

การออกแบบและพัฒนาระบบการสำหรับการพัฒนาซอฟต์แวร์สำเร็จรูป
ที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์



นายยุทธนันท์ กันหาชน

ศูนย์วิทยพัทยาการ จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

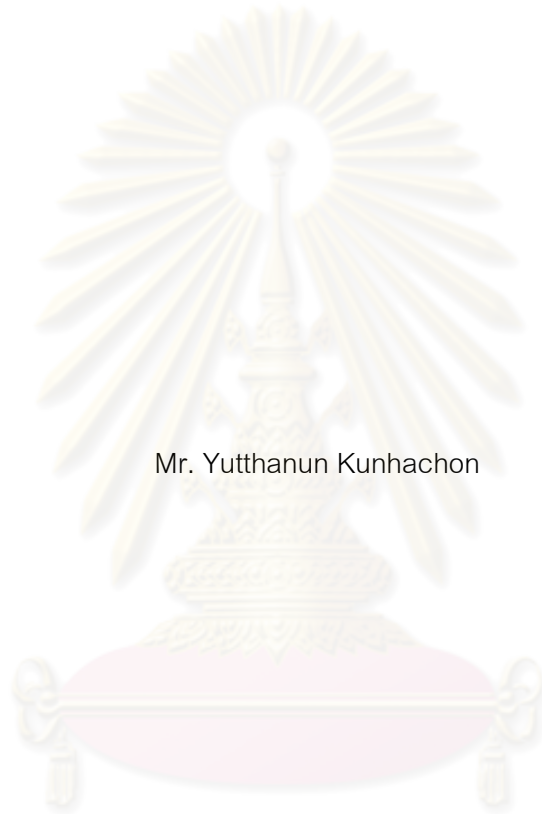
สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2553

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

PROCESS MODEL DESIGN AND DEVELOPMENT FOR COTS SOFTWARE
DEVELOPMENT BASED ON SOFTWARE PRODUCT LINE



Mr. Yutthanun Kunhachon

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Software Engineering

Department of Computer Engineering

Faculty of Engineering

Chulalongkorn University

Academic Year 2010

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์

การออกแบบและพัฒนาระบบการให้บริการพัฒนา

ซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ ซอฟต์แวร์

โดย

นายยุทธนันท์ กันหาชน


สาขาวิชา

วิศวกรรมซอฟต์แวร์

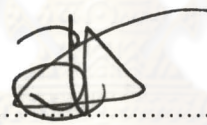
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

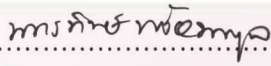
ผู้ช่วยศาสตราจารย์ นครทิพย์ พร้อมพูล

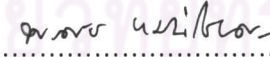
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้บัณฑิตวิทยาลัยเป็นส่วน
หนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

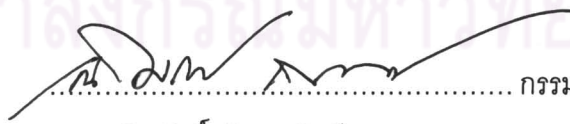
.....  คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.บุญสม เลิศธีรวงษ์)

คณะกรรมการสอบวิทยานิพนธ์

.....  ประธานกรรมการ
(รองศาสตราจารย์ ดร.วันชัย ธีรไพบูลย์)

.....  อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ นครทิพย์ พร้อมพูล)

.....  กรรมการ
(รองศาสตราจารย์ ดร.พรศิริ หมั่นไชยศรี)

.....  กรรมการภายนอกมหาวิทยาลัย
(ดร.เฉลิมศักดิ์ เลิศวงค์เสถียร)

ยุทธนันท์ กัณหาชน : การออกแบบและพัฒนากระบวนการสำหรับการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์. (Process Model Design and Development for COTS Software Development Based on Software Product Line) อ. ที่ปรึกษาวิทยานิพนธ์หลัก : ผศ.นครทิพย์ พร้อมพูล, 260 หน้า.

ซอฟต์แวร์สำเร็จรูปได้นำไปใช้ในการสนับสนุนการทำงานในหลายองค์กร เนื่องจากความยืดหยุ่นในการปรับให้เหมาะสมกับความต้องการของผู้ใช้ อย่างไรก็ตามในหลายองค์กรต้องมีการจัดสรรงบประมาณในการลงทุนเพื่อนำซอฟต์แวร์สำเร็จรูปมาใช้ เนื่องจากได้มีการเปลี่ยนคุณลักษณะของซอฟต์แวร์ให้เป็นไปตามความต้องการที่เปลี่ยนแปลงไปหรือเพื่อเพิ่มความสามารถของซอฟต์แวร์ ดังนั้นในหลายองค์กรจึงให้ความสนใจในการใช้ประโยชน์จากสิ่งที่มีอยู่นั้นให้สามารถสนับสนุนการใช้ซ้ำในการสร้างซอฟต์แวร์ให้เป็นไปตามความต้องการใหม่ๆ ขององค์กรซึ่งจะช่วยให้ประหยัดต้นทุนและเวลาในการพัฒนาได้

งานวิจัยนี้นำเสนอการสร้างแบบจำลองกระบวนการสำหรับการพัฒนาซอฟต์แวร์สำเร็จรูปจากสินทรัพย์ที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน เพื่อให้ได้มาซึ่งแบบจำลองกระบวนการ เอกสารและเครื่องมือสนับสนุน โดยกรอบงานประกอบด้วย 3 ชั้น คือ ชั้นแนวคิด ชั้นกระแสดำเนินการ และชั้นรายละเอียด ซึ่งองค์กรที่ต้องการสร้างผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ที่เป็นซอฟต์แวร์สำเร็จรูปนั้นสามารถนำกรอบงานและเครื่องมือที่นำเสนอนี้ไปประยุกต์ใช้ได้

ภาควิชา : วิศวกรรมคอมพิวเตอร์ลายมือชื่อนิติยุทธนันท์ กัณหาชน
สาขาวิชา : วิศวกรรมซอฟต์แวร์ลายมือชื่อ อ.ที่ปรึกษาวิทยานิพนธ์หลักพรทิพย์ พร้อมพูล
ปีการศึกษา : 2553

5070412821 : MAJOR SOFTWARE ENGINEERING

KEYWORDS : COMMERCIAL-OFF-THE-SHELF / COTS / COTS TAILORING / COTS
INTEGRATION / COTS MAINTENANCE / SOFTWARE PRODUCT LINE / SPL /
SOFTWARE PROCESS IMPROVEMENT

YUTTHANUN KUNHACHON : PROCESS MODEL DESIGN AND
DEVELOPMENT FOR COTS SOFTWARE DEVELOPMENT BASED ON
SOFTWARE PRODUCT LINE. THESIS ADVISOR : ASST.PROF. NAKORNTHIP
PROMPOON, 260 pp.

COTS (Commercial-off-the-shelf) software is widely used by many organizations because of its customization flexibility to comply with user requirements. However, those organizations must grant an investment in order to use new version of COTS software since the features of COTS frequently changes according to the change of requirements or software capability enhancement. Thus, many organizations concern with the utilization of existing components of COTS software to serve the reuse purpose in order to produce software that meets new organization requirements. This may help to reduce cost and time of software development.

This thesis proposes a process model design and development based on the concept of software product lines for COTS software development from assets in order to obtain an organization standard process model, templates of relevant documents and a supporting tool. The framework composed of three layers: conceptual layer, workflow layer and elaboration layer. It can be applied in any organization which wants to create new software from existing assets like COTS components collections.

Department : ..Computer Engineering.....

Field of Study : ..Software Engineering.....

Academic Year : ..2010.....

Student's Signature *Yutthanun Kunhachon*

Advisor's Signature *Nakornthip Prompon*

กิตติกรรมประกาศ

ข้าพเจ้าขอกราบขอบพระคุณพระคุณพระรัตนตรัยอันประเสริฐ มารดาบิดาผู้เลิศคุณที่ซึ่งเป็นที่ยึดเหนี่ยวจิตใจและผู้ให้ทุกสิ่งแก่ข้าพเจ้า บุรพจารย์ทุกท่านที่ได้อบรมสั่งสอนและมีอุปการะคุณแก่ข้าพเจ้ามาก่อนหน้า รวมถึงผู้ช่วยศาสตราจารย์นครทิพย์ พร้อมพล อาจารย์ที่ปรึกษาวิทยานิพนธ์ที่ได้ให้ความรู้ คำแนะนำและแนวทางในการดำเนินชีวิต

ขอกราบขอบพระคุณ รองศาสตราจารย์ ดร.วันชัย รวีไพบูลย์ ประธานกรรมการสอบวิทยานิพนธ์ รองศาสตราจารย์ ดร.พรศิริ หมั่นไชยศรี และ ดร.เฉลิมศักดิ์ เลิศวงศ์เสถียร ที่ได้ให้คำแนะนำในการแก้ไขวิทยานิพนธ์ฉบับนี้ให้มีคุณภาพยิ่งขึ้น และขอกราบขอบพระคุณ คณาจารย์ทุกท่านในภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ขอกราบขอบพระคุณบุคลากรทุกท่าน รวมถึงญาติ พี่น้องทุกท่านที่เป็นกำลังใจให้ข้าพเจ้าเสมอมา

ขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในห้องปฏิบัติการวิศวกรรมซอฟต์แวร์ คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัยทุกคนที่เป็นกำลังใจ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญตาราง.....	ญ
สารบัญภาพ.....	ผ
บทที่	
1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการวิจัย.....	3
1.3 ขอบเขตของการวิจัย.....	3
1.4 ขั้นตอนและวิธีการดำเนินการวิจัย.....	4
1.5 ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย.....	4
1.6 บทความวิชาการที่ได้รับการตีพิมพ์.....	5
2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	6
2.1 ทฤษฎีที่เกี่ยวข้อง.....	6
2.1.1 ซอฟต์แวร์สำเร็จรูป.....	6
2.1.2 วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์.....	8
2.1.3 วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์บนพื้นฐานยูเอ็มแอลหรือพลัส.....	10
2.1.4 มาตรฐานไอทีริฟเฟิลสำหรับเทคโนโลยีสารสนเทศ-กระบวนการ วิจัยชีวิตซอฟต์แวร์-กระบวนการใช้ซ้ำ.....	15
2.2 งานวิจัยที่เกี่ยวข้อง.....	18
2.2.1 COTS-Based Software Product Line Development.....	18
2.2.2 Value-Based Processes for COTS-Based Applications.....	21
2.2.3 Software Product Lines in Action The Best Industrial Practice in Product Line Engineering.....	22

บทที่	หน้า
2.2.4 An Active-Architecture to COTS Integration.....	22
3 กระบวนการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้ สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน.....	25
3.1 การออกแบบและพัฒนาแบบจำลองกระบวนการการพัฒนาซอฟต์แวร์ สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน.....	28
3.1.1 ศึกษาวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์.....	29
3.1.2 ศึกษากระบวนการ กิจกรรมและหน้าที่ของกลุ่มกระบวนการที่เกี่ยวข้อง ของมาตรฐานเพื่อการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์ สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน.....	29
3.1.3 ศึกษาทฤษฎี งานวิจัย บทเรียนรู้และสารสนเทศอื่นๆที่เกี่ยวข้อง.....	30
3.1.4 วิเคราะห์ ออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์ สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน.....	30
3.1.5 วิเคราะห์และออกแบบเอกสารคำอธิบายกระบวนการ.....	31
3.1.6 ทวนสอบกระบวนการและเอกสารคำอธิบายกระบวนการเทียบกับ กระบวนการ กิจกรรมและหน้าที่ของกลุ่มกระบวนการที่เกี่ยวข้อง.....	31
3.1.7 ปรับปรุงกระบวนการและเอกสารคำอธิบายกระบวนการตามผลการ ทวนสอบ.....	31
3.2 การออกแบบและพัฒนาซอฟต์แวร์ต้นแบบที่ใช้สนับสนุนกระบวนการ การพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน.....	32
4 กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน...	33
4.1 แบบจำลองกระบวนการขั้นแนวคิด.....	33
4.2 แบบจำลองกระบวนการขั้นกระแสวน.....	36
4.3 แบบจำลองกระบวนการขั้นรายละเอียด.....	42
4.4 การประเมินกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป.....	83
5 การวิเคราะห์และออกแบบเครื่องมือสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์ สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน.....	87
5.1 การวิเคราะห์ความต้องการของเครื่องมือสนับสนุน.....	87
5.2 การออกแบบหน้าที่การทำงานของเครื่องมือสนับสนุน.....	90

บทที่	ณ หน้า
5.3 การออกแบบสถาปัตยกรรมของเครื่องมือสับสนุน.....	97
5.4 การออกแบบส่วนต่อประสานผู้ใช้.....	100
5.5 การออกแบบฐานข้อมูลเชิงสัมพันธ์.....	105
5.6 การออกแบบความมั่นคงของเครื่องมือสับสนุน.....	105
6 การพัฒนาและประเมินเครื่องมือสับสนุน.....	107
6.1 เครื่องมือที่ใช้ในการพัฒนาเครื่องมือสับสนุน.....	107
6.2 ขั้นตอนในการพัฒนาเครื่องมือสับสนุน.....	109
6.3 กรณีทดสอบเครื่องมือสับสนุน.....	110
6.4 วิธีการประเมินเครื่องมือสับสนุน.....	115
7 สรุปผลการวิจัยและข้อเสนอแนะ.....	118
7.1 สรุปผลการวิจัย.....	118
7.2 ปัญหาและข้อจำกัดในการทำการวิจัย.....	120
7.3 ข้อเสนอแนะ.....	120
รายการอ้างอิง.....	122
ภาคผนวก.....	124
ภาคผนวก ก อภิธานศัพท์	125
ภาคผนวก ข คำย่อและรหัสพจน์.....	126
ภาคผนวก ค ตัวอย่างเอกสารสับสนุนกระบวนการการพัฒนาซอฟต์แวร์ สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน.....	127
ภาคผนวก ง ความต้องการด้านหน้าที่ของเครื่องมือสับสนุนกระบวนการ การพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน	230
ภาคผนวก จ โครงสร้างตารางข้อมูล.....	255
ประวัติผู้เขียนวิทยานิพนธ์.....	260

สารบัญญัตราสาร

ตารางที่		หน้า
2.1	อธิบายโครงสร้างสินทรัพย์ของแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ที่ได้ในแต่ละ ขั้นตอนตามกระบวนการพลัส.....	14
3.1	กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็น ฐานและประเภทของกลุ่มกระบวนการที่ใช้.....	30
3.2	รายละเอียดของคำอธิบายกระบวนการ.....	31
4.1	กระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา.....	45
4.2	กระบวนการกำหนดแผนการปรับแต่ง.....	46
4.3	กระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์.....	46
4.4	กระบวนการประเมินข้อกำหนดความต้องการ.....	47
4.5	กระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์.....	48
4.6	กระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์.....	50
4.7	กระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์.....	50
4.8	กระบวนการพัฒนาส่วนประกอบ.....	51
4.9	กระบวนการประเมินการพัฒนาส่วนประกอบ.....	52
4.10	กระบวนการบูรณาการกับระบบที่มีอยู่.....	53
4.11	กระบวนการประเมินการบูรณาการกับระบบที่มีอยู่.....	54
4.12	กระบวนการทดสอบสายผลิตภัณฑ์.....	55
4.13	กระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป.....	55
4.14	กระบวนการปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ ซอฟต์แวร์.....	56
4.15	กระบวนการวางแผนการติดตั้ง.....	57
4.16	กระบวนการประเมินแผนการติดตั้ง.....	57
4.17	กระบวนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์.....	58
4.18	กระบวนการจัดการการสนับสนุนการฝึกอบรม.....	59
4.19	กระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา.....	60
4.20	กระบวนการกำหนดแผนการบูรณาการ.....	61

ตารางที่	หน้า
4.21	กระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์..... 61
4.22	กระบวนการประเมินข้อกำหนดความต้องการ..... 62
4.23	กระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์..... 63
4.24	กระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์..... 64
4.25	กระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์..... 65
4.26	กระบวนการพัฒนาส่วนประกอบ..... 66
4.27	กระบวนการประเมินการพัฒนาส่วนประกอบ..... 67
4.28	กระบวนการบูรณาการระบบและซอฟต์แวร์..... 68
4.29	กระบวนการประเมินการบูรณาการระบบและซอฟต์แวร์..... 69
4.30	กระบวนการทดสอบสายผลิตภัณฑ์..... 69
4.31	กระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป..... 70
4.32	กระบวนการปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ ซอฟต์แวร์..... 71
4.33	กระบวนการวางแผนการติดตั้ง..... 72
4.34	กระบวนการประเมินแผนการติดตั้ง..... 72
4.35	กระบวนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์..... 73
4.36	กระบวนการจัดการการสนับสนุนการฝึกอบรม..... 74
4.37	กระบวนการกำหนดแผนการบำรุงรักษา..... 74
4.38	กระบวนการสร้างกระบวนการบำรุงรักษา..... 75
4.39	กระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์..... 76
4.40	กระบวนการวิเคราะห์ผลกระทบสินทรัพย์ที่สามารถนำมาใช้ซ้ำ..... 77
4.41	กระบวนการกำหนดวิธีการบำรุงรักษา..... 77
4.42	กระบวนการวางแผนการดัดแปลง ย้ายหรือยกเลิกซอฟต์แวร์..... 78
4.43	กระบวนการดัดแปลง ย้ายหรือยกเลิกซอฟต์แวร์..... 79
4.44	กระบวนการแจ้งผลการบำรุงรักษา..... 79
4.45	กระบวนการสร้างจุดมุ่งหมายของการวัดผล..... 80
4.46	กระบวนการวางแผนกระบวนการวัดผล..... 81
4.47	กระบวนการวัดผลการพัฒนาซอฟต์แวร์..... 82

ตารางที่		หน้า
4.48	กระบวนการวิเคราะห์และจัดเก็บผลิตภัณฑ์.....	82
4.49	กระบวนการรายงานการวัดผลและแนวทางสำหรับการปรับปรุงกระบวนการ....	83
4.50	ตารางรายละเอียดของกิจกรรมที่ผู้วิจัยได้เพิ่มเติมจากการศึกษานอกเหนือจาก มาตรฐานไอทีพีเฟลอี 1517.....	84
4.51	ตารางเปรียบเทียบระหว่างกิจกรรมที่ผู้วิจัยนำเสนอกับกลุ่มกระบวนการการ พัฒนาผลิตภัณฑ์จากสินทรัพย์ (กระบวนการการพัฒนา).....	84
4.52	ตารางเปรียบเทียบระหว่างกิจกรรมที่ผู้วิจัยนำเสนอกับกลุ่มกระบวนการการ พัฒนาผลิตภัณฑ์จากสินทรัพย์ (กระบวนการบำรุงรักษา).....	86
5.1	ความต้องการที่ไม่ใช่หน้าที่ระบบ.....	90
5.2	คำอธิบายแผนภาพคลาสของเครื่องมือสนับสนุน.....	94
6.1	ตัวอย่างกรณีทดสอบการเพิ่มข้อมูลผู้ใช้งานระบบเข้าสู่ระบบ.....	111
6.2	ข้อมูลทดสอบของกรณีทดสอบที่ T0104 (กรณีปกติ).....	112
6.3	ข้อมูลทดสอบของกรณีทดสอบที่ T0104 (กรณีผิดพลาด).....	113
6.4	ตารางเปรียบเทียบระหว่างหน้าที่การทำงานของเครื่องมือสนับสนุนกับ ข้อกำหนดของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีเฟลอี 1517 กลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์(กระบวนการการพัฒนา)..	115
6.5	ตารางเปรียบเทียบระหว่างหน้าที่การทำงานของเครื่องมือสนับสนุนกับ ข้อกำหนดของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีเฟลอี 1517 กลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์(กระบวนการบำรุงรักษา)..	116
6.6	หน้าที่การทำงานอื่นๆ ของเครื่องมือสนับสนุน.....	117
ค.1	รายชื่อเอกสารสนับสนุนประเภทนิยามกระบวนการ.....	127
ค.2	รายชื่อเอกสารสนับสนุนประเภทแบบฟอร์ม.....	127
ค.3	รายชื่อเอกสารสนับสนุนประเภทรายการตรวจสอบ.....	129
ค.4	รายชื่อเอกสารสนับสนุนประเภทแนวทางในการดำเนินการ.....	129
ง.1	ความต้องการด้านหน้าที่ของระบบ รหัส FR0101.....	230
ง.2	ความต้องการด้านหน้าที่ของระบบ รหัส FR0102.....	230
ง.3	ความต้องการด้านหน้าที่ของระบบ รหัส FR0103.....	231
ง.4	ความต้องการด้านหน้าที่ของระบบ รหัส FR0104.....	231

ตารางที่	หน้า
ง.33	ความต้องการด้านหน้าที่ของระบบ รหัส FR0133..... 245
ง.34	ความต้องการด้านหน้าที่ของระบบ รหัส FR0134..... 246
ง.35	ความต้องการด้านหน้าที่ของระบบ รหัส FR0135..... 246
ง.36	ความต้องการด้านหน้าที่ของระบบ รหัส FR0136..... 247
ง.37	ความต้องการด้านหน้าที่ของระบบ รหัส FR0137..... 247
ง.38	ความต้องการด้านหน้าที่ของระบบ รหัส FR0138..... 248
ง.39	ความต้องการด้านหน้าที่ของระบบ รหัส FR0139..... 248
ง.40	ความต้องการด้านหน้าที่ของระบบ รหัส FR0140..... 249
ง.41	ความต้องการด้านหน้าที่ของระบบ รหัส FR0141..... 249
ง.42	ความต้องการด้านหน้าที่ของระบบ รหัส FR0142..... 250
ง.43	ความต้องการด้านหน้าที่ของระบบ รหัส FR0143..... 250
ง.44	ความต้องการด้านหน้าที่ของระบบ รหัส FR0144..... 251
ง.45	ความต้องการด้านหน้าที่ของระบบ รหัส FR0145..... 251
ง.46	ความต้องการด้านหน้าที่ของระบบ รหัส FR0146..... 252
ง.47	ความต้องการด้านหน้าที่ของระบบ รหัส FR0147..... 252
ง.48	ความต้องการด้านหน้าที่ของระบบ รหัส FR0148..... 253
ง.49	ความต้องการด้านหน้าที่ของระบบ รหัส FR0149..... 253
ง.50	ความต้องการด้านหน้าที่ของระบบ รหัส FR0150..... 254
ง.51	ความต้องการด้านหน้าที่ของระบบ รหัส FR0151..... 254
จ.1	สรุปตารางของเครื่องมือสนับสนุน..... 255
จ.2	โครงสร้างตารางข้อมูลของข้อมูลเอกสารสนับสนุนกระบวนการ..... 256
จ.3	โครงสร้างตารางข้อมูลของข้อมูลผู้ใช้งานระบบ..... 257
จ.4	โครงสร้างตารางข้อมูลของข้อมูลเอกสารผลลัพธ์การดำเนินการของผู้จัดการโครงการ... 257
จ.5	โครงสร้างตารางข้อมูลของข้อมูลเอกสารผลลัพธ์การดำเนินการของทีมผู้พัฒนา..... 258
จ.6	โครงสร้างตารางข้อมูลของข้อมูลเอกสารผลลัพธ์การดำเนินการของทีมผู้ประเมิน.... 258
จ.7	โครงสร้างตารางข้อมูลของข้อมูลการประเมินเกณฑ์การใช้ซ้ำของรายการ ตรวจสอบ..... 259
จ.8	โครงสร้างตารางข้อมูลของข้อมูลผลผลิตงาน..... 259

สารบัญภาพ

ภาพที่		หน้า
2.1	ตัวอย่างรายละเอียดของซอฟต์แวร์สำเร็จรูป.....	7
2.2	การพัฒนาสินทรัพย์หลัก.....	9
2.3	การพัฒนาผลิตภัณฑ์จำเพาะ.....	9
2.4	กระบวนการวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์แบบวิวัฒนาการ.....	12
2.5	กระบวนการวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์แบบวิวัฒนาการตามกระบวนการพลัส.....	12
2.6	โครงสร้างสินทรัพย์ของแหล่งเก็บสายผลิตภัณฑ์.....	14
2.7	โครงสร้างสินทรัพย์ของกระบวนการแบบจำลองความต้องการสายผลิตภัณฑ์ตามกระบวนการพลัส.....	15
2.8	โครงสร้างของมาตรฐานไอทีพีพีเอส 1517 ภายในกรอบงานของมาตรฐานไอทีพีพีเอส 12207 และกรอบงานสำหรับกระบวนการวัฏจักรชีวิตใช้ซ้ำ.....	16
2.9	แบบจำลองวาง.....	19
2.10	สถาปัตยกรรมของแบบจำลองวาง.....	20
2.11	กรอบงานของกระบวนการตัดสินใจสำหรับแอปพลิเคชันที่ใช้ซอฟต์แวร์สำเร็จรูปเป็นฐานที่ใช้มูลค่าเป็นฐาน.....	21
2.12	วิวัฒนาการร่วมกันของระบบธุรกิจและสภาพแวดล้อมของเทคโนโลยีสารสนเทศที่ใช้ในการบูรณาการซอฟต์แวร์สำเร็จรูป.....	23
2.13	การบูรณาการซอฟต์แวร์สำเร็จรูปกับสถาปัตยกรรมอาร์ชแวร์.....	24
3.1	กระบวนการออกแบบและพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน.....	25
3.2	การออกแบบและพัฒนาแบบจำลองกระบวนการการพัฒนาซอฟต์แวร์ที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน.....	29
3.3	การออกแบบและพัฒนาซอฟต์แวร์ต้นแบบที่ใช้สนับสนุนกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน.....	32
4.1	แบบจำลองกระบวนการขั้นแนวคิด.....	33
4.2	กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปขั้นกระแสวน.....	37

ภาพที่	หน้า
4.3	กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปชั้นกระแสนงาน..... 39
4.4	กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูปชั้นกระแสนงาน..... 41
4.5	กระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูปชั้นกระแสนงาน..... 42
4.6	การสร้างคำอธิบายกระบวนการการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา..... 43
5.1	แผนภาพยูสเคสแสดงหน้าที่การทำงานของเครื่องมือสนับสนุน..... 92
5.2	แผนภาพคลาสของเครื่องมือสนับสนุน..... 93
5.3	แผนภาพสถาปัตยกรรมระบบ..... 98
5.4	แผนภาพสถาปัตยกรรมซอฟต์แวร์..... 99
5.5	โครงสร้างส่วนต่อประสานกับผู้ใช้..... 100
5.6	หน้าจอแสดงผลการทำงานในกรณีปกติ..... 101
5.7	หน้าจอแสดงผลการทำงานในกรณีผิดพลาด..... 102
5.8	โครงสร้างส่วนต่อประสานกับผู้ใช้ส่วนการออกรายงานการบันทึกผล..... 103
5.9	โครงสร้างส่วนต่อประสานกับผู้ใช้ส่วนการออกรายงานการประเมิน..... 104
5.10	โครงสร้างฐานข้อมูลเชิงสัมพันธ์..... 105
5.11	หน้าจอแสดงผลการทำงานเมื่อผู้ใช้งานระบบกรอกรหัสผู้ใช้งานและรหัสผ่านในกรณีปกติ..... 106
5.12	หน้าจอแสดงผลการทำงานเมื่อผู้ใช้งานระบบกรอกรหัสผู้ใช้งานและรหัสผ่านในกรณีผิดพลาด..... 106
6.1	หน้าจอแสดงผลการบันทึกข้อมูลในกรณีปกติ..... 114
6.2	หน้าจอแสดงผลการบันทึกข้อมูลในผิดพลาด..... 114
จ.1	โครงสร้างฐานข้อมูลเชิงสัมพันธ์..... 256

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ซอฟต์แวร์สำเร็จรูปเป็นซอฟต์แวร์เชิงพาณิชย์ที่ถูกสร้างขึ้นเพื่อตอบสนองต่อความต้องการของท้องตลาด ซึ่งเป็นการพัฒนาซอฟต์แวร์เพื่อตอบสนองต่อความต้องการของผู้ใช้โดยทั่วไป [1-2] ซึ่งซอฟต์แวร์สำเร็จรูปสามารถนำไปปรับใช้กับงานที่เกี่ยวข้องได้อย่างกว้างขวาง โดยองค์กรที่นำเอาซอฟต์แวร์สำเร็จรูปไปใช้นั้นจะต้องเข้าใจลักษณะของซอฟต์แวร์สำเร็จรูปที่มีการเปลี่ยนแปลงอยู่ตลอดเวลาเป็นอย่างดีเสียก่อน [3-5] เนื่องจากการพัฒนาซอฟต์แวร์ที่เป็นซอฟต์แวร์สำเร็จรูปนั้นขึ้นอยู่กับผู้ผลิตที่สามารถเปลี่ยนแปลงคุณลักษณะของซอฟต์แวร์ได้ทุกเมื่อและอาจส่งผลกระทบต่อผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปขององค์กรที่ใช้อยู่ เช่น ปัญหาด้านการบำรุงรักษา ซึ่งผู้ผลิตสามารถกำหนดระยะเวลาและรุ่นของซอฟต์แวร์ในการบำรุงรักษาได้ รวมถึงปัญหาด้านความปลอดภัยของซอฟต์แวร์ เนื่องจากมีความสามารถที่หลากหลายและผลิตมาเพื่อผู้ใช้งานโดยทั่วไป ความสามารถด้านความปลอดภัยจึงลดลงซึ่งอาจเป็นเป้าหมายของการโจมตีได้ง่าย เป็นต้น แต่อย่างไรก็ตามซอฟต์แวร์สำเร็จรูปยังได้รับความสนใจและถูกนำไปใช้โดยหลายองค์กรเนื่องจากช่วยลดเวลาในการพัฒนาซอฟต์แวร์และง่ายต่อการบำรุงรักษาแต่เนื่องจากการเปลี่ยนแปลงอยู่ตลอดเวลาของคุณลักษณะและรุ่นของซอฟต์แวร์สำเร็จรูป จึงทำให้องค์กรที่นำเอาซอฟต์แวร์สำเร็จรูปไปใช้เหล่านั้นจำเป็นต้องทำการปรับเปลี่ยนเป็นซอฟต์แวร์สำเร็จรูปรุ่นใหม่อยู่บ่อยครั้ง โดยที่ซอฟต์แวร์สำเร็จรูปรุ่นเก่าที่ยังสามารถใช้งานได้ถูกเก็บไว้เป็นสินทรัพย์ที่ไม่ได้ใช้ให้เกิดประโยชน์ ซึ่งการนำเอาสินทรัพย์ขององค์กรมาใช้ซ้ำในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์นั้น นอกจากจะเป็นการใช้สินทรัพย์ให้เกิดประโยชน์แล้วยังสามารถลดต้นทุนและเวลาในการพัฒนาซอฟต์แวร์ได้อีกด้วย ดังนั้นผู้วิจัยจึงได้นำหลักการของสายผลิตภัณฑ์ซอฟต์แวร์ (Software Product Lines) มาใช้สำหรับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ซึ่งสายผลิตภัณฑ์ซอฟต์แวร์เป็นกลยุทธ์หนึ่งที่น่าเสนอแนวคิดในการใช้ซ้ำ (Reuse) จากสินทรัพย์ขององค์กรโดยการระบุคุณสมบัติร่วม (Commonality) และส่วนแปรผัน (Variability) เพื่อสร้างคุณลักษณะที่ตอบสนองต่อความต้องการซอฟต์แวร์ขององค์กร [6-8] ข้อดีของการนำสายผลิตภัณฑ์ซอฟต์แวร์มาใช้ คือ ลดต้นทุน ลดเวลาในการพัฒนาและได้ซอฟต์แวร์ที่มีคุณภาพเนื่องจากพัฒนาจากสินทรัพย์ที่ผ่านการทดสอบแล้ว [5]

กระบวนการที่ทำให้ได้มาซึ่งซอฟต์แวร์สำเร็จรูปที่ตรงกับความต้องการซอฟต์แวร์ขององค์กรโดยรวมแล้วมีทั้งหมด 4 กระบวนการ [5, 9-12] ดังนี้

1) การประเมิน (Assessment) เป็นกระบวนการในการประเมินและคัดเลือกผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปจากท้องตลาดเพื่อนำมาใช้กับองค์กร

2) การปรับแต่ง (Tailoring) เป็นกระบวนการในการปรับแต่งซอฟต์แวร์สำเร็จรูปที่ถูกคัดเลือกให้นำมาใช้กับองค์กรให้มีความเหมาะสมและสามารถทำงานร่วมกันได้กับสภาพแวดล้อมของแต่ละองค์กร

3) การบูรณาการ (Integration) เป็นกระบวนการในการออกแบบและพัฒนารหัสคำสั่งเพื่อเชื่อมต่อซอฟต์แวร์สำเร็จรูปเข้าด้วยกันซึ่งเป็นการเพิ่มเติมความสามารถของซอฟต์แวร์ให้สามารถตอบสนองและครอบคลุมต่อความต้องการซอฟต์แวร์ขององค์กรได้

4) การบำรุงรักษา (Maintenance) เป็นกระบวนการในการบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูป ซึ่งทำหน้าที่ในการจัดการกับปัญหาหรือความเสี่ยงที่จะเกิดขึ้นสำหรับผลิตภัณฑ์ซอฟต์แวร์ [13] เช่น ปรับปรุงแก้ไข ย้าย หรือยกเลิกซอฟต์แวร์สำเร็จรูป เป็นต้น

ดังนั้นงานวิจัยนี้จึงได้นำเสนอกรอบงานสำหรับการออกแบบและพัฒนาแบบจำลองกระบวนการเพื่อพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ โดยมุ่งเน้นที่กระบวนการปรับแต่ง การบูรณาการและการบำรุงรักษาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน ซึ่งเป็นการนำเอาซอฟต์แวร์สำเร็จรูปที่เป็นสินทรัพย์ขององค์กรที่มีอยู่ในแหล่งเก็บสายผลิตภัณฑ์ (Software Product Lines Repository) ทั้งที่เป็นซอฟต์แวร์สำเร็จรูปและสินทรัพย์อื่นๆ นั้นมาใช้ซ้ำรวมถึงเพื่อพัฒนาเครื่องมือและเอกสารสนับสนุนสำหรับในแต่ละกระบวนการเพื่อให้สามารถตอบสนองต่อความต้องการซอฟต์แวร์ขององค์กรและนอกจากนั้นเพื่อให้ได้กระบวนการสำหรับการพัฒนาซอฟต์แวร์สำเร็จรูปที่เป็นวิธีปฏิบัติอย่างเป็นระบบ (Systematic Practice) สำหรับการใช้ซ้ำของซอฟต์แวร์ (Software Reuse) งานวิจัยนี้จึงได้นำเอามาตรฐานไอทีพีเฟลอีสำหรับเทคโนโลยีสารสนเทศ-กระบวนการวัฏจักรชีวิตซอฟต์แวร์-กระบวนการใช้ซ้ำ (มาตรฐานไอทีพีเฟลอี 1517) [14] มาประยุกต์ใช้ในการออกแบบและพัฒนากระบวนการสำหรับการพัฒนาซอฟต์แวร์สำเร็จรูปจากสินทรัพย์ดังกล่าวข้างต้นอีกด้วย

1.2 วัตถุประสงค์ของการวิจัย

- 1) เพื่อวิเคราะห์ ออกแบบและพัฒนาระบบการสำหรับการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน ซึ่งประกอบด้วย กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปและกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป
- 2) เพื่อออกแบบและพัฒนาเครื่องมือสนับสนุนแบบจำลองกระบวนการที่ได้วิเคราะห์ ออกแบบและพัฒนาขึ้นตาม ข้อ 1

1.3 ขอบเขตของการวิจัย

- 1) กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานอ้างอิงแบบจำลองเพื่อเป็นแนวทางในการออกแบบกระบวนการสำหรับพัฒนาและบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปจากสายผลิตภัณฑ์
- 2) กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานได้ทำการวิเคราะห์และออกแบบกระบวนการดังต่อไปนี้ กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปและกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูปตามมาตรฐานไอทีพีเฟลลีย์ 1517
- 3) จัดทำเอกสารเพื่อเป็นแนวทางในการดำเนินงานสำหรับกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน
- 4) เครื่องมือที่ได้มีความสอดคล้องกับกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน ซึ่งประกอบด้วยหน้าที่การทำงานดังต่อไปนี้
 - (1) สนับสนุนกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป
 - (2) สนับสนุนกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป
 - (3) สนับสนุนกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป
- 5) ประเมินเครื่องมือโดยใช้รายการตรวจสอบในการเปรียบเทียบหน้าที่การทำงานของเครื่องมือกับกระบวนการตามมาตรฐานไอทีพีเฟลลีย์ 1517

1.4 ขั้นตอนและวิธีการดำเนินการวิจัย

- 1) ศึกษารายละเอียดของสายผลิตภัณฑ์ซอฟต์แวร์ที่ใช้เป็นฐานในการออกแบบและพัฒนาผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูป
- 2) ศึกษากระบวนการและขั้นตอนในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูป
- 3) ศึกษากระบวนการ กิจกรรม หน้าที่และแนวทางในการวิเคราะห์และออกแบบกระบวนการจากกลุ่มกระบวนการของมาตรฐานไอทีพีพีแอลอี 1517 ที่ซึ่งเกี่ยวข้องกับการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน
- 4) วิเคราะห์และออกแบบกระบวนการในการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน
- 5) วิเคราะห์และออกแบบเอกสารตามกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานที่ได้จากข้อ 4
- 6) วิเคราะห์และออกแบบหน้าที่การทำงานของเครื่องมือที่เป็นต้นแบบในการสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานโดยใช้แนวความคิดเชิงวัตถุ
- 7) พัฒนาส่วนต่อประสานและส่วนประกอบภายในระบบ
- 8) สรุปผลการวิจัยและจัดทำรายงานวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย

- 1) ได้กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานที่สร้างกระบวนการตามมาตรฐานไอทีพีพีแอลอี 1517 ซึ่งสามารถนำไปประยุกต์ใช้งานต่อไปได้
- 2) ได้เครื่องมือที่เป็นต้นแบบในการสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

1.6 บทความวิชาการที่ได้รับการตีพิมพ์

ในการวิจัยนี้ ผู้วิจัยมีผลงานวิชาการร่วมกับคณะผู้วิจัย ซึ่งเป็นบทความวิชาการระดับนานาชาติและระดับชาติ รวม 2 บทความ ได้แก่

1) บทความวิชาการเรื่อง “การออกแบบและพัฒนาแบบจำลองกระบวนการปรับแต่งซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์” ซึ่งได้รับการคัดเลือกเพื่อนำเสนอและตีพิมพ์ในงาน “การประชุมวิชาการร่วมระดับนานาชาติด้านวิทยาการคอมพิวเตอร์และวิศวกรรมซอฟต์แวร์ ครั้งที่ 7 (The 7th International Joint Conference on Computer Science and Software Engineering : JCSSE 2010)” ระหว่างวันที่ 12-14 พฤษภาคม 2553 ณ มหาวิทยาลัยรามคำแหง กรุงเทพฯ ประเทศไทย

2) บทความวิชาการเรื่อง “การนิยามแบบจำลองกระบวนการและเครื่องมือสนับสนุนสำหรับการบูรณาการซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์” ซึ่งได้รับการคัดเลือกเพื่อนำเสนอและตีพิมพ์ในงาน “การประชุมวิชาการระดับชาติด้านวิทยาการคอมพิวเตอร์และวิศวกรรมคอมพิวเตอร์ ครั้งที่ 14 (The 14th National Computer Science and Engineering Conference : NCSEC 2010)” ระหว่างวันที่ 17-19 พฤศจิกายน 2553 ณ โรงแรมเซ็นทารา ดวงตะวัน เชียงใหม่ ประเทศไทย

ศูนย์วิทยพัชกร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 ซอฟต์แวร์สำเร็จรูป (Commercial-off-the-Shelf Software: COTS Software)

ซอฟต์แวร์สำเร็จรูป [1-2] คือ ผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปที่ถูกสร้างขึ้นเพื่อแจกจ่ายหรือเพื่อใช้ในเชิงพาณิชย์ซึ่งตอบสนองต่อความต้องการของท้องตลาดและผู้ใช้โดยทั่วไป เป็นซอฟต์แวร์ที่มีความเสถียร มีความสามารถที่หลากหลาย มีเอกสารการใช้งาน และถูกระบุความสามารถและข้อจำกัดไว้อย่างชัดเจน

ปัจจุบันการพัฒนาผลิตภัณฑ์ซอฟต์แวร์เพื่อนำมาใช้กับองค์กรนั้นใช้เวลาในการพัฒนาและต้นทุนค่อนข้างมากเนื่องจากการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ขึ้นใหม่ทั้งหมดหลายองค์กรจึงได้เลือกซอฟต์แวร์สำเร็จรูปมาใช้เนื่องจากสามารถนำมาปรับใช้กับสภาพแวดล้อมขององค์กรเพื่อตอบสนองต่อความต้องการได้อย่างรวดเร็ว แต่อย่างไรก็ตามซอฟต์แวร์สำเร็จรูปนั้นมีการเปลี่ยนแปลงรุ่นและคุณลักษณะอยู่ตลอดเวลาซึ่งอาจทำให้เกิดปัญหา เช่น การบำรุงรักษาซึ่งผู้ผลิตสามารถกำหนดระยะเวลาและรุ่นของซอฟต์แวร์ในการบำรุงรักษาได้ นอกจากนั้นผู้ผลิตสามารถเปลี่ยนแปลงความสามารถของซอฟต์แวร์ได้โดยที่ไม่ต้องแจ้งให้ผู้ใช้ทราบอีกด้วย รวมถึงปัญหาด้านความปลอดภัยของซอฟต์แวร์ เนื่องจากมีความสามารถที่หลากหลายและผลิตมาเพื่อผู้ใช้งานโดยทั่วไป ความสามารถด้านความปลอดภัยจึงลดลงซึ่งอาจเป็นเป้าหมายของการโจมตีได้ง่าย เป็นต้น โดยตัวอย่างรายละเอียดของซอฟต์แวร์สำเร็จรูปแสดงได้ดังรูปที่ 2.1

กระบวนการที่ทำให้ได้มาซึ่งซอฟต์แวร์สำเร็จรูปที่ตอบสนองต่อความต้องการซอฟต์แวร์ขององค์กร ประกอบด้วย 4 ขั้นตอน [5, 9-12] ดังนี้

1) การเลือกและการประเมินซอฟต์แวร์สำเร็จรูป (COTS assessment and selection)

การเลือกและการประเมินความสามารถของซอฟต์แวร์สำเร็จรูปเป็นกระบวนการในการประเมินและเลือกซอฟต์แวร์สำเร็จรูปในท้องตลาดมาใช้กับองค์กรโดยประเมินตามข้อกำหนดความต้องการซอฟต์แวร์ นโยบายขององค์กร หรือเป้าหมายขององค์กร เป็นต้น

2) การปรับแต่งส่วนประกอบซอฟต์แวร์สำเร็จรูป (COTS component tailoring)

การปรับแต่งส่วนประกอบซอฟต์แวร์สำเร็จรูปเป็นกระบวนการปรับแต่งซอฟต์แวร์สำเร็จรูปที่ได้ผ่านกระบวนการเลือกและประเมินซอฟต์แวร์สำเร็จรูปแล้วให้สามารถทำงานร่วมกับสภาพแวดล้อมหรือระบบเดิมที่มีอยู่ขององค์กรได้ ยกตัวอย่างเช่น การระบุรายละเอียดของฐานข้อมูล การระบุรายละเอียดของเครื่องแม่ข่าย หรือระบุนิพิตคลาส (Class path) ของคอมไพเลอร์ที่ใช้ เป็นต้น

COTS category	COTS name	Vendor	COTS assigned code	Description of selected COTS
Business web site	- ASP-Template - HTML-Template - PHP-Template	- HyperTemplates - Boxed Art - Webmasters Plaza	- BW-1 - BW-2 - BW-3	- ASP business web site template - HTML business web site template - PHP business web site template
Payment processing	- PayPal - WorldPay	- eBay - WorldPay	- PP-1 - PP-2	- Payment processing COTS for online secure transactions - Payment processing COTS for online secure transactions
Store management	- UStore - ASPVendor	- Superfreaker Studios - CJW Soft	- SM-1 - SM-2	- ASP-based Microsoft Access content management system - ASP-based Microsoft SQL Server product management system
Shopping cart	- CactuShop - SquirrelCart	- CactuSoft - Lighthouse Development	- SC-1 - SC-2	- ASP-based shopping cart tool to support shopping and order processing - ASP-based shopping cart tool to support shopping and order processing

รูปที่ 2.1 ตัวอย่างรายละเอียดของซอฟต์แวร์สำเร็จรูป [1]

3) การบูรณาการซอฟต์แวร์สำเร็จรูป (COTS integration)

การบูรณาการซอฟต์แวร์สำเร็จรูปเป็นกระบวนการในการออกแบบและพัฒนารหัสคำสั่งเพิ่มเติม เพื่อนำมาบูรณาการกับซอฟต์แวร์สำเร็จรูปให้สามารถครอบคลุมข้อกำหนดความต้องการซอฟต์แวร์ขององค์กร กระบวนการบูรณาการนั้นจะเกิดขึ้นก็ต่อเมื่อมีการเลือกและประเมินซอฟต์แวร์สำเร็จรูป รวมถึงการปรับแต่งซอฟต์แวร์สำเร็จรูปแล้วไม่สามารถทำได้มาซึ่งซอฟต์แวร์ที่สามารถตอบสนองต่อความต้องการขององค์กรได้

4) การบำรุงรักษาซอฟต์แวร์สำเร็จรูป (COTS Maintenance)

การบำรุงรักษาซอฟต์แวร์สำเร็จรูปเป็นกระบวนการในการบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูป ซึ่งทำหน้าที่ในการจัดการกับปัญหาหรือความเสี่ยงที่จะเกิดขึ้นสำหรับผลิตภัณฑ์ซอฟต์แวร์ [13] เช่น ปรับปรุงแก้ไข ย้าย หรือยกเลิกซอฟต์แวร์สำเร็จรูป เป็นต้น

2.1.2 วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Software Product Line Engineering: SPLE)

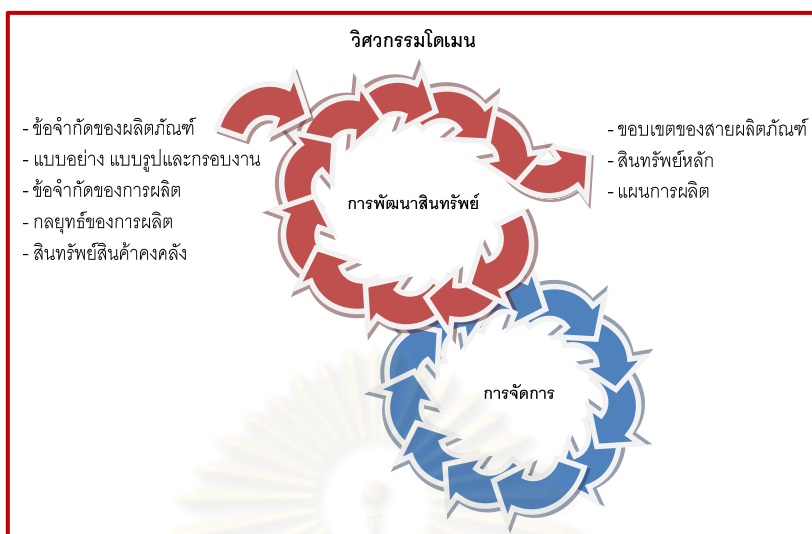
Clements และ Northrop [6] ให้นิยามสายผลิตภัณฑ์ซอฟต์แวร์ไว้ว่า “สายผลิตภัณฑ์ซอฟต์แวร์เป็นเซตของระบบซอฟต์แวร์ที่ใช้คุณสมบัติร่วมกัน จัดการเซตของลักษณะที่ตรงกับความต้องการของตลาดหรือภาระหน้าที่ และพัฒนาซอฟต์แวร์จากเซตของคุณสมบัติร่วมของสินทรัพย์หลักเหล่านั้น” วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์เป็นแนวคิดที่นำเสนอวิธีการนำเอาสินทรัพย์ที่มีอยู่ขององค์กรมาใช้ซ้ำโดยสายผลิตภัณฑ์ซอฟต์แวร์ประกอบด้วยกลุ่มของระบบที่มีคุณสมบัติร่วมและส่วนแปรผันซึ่งเป็นสินทรัพย์ที่สามารถนำมาใช้ซ้ำเพื่อตอบสนองต่อความต้องการซอฟต์แวร์

การพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์โดยใช้สายผลิตภัณฑ์ซอฟต์แวร์ประกอบด้วย 2 ขั้นตอนหลัก [6-8] ดังนี้

1) วิศวกรรมโดเมน (Domain Engineering)

วิศวกรรมโดเมนทำหน้าที่พัฒนาสินทรัพย์หลัก (Core assets) กระบวนการนี้ถูกเรียกว่า การพัฒนาสำหรับการใช้ซ้ำ (Development for reuse) ซึ่งเป็นพื้นฐานของสายผลิตภัณฑ์ซอฟต์แวร์ สินทรัพย์ที่อยู่ในกระบวนการนี้ เช่น สถาปัตยกรรม แบบจำลองโดเมน ข้อกำหนดความต้องการซอฟต์แวร์และกรณีทดสอบ [8] เป็นต้น

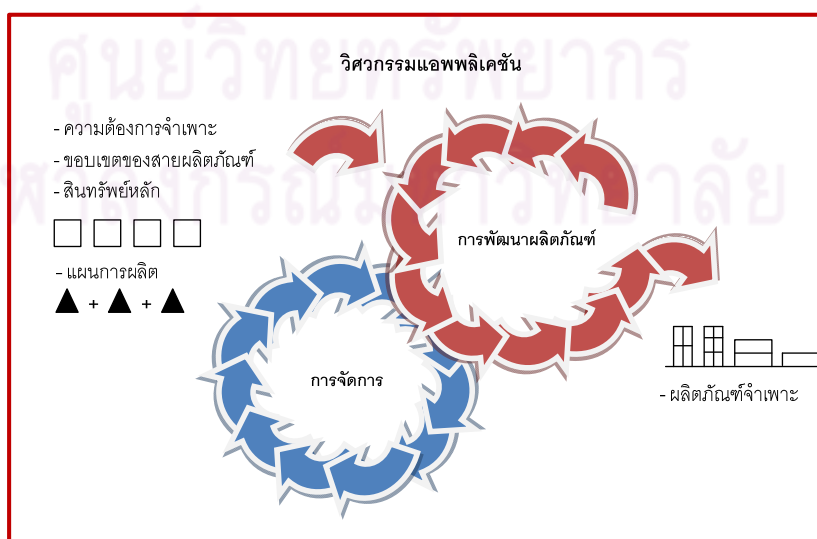
กิจกรรมของการพัฒนาสินทรัพย์นั้นประกอบด้วยข้อมูลนำเข้า คือ ข้อจำกัดของการผลิต แบบอย่าง แบบรูปและกรอบงาน ข้อจำกัดของผลิตภัณฑ์ กลยุทธ์ของการผลิตและสินทรัพย์สินค้าคงคลัง (Inventory of pre-existing assets) ส่วนข้อมูลนำออก คือ ขอบเขตของสายผลิตภัณฑ์ แผนการผลิตและสินทรัพย์หลัก ดังรูปที่ 2.2 ซึ่งข้อมูลนำออกทั้ง 3 ส่วนนี้จะนำไปใช้ในการผลิตในขั้นตอนของการสร้างผลิตภัณฑ์จำเพาะ (Specific product) ต่อไป



รูปที่ 2.2 การพัฒนาสินทรัพย์หลัก [8]

2) วิศวกรรมแอปพลิเคชัน (Application Engineering)

วิศวกรรมแอปพลิเคชันทำหน้าที่ในการพัฒนาผลิตภัณฑ์จำเพาะ กระบวนการนี้ถูกเรียกว่ากระบวนการพัฒนาโดยใช้ซ้ำ (Development with reuse) ซึ่งสินทรัพย์หลักที่นำมาใช้ซ้ำ ได้จากข้อมูลนำออกของขั้นตอนวิศวกรรมโดเมน ซึ่งในการพัฒนาแอปพลิเคชันจำเพาะนั้นนอกจากจะขึ้นอยู่กับข้อมูลนำออกของขั้นตอนวิศวกรรมโดเมนซึ่งเป็นข้อมูลนำเข้าของขั้นตอนวิศวกรรมแอปพลิเคชันแล้ว การพัฒนาแอปพลิเคชันจำเพาะจะขึ้นอยู่กับความต้องการจำเพาะของลูกค้าหรือองค์กรสำหรับผลิตภัณฑ์จำเพาะนั้น ซึ่งข้อมูลนำออกของขั้นตอนนี้ คือ ผลิตภัณฑ์จำเพาะ ที่ได้พัฒนาจากข้อมูลนำเข้าข้างต้น ดังรูปที่ 2.3



รูปที่ 2.3 การพัฒนาผลิตภัณฑ์จำเพาะ [8]

นอกจากนั้น Linda M. Northrop [8] ได้อธิบายว่า นอกจาก 2 ขั้นตอนของสายผลิตภัณฑ์ซอฟต์แวร์ข้างต้นแล้วกระบวนการที่สำคัญอีกกระบวนการหนึ่ง คือ การจัดการ (Management) สายผลิตภัณฑ์ซอฟต์แวร์ทั้งส่วนของการพัฒนาสินทรัพย์หลักและการพัฒนาผลิตภัณฑ์จำเพาะ การจัดการไม่ว่าจะเป็นการจัดการในระดับโครงการหรือการจัดการในระดับองค์กรเป็นสิ่งที่ต้องมีสำหรับสายผลิตภัณฑ์ซอฟต์แวร์ เพื่อความสำเร็จในการพัฒนาสายผลิตภัณฑ์

การจัดการในระดับโครงการ เป็นการตรวจสอบกิจกรรมการพัฒนาสินทรัพย์หลักและกิจกรรมการพัฒนาผลิตภัณฑ์จำเพาะเพื่อรับรองว่ากลุ่มของสินทรัพย์หลักที่สร้างขึ้นและผลิตภัณฑ์จำเพาะที่สร้างขึ้นนั้นเป็นกิจกรรมที่ต้องทำตามกระบวนการที่ได้ระบุไว้สำหรับสายผลิตภัณฑ์และเก็บรวบรวมข้อมูลที่เพียงพอสำหรับการติดตามความก้าวหน้าสำหรับการพัฒนาสายผลิตภัณฑ์ซอฟต์แวร์

การจัดการในระดับองค์กร จะต้องมีโครงสร้างขององค์กรที่เหมาะสมสำหรับองค์กรและรับรองได้ว่าองค์กรจะได้รับทรัพยากรที่ถูกต้องและมีจำนวนที่เพียงพอ การจัดการในระดับองค์กรช่วยกำหนดแบบจำลองกองทุน (Funding model) ที่ซึ่งใช้เพื่อเตรียมกองทุนให้สอดคล้องกับวิวัฒนาการของสินทรัพย์หลักซึ่งจะนำมาใช้สร้างกิจกรรมทางเทคนิคและกิจกรรมแบบวนซ้ำระหว่างการพัฒนาสินทรัพย์หลักและการพัฒนาผลิตภัณฑ์จำเพาะ นอกจากนั้นจะได้มาซึ่งเอกสารสำหรับการดำเนินการต่างๆ เพื่อการสื่อสารของสายผลิตภัณฑ์ซอฟต์แวร์ด้วย ซึ่งการจัดการนั้นสามารถลดความเสี่ยงขององค์กรที่คุกคามความสำเร็จของการพัฒนาสายผลิตภัณฑ์ซอฟต์แวร์และช่วยให้การพัฒนาผลิตภัณฑ์จากสายผลิตภัณฑ์ซอฟต์แวร์มีประสิทธิภาพยิ่งขึ้น

2.1.3 วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์บนพื้นฐานยูเอ็มแอลหรือพลัส (Product Line UML-Based Software Engineering: PLUS)

วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์บนพื้นฐานยูเอ็มแอลหรือพลัส [7] คือ การรวมเอากระบวนการและวิธีการเพื่อรองรับการพัฒนาสายผลิตภัณฑ์ซอฟต์แวร์โดยใช้ยูเอ็มแอลในการอธิบาย วิธีการพลัสนั้นได้นำเอายูเอ็มแอลซึ่งจากเดิมใช้วิธีสำหรับระบบเดี่ยว (Single systems) ซึ่งเป็นการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ขึ้นใหม่ทั้งหมด มาใช้กับสายผลิตภัณฑ์ซอฟต์แวร์ซึ่งเป็นวิธีระบบครอบครัว (Family systems) เนื่องจากยูเอ็มแอลช่วยให้สามารถเข้าใจและจัดการกับส่วนคุณสมบัติร่วมและส่วนแปรผันของสายผลิตภัณฑ์ซอฟต์แวร์จากทัศนมิติที่แตกต่างกันได้เป็นอย่างดีซึ่งแตกต่างจากการพัฒนาระบบโดยทั่วไป

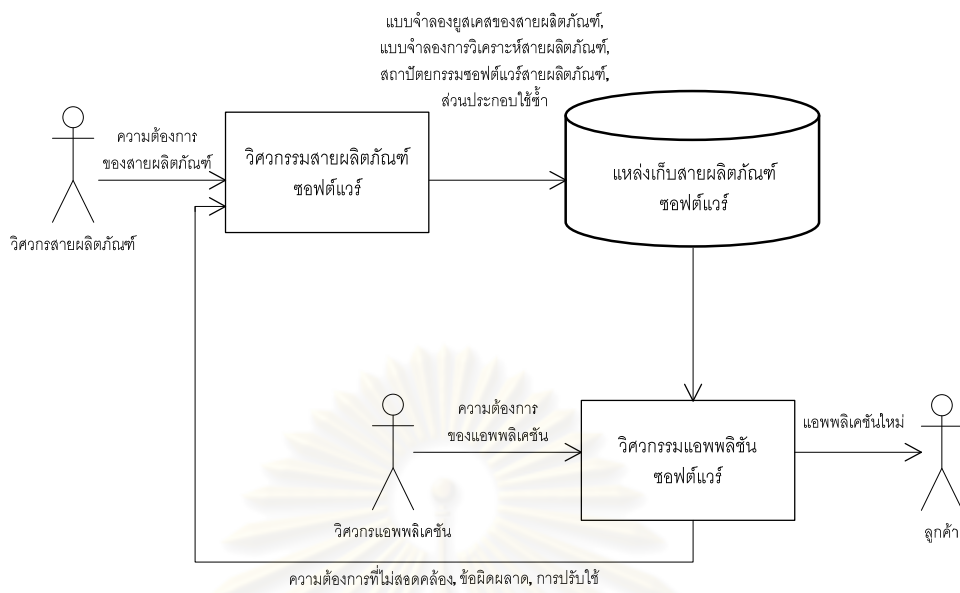
กระบวนการในการพัฒนาสำหรับวิธีการพลัสเป็นกระบวนการซอฟต์แวร์เชิงวัตถุแบบวนซ้ำซึ่งวิธีการของพลัสนั้นเริ่มตั้งแต่ขั้นตอนของการระบุความต้องการ การวิเคราะห์และการออกแบบสำหรับสายผลิตภัณฑ์ซอฟต์แวร์ วิธีการพลัสสามารถรวมเข้ากับกระบวนการวิศวกรรมซอฟต์แวร์ที่มีอยู่แล้วไม่ว่าจะเป็นกระบวนการเรชันนัลหรืออาร์ยูพี (Rational Unified Process: RUP) แบบจำลองสไปรัล (Spiral model) และกระบวนการวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์แบบวิวัฒนาการ (Evolutionary Software Product Line Engineering Process: ESPLEP) [7] เป็นต้น ซึ่งในส่วนของกระบวนการวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์แบบวิวัฒนาการหรืออีเอสพีเอลอีพีนี้เป็นแบบจำลองกระบวนการซอฟต์แวร์ที่ซึ่งกำจัดความแตกต่างระหว่างการพัฒนาและการบำรุงรักษาซึ่งเป็นการพัฒนาหลายรอบแบบวนซ้ำตามความต้องการที่เปลี่ยนแปลงในแต่ละรอบ ดังรูปที่ 2.4 จากเดิมการพัฒนาแบบเดี่ยวนั้นจะใช้เวลาพัฒนาซอฟต์แวร์ที่ใช้แบบจำลองเป็นฐาน (Model-based software development) ซึ่งเป็นกิจกรรมสำหรับระบบเดียว ดังนั้นเมื่อเปลี่ยนเป็น ระบบครอบครัวจึงนำเอาอีเอสพีเอลอีพีมาใช้ในการพัฒนาสายผลิตภัณฑ์ซอฟต์แวร์ซึ่งแบบจำลองอีเอสพีเอลอีพีนั้นประกอบไปด้วย 2 กระบวนการหลัก ดังนี้

1) วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Software Product Line Engineering)

เป็นกระบวนการในการวิเคราะห์และจำแนกส่วนที่ใช้คุณสมบัติร่วมและส่วนแปรผันของความต้องการทั้งหมดของสายผลิตภัณฑ์ที่มีความชัดเจน ขั้นตอนของการพัฒนาประกอบด้วย แบบจำลองยูสเคสของสายผลิตภัณฑ์ (Product Line Use Case Model) แบบจำลองการวิเคราะห์สายผลิตภัณฑ์ (Product Line Analysis Model) สถาปัตยกรรมซอฟต์แวร์สายผลิตภัณฑ์ (Product Line Software Architecture) และส่วนประกอบใช้ซ้ำ (Reusable Component) หลังจากนั้นจะทำการทดสอบและเก็บสิ่งที่ทำขึ้นในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Software Product Line Repository)

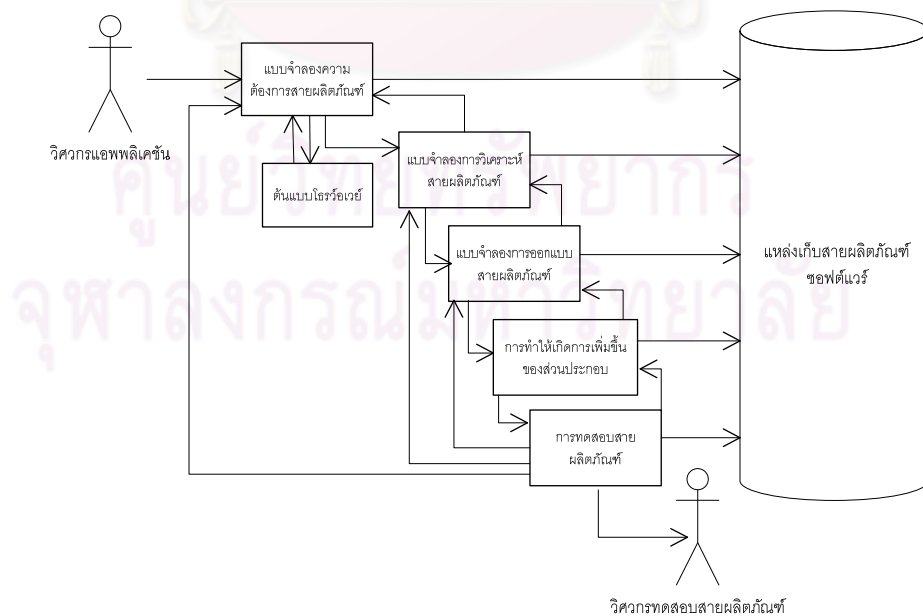
2) วิศวกรรมแอปพลิเคชันซอฟต์แวร์ (Software Application Engineering)

เป็นกระบวนการในการพัฒนาแอปพลิเคชันย่อยที่เป็นสมาชิกของสายผลิตภัณฑ์ซอฟต์แวร์ซึ่งพัฒนาจากสิ่งทำขึ้นที่เกิดจากการพัฒนาในระหว่างวัฏจักรชีวิตของวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ โดยความต้องการของแต่ละแอปพลิเคชันจำเพาะปรับและสืบทอดมาจากกิจกรรมวัฏจักรชีวิตของวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ข้างต้น โดยเลือกส่วนประกอบที่เหมาะสมมาใช้จากแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์มาพัฒนาเป็นแอปพลิเคชันใหม่



รูปที่ 2.4 กระบวนการวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์แบบวิวัฒนาการ [7]

ดังนั้นจึงกล่าวได้ว่า กระบวนการวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์แบบวิวัฒนาการหรืออีเอสพีเอลอีพีเมื่อนำมารวมกับวิธีการพลัส ซึ่งเป็นขั้นตอนของการระบุความต้องการการวิเคราะห์ และการออกแบบสำหรับสายผลิตภัณฑ์ซอฟต์แวร์โดยใช้ยูเอ็มแอลในการอธิบายตามที่ได้อธิบายไว้ข้างต้น จึงเกิดเป็นแบบจำลองกระบวนการที่เกิดจากการบูรณาการของพลัสและอีเอสพีเอลอีพี ดังรูปที่ 2.5



รูปที่ 2.5 กระบวนการวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์แบบวิวัฒนาการตามกระบวนการพลัส [7]

จากรูปที่ 2.5 สามารถอธิบายกระบวนการที่เกิดขึ้นในแต่ละขั้นตอนที่ซึ่งถูกเก็บลงในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ได้ ดังนี้

1) แบบจำลองความต้องการสายผลิตภัณฑ์ (Product Line Requirements Modeling)

สิ่งที่ถูกพัฒนาในขั้นตอนนี้ประกอบด้วยแบบจำลองยูสเคส (Use case model) และแบบจำลองลักษณะ (Feature model) ระบุถึงความต้องการเชิงฟังก์ชันของสายผลิตภัณฑ์ซอฟต์แวร์ในแง่ของผู้กระทำ (Actors) และยูสเคสซึ่งเพิ่มส่วนของคุณสมบัติและส่วนแปรผัน ซึ่งเป็นสิ่งจำเป็นที่จะต้องใช้ในการพัฒนาสายผลิตภัณฑ์ซอฟต์แวร์และถ้าความต้องการของแอปพลิเคชันยังไม่ชัดเจนสามารถใช้ต้นแบบโธรวอเวย์ (Throwaway Prototyping) เพื่อช่วยให้มีความชัดเจนของความต้องการมากขึ้น

2) แบบจำลองการวิเคราะห์สายผลิตภัณฑ์ (Product Line Analysis Modeling)

สิ่งที่ถูกพัฒนาในขั้นตอนนี้ประกอบด้วยแบบจำลองสถิต (Static model) และแบบจำลองพลวัต (Dynamic model) ซึ่งแบบจำลองสถิตอยู่ในรูปแบบของแผนภาพคลาส ทำหน้าที่ระบุความสัมพันธ์เชิงโครงสร้างซึ่งอธิบายถึงส่วนคุณสมบัติร่วมและส่วนแปรผันในระหว่างสมาชิกของสายผลิตภัณฑ์ และแบบจำลองพลวัตอยู่ในรูปแบบของแผนภาพลำดับทำหน้าที่แสดงปฏิสัมพันธ์ (Interact) ของยูสเคสจากแบบจำลองความต้องการ

3) แบบจำลองการออกแบบสายผลิตภัณฑ์ (Product Line Design Modeling)

สิ่งที่ถูกพัฒนาในขั้นตอนนี้คือ สถาปัตยกรรมซอฟต์แวร์ของสายผลิตภัณฑ์ซอฟต์แวร์ที่ใช้ส่วนประกอบพื้นฐาน ถูกออกแบบจากขั้นตอนของการวิเคราะห์ที่ซึ่งระบุถึงสถาปัตยกรรมเชิงโครงสร้างที่อยู่ในรูปแบบของแผนภาพคลาสและสถาปัตยกรรมเชิงพลวัตที่อยู่ในรูปแบบของแผนภาพลำดับในแง่ของการสื่อสารกันของแต่ละส่วนประกอบว่าจะทำได้อย่างไร

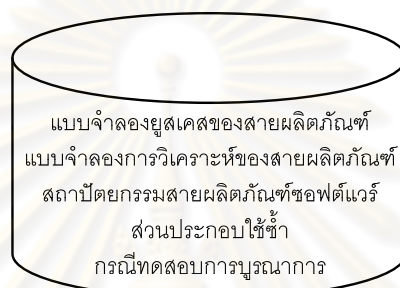
4) การทำให้เกิดการเพิ่มขึ้นของส่วนประกอบ (Incremental Component Implementation)

สิ่งที่ถูกพัฒนาในขั้นตอนนี้คือ เซตย่อยของสายผลิตภัณฑ์จะถูกเลือกนำมาพัฒนาในแต่ละรอบซึ่งในขั้นตอนนี้จะประกอบด้วย การออกแบบรายละเอียด (Detailed design) การให้รหัส (Coding) และการทดสอบส่วนย่อย (Unit testing)

5) การทดสอบสายผลิตภัณฑ์ (Product Line Testing)

ขั้นตอนของการทดสอบสายผลิตภัณฑ์นั้นประกอบด้วย การทดสอบแบบบูรณาการ (Integration testing) และการทดสอบเชิงหน้าที่ (Functional testing) ของสายผลิตภัณฑ์ซอฟต์แวร์

สามารถแสดงโครงสร้างสินทรัพย์ของแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ในกระบวนการสายผลิตภัณฑ์ซอฟต์แวร์แบบวิวัฒนาการตามกระบวนการพลัส ได้ดังรูปที่ 2.6



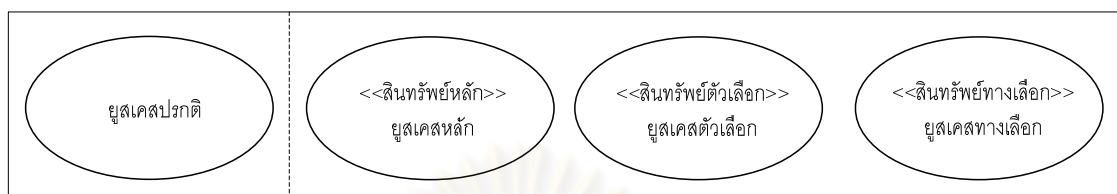
รูปที่ 2.6 โครงสร้างสินทรัพย์ของแหล่งเก็บสายผลิตภัณฑ์ [7]

จากรูปที่ 2.6 สามารถอธิบายในรายละเอียดของสินทรัพย์ที่ได้จากกระบวนการวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์แบบวิวัฒนาการตามกระบวนการพลัสในแต่ละขั้นตอนได้ดังตารางที่ 2.1

ตารางที่ 2.1 อธิบายโครงสร้างสินทรัพย์ของแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ที่ได้ในแต่ละขั้นตอนตามกระบวนการพลัส [7]

กระบวนการพลัส	สินทรัพย์ที่ได้	การนำไปใช้
แบบจำลองความต้องการสายผลิตภัณฑ์	แบบจำลองยูสเคส, แบบจำลองคุณลักษณะ	เพื่อใช้เป็นแบบจำลองยูสเคสของแอปพลิเคชัน
แบบจำลองการวิเคราะห์สายผลิตภัณฑ์	แบบจำลองสถิติ, แบบจำลองพลวัต	เพื่อใช้เป็นแบบจำลองการวิเคราะห์ของแอปพลิเคชัน
แบบจำลองการออกแบบสายผลิตภัณฑ์	สถาปัตยกรรมส่วนประกอบ	เพื่อใช้ในการสร้างสถาปัตยกรรมของแอปพลิเคชันซอฟต์แวร์
การทำให้เกิดการเพิ่มขึ้นของส่วนประกอบ	ส่วนประกอบใช้ซ้ำ	เพื่อใช้ส่วนประกอบที่เหมาะสมในการพัฒนาซอฟต์แวร์
การทดสอบสายผลิตภัณฑ์	กรณีทดสอบการบูรณาการ	เพื่อใช้ในการทดสอบสายผลิตภัณฑ์

นอกจากนั้นสามารถอธิบายในรายละเอียดของโครงสร้างสินทรัพย์ ในที่นี้ ยกตัวอย่างโครงสร้างสินทรัพย์ของกระบวนการแบบจำลองความต้องการของสายผลิตภัณฑ์ตาม กระบวนการพลัส ดังแสดงในรูปที่ 2.7



รูปที่ 2.7 โครงสร้างสินทรัพย์ของกระบวนการแบบจำลองความต้องการสายผลิตภัณฑ์ ตามกระบวนการพลัส [7]

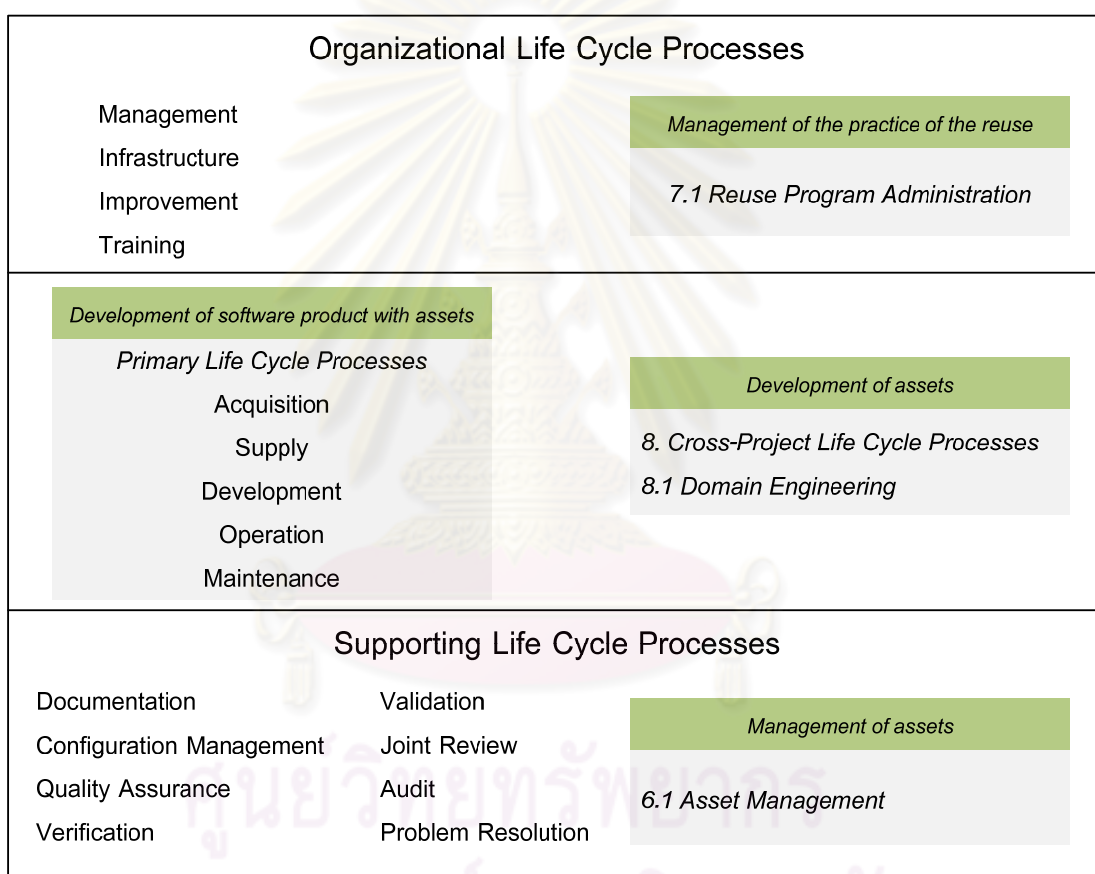
จากรูปที่ 2.7 สามารถอธิบายในรายละเอียดของโครงสร้างสินทรัพย์ของ กระบวนการแบบจำลองความต้องการสายผลิตภัณฑ์ตามกระบวนการพลัสซึ่งแตกต่างจากระบบ ทั่วไป โดยสามารถแบ่งยูสเคสได้เป็น 3 ประเภท ดังนี้

- 1) ยูสเคสหลัก (Kernel Use Case) คือ ยูสเคสที่ทุกสมาชิกที่อยู่ในสายผลิตภัณฑ์ ใช้ร่วมกัน
- 2) ยูสเคสตัวเลือก (Optional Use Case) คือ ยูสเคสที่หลายสมาชิกในสายผลิตภัณฑ์ ที่ใช้ร่วมกัน
- 3) ยูสเคสทางเลือก (Alternative Use Case) คือ ยูสเคสที่มีบางสมาชิกเท่านั้นใน สายผลิตภัณฑ์เลือกนำมาใช้งาน

2.1.4 มาตรฐานไอทีริเฟอิลอีสำหรับเทคโนโลยีสารสนเทศ-กระบวนการวัฏจักร ชีวิตซอฟต์แวร์-กระบวนการใช้ซ้ำ (IEEE Standard for Information Technology-Software Life Cycle Processes-Reuse Processes: IEEE 1517)

มาตรฐานไอทีริเฟอิลอี 1517 เป็นกรอบงานสำหรับกระบวนการวัฏจักรชีวิต ซอฟต์แวร์ที่เป็นวิธีปฏิบัติอย่างเป็นระบบสำหรับการใช้ซ้ำของซอฟต์แวร์โดยระบุกระบวนการ กิจกรรมและหน้าที่ที่ต้องใช้ในการสร้างผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ในแต่ละขั้นตอนที่เกิดขึ้น ในวัฏจักรชีวิตซอฟต์แวร์ซึ่งประกอบด้วย การระบุ (Identification) การสร้าง (Construction) การบำรุงรักษา (Maintenance) และการจัดการสินทรัพย์ (Management of assets)

มาตรฐานไอทีพีเฟิลอี 1517 เป็นมาตรฐานที่อธิบายขยายความเพิ่มเติมจากมาตรฐานไอทีพีเฟิลอี 12207 ซึ่งเป็นมาตรฐานเกี่ยวกับกระบวนการวัฏจักรชีวิตซอฟต์แวร์โดยมุ่งเน้นที่กระบวนการใช้ซ้ำแต่อธิบายในเฉพาะระดับอ้างอิง (Reference level) ซึ่งไม่ได้ระบุถึงรายละเอียดว่าต้องดำเนินการอย่างไรในแต่ละกระบวนการ โดยจุดประสงค์ของมาตรฐานนี้เพื่อสร้างกรอบงานสำหรับการใช้ซ้ำ ระบุกระบวนการ กิจกรรมและหน้าที่ที่จำเป็นต้องดำเนินการทั้งการใช้ซ้ำสำหรับการพัฒนาซอฟต์แวร์โครงการเดียว การพัฒนาซอฟต์แวร์หลายโครงการหรือการพัฒนาซอฟต์แวร์ระหว่างองค์กรซึ่งโครงสร้างของมาตรฐานไอทีพีเฟิลอี 1517 แสดงได้ดังรูปที่ 2.8



รูปที่ 2.8 โครงสร้างของมาตรฐานไอทีพีเฟิลอี 1517 ภายในกรอบงานของมาตรฐานไอทีพีเฟิลอี 12207 และกรอบงานสำหรับกระบวนการวัฏจักรชีวิตใช้ซ้ำ [14]

มาตรฐานไอทีพีเฟิลอี 1517 ได้ระบุกระบวนการ กิจกรรมและหน้าที่ของวัฏจักรชีวิตซอฟต์แวร์ที่ต้องใช้ในวิธีปฏิบัติการใช้ซ้ำ (Reuse practice) ซึ่งถูกแบ่งประเภทการใช้ซ้ำออกเป็น 4 ประเภท ดังนี้

1) การพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ (Development of software products with assets)

กิจกรรมและหน้าที่ในการใช้ซ้ำที่เกี่ยวข้อง คือ การพัฒนา (Development) การปฏิบัติการ (Operation) และการบำรุงรักษา (Maintenance) ผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ ซึ่งนอกจากสินทรัพย์แล้วข้อมูลนำเข้าของวิธีปฏิบัติการใช้ซ้ำนี้ยังรวมถึงแบบจำลองโดเมน (Domain models) และสถาปัตยกรรมโดเมน (Domain architecture) อีกด้วย

ในส่วนของ การพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์นี้ครอบคลุม ส่วนของวิศวกรรมแอปพลิเคชัน (Application engineering) ของสายผลิตภัณฑ์ซึ่งทำหน้าที่ในการพัฒนาผลิตภัณฑ์จำเพาะ (Specific product) จากสินทรัพย์ ดังนั้นงานวิจัยนี้จึงได้มุ่งเน้นที่กลุ่มกระบวนการนี้โดยนำมาใช้เป็นฐานในการวิเคราะห์ ออกแบบ และพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปซึ่งประกอบด้วยกระบวนการปรับแต่งซอฟต์แวร์สำเร็จรูป กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปและกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

2) การพัฒนาสินทรัพย์ (Development of assets)

กลุ่มของกิจกรรมและหน้าที่ของวิธีปฏิบัติการใช้ซ้ำนี้ คือ ส่วนของวิศวกรรมโดเมน (Domain engineering) ในสายผลิตภัณฑ์ ซึ่งเป็นส่วนของการพัฒนาสินทรัพย์ เพื่อนำไปใช้ซ้ำซึ่งสินทรัพย์ที่ได้จากขั้นตอนของวิศวกรรมโดเมนนั้นสามารถนำไปใช้ได้กับการพัฒนาซอฟต์แวร์หลายโครงการ รวมถึงการพัฒนาซอฟต์แวร์ระหว่างองค์กร ซึ่งกระบวนการหลักของวัฏจักรชีวิตในขั้นตอนนี้ประกอบด้วย การได้มา การจัดหา การพัฒนา การปฏิบัติการและการบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์

ในส่วนของ การพัฒนาสินทรัพย์นี้เป็นส่วนที่ครอบคลุมขั้นตอนวิศวกรรมโดเมนของสายผลิตภัณฑ์ซึ่งทำหน้าที่ในการพัฒนาสินทรัพย์หลักเพื่อนำเอาสินทรัพย์ไปใช้ซ้ำ แต่เนื่องจากงานวิจัยนี้ได้มุ่งเน้นในเฉพาะในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ ดังนั้นจึงไม่เกี่ยวข้องกับการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ในงานวิจัยนี้

3) การจัดการวิธีปฏิบัติในการใช้ซ้ำ (Management of the practice of reuse)

กลุ่มของกิจกรรมและหน้าที่ของวิธีปฏิบัติการใช้ซ้ำนี้ คือ กระบวนการการบริหารโปรแกรมใช้ซ้ำ (Reuse program administration process) ซึ่งเป็นกระบวนการที่ระบุถึงวัฏจักรชีวิตซอฟต์แวร์สำหรับการใช้ซ้ำในระดับองค์กร โดยตัวอย่างกระบวนการใช้ซ้ำในระดับองค์กร เช่น การจัดการ (Management) โครงสร้างพื้นฐาน (Infrastructure) การปรับปรุง (Improvement) และการฝึกอบรม (Training) เป็นต้น

ในส่วนของจัดการวิธีปฏิบัติในการใช้ซ้ำนี้เป็นกระบวนการวัฏจักรชีวิตซอฟต์แวร์สำหรับการใช้ซ้ำในระดับองค์กร ดังนั้นจึงไม่เกี่ยวข้องกับการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ในงานวิจัยนี้

4) การจัดการสินทรัพย์ (Management of assets)

กลุ่มของกิจกรรมและหน้าที่ของวิธีปฏิบัติการใช้ซ้ำนี้ คือ กระบวนการจัดการสินทรัพย์ ซึ่งเป็นกระบวนการที่ช่วยสนับสนุนกระบวนการอื่นทำให้ได้มาซึ่งซอฟต์แวร์ที่มีคุณภาพและประสบความสำเร็จ โดยกระบวนการสนับสนุนนั้นยกตัวอย่างเช่น การพัฒนาการบำรุงรักษาและขั้นตอนวิศวกรรมโดเมนในสายผลิตภัณฑ์ โดยระบุกิจกรรมที่จำเป็น เช่น การจัดการรับรองสินทรัพย์ (Manage asset certification) การจำแนก (Classification) การเก็บข้อมูล (Storage) การค้นคืน (Retrieval) การควบคุมเวอร์ชัน (Version control) และการควบคุมการเปลี่ยนแปลง (Change control) เป็นต้น

ในส่วนของจัดการสินทรัพย์นี้เป็นกระบวนการที่ครอบคลุมทุกส่วนของขั้นตอนสำหรับสายผลิตภัณฑ์ซอฟต์แวร์ไม่ว่าจะเป็นขั้นตอนวิศวกรรมโดเมน หรือขั้นตอนวิศวกรรมแอปพลิเคชัน แต่เนื่องจากงานวิจัยนี้ได้มุ่งเน้นในเฉพาะในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ ดังนั้นจึงไม่เกี่ยวข้องกับการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ในงานวิจัยนี้

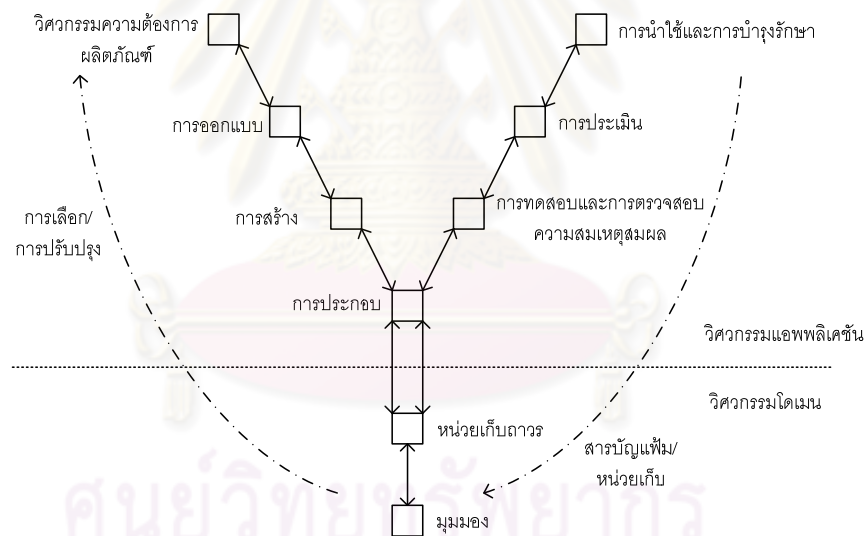
2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 การพัฒนาสายผลิตภัณฑ์ซอฟต์แวร์ที่ใช้ซอฟต์แวร์สำเร็จรูปเป็นฐาน (COTS-based software product line development โดย Luiz Fernando Capretz และคณะ)

งานวิจัยนี้ [1] ได้นำเสนอวิธีในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปโดยใช้สายผลิตภัณฑ์ซอฟต์แวร์ โดยมีจุดประสงค์เพื่อลดต้นทุน เวลาในการพัฒนาและนำหลักการของสายผลิตภัณฑ์ซอฟต์แวร์มาใช้ในการจำแนกส่วนคุณสมบัติร่วมและส่วนแปรผันของซอฟต์แวร์สำเร็จรูปเพื่อนำไปใช้ประโยชน์ในการใช้ซ้ำโดยได้เสนอแบบจำลองวาย (Y-model) เพื่อใช้ในการสร้างผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปจากสายผลิตภัณฑ์ซอฟต์แวร์ ดังรูปที่ 2.9

แบบจำลองวาย แบ่งออกเป็น 2 ระยะเวลาได้แก่

1) วิศวกรรมโดเมน (Domain Engineering) ซึ่งประกอบด้วย 2 มุมมอง คือ มุมมองของโครงสร้างสายผลิตภัณฑ์ซอฟต์แวร์ ทำหน้าที่สร้างโครงสร้างของสายผลิตภัณฑ์ซอฟต์แวร์และมุมมองของหน่วยเก็บถาวรซอฟต์แวร์สำเร็จรูป ทำหน้าที่เก็บซอฟต์แวร์สำเร็จรูปเพื่อใช้ในการสร้างผลิตภัณฑ์ซอฟต์แวร์

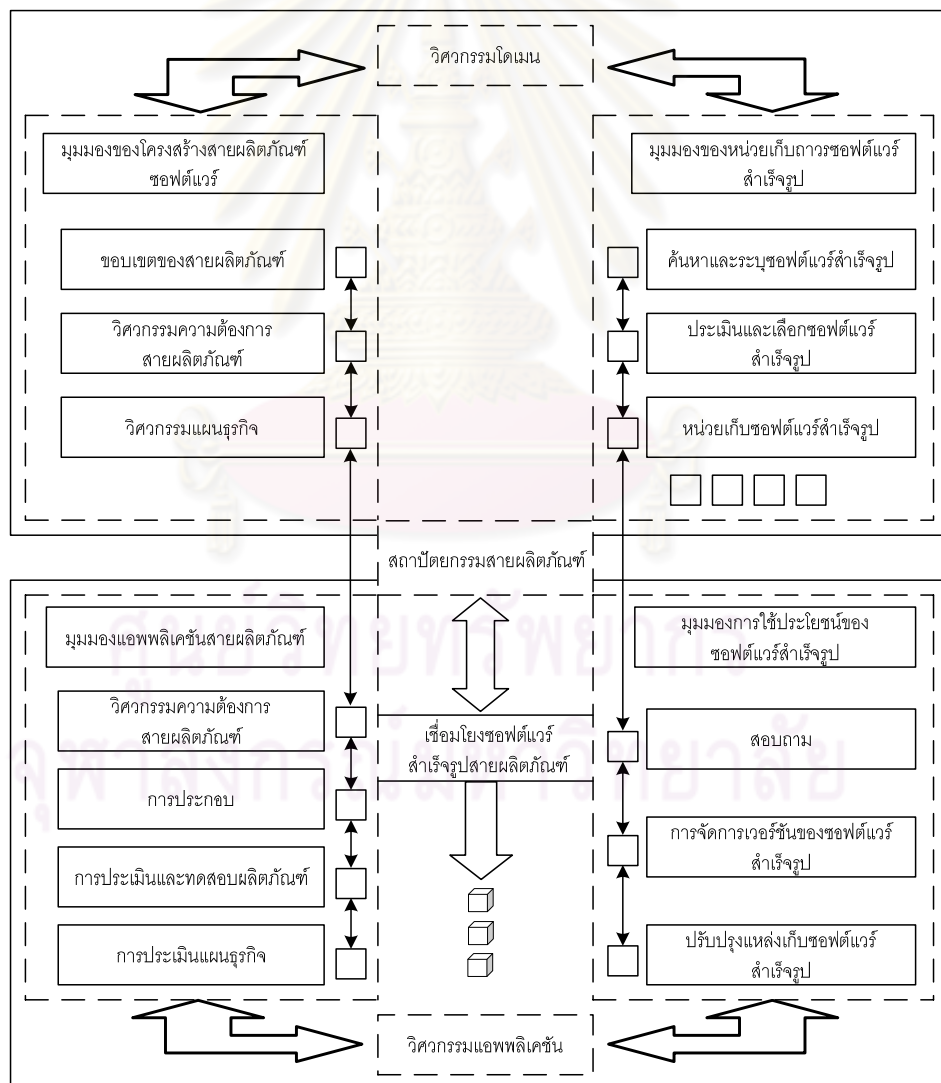


รูปที่ 2.9 แบบจำลองวาย [1]

2) วิศวกรรมแอปพลิเคชัน (Application Engineering) เป็นการพัฒนาผลิตภัณฑ์จำเพาะโดยมีอยู่ 2 มุมมอง คือ มุมมองแอปพลิเคชันสายผลิตภัณฑ์ ทำหน้าที่ระบุสมรรถนะที่อาจจะพัฒนาขึ้นได้จากแผนธุรกิจและมุมมองการใช้ประโยชน์ของซอฟต์แวร์สำเร็จรูป ทำหน้าที่เลือกใช้ซอฟต์แวร์สำเร็จรูปจากแหล่งเก็บมาใช้ในการพัฒนาผลิตภัณฑ์

ในแต่ละมุมมองของทั้งสองระยะจะมีกิจกรรมที่แตกต่างกันที่บอกถึงหน้าที่ของแต่ละมุมมองว่าต้องทำกิจกรรมอะไรบ้างซึ่งแต่ละกิจกรรมมีผลป้อนกลับซึ่งกันและกัน ดังแสดงในรูปที่ 2.10 เพื่อให้ได้ผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปที่ตรงกับความต้องการมากที่สุด

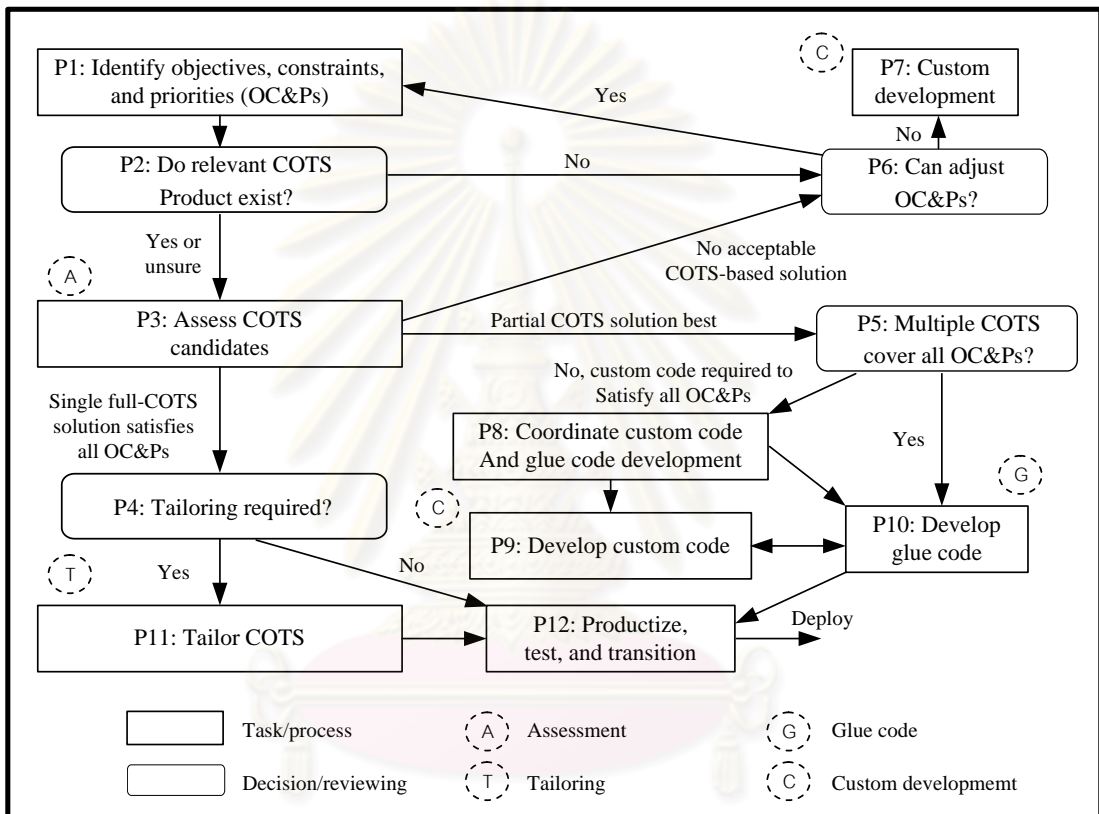
สิ่งที่นำมาพิจารณาใช้ในงานวิจัยนี้คือ สถาปัตยกรรมของแบบจำลองภายในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปจากสายผลิตภัณฑ์ซอฟต์แวร์ เพื่อใช้เป็นแนวทางในวิเคราะห์ออกแบบและพัฒนาแบบจำลองกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป การบูรณาการซอฟต์แวร์สำเร็จรูปและการบำรุงรักษาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์ตามกระบวนการ หน้าที่และกิจกรรมที่ได้กำหนดไว้ในมาตรฐานไอทีพีเฟลอี 1517 ซึ่งอธิบายในหัวข้อ 2.1.4 ข้างต้น



รูปที่ 2.10 สถาปัตยกรรมของแบบจำลองภายใน [15]

2.2.2 กระบวนการให้ค่าเป็นฐานสำหรับแอปพลิเคชันที่ใช้ซอฟต์แวร์สำเร็จรูปเป็นฐาน (Value-Based Processes for COTS-Based Applications โดย Ye Yang และคณะ)

งานวิจัยนี้ [10] ได้นำเสนอกรอบงานของกระบวนการตัดสินใจสำหรับแอปพลิเคชันที่ใช้ซอฟต์แวร์สำเร็จรูปเป็นฐาน (COTS-based Applications: CBAs) ดังรูปที่ 2.11 โดยมีจุดประสงค์เพื่อการได้มาซึ่งซอฟต์แวร์สำเร็จรูปที่เหมาะสมที่สุดสำหรับองค์กร



รูปที่ 2.11 กรอบงานของกระบวนการตัดสินใจสำหรับแอปพลิเคชันที่ใช้ซอฟต์แวร์สำเร็จรูปเป็นฐานที่ใช้มูลค่าเป็นฐาน [10]

กระบวนการตัดสินใจสำหรับแอปพลิเคชันที่ใช้ซอฟต์แวร์สำเร็จรูปเป็นฐานประกอบด้วย 3 กระบวนการหลัก ดังนี้

- 1) การประเมิน (Assessment) คือ การประเมินและเลือกผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปโดยเทียบกับความต้องการขององค์กร
- 2) การปรับแต่ง (Tailoring) คือ การปรับแต่งผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปเพื่อนำมาใช้กับความต้องการจำเพาะขององค์กร

3) การพัฒนารหัสเชื่อมต่อ (Glue-code Development) คือ การออกแบบการพัฒนาและการใช้รหัสคำสั่งเพื่อบูรณาการกับผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปให้สามารถทำงานร่วมกันได้เพื่อสนับสนุนความต้องการซอฟต์แวร์ผู้ใช้

สิ่งที่นำมาพิจารณาใช้ในงานวิจัยนี้คือ กรอบงานกระบวนการตัดสินใจสำหรับแอปพลิเคชันที่ใช้ซอฟต์แวร์สำเร็จรูปเป็นฐาน โดยใช้กระบวนการในการวิเคราะห์ ออกแบบและพัฒนาระบบการสำหรับการพัฒนาซอฟต์แวร์สำเร็จรูปในส่วนของกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปและกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป

2.2.3 สายผลิตภัณฑ์ซอฟต์แวร์ในการปฏิบัติการณ์ปฏิบัติทางอุตสาหกรรมที่ดีที่สุดสำหรับวิศวกรรมสายผลิตภัณฑ์ (Software Product Lines in Action The Best Industrial Practice in Product Line Engineering โดย Frank van der Lindenและคณะ)

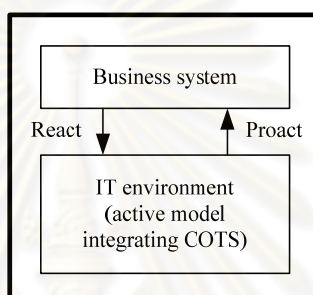
งานวิจัยนี้ [16] ได้นำเสนอการอธิบายโดยสรุปในเชิงปฏิบัติสำหรับการนำเอากระบวนการทางด้านวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์โดยใช้แบบจำลองเบไป (BAPO Model: Business, Architecture, Process and Organization) ไปใช้กับหลายองค์กรซึ่งใช้วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ในการพัฒนาผลิตภัณฑ์โดยมีขนาดและโดเมนที่แตกต่างกัน ยกตัวอย่างเช่น Bosch Gasoline System, Nokia Mobile Phones, Philips Medical System และ Siemen Medical Solutions เป็นต้น ซึ่งมีทั้งหมด 8 บริษัท 10 กรณีศึกษา โดยได้รวบรวมรายงานและประสบการณ์ที่ได้จากองค์กรเหล่านั้นซึ่งนอกจากการอธิบายถึงการพัฒนาระบบการในการสร้างผลิตภัณฑ์ซอฟต์แวร์ที่ใช้สายผลิตภัณฑ์แล้วยังได้อธิบายถึงปัญหาที่อาจเกิดขึ้นโดยองค์กรเหล่านั้นมีวิธีจัดการกับปัญหาเหล่านั้นอย่างไร ทำให้ได้มาซึ่งบทเรียนที่ได้จากอุตสาหกรรมจริงเกี่ยวกับสายผลิตภัณฑ์ซอฟต์แวร์

สิ่งที่นำมาพิจารณาใช้ในงานวิจัยนี้คือ บทเรียนที่ได้จากการนำเอากระบวนการที่ได้ออกแบบและพัฒนาไว้แล้ว นำไปใช้กับอุตสาหกรรมจริงซึ่งได้มาซึ่งข้อดีและข้อเสียรวมถึงความรู้ต่างๆ ซึ่งสามารถนำมาใช้กับงานวิจัยนี้ได้

2.2.4 แนวคิดสถาปัตยกรรมแบบแอ็กทีฟเพื่อบูรณาการซอฟต์แวร์สำเร็จรูป (An Active-Architecture Approach to COTS Integration โดย Brian Warboys และคณะ)

งานวิจัยนี้ [17] ได้นำเสนอกรอบงานอาร์ชแวร์ (ArchWare framework) เป็นแบบจำลองสถาปัตยกรรมแบบแอ็กทีฟซึ่งเป็นการวิวัฒนาการร่วมกันระหว่างระบบธุรกิจที่ไม่มี

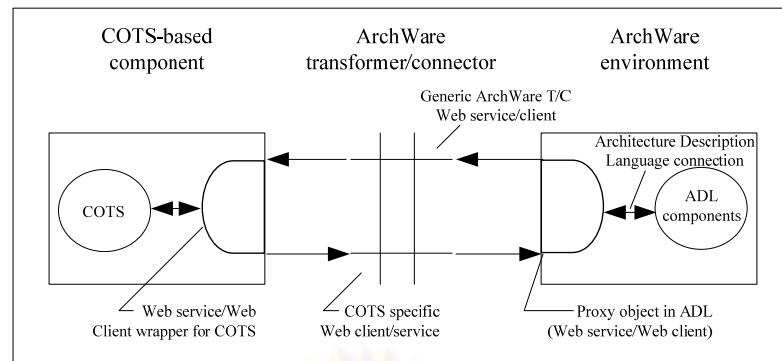
ความเสถียรกับสภาพแวดล้อมของเทคโนโลยีสารสนเทศที่ใช้ในการบูรณาการซอฟต์แวร์สำเร็จรูป ดังแสดงในรูปที่ 2.12 ซึ่งกรอบงานอาร์ชแวร์นี้จะช่วยในการจัดการกับซอฟต์แวร์สำเร็จรูป ไม่ว่าจะเป็นการรวมซอฟต์แวร์สำเร็จรูปใหม่หรือแทนที่โดยซอฟต์แวร์สำเร็จรูปที่มีอยู่แล้ว เพื่อสนองต่อความต้องการที่เปลี่ยนแปลงไปโดยแสดงโครงสร้างของการบูรณาการซอฟต์แวร์สำเร็จรูปกับสถาปัตยกรรมอาร์ชแวร์ได้ดังรูปที่ 2.13 โดยมีจุดประสงค์เพื่อการได้มาซึ่งระบบสารสนเทศที่มีความยืดหยุ่น สามารถนำเอาส่วนประกอบซอฟต์แวร์สำเร็จรูปในแต่ละบทบาทมาปรับใช้ให้ตรงกับสภาพแวดล้อมของระบบธุรกิจในขณะนั้นได้



รูปที่ 2.12 วิวัฒนาการร่วมกันของระบบธุรกิจและสภาพแวดล้อมของเทคโนโลยีสารสนเทศที่ใช้ในการบูรณาการซอฟต์แวร์สำเร็จรูป [17]

ระบบสารสนเทศที่ใช้อาร์ชแวร์ (ArchWare-based information system: AIS) ประกอบด้วยองค์ประกอบพื้นฐาน 4 ส่วน ดังนี้

- 1) ส่วนประกอบซอฟต์แวร์สำเร็จรูป (COTS Components) คือ ซอฟต์แวร์สำเร็จรูปที่ใช้ในการบูรณาการ
- 2) หม้อแปลง/ตัวเชื่อมต่ออาร์ชแวร์ (ArchWare Transformer/Connectors: TCs) คือ ส่วนที่ทำหน้าที่ห่อหุ้มบทบาทของส่วนประกอบซอฟต์แวร์สำเร็จรูปที่อยู่ในระบบ
- 3) ส่วนประกอบภาษาการพรรณาสถาปัตยกรรมอาร์ชแวร์ (ArchWare Architecture Description Language components) คือ ส่วนที่อธิบายรายละเอียดในการบูรณาการซอฟต์แวร์สำเร็จรูป
- 4) ผู้ใช้ (Users) คือ ผู้มีส่วนได้ส่วนเสียกับการพัฒนาระบบ



รูปที่ 2.13 การบูรณาการซอฟต์แวร์สำเร็จรูปกับสถาปัตยกรรมอาร์ชแวร์ [17]

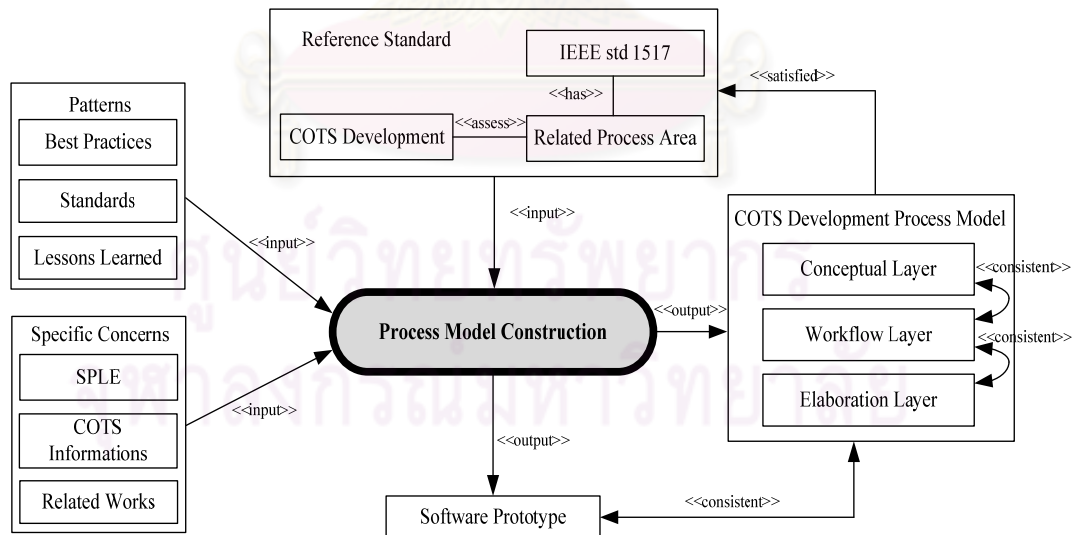
สิ่งที่นำมาพิจารณาใช้ในงานวิจัยนี้คือ กรอบงานอาร์ชแวร์เพื่อใช้ในการวิเคราะห์ และออกแบบการบูรณาการซอฟต์แวร์สำเร็จรูปที่ซึ่งช่วยในการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3

กระบวนการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์ สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

ในบทนี้จะกล่าวถึงกระบวนการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป ซึ่งประกอบด้วย 3 กระบวนการหลัก คือ กระบวนการปรับแต่งซอฟต์แวร์สำเร็จรูป กระบวนการบูรณาการซอฟต์แวร์สำเร็จรูปและกระบวนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป และนอกจากนั้นยังประกอบด้วย 1 กระบวนการสนับสนุน คือ กระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป โดยออกแบบและพัฒนากระบวนการตามมาตรฐานไอทีพีเฟลอี 1517 แบบจำลองวุฒิภาวะความสามารถแบบบูรณาการหรือซีเอ็มเอ็มไอ (Capability Maturity Model Integration: CMMI) [18] ในส่วนของกลุ่มกระบวนการวิเคราะห์และการวัดผล (Measurement and analysis) และวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน เพื่อให้ได้กระบวนการที่จะนำไปวิเคราะห์และออกแบบเครื่องมือสนับสนุนต่อไป สำหรับกระบวนการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานสามารถแสดงได้ดังรูปที่ 3.1



รูปที่ 3.1 กระบวนการออกแบบและพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

จากรูปที่ 3.1 แบบจำลองกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานและซอฟต์แวร์ต้นแบบซึ่งเป็นเครื่องมือสนับสนุนกระบวนการนั้นเป็นผลลัพธ์จากการออกแบบและพัฒนากระบวนการซึ่งประกอบด้วยข้อมูลนำเข้า 3 กลุ่มนำเข้าสามารถอธิบายในรายละเอียดได้ ดังต่อไปนี้

1) มาตรฐานที่ใช้อ้างอิง (Reference Standard)

มาตรฐานที่ใช้อ้างอิง คือ มาตรฐานไอทีพีพีอีสำหรับเทคโนโลยีสารสนเทศ-กระบวนการวิศวกรรมชีวิตซอฟต์แวร์-กระบวนการใช้ซ้ำ (IEEE 1517) ซึ่งใช้สำหรับการพัฒนาซอฟต์แวร์จากสินทรัพย์โดยนำมาใช้เป็นฐานในการออกแบบและพัฒนากระบวนการโดยแบ่งกลุ่มนำเข้าเป็น 2 ส่วน ดังนี้

1.1) กลุ่มกระบวนการที่เกี่ยวข้อง (Related Process Area)

กลุ่มกระบวนการที่เกี่ยวข้องเป็นกระบวนการที่อยู่ในมาตรฐานไอทีพีพีอี 1517 ซึ่งเกี่ยวข้องกับการใช้ซ้ำในการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน ซึ่งประกอบด้วยการปรับแต่ง การบูรณาการและการบำรุงรักษาซอฟต์แวร์สำเร็จรูปจากสินทรัพย์ โดยกลุ่มกระบวนการที่เกี่ยวข้องที่ใช้เป็นหลักในการออกแบบกระบวนการข้างต้นนั้นเป็นกลุ่มของกระบวนการที่อยู่ในประเภทการใช้ซ้ำ (Reuse category) คือ การพัฒนาซอฟต์แวร์จากสินทรัพย์ (Development of software product with assets) ดังที่ได้กล่าวในรายละเอียดไว้ในบทที่ 2 โดยกลุ่มกระบวนการที่อยู่ในประเภทการใช้ซ้ำดังกล่าวเมื่อเทียบกับขั้นตอนของวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์แล้วจะอยู่ในขั้นตอนวิศวกรรมแอปพลิเคชัน (Application Engineering) ซึ่งเป็นการพัฒนาผลิตภัณฑ์เฉพาะจากสินทรัพย์

1.2) การพัฒนาซอฟต์แวร์สำเร็จรูป (COTS Development)

การพัฒนาซอฟต์แวร์สำเร็จรูป คือ เป้าหมายสำหรับการพัฒนากระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป และกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป โดยกระบวนการที่ได้นี้จะถูกประเมิน (Assess) เทียบกับกลุ่มกระบวนการที่เกี่ยวข้องและนอกจากนั้นเมื่อได้กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานแล้วผู้พัฒนาจะนำไปทวนสอบ (Verify) กับกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ได้ว่ามีความสอดคล้องกันหรือไม่

2) แบบรูป (Patterns)

แบบรูปเป็นส่วนของความรู้ที่ได้จากการศึกษาเพื่อใช้เป็นตัวนำร่องในการออกแบบและพัฒนากระบวนการ โดยแบ่งกลุ่มนำเข้าเป็น 3 ส่วน ดังนี้

2.1) วิธีปฏิบัติที่ดีที่สุด (Best Practices)

วิธีปฏิบัติที่ดีที่สุดที่นำมาใช้ในการออกแบบและพัฒนากระบวนการ คือ กลุ่มกระบวนการการวิเคราะห์และการวัดผลของแบบจำลองคุณภาพความสามารถแบบบูรณาการโดยนำมาใช้เพื่อวัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน โดยมีจุดประสงค์เพื่อหาจุดบกพร่องที่เกิดขึ้นและพัฒนากระบวนการให้มีประสิทธิภาพยิ่งขึ้นในการพัฒนาครั้งต่อไป

2.2) มาตรฐาน (Standards)

มาตรฐานในส่วนนี้เป็นมาตรฐานอื่นๆ ที่เกี่ยวข้องกับการออกแบบและพัฒนากระบวนการ คือ มาตรฐานไอทีพีพีเอส 12207-2008 [19] กระบวนการที่นำมาใช้ คือ สนับสนุนการยอมรับซอฟต์แวร์ (Software Acceptance Support) นำมาใช้เพื่อสนับสนุนการนำเอาผลิตภัณฑ์ซอฟต์แวร์ไปใช้งานได้อย่างมีประสิทธิภาพซึ่งประกอบด้วย การส่งมอบผลิตภัณฑ์ซอฟต์แวร์ การฝึกฝนและการจัดทำเอกสารการใช้งานซอฟต์แวร์สำหรับผู้

2.3) บทเรียนรู้ (Lesson Learned)

บทเรียนรู้เป็นประสบการณ์ที่ได้จากการออกแบบและพัฒนากระบวนการหรือประสบการณ์ต่างๆ ที่ได้จากนักวิจัยหลายๆ ท่าน เช่น [3-5, 16] ซึ่งสามารถนำมาประยุกต์ใช้กับงานวิจัยนี้ได้ โดยเฉพาะในส่วนของสายผลิตภัณฑ์ซอฟต์แวร์และซอฟต์แวร์สำเร็จรูปซึ่งบทเรียนรู้ที่ได้เหล่านั้นเกิดจากการนำเอาสิ่งที่ได้พัฒนาขึ้นไปใช้กับสภาพแวดล้อมจริงหรือได้จากศึกษาจากสภาพแวดล้อมจริงทำให้ได้มาซึ่งความรู้ที่สามารถนำมาใช้เป็นแนวทางในการออกแบบและพัฒนากระบวนการให้มีความสมบูรณ์แบบมากยิ่งขึ้น

3) ส่วนเกี่ยวข้องจำเพาะ (Specific concerns)

ส่วนเกี่ยวข้องจำเพาะเป็นสาระสำคัญที่จำเป็นต้องใช้ในการออกแบบและพัฒนากระบวนการ โดยแบ่งกลุ่มนำเข้าเป็น 3 ส่วน ดังนี้

3.1) วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Software Product Lines Engineering: SPLE)

กระบวนการที่ได้ออกแบบและพัฒนานั้นใช้วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานซึ่งประกอบด้วย 2 ขั้นตอนหลัก คือ วิศวกรรมโดเมนและวิศวกรรมแอปพลิเคชัน ซึ่งได้อธิบายในรายละเอียดไว้ในบทที่ 2 โดยการออกแบบและพัฒนากระบวนการโดยใช้วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์นั้นเป็นตัวบ่งชี้ถึงความแตกต่างกันอย่างชัดเจนระหว่างการพัฒนาผลิตภัณฑ์ซอฟต์แวร์โดยวิธีระบบเดี่ยว (Single system) ซึ่งเป็นการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ขึ้นใหม่ทั้งหมดกับวิธีระบบครอบครัว (Family system) ซึ่งเป็นการพัฒนาผลิตภัณฑ์ซอฟต์แวร์โดยนำเอาคุณลักษณะที่มีอยู่ในสายผลิตภัณฑ์มาใช้ซ้ำซึ่งลดต้นทุนและเวลาในการพัฒนาซอฟต์แวร์

3.2) สารสนเทศซอฟต์แวร์สำเร็จรูป (COTS Informations)

สารสนเทศซอฟต์แวร์สำเร็จรูปเป็นเนื้อหาสาระที่เกี่ยวข้องกับซอฟต์แวร์สำเร็จรูปที่ถูกนำมาใช้ในการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

3.3) งานวิจัยที่เกี่ยวข้อง (Related Work)

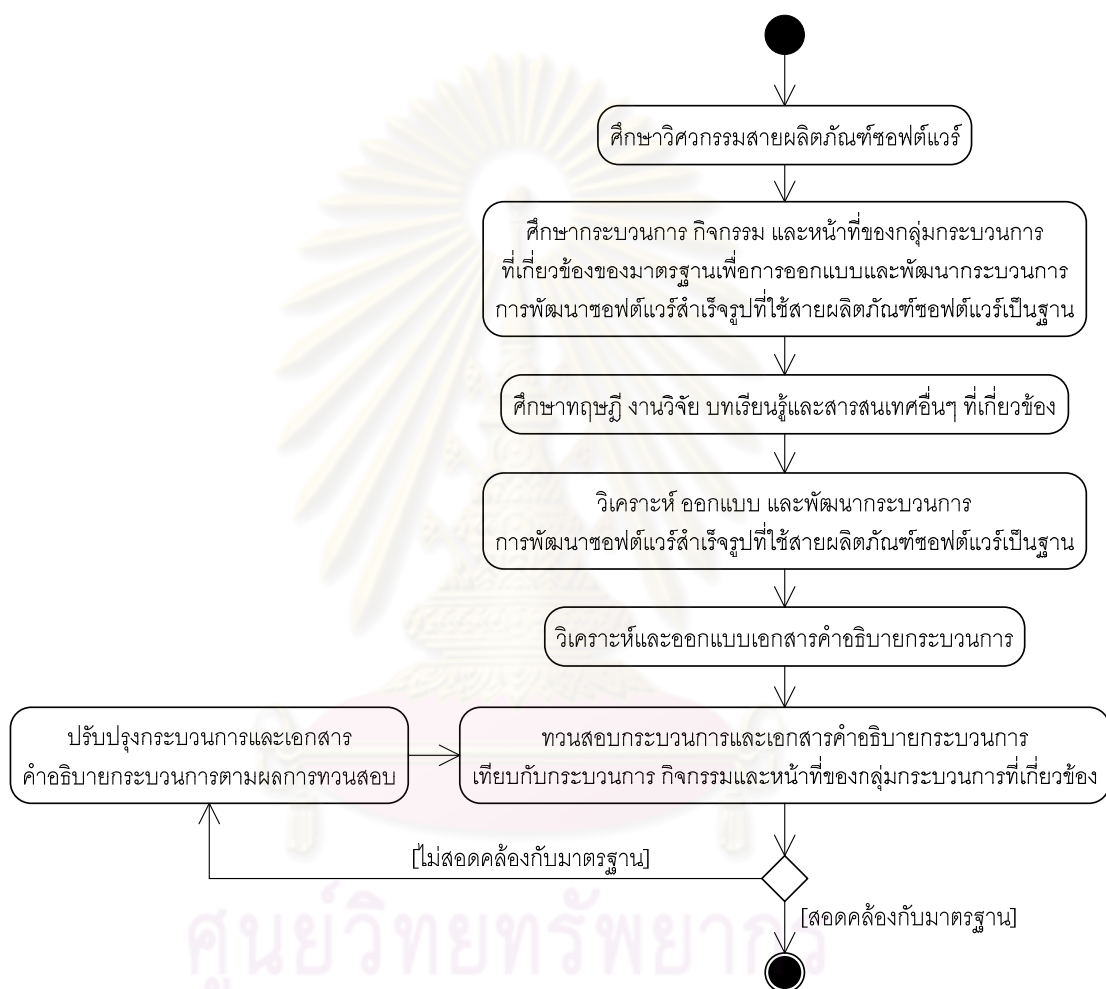
งานวิจัยที่เกี่ยวข้องเป็นงานวิจัยซึ่งนำมาใช้ในการออกแบบและพัฒนากระบวนการสำหรับการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานซึ่งได้อธิบายในรายละเอียดไว้ในบทที่ 2

3.1 การออกแบบและพัฒนาแบบจำลองกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

การศึกษาค้นคว้าข้อมูลพื้นฐานเพื่อใช้ในการวิเคราะห์ ออกแบบและพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานนั้น แบ่งออกเป็น 6 ขั้นตอนดังแสดงในรูปที่ 3.2 โดยสามารถอธิบายในรายละเอียดดังต่อไปนี้

3.1.1 ศึกษาวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์

ขั้นตอนนี้เป็นการศึกษาพื้นฐานของสายผลิตภัณฑ์ซอฟต์แวร์ซึ่งใช้เป็นฐานในการออกแบบและพัฒนากระบวนการสำหรับงานวิจัยนี้ ดังที่ได้อธิบายรายละเอียดของวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ไว้แล้วในบทที่ 2 ข้างต้น



รูปที่ 3.2 การออกแบบและพัฒนาแบบจำลองกระบวนการพัฒนาซอฟต์แวร์ที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

3.1.2 ศึกษากระบวนการ กิจกรรม และหน้าที่ของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานเพื่อการออกแบบและพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

ผู้วิจัยได้ทำการศึกษากระบวนการ กิจกรรม และหน้าที่ของกลุ่มกระบวนการของมาตรฐานไอทีพีเฟลอี 1517 เพื่อพิจารณาและวิเคราะห์ประเภทการนำเข้าของ

มาตรฐานที่สอดคล้องและเหมาะสมกับจุดประสงค์ของการออกแบบและพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานมากที่สุด เพื่อเลือกใช้กลุ่มของกระบวนการที่เกี่ยวข้องในประเภทการใช้ซ้่านั้นเป็นฐานในการออกแบบและพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

