อมูลนำออกในแต่ละเส้นทางที่สร้างได้ 5. ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log	
กระแสเหตุการณ์ทางเลือก:-	

4.3.3 แผนภาพคลาส (Class Diagram)

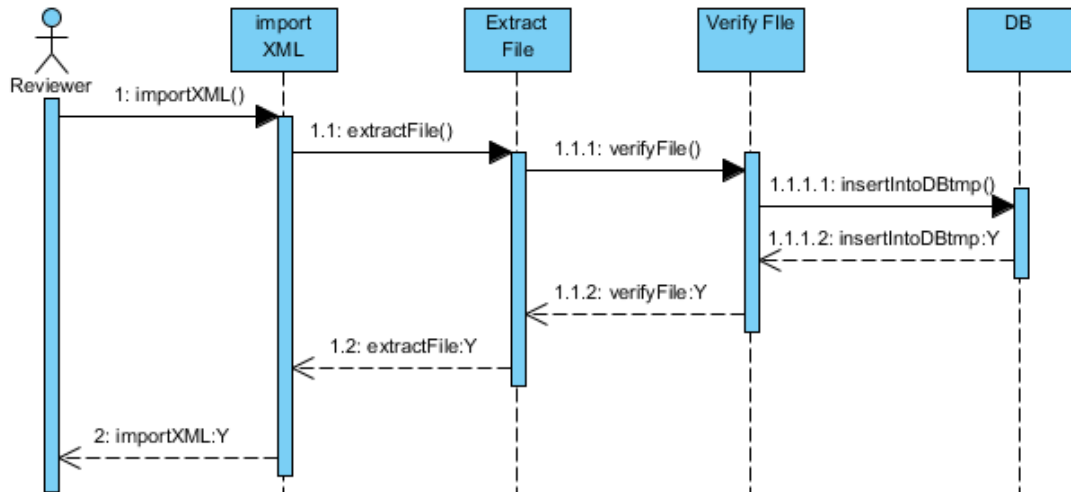
การวิเคราะห์โครงสร้างของระบบใช้แผนภาพคลาส ในการแสดงภาพรวมของระบบและส่วนย่อยของระบบ โดยแผนภาพคลาสดังภาพที่ 15



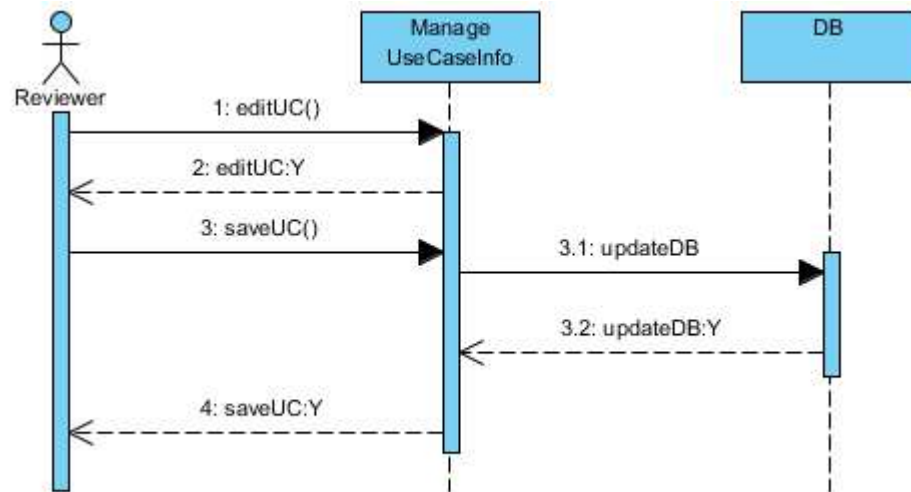
ภาพที่ 15 แผนภาพการออกแบบคลาสของระบบ

4.3.3 แผนภาพซีควเอนซ์ (Sequence Diagram)

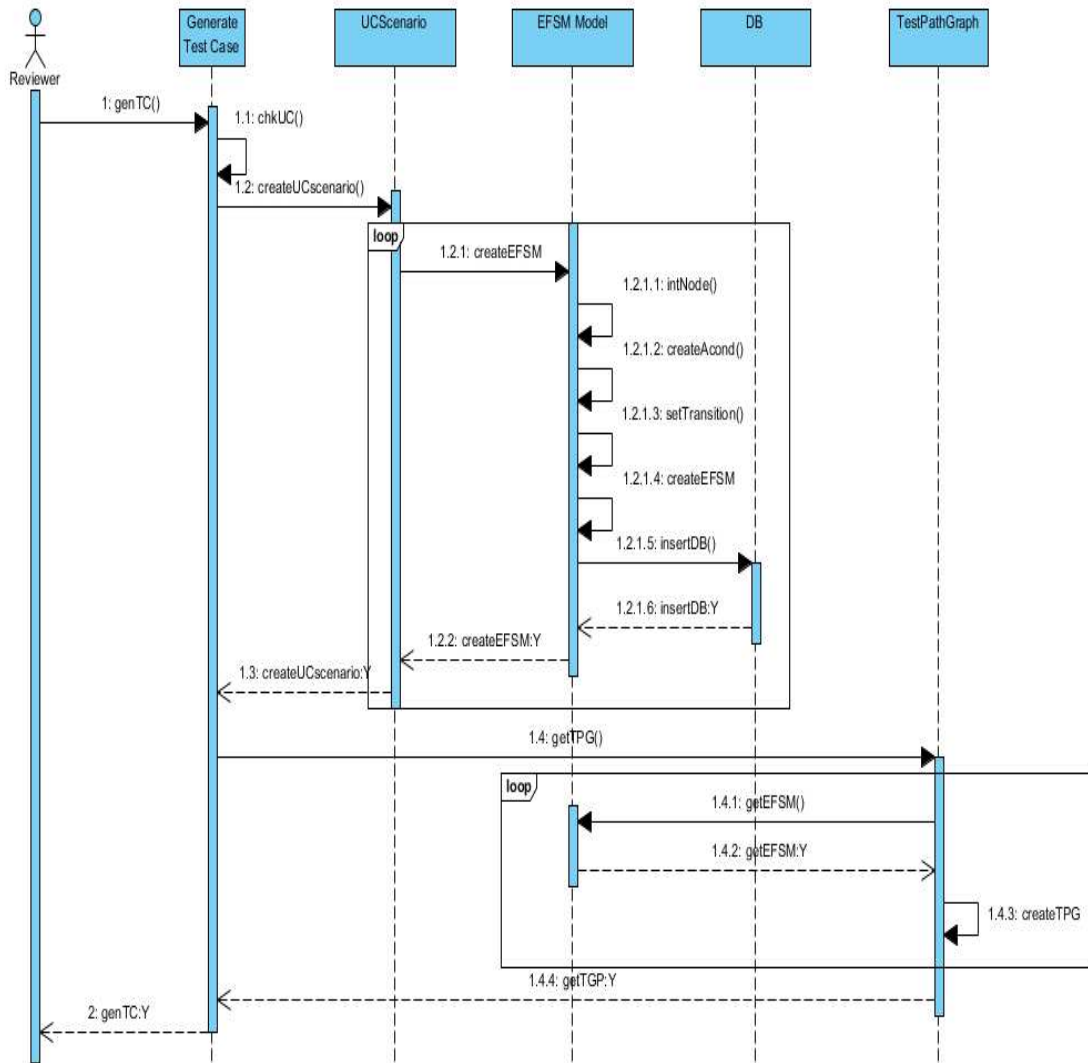
ส่วนการวิเคราะห์พฤติกรรมของระบบ คือ การนำยูสเคสที่สำคัญในแผนภาพยูสเคสมาวิเคราะห์ว่าจะนำคลาสใดบ้างมาทำงานในยูสเคสนั้น เพื่อศึกษาพฤติกรรม (Behavior) ของระบบ แสดงในภาพที่ 16 ถึง ภาพที่ 20



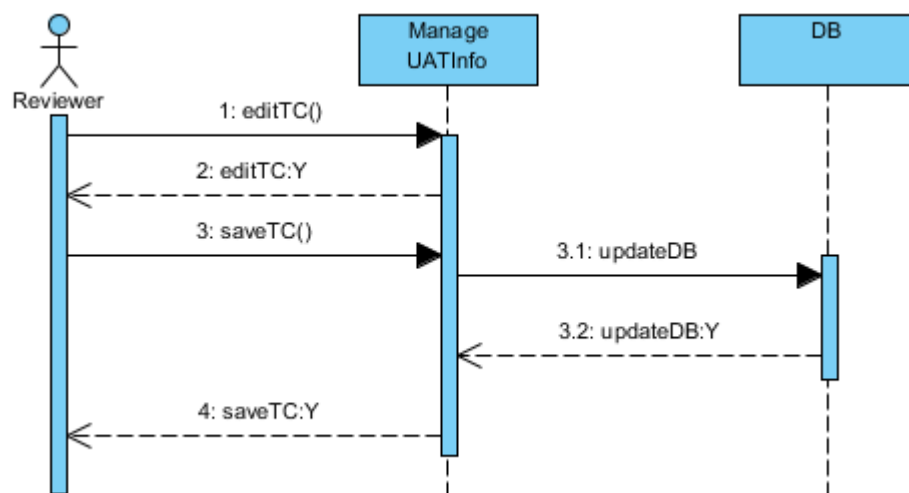
ภาพที่ 16 แผนภาพซีควเอนซ์ Import XML



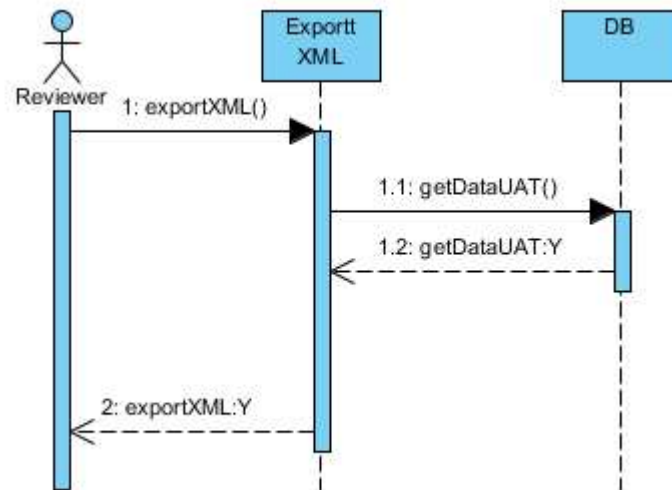
ภาพที่ 17 แผนภาพซีควเอนซ์ Manage Use Case Info



ภาพที่ 18 แผนภาพซีควเอนซ์ Generates Test Case



ภาพที่ 19 แผนภาพซีควเอนซ์ Manage UAT Info



ภาพที่ 20 แผนภาพซีควเอนซ์ Extract XML

4.3.4 การพัฒนาฐานข้อมูล

จากแผนภาพคลาสใน 4.3.4 สามารถพัฒนาฐานข้อมูลที่มีการจัดเก็บตามเค้าร่าง (Schema) และรายละเอียดข้อมูลที่จัดเก็บ ดังต่อไปนี้

4.3.4.1 เค้าร่างฐานข้อมูล (Database Schema)

เป็นส่วนใหญ่ใช้แสดงโครงสร้างตาราง ประกอบไปด้วยคุณลักษณะ (Attribute) อะไรบ้างและมีการกำหนด Primary Key และ Foreign Key ที่มีความสัมพันธ์กับ Entity อื่น ดังต่อไปนี้ สามารถดูรายละเอียดของพจนานุกรมข้อมูล (Data Dictionary) ของระบบ ฐานข้อมูล ได้จากภาคผนวก ก ปรากฏในตารางที่ 25 ถึง 34

tbl_uc_master (projectid, useCaseID, useCaseName, useCaseType, importanceLevel, relatedRequirement, primaryActor, briefDescription, trigger, type, association, include, extend, generalization, preCondition, postCondition, StakeholdersAndInterests)

tbl_importTmp (useCaseName, useCaseType, importanceLevel, relatedRequirement, primaryActor, briefDescription, trigger, type, association, include, extend, generalization, normalFlowOfEvents, altenateFlows,

preCondition, postCondition, StakeholdersAndInterests, project id,
project_name, table_num, data id_relate)

tbl_alternate_flow (projectid, useCaseID, a_condition, a_flowSeq, alternateFlows,
input, output)

tbl_normal_flow (projectid, useCaseID, n_flowSeq, normalFlowOfEvents,
a_condition, e_condition, input, output)

tbl_uat_master (projectid, uat_id, uat_flowtype, relatedUsecase,
relatedRequirement, preCondition, post_cond)

tbl_uat_detail (projectid, uat_id, uat_scenario_no, uat_scenario, input, output)

tbl_efsm_state (projectid, useCaseID, efsm_state_id, FlowOfEvents, input,
output)

tbl_efsm_state_transition (projectid, useCaseID, state_trans_id, source_state,
dest_state, a_condition, FlowOfEvents, input, output)

4.4 การพัฒนาระบบ

4.4.1 สภาพแวดล้อมและเครื่องมือที่ใช้ในการพัฒนา

สภาพแวดล้อมที่ใช้ในการพัฒนาระบบมีสภาพแวดล้อมทางด้านฮาร์ดแวร์และซอฟต์แวร์
ดังต่อไปนี้

ฮาร์ดแวร์

1. หน่วยประมวลผลอินเทล คอร์ 2 ดิวโอ 2.1 กิกะเฮิร์ต (CPU Intel Core 2 Duo 2.1GHz)
2. หน่วยความจำ (RAM) 2,048 เมกกะไบต์ (2,048 MB)
3. ฮาร์ดดิสก์ (Hard disk) 160 กิกะไบต์ (160 GB)

ซอฟต์แวร์

1. ระบบปฏิบัติการ วินโดวส์เอ็กซ์พี โพรเฟสชันนอล (Windows XP Professional)
2. โปรแกรมสภาพแวดล้อมประมวลผลจาวา 6 (Java Runtime Environment 6)

3. โปรแกรมชุดเครื่องมือพัฒนาจาวา 6 (Java Development Kit 6)
4. โปรแกรมเนตบีนส์ ไอดีอี 7.1 (NetBeans IDE 7.1)
5. ระบบจัดการฐานข้อมูลมายเอสคิวแอล 5.5 (MySQL 5.5)

4.4.2 การติดตั้งซอฟต์แวร์ในการพัฒนาระบบ

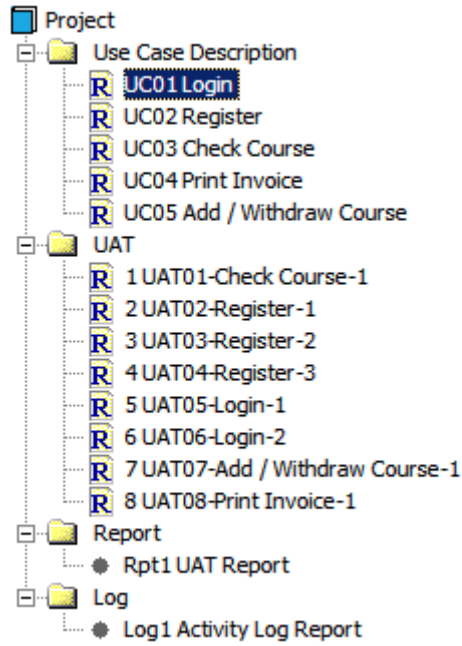
เมื่อเตรียมเครื่องมือสำหรับการพัฒนาระบบเรียบร้อยแล้ว จึงทำการติดตั้งเครื่องมือทั้งหมดลงในเครื่องคอมพิวเตอร์ที่ใช้พัฒนาระบบ โดยมีลำดับการติดตั้งเครื่องมือเป็นไปตามขั้นตอนต่อไปนี้

1. ติดตั้งระบบปฏิบัติการ วินโดวส์ เอ็กซ์พี โปรเฟสชันนอล
2. โปรแกรมสภาพแวดล้อมประมวลผลจาวา 6 (Java Runtime Environment 6)
3. โปรแกรมชุดเครื่องมือพัฒนาจาวา 6 (Java Development Kit 6)
4. โปรแกรมเนตบีนส์ ไอดีอี 7.1 (NetBeans IDE 7.1)
5. ระบบจัดการฐานข้อมูลมายเอสคิวแอล 5.5 (MySQL 5.5)

