

ระบบสารสนเทศเพื่อติดตาม และรายงานข้อผิดพลาดหรือประเด็นการแก้ไขระบบ

DEFECT TRACKING SYSTEM



โดย

สุชาลัย ขาวบาง

SUCHALAI KHAWBANG



อาจารย์ที่ปรึกษา

รศ.ดร. พรฤดี เนติโสภาคกุล

2005

เลขหมู่.....
เลขทะเบียน.....
วัน,เดือน,ปี..... 2 ส.ย. 2557

b. 12608658
i.....

รายงานนี้เป็นส่วนหนึ่งของวิชาการศึกษาระดับ 2

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศและการจัดการ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2555

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DEFECT TRACKING SYSTEM



**A REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS OF THE COURSE**

INDEPENDENT STUDY 2

MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2 / 2012

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2013

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|------------------|---|
| หัวข้อ | ระบบสารสนเทศเพื่อติดตามและรายงานข้อผิดพลาดหรือประเด็นการแก้ไขระบบ |
| นักศึกษา | นางสาวสุชาลัย ขาวบาง |
| รหัสนักศึกษา | 54660801 |
| ปริญญา | วิทยาศาสตรมหาบัณฑิต |
| สาขาวิชา | เทคโนโลยีสารสนเทศ |
| แขนงวิชา | เทคโนโลยีสารสนเทศและการจัดการ |
| ปีการศึกษา | 2555 |
| อาจารย์ที่ปรึกษา | รศ.ดร. พรฤดี เนติโสภาคกุล |

บทคัดย่อ

เนื่องจาก ณ ปัจจุบันธนาคารอิสลามแห่งประเทศไทย กลุ่มงานเทคโนโลยีสารสนเทศใน ส่วนงานทดสอบระบบสารสนเทศ มีการทดสอบระบบต่าง ๆ มากมาย และเมื่อพบข้อผิดพลาดจาก การทดสอบระบบ (Defect) หรือประเด็น (Issue) การแก้ไขปรับเปลี่ยน โปรแกรมหรือพารามิเตอร์ ในโปรแกรมต่าง ๆ นั้น จะบันทึก Defect หรือ Issue ต่าง ๆ ลงใน Microsoft excel ซึ่งยากในการ บริหารจัดการ ติดตามและรายงานข้อผิดพลาดที่เกิดจากการทดสอบระบบ ดังนั้นจึงควรพัฒนา ระบบสารสนเทศเพื่อติดตาม และรายงานข้อผิดพลาดหรือประเด็นการแก้ไขจากการทดสอบระบบ สารสนเทศ (Defect Tracking System) เพื่อให้สามารถบริหารจัดการข้อผิดพลาดที่เกิดขึ้นได้ โดย แบ่งเป็นข้อผิดพลาดที่เกิดบนระบบ Production (ระบบงานจริง) และข้อผิดพลาดจากการทดสอบ ระบบก่อนขึ้นใช้งานจริง อีกทั้งระบบนี้จะเก็บประเด็นการปรับเปลี่ยน แก้ไข ต่างๆ ที่มีผลต่อระบบ อาทิเช่น Requirement / ประเด็นเพิ่มเติมจาก User ทำให้ง่ายในการบริหารจัดการและสามารถเพิ่ม ประสิทธิภาพในการทำงานภายในองค์กร ได้มากยิ่งขึ้น

| | |
|----------------------|---|
| Title | Defect Tracking System |
| Student | Ms. Suchalai Khawbang |
| Student ID | 54660801 |
| Degree | Master of Science |
| Program | Information Technology |
| Major | Information Technology and Management |
| Academic Year | 2012 |
| Advisor | Associate Professor Ponrudee Netisopakul, Ph.D. |

ABSTRACT

Currently, Islamic Bank of Thailand or “ibank” were a in process of developing many IT Systems to support internal operations. Therefore, UAT Division, IT Department has role to test all of systems. When we found any defects or any issues, we record them manually in a work sheet such as Microsoft Excel. This form of defect difficult to manage monitor and to create reports for defect tracking. To solve this problem, this project will develop “defect tracking system” to manage sub process defect tracking process.

Defect Tracking System separate is into sub process defect on production system and the defect on UAT Test sub process. In addition, defect tracking system will record any issues that effect the system, such as, changing requirement, more issues from users. The developed system will support management and increase efficiency for other IT system of ibank.

กิตติกรรมประกาศ

การศึกษาอิสระ ในหัวข้อเรื่องระบบสารสนเทศเพื่อติดตาม และรายงานข้อผิดพลาดหรือประเด็นการแก้ไขระบบนี้ ได้รับการส่งเสริม และสนับสนุนให้สำเร็จลุล่วงไปได้ด้วยดีจากบุคคลหลายท่าน ซึ่งผู้เขียนขอ กราบขอบพระคุณมา ณ ที่นี้ ได้แก่

รศ.ดร. พรฤดี เนติโสภากุล ผู้ซึ่งป็นอาจารย์ที่ปรึกษาโครงการการศึกษาอิสระนี้ ที่ให้ความกรุณารับเป็นอาจารย์ที่ปรึกษา ท่านได้ให้คำปรึกษา ข้อคิดเห็น และคำแนะนำตลอดจนช่วยเหลือในการตรวจสอบ และแก้ไขข้อบกพร่องต่างๆ ในแต่ละขั้นตอนของการจัดทำรายงานฉบับนี้

อาจารย์ทุกท่าน ที่ได้อบรมสั่งสอน และให้ความรู้ในหลักวิชาการต่างๆ ที่สามารถนำมาประยุกต์ใช้ในการพัฒนาระบบงานนี้ได้อย่างสำเร็จลุล่วง

เพื่อนๆ พี่ๆ ITM 25.2 ทุกคนที่ได้ช่วยเหลือ สนับสนุน และเป็นกำลังใจให้กันเสมอมา ทำให้การศึกษาหลักสูตรเป็นไปได้อย่างดี

ผู้บริหาร พี่ๆ เพื่อนร่วมงานทุกท่าน ของกลุ่มงานเทคโนโลยีสารสนเทศ ส่วนงานทดสอบระบบสารสนเทศ ธนาคารอิสลามแห่งประเทศไทย ที่ได้ให้การสนับสนุน และยินยอมให้พัฒนาต้นแบบระบบสารสนเทศเพื่อติดตาม และรายงานข้อผิดพลาดหรือประเด็นการแก้ไขระบบและเอื้อเพื่อให้ข้อมูลต่างๆ ที่จำเป็นในการดำเนินการ รวมทั้งเป็นกำลังใจเสมอมา

สุดท้ายนี้สิ่งที่สำคัญที่สุดคือ ครอบครัวของข้าพเจ้า และนายณัฐวุฒิ วงศ์ชัย ที่เข้าใจและเป็นกำลังใจ และให้การสนับสนุนข้าพเจ้าเสมอมา ตั้งแต่วันแรกที่เริ่มศึกษา จวบจนวันสำเร็จการศึกษา

สุชาลัย ขาวบาง
20 มีนาคม 2556

สารบัญ

หน้า

| | |
|---|------|
| บทคัดย่อภาษาไทย..... | I |
| บทคัดย่อภาษาอังกฤษ..... | II |
| กิตติกรรมประกาศ | III |
| สารบัญ..... | IV |
| สารบัญตาราง..... | VI |
| สารบัญรูป..... | VIII |
| บทที่ 1 บทนำ..... | 1 |
| 1.1 ความเป็นมาและความสำคัญของปัญหา..... | 1 |
| 1.2 วัตถุประสงค์ของโครงการ..... | 1 |
| 1.3 ขอบเขตของการดำเนินงาน..... | 2 |
| 1.4 ขั้นตอนการดำเนินโครงการ..... | 2 |
| 1.5 ประโยชน์ที่คาดว่าจะได้รับ..... | 2 |
| บทที่ 2 ทฤษฎีพื้นฐานและเทคโนโลยีที่เกี่ยวข้อง..... | 3 |
| 2.1 ทฤษฎีการทดสอบซอฟต์แวร์..... | 3 |
| 2.2 การทดสอบถดถอย (Regression Testing)..... | 4 |
| 2.3 การทดสอบและรับรองจากผู้ใช้งาน (User Acceptance Test)..... | 4 |
| 2.4 การทดสอบอื่น ๆ (Non-Functional)..... | 5 |
| 2.5 ความรู้เกี่ยวกับข้อผิดพลาดหรือข้อบกพร่อง (Defect)..... | 5 |
| 2.6 เว็บแอปพลิเคชัน..... | 9 |

สารบัญ (ต่อ)

| | หน้า |
|---|------|
| บทที่ 3 การวิเคราะห์และออกแบบระบบ..... | 11 |
| 3.1 การดำเนินงานปัจจุบัน | 11 |
| 3.2 ปัญหาของการทำงานปัจจุบัน | 29 |
| 3.3 ความต้องการระบบงานใหม่..... | 33 |
| 3.4 แผนภาพที่เกี่ยวข้องกับการวิเคราะห์และออกแบบระบบ | 36 |
| บทที่ 4 การออกแบบฐานข้อมูลและออกแบบหน้าจอส่วนต่อประสานกับผู้ใช้ | 62 |
| 4.1 แบบจำลองความสัมพันธ์ระหว่างเอนทิตี..... | 62 |
| 4.2 พจนานุกรมข้อมูล | 65 |
| 4.3 สิทธิการใช้งานระบบ..... | 72 |
| 4.4 ออกแบบหน้าจอส่วนต่อประสานกับผู้ใช้..... | 73 |
| บทที่ 5 สรุปผลการศึกษา..... | 89 |
| 5.1 สรุปผลการศึกษา..... | 89 |
| 5.2 ข้อจำกัดระบบ..... | 89 |
| 5.3 ข้อเสนอแนะและการพัฒนาระบบเพิ่มเติม..... | 89 |
| บรรณานุกรม | 90 |
| ประวัติผู้เขียน..... | 91 |

สารบัญตาราง

| ตารางที่ | หน้า |
|--|------|
| 3.1 คำอธิบายยูสเคส Create New Defect | 45 |
| 3.2 คำอธิบายยูสเคส Update Defect | 46 |
| 3.3 คำอธิบายยูสเคส Assign Defect | 47 |
| 3.4 คำอธิบายยูสเคส Resolve Defect | 48 |
| 3.5 คำอธิบายยูสเคส Close Defect | 49 |
| 3.6 คำอธิบายยูสเคส Search Defect | 50 |
| 3.7 คำอธิบายยูสเคส Generate Summary Report and Chart | 51 |
| 3.8 คำอธิบายยูสเคส View Defect /Issue Activity | 52 |
| 3.9 คำอธิบายยูสเคส Manage User Profile | 53 |
| 3.10 คำอธิบายยูสเคส Manage Project for Testing | 54 |
| 3.11 คำอธิบายยูสเคส Send Email Notification | 55 |
| 4.1 แสดงรายละเอียดของแต่ละเอนทิตี | 62 |
| 4.2 พจนานุกรมข้อมูลของเอนทิตี DEFECT..... | 66 |
| 4.3 พจนานุกรมข้อมูลของเอนทิตี PRIORITY..... | 67 |
| 4.4 พจนานุกรมข้อมูลของเอนทิตี SEVERITY..... | 67 |
| 4.5 พจนานุกรมข้อมูลของเอนทิตี TYPE..... | 67 |
| 4.6 พจนานุกรมข้อมูลของเอนทิตี STATUS..... | 68 |
| 4.7 พจนานุกรมข้อมูลของเอนทิตี LOG_DEFECT..... | 68 |
| 4.8 พจนานุกรมข้อมูลของเอนทิตี ATTACHMENT | 68 |
| 4.9 พจนานุกรมข้อมูลของเอนทิตี PROJECT | 69 |

สารบัญตาราง (ต่อ)

| ตารางที่ | หน้า |
|----------|---|
| 4.10 | พจนานุกรมข้อมูลของเอนทิตี TESTPLAN_PROJECT 69 |
| 4.11 | พจนานุกรมข้อมูลของเอนทิตี EMPLOYEE..... 70 |
| 4.12 | พจนานุกรมข้อมูลของเอนทิตี TEAM..... 70 |
| 4.13 | พจนานุกรมข้อมูลของเอนทิตี POSITION..... 71 |
| 4.14 | แสดงการเปรียบเทียบสิทธิการใช้งานในระบบ 72 |



สารบัญรูป

| รูปที่ | หน้า |
|--------|--|
| 2.1 | แสดงระดับของการทดสอบโปรแกรม3 |
| 2.2 | แสดงการจัดการข้อผิดพลาด (Defect Management)6 |
| 2.3 | แสดงการจัดกลุ่มความรุนแรงของข้อผิดพลาด8 |
| 2.4 | แสดงการจัดลำดับความสำคัญของข้อผิดพลาด8 |
| 3.1 | แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ ขั้นตอนที่ 1 Planning & Definition 17 |
| 3.2 | แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ ขั้นตอนที่ 2 Gathering & Requirement Analysis..... 18 |
| 3.3 | แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ ขั้นตอนที่ 3 System Design 19 |
| 3.4 | แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ ขั้นตอนที่ 4 Implement Coding20 |
| 3.5 | แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ ขั้นตอนที่ 5 Test Case Preparations.....21 |
| 3.6 | แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ ขั้นตอนที่ 6 Test Executions.....22 |
| 3.7 | แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ ขั้นตอนที่ 7 Deployment.....23 |

สารบัญรูป (ต่อ)

| รูปที่ | หน้า |
|--------|---|
| 3.8 | แสดงแผนภาพและขั้นตอนการปฏิบัติ กรณีบนระบบ Production มี CR หรือ IR26 |
| 3.9 | แสดงแบบฟอร์ม IR (Incident Request)27 |
| 3.10 | แสดงแบบฟอร์ม CR (Change Request)28 |
| 3.11 | ตัวอย่างการบันทึกข้อผิดพลาดจากการทดสอบระบบ ATM.....31 |
| 3.12 | แสดงเอกทิวทัศน์ไออะแกรมของการสร้าง หรือ บันทึกข้อผิดพลาด (Create New Defect).....36 |
| 3.13 | แสดงเอกทิวทัศน์ไออะแกรมของการปรับปรุงแก้ไขข้อผิดพลาดและมอบหมายงานข้อผิดพลาด37 |
| 3.14 | แสดงเอกทิวทัศน์ไออะแกรมของการปรับปรุงแก้ไขข้อผิดพลาดของ Developer38 |
| 3.15 | แสดงเอกทิวทัศน์ไออะแกรมของการตรวจและตัดสินใจเลือก Solution (Resolve Defect)39 |
| 3.16 | แสดงเอกทิวทัศน์ไออะแกรมของการปรับปรุงแก้ไขข้อผิดพลาดและปิดสถานะ40 |
| 3.17 | แสดงเอกทิวทัศน์ไออะแกรมของการสร้างรายงานสรุปต่าง ๆ41 |
| 3.18 | แสดงยูสเคสไออะแกรมของระบบ42 |
| 3.19 | แสดงสเตตไออะแกรมของสถานะข้อผิดพลาด (Status Defect)57 |
| 3.20 | แสดงคลาสไออะแกรมของระบบติดตามเพื่อติดตาม59 |
| 4.1 | แสดงอีอาร์ไออะแกรมของระบบ63 |
| 4.2 | แสดงหน้าจอเข้าสู่ระบบ73 |
| 4.3 | แสดงหน้าจอเข้าสู่ระบบสำหรับผู้ทดสอบระบบ (Tester).....74 |
| 4.4 | แสดงหน้าจอเข้าสู่ระบบสำหรับผู้จัดการส่วนทดสอบระบบ (Test Manager)75 |

สารบัญรูป (ต่อ)

| รูปที่ | หน้า |
|--------|--|
| 4.5 | แสดงหน้าจอเข้าสู่ระบบสำหรับผู้พัฒนาระบบ (Developer) 75 |
| 4.6 | แสดงหน้าจอเข้าสู่ระบบสำหรับ BA, SA,PM..... 76 |
| 4.7 | แสดงหน้าจอ Create Defect /Issue 77 |
| 4.8 | แสดงหน้าจอการค้นหาข้อผิดพลาดจากการทดสอบระบบฯ 78 |
| 4.9 | แสดงหน้าจอเพื่อคลิก Update เปลี่ยนแปลงรายละเอียดหรือสถานะต่าง ๆ..... 79 |
| 4.10 | แสดงหน้าจอสิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้ทดสอบ (Tester) 80 |
| 4.11 | แสดงหน้าจอสิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้จัดการส่วนทดสอบ (Test Manager)..... 81 |
| 4.12 | แสดงหน้าจอสิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้พัฒนาระบบ (Developer) 82 |
| 4.13 | แสดงหน้าจอเพื่อเรียกดูรายละเอียด สถานะ กิจกรรมต่าง ๆ 83 |
| 4.14 | แสดงหน้าจอ Generate Summary Report 84 |
| 4.15 | แสดงหน้าจอรายงานสถานะข้อผิดพลาดทั้งหมด (Defect Status Report) 85 |
| 4.16 | แสดงหน้าจอ Manage User Profile..... 86 |
| 4.17 | แสดงหน้าจอ Create New Account 87 |
| 4.18 | แสดงหน้าจอ Manage Project for Testing and Add Test Plan 88 |

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันธนาคารอิสลามแห่งประเทศไทย มีระบบสารสนเทศใหม่ๆ เกิดขึ้นหลายระบบ และระบบที่กำลังพัฒนาขึ้นใหม่ เนื่องจาก ณ ปัจจุบันกลุ่มงานเทคโนโลยีสารสนเทศในสำนักงานทดสอบระบบสารสนเทศ เมื่อมีการทดสอบระบบต่างๆ เกิดขึ้นนั้นและ เมื่อพบข้อผิดพลาดจากการทดสอบระบบ (Defect) หรือประเด็น (Issue) การแก้ไขปรับเปลี่ยน โปรแกรมหรือพารามิเตอร์ในโปรแกรมต่างๆ นั้น จะบันทึก Defect หรือ Issue ต่างๆ ลงใน Microsoft excel ซึ่งยากในการบริหารจัดการ ติดตามและรายงานข้อผิดพลาดที่เกิดจากการทดสอบระบบ เพราะ Defect แต่ละ Defect มีระดับความรุนแรงหรือลำดับความสำคัญที่แตกต่างกัน อีกทั้งทำให้ในแต่ฝ่ายงานของกลุ่มงานเทคโนโลยีสารสนเทศ ซึ่งมีผู้ดูแลรับผิดชอบในส่วนงานที่เกี่ยวข้อง ไม่ทราบที่มาที่ไปหรือรายละเอียดของข้อผิดพลาดนั้นๆ และไม่สามารถแก้ไขหรือติดตามสถานะข้อผิดพลาดที่เกิดขึ้นได้อย่างถูกต้อง และรวดเร็ว ทั้งนี้เพื่อให้การจัดการและการติดตามข้อผิดพลาดหรือประเด็นต่างๆ จากระบบให้เป็นไปตามกระบวนการทำงาน (Workflow) ของกลุ่มงานเทคโนโลยีสารสนเทศ

ดังนั้น จึงได้นำระบบสารสนเทศดังกล่าวมาพัฒนาให้สามารถบริหารจัดการข้อผิดพลาดที่เกิดขึ้นได้ โดยแบ่งเป็นข้อผิดพลาดที่เกิดบนระบบ Production (ระบบงานจริง) และข้อผิดพลาดจากการทดสอบระบบก่อนขึ้นใช้งานจริง อีกทั้งระบบนี้จะเก็บประเด็นการปรับเปลี่ยน แก้ไข ต่างๆ ที่มีผลกระทบต่อระบบ อาทิเช่น Requirement / ประเด็นเพิ่มเติมจาก User เพื่อใช้ในการบริหารจัดการ

1.2 วัตถุประสงค์ของโครงการ

เพื่อออกแบบ และพัฒนาระบบระบบสารสนเทศเพื่อติดตาม และรายงานข้อผิดพลาดที่เกิดจากระบบของธนาคารอิสลามแห่งประเทศไทย ทางผู้ศึกษาได้กำหนดวัตถุประสงค์ของระบบงานไว้ดังนี้

1. เพื่อปรับปรุงการจัดการข้อมูลและบริหารข้อมูลต่างๆ จากการทดสอบระบบ ให้เป็นระบบ ระเบียบ ลดความซ้ำซ้อน และความผิดพลาดของข้อมูล
2. เพื่อการบริหารจัดการและติดตาม Defect หรือ Issue ที่เกิดขึ้นได้อย่างถูกต้องตรงประเด็นและรวดเร็วขึ้น
3. เพื่อให้ผู้บริหารหรือผู้รับผิดชอบระบบ สามารถนำข้อมูล Defect หรือ Issue ที่เกิดขึ้นจากการทดสอบมาวิเคราะห์หรือประเมินระบบต่างๆ ก่อนนำขึ้นใช้งานจริงบนระบบ Production (ระบบงานจริง)
4. เพื่อพัฒนาความสามารถในการทำดำเนินการธุรกิจขององค์กรให้มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของการดำเนินงาน

สำหรับระบบสารสนเทศเพื่อติดตาม และรายงานข้อผิดพลาดหรือประเด็นการแก้ไขระบบของธนาคารอิสลามเป็น Defect Life Cycle คือ วงจรการทำงานของ Defect ที่เริ่มตั้งแต่เมื่อพบ Defect หรือ Issue แล้ว Assign ให้ทางทีม Developer (bank)/Vendor แก้ไข จนกระทั่งทีม Developer (bank)/Vendor แก้ไขเสร็จ ซึ่งมีรายละเอียด ดังนี้

- ระบบสามารถจัดการสร้าง/บันทึก/ค้นหา ผลการทดสอบกรณีพบ Defect หรือ Issue ตาม Test Case ที่พบในช่วย Phase Testing รวมถึงแสดงรายงานสรุปผลทั้งหมดทุก Status ในรูปแบบเนื้อหาและกราฟ
- ระบบสามารถ Assign และแจ้งเตือนในรูปแบบอีเมลล์ และ Alert บนหน้าจอของระบบดังกล่าว ทุกกรณีที่เมื่อมีการพบ Defect ในกรณีที่ เป็น Defect ที่รุนแรงมากหรือ Defect มีระยะเวลาจำกัด
- ระบบสามารถสร้างรายงานสรุปผลการติดตามข้อผิดพลาดหรือประเด็นต่าง ๆ จากการทดสอบฯ พร้อมสร้างเอกสาร User Acceptance Test

1.4 ขั้นตอนการดำเนินโครงการ

1. ศึกษาปัญหาของการทำงานปัจจุบัน ปัญหาที่เกิดขึ้น โดยการสอบถามจากพนักงานและผู้บริหารเพื่อให้ทราบถึงข้อมูลต่าง ๆ และรวบรวมเอกสารและแบบฟอร์มต่าง ๆ ที่ใช้งานอยู่ในปัจจุบัน
2. วิเคราะห์ปัญหาของระบบงานปัจจุบัน และศึกษาความต้องการของผู้ใช้งานระบบและความเป็นไปได้ของการพัฒนาระบบงานใหม่
3. วิเคราะห์และออกแบบระบบงานใหม่ โดยใช้แนวคิดเชิงวัตถุมาใช้ในการจำลองระบบ รวมทั้งออกแบบฐานข้อมูลที่ใช้ในการเก็บข้อมูลต่าง ๆ ของระบบงาน
4. ออกแบบหน้าจอส่วนต่อประสานกับผู้ใช้และรายงานต่าง ๆ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถนำระบบที่มีการวิเคราะห์และออกแบบ ไปพัฒนาต่อในด้านโปรแกรมมิ่งได้อย่างสมบูรณ์และถูกต้อง
2. ช่วยลดขั้นตอนในการดำเนินงานที่ซับซ้อน และบริหารจัดการข้อมูลให้เป็นไปตามกระบวนการทำงานขององค์กร
3. ทำให้องค์กรสามารถดำเนินธุรกิจได้ดียิ่งขึ้น ซึ่งเป็นภาพลักษณ์ที่ดีขององค์กร
4. ผู้บริหารสามารถวิเคราะห์ ประเมินและตัดสินใจว่าระบบที่กำลัง Implement อยู่และระบบที่แก้ไขบางส่วนบน Production สามารถนำขึ้นใช้งานจริงได้หรือไม่ ซึ่งระบบนี้มีข้อมูลประกอบการตัดสินใจได้อย่างถูกต้องและทันต่อเหตุการณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

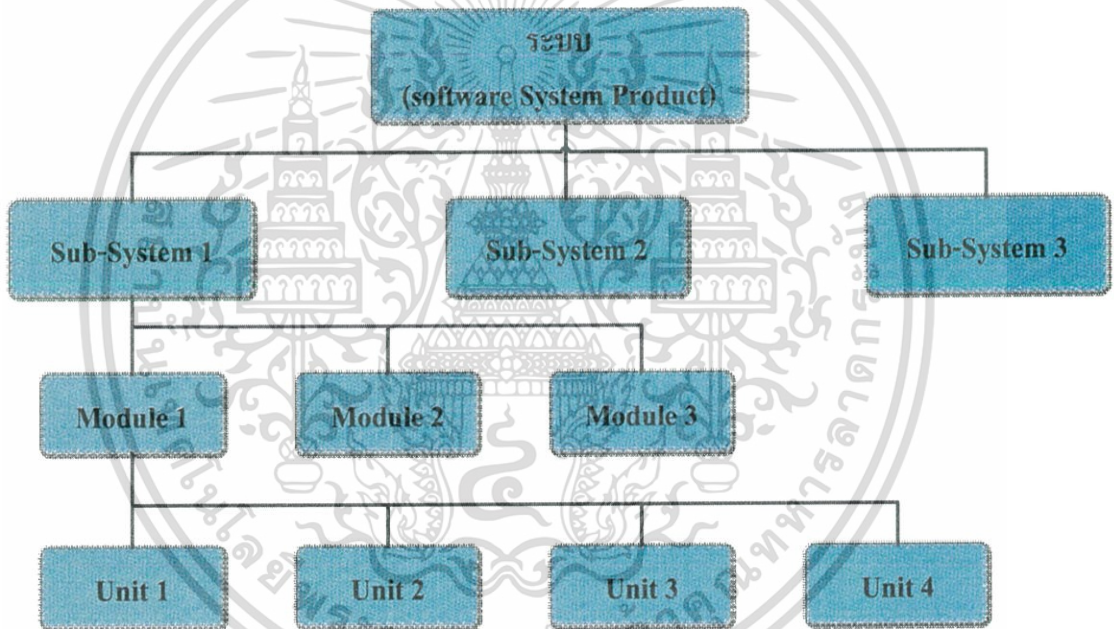
ทฤษฎีพื้นฐานและเทคโนโลยีที่เกี่ยวข้อง

2.1 ทฤษฎีการทดสอบซอฟต์แวร์

การทดสอบซอฟต์แวร์ (Software Testing) คือ กิจกรรมที่จัดทำขึ้นเพื่อปรับปรุงคุณภาพของซอฟต์แวร์ โดยตรวจหาข้อผิดพลาดและปัญหาที่เกิดขึ้น แล้วทำการแก้ไขข้อผิดพลาดนั้น

- ระดับของการทดสอบโปรแกรม

ระดับการทดสอบจะถูกกำหนดไว้ในแผนการทดสอบ (Test Planning) ในส่วน Test strategy เพื่อเห็นแนวทางว่าควรจะใช้การทดสอบแบบใดให้เหมาะกับโปรแกรม และความต้องการของผู้ใช้งาน (User Requirement)



รูปที่ 2.1 แสดงระดับของการทดสอบ โปรแกรม

1. การทดสอบระดับหน่วย (Unit Test) (โอภาส , 2548)

การทดสอบระดับหน่วยเป็นการทดสอบของแต่ละโมดูล (Module) ภายในโปรแกรม โดยแต่ละโมดูลจะถูกทดสอบแยกจากกัน ทำให้บางโมดูล จำเป็นต้องใช้โปรแกรมที่เขียนขึ้นมาโดยเฉพาะ นั่นคือตัวขับ (Driver) และตัวดำเนินการ (stub) โดยตัวขับเป็นโปรแกรมที่ใช้จำลองการเรียกโมดูลที่ต้องการทดสอบ พร้อมกับให้ข้อมูลเข้า (Input) และรับผลลัพธ์ (Output) จากการทำงานของโมดูลนั้น และตัวดำเนินการเป็นโปรแกรมจำลองที่โมดูลที่จะทดสอบนั้นต้องเรียกใช้ ซึ่งกรณีทดสอบหน่วยต้องระบุตัวขับ และตัวดำเนินการสำหรับแต่ละโมดูลไว้ ดำเนินการโดย: โปรแกรมเมอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การทดสอบระดับรวมหน่วย (Integration Testing)

การทดสอบการรวมเป็นการตรวจสอบข้อผิดพลาดที่เกิดขึ้นจากการรวมโมดูลเข้าด้วยกัน และเป็นการตรวจสอบความเข้ากันได้ระหว่างโมดูล ดำเนินการโดย : โปรแกรมเมอร์

3. การทดสอบระบบ (System Testing)

การทดสอบระบบเป็นการตรวจสอบความถูกต้องในการทำงานของส่วน (Element) ต่างๆในระบบทั้งหมด โดยกรณีทดสอบที่สร้างขึ้นต้องครอบคลุมทุก ๆ หน้าที่ (Function) การทำงานของระบบทั้งฮาร์ดแวร์ และซอฟต์แวร์ เพื่อให้ระบบที่พัฒนาไม่เกิดความผิดพลาดการทำงาน จุดประสงค์ เพื่อตรวจสอบระบบว่าระบบทำงานได้ถูกต้องและได้ผลลัพธ์ตรงตาม Requirement โดยจะทำการทดสอบรวมเข้ากับองค์ประกอบอื่นของระบบ คือ อุปกรณ์ บุคลากร และข้อมูล โดยทำการทดสอบทั้ง Functional และ Non- functional Test ดำเนินการโดย: Tester หรือ QA

2.2 การทดสอบถดถอย (Regression Testing)

เป็นการทดสอบซ้ำโดยใช้ชุดทดสอบเดิมที่เคยทำมาแล้ว จุดประสงค์เพื่อดูว่าการเปลี่ยนแปลง (Change) ในจุดหนึ่งจะไปกระทบกับส่วนอื่นๆ ที่ไม่ได้ทำการเปลี่ยนแปลงหรือไม่ ซึ่งการทดสอบเชิงถดถอย นั้นจะเป็นการทดสอบทั้งระบบอีกครั้งหนึ่งเป็นการทดสอบซ้ำกับ Case เดิมที่เคยทดสอบไปแล้ว เพื่อที่จะได้ตรวจสอบ (Verify) ได้ว่าหน้าที่ (Function) เดิมยังใช้งานได้อยู่ โดยแบ่งเป็น 2 แบบดังนี้

1. กรณีที่มีการแก้ไขหน้าที่ (Function) กลางที่ใช้งานร่วมกันของระบบ จำเป็นจะต้องทำการทดสอบใหม่ทั้งหมด เพื่อว่าการแก้ไข function กลางจะไปมีผลกระทบกับส่วนอื่นๆ หรือไม่
2. กรณีที่มีการ Upgrade software เช่นจาก oracle8i เป็น oracle10g ในกรณีนี้จะต้องทำการทดสอบ Application ใหม่ทั้งหมด เพื่อที่จะตรวจสอบ (Verify) ให้ได้ว่าเมื่อ upgrade oracle แล้วจะไม่มีผลกระทบใดๆ ต่อ application เดิม

ดำเนินการโดย : Tester หรือ QA

2.3 การทดสอบและรับรองจากผู้ใช้งาน (User Acceptance Test)

เป็นการทดสอบและรับรองจากผู้ใช้งานระบบงาน จุดประสงค์เป็นกระบวนการการทดสอบเพื่อให้แน่ใจว่าระบบที่พัฒนาพร้อมที่จะใช้งานจริง ตรงตามกระบวนการทางธุรกิจ (Business Process) และความต้องการของผู้ใช้งานที่ได้กำหนดไว้ (Software Requirement) โดยสภาพแวดล้อม ในการทำ UAT (User Acceptance Test) จะต้องเป็นสภาพแวดล้อมที่เหมือนหรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใกล้เคียงกับ สภาพแวดล้อมการใช้งานจริงมากที่สุดและจะใช้ข้อมูลจริงในการทดสอบ โดยแบ่งได้เป็น 2 ประเภทคือ

1. Alpha testing

การทดสอบความสมบูรณ์ของระบบโดยผู้ใช้ และใช้ข้อมูลสมมติในการทดสอบ จะสมมติให้ระบบอยู่ในสถานการณ์ที่อาจจะเกิดขึ้นได้

2. Beta testing

การทดสอบความสมบูรณ์ของระบบโดยผู้ใช้ และใช้ข้อมูลจริงในการทดสอบ และภายใต้สถานการณ์ที่เกิดขึ้นจริง

2.4 การทดสอบอื่น ๆ (Non-Functional)