3.1.3 ศึกษาทฤษฎี งานวิจัย บทเรียนรู้และสารสนเทศอื่นๆ ที่เกี่ยวข้อง

ขั้นตอนนี้ผู้วิจัยได้ทำการศึกษาหาความรู้จากแหล่งความรู้อื่นๆ ที่เกี่ยวข้องซึ่งความรู้ที่ได้นำมาใช้ในการวิเคราะห์ ออกแบบและพัฒนากระบวนการนั้น คือ กลุ่มนำเข้าแบบรูปและกลุ่มนำเข้าส่วนเกี่ยวข้องจำเพาะ ดังที่ได้อธิบายไว้ในหัวข้อ 2) และ 3) ข้างต้น

3.1.4 วิเคราะห์ ออกแบบและพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

เมื่อได้ทำการศึกษาองค์ความรู้ที่จำเป็นต้องใช้ในการออกแบบและพัฒนากระบวนการแล้ว ผู้วิจัยได้ทำการออกแบบและพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน โดยแบ่งเป็น 4 ส่วน คือ กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูปและกระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป ซึ่งสามารถอธิบายในรายละเอียดของการวิเคราะห์ ออกแบบและพัฒนากระบวนการโดยเทียบกับมาตรฐานไอทีพีพีอี 1517 ในประเภทการใช้ซ้ําการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์และกลุ่มกระบวนการการวิเคราะห์และการวัดผลที่ได้จากแบบจำลองวุฒิภาวะความสามารถแบบบูรณาการ ดังแสดงได้ดังตารางที่ 3.1

ตารางที่ 3.1 กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานและประเภทของกลุ่มกระบวนการที่ใช้

COTS Development Process	IEEE 1517 Process (Development of software product with assets reuse category)
COTS tailoring	Development process
COTS integration	Development process
COTS maintenance	Maintenance process
COTS support	Measurement and analysis [18]

3.1.5 วิเคราะห์และออกแบบเอกสารคำอธิบายกระบวนการ

ขั้นตอนนี้เป็นการระบุคำอธิบายกระบวนการของกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานในแต่ละกระบวนการ ซึ่งได้จากการศึกษากระบวนการ กิจกรรม และหน้าที่ของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีเฟลอี 1517 โดยละเอียดซึ่งอธิบายถึงกิจกรรมและกระบวนการที่ต้องเกิดขึ้น ดังแสดงในรายละเอียดได้ดังตารางที่ 2

ตารางที่ 3.2 รายละเอียดของคำอธิบายกระบวนการ

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ชื่อของกระบวนการ
วัตถุประสงค์	อธิบายจุดประสงค์ของกระบวนการ
เกณฑ์การเข้า	อธิบายเงื่อนไขก่อนเริ่มกระบวนการ
ข้อมูลนำเข้า	อธิบายข้อมูลนำเข้าที่จำเป็นต้องใช้ในกระบวนการ
กระบวนการ	อธิบายขั้นตอนของกระบวนการ
ข้อมูลนำออก	อธิบายข้อมูลนำออกที่เป็นผลลัพธ์จากกระบวนการ
เกณฑ์การออก	อธิบายเงื่อนไขของการสิ้นสุดกระบวนการ
ผู้รับผิดชอบ	อธิบายผู้ที่เกี่ยวข้องในสำหรับกระบวนการ

3.1.6 ทวนสอบกระบวนการและเอกสารคำอธิบายกระบวนการเทียบกับกระบวนการ กิจกรรมและหน้าที่ของกลุ่มกระบวนการที่เกี่ยวข้อง

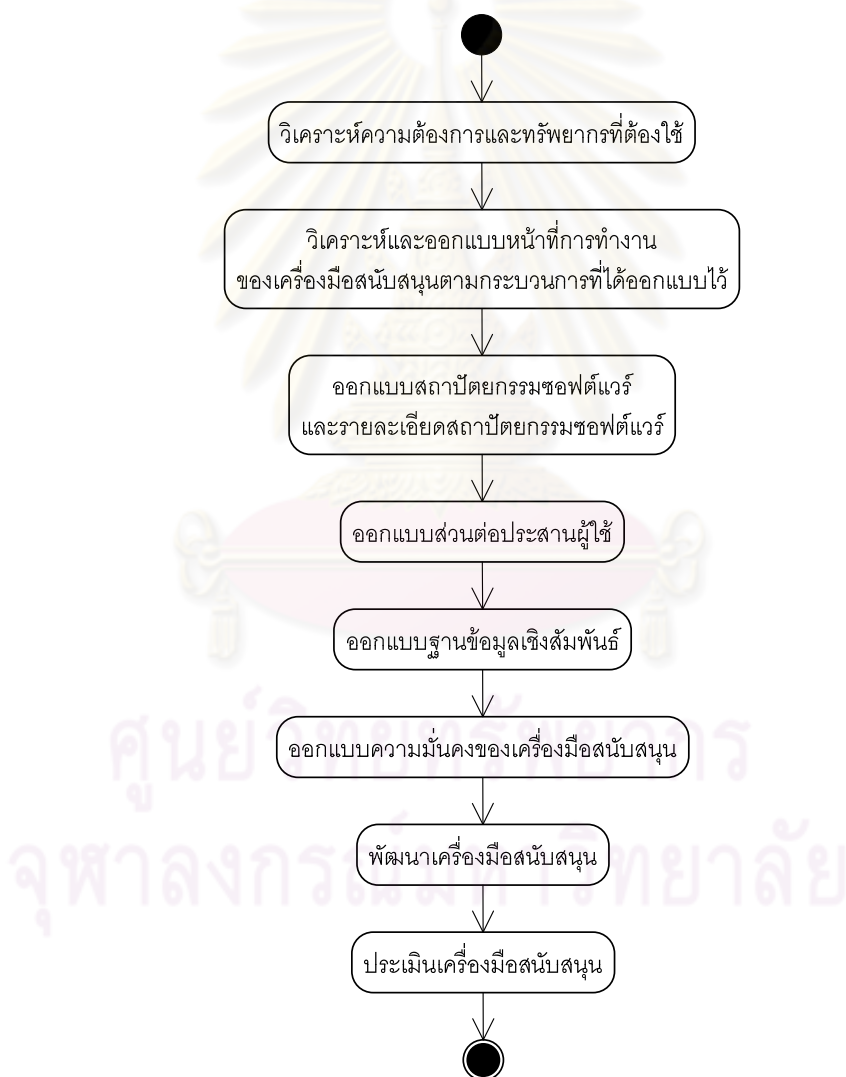
ขั้นตอนนี้เป็นการทวนสอบความสอดคล้องกันระหว่างกระบวนการและเอกสารคำอธิบายกระบวนการเทียบกับมาตรฐานซึ่งเป็นกระบวนการที่เกี่ยวข้องว่ามีความสอดคล้องกันหรือไม่ รวมถึงตรวจสอบความถูกต้องและความสมบูรณ์ของกระบวนการอีกด้วย

3.1.7 ปรับปรุงกระบวนการและเอกสารคำอธิบายกระบวนการตามผลการทวนสอบ

เมื่อกระบวนการที่ผ่านการทวนสอบแล้วมีความผิดพลาดเกิดขึ้นไม่ว่าจะเป็นความไม่สอดคล้องกันหรือความไม่สมบูรณ์ เป็นต้น ขั้นตอนนี้เป็นการปรับปรุงกระบวนการและเอกสารคำอธิบายกระบวนการที่เกิดข้อผิดพลาดนั้นให้สมบูรณ์ ถูกต้องและสอดคล้องกันกับมาตรฐานในกลุ่มกระบวนการที่เกี่ยวข้อง

3.2 การออกแบบและพัฒนาซอฟต์แวร์ต้นแบบที่ใช้สนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

สำหรับขั้นตอนการวิเคราะห์ ออกแบบและพัฒนาซอฟต์แวร์ต้นแบบที่ใช้สนับสนุนกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน ผู้วิจัยใช้แบบจำลองวอเตอร์ฟอลล์ (Waterfall model) เป็นวัฏจักรชีวิตซอฟต์แวร์ของการพัฒนาเครื่องมือสนับสนุนนี้ ดังรูปที่ 3.3 ส่วนรายละเอียดของการวิเคราะห์ ออกแบบและพัฒนาซอฟต์แวร์ต้นแบบซึ่งใช้เป็นเครื่องมือสนับสนุนกระบวนการของงานวิจัยนี้จะอธิบายในรายละเอียดในบทที่ 5 และ 6 ต่อไป



รูปที่ 3.3 การออกแบบและพัฒนาซอฟต์แวร์ต้นแบบที่ใช้สนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

บทที่ 4

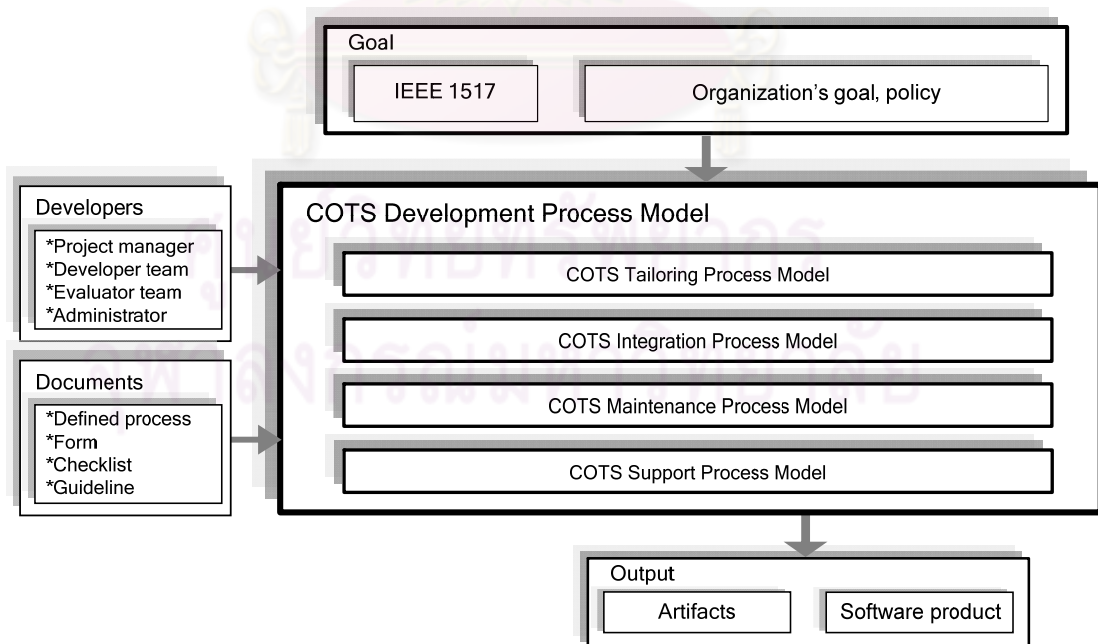
กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์

ซอฟต์แวร์เป็นฐาน

ในบทนี้จะกล่าวถึงผลลัพธ์ที่ได้จากการออกแบบและพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน ซึ่งประกอบด้วย 3 กระบวนการหลัก คือ กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปและกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป นอกจากนี้ยังประกอบด้วย 1 กระบวนการสนับสนุน คือ กระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป โดยแบบจำลองกระบวนการของแต่ละกระบวนการนั้นจะประกอบด้วย 3 ชั้น คือ ชั้นแนวคิด ชั้นกระแสนงานและชั้นรายละเอียด แบบจำลองกระบวนการสำหรับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานมีรายละเอียดดังต่อไปนี้

4.1 แบบจำลองกระบวนการชั้นแนวคิด (Conceptual Layer Process Model)

แบบจำลองกระบวนการการพัฒนาซอฟต์แวร์ชั้นแนวคิดนั้นสามารถแสดงในรายละเอียดได้ดังรูปที่ 4.1



รูปที่ 4.1 แบบจำลองกระบวนการชั้นแนวคิด

จากรูปที่ 4.1 แบบจำลองกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปขั้นแนวคิด ประกอบด้วย 5 องค์ประกอบ ซึ่งสามารถอธิบายในรายละเอียดได้ดังต่อไปนี้

1) ผู้พัฒนา (Developers) เป็นกลุ่มของบทบาทหน้าที่ของผู้ที่เกี่ยวข้องกับกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป

2) เอกสาร (Documents) เป็นเอกสารซึ่งเป็นส่วนนำเข้าสู่กระบวนการที่ใช้สำหรับสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป

3) เป้าหมาย (Goal) เป็นเป้าหมายของกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป

4) แบบจำลองกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป (COTS Development Process) เป็นลำดับของกิจกรรมสำหรับกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป ซึ่งรายละเอียดจะอธิบายในหัวข้อ 4.2 ต่อไป

5) ข้อมูลนำออก (Output) เป็นผลลัพธ์ที่ได้จากกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป

สำหรับผลการออกแบบกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปในแต่ละกระบวนการมีองค์ประกอบดังต่อไปนี้

1) บทบาทและความรับผิดชอบ ที่เกิดขึ้นมีดังต่อไปนี้

(1) ผู้จัดการโครงการ (Project Manager) โดยผู้จัดการโครงการมีความรับผิดชอบดังต่อไปนี้

- ทำหน้าที่วางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป
- ทำหน้าที่วางแผนโครงการการบูรณาการซอฟต์แวร์สำเร็จรูป
- ทำหน้าที่วางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป
- ทำหน้าที่ออกแบบสถาปัตยกรรมการปรับแต่งซอฟต์แวร์สำเร็จรูป
- ทำหน้าที่ออกแบบสถาปัตยกรรมการบูรณาการซอฟต์แวร์สำเร็จรูป
- ทำหน้าที่วิเคราะห์ปัญหาและการดัดแปลง
- ทำหน้าที่วัดผลกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป

(2) ทีมผู้พัฒนา (Developer Team) โดยทีมผู้พัฒนามีความรับผิดชอบดังต่อไปนี้

- ทำหน้าที่พัฒนาปรับแต่งซอฟต์แวร์สำเร็จรูป
- ทำหน้าที่ติดตั้งซอฟต์แวร์
- ทำหน้าที่สนับสนุนการยอมรับซอฟต์แวร์
- ทำหน้าที่พัฒนาบูรณาการซอฟต์แวร์สำเร็จรูป
- ทำหน้าที่ดัดแปลง ย้าย และยกเลิกข้อมูลซอฟต์แวร์สำเร็จรูป

(3) ทีมผู้ประเมิน (Evaluator Team) โดยทีมผู้ประเมินมีความรับผิดชอบ

ดังต่อไปนี้

- ทำหน้าที่ประเมินข้อกำหนดความต้องการ
- ทำหน้าที่ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
- ทำหน้าที่ประเมินการพัฒนาส่วนประกอบ
- ทำหน้าที่ประเมินการบูรณาการกับระบบที่มีอยู่
- ทำหน้าที่ประเมินการบูรณาการระบบและซอฟต์แวร์
- ทำหน้าที่ประเมินแผนการติดตั้ง

(4) ผู้จัดการระบบ (Administrator) โดยผู้จัดการระบบมีความรับผิดชอบ

ดังต่อไปนี้

- ทำหน้าที่จัดการกับข้อมูลผู้ใช้งานระบบ
- ทำหน้าที่จัดการกับเอกสารที่เกิดขึ้นในแต่ละกระบวนการ รวมถึงเอกสารที่ใช้ในการสนับสนุนในแต่ละกระบวนการ
- ทำหน้าที่จัดการกับเกณฑ์การใช้ซ้ำที่ใช้สำหรับสนับสนุนกระบวนการประเมิน

2) เอกสารสนับสนุนกระบวนการที่เกิดขึ้นมีดังต่อไปนี้

(1) นิยามกระบวนการ (Defined process) คือ เอกสารที่ระบุขั้นตอนและหน้าที่ที่ต้องทำในแต่ละกระบวนการ เพื่อใช้สนับสนุนให้ผู้ดำเนินการสามารถนำไปใช้ดำเนินการได้อย่างสมบูรณ์

(2) ฟอร์ม (Form) คือ เอกสารที่นำมาใช้เพื่อการเก็บข้อมูลที่สำคัญเพื่อนำมาใช้ในการวิเคราะห์ข้อมูลของการดำเนินงานของกระบวนการ

(3) รายการตรวจสอบ (Checklist) คือ เอกสารที่ระบุถึงสิ่งที่ต้องทำในกิจกรรมนั้นๆ เพื่อให้ผู้ดำเนินการสามารถทวนสอบกิจกรรมได้ว่าได้กระทำครบถ้วนหรือไม่

(4) แนวทางในการดำเนินการ (Guideline) คือ เอกสารที่ช่วยให้ผู้ดำเนินการทราบถึงวิธีในการทำงานของกิจกรรมนั้นๆ ซึ่งช่วยให้ผู้ดำเนินการปฏิบัติหน้าที่ได้ง่ายขึ้น

3) เป้าหมายของกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป คือ การวิเคราะห์และออกแบบให้สอดคล้องกับกระบวนการที่เกี่ยวข้องตามมาตรฐานไอทีพีเฟลอี 1517

4) กระบวนการที่เกิดขึ้นในกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปนั้นประกอบไปด้วย 4 กลุ่มกระบวนการซึ่งสามารถอธิบายในรายละเอียดในหัวข้อที่ 4.2 ต่อไป

5) ข้อมูลนำออก คือ ผลลัพธ์ที่ได้จากการออกแบบและพัฒนากระบวนการซึ่งประกอบด้วยเอกสารที่เกิดขึ้น (Artifacts) ซึ่งเป็นผลลัพธ์ของแต่ละกระบวนการและผลิตภัณฑ์ซอฟต์แวร์ (Software Product)

4.2 แบบจำลองกระบวนการชั้นกระแสนงาน (Workflow Layer Process Model)

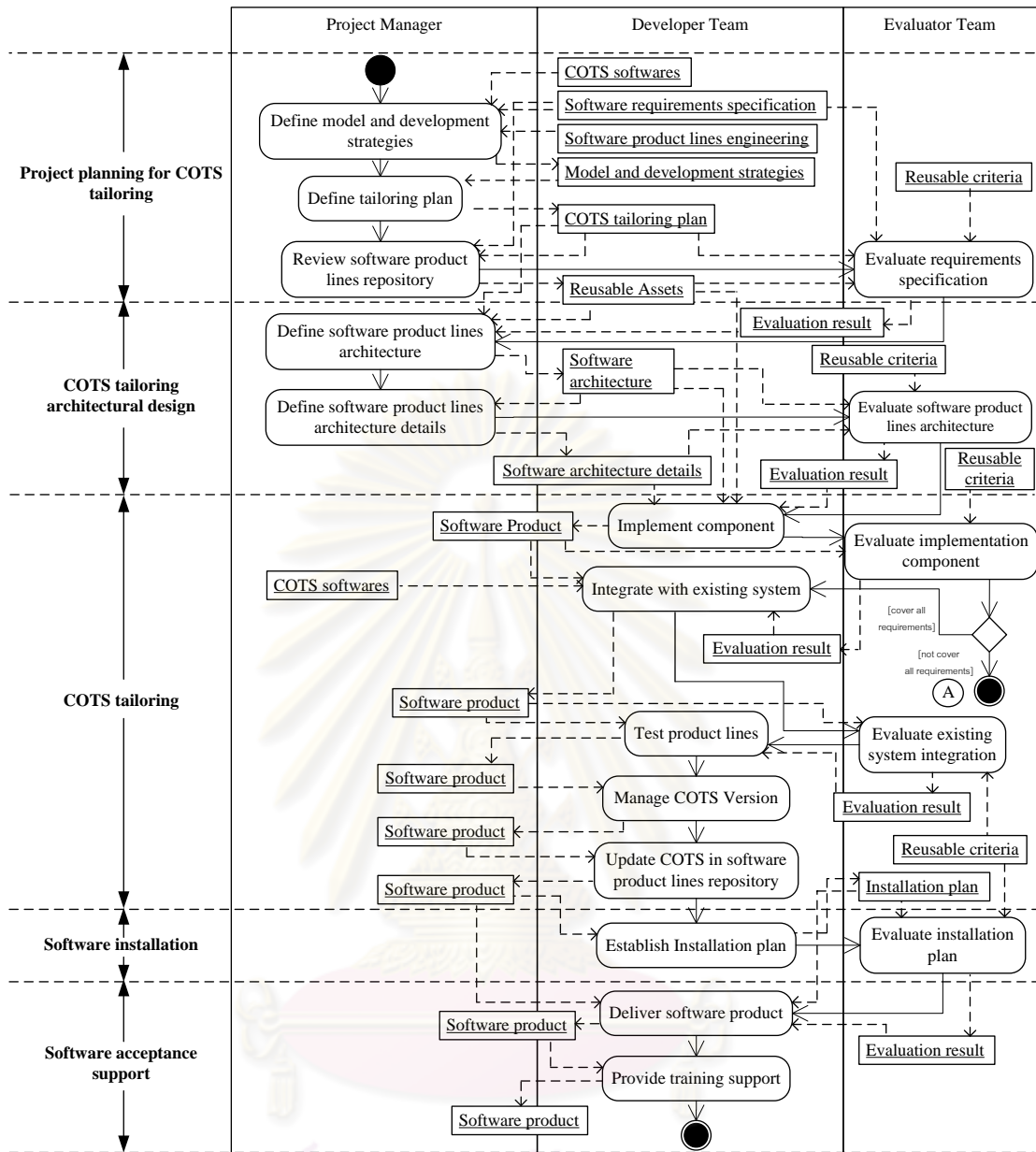
ในการออกแบบขั้นนี้จะระบุถึงขั้นตอนการทำงานของแต่ละกิจกรรมที่เกิดขึ้นโดยนำเสนอด้วยแผนภาพกิจกรรม โดยแบบจำลองกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานชั้นกระแสนงานประกอบด้วย 4 กลุ่มกระบวนการโดยมีรายละเอียดดังต่อไปนี้

4.2.1) กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปชั้นกระแสนงาน

กระบวนการนี้ประกอบด้วย 5 ขั้นตอน ดังรูปที่ 4.2 ซึ่งสามารถอธิบายในรายละเอียดได้ดังต่อไปนี้

1) ขั้นตอนการวางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป (Project planning for COTS tailoring phase)

จุดประสงค์ของขั้นตอนนี้เพื่อได้มาซึ่งแบบจำลอง กลยุทธ์ในการพัฒนา แผนการปรับแต่งและสินทรัพย์ที่สามารถนำมาใช้ซ้ำเพื่อนำมาใช้สำหรับกระบวนการปรับแต่งซอฟต์แวร์สำเร็จรูปซึ่งผลลัพธ์ที่ได้จากขั้นตอนนี้จะถูกประเมินโดยเทียบกับเกณฑ์การใช้ซ้ำรวมถึงข้อกำหนดความต้องการซอฟต์แวร์ก่อนที่จะเข้าสู่ขั้นตอนต่อไป



รูปที่ 4.2 กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปขั้นกระบวนการ

2) ขั้นตอนการออกแบบสถาปัตยกรรมการปรับแต่งซอฟต์แวร์สำเร็จรูป (COTS tailoring architectural design phase)

จุดประสงค์ของขั้นตอนนี้เพื่อได้มาซึ่งสถาปัตยกรรมและรายละเอียดของสถาปัตยกรรมซอฟต์แวร์สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปซึ่งสถาปัตยกรรมที่ได้นี้จะต้องประกอบด้วยส่วนประกอบที่ทำหน้าที่เป็นคุณลักษณะหลัก คุณลักษณะตัวเลือกและคุณลักษณะทางเลือก ซึ่งเป็นส่วนประกอบพื้นฐานของสายผลิตภัณฑ์

ซอฟต์แวร์ นอกจากนั้นรายละเอียดของสถาปัตยกรรมจะถูกระบุ โดยผลลัพธ์ที่ได้จากขั้นตอนนี้จะถูกประเมินเทียบกับเกณฑ์การใช้อีกก่อนที่จะเข้าสู่ขั้นตอนถัดไป

3) ขั้นตอนการปรับแต่งซอฟต์แวร์สำเร็จรูป (COTS tailoring phase)

จุดประสงค์ของขั้นตอนนี้เพื่อทำการปรับแต่งสินทรัพย์ขององค์กรที่อยู่ในรูปของซอฟต์แวร์สำเร็จรูปซึ่งทำหน้าที่เป็นคุณลักษณะหลัก และนำเอาสินทรัพย์ที่สามารถนำมาใช้ซ้ำที่เป็นสินทรัพย์อื่นๆ ที่สามารถนำมาประยุกต์ใช้ในกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปนำมาใช้ซ้ำเป็นคุณลักษณะตัวเลือกหรือคุณลักษณะทางเลือกตามลำดับความสำคัญ โดยผลลัพธ์ที่ได้จากขั้นตอนนี้จะถูกประเมินเทียบกับเกณฑ์การใช้อีกและข้อกำหนดความต้องการก่อนที่จะเข้าสู่ขั้นตอนถัดไป

4) ขั้นตอนการติดตั้งซอฟต์แวร์ (Software installation phase)

จุดประสงค์ของขั้นตอนนี้เพื่อได้มาซึ่งแผนการติดตั้งสำหรับผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากขั้นตอนการปรับแต่งซอฟต์แวร์สำเร็จรูปโดยผลลัพธ์ที่ได้จากขั้นตอนนี้จะถูกประเมินเทียบกับเกณฑ์การใช้อีกก่อนที่จะเข้าสู่ขั้นตอนถัดไป

5) ขั้นตอนการสนับสนุนการยอมรับซอฟต์แวร์ (Software acceptance support phase)

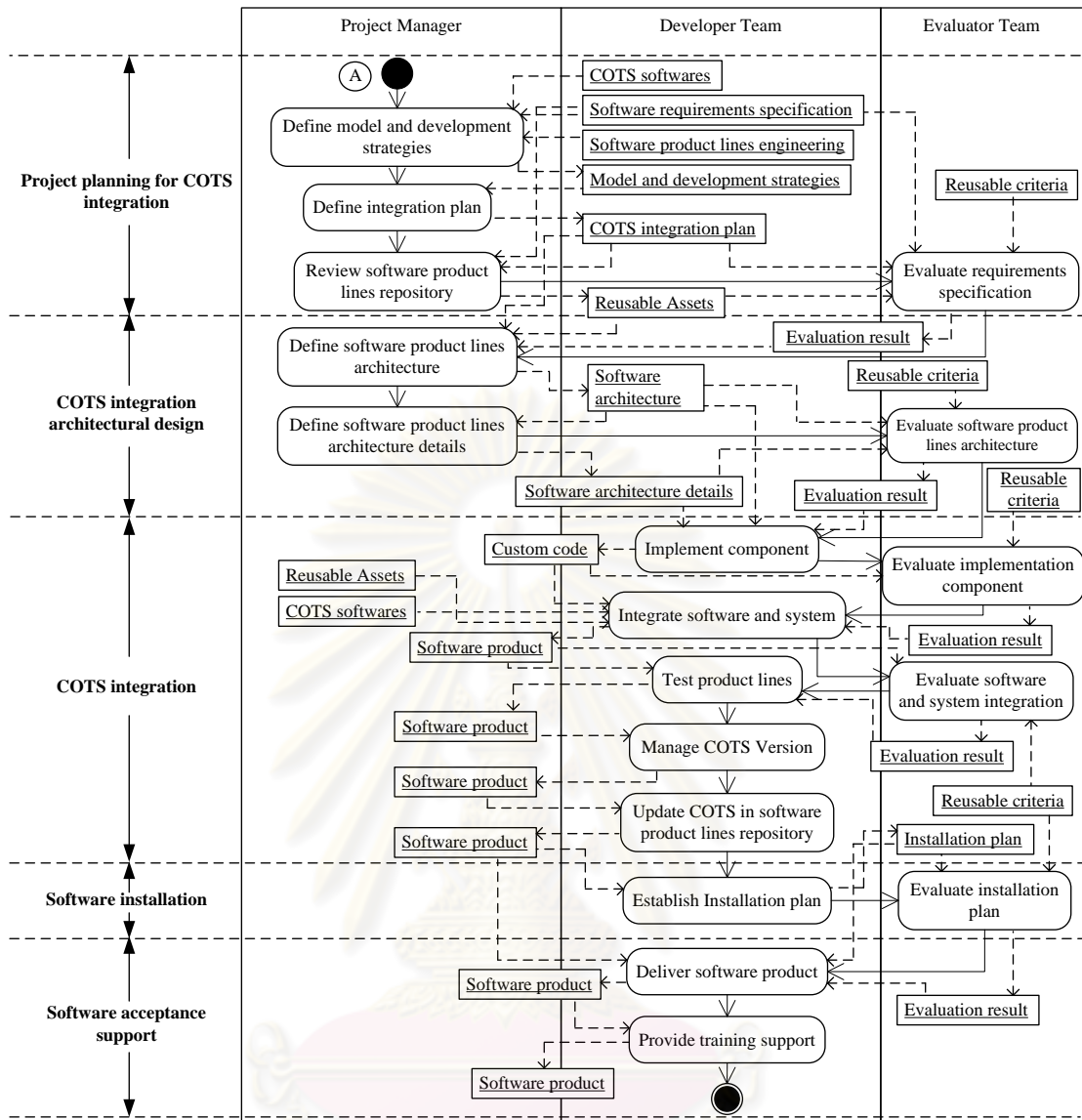
จุดประสงค์ของขั้นตอนนี้เพื่อทำการส่งมอบผลิตภัณฑ์ซอฟต์แวร์ให้กับผู้ใช้และสนับสนุนการนำเอาผลิตภัณฑ์ซอฟต์แวร์ไปใช้งานได้อย่างมีประสิทธิภาพนอกจากนี้ยังรวมถึงการเตรียมการฝึกฝนและจัดทำเอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้อีกด้วย

4.2.2) กระบวนการบูรณาการซอฟต์แวร์สำเร็จรูปขึ้นกระแสนงาน

กระบวนการนี้ประกอบด้วย 5 ขั้นตอน ดังรูปที่ 4.3 ซึ่งสามารถอธิบายในรายละเอียดได้ดังต่อไปนี้

1) ขั้นตอนการวางแผนโครงการการบูรณาการซอฟต์แวร์สำเร็จรูป (Project planning for COTS integration phase)

จุดประสงค์ของขั้นตอนนี้เพื่อได้มาซึ่งแบบจำลอง กลยุทธ์ในการพัฒนา แผนการบูรณาการและสินทรัพย์ที่สามารถนำมาใช้ซ้ำเพื่อนำมาใช้สำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูปซึ่งผลลัพธ์ที่ได้จากขั้นตอนนี้จะถูกประเมินโดยเทียบกับเกณฑ์การใช้อีก รวมถึงข้อกำหนดความต้องการซอฟต์แวร์ก่อนที่จะเข้าสู่ขั้นตอนต่อไป



รูปที่ 4.3 กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปชั้นกระแสนงาน

2) ขั้นตอนการออกแบบสถาปัตยกรรมการบูรณาการซอฟต์แวร์สำเร็จรูป (COTS integration architectural design phase)

จุดประสงค์ของขั้นตอนนี้เพื่อได้มาซึ่งสถาปัตยกรรมและรายละเอียดของสถาปัตยกรรมซอฟต์แวร์สำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปซึ่งสถาปัตยกรรมที่ได้นี้จะต้องประกอบด้วยส่วนประกอบที่ทำหน้าที่เป็นคุณลักษณะหลัก คุณลักษณะตัวเลือกและคุณลักษณะทางเลือก ซึ่งเป็นส่วนประกอบพื้นฐานของสายผลิตภัณฑ์ซอฟต์แวร์ นอกจากนั้นรายละเอียดของสถาปัตยกรรมจะถูกระบุ โดยผลลัพธ์ที่ได้จากขั้นตอนนี้จะถูกประเมินเทียบกับเกณฑ์การใช้ซ้ำก่อนที่จะเข้าสู่ขั้นตอนถัดไป

3) ขั้นตอนการบูรณาการซอฟต์แวร์สำเร็จรูป (COTS integration phase)

จุดประสงค์ของขั้นตอนนี้เพื่อทำการบูรณาการสินทรัพย์ขององค์กรที่อยู่ในรูปของซอฟต์แวร์สำเร็จรูปซึ่งทำหน้าที่เป็นคุณลักษณะหลักโดยทำการออกแบบและพัฒนารหัสคำสั่งเพิ่มเติมซึ่งเป็นส่วนประกอบซอฟต์แวร์ที่เพิ่มเติมความสามารถของสินทรัพย์ที่มีอยู่ให้สามารถครอบคลุมข้อกำหนดความต้องการก่อนที่จะนำเอาสินทรัพย์ที่สามารถนำมาใช้ซ้ำที่เป็นสินทรัพย์อื่นๆ ที่สามารถนำมาประยุกต์ใช้ในกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปนำมาใช้ซ้ำเป็นคุณลักษณะตัวเลือกหรือคุณลักษณะทางเลือกตามลำดับความสำคัญ โดยผลลัพธ์ที่ได้จากขั้นตอนนี้จะถูกประเมินเทียบกับเกณฑ์การใช้ซ้ำและข้อกำหนดความต้องการก่อนที่จะเข้าสู่ขั้นตอนถัดไป

4) ขั้นตอนการติดตั้งซอฟต์แวร์ (Software installation phase)

จุดประสงค์ของขั้นตอนนี้เพื่อได้มาซึ่งแผนการติดตั้งสำหรับผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากขั้นตอนการบูรณาการซอฟต์แวร์สำเร็จรูปโดยผลลัพธ์ที่ได้จากขั้นตอนนี้จะถูกประเมินเทียบกับเกณฑ์การใช้ซ้ำก่อนที่จะเข้าสู่ขั้นตอนถัดไป

5) ขั้นตอนการสนับสนุนการยอมรับซอฟต์แวร์ (Software acceptance support phase)

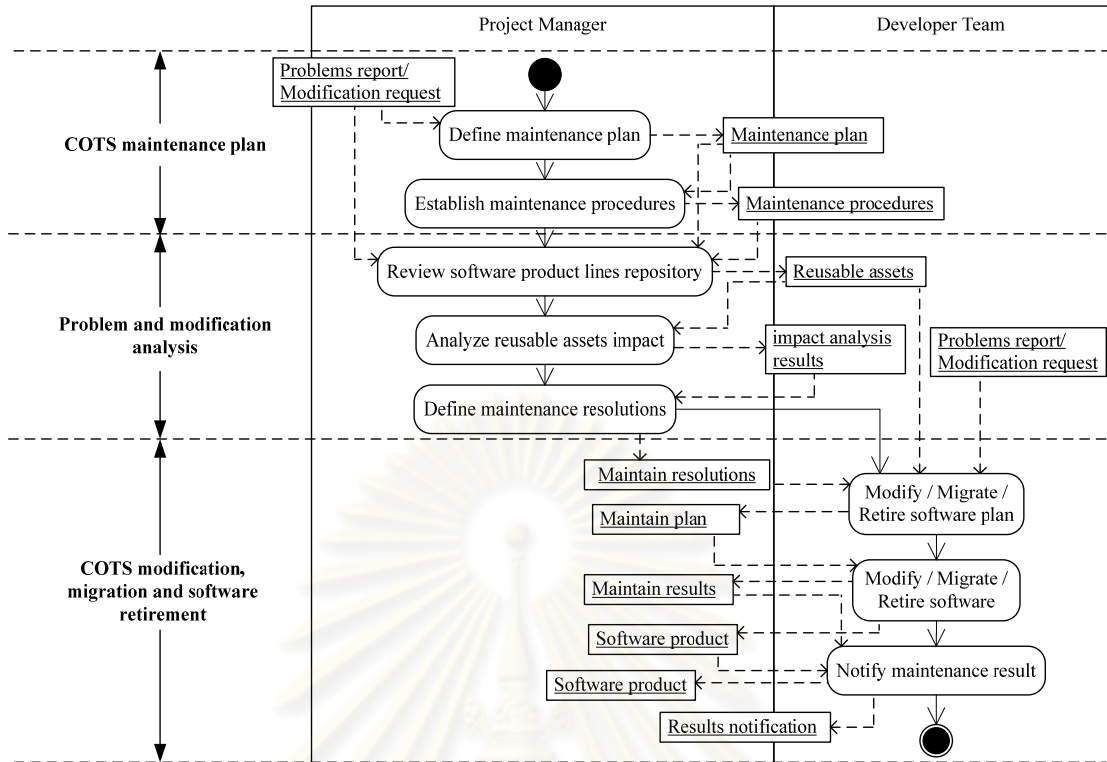
จุดประสงค์ของขั้นตอนนี้เพื่อทำการส่งมอบผลิตภัณฑ์ซอฟต์แวร์ให้กับผู้ใช้และสนับสนุนการนำเอาผลิตภัณฑ์ซอฟต์แวร์ไปใช้งานได้อย่างมีประสิทธิภาพนอกจากนั้นยังรวมถึงการเตรียมการฝึกฝนและจัดทำเอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้อีกด้วย

4.2.3) กระบวนการบำรุงรักษาซอฟต์แวร์สำเร็จรูปขั้นกระแสนาน

กระบวนการนี้ประกอบด้วย 3 ขั้นตอน ดังรูปที่ 4.4 ซึ่งสามารถอธิบายในรายละเอียดได้ดังต่อไปนี้

1) ขั้นตอนการวางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป (COTS maintenance phase)

จุดประสงค์ของขั้นตอนนี้เพื่อได้มาซึ่งแผนการสำหรับการดำเนินการการบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์ที่อยู่ในสายผลิตภัณฑ์โดยขั้นตอนนี้จะเกิดขึ้นได้ก็ต่อเมื่อมีรายงานปัญหาหรือคำร้องขอการดัดแปลง



รูปที่ 4.4 กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูปชั้นกระแสนาน

2) ขั้นตอนการวิเคราะห์ปัญหาและการดัดแปลง (Problem and modification analysis phase)

จุดประสงค์ของขั้นตอนนี้เพื่อได้มาซึ่งพิจารณานำเอาสินทรัพย์ที่อยู่ในสายผลิตภัณฑ์มาใช้สำหรับการบำรุงรักษา วิเคราะห์ผลกระทบต่อการนำสินทรัพย์เหล่านั้นมาใช้ซ้ำและหาวิธีที่ใช้สำหรับการบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ที่อยู่ในสายผลิตภัณฑ์

3) ขั้นตอนการดัดแปลง ย้ายและยกเลิกข้อมูลซอฟต์แวร์สำเร็จรูป (COTS modification, migration and software retirement phase)

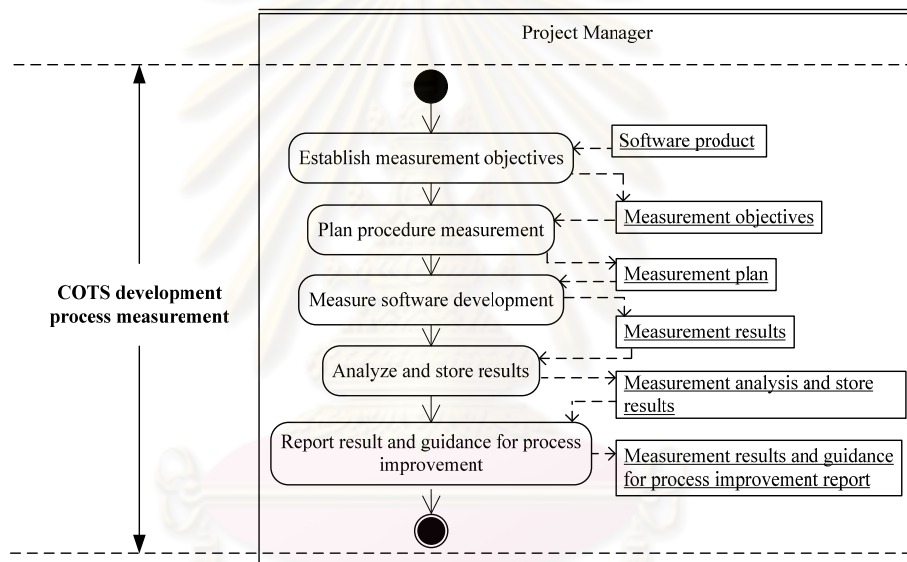
จุดประสงค์ของขั้นตอนนี้เพื่อทำการบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์โดยการดัดแปลง ย้ายและยกเลิกซอฟต์แวร์ตามแผนการบำรุงรักษาและวิธีที่ใช้สำหรับการบำรุงรักษาตามที่ได้กำหนดไว้ก่อนหน้านี้โดยเมื่อได้ทำการบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์แล้วรายละเอียดที่ได้จากขั้นตอนนี้จะถูกแจ้งไปยังผู้ที่เกี่ยวข้อง

4.2.4) กระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูปขั้นกระแสน

กระบวนการนี้ประกอบด้วย 1 ขั้นตอน ดังรูปที่ 4.5 ซึ่งสามารถอธิบายในรายละเอียดได้ดังต่อไปนี้

- 1) ขั้นตอนการวัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป (COTS development process measurement phase)

จุดประสงค์ของขั้นตอนนี้เพื่อทำการวัดผลกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปเพื่อหาจุดบกพร่องที่เกิดขึ้นและพัฒนากระบวนการให้มีประสิทธิภาพมากยิ่งขึ้นสำหรับการพัฒนาครั้งถัดไป



รูปที่ 4.5 กระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูปขั้นกระแสน

4.3 แบบจำลองกระบวนการชั้นรายละเอียด (Elaboration Layer Process Model)

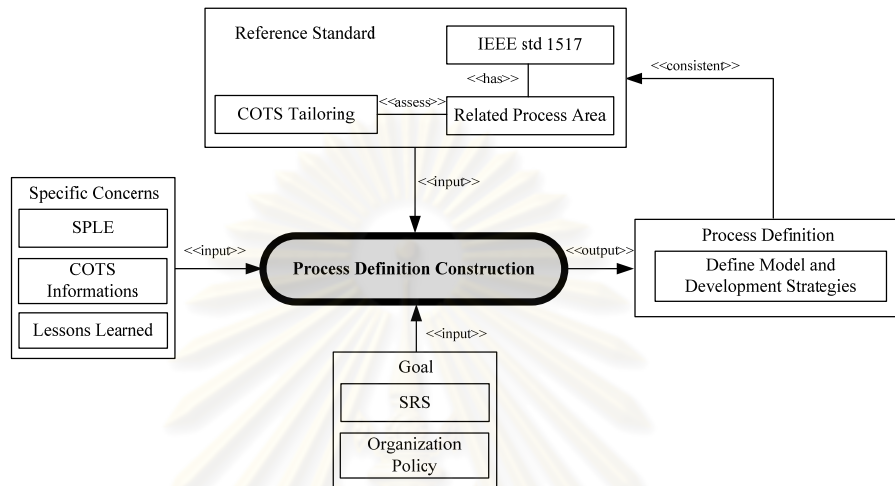
แบบจำลองกระบวนการชั้นรายละเอียดได้กำหนดกระบวนการการทำงานตามรายละเอียดกิจกรรมที่ได้จากบทที่ 3 ซึ่งสามารถอธิบายในรายละเอียดของคำอธิบายกระบวนการสำหรับชั้นรายละเอียดได้ดังต่อไปนี้

4.3.1 กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป

กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปประกอบด้วย 5 ขั้นตอน ซึ่งสามารถอธิบายในรายละเอียดได้ดังต่อไปนี้

4.3.1.1 ขั้นตอนวางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป

ในส่วนนี้เป็นส่วนที่อธิบายถึงวิธีการสร้างคำอธิบายกระบวนการโดยได้ยกตัวอย่างการสร้างคำอธิบายกระบวนการสำหรับการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา ซึ่งเป็นส่วนหนึ่งของขั้นตอนวางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป ดังแสดงในรูปที่ 4.6



รูปที่ 4.6 การสร้างคำอธิบายกระบวนการการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา

จากรูปที่ 4.6 ได้อธิบายถึงการสร้างคำอธิบายกระบวนการของกระบวนการการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา ซึ่งสามารถอธิบายในรายละเอียดได้ดังต่อไปนี้

1) มาตรฐานที่ใช้อ้างอิง (Reference Standard)

มาตรฐานที่ใช้อ้างอิง คือ มาตรฐานไอทีพีเฟลอี 1517 ซึ่งใช้สำหรับการพัฒนาซอฟต์แวร์จากสินทรัพย์โดยใช้เป็นฐานในการสร้างคำอธิบายกระบวนการโดยแบ่งกลุ่มนำเข้าเป็น 2 ส่วน ดังนี้

1.1) กลุ่มกระบวนการที่เกี่ยวข้อง (Related Process Area)

กลุ่มกระบวนการที่เกี่ยวข้องเป็นกระบวนการที่อยู่ในมาตรฐานไอทีพีเฟลอี 1517 ซึ่งเกี่ยวข้องกับการใช้ซ้ำในการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปจากสินทรัพย์

1.2) การปรับแต่งซอฟต์แวร์สำเร็จรูป (COTS Tailoring)

การปรับแต่งซอฟต์แวร์สำเร็จรูป คือ เป้าหมายสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปโดยกระบวนการที่ได้นั้นจะถูกประเมิน (Assess) เทียบกับกลุ่ม

กระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีพีเอส 1517 และผู้พัฒนาจะทวนสอบคำอธิบายกระบวนการกับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปที่ได้ว่ามีความสอดคล้องกันหรือไม่

2) ส่วนเกี่ยวข้องจำเพาะ (Specific concerns)

ส่วนเกี่ยวข้องจำเพาะเป็นสาระสำคัญที่จำเป็นต้องใช้ในการออกแบบและพัฒนาคำอธิบายกระบวนการ โดยแบ่งกลุ่มนำเข้าเป็น 3 ส่วน ดังนี้

2.1) วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Software Product Lines Engineering: SPLE)

วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ใช้เป็นฐานในการสร้างคำอธิบายกระบวนการ โดยวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ได้อธิบายในรายละเอียดไว้ในบทที่ 2

2.2) สารสนเทศซอฟต์แวร์สำเร็จรูป (COTS Informations)

สารสนเทศซอฟต์แวร์สำเร็จรูปเป็นเนื้อหาสาระที่เกี่ยวข้องกับซอฟต์แวร์สำเร็จรูปที่ถูกนำมาใช้ในการสร้างคำอธิบายกระบวนการ

2.3) บทเรียนรู้ (Lesson Learned)

บทเรียนรู้เป็นประสบการณ์ที่ได้จากการออกแบบและพัฒนากระบวนการหรือประสบการณ์ต่างๆ ซึ่งสามารถนำมาประยุกต์ใช้กับงานวิจัยนี้ได้

3) เป้าหมาย (Goals)

เป้าหมายของการสร้างคำอธิบายกระบวนการ แบ่งกลุ่มนำเข้าเป็น 2 ส่วน ดังนี้

3.1) ข้อกำหนดความต้องการซอฟต์แวร์ (Software Requirements Specification: SRS)

ข้อกำหนดความต้องการซอฟต์แวร์เป็นคุณลักษณะของผลิตภัณฑ์ซอฟต์แวร์ที่องค์กรต้องการจากการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์

3.2) นโยบายขององค์กร (Organization Policy)

นโยบายขององค์กรเป็นส่วนที่ระบุถึงขอบเขตหรือสิ่งที่สนใจซึ่งเป็นเป้าหมายของการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ขององค์กร

นอกจากนั้นกลุ่มนำออกที่ได้จากการสร้างคำอธิบายกระบวนการ คือ คำอธิบายกระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนาซึ่งเป็นส่วนหนึ่งของขั้นตอนวางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป โดยขั้นตอนนี้ประกอบด้วย 4 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.1-4.4

ตารางที่ 4.1 กระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา (Define model and development strategies)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	กำหนดแบบจำลองและกลยุทธ์ในการพัฒนา
วัตถุประสงค์	เพื่อให้ได้แบบจำลองและกลยุทธ์ในการพัฒนาที่สามารถนำมาใช้กับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	1. เมื่อมีซอฟต์แวร์สำเร็จรูปที่เป็นสินทรัพย์ขององค์กรซึ่งผ่านการคัดเลือกเพื่อเข้าสู่กระบวนการการปรับแต่งฯ หรือ 2. เมื่อต้องการวัดผลกระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา
ข้อมูลนำเข้า	1. <<สินทรัพย์>> ¹ ผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูป 2. <<เอกสาร>> ² ข้อกำหนดความต้องการซอฟต์แวร์ 3. <<เอกสาร>> วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	1. ผู้จัดการโครงการทำการพิจารณาข้อมูลนำเข้าของกระบวนการ 2. ผู้จัดการโครงการทำการเลือก ประเมิน และจัดกลุ่มแบบจำลองและกลยุทธ์ในการพัฒนาซอฟต์แวร์ที่สามารถนำมาใช้ได้กับโครงการ 3. ผู้จัดการโครงการทำกระบวนการระบุแบบจำลองที่ใช้สำหรับโครงการ 4. ผู้จัดการโครงการทำการระบุกลยุทธ์ในการพัฒนาที่ใช้สำหรับโครงการ ซึ่งประกอบด้วยกระบวนการ ทักษะ มาตรฐาน วิธีเครื่องมือและภาษาโปรแกรมที่ใช้ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> แบบจำลองและกลยุทธ์ในการพัฒนา
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

¹<<สินทรัพย์>> สัญลักษณ์ แสดงสินทรัพย์ที่อยู่ในสายผลิตภัณฑ์ซึ่งถูกเลือกนำมาใช้กับกระบวนการ

²<<เอกสาร>> สัญลักษณ์ แสดงเอกสารที่เป็นผลลัพธ์มาจากกิจกรรมอื่น

ตารางที่ 4.2 กระบวนการกำหนดแผนการปรับแต่ง (Define tailoring plan)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	กำหนดแผนการปรับแต่ง
วัตถุประสงค์	เพื่อให้ได้แผนการปรับแต่งที่ใช้สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา หรือ 2. เมื่อต้องการวัดผลกระบวนการกำหนดแผนการปรับแต่ง
ข้อมูลนำเข้า	<<เอกสาร>> แบบจำลองและกลยุทธ์ในการพัฒนา
กระบวนการ	ผู้จัดการโครงการทำการกำหนดแผนการปรับแต่งซอฟต์แวร์สำเร็จรูปตามแบบจำลองและกลยุทธ์ในการพัฒนาที่ได้ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> แผนการปรับแต่งซอฟต์แวร์สำเร็จรูป
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.3 กระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Review software product lines repository)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อพิจารณา จำแนกคุณลักษณะและนำเอาสินทรัพย์ที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ขององค์กรมาใช้ซ้ำ
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการกำหนดแผนการปรับแต่ง หรือ 2. เมื่อต้องการวัดผลกระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	1. <<เอกสาร>> ข้อกำหนดความต้องการซอฟต์แวร์ 2. <<เอกสาร>> แผนการปรับแต่งซอฟต์แวร์สำเร็จรูป

ตารางที่ 4.3 กระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Review software product lines repository) (ต่อ)

หัวข้อ	คำอธิบาย
กระบวนการ	<ol style="list-style-type: none"> 1. ผู้จัดการโครงการทำการพิจารณาข้อกำหนดความต้องการและขอบเขตของสายผลิตภัณฑ์ซอฟต์แวร์ เพื่อหาขอบเขตของสินทรัพย์ที่สามารถนำมาใช้ซ้ำสำหรับโครงการได้ 2. ผู้จัดการโครงการทำการระบุสินทรัพย์ที่สามารถนำมาประยุกต์ใช้ซ้ำกับโครงการได้ 3. ผู้จัดการโครงการทำการจำแนกคุณลักษณะตามลำดับความสำคัญของสินทรัพย์ที่สามารถนำมาใช้ซ้ำได้ โดยแบ่งเป็นคุณลักษณะหลัก (Core feature) คุณลักษณะตัวเลือก (Optional feature) และคุณลักษณะทางเลือก (Alternative feature) ตามลำดับ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	<ol style="list-style-type: none"> 1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> สินทรัพย์ที่สามารถนำมาใช้ซ้ำ
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.4 กระบวนการประเมินข้อกำหนดความต้องการ (Evaluate requirements specification)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ประเมินข้อกำหนดความต้องการ
วัตถุประสงค์	เพื่อประเมินแผนการปรับแต่งซอฟต์แวร์สำเร็จรูปและสินทรัพย์ที่สามารถนำมาใช้ซ้ำเทียบกับข้อกำหนดความต้องการซอฟต์แวร์และเกณฑ์การใช้ซ้ำ
เกณฑ์การเข้า	<ol style="list-style-type: none"> 1. เมื่อสิ้นสุดกระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการประเมินข้อกำหนดความต้องการ
ข้อมูลนำเข้า	<ol style="list-style-type: none"> 1. <<เอกสาร>> แผนการปรับแต่งซอฟต์แวร์สำเร็จรูป 2. <<เอกสาร>> สินทรัพย์ที่สามารถนำมาใช้ซ้ำ 3. <<เอกสาร>> เกณฑ์การใช้ซ้ำ

ตารางที่ 4.4 กระบวนการประเมินข้อกำหนดความต้องการ (Evaluate requirements specification) (ต่อ)

หัวข้อ	คำอธิบาย
กระบวนการ	ทีมผู้ประเมินทำการประเมินแผนการปรับแต่งซอฟต์แวร์สำเร็จรูปสินทรัพย์ที่สามารถนำมาใช้ซ้ำ โดยพิจารณาว่าครอบคลุมและสามารถตอบสนองต่อข้อกำหนดความต้องการซอฟต์แวร์รวมถึงเกณฑ์การใช้ซ้ำขององค์กรหรือไม่และมีรายการใดที่ต้องทำการปรับปรุงแก้ไข พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการประเมินข้อกำหนดความต้องการ
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้ประเมิน

4.3.1.2 ขั้นตอนออกแบบสถาปัตยกรรมการปรับแต่งซอฟต์แวร์สำเร็จรูป

ขั้นตอนนี้ประกอบด้วย 3 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.5-4.7

ตารางที่ 4.5 กระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อให้ได้มาซึ่งสถาปัตยกรรมซอฟต์แวร์สำหรับใช้ในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์โดยกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการประเมินข้อกำหนดความต้องการ หรือ 2. เมื่อต้องการวัดผลกระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	1. <<เอกสาร>> แผนการปรับแต่งซอฟต์แวร์สำเร็จรูป 2. <<เอกสาร>> สินทรัพย์ที่สามารถนำมาใช้ซ้ำ 3. <<เอกสาร>> ผลการประเมินข้อกำหนดความต้องการ

ตารางที่ 4.5 กระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture) (ต่อ)

หัวข้อ	คำอธิบาย
กระบวนการ	ผู้จัดการโครงการทำการกำหนดสถาปัตยกรรมซอฟต์แวร์โดยพิจารณาจากแผนการปรับแต่งซอฟต์แวร์สำเร็จรูป สิ้นทรัพย์ที่สามารถนำมาใช้ซ้ำและผลการประเมินข้อกำหนดความต้องการ ซึ่งสถาปัตยกรรมต้องประกอบด้วยส่วนประกอบ Kernel, Optional และ Variant โดยจำแนกสถาปัตยกรรมออกเป็น 2 ส่วน คือ 1) สถาปัตยกรรมเชิงโครงสร้าง แสดงถึงโครงสร้างของสถาปัตยกรรมซอฟต์แวร์ทั้งหมด โดยมีตัวอย่างแบบรูปสถาปัตยกรรม เช่น Kernel Architectural Pattern และ Client/Server Architectural Pattern เป็นต้น 2) สถาปัตยกรรมเชิงพลวัต แสดงการสื่อสารกันของแต่ละส่วนประกอบ ซึ่งอยู่ในรูปแบบของแผนภาพลำดับ โดยมีตัวอย่างแบบรูปสถาปัตยกรรม เช่น Communication Pattern และ Broker Communication Pattern เป็นต้น พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> สถาปัตยกรรมซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.6 กระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture details)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อให้ได้มาซึ่งรายละเอียดส่วนประกอบของสถาปัตยกรรมซอฟต์แวร์ที่ใช้สำหรับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์โดยกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	<<เอกสาร>> สถาปัตยกรรมซอฟต์แวร์
กระบวนการ	ผู้จัดการโครงการทำการกำหนดรายละเอียดสถาปัตยกรรมซอฟต์แวร์โดยพิจารณาจากสถาปัตยกรรมซอฟต์แวร์ที่ได้ เช่น Client Component, User Interface Component และ Server Component เป็นต้น พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> รายละเอียดสถาปัตยกรรมซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.7 กระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Evaluate software product lines architecture)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อทำการประเมินสถาปัตยกรรมซอฟต์แวร์ รวมถึงรายละเอียดของสถาปัตยกรรมซอฟต์แวร์ที่ใช้สำหรับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์โดยกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป

ตารางที่ 4.7 กระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Evaluate software product lines architecture) (ต่อ)

หัวข้อ	คำอธิบาย
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	1. <<เอกสาร>> สถาปัตยกรรมซอฟต์แวร์ 2. <<เอกสาร>> รายละเอียดสถาปัตยกรรมซอฟต์แวร์ 3. <<เอกสาร>> เกณฑ์การให้ซ้ำ
กระบวนการ	ทีมผู้ประเมินทำการประเมินสถาปัตยกรรมซอฟต์แวร์ รวมถึงรายละเอียดสถาปัตยกรรมโดยเทียบกับเกณฑ์การให้ซ้ำว่ามีความสอดคล้องกันหรือไม่และมีรายการใดที่ต้องทำการปรับปรุงแก้ไขพร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้ประเมิน

4.3.1.3 ขั้นตอนปรับแต่งซอฟต์แวร์สำเร็จรูป

ขั้นตอนนี้ประกอบด้วย 7 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.8-4.14

ตารางที่ 4.8 กระบวนการพัฒนาส่วนประกอบ (Implement component)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	พัฒนาส่วนประกอบ
วัตถุประสงค์	เพื่อปรับแต่งซอฟต์แวร์สำเร็จรูปโดยสินทรัพย์ให้สามารถตอบสนองต่อความต้องการซอฟต์แวร์ขององค์กร

ตารางที่ 4.8 กระบวนการพัฒนาส่วนประกอบ (Implement component) (ต่อ)

หัวข้อ	คำอธิบาย
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการพัฒนาส่วนประกอบ
ข้อมูลนำเข้า	1. <<เอกสาร>> สถาปัตยกรรมซอฟต์แวร์ 2. <<เอกสาร>> รายละเอียดสถาปัตยกรรมซอฟต์แวร์ 3. <<เอกสาร>> ผลการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	ทีมผู้พัฒนาทำการพัฒนาปรับแต่งซอฟต์แวร์สำเร็จรูป โดยใช้วิธีการ Kernel first approach ซึ่งหมายถึงซอฟต์แวร์สำเร็จรูปที่ใช้เป็นคุณลักษณะหลักในการปรับแต่ง หลังจากนั้นทีมผู้พัฒนาทำการพัฒนาซอฟต์แวร์โดยการเลือกเอาสินทรัพย์ที่ถูกกำหนดให้เป็นคุณลักษณะตัวเลือกและคุณลักษณะทางเลือกมาประกอบเข้ากับซอฟต์แวร์สำเร็จรูป โดยเพิ่มตามลำดับความสำคัญของคุณลักษณะ เมื่อได้ผลิตภัณฑ์ซอฟต์แวร์ซึ่งเกิดจากการพัฒนาส่วนประกอบแล้วให้จัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.9 กระบวนการประเมินการพัฒนาส่วนประกอบ (Evaluate implementation component)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ประเมินการพัฒนาส่วนประกอบ
วัตถุประสงค์	เพื่อทำการประเมินผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากการปรับแต่งซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการพัฒนาส่วนประกอบ หรือ 2. เมื่อต้องการวัดผลกระบวนการประเมินการพัฒนาส่วนประกอบ

ตารางที่ 4.9 กระบวนการประเมินการพัฒนาส่วนประกอบ (Evaluate implementation component) (ต่อ)

หัวข้อ	คำอธิบาย
ข้อมูลนำเข้า	1. ผลิตภัณฑ์ซอฟต์แวร์ 2. <<เอกสาร>> เกณฑ์การใช้ซ้ำ 3. <<เอกสาร>> ข้อกำหนดความต้องการ
กระบวนการ	ทีมผู้ประเมินทำการประเมินผลิตภัณฑ์ซอฟต์แวร์ที่ได้เทียบกับข้อกำหนดความต้องการซอฟต์แวร์และเกณฑ์การใช้ซ้ำ และพิจารณาว่ามีรายการใดที่ต้องทำการปรับปรุงแก้ไขพร้อมจัดทำเอกสารบันทึกผล โดยในกรณีที่ผลิตภัณฑ์ซอฟต์แวร์ที่ได้นั้นไม่ครอบคลุมความต้องการซอฟต์แวร์จะสิ้นสุดกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป เพื่อเข้าสู่กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปต่อไป
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการประเมินการพัฒนาส่วนประกอบ
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้ประเมิน