4.4.3 การพัฒนาส่วนต่อประสาน

พัฒนาส่วนต่อประสานผู้ใช้ ให้สอดคล้องกับขอบเขตของระบบจากบทที่ 1 และมีการทำงานตรงกับตามความต้องการด้านหน้าที่และความต้องการที่ไม่ใช่หน้าที่ โดยการพัฒนาร่วมต่อประสานกับผู้ใช้จะใช้ไลบรารีสวิง (Swing) ของภาษาจาวาในการพัฒนา โดยเรียกใช้ไลบรารีผ่านส่วนต่อประสานโปรแกรมประยุกต์ที่ภาษาจาวาจัดเตรียมไว้ นอกจากนี้โปรแกรมเนตบีนส์ ไอดีอี 7 ยังมีเครื่องมือสำหรับสร้างส่วนต่อประสานกราฟิกกับผู้ใช้ชื่อว่า ตัวสร้างจิวไอ (GUI Builder) ผู้ใช้สามารถลากและวางคอมโพเนนต์ของส่วนต่อประสานกราฟิกกับผู้ใช้ที่ต้องการได้ทันที จากนั้นโปรแกรมเนตบีนส์ยังสามารถสร้างซอสโค้ดภาษาจาวาให้อัตโนมัติ ช่วยให้พัฒนาส่วนต่อประสานกับผู้ใช้ได้อย่างรวดเร็ว

หลังจากเปิดโปรแกรมหน้าจอสำหรับการใช้งานหลักจะปรากฏขึ้นแบ่งออกเป็น 2 ส่วน คือ ส่วนต้นไม้นำทางจะปรากฏอยู่ทางซ้ายของโปรแกรม และส่วนหน้าจอทำงานหลักจะปรากฏอยู่ทางขวาของโปรแกรม แสดงหน้าจอโปรแกรมสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้ในภาพที่ 21 และภาพที่ 22



ภาพที่ 21 หน้าจอส่วนต้นไม่นำทาง

Use Case Information | Data State Table | Graph View

Type: Use Case

Prefix: UC 01

Name: Login

Primary Actor: AC01

Relate Req: FRO1

UC Description: นักศึกษาทำการเข้าสู่ระบบลงทะเบียนเรียนออนไลน์ โดยการกรอกรหัสนักศึกษา และรหัสผ่านให้ถูกต้อง เพื่อเข้าสู่ระบบ และทำการลงทะเบียนเรียน

Pre-Condition: นักศึกษาที่เข้าสู่ระบบลงทะเบียนเรียนออนไลน์ จะต้องมีสถานะเป็นนักศึกษามีจวัน และกำลังศึกษาอยู่เท่านั้น

Post-Condition: นักศึกษาที่เข้าสู่ระบบลงทะเบียนเรียนออนไลน์ และอยู่ในสถานะเข้าสู่ระบบ

Include:

Normal-Flow of Event (Main Success Scenario)

No	Scenario Description	Input / Parameter	Output / Expected Result
1.	นักศึกษาเข้าสู่ระบบลงทะเบียนเรียน		
2.	ระบบแสดงหน้าจอให้กรอกรหัสนักศึกษา และรหัสผ่าน		
3.	นักศึกษกรอกรหัสนักศึกษา และรหัสผ่าน		
4.	ระบบตรวจสอบความถูกต้องของรหัสนักศึกษาและรหัสผ่าน		
5.	รหัสนักศึกษาและรหัสผ่านถูกต้อง ระบบตรวจสอบสถานะการเป็นนักศึกษา ของนักศึกษา		
6.	นักศึกษาเข้าสู่ระบบสำเร็จ และอยู่ในสถานะเข้าสู่ระบบ		

Alternate / Exception Flow

No	Scenario Description	Alternate Condition	Input / Parameter	Output / Expected R...
1		4a: นักศึกษกรอกรหัสนักศึกษา และรหัสผ่านไม่ถูกต้อง		
2	4a1. ระบบแสดงข้อความผิดพลาด ? รหัสนักศึกษา และรหัสผ่านไม่ถูกต้อง?			
3	4a2. ระบบแสดงหน้าจอให้กรอกรหัสนักศึกษา และรหัสผ่านอีกครั้ง			
4	4a3. Go to 2			

Priority: High

Responsibility:

Status: Proposed

ภาพที่ 22 หน้าจอโปรแกรมสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้

4.4.3.1 เมนู (Menu)

เมนูของระบบประกอบด้วย

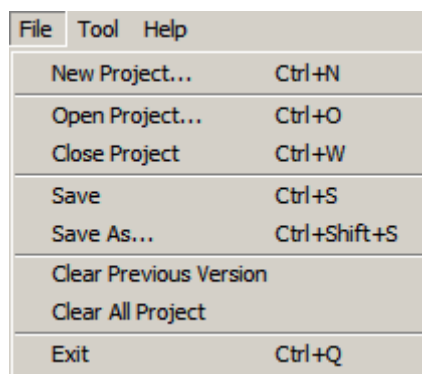
1) เมนูบาร์

เมนูบาร์ในระบบจะมีด้วยกัน 3 เมนูได้แก่ เมนูไฟล์ เมนูเครื่องมือและเมนูช่วยเหลือ

1.เมนูไฟล์ จะมีเมนูย่อย ดังภาพที่ 23 มีดังนี้

1.1.เมนู New Project สำหรับการสร้าง Project ใหม่

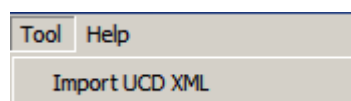
1.2.เมนู Open Project สำหรับการเปิด Project ของเดิมมาตรวจสอบหรือแก้ไข



ภาพที่ 23 หน้าจอเมนูไฟล์

2.เมนูเครื่องมือหน้าที่จัดการการตั้งค่าของระบบจะมีการเพิ่มในส่วนเมนูย่อย ดังภาพที่ 24 มีดังนี้

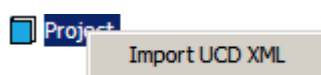
2.1 Import UCD XML



ภาพที่ 24 หน้าจอเมนูเครื่องมือ

2) เมนูคลิกขวา

เมนูคลิกขวา คือ เมนูที่ปรากฏขึ้นเมื่อคลิกขวาบนโนหนดของต้นไม้แนวทาง เมนูย่อยที่เพิ่มเติมในระบบคือ "Import UCD XML" เพื่อเป็นช่องทางในการสร้างกรณีทดสอบจากยูสเคส ดังภาพที่ 25 มีดังนี้



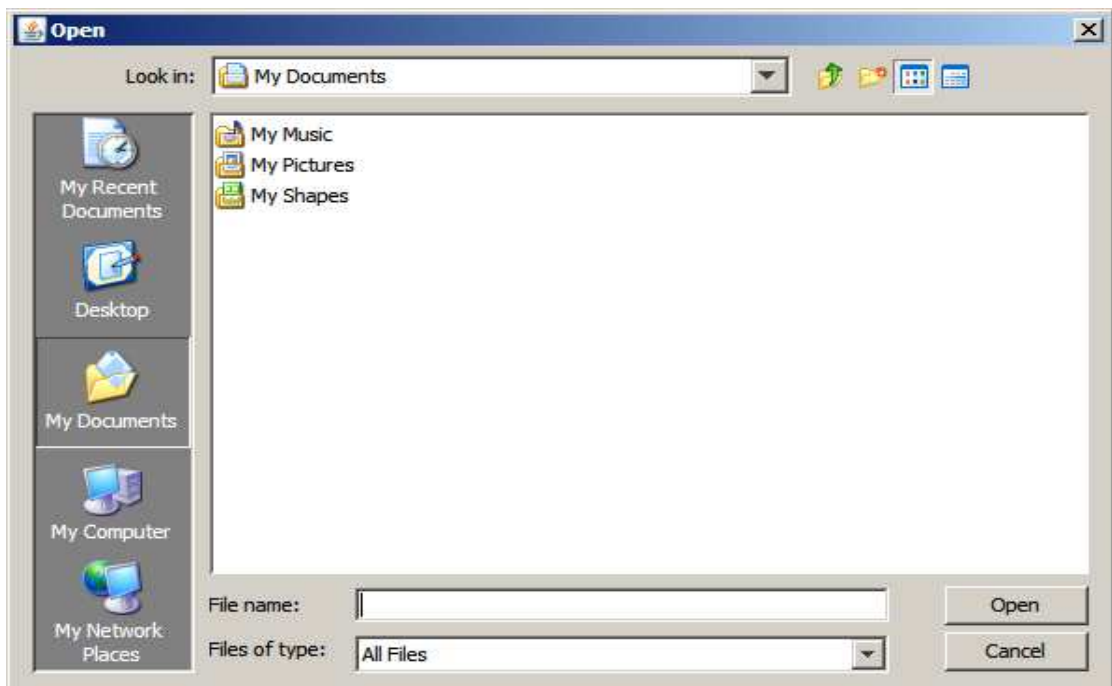
ภาพที่ 25 หน้าจอเมนูคลิกขวา

4.4.3.2 หน้าจอ (Screen)

1) หน้าจอ Import XML เมื่อผู้ใช้ระบบไปที่ เมนูบาร์ > Tool > Import XML File ระบบจะแสดงหน้าจอเพื่อให้ผู้ใช้ได้ทำการเลือกไฟล์เอกสารคำอธิบายยูสเคสในรูปแบบเอกสารเอ็กซ์เอ็มแอล ดังภาพที่ 26 และเพื่อเตรียมประมวลผลในขั้นตอนถัดไป ดังภาพที่ 27 มีดังนี้

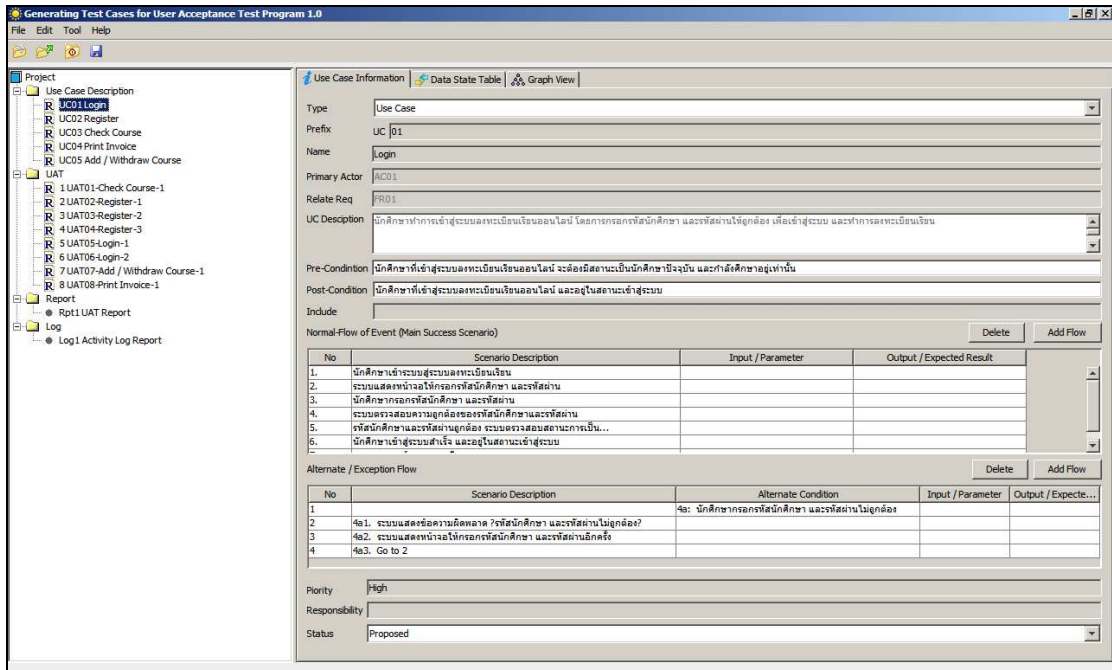
```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
- <project id="25550108190823940" name="NPA SRS V3.0">
  <package id="0" name="User Requirements" />
  <package id="1" name="Product Feature" />
+ <package id="2" name="Functional Requirement">
+ <package id="3" name="Non-Functional Requirement">
- <package id="4" name="Use Case Description">
  <data id_relate="25550108190842740000" name="Alternate flows" table_num="1">-</data>
  <data id_relate="255501081908426280002" name="Brief description" table_num="1">เป็นการกำหนด Security ในการเข้าถึงข้อมูลให้กับผู้ปฏิบัติงาน</data>
  <data id_relate="255501081908433420006" name="Importance level" table_num="1">High</data>
  <data id_relate="255501081908448170008" name="Normal flow of Events" table_num="1">1. ระบบแสดงรายชื่อ User\n2. User Admin เลือกชื่อ User ที่จะกำหนดสิทธิ์\n3. User Admin กำหนดสิทธิ์การเข้าถึงข้อมูลของผู้ปฏิบัติงาน โดยแบ่งกลุ่มดังนี้\n3.1 กลุ่มผู้ใช้ทั่วไปสามารถบันทึกและแก้ไขข้อมูลของตนเองได้ และสามารถดูข้อมูลของทั้งหมดได้\n3.2 กลุ่มผู้บริหารสามารถเรียกดูข้อมูลได้ทั้งหมด และสามารถแก้ไขข้อมูลที่ความดูแลของตนเองได้ทั้งหมด\n4. ระบบบันทึกข้อมูล</data>
  <data id_relate="255501081908441850009" name="Post-Condition" table_num="1">User สามารถกำหนด Security ในการเข้าถึงข้อมูลให้กับผู้ปฏิบัติงานได้สำเร็จ</data>
  <data id_relate="2555010819084422200010" name="Pre-condition" table_num="1">มีรายชื่อ User ที่จะกำหนดสิทธิ์ในระบบ</data>
  <data id_relate="2555010819084411800011" name="Primary actor" table_num="1">AC02,AC03</data>
  <data id_relate="2555010819084493800012" name="Related Requirement" table_num="1">FR09</data>
  <data id_relate="2555010819084432300015" name="Use Case Name" table_num="1">UC1: Setup Security</data>
  <data id_relate="255501081908466100016" name="Use case type" table_num="1">Real Use Case</data>
</package>
<package id="5" name="User Acceptance Test" />
<package id="6" name="User" />
</project>
```

ภาพที่ 26 ข้อมูลนำเข้าคำอธิบายยูสเคสในรูปแบบเอ็กซ์เอ็มแอล



ภาพที่ 27 หน้าจอเลือกไฟล์เอกสารเอ็กซ์เอ็มแอล

2) หน้าจอ Use Case Information เป็นหน้าจอสำหรับทวนสอบ ปรับปรุงแก้ไข ยูสเคส โดยก่อนที่จะใช้งานผู้ใช้ระบบต้องทำการนำเข้าข้อมูลในรูปแบบเอกสารเอกซ์เอ็มแอลให้เสร็จสิ้นก่อน หรือเปิด Project เดิม เพื่อทำการปรับปรุงแก้ไขยูสเคส ดังภาพที่ 28 มีดังนี้



ภาพที่ 28 หน้าจอ Use Case Information

การทำงานของหน้านี้ เริ่มจากผู้ใช้งานเลือกยูสเคสที่ต้องการปรับปรุงแก้ไขจากต้นไม้ทางด้านซ้ายมือ จากนั้นระบบจะแสดงข้อมูลคำอธิบายของยูสเคส โดยผู้ใช้สามารถแก้ไขได้โดยคลิกที่ปุ่ม Edit ระบบจะเปิดไฟล์ให้แก้ไขในส่วนขั้นตอนการทำงานปกติของยูสเคส (Normal Flow of event) และขั้นตอนแสดงเงื่อนไขที่ทำให้เกิดเหตุการณ์อื่นที่อาจเกิดขึ้นได้ในบางกรณี (Alternate/Exceptional flows) ทั้งยังสามารถระบุตัวแปรนำเข้า และผลลัพธ์ที่คาดหวังในแต่ละขั้นตอนการทำงานของยูสเคสได้ เมื่อแก้ไขเสร็จสิ้นให้คลิกที่ปุ่ม Save เพื่อทำการบันทึกข้อมูลดังกล่าวลงในฐานข้อมูล