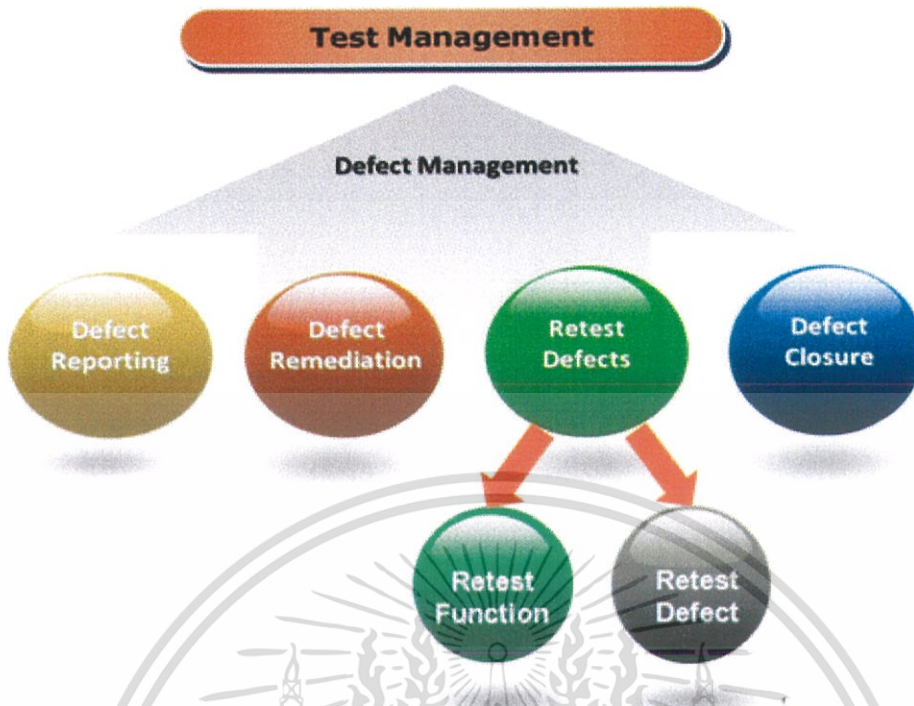
การทดสอบอื่น ๆ ประกอบด้วย การทดสอบประสิทธิภาพในการทำงานของระบบสามารถจำแนกเป็น การทดสอบการทำงานสูงสุด (Peak Load Testing) การทดสอบประสิทธิภาพของเวลา (Performance Testing) การทดสอบการกู้ระบบ (Recovery Testing) การทดสอบการเก็บข้อมูล (Storage Testing) การทดสอบกระบวนการ (Procedure Testing) และการทดสอบผู้ใช้ (User)

2.5 ความรู้เกี่ยวกับข้อผิดพลาดหรือข้อบกพร่อง (Defect)

ข้อผิดพลาดหรือข้อบกพร่อง (Defect) คือ ปัญหาที่พบในการทดสอบระบบ ซึ่งปัญหาเหล่านี้ อาจจะกระทบต่อหน้าที่ (Function) การทำงานของระบบ เช่น ระบบแสดง Error ต่าง ๆ หรือ Defect ที่พบอาจจะไม่กระทบกับ Function การทำงาน เช่นการแสดงผลที่อาจเกิดจาก Design หรือ การแสดงผลของข้อความ ซึ่งปัญหาเหล่านี้จะไม่ส่งผลกระทบต่อระบบ

- การจัดการข้อผิดพลาด (Defect Management)

การจัดการข้อผิดพลาด (Defect) ที่พบในช่วงขั้นตอนการทดสอบในแต่ละโปรเจกต์นั้น ๆ โดยการจัดทำสรุปรายงานข้อผิดพลาดที่พบและแจ้งทีมที่เกี่ยวข้อง เพื่อใช้ข้อมูลสรุปปัญหาของโปรเจกต์นั้น ๆ ว่าจะสามารถ Launch ได้ตาม Plan หรือไม่ โดยมีการจัดการข้อผิดพลาดดังนี้



รูปที่ 2.2 แสดงการจัดการข้อผิดพลาด (Defect Management)

1. Defect Report

สรุปรายงานแสดงผลการทดสอบระบบงานที่เกิดข้อผิดพลาดหรือ ไม่สอดคล้องกับ Functional specification การเขียน defect report ที่ดีนั้น จะช่วยให้แก้ไขความผิดพลาดที่เกิดขึ้นได้อย่างรวดเร็ว, การทดสอบเกิดประสิทธิภาพ, ลดจำนวนข้อผิดพลาดในอนาคต, ช่วยเรื่องการประสานงานระหว่าง ทีม tester และ ทีม developer

2. Defect Remediation

หลังจากสรุปรายงานข้อผิดพลาดจากการทดสอบนั้น ๆ ขั้นตอนนี้ทีมผู้เกี่ยวข้องก็จะทำการแก้ไขข้อผิดพลาดต่าง ๆ ให้ถูกต้อง

3. Retest Defect

เป็นการทดสอบระบบอีกครั้ง เพื่อที่จะตรวจสอบว่า การแก้ไขหรือการเปลี่ยนแปลงไม่มีผลทำให้เกิดการเปลี่ยนแปลงของระบบ พร้อมทั้งทดสอบว่า Code ใหม่สามารถทำงานได้ตามปกติ และ ไม่มีผลเสียหายต่อฟังก์ชันการทำงานใดๆ ก่อนหน้านี้

Entry Criteria

- ทำการ Test Execution เรียบร้อยแล้ว
- ทีมผู้พัฒนาทำการแจ้งยืนยันความพร้อมของระบบ ทั้งนี้ ทีมผู้พัฒนาจะต้องดำเนินการต่าง ๆ ดังต่อไปนี้ เรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทำการแก้ไขปัญหาจากรายงานปัญหา (Defects Report) ที่พบเรียบร้อยแล้ว
- ทำการ Update / Deployment Source code ไปที่ Test Environment เรียบร้อยแล้ว
- ทำการแจ้งปรับสถานะต่าง ๆ ของรายงานปัญหาที่แจ้งเรียบร้อยแล้ว

4. Defect Closure

สรุปผลการทดสอบทั้งระบบ พร้อมทั้งระบุถึงความเสี่ยงของระบบนั้น ๆ โดยทำการสรุปภาพรวมของงานทั้งหมด เพื่อให้ทุกฝ่ายที่เกี่ยวข้องได้ทราบถึงข้อมูลเกี่ยวกับการทดสอบ รวมถึงปัญหาต่าง ๆ ที่ยังคงค้างอยู่ ซึ่งจะมีการแก้ไขใน Release ต่อ ๆ ไป ทั้งนี้ทางทีมทดสอบจะดำเนินการจัดทำเอกสาร Test Report ส่งให้ทีมต่าง ๆ ที่เกี่ยวข้อง โดยสรุปรายละเอียดทั้งหมด และส่งมอบให้กับ UAT เพื่อดำเนินการทดสอบระบบต่อไปพร้อมดำเนินการจัดทำ Summary Test Report เป็นต้น

Entry Criteria

- ทำการ Test Execution และ Regression Testing เรียบร้อยแล้ว
- ปัญหาต่าง ๆ ที่พบได้รับการแก้ไขเรียบร้อยแล้ว หรือ ตามตกลงแล้วแต่กรณี ทั้งนี้เนื่องจาก ปัญหาที่พบบางข้อนั้น อาจจะมีการแก้ไขใน Release ต่อ ๆ มา อีกครั้ง ทั้งนี้ การตกลงต่าง ๆ นั้น จำเป็นจะต้องมีผู้เกี่ยวข้องต่าง ๆ เข้ามาปรึกษากัน และ ยังต้องพิจารณาถึงความเสี่ยงต่าง ๆ ต่อไป

- การจัดประเภทของข้อผิดพลาด (Classification of Defects)

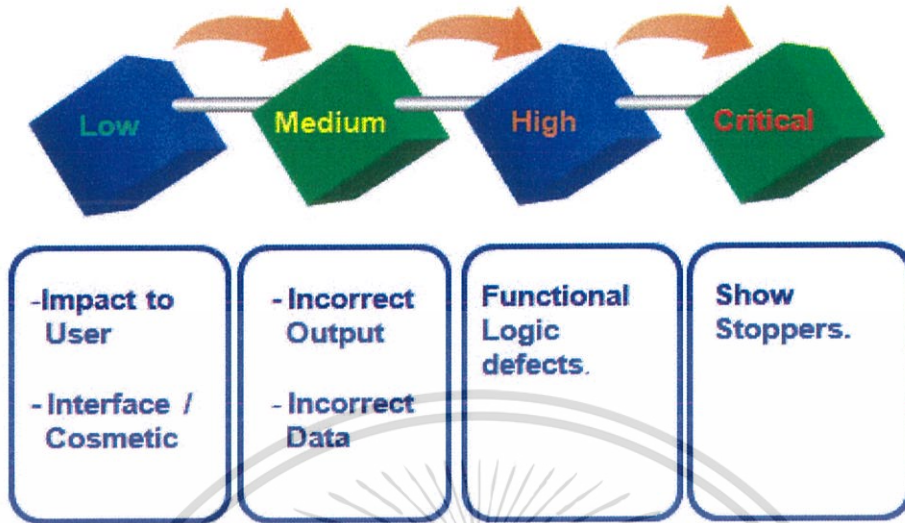
มีหลายวิธีซึ่งเราสามารถแบ่งประเภทบางส่วนได้ดังนี้

1. ความรุนแรงของข้อผิดพลาด (Defect Severity) (Younessi Houman,2002)

- ข้อผิดพลาดหลัก : ข้อผิดพลาด ซึ่งจะทำให้เกิดการล้มเหลวของผลลัพธ์ที่สามารถสังเกตได้หรือมีความแตกต่างจากความต้องการ
- ข้อผิดพลาดรอง : ข้อผิดพลาดที่จะไม่ก่อให้เกิดความล้มเหลวในการดำเนินงาน
- ร้ายแรง : ข้อผิดพลาดที่จะทำให้เกิดระบบล้มเหลว หรือหยุดชะงัก หรือมีผลกระทบต่อโปรแกรม

ข้อผิดพลาดที่อาจจะกำหนดได้อย่างใดอย่างหนึ่งที่เป็นสาเหตุของความเสียหายของข้อมูลหรือความผิดพลาดของระบบ หรือการละเมิดด้านความปลอดภัยของระบบ ในการบันทึกข้อผิดพลาดจะต้องใส่ระดับความรุนแรงของ Error ด้วย ซึ่งในส่วนนี้จะต้องเป็นข้อตกลงร่วมกันระหว่างทุกคนที่อยู่ในทีมงานของ Project นั้นๆ ว่า จะจัด Error ต่างๆ ไว้ที่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

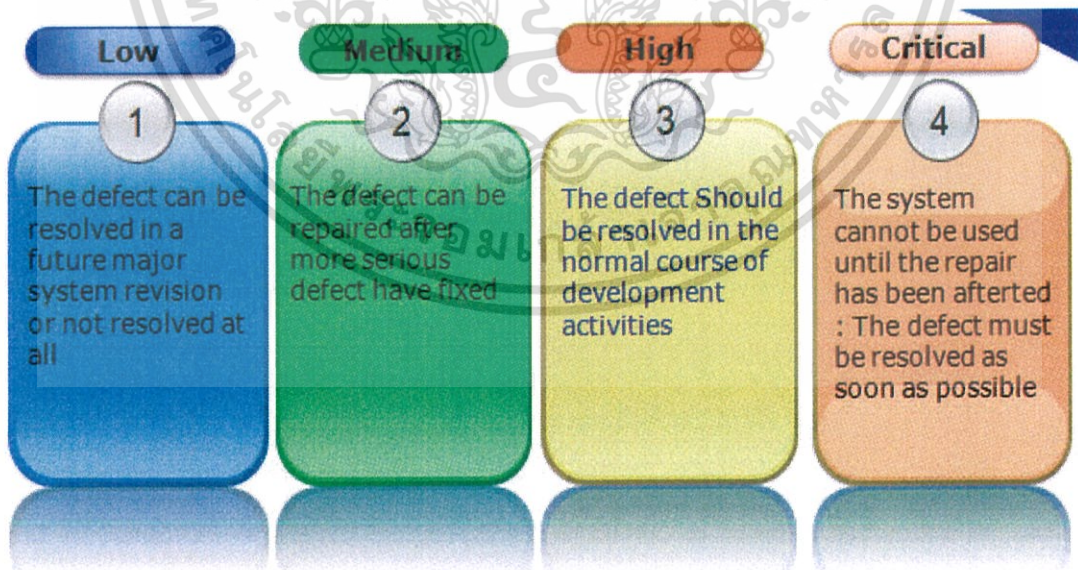
ระดับใดบ้าง โดยในแต่ละระดับก็จะมีนิยามที่แตกต่างกัน โดยแบ่งระดับความรุนแรงตามภาพ
ได้ดังนี้



รูปที่ 2.3 การจัดกลุ่มความรุนแรงของข้อผิดพลาด

2. การจัดลำดับความสำคัญของข้อผิดพลาด (Defect Priority)

เป็นการจัดลำดับของ Case ต่างๆ ว่า Case ไหนจำเป็นที่จะต้องกำจัด Error ไม่ให้เกิดขึ้น ถ้าเกิด error ประเภทนี้ขึ้นแล้วจะก่อให้เกิดผลเสียต่อองค์กรมาก ก็จะถูกจัดไว้เป็น Priority แรกๆ Case เหล่านี้จะถูกเรียกว่า Critical Case เป็นต้น ซึ่งแบ่งระดับความสำคัญได้ดังนี้



รูปที่ 2.4 การจัดลำดับความสำคัญของข้อผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ประเภทของข้อผิดพลาด (Defect Type) (Myers Glenford J., 1946)

- ข้อผิดพลาดจากการศึกษาระบบและคุณสมบัติการทำงานจากระบบ
 - ในหลายๆงานหรือโครงการ ไม่เอกสารการศึกษาระบบและคุณสมบัติการทำงาน
 - มีเอกสารการศึกษาระบบและคุณสมบัติการทำงานระบบไม่ละเอียดครบถ้วนสมบูรณ์
 - การสื่อสารเข้าใจ ข้อความในคุณสมบัติการทำงานไม่ตรงกัน
 - เงื่อนไขขอบเขตสเปคของความต้องการถูกละเลย ไม่ให้ความสำคัญ
- ข้อผิดพลาดจากการออกแบบสถาปัตยกรรมและรายละเอียดการออกแบบระบบ
 - การวางแผนและการออกแบบอาจทำให้เกิดข้อผิดพลาด โดยเปรียบเทียบได้กับแม่พิมพ์ในการสร้างอาคาร ข้อผิดพลาดอาจเกิดได้ด้วยเหตุผลเดียวกับที่เกิดจากสเปคการวางแผนอาจไม่ได้ทำให้ละเอียดพอและไม่สอดคล้องตามความต้องการของผู้ใช้งาน
 - อาจทำอย่างเร่งรีบ มีการเปลี่ยนแปลง ไม่ได้ทำความเข้าใจแบบอย่างเพียงพอ
- ข้อผิดพลาดจากซอสโค้ด
 - เกิดจากความซับซ้อนของซอฟต์แวร์และเอกสารที่ไม่ครบถ้วนสมบูรณ์ รวมถึงความกดดันในการทำงานให้ทันเวลา อาจจะเป็นความผิดพลาดของโปรแกรมเมอร์เอง แต่ความผิดพลาดจำนวนมาก สามารถค้นต้นตอไปยังขั้นการออกแบบหรือสเปคได้ ข้อผิดพลาดที่เกิดจากซอสโค้ด อาทิเช่น ข้อผิดพลาดจากการคำนวณที่ไม่ถูกต้อง , ความเข้าใจผิดหรือการขาดหายไปของซอสโค้ด, ตรรกะผิดพลาด, ข้อผิดพลาดจากการประกาศตัวแปร และการแสดงข้อความในซอสโค้ดที่ผิดพลาด เป็นต้น

2.6 เว็บแอปพลิเคชัน

เว็บแอปพลิเคชัน (Web Application) คือ แอปพลิเคชันที่เข้าถึงด้วยเว็บเบราว์เซอร์ผ่านเครือข่ายคอมพิวเตอร์อย่างอินเทอร์เน็ต หรืออินทราเน็ต โดยใช้เครื่องมือที่พัฒนาร่วมกัน ดังต่อไปนี้

1. Adobe Dreamweaver CS5 (นwor, 2553)

Adobe Dreamweaver CS5 เป็นโปรแกรมสำหรับพัฒนาเว็บไซต์ ซึ่งมีคุณสมบัติครอบคลุม ตั้งแต่การออกแบบและสร้างเว็บเพจ การบริหารจัดการเว็บไซต์ ตลอดจนจนถึงการพัฒนาเว็บแอปพลิเคชัน Dreamweaver ได้รับความนิยมเป็นอย่างสูงในบรรดาโปรแกรมประเภทเดียวกันเนื่องจากคุณสมบัติเด่น คือใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานง่าย มีเครื่องมืออำนวยความสะดวกในการจัดวางข้อความภาพ กราฟิก ตารางข้อมูล แบบฟอร์ม รวมทั้งองค์ประกอบต่างๆ ที่ใช้ได้ตอบกับผู้ชมในเว็บเพจได้อย่างง่ายดาย โดยผู้ใช้ไม่จำเป็นต้องรู้จักภาษา HTML CSS และ JavaScript จึงเหมาะสำหรับผู้เริ่มต้นและผู้ใช้ทั่วไป

2. Microsoft Visual Studio.Net (สุรสิทธิ์และนันทน์, 2546)

ไมโครซอฟท์วิซวลสตูดิโอได้ออกเน็ต (Visual Studio .NET) คือ เครื่องมือสำหรับพัฒนาโปรแกรมแบบกราฟิกสำหรับระบบปฏิบัติการวินโดวส์ (Windows Operating System) โดยมีรากฐานภาษามาจากภาษาเบสิก และทำงานบนได้ออกเน็ตเฟรมเวิร์ค (Dotnet Framework) ถูกออกแบบให้มีความสามารถในการพัฒนาโปรแกรมเชิงวัตถุอย่างแท้จริง และรองรับการออกแบบด้วยยูเอ็มแอล (UML = Unified Modeling Language) เป็นการรวมเครื่องมือที่จำเป็นและครบวงจรต่อการพัฒนาโปรแกรมอย่างครบถ้วน (IDE = Integrated Development Environment) รวมบริการการพัฒนาภาษาโปรแกรม บริการคลาสด์พื้นฐานให้นำมาใช้ร่วมกันได้อย่างเป็นระบบ เช่น เอสคิวแอลเซิร์ฟเวอร์ (SQL Server) และเอเอสพีได้ออกเน็ต (ASP .NET) เป็นต้น โดยทั้งหมดทำงานอยู่บนซีแอลอาร์ (CLR = Common Language Runtime) ที่รองรับการประมวลผลและเข้าใช้ทรัพยากรในเครื่องได้อย่างมีประสิทธิภาพ

3. Microsoft SQL Server (ธาริน, 2548)

ไมโครซอฟท์ซีคิวลเซิร์ฟเวอร์ (Microsoft SQL Server) คือระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Database Management System (DBMS)) หรือฐานข้อมูลเป็นกลุ่มรวบรวมหน่วยข้อมูลที่จัดขึ้นเป็นชุดของอธิบายเป็นตารางข้อมูลอย่างเป็นทางการ พัฒนาโดยไมโครซอฟท์ มีลักษณะเป็นโครงสร้างในการเก็บรวบรวมข้อมูล การเพิ่ม การเข้าถึง หรือการประมวลผลข้อมูล โดยใช้ภาษา T-SQL มีระบบบริหารข้อมูลระดับเอนเตอร์ไพรซ์ มีความน่าเชื่อถือ ในการรักษาความปลอดภัย (Security) ความพร้อมใช้งาน (Availability) และความพร้อมในการขยายระบบ (Scalability) และมีประสิทธิภาพสูง ประหยัดเวลาและลดค่าใช้จ่ายในการพัฒนาและจัดการฐานข้อมูล สนับสนุนการพัฒนาแบบร่วมกับ Microsoft Visual Studio

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การวิเคราะห์และออกแบบระบบ

การวิเคราะห์การทำงานจะกล่าวถึงขั้นตอนการทำงานของกลุ่มงานเทคโนโลยีสารสนเทศ ในส่วนงานทดสอบระบบสารสนเทศ ธนาคารอิสลามแห่งประเทศไทย เป็นกรณีศึกษา โดยการศึกษา และวิเคราะห์ถึงกระบวนการทำงาน ปัญหาที่พบและข้อจำกัดของระบบงานปัจจุบัน

3.1 การดำเนินงานปัจจุบัน

สำหรับกลุ่มงานเทคโนโลยีสารสนเทศ เมื่อเริ่มจะมีการ Implement ระบบต่าง ๆ ก็จะมีการจัดประชุมเพื่อชี้แจงรายละเอียดของ Project นั้น ๆ และมอบหมายความรับผิดชอบให้แก่ส่วนงานต่าง ๆ โดยจัดตั้งทีมงาน โครงการ (Project Team) เพื่อแบ่งความรับผิดชอบให้แก่ทีมที่เกี่ยวข้อง ซึ่งในส่วนการทดสอบระบบสารสนเทศจะแบ่งการทดสอบออกเป็น 2 แบบดังนี้

1. การทดสอบสำหรับการพัฒนาระบบงานใหม่ (Testing in Development Phase) เรียงลำดับขั้นตอนการทำงาน ดังแสดงดังรูปที่ 3.1-3.7 โดยขั้นตอนการทดสอบระบบจะแสดงดังรูปที่ 3.5-3.6

1. การวางแผนการพัฒนาระบบ (Planning & Definition)

เป็นขั้นตอนของการวางแผนโครงการตั้งแรกเริ่มในการพัฒนาระบบ โดย Project Manager (PM) จะบริหารจัดการโครงการ ซึ่งจะเป็นผู้คัดสรรทรัพยากร หรือบุคคลกรที่เกี่ยวข้องเพื่อเข้าร่วมโครงการที่จะพัฒนาระบบเช่น Business Analyst (BA), System Analyst (SA), Test Manager เป็นต้น รวมถึงจัดทำเอกสารทั้งหมดที่เกี่ยวข้องกับโครงการ, แผนการดำเนินงานโครงการให้บรรลุตามเป้าหมายโครงการ สามารถแสดงขั้นตอนการทำงานดังแสดงในรูปที่ 3.1

2. การวิเคราะห์ความต้องการระบบ (Gathering & Requirement Analysis)

1. Business Analyst (BA) จัดเก็บความต้องการด้านธุรกิจจากผู้ใช้งาน วิเคราะห์ความเป็นไปได้เชิงธุรกิจจากความต้องการ และนำความต้องการมาทำการสร้างแผนงานด้านธุรกิจ และจัดทำเอกสารความต้องการจากผู้ใช้งาน URD (User Requirement Document)
2. System Analyst (SA) นำเอกสาร URD Part: Business Requirement มาวิเคราะห์ความต้องการในเชิงระบบสารสนเทศ และจัดทำเอกสาร URD เพิ่มเติมในส่วนวิเคราะห์ความเป็นไปได้เชิงระบบและแผนภาพต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Project Manager (PM) ทำการรีวิวกเอกสาร URD อีกครั้งหลังจาก BA และ SA ทำการวิเคราะห์ความเป็นไปได้แล้วในเชิงธุรกิจและเชิงระบบสารสนเทศ หลังจากนั้นนำเสนอให้ผู้ใช้งานตรวจสอบ

4. User Requestor ทำการตรวจสอบเอกสาร URD ว่าตรงตามความต้องการที่ต้องการหรือไม่ ถ้าตรงตามความต้องการก็ยืนยันความถูกต้องของเอกสาร URD ถ้าไม่ย้อนกลับไปแก้ไขใน

ขั้นตอนวิเคราะห์ความเป็นไปได้เชิงธุรกิจใหม่อีกครั้ง สามารถแสดงขั้นตอนการทำงานดังแสดงในรูปที่ 3.2

ทั้งนี้ในเอกสารความต้องการจากผู้ใช้งาน URD (User Requirement Document) จะมีเอกสารย่อยดังนี้ เอกสารการศึกษาระบบ (System Study document) และเอกสารคุณสมบัติการทำงาน (Functional Specification document) ซึ่งอาจพบปัญหาจากการทำงานปัจจุบันในเอกสารดังกล่าว อาทิเช่น ข้อผิดพลาดจากเอกสารที่มีการเปลี่ยนความต้องการบ่อยหรือเอกสารไม่ละเอียดครบถ้วนสมบูรณ์ ทำให้การทดสอบเกิดข้อผิดพลาดได้ง่าย เป็นต้น

3. การออกแบบระบบ (System Design)

1. Business Analyst (BA) วิเคราะห์ผลกระทบที่เกิดจากธุรกิจในการพัฒนาระบบสารสนเทศ และสร้างเมทริกซ์ติดตามความต้องการ (Traceability Matrix) เพื่อการติดตามและจัดการกับความต้องการต่างๆ ตลอดทั้งโครงการที่พัฒนาทำให้เราเห็นความเป็นไปหรือความเปลี่ยนแปลงของ ความต้องการนั้น ๆ รวมถึงผลกระทบที่จะกระทบกับเอกสารต่างๆ ภายในโครงการฯ

2. System Analyst (SA) ออกแบบระบบตามความต้องการที่กำหนดในเอกสาร URD วิเคราะห์ผลกระทบที่เกิดขึ้นระหว่างธุรกิจกับระบบสารสนเทศรวมถึงออกแบบแผนภาพที่เกี่ยวข้องกับการวิเคราะห์และออกแบบระบบทั้งหมด

3. Project Manager (PM) ทำการรีวิวกเอกสาร URD อีกครั้งหลังจาก BA และ SA ทำการวิเคราะห์ความเป็นไปได้แล้วในเชิงธุรกิจและเชิงระบบสารสนเทศ หลังจากนั้นนำเสนอให้ผู้ใช้งานตรวจสอบ 4. User Requestor ทำการตรวจสอบเอกสาร URD ว่าตรงตามความต้องการที่ต้องการหรือไม่ในครั้งท้าย เนื่องจากครบทุกฟังก์ชันการทำงานทั้งหมดแล้ว พร้อมลงนามในเอกสาร URD สามารถแสดงขั้นตอนการทำงานดังแสดงในรูปที่ 3.3

ทั้งนี้ในเอกสารความต้องการจากผู้ใช้งาน URD (User Requirement Document) จะมีเอกสารย่อยดังนี้ เอกสารการออกแบบสถาปัตยกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Architectural Design Document) และเอกสารรายละเอียดการออกแบบ (Detailed Design document) ซึ่งอาจพบปัญหาจากการทำงานปัจจุบันในเอกสารดังกล่าว อาทิเช่น การวิเคราะห์และออกแบบไม่สอดคล้องตรงตามความต้องการของผู้ใช้งาน ทำให้การออกแบบภาพไม่ถูกต้องมีผลกระทบไปยังหลาย ๆ ทิมที่เกี่ยวข้อง โดยเฉพาะผลการทดสอบที่เกิดขึ้นมีข้อผิดพลาดจากการออกแบบ ทำให้ระบบไม่ประสบผลสำเร็จในการออกแบบระบบฯ เป็นต้น

4. พัฒนาระบบขั้นตอนการเขียนโปรแกรม (Implement Coding)

1. Developer ทำการพัฒนาระบบตามเอกสาร URD ที่ถูกต้องสมบูรณ์แล้ว หลังจากพัฒนาระบบเสร็จแล้วในเบื้องต้น Developer ทำการจัดทำทดสอบพร้อมทดสอบในส่วนการทดสอบระดับหน่วย (Unit Test) และการทดสอบระดับรวมหน่วย (Integration Testing) ด้านเทคนิคต่างๆ ถ้ากรณีที่พบข้อผิดพลาด Developer ก็จะทำการบันทึกไว้ใน File excel แล้วทำการแก้ไขจนผ่านและสรุปผลการทดสอบในส่วน Unit Integrated พร้อมนำส่งระบบที่พัฒนาให้ Test Manager เพื่อทำการทดสอบระบบต่อไป

2. Test Manager รับสรุปผลการทดสอบในส่วน Unit Integrated พร้อมระบบที่พัฒนาเสร็จเรียบร้อยแล้ว มาเพื่อเตรียมตัวสำหรับการทดสอบต่อไป สามารถแสดงขั้นตอนการทำงานดังแสดงในรูปที่ 3.4

ทั้งนี้ในขั้นตอนนี้อาจพบข้อผิดพลาดจากข้อสัปดาห์ อาทิเช่น ข้อผิดพลาดจากการประกาศตัวแปร , ข้อผิดพลาดจากตรรกะ หรือการคำนวณไม่ถูกต้อง ทำให้การทดสอบเกิดข้อผิดพลาดขึ้นเยอะ เนื่องจาก Developer อาจไม่เข้าใจตั้งแต่ความต้องการของระบบ และการออกแบบระบบ เป็นต้น

5. การจัดเตรียมการทดสอบระบบ (Test Case Preparation)

1. Test Manager จัดทำกลยุทธ์การทดสอบซอฟต์แวร์โดยการออกแบบกรณีทดสอบและการวางแผนทดสอบ เพื่อให้ได้ชุดของขั้นตอนตามที่ปฏิบัติ เป็นการยืนยันว่าการสร้างซอฟต์แวร์ประสบผลสำเร็จ และกำหนดแผนการทดสอบหลัก

2. Tester จัดทำแผนการทดสอบสำหรับการทดสอบ UAT (User Acceptance Test) และจัด Test Case และ Test Script

3. Project Manager (PM) Review Test Case และ Test Script ก่อนนำส่งให้ผู้ใช้งาน

4. User Requestor ทำการตรวจสอบ Test Case และ Test Script ว่าถูกต้องตามฟังก์ชันการทำงานหรือไม่พร้อมลงนามยืนยัน Test Case และ Test Script

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Tester หลังจากได้รับการยืนยันจากผู้ใช้งานว่า Test Case และ Test Script ถูกต้องตรงตามฟังก์ชันการทำงาน Tester ก็ทำการจัดเตรียม Test Environment เพื่อทำการทดสอบต่อไปสามารถแสดงขั้นตอนการทำงานดังแสดงในรูปที่ 3.5

ทั้งนี้ในขั้นตอนนี้อาจพบข้อผิดพลาดจากแผนการทดสอบ อาทิเช่น อาจจะมีการวางแผนการทดสอบที่ไม่ถูกต้องสมบูรณ์ ทำให้ทดสอบผิดพลาดไม่เป็นไปตามแผนการทดสอบ เป็นต้น

6. การดำเนินการทดสอบระบบ (Test Execution)

1. Tester ดำเนินการทดสอบระบบ (System Testing) เพื่อตรวจสอบระบบว่าระบบทำงานได้ถูกต้องและได้ผลลัพธ์ตรงตามความต้องการ ตาม Test Case หรือ Test Script ที่ได้จัดทำหรือไม่

2. กรณีถ้าพบข้อผิดพลาดในส่วนของ การทดสอบระบบ (System Testing) Tester จะทำการบันทึกลงไฟล์ Excel และทำการ Regression Test จนไม่พบข้อผิดพลาด

3. ถ้าไม่พบข้อผิดพลาดจากทดสอบระบบก็จัดทำสรุปในส่วนทดสอบระบบ (System Testing)

4. Project Manager, Business Analyst, System Analyst, Test Manager ตรวจสอบและลงนามในส่วนทดสอบระบบ (System Testing)

5. Tester ดำเนินการทดสอบและรับรองจากผู้ใช้งาน (User Acceptance Test) เพื่อการทดสอบความสมบูรณ์ของระบบโดยผู้ใช้ และใช้ข้อมูลสมมุติโดย Tester และใช้ข้อมูลจริงในการทดสอบ โดยผู้ใช้งานจริง ทดสอบเพื่อให้แน่ใจว่าระบบที่พัฒนาพร้อมที่จะใช้งานจริง ตรงตามกระบวนการทางธุรกิจ (Business Process) และความต้องการของผู้ใช้งานที่ได้กำหนดไว้ในเอกสารความต้องการจากผู้ใช้ (User Requirement Document)

6. กรณีพบข้อผิดพลาดในการทดสอบจากผู้ใช้ (User Acceptance Test) Tester จะทำการบันทึกลงไฟล์ Excel และทำการ Regression Test จนไม่พบข้อผิดพลาด

7. ถ้าไม่พบข้อผิดพลาดจากทดสอบระบบก็จัดทำสรุปในส่วนทดสอบและรับรองจากผู้ใช้งาน (User Acceptance Test)

8. จัดทำสรุปทั้งหมดสำหรับการทดสอบของโครงการฯและจัดประชุมทีมที่เกี่ยวข้องและผู้ใช้งานระบบเพื่อแจ้งผลการทดสอบทั้งหมด รวมถึงจัดหาข้อสรุปสำหรับโครงการฯ พร้อมยืนยันผลการทดสอบทั้งหมดจากทีมผู้เกี่ยวข้องทั้งหมด

9. จัดทำเอกสารสรุปผลการทดสอบเพื่อยืนยันการทดสอบ และให้ผู้เกี่ยวข้องทั้งหมดลงนาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. Project Manager นำส่งเอกสารสรุปผลการทดสอบทั้งหมดว่าระบบสามารถขึ้นใช้งานจริงได้ อาทิเช่น User Acceptance Test Form, Migration Form เป็นต้น สามารถแสดงขั้นตอนการทำงานดังแสดงในรูปที่ 3.6

7. การนำระบบขึ้นใช้งานจริง (Deployment)

1. IT Operation เตรียมนำระบบที่พัฒนาขึ้นใช้งานจริง โดยจัดเตรียมฮาร์ดแวร์ ซอฟต์แวร์ที่เกี่ยวข้องกับระบบทั้งหมด และอนุมัติการขึ้นระบบ โดยผู้บริหาร
2. Project Manager จัดทำสรุปเพื่อปิดโครงการฯ และจัดประชุมเพื่อสรุปภาพรวมโครงการทั้งหมดก่อนปิดโครงการ รวมถึงให้ผู้เกี่ยวข้องทั้งหมดลงนามปิดโครงการที่พัฒนาฯ สามารถแสดงขั้นตอนการทำงานดังแสดงในรูปที่ 3.7

2. การทดสอบสำหรับระบบงานจริง (Production) กรณีมีข้อผิดพลาดหรือการปรับเปลี่ยนแปลงบนระบบงานจริงแล้วต้องแก้ไขโปรแกรม

ทั้งนี้ในส่วนของ ระบบงานจริง (Production) แล้วเกิดกรณีมีข้อผิดพลาดหรือการปรับเปลี่ยนแปลงบนระบบงานจริงแล้วต้องแก้ไข โปรแกรม ก็จะมีการทดสอบระบบทุกครั้ง โดยเรียกว่า IR (Incident Request) คือ เกิดข้อผิดพลาดบนระบบงานจริง หรือ Change Request (CR) คือการปรับปรุงเปลี่ยนแปลงพารามิเตอร์หรือเพิ่ม Function การทำงานใหม่กับระบบ เป็นต้น ทั้งนี้มีแผนภาพแสดงขั้นตอนการปฏิบัติงาน กรณีได้รับ IR (Incident Request) และ Change Request (CR) สามารถแสดงขั้นตอนการทำงานดังแสดงในรูปที่ 3.8