ตารางที่ 4.10 กระบวนการบูรณาการกับระบบที่มีอยู่ (Integrate with existing system)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	บูรณาการกับระบบที่มีอยู่
วัตถุประสงค์	เพื่อทำการบูรณาการผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากการปรับแต่งซอฟต์แวร์สำเร็จรูปเข้ากับระบบหรือสภาพแวดล้อมที่มีอยู่เดิมขององค์กรให้สามารถทำงานร่วมกันได้
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการประเมินการพัฒนาส่วนประกอบ หรือ 2. เมื่อต้องการวัดผลกระบวนการบูรณาการกับระบบที่มีอยู่
ข้อมูลนำเข้า	1. ผลิตภัณฑ์ซอฟต์แวร์ 2. <<เอกสาร>> ผลการประเมินการพัฒนาส่วนประกอบ
กระบวนการ	ทีมผู้พัฒนาทำการบูรณาการผลิตภัณฑ์ซอฟต์แวร์ที่ได้กับระบบและสภาพแวดล้อมที่มีอยู่เดิม เพื่อให้ผลิตภัณฑ์ซอฟต์แวร์สามารถใช้งานร่วมกันกับระบบที่มีอยู่เดิมได้ พร้อมจัดทำเอกสารบันทึกผล

ตารางที่ 4.10 กระบวนการบูรณาการกับระบบที่มีอยู่ (Integrate with existing system) (ต่อ)

หัวข้อ	คำอธิบาย
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.11 กระบวนการประเมินการบูรณาการกับระบบที่มีอยู่ (Evaluate existing system integration)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ประเมินการบูรณาการกับระบบที่มีอยู่
วัตถุประสงค์	เพื่อประเมินการบูรณาการผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากการปรับแต่งซอฟต์แวร์สำเร็จรูปเข้ากับระบบหรือสภาพแวดล้อมที่มีอยู่เดิมขององค์กร
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการบูรณาการกับระบบที่มีอยู่ หรือ 2. เมื่อต้องการวัดผลกระบวนการประเมินการบูรณาการกับระบบที่มีอยู่
ข้อมูลนำเข้า	1. ผลิตภัณฑ์ซอฟต์แวร์ 2. <<เอกสาร>> เกณฑ์การให้ซ้ำ
กระบวนการ	ทีมผู้ประเมินทำการประเมินการบูรณาการผลิตภัณฑ์ซอฟต์แวร์กับระบบที่มีอยู่เดิมเทียบกับเกณฑ์การให้ซ้ำ ว่าสามารถตอบสนองต่อความต้องการซอฟต์แวร์หรือไม่และพิจารณาว่ามีรายการใดที่ต้องทำการปรับปรุงแก้ไข พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการประเมินการบูรณาการกับระบบที่มีอยู่
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้ประเมิน

ตารางที่ 4.12 กระบวนการทดสอบสายผลิตภัณฑ์ (Test product lines)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ทดสอบสายผลิตภัณฑ์
วัตถุประสงค์	เพื่อหาข้อผิดพลาดที่เกิดขึ้นกับผลิตภัณฑ์ซอฟต์แวร์ และปรับปรุงแก้ไขผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากการปรับแต่งซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการประเมินการบูรณาการกับระบบที่มีอยู่ หรือ 2. เมื่อต้องการวัดผลกระบวนการทดสอบสายผลิตภัณฑ์
ข้อมูลนำเข้า	1. ผลิตภัณฑ์ซอฟต์แวร์ 2. <<เอกสาร>> ผลการประเมินการบูรณาการกับระบบที่มีอยู่
กระบวนการ	ทีมผู้พัฒนาทำการทดสอบผลิตภัณฑ์ซอฟต์แวร์ที่ได้ โดยประกอบด้วย การทดสอบการบูรณาการ (Integration test) เพื่อทดสอบว่าผลิตภัณฑ์ซอฟต์แวร์มีความเข้ากันได้หรือไม่และการทดสอบเชิงหน้าที่ (Functional test) เพื่อทดสอบว่าผลิตภัณฑ์ซอฟต์แวร์ตรงกับความต้องการของซอฟต์แวร์ที่กำหนดไว้หรือไม่ โดยทีมผู้พัฒนาทำการทดสอบผลิตภัณฑ์ซอฟต์แวร์ตามลำดับความสำคัญของคุณลักษณะ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.13 กระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป (Manage COTS Version)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	จัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป
วัตถุประสงค์	เพื่อระบุเวอร์ชันของผลิตภัณฑ์ซอฟต์แวร์และสารสนเทศที่เกี่ยวข้อง
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการทดสอบสายผลิตภัณฑ์ หรือ 2. เมื่อต้องการวัดผลกระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป

ตารางที่ 4.13 กระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป (Manage COTS Version) (ต่อ)

หัวข้อ	คำอธิบาย
ข้อมูลนำเข้า	ผลิตภัณฑ์ซอฟต์แวร์
กระบวนการงาน	ทีมผู้พัฒนาทำการกำหนดเวอร์ชันให้กับผลิตภัณฑ์ซอฟต์แวร์และระบุข้อมูลสารสนเทศ ข้อมูลที่เกี่ยวข้องรวมถึงการจัดทำเอกสารเพื่ออำนวยความสะดวกนำมาใช้ประโยชน์ในครั้งต่อไป
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.14 กระบวนการปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Update COTS in software product lines repository)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อนำเอาผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป เก็บไว้เป็นสินทรัพย์ใหม่ที่อยู่ในสายผลิตภัณฑ์
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป หรือ 2. เมื่อต้องการวัดผลกระบวนการปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	ผลิตภัณฑ์ซอฟต์แวร์
กระบวนการงาน	ทีมผู้พัฒนาทำการจัดเก็บผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปไว้ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์เพื่อรอโอกาสที่จะนำกลับมาใช้ซ้ำในอนาคต พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

4.3.1.4 ขั้นตอนติดตั้งซอฟต์แวร์

ขั้นตอนนี้ประกอบด้วย 2 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียด
ดังแสดงในตารางที่ 4.15-4.16

ตารางที่ 4.15 กระบวนการวางแผนการติดตั้ง (Establish installation plan)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	วางแผนการติดตั้ง
วัตถุประสงค์	เพื่อให้ได้มาซึ่งแผนสำหรับการติดตั้งผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการวางแผนการติดตั้ง
ข้อมูลนำเข้า	ผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	ทีมผู้พัฒนาทำการวางแผนการติดตั้งสำหรับผลิตภัณฑ์ซอฟต์แวร์ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> แผนการติดตั้ง
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.16 กระบวนการประเมินแผนการติดตั้ง (Evaluate installation plan)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ประเมินแผนการติดตั้ง
วัตถุประสงค์	เพื่อประเมินแผนสำหรับการติดตั้งผลิตภัณฑ์ซอฟต์แวร์ให้ได้มาซึ่งแผนการติดตั้งที่มีความสมบูรณ์ที่สุด
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการวางแผนการติดตั้ง หรือ 2. เมื่อต้องการวัดผลกระบวนการประเมินแผนการติดตั้ง
ข้อมูลนำเข้า	1. <<เอกสาร>> แผนการติดตั้ง 2. <<เอกสาร>> เกณฑ์การให้ซ้ำ
กระบวนการ	ทีมผู้ประเมินทำการประเมินแผนการติดตั้งเทียบกับเกณฑ์การให้ซ้ำและพิจารณาว่ามีรายการใดที่ต้องทำการปรับปรุงแก้ไข พร้อมจัดทำเอกสารบันทึกผล

ตารางที่ 4.16 กระบวนการประเมินแผนการติดตั้ง (Evaluate installation plan) (ต่อ)

หัวข้อ	คำอธิบาย
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการประเมินแผนการติดตั้ง
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้ประเมิน

4.3.1.5 ขั้นตอนสนับสนุนการยอมรับซอฟต์แวร์

ขั้นตอนนี้ประกอบด้วย 2 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.17-4.18

ตารางที่ 4.17 กระบวนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์ (Deliver software product)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ส่งมอบผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อทำการส่งมอบผลิตภัณฑ์ซอฟต์แวร์ให้กับผู้ใช้
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการประเมินแผนการติดตั้ง หรือ 2. เมื่อต้องการวัดผลกระบวนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	1. ผลิตภัณฑ์ซอฟต์แวร์ 2. <<เอกสาร>> แผนการติดตั้ง 3. <<เอกสาร>> ผลการประเมินแผนการติดตั้ง
กระบวนการ	ทีมผู้พัฒนาซอฟต์แวร์ทำการส่งมอบผลิตภัณฑ์ซอฟต์แวร์ให้กับผู้ใช้
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.18 กระบวนการจัดการสนับสนุนการฝึกอบรม (Provide training support)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	จัดการสนับสนุนการฝึกอบรม
วัตถุประสงค์	เพื่อสนับสนุนการฝึกอบรมการใช้งานผลิตภัณฑ์ซอฟต์แวร์ให้กับผู้ใช้ให้สามารถใช้งานผลิตภัณฑ์ซอฟต์แวร์ได้อย่างมีประสิทธิภาพ
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการจัดการสนับสนุนการฝึกอบรม
ข้อมูลนำเข้า	ผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	ทีมผู้พัฒนาเตรียมการสำหรับการฝึกอบรมการใช้งานผลิตภัณฑ์ซอฟต์แวร์ที่ได้ส่งมอบให้กับผู้ใช้ เพื่อให้สามารถใช้งานผลิตภัณฑ์ซอฟต์แวร์อย่างมีประสิทธิภาพ พร้อมทั้งจัดทำเอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

4.3.2 กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป

กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปประกอบด้วย 5 ขั้นตอน ซึ่งสามารถอธิบายในรายละเอียดได้ดังต่อไปนี้

4.3.2.1 ขั้นตอนวางแผนโครงการการบูรณาการซอฟต์แวร์สำเร็จรูป

ขั้นตอนนี้ประกอบด้วย 4 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.19-4.22

ตารางที่ 4.19 กระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา (Define model and development strategies)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	กำหนดแบบจำลองและกลยุทธ์ในการพัฒนา
วัตถุประสงค์	เพื่อให้ได้แบบจำลองและกลยุทธ์ในการพัฒนาที่สามารถนำมาใช้กับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	<ol style="list-style-type: none"> เมื่อกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปไม่สามารถทำได้มาซึ่งผลิตภัณฑ์ซอฟต์แวร์ที่ตอบสนองต่อความต้องการซอฟต์แวร์ได้ หรือ เมื่อต้องการวัดผลกระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา
ข้อมูลนำเข้า	<ol style="list-style-type: none"> <<สินทรัพย์>> ผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูป <<เอกสาร>> ข้อกำหนดความต้องการซอฟต์แวร์ <<เอกสาร>> วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	<ol style="list-style-type: none"> ผู้จัดการโครงการทำการพิจารณาข้อมูลนำเข้าของกระบวนการ ผู้จัดการโครงการทำการเลือก ประเมิน และจัดกลุ่มแบบจำลองและกลยุทธ์ในการพัฒนาซอฟต์แวร์ที่สามารถนำมาใช้ได้กับโครงการ ผู้จัดการโครงการทำการระบุแบบจำลองที่ใช้สำหรับโครงการ ผู้จัดการโครงการทำการระบุกลยุทธ์ในการพัฒนาที่ใช้สำหรับโครงการ ซึ่งประกอบด้วยกระบวนการ ทักษะ มาตรฐาน วิธีเครื่องมือและภาษาโปรแกรมที่ใช้ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	<ol style="list-style-type: none"> <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ <<เอกสาร>> แบบจำลองและกลยุทธ์ในการพัฒนา
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.20 กระบวนการกำหนดแผนการบูรณาการ (Define integration plan)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	กำหนดแผนการบูรณาการ
วัตถุประสงค์	เพื่อให้ได้แผนการบูรณาการที่ใช้สำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา หรือ 2. เมื่อต้องการวัดผลกระบวนการกำหนดแผนการบูรณาการ
ข้อมูลนำเข้า	<<เอกสาร>> แบบจำลองและกลยุทธ์ในการพัฒนา
กระบวนการ	ผู้จัดการโครงการทำการกำหนดแผนการบูรณาการซอฟต์แวร์สำเร็จรูปตามแบบจำลองและกลยุทธ์ในการพัฒนาที่ได้ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> แผนการบูรณาการซอฟต์แวร์สำเร็จรูป
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.21 กระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Review software product lines repository)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อพิจารณา จำแนกคุณลักษณะและนำเอาสินทรัพย์ที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ขององค์กรมาใช้ซ้ำ
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการกำหนดแผนการบูรณาการ หรือ 2. เมื่อต้องการวัดผลกระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	1. <<เอกสาร>> ข้อกำหนดความต้องการซอฟต์แวร์ 2. <<เอกสาร>> แผนการบูรณาการซอฟต์แวร์สำเร็จรูป

ตารางที่ 4.21 กระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Review software product lines repository) (ต่อ)

หัวข้อ	คำอธิบาย
กระบวนการ	<ol style="list-style-type: none"> 1. ผู้จัดการโครงการทำการพิจารณาข้อกำหนดความต้องการและขอบเขตของสายผลิตภัณฑ์ซอฟต์แวร์ เพื่อหาขอบเขตของสินทรัพย์ที่สามารถนำมาใช้ซ้ำสำหรับโครงการได้ 2. ผู้จัดการโครงการทำการระบุสินทรัพย์ที่สามารถนำมาประยุกต์ใช้ซ้ำกับโครงการได้ 3. ผู้จัดการโครงการทำการจำแนกคุณลักษณะตามลำดับความสำคัญของสินทรัพย์ที่สามารถนำมาใช้ซ้ำได้ โดยแบ่งเป็นคุณลักษณะหลัก (Core feature) คุณลักษณะตัวเลือก (Optional feature) และคุณลักษณะทางเลือก (Alternative feature) ตามลำดับ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	<ol style="list-style-type: none"> 1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> สินทรัพย์ที่สามารถนำมาใช้ซ้ำ
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.22 กระบวนการประเมินข้อกำหนดความต้องการ (Evaluate requirements specification)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ประเมินข้อกำหนดความต้องการ
วัตถุประสงค์	เพื่อประเมินแผนการบูรณาการซอฟต์แวร์สำเร็จรูปและสินทรัพย์ที่สามารถนำมาใช้ซ้ำเทียบกับข้อกำหนดความต้องการซอฟต์แวร์และเกณฑ์การใช้ซ้ำ
เกณฑ์การเข้า	<ol style="list-style-type: none"> 1. เมื่อสิ้นสุดกระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการประเมินข้อกำหนดความต้องการ
ข้อมูลนำเข้า	<ol style="list-style-type: none"> 1. <<เอกสาร>> แผนการบูรณาการซอฟต์แวร์สำเร็จรูป 2. <<เอกสาร>> สินทรัพย์ที่สามารถนำมาใช้ซ้ำ 3. <<เอกสาร>> เกณฑ์การใช้ซ้ำ

ตารางที่ 4.22 กระบวนการประเมินข้อกำหนดความต้องการ (Evaluate requirements specification) (ต่อ)

หัวข้อ	คำอธิบาย
กระบวนการ	ทีมผู้ประเมินทำการประเมินแผนการบูรณาการซอฟต์แวร์สำเร็จรูปสินทรัพย์ที่สามารถนำมาใช้ซ้ำ โดยพิจารณาว่าครอบคลุมและสามารถตอบสนองต่อข้อกำหนดความต้องการซอฟต์แวร์รวมถึงเกณฑ์การใช้ซ้ำขององค์กรหรือไม่และมีรายการใดที่ต้องทำการปรับปรุงแก้ไข พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการประเมินข้อกำหนดความต้องการ
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้ประเมิน

4.3.2.2 ขั้นตอนออกแบบสถาปัตยกรรมการบูรณาการซอฟต์แวร์สำเร็จรูป

ขั้นตอนนี้ประกอบด้วย 3 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.23-4.25

ตารางที่ 4.23 กระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อให้ได้มาซึ่งสถาปัตยกรรมซอฟต์แวร์สำหรับใช้ในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์โดยกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการประเมินข้อกำหนดความต้องการ หรือ 2. เมื่อต้องการวัดผลกระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	1. <<เอกสาร>> แผนการบูรณาการซอฟต์แวร์สำเร็จรูป 2. <<เอกสาร>> สินทรัพย์ที่สามารถนำมาใช้ซ้ำ 3. <<เอกสาร>> ผลการประเมินข้อกำหนดความต้องการ

ตารางที่ 4.23 กระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture) (ต่อ)

หัวข้อ	คำอธิบาย
กระบวนการงาน	ผู้จัดการโครงการทำการกำหนดสถาปัตยกรรมซอฟต์แวร์โดยพิจารณาจากแผนการบูรณาการซอฟต์แวร์สำเร็จรูป สิ้นทรัพย์ที่สามารถนำมาใช้ซ้ำและผลการประเมินข้อกำหนดความต้องการ ซึ่งสถาปัตยกรรมต้องประกอบด้วยส่วนประกอบ Kernel, Optional และ Variant โดยจำแนกสถาปัตยกรรมออกเป็น 2 ส่วน คือ 1) สถาปัตยกรรมเชิงโครงสร้าง แสดงถึงโครงสร้างของสถาปัตยกรรมซอฟต์แวร์ทั้งหมด โดยมีตัวอย่างแบบรูปสถาปัตยกรรม เช่น Kernel Architectural Pattern และ Client/Server Architectural Pattern เป็นต้น 2) สถาปัตยกรรมเชิงพลวัต แสดงการสื่อสารกันของแต่ละส่วนประกอบ ซึ่งอยู่ในรูปแบบของแผนภาพลำดับ โดยมีตัวอย่างแบบรูปสถาปัตยกรรม เช่น Communication Pattern และ Broker Communication Pattern เป็นต้น พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> สถาปัตยกรรมซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.24 กระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture details)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อให้ได้มาซึ่งรายละเอียดส่วนประกอบของสถาปัตยกรรมซอฟต์แวร์ที่ใช้สำหรับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสิ้นทรัพย์โดยกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป

ตารางที่ 4.24 กระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture details) (ต่อ)

หัวข้อ	คำอธิบาย
เกณฑ์การเข้า	<ol style="list-style-type: none"> 1. เมื่อสิ้นสุดกระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	<<เอกสาร>> สถาปัตยกรรมซอฟต์แวร์
กระบวนการ	<p>ผู้จัดการโครงการทำการกำหนดรายละเอียดสถาปัตยกรรมซอฟต์แวร์โดยพิจารณาจากสถาปัตยกรรมซอฟต์แวร์ที่ได้ เช่น Client Component, User Interface Component และ Server Component เป็นต้น นอกจากนี้ผู้จัดการโครงการทำการออกแบบทอพอโลยีการเชื่อมต่อสำหรับการเชื่อมโยงกันระหว่างซอฟต์แวร์สำเร็จรูป สินทรัพย์ที่สามารถนำมาใช้ซ้ำ และซอฟต์แวร์ที่ได้จากการพัฒนารหัสคำสั่งเพิ่มเติม เพื่อบูรณาการซอฟต์แวร์สำเร็จรูปให้สามารถทำงานร่วมกันได้ เช่น กรอบงานอาร์ชแวร์ (Archware framework) พร้อมจัดทำเอกสารบันทึกผล</p>
ข้อมูลนำออก	<ol style="list-style-type: none"> 1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> รายละเอียดสถาปัตยกรรมซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.25 กระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Evaluate software product lines architecture)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อทำการประเมินสถาปัตยกรรมซอฟต์แวร์ รวมถึงรายละเอียดของสถาปัตยกรรมซอฟต์แวร์ที่ใช้สำหรับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์โดยกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป

ตารางที่ 4.25 กระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Evaluate software product lines architecture) (ต่อ)

หัวข้อ	คำอธิบาย
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	1. <<เอกสาร>> สถาปัตยกรรมซอฟต์แวร์ 2. <<เอกสาร>> รายละเอียดสถาปัตยกรรมซอฟต์แวร์ 3. <<เอกสาร>> เกณฑ์การให้ซ้ำ
กระบวนการ	ทีมผู้ประเมินทำการประเมินสถาปัตยกรรมซอฟต์แวร์ รวมถึงรายละเอียดสถาปัตยกรรมโดยเทียบกับเกณฑ์การให้ซ้ำว่ามีความสอดคล้องกันหรือไม่และมีรายการใดที่ต้องทำการปรับปรุงแก้ไขพร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้ประเมิน

4.3.2.3 ขั้นตอนบูรณาการซอฟต์แวร์สำเร็จรูป

ขั้นตอนนี้ประกอบด้วย 7 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.26-4.32

ตารางที่ 4.26 กระบวนการพัฒนาส่วนประกอบ (Implement component)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	พัฒนาส่วนประกอบ
วัตถุประสงค์	เพื่อบูรณาการซอฟต์แวร์สำเร็จรูปโดยสินทรัพย์ให้สามารถตอบสนองต่อความต้องการซอฟต์แวร์ขององค์กร

ตารางที่ 4.26 กระบวนการพัฒนาส่วนประกอบ (Implement component) (ต่อ)

หัวข้อ	คำอธิบาย
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการพัฒนาส่วนประกอบ
ข้อมูลนำเข้า	1. <<เอกสาร>> สถาปัตยกรรมซอฟต์แวร์ 2. <<เอกสาร>> รายละเอียดสถาปัตยกรรมซอฟต์แวร์ 3. <<เอกสาร>> ผลการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	ทีมผู้พัฒนาทำการพัฒนารหัสคำสั่งเพิ่มเติม (Custom code) ที่ใช้ในการบูรณาการซอฟต์แวร์สำเร็จรูปเพื่อให้ได้ผลิตภัณฑ์ซอฟต์แวร์ที่ครอบคลุมความต้องการ โดยทีมผู้พัฒนาใช้วิธี Kernel first approach คือ การพัฒนาซอฟต์แวร์โดยเริ่มจากส่วนที่ทำหน้าที่คุณลักษณะหลักก่อนจากนั้นจึงสร้างส่วนที่ทำหน้าที่คุณลักษณะตัวเลือกและคุณลักษณะทางเลือกตามลำดับความสำคัญ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.27 กระบวนการประเมินการพัฒนาส่วนประกอบ (Evaluate implementation component)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ประเมินการพัฒนาส่วนประกอบ
วัตถุประสงค์	เพื่อทำการประเมินผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากการบูรณาการซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการพัฒนาส่วนประกอบ หรือ 2. เมื่อต้องการวัดผลกระบวนการประเมินการพัฒนาส่วนประกอบ
ข้อมูลนำเข้า	1. ผลิตภัณฑ์ซอฟต์แวร์ 2. <<เอกสาร>> เกณฑ์การให้ค่า 3. <<เอกสาร>> ข้อกำหนดความต้องการ

ตารางที่ 4.27 กระบวนการประเมินการพัฒนาส่วนประกอบ (Evaluate implementation component) (ต่อ)

หัวข้อ	คำอธิบาย
กระบวนการ	ทีมผู้ประเมินพิจารณาประเมินผลิตภัณฑ์ซอฟต์แวร์ เทียบกับข้อกำหนดความต้องการซอฟต์แวร์และเกณฑ์การใช้ซ้ำ และพิจารณาว่ามีรายการใดที่ต้องทำการปรับปรุงแก้ไขพร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการประเมินการพัฒนาส่วนประกอบ
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้ประเมิน

ตารางที่ 4.28 กระบวนการบูรณาการระบบและซอฟต์แวร์ (Software and system integration)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	บูรณาการระบบและซอฟต์แวร์
วัตถุประสงค์	เพื่อทำการบูรณาการผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากการบูรณาการซอฟต์แวร์สำเร็จรูปเข้ากับระบบหรือสภาพแวดล้อมที่มีอยู่เดิมขององค์กรให้สามารถทำงานร่วมกันได้
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการประเมินการพัฒนาส่วนประกอบ หรือ 2. เมื่อต้องการวัดผลกระบวนการบูรณาการระบบและซอฟต์แวร์
ข้อมูลนำเข้า	1. ผลิตภัณฑ์ซอฟต์แวร์ 2. <<เอกสาร>> ผลการประเมินการพัฒนาส่วนประกอบ
กระบวนการ	ทีมผู้พัฒนาทำการบูรณาการผลิตภัณฑ์ซอฟต์แวร์ โดยนำเอาซอฟต์แวร์สำเร็จรูป สินทรัพย์ที่สามารถนำมาใช้ซ้ำ และรหัสคำสั่งเพิ่มเติม เข้าด้วยกันตามทอพอโลยีการเชื่อมต่อที่ได้กำหนดไว้พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.29 กระบวนการประเมินการบูรณาการระบบและซอฟต์แวร์ (Evaluate software and system integration)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ประเมินการบูรณาการระบบและซอฟต์แวร์
วัตถุประสงค์	เพื่อประเมินการบูรณาการผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากการบูรณาการซอฟต์แวร์สำเร็จรูปเข้ากับระบบหรือสภาพแวดล้อมที่มีอยู่เดิมขององค์กร
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการบูรณาการระบบและซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการประเมินการบูรณาการระบบและซอฟต์แวร์
ข้อมูลนำเข้า	1. ผลิตภัณฑ์ซอฟต์แวร์ 2. <<เอกสาร>> เกณฑ์การให้ซ้ำ
กระบวนการ	ทีมผู้ประเมินทำการประเมินผลิตภัณฑ์ซอฟต์แวร์ที่ได้เทียบกับเกณฑ์การให้ซ้ำและพิจารณาว่ามีรายการใดที่ต้องทำการปรับปรุงแก้ไขพร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการประเมินการบูรณาการระบบและซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้ประเมิน

ตารางที่ 4.30 กระบวนการทดสอบสายผลิตภัณฑ์ (Test product lines)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ทดสอบสายผลิตภัณฑ์
วัตถุประสงค์	เพื่อหาข้อผิดพลาดที่เกิดขึ้นกับผลิตภัณฑ์ซอฟต์แวร์ และปรับปรุงแก้ไขผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากการบูรณาการซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการประเมินการบูรณาการระบบและซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการทดสอบสายผลิตภัณฑ์

ตารางที่ 4.30 กระบวนการทดสอบสายผลิตภัณฑ์ (Test product lines) (ต่อ)

หัวข้อ	คำอธิบาย
ข้อมูลนำเข้า	1. ผลิตภัณฑ์ซอฟต์แวร์ 2. <<เอกสาร>> ผลการประเมินการบูรณาการระบบและซอฟต์แวร์
กระบวนการ	ทีมผู้พัฒนาทำการทดสอบผลิตภัณฑ์ซอฟต์แวร์ที่ได้ โดยประกอบด้วยทดสอบการบูรณาการ (Integration test) เพื่อทดสอบว่าผลิตภัณฑ์ซอฟต์แวร์มีความเข้ากันได้หรือไม่และการทดสอบเชิงหน้าที่ (Functional test) เพื่อทดสอบว่าผลิตภัณฑ์ซอฟต์แวร์ตรงกับความต้องการของซอฟต์แวร์ที่กำหนดไว้หรือไม่ โดยทีมผู้พัฒนาทำการทดสอบผลิตภัณฑ์ซอฟต์แวร์ตามลำดับความสำคัญของคุณลักษณะ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.31 กระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป (Manage COTS Version)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	จัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป
วัตถุประสงค์	เพื่อระบุเวอร์ชันของผลิตภัณฑ์ซอฟต์แวร์และสารสนเทศที่เกี่ยวข้อง
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการทดสอบสายผลิตภัณฑ์ หรือ 2. เมื่อต้องการวัดผลกระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป
ข้อมูลนำเข้า	ผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	ทีมผู้พัฒนาทำการกำหนดเวอร์ชันให้กับผลิตภัณฑ์ซอฟต์แวร์และระบุข้อมูลสารสนเทศ ข้อมูลที่เกี่ยวข้องรวมถึงการจัดทำเอกสารเพื่อถ่ายทอดการนำมาใช้ประโยชน์ในครั้งต่อไป

ตารางที่ 4.31 กระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป (Manage COTS Version) (ต่อ)

หัวข้อ	คำอธิบาย
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.32 กระบวนการปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Update COTS in software product lines repository)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อนำเอาผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป เก็บไว้เป็นสินทรัพย์ใหม่ที่อยู่ในสายผลิตภัณฑ์
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป หรือ 2. เมื่อต้องการวัดผลกระบวนการปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	ผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	ทีมผู้พัฒนาทำการจัดเก็บผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปไว้ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์เพื่อรอโอกาสที่จะนำกลับมาใช้ซ้ำในอนาคต พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

4.3.2.4 ขั้นตอนติดตั้งซอฟต์แวร์

ขั้นตอนนี้ประกอบด้วย 2 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.33-4.34

ตารางที่ 4.33 กระบวนการวางแผนการติดตั้ง (Establish installation plan)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	วางแผนการติดตั้ง
วัตถุประสงค์	เพื่อให้ได้มาซึ่งแผนสำหรับการติดตั้งผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บสายผลิตภัณฑ์ หรือ 2. เมื่อต้องการวัดผลกระบวนการวางแผนการติดตั้ง
ข้อมูลนำเข้า	ผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	ทีมผู้พัฒนาทำการวางแผนการติดตั้งสำหรับผลิตภัณฑ์ซอฟต์แวร์ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> แผนการติดตั้ง
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.34 กระบวนการประเมินแผนการติดตั้ง (Evaluate installation plan)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ประเมินแผนการติดตั้ง
วัตถุประสงค์	เพื่อประเมินแผนสำหรับการติดตั้งผลิตภัณฑ์ซอฟต์แวร์ให้ได้มาซึ่งแผนการติดตั้งที่มีความสมบูรณ์ที่สุด
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการวางแผนการติดตั้ง หรือ 2. เมื่อต้องการวัดผลกระบวนการประเมินแผนการติดตั้ง
ข้อมูลนำเข้า	1. <<เอกสาร>> แผนการติดตั้ง 2. <<เอกสาร>> เกณฑ์การให้ซ้ำ

ตารางที่ 4.34 กระบวนการประเมินแผนการติดตั้ง (Evaluate installation plan) (ต่อ)

หัวข้อ	คำอธิบาย
กระบวนการ	ทีมผู้ประเมินทำการประเมินแผนการติดตั้งเทียบกับเกณฑ์การใช้ ซ้ำและพิจารณาว่ามีรายการใดที่ต้องทำการปรับปรุงแก้ไข พร้อม จัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการประเมินแผนการติดตั้ง
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้ประเมิน

4.3.2.5 ขั้นตอนสนับสนุนการยอมรับซอฟต์แวร์

ขั้นตอนนี้ประกอบด้วย 2 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียด
ดังแสดงในตารางที่ 4.35-4.36

ตารางที่ 4.35 กระบวนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์ (Deliver software product)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ส่งมอบผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อทำการส่งมอบผลิตภัณฑ์ซอฟต์แวร์ให้กับผู้ใช้
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการประเมินแผนการติดตั้ง หรือ 2. เมื่อต้องการวัดผลกระบวนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	1. ผลิตภัณฑ์ซอฟต์แวร์ 2. <<เอกสาร>> แผนการติดตั้ง 3. <<เอกสาร>> ผลการประเมินแผนการติดตั้ง
กระบวนการ	ทีมผู้พัฒนาซอฟต์แวร์ทำการส่งมอบผลิตภัณฑ์ซอฟต์แวร์ให้กับ ผู้ใช้
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.36 กระบวนการจัดการสนับสนุนการฝึกอบรม (Provide training support)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	จัดการสนับสนุนการฝึกอบรม
วัตถุประสงค์	เพื่อสนับสนุนการฝึกอบรมการใช้งานผลิตภัณฑ์ซอฟต์แวร์ให้กับผู้ใช้ให้สามารถใช้งานผลิตภัณฑ์ซอฟต์แวร์ได้อย่างมีประสิทธิภาพ
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระบวนการจัดการสนับสนุนการฝึกอบรม
ข้อมูลนำเข้า	ผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	ทีมผู้พัฒนาเตรียมการสำหรับการฝึกอบรมการใช้งานผลิตภัณฑ์ซอฟต์แวร์ที่ได้ส่งมอบให้กับผู้ใช้ เพื่อให้สามารถใช้งานผลิตภัณฑ์ซอฟต์แวร์อย่างมีประสิทธิภาพ พร้อมทั้งจัดทำเอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

4.3.3 กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูปประกอบด้วย 3 ขั้นตอน ซึ่งสามารถอธิบายในรายละเอียดได้ดังต่อไปนี้

4.3.3.1 ขั้นตอนวางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

ขั้นตอนนี้ประกอบด้วย 2 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.37-4.38

ตารางที่ 4.37 กระบวนการกำหนดแผนการบำรุงรักษา (Define maintenance plan)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	กำหนดแผนการบำรุงรักษา
วัตถุประสงค์	เพื่อได้มาซึ่งแผนสำหรับการบำรุงรักษาซอฟต์แวร์สำเร็จรูปจากสินทรัพย์

ตารางที่ 4.37 กระบวนการกำหนดแผนการบำรุงรักษา (Define maintenance plan) (ต่อ)

หัวข้อ	คำอธิบาย
เกณฑ์การเข้า	1. เมื่อต้องการทำการบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์โดยมีรายงานปัญหา/คำร้องขอการดัดแปลง หรือ 2. เมื่อต้องการวัดผลกระบวนการกำหนดแผนการบำรุงรักษา
ข้อมูลนำเข้า	<<เอกสาร>> รายงานปัญหา/คำร้องขอการดัดแปลง
กระบวนการ	ผู้จัดการโครงการทำการวางแผนการบำรุงรักษาโดยพิจารณาจากรายงานปัญหาหรือคำร้องขอการดัดแปลง พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> แผนการบำรุงรักษา
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.38 กระบวนการสร้างกระบวนการบำรุงรักษา (Establish maintenance procedures)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	สร้างกระบวนการบำรุงรักษา
วัตถุประสงค์	เพื่อให้ได้มาซึ่งกระบวนการที่ใช้สำหรับการบำรุงรักษาซอฟต์แวร์สำเร็จรูปจากสินทรัพย์
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการกำหนดแผนการบำรุงรักษา หรือ 2. เมื่อต้องการวัดผลกระบวนการสร้างกระบวนการบำรุงรักษา
ข้อมูลนำเข้า	<<เอกสาร>> แผนการบำรุงรักษา
กระบวนการ	ผู้จัดการโครงการทำการสร้างกระบวนการสำหรับการบำรุงรักษาเพื่อตอบสนองต่อแผนการบำรุงรักษานั้น เช่น กระบวนการรับบันทึก ดัดแปลง ย้าย ยกเลิกหรือติดตามปัญหา เป็นต้น พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> กระบวนการบำรุงรักษา
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

4.3.3.2 ขั้นตอนวิเคราะห์ปัญหาและการดัดแปลง

ขั้นตอนนี้ประกอบด้วย 3 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.39-4.41

ตารางที่ 4.39 กระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Review software product lines repository)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
วัตถุประสงค์	เพื่อพิจารณา จำแนกคุณลักษณะและนำเอาสินทรัพย์ที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ขององค์กรมาใช้ซ้ำ
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการสร้างกระบวนการบำรุงรักษา หรือ 2. เมื่อต้องการวัดผลกระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
ข้อมูลนำเข้า	1. <<เอกสาร>> รายงานปัญหา/คำร้องขอการดัดแปลง 2. <<เอกสาร>> แผนการบำรุงรักษา 3. <<เอกสาร>> กระบวนการบำรุงรักษา
กระบวนการ	1. ผู้จัดการโครงการทำการพิจารณาข้อกำหนดความต้องการและขอบเขตของสายผลิตภัณฑ์ซอฟต์แวร์ เพื่อหาขอบเขตของสินทรัพย์ที่สามารถนำมาใช้ซ้ำสำหรับโครงการได้ 2. ผู้จัดการโครงการทำการระบุสินทรัพย์ที่สามารถนำมาประยุกต์ใช้ซ้ำกับโครงการได้ 3. ผู้จัดการโครงการทำการจำแนกคุณลักษณะตามลำดับความสำคัญของสินทรัพย์ที่สามารถนำมาใช้ซ้ำได้ โดยแบ่งเป็นคุณลักษณะหลัก (Core feature) คุณลักษณะตัวเลือก (Optional feature) และคุณลักษณะทางเลือก (Alternative feature) ตามลำดับ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> สินทรัพย์ที่สามารถนำมาใช้ซ้ำ
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.40 กระบวนการวิเคราะห์ผลกระทบสินทรัพย์ที่สามารถนำมาใช้ซ้ำ (Analyze reusable assets impact)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	วิเคราะห์ผลกระทบสินทรัพย์ที่สามารถนำมาใช้ซ้ำ
วัตถุประสงค์	เพื่อวิเคราะห์สินทรัพย์ที่สามารถนำมาใช้ซ้ำถึงผลกระทบ ความเสี่ยงที่อาจเกิดขึ้นได้
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ หรือ 2. เมื่อต้องการวัดผลกระทบจากการวิเคราะห์ผลกระทบสินทรัพย์ที่สามารถนำมาใช้ซ้ำ
ข้อมูลนำเข้า	<<เอกสาร>> สินทรัพย์ที่สามารถนำมาใช้ซ้ำ
กระบวนการ	ผู้จัดการโครงการทำการวิเคราะห์ผลกระทบจากสินทรัพย์ที่สามารถนำมาใช้ซ้ำที่มีต่อการแก้ไขปัญหาและดัดแปลงซอฟต์แวร์ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลลัพธ์การวิเคราะห์ผลกระทบ
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.41 กระบวนการกำหนดวิธีการบำรุงรักษา (Define maintenance resolution)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	กำหนดวิธีการบำรุงรักษา
วัตถุประสงค์	เพื่อให้ได้มาซึ่งวิธีในการบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการวิเคราะห์ผลกระทบที่สามารถนำมาใช้ซ้ำ หรือ 2. เมื่อต้องการวัดผลกระทบจากการกำหนดวิธีการบำรุงรักษา
ข้อมูลนำเข้า	<<เอกสาร>> ผลลัพธ์การวิเคราะห์ผลกระทบ
กระบวนการ	ผู้จัดการโครงการทำการพิจารณาผลลัพธ์การวิเคราะห์ผลกระทบของสินทรัพย์เพื่อวิเคราะห์หาวิธีการแก้ไขปัญหา เพื่อใช้ในการแก้ไขปัญหาลงและดัดแปลงซอฟต์แวร์ พร้อมจัดทำเอกสารบันทึกผล

ตารางที่ 4.41 กระบวนการกำหนดวิธีการบำรุงรักษา (Define maintenance resolution) (ต่อ)

หัวข้อ	คำอธิบาย
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> วิธีการบำรุงรักษา
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

4.3.3.3 ขั้นตอนดัดแปลง ย้าย และยกเลิกข้อมูลซอฟต์แวร์สำเร็จรูป

ขั้นตอนนี้ประกอบด้วย 3 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.42-4.44

ตารางที่ 4.42 กระบวนการวางแผนการดัดแปลง ย้ายหรือยกเลิกซอฟต์แวร์ (Modify / Migrate / Retire software plan)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	วางแผนการดัดแปลง ย้ายหรือยกเลิกซอฟต์แวร์
วัตถุประสงค์	เพื่อให้ได้มาซึ่งแผนสำหรับการดำเนินการดัดแปลง ย้ายหรือยกเลิกข้อมูลของผลิตภัณฑ์ซอฟต์แวร์ในสายผลิตภัณฑ์
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการกำหนดวิธีการบำรุงรักษา หรือ 2. เมื่อต้องการวัดผลกระบวนการวางแผนการดัดแปลง ย้ายหรือยกเลิกข้อมูล
ข้อมูลนำเข้า	1. <<เอกสาร>> รายงานปัญหา/คำร้องขอการดัดแปลง 2. <<เอกสาร>> สินทรัพย์ที่สามารถนำมาใช้ซ้ำ 3. <<เอกสาร>> วิธีการบำรุงรักษา
กระบวนการ	ทีมผู้พัฒนาทำการวางแผนปฏิบัติการ เพื่อดัดแปลง ย้าย หรือยกเลิกข้อมูล ตามผลการวิเคราะห์การแก้ไขปัญหาที่ได้ เพื่อตอบสนองต่อรายงานปัญหาหรือคำร้องขอการดัดแปลง กรณีที่ต้องทำการดัดแปลงผลิตภัณฑ์ซอฟต์แวร์จะมีข้อมูลนำเข้า คือสินทรัพย์ที่สามารถนำมาใช้ซ้ำ พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> แผนปฏิบัติการบำรุงรักษา

ตารางที่ 4.42 กระบวนการวางแผนการดัดแปลง ย้ายหรือยกเลิกซอฟต์แวร์ (Modify / Migrate / Retire software plan) (ต่อ)

หัวข้อ	คำอธิบาย
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.43 กระบวนการดัดแปลง ย้ายหรือยกเลิกซอฟต์แวร์ (Modify / Migrate / Retire software)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	ดัดแปลง ย้ายหรือยกเลิกซอฟต์แวร์
วัตถุประสงค์	เพื่อดำเนินการดัดแปลง ย้ายหรือยกเลิกข้อมูลของผลิตภัณฑ์ซอฟต์แวร์ในสายผลิตภัณฑ์ตามแผนปฏิบัติการบำรุงรักษา
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการวางแผนการดัดแปลง ย้ายหรือยกเลิกข้อมูล หรือ 2. เมื่อต้องการวัดผลกระบวนการดัดแปลง ย้ายหรือยกเลิกข้อมูล
ข้อมูลนำเข้า	<<เอกสาร>> แผนปฏิบัติการบำรุงรักษา
กระบวนการ	ทีมผู้พัฒนาทำการดัดแปลง ย้าย หรือยกเลิกข้อมูลตามแผนที่ได้พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการบำรุงรักษา 3. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

ตารางที่ 4.44 กระบวนการแจ้งผลการบำรุงรักษา (Notify maintenance result)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	แจ้งผลการบำรุงรักษา
วัตถุประสงค์	เพื่อแจ้งผลการบำรุงรักษาต่อผู้ที่เกี่ยวข้อง
เกณฑ์การเข้า	1. เมื่อสิ้นสุดกระบวนการดัดแปลง ย้ายหรือยกเลิกข้อมูล หรือ 2. เมื่อต้องการวัดผลกระบวนการแจ้งผลการบำรุงรักษา

ตารางที่ 4.44 กระบวนการแจ้งผลการบำรุงรักษา (Notify maintenance result) (ต่อ)

หัวข้อ	คำอธิบาย
ข้อมูลนำเข้า	1. <<เอกสาร>> ผลการบำรุงรักษา 2. ผลิตภัณฑ์ซอฟต์แวร์
กระบวนการ	ทีมผู้พัฒนาทำการแจ้งผลการดัดแปลง ย้ายหรือยกเลิกข้อมูลนั้นๆ ให้กับผู้มีส่วนได้เสีย พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> แจ้งผลการบำรุงรักษา 2. ผลิตภัณฑ์ซอฟต์แวร์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ทีมผู้พัฒนา

4.3.4 กระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป

กระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูปประกอบด้วย 1 ขั้นตอน ซึ่งสามารถอธิบายในรายละเอียดได้ดังต่อไปนี้

4.3.4.1 ขั้นตอนวัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป

ขั้นตอนนี้ประกอบด้วย 5 กระบวนการ ซึ่งสามารถอธิบายในรายละเอียดดังแสดงในตารางที่ 4.45-4.49

ตารางที่ 4.45 กระบวนการสร้างจุดมุ่งหมายของการวัดผล (Establish measurement objectives)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	สร้างจุดมุ่งหมายของการวัดผล
วัตถุประสงค์	เพื่อได้มาซึ่งจุดมุ่งหมายสำหรับการวัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	เมื่อต้องการวัดผลกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป การบูรณาการซอฟต์แวร์สำเร็จรูป หรือการบำรุงรักษาซอฟต์แวร์สำเร็จรูป
ข้อมูลนำเข้า	ผลิตภัณฑ์ซอฟต์แวร์

ตารางที่ 4.45 กระบวนการสร้างจุดมุ่งหมายของการวัดผล (Establish measurement objectives) (ต่อ)

หัวข้อ	คำอธิบาย
กระบวนการงาน	ผู้จัดการโครงการทำการสร้างจุดมุ่งหมายของการวัดผล เช่น เพื่อลดเวลาในการส่งมอบซอฟต์แวร์ เพื่อลดต้นทุน หรือเพื่อเพิ่มคุณภาพซอฟต์แวร์ เป็นต้น รวมถึงทรัพยากรที่ต้องใช้ในการวัดผล เช่น กระบวนการ การจัดการ หรือ เทคนิคต่างๆ เป็นต้น พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> จุดมุ่งหมายการวัดผล
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.46 กระบวนการวางแผนกระบวนการวัดผล (Plan procedure measurement)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	วางแผนกระบวนการวัดผล
วัตถุประสงค์	เพื่อให้ได้มาซึ่งกระบวนการที่ใช้ในการวัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	เมื่อสิ้นสุดกระบวนการสร้างจุดมุ่งหมายของการวัดผล
ข้อมูลนำเข้า	<<เอกสาร>> จุดมุ่งหมายการวัดผล
กระบวนการงาน	ผู้จัดการโครงการทำการวางแผนกระบวนการที่ใช้ในการวัดผล โดยทำการระบุการวัดผลที่ตอบสนองต่อจุดมุ่งหมาย ระบุการจัดเก็บข้อมูล ระบุกระบวนการวิเคราะห์การวัดผล พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> แผนการวัดผล
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.47 กระบวนการวัดผลการพัฒนาซอฟต์แวร์ (Measure software development)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	วัดผลการพัฒนาซอฟต์แวร์
วัตถุประสงค์	เพื่อทำการวัดผลผลิตภัณฑ์ซอฟต์แวร์หรือกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	เมื่อสิ้นสุดกระบวนการวางแผนกระบวนการงานวัดผล
ข้อมูลนำเข้า	<<เอกสาร>> แผนการวัดผล
กระบวนการ	ผู้จัดการโครงการทำการวัดผลการพัฒนาซอฟต์แวร์ตามแผนที่ได้พร้อมจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลลัพธ์การวัดผล
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.48 กระบวนการวิเคราะห์และจัดเก็บผลลัพธ์ (Analyze and store results)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	วิเคราะห์และจัดเก็บผลลัพธ์
วัตถุประสงค์	เพื่อทำการวิเคราะห์ผลลัพธ์การวัดผลและจัดเก็บผลลัพธ์ที่เกิดจากกระบวนการการวัดผลผลิตภัณฑ์ซอฟต์แวร์หรือกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป
เกณฑ์การเข้า	เมื่อสิ้นสุดกระบวนการวัดผลการพัฒนาซอฟต์แวร์
ข้อมูลนำเข้า	<<เอกสาร>> ผลลัพธ์การวัดผล
กระบวนการ	ผู้จัดการโครงการทำการรวบรวมผลการวัดผล วิเคราะห์ผลการวัดผลและจัดเก็บผลการวัดผล พร้อมทั้งจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> ผลการวิเคราะห์และจัดเก็บผลลัพธ์
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

ตารางที่ 4.49 กระบวนการรายงานการวัดผลและแนวทางสำหรับการปรับปรุงกระบวนการ (Report result and guidance for process improvement)

หัวข้อ	คำอธิบาย
ชื่อกระบวนการ	รายงานการวัดผลและแนวทางสำหรับการปรับปรุงกระบวนการ
วัตถุประสงค์	เพื่อรายงานผลการวัดผลพร้อมทั้งแนวทางในการปรับปรุงกระบวนการให้มีความสมบูรณ์มากขึ้นในการพัฒนาซอฟต์แวร์ในครั้งถัดไป
เกณฑ์การเข้า	เมื่อสิ้นสุดกระบวนการวิเคราะห์และจัดเก็บผลลัพธ์
ข้อมูลนำเข้า	<<เอกสาร>> ผลการวิเคราะห์และจัดเก็บผลลัพธ์
กระบวนการ	ผู้จัดการโครงการทำการรายงานผลการวัดผลและแนวทางสำหรับการปรับปรุงกระบวนการต่อผู้มีส่วนได้เสียในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ เพื่อใช้พัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปให้มีประสิทธิภาพยิ่งขึ้นในการพัฒนาครั้งต่อไป พร้อมทั้งจัดทำเอกสารบันทึกผล
ข้อมูลนำออก	1. <<เอกสาร>> เอกสารบันทึกผลของกระบวนการ 2. <<เอกสาร>> รายงานผลการวัดผลและแนวทางสำหรับการปรับปรุงกระบวนการ
เกณฑ์การออก	เมื่อได้ข้อมูลนำออกครบถ้วน
ผู้รับผิดชอบ	ผู้จัดการโครงการ

4.4 การประเมินกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป (COTS Development Process Model Evaluation)

หลังจากที่ได้กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปแล้วนั้น ผู้วิจัยจึงประเมินแบบจำลองที่นำเสนอ เพื่อทวนสอบเปรียบเทียบกับกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีเพิลอี 1517 กลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ซึ่งการประเมินนี้ทางผู้วิจัยใช้วิธีการทวนสอบแบบวิธีการตรวจสอบ หรือ Walkthrough ซึ่งวิธีการนี้ทางผู้วิจัยได้ใช้รายการตรวจสอบที่กำหนดตามกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีเพิลอี 1517 เพื่อตรวจสอบว่ากิจกรรมและอาร์ทิแฟกของกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่นำเสนอนั้นสอดคล้องกับกลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ของมาตรฐานหรือไม่โดยพิจารณาประเมินเฉพาะกิจกรรมที่เกิดจากการออกแบบและพัฒนากระบวนการที่ได้จากมาตรฐาน

ไอทริฟเฟิลอี 1517 ซึ่งกิจกรรมอื่นนอกจากนั้นเป็นส่วนที่ผู้วิจัยได้ทำการเพิ่มเติมขึ้นจากการศึกษาพื้นฐานวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ บทเรียนรู้ วิธีปฏิบัติที่ดีที่สุด มาตรฐานอื่นๆ และงานวิจัยที่เกี่ยวข้องเพื่อให้ได้มาซึ่งกระบวนการที่มีความสมบูรณ์มากยิ่งขึ้น โดยประกอบด้วยรายละเอียดของกิจกรรม ดังแสดงในตารางที่ 4.50 ต่อไปนี้

ตารางที่ 4.50 ตารางรายละเอียดของกิจกรรมที่ผู้วิจัยได้เพิ่มเติมจากการศึกษานอกเหนือจากมาตรฐานไอทริฟเฟิลอี 1517

กิจกรรมในกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป	กลุ่มกระบวนการที่ใช้
จัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป	COTS tailoring, COTS integration
ปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์	COTS tailoring, COTS integration
ส่งมอบผลิตภัณฑ์ซอฟต์แวร์	COTS tailoring, COTS integration
สนับสนุนการยอมรับซอฟต์แวร์	COTS tailoring, COTS integration
สร้างจุดมุ่งหมายของการวัดผล	COTS support
วางแผนกระบวนการวัดผล	COTS support
วัดผลการพัฒนาซอฟต์แวร์	COTS support
วิเคราะห์และจัดเก็บผลลัพธ์	COTS support
รายงานการวัดผลและแนวทางสำหรับการปรับปรุงกระบวนการ	COTS support

ผลของการประเมินสามารถแสดงในรายละเอียดได้ดังตารางที่ 4.51-4.52 โดยแสดงการเปรียบเทียบระหว่างกิจกรรมที่ผู้วิจัยนำเสนอกับกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทริฟเฟิลอี 1517 กลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ ดังต่อไปนี้

ตารางที่ 4.51 ตารางเปรียบเทียบระหว่างกิจกรรมที่ผู้วิจัยนำเสนอกับกลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ (กระบวนการการพัฒนา)

กิจกรรมในกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป	กระบวนการพัฒนาที่ใช้เป็นฐานในการออกแบบและพัฒนากระบวนการ (Development Process)
กำหนดแบบจำลองและกลยุทธ์ในการพัฒนา	Process implementation, System requirements analysis, System architectural design

ตารางที่ 4.51 ตารางเปรียบเทียบระหว่างกิจกรรมที่ผู้วิจัยนำเสนอกับกลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ (กระบวนการการพัฒนา) (ต่อ)

กิจกรรมในกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป	กระบวนการพัฒนาที่ใช้เป็นฐานในการออกแบบและพัฒนากระบวนการ (Development Process)
กำหนดแผนการปรับแต่ง	Process implementation
ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์	Process implementation, System requirements analysis, System architectural design, Software requirements analysis
ประเมินข้อกำหนดความต้องการ	System requirements analysis, Software requirements analysis, System architectural design
กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	Software architectural design
กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	Software detailed design
ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	Software architectural design, Software detailed design
พัฒนาส่วนประกอบ	Software coding and testing
ประเมินการพัฒนาส่วนประกอบ	Software coding and testing
การบูรณาการกับระบบที่มีอยู่	Software integration, System integration
ประเมินการบูรณาการกับระบบที่มีอยู่	Software integration, System integration
ทดสอบสายผลิตภัณฑ์	Software qualification testing, System qualification testing
วางแผนการติดตั้ง	Software installation
ประเมินแผนการติดตั้ง	Software installation
กำหนดแผนการบูรณาการ	Process implementation

ตารางที่ 4.51 ตารางเปรียบเทียบระหว่างกิจกรรมที่ผู้วิจัยนำเสนอกับกลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ (กระบวนการการพัฒนา) (ต่อ)

กิจกรรมในกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป	กระบวนการพัฒนาที่ใช้เป็นฐานในการออกแบบและพัฒนากระบวนการ (Development Process)
บูรณาการระบบและซอฟต์แวร์	Software integration, System integration
ประเมินการบูรณาการระบบและซอฟต์แวร์	Software integration, System integration

ตารางที่ 4.52 ตารางเปรียบเทียบระหว่างกิจกรรมที่ผู้วิจัยนำเสนอกับกลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ (กระบวนการการบำรุงรักษา)

กิจกรรมในกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป	กระบวนการบำรุงรักษาที่ใช้เป็นฐานในการออกแบบและพัฒนากระบวนการ (Maintenance Process)
กำหนดแผนการบำรุงรักษา	Process implementation
สร้างกระบวนการบำรุงรักษา	Process implementation
ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์	Process implementation, Problem and modification analysis
วิเคราะห์ผลกระทบสินทรัพย์ที่สามารถนำมาใช้ซ้ำ	Problem and modification analysis
กำหนดวิธีการบำรุงรักษา	Problem and modification analysis
วางแผนการดัดแปลง ย้ายหรือยกเลิกข้อมูล	Modification implementation, Migration, Software retirement
ดัดแปลง ย้ายหรือยกเลิกข้อมูล	Modification implementation, Migration, Software retirement
แจ้งผลการบำรุงรักษา	Modification implementation, Migration, Software retirement

บทที่ 5

การวิเคราะห์และออกแบบเครื่องมือนับสนุนกระบวนการ การพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

เมื่อผู้วิจัยได้วิเคราะห์และออกแบบกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปเป็นที่เรียบร้อยแล้วนั้น ในบทนี้ผู้วิจัยจะวิเคราะห์ความต้องการของเครื่องมือนับสนุน ออกแบบหน้าที่การทำงานของเครื่องมือ ออกแบบสถาปัตยกรรมของเครื่องมือ ออกแบบฐานข้อมูล เชิงสัมพันธ์และออกแบบส่วนต่อประสานผู้ใช้เพื่อนำไปใช้ในการพัฒนาเครื่องมือนับสนุนต่อไป

5.1 การวิเคราะห์ความต้องการของเครื่องมือนับสนุน

จากการวิเคราะห์และออกแบบกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ในบทที่ 4 แล้วนั้น ผู้วิจัยจึงได้วิเคราะห์และสรุปถึงความต้องการของระบบ โดยแบ่งออกเป็น 2 ส่วนหลัก คือ ความต้องการด้านหน้าที่ และความต้องการที่ไม่ใช่หน้าที่ โดยมีรายละเอียดดังต่อไปนี้

5.1.1 ความต้องการด้านหน้าที่

เครื่องมือนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์สามารถแบ่งออกเป็นระบบงานหลัก 2 ระบบใหญ่ตามจุดประสงค์การใช้งานของระบบ ซึ่งแต่ละระบบงานมีความต้องการด้านหน้าที่ดังต่อไปนี้

1) ระบบงานผู้ดูแลระบบ

ระบบงานผู้ดูแลระบบเป็นระบบจัดการข้อมูลที่จะนำไปใช้ต่อไปในระบบจัดการการดำเนินงาน สามารถแบ่งหน้าที่ออกเป็น 3 กลุ่มหลัก คือ

- (1) กลุ่มการจัดการสารสนเทศอาร์ทิแฟก โดยมีรายละเอียดดังต่อไปนี้
 - (1.1) บันทึกรหัสข้อมูลของเอกสารนับสนุนกระบวนการและกระบวนการที่ต้องใช้
 - (1.2) แก้ไขและเรียกดูข้อมูลของเอกสารนับสนุน
 - (1.3) เรียกดูข้อมูลเอกสารผลลัพธ์ที่เกิดขึ้นจากการดำเนินการของกระบวนการ

- (2) กลุ่มการจัดการสารสนเทศผู้ใช้งานระบบ
 - (2.1) บันทึกข้อมูลของผู้ใช้งานระบบ
 - (2.2) แก้ไขและเรียกดูข้อมูลของผู้ใช้งานระบบ
- (3) กลุ่มการจัดการเกณฑ์รายการตรวจสอบ
 - (2.1) บันทึกข้อมูลรายการตรวจสอบที่ต้องใช้สำหรับกระบวนการประเมิน
 - (2.2) แก้ไขและเรียกดูข้อมูลของรายการตรวจสอบที่ต้องใช้สำหรับกระบวนการประเมิน

2) ระบบงานจัดการการดำเนินงาน

ระบบงานจัดการการดำเนินงานเป็นระบบจัดการเกี่ยวกับข้อมูลสามารถแบ่งหน้าที่ออกเป็น 22 กลุ่มหลัก คือ

- (1) บันทึกและเรียกดูข้อมูลการวางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป
- (2) บันทึกและเรียกดูข้อมูลการออกแบบสถาปัตยกรรมการปรับแต่งซอฟต์แวร์สำเร็จรูป
- (3) บันทึกและเรียกดูข้อมูลการปรับแต่งซอฟต์แวร์สำเร็จรูป
- (4) บันทึกและเรียกดูข้อมูลการติดตั้งซอฟต์แวร์สำหรับกระบวนการปรับแต่งซอฟต์แวร์สำเร็จรูป
- (5) บันทึกและเรียกดูข้อมูลการสนับสนุนการยอมรับซอฟต์แวร์สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป
- (6) บันทึกและเรียกดูข้อมูลการวางแผนโครงการการบูรณาการซอฟต์แวร์สำเร็จรูป
- (7) บันทึกและเรียกดูข้อมูลการออกแบบสถาปัตยกรรมการบูรณาการซอฟต์แวร์สำเร็จรูป
- (8) บันทึกและเรียกดูข้อมูลการบูรณาการซอฟต์แวร์สำเร็จรูป
- (9) บันทึกและเรียกดูข้อมูลการติดตั้งซอฟต์แวร์สำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป

- (10) บันทึกและเรียกดูข้อมูลการสนับสนุนการยอมรับซอฟต์แวร์สำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป
- (11) บันทึกและเรียกดูข้อมูลการวางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป
- (12) บันทึกและเรียกดูข้อมูลการวิเคราะห์ปัญหาและการตัดแปลง
- (13) บันทึกและเรียกดูข้อมูลการตัดแปลง ย้ายหรือยกเลิกซอฟต์แวร์
- (14) บันทึกและเรียกดูข้อมูลการวัดผลกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป
- (15) บันทึกและเรียกดูข้อมูลการประเมินการวางแผนโครงการ การปรับแต่งซอฟต์แวร์สำเร็จรูป
- (16) บันทึกและเรียกดูข้อมูลการประเมินการออกแบบสถาปัตยกรรม การปรับแต่งซอฟต์แวร์สำเร็จรูป
- (17) บันทึกและเรียกดูข้อมูลการประเมินการปรับแต่งซอฟต์แวร์สำเร็จรูป
- (18) บันทึกและเรียกดูข้อมูลการประเมินการติดตั้งซอฟต์แวร์สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป
- (19) บันทึกและเรียกดูข้อมูลการประเมินการวางแผนโครงการการบูรณาการซอฟต์แวร์สำเร็จรูป
- (20) บันทึกและเรียกดูข้อมูลการประเมินการออกแบบสถาปัตยกรรมการบูรณาการซอฟต์แวร์สำเร็จรูป
- (21) บันทึกและเรียกดูข้อมูลการประเมินการบูรณาการซอฟต์แวร์สำเร็จรูป
- (22) บันทึกและเรียกดูข้อมูลการประเมินการติดตั้งซอฟต์แวร์สำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป

ความต้องการด้านหน้าที่นี้ ผู้วิจัยได้สรุปความต้องการให้อยู่ในรูปแบบของตาราง แสดงความต้องการหน้าที่ ซึ่งประกอบด้วยรายละเอียดดังต่อไปนี้ รหัสความต้องการ ชื่อหน้าที่ คำอธิบายหน้าที่ ข้อมูลนำเข้าระบบ ข้อมูลนำออก และผู้มีสิทธิ์ใช้งาน เพื่อให้ความต้องการด้านหน้าที่เก็บอยู่ในรูปแบบที่ง่ายต่อการอ้างอิง สะดวกต่อการแก้ไขและสืบค้น รวมทั้งง่ายต่อการนำไปพัฒนาระบบในภายหลัง ซึ่งความต้องการด้านหน้าที่ของระบบแสดงได้ดังตารางที่ ง.1-ง.51 ในภาคผนวก ง.