3) หน้าจอ Data State Table เป็นหน้าจอสำหรับตรวจสอบเส้นทางของโหนดในแต่ละยูสเคสซีนารีโอ และเงื่อนไขที่ทำให้เกิดเหตุการณ์อื่น ดังภาพที่ 28

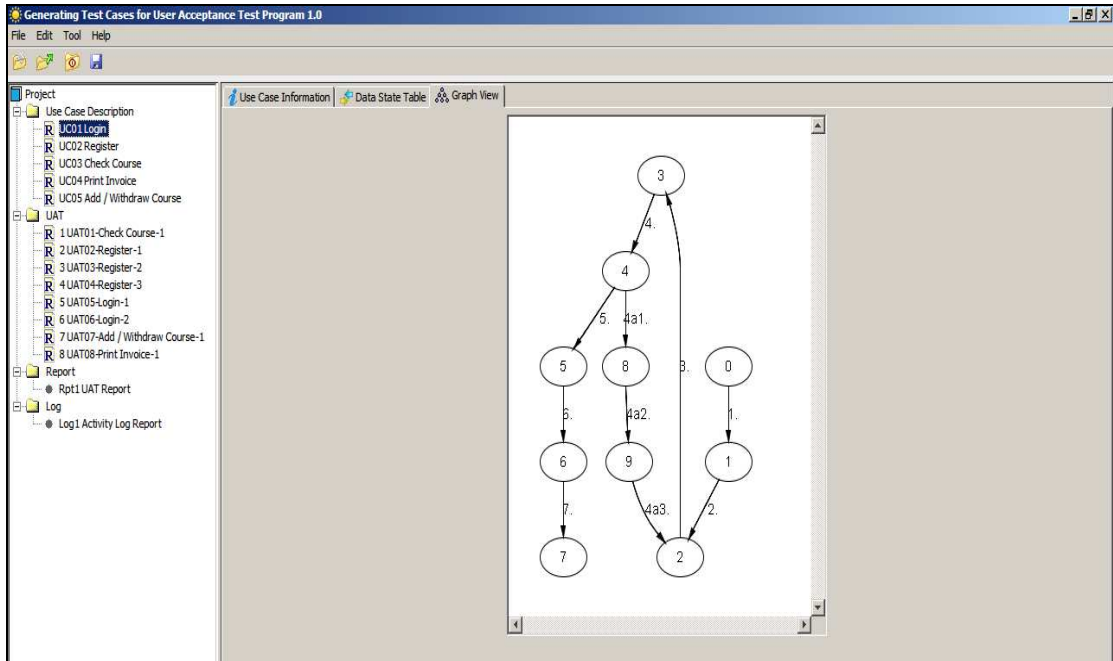
State No	Scenario No	Scenario Description	A-Condition No	A-Condition Description	Input / Para...	Output / Exp...	Start State	Finish State
1	1.	นักศึกษาชำระระบบลงทะเบียนเรียน					0	1
2	2.	ระบบแสดงหน้าจอโปรแกรมกรทิส...					1	2
3	3.	นักศึกษากรทิสนักศึกษา และรหัสผ่าน					2	3
4	4.	ระบบตรวจสอบความถูกต้องของ...					3	4
5	5.	รหัสนักศึกษาและรหัสผ่านถูกต้อง...					4	5
6	6.	นักศึกษาเข้าสู่ระบบสำเร็จ และอยู่ใน...					5	6
7	7.	ระบบแสดงหน้าจอลงทะเบียน					6	7
8	4a1.	ระบบแสดงข้อความผิดพลาด 7รหัส...	4a:	นักศึกษากรทิสนักศึกษา และ...			4	8
9	4a2.	ระบบแสดงหน้าจอโปรแกรมกรทิส...					8	9
10	4a3.	Go to 2					9	2

Path Number	Path Description	Flow Type
1	Start->1->2->3->4->4a1->4a2->4a3.	Alternate Flow
2	Start->1->2->3->4->5->6->7.	Normal Flow

ภาพที่ 29 หน้าจอ Data State Table

การทำงานของหน้าจอนี้เริ่มจากผู้ใช้งานเลือกยูสเคสที่ต้องการจากต้นไม้ทางด้านซ้ายมือ จากนั้นระบบจะแสดงข้อมูลในแท็บ Data State Table โดยข้อมูลนี้จะได้มาจากการที่ผู้ใช้เลือก เมฆคลิกขวา คือ เมนูที่ปรากฏขึ้นเมื่อคลิกขวาบนโหนดของต้นไม้ทาง คือ “Generate UAT All” และ “Generate UAT by UC” ในหน้าจอนี้ผู้ใช้ไม่สามารถแก้ไขได้สามารถดูได้อย่างเดียว

4) หน้าจอ EFSM Graph เป็นหน้าจอสำหรับตรวจสอบเส้นทางของโหนดในแต่ละยูสเคส ซีนารีโอ และเงื่อนไขที่ทำให้เกิดเหตุการณ์อื่น ดังภาพที่ 30



ภาพที่ 30 หน้าจอ EFSM Graph

การทำงานของหน้าจอนี้เริ่มจากผู้ใช้งานเลือกยูสเคสที่ต้องการจากต้นไม้ทางด้านซ้ายมือ จากนั้นระบบจะแสดงข้อมูลในแท็บ EFSM Graph จะเป็นการวาดกราฟโดยนำข้อมูลที่ได้จากหน้าจอแท็บ Data State Table ในหน้าจอนี้ผู้ใช้ไม่สามารถแก้ไขได้สามารถดูได้อย่างเดียว

5) หน้าจอ UAT เป็นหน้าจอสำหรับทวนสอบ ปรับปรุงแก้ไขกรณีทดสอบการยอมรับของผู้ใช้ ดังภาพที่ 31

No	Scenario Description	Input / Parameter	Output / Expected Result
1.	นักศึกษาเข้าสู่ระบบลงทะเบียนเรียน		
2.	ระบบแสดงหน้าจอให้กรอกรหัสนักศึกษา และรหัสผ่าน		
3.	นักศึกษากรอกรหัสนักศึกษา และรหัสผ่าน		
4.	ระบบตรวจสอบความถูกต้องของรหัสนักศึกษาและรหัสผ่าน		
4a1.	ระบบแสดงข้อความผิดพลาด รหัสนักศึกษา และรหัสผ่านไม่ถูกต้อง?		
4a2.	ระบบแสดงหน้าจอให้กรอกรหัสนักศึกษา และรหัสผ่านอีกครั้ง		
4a3.	Go to 2null		

Test Path: Start->1.->2.->3.->4.->4a1.->4a2.->4a3.

ภาพที่ 31 หน้าจอ UAT

การทำงานของหน้าจอเริ่มจาก ผู้ใช้งานเลือกกรณีทดสอบที่ต้องการจากต้นไม้ นำทางด้านซ้ายมือ จากนั้นระบบจะแสดงข้อมูลกรณีทดสอบที่ต้องการ โดยก่อนที่จะใช้งานผู้ใช้ระบบ ต้องทำการสร้างกรณีทดสอบจากขั้นตอน “Generate UAT All” และ “Generate UAT by UC” ให้เสร็จสิ้น ในหน้าจอนี้ผู้ใช้สามารถแก้ไขโดยการคลิกที่ปุ่ม Edit ระบบจะเปิดฟิลด์ให้แก้ไขในส่วนของลำดับขั้นการทดสอบ (Test Scenarios/Step) ประกอบด้วย No, Description, Input (ระบุตัวแปรนำเข้า) และ Output/Expected Result (ผลลัพธ์ที่คาดหวัง) ในแต่ละขั้นตอนการทำงาน of กรณีทดสอบเพื่อครอบคลุมกรณีทดสอบได้ดียิ่งขึ้น เมื่อแก้ไขเสร็จสิ้นให้คลิกที่ปุ่ม Save เพื่อทำการบันทึกข้อมูลดังกล่าวลงในฐานข้อมูล

4.4.3.3 รายงาน (Report)

1) Report UAT เป็นหน้าจอสำหรับการออกรายงานกรณีทดสอบการยอมรับของผู้ใช้ ซึ่งแสดงสามารถความสัมพันธ์ ระหว่างความต้องการเชิงหน้าที่ ความต้องการที่ไม่ใช่หน้าที่ และยูสเคส กับกรณีทดสอบการยอมรับของผู้ใช้ ดังภาพที่ 32

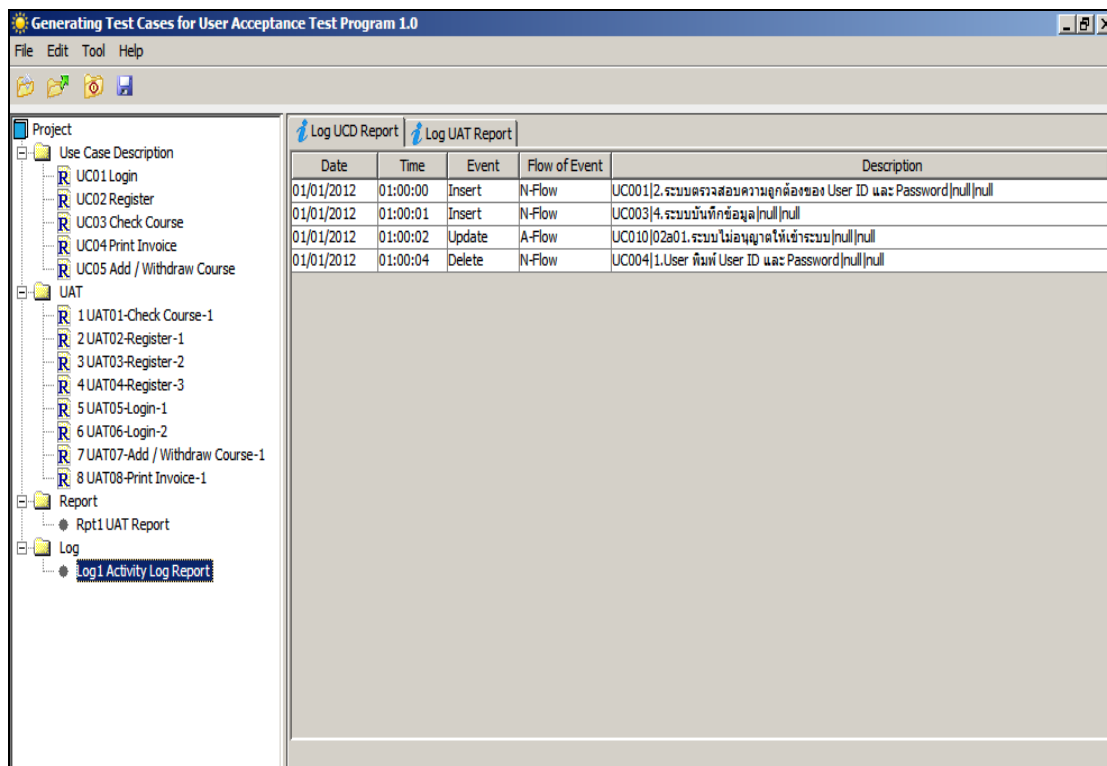
uat_id	precond	uat_scenario	postcond	relreq	usecase_id
UAT01 Check Course-1		1. นักศึกษาเข้าสู่หน้าจอ...	รายชื่อที่ลงทะเบียนจะสัมพันธ์กับนักศึกษา	FR03	UC03
UAT02 Register-1	1. นักศึกษาเข้าสู่ระบบลงทะเบียนเรียน...	1. นักศึกษาเข้าสู่หน้าจอ...	1. นักศึกษาทำการลงทะเบียน และรายชื่อจะท...	FR02	UC02
UAT03 Register-2	1. นักศึกษาเข้าสู่ระบบลงทะเบียนเรียน...	1. นักศึกษาเข้าสู่หน้าจอ...	1. นักศึกษาทำการลงทะเบียน และรายชื่อจะท...	FR02	UC02
UAT04 Register-3	1. นักศึกษาเข้าสู่ระบบลงทะเบียนเรียน...	1. นักศึกษาเข้าสู่หน้าจอ...	1. นักศึกษาทำการลงทะเบียน และรายชื่อจะท...	FR02	UC02
UAT05 Login-1	นักศึกษเข้าสู่ระบบลงทะเบียนเรียนออนไลน์ ...	1. นักศึกษาเข้าสู่ระบบ...	นักศึกษเข้าสู่ระบบลงทะเบียนเรียนออนไลน์ ...	FR01	UC01
UAT06 Login-2	นักศึกษเข้าสู่ระบบลงทะเบียนเรียนออนไลน์ ...	1. นักศึกษาเข้าสู่ระบบ...	นักศึกษเข้าสู่ระบบลงทะเบียนเรียนออนไลน์ ...	FR01	UC01
UAT07 Add / Withdraw Course-1		1. ระบบเห็นรายชื่อที่...		FR04	UC05
UAT08 Print Invoice-1		1. นักศึกษาเข้าสู่หน้าจอ...	การพิมพ์ใบแจ้งหนี้ จะทำสำเร็จ	FR05	UC04

ภาพที่ 32 หน้าจอ Report UAT

การทำงานของหน้าจอเริ่มจาก ผู้ใช้งานเลือกรายงานกรณีทดสอบที่ต้องการจากต้นไม้ นำทางด้านซ้ายมือ และเลือก UAT Report จากนั้นระบบจะแสดงข้อมูลรายงานกรณีทดสอบ ประกอบไปด้วยข้อมูล Test Case ID, Pre-Condition, Test-Scenarios, Expected Result, Relate Use Case และ Relate Requirements ในหน้าจอนี้ผู้ใช้สามารถทำการ Export XML

ข้อมูลกรณีทดสอบการยอมรับของผู้ใช้เพื่อส่งข้อมูลความสัมพันธ์ระหว่างยูสเคสกับกรณีทดสอบ ความต้องการของยูสเคสนั้นไปยังเมตริกซ์ความสามารถตามรอยความต้องการ

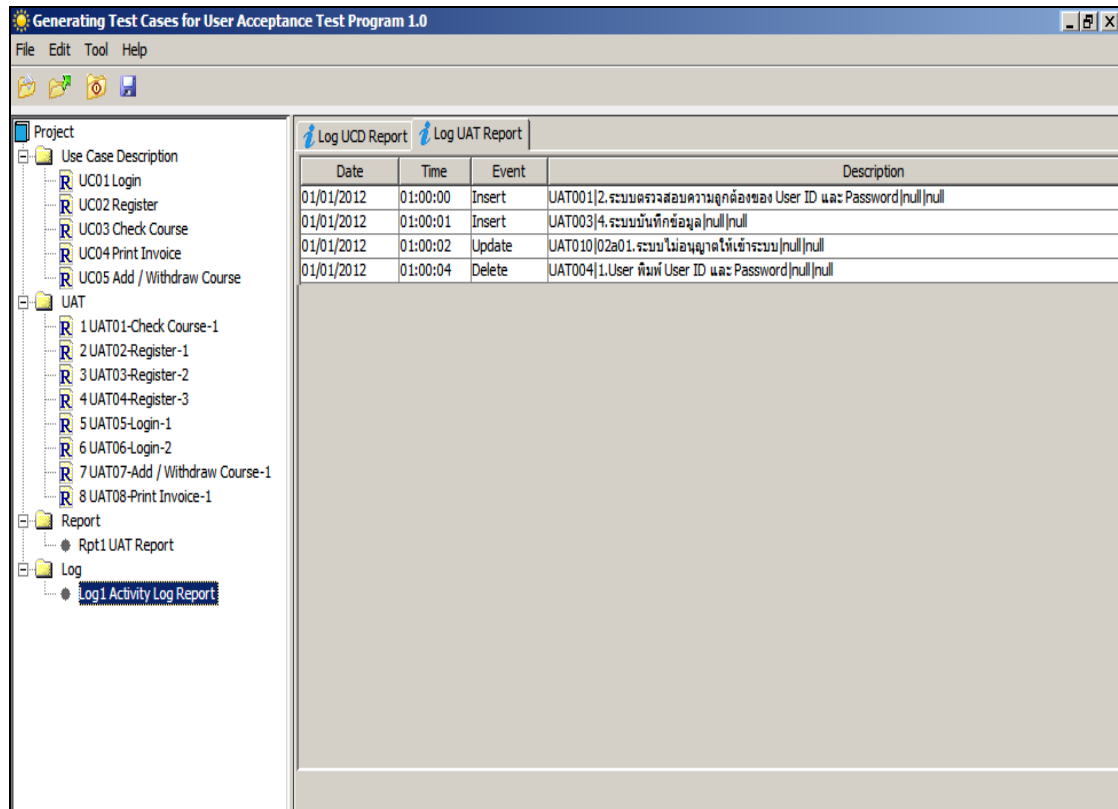
2)Report Logging Use Case Information เป็นหน้าจอสำหรับการออกรายงาน log การแก้ไขยูสเคส และรายละเอียด ดังภาพที่ 33