โดยอธิบายแผนภาพและขั้นตอนงานการปฏิบัติกรณีบนระบบ Production มี CR (Change Request) เรียงลำดับขั้นตอนการทำงานดังนี้

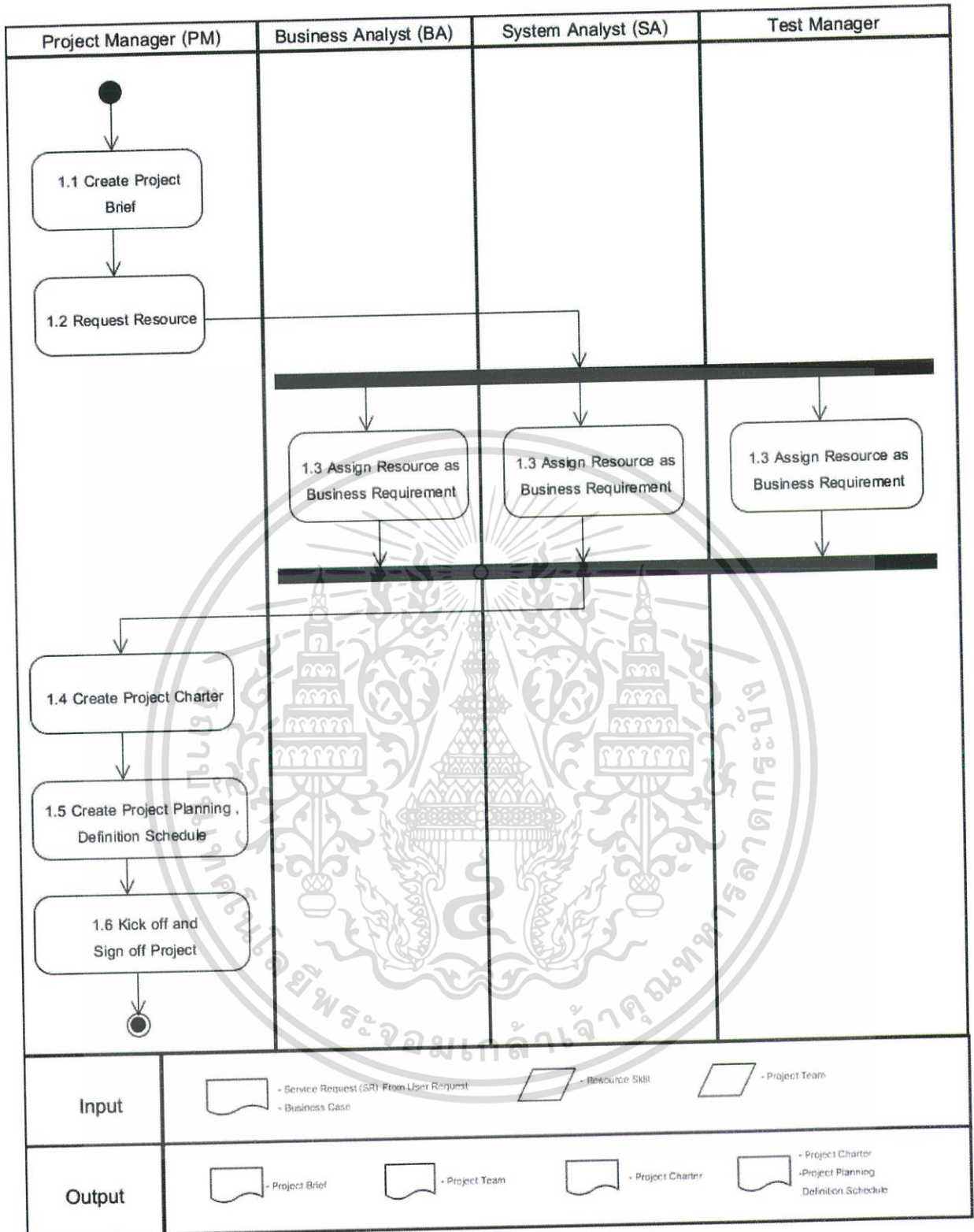
1. User ส่ง CR ให้กับ IT Help Desk โดยมีผู้บริหารฝ่ายของ User เป็นผู้อนุมัติ ซึ่ง Help Desk จะเป็นผู้ให้คำปรึกษากับ User ว่าสิ่งที่ User ต้องการนั้นควรเป็น CR หรือไม่อย่างไร และทำการชี้แจงเหตุผลเบื้องต้นให้ User ทราบ ทั้งนี้มีแบบฟอร์มการกรอก ข้อมูล IR (Incident Request) และ Change Request (CR) ดังแสดงในรูปที่ 3.9 – 3.10
2. Help Desk ทำการส่ง CR มายัง BA เพื่อทำการประเมิน Requirement ตามคำขอของ User และ ส่งมาถึง SA
3. BA และ SA จะดำเนินการพิจารณา CR วิเคราะห์ทั้ง Business และ System และประสานคุยทำความเข้าใจกับ Developer
4. ทบทวนและอนุมัติโดยผู้บริหารกลุ่มงานเทคโนโลยีสารสนเทศ และผู้บริหารฝ่ายของ User ลงนามเพื่อรับทราบ จากนั้นจึงทำการส่ง CR ให้กับ IT Help Desk
5. Developer ดำเนินการปรับแก้ไขโปรแกรมตามคำขอ CR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ส่วนทดสอบระบบสารสนเทศ ทำการทดสอบ CR ถ้าทดสอบไม่ผ่านจะทำการส่งให้ผู้ให้บริการภายนอก(Outsource)/ IT Develop ดำเนินการแก้ไขใหม่
 - กรณีทดสอบแล้วพบ Defect จะจัดเก็บและติดตามการแก้ไขในรูปแบบไฟล์ Excel
7. ถ้าทดสอบผ่านแล้ว ส่วนทดสอบระบบฯ จัดทำเอกสาร UAT Summary Signoff เพื่อให้ผู้บริหารฝ่ายของ User และ ส่วนทดสอบระบบสารสนเทศ ร่วมกันเซ็นรับทราบผลการทดสอบ
8. เข้าสู่กระบวนการนำระบบขึ้นใช้งานจริง โดย ผู้ให้บริการภายนอก(Outsource) / IT Operation



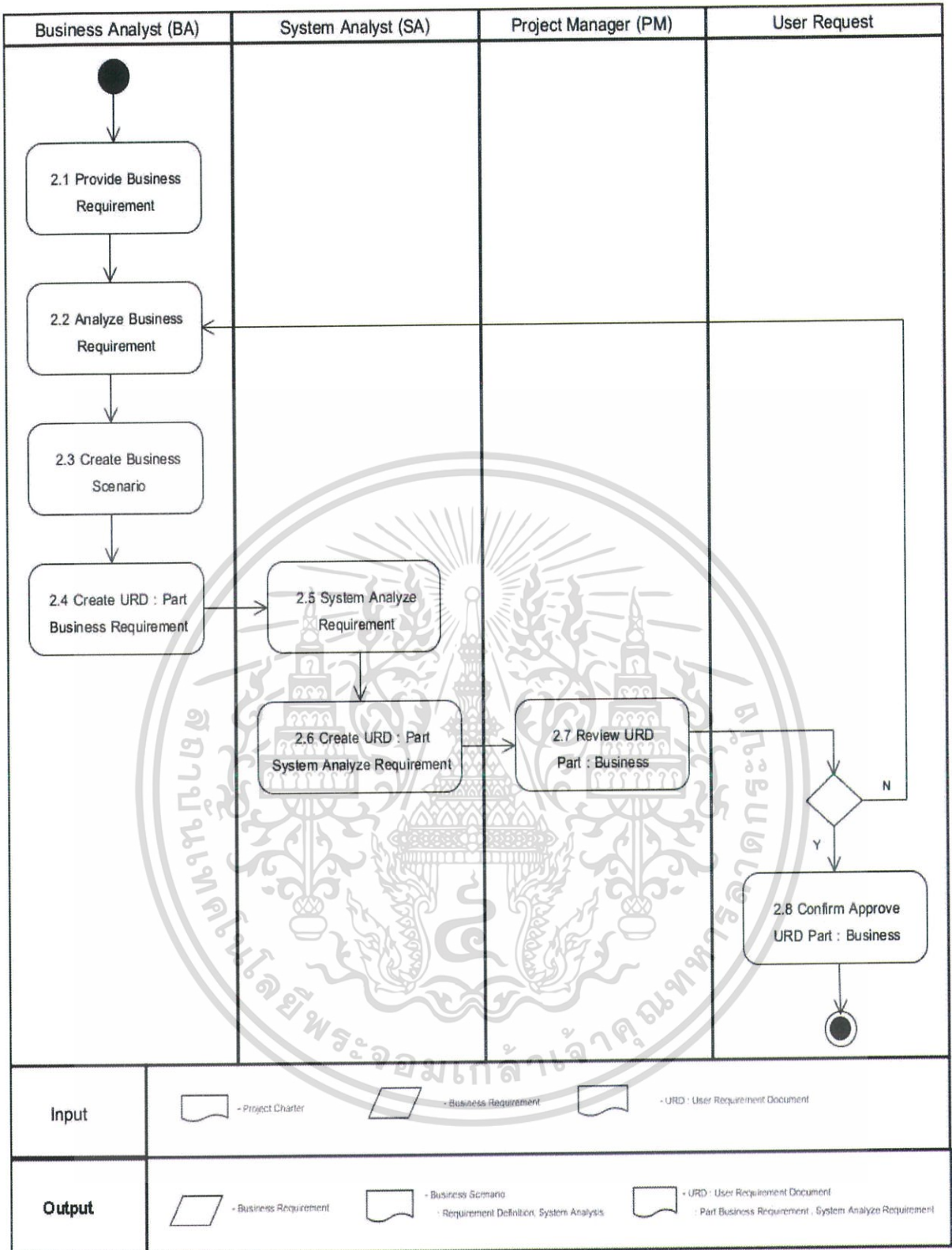
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนาระบบกับการทดสอบระบบ
ขั้นตอนที่: 1. Planning & Definition

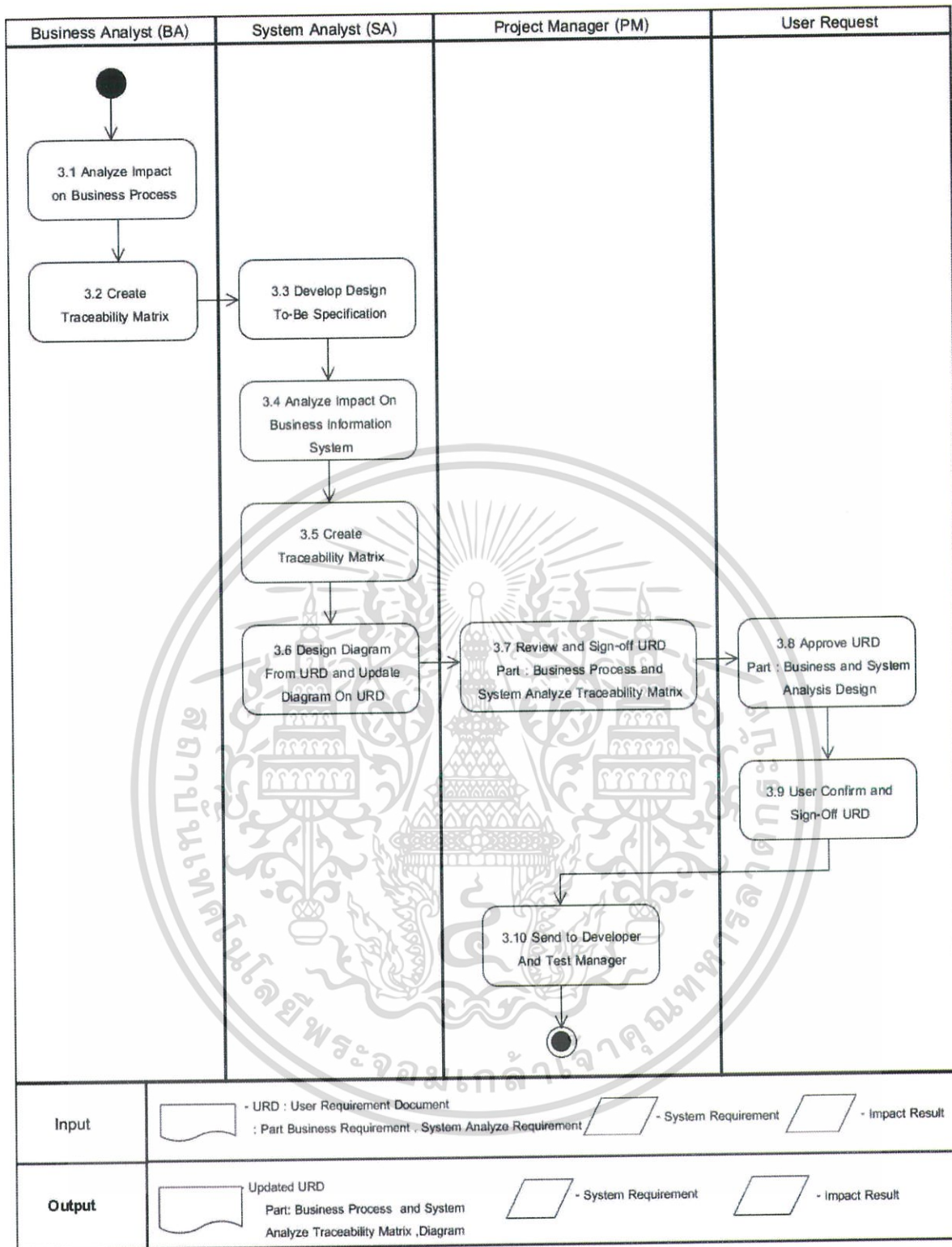
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

131351



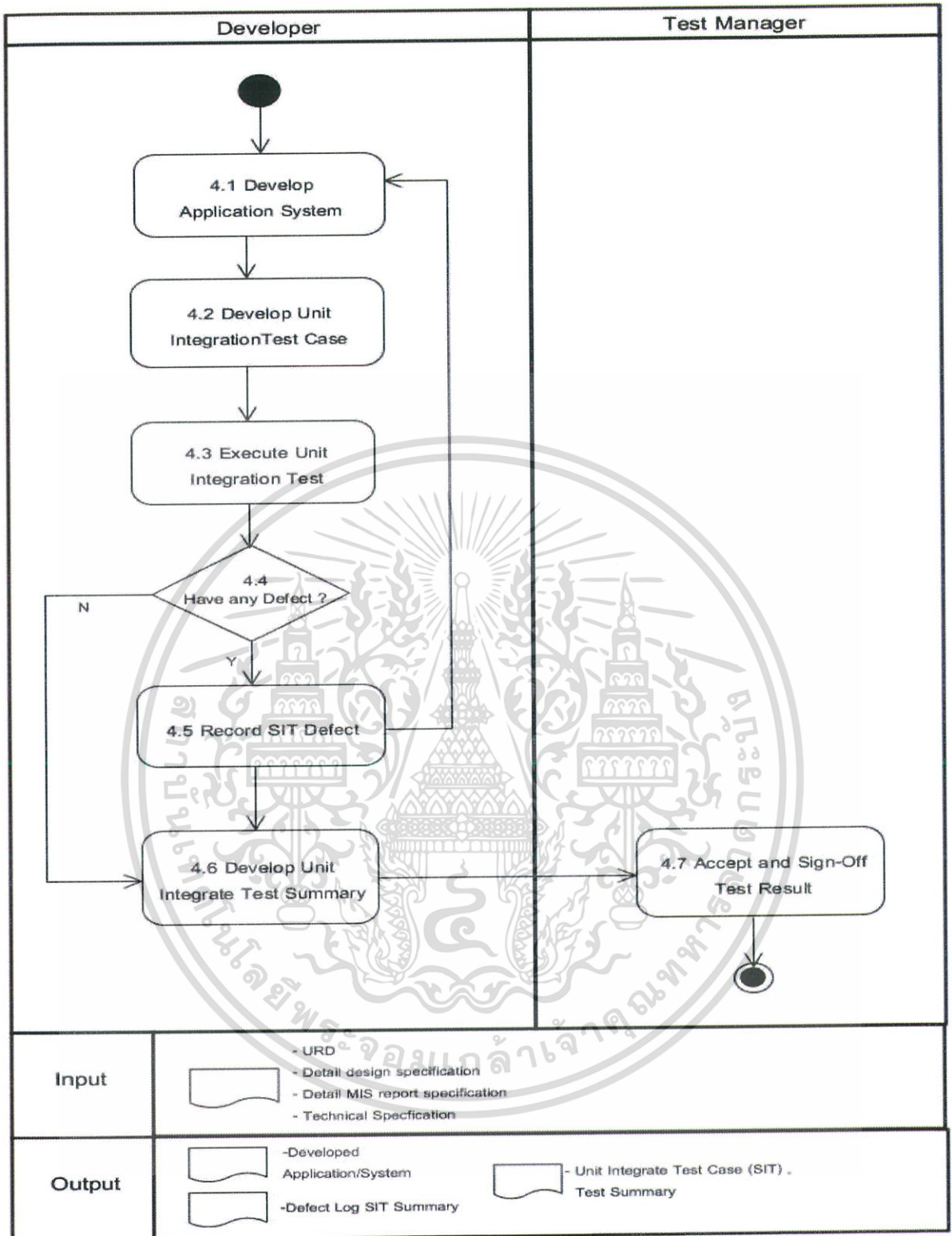
รูปที่ 3.2 แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ
ขั้นตอนที่ 2 Gathering & Requirement Analysis

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



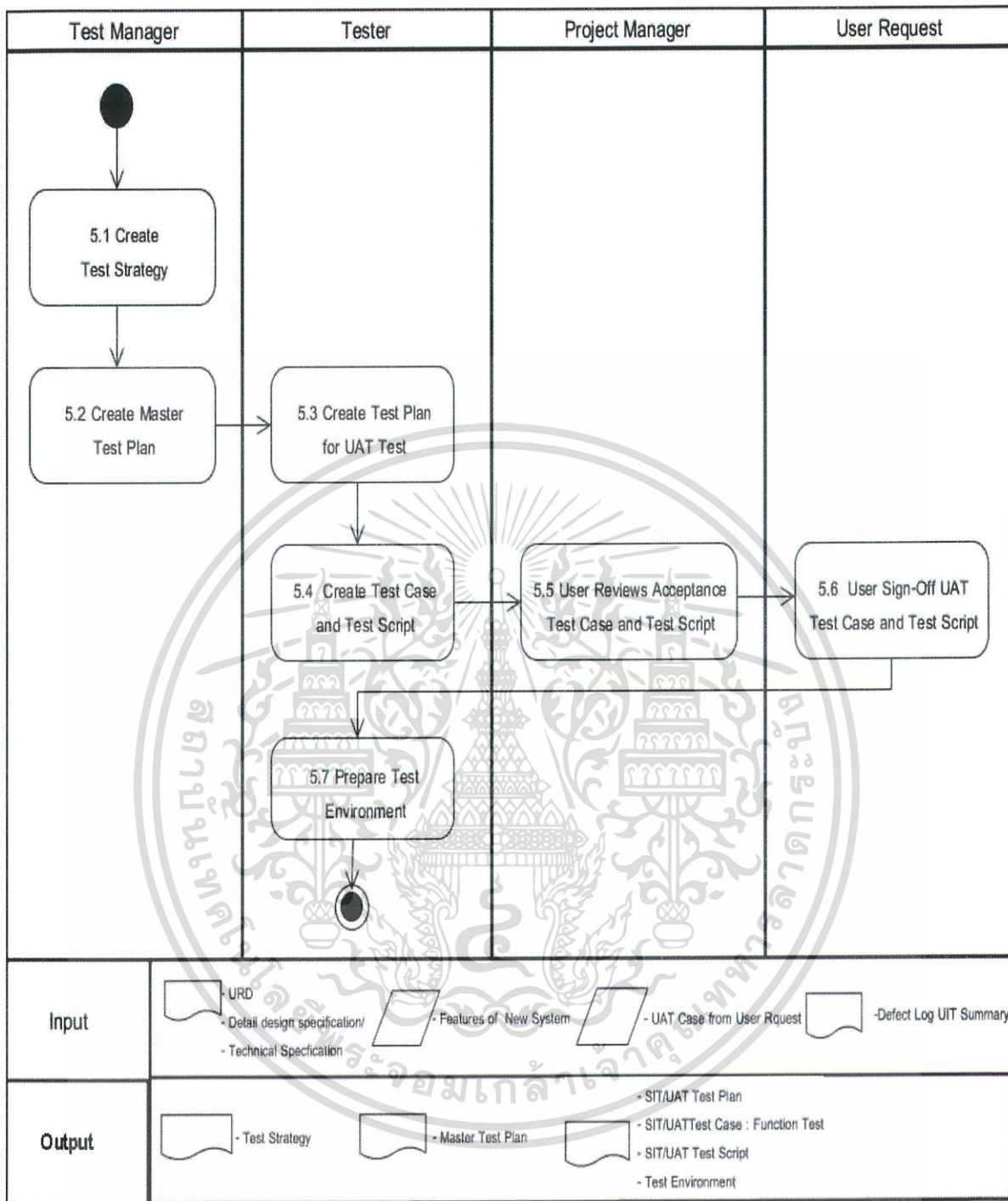
รูปที่ 3.3 แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ
 ขั้นตอนที่ 3. System Design

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



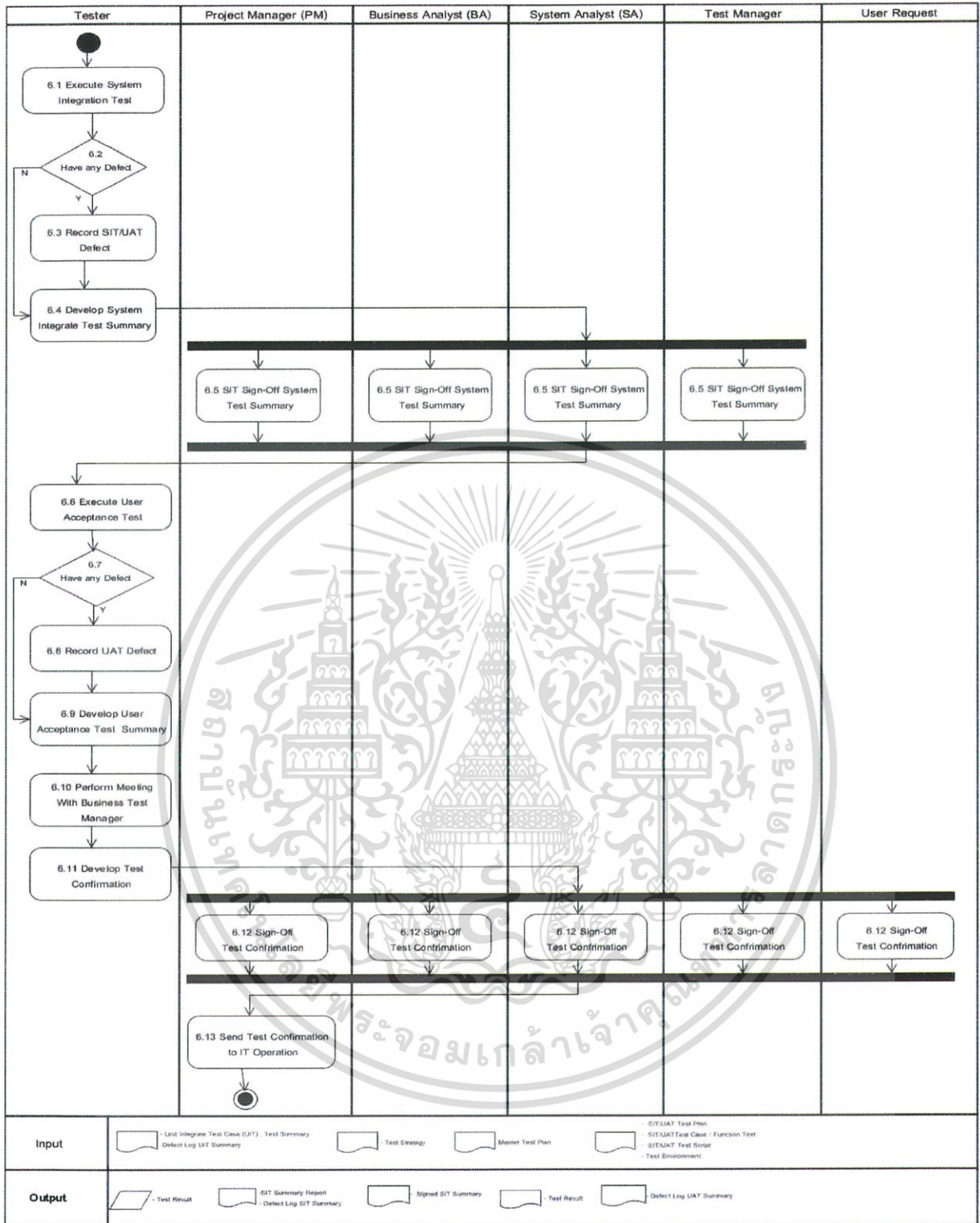
รูปที่ 3.4 แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนาการระบบกับการทดสอบระบบ
ขั้นตอนที่ 4. Implement Coding

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



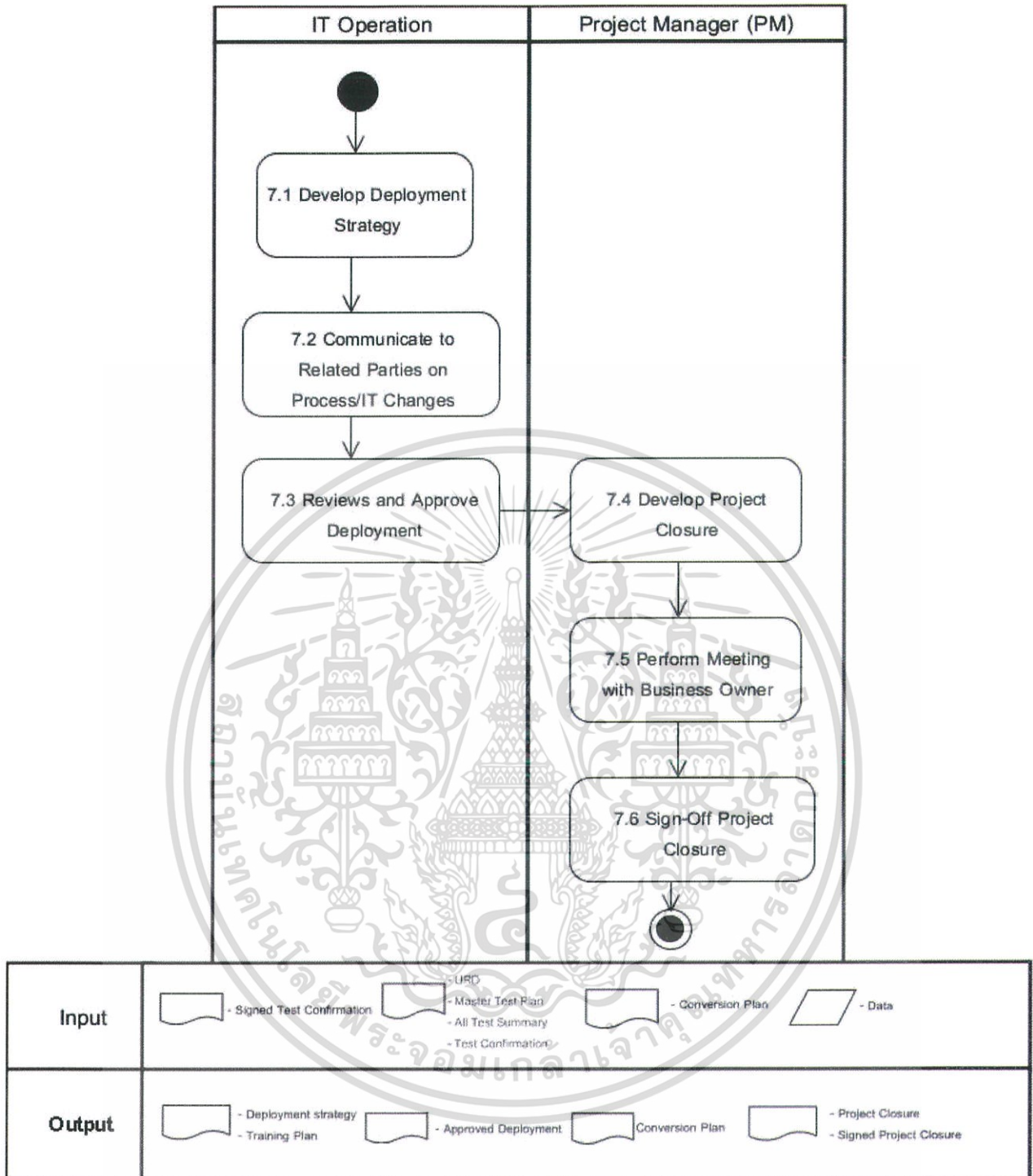
รูปที่ 3.5 แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนาระบบกับการทดสอบระบบ
ขั้นตอนที่ 5 Test Case Preparations

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ
ขั้นตอนที่ 6 Test Executions

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 แผนภาพแสดงวงจรความสัมพันธ์ระหว่างการพัฒนากระบวนการกับการทดสอบระบบ
ขั้นตอนที่: 7 Deployments

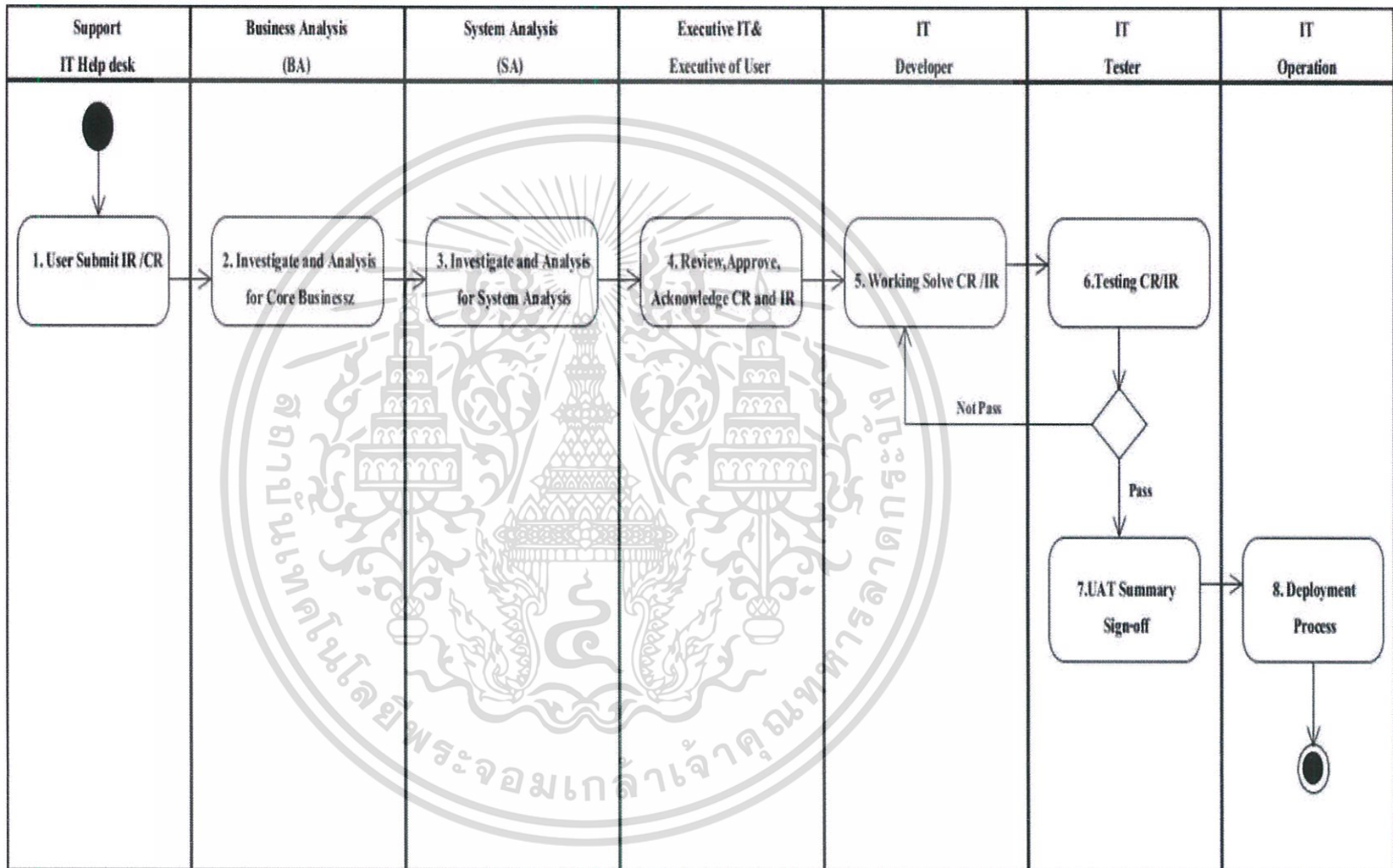
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รายละเอียดความรับผิดชอบแต่ละทีมที่เกี่ยวข้องกับความสัมพันธ์การพัฒนาระบบกับการทดสอบระบบ
 1. Project Manager (PM)
 - บริหารโครงการ สร้างแผนการทำงาน และคอย monitor ความคืบหน้าของโครงการ และคอยบริหารความเสี่ยง และ issue ต่างๆของโครงการ
 2. Business Analysis (BA)
 - จัดเก็บ Requirement จาก User เพื่อจัดทำ User Requirement Document ,Process (As-Is,To-be), Impact Analysis , Traceability matrixs
 3. System Analysis (SA)
 - วิเคราะห์ความเป็นไปได้และออกแบบ Diagram ตาม User Requirement Document, Process (As-Is, To-be), Impact Analysis, Traceability matrixs เพื่อดำเนินการพัฒนาระบบหรือประสานงานในการพัฒนาระบบกับบริษัทผู้ให้บริการ
 4. Developer
 - เขียนโปรแกรมตามเอกสาร User Requirement Document หรือแก้ไขโปรแกรมต่างๆ
 5. Test Manager
 - ออกแบบ Strategy ในการ Test และ จัดทำ Test plan ควบคุมการ Test ให้มีประสิทธิภาพและ Monitor ความคืบหน้าการทดสอบหรือ Defect จากการทดสอบของโครงการนั้น ๆ
 6. Tester / UAT (User Acceptance Test)
 - ทดสอบระบบและรายงานข้อผิดพลาด หรือประเด็นต่างๆ ที่เกิดจากการทดสอบระบบ รวมทั้งจัดทำสรุปภาพรวมการทดสอบของโครงการนั้น โดยส่วนงานทดสอบระบบสารสนเทศ มีหน้าที่การทำงานดังนี้
 1. ทีมทดสอบระบบทำการ Review เอกสาร User Requirement Document (URD) พร้อมทำการวิเคราะห์และศึกษาทำความเข้าใจกับระบบที่ต้องทำการทดสอบ ซึ่งเอกสารทั้งหมด PM ,PMO,BA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. จัดทำแผนการทดสอบ Test Plan เพื่อกำหนดระยะเวลาที่จะต้องใช้ในการทดสอบระบบนั้น ๆ และส่งข้อมูลดังกล่าวไปยังทีมงาน โครงการทั้งหมดที่เกี่ยวข้องกับ Project นั้น ๆ
3. ศึกษาและวิเคราะห์ระบบที่รับผิดชอบเพื่อจัดทำ Test Case , Test Data , Test Environment , Test Script ที่จะใช้ในการทดสอบระบบตาม Functional Requirement และ Non Functional Requirement ในเอกสาร URD
4. ดำเนินการทดสอบตาม Test Case / Test Script และเมื่อพบ Defect / Issue ระหว่างการทดสอบก็จะทำการ Log Defect ที่พบลงใน Microsoft Excel / Word เพื่อ Report Defect / Issue ที่พบแจ้งทีมที่เกี่ยวข้องดำเนินการแก้ไข
5. เมื่อทีมพัฒนาแก้ไข Defect ต่าง ๆ ที่พบเรียบร้อยแล้ว ก็จะทำการ Update Defect ที่แจ้งลงในไฟล์ Excel หลังจากที Tester นำส่งให้ ซึ่งไฟล์ Excel ดังกล่าวจะถูกเก็บไว้ใน Drive Share ของกลุ่มงานเทคโนโลยีสารสนเทศ
6. ทีมทดสอบระบบ ดำเนินการทดสอบระบบอีกครั้ง (Regression Test) ตาม Defect Report ที่ได้แจ้งแก้ไขแล้วจนถูกต้องครบทั้งหมดที่แจ้งไป
7. จัดทำสรุปภาพรวมผลการทดสอบทั้งหมด เพื่อให้ทีมงาน โครงการที่เกี่ยวข้องได้ทราบถึงข้อมูลทั้งหมดเกี่ยวกับการทดสอบระบบ รวมถึงปัญหา / ประเด็นต่างที่ยังคงค้างอยู่ ซึ่งจะมีการแก้ไขใน Release ต่อ ๆ ไป เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดงแผนภาพและขั้นตอนงานการปฏิบัติกรณีนระบบ Production มี CR (Change Request) หรือ IR (Incident Request)