5.1.2 ความต้องการที่ไม่ใช่หน้าที่

ความต้องการที่ไม่ใช่หน้าที่ของระบบ มีรายละเอียดดังตารางที่ 5.1

ตารางที่ 5.1 ความต้องการที่ไม่ใช่หน้าที่ระบบ

รหัส	ชื่อความต้องการที่ไม่ใช่หน้าที่	คำอธิบาย
NFR01	ด้านความสามารถการใช้งาน (Usability)	ระบบได้รับการออกแบบให้ง่ายต่อการใช้งานและสามารถเรียนรู้ได้ง่าย
NFR02	ความมั่นคงของระบบ (Security)	ระบบควรสามารถจำกัดสิทธิ์การเข้าใช้ระบบตามข้อกำหนดความต้องการระบบ
NFR03	ความละเอียดถูกต้องของข้อมูลนำเข้า (Accuracy)	ระบบมีการตรวจสอบความถูกต้องของข้อมูลก่อนการนำเข้า
NFR04	การจัดการกับความผิดพลาด (Error management)	ระบบมีความสามารถในการจัดการกับความผิดพลาดที่อาจจะเกิดขึ้นได้
NFR05	ความต้องการด้านการเคลื่อนย้ายระบบ (Portability requirements)	ระบบควรมีรูปแบบการติดตั้งที่ไม่ซับซ้อนโดยใช้สถาปัตยกรรมแบบเว็บเบส แอปพลิเคชันซึ่งผู้ใช้งานปลายทางติดตั้งแค่โปรแกรมเว็บเบราว์เซอร์เท่านั้นก็สามารถติดต่อผ่านเครือข่ายอินเทอร์เน็ตเข้ามาใช้งานระบบได้
NFR06	การบำรุงรักษาได้ของระบบ (Maintainability)	รหัสต้นฉบับ (Source code) ของระบบควรมีการเขียนหมายเหตุ (Comment) เพื่อให้ง่ายต่อผู้ดูแลและบำรุงรักษาระบบต่อไปในภายหลัง
NFR07	การออกแบบส่วนต่อประสานของระบบ (User interface design)	ส่วนต่อประสานของระบบที่ได้ เป็นไปตามการออกแบบ เช่น การกำหนดรูปแบบของหน้าจอ รูปแบบของรายงาน เอกสาร เป็นต้น

5.2 การออกแบบหน้าที่การทำงานของเครื่องมือสนับสนุน

เครื่องมือสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์มีกลุ่มผู้ใช้งานทั้งหมด 4 ระดับ ดังต่อไปนี้

1) ผู้จัดการโครงการ มีหน้าที่สร้างโครงสร้างพื้นฐานของโครงการสำหรับกระบวนการพัฒนาผลิตภัณฑ์ซอฟต์แวร์เข้าสู่ระบบ

2) ทีมผู้พัฒนา มีหน้าที่นำเข้าข้อมูลโครงการสำหรับกระบวนการพัฒนาผลิตภัณฑ์ซอฟต์แวร์เข้าสู่ระบบ

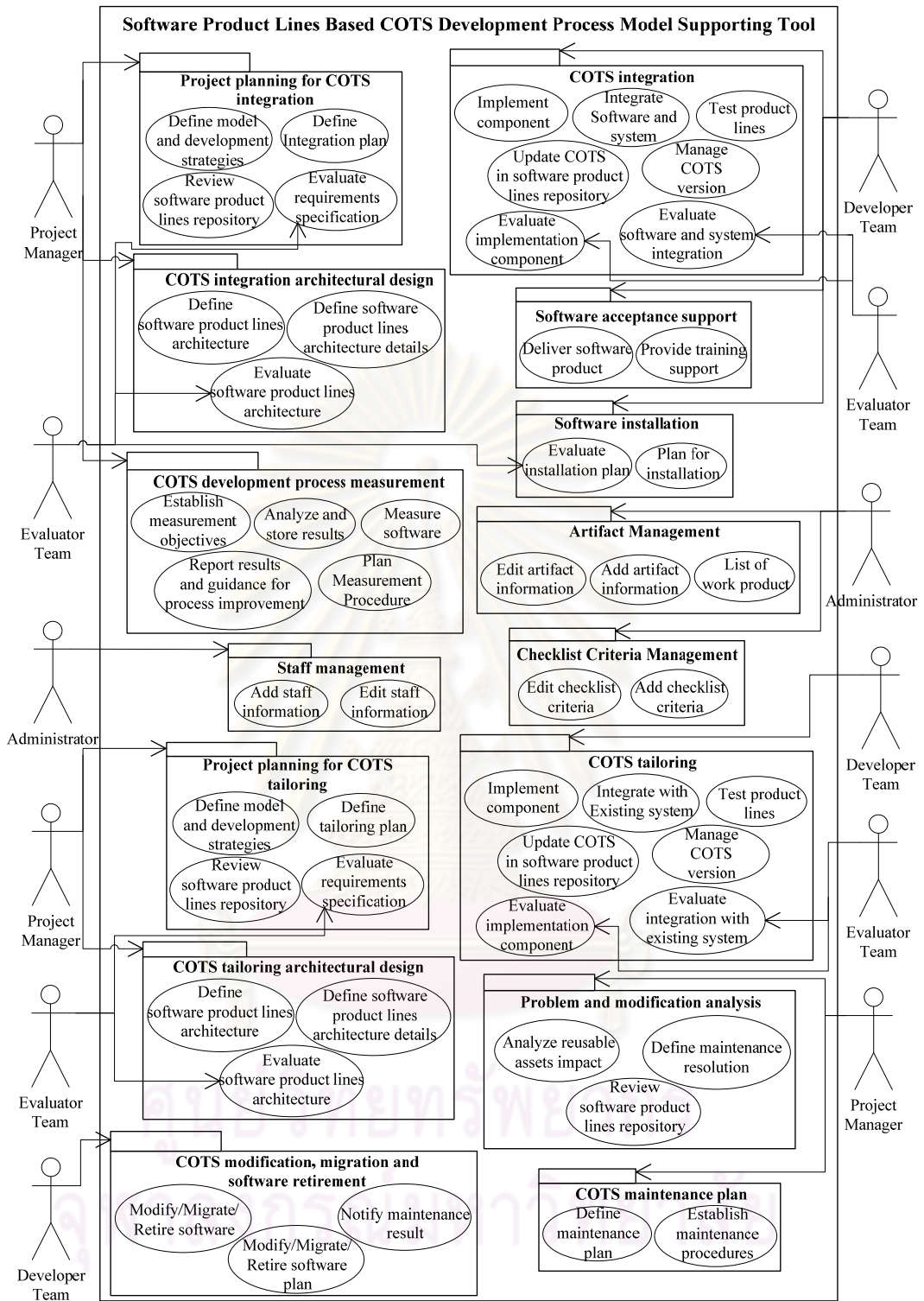
3) ทีมผู้ประเมิน มีหน้าที่นำเข้าข้อมูลโครงการสำหรับการประเมินสำหรับกระบวนการพัฒนาผลิตภัณฑ์ซอฟต์แวร์เข้าสู่ระบบ

4) ผู้จัดการระบบ มีหน้าที่นำเข้าเอกสารสนับสนุนกระบวนการ เกณฑ์การใช้อำนาจข้อมูลผู้ใช้งานระบบในแต่ละหน้าที่และจัดการกับสารสนเทศที่เกิดขึ้นกับระบบ

ทั้งนี้ผู้วิจัยได้ออกแบบหน้าที่การทำงานของเครื่องมือออกเป็น 15 ระบบย่อยตามลักษณะของการทำงานที่เกิดขึ้น คือ

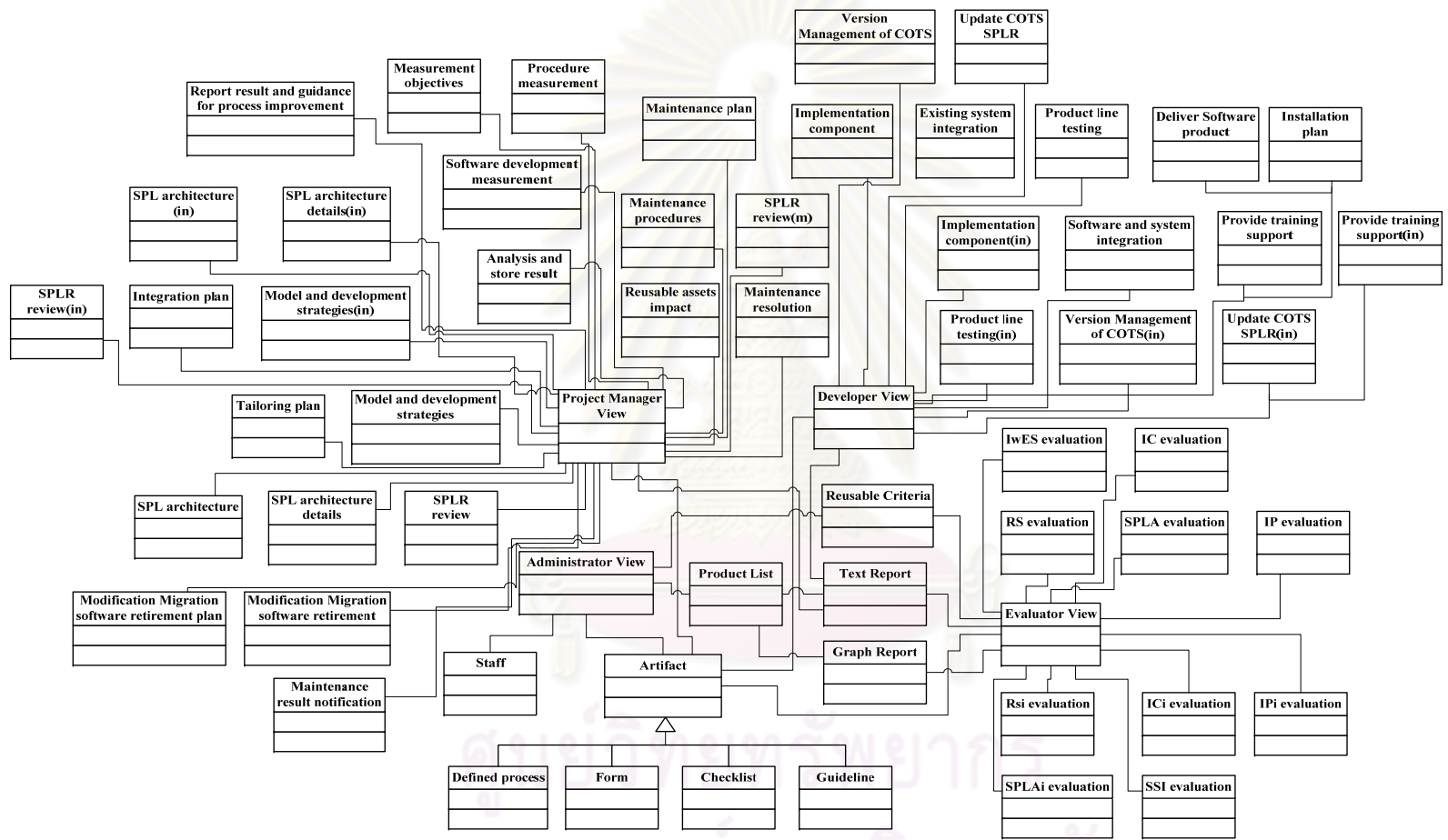
- 1) ระบบจัดการวางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป
- 2) ระบบจัดการออกแบบสถาปัตยกรรมการปรับแต่งซอฟต์แวร์สำเร็จรูป
- 3) ระบบจัดการปรับแต่งซอฟต์แวร์สำเร็จรูป
- 4) ระบบจัดการติดตั้งซอฟต์แวร์
- 5) ระบบจัดการสนับสนุนการยอมรับซอฟต์แวร์
- 6) ระบบจัดการวางแผนโครงการการบูรณาการซอฟต์แวร์สำเร็จรูป
- 7) ระบบจัดการออกแบบสถาปัตยกรรมการบูรณาการซอฟต์แวร์สำเร็จรูป
- 8) ระบบจัดการบูรณาการซอฟต์แวร์สำเร็จรูป
- 9) ระบบจัดการวางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป
- 10) ระบบจัดการวิเคราะห์ปัญหาและการดัดแปลง
- 11) ระบบจัดการดัดแปลง ย้ายและยกเลิกซอฟต์แวร์
- 12) ระบบจัดการวัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป
- 13) ระบบจัดการอาร์ทิแฟกต์และเอกสารสนับสนุน
- 14) ระบบจัดการข้อมูลผู้ใช้
- 15) ระบบจัดการเกณฑ์รายการตรวจสอบ

ผู้วิจัยนำเสนอแนวคิดการออกแบบหน้าที่การทำงานของเครื่องมือสนับสนุนด้วยแผนภาพยูสเคสที่แสดงถึงหน้าที่การทำงานของเครื่องมือสนับสนุนและผู้ใช้ที่มีส่วนเกี่ยวข้องในแต่ละระบบย่อยอย่างชัดเจนดังรูปที่ 5.1



รูปที่ 5.1 แผนภาพยูสเคสแสดงหน้าที่การทำงานของเครื่องมือสนับสนุน

นอกจากออกแบบแผนภาพยูสเคสของเครื่องมือสนับสนุน ผู้วิจัยได้ออกแบบคลาสและความสัมพันธ์ของแต่ละคลาส เพื่อแสดงความสัมพันธ์ขององค์ประกอบแต่ละวัตถุที่เกิดขึ้นแสดงดังรูปที่ 5.2 และนำไปใช้ในการออกแบบฐานข้อมูลเชิงสัมพันธ์ต่อไป สำหรับอธิบายแผนภาพคลาสต่างๆ ที่เกิดขึ้นสามารถแสดงดังตารางที่ 5.2



รูปที่ 5.2 แผนภาพคลาสของเครื่องมือสนับสนุน

ตารางที่ 5.2 คำอธิบายแผนภาพคลาสของเครื่องมือสนับสนุน

ชื่อคลาส	คำอธิบาย
ระบบจัดการวางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
Model and development strategies	ทำหน้าที่จัดเก็บข้อมูลแบบจำลองและกลยุทธ์ในการพัฒนา
Tailoring plan	ทำหน้าที่จัดเก็บข้อมูลแผนการปรับแต่งซอฟต์แวร์สำเร็จรูป
SPLR review	ทำหน้าที่จัดเก็บข้อมูลสินทรัพย์ที่สามารถนำมาใช้ซ้ำ
RS evaluation	ทำหน้าที่จัดเก็บข้อมูลและสนับสนุนการประเมินข้อกำหนดความต้องการซอฟต์แวร์
ระบบจัดการออกแบบสถาปัตยกรรมการปรับแต่งซอฟต์แวร์สำเร็จรูป	
SPL architecture	ทำหน้าที่จัดเก็บข้อมูลการออกแบบสถาปัตยกรรมซอฟต์แวร์
SPL architecture details	ทำหน้าที่จัดเก็บข้อมูลการออกแบบรายละเอียดสถาปัตยกรรมซอฟต์แวร์
SPLA evaluation	ทำหน้าที่จัดเก็บข้อมูลและสนับสนุนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
ระบบจัดการปรับแต่งซอฟต์แวร์สำเร็จรูป	
Implementation component	ทำหน้าที่จัดเก็บข้อมูลการพัฒนาส่วนประกอบ
IC evaluation	ทำหน้าที่จัดเก็บข้อมูลและสนับสนุนการประเมินการพัฒนาส่วนประกอบ
Existing system integration	ทำหน้าที่จัดเก็บข้อมูลการบูรณาการกับระบบที่มีอยู่
IwES evaluation	ทำหน้าที่จัดเก็บข้อมูลและสนับสนุนการประเมินการบูรณาการกับระบบที่มีอยู่
Product line testing	ทำหน้าที่จัดเก็บข้อมูลการทดสอบผลิตภัณฑ์ซอฟต์แวร์
Version management of COTS	ทำหน้าที่จัดเก็บข้อมูลการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป
Update COTS SPLR	ทำหน้าที่จัดเก็บข้อมูลการปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บสายผลิตภัณฑ์
ระบบจัดการติดตั้งซอฟต์แวร์	
Installation plan	ทำหน้าที่จัดเก็บข้อมูลการวางแผนการติดตั้งซอฟต์แวร์

ตารางที่ 5.2 คำอธิบายแผนภาพคลาสของเครื่องมือสนับสนุน (ต่อ)

ชื่อคลาส	คำอธิบาย
IP evaluation	ทำหน้าที่จัดเก็บข้อมูลและสนับสนุนการประเมินแผนการติดตั้งซอฟต์แวร์
ระบบจัดการสนับสนุนการยอมรับซอฟต์แวร์	
Deliver software product	ทำหน้าที่จัดเก็บข้อมูลการสนับสนุนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์
Provide training support	ทำหน้าที่จัดเก็บข้อมูลการเตรียมการสนับสนุนการฝึกอบรม
ระบบจัดการวางแผนโครงการบูรณาการซอฟต์แวร์สำเร็จรูป	
Model and development strategies(in)	ทำหน้าที่จัดเก็บข้อมูลแบบจำลองและกลยุทธ์ในการพัฒนา
Integration plan	ทำหน้าที่จัดเก็บข้อมูลแผนการบูรณาการซอฟต์แวร์สำเร็จรูป
SPLR review(in)	ทำหน้าที่จัดเก็บข้อมูลสินทรัพย์ที่สามารถนำมาใช้ซ้ำ
RSi evaluation	ทำหน้าที่จัดเก็บข้อมูลและสนับสนุนการประเมินข้อกำหนดความต้องการซอฟต์แวร์
ระบบจัดการออกแบบสถาปัตยกรรมการบูรณาการซอฟต์แวร์สำเร็จรูป	
SPL architecture(in)	ทำหน้าที่จัดเก็บข้อมูลการออกแบบสถาปัตยกรรมซอฟต์แวร์
SPL architecture details(in)	ทำหน้าที่จัดเก็บข้อมูลการออกแบบรายละเอียดสถาปัตยกรรมซอฟต์แวร์
SPLAi evaluation	ทำหน้าที่จัดเก็บข้อมูลและสนับสนุนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
ระบบจัดการบูรณาการซอฟต์แวร์สำเร็จรูป	
Implementation component(in)	ทำหน้าที่จัดเก็บข้อมูลการพัฒนาส่วนประกอบ
ICi evaluation	ทำหน้าที่จัดเก็บข้อมูลและสนับสนุนการประเมินการพัฒนาส่วนประกอบ
Software and system integration	ทำหน้าที่จัดเก็บข้อมูลการบูรณาการระบบและซอฟต์แวร์

ตารางที่ 5.2 คำอธิบายแผนภาพคลาสของเครื่องมือสนับสนุน (ต่อ)

ชื่อคลาส	คำอธิบาย
SSI evaluation	ทำหน้าที่จัดเก็บข้อมูลและสนับสนุนการประเมินการบูรณาการระบบและซอฟต์แวร์
Product line testing	ทำหน้าที่จัดเก็บข้อมูลการทดสอบผลิตภัณฑ์ซอฟต์แวร์
Version management of COTS(in)	ทำหน้าที่จัดเก็บข้อมูลการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป
Update COTS SPLR(in)	ทำหน้าที่จัดเก็บข้อมูลการปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บสายผลิตภัณฑ์
ระบบจัดการวางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป	
Maintenance plan	ทำหน้าที่จัดเก็บข้อมูลการวางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป
Maintenance procedures	ทำหน้าที่จัดเก็บข้อมูลการดำเนินงานการบำรุงรักษาซอฟต์แวร์สำเร็จรูป
ระบบจัดการวิเคราะห์ปัญหาและการดัดแปลง	
SPLR review(m)	ทำหน้าที่จัดเก็บข้อมูลสินทรัพย์ที่สามารถนำมาใช้ซ้ำ
Reusable assets impact	ทำหน้าที่จัดเก็บข้อมูลผลกระทบสินทรัพย์ที่สามารถนำมาใช้ซ้ำ
Maintenance resolution	ทำหน้าที่จัดเก็บข้อมูลกำหนดวิธีการบำรุงรักษา
ระบบจัดการดัดแปลง ย้ายและยกเลิกซอฟต์แวร์	
Modification Migration Software retirement plan	ทำหน้าที่จัดเก็บข้อมูลการวางแผนการดัดแปลง ย้ายและยกเลิกซอฟต์แวร์
Modification Migration Software retirement	ทำหน้าที่จัดเก็บข้อมูลการดัดแปลง ย้ายและยกเลิกซอฟต์แวร์
Maintenance result notification	ทำหน้าที่จัดเก็บข้อมูลการแจ้งผลการบำรุงรักษา
ระบบจัดการวัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป	
Measurement objectives	ทำหน้าที่จัดเก็บข้อมูลการสร้างจุดมุ่งหมายของการวัดผล
Procedure measurement	ทำหน้าที่จัดเก็บข้อมูลการวางแผนกระบวนการวัดผล
Software development measurement	ทำหน้าที่จัดเก็บข้อมูลการวัดผลการพัฒนาซอฟต์แวร์

ตารางที่ 5.2 คำอธิบายแผนภาพคลาสของเครื่องมือสนับสนุน (ต่อ)

ชื่อคลาส	คำอธิบาย
Analysis and store result	ทำหน้าที่จัดเก็บข้อมูลการวิเคราะห์และจัดเก็บผลลัพธ์
Report result and guidance for process improvement	ทำหน้าที่จัดเก็บข้อมูลการรายงานการวัดผลและแนวทางสำหรับการปรับปรุงกระบวนการ
ระบบจัดการอาร์ทิแฟกต์และเอกสารสนับสนุน	
Artifact	ทำหน้าที่จัดเก็บและแก้ไขข้อมูลเอกสารสนับสนุนกระบวนการ
Defined Process	ทำหน้าที่จัดเก็บข้อมูลเอกสารนิยามกระบวนการ
Form	ทำหน้าที่จัดเก็บข้อมูลฟอร์ม
Checklist	ทำหน้าที่จัดเก็บข้อมูลรายการตรวจสอบ
Guideline	ทำหน้าที่จัดเก็บข้อมูลเอกสารแนวทางในการดำเนินการ
Product List	ทำหน้าที่จัดเก็บและเรียกดูข้อมูลเอกสารที่เกิดขึ้นในการดำเนินงานกระบวนการ
Text report	ทำหน้าที่จัดเก็บและแสดงผลบันทึกข้อมูลการดำเนินงานกระบวนการในรูปแบบของข้อความ
Graph report	ทำหน้าที่จัดเก็บและแสดงผลบันทึกข้อมูลการดำเนินงานกระบวนการในรูปแบบของกราฟ
ระบบจัดการข้อมูลผู้ใช้	
Staff	ทำหน้าที่จัดการกับข้อมูลผู้ใช้งานระบบ
ระบบจัดการเกณฑ์รายการตรวจสอบ	
Reusable Criteria	ทำหน้าที่จัดการกับข้อมูลเกณฑ์การใช้ซ้ำที่ใช้สำหรับกระบวนการประเมิน

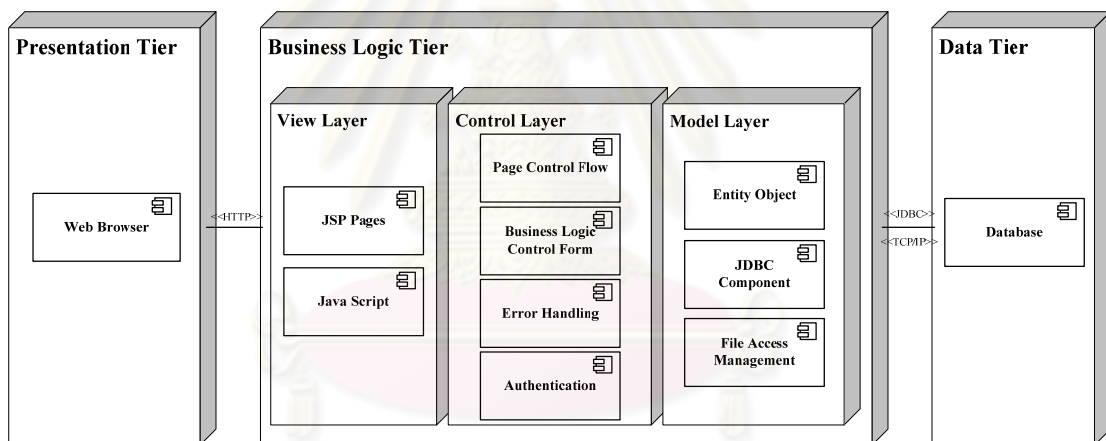
5.3 การออกแบบสถาปัตยกรรมของเครื่องมือสนับสนุน

ในการออกแบบสถาปัตยกรรมของเครื่องมือ ผู้วิจัยได้ทำการแบ่งการออกแบบในส่วนนี้ออกเป็น 2 ส่วนคือ

- 1) การออกแบบสถาปัตยกรรมระบบ

เมื่อได้กำหนดหน้าที่การทำงานของเครื่องมือเรียบร้อยแล้ว ผู้วิจัยจึงออกแบบสถาปัตยกรรมระบบ ที่จะแสดงโครงสร้างเทคโนโลยีของเครื่องมือที่สร้างขึ้น โดยสถาปัตยกรรมที่ใช้เป็นการเชื่อมต่อผ่านโพรโทคอลที่ซีพีไอพี (TCP/IP) และระบบที่พัฒนาขึ้นมีฐานข้อมูลแบบรวมศูนย์ (Centralized Database) และมีส่วนหน้าทำงานสนับสนุนที่มีลักษณะโครงสร้างสถาปัตยกรรมแบบเว็บเบสแอปพลิเคชัน (Web-Based Application) โดยที่สถาปัตยกรรมเทคโนโลยีนี้จะมีลักษณะของโครงสร้างเป็นแบบหลายชั้น (Multi-Tier) ดังแสดงในรูปที่ 5.3 ซึ่งมีรายละเอียดดังต่อไปนี้

- (1) ชั้นส่วนการนำเสนอ (Presentation Tier) ทำหน้าที่เป็นส่วนต่อประสานกับผู้ใช้ระบบ
- (2) ชั้นส่วนตรรกะทางธุรกิจ (Business Logic Tier) ทำหน้าที่ให้บริการข้อมูลและประมวลผลการดำเนินงานให้แก่เครื่องลูกข่าย
- (3) ชั้นส่วนหน่วยข้อมูล (Data Tier) ทำหน้าที่เก็บข้อมูลของระบบทั้งหมด



รูปที่ 5.3 แผนภาพสถาปัตยกรรมระบบ

ประโยชน์ของสถาปัตยกรรมแบบเว็บเบสแอปพลิเคชันนั้น เพื่อให้ผู้ใช้งานระบบสามารถเข้าใช้งานจากเครื่องลูกข่ายในสถานที่ใดๆ ผ่านทางเว็บเบราว์เซอร์โดยไม่จำกัดการทำงานอยู่ที่บริษัทของผู้ใช้เพียงแห่งเดียว และเกิดความคล่องตัวในการทำงานเมื่อต้องออกนอกสถานที่ รวมทั้งยังสะดวกและง่ายต่อการบำรุงรักษาระบบอีกด้วย

2) การออกแบบสถาปัตยกรรมซอฟต์แวร์

เนื่องจากสถาปัตยกรรมระบบเป็นเว็บเบสแอปพลิเคชัน ในการออกแบบสถาปัตยกรรมซอฟต์แวร์ในชั้นส่วนตรรกะทางธุรกิจ ผู้วิจัยได้มีการแบ่งแยกการออกแบบเป็น 4 ส่วนหลักๆ ดังรูปที่ 5.4 คือ

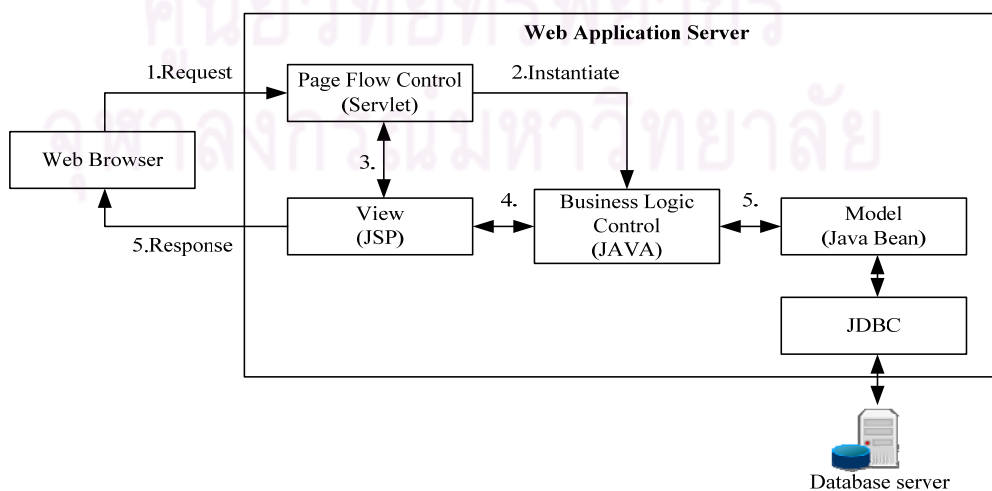
(1) ส่วนการแสดงผล (View) เป็นส่วนที่รับข้อมูลจากผู้ใช้งาน และส่งต่อไปยังส่วนควบคุมการไหลของการแสดงผลต่อไป รวมทั้งแสดงผลการทำงานตามผลของส่วนการควบคุมด้านตรรกะธุรกิจต่อไป

(2) ส่วนการควบคุมการไหลของการแสดงผล (Page Flow Control) เป็นส่วนควบคุมการแสดงผลตามผลที่เกิดขึ้นจากการควบคุมด้านส่วนตรรกะธุรกิจ เช่น เมื่อผู้ใช้กรอกข้อมูลผิดพลาด ระบบต้องเรียกส่วนการแสดงผลส่วนใดมาแสดง นอกจากนั้นยังรองรับการเรียกใช้งานจากส่วนแสดงผลว่าต้องการใช้ส่วนการควบคุมด้านตรรกะธุรกิจส่วนใด เพื่อนำผลลัพธ์ไปแสดงผลในส่วนแสดงผลต่อไป

(3) ส่วนการควบคุมด้านตรรกะธุรกิจ (Business Logic Control) เป็นส่วนควบคุมการทำงานของเครื่องมือให้เป็นไปตามตรรกะทางธุรกิจที่ได้ออกแบบไว้เช่น ในการบันทึกการดำเนินการการวางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป ระบบต้องการบังคับให้ผู้ใช้งานกรอกข้อมูลอะไรบ้าง รวมทั้งตรวจสอบประเภทของข้อมูลที่ใช้กรอกตรงตามที่ระบบกำหนดหรือไม่

(4) ส่วนตรรกะธุรกิจ (Model) เป็นส่วนเชื่อมต่อกับตารางในฐานข้อมูลเพื่อสนับสนุนให้ส่วนการควบคุมด้านตรรกะธุรกิจมาเรียกใช้ เช่น บันทึกข้อมูล เรียกดูข้อมูล ซึ่งทำให้ส่วนการควบคุมด้านตรรกะธุรกิจไม่ต้องเกี่ยวข้องกับภาษาเอสควิแอลเพราะส่วนการทำงานด้านข้อมูลจะเป็นตัวบริหารและจัดการแทน

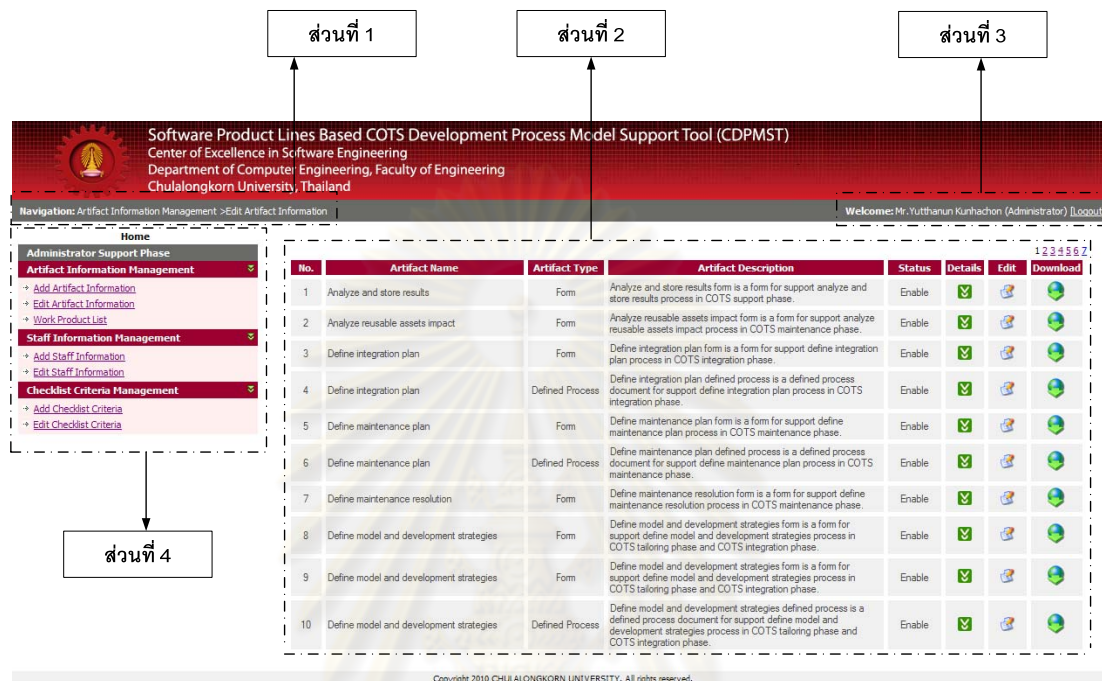
ประโยชน์ของการออกแบบสถาปัตยกรรมซอฟต์แวร์นี้ ทำให้เครื่องมือเกิดคุณลักษณะการนำกลับไปใช้ใหม่หรือเพิ่ม แก้ไข เปลี่ยนแปลง ระบบได้ง่ายขึ้น ยกตัวอย่างเช่น เมื่อองค์กรต้องการเปลี่ยนระบบฐานข้อมูล องค์กรเพียงแก้ไขส่วนการทำงานด้านข้อมูลที่ติดต่อยังโครงสร้างฐานข้อมูลตัวใหม่เท่านั้น ซึ่งจะไม่กระทบต่อส่วนการทำงานหลักอื่นๆ



รูปที่ 5.4 แผนภาพสถาปัตยกรรมซอฟต์แวร์

5.4 การออกแบบส่วนต่อประสานผู้ใช้

ในการออกแบบส่วนต่อประสานผู้ใช้นั้น ผู้วิจัยได้พิจารณาโครงสร้างส่วนต่อประสานกับผู้ใช้โดยแยกออกเป็น 4 ส่วน ดังรูป 5.5



รูปที่ 5.5 โครงสร้างส่วนต่อประสานกับผู้ใช้

ส่วนที่ 1 เป็นส่วนที่แสดงเส้นทางที่ผู้ใช้งานกำลังใช้งาน ซึ่งเริ่มต้นแสดงจากหน้าหลักแล้วเรียงลำดับไปยังส่วนของโปรแกรมต่างๆ ที่ผู้ใช้งานกำลังใช้งานอยู่


ส่วนที่ 2 เป็นส่วนที่แสดงเนื้อหาหรือผลข้อมูลจากการทำงานของระบบ เช่น จากรูปที่ 5.5 เป็นการแสดงผลลัพธ์ของการเรียกดูข้อมูลรายการเอกสารสนับสนุนกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์

ส่วนที่ 3 เป็นส่วนที่แสดงชื่อและนามสกุลของผู้ใช้ที่กำลังใช้งาน พร้อมทั้งบทบาทความรับผิดชอบที่มีต่อโครงการ

ส่วนที่ 4 เป็นส่วนที่แสดงเมนูหลักและเมนูย่อยตามลำดับ โดยระบบจะเปิดการใช้งานเมนูตามลำดับขั้นตอนของการดำเนินงานกระบวนการ





สำหรับการแสดงข้อความเตือนให้ผู้ใช้ทราบถึงผลของการทำงานนั้น ผู้วิจัยได้พิจารณาโครงสร้างส่วนต่อประสาน โดยแบ่งการแสดงข้อความเตือนออกเป็น 2 กรณีคือ กรณีปกติ และกรณีผิดพลาด ในกรณีที่เครื่องมือทำงานได้ปกติระบบจะแสดงผลของการทำงานไว้ด้านล่างของชื่อฟอร์มข้อมูล ดังแสดงได้ดังรูปที่ 5.6

ข้อความแสดงผลการทำงานในกรณีปกติ



Staff Information Form

Inserting staff information is completed.

Title:	Mr.
First Name:	Yutthanun
Last Name:	Kunhachon
Address:	Dept. of Computer Engineering, Faculty of Engineering, Chulalongkorn University Bangkok 10330, THAILAND
Telephone No.:	 +66-(0)2-218-6956 
Mobile No.:	 +66-(0)2-218-5555 
Email:	Yutthanun.K@student.chula.ac.th
Department Name:	Software Engineering Lab.
Skill Information:	<ul style="list-style-type: none"> - Software Process Improvement - Java, JSP and J2ME programming - Unified Modeling Language
Username:	yutthanun
Account Type:	Project Manager

รูปที่ 5.6 หน้าจอแสดงผลการทำงานในกรณีปกติ

สำหรับกรณีผิดพลาดซึ่งในการแสดงข้อความเตือนกรณีผิดพลาดนั้น เครื่องมือจะแสดงข้อความเตือนที่ตำแหน่งที่ผิดพลาด เช่น กรณีที่ผู้ใช้ต้องกรอกข้อมูลทักษะ แล้วผู้ใช้ไม่ได้กรอกข้อมูล เครื่องมือจะแสดงข้อความและสัญลักษณ์เตือนตรงตำแหน่งที่ผู้ใช้ต้องกรอกข้อมูล เป็นต้น ดังแสดงได้ดังรูปที่ 5.7

จุฬาลงกรณ์มหาวิทยาลัย

Add Staff Information Form

Fields marked with an asterisk * are required.

* Title:	<input checked="" type="radio"/> Mr. <input type="radio"/> Ms. <input type="radio"/> Mrs.
* First Name:	Yutthanun
* Last Name:	Kunhachon
* Address:	Dept. of Computer Engineering, Faculty of Engineering, Chulalongkorn University Bangkok 10330, THAILAND
* Telephone No.:	+66-(0)2-218-6956
* Mobile No.:	+66-(0)2-218-5555
* Email:	Yutthanun.K@student.chula.ac.th
* Department Name:	Software Engineering Lab.
* Skill Information:	
* Username:	pm This username already used ID may consist of a-z, 0-9, Underscores, and a single dot (.) Max lengths is ten characters.
* Password:	<input type="password"/> Six characters or more; capitalization matters!
* Account Type:	Project Manager

ข้อความแสดงผลการทำงานในกรณีผิดพลาด

please enter your skill information

This username already used

Six characters or more; capitalization matters!

All of informations are required.

Submit

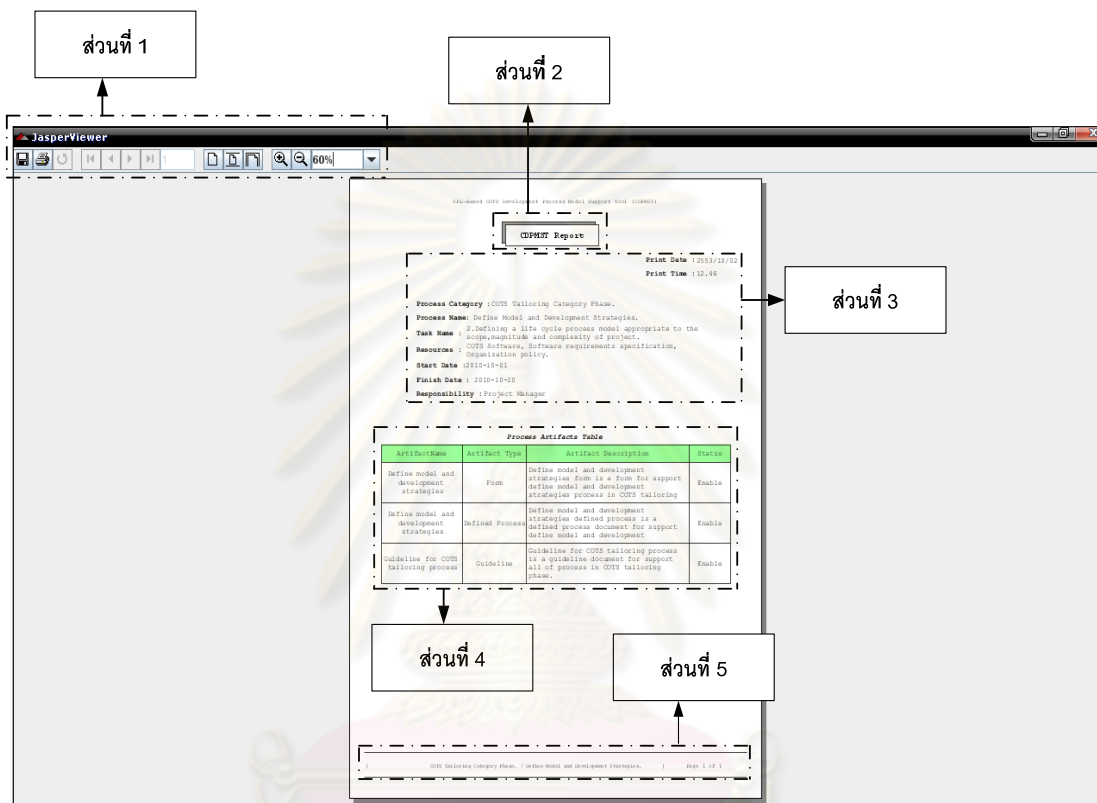
รูปที่ 5.7 หน้าจอแสดงผลการทำงานในกรณีผิดพลาด

นอกจากการออกแบบส่วนต่อประสานผู้ใช้ของเครื่องมือสนับสนุนที่ใช้ในการบันทึก ปรับปรุงข้อมูลและเรียกดูข้อมูลแล้วนั้น ผู้วิจัยได้ออกแบบส่วนต่อประสานกับผู้ใช้ในส่วนของการออกรายงานอีกด้วย เนื่องจากเครื่องมือสนับสนุนที่ได้พัฒนาขึ้นมีการเก็บข้อมูลที่จำเป็นต่อการนำไปใช้ในอนาคตกับโครงการอื่นๆ หรือเพื่อใช้สำหรับการตัดสินใจอื่นๆ ที่เกี่ยวเนื่องกันโดยส่วนของการออกรายงานนั้นแบ่งออกเป็น 2 ส่วน ดังนี้

1) รายงานการบันทึกผล เป็นเอกสารรายงานบันทึกผลซึ่งเป็นผลลัพธ์จากการดำเนินการที่เกิดขึ้นในแต่ละกระบวนการ ซึ่งจะแสดงผลในรูปแบบของข้อความ บันทึกรายละเอียดของกระบวนการที่ได้ถูกดำเนินการ เพื่อบันทึกผลการดำเนินการเพื่อง่ายต่อการบำรุงรักษา

2) รายงานการประเมิน เป็นเอกสารรายงานที่ใช้ในการวิเคราะห์ซึ่งใช้ในกระบวนการประเมินเพื่อช่วยให้ทีมผู้ประเมินสามารถดำเนินการประเมินได้ง่ายขึ้น

โครงสร้างส่วนต่อประสานกับผู้ใช้ในส่วนการออกรายงานการบันทึกผลแบ่งออกเป็น 5 ส่วน ได้ดังรูปที่ 5.8



รูปที่ 5.8 โครงสร้างส่วนต่อประสานกับผู้ใช้ส่วนการออกรายงานการบันทึกผล

ส่วนที่ 1 เป็นส่วนที่ใช้ในการควบคุมรายงาน เช่น ปุ่มบันทึกรายงาน ปุ่มพิมพ์รายงานและส่วนการแสดงผลของรายงาน เป็นต้น

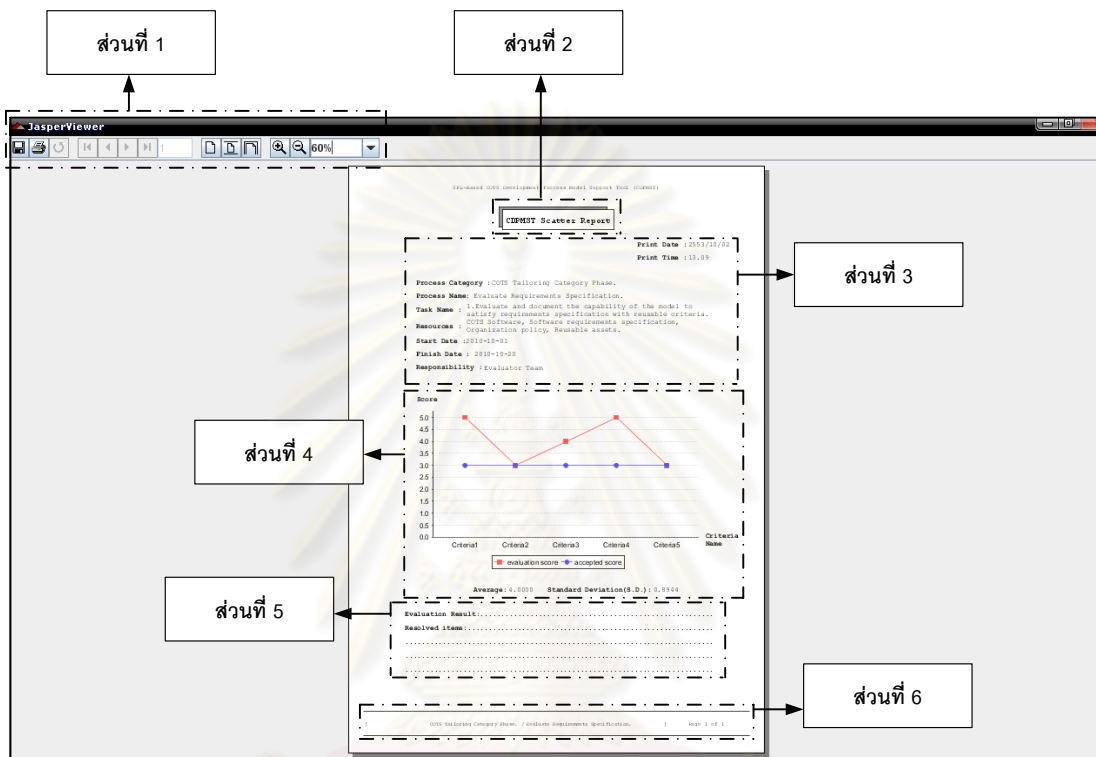
ส่วนที่ 2 เป็นส่วนที่แสดงชื่อของเอกสารรายงาน

ส่วนที่ 3 เป็นส่วนที่บันทึกข้อมูลที่เกิดจากข้อมูลนำเข้าของผู้ใช้งาน

ส่วนที่ 4 เป็นส่วนที่แสดงถึงรายการเอกสารสนับสนุนกระบวนการในแต่ละกระบวนการนั้นๆ ที่จำเป็นต้องใช้ พร้อมรายละเอียด

ส่วนที่ 5 เป็นส่วนที่บอกถึงชื่อกระบวนการที่ดำเนินการ พร้อมทั้งระยะของขั้นตอนการดำเนินการ และจำนวนหน้าที่แสดงรายงาน

โครงสร้างส่วนต่อประสานกับผู้ใช้ในส่วนการออกรายงานการประเมิน แบ่งออกเป็น 6 ส่วน ได้ดังรูปที่ 5.9



รูปที่ 5.9 โครงสร้างส่วนต่อประสานกับผู้ใช้ในส่วนการออกรายงานการประเมิน

ส่วนที่ 1 เป็นส่วนที่ใช้ในการควบคุมรายงาน เช่น ปุ่มบันทึกรายงาน ปุ่มพิมพ์รายงานและส่วนการแสดงผลของรายงาน เป็นต้น

ส่วนที่ 2 เป็นส่วนที่แสดงชื่อของเอกสารรายงาน

ส่วนที่ 3 เป็นส่วนที่บันทึกข้อมูลที่เกิดจากข้อมูลนำเข้าของผู้ใช้งาน

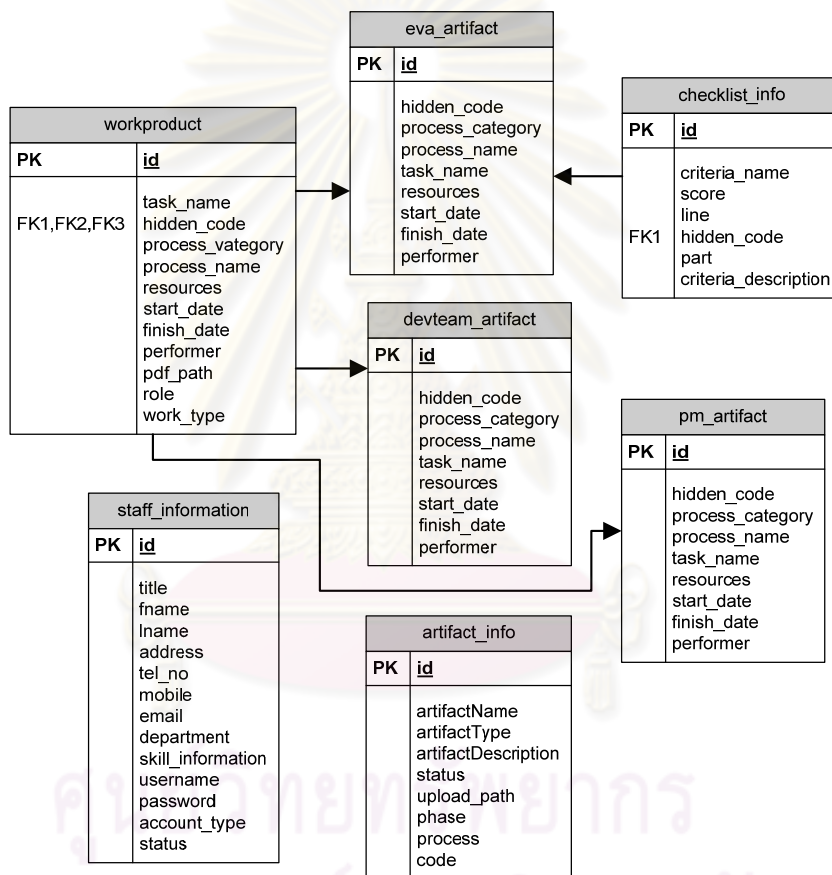
ส่วนที่ 4 เป็นส่วนที่แสดงถึงกราฟการกระจาย (Scatter graph) ซึ่งเป็นผลลัพธ์จากการประเมิน นอกจากนั้นยังประกอบด้วย ค่าเฉลี่ย (\bar{X}) และค่าเบี่ยงเบนมาตรฐาน (Standard Deviation: S.D.) เพื่อใช้ในการพิจารณาการประเมินสำหรับทีมผู้ประเมิน

ส่วนที่ 5 เป็นส่วนที่ใช้ในการบันทึกผลการประเมิน พร้อมทั้งบันทึกรายการที่ต้องแก้ไขที่เกิดจากผลการประเมินที่ได้ เพื่อปรับปรุงกระบวนการให้มีประสิทธิภาพมากขึ้นต่อไป

ส่วนที่ 6 เป็นส่วนที่บอกถึงชื่อกระบวนการที่ดำเนินการ พร้อมทั้งระยะของขั้นตอนการดำเนินการ และจำนวนหน้าที่แสดงรายงาน

5.5 การออกแบบฐานข้อมูลเชิงสัมพันธ์

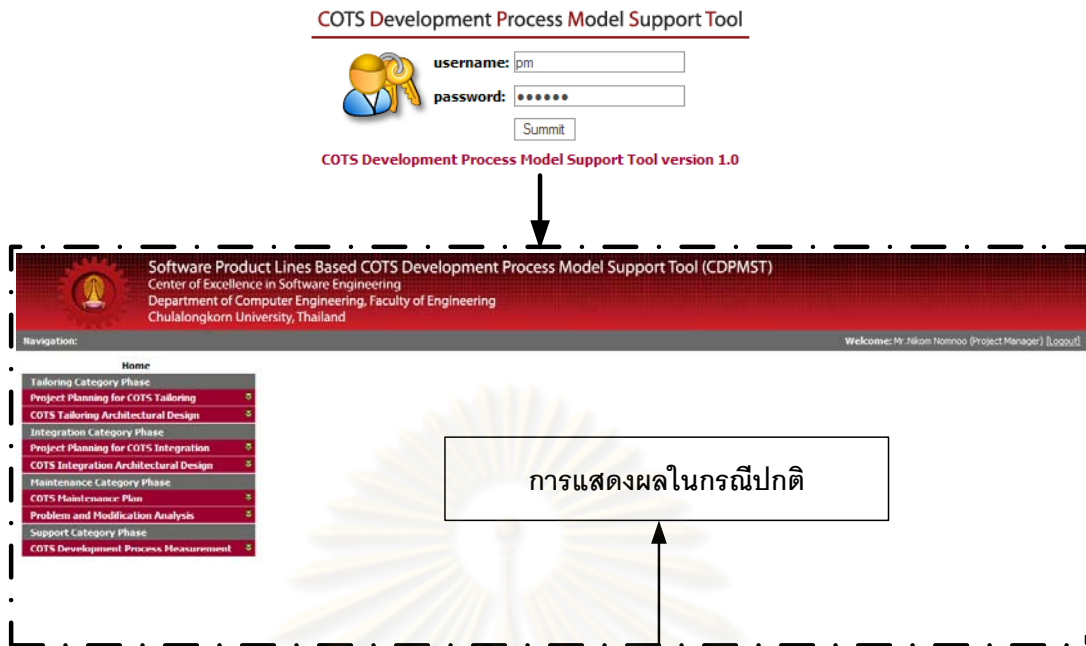
การออกแบบฐานข้อมูลของระบบตามแผนภาพคลาสที่ได้ออกแบบไว้ที่มีลักษณะเป็นฐานข้อมูลเชิงสัมพันธ์ผู้วิจัยได้ออกแบบฐานข้อมูลเชิงกายภาพ ซึ่งแสดงให้เห็นตารางข้อมูลและความสัมพันธ์ระหว่างตารางข้อมูลต่างๆ ภายในระบบ ดังรูปที่ 5.10 (รายละเอียดตารางข้อมูลของระบบแสดงในภาคผนวก จ.)