ภาพที่ 33 หน้าจอ Log UC

การทำงานของหน้าจอนี้เริ่มจาก ผู้ใช้งานเลือกรายงานที่ต้องการจากต้นไม้ทางด้านซ้ายมือ และเลือก Log Report จากนั้นระบบจะแสดงข้อมูลรายงาน UC Log

3)Report Logging UAT Information เป็นหน้าจอสำหรับการออกรายงาน log การแก้ไขกรณีทดสอบ และรายละเอียด ดังภาพที่ 34



ภาพที่ 34 หน้าจอ Log UAT

การทำงานของหน้าจอนี้เริ่มจาก ผู้ใช้งานเลือกรายงานที่ต้องการจากต้นไม้ทางด้านซ้ายมือ และเลือก Log Report จากนั้นระบบจะแสดงข้อมูลรายงาน TC Log

บทที่ 5

การทดสอบระบบ

การทดสอบระบบสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้มีจุดประสงค์เพื่อดำเนินการตรวจสอบความถูกต้องของฟังก์ชันการทำงานต่างๆ ของระบบว่าทำงานได้อย่างถูกต้อง และทำงานได้ครอบคลุมตามขอบเขตที่งานวิจัยได้กำหนดไว้หรือไม่ นอกจากนี้ยังเป็นการช่วยตรวจทานข้อผิดพลาดของระบบด้วย

5.1 การทดสอบความถูกต้องฟังก์ชันการทำงานของระบบ

การทดสอบความถูกต้องฟังก์ชันการทำงานของระบบ เป็นการทดสอบหน้าที่การทำงานในส่วนต่างๆ ของระบบ ตามที่ได้ระบุไว้ในหัวข้อความต้องการด้านฟังก์ชันการทำงานในบทที่ 4 ทั้งนี้ การทดสอบระบบได้อาศัยวิธีการทดสอบฟังก์ชันการทำงาน (Black Box Testing) ตามกรณีทดสอบที่ได้ดำเนินการออกแบบไว้ ซึ่งมีการทดสอบ และผลการทดสอบที่ได้ สามารถแสดงได้ดังตารางที่ 20 ดังนี้

ตารางที่ 20 ผลการทดสอบความต้องการเชิงหน้าที่

หมายเลขกรณีทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่คาดหวัง	ผลการทดสอบจริง
TC001	ทดสอบการนำเข้าเอกสารในรูปแบบเอ็กซ์เอ็มแอล	1.เข้าระบบไปที่เมนูImport XML 2.นำเข้าข้อมูลเอกสารข้อกำหนดความต้องการซอฟต์แวร์ที่เป็นไฟล์ในรูปแบบเอ็กซ์เอ็มแอล 3.ระบบตรวจสอบโครงสร้างเอ็กซ์เอ็มแอล 4.ระบบจัดเก็บข้อมูลลงแฟ้มข้อมูลชั่วคราว 5.ระบบแสดงข้อความทำรายการสำเร็จ 6.ยืนยันการบันทึกข้อมูล	ระบบสามารถนำเข้าข้อมูลจากเอกสารข้อกำหนดความต้องการซอฟต์แวร์ที่เป็นไฟล์ในรูปแบบเอ็กซ์เอ็มแอลได้	ทำงานถูกต้อง

		7.ระบบจัดเก็บข้อมูลที่ได้ลง ฐานข้อมูล และบันทึก log		
TC002	ทดสอบแก้ไข เปลี่ยนแปลง ข้อมูลคำอธิบาย ยูสเคส	1.เข้าระบบไปที่เมนู Use Case Information 2.ระบุยูสเคสที่ต้องการ ปรับปรุงแก้ไข 3.ระบบแสดงยูสเคสที่ ผู้ ทบทวนต้องการ 4.คลิกที่ปุ่ม Edit เพื่อทำการ ปรับปรุงแก้ไข 5.ระบบเปิดฟิลด์ตามที่ กำหนดเพื่อให้แก้ไขยูสเคส 6.คลิกที่ปุ่ม Save เพื่อบันทึก ข้อมูลปรับปรุงแก้ไข 7.ระบบแสดงข้อความเตือน เพื่อบันทึกข้อมูล 8.ระบบตรวจสอบข้อมูลที่ บันทึก 9.ระบบจัดเก็บข้อมูลที่ได้ลง ฐานข้อมูล และบันทึก log	ผู้ใช้ระบบสามารถ บริหารจัดการ แก้ไข เปลี่ยนแปลงข้อมูล คำอธิบายยูสเคสได้	ทำงาน ถูกต้อง
TC003	ทดสอบสร้าง EFSM Model จากยูสเคสซีนารี ไอ	1.ไปที่ต้นไม้แนวทาง 2.คลิกขวาที่ยูสเคสหลักที่ ต้องการสร้างกรณีทดสอบ (สร้างกรณีทดสอบทุกยูส เคส) 3.ระบบทำการสร้างยูสเคสซี นารีไอ 4.ระบบสร้าง EFSM Model 5.ระบบสร้างเส้นทางกรณี ทดสอบจาก EFSM	ระบบสามารถสร้าง EFSM Model จาก ยูสเคสซีนารีไอได้	ทำงาน ถูกต้อง

		<p>6.ระบบแสดงข้อความทำรายการสำเร็จ</p> <p>7.ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log</p>		
TC004	ทดสอบวาดกราฟจาก EFSM Model	<p>1.ไปที่ต้นไม้นำทาง</p> <p>2.คลิกขวาที่ยูสเคสหลักที่ต้องการสร้างกรณีทดสอบ (สร้างกรณีทดสอบทุกยูสเคส)</p> <p>3.ระบบทำการสร้างยูสเคสขึ้นมาฟรีไอ</p> <p>4.ระบบสร้าง EFSM Model</p> <p>5.ระบบสร้างเส้นทางกรณีทดสอบจาก EFSM</p> <p>6.ระบบแสดงข้อความทำรายการสำเร็จ</p> <p>7.ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log</p>	ระบบสามารถวาดกราฟจาก EFSM Model ได้	ทำงานถูกต้อง
TC005	ทดสอบสร้างเส้นทางกรณีทดสอบจากกราฟ EFSM model	<p>1.ไปที่ต้นไม้นำทาง</p> <p>2.คลิกขวาที่ยูสเคสหลักที่ต้องการสร้างกรณีทดสอบ (สร้างกรณีทดสอบทุกยูสเคส)</p> <p>3.ระบบทำการสร้างยูสเคสขึ้นมาฟรีไอ</p> <p>4.ระบบสร้าง EFSM Model</p> <p>5.ระบบสร้างเส้นทางกรณีทดสอบจาก EFSM</p> <p>6.ระบบแสดงข้อความทำรายการสำเร็จ</p>	ระบบสามารถสร้างเส้นทางกรณีทดสอบจากกราฟ EFSM model ได้ (Generated Test Path)	ทำงานถูกต้อง

		7.ระบบจัดเก็บข้อมูลที่ได้ลง ฐานข้อมูล และบันทึก log		
TC006	ทดสอบจัดการ แก้ไข เปลี่ยนแปลงกรณี ทดสอบ	1.เข้าระบบไปที่เมนูTest Case Information 2.ระบุกรณีทดสอบที่ต้องการ ปรับปรุงแก้ไข 3.ระบบแสดงกรณีทดสอบที่ ผู้ทบทวนต้องการ 4.คลิกที่ปุ่ม Edit เพื่อทำการ ปรับปรุงแก้ไข 5.ระบบเปิดฟิลด์ตามที่ กำหนดเพื่อให้แก้ไขกรณี ทดสอบ 6.คลิกที่ปุ่ม Save เพื่อบันทึก ข้อมูลปรับปรุงแก้ไข 7.ระบบแสดงข้อความเตือน เพื่อบันทึกข้อมูล 8.ระบบตรวจสอบข้อมูลที่ บันทึก 9.ระบบจัดเก็บข้อมูลที่ได้ลง ฐานข้อมูล และบันทึก log	ผู้ใช้ระบบสามารถ บริหารจัดการ แก้ไข เปลี่ยนแปลงกรณี ทดสอบได้	ทำงาน ถูกต้อง
TC007	ทดสอบส่งออก ข้อมูลของกรณี ทดสอบที่มี ความสัมพันธ์กับ คำอธิบายยูสเคส ไปให้โปรแกรม สร้าง ความสามารถ	1.เข้าระบบไปที่เมนู Report 2.คลิกที่ปุ่ม Export XML 3.ระบบตรวจสอบข้อมูลที่ จำเป็น 4.ระบบสร้างกรณีทดสอบใน รูปแบบเอ็กซ์เอ็มแอล 5.ระบบส่งออกข้อมูลตาม เงื่อนไขที่ระบุ	ระบบสามารถ ส่งออกข้อมูลของ กรณีทดสอบที่มี ความสัมพันธ์กับ คำอธิบายยูสเคสไป ให้โปรแกรมสร้าง ความสามารถตาม รอยความต้องการ	ทำงาน ถูกต้อง

	ตามรอยความต้องการใน รูปแบบเอ็กซ์เอ็มแอล	6.ระบบจัดเก็บข้อมูลที่ได้ลง ฐานข้อมูล และบันทึก log	ในรูปแบบเอ็กซ์เอ็มแอลได้ ตามรูปแบบ XML Schema ที่ กำหนดได้	
--	--	--	--	--

ตารางที่ 21 ผลการทดสอบความต้องการที่ไม่ใช่เชิงหน้าที่

หมายเลข กรณี ทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่ คาดหวัง	ผลการ ทดสอบ จริง
TC008	ทดสอบเก็บ Log การแก้ไข เปลี่ยนแปลงกรณี ทดสอบได้	1.เข้าระบบไปที่เมนู Test Case Information 2.ระบุกรณีทดสอบที่ ต้องการปรับปรุงแก้ไข 3.ระบบแสดงกรณีทดสอบ ที่ต้องการ 4.คลิกที่ปุ่ม Edit เพื่อทำ การปรับปรุงแก้ไข 5.ระบบเปิดฟิลด์ตามที่ กำหนดเพื่อให้แก้ไขกรณี ทดสอบ 6.คลิกที่ปุ่ม Save เพื่อ บันทึกข้อมูลปรับปรุงแก้ไข 7.ระบบแสดงข้อความ เตือนเพื่อบันทึกข้อมูล 8.ระบบตรวจสอบข้อมูลที่ บันทึก 9.ระบบจัดเก็บข้อมูลที่ได้ ลงฐานข้อมูล และบันทึก log 10.ตรวจสอบความถูกต้อง จาก Report UAT Log	ระบบสามารถเก็บ Log การแก้ไข เปลี่ยนแปลงกรณี ทดสอบได้	ทำงาน ถูกต้อง

TC009	ทดสอบเก็บ Log การแก้ไข เปลี่ยนแปลง ข้อมูลคำอธิบาย ยูสเคส	1.เข้าระบบไปที่เมนู Use Case Information 2.ระบุยูสเคสที่ต้องการปรับปรุงแก้ไข 3.ระบบแสดงยูสเคสที่ต้องการ 4.คลิกที่ปุ่ม Edit เพื่อทำการปรับปรุงแก้ไข 5.ระบบเปิดฟิลด์ตามที่กำหนดเพื่อให้แก้ไขยูสเคส 6.คลิกที่ปุ่ม Save เพื่อบันทึกข้อมูลปรับปรุงแก้ไข 7.ระบบแสดงข้อความเตือนเพื่อบันทึกข้อมูล 8.ระบบตรวจสอบข้อมูลที่บันทึก 9.ระบบจัดเก็บข้อมูลที่ได้ลงฐานข้อมูล และบันทึก log 10.ตรวจสอบความถูกต้องจาก Report UAT Log	ระบบสามารถเก็บ Log การแก้ไข เปลี่ยนแปลงข้อมูล คำอธิบายยูสเคสได้	ทำงานถูกต้อง
-------	--	---	---	--------------

5.2 การทดสอบความถูกต้องกระบวนการทำงานของระบบ

การทดสอบความถูกต้องกระบวนการทำงานของระบบ เป็นการทดสอบกระบวนการทำงานตามขอบเขตแนวทางการสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้ ซึ่งได้ระบุไว้ในบทที่ 3

ทั้งนี้การทดสอบได้ใช้กรณีศึกษาคือ ข้อกำหนดความต้องการซอฟต์แวร์จากบริษัทแห่งหนึ่ง ที่ครอบคลุมตามขอบเขตรูปแบบพื้นฐานคำอธิบายยูสเคสที่อธิบายไว้ในงานวิจัยชิ้นนี้มาสร้างเป็นกรณีทดสอบ โดยที่การทดสอบ และผลการทดสอบที่ได้ ซึ่งสามารถแสดงได้ดังตารางที่ 22 ดังนี้

ตารางที่ 22 การทดสอบความถูกต้องกระบวนการทำงานของระบบ

หมายเลข กรณี ทดสอบ	การทดสอบ	คำอธิบาย	ผลการทดสอบที่ คาดหวัง	ผลการ ทดสอบ จริง
TC010	ตรวจสอบเอกสาร เอ็กซ์เอ็มแอลที่ ระบบนำเข้า	เอ็กซ์เอ็มแอลของระบบ เป็นการยืนยันการนำเข้า ข้อมูลต่างๆเข้าสู่ระบบ	ข้อมูลเอ็กซ์เอ็มแอล เป็นไปข้อมูลที่เข้าสู่ ระบบ	ถูกต้อง
TC011	การสร้างกรณี ทดสอบการ ยอมรับของผู้ใช้ จากคำอธิบายยูส เคส	กรณีทดสอบการยอมรับ ของผู้ใช้ถูกสร้างจาก คำอธิบายยูสเคส	กรณีทดสอบการ ยอมรับของผู้ใช้จาก คำอธิบายยูสเคส ครอบคลุมข้อมูล คำอธิบายยูสเคสที่อยู่ ในเอกสารกับที่ระบบ สร้างขึ้น	ถูกต้อง
TC012	ตรวจสอบ ความถูกต้องของ ข้อมูลกรณี ทดสอบการ ยอมรับของผู้ใช้ที่ ระบบสร้าง	ข้อมูลกรณีทดสอบการ ยอมรับของผู้ใช้ที่ระบบ สร้างมีความถูกต้อง	ข้อมูลกรณีทดสอบการ ยอมรับของผู้ใช้ใน ระบบมีความถูกต้อง	ถูกต้อง
TC013	การตรวจสอบ สถานะฐานข้อมูล ที่เกิดขึ้น	ตรวจสอบฐานข้อมูลที่เกิด ขึ้นกับที่คาดหวังจะได้จาก เอกสาร	ข้อมูลเป็นไปตาม เอกสารเอ็กซ์เอ็มแอล นำเข้า	ถูกต้อง
TC014	ตรวจสอบเอกสาร เอ็กซ์เอ็มแอลที่ ระบบสร้างขึ้นให้ ระบบอื่น	เอ็กซ์เอ็มแอลของระบบ สร้างขึ้นครอบคลุมเนื้อหาที่ ใช้	ข้อมูลเอ็กซ์เอ็มแอลนำ ออกข้อมูลกรณี ทดสอบเพื่อการ ยอมรับของผู้ใช้ เพื่อที่ ให้ระบบอื่นนำไปใช้ต่อ ได้	ถูกต้อง