ธนาคารอิสลาม
แห่งประเทศไทย
ISLAMIC BANK OF THAILAND


ISLAMIC BANK OF THAILAND
IR AND CR FORM

| | | | |
|---|---|---|--|
| For IBANK User | <input checked="" type="checkbox"/> Production IR <input type="checkbox"/> UAT IR <input type="checkbox"/> Change Request | For IT Group | Reference No. : ๓๒.(๒.) 182/2555 IR Cedar/9555-0001 |
| | | <input type="checkbox"/> IT <input type="checkbox"/> Bio <input checked="" type="checkbox"/> Outsource : Cedar | |
| For IBANK User | Application: <input type="checkbox"/> CIF <input type="checkbox"/> DEPOSIT <input type="checkbox"/> FINANCE <input type="checkbox"/> TRADE FINANCE <input type="checkbox"/> REMITTANCE <input type="checkbox"/> GL <input type="checkbox"/> ATM <input type="checkbox"/> DW <input type="checkbox"/> REPORT <input type="checkbox"/> Hire Purchase <input checked="" type="checkbox"/> OTHER SVS | | |
| Priority | <input checked="" type="checkbox"/> Urgent <input type="checkbox"/> High <input type="checkbox"/> Medium <input type="checkbox"/> Low | Attachment | <input checked="" type="radio"/> Y / <input type="radio"/> N Number of Pages 3 |
| Description : ลูกค้าสาขาลำปางบัญชีกระแสรายวันเลขที่ 298-6-01310-4 ชื่อบริษัท นำพสิ่ง จำกัด และ เลขที่ 298-6-01302-3 ชื่อบัญชี บริษัท บ่อมคลัง จำกัด ต้องการเปลี่ยนแปลงเงื่อนไขซึ่งจ่าย แต่สาขาไม่สามารถ SCAN เพิ่มลายเซ็นในระบบ Signature โดย image No ที่แสดงใน ระบบไม่ถูกต้อง เนื่องจากเดิมมีลายเซ็นถึง Image No.000027 แต่เมื่อสาขาทำการ scan ลายเซ็นที่เพิ่ม ในระบบแสดงจะ Image No.000026 ซึ่งเป็น Image No. เดียวกันที่มีอยู่แล้วและที่มีสถานะ Suspense แล้ว | | | |
| Raised By | นางอัสมา มุรทารี | Approved By | นายนคร นันทน่อ |
| Department | สาขาลำปาง | Date | 18 / 04 / 2555 |
| Contact No | 081-8811381 | Classification : <input type="checkbox"/> Error(E) <input type="checkbox"/> Enhancement(H) <input type="checkbox"/> Understanding(U) <input type="checkbox"/> Parameter(P) | |
| Date | 18 / 04 / 2555 | For IT Group/Adapt/Outsource | |
| Solution : | | | |
| Attachment Pages | | | |
| Resolved By | | Authorised By | |
| Contact No | | Date | |
| Date | | Tested/Closed By : Date : | |
| For IBANK user | Installation authorised by : Date : | | |

E-Mail : it_helpdesk@ibank.co.th / 02-650-6999 ต่อ 1111, 1101, 1100, 5134 / 089-896-9944, 089-810-0861

รูปที่ 3.9 แบบฟอร์ม Incident Request ก่อนนำมาแต่ละ Case มาทดสอบระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|--|---|--|
|  | | สำหรับโครงการ Core Banking System Change Request Form | |
| Reference Number : <u>CR CBS 4730</u> | | Date: 28 / 2 / 54 | |
| 1. การขอปรับปรุง/เปลี่ยนแปลง/เพิ่มเติม (Request for Change) | | | |
| ระบบงาน (System/Application): <input checked="" type="checkbox"/> Authorization <input type="checkbox"/> Check Processing <input type="checkbox"/> CIF <input type="checkbox"/> Deposit <input type="checkbox"/> Financing <input type="checkbox"/> Foreign Service <input type="checkbox"/> Interface <input type="checkbox"/> MIS <input type="checkbox"/> O/D <input type="checkbox"/> Printing <input type="checkbox"/> Report <input type="checkbox"/> Trade Finance <input type="checkbox"/> Other Service | | | |
| ขอบเขตที่ต้องการปรับปรุงเปลี่ยนแปลง (Scope): <input type="checkbox"/> โครงสร้างและข้อมูลหลัก (Structure and Main data) <input checked="" type="checkbox"/> กระบวนการทำงาน (Work of Processes) <input type="checkbox"/> เงื่อนไข และรูปแบบรายงานใหม่ (Condition and New Report Form) <input type="checkbox"/> แบบฟอร์มใหม่ (New Form) <input type="checkbox"/> อื่นๆ (Others) | | | |
| รายละเอียดงานที่ต้องการปรับปรุงเปลี่ยนแปลง (Description): ขอเพิ่มสิทธิ์ User Class สำหรับการทํารายการ Inter Unit ใน PD Teller สำหรับ User ID ดังนี้ 1. 540580 2. 520234 3. 541014 4. 540221 <input type="checkbox"/> Attachment Pages | | | |
| Priority : <input checked="" type="checkbox"/> Urgent <input type="checkbox"/> High <input type="checkbox"/> Medium <input type="checkbox"/> Low | | | |
| Requested By : <u>[Signature]</u> Date : <u>28/2/12</u> (iBank : Functional Team Leader) | | Approved By : <u>[Signature]</u> Date : <u>28.2.54</u> (iBank : IT Develop Team Leader) | |
| Review By : <u>[Signature]</u> Date : <u>29/2/12</u> (iBank : IT Develop Team Leader) | | | |
| 2. อนุมัติค่าใช้จ่ายในการดำเนินการ | | | |
| Total Manday: _____ | | Manday(s) | |
| Cost Manday: _____ | | Baht | |
| Total Amount: _____ | | Baht | |
| รายละเอียดเพิ่มเติม (Detail): | | | |
| <input type="checkbox"/> Attachment Pages | | | |
| Estimated By : <u>Without Bandana</u> (TNIS : Asst. PM.) | | Accepted By : <u>[Signature]</u> (iBank : IT Develop Team Leader) | |
| Approved By : <u>Without Bandana</u> (TNIS : Asst. PM.) | | Approved By : <u>[Signature]</u> (iBank : <u>นายวิญญู สุจิวัฒน์พงศ์</u>) | |
| Date : <u>29/2/2012</u> (iBank : Functional Team Leader) | | Date : _____ Date : _____ 29 ก.พ. 2555 | |

รูปที่ 3.10 แบบฟอร์ม Change Request ก่อนนำมาแต่ละ Case มาทดสอบระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ปัญหาของการทำงานปัจจุบัน

จากการศึกษาและวิเคราะห์ข้อมูลต่าง ๆ ของระบบงานปัจจุบันพบปัญหาดังนี้

1. ปัญหาจากการทำงานปัจจุบันที่พบในเอกสารต่าง ๆ ที่เกี่ยวข้องกับการทดสอบระบบ
 - พบข้อผิดพลาดจากเอกสารการศึกษาระบบ (System Study document)
 - พบข้อผิดพลาดจากเอกสารคุณสมบัติการทำงาน (Functional Specification document)
 - พบข้อผิดพลาดจากเอกสารการออกแบบสถาปัตยกรรม (Architectural Design Document)
 - พบข้อผิดพลาดจากเอกสารรายละเอียดการออกแบบ (Detailed Design document)
 - พบข้อผิดพลาดจาก Source Code
 - พบข้อผิดพลาดจากแผนทดสอบ / กรณีทดสอบ
 - พบข้อผิดพลาดจากเอกสารที่ใช้ประกอบ: คู่มือการใช้งาน คู่มือการปฏิบัติ

ข้อผิดพลาดจากเอกสารต่างๆ ที่พบในช่วงต้นอาจจะมีปัญหาอาทิเช่น เอกสารไม่ละเอียด มีการแก้ไขเปลี่ยนแปลงบ่อย หรือเอกสาร ไม่สอดคล้องตรงตามความต้องการของผู้ใช้ ทำให้มีผลกระทบ หรือพบข้อผิดพลาดจากการทดสอบระบบในทุก ๆ ขั้นตอนทำให้ผลการทดสอบมีข้อผิดพลาดเยอะ และเกิดความล่าช้าในการพัฒนาระบบนั้นๆ เป็นเหตุให้โครงการที่พัฒนาระบบไม่เป็นไปตามแผนการดำเนินงาน โครงการฯ
2. การบันทึก Defect/Issue ที่เกิดจากการทดสอบระบบกระจัดกระจาย ไปตามความรับผิดชอบของผู้ทดสอบแต่ละคน เนื่องจากเก็บบันทึกในรูปแบบ ไฟล์ Excel /Word ซึ่งยากในการสืบค้นและติดตาม รวมถึงไม่สามารถกำหนดสิทธิการเข้าถึงข้อมูลและการปรับปรุงข้อมูลได้ อีกทั้งยังไม่สนับสนุนการทำงานในสถานะพร้อมกัน แสดงดังในรูปที่ 3.11
3. ระบบงานปัจจุบันไม่สามารถจัดการ Assign ให้กับทีมที่เกี่ยวข้องได้ทันที ทำให้เสียเวลาและเกิดการล่าช้าในการแก้ไข Defect/Issue
4. ทีมผู้เกี่ยวข้องใน Project นั้น ๆ ไม่ทราบถึงสถานะที่แน่นอนของ Defect/Issue นั้นว่าดำเนินการแก้ไขถึงไหนแล้ว และต้องใช้เวลาการแก้ไขนานแค่ไหน ทำให้งานไม่เป็นไปตามกระบวนการทำงานของ IT
5. ไม่สามารถติดตามและรายงานข้อผิดพลาดของระบบ ได้อย่างถูกต้องและรวดเร็ว เมื่อผู้บริหารต้องการข้อมูลสรุปปัญหาของ Project นั้น ๆ ว่าสามารถ Launch ได้ตาม Plan หรือไม่และพร้อมนำขึ้นใช้งานจริงได้หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ผู้ทดสอบระบบต้องเสียเวลาในการจัดการด้านเอกสารจำนวนมาก ทำให้การทดสอบล่าช้าไม่เป็นตามแผนการทดสอบ
7. ไม่สามารถสรุปภาพรวมผลการทดสอบทั้งหมดได้ทันที ทำให้ไม่สามารถปิดผลการทดสอบที่พบข้อผิดพลาดได้ จึงทำให้เกิดความล่าช้าและมีผลกระทบในการนำระบบขึ้นใช้งานจริง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| Defect Log - ทดสอบโปรแกรม (ATM Activate Card On BDS System) | | | | | | | | | | | | | | | |
|---|-----------|---------------------------------|---|---------------|------------------------|-----------|---------------|---|-------------|------------|----------|------------|-------------|-------------------|-------------------|
| DF_ID | DF_Status | Summary | Description | Test Case No. | Sheet Refer | Assign To | Assigned Date | Solution | Defect Date | Severity | Priority | Close Date | Defected By | Defect in Version | Defected In Cycle |
| DF_001 | Closed | search by ID number not success | Can not find customer information by using search by ID number | - | DF_001 | Developer | - | SolveAlready :About Source code Incorrect | 15/3/2011 | Severity 1 | High | 23/3/2011 | Suchalai | 1.0 | 1.0 |
| DF_002 | Closed | Can not add new account | Customer have two deposit account, we created primary account link with ATM card and want to link second deposit account but can not found account on dropdown list | - | DF_002 | Developer | - | SolveAlready | 15/3/2011 | Severity 1 | High | 23/3/2011 | Suchalai | 1.0 | 1.0 |
| DF_003 | Closed | Card Activate | Why can card activate again and replace the existing record too you can see account that change | - | DF_003 | Developer | - | SolveAlready | 15/3/2011 | Severity 1 | High | 21/3/2011 | Suchalai | 1.0 | 1.0 |
| DF_004 | Closed | change status | new card status should show only 3 status following : 1 open 4 Restricted 9 Closed | - | DF_004 | Developer | - | SolveAlready :Add Status already | 16/3/2011 | Severity 1 | High | 21/3/2011 | Suchalai | 1.0 | 1.0 |
| DF_005 | Reject | ATM Report | Unable to display the report | - | DF_005 | Developer | - | SolveAlready :About Source code Incorrect and Message send to Web service Incorrect | 16/3/2011 | Severity 1 | High | 23/3/2011 | Suchalai | 1.0 | 1.0 |

รูปที่ 3.11 ตัวอย่างการบันทึกข้อผิดพลาดจากการทดสอบระบบ ATM

- อธิบายตัวอย่างการบันทึกข้อผิดพลาดจากการทดสอบระบบ ซึ่งในบันทึกแต่ละครั้งผู้ทดสอบแต่ละคนอาจจะมีการบันทึกข้อผิดพลาดที่แตกต่างกัน หรือคล้ายกันได้ ทั้งนี้มีรายละเอียดตัวอย่างแต่ฟิลด์ข้อมูลดังนี้

1. DF_ID : เลขที่ข้อผิดพลาด Run Number Manual
2. DF_Status : บอกถึงสถานะของ Defect ที่พบ Status ที่ใช้ปัจจุบันจะมีอยู่ 6 status ดังนี้ New, Assign ,Inprogress,Reopen,Closed,Reject
3. Summary : สรุปหัวข้อที่พบ Defect
4. Description : รายละเอียด Defect ที่พบ
5. Test Case : อ้างอิง Test case ที่ทดสอบพบ Defect
6. Sheet Refer : ลิงก์ Sheet ในMicrosoft Excel ที่มีการ Capture หน้าจอ Defect ที่พบในการทดสอบ
7. Assign To : เลือกชื่อผู้รับผิดชอบ ในส่วนของ Defect ที่พบเพื่อส่ง Defect ให้แก้ไข
8. Assign Date : วันที่ที่เลือกผู้รับผิดชอบ เพื่อส่ง Defect ให้แก้ไข
9. Solution : รายละเอียดการแก้ไข ว่าได้รับการแก้ไข Defect อย่างไร หรือได้แก้ไขแล้วเรียบร้อย
10. Defect Date : วันที่ที่พบ Defect
11. Severity : ระดับความรุนแรงของ Defect นั้น ๆ
12. Priority : ความสำคัญของ Defect นั้น ๆ
13. Closed Date : วันที่ที่ Defect ได้รับการแก้ไข และทดสอบผ่านแล้ว
14. Defect By : ผู้ทดสอบที่พบ Defect
15. Defect in version : การทดสอบระบบ Version ของระบบที่พบ Defect
16. Defect Cycle : รอบในการทดสอบ Defect นั้น ๆ

3.3 ความต้องการระบบงานใหม่

1. ความต้องการเชิงฟังก์ชัน (Functional Requirements)

จากศึกษาปัญหาของระบบงานในปัจจุบัน ทำให้ทราบถึงความต้องการของระบบใหม่ ซึ่งจะช่วยให้การติดตามและรายงานข้อผิดพลาดหรือประเด็นต่าง ๆ ที่เกิดจากทดสอบระบบ แสดงข้อมูลได้อย่างถูกต้อง สะดวก รวดเร็วและลดความซ้ำซ้อน ทำให้เกิดประสิทธิภาพในกระบวนการทดสอบระบบมากยิ่งขึ้น โดยความต้องการในระบบงานใหม่มีดังต่อไปนี้

1.1 ระบบสามารถสร้าง/บันทึกผลการทดสอบกรณีเจอ Defect หรือ Issue ตาม Test Case ที่ทำได้ไว้ โดยกรอกรายละเอียดเบื้องต้นดังนี้

- **Category** : กลุ่มประเภทข้อมูล อาทิ ข้อผิดพลาดที่พบจากการทดสอบระบบ (Defect) หรือ ประเด็นเพิ่มเติมจากการทดสอบระบบ (Issue)
- **Topic** : หัวข้อรายละเอียดที่พบ Defect
- **Priority** : ระบุความสำคัญของ Defect ที่พบ (Non, Low, High, Urgent, Immediate)
- **Assign To** : เลือกชื่อผู้รับผิดชอบ ในส่วนของ Defect ที่พบ
- **Defect Detail** : เขียนรายละเอียดของ Defect ที่พบ รวมทั้งระบบ URL, Test Data ที่ใช้ในการทดสอบ
- **Test Steps** : เขียนขั้นตอนการทดสอบที่พบ Defect
- **Note Log defect**: ระบุผลการทดสอบที่พบ, ผลการทดสอบที่ควรจะเป็น และ ไฟล์แสดงผลการทดสอบ เพื่อใช้อธิบาย Defect ที่พบ
- **Attachment** : Attach file แสดงผลการทดสอบ ขนาดไฟล์ต้องไม่เกิน 5 เมกะไบต์
- **Defect Status** เป็นการบอกถึงสถานะของ Defect ที่พบ Status ที่ใช้ปัจจุบันจะมีอยู่ 5 status ดังนี้
 - **New** : เมื่อพบ Defect ใหม่
 - **Assign** : เมื่อพบ defect จะ Assign ไปให้ทาง Team Develop ที่รับผิดชอบ
 - **Fixed** : Developer แก้ไข เมื่อ Developer แก้ไข Defect ที่พบ และ upload Code ขึ้น Test Environment
 - **Resolved** : Developer แก้ไขแล้ว และ Test Manager ทำการตรวจสอบว่าที่ Fixed มาตรฐานกับ Defect ที่พบ และส่งให้ Tester Verify defect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Retest** : Tester ทำการ Retest Defect อีกครั้งหลังจาก Fixed และ Resolved แล้วเรียบร้อย
 - **Reopen** : กรณีทำการ Retest Defect ไม่ผ่านต้องเปิด Defect ตัวเดิมใหม่อีกครั้ง
 - **Reject** : กรณี Defect ที่พบไม่ใช่ Defect หรือไม่เกี่ยวข้องกับระบบฯ
 - **Closed** : Defect ได้รับการแก้ไขถูกต้อง
- **Defect Type** เป็นการแบ่งประเภทของ Defect ที่พบ ว่าเกิดจากส่วนใดในการทำงาน สามารถแบ่งออกได้เป็น 6 ประเภท
- **Requirement** : เป็น Defect ที่เกิดจากการแก้ไข Requirement ของทาง Business โดยไม่แจ้งทีมที่เกี่ยวข้อง (QA ,Developer, Tester ,Project PM)
 - **Coding** : เป็น Defect ที่เกิดจากการ Coding ของทาง Developer ที่ไม่ตรวจสอบในส่วนนั้นๆ
 - **Graphic Design** : เป็น Defect ที่เกิดจากการ Design ที่ไม่รองรับกับ Browser ต่างๆ หรือ เมื่อ developer นำ Design มาประกอบกับ Code แล้วทำให้การแสดงผลไม่ถูกต้อง
 - **Tester** : เป็น Defect ที่เกิดจากความเข้าใจผิด หรือเกิดจากความผิดพลาดของ Tester
 - **Data Test** : เป็น Defect ที่เกิดจาก Test data อาจไม่มีใน Environment หรือระบบอาจไม่ Support data ส่วนนี้
 - **Other** : ข้อจำกัดของระบบ, ข้อจำกัดของ Environment
- **Defect Severity** เป็นการระบุความรุนแรงของ Defect ที่พบใน Test Phase โดยยึดตาม Standard Severity Code แบ่งออกเป็น 4 ระดับ ดังนี้
- **Critical** : Defect ที่ไม่สามารถทดสอบโปรแกรมในส่วนของ Function นั้นต่อได้เลย
 - **High** : Defect ที่เกิดจากการใส่ข้อมูลถูกต้อง แต่ระบบแสดงผลผิดพลาด เช่น Error ต่างๆ
 - **Medium** : ระบบจะแสดงผลถูกต้องเมื่อใส่ข้อมูลถูกต้อง แต่เมื่อใส่ข้อมูลไม่ถูกต้อง ระบบจะแสดงผลผิดพลาด เช่น Filed ที่มีการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Validate ผล เมื่อใส่ ค่าว่าง,อักขระพิเศษ (;',%,&) และ Script ที่มีผลต่อการแสดงผลของระบบ ๆ จะแสดงผลผิดพลาด

- **Low** : Defect ที่เกิดจากการแสดงผลของข้อความ หรือ เรื่องของการ Design ซึ่ง Defect เหล่านี้จะไม่มีผลกระทบต่อการทำงานของระบบ

1.2 ระบบสามารถแสดงรายงานสรุปผล Defect ทั้งหมดทุกสถานะของการทดสอบ และจำนวนข้อผิดพลาดในการทดสอบ ในรูปแบบเนื้อหาและกราฟ

1.3 ระบบสามารถ Assign และแจ้งเตือนในรูปแบบอีเมล และ Alert บนหน้าจอของระบบดังกล่าว ทุกกรณีที่มีเมื่อมีการพบ Defect ในกรณีที่ เป็น Defect ที่รุนแรงมากหรือ Defect มีระยะเวลาจำกัด ระบบจะมีสัญลักษณ์ Urgent เพื่อบอกความสำคัญของ Defect นั้น โดยระบบจะแจ้งเตือนผู้รับผิดชอบใน Project นั้น ๆ ทีมงานโครงการ (Project Team) ได้แก่ Project Manager, Business Analysis (BA), System Analysis, Test Manager เป็นต้น

1.4 ระบบสามารถออกเอกสาร User Acceptance Test พร้อมสรุปรายงานข้อผิดพลาดของระบบ

1.5 ระบบสามารถกำหนดสิทธิ์การใช้งานของผู้ใช้งานได้ โดยผู้บริหารและพนักงานปฏิบัติการแต่ฝ่ายงานของ IT จะมีสิทธิ์การใช้แตกต่างตามความรับผิดชอบ

2. ความต้องการที่ไม่เกี่ยวกับฟังก์ชัน (Non-functional Requirements)

2.1 Operational Requirement

- ระบบเป็นเว็บแอปพลิเคชันเพื่อให้สามารถใช้งานได้ทุกที่
- เขียนโปรแกรมของระบบด้วยภาษา ASP.NET, HTML, Adobe Dreamweaver CS5 และใช้ฐานข้อมูล Microsoft SQL Server

2.2 Performance Requirement

- ระบบมีประสิทธิภาพในการรองรับผู้ใช้งานได้ถึง 50 เวิร์กสเตชันในเวลาเดียวกัน และสามารถแสดงผลข้อมูลบนจอภาพ ไม่เกิน 10 วินาที
- ระบบมีความน่าเชื่อถือต่อการใช้งาน โดยจะไม่มีข้อผิดพลาดต่อการใช้งาน และมีความถูกต้อง รวดเร็ว สามารถตรวจสอบได้
- ระบบมีการออกแบบมาให้สามารถใช้งานได้สะดวกและง่ายต่อการใช้งาน

2.3 Security Requirement

- ระบบมีความปลอดภัยต่อการใช้งานเพราะทีมผู้เกี่ยวข้องกับระบบต้องมี ผู้เข้าใช้ระบบ (Username) และรหัสผ่าน (Password) ในการยืนยันตัวตนเมื่อเข้าใช้ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 แผนภาพที่เกี่ยวข้องกับการวิเคราะห์และออกแบบระบบ

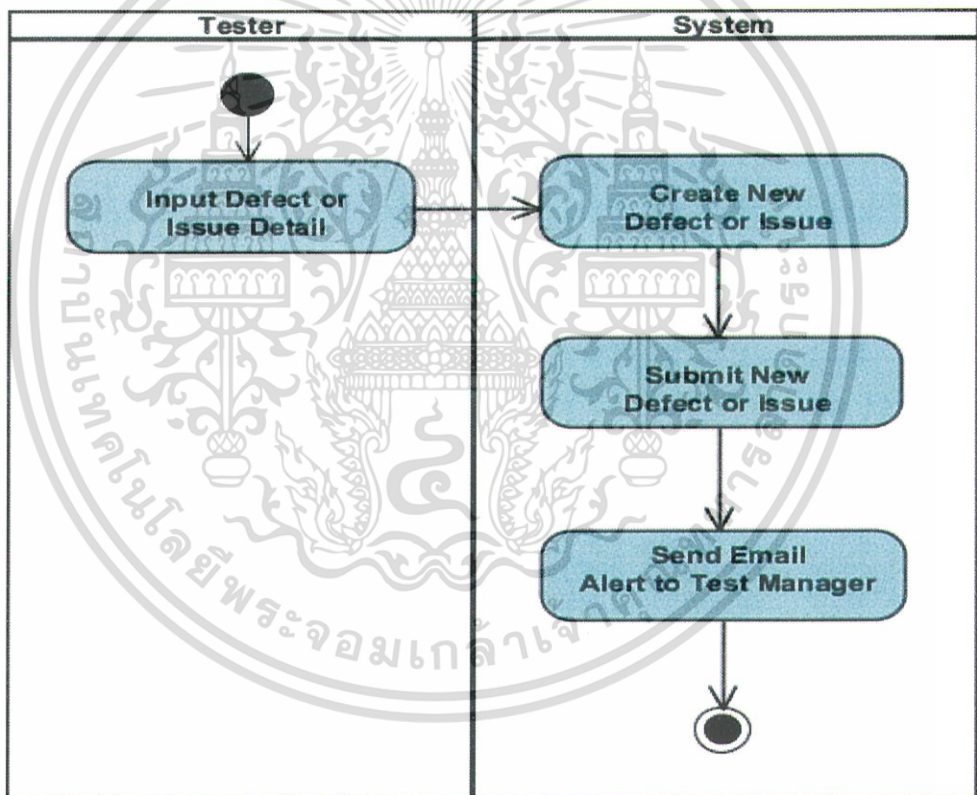
ระบบติดตามเพื่อติดตาม และรายงานข้อผิดพลาดที่เกิดจากการทดสอบระบบ ได้นำยูเอ็มแอลมาช่วยในการวิเคราะห์และออกแบบระบบ ประกอบด้วยไดอะแกรมต่าง ๆ ดังต่อไปนี้

3.4.1 ภาพแสดงแอกทิวิตีไดอะแกรม (Activity diagram)

ไดอะแกรมแสดงให้เห็นลำดับการดำเนินกิจกรรม ซึ่งมีกิจกรรมดังนี้

1. แอกทิวิตีไดอะแกรมของการสร้าง หรือ บันทึกข้อผิดพลาด (Create New Defect)

ไดอะแกรมแสดงให้เห็นถึงลำดับการดำเนินกิจกรรมของการสร้างหรือบันทึกข้อผิดพลาด โดยการกรอกข้อมูลข้อผิดพลาด หรือประเด็นการแก้ไขระบบ ฯ ที่เกิดจากการทดสอบระบบ และทำการบันทึกข้อมูล หลังจากระบบจะส่งงานไปยัง Test Manager ซึ่งรูปแบบแจ้งเตือนอีเมลและกล่องงานส่วนตัวในระบบของ Test Manager ดังแสดงในรูปที่ 3.12

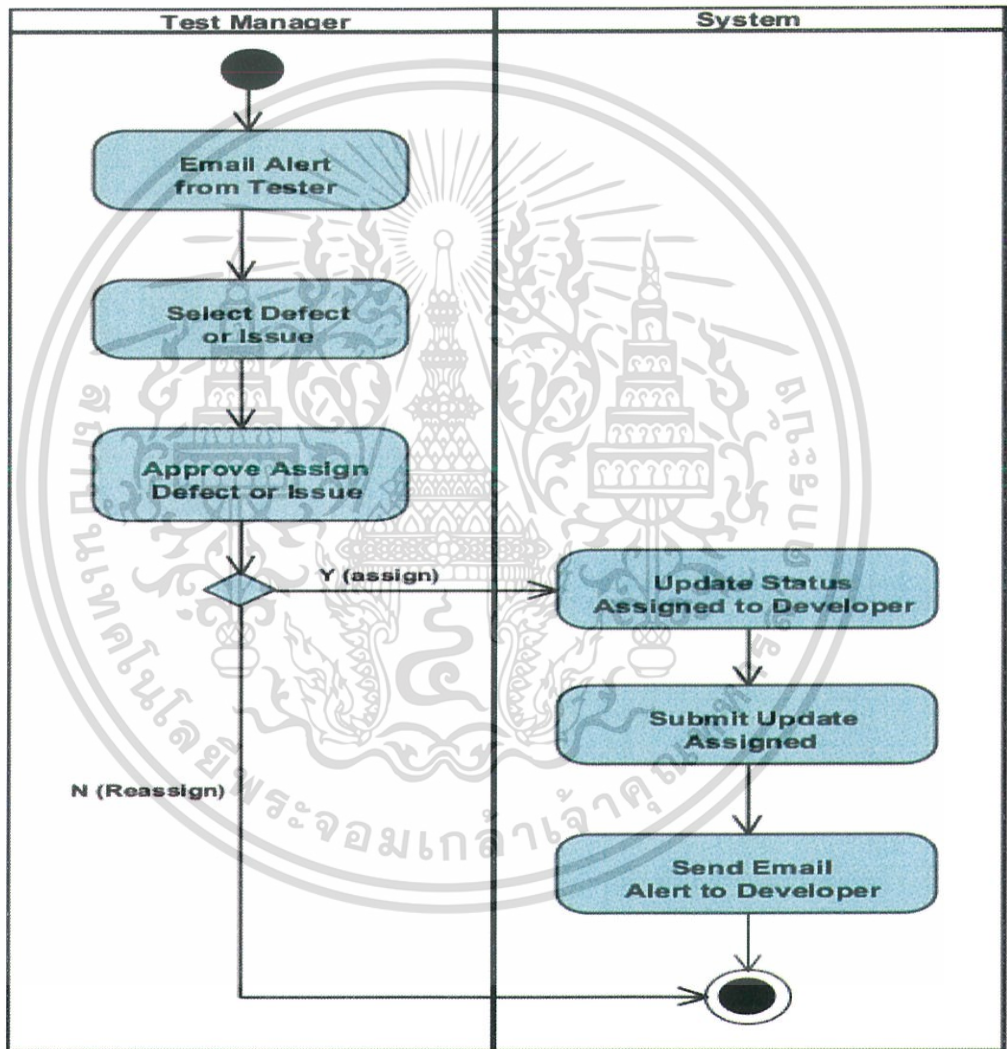


รูปที่ 3.12 แสดงแอกทิวิตีไดอะแกรมของการสร้าง หรือ บันทึกข้อผิดพลาด
(Create New Defect)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. แยกทิวทัศน์ไดอะแกรมของการปรับปรุงแก้ไขข้อผิดพลาดและมอบหมายงานข้อผิดพลาดของ Test Manager (Update and Assign Defect)

ไดอะแกรมแสดงให้เห็นถึงลำดับการดำเนินกิจกรรมของการปรับปรุงแก้ไขข้อผิดพลาดและมอบหมายงานข้อผิดพลาดไปยัง Developer โดย Test Manager จะได้รับอีเมลแจ้งเตือนงานที่ส่งมาจาก Tester หลังจากนั้นจะทำการเลือกข้อผิดพลาดหรือประเด็นแก้ไขเพื่อทำการตรวจสอบเพื่ออนุมัติข้อผิดพลาดและมอบหมายงานไปยัง Developer เพื่อแก้ไขต่อไป ซึ่งเมื่อบันทึกข้อมูลมอบหมายแล้วระบบจะแจ้งเตือนอีเมลและกล่องงานส่วนตัวในระบบของ Developer ดังแสดงในรูปที่ 3.13