รูปที่ 5.10 โครงสร้างฐานข้อมูลเชิงสัมพันธ์

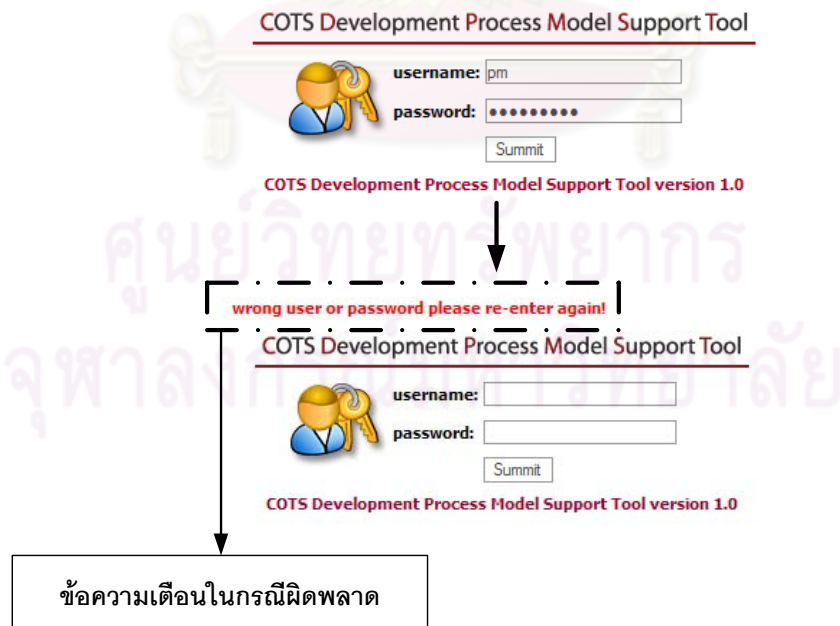
5.6 การออกแบบความมั่นคงของเครื่องมือสนับสนุน

การออกแบบความมั่นคงของเครื่องมือสนับสนุนในส่วนของสิทธิในการใช้งานระบบเมื่อผู้ใช้งานระบบกรอกรหัสผู้ใช้งานและรหัสผ่าน ผู้วิจัยได้พิจารณาโครงสร้างโดยแบ่งออกเป็น 2 กรณี คือ กรณีปกติและกรณีผิดพลาด ในกรณีปกติเมื่อผู้ใช้งานระบบกรอกรหัสผู้ใช้งานและรหัสผ่านถูกต้อง ดังแสดงได้ดังรูปที่ 5.11



รูปที่ 5.11 หน้าจอแสดงผลการทำงานเมื่อผู้ใช้งานระบบกรอกรหัสผู้ใช้งานและรหัสผ่าน
ในกรณีปกติ

สำหรับกรณีผิดพลาดเมื่อผู้ใช้งานระบบกรอกรหัสผู้ใช้งานและรหัสผ่านไม่ถูกต้อง
เครื่องมือจะแสดงข้อความเตือน ดังแสดงได้ดังรูปที่ 5.12



รูปที่ 5.12 หน้าจอแสดงผลการทำงานเมื่อผู้ใช้งานระบบกรอกรหัสผู้ใช้งานและรหัสผ่าน
ในกรณีผิดพลาด

บทที่ 6

การพัฒนาและประเมินเครื่องมือสนับสนุน

เมื่อผู้วิจัยได้วิเคราะห์และออกแบบเครื่องมือสนับสนุนเป็นที่เรียบร้อยแล้วนั้น ในขั้นตอนถัดไปผู้วิจัยจะพัฒนาและทดสอบเครื่องมือสนับสนุน ซึ่งในบทนี้จะกล่าวถึงเครื่องมือที่ใช้ในการพัฒนาเครื่องมือสนับสนุน ขั้นตอนในการพัฒนา และวิธีการประเมินเครื่องมือสนับสนุน

6.1 เครื่องมือที่ใช้ในการพัฒนาเครื่องมือสนับสนุน

สำหรับเครื่องมือที่ใช้ในการพัฒนาเครื่องมือสนับสนุนนั้น ผู้วิจัยได้แบ่งออกเป็น 2 กลุ่ม โดยมีรายละเอียดดังต่อไปนี้

1) ฮาร์ดแวร์

การกำหนดคุณสมบัติของฮาร์ดแวร์ที่ใช้ในการพัฒนาเครื่องมือสนับสนุน สามารถจำแนกออกเป็น 3 กลุ่ม คือ ฮาร์ดแวร์ของเครื่องคอมพิวเตอร์ที่ใช้ในการพัฒนาเครื่องมือสนับสนุน ฮาร์ดแวร์ของเครื่องลูกข่ายและฮาร์ดแวร์ของเครื่องแม่ข่าย โดยมีรายละเอียดดังต่อไปนี้

(1) เครื่องคอมพิวเตอร์ที่ใช้ในการพัฒนาเครื่องมือสนับสนุน

- หน่วยประมวลผล อินเทลคอทดูโอ 1.73 กิกะเฮิร์ตซ์
- หน่วยความจำหลัก 1 กิกะไบต์
- ฮาร์ดดิสก์ความจุ 80 กิกะไบต์
- รองรับการโอนถ่ายข้อมูลเครือข่ายที่ความเร็ว 10/100 เมกกะบิตต่อวินาที

(2) เครื่องคอมพิวเตอร์ลูกข่ายที่รองรับการประมวลผลของเครื่องมือสนับสนุน

- หน่วยประมวลผล อินเทลเซลเลอลอน ความเร็ว 2.4 กิกะเฮิร์ตซ์ ขึ้นไป
- หน่วยความจำหลัก ดิดีอาร์-เอสดีแรม 512 เมกกะไบต์
- ฮาร์ดดิสก์ไอดีอีเอทีเอ-100 หรือ 133 เมกกะเฮิร์ตซ์ความจุ 30 กิกะไบต์
- รองรับการโอนถ่ายข้อมูลเครือข่ายที่ความเร็ว 10/100 เมกกะบิตต่อวินาที

- (3) เครื่องคอมพิวเตอร์แม่ข่ายที่รองรับการให้บริการและประมวลผลด้านตรรกะทางธุรกิจและด้านฐานข้อมูล
- หน่วยประมวลผล อินเทลคอทดูโอ 1.73 กิกะเฮิร์ตซ์
 - หน่วยความจำหลัก 1 กิกะไบต์
 - ฮาร์ดดิสก์ไอดีอีเอทีเอ-100 หรือ 133 เมกกะเฮิร์ตซ์ความจุ 80 กิกะไบต์
 - รองรับการโอนถ่ายข้อมูลเครือข่ายที่ความเร็ว 10/100 เมกกะบิตต่อวินาที

2) ซอฟต์แวร์

ในส่วนของซอฟต์แวร์สามารถจำแนกตามลักษณะการใช้งานในช่วงการพัฒนาโดยสามารถจำแนกออกเป็น 4 กลุ่ม คือ

(1) ระบบปฏิบัติการ

- วินโดวส์เอ็กซ์พีโพรเฟสชันแนล สำหรับเครื่องคอมพิวเตอร์พัฒนาระบบ
- วินโดวส์ 2003 เซิร์ฟเวอร์ขึ้นไป สำหรับเครื่องแม่ข่ายที่รองรับการให้บริการและประมวลผลด้านตรรกะทางธุรกิจ
- วินโดวส์ 2003 เซิร์ฟเวอร์ขึ้นไป สำหรับเครื่องแม่ข่ายที่รองรับการให้บริการและประมวลผลด้านฐานข้อมูล
- วินโดวส์เอ็กซ์พี ขึ้นไป สำหรับเครื่องลูกข่าย

(2) เครื่องมือที่ใช้ในการออกแบบและจัดทำเอกสาร

- ไมโครซอฟต์แอฟฟิซ 2007
- อะโดบีอะโครแบท 9.0
- ไมโครซอฟต์แอฟฟิซ 2007

(3) เครื่องมือที่ใช้ในการพัฒนาส่วนต่อประสานผู้ใช้

- เว็บบราวน์เซอร์อินเทอร์เน็ตเอ็กซ์โพลเลอร์เวอร์ชัน 7
- อะโดบีไฟโตชอปซีเอส 3
- มาโครมีเดียดรีมวีเวอร์

- (4) เครื่องมือที่ใช้ในการพัฒนาส่วนให้บริการตระกะทางธุรกิจและฐานข้อมูล
- เว็บเซิร์ฟเวอร์อาปาเช่ จากاتا ทอมแคท เวอร์ชัน 6.0.18
 - คอมไพเลอร์จาวา สแตนด์ดาร์ดเอ็ดิชั่น เวอร์ชัน 1.6.0_02
 - ฐานข้อมูลเชิงสัมพันธ์มายเอสคิวแอล 5.0.84
 - โปรแกรมจัดการฐานข้อมูลเชิงสัมพันธ์มายเอสคิวแอลยอค เวอร์ชัน 8.14
 - เอ็ดิตพัลส์ เท็กซ์เอ็ดิเตอร์เวอร์ชัน 2.0
 - เน็ตปีนส์ ไอดีอี เวอร์ชัน 6.7.1
 - เว็บเบราวน์เซอร์อินเทอร์เน็ตเอ็กโพลเลอร์เวอร์ชัน 7

6.2 ขั้นตอนในการพัฒนาเครื่องมือสนับสนุน

ในการพัฒนาเครื่องมือสนับสนุนผู้วิจัยได้เลือกใช้ภาษาโปรแกรมในการพัฒนาเครื่องมือสนับสนุน คือ ภาษาโปรแกรมจาวา ภาษาโปรแกรมจาวาเซิร์ฟเล็ท และภาษาโปรแกรมเจเอสพีเนื่องจากในการวิเคราะห์และออกแบบเครื่องมือสนับสนุน ผู้วิจัยได้ใช้การวิเคราะห์การออกแบบเชิงวัตถุซึ่งภาษาโปรแกรมทั้งสองสนับสนุนคุณสมบัติและลักษณะของการออกแบบเชิงวัตถุซึ่งขั้นตอนในการพัฒนาเครื่องมือสนับสนุน มีดังต่อไปนี้

1) พัฒนาหน้าจอต้นแบบของเครื่องมือสนับสนุน

ในส่วนนี้ ผู้วิจัยได้เริ่มต้นพัฒนาหน้าจอต้นแบบของเครื่องมือสนับสนุน โดยแสดงภาพรวมและหน้าที่การทำงานของเครื่องมือสนับสนุนทั้งหมด ทั้งรูปแบบโครงสร้างของส่วนต่อประสานกับผู้ใช้ในแต่ละหน้าจอการทำงาน รวมถึงข้อมูลนำเข้า และข้อมูลนำออกที่เกิดขึ้นในแต่ละหน้าที่การทำงาน ซึ่งข้อดีของการพัฒนาในส่วนนี้ คือ สามารถทำให้ผู้ใช้งานเข้าใจถึงภาพรวมและหน้าที่การทำงานของเครื่องมือสนับสนุนที่พัฒนาขึ้นได้อีกด้วย

2) พัฒนาระบบฐานข้อมูล

ในส่วนนี้ ผู้วิจัยได้สร้างพิวลในแต่ละตาราง รวมถึงความสัมพันธ์ของแต่ละตารางบนระบบฐานข้อมูลมายเอสคิวแอล จากแบบจำลองเชิงกายภาพที่ได้ออกแบบไว้ในบทที่ 5

3) พัฒนาจาวาปีน

ในส่วนนี้ ผู้วิจัยได้สร้างจาวาปีน ซึ่งเป็นส่วนประกอบหนึ่งของเครื่องมือสนับสนุนเพื่อใช้ในการบันทึกข้อมูล การเปลี่ยนแปลงข้อมูล รวมถึงการตรวจสอบความต้องการของข้อมูลก่อนบันทึกหรือเปลี่ยนแปลงข้อมูลในแต่ละตารางบนระบบฐานข้อมูลที่ได้พัฒนาขึ้นไว้

4) พัฒนาเจเอสพี

ในส่วนนี้ ผู้วิจัยได้สร้างเจเอสพี ซึ่งเป็นส่วนนำเข้าข้อมูล และ/หรือนำออกข้อมูลจากจาวาปีนไปแสดงผลบนหน้าจอ ตามโครงสร้างส่วนต่อประสานกับผู้ใช้ที่ได้ออกแบบไว้

5) พัฒนาจาวาเซิร์ฟเล็ต

ในส่วนนี้ ผู้วิจัยได้สร้างจาวาเซิร์ฟเล็ต ซึ่งเป็นตัวควบคุมการไหลของการแสดงผลของเจเอสพีที่สร้างขึ้น รวมทั้งควบคุมการทำงานของจาวาปีนที่สร้างขึ้น เพื่อให้จาวาปีนและเจเอสพีที่สร้างขึ้นทำงานสอดคล้องกันตามตรรกะทางธุรกิจที่ได้กำหนดไว้ เช่น ก่อนการบันทึกข้อมูลลงฐานข้อมูลจาวาเซิร์ฟเล็ตจะเรียกให้จาวาปีนตรวจสอบข้อมูลว่าครบถ้วนและถูกต้องหรือไม่ ถ้ามีข้อผิดพลาดเกิดขึ้น จาวาเซิร์ฟเล็ตควบคุมให้เจเอสพีแสดงผลข้อผิดพลาดตามข้อมูลที่จาวาปีนได้ตรวจสอบไว้

6.3 กรณีทดสอบเครื่องมือสนับสนุน

ในการทดสอบเครื่องมือสนับสนุนที่พัฒนาขึ้นนั้น ผู้วิจัยได้ใช้การทดสอบแบบกล่องดำ (Black Box Testing) ซึ่งข้อมูลที่ใช้ในการทดสอบนั้นผู้วิจัยได้จำลองขึ้นให้มีลักษณะใกล้เคียงกับข้อมูลจริงของระบบมากที่สุด สำหรับข้อมูลที่เกี่ยวข้องในการทดสอบเครื่องมือ ได้แก่

- 1) ข้อมูลผู้ใช้งานระบบ
- 2) ข้อมูลอาร์ทิแฟกและเอกสารสนับสนุนกระบวนการ
- 3) ข้อมูลเกณฑ์การใช้จ่าย
- 4) ข้อมูลการประเมินกระบวนการที่เกิดขึ้นกับโครงการ
- 5) ข้อมูลรายการกิจกรรมที่เกิดขึ้นในโครงการ

ในการสร้างกรณีทดสอบขึ้น ผู้วิจัยเริ่มต้นจากพิจารณาเป้าหมายและผลลัพธ์ที่คาดหวังไว้ตามความต้องการด้านหน้าที่ของเครื่องมือสนับสนุน โดยกำหนดข้อมูลทดสอบและบันทึกผลการทดสอบดังตารางที่ 6.1 ที่ใช้ทดสอบการเพิ่มข้อมูลผู้ใช้งานระบบเข้าสู่ระบบ

ตารางที่ 6.1 ตัวอย่างกรณีทดสอบการเพิ่มข้อมูลผู้ใช้งานระบบเข้าสู่ระบบ

ชื่อระบบ	เครื่องมือสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์
ชื่อหน้าที่งานหลัก	จัดการข้อมูลผู้ใช้งานระบบ
เลขที่กรณีทดสอบ	T0104 (อ้างอิงความต้องการด้านหน้าที่รหัส FR0104)
ชื่อกรณีทดสอบ	เพิ่มข้อมูลสารสนเทศผู้ใช้ระบบ
วัตถุประสงค์	เพื่อทดสอบการเพิ่มข้อมูลผู้ใช้งานระบบเข้าสู่ระบบ
บทบาทผู้ใช้งานระบบ	ผู้จัดการระบบ
ข้อมูลนำเข้า	1) คำนำหน้าชื่อ 2) ชื่อ 3) ชื่อสกุล 4) ที่อยู่ 5) เบอร์โทรศัพท์ 6) เบอร์โทรศัพท์เคลื่อนที่ 7) ไปรษณีย์อิเล็กทรอนิกส์ 8) ชื่อหน่วยงาน 9) ข้อมูลทักษะหรือความชำนาญ 10) รหัสผู้ใช้งาน 11) รหัสผ่าน 12) ประเภทผู้ใช้งาน
ผลลัพธ์ที่คาดหวัง (กรณีปกติ)	ระบบแสดงข้อความให้ผู้ใช้ทราบว่าได้เก็บข้อมูลที่กรอกจากหน้าจอก่อนลงในฐานข้อมูลถูกต้องและครบถ้วน
กรณีคาดหวัง (กรณีผิดพลาด)	กรณีผู้ใช้กรอกข้อมูลไม่ครบถ้วน หน้าจอจะแจ้งข้อความเตือนให้กับผู้ใช้ระบุข้อมูลในฟิลด์ที่กำหนดไว้

ตารางที่ 6.1 ตัวอย่างกรณีทดสอบการเพิ่มข้อมูลผู้ใช้งานระบบเข้าสู่ระบบ (ต่อ)

ข้อมูลทดสอบ	กรณีปกติ แสดงดังตารางที่ 6.2 กรณีผิดพลาด แสดงดังตารางที่ 6.3
ผลการทดสอบ	เครื่องมือทำงานในกรณีปกติได้ถูกต้องครบถ้วน และในกรณีผิดพลาดเครื่องมือแสดงข้อความเตือนความผิดพลาดที่เกิดขึ้นได้ ดังแสดงได้ดังรูปที่ 6.1 และ 6.2 ตามลำดับ
สรุปผลการทดสอบ	<input checked="" type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน หมายเหตุ:

ตารางที่ 6.2 ข้อมูลทดสอบของกรณีทดสอบที่ T0104 (กรณีปกติ)

ข้อมูลนำเข้า		ตัวอย่างข้อมูล
รายชื่อฟิลด์ข้อมูล	บังคับ/ไม่บังคับ กรอกข้อมูล	กรณีปกติ
1) คำนำหน้าชื่อ	บังคับ	Mr.
2) ชื่อ	บังคับ	Yutthanun
3) ชื่อสกุล	บังคับ	Kunhachon
4) ที่อยู่	บังคับ	Dept. of Computer Engineering, Faculty of Engineering, Chulalongkorn University Bangkok 10330, THAILAND
5) เบอร์โทรศัพท์	บังคับ	+66-(0)2-218-6956
6) เบอร์โทรศัพท์เคลื่อนที่	บังคับ	+66-(0)2-218-5555
7) ไปรษณีย์อิเล็กทรอนิกส์	บังคับ	Yutthanun.K@student.chula.ac.th
8) ชื่อหน่วยงาน	บังคับ	Software Engineering Lab.
9) ข้อมูลทักษะหรือความ ชำนาญ	บังคับ	- Software Process Improvement - Java, JSP and J2ME programming - Unified Modeling Language
10) รหัสผู้ใช้งาน	บังคับ	yutthanun

ตารางที่ 6.2 ข้อมูลทดสอบของกรณีทดสอบที่ T0104 (กรณีปกติ) (ต่อ)

ข้อมูลนำเข้า		ตัวอย่างข้อมูล
รายชื่อฟิลด์ข้อมูล	บังคับ/ไม่บังคับ กรอกข้อมูล	กรณีปกติ
11) รหัสผ่าน	บังคับ	*****
12) ประเภทผู้ใช้งาน	บังคับ	Project Manager

ตารางที่ 6.3 ข้อมูลทดสอบของกรณีทดสอบที่ T0104 (กรณีผิดพลาด)

ข้อมูลนำเข้า		ตัวอย่างข้อมูล
รายชื่อฟิลด์ข้อมูล	บังคับ/ไม่บังคับ กรอกข้อมูล	กรณีปกติ
1) คำนำหน้าชื่อ	บังคับ	Mr.
2) ชื่อ	บังคับ	Yutthanun
3) ชื่อสกุล	บังคับ	Kunhachon
4) ที่อยู่	บังคับ	Dept. of Computer Engineering, Faculty of Engineering, Chulalongkorn University Bangkok 10330, THAILAND
5) เบอร์โทรศัพท์	บังคับ	+66-(0)2-218-6956
6) เบอร์โทรศัพท์เคลื่อนที่	บังคับ	+66-(0)2-218-5555
7) ไปรษณีย์อิเล็กทรอนิกส์	บังคับ	Yutthanun.K@student.chula.ac.th
8) ชื่อหน่วยงาน	บังคับ	Software Engineering Lab.
9) ข้อมูลทักษะหรือความ ชำนาญ	บังคับ	ผู้ใช้ไม่กรอกข้อมูล
10) รหัสผู้ใช้งาน	บังคับ	pm (ชื่อรหัสผู้ใช้งานซ้ำ)
11) รหัสผ่าน	บังคับ	*****
12) ประเภทผู้ใช้งาน	บังคับ	Project Manager

ในกรณีหน้าจอของเครื่องมือจะแสดงผลการทำงานดังรูปที่ 6.1 และในกรณีผิดพลาดหน้าจอของเครื่องมือจะแสดงผลการทำงานดังรูปที่ 6.2

Staff Information Form	
Inserting staff information is completed.	
Title:	Mr.
First Name:	Yutthanun
Last Name:	Kunhachon
Address:	Dept. of Computer Engineering, Faculty of Engineering, Chulalongkorn University Bangkok 10330, THAILAND
Telephone No.:	+66-(0)2-218-6956
Mobile No.:	+66-(0)2-218-5555
Email:	Yutthanun.K@student.chula.ac.th
Department Name:	Software Engineering Lab.
Skill Information:	- Software Process Improvement - Java, JSP and J2ME programming - Unified Modeling Language
Username:	yutthanun
Account Type:	Project Manager

รูปที่ 6.1 หน้าจอแสดงผลการบันทึกข้อมูลในกรณีปกติ

Add Staff Information Form	
Fields marked with an asterisk * are required.	
* Title:	<input checked="" type="radio"/> Mr. <input type="radio"/> Ms. <input type="radio"/> Mrs.
* First Name:	<input type="text" value="Yutthanun"/>
* Last Name:	<input type="text" value="Kunhachon"/>
* Address:	<input type="text" value="Dept. of Computer Engineering, Faculty of Engineering, Chulalongkorn University Bangkok 10330, THAILAND"/>
* Telephone No.:	<input type="text" value="+66-(0)2-218-6956"/>
* Mobile No.:	<input type="text" value="+66-(0)2-218-5555"/>
* Email:	<input type="text" value="Yutthanun.K@student.chula.ac.th"/>
* Department Name:	<input type="text" value="Software Engineering Lab."/>
* Skill Information:	<input type="text" value=""/> <small>please enter your skill information</small>
* Username:	<input type="text" value="pm"/> <small>This username already used ID may consist of a-z, 0-9, underscores, and a single dot (.) Max lengths is ten characters.</small>
* Password:	<input type="text"/> <small>Six characters or more; capitalization matters!</small>
* Account Type:	<input type="text" value="Project Manager"/>
All of informations are required.	<input type="button" value="Submit"/>

รูปที่ 6.2 หน้าจอแสดงผลการบันทึกข้อมูลในกรณีผิดพลาด

6.4 วิธีการประเมินเครื่องมือสนับสนุน

เนื่องจากเครื่องมือสนับสนุนที่ได้นำเสนอขึ้น เพื่อสนับสนุนกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์ที่ผู้วิจัยได้นำเสนอขึ้นในบทที่ 4 ดังนั้นในส่วนนี้ผู้วิจัยจะกล่าวถึงวิธีการประเมินเครื่องมือสนับสนุนว่ามีองค์ประกอบครบถ้วนตามข้อกำหนดของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีเฟลลี้ 1517 กลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ ซึ่งวิธีการประเมินผู้วิจัยได้ใช้รายการตรวจสอบข้อกำหนดของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีเฟลลี้ 1517 กลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์นั้นสามารถใช้ระบบหรือหน้าที่การทำงานใด

ตารางที่ 6.4 ตารางเปรียบเทียบระหว่างหน้าที่การทำงานของเครื่องมือสนับสนุนกับข้อกำหนดของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีเฟลลี้ 1517 กลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ (กระบวนการการพัฒนา)

หน้าที่การทำงานของเครื่องมือสนับสนุน	ข้อกำหนดของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีเฟลลี้ 1517 กลุ่มกระบวนการพัฒนาผลิตภัณฑ์จากสินทรัพย์ (Development Process)
ระบบจัดการวางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป	Process implementation, System requirements analysis, System architectural design, Software requirements analysis, System architectural design
ระบบจัดการออกแบบสถาปัตยกรรมการปรับแต่งซอฟต์แวร์สำเร็จรูป	Software architectural design, Software detailed design
ระบบจัดการปรับแต่งซอฟต์แวร์สำเร็จรูป	Software coding and testing, Software integration, System integration, Software qualification testing, System qualification testing
ระบบจัดการติดตั้งซอฟต์แวร์	Software installation

ตารางที่ 6.4 ตารางเปรียบเทียบระหว่างหน้าที่การทำงานของเครื่องมือสนับสนุนกับข้อกำหนดของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีพีแอล 1517 กลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ (กระบวนการการพัฒนา) (ต่อ)

หน้าที่การทำงานของเครื่องมือสนับสนุน	ข้อกำหนดของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีพีแอล 1517 กลุ่มกระบวนการพัฒนาผลิตภัณฑ์จากสินทรัพย์ (Development Process)
ระบบจัดการวางแผนโครงการบูรณาการซอฟต์แวร์สำเร็จรูป	Process implementation, System requirements analysis, System architectural design, Software requirements analysis, System architectural design
ระบบจัดการออกแบบสถาปัตยกรรมการบูรณาการซอฟต์แวร์สำเร็จรูป	Software architectural design, Software detailed design
ระบบจัดการบูรณาการซอฟต์แวร์สำเร็จรูป	Software coding and testing, Software integration, System integration, Software qualification testing, System qualification testing

ตารางที่ 6.5 ตารางเปรียบเทียบระหว่างหน้าที่การทำงานของเครื่องมือสนับสนุนกับข้อกำหนดของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีพีแอล 1517 กลุ่มกระบวนการการพัฒนาผลิตภัณฑ์จากสินทรัพย์ (กระบวนการบำรุงรักษา)

หน้าที่การทำงานของเครื่องมือสนับสนุน	ข้อกำหนดของกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีพีแอล 1517 กลุ่มกระบวนการพัฒนาผลิตภัณฑ์จากสินทรัพย์ (Maintenance Process)
ระบบจัดการวางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป	Process implementation, Problem and modification analysis
ระบบจัดการวิเคราะห์ปัญหาและการดัดแปลง	Problem and modification analysis
ระบบจัดการดัดแปลง ย้ายและยกเลิกซอฟต์แวร์	Modification implementation, Migration, Software retirement

ตารางที่ 6.6 หน้าี่การทำงานอื่นๆ ของเครื่องมือสนับสนุน

หน้าี่การทำงานของเครื่องมือสนับสนุน	คำอธิบาย
ระบบจัดการสนับสนุนการยอมรับซอฟต์แวร์	มีหน้าี่จัดการกับข้อมูลการส่งมอบผลิตภัณฑ์ซอฟต์แวร์และจัดเตรียมการฝึกฝนการใช้งานของผู้ใช้ผลิตภัณฑ์ซอฟต์แวร์
ระบบจัดการวัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป	มีหน้าี่จัดการกับข้อมูลการวัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป
ระบบจัดการอาร์ทิแฟกต์และเอกสารสนับสนุนและเอกสารผลลัพธ์ที่ได้จากการดำเนินกระบวนการ	มีหน้าี่จัดการกับข้อมูลเอกสารสนับสนุนและเอกสารผลลัพธ์ที่ได้จากการดำเนินกระบวนการ
ระบบจัดการข้อมูลผู้ใช้	มีหน้าี่จัดการกับข้อมูลสารสนเทศของผู้ใช้งานระบบ
ระบบจัดการเกณฑ์รายการตรวจสอบ	มีหน้าี่จัดการกับข้อมูลเกณฑ์ของรายการตรวจสอบเพื่อใช้ในกระบวนการประเมิน

บทที่ 7

สรุปผลการวิจัยและข้อเสนอแนะ

จากการวิจัยเพื่อออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานนั้นผู้วิจัยได้สรุปผลการวิจัย และมีข้อเสนอแนะต่างๆ ดังนี้

7.1 สรุปผลการวิจัย

ในงานวิจัยนี้ผู้วิจัยได้นำเสนอกระบวนการออกแบบและพัฒนากระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน โดยกระบวนการที่นำเสนอ นั้นกล่าวถึงองค์ประกอบพื้นฐานที่ผู้วิจัยได้นำมาใช้ในการวิเคราะห์และออกแบบ เพื่อให้ได้กระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่จะนำไปพัฒนาเป็นเครื่องมือสนับสนุนต่อไป ซึ่งผลของการวิจัยสามารถสรุปได้ดังต่อไปนี้

1) กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานประกอบด้วย 3 กลุ่มกระบวนการหลักและ 1 กลุ่มกระบวนการสนับสนุน ซึ่งกลุ่มกระบวนการหลักประกอบด้วย กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป กระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป และกระบวนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป โดยกลุ่มกระบวนการสนับสนุนประกอบด้วย กระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป ซึ่งกระบวนการที่ได้นั้นแบ่งออกเป็น 3 ส่วนหลัก คือ ชั้นแบบจำลองเชิงแนวคิด ชั้นแบบจำลองเชิงกระแสนงานและชั้นแบบจำลองเชิงรายละเอียด โดยมุมมองการออกแบบจะเริ่มจากชั้นแบบจำลองเชิงแนวคิดไปยังชั้นแบบจำลองเชิงรายละเอียด รวมทั้งแสดงถึงวิธีการในการประเมินกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานที่นำเสนอขึ้นโดยการนำวิธีการทวนสอบแบบวิธีการตรวจตลอด (Walkthrough) มาใช้ในการประเมินกระบวนการที่นำเสนอซึ่งผลของการวิเคราะห์และออกแบบกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานนั้น ได้แสดงให้เห็นถึงองค์ประกอบพื้นฐานที่จำเป็นต่อองค์กร เพื่อให้องค์กรสามารถนำไปประยุกต์ใช้ให้เข้ากับบรรทัดฐานของแต่ละองค์กรต่อไป

2) จากการออกแบบและพัฒนากระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานได้มาซึ่งเอกสารสนับสนุนกระบวนการ ซึ่งประกอบด้วย 4 ส่วนดังต่อไปนี้

(1) นิยามกระบวนการ (Defined process) คือ เอกสารที่ระบุขั้นตอนและหน้าที่ที่ต้องทำในแต่ละกระบวนการเพื่อใช้สนับสนุนให้ผู้ดำเนินการสามารถนำไปใช้ดำเนินการได้อย่างสมบูรณ์

(2) ฟอร์ม (Form) คือ เอกสารที่นำมาใช้เพื่อการเก็บข้อมูลที่สำคัญเพื่อนำมาใช้ในการวิเคราะห์ข้อมูลของการดำเนินงานของกระบวนการ

(3) รายการตรวจสอบ (Checklist) คือ เอกสารที่ระบุถึงสิ่งที่ต้องทำในกิจกรรมนั้นๆ เพื่อให้ผู้ดำเนินการสามารถทวนสอบกิจกรรมได้ว่าได้กระทำครบถ้วนหรือไม่

(4) แนวทางในการดำเนินการ (Guideline) คือ เอกสารที่ช่วยให้ผู้ดำเนินการทราบถึงวิธีในการทำงานของกิจกรรมนั้นๆ ซึ่งช่วยให้ผู้ดำเนินการปฏิบัติหน้าที่ได้ง่ายขึ้น

3) เครื่องมือสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานประกอบด้วยระบบย่อยทั้งหมด 15 ระบบตามลักษณะของการทำงานที่เกิดขึ้นของกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานซึ่งความต้องการด้านหน้าที่ของแต่ละระบบย่อยที่นำเสนอขึ้นนั้นเป็นเพียงความต้องการพื้นฐานที่ช่วยสนับสนุนให้องค์กรสามารถปฏิบัติตามได้ตามกระบวนการที่ได้นำเสนอขึ้นเท่านั้นซึ่งองค์กรสามารถพัฒนาเครื่องมือสนับสนุนให้มีความซับซ้อนตามการใช้งานได้ในภายหลังต่อไป

4) สำหรับการนำเครื่องมือสนับสนุนไปใช้งานจริง เครื่องแม่ข่ายให้บริการควรใช้ฮาร์ดแวร์ที่มีคุณสมบัติขั้นต่ำ คือ

(1) หน่วยประมวลผล อินเทลคอทดูโอ 1.73 กิกะเฮิร์ตซ์

(2) หน่วยความจำหลัก 1 กิกะไบต์

(3) ฮาร์ดดิสก์ความจุ 80 กิกะไบต์

(4) รองรับการโอนถ่ายข้อมูลเครือข่ายที่ความเร็ว 10/100 เมกกะบิตต่อวินาที

และเครื่องลูกข่ายรับบริการควรใช้ฮาร์ดแวร์ที่มีคุณสมบัติขั้นต่ำ คือ

(1) หน่วยประมวลผล อินเทลเซเลลลอน ความเร็ว 2.4 กิกะเฮิร์ตซ์ขึ้นไป

(2) หน่วยความจำหลัก 512 เมกกะไบต์

(3) ฮาร์ดดิสก์ความจุ 30 กิกะไบต์

(4) รองรับการโอนถ่ายข้อมูลเครือข่ายที่ความเร็ว 10/100 เมกกะบิตต่อวินาที

นอกจากการออกแบบและพัฒนากระบวนการสนับสนุนการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานแล้วนั้น ผู้วิจัยได้ร่วมทำบทความวิชาการกับอาจารย์ที่ปรึกษา ซึ่งมีบทความที่ได้รับการคัดเลือกเพื่อนำเสนอและตีพิมพ์ในงานประชุมวิชาการรวมทั้งสิ้น 2 บทความ โดยเป็นงานประชุมวิชาการระดับชาติ 1 บทความและงานประชุมวิชาการระดับนานาชาติ 1 บทความ

7.2 ปัญหาและข้อจำกัดในการทำวิจัย

ผู้วิจัยพบปัญหาและมีข้อจำกัดในการทำวิจัย คือ เนื่องจากการวิจัยนี้เป็นการวิเคราะห์และออกแบบกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานซึ่งผู้วิจัยได้ทำการวิเคราะห์ ออกแบบและพัฒนากระบวนการอ้างอิงจากกลุ่มกระบวนการที่เกี่ยวข้องของมาตรฐานไอทีพีเฟลอี 1517 กลุ่มกระบวนการพัฒนาผลิตภัณฑ์จากสินทรัพย์และองค์ความรู้อื่นๆ ดังที่ได้นำเสนอในงานวิจัยนี้ ซึ่งกระบวนการที่ได้นั้นยังไม่ได้ถูกนำไปใช้กับอุตสาหกรรมจริง ถึงแม้ว่าผู้วิจัยได้เติมเต็มโดยการศึกษาค้นคว้าหาความรู้เพิ่มเติมในส่วนบทเรียนรู้ที่เกิดจากการนำเอาซอฟต์แวร์สำเร็จรูป รวมถึงการพัฒนากระบวนการแล้วนำไปใช้กับอุตสาหกรรมจริงแล้วก็ตาม ดังนั้นงานวิจัยนี้จึงเป็นเพียงการนำเสนอแนวคิดในการพัฒนาซอฟต์แวร์สำเร็จรูปจากสินทรัพย์หนึ่งที่ได้จากการศึกษาจากองค์ความรู้ต่างๆ เท่านั้น ซึ่งจุดชี้วัดถึงความสมบูรณ์ของกระบวนการนี้จะเกิดขึ้นก็ต่อเมื่อถูกนำไปใช้และประเมินถึงข้อดี ข้อเสียและพัฒนากระบวนการต่อไป

7.3 ข้อเสนอแนะ

- 1) เครื่องมือที่พัฒนาในวิทยานิพนธ์นี้เป็นเพียงระบบต้นแบบ มีหน้าที่พื้นฐานเพื่อสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานเท่านั้น ซึ่งสามารถนำไปพัฒนาหน้าที่การทำงานอื่นเพิ่มเติมได้ต่อไป เช่น เพิ่มความสามารถของการดำเนินการให้เหมาะสมกับหน้าที่พื้นฐานรวมถึงความต้องการของแต่ละองค์กร เป็นต้น
- 2) เครื่องมือที่พัฒนาในวิทยานิพนธ์นี้ยังขาดระบบที่ใช้ในการติดตามผลการดำเนินการในแต่ละบทบาทหน้าที่ของกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานซึ่งทำหน้าที่ในการทวนสอบว่าผู้ดำเนินการในแต่ละบทบาทหน้าที่นั้น ได้ดำเนินการในกิจกรรมใดแล้วบ้างและยังขาดกิจกรรมใดที่ต้องดำเนินการ

3) เนื่องจากการออกแบบและพัฒนาระบบการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ที่ได้นี้ ประกอบด้วย 3 กลุ่มกระบวนการหลัก ดังที่ได้กล่าวมาแล้วข้างต้น ดังนั้นกระบวนการที่ได้นี้จึงอธิบายในเชิงกว้างซึ่งอยู่เพียงระดับส่วนประกอบเป็นฐาน (Component-Based) เท่านั้น ซึ่งแต่ละกระบวนการหลักที่ได้สามารถระบุรายละเอียดลงไปในระดับปฏิบัติการได้

4) กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานซึ่งเป็นผลลัพธ์จากงานวิจัยนี้ ยังไม่มีผลการทดลอง เนื่องจากยังไม่ได้นำไปใช้จริงกับอุตสาหกรรม ซึ่งองค์การที่นำไปใช้นั้นต้องเป็นองค์การที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์ในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ และนอกจากนั้นองค์การจะต้องมีผลิตภัณฑ์ซอฟต์แวร์สำเร็จรูปที่เป็นสินทรัพย์ที่มีอยู่ในแหล่งเก็บสายผลิตภัณฑ์ขององค์การอีกด้วย



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

- [1] Capretz, L.F., Ahmed, F., Al-maati, S., and Aghbari, Z. A. COTS-based software product line development. International Journal of Web Information Systems 4, 2 (2008) : 168-180.
- [2] IEEE-SA Standards Board. IEEE Recommended Practice for Software Acquisition. NY : The Institute of Electrical and Electronics Engineers, 1998.
- [3] Lewis, P., and others. Lessons learned in developing commercial off-the-shelf(COTS) intensive software systems. FAA Software Engineering Resource Center, 2000.
- [4] Coutts, C. T. and Gerdes, P. F. Integrating COTS Software: Lessons from a Large Healthcare Organization. IT Professional 12, 2 (March/April 2010) : 50-58.
- [5] Jingyu, L., and others. Development with Off-the-Shelf Component: 10 Facts. IEEE SOFTWARE 26, 2 (March/April 2009) : 80-87.
- [6] Clements, P. and Northrop, L. Software Product Lines: Practices and Patterns. Addison-Wesley Professional, 2001.
- [7] Gomaa, H. Designing Software Product Lines with UML: From Use Cases to Pattern-Based Software Architectures. Addison-Wesley Professional, 2004.
- [8] Northrop, L. M. SEI's Software Product Lines Tenets. IEEE SOFTWARE 19, 4 (July/August 2002) : 32-40.
- [9] Carney, D. Assembling Large Systems from COTS Components : Opportunities, Cautions, and Complexities. SEI, 1997.
- [10] Yang, Y., Boehm, B., Bhuta J., and Port, D. N. Value-Based Processes for COTS-Based Applications. IEEE SOFTWARE 22, 4 (July/August 2005) : 54-62.
- [11] Abst, C., Boehm, B., and Clark, E. B. COCOTS: A COTS Software Integration Lifecycle Cost Model-Model Overview and Preliminary Data Collection Findings, 11th ESCOM Conference, pp.325-333. Munich, 2000.

- [12] Boehm, B., Port, D. N., Yang, Y., and Bhuta, J. Composable Process Elements for Developing COTS-Based Applications, Empirical Software Engineering, 2003. ISESE 2003. Proceedings. 2003 International Symposium on, pp. 8-17. Rome, 2003.
- [13] Clapp, J. A. and Taub, A. E. A Management Guide to Software Maintenance in COTS-Based Systems. MITRE Center for Air Force C2 Systems, 1998.
- [14] IEEE-SA Standards Board. IEEE Standard for Information Technology-Software Life Cycle Processes-Reuse Processes. NY : The Institute of Electrical and Electronics Engineers, 1999(R2004).
- [15] Ahmed, F., and others. Setting Up COTS-Based Software Product Lines, Commercial- off-the-Shelf (COTS)-Based Software Systems, 2007. ICCBSS '07. Sixth International IEEE Conference on. pp.249-249. Canada, 2007.
- [16] Linden, F. V. D., and others. Software Product Lines in Action The Best Industrial Practice in Product Line Engineering. New York : Springer Berlin Heidelberg New York, 2007.
- [17] Warboys, B., and others. An Active-Architecture Approach to COTS Integration. IEEE SOFTWARE 22, 4 (July/August 2005) : 20-27.
- [18] CMMI Product Team. CMMI[®] for Development, Version 1.2. Pittsburgh : Software Engineering Institute, 2006.
- [19] IEEE-SA Standards Board. Systems and Software engineering-Software life cycle processes. NJ : The Institute of Electrical and Electronics Engineers, 2008.



ภาคผนวก

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก
อภิธานศัพท์

Commercial-off-the-shelf Software	ซอฟต์แวร์สำเร็จรูป
Software Product Lines	สายผลิตภัณฑ์ซอฟต์แวร์
Software Product Lines Repository	แหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
Software Product Lines Engineering	วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์
Systematic Practice	วิธีปฏิบัติอย่างเป็นระบบ
Domain Engineering	วิศวกรรมโดเมน
Application Engineering	วิศวกรรมแอปพลิเคชัน
Process Model	แบบจำลองกระบวนการ
Software Product	ผลิตภัณฑ์ซอฟต์แวร์



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ข
คำย่อและรหัสพจน์

COTS Software	Commercial-off-the-shelf Software
SPLE	Software Product Line Engineering
IEEE	Institute of Electrical and Electronics Engineers
IEEE 1517	IEEE Standard for Information Technology- Software Life Cycle Processes-Reuse Processes
MA	Measurement and Analysis
PA	Process Area
PLUS	Product Line UML-Based Software Engineering
ESPLEP	Evolutionary Software Product Line Engineering Process
CMMI	Capability Maturity Model Integration

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ค

ตัวอย่างเอกสารสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูป ที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

ในกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐานที่ผู้วิจัยได้นำเสนอนั้น มีประเภทของเอกสารสนับสนุนทั้งหมด 4 ประเภท ซึ่งแต่ละประเภทมีรายชื่อดังแสดงได้ตามตารางที่ ค.1-ค.4

ตารางที่ ค.1 รายชื่อเอกสารสนับสนุนประเภทนิยามกระบวนการ

นิยามกระบวนการ	หน้าที่
กำหนดแบบจำลองและกลยุทธ์การพัฒนา	130
กำหนดแผนการปรับแต่ง	136
กำหนดแผนการบูรณาการ	142
ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์	148
กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	154
กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	160
กำหนดแผนการบำรุงรักษา	166
สร้างกระบวนการบำรุงรักษา	172
เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้	178

ตารางที่ ค.2 รายชื่อเอกสารสนับสนุนประเภทแบบฟอร์ม

แบบฟอร์ม	หน้าที่
แบบฟอร์มการประเมิน	184
แบบฟอร์มการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา	185
แบบฟอร์มการกำหนดแผนการปรับแต่ง	186
แบบฟอร์มการกำหนดแผนการบูรณาการ	187
แบบฟอร์มการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์	188
แบบฟอร์มการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	189
แบบฟอร์มการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	190

ตารางที่ ค.2 รายชื่อเอกสารสนับสนุนประเภทแบบฟอร์ม (ต่อ)

แบบฟอร์ม	หน้าที่
แบบฟอร์มการพัฒนาส่วนประกอบ	191
แบบฟอร์มการบูรณาการกับระบบที่มีอยู่	192
แบบฟอร์มการบูรณาการระบบและซอฟต์แวร์	193
แบบฟอร์มการทดสอบสายผลิตภัณฑ์	194
แบบฟอร์มการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป	195
แบบฟอร์มการปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์	196
แบบฟอร์มการวางแผนการติดตั้ง	197
แบบฟอร์มการส่งมอบผลิตภัณฑ์ซอฟต์แวร์	198
แบบฟอร์มการจัดหาการสนับสนุนการฝึกอบรม	199
แบบฟอร์มการกำหนดแผนการบำรุงรักษา	200
แบบฟอร์มการสร้างกระบวนการบำรุงรักษา	201
แบบฟอร์มการวิเคราะห์ผลกระทบสินทรัพย์ที่สามารถนำมาใช้ซ้ำ	202
แบบฟอร์มการกำหนดวิธีการบำรุงรักษา	203
แบบฟอร์มการวางแผนการดัดแปลง ย้ายหรือยกเลิกข้อมูล	204
แบบฟอร์มการดัดแปลง ย้ายหรือยกเลิกข้อมูล	205
แบบฟอร์มการแจ้งผลการบำรุงรักษา	206
แบบฟอร์มการสร้างจุดมุ่งหมายของการวัดผล	207
แบบฟอร์มการวางแผนกระบวนการวัดผล	208
แบบฟอร์มการวัดผลการพัฒนาซอฟต์แวร์	209
แบบฟอร์มการวิเคราะห์และจัดเก็บผลลัพธ์	210
แบบฟอร์มการรายงานการวัดผลและแนวทางสำหรับปรับปรุงกระบวนการ	211

ตารางที่ ค.3 รายชื่อเอกสารสนับสนุนประเภทรายการตรวจสอบ

รายการตรวจสอบ	หน้าที่
รายการตรวจสอบเกณฑ์การใช้ซ้ำสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	212
รายการตรวจสอบเกณฑ์การใช้ซ้ำสำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป	217

ตารางที่ ค.4 รายชื่อเอกสารสนับสนุนประเภทแนวทางในการดำเนินการ

แนวทางในการดำเนินการ	หน้าที่
แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	222
แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป	225
แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป	228

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

กำหนดแบบจำลองและกลยุทธ์การพัฒนา (Define model and development strategies)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแบบจำลองและกลยุทธ์การพัฒนา (Define model and development strategies)		
ตัวแบบอ้างอิง : [COTS tailoring process, COTS integration process]	ระดับการใช้งาน : [โครงการ หรือ องค์การ]	เวอร์ชัน [n]
ชื่อโครงการ / แผนงาน : [.....]		

[ชื่อหน่วยงาน]

กำหนดแบบจำลองและกลยุทธ์การพัฒนา
(Define Model and Development Strategies)

[ชื่อโครงการ/แผนงาน]

เอกสารควบคุม เอกสารไม่ควบคุม

วันที่จัดทำเอกสาร : [วัน / เดือน / ปี]

สถานะเอกสาร : [ชื่อสถานะ]

จัดทำโดย : [ระบุส่วนงานหรือแผนกที่จัดทำ]

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของวิทยานิพนธ์ ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ปีการศึกษา 2553 ของนายยุทธนันท์ กันหาชน รหัสประจำตัวนิสิต 507 04128 21
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแบบจำลองและกลยุทธ์การพัฒนา (Define model and development strategies)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

บันทึกการแก้ไข					
เวอร์ชัน	แก้ไขครั้งที่	วันที่แก้ไข	รายละเอียด	แก้ไขโดย	ผู้อนุมัติ
[เลขที่]	[ครั้งที่]/[พ.ศ.]	[วัน / เดือน / ปี]	[หัวข้อ-รายละเอียดการแก้ไข]	[ชื่อ-สกุล]	[ชื่อ-สกุล]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแบบจำลองและกลยุทธ์การพัฒนา (Define model and development strategies)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สารบัญ		
ลำดับ	เนื้อหา	หน้า
1	บทนำ	[เลขหน้า]
2	เอกสารอ้างอิง	[เลขหน้า]
3	คำนิยาม	[เลขหน้า]
4	วัตถุประสงค์	[เลขหน้า]
5	วิเคราะห์ข้อกำหนดความต้องการซอฟต์แวร์ ซอฟต์แวร์สำเร็จรูปและ วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์	[เลขหน้า]
6	เลือกและจัดลำดับแบบจำลองและกลยุทธ์ในการพัฒนา	[เลขหน้า]
7	กำหนดแบบจำลองวัฏจักรชีวิตซอฟต์แวร์	[เลขหน้า]
8	กำหนดทรัพยากร กระบวนการและกลยุทธ์ในการพัฒนาซอฟต์แวร์	[เลขหน้า]
	ภาคผนวก ก - บทนิยาม	[เลขหน้า]
	ภาคผนวก ข - คำย่อและรหัสพจน์	[เลขหน้า]
	ภาคผนวก ค - เอกสารอ้างอิง	[เลขหน้า]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแบบจำลองและกลยุทธ์การพัฒนา (Define model and development strategies)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแบบจำลองและกลยุทธ์การพัฒนา

1. บทนำ (Introduction)

[ระบุรายละเอียดของบทนำ]

2. เอกสารอ้างอิง (Reference)

[ระบุรายชื่อเอกสารอ้างอิง]

3. บทนิยาม (Definition)

[ระบุบทนิยาม]

4. วัตถุประสงค์ (Objective)

[ระบุวัตถุประสงค์]

5. วิเคราะห์ข้อกำหนดความต้องการซอฟต์แวร์ ซอฟต์แวร์สำเร็จรูปและวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (software requirements specification, COTS software and Software product lines engineering analysis)

[วิเคราะห์ข้อกำหนดความต้องการซอฟต์แวร์ ซอฟต์แวร์สำเร็จรูปและวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ซึ่งเป็นข้อมูลนำเข้าของกระบวนการ เพื่อวิเคราะห์และกำหนดแบบจำลอง กำหนดวัฏจักรชีวิตซอฟต์แวร์ และกลยุทธ์ในการพัฒนาซอฟต์แวร์จากสินทรัพย์ที่สามารถตอบสนองต่อความต้องการซอฟต์แวร์ขององค์กร]

6. เลือกและจัดลำดับแบบจำลองและกลยุทธ์ในการพัฒนา (Select and prioritize model and development strategies)

[เลือกและจัดกลุ่มแบบจำลองและกลยุทธ์ในการพัฒนาซอฟต์แวร์ที่สามารถนำมาใช้กับกระบวนการพัฒนาซอฟต์แวร์ได้ โดยจัดลำดับตามความเหมาะสมกับโครงการ]

7. กำหนดแบบจำลองวัฏจักรชีวิตซอฟต์แวร์ (Define software life cycle model)

[กำหนดแบบจำลองที่ใช้เป็นวัฏจักรชีวิตซอฟต์แวร์ของโครงการที่ตอบสนองต่อข้อกำหนดความต้องการและโครงสร้างพื้นฐานของวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์]

8. กำหนดทรัพยากร กระบวนการและกลยุทธ์ในการพัฒนาซอฟต์แวร์ (Define resources, procedures and software development strategies)

[กำหนดทรัพยากร กระบวนการและกลยุทธ์ในการพัฒนาซอฟต์แวร์ ซึ่งรวมถึง มาตรฐานที่ใช้ วิธี เครื่องมือ และภาษาโปรแกรม]

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแบบจำลองและกลยุทธ์การพัฒนา (Define model and development strategies)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแบบจำลองและกลยุทธ์การพัฒนา

ภาคผนวก ก - บทนิยาม (Definitions)

ภาคผนวก ข - คำย่อและรหัศพน (Abbreviations and Acronyms)

ภาคผนวก ค - เอกสารอ้างอิง (Reference)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแบบจำลองและกลยุทธ์การพัฒนา (Define model and development strategies)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแบบจำลองและกลยุทธ์การพัฒนา

จัดทำโดย (.....ลงนาม.....) (....วัน / เดือน / ปี....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ตรวจสอบโดย (.....ลงนาม.....) (....วัน / เดือน / ปี....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

อนุมัติโดย (.....ลงนาม.....) (....วัน / เดือน / ปี....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการปรับแต่ง (Define tailoring plan)	รหัสเอกสาร : [DP-T-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการปรับแต่ง (Define tailoring plan)		
ตัวแบบอ้างอิง : [COTS tailoring process]	ระดับการใช้งาน : [โครงการ หรือ องค์การ]	เวอร์ชัน [n]
ชื่อโครงการ / แผนงาน : [.....]		

[ชื่อหน่วยงาน]

กำหนดแผนการปรับแต่ง
(Define Tailoring Plan)

[ชื่อโครงการ/แผนงาน]

เอกสารควบคุม เอกสารไม่ควบคุม

วันที่จัดทำเอกสาร : [วัน / เดือน / ปี]

สถานะเอกสาร : [ชื่อสถานะ]

จัดทำโดย : [ระบุส่วนงานหรือแผนกที่จัดทำ]

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของวิทยานิพนธ์ ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ปีการศึกษา 2553 ของนายยุทธนันท์ กันหาชน รหัสประจำตัวนิสิต 507 04128 21 ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการปรับแต่ง (Define tailoring plan)	รหัสเอกสาร : [DP-T-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

บันทึกการแก้ไข					
เวอร์ชัน	แก้ไขครั้งที่	วันที่แก้ไข	รายละเอียด	แก้ไขโดย	ผู้อนุมัติ
[เลขที่]	[ครั้งที่]/[พ.ศ.]	[วัน / เดือน / ปี]	[หัวข้อ-รายละเอียดการแก้ไข]	[ชื่อ-สกุล]	[ชื่อ-สกุล]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการปรับแต่ง (Define tailoring plan)	รหัสเอกสาร : [DP-T-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สารบัญ		
ลำดับ	เนื้อหา	หน้า
1	บทนำ	[เลขหน้า]
2	เอกสารอ้างอิง	[เลขหน้า]
3	คำนิยาม	[เลขหน้า]
4	วัตถุประสงค์	[เลขหน้า]
5	วิเคราะห์แบบจำลองและกลยุทธ์ในการพัฒนา	[เลขหน้า]
6	กำหนดแผนการปรับแต่งซอฟต์แวร์สำเร็จรูป	[เลขหน้า]
	ภาคผนวก ก - บทนิยาม	[เลขหน้า]
	ภาคผนวก ข - คำย่อและรหัสพจน์	[เลขหน้า]
	ภาคผนวก ค - เอกสารอ้างอิง	[เลขหน้า]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการปรับแต่ง (Define tailoring plan)	รหัสเอกสาร : [DP-T-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการปรับแต่ง

1. บทนำ (Introduction)

[ระบุรายละเอียดของบทนำ]

2. เอกสารอ้างอิง (Reference)

[ระบุรายชื่อเอกสารอ้างอิง]

3. บทนิยาม (Definition)

[ระบุบทนิยาม]

4. วัตถุประสงค์ (Objective)

[ระบุวัตถุประสงค์]

5. วิเคราะห์แบบจำลองและกลยุทธ์ในการพัฒนา (Model and development strategies analysis)

[วิเคราะห์แบบจำลองและกลยุทธ์ในการพัฒนาเพื่อทำการวางแผนสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป ซึ่งจะช่วยให้แผนของการปรับแต่งที่ได้นั้น สามารถตอบสนองต่อแบบจำลองและกลยุทธ์ในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ตามที่ได้กำหนดไว้ รวมถึงสามารถตอบสนองต่อข้อกำหนดความต้องการซอฟต์แวร์และวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์อีกด้วย]

6. กำหนดแผนการปรับแต่ง (Define COTS tailoring plan)

[กำหนดแผนการปรับแต่งที่ใช้สำหรับโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป ซึ่งระบุถึงกิจกรรมและหน้าที่สำหรับการปรับแต่งซอฟต์แวร์สำเร็จรูปทุกกระบวนการที่จะเกิดขึ้นในการพัฒนาซอฟต์แวร์ตามวัฏจักรชีวิตซอฟต์แวร์ รวมถึงแบบจำลองและกลยุทธ์ในการพัฒนาที่ได้ โดยแผนการปรับแต่งที่ได้นั้นใช้วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน]

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการปรับแต่ง (Define tailoring plan)	รหัสเอกสาร : [DP-T-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการปรับแต่ง

ภาคผนวก ก - บทนิยาม (Definitions)

ภาคผนวก ข - คำย่อและรหัศพนธ์ (Abbreviations and Acronyms)

ภาคผนวก ค - เอกสารอ้างอิง (Reference)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการปรับแต่ง (Define tailoring plan)	รหัสเอกสาร : [DP-T-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการปรับแต่ง

จัดทำโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ตรวจสอบโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

อนุมัติโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบูรณาการ (Define integration plan)	รหัสเอกสาร : [DP-I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการบูรณาการ (Define integration plan)		
ตัวแบบอ้างอิง : [COTS integration process]	ระดับการใช้งาน : [โครงการ หรือ องค์การ]	เวอร์ชัน [n]
ชื่อโครงการ / แผนงาน : [.....]		

[ชื่อหน่วยงาน]

กำหนดแผนการบูรณาการ
(Define Integration Plan)

[ชื่อโครงการ/แผนงาน]

เอกสารควบคุม เอกสารไม่ควบคุม

วันที่จัดทำเอกสาร : [วัน / เดือน / ปี]

สถานะเอกสาร : [ชื่อสถานะ]

จัดทำโดย : [ระบุส่วนงานหรือแผนกที่จัดทำ]

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของวิทยานิพนธ์ ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ปีการศึกษา 2553 ของนายยุทธนันท์ กันหาชน รหัสประจำตัวนิสิต 507 04128 21
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบูรณาการ (Define integration plan)	รหัสเอกสาร : [DP-I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

บันทึกการแก้ไข					
เวอร์ชัน	แก้ไขครั้งที่	วันที่แก้ไข	รายละเอียด	แก้ไขโดย	ผู้อนุมัติ
[เลขที่]	[ครั้งที่]/[พ.ศ.]	[วัน / เดือน / ปี]	[หัวข้อ-รายละเอียดการแก้ไข]	[ชื่อ-สกุล]	[ชื่อ-สกุล]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบูรณาการ (Define integration plan)	รหัสเอกสาร : [DP-I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สารบัญ		
ลำดับ	เนื้อหา	หน้า
1	บทนำ	[เลขหน้า]
2	เอกสารอ้างอิง	[เลขหน้า]
3	คำนิยาม	[เลขหน้า]
4	วัตถุประสงค์	[เลขหน้า]
5	วิเคราะห์แบบจำลองและกลยุทธ์ในการพัฒนา	[เลขหน้า]
6	กำหนดแผนการบูรณาการซอฟต์แวร์สำเร็จรูป	[เลขหน้า]
	ภาคผนวก ก - บทนิยาม	[เลขหน้า]
	ภาคผนวก ข - คำย่อและรหัสพจน์	[เลขหน้า]
	ภาคผนวก ค - เอกสารอ้างอิง	[เลขหน้า]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบูรณาการ (Define integration plan)	รหัสเอกสาร : [DP-I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการบูรณาการ

1. บทนำ (Introduction)

[ระบุรายละเอียดของบทนำ]

2. เอกสารอ้างอิง (Reference)

[ระบุรายชื่อเอกสารอ้างอิง]

3. บทนิยาม (Definition)

[ระบุบทนิยาม]

4. วัตถุประสงค์ (Objective)

[ระบุวัตถุประสงค์]

5. วิเคราะห์แบบจำลองและกลยุทธ์ในการพัฒนา (Model and development strategies analysis)

[วิเคราะห์แบบจำลองและกลยุทธ์ในการพัฒนาเพื่อทำการวางแผนสำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป ซึ่งจะช่วยให้แผนของการบูรณาการที่ได้นั้น สามารถตอบสนองต่อแบบจำลองและกลยุทธ์ในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ตามที่ได้กำหนดไว้ รวมถึงสามารถตอบสนองต่อข้อกำหนดความต้องการซอฟต์แวร์และวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์อีกด้วย]

6. กำหนดแผนการบูรณาการซอฟต์แวร์สำเร็จรูป (Define COTS integration plan)

[กำหนดแผนการบูรณาการที่ใช้สำหรับโครงการการบูรณาการซอฟต์แวร์สำเร็จรูป ซึ่งระบุถึงกิจกรรมและหน้าที่สำหรับการบูรณาการซอฟต์แวร์สำเร็จรูปทุกกระบวนการที่จะเกิดขึ้นในการพัฒนาซอฟต์แวร์ตามวัฏจักรชีวิตซอฟต์แวร์ รวมถึงแบบจำลองและกลยุทธ์ในการพัฒนาที่ได้ โดยแผนการบูรณาการนั้นใช้วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน]

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบูรณาการ (Define integration plan)	รหัสเอกสาร : [DP-I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการบูรณาการ

ภาคผนวก ก - บทนิยาม (Definitions)

ภาคผนวก ข - คำย่อและรหัสพจน์ (Abbreviations and Acronyms)

ภาคผนวก ค - เอกสารอ้างอิง (Reference)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบูรณาการ (Define integration plan)	รหัสเอกสาร : [DP-I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการบูรณาการ

จัดทำโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ตรวจสอบโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

อนุมัติโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Software product lines repository review)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Software product lines repository review)		
ตัวแบบอ้างอิง : [COTS tailoring process, COTS integration process, COTS maintenance process]	ระดับการใช้งาน : [โครงการ หรือ องค์การ]	เวอร์ชัน [n]
ชื่อโครงการ / แผนงาน : [.....]		

[ชื่อหน่วยงาน]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
(Software Product Lines Repository Review)

[ชื่อโครงการ/แผนงาน]

เอกสารควบคุม เอกสารไม่ควบคุม

วันที่จัดทำเอกสาร : [วัน / เดือน / ปี]

สถานะเอกสาร : [ชื่อสถานะ]

จัดทำโดย : [ระบุส่วนงานหรือแผนกที่จัดทำ]

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของวิทยานิพนธ์ ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ปีการศึกษา 2553 ของนายยุทธนันท์ กันหาชน รหัสประจำตัวนิสิต 507 04128 21 ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Software product lines repository review)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

บันทึกการแก้ไข					
เวอร์ชัน	แก้ไขครั้งที่	วันที่แก้ไข	รายละเอียด	แก้ไขโดย	ผู้อนุมัติ
[เลขที่]	[ครั้งที่]/[พ.ศ.]	[วัน / เดือน / ปี]	[หัวข้อ-รายละเอียดการแก้ไข]	[ชื่อ-สกุล]	[ชื่อ-สกุล]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Software product lines repository review)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สารบัญ		
ลำดับ	เนื้อหา	หน้า
1	บทนำ	[เลขหน้า]
2	เอกสารอ้างอิง	[เลขหน้า]
3	คำนิยาม	[เลขหน้า]
4	วัตถุประสงค์	[เลขหน้า]
5	กำหนดขอบเขตของสายผลิตภัณฑ์สำหรับการใช้ซ้ำ	[เลขหน้า]
6	กำหนดสินทรัพย์ที่สามารถนำมาใช้ซ้ำ	[เลขหน้า]
7	กำหนดคุณลักษณะของสินทรัพย์ที่สามารถนำมาใช้ซ้ำ	[เลขหน้า]
	ภาคผนวก ก - บทนิยาม	[เลขหน้า]
	ภาคผนวก ข - คำย่อและรหัสพจน์	[เลขหน้า]
	ภาคผนวก ค - เอกสารอ้างอิง	[เลขหน้า]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Software product lines repository review)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์

1. บทนำ (Introduction)
[ระบุนายละเอียดของบทนำ]
2. เอกสารอ้างอิง (Reference)
[ระบุนายสิ่งเอกสารอ้างอิง]
3. บทนิยาม (Definition)
[ระบุบทนิยาม]
4. วัตถุประสงค์ (Objective)
[ระบุวัตถุประสงค์]
5. กำหนดขอบเขตของสายผลิตภัณฑ์สำหรับการใช้ซ้ำ (Define product lines scope for software reuse)
[กำหนดขอบเขตของสายผลิตภัณฑ์ที่ใช้สำหรับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ของโครงการ]
6. กำหนดสินทรัพย์ที่สามารถนำมาใช้ซ้ำ (Define reusable assets)
[กำหนดสินทรัพย์ที่สามารถนำมาใช้ซ้ำ]
7. กำหนดคุณลักษณะของสินทรัพย์ที่สามารถนำมาใช้ซ้ำ (Define reusable assets features)
[กำหนดคุณลักษณะของสินทรัพย์ที่สามารถนำมาใช้ซ้ำได้ตามลำดับความสำคัญซึ่งแบ่งเป็นคุณลักษณะหลัก (Core feature) คุณลักษณะตัวเลือก (Optional feature) และคุณลักษณะทางเลือก (Alternative feature)]

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Software product lines repository review)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์

ภาคผนวก ก - บทนิยาม (Definitions)

ภาคผนวก ข - คำย่อและรหัสพจน์ (Abbreviations and Acronyms)

ภาคผนวก ค - เอกสารอ้างอิง (Reference)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ (Software product lines repository review)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์

จัดทำโดย (.....ลงนาม.....) (....วัน / เดือน / ปี....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ตรวจสอบโดย (.....ลงนาม.....) (....วัน / เดือน / ปี....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

อนุมัติโดย (.....ลงนาม.....) (....วัน / เดือน / ปี....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture)		
ตัวแบบอ้างอิง : [COTS tailoring process, COTS integration process]	ระดับการใช้งาน : [โครงการ หรือ องค์การ]	เวอร์ชัน [n]
ชื่อโครงการ / แผนงาน : [.....]		