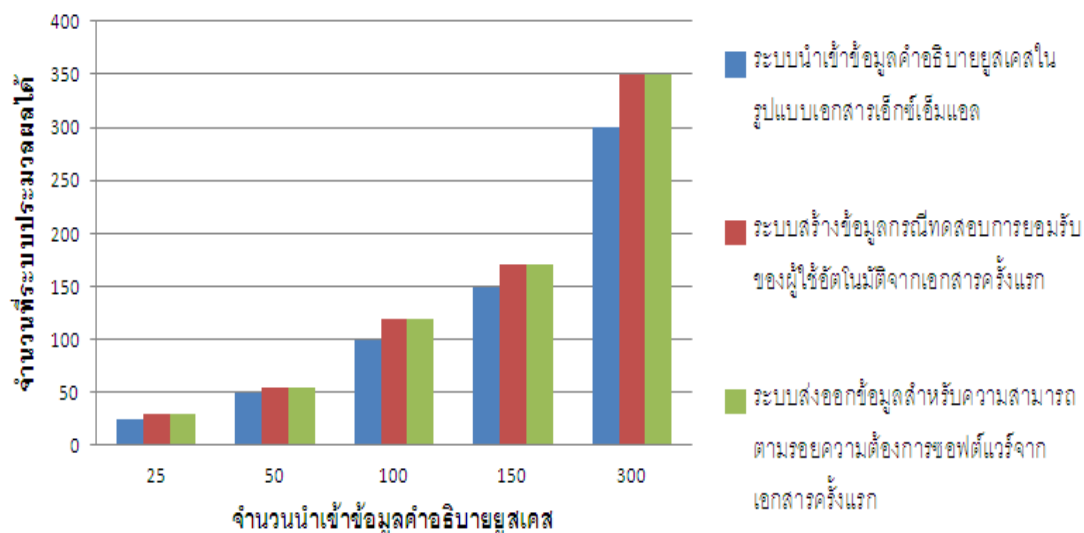
5.3 การประเมินผลระบบสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้

การประเมินผลงานวิจัยนี้ วัตถุประสงค์การสร้างกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติและการส่งออกข้อมูลสำหรับความสามารถตามรอยความต้องการซอฟต์แวร์ได้อย่างถูกต้องแม่นยำ และ ครบถ้วน ตรงตามเอกสารนำเข้ต้นทาง โดยได้ทำการทดสอบนำเข้ข้อมูลคำอธิบายยูสเคสในรูปแบบเอกสารเอ็กซ์เอ็มแอล จำนวน 25, 50, 125, 100 ตามลำดับ เพื่อสร้างกรณีทดสอบบนระบบและส่งออกข้อมูลสำหรับความสามารถตามรอยความต้องการซอฟต์แวร์ดังแสดงตามรายการบนคอลัมน์ของตารางที่ 23 โดยวัดตามตามรายการต่อไปนี้ และแสดงกราฟความแตกต่างของจำนวนนำเข้ข้อมูลคำอธิบายยูสเคสจากเอกสารครั้งแรก กับจำนวนที่ระบบสร้างข้อมูลกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติ ดังแสดงในภาพที่ 35

- ระบบนำเข้ข้อมูลคำอธิบายยูสเคสในรูปแบบเอกสารเอ็กซ์เอ็มแอล
- ระบบสร้างข้อมูลกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติจากเอกสารครั้งแรก
- ระบบส่งออกข้อมูลสำหรับความสามารถตามรอยความต้องการซอฟต์แวร์จากเอกสารครั้งแรก
- ระบบนำเข้ข้อมูลคำอธิบายยูสเคสในรูปแบบเอกสารเอ็กซ์เอ็มแอลเมื่อเอกสารเกิดการเปลี่ยนแปลง
- ระบบสร้างข้อมูลกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติเมื่อเอกสารเกิดการเปลี่ยนแปลง
- ระบบส่งออกข้อมูลสำหรับความสามารถตามรอยความต้องการซอฟต์แวร์จากการเปลี่ยนแปลง

ตารางที่ 23 วิธีการทำการทดลองใช้ในการเปรียบเทียบจำนวนนำเข้ข้อมูลคำอธิบายยูสเคสจากเอกสารครั้งแรก

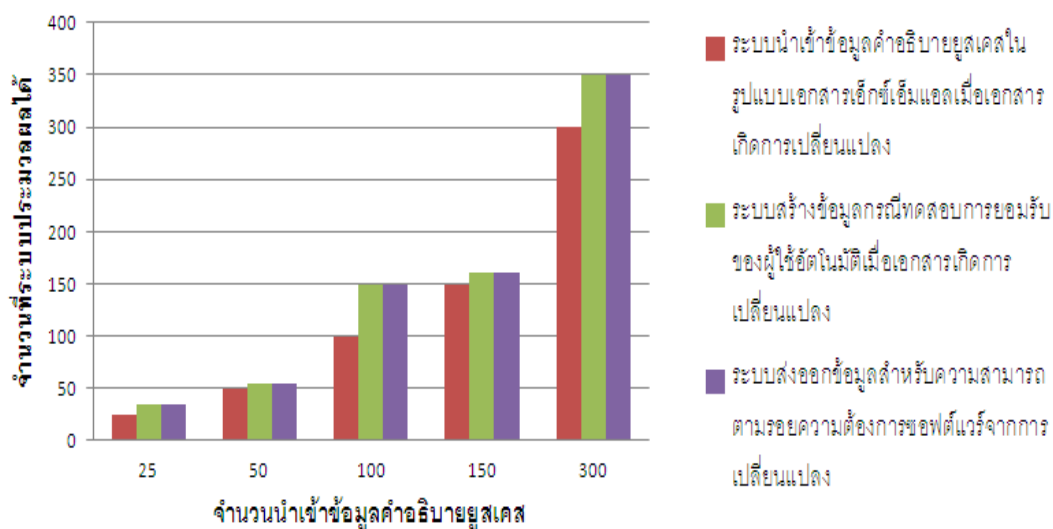
ลำดับที่	วิธีการทำการทดลอง	จำนวนนำเข้ข้อมูลคำอธิบายยูสเคส				
		25	50	100	150	300
1	ระบบนำเข้ข้อมูลคำอธิบายยูสเคสในรูปแบบเอกสารเอ็กซ์เอ็มแอล	25	50	100	150	300
2	ระบบสร้างข้อมูลกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติจากเอกสารครั้งแรก	30	54	120	170	350
3	ระบบส่งออกข้อมูลสำหรับความสามารถตามรอยความต้องการซอฟต์แวร์จากเอกสารครั้งแรก	30	54	120	170	350



ภาพที่ 35 กราฟความแตกต่างของจำนวนนำเข้าข้อมูลคำอธิบายยูสเคสจากเอกสารครั้งแรก กับจำนวนที่ระบบสร้างข้อมูลกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติ

ตารางที่ 24 วิธีการทำการทดลองใช้ในการเปรียบเทียบจำนวนนำเข้าข้อมูลคำอธิบายยูสเคสจากเมื่อเอกสารเกิดการเปลี่ยนแปลง

ลำดับที่	วิธีการทำการทดลอง	จำนวนนำเข้าข้อมูลคำอธิบายยูสเคส				
		25	50	100	150	300
1	ระบบนำเข้าข้อมูลคำอธิบายยูสเคสในรูปแบบเอกสารเอ็กซ์เอ็มแอลเมื่อเอกสารเกิดการเปลี่ยนแปลง	25	50	100	150	300
2	ระบบสร้างข้อมูลกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติเมื่อเอกสารเกิดการเปลี่ยนแปลง	30	54	120	170	350
3	ระบบส่งออกข้อมูลสำหรับความสามารถตามรอยความต้องการซอฟต์แวร์จากการเปลี่ยนแปลง	30	54	120	170	350



ภาพที่ 36 กราฟความแตกต่างของจำนวนนำเข้าข้อมูลคำอธิบายยูสเคสเมื่อเอกสารเกิดการเปลี่ยนแปลง กับจำนวนที่ระบบสร้างข้อมูลกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติ

โดยสรุปเชิงปริมาณงานที่ระบบสามารถสร้างกรณีทดสอบการยอมรับของผู้ใช้ได้ว่า การทำระบบการสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้สามารถสร้างกรณีทดสอบการยอมรับของผู้ใช้ และส่งออกข้อมูลสำหรับความสามารถตามรอยความต้องการซอฟต์แวร์ได้เท่ากับ หรือมากกว่าปริมาณจำนวนนำเข้าข้อมูลคำอธิบายยูสเคส

5.4 สรุปผลการทดลอง

จากผลการทดลองข้างต้นสรุปได้ว่า ระบบการสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้สามารถทำงานได้อย่างถูกต้องตามฟังก์ชันการทำงาน และช่วยลดทรัพยากร เวลา และแรงงานที่ใช้ในการสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้ โดยเฉพาะอย่างยิ่ง ในกรณีที่มีการเปลี่ยนแปลงความต้องการ

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

6.1 สรุปผลการวิจัย

ผลลัพธ์ที่ได้จากงานวิจัยนี้มีดังนี้

1. ได้วิธีการและระบบต้นแบบสำหรับสร้างกรณีทดสอบจากเอกสารข้อกำหนดความต้องการซอฟต์แวร์แบบอัตโนมัติ เพื่อสนับสนุนการทดสอบการยอมรับของผู้ใช้
2. ได้ข้อมูลสนับสนุนการสร้างเมตริกซ์ตามรอยความต้องการโดยอัตโนมัติระหว่างยูสเคสกับกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้

งานวิจัยนี้ได้นำเสนอแนวทางการสร้างชุดกรณีทดสอบการยอมรับของผู้ใช้จากคำอธิบายยูสเคสแบบอัตโนมัติ และได้พัฒนาระบบให้อยู่ในรูปแบบซอฟต์แวร์แอปพลิเคชัน (Software Application) ที่สามารถทำงานตอบสนองความต้องการสร้างกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้ได้จริง โดยมีการจัดเก็บข้อมูล เพื่อความสะดวก รวดเร็ว ประหยัดทรัพยากรในการดำเนินงานที่เกี่ยวข้องกับกระบวนการดังกล่าว พร้อมทั้งให้ข้อมูลกรณีทดสอบสำหรับการทดสอบการยอมรับของผู้ใช้ที่ถูกต้อง ทำให้ค่าใช้จ่ายการทดสอบลดลง และสามารถจัดการการเปลี่ยนแปลงความต้องการได้ดียิ่งขึ้น จึงเหมาะสำหรับระบบที่มีการเปลี่ยนแปลงข้อกำหนดความต้องการบ่อยๆ หรือที่มีข้อมูลจำนวนมาก และระบบสามารถเป็นส่วนหนึ่งของระบบเก็บเกี่ยวความต้องการ สามารถส่งออกความสัมพันธ์ระหว่างยูสเคสกับกรณีทดสอบความต้องการของยูสเคสนั้นไปยังเมตริกซ์ความสามารถตามรอยความต้องการโดยอัตโนมัติ

6.2 ข้อจำกัด

จากการดำเนินงานวิจัย ปัญหาและข้อจำกัดที่พบ คือ ถ้ามีการผิดพลาดในการนำเข้าเอกสารผิดพลาด หรือระบุไม่ตรงตามเงื่อนไขที่กำหนด จะทำให้ระบบไม่สามารถได้ข้อมูลเพื่อนำไปสร้างกรณีทดสอบการยอมรับของผู้ใช้ตามที่ต้องการได้

6.3 แนวทางการวิจัยต่อ

สำหรับการวิจัยในอนาคตสามารถหาวิธีอื่นที่สามารถสร้างชุดกรณีทดสอบการยอมรับของผู้ใช้จากคำอธิบายยูสเคสแบบอัตโนมัติได้แม่นยำและครอบคลุมกรณีทดสอบการยอมรับของผู้ใช้มากยิ่งขึ้น และจำเป็นต้องให้ความสนใจการปรับปรุงให้ระบบมีความฉลาด จากการเรียนรู้การสร้างชุดกรณีทดสอบ

รายการอ้างอิง

- [1] IEEE. IEEE std. 830-1998 Recommended Practice for Software Requirements Specifications[Online].1998. Available from : www.ieee.org
[2006,September 1]
- [2] Paul C.Jorgensen.Software Testing A Craftsman's Approach.3rded.
: Auerbach Publications,2008.
- [3] World Wide Web Consortium (W3C). XML Schema [Online]. 2005. Available
from : <http://www.3c.org/TR/xmlschema-0> [2011,Feb 4]
- [4] Object Management group.Unified Modeling Language: Superstructure version 2.3
[Online]. 2010. Available from : <http://www.omg.org> [2011, Feb 4]
- [5] เพชรน้ำค้าง วงษ์ศุทธิภากร. การสร้างกรณีทดสอบจื๊อจากยูเอ็มแอล. วิทยานิพนธ์ปริญญา
มหาบัณฑิต, สาขาวิชาวิศวกรรมซอฟต์แวร์ คณะวิศวกรรมศาสตร์
จุฬาลงกรณ์มหาวิทยาลัย,2548.
- [6] Cle´mentine Nebut, Franck Fleurey, Yves Le Traon, Member, IEEE, and Jean-Marc
Je´ze´quel, Member, IEEE. Automatic Test Generation:A Use Case Driven
Approach.IEEE Transactios on Software Engineering,Vol,32,No.3,March,
2006
- [7] A. Sinha, A. Paradkar, and C. Williams. On Generating EFSM Models from Use
Cases. Proceedings of the Sixth International Workshop on Scenarios and
State Machines (SCESM'07: ICSE Workshops 2007), Minneapolis, MN, USA,
May 20-26, 2007
- [8] สุรเชษฐ์ สุรย์ส่องธานี. การขยายความสามารถตามรอยความต้องการด้วย พฤติกรรมของ
ความสัมพันธ์ระหว่างความต้องการ. วิทยานิพนธ์ปริญญาามหาบัณฑิต, สาขาวิชา
วิทยาศาสตร์คอมพิวเตอร์ ภาควิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
จุฬาลงกรณ์มหาวิทยาลัย. 2553.
- [9] Y. Phopan, and Y. Limpiyakorn, Approach to Automating Input Data for
Requirements Traceability Matrix. Proceedings of the National Graduate
Research Conference 2011, Buriram, Thailand, Aug 11, 2011

ภาคผนวก

ภาคผนวก ก.
พจนานุกรมข้อมูลของระบบฐานข้อมูล

ตารางที่ 25 รายชื่อตารางทั้งหมด

ลำดับ	ชื่อตาราง	คำอธิบาย
1	tbl_importTmp	เป็นตารางเก็บข้อมูลชั่วคราว ใช้เพื่อรับข้อมูลจากเอกสาร XML
2	tbl_uc_master	เป็นตารางเก็บข้อมูลยูสเคส
3	tbl_normal_flow	เป็นตารางเก็บข้อมูลลำดับขั้นตอนการทำงานปกติ
4	tbl_alternate_flow	เป็นตารางเก็บข้อมูลลำดับขั้นตอนการทำงานที่ไม่ปกติ (alternation / exception flow)
5	tbl_uat_master	เป็นตารางเก็บข้อมูลกรณีทดสอบการยอมรับของผู้ใช้
6	tbl_uat_detail	เป็นตารางเก็บรายละเอียดข้อมูลกรณีทดสอบการยอมรับของผู้ใช้
7	tbl_efsm_state	เป็นตารางเก็บข้อมูล efsm state
8	tbl_efsm_state_transition	เป็นตารางเก็บข้อมูล efsm state transition
9	tbl_project_uat	เป็นตารางเก็บข้อมูล Project

ตารางที่ 26 พจนานุกรมข้อมูล ตาราง TBL_PROJECT_UAT

ตาราง: tbl_project_uat				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูล Project				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
1	project id	varchar	50	หมายเลขประจำตัวของโปรเจค
2	project seq	varchar	20	หมายเลขลำดับโปรเจค
3	project_name	varchar	20	ชื่อโปรเจค
Primary Key : project id				
Foreign Key : -				