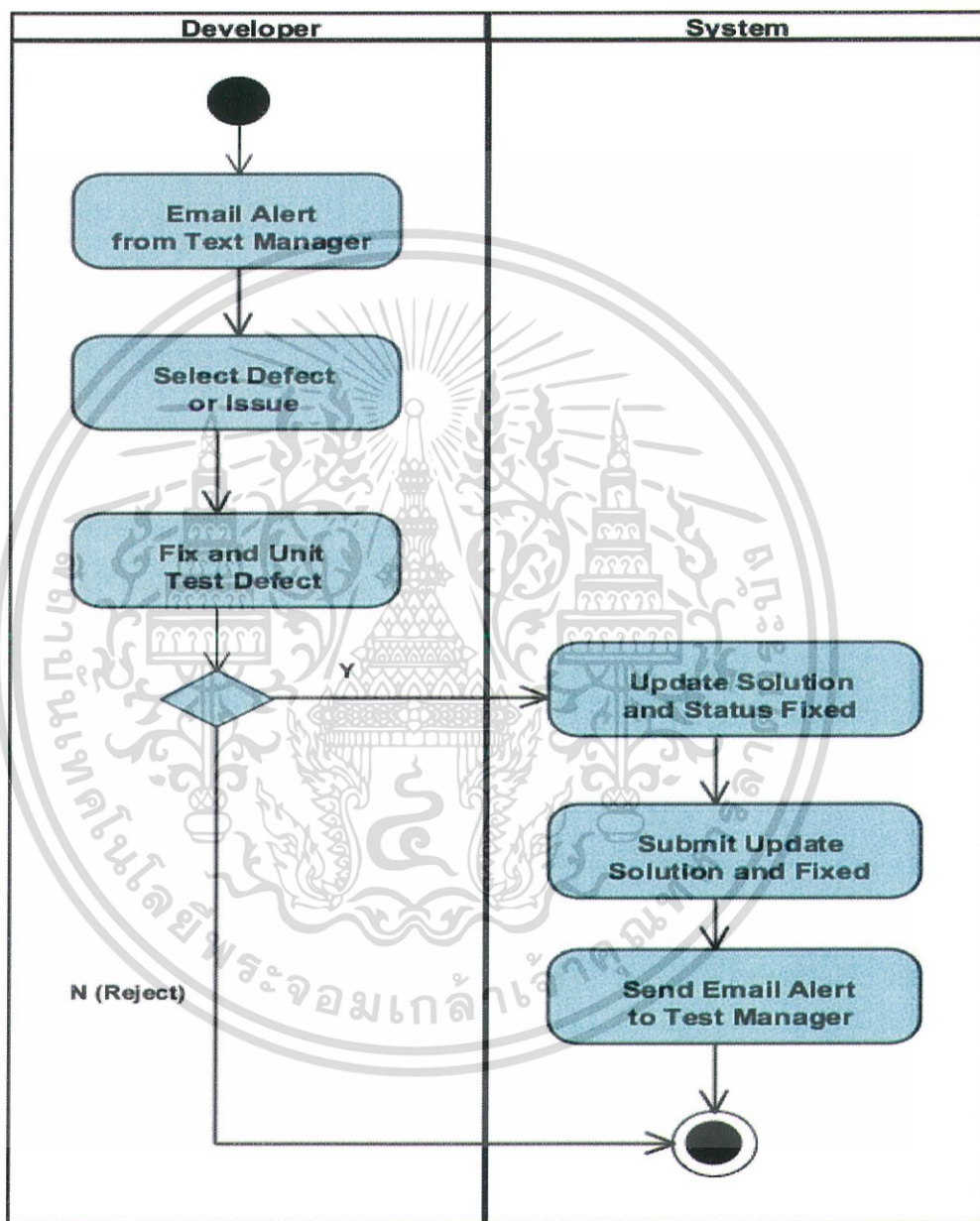


รูปที่ 3.13 แยกทิวทัศน์ไดอะแกรมของการปรับปรุงแก้ไขข้อผิดพลาดและมอบหมายงานข้อผิดพลาดของ Test Manager (Update and Assign Defect)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. แอกทิวิตีไดอะแกรมของการปรับปรุงแก้ไขข้อผิดพลาดของ Developer (Update Defect)

ไดอะแกรมแสดงให้เห็นถึงลำดับการดำเนินการกิจกรรมของการปรับปรุงแก้ไขข้อผิดพลาด เป็นการแก้ไขโปรแกรมที่พบข้อผิดพลาดและเพิ่มข้อมูล Solution ต่าง ๆ ที่เกี่ยวกับข้อผิดพลาดนั้น ๆ และเมื่อทำการบันทึกข้อมูลแล้ว ระบบจะแจ้งเตือนกลับมายัง Test Manager เพื่อทำการตรวจสอบ Solution ต่อไป ดังแสดงในรูปที่ 3.14

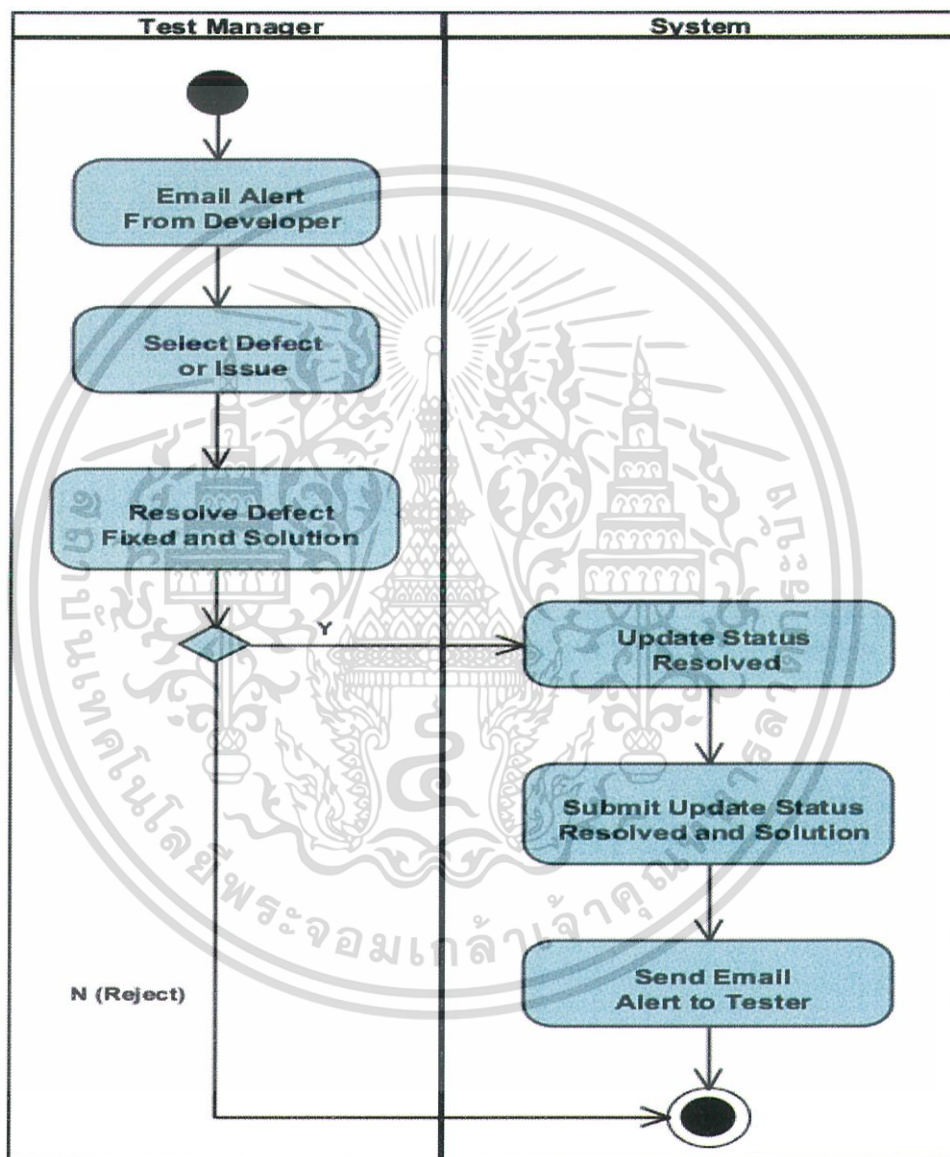


รูปที่ 3.14 แสดงแอกทิวิตีไดอะแกรมของการปรับปรุงแก้ไขข้อผิดพลาดของ Developer (Update Defect)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. แยกทิวทัศน์ไคอะแกรมของการตรวจและตัดสินใจเลือก Solution (Resolve Defect)

ไคอะแกรมแสดงให้เห็นถึงลำดับการดำเนินงานกิจกรรมของการตรวจและตัดสินใจเลือก Solution เพื่อให้ Tester ทดสอบระบบหรือการ Retest ข้อผิดพลาดอีกครั้ง โดย Test Manager จะได้รับอีเมลแจ้งเตือนงานที่ส่งมาจาก Developer ว่าข้อผิดพลาดได้รับการแก้ไขแล้ว ซึ่ง Test Manager จะต้องตรวจสอบว่า Solution ที่ทาง Developer แก้ไขสามารถทำงานได้จริง เป็นไปได้หรือไม่ ดังแสดงในรูปที่ 3.15

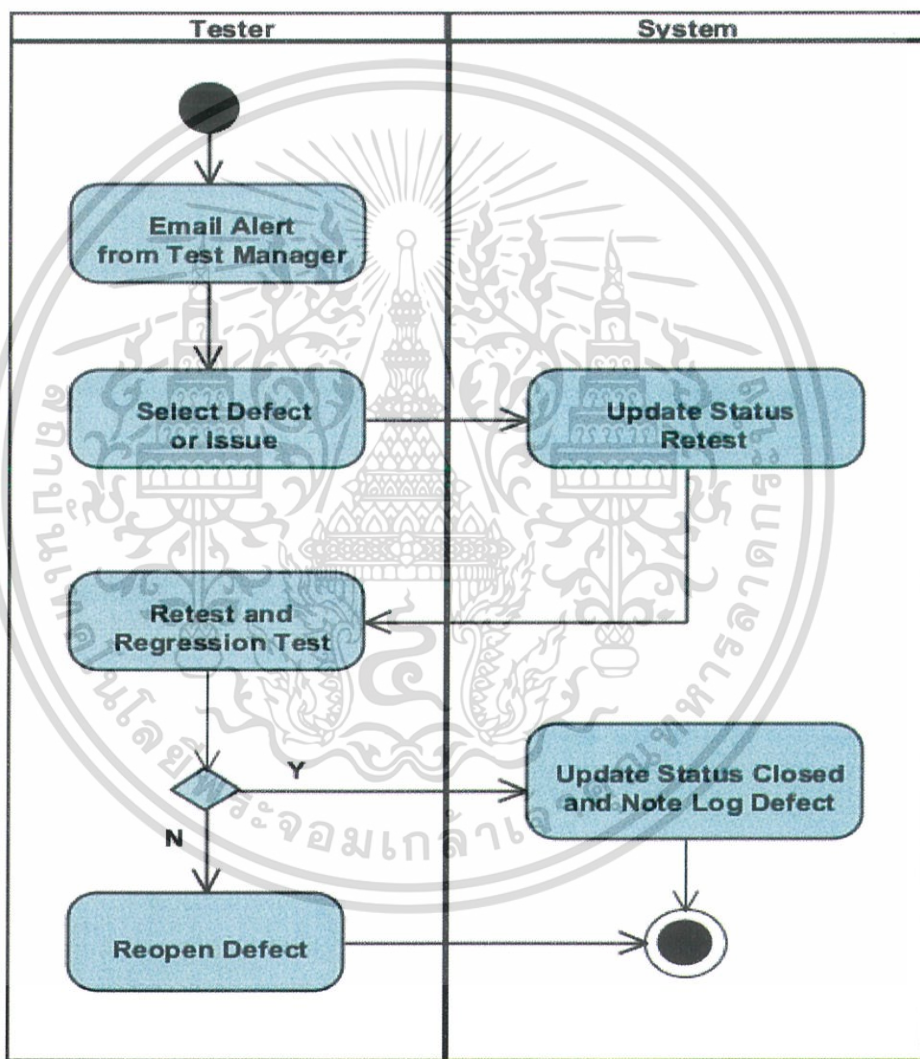


รูปที่ 3.15 แสดงแยกทิวทัศน์ไคอะแกรมของการตรวจและตัดสินใจเลือก Solution (Resolve Defect)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. แยกทิวทัศน์ไดอะแกรมของการปรับปรุงแก้ไขข้อผิดพลาดและปิดสถานะข้อผิดพลาด (Update and Closed Defect)

ไดอะแกรมแสดงให้เห็นถึงลำดับการดำเนินงานของการปรับปรุงแก้ไขข้อผิดพลาดและปิดสถานะข้อผิดพลาด โดย Tester จะได้รับอีเมลแจ้งเตือนที่ส่งมาจาก Test Manager ว่าข้อผิดพลาดได้รับการแก้ไขและตรวจสอบแล้ว หลังจากนั้น Tester ต้องทำการ Retest และ Regression Test ถ้าทำการทดสอบข้อผิดพลาดผ่าน ก็ทำการบันทึกข้อมูลและปรับปรุงแก้ไขสถานะเป็น Closed และกรณีไม่ผ่าน ก็ทำการ Reopen ข้อผิดพลาดเพื่อทำการส่งไปยังผู้เกี่ยวข้องต่อไป ดังแสดงในรูปที่ 3.16

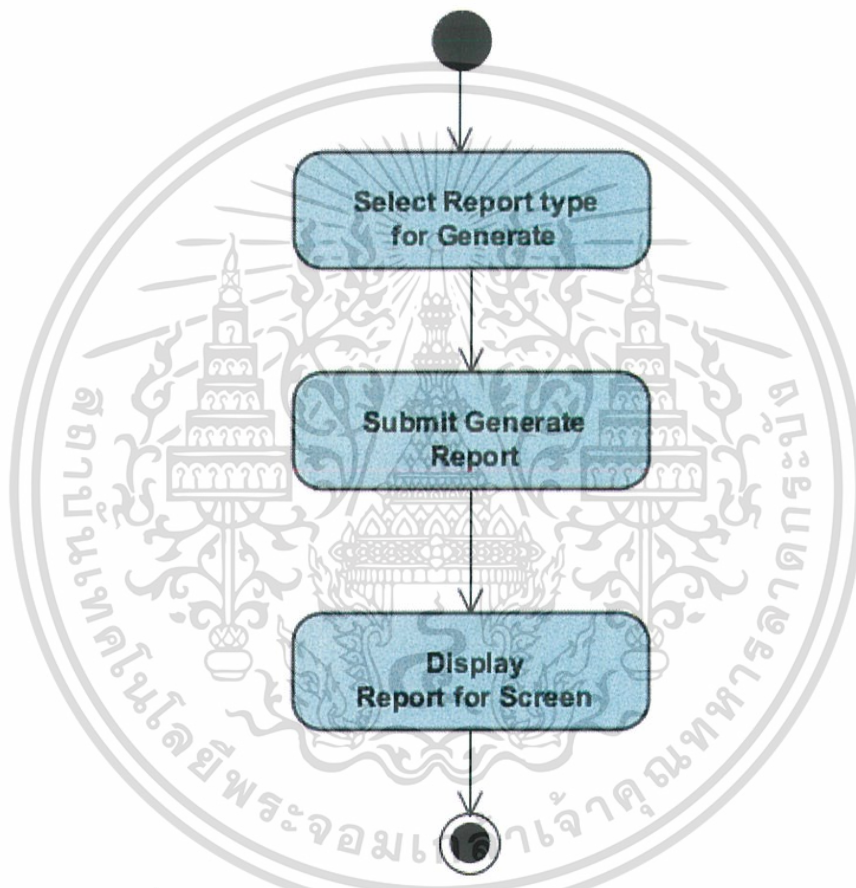


รูปที่ 3.16 แสดงแยกทิวทัศน์ไดอะแกรมของการปรับปรุงแก้ไขข้อผิดพลาดและปิดสถานะข้อผิดพลาด (Update and Closed Defect)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. แยกทิวทัศน์ไคอะแกรมของการสร้างรายงานสรุปต่าง ๆ (Generate Summary Report and Chart)

ไคอะแกรมแสดงให้เห็นถึงลำดับการดำเนินงานกิจกรรมของการสร้างรายงานสรุปประเภทต่าง ๆ ซึ่งผู้เกี่ยวข้องกับขั้นตอนนี้ได้แก่ Test Manager, Project Manager, Business Analyst, System Analyst สามารถเลือกดูรายงาน อาทิเช่น รายงานสรุปสถานะทั้งหมดของระบบฯ หรือ รายงานสรุปข้อผิดพลาดของโครงการที่ทำการทดสอบ ดังแสดงในรูปที่ 3.17

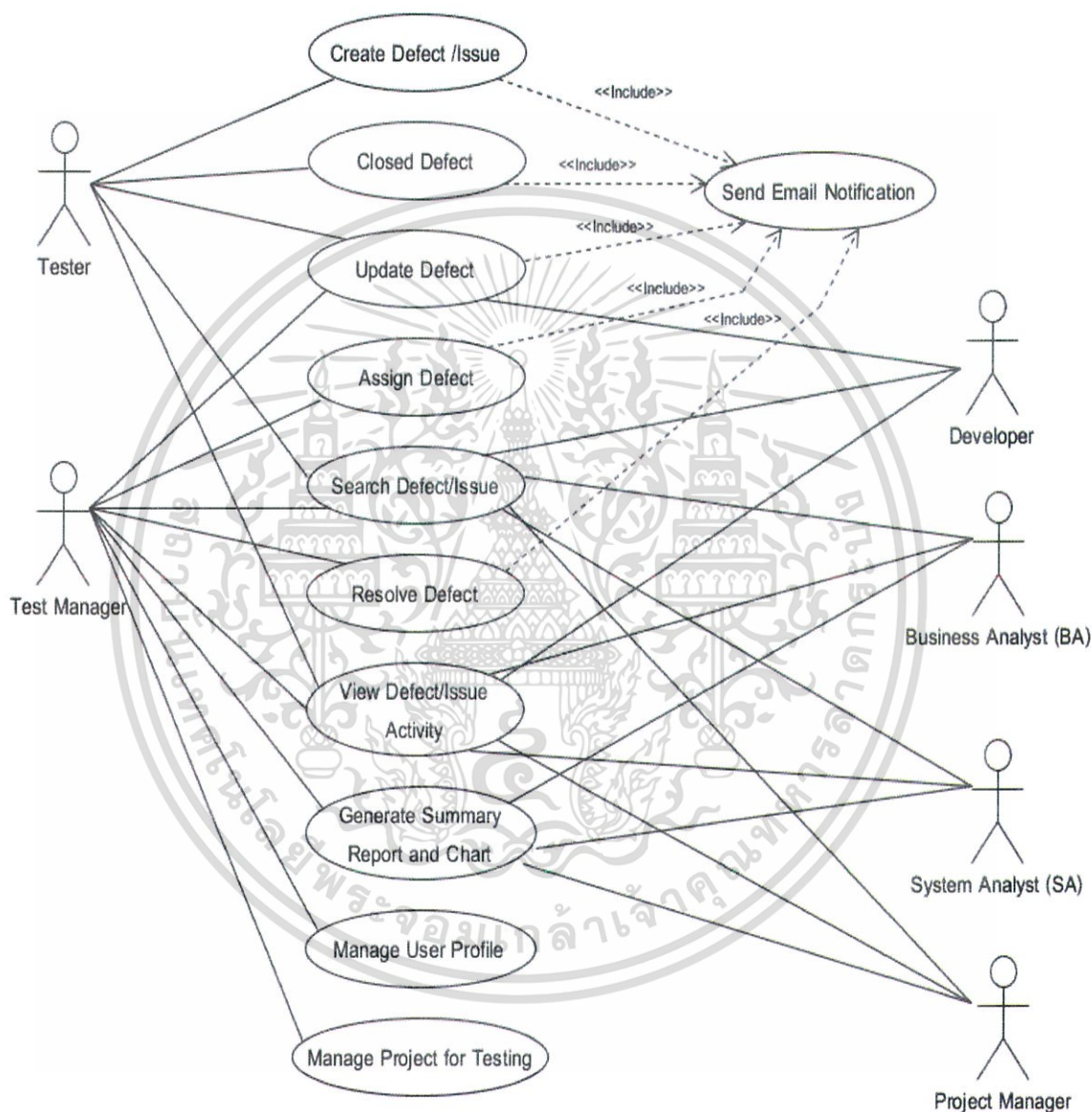


รูปที่ 3.17 แสดงแยกทิวทัศน์ไคอะแกรมของการสร้างรายงานสรุปต่าง ๆ
(Generate Summary Report and Chart)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2 ภาพแสดง ยูสเคสไดอะแกรม (Use Case Diagram)

จากการศึกษาความต้องการระบบ สามารถสร้างยูสเคสไดอะแกรมซึ่งแสดงฟังก์ชันหลักและอธิบายการทำงานของระบบยูสเคสไดอะแกรมจะแสดงรายละเอียดของแอกเตอร์และยูสเคส ที่แสดงการทำงานของระบบโดยรวมทั้งหมด ซึ่งยูสเคสไดอะแกรมที่ออกแบบมียูสเคสและแอกเตอร์ต่างๆ ดังรูปภาพด้านล่าง



รูปที่ 3.18 แสดงยูสเคสไดอะแกรมของระบบติดตามเพื่อติดตามและรายงานข้อผิดพลาดที่เกิดจากการทดสอบระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากยูสเคสไดอะแกรม ข้างต้น ประกอบด้วย แอ็กเตอร์ซึ่งเป็นส่วนแสดงบุคคล ผู้เกี่ยวข้องในกระบวนการทดสอบซอฟต์แวร์ ซึ่งมีดังนี้

1. **Tester** หมายถึง ผู้ทดสอบระบบและบันทึกรายงานข้อผิดพลาด หรือประเด็นต่างๆ ที่เกิดจากการทดสอบระบบ รวมถึงเป็นผู้ดูแลระบบ
2. **Test Manager** หมายถึง ผู้จัดการส่วนทดสอบระบบตรวจสอบข้อผิดพลาดที่เกิดขึ้นจากการทดสอบและมอบหมายงานข้อผิดพลาดไปยังผู้ที่เกี่ยวข้องในทีม
3. **Developer** หมายถึง นักพัฒนาระบบ หรือ โปรแกรมเมอร์ แก้ไข โปรแกรมหรือนำเสนอการแก้ปัญหาลักษณะต่างๆ ที่เกี่ยวข้องกับโปรแกรม
4. **Project Manager** หมายถึง ผู้ควบคุมดูแลภาพรวมของโครงการต่างๆ ที่เกี่ยวข้องกับทดสอบ และติดตามข้อผิดพลาดหรือประเด็นต่างๆ ที่เกี่ยวข้องกับโครงการนั้น ๆ
5. **Business Analyst** หมายถึง นักวิเคราะห์ธุรกิจ ตรวจสอบผลการทดสอบว่าตรงกับ User Requirement Document Part : Business และติดตามข้อผิดพลาดหรือประเด็นต่าง ๆ
6. **System Analyst** หมายถึง นักวิเคราะห์ระบบ ตรวจสอบผลการทดสอบว่าตรงกับ User Requirement Document Part : System Analyst และติดตามข้อผิดพลาดหรือประเด็นต่าง ๆ

ส่วนฟังก์ชันการทำงานหลักได้แยกแสดงออกเป็นยูสเคส ซึ่งประกอบด้วย

1. Use case Create New Defect เป็นการสร้างหรือบันทึก Defect ที่พบจากการทดสอบระบบฯ
2. Use case Closed Defect เป็นการปิด Defect ที่เกิดขึ้นและเปลี่ยนสถานะเป็น Closed
3. Use case Update Defect ปรับปรุงแก้ไข Defect ที่พบครั้งแรก และมีการ Fix Defect จากทีมผู้เกี่ยวข้อง
4. Use case Assign Defect เป็น การมอบหมาย Defect ที่เกิดขึ้นให้ Developer ทำการแก้ไขหรือนำเสนอ Solution
5. Use case Search Defect การค้นหา Defect แบบ Advanced Search สามารถค้นหาได้ตาม Criteria ที่ต้องการ
6. Use case Resolve Solution Defect การตรวจสอบและตัดสินใจเลือก Solution เพื่อให้ Tester ทดสอบระบบคือการ Regression Test

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. Use Case View Defect /Issue Activity เพื่อดูรายละเอียดข้อผิดพลาดหรือประเด็นต่างๆ ที่เกิดจากการทดสอบ อาทิเช่น ข้อมูลอะไร , สถานะ,ใครเป็นผู้รับผิดชอบ เป็นต้น
8. Use case Generate Summary Report and Chart เป็นการสร้างรายงานสรุปผลการทดสอบประเภทต่างๆ
9. Manage User Profile บริหารจัดการข้อมูลผู้ใช้งาน จัดการกลุ่มผู้ใช้งานและสิทธิในการเข้าใช้งาน
10. Manage Project for Testing บริหารจัดการเพิ่มโปรเจกต์สำหรับการทดสอบ ว่ามีการทดสอบโปรเจกต์ไหนบ้าง เพื่อการอ้างอิงในการทดสอบในแต่ละทีม
11. Send Email Notification เป็นส่งเมลแจ้งเตือนทีมผู้เกี่ยวข้องในการสร้าง หรือปรับปรุงแก้ไขสถานะต่างๆของข้อผิดพลาดที่เกิดขึ้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 รายละเอียดยูสเคส Create New Defect

| | | |
|-----------------------|---|--|
| Use Case Name: | Create New Defect | |
| Scenario: | สร้าง/บันทึกผลการทดสอบจากการพบ Defect | |
| Brief Description: | ผู้ทดสอบระบบพบ Defect จากการทดสอบฯ ทำให้ต้องสร้างข้อมูลผลกระทบ (Defect) ที่เกิดจากการทดสอบ เพื่อแจ้งให้ผู้ที่เกี่ยวข้องทำการแก้ไข Defect ต่อ ไป | |
| Actors: | Tester (ผู้ทดสอบ) | |
| Related Use Cases: | Includes: Send Email Notification | |
| Preconditions: | สิทธิ์ที่สามารถสร้าง/บันทึกผลการทดสอบจากการพบ Defect หรือ Bug ได้ คือ สิทธิ์ของ Tester เท่านั้น Log in เข้าระบบโดยสิทธิ์ Tester | |
| Post conditions: | ระบบสร้างข้อมูล Defect และ Generate Defect number ได้สำเร็จ พร้อมส่งอีเมลล์ แจ้งเตือน ไปยังผู้ที่เกี่ยวข้อง | |
| Flow of Activities: | Actor | System |
| | <ol style="list-style-type: none"> พบ Defect ใหม่ที่เกิดขึ้นจากทดสอบ Tester Log in เข้าสู่ระบบตามสิทธิ์ที่กำหนด และคลิกปุ่ม Create Defect Tester กรอกข้อมูล Defect ทั้งหมดที่เกิดขึ้นจากการทดสอบฯและคลิกปุ่มบันทึก หลังจากนั้นคลิกปุ่ม Send เพื่อส่งงาน ไปยังทีมผู้เกี่ยวข้อง ได้แก่ Test Manager | <ol style="list-style-type: none"> 3. แสดงหน้าจอให้สร้างข้อมูล Defect ใหม่ 5. บันทึกข้อมูล Defect ลงฐานข้อมูลและระบบ Auto send mail และส่งงาน Defect ที่สร้างใหม่ไปยัง Personal Inbox ของผู้เกี่ยวข้อง ได้แก่ Test Manager |
| Exception Conditions: | <ol style="list-style-type: none"> ถ้ากรอกข้อมูล ไม่ครบทุกช่องตาม Required field <ol style="list-style-type: none"> ระบบแจ้งเตือนข้อผิดพลาด กรอกข้อมูลให้ครบทุกช่องตาม Required filed หลังจากคลิกปุ่ม Submit และSend mail ไปยัง Test Manager แต่ปรากฏว่าเกิดข้อผิดพลาดไม่สามารถส่งได้ <ol style="list-style-type: none"> ระบบจะมีข้อความแจ้งกลับมายังต้นทางว่า Defect ไม่สามารถส่งไปยัง Mail ที่ต้องการส่งได้ มีข้อผิดพลาดกรุณาตรวจสอบใหม่ | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 รายละเอียดยูสเคส Update Defect

| | | |
|-----------------------|--|---|
| Use Case Name: | Update Defect | |
| Scenario: | ปรับปรุง แก้ไขข้อผิดพลาดที่เกิดจากทดสอบระบบฯ | |
| Brief Description: | ผู้ที่เกี่ยวข้องตามสิทธิ์สามารถเข้ามาทำการปรับปรุงแก้ไข Defect ต่างๆ ในระบบได้ อาทิเช่น Tester และ Test Manager เข้ามาตรวจสอบสถานะต่าง ๆ ข้อมูลต่างๆ ที่อยู่ในระบบ ส่วน Developer สิทธิ์จะสามารถเห็นและปรับปรุงแก้ไขได้ เช่น แก้ไขโปรแกรมตาม Defect ที่เจอและเพิ่มข้อมูล Solution ต่างๆ ที่เกี่ยวกับ Defect นั้น รวมถึงเปลี่ยนสถานะได้ เป็นต้น | |
| Actors: | Tester ,Test Manager, Developer | |
| Related Use Cases: | Includes: Send Email Notification | |
| Preconditions: | สิทธิ์ที่สามารถปรับปรุงแก้ไข Defect ได้ คือสิทธิ์ของ Tester, Test Manager ,Developer เท่านั้น | |
| Post conditions: | แสดงข้อมูล Defect ที่มีการปรับปรุงแก้ไขได้สำเร็จและถูกต้อง พร้อมส่งอีเมลแจ้งเตือนไปยังผู้ที่เกี่ยวข้อง | |
| Flow of Activities: | Actor | System |
| | <ol style="list-style-type: none"> 1. ผู้มีสิทธิ์ Log in เข้าสู่ระบบตามสิทธิ์ที่กำหนด และคลิกปุ่มเพื่อค้นหา Defect ที่ต้องการปรับปรุงแก้ไข 3. ผู้มีสิทธิ์ทำการปรับแก้ไข หรือ Update ข้อมูลเพิ่มเติม และคลิกปุ่มยืนยันการ Update | <ol style="list-style-type: none"> 2.แสดง Alert ข้อความงานที่ได้รับมอบหมายหรือตอบกลับ (ทั้งนี้สิทธิ์แต่ละสิทธิ์จะเห็นเฉพาะงานของตัวเองเท่านั้นตาม Role กำหนด) 4. บันทึกข้อมูล Defect ที่ Update ลงฐานข้อมูลและระบบ Auto send mail แจ้งส่ง Defect ที่ Update แจ้งผู้เกี่ยวข้อง |
| Exception Conditions: | <p>หลังจากคลิกปุ่ม Submit และSend mail ไปยังผู้ที่เกี่ยวข้องแต่ปรากฏว่าเกิดข้อผิดพลาดไม่สามารถส่งได้</p> <p>a. ระบบจะมีข้อความแจ้งเตือนมายังต้นทางว่า Defect ไม่สามารถส่งไปยัง Mail ที่ต้องการส่งได้ มีข้อผิดพลาดกรุณาตรวจสอบใหม่</p> | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 รายละเอียดยูสเคส Assign Defect

| | | |
|-----------------------|---|---|
| Use Case Name: | Assign Defect | |
| Scenario: | Assign Defect การมอบหมาย Defect ที่เกิดขึ้นให้ Developer ทำการแก้ไขหรือนำเสนอ Solution | |
| Brief Description: | Test manager ได้รับ Email แจ้งเตือน Defect ใหม่ที่ถูกสร้างขึ้นโดย Tester ซึ่ง Test Manager ต้องเข้าสู่ระบบเพื่อทำการ Assign defect ไปยัง Developer เพื่อทำการแก้ไข หรือเสนอ Solution ทั้งนี้ Test manager จะต้องตรวจสอบก่อนว่า Defect ที่เกิดขึ้นรุนแรงและสมควรได้รับการแก้ไขหรือไม่ เงื่อนไขดังนี้ ถ้าสมควร (Y) Assign ไปยัง Developer แต่ถ้า ไม่สมควร(N) Reassign กลับไปยัง Tester ทำการทดสอบใหม่ | |
| Actors: | Test Manager | |
| Related Use Cases: | Includes: Send Email Notification (Auto mail แจ้งเตือนผู้ที่เกี่ยวข้อง) | |
| Preconditions: | สิทธิ์ที่สามารถ Assign Defect ได้ คือสิทธิ์ของ Test Manager เท่านั้น Log in เข้าระบบโดยสิทธิ์ Test Manager | |
| Post conditions: | สามารถ Assign Defect ได้สำเร็จและถูกต้องตาม Role ที่กำหนด พร้อมส่งอีเมลแจ้งเตือน ไปยัง Developer | |
| Flow of Activities: | Actor | System |
| | <p>1. Test Manager Log in เข้าสู่ระบบตามสิทธิ์ที่กำหนด และคลิกปุ่มเพื่อค้นหา Defect ที่ต้องการ Assign</p> <p>3. แก้ไข Status และ ระบุ Assign to ไปยัง Developer แก้ไข และคลิกปุ่มยืนยัน</p> | <p>2. แสดง Alert ข้อความงานที่ได้รับการมอบหมายหรือตอบกลับ (ทั้งนี้สิทธิ์แต่ละสิทธิ์จะเห็นเฉพาะงานของตัวเอง เท่านั้นตาม Role กำหนด)</p> <p>4. บันทึก Status และเก็บข้อมูล Assign to ลงฐานข้อมูลและระบบ Auto send mail แจ้งส่ง Defect ไปยัง Developer แก้ไข</p> |
| Exception Conditions: | <p>หลังจากคลิกปุ่ม Submit และ Send mail ไปยัง Developer แต่ปรากฏว่าเกิดข้อผิดพลาดไม่สามารถส่งได้</p> <p>a. ระบบจะมีข้อความแจ้งกลับมายังต้นทางว่า Defect ไม่สามารถส่งไปยัง Mail ที่ต้องการส่งได้ มีข้อผิดพลาดกรุณาตรวจสอบใหม่</p> | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.4 รายละเอียดยูสเคส Resolve Defect