[ชื่อหน่วยงาน]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
(Define Software Product Lines Architecture)

[ชื่อโครงการ/แผนงาน]

เอกสารควบคุม เอกสารไม่ควบคุม

วันที่จัดทำเอกสาร : [วัน / เดือน / ปี]

สถานะเอกสาร : [ชื่อสถานะ]

จัดทำโดย : [ระบุส่วนงานหรือแผนกที่จัดทำ]

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของวิทยานิพนธ์ ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ปีการศึกษา 2553 ของนายยุทธนันท์ กันหาชน รหัสประจำตัวนิสิต 507 04128 21
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

บันทึกการแก้ไข					
เวอร์ชัน	แก้ไขครั้งที่	วันที่แก้ไข	รายละเอียด	แก้ไขโดย	ผู้อนุมัติ
[เลขที่]	[ครั้งที่]/[พ.ศ.]	[วัน / เดือน / ปี]	[หัวข้อ-รายละเอียดการแก้ไข]	[ชื่อ-สกุล]	[ชื่อ-สกุล]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สารบัญ		
ลำดับ	เนื้อหา	หน้า
1	บทนำ	[เลขหน้า]
2	เอกสารอ้างอิง	[เลขหน้า]
3	คำนิยาม	[เลขหน้า]
4	วัตถุประสงค์	[เลขหน้า]
5	วิเคราะห์สินทรัพย์ที่สามารถนำมาใช้ซ้ำ	[เลขหน้า]
6	วิเคราะห์ผลการประเมินข้อกำหนดความต้องการซอฟต์แวร์	[เลขหน้า]
7	กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์สำหรับการปรับแต่ง/ การบูรณาการซอฟต์แวร์สำเร็จรูป	[เลขหน้า]
	- สถาปัตยกรรมเชิงโครงสร้าง	[เลขหน้า]
	- สถาปัตยกรรมเชิงพลวัต	[เลขหน้า]
	ภาคผนวก ก - บทนิยาม	[เลขหน้า]
	ภาคผนวก ข - คำย่อและรหัสพจน์	[เลขหน้า]
	ภาคผนวก ค - เอกสารอ้างอิง	[เลขหน้า]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

1. บทนำ (Introduction)

[ระบุนรายละเอียดของบทนำ]

2. เอกสารอ้างอิง (Reference)

[ระบุนายสิ่งเอกสารอ้างอิง]

3. บทนิยาม (Definition)

[ระบุนบทนิยาม]

4. วัตถุประสงค์ (Objective)

[ระบุวัตถุประสงค์]

5. วิเคราะห์สินทรัพย์ที่สามารถนำมาใช้ (Reusable assets analysis)

[วิเคราะห์และจำแนกสินทรัพย์ที่สามารถนำมาใช้ซ้ำตามลำดับความสำคัญแบ่งเป็น คุณสมบัติหลัก (Core feature) คุณสมบัติตัวเลือก (Optional feature) และคุณสมบัติทางเลือก (Alternative feature)]

6. วิเคราะห์ผลการประเมินข้อกำหนดความต้องการซอฟต์แวร์ (Software Requirements specification evaluation result analysis)

[พิจารณาผลการประเมินข้อกำหนดความต้องการซอฟต์แวร์ ซึ่งเกิดจากการประเมินข้อกำหนดความต้องการเทียบกับ แผนการปรับแต่ง/บูรณาการซอฟต์แวร์สำเร็จรูป สินทรัพย์ที่สามารถนำมาใช้ซ้ำและเกณฑ์การใช้ซ้ำ ซึ่งนำมาใช้พิจารณากำหนดสถาปัตยกรรมซอฟต์แวร์และรายละเอียดสถาปัตยกรรมซอฟต์แวร์ เพื่อตอบสนองต่อข้อกำหนดความต้องการซอฟต์แวร์]

7. กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์สำหรับการปรับแต่ง/การบูรณาการซอฟต์แวร์สำเร็จรูป (define software product lines architecture for COTS tailoring / COTS integration)

[กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์สำหรับการปรับแต่ง/การบูรณาการซอฟต์แวร์สำเร็จรูป โดยประกอบด้วยสถาปัตยกรรมเชิงโครงสร้าง (Structural architecture) แสดงถึงโครงสร้างซอฟต์แวร์ทั้งหมด และสถาปัตยกรรมเชิงพลวัต (Dynamical architecture) แสดงถึงการติดต่อสื่อสารกันของแต่ละส่วนประกอบ โดยกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปจะรวมถึงสถาปัตยกรรมสำหรับการบูรณาการกับระบบที่มีอยู่ และกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปจะรวมถึงสถาปัตยกรรมของทอพอโลยีการเชื่อมต่อซึ่งใช้เป็นส่วนเชื่อมต่อเพื่อบูรณาการซอฟต์แวร์สำเร็จรูป สินทรัพย์ที่สามารถนำมาใช้ซ้ำและรหัสคำสั่งเพิ่มเติมเข้าด้วยกัน]

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

ภาคผนวก ก - บทนิยาม (Definitions)

ภาคผนวก ข - คำย่อและรหัสพจน์ (Abbreviations and Acronyms)

ภาคผนวก ค - เอกสารอ้างอิง (Reference)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

จัดทำโดย (.....ลงนาม.....) (....วัน / เดือน / ปี....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ตรวจสอบโดย (.....ลงนาม.....) (....วัน / เดือน / ปี....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

อนุมัติโดย (.....ลงนาม.....) (....วัน / เดือน / ปี....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture details)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture details)		
ตัวแบบอ้างอิง : [COTS tailoring process, COTS integration process]	ระดับการใช้งาน : [โครงการ หรือ องค์การ]	เวอร์ชัน [n]
ชื่อโครงการ / แผนงาน : [.....]		

[ชื่อหน่วยงาน]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
(Define Software Product Lines Architecture Details)

[ชื่อโครงการ/แผนงาน]

เอกสารควบคุม เอกสารไม่ควบคุม

วันที่จัดทำเอกสาร : [วัน / เดือน / ปี]

สถานะเอกสาร : [ชื่อสถานะ]

จัดทำโดย : [ระบุส่วนงานหรือแผนกที่จัดทำ]

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของวิทยานิพนธ์ ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ปีการศึกษา 2553 ของนายยุทธนันท์ กันหาชน รหัสประจำตัวนิสิต 507 04128 21
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture details)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

บันทึกการแก้ไข					
เวอร์ชัน	แก้ไขครั้งที่	วันที่แก้ไข	รายละเอียด	แก้ไขโดย	ผู้อนุมัติ
[เลขที่]	[ครั้งที่]/[พ.ศ.]	[วัน / เดือน / ปี]	[หัวข้อ-รายละเอียดการแก้ไข]	[ชื่อ-สกุล]	[ชื่อ-สกุล]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture details)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สารบัญ		
ลำดับ	เนื้อหา	หน้า
1	บทนำ	[เลขหน้า]
2	เอกสารอ้างอิง	[เลขหน้า]
3	คำนิยาม	[เลขหน้า]
4	วัตถุประสงค์	[เลขหน้า]
5	วิเคราะห์โครงสร้างของสถาปัตยกรรมซอฟต์แวร์	[เลขหน้า]
6	กำหนดรายละเอียดสถาปัตยกรรมซอฟต์แวร์	[เลขหน้า]
	- รายละเอียดสถาปัตยกรรมเชิงโครงสร้าง	[เลขหน้า]
	- รายละเอียดสถาปัตยกรรมเชิงพลวัต	[เลขหน้า]
	ภาคผนวก ก - บทนิยาม	[เลขหน้า]
	ภาคผนวก ข - คำย่อและรหัสพจน์	[เลขหน้า]
	ภาคผนวก ค - เอกสารอ้างอิง	[เลขหน้า]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture details)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

1. บทนำ (Introduction)

[ระบุรายละเอียดของบทนำ]

2. เอกสารอ้างอิง (Reference)

[ระบุรายชื่อเอกสารอ้างอิง]

3. บทนิยาม (Definition)

[ระบุบทนิยาม]

4. วัตถุประสงค์ (Objective)

[ระบุวัตถุประสงค์]

5. วิเคราะห์โครงสร้างของสถาปัตยกรรมซอฟต์แวร์ (Software architecture Infrastructure analysis)

[วิเคราะห์โครงสร้างของสถาปัตยกรรมซอฟต์แวร์ซึ่งได้ถูกออกแบบไว้แล้ว เพื่อกำหนดรายละเอียดสถาปัตยกรรมที่ใช้สำหรับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์]

6. กำหนดรายละเอียดสถาปัตยกรรมซอฟต์แวร์ (Define software architecture details)

[กำหนดรายละเอียดสถาปัตยกรรมซอฟต์แวร์ ซึ่งประกอบด้วย 2 ส่วน คือ สถาปัตยกรรมเชิงโครงสร้าง (Structural architecture) ซึ่งแสดงถึงโครงสร้างของซอฟต์แวร์ทั้งหมด โดยมีตัวอย่างแบบรูป เช่น Client/Server Architectural Pattern เป็นต้น และสถาปัตยกรรมเชิงพลวัต (Dynamical architecture) ซึ่งแสดงถึงการติดต่อสื่อสารระหว่างแต่ละส่วนประกอบโดยมีตัวอย่างแบบรูป เช่น Broker Communication Pattern เป็นต้น โดยรายละเอียดของสถาปัตยกรรม ประกอบด้วยรายละเอียด เช่น Client Component, User Interface Component, Server Component, Control Component, Coordinator Component, Data Collection Component, Data Analysis Component, I/O Component และ Interconnection Topology เป็นต้น]

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture details)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

ภาคผนวก ก - บทนิยาม (Definitions)

ภาคผนวก ข - คำย่อและรหัสนิยม (Abbreviations and Acronyms)

ภาคผนวก ค - เอกสารอ้างอิง (Reference)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Define software product lines architecture details)	รหัสเอกสาร : [DP-T/I-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

จัดทำโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ตรวจสอบโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

อนุมัติโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบำรุงรักษา (Define maintenance plan)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการบำรุงรักษา (Define maintenance plan)		
ตัวแบบอ้างอิง : [COTS maintenance process]	ระดับการใช้งาน : [โครงการ หรือ องค์กร]	เวอร์ชัน [n]
ชื่อโครงการ / แผนงาน : [.....]		

[ชื่อหน่วยงาน]

กำหนดแผนการบำรุงรักษา
(Define Maintenance Plan)

[ชื่อโครงการ/แผนงาน]

เอกสารควบคุม เอกสารไม่ควบคุม

วันที่จัดทำเอกสาร : [วัน / เดือน / ปี]

สถานะเอกสาร : [ชื่อสถานะ]

จัดทำโดย : [ระบุส่วนงานหรือแผนกที่จัดทำ]

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของวิทยานิพนธ์ ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ปีการศึกษา 2553 ของนายยุทธนันท์ กันหาชน รหัสประจำตัวนิสิต 507 04128 21
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบำรุงรักษา (Define maintenance plan)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

บันทึกการแก้ไข					
เวอร์ชัน	แก้ไขครั้งที่	วันที่แก้ไข	รายละเอียด	แก้ไขโดย	ผู้อนุมัติ
[เลขที่]	[ครั้งที่]/[พ.ศ.]	[วัน / เดือน / ปี]	[หัวข้อ-รายละเอียดการแก้ไข]	[ชื่อ-สกุล]	[ชื่อ-สกุล]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบำรุงรักษา (Define maintenance plan)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สารบัญ		
ลำดับ	เนื้อหา	หน้า
1	บทนำ	[เลขหน้า]
2	เอกสารอ้างอิง	[เลขหน้า]
3	คำนิยาม	[เลขหน้า]
4	วัตถุประสงค์	[เลขหน้า]
5	วิเคราะห์รายงานปัญหา/คำร้องขอการดัดแปลง	[เลขหน้า]
6	จัดกลุ่มและเรียงลำดับความสำคัญของปัญหา/คำร้องขอการดัดแปลง	[เลขหน้า]
7	ประเมินความสามารถในการแก้ไขปัญหาที่ตอบสนองต่อรายงานปัญหา/คำร้องขอการดัดแปลง	[เลขหน้า]
8	กำหนดกิจกรรมและหน้าที่สำหรับกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป	[เลขหน้า]
	ภาคผนวก ก - บทนิยาม	[เลขหน้า]
	ภาคผนวก ข - คำย่อและรหัสพจน์	[เลขหน้า]
	ภาคผนวก ค - เอกสารอ้างอิง	[เลขหน้า]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบำรุงรักษา (Define maintenance plan)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการบำรุงรักษา

1. บทนำ (Introduction)

[ระบุนรายละเอียดของบทนำ]

2. เอกสารอ้างอิง (Reference)

[ระบุนายสิ่งเอกสารอ้างอิง]

3. บทนิยาม (Definition)

[ระบุนบทนิยาม]

4. วัตถุประสงค์ (Objective)

[ระบุวัตถุประสงค์]

5. วิเคราะห์รายงานปัญหา/คำร้องขอการดัดแปลง (Problems report/modification request analysis)

[วิเคราะห์รายงานปัญหา/คำร้องขอการดัดแปลง เพื่อพิจารณาถึงความเป็นไปได้ในการตอบสนองต่อรายงานปัญหา/คำร้องขอการดัดแปลง รวมถึงความสมบูรณ์ของรายงานปัญหา/คำร้องขอการดัดแปลงว่ามีความเสถียรแล้วหรือไม่]

6. จัดกลุ่มและเรียงลำดับความสำคัญของรายงานปัญหา/คำร้องขอการดัดแปลง (Categorize and prioritize the problems report/modification request)

[จัดกลุ่มและเรียงลำดับความสำคัญของปัญหา/คำร้องขอการดัดแปลง และความยากง่ายของการดำเนินการบำรุงรักษา รวมถึงเวลาที่ใช้ในการบำรุงรักษาตามลำดับ]

7. ประเมินความสามารถในการแก้ไขปัญหาที่ตอบสนองต่อรายงานปัญหา/คำร้องขอการดัดแปลง (Evaluate capability for resolve problems report/modification request)

[ประเมินความสามารถในการแก้ไขปัญหาที่ตอบสนองต่อปัญหา/คำร้องขอการดัดแปลง ว่าสามารถตอบสนองต่อรายงานปัญหา/คำร้องขอการดัดแปลง อยู่ในระดับใด รายการใดที่สามารถดำเนินการบำรุงรักษาได้ และรายการใดที่ไม่สามารถดำเนินการบำรุงรักษาได้พร้อมระบุเหตุผล]

8. กำหนดกิจกรรมและหน้าที่สำหรับกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป (Define activity and task for COTS maintenance process)

[กำหนดกิจกรรมและหน้าที่สำหรับกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป ในรายการที่สามารถดำเนินการบำรุงรักษาตามรายงานปัญหา/คำร้องขอการดัดแปลงได้]

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบำรุงรักษา (Define maintenance plan)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการบำรุงรักษา

ภาคผนวก ก - บทนิยาม (Definitions)

ภาคผนวก ข - คำย่อและรศพจน์ (Abbreviations and Acronyms)

ภาคผนวก ค - เอกสารอ้างอิง (Reference)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

กำหนดแผนการบำรุงรักษา (Define maintenance plan)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

กำหนดแผนการบำรุงรักษา

จัดทำโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ตรวจสอบโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

อนุมัติโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

สร้างกระบวนการบำรุงรักษา (Establish maintenance procedures)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สร้างกระบวนการบำรุงรักษา (Establish maintenance procedures)		
ตัวแบบอ้างอิง : [COTS maintenance process]	ระดับการใช้งาน : [โครงการ หรือ องค์กร]	เวอร์ชัน [n]
ชื่อโครงการ / แผนงาน : [.....]		

[ชื่อหน่วยงาน]

สร้างกระบวนการบำรุงรักษา
(Establish Maintenance Procedures)

[ชื่อโครงการ/แผนงาน]

เอกสารควบคุม เอกสารไม่ควบคุม

วันที่จัดทำเอกสาร : [วัน / เดือน / ปี]

สถานะเอกสาร : [ชื่อสถานะ]

จัดทำโดย : [ระบุส่วนงานหรือแผนกที่จัดทำ]

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของวิทยานิพนธ์ ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ปีการศึกษา 2553 ของนายยุทธนันท์ กันหาชน รหัสประจำตัวนิสิต 507 04128 21
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

สร้างกระบวนการบำรุงรักษา (Establish maintenance procedures)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

บันทึกการแก้ไข					
เวอร์ชัน	แก้ไขครั้งที่	วันที่แก้ไข	รายละเอียด	แก้ไขโดย	ผู้อนุมัติ
[เลขที่]	[ครั้งที่]/[พ.ศ.]	[วัน / เดือน / ปี]	[หัวข้อ-รายละเอียดการแก้ไข]	[ชื่อ-สกุล]	[ชื่อ-สกุล]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

สร้างกระบวนการบำรุงรักษา (Establish maintenance procedures)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สารบัญ		
ลำดับ	เนื้อหา	หน้า
1	บทนำ	[เลขหน้า]
2	เอกสารอ้างอิง	[เลขหน้า]
3	คำนิยาม	[เลขหน้า]
4	วัตถุประสงค์	[เลขหน้า]
5	วิเคราะห์แผนการบำรุงรักษา	[เลขหน้า]
6	กำหนดกระบวนการสนับสนุนการบำรุงรักษาซอฟต์แวร์	[เลขหน้า]
	- การรับ (Receive)	[เลขหน้า]
	- การบันทึก (Record)	[เลขหน้า]
	- การแก้ปัญหา (Resolve)	[เลขหน้า]
	- การติดตามปัญหา (Tracking problems)	[เลขหน้า]
	ภาคผนวก ก - บทนิยาม	[เลขหน้า]
	ภาคผนวก ข - คำย่อและรหัสพจน์	[เลขหน้า]
	ภาคผนวก ค - เอกสารอ้างอิง	[เลขหน้า]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

สร้างกระบวนการบำรุงรักษา (Establish maintenance procedures)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สร้างกระบวนการบำรุงรักษา

1. บทนำ (Introduction)
[ระบุนายละเอียดของบทนำ]
2. เอกสารอ้างอิง (Reference)
[ระบุนายสิ่งเอกสารอ้างอิง]
3. บทนิยาม (Definition)
[ระบุนิยาม]
4. วัตถุประสงค์ (Objective)
[ระบุวัตถุประสงค์]
5. วิเคราะห์แผนการบำรุงรักษา (Maintenance plan analysis)
[วิเคราะห์แผนการบำรุงรักษาเพื่อสร้างกระบวนการสำหรับการบำรุงรักษาโดยกระบวนการที่ได้นั้นจะต้องตอบสนองต่อแผนการบำรุงรักษา]
6. กำหนดกระบวนการสนับสนุนการบำรุงรักษาซอฟต์แวร์ (Define support procedures for software maintenance)
[กำหนดกระบวนการสนับสนุนการบำรุงรักษาซอฟต์แวร์ ซึ่งประกอบด้วย การรับ (Receive), การบันทึก (Record), การแก้ปัญหา (Resolve) และการติดตามปัญหา (Tracking problems)]

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

สร้างกระบวนการบำรุงรักษา (Establish maintenance procedures)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สร้างกระบวนการบำรุงรักษา

ภาคผนวก ก - บทนิยาม (Definitions)

ภาคผนวก ข - คำย่อและรหัศพนธ์ (Abbreviations and Acronyms)

ภาคผนวก ค - เอกสารอ้างอิง (Reference)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

สร้างกระบวนการบำรุงรักษา (Establish maintenance procedures)	รหัสเอกสาร : [DP-M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สร้างกระบวนการบำรุงรักษา

จัดทำโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ตรวจสอบโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

อนุมัติโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้ (User documentation)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้ (User documentation)		
ตัวแบบอ้างอิง : [COTS tailoring process, COTS integration process, COTS maintenance process]	ระดับการใช้งาน : [โครงการ หรือ องค์การ]	เวอร์ชัน [n]
ชื่อโครงการ / แผนงาน : [.....]		

[ชื่อหน่วยงาน]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้
(User Documentation)

[ชื่อโครงการ/แผนงาน]

เอกสารควบคุม เอกสารไม่ควบคุม

วันที่จัดทำเอกสาร : [วัน / เดือน / ปี]

สถานะเอกสาร : [ชื่อสถานะ]

จัดทำโดย : [ระบุส่วนงานหรือแผนกที่จัดทำ]

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของวิทยานิพนธ์ ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิศวกรรมซอฟต์แวร์ ปีการศึกษา 2553 ของนายยุทธนันท์ กันหาชน รหัสประจำตัวนิสิต 507 04128 21 ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้ (User documentation)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

บันทึกการแก้ไข					
เวอร์ชัน	แก้ไขครั้งที่	วันที่แก้ไข	รายละเอียด	แก้ไขโดย	ผู้อนุมัติ
[เลขที่]	[ครั้งที่]/[พ.ศ.]	[วัน / เดือน / ปี]	[หัวข้อ-รายละเอียดการแก้ไข]	[ชื่อ-สกุล]	[ชื่อ-สกุล]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้ (User documentation)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

สารบัญ		
ลำดับ	เนื้อหา	หน้า
1	บทนำ	[เลขหน้า]
2	เอกสารอ้างอิง	[เลขหน้า]
3	คำนิยาม	[เลขหน้า]
4	วัตถุประสงค์	[เลขหน้า]
5	รายการที่ต้องเตรียม	[เลขหน้า]
6	ขั้นตอนการติดตั้งซอฟต์แวร์	[เลขหน้า]
7	รายละเอียดซอฟต์แวร์	[เลขหน้า]
8	การใช้งานสำหรับผู้ดูแลระบบ	[เลขหน้า]
9	การใช้งานสำหรับผู้ใช้งานระบบทั่วไป	[เลขหน้า]
	ภาคผนวก ก - บทนิยาม	[เลขหน้า]
	ภาคผนวก ข - คำย่อและรหัสพจน์	[เลขหน้า]
	ภาคผนวก ค - เอกสารอ้างอิง	[เลขหน้า]

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้ (User documentation)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้

1. **บทนำ (Introduction)**
[ระบุนายละเอียดของบทนำ]
2. **เอกสารอ้างอิง (Reference)**
[ระบุนายสิ่งเอกสารอ้างอิง]
3. **บทนิยาม (Definition)**
[ระบุนบทนิยาม]
4. **วัตถุประสงค์ (Objective)**
[ระบุวัตถุประสงค์]
5. **รายการที่ต้องเตรียม (Provide list)**
[รายการที่ต้องเตรียมซึ่งเป็นสภาพแวดล้อมของซอฟต์แวร์ที่ต้องใช้ในการติดตั้งซอฟต์แวร์]
6. **ขั้นตอนการติดตั้งซอฟต์แวร์ (Software installation step by step)**
[ขั้นตอนการติดตั้งซอฟต์แวร์ ซึ่งรวมถึงการติดตั้งสภาพแวดล้อมอื่นที่จำเป็นต้องใช้ด้วย]
7. **รายละเอียดซอฟต์แวร์ (Software details)**
[รายละเอียดซอฟต์แวร์ที่ติดตั้งโดยสรุป ซึ่งรวมถึงคำนิยามหรือข้อกำหนดต่างๆ]
8. **การใช้งานสำหรับผู้ดูแลระบบ (Administrator guide)**
[การใช้งานสำหรับผู้ดูแลระบบ]
9. **การใช้งานสำหรับผู้ใช้งานระบบทั่วไป (User guide)**
[การใช้งานสำหรับผู้ใช้งานระบบทั่วไป]

ชื่อเพิ่มข้อมูล: [ชื่อเพิ่ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้ (User documentation)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้

ภาคผนวก ก - บทนิยาม (Definitions)

ภาคผนวก ข - คำย่อและรศพจน์ (Abbreviations and Acronyms)

ภาคผนวก ค - เอกสารอ้างอิง (Reference)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้ (User documentation)	รหัสเอกสาร : [DP-T/I/M-n]
[ชื่อโครงการ/แผนงาน]	เวอร์ชัน [n]

เอกสารการใช้งานซอฟต์แวร์สำหรับผู้ใช้

จัดทำโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ตรวจสอบโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

อนุมัติโดย (.....ลงนาม.....) (.....วัน / เดือน / ปี.....)
 [ชื่อ-สกุล] วันที่ลงนาม
 [ตำแหน่ง]

ศูนย์วิทยทรัพยากร
 จุฬาลงกรณ์มหาวิทยาลัย

ชื่อแฟ้มข้อมูล: [ชื่อแฟ้ม.นามสกุล]	[ชื่อโครงการ/แผนงาน]	หน้าที่ [หน้าที่]/[จำนวนหน้า]
	[ประเภทเอกสาร]	วันที่พิมพ์ [วัน / เดือน / ปี]

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการประเมิน	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการประเมิน

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ถูประเมิน :

ชื่อเกณฑ์การให้ซ้ำที่ใช้ประเมิน (รายการตรวจสอบ) :

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ประเมิน :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการประเมิน

ลำดับ	รายการที่ประเมิน	เกณฑ์การให้ซ้ำ	ผลลัพธ์การประเมิน	คำอธิบาย
[ลำดับ]	[ระบุรายการที่ประเมิน]	[ระบุประเภทและลำดับเลขที่ของเกณฑ์การให้ซ้ำที่ใช้ประเมิน]	[ระบุผลลัพธ์การประเมิน]	[ระบุคำอธิบาย]

สรุปผลลัพธ์การประเมิน ผ่าน ไม่ผ่าน

รายการที่ต้องปรับปรุงแก้ไข :

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : การกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา

ลำดับ	แบบจำลอง/วัฏจักรชีวิตซอฟต์แวร์ที่ใช้ในการพัฒนา	คำอธิบาย
[ลำดับ]	[ระบุแบบจำลอง/วัฏจักรชีวิตซอฟต์แวร์ที่ใช้ในการพัฒนา]	[ระบุคำอธิบาย]
ลำดับ	กลยุทธ์ที่ใช้ในการพัฒนา	คำอธิบาย
[ลำดับ]	[ระบุกลยุทธ์ที่ใช้ในการพัฒนา]	[ระบุคำอธิบาย]

หมายเหตุ :

.....

.....

.....

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-T-n]	แบบฟอร์มการกำหนดแผนการปรับแต่ง	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการกำหนดแผนการปรับแต่ง

ประเภทของกระบวนการ : กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป
 ชื่อกระบวนการหลักที่ดำเนินการ : วางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป
 ชื่อกระบวนการย่อยที่ดำเนินการ : กำหนดแผนการปรับแต่ง
 ข้อมูลนำเข้าของกระบวนการ :

 รหัสเอกสารอ้างอิง :
 วัน / เดือน / ปี ที่ดำเนินการ :
 ชื่อผู้บันทึกผล :
 หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการกำหนดแผนการปรับแต่ง

ลำดับ	วัฏจักรชีวิตซอฟต์แวร์	กิจกรรม	หน้าที่	คำอธิบาย
[ลำดับ]	[ระบุวัฏจักรชีวิตซอฟต์แวร์ ของแบบจำลอง กระบวนการที่เลือก ตามลำดับ]	[ระบุกิจกรรม]	[ระบุหน้าที่]	[ระบุคำอธิบาย]

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-I-n]	แบบฟอร์มการกำหนดแผนการบูรณาการ	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการกำหนดแผนการบูรณาการ

ประเภทของกระบวนการ : กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป
 ชื่อกระบวนการหลักที่ดำเนินการ : วางแผนโครงการการบูรณาการซอฟต์แวร์สำเร็จรูป
 ชื่อกระบวนการย่อยที่ดำเนินการ : กำหนดแผนการบูรณาการ
 ข้อมูลนำเข้าของกระบวนการ :

 รหัสเอกสารอ้างอิง :
 วัน / เดือน / ปี ที่ดำเนินการ :
 ชื่อผู้บันทึกผล :
 หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการกำหนดแผนการบูรณาการ

ลำดับ	วัฏจักรชีวิตซอฟต์แวร์	กิจกรรม	หน้าที่	คำอธิบาย
[ลำดับ]	[ระบุวัฏจักรชีวิตซอฟต์แวร์ ของแบบจำลอง กระบวนการที่เลือก ตามลำดับ]	[ระบุกิจกรรม]	[ระบุหน้าที่]	[ระบุคำอธิบาย]

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-T/I/M-n]	แบบฟอร์มการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ
 กระบวนการการบำรุงรักษา

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : ทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์

ลำดับ	ชื่อสายผลิตภัณฑ์	ชื่อสินทรัพย์	คุณลักษณะ	คำอธิบาย
[ลำดับ]	[รายชื่อสายผลิตภัณฑ์ที่ทำการทบทวน]	[รายชื่อสินทรัพย์ที่สามารถนำมาใช้ซ้ำได้]	[ระบุคุณลักษณะของสินทรัพย์ที่ถูกนำมา ใช้ซ้ำ]	[ระบุคำอธิบาย]

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : กำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

สถาปัตยกรรม	แบบรูป / ส่วนประกอบสถาปัตยกรรมที่ใช้	คำอธิบาย
สถาปัตยกรรมเชิงโครงสร้าง	[ระบุลำดับ] [แบบรูป / ส่วนประกอบสถาปัตยกรรมที่ใช้]	[ระบุลำดับ] [ระบุคำอธิบายจุดประสงค์การใช้งานของแบบรูป / ส่วนประกอบสถาปัตยกรรมที่ใช้]
สถาปัตยกรรมเชิงพลวัต	[ระบุลำดับ] [แบบรูป / ส่วนประกอบสถาปัตยกรรมที่ใช้]	[ระบุลำดับ] [ระบุคำอธิบายจุดประสงค์การใช้งานของแบบรูป / ส่วนประกอบสถาปัตยกรรมที่ใช้]

หมายเหตุ :

[ชื่อแฟ้ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
--------------------	--------------------------------

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการกำหนดรายละเอียดสถาปัตยกรรม	หน้าที่ n/m
	สายผลิตภัณฑ์ซอฟต์แวร์	แก้ครั้งที่ n

แบบฟอร์มการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : กำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์

สถาปัตยกรรม	แบบรูป / ส่วนประกอบ สถาปัตยกรรมที่ใช้	รายละเอียดส่วนประกอบ สถาปัตยกรรม	คำอธิบาย
สถาปัตยกรรม เชิงโครงสร้าง	[ระบุลำดับ] [แบบรูป / ส่วนประกอบสถาปัตยกรรมที่ใช้]	[ระบุลำดับ] [ระบุรายละเอียด ส่วนประกอบสถาปัตยกรรม]	[ระบุลำดับ] [ระบุ คำอธิบาย]
สถาปัตยกรรม เชิงพลวัต	[ระบุลำดับ] [แบบรูป / ส่วนประกอบสถาปัตยกรรมที่ใช้]	[ระบุลำดับ] [ระบุรายละเอียด ส่วนประกอบสถาปัตยกรรม]	[ระบุลำดับ] [ระบุ คำอธิบาย]

หมายเหตุ :

.....

.....

.....

.....

.....

.....

ชื่อแฟ้ม.นามสกุล	วันที่พิมพ์ [วัน / เดือน / ปี]
------------------	--------------------------------

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการพัฒนาส่วนประกอบ	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการพัฒนาส่วนประกอบ

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : การพัฒนาส่วนประกอบ

ข้อมูลนำเข้าของกระบวนการ :

.....

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการพัฒนาส่วนประกอบ

ลำดับ	รายการที่ต้องพัฒนา	คุณลักษณะ	สถานะการพัฒนา	คำอธิบาย
[ลำดับ]	[ระบุรายการที่ต้องพัฒนา]	[ระบุคุณลักษณะในการพัฒนา/ คุณลักษณะที่ต้องใช้]	[ระบุสถานะการพัฒนา]	[ระบุคำอธิบาย]

หมายเหตุ :

.....

.....

.....

.....

.....

.....

.....

.....

[ชื่อแฟ้ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
--------------------	--------------------------------

รหัสเอกสาร : [F-T-n]	แบบฟอร์มการบูรณาการกับระบบที่มีอยู่	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการบูรณาการกับระบบที่มีอยู่

ประเภทของกระบวนการ : กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการหลักที่ดำเนินการ : ปรับแต่งซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : บูรณาการกับระบบที่มีอยู่

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการบูรณาการกับระบบที่มีอยู่

ลำดับ	รายการสภาพแวดล้อมที่ต้องบูรณาการ	ผลการบูรณาการ	คำอธิบาย
[ลำดับ]	[ระบุรายการสภาพแวดล้อมที่ต้องใช้ในการบูรณาการซอฟต์แวร์]	[ระบุผลการบูรณาการกับระบบที่มีอยู่]	[ระบุคำอธิบาย]

สรุปผลลัพธ์การบูรณาการกับระบบที่มีอยู่ สมบูรณ์ ไม่สมบูรณ์

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

.....

หมายเหตุ :

.....

.....

.....

ชื่อแฟ้ม.นามสกุล	วันที่พิมพ์ [วัน / เดือน / ปี]
------------------	--------------------------------

รหัสเอกสาร : [F-I-n]	แบบฟอร์มการบูรณาการระบบและซอฟต์แวร์	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการบูรณาการระบบและซอฟต์แวร์

ประเภทของกระบวนการ : กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการหลักที่ดำเนินการ : บูรณาการซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : บูรณาการระบบและซอฟต์แวร์

ข้อมูลนำเข้าของกระบวนการ :

.....

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการบูรณาการระบบและซอฟต์แวร์

ลำดับ	รายการรหัสคำสั่งเพิ่มเติม / สินทรัพย์ / ซอฟต์แวร์สำเร็จรูป	คุณลักษณะ	ผลการบูรณาการ	คำอธิบาย
[ลำดับ]	[ระบุรายการรหัสคำสั่งเพิ่มเติม / สินทรัพย์ / ซอฟต์แวร์สำเร็จรูป ที่ใช้ ในการบูรณาการ]	[ระบุคุณลักษณะ ของส่วนประกอบ ซอฟต์แวร์]	[ระบุผลการ บูรณาการระบบ และซอฟต์แวร์]	[ระบุคำอธิบาย]

สรุปผลลัพธ์การบูรณาการระบบและซอฟต์แวร์ สมบูรณ์ ไม่สมบูรณ์

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

หมายเหตุ :

.....

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการทดสอบสายผลิตภัณฑ์	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการทดสอบสายผลิตภัณฑ์

ประเภทของกระบวนการ : กระบวนการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : ทดสอบสายผลิตภัณฑ์

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการทดสอบสายผลิตภัณฑ์

ลำดับ	รายการทดสอบ ซอฟต์แวร์	ประเภทของการทดสอบ		ผลการทดสอบ	คำอธิบาย
		การบูรณาการ	เชิงหน้าที่		
[ลำดับ]	[ระบุรายการที่จะทำการ ทดสอบซอฟต์แวร์]	[ระบุประเภทการ ทดสอบการ บูรณาการ]	[ระบุประเภทการ ทดสอบเชิง หน้าที่]	[ระบุผลการ ทดสอบ]	[ระบุ คำอธิบาย]

สรุปผลลัพธ์การทดสอบสายผลิตภัณฑ์ สมบูรณ์ ไม่สมบูรณ์

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

.....

หมายเหตุ :

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : จัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป

ลำดับ	ชื่อผลิตภัณฑ์ซอฟต์แวร์	ชื่อสายผลิตภัณฑ์	เวอร์ชันของซอฟต์แวร์	ข้อมูลสารสนเทศ	คำอธิบาย
[ลำดับ]	[ระบุชื่อผลิตภัณฑ์ซอฟต์แวร์]	[ระบุชื่อสายผลิตภัณฑ์ที่เป็นต้นแบบของผลิตภัณฑ์ซอฟต์แวร์ที่ได้]	[ระบุเวอร์ชันของซอฟต์แวร์]	[ระบุข้อมูลสารสนเทศ]	[ระบุคำอธิบาย]

หมายเหตุ :

[ชื่อแฟ้ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
--------------------	--------------------------------

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บ	หน้าที่ n/m
	สายผลิตภัณฑ์ซอฟต์แวร์	แก้ไขครั้งที่ n

**แบบฟอร์มการปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บ
สายผลิตภัณฑ์ซอฟต์แวร์**

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : ปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการปรับปรุงซอฟต์แวร์สำเร็จรูปที่อยู่ในแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์

ลำดับ	ชื่อผลิตภัณฑ์ซอฟต์แวร์	ชื่อสายผลิตภัณฑ์	เลขที่เอกสารจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป	คำอธิบาย
[ลำดับ]	[ระบุชื่อผลิตภัณฑ์ซอฟต์แวร์]	[ระบุชื่อสายผลิตภัณฑ์ที่จัดเก็บซอฟต์แวร์]	[ระบุเลขที่เอกสารจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป]	[ระบุคำอธิบาย]

หมายเหตุ :

.....

.....

.....

.....

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการวางแผนการติดตั้ง	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการวางแผนการติดตั้ง

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : วางแผนการติดตั้ง

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการวางแผนการติดตั้ง

ลำดับ	กระบวนการติดตั้งซอฟต์แวร์	สภาพแวดล้อมที่ต้องใช้	คำอธิบาย
[ลำดับ]	[ระบุกระบวนการติดตั้งซอฟต์แวร์]	[ระบุสภาพแวดล้อมที่ต้องใช้]	[ระบุคำอธิบาย]

หมายเหตุ :

.....

.....

.....

.....

.....

.....

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการส่งมอบผลิตภัณฑ์ซอฟต์แวร์	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการส่งมอบผลิตภัณฑ์ซอฟต์แวร์

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : ส่งมอบผลิตภัณฑ์ซอฟต์แวร์

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการส่งมอบผลิตภัณฑ์ซอฟต์แวร์

ลำดับ	ชื่อผลิตภัณฑ์ซอฟต์แวร์	รายการที่ต้องส่งมอบ	คำอธิบาย
[ลำดับ]	[ระบุชื่อผลิตภัณฑ์ซอฟต์แวร์]	[ระบุรายการที่ต้องส่งมอบ]	[ระบุคำอธิบาย]

วันที่ส่งมอบ :

ลงนามผู้ส่งมอบผลิตภัณฑ์ซอฟต์แวร์:

(.....)

ลงนามผู้รับมอบผลิตภัณฑ์ซอฟต์แวร์:

(.....)

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-T/I-n]	แบบฟอร์มการจัดการสนับสนุนการฝึกอบรม	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการจัดการสนับสนุนการฝึกอบรม

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ :

ชื่อกระบวนการย่อยที่ดำเนินการ : จัดการสนับสนุนการฝึกอบรม

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

ชื่อหลักสูตร

คำอธิบายหลักสูตร

.....

วันเริ่มต้น

วันสิ้นสุด

รายชื่อบุคลากรที่เข้ารับการฝึกอบรม

ลำดับ	ชื่อ-นามสกุล	ตำแหน่ง	ลายเซ็น
[ลำดับ]	[ชื่อ-นามสกุล]	[ตำแหน่ง]	[ลายเซ็น]

หมายเหตุ :

[ชื่อแฟ้ม-นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
--------------------	--------------------------------

รหัสเอกสาร : [F-M-n]	แบบฟอร์มการกำหนดแผนการบำรุงรักษา	หน้าที่ n/m
		แก้ไขครั้งที่ n

แบบฟอร์มการกำหนดแผนการบำรุงรักษา

ประเภทของกระบวนการ : กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการหลักที่ดำเนินการ : วางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : กำหนดแผนการบำรุงรักษา

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

เลขที่รายการรายงานปัญหา/คำร้องขอการดัดแปลง :

เลขที่รายการผลการประเมินความสามารถในการแก้ไขปัญหาที่ตอบสนองต่อปัญหา/คำร้องขอการ

ดัดแปลง :

แบบฟอร์มการกำหนดแผนการบำรุงรักษา

ลำดับ	กระบวนการบำรุงรักษา	กิจกรรม	หน้าที่	คำอธิบาย
[ลำดับ]	[ระบุกระบวนการบำรุงรักษา]	[ระบุกิจกรรม]	[ระบุหน้าที่]	[ระบุคำอธิบาย]

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-M-n]	แบบฟอร์มการสร้างกระบวนการบำรุงรักษา	หน้าที่ n/m
		แก้ไขครั้งที่ n

แบบฟอร์มการสร้างกระบวนการบำรุงรักษา

ประเภทของกระบวนการ : กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการหลักที่ดำเนินการ : วางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : สร้างกระบวนการบำรุงรักษา

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการสร้างกระบวนการบำรุงรักษา

ลำดับ	กระบวนการสนับสนุนการบำรุงรักษาซอฟต์แวร์	กิจกรรม	หน้าที่	คำอธิบาย
[ลำดับ]	[ระบุกระบวนการสนับสนุนการบำรุงรักษาซอฟต์แวร์]	[ระบุกิจกรรม]	[ระบุหน้าที่]	[ระบุคำอธิบาย]

หมายเหตุ :

.....

.....

.....

.....

.....

.....

.....

.....

[ชื่อแฟ้ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
--------------------	--------------------------------

รหัสเอกสาร : [F-M-n]	แบบฟอร์มการวิเคราะห์ผลกระทบสิ่งแวดล้อมที่สามารถนำมาใช้ซ้ำ	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการวิเคราะห์ผลกระทบสิ่งแวดล้อมที่สามารถนำมาใช้ซ้ำ

ประเภทของกระบวนการ : กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการหลักที่ดำเนินการ : วิเคราะห์ปัญหาและการดัดแปลง

ชื่อกระบวนการย่อยที่ดำเนินการ : วิเคราะห์ผลกระทบสิ่งแวดล้อมที่สามารถนำมาใช้ซ้ำ

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการวิเคราะห์ผลกระทบสิ่งแวดล้อมที่สามารถนำมาใช้ซ้ำ

ลำดับ	ชื่อสิ่งแวดล้อมที่สามารถนำมาใช้ซ้ำ	คุณลักษณะ	ผลกระทบของสิ่งแวดล้อม
[ลำดับ]	[ระบุชื่อสิ่งแวดล้อมที่สามารถนำมาใช้ซ้ำ]	[ระบุคุณลักษณะการนำมาใช้ซ้ำของสิ่งแวดล้อม]	[ผลกระทบของสิ่งแวดล้อมต่อรายงานปัญหาและคำร้องขอการดัดแปลง]

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-M-n]	แบบฟอร์มการกำหนดวิธีการบำรุงรักษา	หน้าที่ n/m
		แก้ไขครั้งที่ n

แบบฟอร์มการกำหนดวิธีการบำรุงรักษา

ประเภทของกระบวนการ : กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการหลักที่ดำเนินการ : วิเคราะห์ปัญหาและการดัดแปลง

ชื่อกระบวนการย่อยที่ดำเนินการ : กำหนดวิธีการบำรุงรักษา

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการกำหนดวิธีการบำรุงรักษา

ลำดับ	รายการรายงานปัญหา / คำร้องขอการดัดแปลง	วิธีการแก้ไขปัญหา	คำอธิบาย
[ลำดับ]	[ระบุรายการรายงานปัญหา / คำร้องขอการดัดแปลง]	[ระบุวิธีการแก้ไขปัญหา ซึ่งประกอบด้วย การ ดัดแปลง / ย้าย / ยกเลิกข้อมูล]	[ระบุคำอธิบาย]

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-M-n]	แบบฟอร์มการวางแผนการดัดแปลง ย้ายหรือยกเลิกข้อมูล	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการวางแผนการดัดแปลง ย้ายหรือยกเลิกข้อมูล

ประเภทของกระบวนการ : กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการหลักที่ดำเนินการ : ดัดแปลง ย้าย และยกเลิกข้อมูลซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : วางแผนการดัดแปลง ย้าย หรือยกเลิกข้อมูล

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการวางแผนการดัดแปลง ย้ายหรือยกเลิกข้อมูล

ลำดับ	รายการรายงาน ปัญหาและคำร้อง ขอการดัดแปลง	วิธีการแก้ไขปัญหา	กิจกรรม	หน้าที่	คำอธิบาย
[ลำดับ]	[ระบุรายการ รายงานปัญหาและ คำร้องขอการ ดัดแปลง]	[ระบุวิธีการแก้ไขปัญหา ซึ่งประกอบด้วย การดัดแปลง / ย้าย / ยกเลิก ข้อมูล]	[ระบุกิจกรรม]	[ระบุหน้าที่]	[ระบุ คำอธิบาย]

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-M-n]	แบบฟอร์มการดัดแปลง ย้ายหรือยกเลิกข้อมูล	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการดัดแปลง ย้ายหรือยกเลิกข้อมูล

ประเภทของกระบวนการ : กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการหลักที่ดำเนินการ : ดัดแปลง ย้าย และยกเลิกข้อมูลซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : ดัดแปลง ย้าย หรือยกเลิกข้อมูล

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการดัดแปลง ย้ายหรือยกเลิกข้อมูล

ลำดับ	รายการรายงานปัญหา และคำร้องขอการ ดัดแปลง	วิธีการแก้ไขปัญหา	ผลการปฏิบัติ	คำอธิบาย
[ลำดับ]	[ระบุรายการรายงาน ปัญหาและคำร้องขอการ ดัดแปลง]	[ระบุวิธีการแก้ไข ปัญหาและคำร้อง ขอการดัดแปลง]	[ระบุผลการปฏิบัติ]	[ระบุคำอธิบาย]

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-M-n]	แบบฟอร์มการแจ้งผลการบำรุงรักษา	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการแจ้งผลการบำรุงรักษา

ประเภทของกระบวนการ : กระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการหลักที่ดำเนินการ : ดัดแปลง ย้าย และยกเลิกข้อมูลซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : แจ้งผลการบำรุงรักษา

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการแจ้งผลการบำรุงรักษา

ลำดับ	วิธีการแก้ไขปัญหาและ คำร้องขอการดัดแปลง	รายงานการปฏิบัติการ	รายการสินทรัพย์ที่มีการ เปลี่ยนแปลง
[ลำดับ]	[ระบุวิธีการแก้ไขปัญหาและ คำร้องขอการดัดแปลง]	[ระบุรายงานการปฏิบัติการ (ดัดแปลง/ย้าย/ยกเลิกข้อมูล)]	[ระบุผลรายการสินทรัพย์ที่มี การเปลี่ยนแปลง]

หมายเหตุ :

.....

.....

วันที่แจ้งผลการบำรุงรักษา :

ลงนามผู้แจ้งผล:

(.....)

ลงนามผู้รับแจ้งผล:

(.....)

ผู้จัดการสินทรัพย์ (Assets Manager)

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-S-n]	แบบฟอร์มการสร้างจุดมุ่งหมายของการวัดผล	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการสร้างจุดมุ่งหมายของการวัดผล

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ
 กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ : วัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : สร้างจุดมุ่งหมายของการวัดผล

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการสร้างจุดมุ่งหมายของการวัดผล

ลำดับ	จุดมุ่งหมายของการวัดผล	มาตรวัด	คำอธิบาย
[ลำดับ]	[ระบุจุดมุ่งหมายของการวัดผล]	[ระบุมาตรวัด]	[ระบุคำอธิบาย]

หมายเหตุ :

.....

.....

.....

.....

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-S-n]	แบบฟอร์มการวางแผนกระบวนการงานวัดผล	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการวางแผนกระบวนการงานวัดผล

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ
 กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ : วัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : วางแผนกระบวนการงานวัดผล

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการวางแผนกระบวนการงานวัดผล

ลำดับ	จุดมุ่งหมายของการวัดผล	มาตรวัดที่ใช้	กิจกรรม	หน้าที่	คำอธิบาย
[ลำดับ]	[ระบุจุดมุ่งหมายของการวัดผล]	[ระบุมาตรวัดที่ใช้]	[ระบุกิจกรรม]	[ระบุหน้าที่]	[ระบุคำอธิบาย]

หมายเหตุ :

.....

.....

.....

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-S-n]	แบบฟอร์มการวัดผลการพัฒนาซอฟต์แวร์	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการวัดผลการพัฒนาซอฟต์แวร์

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ
 กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ : วัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : วัดผลการพัฒนาซอฟต์แวร์

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

แบบฟอร์มการวัดผลการพัฒนาซอฟต์แวร์

ลำดับ	จุดมุ่งหมายของการวัดผล	มาตรวัดที่ใช้	ผลลัพธ์
[ลำดับ]	[ระบุจุดมุ่งหมายของการวัดผล]	[ระบุมาตรวัดที่ใช้]	[ระบุผลลัพธ์ที่ได้จากการวัดผล]

สรุปผลลัพธ์การวัดผลการพัฒนาซอฟต์แวร์ :

ดีมาก ดี ปานกลาง น้อย น้อยมาก

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

.....

หมายเหตุ :

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-S-n]	แบบฟอร์มการวิเคราะห์และจัดเก็บผลลัพธ์	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการวิเคราะห์และจัดเก็บผลลัพธ์

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ
 กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ : วัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : วิเคราะห์และจัดเก็บผลลัพธ์

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

วัน / เดือน / ปี (วันที่วิเคราะห์):

วัน / เดือน / ปี (วันที่จัดเก็บผลลัพธ์):

แบบฟอร์มการวิเคราะห์และจัดเก็บผลลัพธ์

ลำดับ	จุดมุ่งหมายของการวัดผล	ผลลัพธ์	รายการผลลัพธ์การวิเคราะห์ที่จัดเก็บ
[ลำดับ]	[ระบุจุดมุ่งหมายของการวัดผล]	[ระบุผลลัพธ์ที่ได้จากการวัดผล]	[ระบุรายการผลลัพธ์การวิเคราะห์ที่จัดเก็บ]

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [F-S-n]	แบบฟอร์มการรายงานการวัดผลและแนวทางสำหรับ ปรับปรุงกระบวนการ	หน้าที่ n/m
		แก้ครั้งที่ n

แบบฟอร์มการรายงานการวัดผลและแนวทางสำหรับปรับปรุงกระบวนการ

ประเภทของกระบวนการ : กระบวนการการปรับแต่งฯ กระบวนการการบูรณาการฯ
 กระบวนการการบูรณาการฯ

ชื่อกระบวนการหลักที่ดำเนินการ : วัดผลกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูป

ชื่อกระบวนการย่อยที่ดำเนินการ : รายงานการวัดผลและแนวทางสำหรับปรับปรุงกระบวนการ

ข้อมูลนำเข้าของกระบวนการ :

.....

รหัสเอกสารอ้างอิง :

วัน / เดือน / ปี ที่ดำเนินการ :

ชื่อผู้บันทึกผล :

หน้าที่รับผิดชอบ : ผู้จัดการโครงการ ทีมผู้พัฒนา ทีมผู้ประเมิน

วัน / เดือน / ปี (วันที่รายงานการวัดผล):

แบบฟอร์มการรายงานการวัดผลและแนวทางสำหรับปรับปรุงกระบวนการ

ลำดับ	จุดมุ่งหมายของการวัดผล	มาตรวัด	ผลลัพธ์	แนวทางสำหรับปรับปรุงกระบวนการ
[ลำดับ]	[ระบุจุดมุ่งหมายของการวัดผล]	[ระบุมาตรวัดที่ใช้]	[ระบุผลลัพธ์ที่ได้จากการวัดผล]	[ระบุแนวทางสำหรับปรับปรุงกระบวนการ]

หมายเหตุ :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รายการตรวจสอบเกณฑ์การใช้ซ้ำสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป (Reusable Criteria Checklists for COTS Tailoring Process)

1. ประเมินข้อกำหนดความต้องการ (Evaluate Requirements Specification)

รหัสเอกสาร : [C-T-n]	รายการตรวจสอบเกณฑ์การใช้ซ้ำ (ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ)	หน้าที่ 1/1
----------------------	---	-------------

ชื่อกระบวนการ (Process Name) : กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป					
ชื่อรายการตรวจสอบ (Checklist Name) : เกณฑ์การใช้ซ้ำ (Reusable Criteria)					
คำอธิบาย (Description) : รายการตรวจสอบสำหรับเกณฑ์การใช้ซ้ำ ประกอบด้วย 5 ส่วน ดังนี้ ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ ส่วนที่ 4 ประเมินการบูรณาการกับระบบที่มีอยู่ และส่วนที่ 5 ประเมินการวางแผนการติดตั้ง					
ส่วนที่ 1 - ประเมินข้อกำหนดความต้องการ					
เกณฑ์การใช้ซ้ำที่อยู่ในส่วนนี้มีความสำคัญต่อความสำเร็จในการปรับแต่งซอฟต์แวร์สำเร็จรูปเพราะเป็นส่วนที่ประเมินความต้องการซอฟต์แวร์ ซึ่งเป็นเป้าหมายหลักของการปรับแต่งซอฟต์แวร์สำเร็จรูป ให้พิจารณาตามรายการตรวจสอบจากเกณฑ์การใช้ซ้ำโดยให้คะแนนตามความถูกต้องและความสำคัญจากมากไปหาน้อยตามลำดับ					
4	3	2	1	N/A	เกณฑ์การใช้ซ้ำ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ข้อกำหนดความต้องการซอฟต์แวร์มีความชัดเจน และสามารถดำเนินการปรับแต่งซอฟต์แวร์สำเร็จรูปเพื่อตอบสนองต่อความต้องการได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ซอฟต์แวร์สำเร็จรูปที่เป็นสินทรัพย์ขององค์กรนั้นมีความยืดหยุ่นต่อการปรับแต่งเพื่อบรรลุเป้าหมายได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการปรับแต่งที่ได้ นั้นออกแบบตามกระบวนการวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สินทรัพย์ขององค์กรที่สามารถนำมาใช้ซ้ำได้นั้น มีเพียงพอต่อการปรับแต่งซอฟต์แวร์สำเร็จรูปและมีความยืดหยุ่นเมื่อเกิดปัญหาที่สินทรัพย์ไม่สามารถตอบสนองต่อข้อกำหนดความต้องการได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการปรับแต่งสามารถตอบสนองต่อแบบจำลองและกลยุทธ์ในการพัฒนาซอฟต์แวร์ที่ได้ รวมถึงการนำสินทรัพย์มาใช้ซ้ำได้เป็นอย่างดี
ผลการประเมิน <input type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน					
ชื่อผู้ประเมิน / ทีมผู้ประเมิน					
*ผลการประเมินจะผ่านหรือไม่ขึ้นอยู่กับพิจารณาของทีมผู้ประเมิน					

รายการที่ต้องปรับปรุงแก้ไข :

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

2. ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Evaluate Software Product Lines Architecture)

รหัสเอกสาร : [C-T-n]	รายการตรวจสอบเกณฑ์การใช้ซ้ำ (ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์)	หน้าที่ 1/1
----------------------	--	-------------

ชื่อกระบวนการ (Process Name) : กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป					
ชื่อรายการตรวจสอบ (Checklist Name) : เกณฑ์การใช้ซ้ำ (Reusable Criteria)					
คำอธิบาย (Description) : รายการตรวจสอบสำหรับเกณฑ์การใช้ซ้ำ ประกอบด้วย 5 ส่วน ดังนี้ ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ ส่วนที่ 4 ประเมินการบูรณาการกับระบบที่มีอยู่ และส่วนที่ 5 ประเมินการวางแผนการติดตั้ง					
ส่วนที่ 2 - ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์					
เกณฑ์การใช้ซ้ำที่อยู่ในส่วนนี้เป็นการประเมินรายละเอียดความสมบูรณ์ของสถาปัตยกรรมซอฟต์แวร์ที่ใช้ในกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป ให้พิจารณาตามรายการตรวจสอบจากเกณฑ์การใช้ซ้ำโดยให้คะแนนตามความถูกต้องและความสำคัญจากมากไปหาน้อยตามลำดับ					
4	3	2	1	N/A	เกณฑ์การใช้ซ้ำ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ข้อมูลนำเข้าของกระบวนการที่ใช้ในการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์มีความสมบูรณ์
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สามารถออกแบบสถาปัตยกรรมเชิงโครงสร้างที่ตอบสนองต่อความต้องการได้ครอบคลุมเป็นอย่างดี
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สามารถออกแบบสถาปัตยกรรมเชิงพลวัตที่ตอบสนองต่อความต้องการได้ครอบคลุมเป็นอย่างดี
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	รายละเอียดสถาปัตยกรรมซอฟต์แวร์ที่ได้นั้น มีความสมบูรณ์และสอดคล้องกับสถาปัตยกรรมซอฟต์แวร์ที่ได้ออกแบบไว้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สถาปัตยกรรมและรายละเอียดสถาปัตยกรรมฯ ที่ได้นั้นสามารถตอบสนองต่อข้อกำหนดความต้องการซอฟต์แวร์และแผนการปรับแต่งได้เป็นอย่างดี
ผลการประเมิน <input type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน					
ชื่อผู้ประเมิน / ทีมผู้ประเมิน					
*ผลการประเมินจะผ่านหรือไม่ขึ้นอยู่กับพิจารณาของทีมผู้ประเมิน					

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

3. ประเมินการพัฒนาส่วนประกอบ (Evaluate Implementation Component)

รหัสเอกสาร : [C-T-n]	รายการตรวจสอบเกณฑ์การใช้ซ้ำ (ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ)	หน้าที่ 1/1
----------------------	--	-------------

ชื่อกระบวนการ (Process Name) : กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป					
ชื่อรายการตรวจสอบ (Checklist Name) : เกณฑ์การใช้ซ้ำ (Reusable Criteria)					
คำอธิบาย (Description) : รายการตรวจสอบสำหรับเกณฑ์การใช้ซ้ำ ประกอบด้วย 5 ส่วน ดังนี้ ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ ส่วนที่ 4 ประเมินการบูรณาการกับระบบที่มีอยู่ และส่วนที่ 5 ประเมินการวางแผนการติดตั้ง					
ส่วนที่ 3 - ประเมินการพัฒนาส่วนประกอบ เกณฑ์การใช้ซ้ำที่อยู่ในส่วนนี้เป็นการประเมินการพัฒนาส่วนประกอบของการปรับแต่งซอฟต์แวร์สำเร็จรูป ให้พิจารณาตามรายการตรวจสอบจากเกณฑ์การใช้ซ้ำโดยให้คะแนนตามความถูกต้องและความสำคัญจากมากไปหาน้อยตามลำดับ					
4	3	2	1	N/A	เกณฑ์การใช้ซ้ำ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากกระบวนการการพัฒนาส่วนประกอบนั้นสอดคล้องกับข้อกำหนดความต้องการซอฟต์แวร์ แผนการปรับแต่ง รวมถึงสถาปัตยกรรมและรายละเอียดสถาปัตยกรรมซอฟต์แวร์ที่ได้ออกแบบไว้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ทีมผู้พัฒนาได้ดำเนินการพัฒนาส่วนประกอบซอฟต์แวร์ โดยปรับแต่งซอฟต์แวร์จากสินทรัพย์ตามลำดับความสำคัญของคุณลักษณะตามหลักการของสายผลิตภัณฑ์ซอฟต์แวร์
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สินทรัพย์ที่สามารถนำมาใช้ซ้ำสำหรับกระบวนการการปรับแต่งนั้นสามารถนำมาบูรณาการกับซอฟต์แวร์สำเร็จรูปได้ มีจำนวนที่เพียงพอและมีความยืดหยุ่นในกรณีที่เกิดปัญหาขึ้น
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ผลิตภัณฑ์ซอฟต์แวร์ที่ได้สามารถตอบสนองต่อข้อกำหนดความต้องการซอฟต์แวร์ ซึ่งอยู่ในเกณฑ์ที่สามารถยอมรับได้
ผลการประเมิน <input type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน					
ชื่อผู้ประเมิน / ทีมผู้ประเมิน					
*ผลการประเมินจะผ่านหรือไม่ขึ้นอยู่กับการพิจารณาของทีมผู้ประเมิน					

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

.....