ตารางที่ 27 พจนานุกรมข้อมูล ตาราง TBL_IMPORTTMP

ตาราง: tbl_importTmp				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูลชั่วคราว ใช้เพื่อรับข้อมูลจากเอกสาร XML				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
1	useCaseName	varchar	50	ชื่อยูสเคส
2	useCaseType	varchar	20	ประเภทของยูสเคส โดยระบุว่า "ภาพรวม" หรือ "เชิงละเอียด" (Overview/Detail) และ "พื้นฐาน" หรือ "จริง" (Essential/Real)
3	importanceLevel	varchar	20	ระดับความสำคัญของยูสเคส เมื่อเปรียบเทียบกับยูสเคสอื่นในระบบเดียวกัน มีไว้เพื่อช่วยในการจัดลำดับความสำคัญเร่งด่วนของงานที่ต้องทำ เช่น สูง/ปานกลาง/ต่ำ
4	relatedRequirement	varchar	20	ความสัมพันธ์ระหว่างยูสเคสกับความต้องการ
5	primaryActor	varchar	20	แสดงชื่อบุคคล หรือระบบที่จะก่อให้เกิดการดำเนินงานตามยูสเคส
6	briefDescription	varchar	200	เป็นข้อความสาระสำคัญของยูสเคส ซึ่งโดยทั่วไปจะเป็นการอธิบายสั้นๆ เพื่อให้เข้าใจว่ายูสเคสนี้ให้บริการใด แก่ผู้ใดบ้าง และจะมีการใช้งานเมื่อใด
7	trigger	varchar	200	เป็นข้อความแสดงเหตุการณ์ที่ก่อให้เกิดการดำเนินงานตามยูสเคส สามารถจำแนกได้เป็น สองประเภท สิ่งกระตุ้นภายนอก และสิ่งกระตุ้นตามเวลา
8	type	varchar	20	ประเภทของยูสเคส โดยระบุว่า "ภาพรวม" หรือ "เชิงละเอียด" (Overview/Detail) และ "พื้นฐาน" หรือ "จริง" (Essential/Real)
9	association	varchar	20	ความสัมพันธ์ระหว่างยูสเคส และผู้กระทำแบบเกี่ยวเนื่อง

ตาราง: tbl_importTmp				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูลชั่วคราว ใช้เพื่อรับข้อมูลจากเอกสาร XML				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
10	include	varchar	20	เป็นการระบุความสัมพันธ์แบบรวมระหว่างยูสเคส
11	extend	varchar	20	เป็นการระบุความสัมพันธ์แบบขยายระหว่างยูสเคส
12	generalization	varchar	20	ความสัมพันธ์ระหว่างยูสเคส กับยูสเคสที่มีการรับผิดชอบต่อคุณสมบัติ
13	normalFlowOfEvents	varchar	200	เป็นข้อความแสดงลำดับขั้นตอนการทำงานของแต่ละยูสเคส
14	alternateFlows	varchar	200	เป็นข้อความแสดงเงื่อนไขที่ทำให้เกิดเหตุการณ์อื่น ที่อาจเกิดขึ้นได้ในบางกรณี
15	preCondition	varchar	200	เงื่อนไขก่อนหน้า เพื่อใช้บอกเหตุการณ์ที่ต้องเกิดขึ้นก่อนจะเกิดเหตุการณ์ในยูสเคสจะสามารถเกิดขึ้นได้
16	postCondition	varchar	200	เงื่อนไขตามหลัง เพื่อใช้บอกเหตุการณ์ที่เกิดขึ้นหลังจากทำยูสเคสสำเร็จ
17	StakeholdersAndInterests	varchar	200	เป็นข้อความแสดงชื่อบุคคล หรือระบบที่ได้รับผลประโยชน์จากการทำงานของยูสเคส ได้แก่ ผู้กระทำหลัก หรือ อาจเป็นผู้ที่เกี่ยวข้องกับยูสเคส
18	project id	varchar	20	หมายเลขลำดับโปรเจค
19	project_name	varchar	20	ชื่อโปรเจค
20	table_num	varchar	20	หมายเลขบอกลำดับตาราง
21	data id_relate	varchar	20	หมายเลขลำดับความสัมพันธ์กับข้อมูล
Primary Key : -				
Foreign Key : -				

ตารางที่ 28 พจนานุกรมข้อมูล ตาราง TBL_UC_MASTER

ตาราง: tbl_uc_master				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูลยูสเคส				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
1	project id	varchar	20	หมายเลขลำดับโปรเจค
2	useCaseID	varchar	50	หมายเลขยูสเคส
3	useCaseName	varchar	50	ชื่อยูสเคส
4	useCaseType	varchar	20	ประเภทของยูสเคส โดยระบุว่า "ภาพรวม" หรือ "เชิงละเอียด" (Overview/Detail) และ "พื้นฐาน" หรือ "จริง" (Essential/Real)
5	importanceLevel	varchar	20	ระดับความสำคัญของยูสเคส เมื่อ เปรียบเทียบกับยูสเคสอื่นในระบบ เดียวกัน มีไว้เพื่อช่วยในการจัดลำดับ ความสำคัญเร่งด่วนของงานที่ต้องทำ เช่น สูง/ปานกลาง/ต่ำ
6	relatedRequirement	varchar	20	ความสัมพันธ์ระหว่างยูสเคสกับความ ต้องการ
7	primaryActor	varchar	20	แสดงชื่อบุคคล หรือระบบที่จะ ก่อให้เกิดการดำเนินงานตามยูสเคส
8	briefDescription	varchar	200	เป็นข้อความสาระสำคัญของยูสเคส ซึ่ง โดยทั่วไปจะเป็นการอธิบายสั้นๆ เพื่อให้ เข้าใจว่ายูสเคสนี้ให้บริการใด แก่ผู้ ใดบ้าง และจะมีการใช้งานเมื่อใด
9	trigger	varchar	200	เป็นข้อความแสดงเหตุการณ์ที่ ก่อให้เกิดการดำเนินงานตามยูสเคส สามารถจำแนกได้เป็น สองประเภท สิ่ง กระตุ้นภายนอก และสิ่งกระตุ้นตาม เวลา

ตาราง: tbl_uc_master				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูลยูสเคส				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
10	type	varchar	20	ประเภทของยูสเคส โดยระบุว่า "ภาพรวม" หรือ "เชิงละเอียด" (Overview/Detail) และ "พื้นฐาน" หรือ "จริง" (Essential/Real)
11	association	varchar	20	ความสัมพันธ์ระหว่างยูสเคส และ ผู้กระทำแบบเกี่ยวเนื่อง
12	include	varchar	20	เป็นการระบุความสัมพันธ์แบบรวม ระหว่างยูสเคส
13	extend	varchar	20	เป็นการระบุความสัมพันธ์แบบขยาย ระหว่างยูสเคส
14	generalization	varchar	20	ความสัมพันธ์ระหว่างยูสเคส กับยูสเคส ที่มีการรับทอดคุณสมบัติ
15	preCondition	varchar	200	เงื่อนไขก่อนหน้า เพื่อใช้บอกเหตุการณ์ ที่ต้องเกิดขึ้นก่อนจะเกิดเหตุการณ์ในยูส เคสจะสามารถเกิดขึ้นได้
16	postCondition	varchar	200	เงื่อนไขตามหลัง เพื่อใช้บอกเหตุการณ์ ที่เกิดขึ้นหลังจากทำยูสเคสสำเร็จ
17	StakeholdersAndInterests	varchar	200	เป็นข้อความแสดงชื่อบุคคล หรือระบบ ที่ได้รับผลประโยชน์จากการทำงานของ ยูสเคส ได้แก่ ผู้กระทำหลัก หรือ อาจ เป็นผู้ที่เกี่ยวข้องกับยูสเคส
Primary Key : project id,useCaseID				
Foreign Key : -				

ตารางที่ 29 พจนานุกรมข้อมูล ตาราง TBL_NORMAL_FLOW

ตาราง: tbl_normal_flow				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูลลำดับขั้นตอนการทำงานปกติ				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
1	project id	varchar	20	หมายเลขลำดับโปรเจค
2	useCaseID	varchar	50	หมายเลขยูสเคส
3	n_flowSeq	int	5	ลำดับที่ของลำดับขั้นตอนการทำงานปกติ
4	normalFlowOfEvents	varchar	200	เป็นข้อความแสดงลำดับขั้นตอนการทำงานของแต่ละยูสเคส
5	a_condition	varchar	200	ความสัมพันธ์ระหว่างลำดับขั้นตอนการทำงานปกติ กับขั้นตอนไม่ปกติ (alternative / exception flow)
6	e_condition	varchar	200	ความสัมพันธ์ระหว่างลำดับขั้นตอนการทำงานปกติ กับขั้นตอนส่วนขยาย (extendtion point)
7	input	varchar	200	พารามิเตอร์นำเข้า
8	output	varchar	200	ผลลัพธ์ของพารามิเตอร์
Primary Key : project id,useCaseID,table_num				
Foreign Key : -				

ตารางที่ 30 พจนานุกรมข้อมูล ตาราง TBL_ALTERNATE_FLOW

ตาราง: tbl_alternate_flow				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูลลำดับขั้นตอนการทำงานที่ไม่ปกติ (alternation / exception flow)				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
1	project id	varchar	20	หมายเลขลำดับโปรเจค
2	useCaseID	varchar	50	หมายเลขยูสเคส
3	a_condition	varchar	200	ความสัมพันธ์ระหว่างลำดับขั้นตอนการทำงานปกติ

ตาราง: tbl_altermate_flow				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูลลำดับขั้นตอนการทำงานที่ไม่ปกติ (alternation / exception flow)				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
4	a_conditionFlows	varchar	200	เป็นข้อความแสดงเงื่อนไข
5	a_flowSeq	int	5	ลำดับที่ของลำดับขั้นตอนการทำงานไม่ปกติ
6	altenateFlows	varchar	200	เป็นข้อความแสดงเงื่อนไขที่ทำให้เกิดเหตุการณ์อื่น ที่อาจเกิดขึ้นได้ในบางกรณี
7	input	varchar	200	พารามิเตอร์นำเข้า
8	output	varchar	200	ผลลัพธ์ของพารามิเตอร์
Primary Key : project id,useCaseID,table_num,a_condition				
Foreign Key : -				

ตารางที่ 31 พจนานุกรมข้อมูล ตาราง TBL_UAT_MASTER

ตาราง: tbl_uat_master				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูลกรณีทดสอบการยอมรับของผู้ใช้				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
1	project id	varchar	20	หมายเลขลำดับโปรเจค
2	uat_id	varchar	50	หมายเลขกรณีทดสอบ
3	uat_flowtype	varchar	50	ประเภทของขั้นตอนการทำงาน
4	relatedUsecase	varchar	20	ความสัมพันธ์ระหว่างกรณีทดสอบกับความต้องการ
5	relatedRequirement	varchar	20	ความสัมพันธ์ระหว่างกรณีทดสอบกับความต้องการ
6	preCondition	varchar	200	เงื่อนไขก่อนหน้า เพื่อใช้บอกเหตุการณ์ที่ต้องเกิดขึ้นก่อนจะเกิดเหตุการณ์ในยูสเคสจะสามารถเกิดขึ้นได้
7	post_cond	varchar	200	เงื่อนไขตามหลัง เพื่อใช้บอกเหตุการณ์ที่เกิดขึ้นหลังจากทำยูสเคสสำเร็จ

ตาราง: tbl_uat_master				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูลกรณีทดสอบการยอมรับของผู้ใช้				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
Primary Key : project id, uat_id				
Foreign Key : relatedUsecase, relatedRequirement				

ตารางที่ 32 พจนานุกรมข้อมูล ตาราง TBL_UAT_DETAIL

ตาราง: tbl_uat_detail				
คำอธิบายตาราง : เป็นตารางเก็บรายละเอียดข้อมูลกรณีทดสอบการยอมรับของผู้ใช้				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
1	project id	varchar	20	หมายเลขลำดับโปรเจค
2	uat_id	varchar	50	หมายเลขกรณีทดสอบ
3	uat_scenario_no	int	5	ลำดับเหตุการณ์กรณีทดสอบ
4	uat_scenario	varchar	200	รายละเอียดเหตุการณ์กรณีทดสอบ
5	input	varchar	200	พารามิเตอร์นำเข้า
6	output	varchar	200	ผลลัพธ์ของพารามิเตอร์
Primary Key : project id, uat_id				
Foreign Key : relatedUsecase, relatedRequirement				

ตารางที่ 33 พจนานุกรมข้อมูล ตาราง TBL_EFSM_STATE

ตาราง: tbl_efs_state				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูล efsm state				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
1	project id	varchar	20	หมายเลขลำดับโปรเจค
2	useCaseID	varchar	50	หมายเลขยูสเคส
3	efs_state_id	varchar	20	หมายเลขลำดับความสัมพันธ์กับข้อมูล
4	FlowOfEvents	varchar	200	เป็นข้อความแสดงเหตุการณ์
5	input	varchar	200	พารามิเตอร์นำเข้า
6	output	varchar	200	ผลลัพธ์ของพารามิเตอร์ (block)

ตาราง: tbl_efsm_state				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูล efsm state				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
Primary Key : project id,useCaseID,efsm_state_id				
Foreign Key : -				

ตารางที่ 34 พจนานุกรมข้อมูล ตาราง TBL_EFSM_STATE_TRANSITION

ตาราง: tbl_efsm_state_transition				
คำอธิบายตาราง : เป็นตารางเก็บข้อมูล efsm state transition				
ลำดับ	ชื่อวัตถุ	ชนิดข้อมูล	ขนาดข้อมูล	ความหมาย
1	project id	varchar	20	หมายเลขลำดับโปรเจค
2	useCaseID	varchar	50	หมายเลขยูสเคส
3	state_trans_id	varchar	20	หมายเลขสถานะการเปลี่ยนแปลง (ลำดับ)
4	source_state	varchar	20	หมายเลขสถานะต้นทาง
5	dest_state	varchar	20	หมายเลขสถานะปลายทาง
6	a_condition	varchar	200	ความสัมพันธ์ระหว่างลำดับขั้นตอนการทำงานปกติ (predicate)
7	FlowOfEvents	varchar	200	เป็นข้อความแสดงเหตุการณ์
8	input	varchar	200	พารามิเตอร์นำเข้า (predicate)
9	output	varchar	200	ผลลัพธ์ของพารามิเตอร์ (block)
Primary Key : project id,useCaseID,state_trans_id				
Foreign Key : source_state,dest_state				

ภาคผนวก ข.