| Use Case Name: | Resolve Defect | | | | | | |
|--|--|-------|--------|--|--|---|--|
| Scenario: | Resolve Defect การตรวจสอบและตัดสินใจเลือก Solution เพื่อให้ Tester ทดสอบระบบฯ | | | | | | |
| Brief Description: | Test manager ได้รับ Email แจ้งกลับจาก Developer ว่า Defect ได้รับการแก้ไขแล้ว ทั้งนี้ Test Manager ต้องตรวจสอบว่า Solution ที่ทาง Developer แก้ไขมาสามารถทำงานได้จริง โดยมีเงื่อนไขดังนี้ ถ้า สมควร (Y) เข้าสู่ระบบเพื่อ Update Status resolved ไปยัง Tester แต่ถ้า ไม่สมควร(N) Reopened ส่งกลับไปให้ Developer แก้ไขใหม่อีกครั้ง | | | | | | |
| Actors: | Test Manager | | | | | | |
| Related Use Cases: | Includes: Send Email Notification (Auto mail แจ้งเตือนผู้ที่เกี่ยวข้อง) | | | | | | |
| Preconditions: | สิทธิ์ที่สามารถ Resolve Defect ได้ คือสิทธิ์ของ Test Manager เท่านั้น Log in เข้าสู่ระบบโดยสิทธิ์ Test Manager | | | | | | |
| Post conditions: | สามารถ Update Status เปลี่ยนเป็น Resolved Defect ได้สำเร็จและถูกต้องตาม Role ที่กำหนด พร้อมส่งอีเมลแจ้งเตือนไปยัง Tester | | | | | | |
| Flow of Activities: | <table border="1"> <thead> <tr> <th>Actor</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>1. Test Manager Log in เข้าสู่ระบบตามสิทธิ์ที่กำหนด และคลิกปุ่มเพื่อค้นหา Defect ที่ต้องการ ตรวจสอบ Solution จาก Developer</td> <td>2. แสดง Alert ข้อความงานที่ได้รับ การมอบหมายหรือตอบกลับ (ทั้งนี้ สิทธิ์แต่ละสิทธิ์จะเห็นเฉพาะงานของตัวเองเท่านั้นตาม Role กำหนด)</td> </tr> <tr> <td>3. แก้ไข Status เป็น Resolved และคลิกปุ่มยืนยัน</td> <td>4. บันทึก Status เป็น Resolved และระบบ Auto send mail แจ้งส่ง Defect ที่แก้ไขและตรวจสอบแล้ว ไปยัง Tester</td> </tr> </tbody> </table> | Actor | System | 1. Test Manager Log in เข้าสู่ระบบตามสิทธิ์ที่กำหนด และคลิกปุ่มเพื่อค้นหา Defect ที่ต้องการ ตรวจสอบ Solution จาก Developer | 2. แสดง Alert ข้อความงานที่ได้รับ การมอบหมายหรือตอบกลับ (ทั้งนี้ สิทธิ์แต่ละสิทธิ์จะเห็นเฉพาะงานของตัวเองเท่านั้นตาม Role กำหนด) | 3. แก้ไข Status เป็น Resolved และคลิกปุ่มยืนยัน | 4. บันทึก Status เป็น Resolved และระบบ Auto send mail แจ้งส่ง Defect ที่แก้ไขและตรวจสอบแล้ว ไปยัง Tester |
| Actor | System | | | | | | |
| 1. Test Manager Log in เข้าสู่ระบบตามสิทธิ์ที่กำหนด และคลิกปุ่มเพื่อค้นหา Defect ที่ต้องการ ตรวจสอบ Solution จาก Developer | 2. แสดง Alert ข้อความงานที่ได้รับ การมอบหมายหรือตอบกลับ (ทั้งนี้ สิทธิ์แต่ละสิทธิ์จะเห็นเฉพาะงานของตัวเองเท่านั้นตาม Role กำหนด) | | | | | | |
| 3. แก้ไข Status เป็น Resolved และคลิกปุ่มยืนยัน | 4. บันทึก Status เป็น Resolved และระบบ Auto send mail แจ้งส่ง Defect ที่แก้ไขและตรวจสอบแล้ว ไปยัง Tester | | | | | | |
| Exception Conditions: | <p>หลังจากคลิกปุ่ม Submit และ Send mail ไปยัง Tester แต่ปรากฏว่าเกิดข้อผิดพลาดไม่สามารถส่งได้</p> <p>a. ระบบจะมีข้อความแจ้งกลับมายังต้นทางว่า Defect ไม่สามารถส่งไปยัง Mail ที่ต้องการส่งได้ มีข้อผิดพลาดกรุณาตรวจสอบใหม่</p> | | | | | | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 รายละเอียดยูสเคส Close Defect

| | | |
|-----------------------|---|---|
| Use Case Name: | Close Defect | |
| Scenario: | Close Defect การปิด Defect ที่เกิดขึ้นและเปลี่ยนสถานะเป็น Closed | |
| Brief Description: | Tester ได้รับ Email แจ้งกลับจาก Test Manager ว่า Defect ได้รับการแก้ไขและตรวจสอบ สามารถทำการทดสอบได้แล้ว ซึ่ง Tester ต้องนำ Solution ที่ได้หรือโปรแกรมที่แก้ไขไปทำการ Retest อีกครั้ง ถ้า Retest สมบูรณ์ Tester จึง Log in เข้าสู่ระบบเพื่อปิด Defect และ Update status เป็น Closed | |
| Actors: | Tester | |
| Related Use Cases: | Includes: Send Email Notification (Auto mail แจ้งเตือนผู้ที่เกี่ยวข้องในทีม) | |
| Preconditions: | สิทธิ์ที่สามารถปิด Defect หรือ Bug ได้ คือสิทธิ์ของ Tester เท่านั้น Log in เข้าสู่ระบบ โดยสิทธิ์ Tester | |
| Post conditions: | สามารถ Update Status เปลี่ยนเป็น Closed Defect ได้สำเร็จและถูกต้องตาม Role ที่กำหนด พร้อมส่งอีเมลแจ้งเตือนไปยัง Test Manager ,Developer ,Project Manager,BA,SA | |
| Flow of Activities: | Actor | System |
| | <p>1. Tester Log in เข้าสู่ระบบตามสิทธิ์ที่กำหนด และคลิกปุ่มเพื่อค้นหา Defect ที่ต้องการ Close</p> <p>3. แก้ไข Status เป็น Closed พร้อมกรอกรายละเอียดที่ปิด Defect และคลิกปุ่มยืนยัน</p> | <p>2. แสดง Alert ข้อความงานที่ได้รับการมอบหมายหรือตอบกลับ (ทั้งนี้สิทธิ์แต่ละสิทธิ์จะเห็นเฉพาะงานของตัวเองเท่านั้นตาม Role กำหนด)</p> <p>4. บันทึก Status เป็น Closed และระบบ Auto send mail แจ้งส่ง Defect ที่ Tester ได้ทำการทดสอบผ่านแล้ว ไปยังผู้ที่เกี่ยวข้องทั้งหมด</p> |
| Exception Conditions: | <p>หลังจากคลิกปุ่ม Submit และ Send mail ไปยังผู้ที่เกี่ยวข้อง แต่ปรากฏว่าเกิดข้อผิดพลาดไม่สามารถส่งได้</p> <p>a. ระบบจะมีข้อความแจ้งกลับมายังต้นทางว่า Defect ไม่สามารถส่งไปยัง Mail ที่ต้องการส่งได้ มีข้อผิดพลาดกรุณาตรวจสอบใหม่</p> | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.6 รายละเอียดยูสเคส Search Defect / Issue

| | | |
|-----------------------|---|--|
| Use Case Name: | Search Defect /Issue | |
| Scenario: | ค้นหารายละเอียดข้อผิดพลาด หรือ ประเด็นต่าง ๆ ที่เกิดจากการทดสอบระบบฯ | |
| Brief Description: | ทีมผู้เกี่ยวข้องทั้งหมดสามารถทำการค้นหา Defect หรือ Issue เพื่อค้นหา รายละเอียดข้อผิดพลาด หรือประเด็นต่างๆ ที่เกิดขึ้นจากการทดสอบระบบฯ เพื่อเข้าไปแก้ไขหรือเพิ่มรายละเอียดต่าง ๆ ที่เกี่ยวข้อง | |
| Actors: | Tester, Test Manager , Project Manager, Business Analyst (BA), System Analyst(SA), Developer | |
| Related Use Cases: | - | |
| Preconditions: | เมื่อต้องการเข้าไปดูรายละเอียด หรือ ปรับปรุงแก้ไขสถานะต่าง ๆ ของ ข้อผิดพลาดหรือประเด็นต่างๆ โดยสิทธิ์ที่สามารถค้นหาข้อมูลได้แก่ Tester, Test Manager , Project Manager, Business Analyst (BA), System Analyst(SA), Developer โดย Log in เข้าระบบ โดยสิทธิ์แต่ละคนเท่านั้น | |
| Post conditions: | แสดงรายละเอียดข้อมูลที่ค้นหา โดยบางสิทธิ์ที่ค้นหาได้จะไม่สามารถทำการแก้ไขได้ แต่จะสามารถ View ดูได้อย่างเดียวขึ้นอยู่กับข้อกำหนดสิทธิ์การใช้งาน | |
| Flow of Activities: | Actor | System |
| | 1. ผู้มีสิทธิ์ทำการระบุเงื่อนไขต่าง ๆ ที่ต้องการค้นหา Defect หรือ Issue และคลิกปุ่มค้นหา | 2. ระบบทำการค้นหา Defect หรือ Issue ตามเงื่อนไขที่ระบุไว้ 4. ระบบแสดงรายละเอียดที่ค้นหา |
| Exception Conditions: | หลังจากคลิกปุ่ม ค้นหา แต่ไม่มีข้อมูลในระบบจะแสดงข้อความดังนี้ A. ระบบไม่สามารถทำการค้นหาได้ โดยระบบจะแจ้งเตือนข้อความ | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 รายละเอียดยูสเคส Generate Summary Report and Chart

| | | |
|-----------------------|---|--|
| Use Case Name: | Generate Summary Report and Chart | |
| Scenario: | สร้างรายงานสรุปประเภทต่างๆ | |
| Brief Description: | <p>1. Test Manager สามารถเลือกเรียกดูรายงานสรุปพร้อมแสดง Chart ตามเงื่อนไขที่เลือก ได้ทุกประเภทที่มีในระบบ</p> <p>2. Project Manager สามารถเลือกเรียกดูรายงานสรุปพร้อมแสดง Chart ตามเงื่อนไขที่เลือก ได้บางประเภทที่เกี่ยวข้องสำหรับการบริหาร Project ที่เกี่ยวข้องเท่านั้น</p> <p>3. Business Analyst สามารถเลือกเรียกดูรายงานสรุปพร้อมแสดง Chart ตามเงื่อนไขที่เลือก ได้บางประเภทที่เกี่ยวข้องเท่านั้น</p> <p>4. System Analyst สามารถเลือกเรียกดูรายงานสรุปพร้อมแสดง Chart ตามเงื่อนไขที่เลือก ได้บางประเภทที่เกี่ยวข้องเท่านั้น</p> | |
| Actors: | Test Manager , Project Manager, Business Analyst (BA),System Analyst(SA) | |
| Related Use Cases: | - | |
| Preconditions: | สิทธิ์ที่สามารถเรียกดูรายงานสรุปได้แก่ Test Manager , Project Manager, Business Analyst (BA) และ System Analyst(SA) โดย Log in เข้าระบบโดยสิทธิ์แต่ละคนเท่านั้น | |
| Post conditions: | แสดงรายงานสรุปและChart แต่ละประเภทที่เลือก | |
| Flow of Activities: | Actor | System |
| | 1. ผู้มีสิทธิ์เรียกดูรายงาน Log in เข้าสู่ระบบตามสิทธิ์ที่กำหนด | 2.แสดงรายงานประเภทต่างๆ และ Criteria ต่างๆ |
| | 3. เลือกประเภทรายงานตามเงื่อนไขที่จะเรียกดู และคลิกปุ่ม Submit | 4. แสดงผลรายงานตามเงื่อนไขที่เลือก |
| | Generate report | |
| Exception Conditions: | หลังจากคลิกปุ่ม Submit Generate report แต่ประเภทรายงานที่เลือกไม่มีข้อมูลในระบบจะแสดงข้อความดังนี้ | |
| | A. ระบบไม่สามารถ Generate report ได้เนื่องจากไม่มีข้อมูลในระบบ | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.8 รายละเอียดยูสเคส View Defect/Issue Activity

| | | |
|-----------------------|--|---|
| Use Case Name: | View Defect/Issue Activity | |
| Scenario: | เรียกดูรายละเอียด สถานะ กิจกรรมต่างๆ ของข้อผิดพลาดหรือประเด็นต่าง ๆ | |
| Brief Description: | ทีมผู้เกี่ยวข้องสามารถเรียกดูรายละเอียด สถานะต่างๆ หรือกิจกรรมที่เกิดขึ้นกับข้อผิดพลาดหรือประเด็นต่าง ๆ ที่ถูกสร้างขึ้นในระบบฯ | |
| Actors: | Tester , Test Manager , Project Manager, Business Analyst (BA), System Analyst(SA) | |
| Related Use Cases: | Search Defect / Issue | |
| Preconditions: | ต้องทำการค้นหาข้อมูล Defect หรือ Issue ก่อนเข้าไปดูรายละเอียดกิจกรรมต่างๆ หลังจากนั้นจะสามารถเข้าดูรายละเอียดได้ โดยสิทธิ์ที่สามารถค้นหาข้อมูลได้แก่ Tester, Test Manager , Project Manager, Business Analyst (BA), System Analyst(SA) โดย Log in เข้าสู่ระบบ โดยสิทธิ์แต่ละคนเท่านั้น | |
| Post conditions: | แสดงรายละเอียดกิจกรรมต่างๆ ที่เกิดขึ้นของ Defect /Issue ว่าสถานะเป็นอย่างไร | |
| Flow of Activities: | Actor | System |
| | <ol style="list-style-type: none"> ผู้มีสิทธิ์ทำการระบุเงื่อนไขต่างๆ ที่ต้องการค้นหา Defect หรือ Issue และคลิกปุ่มค้นหา หลังจากและเลือกเลขที่ Defect หรือ Issue เพื่อดูรายละเอียด | <ol style="list-style-type: none"> ระบบทำการค้นหา Defect หรือ Issue ตามเงื่อนไขที่ระบุไว้ ระบบแสดงรายละเอียดที่ค้นหาในเบื้องต้น ระบบแสดงรายละเอียดทั้งหมดของข้อผิดพลาดว่ามีการปรับปรุงแก้ไขอย่างไรหรือสถานะเป็นอย่างไร เป็นต้น |
| Exception Conditions: | หลังจากคลิกปุ่ม ค้นหา แต่ไม่มีข้อมูลในระบบจะแสดงข้อความดังนี้ A. ระบบไม่สามารถทำการค้นหาได้ โดยระบบจะแจ้งเตือนข้อความ | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.9 รายละเอียดยูสเคส Manage User Profile

| | | |
|-----------------------|--|---|
| Use Case Name: | Manage User Profile | |
| Scenario: | บริหารจัดการข้อมูลผู้ใช้งาน จัดการกลุ่มผู้ใช้งานและสิทธิในการเข้าใช้งาน | |
| Brief Description: | เพื่อสามารถจัดการข้อมูลผู้ใช้งาน กลุ่มผู้ใช้งานและสิทธิในการเข้าใช้งาน โดยการเพิ่ม ลบ หรือแก้ไขข้อมูลแต่ทีมในระบบ | |
| Actors: | Test Manager | |
| Related Use Cases: | - | |
| Preconditions: | ต้อง Login เข้าระบบเพื่อสร้างผู้ใช้งานใหม่ หรือเปลี่ยนแปลงสิทธิ์การใช้งาน | |
| Post conditions: | สร้าง User Profile สำเร็จพร้อมสิทธิ์ที่สามารถใช้งานได้ | |
| Flow of Activities: | Actor | System |
| | <p>1. Test Manager จัดการ (เพิ่ม,แก้ไข ,ลบ) User Profile ของผู้ใช้งานที่เพิ่มใหม่หรือเปลี่ยนแปลง</p> <p>2. กรอกรายละเอียดข้อมูลผู้ใช้งาน พร้อมกำหนดสิทธิหน้าที่การใช้งาน</p> | <p>3.ระบบบันทึกผู้ใช้งานและสร้างรหัสผ่านให้อัตโนมัติ พร้อมกำหนดสิทธิผู้ใช้งาน</p> |
| Exception Conditions: | <p>หลังจากคลิกปุ่ม Submit ถ้ากรอกรายละเอียดของข้อมูลของผู้ใช้งานไม่ครบระบบแสดงข้อความดังนี้</p> <p>A. ระบบแสดงข้อความเตือนและให้กรอกข้อมูลให้ครบ</p> | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.10 รายละเอียดยูสเคส Manage Project for Testing

| | | |
|--------------------------|--|---|
| Use Case Name: | Manage Project for Testing | |
| Scenario: | บริหารจัดการเพิ่มโปรเจกต์สำหรับการทดสอบ | |
| Brief Description: | เพื่อสามารถจัดการบริหารจัดการเพิ่ม แก้ไข ลบ โปรเจกต์สำหรับการทดสอบ ว่ามีทดสอบโปรเจกต์ไหนบ้างที่กำลังดำเนินพัฒนา เพื่อการอ้างอิงในการ ทดสอบในแต่ละทีม | |
| Actors: | Test Manager | |
| Related Use Cases: | - | |
| Preconditions: | ต้อง Login เข้าสู่ระบบเพื่อสร้างโปรเจกต์ใหม่ หรือเปลี่ยนแปลงแก้ไขโปรเจกต์ นั้น | |
| Post conditions: | สร้างโปรเจกต์ หรือแก้ไขโปรเจกต์ สำเร็จเพื่อให้ใช้อย่างในระบบ | |
| Flow of Activities: | Actor | System |
| | 1. Test Manager จัดการ (เพิ่ม, แก้ไข,ลบ) โปรเจกต์ที่เพิ่มใหม่หรือ เปลี่ยนแปลงแก้ไข 2. กรอกรายละเอียดข้อมูลของโปร เจกต์นั้นๆ | 3.ระบบบันทึกข้อมูลโปรเจกต์สำเร็จ เรียบร้อย |
| Exception Conditions: | หลังจากคลิกปุ่ม Submit ถ้ากรอกรายละเอียดของข้อมูลโปรเจกต์ไม่ครบ ระบบแสดงข้อความดังนี้ A. ระบบแสดงข้อความเตือนและให้กรอกข้อมูลให้ครบ | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.11 รายละเอียดยูสเคส Send Email Notification

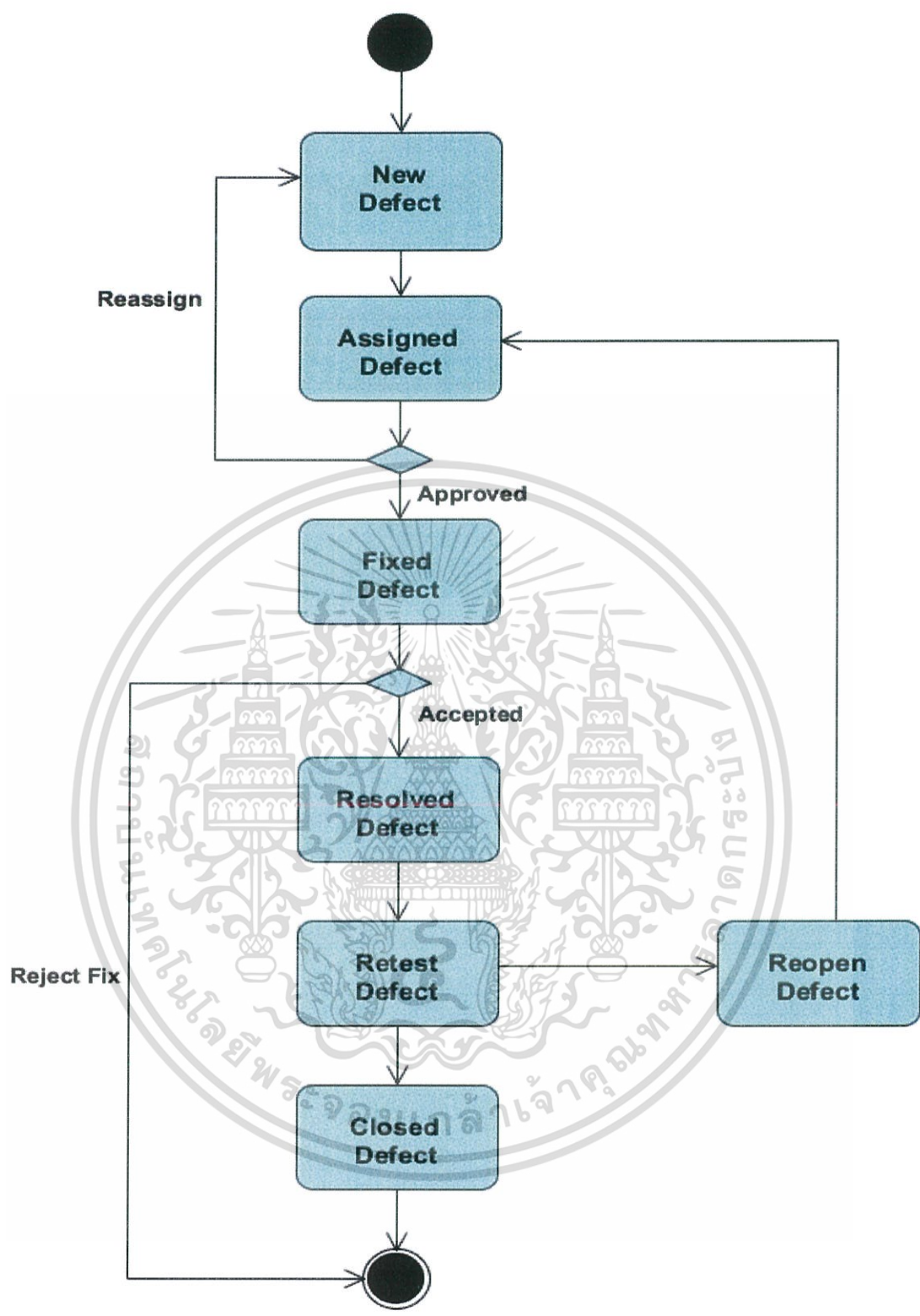
| | | |
|-----------------------|--|--|
| Use Case Name: | Send Email Notification | |
| Scenario: | การส่งเมลแจ้งเตือนทีมผู้เกี่ยวข้อง | |
| Brief Description: | เป็นการส่งเมลแจ้งเตือนทีมผู้เกี่ยวข้องในการสร้าง หรือ ปรับปรุงแก้ไขสถานะต่าง ๆ ของข้อผิดพลาดที่เกิดขึ้น | |
| Actors: | - | |
| Related Use Cases: | Create Defect,Closed Defect ,Update Defect ,Assign Defect,Resolve Defect | |
| Preconditions: | ต้องมีการสร้าง,ปรับปรุงแก้ไข,มอบหมายงาน,การตรวจสอบและตัดสินใจเลือก Solution เพื่อจบงาน | |
| Post conditions: | ส่งเมลแจ้งเตือนผู้เกี่ยวข้องสำเร็จ | |
| Flow of Activities: | Actor | System |
| | 1. ต้องมีการสร้าง,ปรับปรุงแก้ไข, มอบหมายงาน,การตรวจสอบและตัดสินใจเลือก Solution เพื่อจบงาน และเมื่อคลิกปุ่ม Submit | 3.ระบบทำการส่งเมลแจ้งเตือนทีมผู้เกี่ยวข้อง |
| Exception Conditions: | หลังจากคลิกปุ่ม Submit ถ้าส่งเมลไม่สำเร็จ ระบบแสดงข้อความดังนี้ A. ระบบแสดงข้อความเตือนว่าไม่สามารถส่งเมลได้ | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.3 ภาพแสดง สเตตไดอะแกรม (State diagram)

ไดอะแกรมแสดงการอธิบายการเปลี่ยนแปลงสถานะต่าง ๆ ของข้อผิดพลาดที่เกิดจากการทดสอบระบบ ซึ่งสามารถอธิบายได้ดังนี้

1. สถานะ New ของข้อผิดพลาด เป็นสถานะหลังจากที่ Tester พบข้อผิดพลาดที่เกิดจากการทดสอบระบบ
2. สถานะ Assigned ของข้อผิดพลาด จะเปลี่ยนเป็นสถานะนี้ก็ต่อเมื่อ Test Manager มีการ Approved ข้อผิดพลาดและยืนยันว่าเป็นข้อผิดพลาดที่เกิดจากการทดสอบจริง
3. สถานะ Reassign ของข้อผิดพลาด จะเปลี่ยนเป็นสถานะนี้ก็ต่อเมื่อ Test Manager พิจารณาจากข้อผิดพลาดแล้ว แต่ปรากฏว่าข้อผิดพลาดที่พบไม่ใช่ข้อผิดพลาดจากการทดสอบ หรือ ข้อมูลข้อผิดพลาดอาจไม่ครบถ้วนสมบูรณ์เป็นต้น
4. สถานะ Fixed ของข้อผิดพลาด เป็นสถานะที่ Developer มีการแก้ไขข้อผิดพลาดเรียบร้อยแล้วจึงเปลี่ยนเป็นสถานะดังกล่าว
5. สถานะ Reject Fix ของข้อผิดพลาด จะเปลี่ยนเป็นสถานะนี้ก็ต่อเมื่อ Developer พิจารณาข้อผิดพลาดที่เกิดจากการทดสอบระบบ ไม่ได้เกี่ยวข้องกับโปรแกรม หรือ ข้อมูลข้อผิดพลาดอาจไม่ครบถ้วนสมบูรณ์เป็นต้น
6. สถานะ Resolved ของข้อผิดพลาด จะเปลี่ยนเป็นสถานะนี้ก็ต่อเมื่อ Test Manager มีการตรวจสอบและตัดสินใจแล้วว่าข้อผิดพลาดที่ทาง Developer แก้ไขถูกต้อง และเหมาะสม สามารถทำการทดสอบระบบต่อไปได้
7. สถานะ Retest ของข้อผิดพลาด จะเปลี่ยนเป็นสถานะนี้ก็ต่อเมื่อ Tester เริ่มดำเนินการ Retest ข้อผิดพลาดอีกครั้ง
8. สถานะ Reopen ของข้อผิดพลาด จะเปลี่ยนแปลงเป็นสถานะนี้ก็ต่อเมื่อ Tester ทำการ Retest ข้อผิดพลาดแต่ปรากฏว่า ยังไม่ผ่าน ยังพบข้อผิดพลาดจากจุดเดิมอยู่
9. สถานะ Closed ของข้อผิดพลาด จะเปลี่ยนแปลงเป็นสถานะนี้ก็ต่อเมื่อ Tester ทดสอบข้อผิดพลาดที่พบผ่านและถูกต้อง



รูปที่ 3.19 แสดงสเตตัสไดอะแกรมของสถานะข้อผิดพลาด (Status Defect)

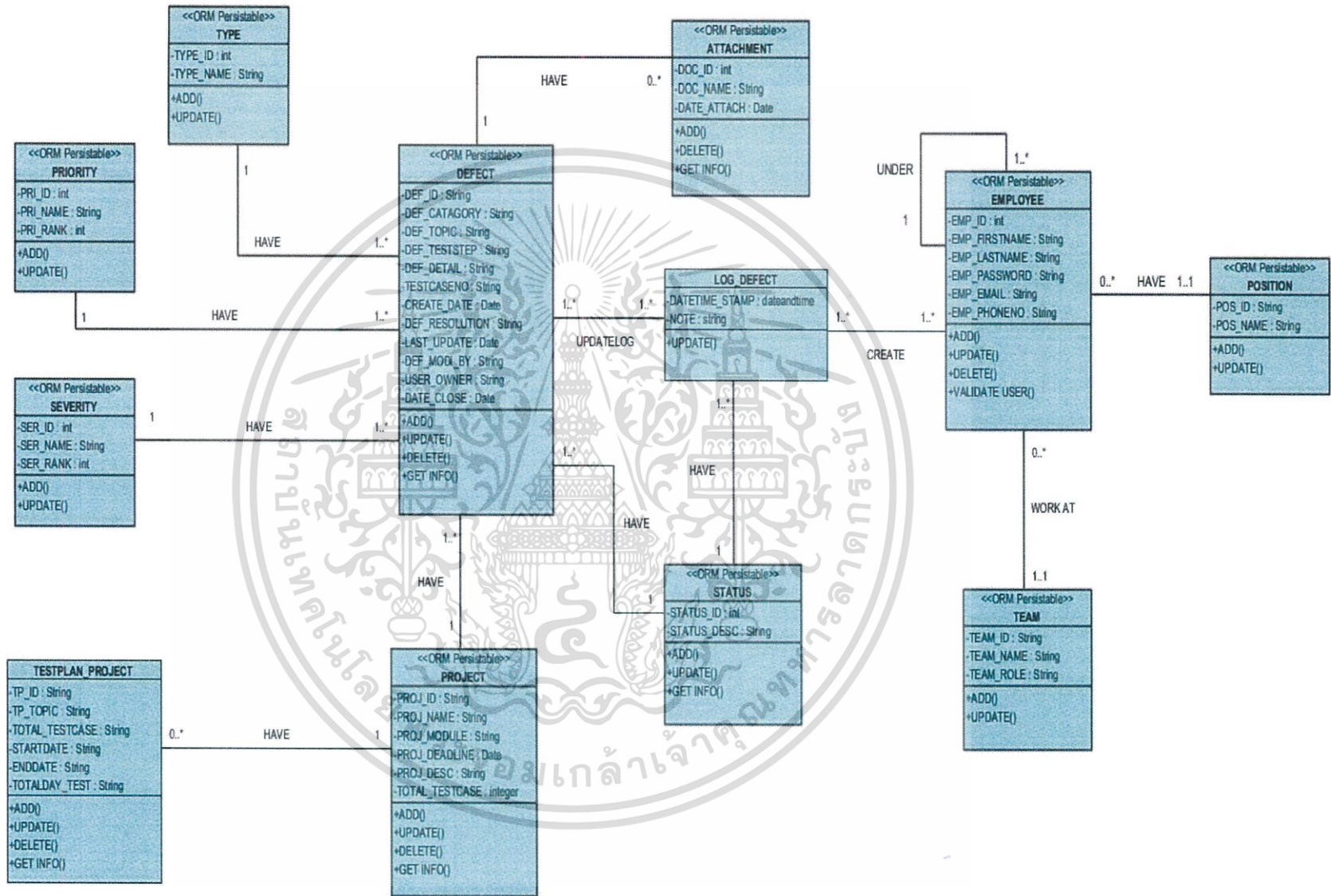
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.4 ภาพแสดง คลาสไดอะแกรม (Class diagram)

ไดอะแกรมที่ใช้ในการแสดงกลุ่มของคลาส โครงสร้างของคลาสอินเตอร์เฟซ (Interface) และ แสดงความสัมพันธ์ (Relationship) ระหว่างคลาส ซึ่งการพัฒนาระบบนี้สามารถแสดงกลุ่มของ คลาสต่าง ๆ แสดงได้ดังนี้

- คลาส DEFECT เป็นคลาสของข้อผิดพลาดหรือประเด็นแก้ไขจากการ ทดสอบระบบ
- คลาส PRIORITY เป็นคลาสของระดับความสำคัญของข้อผิดพลาด
- คลาส SEVERITY เป็นคลาสของระดับความรุนแรงของข้อผิดพลาด
- คลาส TYPE เป็นคลาสของประเภทของข้อผิดพลาดที่พบว่าเกิดจากส่วนใด ในการทำงานระบบ
- คลาส STATUS เป็นคลาสของข้อมูลสถานะข้อผิดพลาดและการเปลี่ยนแปลง ของสถานะข้อผิดพลาดต่าง ๆ
- คลาส PROJECT เป็นคลาสของข้อมูลโครงการระบบสารสนเทศที่ต้อง ดำเนินการทดสอบ
- คลาส ATTACHMENT เป็นคลาสของข้อมูลเอกสารแนบของข้อผิดพลาด หรือประเด็นแก้ไขจากการทดสอบระบบ
- คลาส EMPLOYEE เป็นคลาสของพนักงานผู้ใช้งานที่มีสิทธิในการใช้งาน ของระบบ
- คลาส POSITION เป็นคลาสของตำแหน่งของพนักงาน
- คลาส TEAM เป็นคลาสของทีมผู้ใช้งานที่ EMPLOYEE สังกัดในทีมนั้นๆ และกำหนดสิทธิในการเข้าใช้งานในระบบ
- คลาส LOG DEFECT เป็นคลาสเพื่อเก็บข้อมูลข้อผิดพลาดหรือประเด็นแก้ไข ที่พนักงานผู้ใช้งานระบบมีการเปลี่ยนแปลงต่าง ๆ ของระบบงาน
- คลาส TESTPLAN_PROJECT เป็นคลาสของแผนการทดสอบของโครงการที่ ทำการทดสอบในระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 แสดงคลาสไดอะแกรมของระบบติดตามเพื่อติดตาม และรายงานข้อผิดพลาดที่เกิดจากการทดสอบระบบธนาคารฯ

จากคลาสต่างๆ สามารถอธิบายความสัมพันธ์ของคลาสได้ดังนี้

- ความสัมพันธ์ระหว่างคลาส POSITION กับ EMPLOYEE ชื่อ HAVE แสดงถึงความสัมพันธ์ ของพนักงานผู้ใช้งานระบบ 1 คน ต้องมีตำแหน่งงานได้เพียง 1 ตำแหน่ง แต่ในตำแหน่งงานเดียวกันอาจมีพนักงานผู้ใช้งานระบบได้หลายคน
- ความสัมพันธ์ระหว่างคลาส TEAM กับ EMPLOYEE ชื่อ WORK AT แสดงถึงความสัมพันธ์ ของพนักงานผู้ใช้งานระบบ 1 คน สังกัดทีมได้ 1 ทีม และในแต่ละทีม อาจมีพนักงานสังกัดอยู่ได้มากกว่า 1 คน และในทีมแต่ละทีมต้องมีพนักงานสังกัดอยู่อย่างน้อย 1 คน
- ความสัมพันธ์ระหว่างคลาส DEFECT กับ LOG_DEFECT ชื่อ UPDATELOG แสดงถึงความสัมพันธ์ของข้อผิดพลาดที่เกิดขึ้นจากการทดสอบระบบ 1 ข้อผิดพลาด จะสามารถเก็บข้อมูลข้อผิดพลาดหรือประเด็นแก้ไขที่มีการ Update เปลี่ยนแปลงต่าง ๆ ของระบบงานได้หลายครั้ง
- ความสัมพันธ์ระหว่างคลาส EMPLOYEE กับ LOG_DEFECT ชื่อ CREATE แสดงถึงความสัมพันธ์ของพนักงาน 1 คน สามารถทำการสร้าง บันทึกหรือปรับปรุงแก้ไขข้อผิดพลาดได้หลาย ๆ ครั้ง
- ความสัมพันธ์ระหว่างคลาส STATUS กับ LOG_DEFECT ชื่อ HAVE แสดงถึงความสัมพันธ์ ของสถานะข้อผิดพลาด 1 สถานะ ปรากฏอยู่ในการเก็บข้อมูลข้อผิดพลาดหรือประเด็นแก้ไขที่พนักงานผู้ใช้งานระบบมีการเปลี่ยนแปลงต่าง ๆ ของระบบงานได้หลายเรคอร์ด
- ความสัมพันธ์ระหว่างคลาส STATUS กับ DEFECT ชื่อ HAVE แสดงถึงความสัมพันธ์ โดยสถานะของข้อผิดพลาด 1 สถานะสามารถปรากฏอยู่ในข้อผิดพลาดที่พบจากการทดสอบระบบได้หลายข้อผิดพลาด แต่ 1 ข้อผิดพลาดสามารถมีได้เพียง 1 สถานะเท่านั้น
- ความสัมพันธ์ระหว่างคลาส DEFECT กับ ATTACHMENT ชื่อ HAVE แสดงถึงความสัมพันธ์ โดย 1 ข้อผิดพลาดที่เกิดจากการทดสอบระบบจะมีเอกสารแนบได้หลาย ๆ ฉบับ และเอกสารแนบ 1 ฉบับ จะมีการอ้างอิงข้อผิดพลาดเพียง 1 ข้อผิดพลาด
- ความสัมพันธ์ระหว่างคลาส TYPE กับ DEFECT ชื่อ HAVE แสดงถึงความสัมพันธ์ โดยประเภทของข้อผิดพลาด 1 ประเภทปรากฏอยู่ในข้อผิดพลาดที่พบจากการทดสอบระบบได้หลายข้อผิดพลาด แต่ 1 ข้อผิดพลาดสามารถระบุประเภทได้ 1 ประเภทของข้อผิดพลาดเท่านั้น
- ความสัมพันธ์ระหว่างคลาส PRIORITY กับ DEFECT ชื่อ HAVE แสดงถึงความสัมพันธ์ โดยความสำคัญของข้อผิดพลาด 1 ความสำคัญปรากฏอยู่ในข้อผิดพลาดที่พบจากการทดสอบระบบได้หลายข้อผิดพลาด แต่ 1 ข้อผิดพลาดสามารถระบุความสำคัญได้ 1 ความสำคัญของข้อผิดพลาดเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ความสัมพันธ์ระหว่างคลาส SEVERITY กับ DEFECT ชื่อ HAVE แสดงถึงความสัมพันธ์ โดยความรุนแรงของข้อผิดพลาด 1 ความรุนแรงปรากฏอยู่ในข้อผิดพลาดที่พบจากการทดสอบระบบได้หลายข้อผิดพลาด แต่ 1 ข้อผิดพลาดสามารถระบุความรุนแรงได้ 1 ความรุนแรงของข้อผิดพลาดเท่านั้น

- ความสัมพันธ์ระหว่างคลาส PROJECT กับ DEFECT ชื่อ HAVE แสดงถึงความสัมพันธ์ โดย PROJECT 1 PROJECT สามารถมีข้อผิดพลาดที่เกิดจากการทดสอบระบบได้หลายข้อผิดพลาด แต่ใน 1 ข้อผิดพลาดสามารถอยู่ภายใต้ PROJECT นั้น ๆ ได้เพียง 1 PROJECT เท่านั้น

- ความสัมพันธ์ระหว่างคลาส PROJECT กับ TESTPLAN_PROJECT ชื่อ HAVE แสดงถึงความสัมพันธ์ โดย PROJECT 1 PROJECT สามารถมีแผนการทดสอบได้หลายแผนการทดสอบ แต่ใน 1 แผนการทดสอบสามารถอยู่ภายใต้ PROJECT นั้น ๆ ได้เพียง 1 PROJECT เท่านั้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การออกแบบฐานข้อมูลและออกแบบหน้าจอส่วนต่อประสานกับผู้ใช้

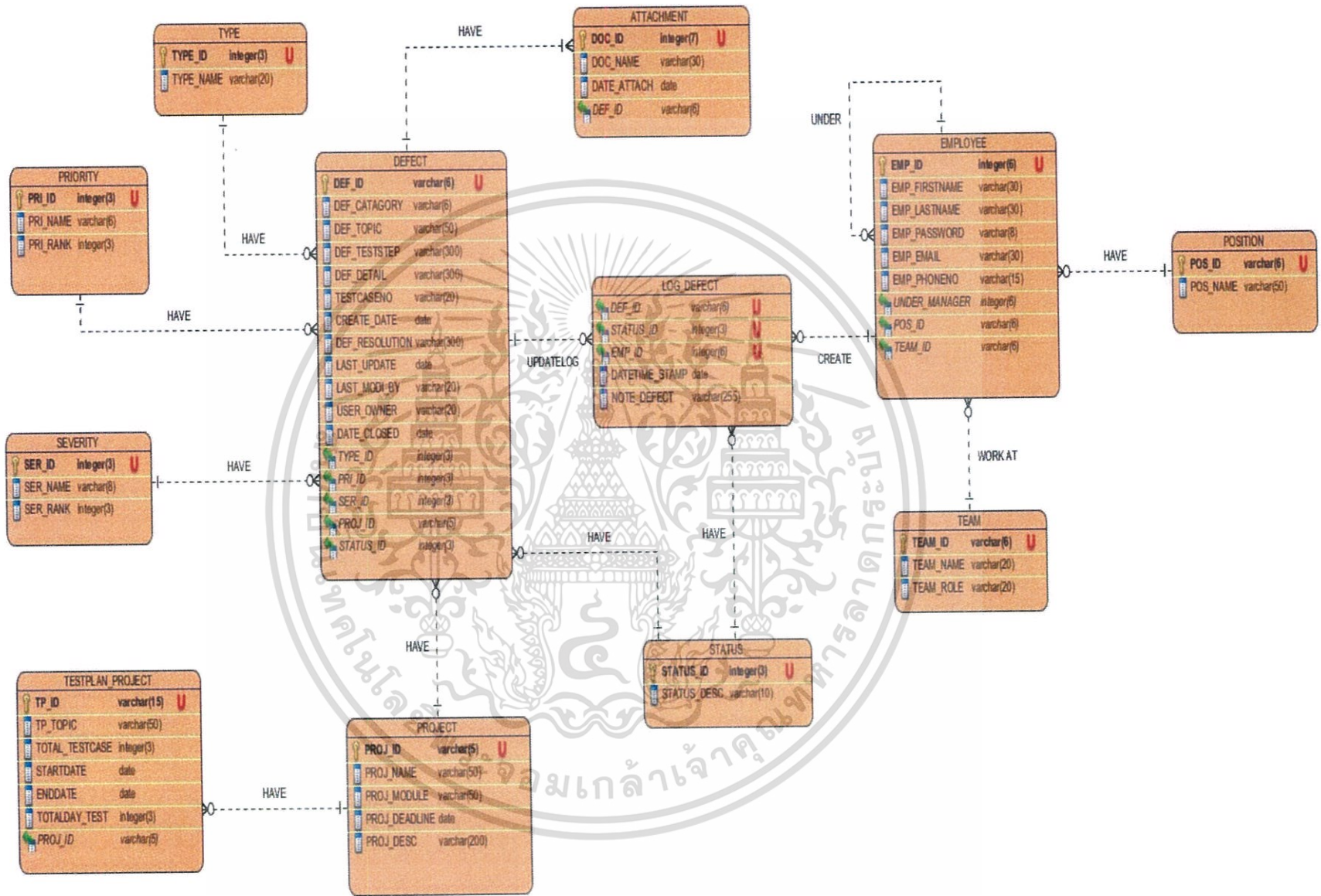
การออกแบบฐานข้อมูลสำหรับระบบติดตามเพื่อติดตาม และรายงานข้อผิดพลาดที่เกิดจากการทดสอบระบบธนาคารฯ ซึ่งได้ออกแบบฐานข้อมูลที่เป็นฐานข้อมูลเชิงสัมพันธ์และนำเสนอผ่านแบบจำลองความสัมพันธ์ระหว่างเอนทิตี เพื่อนำเสนอรายละเอียดทางด้าน โครงสร้างของฐานข้อมูล และแสดงให้เห็นถึงความสัมพันธ์ระหว่างข้อมูล โดยมีรายละเอียดดังต่อไปนี้

4.1 แบบจำลองความสัมพันธ์ระหว่างเอนทิตี

ตารางที่ 4.1 ตารางแสดงรายละเอียดของแต่ละเอนทิตีของระบบดังนี้

| ลำดับ | เอนทิตี | คำอธิบายเอนทิตี |
|-------|------------------|--|
| 1 | DEFECT | เอนทิตีข้อผิดพลาดหรือประเด็นแก้ไขจากการทดสอบระบบ |
| 2 | PRIORITY | เอนทิตีเก็บข้อมูลระบุความสำคัญของข้อผิดพลาด |
| 3 | SEVERITY | เอนทิตีเก็บข้อมูลระบุความรุนแรงของข้อผิดพลาด |
| 4 | TYPE | เอนทิตีเก็บประเภทของข้อผิดพลาดที่พบว่าเกิดจากส่วนใดในการทำงาน |
| 5 | STATUS | เอนทิตีเก็บข้อมูลสถานะข้อผิดพลาดต่างๆ |
| 6 | PROJECT | เอนทิตีเก็บข้อมูล โครงการระบบสารสนเทศที่ต้องดำเนินการทดสอบ |
| 7 | LOG_DEFECT | เอนทิตีเก็บข้อมูลข้อผิดพลาดหรือประเด็นแก้ไขที่ผู้ใช้ระบบมีการเปลี่ยนแปลงต่างๆ ของระบบงาน |
| 8 | ATTACHMENT | เอนทิตีเก็บข้อมูลเอกสารแนบของข้อผิดพลาดหรือประเด็นแก้ไขจากการทดสอบระบบงาน |
| 9 | EMPLOYEE | เอนทิตีเก็บข้อมูลพนักงานที่มีสิทธิในการใช้งานของระบบ |
| 10 | POSITION | เอนทิตีตำแหน่งของพนักงาน |
| 11 | TEAM | เอนทิตีทีมที่พนักงานสังกัด และกำหนดสิทธิในการเข้าใช้งานระบบ |
| 12 | TESTPLAN_PROJECT | เอนทิตีแผนการทดสอบของโครงการที่ทำการทดสอบในระบบ |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 แสดงอีอาร์ไดอะแกรมของระบบติดตามเพื่อติดตาม และรายงานข้อผิดพลาดที่เกิดจากการทดสอบระบบธนาคารฯ

จากอีอาร์ไออะแกรม สามารถอธิบายความสัมพันธ์ระหว่างเอนทิตีได้ดังนี้

- ความสัมพันธ์ระหว่างเอนทิตี EMPLOYEE กับ เอนทิตี EMPLOYEE เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม(1: M relationship) โดยพนักงานผู้ใช้งานระบบ 1 คน ซึ่งเป็นผู้บังคับบัญชา (Test Manager) มีพนักงานผู้ใช้งานระบบซึ่งเป็นผู้ใต้บังคับบัญชาได้หลายคน หรือพนักงานผู้ใช้งานระบบที่เป็นผู้ใต้บังคับบัญชาหลายคนมีพนักงานผู้ใช้งานระบบที่เป็นผู้บังคับบัญชาได้เพียงคนเดียว

- ความสัมพันธ์ระหว่างเอนทิตี POSITION กับ เอนทิตี EMPLOYEE เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1: M relationship) โดยพนักงานผู้ใช้งานระบบ 1 คน ต้องมีตำแหน่งงานได้เพียง 1 ตำแหน่ง แต่ในตำแหน่งงานเดียวกันอาจมีพนักงานผู้ใช้งานระบบได้หลายคน

- ความสัมพันธ์ระหว่างเอนทิตี TEAM กับ เอนทิตี EMPLOYEE เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1: M relationship) โดยพนักงานผู้ใช้งานระบบ 1 คน สังกัดทีมได้ 1 ทีม และในแต่ละทีม อาจมีพนักงานสังกัดอยู่ได้มากกว่า 1 คน และในทีมแต่ละทีมต้องมีพนักงานสังกัดอย่างน้อย 1 คน

- ความสัมพันธ์ระหว่างเอนทิตี DEFECT กับ เอนทิตี EMPLOYEE เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (M: N relationship) โดยพนักงานผู้ใช้งานระบบจะสามารถบันทึกหรือแก้ไขข้อผิดพลาดได้หลาย ๆ ข้อผิดพลาดที่เกิดขึ้นจากการทดสอบระบบ และแต่ละข้อผิดพลาดที่เกิดขึ้นจะสามารถบันทึกหรือแก้ไขได้โดยพนักงานหลาย ๆ คน ซึ่งเมื่อเกิดเหตุการณ์ลักษณะนี้ขึ้น ในการออกแบบฐานข้อมูลจำเป็นต้องสร้าง Composit Entity ซึ่งก็คือต้องทำการแตกเป็นความสัมพันธ์แบบ 1: M โดยมีชื่อ LOG_DEFECT เพื่อเก็บข้อมูลข้อผิดพลาดหรือประเด็นแก้ไขที่พนักงานผู้ใช้งานระบบมีการเปลี่ยนแปลงต่าง ๆ ของระบบงาน

- ความสัมพันธ์ระหว่างเอนทิตี STATUS กับ LOG_DEFECT เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1: M relationship) โดยสถานะของข้อผิดพลาด 1 สถานะ ปรากฏอยู่ในการเก็บข้อมูลข้อผิดพลาดหรือประเด็นแก้ไขที่พนักงานผู้ใช้งานระบบมีการเปลี่ยนแปลงต่าง ๆ ของระบบงานได้หลายเรคอร์ด

- ความสัมพันธ์ระหว่างเอนทิตี STATUS กับ DEFECT เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1: M relationship) โดยสถานะของข้อผิดพลาด 1 สถานะสามารถปรากฏอยู่ในข้อผิดพลาดที่พบจากการทดสอบระบบได้หลายข้อผิดพลาด แต่ 1 ข้อผิดพลาดสามารถมีได้เพียง 1 สถานะเท่านั้น

- ความสัมพันธ์ระหว่างเอนทิตี DEFECT กับ เอนทิตี ATTACHMENT เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1: M relationship) โดย 1 ข้อผิดพลาดที่เกิดจากการทดสอบระบบจะมีเอกสารแนบได้หลาย ๆ ฉบับ และเอกสารแนบ 1 ฉบับ จะมีการอ้างอิงข้อผิดพลาดเพียง 1 ข้อผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ความสัมพันธ์ระหว่างเอนทิตี TYPE เอนทิตี DEFECT เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1: M relationship) โดยประเภทของข้อผิดพลาด 1 ประเภทปรากฏอยู่ในข้อผิดพลาดที่พบจากการทดสอบระบบได้หลายข้อผิดพลาด แต่ 1 ข้อผิดพลาดสามารถระบุประเภทได้ 1 ประเภทของข้อผิดพลาดเท่านั้น

- ความสัมพันธ์ระหว่างเอนทิตี PRIORITY กับ เอนทิตี DEFECT เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1: M relationship) โดยความสำคัญของข้อผิดพลาด 1 ความสำคัญปรากฏอยู่ในข้อผิดพลาดที่พบจากการทดสอบระบบได้หลายข้อผิดพลาด แต่ 1 ข้อผิดพลาดสามารถระบุความสำคัญได้ 1 ความสำคัญของข้อผิดพลาดเท่านั้น

- ความสัมพันธ์ระหว่างเอนทิตี SEVERITY กับ เอนทิตี DEFECT เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1: M relationship) โดยความรุนแรงของข้อผิดพลาด 1 ความรุนแรงปรากฏอยู่ในข้อผิดพลาดที่พบจากการทดสอบระบบได้หลายข้อผิดพลาด แต่ 1 ข้อผิดพลาดสามารถระบุความรุนแรงได้ 1 ความรุนแรงของข้อผิดพลาดเท่านั้น

- ความสัมพันธ์ระหว่างเอนทิตี PROJECT กับ เอนทิตี DEFECT เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1: M relationship) โดย PROJECT 1 PROJECT สามารถมีข้อผิดพลาดที่เกิดจากการทดสอบระบบได้หลายข้อผิดพลาด แต่ใน 1 ข้อผิดพลาดสามารถอยู่ภายใต้ PROJECT นั้น ๆ ได้เพียง 1 PROJECT เท่านั้น

- ความสัมพันธ์ระหว่างเอนทิตี PROJECT กับ เอนทิตี TESTPLAN PROJECT เป็นความสัมพันธ์แบบหนึ่งต่อกลุ่ม (1: M relationship) โดย PROJECT 1 PROJECT สามารถมีแผนการทดสอบได้หลายแผนการทดสอบ แต่ใน 1 แผนการทดสอบสามารถอยู่ภายใต้ PROJECT นั้น ๆ ได้เพียง 1 PROJECT เท่านั้น

4.2 พจนานุกรมข้อมูล (Data Dictionary)

จากอีอาร์โคอะแกรมที่ได้นั้นสามารถแปลงเอนทิตีให้เป็นฐานข้อมูลเชิงสัมพันธ์ได้ โดยนำเสนอพจนานุกรมข้อมูลที่จะแสดงให้เห็นรายละเอียดของข้อมูลต่าง ๆ ที่เกี่ยวข้องกับการทำงานของระบบ ซึ่งได้เป็นตารางที่มีความสัมพันธ์กันทั้งหมด 11 ตาราง ดังรายการตามตารางที่ 4.2 ถึงตารางที่ 4.13

ตารางที่ 4.2 พจนานุกรมข้อมูลของเอนทิตี DEFECT

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิดข้อมูล | ความยาว | คีย์ | ตารางอ้างอิง |
|-----------------|---|------------|---------|------|--------------|
| DEF_ID | รหัสข้อผิดพลาด | varchar | 6 | PK | |
| DEF_CATAGORY | กลุ่มประเภทข้อมูล (อาทิ Defect or Issue) | varchar | 6 | | |
| DEF_TOPIC | หัวข้อข้อผิดพลาด | varchar | 50 | | |
| DEF_DETAIL | รายละเอียดข้อผิดพลาด | varchar | 300 | | |
| DEF_TESTSTEP | ขั้นตอนการทดสอบที่พบข้อผิดพลาด | varchar | 300 | | |
| TESTCASENO | รหัสเทสเคสที่ทดสอบ | varchar | 20 | | |
| CREATE_DATE | วันที่บันทึกข้อผิดพลาด | date | | | |
| DEF_RESOLUTION | การแก้ไขข้อผิดพลาด | varchar | 300 | | |
| LAST_UPDATE | วันที่แก้ไขข้อผิดพลาดล่าสุด | Date | | | |
| LAST_MODI_BY | พนักงานที่แก้ไขข้อผิดพลาดล่าสุด | varchar | 20 | | |
| USER_OWNER | ชื่อผู้ใช้งานระบบ | varchar | 20 | | |
| DATE_CLOSED | วันที่ปิดข้อผิดพลาดแล้ว | Date | | | |
| TYPE_ID | รหัสประเภทข้อผิดพลาด | int | 3 | FK | TYPE |
| PRI_ID | รหัสระดับความสำคัญ | int | 3 | FK | PRIORITY |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 (ต่อ) พจนานุกรมข้อมูลของเอนทิตี DEFECT

| | | | | | |
|-----------|-------------------------|---------|---|----|----------|
| SER_ID | รหัสระบุความรุนแรง | int | 3 | FK | SEVERITY |
| PROJ_ID | รหัสโครงการที่ ทดสอบ | varchar | 5 | FK | PROJECT |
| STATUS_ID | รหัสสถานะ | int | 3 | FK | STATUS |

ตารางที่ 4.3 พจนานุกรมข้อมูลของเอนทิตี PRIORITY

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิด ข้อมูล | ความ ยาว | คีย์ | ตารางอ้างอิง |
|-----------------|-----------------------|----------------|-------------|------|--------------|
| PRI_ID | รหัสระบุ ความสำคัญ | int | 3 | PK | |
| PRI_NAME | ชื่อความสำคัญ | varchar | 6 | | |
| PRI_RANK | ระดับความสำคัญ | int | 3 | | |

ตารางที่ 4.4 พจนานุกรมข้อมูลของเอนทิตี SEVERITY

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิด ข้อมูล | ความ ยาว | คีย์ | ตารางอ้างอิง |
|-----------------|------------------------|----------------|-------------|------|--------------|
| SER_ID | รหัสระบุความ รุนแรง | int | 3 | PK | |
| SER_NAME | ชื่อความรุนแรง | varchar | 8 | | |
| SER_RANK | ระดับความรุนแรง | varchar | 3 | | |

ตารางที่ 4.5 พจนานุกรมข้อมูลของเอนทิตี TYPE

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิด ข้อมูล | ความ ยาว | คีย์ | ตารางอ้างอิง |
|-----------------|--------------------------|----------------|-------------|------|--------------|
| TYPE_ID | รหัสประเภท ข้อผิดพลาด | int | 3 | PK | |
| TYPE_NAME | ชื่อประเภทข้อผิดพลาด | varchar | 20 | | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 พจนานุกรมข้อมูลของเอนทิตี STATUS

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิดข้อมูล | ความยาว | คีย์ | ตารางอ้างอิง |
|-----------------|-----------------|------------|---------|------|--------------|
| STATUS_ID | รหัสสถานะ | int | 3 | PK | |
| STATUS_DESC | รายละเอียดสถานะ | varchar | 10 | | |

ตารางที่ 4.7 พจนานุกรมข้อมูลของเอนทิตี LOG_DEFECT

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิดข้อมูล | ความยาว | คีย์ | ตารางอ้างอิง |
|--------------------|--|------------------|---------|--------|--------------|
| DEF_ID | รหัสข้อผิดพลาด | varchar | 6 | PK,FK2 | DEFECT |
| STATUS_ID | รหัสสถานะ | int | 3 | PK,FK2 | STATUS |
| EMP_ID | รหัสพนักงาน | int | 6 | PK,FK1 | EMPLOYEE |
| DATE TIME_STAMP | วันเวลาที่ทำการ เปลี่ยนแปลง สถานะ Defect | Date and Time | | | |
| NOTE | รายละเอียด ข้อมูลเพิ่มเติม | varchar | 300 | | |

ตารางที่ 4.8 พจนานุกรมข้อมูลของเอนทิตี ATTACHMENT

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิดข้อมูล | ความยาว | คีย์ | ตารางอ้างอิง |
|-----------------|-------------------------------|------------|---------|------|--------------|
| DOC_ID | รหัสเอกสาร | integer | 7 | PK | |
| DOC_NAME | ชื่อเอกสาร | varchar | 30 | | |
| DATE_ATTACH | วันที่เพิ่มเอกสาร เข้าระบบ | date | | | |
| DEF_ID | รหัสข้อผิดพลาด | varchar | 6 | FK | DEFECT |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.9 พจนานุกรมข้อมูลของเอนทิตี PROJECT

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิดข้อมูล | ความยาว | คีย์ | ตารางอ้างอิง |
|-----------------|---------------------------------------|------------|---------|------|--------------|
| PROJ_ID | รหัสโครงการ | varchar | 5 | PK | |
| PROJ_NAME | ชื่อโครงการ | varchar | 50 | | |
| PROJ_MODULE | โมดูลย่อยสำหรับโครงการนั้น ๆ ที่ทดสอบ | varchar | 50 | | |
| PROJ_DEADLINE | กำหนดวันที่สิ้นสุดของโครงการ | date | | | |
| PROJ_DESC | รายละเอียดโครงการ | varchar | 200 | | |

ตารางที่ 4.10 พจนานุกรมข้อมูลของเอนทิตี TESTPLAN_PROJECT

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิดข้อมูล | ความยาว | คีย์ | ตารางอ้างอิง |
|-----------------|---------------------------------|------------|---------|------|--------------|
| TP_ID | รหัสแผนการทดสอบ | varchar | 15 | PK | |
| TP_TOPIC | หัวข้อแผนการทดสอบ | varchar | 50 | | |
| TOTAL_TESTCASE | จำนวนเทสเคสที่ทำการทดสอบทั้งหมด | integer | 3 | | |
| STARTDATE | วันที่เริ่มต้นสอบ | date | | | |
| ENDDATE | วันที่สิ้นสุดการทดสอบ | date | | | |
| TOTALDAY_TEST | จำนวนวันที่ทำการทดสอบทั้งหมด | integer | 3 | | |
| PROJ_ID | รหัสโครงการ | varchar | 5 | FK | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.11 พจนานุกรมข้อมูลของเอนทิตี EMPLOYEE

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิดข้อมูล | ความยาว | คีย์ | ตารางอ้างอิง |
|-----------------|---------------------------------|------------|---------|------|--------------|
| EMP_ID | รหัสพนักงาน | varchar | 6 | PK | |
| EMP_FIRSTNAME | ชื่อพนักงาน | varchar | 30 | | |
| EMP_LASTNAME | นามสกุล พนักงาน | varchar | 30 | | |
| EMP_PASSWORD | รหัสผ่านผู้ใช้ ระบบงาน | varchar | 8 | | |
| EMP_EMAIL | อีเมลพนักงาน | varchar | 30 | | |
| EMP_PHONENO | หมายเลข โทรศัพท์ | varchar | 15 | | |
| UNDER_MANAGER | อยู่ภายใต้บังคับ บัญชาของใคร | int | 6 | FK | EMPLOYEE |
| POS_ID | รหัสตำแหน่ง | varchar | 6 | FK | POSITION |
| TEAM_ID | รหัสกลุ่ม ผู้ใช้งาน | varchar | 6 | FK | TEAM |

ตารางที่ 4.12 พจนานุกรมข้อมูลของเอนทิตี TEAM

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิดข้อมูล | ความยาว | คีย์ | ตารางอ้างอิง |
|-----------------|----------------------------------|------------|---------|------|--------------|
| TEAM_ID | รหัสกลุ่มงานที่ พนักงานสังกัด | varchar | 6 | PK | |
| TEAM_NAME | ชื่อกลุ่มงานที่ พนักงานสังกัด | varchar | 20 | | |
| TEAM_ROLE | กำหนดสิทธิแต่ กลุ่มงาน | varchar | 20 | | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.13 พจนานุกรมข้อมูลของเอนทิตี POSITION

| ชื่อแอททริบิวต์ | คำอธิบาย | ชนิดข้อมูล | ความยาว | คีย์ | ตารางอ้างอิง |
|-----------------|-------------|------------|---------|------|--------------|
| POS_ID | รหัสตำแหน่ง | varchar | 6 | PK | |
| POS_NAME | ชื่อตำแหน่ง | varchar | 50 | | |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 สิทธิการใช้งานระบบสารสนเทศเพื่อติดตาม และรายงานข้อผิดพลาดหรือประเด็นการแก้ไขระบบ

ในการพัฒนาโปรแกรมสามารถสรุปกลุ่มผู้ใช้งานระบบจากยูสเคสไดอะแกรม (Use Case Diagram) โดยแสดงสิทธิการใช้งานดังนี้

ตารางที่ 4.14 ตารางแสดงการเปรียบเทียบสิทธิการใช้งานในระบบดังนี้

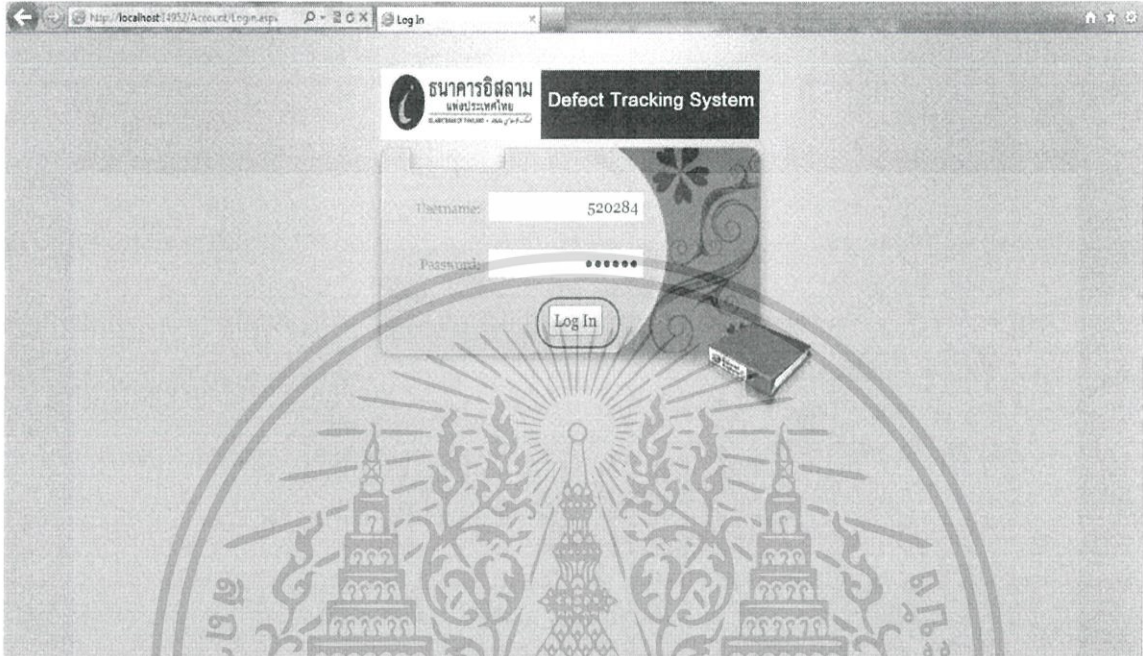
| Authorization | Tester | Test Manager | Developer | BA | SA | PM |
|------------------------------|--------|--------------|-----------|----|----|----|
| Create Defect / Issue | ✓ | | | | | |
| Closed Defect / Issue | ✓ | | | | | |
| Update Defect / Issue | ✓ | ✓ | ✓ | | | |
| Search Defect / Issue | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| View Defect / Issue Activity | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Assign Defect | | ✓ | | | | |
| Resolve Defect | | ✓ | | | | |
| Generate Summary Report | | ✓ | | ✓ | ✓ | ✓ |
| Manage User Profile | | ✓ | | | | |
| Manage for Testing | | ✓ | | | | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ออกแบบหน้าจอส่วนต่อประสานกับผู้ใช้

4.4.1 การเข้าสู่ระบบ (Log in)

เมื่อเริ่มโปรแกรม โปรแกรมจะแสดงหน้าจอเข้าสู่ระบบ ให้ผู้ใช้งานระบบระบุผู้ใช้งาน (Username) และ รหัสผ่าน (Password)



รูปที่ 4.2 แสดงหน้าจอเข้าสู่ระบบ

เมื่อระบุชื่อผู้ใช้งานและรหัสผ่านได้ถูกต้อง จะเข้าสู่หน้าหลักของระบบ ซึ่งจะประกอบด้วยเมนูหลัก ดังรูปที่ 4.3 แสดงหน้าจอเข้าสู่ระบบ โดยแบ่งสิทธิ์การใช้งานระบบและแสดงเมนูการทำงานได้ดังนี้

- **สิทธิ์สำหรับผู้ทดสอบระบบ (Tester)**
 - เมนู Personal Inbox
 - เมนู Create Defect / Issue
 - เมนู Search Defect / Issue
- **สิทธิ์สำหรับผู้จัดการส่วนทดสอบระบบตรวจสอบการทดสอบ (Test Manager)**
 - เมนู Personal Inbox
 - เมนู Search Defect / Issue
 - เมนู Generate Summary Report
 - เมนู Administration
 -

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **สิทธิ์สำหรับผู้พัฒนาระบบ (Developer)**
 - เมนู Personal Inbox
 - เมนู Search Defect / Issue
- **สิทธิ์สำหรับ นักวิเคราะห์ธุรกิจ (Business Analyst), นักวิเคราะห์ระบบสารสนเทศ (System Analyste), ผู้บริหารงานโครงการสารสนเทศ (Project Manager)**
 - เมนู Search Defect / Issue
 - เมนู Generate Summary Report

4.4.2 หน้าจอแสดงกล่องงานส่วนตัว (Personal Inbox)

แสดงกล่องงานส่วนตัวที่มีการ Assign ข้อผิดพลาดมายังผู้เกี่ยวข้อง ซึ่งสามารถทำการปรับปรุงแก้ไข, ดูรายละเอียด และ ลบข้อผิดพลาดนั้นๆ ได้

- แสดงกล่องงานส่วนตัวของผู้ทดสอบระบบ (Tester)

| Defect No. | Topic | Priority | Severity | Status | Type | Assign To | Function |
|------------|---|----------|----------|--------|--------|--------------------|------------------------|
| DF002 | Cannot open Database | High | Critical | New | Coding | Test Manager Area | Update / View / Delete |
| DF003 | สร้างหลักรับระบบผิดพลาด สร้างหลักรับระบบไม่เสร็จ | High | Critical | New | Coding | Test Manager Area | Update / View / Delete |
| DF004 | Tab หลักรับระบบผิดพลาด | Medium | Moderate | Assign | Coding | Developer-Nattapon | Update / View / Delete |
| DF005 | สัญญาสิ้นเชิงหรือลดราคา ไม่ส่งลงคอมมู เป็นตาราง | High | Critical | Assign | Coding | Developer-Nattapon | Update / View / Delete |
| DF006 | สร้างใบคำสิ้นเชิง ขึ้นคณณณ หรือเป็น นสคง "CALINCBFailure" | High | Critical | Assign | Coding | Developer-Nattapon | Update / View / Delete |