ตัวอย่างแสดงการทำงานกรณีทดสอบ

รายละเอียดขั้นตอนการทำงานกรณีทดสอบ สามารถทำตามขั้นตอนได้ดังนี้

1. เข้าไปที่ระบบนำเข้าข้อมูลอัตโนมัติ [7] จัดสร้างเอกสารให้ตรงตามข้อกำหนดและเงื่อนไขการทำงานของระบบนำเข้าข้อมูลอัตโนมัติ โดยในข้อกำหนดความต้องการซอฟต์แวร์ในรูปแบบเอกสาร docx จะสนใจในหัวข้อ Functional Requirements ดังตัวอย่างในภาพที่ 360 ,Non-Functional Requirements ดังตัวอย่างในภาพที่ 370 และ Use Case Description ดังตัวอย่างในภาพที่ 37

4. Functional Requirements		
Functional Requirements	Related Requirements	Primary Actors
FR01: Setup	FR10, NR06, NR07	AC02, AC03
FR02: การ Login เข้าสู่ระบบ	FR09, NR01, NR09	AC01, AC02, AC03
FR03: รายการเอกสารป่าสง	FR10, FR09, NR09	AC01
FR04:งานเตรียมเอกสารการ	FR10, FR09, NR03, NR09	AC01
FR05:งานประชาสัมพันธ์	FR10, FR09, NR03, NR09	AC01

ภาพที่ 37 ตัวอย่างข้อมูลในหัวข้อ Functional Requirements

5. Non-functional Requirements		
Non-Functional Requirements	Related Requirements	Primary Actors
NR01: ระบบสามารถรองรับจำนวนผู้ใช้ระบบงานได้จำนวน 400 Users		AC03
NR02: สามารถใช้งานได้พร้อมๆกันที่จำนวน 250 Users		AC03
NR03: System response ภายใน 15 วินาที		AC02,AC03
NR04 :จัดแบ่งเป็นกลุ่มตามลักษณะการปฏิบัติงาน	NR06	AC02,AC03

ภาพที่ 38 ตัวอย่างข้อมูลในหัวข้อ Non-Functional Requirements

3.4 Use Case Description	
Use Case Name:	UC01: พิมพ์รายงาน
Use case type:	Real Use Case
Importance level:	High
Related Requirement:	FR02
Primary actor:	AC01
Brief description:	เป็นการพิมพ์รายงานเพื่อใช้ตรวจสอบข้อมูลจากการปฏิบัติกรและเพื่อการบริหาร
Trigger:	-
Type:	-
Association:	-
Include:	UC01.1,UC01.2
Extend:	-
Generalization:	-
Normal flow of Events:	1. User เลือกทำข้อ 2 และ 3 โดยไม่เรียงลำดับ 2. User เลือกทำ UC01.1 3. User เลือกทำ UC01.2
Alternate flows:	
Exceptional flows:	-
Pre-condition	-
Post-Condition	ระบบสามารถพิมพ์รายงานเพื่อใช้ตรวจสอบข้อมูลจากการปฏิบัติกรและเพื่อการบริหารได้สำเร็จ
Use Case Name:	UC01.1: พิมพ์รายงานเพื่อการบริหาร
Use case type:	Real Use Case
Importance level:	High
Related Requirement:	FR02
Primary actor:	AC01

ภาพที่ 39 ตัวอย่างข้อมูลในหัวข้อ Use Case Description

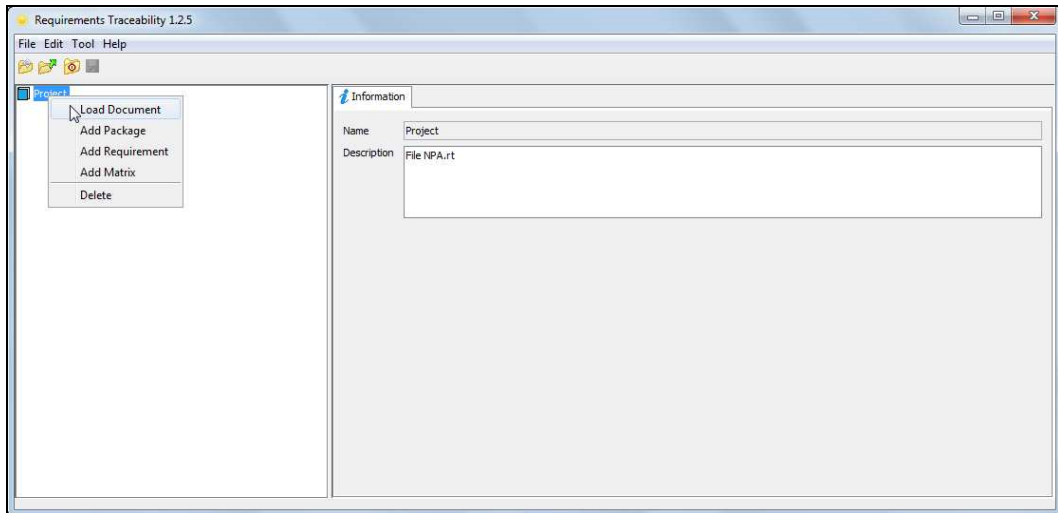
2. เมื่อมีเอกสารที่ตรงตามข้อกำหนดและเงื่อนไขแล้วทำการเปิดโปรแกรมระบบนำเข้าข้อมูลอัตโนมัติขึ้นมาจะได้ดังภาพที่ และทำการสร้างโครงการตามปกติ



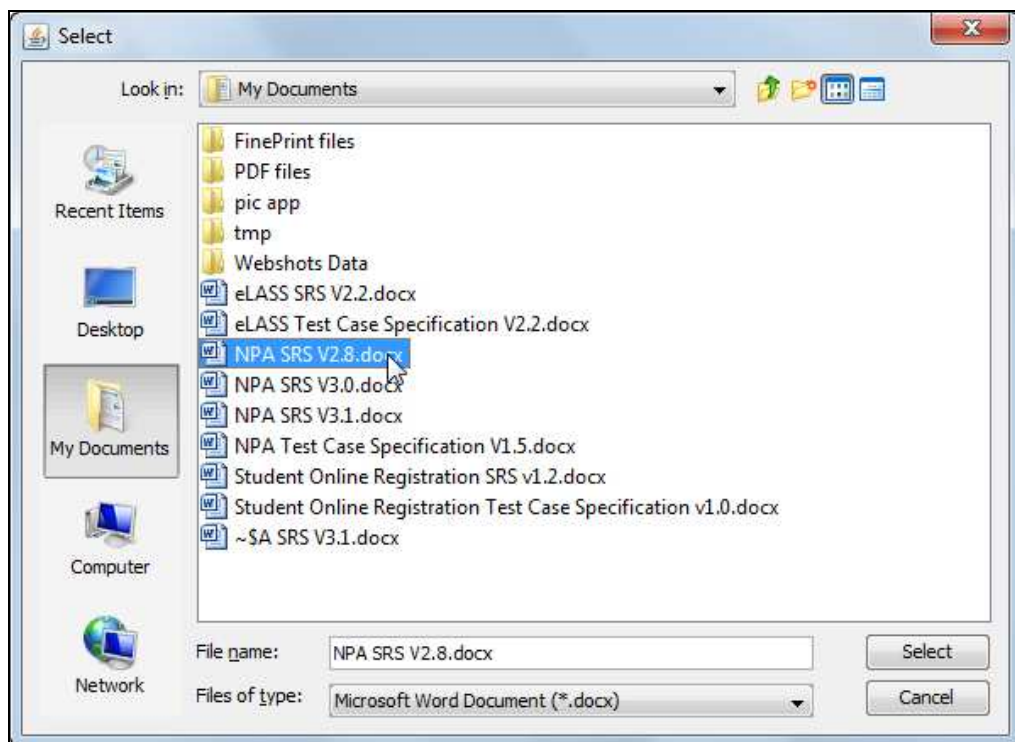
ภาพที่ 40 ตัวอย่างโปรแกรมระบบนำเข้าข้อมูลอัตโนมัติ

3. จากนั้นทำการคลิกขวาที่ชื่อโครงการและกด "Load Document" ดังภาพที่ 40 จากนั้นระบบก็จะทำให้มีไดอะล็อกให้เลือกไฟล์เอกสาร (.docx) เข้าสู่ระบบโดยจะเลือกข้อกำหนดความ

ต้องการซอฟต์แวร์ของโครงการนี้เมื่อทำการเลือกแล้ว ถ้าไม่เคยกำหนดค่าเอกสารเข้ามาก่อนจะมีไดอะล็อกให้เข้าไปทำข้อกำหนดของแท็กดัง แต่ถ้ามีการเคยกำหนดค่าเอกสารเข้าแล้วจะมีไดอะล็อกบอกว่ามีไฟล์กฎของเก่า ระบบสามารถใช้อันเก่าได้ทันที



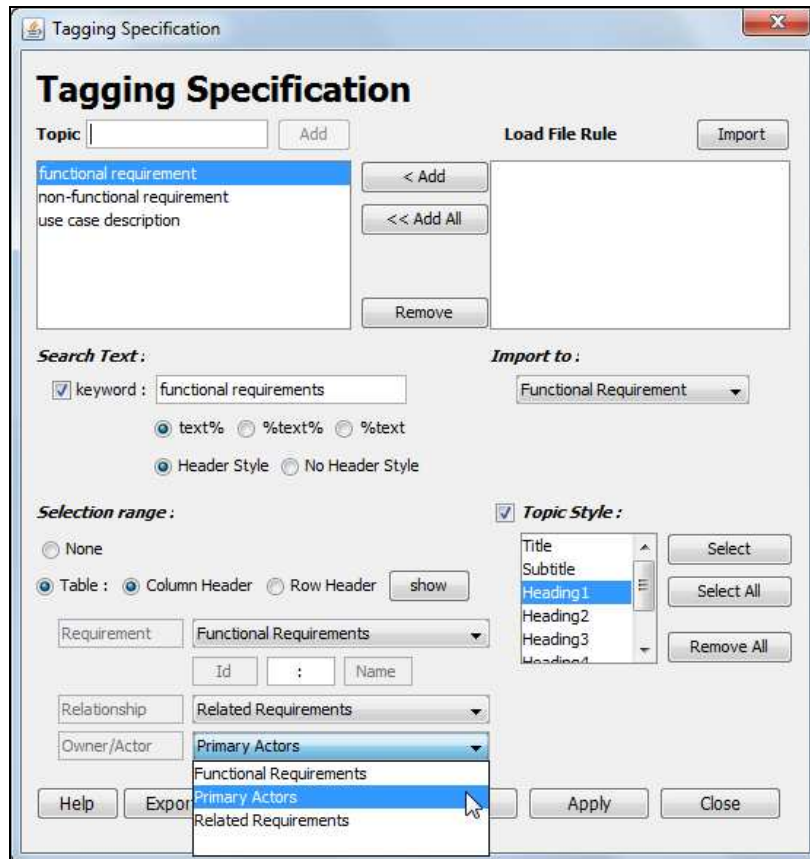
ภาพที่ 41 เมื่อทำการคลิกขวาที่โครงการ



ภาพที่ 42 ไดอะล็อกสำหรับเลือกเอกสารเข้าสู่ระบบ

4. ทำการกำหนดค่าต่างๆใน ไดอะล็อกสำหรับกำหนดค่าเอกสารการนำเข้าโดยผ่านทางข้อกำหนดของแท็กในภาพที่ 42

โดยจะต้องมีการระบุค่า เลือกขอบเขตของสิ่งที่ต้องการ เลือกว่าจะป็นเนื้อหาความต้องการประเภทใด และทำการกำหนดหัวข้อรูปแบบ เพื่อที่จะให้ครอบคลุมกับสิ่งที่ต้องการจากการสกัดข้อมูลเพื่อนำไปใช้ในแอปพลิเคชันอื่นๆอีกด้วย



ภาพที่ 43 ไดอะล็อกสำหรับกำหนดค่าเอกสารการนำเข้าโดยผ่านทางข้อกำหนดของแท็ก

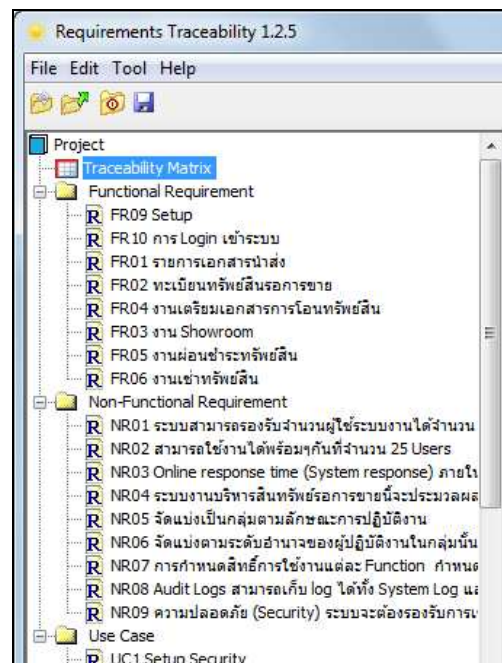
5. ทวนสอบผลลัพธ์ผ่านทางไดอะล็อกรายงานผลการสกัดข้อมูลอย่างย่อ กรณีผลลัพธ์ยังไม่ตรงกับที่คาดหวังให้กลับไปกำหนดค่าเอกสารการนำเข้าใหม่จนกว่าจะได้ผลลัพธ์ที่คาดหวังไว้ ถ้าผลลัพธ์เป็นไปตามที่คาดหวังให้กด “Accept”

id	name	relate	owner	import
FR03	งาน Showroom	FR10, FR09, NR03, NR09	AC01	Functional Requirement
FR05	งานผ่อนชำระทรัพย์สิน	FR10, FR09, NR03, NR09	AC01	Functional Requirement
FR06	งานเช่าทรัพย์สิน	FR10, FR09, NR03, NR09	AC01	Functional Requirement
NR01	ระบบสามารถรองรับจำนวนผู้ใช้ระบบ...		AC03	Non-Functional Requirement
NR02	สามารถใช้งานได้พร้อมกันที่จำนวน 25 Users		AC03	Non-Functional Requirement
NR03	Online response time (System ...		AC02,AC03	Non-Functional Requirement
NR04	ระบบงานบริหารสินทรัพย์รอการขาย...		AC03	Non-Functional Requirement
NR05	จัดแบ่งเป็นกลุ่มตามลักษณะการปฏิบัติงาน	NR07	AC02,AC03	Non-Functional Requirement
NR06	จัดแบ่งตามระดับอำนาจของผู้ปฏิบัติงาน...	NR01, NR02, NR03	AC03	Non-Functional Requirement
NR07	การกำหนดสิทธิ์การใช้งานแต่ละ...	NR05, NR06	AC02,AC03	Non-Functional Requirement
NR08	Audit Logs สามารถเก็บ log ได้ทั้ง...		AC02,AC03	Non-Functional Requirement
NR09	ความปลอดภัย (Security) ระบบจะต้อง...	NR05, NR06	AC02,AC03	Non-Functional Requirement
UC1	Setup Security	FR09	AC02,AC03	Use Case Description
UC2	Setup Statics Data	FR09	AC02,AC03	Use Case Description
UC3	Login	FR10	AC01	Use Case Description
UC4	บันทึกเอกสารนำส่ง	FR01	AC01	Use Case Description

Delete Old Item + Create Traceability Matrix First

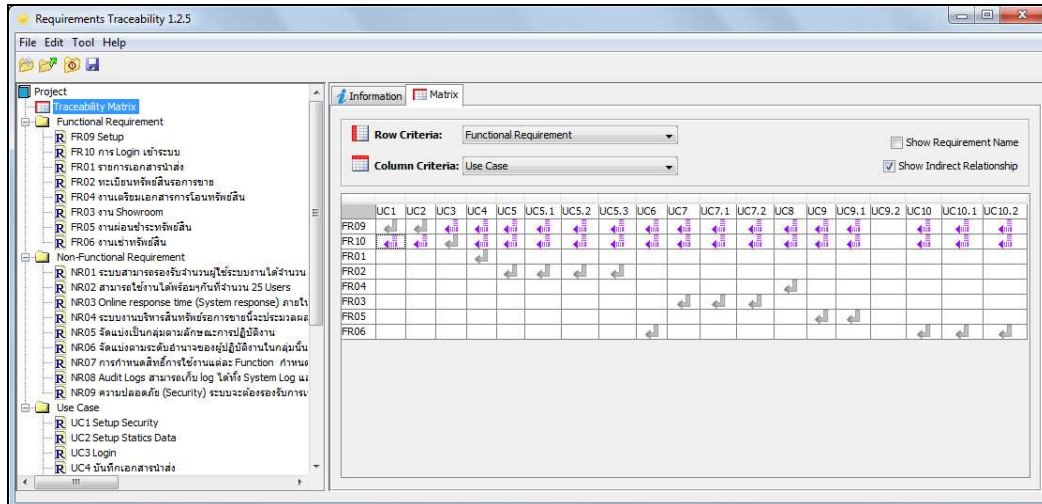
ภาพที่ 44 ได้อะลอรายงานผลการสกัดข้อมูลอย่างย่อ

6. ข้อมูลจะเข้าสู่โปรแกรมสร้างความสามารถตามรอยความต้องการได้ทันที ดังภาพที่ 45



ภาพที่ 45 หน้าจอส่วนต้นไม่นำทาง

7. ตรวจสอบเมตริกซ์ตามรอยโดยคลิกขวาที่โครงการและกดสร้างเมตริกซ์ ก็จะได้เมตริกซ์ตามรอย และสามารถเลือกกำหนดแกนหลัก และแกนแถวได้ตามต้องการ และเลือกดูความสัมพันธ์ทางอ้อมได้ด้วย ดังภาพที่ 46



ภาพที่ 46 โปรแกรมเมตริกซ์ความสามารถตามรอยความต้องการ

8. ถ้าต้องการนำข้อมูลนำไปใช้ภายนอกสามารถเลือกส่งออกมาเป็นเอ็กซ์เอ็มแอล โดยกดที่เมนู เครื่องมือ แล้วเลือก ส่งออกเอ็กซ์เอ็มแอลของระบบ หรือ ส่งออกเอ็กซ์เอ็มแอลให้ระบบภายนอก

8.1 ส่งออกเอ็กซ์เอ็มแอลโดยกดที่เมนูเครื่องมือแล้วเลือก “Export XML System” หรือ “Export XML every topic” เอ็กซ์เอ็มแอลที่ระบบส่งออกไปให้ภายนอกจะมีโครงสร้างของข้อมูลสามารถดูได้จากภาพที่ 47

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<project id="25540902110142" name="Software Requirement Specification 1.0">
  <package id="0" name="User Requirements"/>
  <package id="1" name="Product Feature"/>
  <package id="2" name="Functional Requirement">
  </package>
  <package id="4" name="Tests"/>
  <package id="5" name="Use Case Description"/>
    <data name="ID" t_id="TP255408310512270003" table_num="1">UC-ID0002:เข้าสู่ระบบ
    2</data>
    <data name="Normal flow of Events" t_id="TP255408310512270004" table_num="1">1.
    นักศึกษาเข้าสู่ระบบลงทะเบียนเรียน\๓1.1 ระบบแสดงหน้าจอให้กรอกรหัสนักศึกษา และ
    รหัสผ่าน\๓2. นักศึกษากรอกรหัสนักศึกษา และรหัสผ่าน\๓2.1 ระบบตรวจสอบความถูกต้องของรหัส
    นักศึกษาและรหัสผ่าน\๓2.2 รหัสนักศึกษาและรหัสผ่านถูกต้อง ระบบตรวจสอบสถานะการเป็นนักศึกษา
    ของนักศึกษา\๓3. นักศึกษาเข้าสู่ระบบสำเร็จ และอยู่ในสถานะเข้าสู่ระบบ\๓3.1 ระบบแสดงหน้าจอ
    ลงทะเบียน</data>
    <data name="Trigger" t_id="TP255408310512270004" table_num="1">นักศึกษากรอกรหัส
    นักศึกษา และรหัสผ่านให้ถูกต้อง</data>
    <data name="Relate" t_id="TP255408310512270004" table_num="1">FR-ID0002</data>
  <package id="6" name="User"/>
</project>
```

ภาพที่ 47 ตัวอย่างเอ็กซ์เอ็มแอลที่ระบบสร้างขึ้นให้ระบบภายนอก

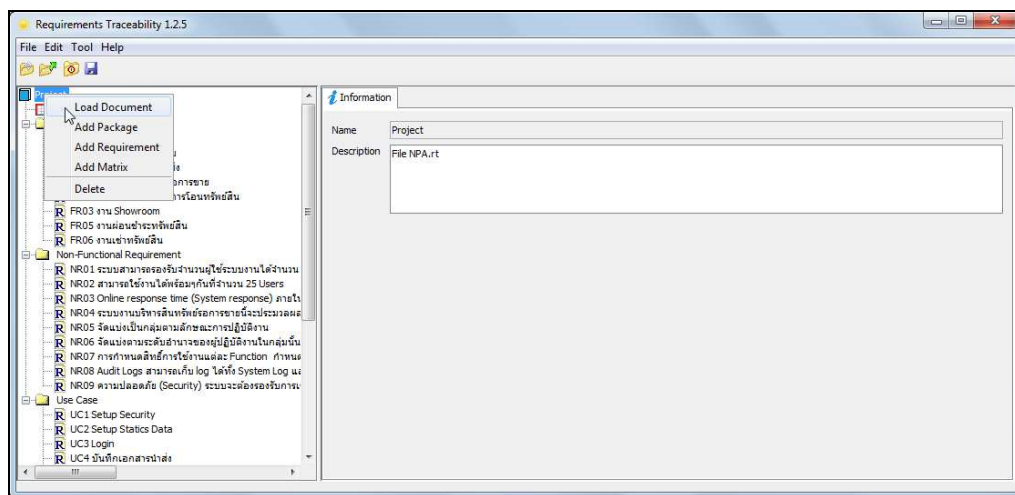
10. ระบบสร้างกรณีทดสอบการยอมรับของผู้ใช้อัตโนมัติจากเอกสารข้อกำหนดความ (งานวิจัยนี้) จัดสร้างเอกสารข้อกำหนดการทดสอบซอฟต์แวร์ให้ตรงตามข้อกำหนดและเงื่อนไข

การทำงานของระบบระบบนำเข้าข้อมูลอัตโนมัติ โดยในข้อกำหนดการทดสอบซอฟต์แวร์ในรูปแบบเอกสาร docx จะสนใจในหัวข้อ User Acceptance Test ดังตัวอย่างในภาพที่ 48

2. User Acceptance Test					
UAT No	Pre-Condition	Test Scenario	Expected Result	Related Requirement	Related Use-Case
UAT01	1.มีรายชื่อ User ที่จะกำหนดสิทธิ์ในระบบ 2.User ที่สามารถกำหนดค่าคงที่ควรได้รับสิทธิ์เป็น User Admin	1. ระบบแสดงรายชื่อ User 2. User Admin เลือกชื่อ User ที่จะกำหนดสิทธิ์ 3. User Admin กำหนดสิทธิ์การเข้าถึงข้อมูลของผู้ใช้งาน โดยแบ่งกลุ่มดังนี้ 3.1 กลุ่มผู้ใช้งานทั่วไปสามารถบันทึกและแก้ไขข้อมูลของตนเองได้ และสามารถดูข้อมูลของทั้งหมดได้ 3.2กลุ่มผู้บริหารสามารถเรียกดูข้อมูลได้ทั้งหมด และสามารถแก้ไขข้อมูลที่อยู่ภายใต้ความดูแลของตนเองได้ทั้งหมด 4. ระบบบันทึกข้อมูล	1.User สามารถกำหนด Security ในการเข้าถึงข้อมูลให้กับผู้ที่มีงานได้สำเร็จ	FR09	UC1
UAT02	User ที่สามารถกำหนดค่าคงที่ควรได้รับสิทธิ์เป็น User Admin	1. User Admin บอกรหัสข้อมูลประกอบด้วย - รายชื่อลูกหนี้ - สาขา - เขต - สาขางาน - ประเภทของทรัพย์สิน - รหัสเอกสาร - แหล่งที่มาของทรัพย์สิน 2. ระบบตรวจสอบความครบถ้วนของข้อมูล 3. ระบบบันทึกข้อมูล	1.User สามารถกำหนดค่าคงที่ที่จะเป็นข้อมูลในระบบทั้งหมดได้สำเร็จ	FR09	UC2

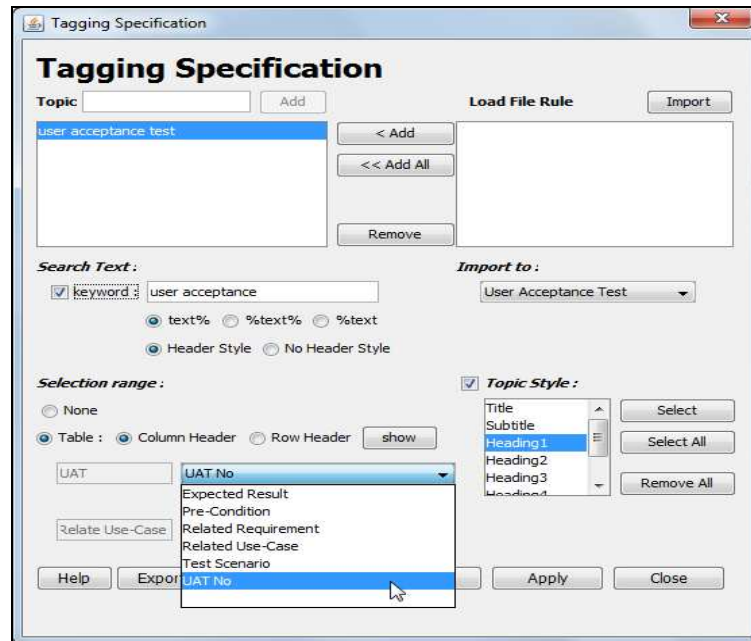
ภาพที่ 48 ตัวอย่างข้อมูลในหัวข้อ User Acceptance Test

11. จากนั้นทำการคลิกขวาที่ชื่อโครงการและกด “Load Document” ดังภาพที่ 49 จากนั้นระบบก็จะทำให้มีไดอะล็อกให้เลือกไฟล์เอกสาร (.docx) เข้าสู่ระบบโดยจะเลือกข้อกำหนดความต้องการซอฟต์แวร์ของโครงการนี้



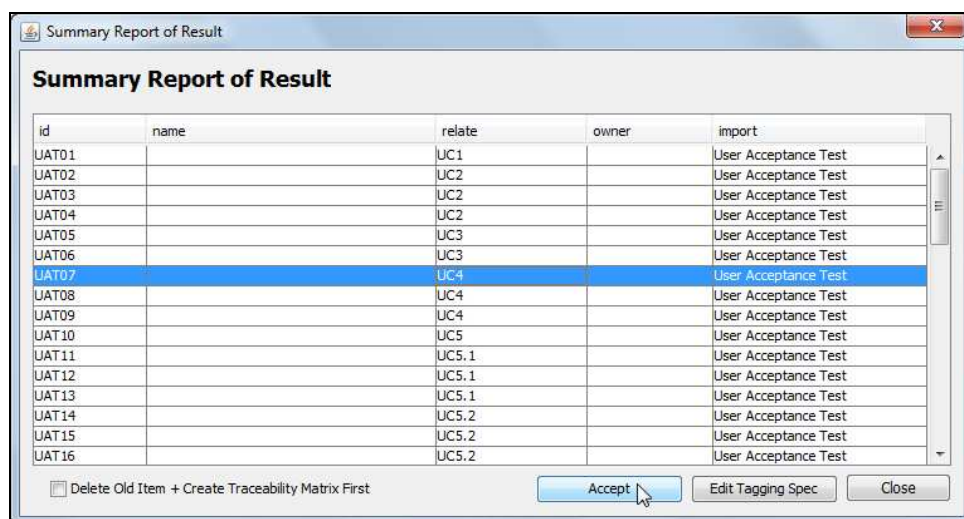
ภาพที่ 49 โหลดเอกสารข้อกำหนดการทดสอบซอฟต์แวร์

12. ทำการกำหนดค่าต่างๆใน ไดอะล็อกสำหรับกำหนดค่าเอกสารการนำเข้าโดยผ่านทางข้อกำหนดของแท็กใน ให้ครบตามข้อกำหนด แล้วเลือกประเภทนำเข้าเป็น User Acceptance Test ดังภาพที่ 50 จากนั้นกด “Apply”



ภาพที่ 50 ได้อะลอกสำหรับกำหนดค่าเอกสารการนำเข้าโดยผ่านทางข้อกำหนดของแท็ก ในส่วนเอกสารข้อกำหนดการทดสอบซอฟต์แวร์

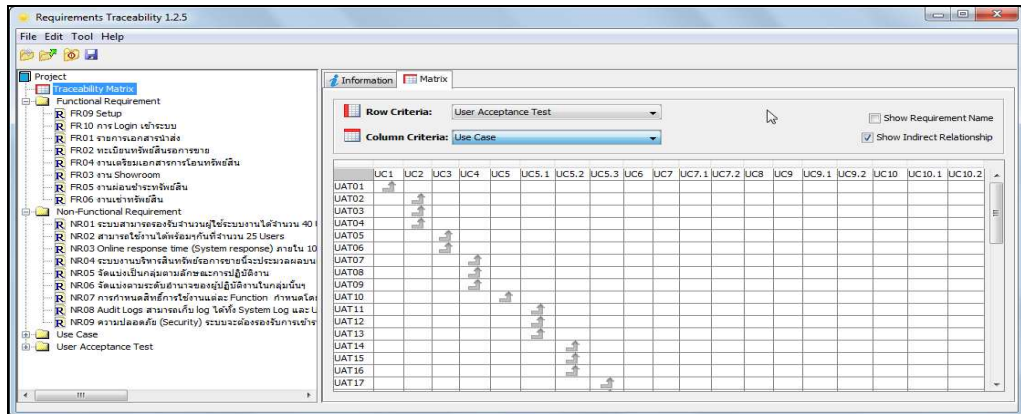
13. ทวนสอบผลลัพธ์ผ่านทางไดอะล็อกรายงานผลการสกัดข้อมูลอย่างย่อ กรณีผลลัพธ์ยังไม่ตรงกับที่คาดหวังให้กลับไปกำหนดค่าเอกสารการนำเข้าใหม่จนกว่าจะได้ผลลัพธ์ที่คาดหวังไว้ ถ้าผลลัพธ์เป็นไปตามที่คาดหวังให้กด “Accept” ดังภาพที่ 51



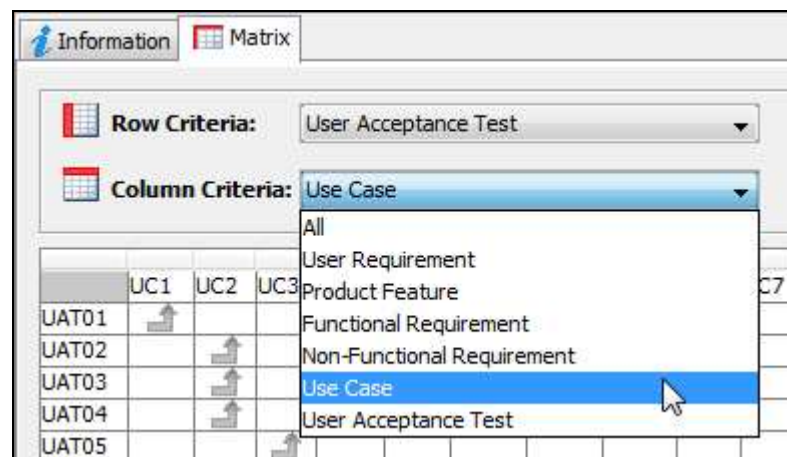
ภาพที่ 51 รายงานผลการสกัดอย่างย่อ

14. ตรวจสอบเมตริกซ์ตามรอยโดยคลิกขวาที่โครงการและกดสร้างเมตริกซ์ ก็จะได้เมตริกซ์ตามรอย และสามารถเลือกกำหนดแกนหลัก และแกนแถวได้ตามต้องการ และเลือกดูความสัมพันธ์ทางอ้อมได้ด้วย ดังภาพที่ 52 และเมตริกซ์ความสามารถตามรอย จะคำนวณให้โดยอัตโนมัติ และสามารถเลือกชนิดความต้องการในแนวแถวและแนวหลัก จากคอมโบบ็อกซ์ ในแท็บ

เมตริกซ์ในภาพที่ 54 ข้อมูลที่ได้เพิ่มจากความสามารถตามรอยความต้องการคือ ความสามารถตามรอยความต้องการทางอ้อม ซึ่งเมตริกซ์ความสามารถตามรอยในงานวิจัยจะคำนวณให้โดยอัตโนมัติ



ภาพที่ 52 โปรแกรมเมตริกซ์ความสามารถตามรอยความต้องการ



ภาพที่ 53 เลือกดูความสัมพันธ์ระหว่างแถวและหลัก

ประวัติผู้เขียนวิทยานิพนธ์

นายชวลิต เขี่ยมสะอาด เกิดเมื่อวันที่ 17 มกราคม พ.ศ. 2525 ที่จังหวัดชลบุรี สำเร็จ การศึกษาหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยรามคำแหง ในปีการศึกษา 2548 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตร มหาบัณฑิต สาขาวิชาวิศวกรรมซอฟต์แวร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2552