© Copyright 2013. ISLAMIC BANK All Rights Reserved.
Designed by Suchalai.K
Ver.3.0-build Date 17-02-2013

รูปที่ 4.3 แสดงหน้าจอเข้าสู่ระบบสำหรับผู้ทดสอบระบบ (Tester) ซึ่งแสดงกล่องงานส่วนตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แสดงกล่องงานส่วนตัวของผู้จัดการส่วนทดสอบระบบตรวจสอบการทดสอบ (Test Manager)

© Copyright 2013, ISLAMIC BANK All Rights Reserved.
Designed by Suchalai.K
Ver.3.0-build Date 17-02-2013

รูปที่ 4.4 แสดงหน้าจอกล่องงานส่วนตัวของผู้จัดการส่วนทดสอบระบบ (Test Manager)

- แสดงกล่องงานส่วนตัวของผู้พัฒนาระบบ (Developer)

© Copyright 2013, ISLAMIC BANK All Rights Reserved.
Designed by Suchalai.K
Ver.3.0-build Date 17-02-2013

รูปที่ 4.5 แสดงหน้าจอกล่องงานส่วนตัวของผู้พัฒนาระบบ (Developer)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- แสดงเมนูค้นหาและรายงานเท่านั้น สำหรับนักวิเคราะห์ธุรกิจ (Business Analyst), นักวิเคราะห์ระบบสารสนเทศ (System Analyste), ผู้บริหารงานโครงการสารสนเทศ (Project Manager) จะไม่แสดงกล่องส่วนตัวเนื่องจากไม่ต้องมีการปรับปรุงแก้ไขสถานะใดๆ สามารถติดตามสถานะต่างๆ ของข้อผิดพลาดเท่านั้น

ธนาคารอิสลามแห่งประเทศไทย

Defect Tracking System

Search Defect/Issue Generate Summary Report

รหัสผู้ใช้งาน : 540360 ชื่อผู้ใช้งาน: Premwit ทีมงาน : Business Analyst [Log out]

Search Defect / Issue

Category : Defect Issue

Defect / Issue No. :

Status : Severity : Priority : Type :

| Defect No. | Description | Priority | Severity | Status | Type | Assign To |
|--------------|--|----------|----------|--------|--------|-------------------|
| <u>DF003</u> | สร้างหลักประกันระบบแสดง สร้างหลักประกันไม่สร้าง | High | High | New | Coding | Test Manager Aree |

© Copyright 2013 ISLAMIC BANK All Rights Reserved.
Designed by Suchalai.K
Ver.3.0-build Date 17-02-2013

รูปที่ 4.6 แสดงหน้าจอเข้าสู่ระบบสำหรับ BA, SA, PM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.3 หน้าจอแสดงการสร้าง/บันทึกข้อผิดพลาดหรือประเด็นแก้ไขที่พบจากทดสอบฯ (Create Defect/Issue)

สามารถสร้างข้อมูลข้อผิดพลาดหรือประเด็นแก้ไขที่พบจากทดสอบฯ โดยผู้ทดสอบระบบเท่านั้น ดังแสดงรูปภาพที่ 4.7 แสดงหน้าจอ Create Defect /Issue

ธนาคารอิสลามแห่งประเทศไทย Defect Tracking System

Personal Inbox Create Defect/Issue Search Defect/Issue

กรุณาใช้งาน : S20299 ชื่อใช้งาน: Suchalai ตำแหน่ง : Tester [Log out]

Create Defect / Issue

New Retest Closed

Generate No.DF0009

Category : Defect Issue

Defect Topic : Offering Letter ไม่แสดงค่า Front end Fee Test Case No : CAP-PIN-007520299-190

Status : Priority : Severity :

Project : Module : Type :

Reported by : Assign Team : Name List :

To : From : User Owner :

Last Modified by : Last Update : Create Date :

Test Step : Defect Detail

- ทดสอบถึงขั้นตอนตรวจสอบสัญญาในระบบ CAP Role ที่มีการสิ้นเชื่อ
- เลือกชุดสัญญาสิ้นเชื่อ เลือกเอกสาร Offering Letter เพื่อ Preview แต่ข้อมูลที่แสดงในเอกสารใบถูกต้อง ได้แก่ตัวต้นแสดงค่า Front end Fee ไม่แสดงยอดเงินค่าธรรมเนียม

แบบฟอร์ม Offering Letter ไม่แสดงค่า Front end Fee ซึ่งต้องแสดงค่าธรรมเนียมทั้งในหน้า Screen บทสรุปและ เอกสาร Offering Letter 2% ของวงสิ้นเชื่อ

Note Log :

Tester : ทหารสร้างข้อมูลข้อผิดพลาดใหม่

Select Document for Upload :

Attach File : C:\Users\suchalai\Desktop [Browse...] [Capture1_DF0009]

Attach File : C:\Users\suchalai\Desktop [Browse...] [Capture2_DF0009]

Attach File : [Browse...] [Add Attach File]

[Save] [OK] [Cancel]

© Copyright 2013. ISLAMIC BANK All Rights Reserved.
Designed by Suchalai.K
Ver.3.0-build Date 17-02-2013

รูปที่ 4.7 แสดงหน้าจอ Create Defect /Issue

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.4 หน้าจอแสดงการค้นหารายละเอียดข้อผิดพลาด หรือประเด็นแก้ไขที่พบจากทดสอบฯ

(Search Defect/Issue)

สามารถทำการค้นหา Defect หรือ Issue เพื่อค้นหารายละเอียดข้อผิดพลาด เพื่อเข้าไปดูรายละเอียด หรือปรับปรุงแก้ไขสถานะต่าง ๆ ของข้อผิดพลาดและประเด็นต่าง ๆ ตามสิทธิ์การใช้งาน ดังรูปที่ 4.8 แสดงหน้าจอการค้นหาข้อผิดพลาด

ธนาคารอิสลามแห่งประเทศไทย
ISLAMIC BANK OF THAILAND

Defect Tracking System

Personal Inbox Create Defect/Issue Search Defect/Issue

รหัสผู้ใช้งาน : 520284 ชื่อผู้ใช้งาน : Suchalai ทีมงาน : Tester | Log Out

Search Defect / Issue

Category : Defect Issue

Defect / Issue No : DF001

Status : Assigned Severity : Critical Priority : High Type : Requirement

Search Cancel

| Defect No. | Description | Priority | Severity | Status | Type | Assign To |
|------------|-------------------|----------|----------|----------|-------------|--------------------|
| DF001 | ATM Activate Card | High | Critical | Assigned | Requirement | Test Manager, Aree |

© Copyright 2013. ISLAMIC BANK All Rights Reserved.
Designed by Suchalai K.
Ver.3.0-build Date 17-02-2013

รูปที่ 4.8 แสดงหน้าจอการค้นหาคือข้อผิดพลาดจากการทดสอบระบบฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.5 หน้าจอปรับปรุงแก้ไขข้อผิดพลาดที่เกิดจากการทดสอบระบบ

สามารถคลิก Icon Update ข้อผิดพลาดได้จากหน้าจอเมนู Personal Inbox เพื่อทำการปรับปรุง แก้ไขข้อผิดพลาดที่เกิดจากการทดสอบระบบ ฯแสดงดังรูปที่ 4.9 แสดงหน้าจอเพื่อคลิก Update เปลี่ยนแปลงรายละเอียดหรือสถานะต่าง ๆ ทั้งนี้สามารถเปลี่ยนแปลงสถานะต่าง ๆ ตามสิทธิ์การใช้งานได้ดังนี้

| Personal Inbox | | | | | | | |
|----------------|--|----------|----------|--------|--------|-------------------|------------------------|
| Defect No. | Description | Priority | Severity | Status | Type | Assign To | Funtion |
| DF002 | Cannot open Database | High | Critical | New | Coding | Test Manager,Aree | Update / View / Delete |
| DF003 | สร้างหลักกับกรรมากรแสดง สร้างหลักกับกรรมากรแสดง | High | Critical | New | Coding | Test Manager,Aree | Update / View / Delete |

รูปที่ 4.9 แสดงหน้าจอเพื่อคลิก Update เปลี่ยนแปลงรายละเอียดหรือสถานะต่าง ๆ

- สิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้ทดสอบ

สามารถเปลี่ยนแปลงสถานะได้ 3 สถานะดังนี้ New, Retest, Closed ซึ่งแสดงดังรูปที่ 4.10 แสดงแสดงหน้าจอสิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้ทดสอบ

Update Defect / Issue

New

Retest

Closed

Generate No.DF003

Category : Defect Issue

Defect Topic : สร้างหลักประกันระบบแสดง "สร้างหลักประกันไม่สำเร็จ" Test Case No : CAP-PIN-002520299-080111

Status : Priority : Severity :

Project : Module : Type :

Reported by : Assign To Team : Name List :

User Owner : Yanee Last Modified by : Last Update :

Create Date :

Test Step :

- คลิกปุ่มเพิ่มหลักประกัน ในรายละเอียดหลักประกันประเภทที่ดินพร้อมสิ่งปลูกสร้าง 1หลัก
- สงงานไปถึงขั้นตอนเบิกจ่ายงวดแรก ระบบแสดง สร้างประกันไม่สำเร็จ

Defect Detail :

- ระบบแสดง สร้างหลักประกันไม่สำเร็จ ซึ่งทำให้ผลกระทบในการส่งไปสร้างข้อมูลบนระบบ CBS

Select Document for Upload :

Attach File :

Attach File :

Attach File :

รูปที่ 4.10 แสดงหน้าจอสิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้ทดสอบ (Tester)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้จัดการส่วนทดสอบระบบตรวจสอบการทดสอบ (Test Manager)

สามารถเปลี่ยนแปลงสถานะได้ 2 สถานะดังนี้ Assigned, Resolve ซึ่งแสดงดังรูปที่

4.11 แสดงหน้าจอสิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้จัดการส่วนทดสอบระบบฯ

The screenshot displays the 'Update Defect / Issue' form in the Defect Tracking System. The form is titled 'Update Defect / Issue' and has two tabs: 'Assign' and 'Resolve'. The 'Assign' tab is active. The form includes the following fields and options:

- Status state:** A diagram showing a transition from 'Assign' to 'Assigned'.
- Generate No.:** DF003
- Category:** Defect (selected), Issue
- Defect Topic:** สร้างหลักประกันระบบแสดง"สร้างหลักประกันไม่สำเร็จ"
- Test Case No.:** CAP PIN 002520299 08011
- Status:** Assigne (selected), Priority: High, Severity: Critical
- Project:** CAP, Module: Product, Type: Coding
- Reported by:** Suchalal, Assign To Team: Test Manager, Name List: Aree
- User Owner:** Yanee, Last Modified by: Suchalal, Last Update: 16/2/2556
- Create Date:** 1/3/2556
- Test Step:**
 - คลิกปุ่มเพิ่มหลักประกัน ใสรายละเอียดหลักประกันประเภทที่ดินพร้อมสิ่งปลูกสร้าง 1 หลัก
 - สงงานไปถึงชั้นต้งเบิกจ่ายงวดแรก ระบบแสดง สร้างประกันไม่สำเร็จ
- Defect Detail:** ระบบแสดง สร้างหลักประกันไม่สำเร็จ ซึ่งหาไม่พบกระทบในการส่งไปสร้างข้อมูลระบบ CBS
- Note Log:**
- Select Document for Upload:** Attach File: Browse... (three times), Add Attach File
- Buttons:** Save, OK, Cancel

At the bottom of the page, there is a copyright notice: © Copyright 2013, ISLAMIC BANK All Rights Reserved. Designed by Suchalal K Ver.3.0-build Date 17-02-2013


รูปที่ 4.11 แสดงหน้าจอสิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้จัดการส่วนทดสอบ (Test Manager)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้พัฒนาระบบ (Developer)

สามารถเปลี่ยนแปลงสถานะ ได้ 2 สถานะดังนี้ Fixed, Reject Fix ซึ่งแสดงดังรูปที่

4.12 แสดงหน้าจอสิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้พัฒนาระบบ (Developer)



ธนาคารอิสลาม
ประเทศไทย

Defect Tracking System


Personal Inbox Search Defect/Issue
รหัสผู้ใช้งาน : 530125 ชื่อผู้ใช้งาน: Nathapon ชื่อนาม : Developer [Log out]

Update Defect / Issue

Fixed
Reject Fix

Generate No.DF005

Status state :



Category : Defect Issue

Defect Topic : สัญญาสินเชื่อหัวขอตราสารไม่แสดงข้อมูล เป็นค่างง Test Case No : CAP-PIN-004520299-08011:

Status : Priority : Severity :

Project : Module : Type :

Reported by : Assign-To Team : Name List :

User Owner: Last Modified by : Last Update :

Create Date :

Test Step :

-ทดสอบถึงขั้นตอนตรวจสอบสัญญาในระบบ CAP Role ที่มีการคืนเงิน
-เลือกชุดสัญญาคืนเงิน เลือกเอกสาร สัญญาคืนเงิน Preview แล้วหัวขอตราสารไม่แสดงยอดเงินตรา

Defect Detail :

-สัญญาคืนเงินหัวขอตราสารไม่แสดงข้อมูล เป็นค่างง ข้อมูลที่ต้องแสดง อาทิ วงเงินของตราสาร รวมถึงเอกสารสารบัญชาระยะ ค่าข้อมูลทุกค่าที่ไม่แสดงในเอกสารไปด้วยกัน เป็นค่างงหมด

Note Log :

>> Developer ทดสอบแก้ไขเรียบร้อยแล้ว สาเหตุเกิดจากใน Table Database ดังตารางมาแสดงค่า Null ซึ่งได้ทำการแก้ไข Coding ให้สามารถดึงข้อมูลที่สถานะแสดงในเอกสารเรียบร้อยแล้ว สามารถทำการตรวจสอบและ Retest ได้เลย

Select Document for Upload :

Attach File :

Attach File :

Attach File :

© Copyright 2013. ISLAMIC BANK All Rights Reserved.
Designed By Suchalai.K
Ver.3.0-build Date 17-02-2013

รูปที่ 4.12 แสดงหน้าจอสิทธิ์เปลี่ยนแปลงสถานะสำหรับผู้พัฒนาระบบ (Developer)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.6 หน้าจอเรียกดูรายละเอียด สถานะ กิจกรรมต่าง ๆ ของข้อผิดพลาดหรือประเด็นต่าง ๆ

สามารถรายละเอียดข้อมูลโดยการคลิกเมนู Personal Inbox และคลิกปุ่ม View เพื่อเรียกดูรายละเอียด สถานะต่าง หรือกิจกรรมที่เกิดขึ้นกับข้อผิดพลาดหรือประเด็นต่าง ๆ ที่ถูกสร้างขึ้นในระบบฯ ดังรูปที่ 4.13 แสดงหน้าจอเพื่อเรียกดูรายละเอียด สถานะ กิจกรรมต่าง ๆ ของข้อผิดพลาดหรือประเด็นต่าง ๆ

The screenshot displays the Defect Tracking System interface. At the top, there is a 'Personal Inbox' section with a table listing defects. Below this is a navigation bar with options like 'Personal Inbox', 'Create Defect/Issue', and 'Search Defect/Issue'. The main content area shows the 'View Defect / Issue Activity' for defect DF003. The form includes fields for Category, Defect Topic, Status, Project, Reported by, User Owner, Create Date, Test Step, and Defect Detail. There are also buttons for 'New', 'Retest', and 'Closed' at the top of the form area. At the bottom, there are 'Save', 'OK', and 'Cancel' buttons.

| Defect No. | Description | Priority | Severity | Status | Type | Assign To | Funtion |
|------------|---|----------|----------|--------|--------|-------------------|------------------------|
| DF002 | Cannot open Database | High | Critical | New | Coding | Test Manager,Arez | Update / View / Delete |
| DF003 | สร้างกล่องกิจกรรมแสดง "สร้างหลักประกันสำเร็จ" | High | Critical | New | Coding | Test Manager,Arez | Update / View / Delete |

Defect Tracking System

Personal Inbox Create Defect/Issue Search Defect/Issue

View Defect / Issue Activity

Generate No: DF003

Category: Defect Issue

Defect Topic: ... Test Case No: CAP-PIB-001-2520299-000

Status: New Priority: High Severity: Critical

Project: CAP Module: ... Type: Coding

Reported by: ... Assign To Team: Test Manager Name List: ...

User Owner: ... Last Modified by: ... Last Update: 16/02/2013

Create Date: 16/02/2013

Test Step: ... Defect Detail: ...

Select Document for Upload:

Attach File: ... Attach File: ... Attach File: ... Add Attach File

Save OK Cancel

© Copyright 2013, ISLAMIC BANK. All Rights Reserved.
Designed by Suchalai.K
Ver.3.0-build Date 17-02-2013

รูปที่ 4.13 แสดงหน้าจอเพื่อเรียกดูรายละเอียด สถานะ กิจกรรมต่าง ๆ ของข้อผิดพลาดหรือประเด็นต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.7 หน้าจอสร้างรายงานสรุปประเภทต่าง ๆ (Generate Summary Report and Chart)

การเรียกดูรายงานจะมีสิทธิ์ที่สามารถเรียกดูรายงานสรุปได้แก่ Test Manager, Project Manager, Business Analyst (BA), System Analyst (SA) เป็นต้น โดยสามารถแสดงรายงานหลัก ๆ ได้ดังนี้


- Defect Status Report: รายงานสถานะข้อผิดพลาดทั้งหมด
- Summary Defect for Project Report: รายงานข้อผิดพลาดแต่ละโครงการที่ทดสอบ

ทั้งนี้สามารถแสดงรายงานในรูปแบบบนหน้าจอ และแปลงเป็นรูปแบบไฟล์เอกสาร PDF, Excel แสดงตัวอย่างหน้าจอรายงานได้ดังรูปที่ 4.14 – 4.15

The screenshot displays the 'Defect Tracking System' interface. At the top, there is a header with the Islamic Bank of Thailand logo and the text 'ธนาคารอิสลามแห่งประเทศไทย' and 'ISLAMIC BANK OF THAILAND'. The main navigation bar includes 'Personal Inbox', 'Search Defect/Issue', 'Generate Summary Report', and 'Administration'. Below this, the user's name 'นายสุโขทัย : 490183' and role 'ชื่อผู้ใช้งาน: Aree ทีมงาน : Test Manager' are shown, along with a 'Log out' link. The 'Generate Summary Report' section is highlighted, showing two options: '>> Defect Status Report : รายงานสถานะข้อผิดพลาดทั้งหมด' and '>> Summary Defect for Project Report : รายงานข้อผิดพลาดแต่ละโครงการที่ทดสอบ'. Both options have PDF and Excel icons next to them. The footer contains copyright information: '© Copyright 2013. ISLAMIC BANK All Rights Reserved. Designed by Suchalai.K Ver.3.0-build Date 17-02-2013'.

รูปที่ 4.14 แสดงหน้าจอ Generate Summary Report

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



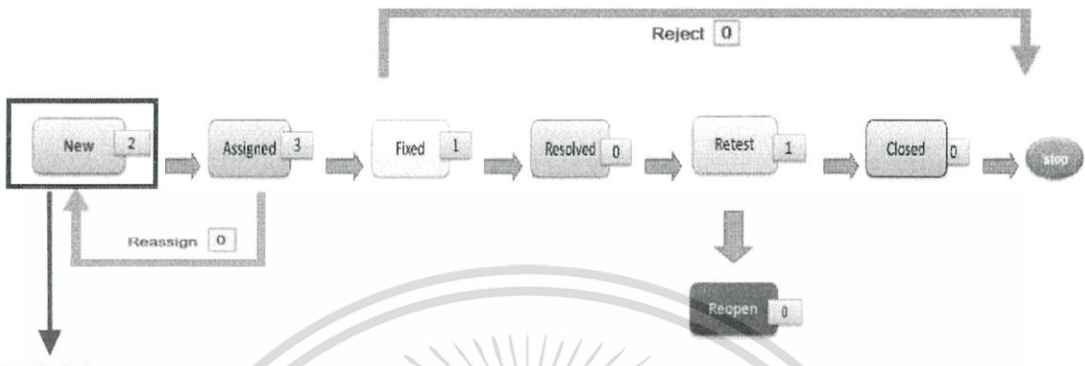
ธนาคารอิสลาม
แห่งประเทศไทย
ISLAMIC BANK OF THAILAND

Defect Tracking System

Personal Inbox
Search Defect/Issue
Generate Summary Report
Administration

รหัสผู้ใช้งาน : 490183 ชื่อผู้ใช้งาน : Aree ทีมงาน : Test Manager [[Log out](#)]

Defect Status Report



```

    graph LR
      New[New 2] --> Assigned[Assigned 3]
      Assigned --> Fixed[Fixed 1]
      Fixed --> Resolved[Resolved 0]
      Resolved --> Retest[Retest 1]
      Retest --> Closed[Closed 0]
      Closed --> Stop((stop))
      Closed --> Reopen[Reopen 0]
      Reopen --> New
      Closed --> Reject[Reject 0]
      Reassign[Reassign 0] --> New
  
```

Status : New

| Defect No. | Topic | Priority | Severity | Status | Type | Assign To |
|------------|--|----------|----------|--------|--------|-------------------|
| DF0006 | Cannot open Database | High | Critical | New | Coding | Test Manager Aree |
| DF0005 | สร้างหลักประกันแบบแสดง สร้างหลักประกันไม่สำเร็จ | High | Critical | New | Coding | Test Manager Aree |

[Back](#)

© Copyright 2013. ISLAMIC BANK All Rights Reserved.
 Designed by Suchalax K.
 ver.3.0-build Date 17-02-2013

รูปที่ 4.15 แสดงหน้าจอรายงานสถานะข้อผิดพลาดทั้งหมด (Defect Status Report)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.8 หน้าจอบริหารจัดการข้อมูลผู้ใช้งาน และสิทธิ์การเข้าใช้งาน

คลิกเมนู Administration และ เลือก Tab User Profile เพื่อทำการกรอกข้อมูลผู้ใช้งาน และ สิทธิ์การใช้งานระบบ โดยเพิ่มลบ แก้ไขข้อมูลแต่ทีมในระบบ หลังจากนั้นทำการคลิกปุ่ม Create User เพื่อสร้างชื่อผู้ใช้งาน (Username) และ กำหนดรหัสผู้ใช้งาน (Password) ดังรูปที่ 4.16 -4.17 แสดงหน้าจอ Manage User Profile และ ดังรูปที่ 4.10 แสดงหน้าจอ Create New Account


The screenshot shows the 'Administration' section of the 'Defect Tracking System' interface. The 'User Profile' tab is active, and the 'Create New User' form is displayed. The form contains the following fields and values:

- Employee ID : 550299
- Firstname : Suchada
- Lastname : Manee
- Email : chada_ma@ibank.co.th
- Phone Number : 1099
- Under Manager : Aree
- Team : Tester
- Position : เจ้าหน้าที่ 1
- Create Date User : 17/2/2556

Buttons for 'Create New User', 'Save', and 'Cancel' are visible. The footer of the page includes the copyright information: © Copyright 2012, ISLAMIC BANK All Rights Reserved. Designed by Suchalai.K Ver.2.0-build Date 13-01-2013.

รูปที่ 4.16 แสดงหน้าจอ Manage User Profile

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ธนาคารอิสลาม
แห่งประเทศไทย
ISLAMIC BANK OF THAILAND • BANK ISLAMIC THAÏLANDE

Defect Tracking System

Personal Inbox
Search Defect/Issue
Generate Summary Report
Administration

รหัสผู้ใช้งาน :490183 ชื่อผู้ใช้งาน: Aree ทีมงาน :Test Manager [Log out]

CREATE A NEW ACCOUNT

Use the form below to create a new account.

Passwords are required to be a minimum of 6 characters in length.

Account Information

User Name:
550299

E-mail:
suchada_ma@ibank.co.th

Password:
●●●●●●

Confirm Password:
●●●●●●


© Copyright 2012. ISLAMIC BANK All Rights Reserved.
Designed by Suchalai.K
Ver.2.0-build Date 13-01-2013

รูปที่ 4.17 แสดงหน้าจอ Create New Account

4.4.9 หน้าจอบริหารจัดการเพิ่มโปรเจกต์สำหรับการทดสอบ

คลิกเมนู Administration และ เลือก Tab Manage Project เพื่อทำการบริหารจัดการเพิ่มแก้ไข โปรเจกต์สำหรับการทดสอบ ว่ามีการทดสอบโปรเจกต์ไหนที่กำลังทดสอบ และเพิ่มแผนการทดสอบสำหรับโครงการนั้น ๆ ว่างเริ่มการทดสอบเมื่อไหร่ เรื่องอะไรที่ทดสอบ และมีจำนวนทดสอบทั้งหมดเท่าไร เพื่ออ้างอิงในคุณภาพรวมโครงการว่าพบข้อผิดพลาดมากน้อยแค่ไหน เป็นต้น ดังรูปที่ 4.18 แสดงหน้าจอ Manage Project for Testing และ Add Test Plan

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**ธนาคารอิสลาม
แห่งประเทศไทย**
ISLAMIC BANK OF THAILAND

Defect Tracking System

Personal Inbox
Search Defect/Issue
Generate Summary Report
Administration

รหัสผู้ใช้งาน : 490183 ชื่อผู้ใช้งาน : Aree ทีมงาน : Test Manager [Log out]

Administration

User Profile
Manage Project

Add Project for Test :

Project ID : _____ Project Name : _____

Module : _____ Project Description : _____

Create Date Project : 1/3/2556

Due Date Project : 1/3/2556

Add Test Plan :

Test Plan ID : _____ Test Plan Topic : _____

Total_Testcase : _____ Totalday_Test : _____

Startdate for Test : 1/3/2556

Enddate for Test : 1/3/2556

© Copyright 2013. ISLAMIC BANK All Rights Reserved.
 Designed by Suchalai.K
 Ver.3.0-build Date 17-02-2013

รูปที่ 4.18 แสดงหน้าจอ Manage Project for Testing and Add Test Plan

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการศึกษา

5.1 สรุปผลการศึกษา

การศึกษานี้เป็นการวิเคราะห์และออกแบบระบบสารสนเทศเพื่อติดตาม และรายงาน ข้อผิดพลาดหรือประเด็นการแก้ไขระบบ ของกลุ่มงานเทคโนโลยีสารสนเทศ ในส่วนทดสอบระบบสารสนเทศ โดยออกแบบและวิเคราะห์ระบบในการติดตามข้อผิดพลาดจากการทดสอบระบบ (Defect) หรือประเด็น (Issue) การแก้ไขปรับเปลี่ยน โปรแกรมหรือพารามิเตอร์ในโปรแกรมต่าง ๆ นั้น ซึ่งฝ่ายงานอื่นที่เกี่ยวข้องได้แก่ ส่วนวิเคราะห์ระบบงานธุรกิจ, ส่วนวิเคราะห์ระบบสารสนเทศ, ส่วนพัฒนาระบบสารสนเทศ, ส่วนบริหารงาน โครงการสารสนเทศ เป็นต้น สามารถจัดการและติดตามข้อผิดพลาดหรือประเด็นต่าง ๆ จากระบบให้เป็นไปตามกระบวนการทำงานขององค์กร

อนึ่งการศึกษากาวิเคราะห์และออกแบบระบบนี้ มีการนำขั้นตอนของวงจรการพัฒนาระบบประยุกต์ใช้ โดยเริ่มศึกษาปัญหาของการทำงานปัจจุบัน กำหนดวัตถุประสงค์ กำหนดภาพรวมของโครงการ จนกระทั่งถึงพิจารณาความต้องการของระบบงานใหม่ ในการศึกษาครั้งนี้ได้ครอบคลุมถึงการออกแบบจำลองเชิงแนวคิดของระบบงานใหม่เพื่อเป็นสื่อกลางในการสื่อสารระหว่างนักวิเคราะห์ระบบกับผู้ใช้งานและพัฒนาระบบให้เข้าใจความต้องการที่ชัดเจน ถูกต้อง ครบถ้วนในระบบงานใหม่ได้ตรงกัน ซึ่งนำไปสู่การพัฒนาระบบงานที่มีประสิทธิภาพสามารถตอบสนองความต้องการของผู้ใช้งานภายในองค์กร ได้เป็นอย่างดี

5.2 ข้อจำกัดของระบบ

ระบบไม่รวมกระบวนการ Test Management ระบบนี้จะทำตั้งแต่ Defect Tracking Process และการ Regression Test ของแต่ละ Defect

5.3 ข้อเสนอแนะและการพัฒนาระบบเพิ่มเติม

เมื่อมีการนำระบบดังกล่าวไปใช้งานแล้ว หากในอนาคตฝ่ายงานมีการขยายตัว หรือมีการเปลี่ยนแปลงขั้นตอนการทำงาน หรือต้องการเพิ่มเติมตามที่กล่าวไว้ในข้อจำกัดของระบบนั้น อาจต้องมีการปรับเปลี่ยนการทำงานของโปรแกรมเพิ่มเติม เพื่อให้รองรับกับกระบวนการทำงาน ให้เกิดประโยชน์สูงสุดกับองค์กร

บรรณานุกรม

ธาริน สัทธีธรรมชารี. 2548. **บริหารและจัดการฐานข้อมูลระดับมืออาชีพ SQLserver 2000 ฉบับสมบูรณ์**. กรุงเทพฯ : ชัคเซส มีเดีย.

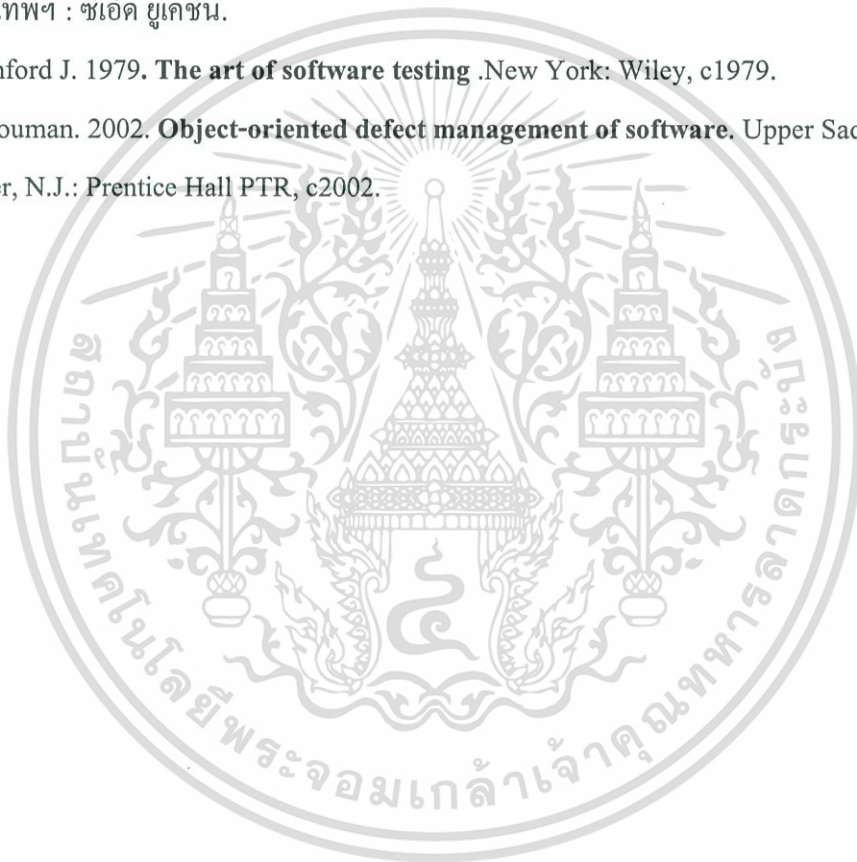
นwor แจ่มจำ. 2553. **สร้างเว็บไซต์ด้วย Dreamweaver ใช้ได้ถึง CS4+CS5**. กรุงเทพฯ : โปรวิชั่น.

สุรสิทธิ์ คิวประสพศักดิ์ และ นันทนี แขวงโสภณ. 2546. **อินไซต์ Visual Basic .NET**. กรุงเทพฯ : โปรวิชั่น.

โอภาส เขียมสิริวงศ์. 2549. **การวิเคราะห์และออกแบบระบบ Systems analysis and design** กรุงเทพฯ : ซีเอ็ด ยูเคชั่น.

Myers, Glenford J. 1979. **The art of software testing**. New York: Wiley, c1979.

Younessi Houman. 2002. **Object-oriented defect management of software**. Upper Saddle River, N.J.: Prentice Hall PTR, c2002.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

| | |
|----------------------|---|
| ชื่อ-นามสกุล | นางสาวสุชาลัย ขาวบาง |
| วัน เดือน ปีเกิด | 31 พฤษภาคม พ.ศ.2527 |
| ที่อยู่ | 298 ม.ประชากรไทย ซ.93 แขวงวังทองหลาง เขตวังทองหลาง กรุงเทพฯ 10310 |
| ประวัติการศึกษา | |
| พ.ศ. 2545 – 2547 | อนุปริญญา บริหารธุรกิจ สาขาคอมพิวเตอร์ธุรกิจ สถาบันเทคโนโลยีราชมงคล วิทยาเขตจรัลพงษ์ภูวนารอด |
| พ.ศ. 2547 – 2549 | วิทยาศาสตรบัณฑิต หลักสูตร 2 ปีต่อเนื่อง สาขาวิชาเทคโนโลยีสารสนเทศ มหาวิทยาลัยนอร์ทกรุงเทพ |
| ประสบการณ์การทำงาน | |
| พ.ศ. 2552 – ปัจจุบัน | เจ้าหน้าที่ 2 ทดสอบระบบสารสนเทศ ธนาคารอิสลามแห่งประเทศไทย |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้