ชื่อเพิ่ม.นามสกุล	วันที่พิมพ์ [วัน / เดือน / ปี]
-------------------	--------------------------------

4. ประเมินการบูรณาการกับระบบที่มีอยู่ (Evaluate Existing System Integration)

รหัสเอกสาร : [C-T-n]	รายการตรวจสอบเกณฑ์การใช้ซ้ำ (ส่วนที่ 4 ประเมินการบูรณาการกับระบบที่มีอยู่)	หน้าที่ 1/1
----------------------	---	-------------

ชื่อกระบวนการ (Process Name) : กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป					
ชื่อรายการตรวจสอบ (Checklist Name) : เกณฑ์การใช้ซ้ำ (Reusable Criteria)					
คำอธิบาย (Description) : รายการตรวจสอบสำหรับเกณฑ์การใช้ซ้ำ ประกอบด้วย 5 ส่วน ดังนี้ ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ ส่วนที่ 4 ประเมินการบูรณาการกับระบบที่มีอยู่ และส่วนที่ 5 ประเมินการวางแผนการติดตั้ง					
ส่วนที่ 4 - ประเมินการบูรณาการกับระบบที่มีอยู่ เกณฑ์การใช้ซ้ำที่อยู่ในส่วนนี้เป็นการประเมินการบูรณาการผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากกระบวนการพัฒนาส่วนประกอบกับระบบหรือสภาพแวดล้อมที่มีอยู่ขององค์กร ให้พิจารณาตามรายการตรวจสอบจากเกณฑ์การใช้ซ้ำโดยให้คะแนนตามความถูกต้องและความสำคัญจากมากไปหาน้อยตามลำดับ					
4	3	2	1	N/A	เกณฑ์การใช้ซ้ำ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ผลิตภัณฑ์ซอฟต์แวร์ที่ได้ สามารถบูรณาการและทำงานร่วมกับระบบหรือสภาพแวดล้อมที่มีอยู่เดิมขององค์กรได้เป็นอย่างดี
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ไม่มีความต้องการเพิ่มเติมสำหรับระบบและสภาพแวดล้อมที่มีอยู่ในการติดตั้งซอฟต์แวร์อื่นที่ช่วยให้ผลิตภัณฑ์ซอฟต์แวร์สามารถทำงานร่วมกับระบบที่มีอยู่เดิมได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สามารถบูรณาการฐานข้อมูลที่มีอยู่เดิมของระบบกับฐานข้อมูลของผลิตภัณฑ์ซอฟต์แวร์ให้สามารถทำงานร่วมกันได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สามารถบูรณาการกระบวนการที่มีอยู่เดิมกับการใช้งานผลิตภัณฑ์ซอฟต์แวร์ที่สามารถสนับสนุนกระบวนการนั้นๆ ได้เป็นอย่างดี
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ไม่มีความต้องการที่จะแปลงผันข้อมูลของผลิตภัณฑ์ซอฟต์แวร์ให้สามารถทำงานร่วมกับระบบหรือสภาพแวดล้อมที่มีอยู่
ผลการประเมิน <input type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน					
ชื่อผู้ประเมิน / ทีมผู้ประเมิน					
*ผลการประเมินจะผ่านหรือไม่ขึ้นอยู่กับพิจารณาของทีมผู้ประเมิน					

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

5. ประเมินการวางแผนการติดตั้ง (Evaluate Installation Plan)

รหัสเอกสาร : [C-T-n]	รายการตรวจสอบเกณฑ์การใช้ซ้ำ (ส่วนที่ 5 ประเมินการวางแผนการติดตั้ง)	หน้าที่ 1/1
----------------------	---	-------------

ชื่อกระบวนการ (Process Name) : กระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป					
ชื่อรายการตรวจสอบ (Checklist Name) : เกณฑ์การใช้ซ้ำ (Reusable Criteria)					
คำอธิบาย (Description) : รายการตรวจสอบสำหรับเกณฑ์การใช้ซ้ำ ประกอบด้วย 5 ส่วน ดังนี้ ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ ส่วนที่ 4 ประเมินการบูรณาการกับระบบที่มีอยู่ และส่วนที่ 5 ประเมินการวางแผนการติดตั้ง					
ส่วนที่ 5 - ประเมินการวางแผนการติดตั้ง					
เกณฑ์การใช้ซ้ำที่อยู่ในส่วนนี้เป็นการประเมินการวางแผนการติดตั้งผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากกระบวนการปรับแต่งซอฟต์แวร์สำเร็จรูป ให้พิจารณาตามรายการตรวจสอบจากเกณฑ์การใช้ซ้ำโดยให้คะแนนตามความถูกต้องและความสำคัญจากมากไปหาน้อยตามลำดับ					
4	3	2	1	N/A	เกณฑ์การใช้ซ้ำ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการติดตั้งที่ได้นั้นมีความครอบคลุมกระบวนการและสภาพแวดล้อมที่จำเป็นสำหรับการติดตั้งซอฟต์แวร์
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการติดตั้งที่ได้นั้นมีส่วนที่จัดการแก้ไขปัญหาในกรณีที่ไม่สามารถติดตั้งผลิตภัณฑ์ซอฟต์แวร์ได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการติดตั้งที่ได้นั้นมีส่วนที่อธิบายกระบวนการสำหรับการติดตั้งซอฟต์แวร์โดยละเอียด
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	กระบวนการที่ได้จากการวางแผนการติดตั้งนั้นสามารถดำเนินการได้อย่างเป็นระบบ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการติดตั้งที่ได้นั้นสอดคล้องต่อข้อกำหนดความต้องการและแผนการปรับแต่ง
ผลการประเมิน <input type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน					
ชื่อผู้ประเมิน / ทีมผู้ประเมิน					
*ผลการประเมินจะผ่านหรือไม่ขึ้นอยู่กับการศึกษาของทีมผู้ประเมิน					

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รายการตรวจสอบเกณฑ์การให้ซ้ำสำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป (Reusable Criteria Checklists for COTS Integration Process)

1. ประเมินข้อกำหนดความต้องการ (Evaluate Requirements Specification)

รหัสเอกสาร : [C-I-n]	รายการตรวจสอบเกณฑ์การให้ซ้ำ (ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ)	หน้าที่ 1/1
----------------------	---	-------------

ชื่อกระบวนการ (Process Name) : กระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป					
ชื่อรายการตรวจสอบ (Checklist Name) : เกณฑ์การให้ซ้ำ (Reusable Criteria)					
คำอธิบาย (Description) : รายการตรวจสอบสำหรับเกณฑ์การให้ซ้ำ ประกอบด้วย 5 ส่วน ดังนี้ ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ ส่วนที่ 4 ประเมินการบูรณาการระบบและซอฟต์แวร์ และส่วนที่ 5 ประเมินการวางแผนการติดตั้ง					
ส่วนที่ 1 - ประเมินข้อกำหนดความต้องการ					
เกณฑ์การให้ซ้ำที่อยู่ในส่วนนี้มีความสำคัญต่อความสำเร็จในการบูรณาการซอฟต์แวร์สำเร็จรูปเพราะเป็นส่วนที่ประเมินข้อกำหนดความต้องการซอฟต์แวร์ ซึ่งเป็นเป้าหมายหลักของการบูรณาการซอฟต์แวร์สำเร็จรูปให้พิจารณาตามรายการตรวจสอบจากเกณฑ์การให้ซ้ำโดยให้คะแนนตามความถูกต้องและความสำคัญจากมากไปหาน้อยตามลำดับ					
4	3	2	1	N/A	เกณฑ์การให้ซ้ำ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ข้อกำหนดความต้องการซอฟต์แวร์มีความชัดเจนและสามารถดำเนินการบูรณาการซอฟต์แวร์สำเร็จรูปเพื่อตอบสนองต่อความต้องการได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ซอฟต์แวร์สำเร็จรูปที่เป็นสินทรัพย์ขององค์กรนั้น มีความยืดหยุ่นต่อการบูรณาการเพื่อบรรลุเป้าหมายได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการบูรณาการที่ได้นั้น ได้ออกแบบตามกระบวนการวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ได้เป็นอย่างดี
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สินทรัพย์ขององค์กรที่สามารถนำมาใช้ซ้ำได้นั้น มีเพียงพอต่อการบูรณาการซอฟต์แวร์สำเร็จรูป ซึ่งสามารถลดการพัฒนาที่ซ้ำซ้อนได้เป็นอย่างดี
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการบูรณาการสามารถตอบสนองต่อแบบจำลองและกลยุทธ์ในการพัฒนาซอฟต์แวร์ที่ได้ รวมถึงการนำสินทรัพย์มาใช้ซ้ำได้เป็นอย่างดี
ผลการประเมิน <input type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน					
ชื่อผู้ประเมิน / ทีมผู้ประเมิน					
*ผลการประเมินจะผ่านหรือไม่ขึ้นอยู่กับพิจารณาของทีมผู้ประเมิน					

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

2. ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ (Evaluate Software Product Lines Architecture)

รหัสเอกสาร : [C-I-n]	รายการตรวจสอบเกณฑ์การใช้ซ้ำ (ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์)	หน้าที่ 1/1
----------------------	--	-------------

ชื่อกระบวนการ (Process Name) : กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป					
ชื่อรายการตรวจสอบ (Checklist Name) : เกณฑ์การใช้ซ้ำ (Reusable Criteria)					
คำอธิบาย (Description) : รายการตรวจสอบสำหรับเกณฑ์การใช้ซ้ำ ประกอบด้วย 5 ส่วน ดังนี้ ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ ส่วนที่ 4 ประเมินการบูรณาการระบบและซอฟต์แวร์ และส่วนที่ 5 ประเมินการวางแผนการติดตั้ง					
ส่วนที่ 2 - ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ เกณฑ์การใช้ซ้ำที่อยู่ในส่วนนี้เป็นการประเมินรายละเอียดความสมบูรณ์ของสถาปัตยกรรมซอฟต์แวร์ที่ใช้ในกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป ให้พิจารณาตามรายการตรวจสอบจากเกณฑ์การใช้ซ้ำโดยให้คะแนนตามความถูกต้องและความสำคัญจากมากไปหาน้อยตามลำดับ					
4	3	2	1	N/A	เกณฑ์การใช้ซ้ำ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ข้อมูลนำเข้าของกระบวนการที่ใช้ในการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์มีความสมบูรณ์
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สามารถออกแบบสถาปัตยกรรมเชิงโครงสร้างที่ตอบสนองต่อความต้องการได้ครอบคลุมเป็นอย่างดี
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สามารถออกแบบสถาปัตยกรรมเชิงพลวัตที่ตอบสนองต่อความต้องการได้ครอบคลุมเป็นอย่างดี
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	รายละเอียดสถาปัตยกรรมซอฟต์แวร์ที่ได้นั้น มีความสมบูรณ์และสอดคล้องกับสถาปัตยกรรมซอฟต์แวร์ที่ได้ออกแบบไว้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สถาปัตยกรรมและรายละเอียดสถาปัตยกรรมซอฟต์แวร์ที่ได้นั้นสามารถตอบสนองต่อข้อกำหนดความต้องการซอฟต์แวร์และแผนการบูรณาการได้เป็นอย่างดี
ผลการประเมิน <input type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน					
ชื่อผู้ประเมิน / ทีมผู้ประเมิน					
*ผลการประเมินจะผ่านหรือไม่ขึ้นอยู่กับผลการพิจารณาของทีมผู้ประเมิน					

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

3. ประเมินการพัฒนาส่วนประกอบ (Evaluate Implementation Component)

รหัสเอกสาร : [C-I-n]	รายการตรวจสอบเกณฑ์การใช้ซ้ำ (ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ)	หน้าที่ 1/1
----------------------	--	-------------

ชื่อกระบวนการ (Process Name) : กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป					
ชื่อรายการตรวจสอบ (Checklist Name) : เกณฑ์การใช้ซ้ำ (Reusable Criteria)					
คำอธิบาย (Description) : รายการตรวจสอบสำหรับเกณฑ์การใช้ซ้ำ ประกอบด้วย 5 ส่วน ดังนี้ ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ ส่วนที่ 4 ประเมินการบูรณาการระบบและซอฟต์แวร์ และส่วนที่ 5 ประเมินการวางแผนการติดตั้ง					
ส่วนที่ 3 - ประเมินการพัฒนาส่วนประกอบ					
เกณฑ์การใช้ซ้ำที่อยู่ในส่วนนี้เป็นการประเมินการพัฒนาส่วนประกอบของการบูรณาการซอฟต์แวร์สำเร็จรูปซึ่งเป็นส่วนของการพัฒนารหัสคำสั่งเพิ่มเติม (Custom code) ซึ่งนำมาบูรณาการกับสินทรัพย์ที่สามารถนำมาใช้ซ้ำเพื่อให้สามารถครอบคลุมความต้องการซอฟต์แวร์และซอฟต์แวร์สำเร็จรูปให้พิจารณาตามรายการตรวจสอบจากเกณฑ์การใช้ซ้ำโดยให้คะแนนตามความถูกต้องและความสำคัญจากมากไปหาน้อยตามลำดับ					
4	3	2	1	N/A	เกณฑ์การใช้ซ้ำ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	รหัสคำสั่งเพิ่มเติมที่ได้จากกระบวนการพัฒนาส่วนประกอบนั้นสอดคล้องกับข้อกำหนดความต้องการซอฟต์แวร์และแผนการบูรณาการ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ทีมผู้พัฒนาสามารถดำเนินการพัฒนารหัสคำสั่งเพิ่มเติมตามลำดับความสำคัญของคุณลักษณะซึ่งเป็นวิธีการของสายผลิตภัณฑ์ซอฟต์แวร์
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	รหัสคำสั่งเพิ่มเติมมีความสมบูรณ์และพร้อมสำหรับการบูรณาการได้โดยไม่มีปัญหาเกิดขึ้น
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	รหัสคำสั่งเพิ่มเติมที่สามารถตอบสนองต่อข้อกำหนดความต้องการซอฟต์แวร์ซึ่งอยู่ในเกณฑ์ที่ยอมรับได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	รหัสคำสั่งเพิ่มเติมที่พัฒนานั้นได้ออกแบบและพัฒนาตามสถาปัตยกรรมเชิงโครงสร้าง สถาปัตยกรรมเชิงพลวัตและรายละเอียดสถาปัตยกรรมที่ได้ออกแบบไว้
ผลการประเมิน <input type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน					
ชื่อผู้ประเมิน / ทีมผู้ประเมิน					
*ผลการประเมินจะผ่านหรือไม่ขึ้นอยู่กับพิจารณาของทีมผู้ประเมิน					

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

.....

ชื่อเพิ่ม.นามสกุล	วันที่พิมพ์ [วัน / เดือน / ปี]
-------------------	--------------------------------

4. ประเมินการบูรณาการระบบและซอฟต์แวร์ (Evaluate Software and System Integration)

รหัสเอกสาร : [C-I-n]	รายการตรวจสอบเกณฑ์การใช้ซ้ำ (ส่วนที่ 4 ประเมินการบูรณาการระบบและซอฟต์แวร์)	หน้าที่ 1/1
----------------------	---	-------------

ชื่อกระบวนการ (Process Name) : กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป					
ชื่อรายการตรวจสอบ (Checklist Name) : เกณฑ์การใช้ซ้ำ (Reusable Criteria)					
คำอธิบาย (Description) : รายการตรวจสอบสำหรับเกณฑ์การใช้ซ้ำ ประกอบด้วย 5 ส่วน ดังนี้ ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ ส่วนที่ 4 ประเมินการบูรณาการระบบและซอฟต์แวร์ และส่วนที่ 5 ประเมินการวางแผนการติดตั้ง					
ส่วนที่ 4 - ประเมินการบูรณาการระบบและซอฟต์แวร์					
เกณฑ์การใช้ซ้ำที่อยู่ในส่วนนี้เป็นการประเมินการบูรณาการผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากกระบวนการพัฒนาส่วนประกอบซึ่งหมายถึง รหัสคำสั่งเพิ่มเติม สิ้นทรัพย์ที่สามารถนำมาใช้ซ้ำ ซอฟต์แวร์สำเร็จรูปและระบบหรือสภาพแวดล้อมที่มีอยู่ขององค์กรโดยให้พิจารณาตามรายการตรวจสอบจากเกณฑ์การใช้ซ้ำโดยให้คะแนนตามความถูกต้องและความสำคัญจากมากไปหาน้อยตามลำดับ					
4	3	2	1	N/A	เกณฑ์การใช้ซ้ำ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ผลิตภัณฑ์ซอฟต์แวร์ที่ได้สามารถบูรณาการและทำงานร่วมกับระบบหรือสภาพแวดล้อมที่มีอยู่เดิมขององค์กรได้เป็นอย่างดี
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ไม่มีความต้องการเพิ่มเติมสำหรับระบบและสภาพแวดล้อมที่มีอยู่ในการติดตั้งซอฟต์แวร์อื่นที่ช่วยให้ผลิตภัณฑ์ซอฟต์แวร์สามารถทำงานร่วมกับระบบที่มีอยู่เดิมได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สามารถบูรณาการฐานข้อมูลที่มีอยู่เดิมของระบบกับฐานข้อมูลของผลิตภัณฑ์ซอฟต์แวร์ให้สามารถทำงานร่วมกันได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สามารถบูรณาการกระบวนการที่มีอยู่เดิมกับการใช้งานผลิตภัณฑ์ซอฟต์แวร์ที่สามารถสนับสนุนกระบวนการนั้นๆ ได้เป็นอย่างดี
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	ไม่มีความต้องการที่จะแปลงผ่านข้อมูลของผลิตภัณฑ์ซอฟต์แวร์ให้สามารถทำงานร่วมกับระบบหรือสภาพแวดล้อมที่มีอยู่
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	สิ้นทรัพย์ที่สามารถนำมาใช้ซ้ำ ซอฟต์แวร์สำเร็จรูปและรหัสคำสั่งเพิ่มเติมสามารถนำมาบูรณาการได้ตามแผนการบูรณาการและครอบคลุมข้อกำหนดความต้องการซอฟต์แวร์และอยู่ในเกณฑ์ที่ยอมรับได้
ผลการประเมิน <input type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน					
ชื่อผู้ประเมิน / ทีมผู้ประเมิน					
*ผลการประเมินจะผ่านหรือไม่ขึ้นอยู่กับพิจารณาของทีมผู้ประเมิน					

รายการที่ต้องปรับปรุงแก้ไข :

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

5. ประเมินการวางแผนการติดตั้ง (Evaluate Installation Plan)

รหัสเอกสาร : [C-I-n]	รายการตรวจสอบเกณฑ์การใช้ซ้ำ (ส่วนที่ 5 ประเมินการวางแผนการติดตั้ง)	หน้าที่ 1/1
----------------------	---	-------------

ชื่อกระบวนการ (Process Name) : กระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป					
ชื่อรายการตรวจสอบ (Checklist Name) : เกณฑ์การใช้ซ้ำ (Reusable Criteria)					
คำอธิบาย (Description) : รายการตรวจสอบสำหรับเกณฑ์การใช้ซ้ำ ประกอบด้วย 5 ส่วน ดังนี้ ส่วนที่ 1 ประเมินข้อกำหนดความต้องการ ส่วนที่ 2 ประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ส่วนที่ 3 ประเมินการพัฒนาส่วนประกอบ ส่วนที่ 4 ประเมินการบูรณาการระบบและซอฟต์แวร์ และส่วนที่ 5 ประเมินการวางแผนการติดตั้ง					
ส่วนที่ 5 - ประเมินการวางแผนการติดตั้ง					
เกณฑ์การใช้ซ้ำที่อยู่ในส่วนนี้เป็นการประเมินการวางแผนการติดตั้งผลิตภัณฑ์ซอฟต์แวร์ที่ได้จากกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป ให้พิจารณาตามรายการตรวจสอบจากเกณฑ์การใช้ซ้ำโดยให้คะแนนตามความถูกต้องและความสำคัญจากมากไปหาน้อยตามลำดับ					
4	3	2	1	N/A	เกณฑ์การใช้ซ้ำ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการติดตั้งที่ได้นั้นมีความครอบคลุมกระบวนการและสภาพแวดล้อมที่จำเป็นสำหรับการติดตั้งซอฟต์แวร์
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการติดตั้งที่ได้นั้นมีส่วนที่จัดการแก้ไขปัญหาในกรณีที่ไม่สามารถติดตั้งผลิตภัณฑ์ซอฟต์แวร์ได้
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการติดตั้งที่ได้นั้นมีส่วนที่อธิบายกระบวนการสำหรับการติดตั้งซอฟต์แวร์โดยละเอียด
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	กระบวนการที่ได้จากการวางแผนการติดตั้งนั้นสามารถดำเนินการได้อย่างเป็นระบบ
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	แผนการติดตั้งที่ได้นั้นสอดคล้องต่อข้อกำหนดความต้องการและแผนการบูรณาการ
ผลการประเมิน <input type="checkbox"/> ผ่าน <input type="checkbox"/> ไม่ผ่าน					
ชื่อผู้ประเมิน / ทีมผู้ประเมิน					
*ผลการประเมินจะผ่านหรือไม่ขึ้นอยู่กับผลการพิจารณาของทีมผู้ประเมิน					

รายการที่ต้องปรับปรุงแก้ไข :

.....

.....

.....

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [G-T-n]	แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การปรับแต่งซอฟต์แวร์สำเร็จรูป	หน้าที่ 1/3
----------------------	--	-------------

แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การปรับแต่งซอฟต์แวร์สำเร็จรูป

แนวทางในการปรับปรุงกระบวนการสำหรับการพัฒนาการปรับแต่งซอฟต์แวร์สำเร็จรูปในแต่ละขั้นตอนของกระบวนการ ควรประกอบด้วยกิจกรรมสำหรับแต่ละขั้นตอนดังต่อไปนี้

วางแผนโครงการการปรับแต่งซอฟต์แวร์สำเร็จรูป

1. การวิเคราะห์ข้อมูลนำเข้าของกระบวนการเพื่อทำการหาแบบจำลองกระบวนการ กลยุทธ์ในการพัฒนา แผนการปรับแต่ง และสินทรัพย์ที่สามารถนำมาใช้ซ้ำนั้นจะต้องอยู่บนฐานของวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ซึ่งเน้นที่กระบวนการวิศวกรรมแอปพลิเคชัน คือ การพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์ และเมื่อได้ผลลัพธ์ของกระบวนการแล้วให้พิจารณาถึงผลลัพธ์ที่ได้จากกระบวนการว่ามีความสมบูรณ์หรือไม่ รวมถึงการบันทึกผลข้อบกพร่องที่เกิดขึ้นในระหว่างการดำเนินการ ซึ่งเป็นสิ่งที่ต้องนำมาปรับปรุงแก้ไข

2. สินทรัพย์ที่สามารถนำมาใช้ซ้ำนั้นจะต้องมีจำนวนที่เพียงพอต่อการปรับแต่งซอฟต์แวร์และมีความยืดหยุ่นในกรณีที่เกิดปัญหาขึ้น

3. จากผลการประเมินข้อกำหนดความต้องการที่ได้ซึ่งได้จากทีมผู้ประเมินนั้น จะมีส่วนที่บ่งชี้ถึงรายการที่ต้องทำการปรับปรุงแก้ไข ซึ่งจากส่วนนี้ผู้จัดการโครงการสามารถนำเอารายการที่ต้องทำการปรับปรุงแก้ไขเหล่านั้น มาดำเนินการปรับปรุงแก้ไขให้ได้มาซึ่งผลลัพธ์ของกระบวนการที่สมบูรณ์ขึ้นได้ เพื่อหลีกเลี่ยงความล้มเหลวที่อาจเกิดขึ้นกับโครงการได้

ออกแบบสถาปัตยกรรมการปรับแต่งซอฟต์แวร์สำเร็จรูป

1. การออกแบบสถาปัตยกรรมซอฟต์แวร์สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปจะต้องอยู่บนพื้นฐานของข้อกำหนดความต้องการ แผนการปรับแต่ง วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ และสินทรัพย์ที่สามารถนำมาใช้ซ้ำ

2. โครงสร้างของสถาปัตยกรรมซอฟต์แวร์จะต้องประกอบด้วย สถาปัตยกรรมเชิงโครงสร้าง (Structural architecture) และสถาปัตยกรรมเชิงพลวัต (Dynamical architecture) และต้องมีส่วนของคุณลักษณะของสินทรัพย์ที่สามารถนำมาใช้ซ้ำ ที่ใช้ในการพัฒนาซอฟต์แวร์โดยแบ่งเป็น คุณสมบัติหลัก (Core feature) คุณสมบัติตัวเลือก (Optional feature) และคุณสมบัติทางเลือก (Alternative feature)

[ชื่อแฟ้ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
--------------------	--------------------------------

รหัสเอกสาร : [G-T-n]	แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การปรับแต่งซอฟต์แวร์สำเร็จรูป	หน้าที่ 2/3
----------------------	--	-------------

3. รายละเอียดสถาปัตยกรรมซอฟต์แวร์จะต้องสอดคล้องและสามารถตอบสนองต่อสถาปัตยกรรมซอฟต์แวร์ที่ได้ และอยู่บนพื้นฐานของการใช้ซ้ำจากสินทรัพย์

4. จากผลการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ซึ่งได้จากทีมผู้ประเมินนั้น จะมีส่วนที่บ่งชี้ถึงรายการที่ต้องทำการปรับปรุงแก้ไข ซึ่งจากส่วนนี้ผู้จัดการโครงการสามารถนำเอารายการที่ต้องทำการปรับปรุงแก้ไขเหล่านั้น มาดำเนินการปรับปรุงแก้ไขให้ได้มาซึ่งผลลัพธ์ของกระบวนการที่สมบูรณ์ขึ้นได้เพื่อหลีกเลี่ยงความล้มเหลวที่อาจเกิดขึ้นกับโครงการได้

ปรับแต่งซอฟต์แวร์สำเร็จรูป

1. การพัฒนาส่วนประกอบ จะต้องดำเนินการพัฒนาการปรับแต่งซอฟต์แวร์สำเร็จรูป ตามลำดับความสำคัญของคุณลักษณะ กล่าวคือ คุณลักษณะหลัก คุณลักษณะตัวเลือก และคุณลักษณะทางเลือกเพื่อการพัฒนาปรับแต่งซอฟต์แวร์จากสินทรัพย์อย่างมีระบบ

2. เมื่อได้ทำการพัฒนาส่วนประกอบแล้ว ผลลัพธ์ที่ได้ซึ่งเป็นผลิตภัณฑ์ซอฟต์แวร์ที่ผ่านการปรับแต่งแล้วจะถูกประเมินความสามารถเทียบกับข้อกำหนดความต้องการ ซึ่งในกรณีที่มีข้อผิดพลาดในส่วนของสินทรัพย์เกิดขึ้นซึ่งอาจทำให้โครงการต้องล้มเหลวได้นั้น ให้ทีมผู้พัฒนาดำเนินการแก้ไขปรับปรุงรายการนั้นๆ โดยนำเอาสินทรัพย์อื่นที่สามารถสนองต่อความต้องการในระดับที่พึงพอใจได้มาใช้

3. ในกรณีที่ได้ผลิตภัณฑ์ซอฟต์แวร์ที่ไม่ครอบคลุมความต้องการ ให้พิจารณาถึงสาเหตุว่าเกิดข้อผิดพลาดที่กระบวนการใดและเหตุใด เพื่อการดำเนินการที่รอบคอบและละเอียดมากขึ้นสำหรับการพัฒนาการปรับแต่งซอฟต์แวร์สำเร็จรูปในอนาคต

4. การทดสอบสายผลิตภัณฑ์ซอฟต์แวร์ โดยแบ่งเป็น การทดสอบการบูรณาการ (Integration test) และการทดสอบเชิงหน้าที่ (Functional test) ให้พิจารณาว่าผลิตภัณฑ์ซอฟต์แวร์ที่ผ่านการทดสอบแล้วนั้นอยู่ในเกณฑ์ที่ยอมรับได้หรือไม่ และมีข้อบกพร่องใดเกิดขึ้นบ้างซึ่งเป็นที่ต้องนำไปปรับปรุงแก้ไขต่อไป

5. การจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป ควรระบุสารสนเทศที่เกี่ยวกับผลิตภัณฑ์ซอฟต์แวร์ให้ครบและละเอียดเพื่อความสะดวกต่อการนำมาใช้ในอนาคต

6. จากผลการประเมินที่เกิดขึ้นในกระบวนการการปรับแต่งซึ่งได้จากทีมผู้ประเมินนั้น จะมีส่วนที่บ่งชี้ถึงรายการที่ต้องทำการปรับปรุงแก้ไข ซึ่งจากส่วนนี้ทีมผู้พัฒนาสามารถนำเอารายการที่ต้องทำการปรับปรุงแก้ไขเหล่านั้น มาดำเนินการปรับปรุงแก้ไขให้ได้มาซึ่งผลลัพธ์ของกระบวนการที่สมบูรณ์ขึ้นได้ เพื่อหลีกเลี่ยงความล้มเหลวที่อาจเกิดขึ้นกับโครงการได้

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [G-T-n]	แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การปรับแต่งซอฟต์แวร์สำเร็จรูป	หน้าที่ 3/3
----------------------	--	-------------

ติดตั้งซอฟต์แวร์

1. แผนการติดตั้งผลิตภัณฑ์ซอฟต์แวร์ที่ได้นั้นจะต้องครอบคลุมในกรณีที่สภาพแวดล้อมขององค์กรนั้นๆ ไม่สามารถดำเนินการติดตั้งได้ โดยมีทางเลือกสำหรับการดำเนินการติดตั้งที่หลากหลายและควรมีเอกสารอธิบายในรายละเอียดสำหรับการติดตั้งอีกด้วย

2. จากผลการประเมินแผนการติดตั้งที่ได้จากทีมผู้ประเมินนั้น จะมีส่วนที่บ่งชี้ถึงรายการที่ต้องทำการปรับปรุงแก้ไข ซึ่งจากส่วนนี้ทีมผู้พัฒนาสามารถนำเอารายการที่ต้องทำการปรับปรุงแก้ไขเหล่านั้นมาดำเนินการปรับปรุงแก้ไขให้ได้มาซึ่งผลลัพธ์ของกระบวนการที่สมบูรณ์ขึ้นได้ เพื่อหลีกเลี่ยงความล้มเหลวที่อาจเกิดขึ้นกับโครงการได้

สนับสนุนการยอมรับซอฟต์แวร์

1. กระบวนการสนับสนุนการยอมรับซอฟต์แวร์นั้น จะต้องมีส่วนของเอกสารการใช้งานสำหรับผู้ใช้งาน (User Documentation) โดยมีคำอธิบายที่ผู้ใช้งานสามารถทำความเข้าใจได้ง่าย ซึ่งจะช่วยให้ผู้ใช้งานสามารถใช้งานผลิตภัณฑ์ซอฟต์แวร์ได้อย่างมีประสิทธิภาพ

2. ทีมผู้พัฒนาควรมีการฝึกอบรมการใช้งานผลิตภัณฑ์ซอฟต์แวร์ให้กับผู้ใช้

วัดผลกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป

1. การวัดผลกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปเป็นกลุ่มกระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป ที่สามารถดำเนินการไปพร้อมๆ กับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ของกลุ่มกระบวนการหลักได้

2. การวัดผลของกระบวนการนั้นจะต้องมีจุดมุ่งหมายของการวัดผลเสียก่อน ยกตัวอย่างเช่น เพื่อลดเวลาในการพัฒนาและส่งมอบผลิตภัณฑ์ซอฟต์แวร์ เพื่อเพิ่มประสิทธิภาพของซอฟต์แวร์ เพื่อลดต้นทุนที่ใช้ในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ เป็นต้น โดยเพื่อนำเอาผลที่ได้จากการวัดผลนั้นนำไปปรับปรุงกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปนั้นๆ ให้มีผลลัพธ์ที่ดีขึ้นในอนาคต

3. ทีมผู้มีส่วนร่วมในการพัฒนาซอฟต์แวร์ อาจจะต้องทำการประเมินความสามารถในการใช้งานของซอฟต์แวร์ เช่น วิธีการสัมภาษณ์ผู้ใช้งาน เป็นต้น โดยมีจุดประสงค์เพื่อได้รับผลตอบรับจากการใช้งานจริงจากผู้ใช้งานซึ่งทำให้ได้มาซึ่งข้อบกพร่องที่ผู้พัฒนาไม่สามารถเห็นถึงข้อบกพร่องนั้นได้ เช่น ส่วนต่อประสานผู้ใช้ เป็นต้น

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [G-I-n]	แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การบูรณาการซอฟต์แวร์สำเร็จรูป	หน้าที่ 1/3
----------------------	--	-------------

แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การบูรณาการซอฟต์แวร์สำเร็จรูป

แนวทางในการปรับปรุงกระบวนการสำหรับการพัฒนาการบูรณาการซอฟต์แวร์สำเร็จรูปในแต่ละขั้นตอนของกระบวนการ ควรประกอบด้วยกิจกรรมสำหรับแต่ละขั้นตอนดังต่อไปนี้

วางแผนโครงการการบูรณาการซอฟต์แวร์สำเร็จรูป

1. การวิเคราะห์ข้อมูลนำเข้าของกระบวนการเพื่อทำการหาแบบจำลองกระบวนการ กลยุทธ์ในการพัฒนา แผนการบูรณาการ และสินทรัพย์ที่สามารถนำมาใช้ซ้ำนั้นจะต้องอยู่บนฐานของวิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์ ซึ่งเน้นที่กระบวนการวิศวกรรมแอปพลิเคชัน คือ การพัฒนาผลิตภัณฑ์ซอฟต์แวร์จากสินทรัพย์และเมื่อได้ผลลัพธ์ของกระบวนการแล้วให้พิจารณาถึงผลลัพธ์ที่ได้จากกระบวนการว่ามีความสมบูรณ์หรือไม่ รวมถึงการบันทึกผลข้อบกพร่องที่เกิดขึ้นในระหว่างการดำเนินการ ซึ่งเป็นสิ่งที่ต้องนำมาปรับปรุงแก้ไข

2. สินทรัพย์ที่สามารถนำมาใช้ซ้ำนั้นจะต้องมีจำนวนที่เพียงพอต่อการบูรณาการซอฟต์แวร์และมีความยืดหยุ่นในกรณีที่เกิดปัญหาขึ้น

3. จากผลการประเมินข้อกำหนดความต้องการที่ได้ซึ่งได้จากทีมผู้ประเมินนั้น จะมีส่วนที่บ่งชี้ถึงรายการที่ต้องทำการปรับปรุงแก้ไข ซึ่งจากส่วนนี้ผู้จัดการโครงการสามารถนำเอารายการที่ต้องทำการปรับปรุงแก้ไขเหล่านั้น มาดำเนินการปรับปรุงแก้ไขให้ได้มาซึ่งผลลัพธ์ของกระบวนการที่สมบูรณ์ขึ้นได้ เพื่อหลีกเลี่ยงความล้มเหลวที่อาจเกิดขึ้นกับโครงการได้

ออกแบบสถาปัตยกรรมการบูรณาการซอฟต์แวร์สำเร็จรูป

1. การออกแบบสถาปัตยกรรมซอฟต์แวร์สำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูปจะต้องอยู่บนพื้นฐานของข้อกำหนดความต้องการ แผนการบูรณาการ วิศวกรรมสายผลิตภัณฑ์ซอฟต์แวร์และสินทรัพย์ที่สามารถนำมาใช้ซ้ำ

2. โครงสร้างของสถาปัตยกรรมซอฟต์แวร์จะต้องประกอบด้วย สถาปัตยกรรมเชิงโครงสร้าง (Structural architecture) และสถาปัตยกรรมเชิงพลวัต (Dynamical architecture) และต้องมีส่วนของคุณลักษณะของสินทรัพย์ที่สามารถนำมาใช้ซ้ำ ที่ใช้ในการพัฒนาซอฟต์แวร์โดยแบ่งเป็น คุณสมบัติหลัก (Core feature) คุณสมบัติตัวเลือก (Optional feature) และคุณสมบัติทางเลือก (Alternative feature) และนอกจากนั้นสำหรับการบูรณาการซอฟต์แวร์สำเร็จรูปนั้นจะต้องใช้ทอพอโลยีการเชื่อมต่อ เช่น ArchWare framework เป็นต้น ซึ่งส่วนของการออกแบบสถาปัตยกรรมนั้นจะต้องมี

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

รหัสเอกสาร : [G-I-n]	แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การบูรณาการซอฟต์แวร์สำเร็จรูป	หน้าที่ 2/3
----------------------	--	-------------

3. รายละเอียดสถาปัตยกรรมซอฟต์แวร์จะต้องสอดคล้องและสามารถตอบสนองต่อสถาปัตยกรรมซอฟต์แวร์ที่ได้ และอยู่บนพื้นฐานของการใช้ซ้ำจากสินทรัพย์

4. จากผลการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ซึ่งได้จากทีมผู้ประเมินนั้นจะมีส่วนที่บ่งชี้ถึงรายการที่ต้องทำการปรับปรุงแก้ไข ซึ่งจากส่วนนี้ผู้จัดการโครงการสามารถนำเอารายการที่ต้องทำการปรับปรุงแก้ไขเหล่านั้น มาดำเนินการปรับปรุงแก้ไขให้ได้มาซึ่งผลลัพธ์ของกระบวนการที่สมบูรณ์ขึ้นได้เพื่อหลีกเลี่ยงความล้มเหลวที่อาจเกิดขึ้นกับโครงการได้

บูรณาการซอฟต์แวร์สำเร็จรูป

1. สำหรับการพัฒนาร่วมประกอบในกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูปนั้นเป็นการออกแบบและพัฒนารหัสคำสั่งเพิ่มเติม (Custom code) โดยอาจใช้เป็น คุณลักษณะตัวเลือกหรือคุณลักษณะทางเลือกก็ได้ โดยมีจุดประสงค์เพื่อเติมเต็มความสามารถของซอฟต์แวร์สำเร็จรูปให้สามารถตอบสนองต่อข้อกำหนดความต้องการซอฟต์แวร์ได้

2. สำหรับการบูรณาการซอฟต์แวร์สำเร็จรูป ซึ่งเกิดจากการบูรณาการระหว่าง ซอฟต์แวร์สำเร็จรูปสินทรัพย์ที่สามารถนำมาใช้ซ้ำ และรหัสคำสั่งเพิ่มเติม การบูรณาการนั้นจะต้องดำเนินการโดยบูรณาการจากสินทรัพย์ที่มีความสำคัญจากมากไปหาน้อยตามลำดับ กล่าวคือ สินทรัพย์ที่มีคุณสมบัติดังต่อไปนี้ คุณสมบัติหลัก คุณสมบัติตัวเลือกและคุณสมบัติทางเลือก

3. การทดสอบสายผลิตภัณฑ์ซอฟต์แวร์ โดยแบ่งเป็น การทดสอบการบูรณาการ (Integration test) และการทดสอบเชิงหน้าที่ (Functional test) ให้พิจารณาว่าผลิตภัณฑ์ซอฟต์แวร์ที่ผ่านการทดสอบแล้วนั้นอยู่ในเกณฑ์ที่ยอมรับได้หรือไม่ และมีข้อบกพร่องใดเกิดขึ้นบ้างซึ่งเป็นที่ต้องนำไปปรับปรุงแก้ไขต่อไป

4. การจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป ควรระบุสารสนเทศที่เกี่ยวกับผลิตภัณฑ์ซอฟต์แวร์ให้ครบและละเอียดเพื่อความสะดวกต่อการนำมาใช้ในอนาคต

5. จากผลการประเมินที่เกิดขึ้นในกระบวนการบูรณาการซึ่งได้จากทีมผู้ประเมินนั้น จะมีส่วนที่บ่งชี้ถึงรายการที่ต้องทำการปรับปรุงแก้ไข ซึ่งจากส่วนนี้ทีมผู้พัฒนาสามารถนำเอารายการที่ต้องทำการปรับปรุงแก้ไขเหล่านั้น มาดำเนินการปรับปรุงแก้ไขให้ได้มาซึ่งผลลัพธ์ของกระบวนการที่สมบูรณ์ขึ้นได้ เพื่อหลีกเลี่ยงความล้มเหลวที่อาจเกิดขึ้นกับโครงการได้

ชื่อเพิ่ม.นามสกุล	วันที่พิมพ์ [วัน / เดือน / ปี]
-------------------	--------------------------------

รหัสเอกสาร : [G-I-n]	แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การบูรณาการซอฟต์แวร์สำเร็จรูป	หน้าที่ 3/3
----------------------	--	-------------

ติดตั้งซอฟต์แวร์

1. แผนการติดตั้งผลิตภัณฑ์ซอฟต์แวร์ที่ได้นั้นจะต้องครอบคลุมในกรณีที่สภาพแวดล้อมขององค์กรนั้นๆ ไม่สามารถดำเนินการติดตั้งได้ โดยมีทางเลือกสำหรับการดำเนินการติดตั้งที่หลากหลายและควรมีเอกสารอธิบายในรายละเอียดสำหรับการติดตั้งอีกด้วย

2. จากผลการประเมินแผนการติดตั้งที่ได้จากทีมผู้ประเมินนั้น จะมีส่วนที่บ่งชี้ถึงรายการที่ต้องทำการปรับปรุงแก้ไข ซึ่งจากส่วนนี้ที่ผู้พัฒนาสามารถนำเอารายการที่ต้องทำการปรับปรุงแก้ไขเหล่านั้นมาดำเนินการปรับปรุงแก้ไขให้ได้มาซึ่งผลลัพธ์ของกระบวนการที่สมบูรณ์ขึ้นได้ เพื่อหลีกเลี่ยงความล้มเหลวที่อาจเกิดขึ้นกับโครงการได้

สนับสนุนการยอมรับซอฟต์แวร์

1. กระบวนการสนับสนุนการยอมรับซอฟต์แวร์นั้น จะต้องมีส่วนของเอกสารการใช้งานสำหรับผู้ใช้งาน (User Documentation) โดยมีคำอธิบายที่ผู้ใช้งานสามารถทำความเข้าใจได้ง่าย ซึ่งจะช่วยให้ผู้ใช้งานผลิตภัณฑ์ซอฟต์แวร์ได้อย่างมีประสิทธิภาพ

2. ทีมผู้พัฒนาควรมีการฝึกอบรมการใช้งานผลิตภัณฑ์ซอฟต์แวร์ให้กับผู้ใช้

วัดผลกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป

1. การวัดผลกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูปเป็นกลุ่มกระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูปที่สามารถดำเนินการไปพร้อมๆ กับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ของกลุ่มกระบวนการหลักได้

2. การวัดผลของกระบวนการนั้นจะต้องมีจุดมุ่งหมายของการวัดผลเสียก่อน ยกตัวอย่างเช่น เพื่อลดเวลาในการพัฒนาและส่งมอบผลิตภัณฑ์ซอฟต์แวร์ เพื่อเพิ่มประสิทธิภาพของซอฟต์แวร์ เพื่อลดต้นทุนที่ใช้ในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ เป็นต้น โดยเพื่อนำเอาผลที่ได้จากการวัดผลนั้นนำไปปรับปรุงกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปนั้นๆ ให้มีผลลัพธ์ที่ดีขึ้นในอนาคต

3. ทีมผู้มีส่วนร่วมในการพัฒนาซอฟต์แวร์อาจจะต้องทำการประเมินความสามารถในการใช้งานของซอฟต์แวร์ เช่น วิธีการสัมภาษณ์ผู้ใช้งาน เป็นต้น โดยมีจุดประสงค์เพื่อได้รับผลตอบรับจากการใช้งานจริงจากผู้ใช้งานที่ได้มาซึ่งข้อบกพร่องที่ผู้พัฒนาไม่สามารถเห็นถึงข้อบกพร่องนั้นได้ เช่น ส่วนต่อประสานผู้ใช้ เป็นต้น

[ชื่อแถม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
-------------------	--------------------------------

รหัสเอกสาร : [G-M-n]	แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การบำรุงรักษาซอฟต์แวร์สำเร็จรูป	หน้าที่ 1/2
----------------------	--	-------------

แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การบำรุงรักษาซอฟต์แวร์สำเร็จรูป

แนวทางในการปรับปรุงกระบวนการสำหรับการพัฒนาการบำรุงรักษาซอฟต์แวร์สำเร็จรูปในแต่ละขั้นตอนของกระบวนการ ควรประกอบด้วยกิจกรรมสำหรับแต่ละขั้นตอนดังต่อไปนี้

วางแผนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

1. รายงานปัญหา/คำร้องขอการดัดแปลง ที่ได้นั้นจะต้องถูกพิจารณาถึงความเป็นไปได้ว่าทีมผู้พัฒนาสามารถดำเนินการบำรุงรักษาตามรายงานปัญหา/คำร้องขอการดัดแปลงนั้นได้หรือไม่เสียก่อน เพื่อหลีกเลี่ยงความล้มเหลวที่อาจเกิดขึ้นกับโครงการได้ เช่น ในกรณีที่ไม่มีบุคลากรที่มีความรู้เฉพาะทาง เทคโนโลยีที่สามารถดำเนินการ หรือสินทรัพย์ที่สามารถใช้ซ้ำได้ นั้นมีไม่เพียงพอต่อการบำรุงรักษา เป็นต้น หลังจากนั้นให้หาวิธีการแก้ไขปัญหาเพื่อดำเนินการบำรุงรักษาต่อไป

2. แผนการบำรุงรักษาซึ่งประกอบด้วย แผนการบำรุงรักษาและกระบวนการบำรุงรักษา ต้องสามารถตอบสนองต่อรายงานปัญหา/คำร้องขอการดัดแปลง

วิเคราะห์ปัญหาและการดัดแปลง

1. พิจารณาสินทรัพย์ที่สามารถนำมาใช้ซ้ำว่ามีสินทรัพย์ที่สามารถนำมาใช้ในการบำรุงรักษาได้ตามแผนการบำรุงรักษาและกระบวนการบำรุงรักษาหรือไม่ โดยสินทรัพย์เหล่านั้นจะต้องมีความหลากหลายซึ่งมีสินทรัพย์ที่มีจำนวนที่เพียงพอและมีความยืดหยุ่นในกรณีที่เกิดปัญหาขึ้นได้

2. เมื่อได้สินทรัพย์ที่สามารถนำมาใช้ซ้ำได้ต้องดำเนินการวิเคราะห์ผลลัพธ์ของแต่ละสินทรัพย์นั้นเพื่อเลือกใช้สินทรัพย์ที่มีผลกระทบต่อการใช้งานมากที่สุดเพราะจะได้ลดความเพียรพยายามที่เกินจำเป็นต่อโครงการ

3. การกำหนดวิธีการบำรุงรักษาซึ่งประกอบด้วย ดัดแปลง ย้าย หรือ ยกเลิกข้อมูลนั้นสำหรับการดำเนินการบำรุงรักษาแต่ละครั้งนั้นสามารถทำได้หลายวิธีขึ้นอยู่กับความต้องการที่มีอยู่ในรายงานปัญหา/คำร้องขอการดัดแปลง

[ชื่อแฟ้ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
--------------------	--------------------------------

รหัสเอกสาร : [G-M-n]	แนวทางในการดำเนินการสำหรับปรับปรุงกระบวนการ การบำรุงรักษาซอฟต์แวร์สำเร็จรูป	หน้าที่ 2/2
----------------------	--	-------------

ดัดแปลง ย้าย และยกเลิกข้อมูล

1. เมื่อทำการดำเนินการดัดแปลง ย้าย หรือยกเลิกข้อมูลแล้วให้พิจารณาถึงความสมบูรณ์ของผลิตภัณฑ์ซอฟต์แวร์ว่าสามารถตอบสนองต่อรายงานปัญหา/คำร้องขอการดัดแปลงหรือไม่และระบุข้อบกพร่องที่เกิดขึ้นและรายการที่ต้องปรับปรุงแก้ไขซึ่งจะต้องดำเนินการปรับปรุงแก้ไขต่อไป

2. แจ้งผลการบำรุงรักษาผลิตภัณฑ์ซอฟต์แวร์แก่ผู้ที่มีส่วนได้ส่วนเสียสำหรับโครงการทั้งหมดให้รับทราบ

วัดผลกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป

1. การวัดผลกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูปเป็นกลุ่มกระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูปที่สามารถดำเนินการไปพร้อมๆ กับการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ของกลุ่มกระบวนการหลักได้

2. การวัดผลของกระบวนการนั้นจะต้องมีจุดมุ่งหมายของการวัดผลเสียก่อน ยกตัวอย่างเช่น เพื่อลดเวลาในการพัฒนาและส่งมอบผลิตภัณฑ์ซอฟต์แวร์ เพื่อเพิ่มประสิทธิภาพของซอฟต์แวร์ เพื่อลดต้นทุนที่ใช้ในการพัฒนาผลิตภัณฑ์ซอฟต์แวร์ เป็นต้น โดยเพื่อนำเอาผลที่ได้จากการวัดผลนั้นนำไปปรับปรุงกระบวนการพัฒนาซอฟต์แวร์สำเร็จรูปนั้นๆ ให้มีผลลัพธ์ที่ดีขึ้นในอนาคต

3. ทีมผู้มีส่วนร่วมในการพัฒนาซอฟต์แวร์อาจจะต้องการประเมินความสามารถในการใช้งานของซอฟต์แวร์ เช่น วิธีการสัมภาษณ์ผู้ใช้งาน เป็นต้น โดยมีจุดประสงค์เพื่อได้รับผลตอบรับจากการใช้งานจริงจากผู้ใช้ซึ่งทำให้ได้มาซึ่งข้อบกพร่องที่ผู้พัฒนาไม่สามารถเห็นถึงข้อบกพร่องนั้นได้ เช่น ส่วนต่อประสานผู้ใช้ เป็นต้น

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

[ชื่อเพิ่ม.นามสกุล]	วันที่พิมพ์ [วัน / เดือน / ปี]
---------------------	--------------------------------

ภาคผนวก ง

ความต้องการด้านหน้าที่ของเครื่องมือสนับสนุน

กระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน

ตารางความต้องการด้านหน้าที่ของเครื่องมือสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์นั้น มีทั้งหมด 51 ตาราง ซึ่งสามารถสรุปได้ดังตารางที่ ง.1-ง.51

ตารางที่ ง.1 ความต้องการด้านหน้าที่ของระบบ รหัส FR0101

รหัสความต้องการ : FR0101	ชื่อหน้าที่ : เพิ่มข้อมูลสารสนเทศอาร์ทิแฟก
คำอธิบายหน้าที่ :	เพิ่มเอกสารสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จที่ใช้สายผลิตภัณฑ์รูปเข้าสู่ระบบ
ข้อมูลนำเข้าระบบ :	1. ชื่อเอกสารสนับสนุน 2. ชนิดของเอกสารสนับสนุน 3. ไฟล์เอกสารสนับสนุน 4. ชื่อกระบวนการที่ใช้เอกสารสนับสนุน 5. คำอธิบายเอกสารสนับสนุน 6. สถานะของเอกสารสนับสนุน
ข้อมูลนำออก :	(ไม่มี) //บันทึกข้อมูลลงฐานข้อมูล
ผู้มีสิทธิ์ใช้งาน :	ผู้จัดการระบบ

ตารางที่ ง.2 ความต้องการด้านหน้าที่ของระบบ รหัส FR0102

รหัสความต้องการ : FR0102	ชื่อหน้าที่ : ปรับปรุงข้อมูลสารสนเทศอาร์ทิแฟก
คำอธิบายหน้าที่ :	ปรับปรุงข้อมูลเอกสารสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จที่ใช้สายผลิตภัณฑ์รูปเข้าสู่ระบบ
ข้อมูลนำเข้าระบบ :	1. ชื่อเอกสารสนับสนุน 2. ชนิดของเอกสารสนับสนุน 3. ไฟล์เอกสารสนับสนุน 4. ชื่อกระบวนการที่ใช้เอกสารสนับสนุน 5. คำอธิบายเอกสารสนับสนุน 6. สถานะของเอกสารสนับสนุน
ข้อมูลนำออก :	(ไม่มี) //บันทึกข้อมูลลงฐานข้อมูล
ผู้มีสิทธิ์ใช้งาน :	ผู้จัดการระบบ

ตารางที่ ง.3 ความต้องการด้านหน้าที่ของระบบ รหัส FR0103

รหัสความต้องการ : FR0103	ชื่อหน้าที่ : รายการผลผลิตงาน
คำอธิบายหน้าที่ : แสดงรายการและดาวน์โหลดผลผลิตงานซึ่งเป็นเอกสารผลลัพธ์ที่ได้จากการดำเนินการของผู้ดำเนินการในแต่ละบทบาท	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อผู้ดำเนินการ 2. ชื่อกลุ่มกระบวนการหลักของการพัฒนาซอฟต์แวร์ 3. ชื่อกลุ่มกระบวนการ 4. หน้าที่รับผิดชอบ 	
ข้อมูลนำออก : ไฟล์เอกสารผลผลิตงาน	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการระบบ	

ตารางที่ ง.4 ความต้องการด้านหน้าที่ของระบบ รหัส FR0104

รหัสความต้องการ : FR0104	ชื่อหน้าที่ : เพิ่มข้อมูลผู้ใช้งานระบบ
คำอธิบายหน้าที่ : เพิ่มข้อมูลผู้ใช้งานเข้าสู่ระบบ	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. คำนำหน้าชื่อ 2. ชื่อ 3. ชื่อสกุล 4. ที่อยู่ 5. เบอร์โทรศัพท์ 6. เบอร์โทรศัพท์เคลื่อนที่ 7. ไปรษณีย์อิเล็กทรอนิกส์ 8. ชื่อหน่วยงาน 9. ข้อมูลทักษะหรือความชำนาญ 10. รหัสผู้ใช้งาน 11. รหัสผ่าน 12. ประเภทผู้ใช้งาน 	
ข้อมูลนำออก : (ไม่มี)	//บันทึกข้อมูลลงฐานข้อมูล
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการระบบ	

ตารางที่ ง.5 ความต้องการด้านหน้าที่ของระบบ รหัส FR0105

รหัสความต้องการ : FR0105	ชื่อหน้าที่ : ปรับปรุงข้อมูลผู้ใช้งานระบบ
คำอธิบายหน้าที่ : ปรับปรุงข้อมูลผู้ใช้งานเข้าสู่ระบบ	
ข้อมูลนำเข้ระบบ : <ol style="list-style-type: none"> 1. คำนำหน้าชื่อ 2. ชื่อ 3. ชื่อสกุล 4. ที่อยู่ 5. เบอร์โทรศัพท์ 6. เบอร์โทรศัพท์เคลื่อนที่ 7. ไปรษณีย์อิเล็กทรอนิกส์ 8. ชื่อหน่วยงาน 9. ข้อมูลทักษะหรือความชำนาญ 10. รหัสผู้ใช้งาน 11. รหัสผ่าน 12. ประเภทผู้ใช้งาน 	
ข้อมูลนำออก : (ไม่มี)	//บันทึกข้อมูลลงฐานข้อมูล
ผู้มีสิทธิใช้งาน : ผู้จัดการระบบ	

ตารางที่ ง.6 ความต้องการด้านหน้าที่ของระบบ รหัส FR0106

รหัสความต้องการ : FR0106	ชื่อหน้าที่ : เพิ่มข้อมูลเกณฑ์การให้ซ้ำของรายการตรวจสอบ
คำอธิบายหน้าที่ : เพิ่มข้อมูลเกณฑ์การให้ซ้ำของรายการตรวจสอบที่ใช้สำหรับกระบวนการประเมิน	
ข้อมูลนำเข้ระบบ : <ol style="list-style-type: none"> 1. ชื่อกระบวนการ 2. คำอธิบายเกณฑ์การให้ซ้ำ 	
ข้อมูลนำออก : (ไม่มี)	//บันทึกข้อมูลลงฐานข้อมูล
ผู้มีสิทธิใช้งาน : ผู้จัดการระบบ	

ตารางที่ ง.7 ความต้องการด้านหน้าที่ของระบบ รหัส FR0107

รหัสความต้องการ : FR0107	ชื่อหน้าที่ : ปรับปรุงข้อมูลเกณฑ์การให้ซ้ำของรายการตรวจสอบ
คำอธิบายหน้าที่ : ปรับปรุงข้อมูลเกณฑ์การให้ซ้ำของรายการตรวจสอบที่ใช้สำหรับกระบวนการประเมิน	
ข้อมูลนำเข้าระบบ : 1. คำอธิบายเกณฑ์การให้ซ้ำ	
ข้อมูลนำออก : (ไม่มี)	//บันทึกข้อมูลลงฐานข้อมูล
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการระบบ	

ตารางที่ ง.8 ความต้องการด้านหน้าที่ของระบบ รหัส FR0108

รหัสความต้องการ : FR0108	ชื่อหน้าที่ : เพิ่มข้อมูลกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา
คำอธิบายหน้าที่ : เพิ่มข้อมูลกำหนดแบบจำลองและกลยุทธ์ในการพัฒนาสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.9 ความต้องการด้านหน้าที่ของระบบ รหัส FR0109

รหัสความต้องการ : FR0109	ชื่อหน้าที่ : เพิ่มข้อมูลกำหนดแผนการปรับแต่ง
คำอธิบายหน้าที่ : เพิ่มข้อมูลกำหนดแผนการปรับแต่งสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการกำหนดแผนการปรับแต่ง	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.10 ความต้องการด้านหน้าที่ของระบบ รหัส FR0110

รหัสความต้องการ : FR0110	ชื่อหน้าที่ : เพิ่มข้อมูลทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.11 ความต้องการด้านหน้าที่ของระบบ รหัส FR0111

รหัสความต้องการ : FR0111	ชื่อหน้าที่ : เพิ่มข้อมูลกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.12 ความต้องการด้านหน้าที่ของระบบ รหัส FR0112

รหัสความต้องการ : FR0112	ชื่อหน้าที่ : เพิ่มข้อมูลกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	
ผู้มีสิทธิใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.13 ความต้องการด้านหน้าที่ของระบบ รหัส FR0113

รหัสความต้องการ : FR0113	ชื่อหน้าที่ : เพิ่มข้อมูลกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา
คำอธิบายหน้าที่ : เพิ่มข้อมูลกำหนดแบบจำลองและกลยุทธ์ในการพัฒนาสำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการกำหนดแบบจำลองและกลยุทธ์ในการพัฒนา	
ผู้มีสิทธิใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.14 ความต้องการด้านหน้าที่ของระบบ รหัส FR0114

รหัสความต้องการ : FR0114	ชื่อหน้าที่ : เพิ่มข้อมูลกำหนดแผนการบูรณาการ
คำอธิบายหน้าที่ : เพิ่มข้อมูลกำหนดแผนการบูรณาการสำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 	
ข้อมูลนำออก : <ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการกำหนดแผนการบูรณาการ 	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.15 ความต้องการด้านหน้าที่ของระบบ รหัส FR0115

รหัสความต้องการ : FR0115	ชื่อหน้าที่ : เพิ่มข้อมูลทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 	
ข้อมูลนำออก : <ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์ 	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.16 ความต้องการด้านหน้าที่ของระบบ รหัส FR0116

รหัสความต้องการ : FR0116	ชื่อหน้าที่ : เพิ่มข้อมูลกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการกำหนดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	
ผู้มีสิทธิใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.17 ความต้องการด้านหน้าที่ของระบบ รหัส FR0117

รหัสความต้องการ : FR0117	ชื่อหน้าที่ : เพิ่มข้อมูลกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการกำหนดรายละเอียดสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์	
ผู้มีสิทธิใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.18 ความต้องการด้านหน้าที่ของระบบ รหัส FR0118

รหัสความต้องการ : FR0118	ชื่อหน้าที่ : เพิ่มข้อมูลกำหนดแผนการบำรุงรักษา
คำอธิบายหน้าที่ : เพิ่มข้อมูลกำหนดแผนการบำรุงรักษาสำหรับกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ :	<ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ
ข้อมูลนำออก :	<ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการกำหนดแผนการบำรุงรักษา
ผู้มีสิทธิ์ใช้งาน :	ผู้จัดการโครงการ

ตารางที่ ง.19 ความต้องการด้านหน้าที่ของระบบ รหัส FR0119

รหัสความต้องการ : FR0119	ชื่อหน้าที่ : เพิ่มข้อมูลสร้างกระบวนการบำรุงรักษา
คำอธิบายหน้าที่ : เพิ่มข้อมูลสร้างกระบวนการบำรุงรักษาสำหรับกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ :	<ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ
ข้อมูลนำออก :	<ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการสร้างกระบวนการบำรุงรักษา
ผู้มีสิทธิ์ใช้งาน :	ผู้จัดการโครงการ

ตารางที่ ง.20 ความต้องการด้านหน้าที่ของระบบ รหัส FR0120

รหัสความต้องการ : FR0120	ชื่อหน้าที่ : เพิ่มข้อมูลทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการทบทวนแหล่งเก็บสายผลิตภัณฑ์ซอฟต์แวร์	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.21 ความต้องการด้านหน้าที่ของระบบ รหัส FR0121

รหัสความต้องการ : FR0121	ชื่อหน้าที่ : เพิ่มข้อมูลวิเคราะห์ผลกระทบสินทรัพย์ที่สามารถนำมาใช้ซ้ำ
คำอธิบายหน้าที่ : เพิ่มข้อมูลวิเคราะห์ผลกระทบสินทรัพย์ที่สามารถนำมาใช้ซ้ำสำหรับกระบวนการบำรุงรักษาซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการวิเคราะห์ผลกระทบสินทรัพย์ที่สามารถนำมาใช้ซ้ำ	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.22 ความต้องการด้านหน้าที่ของระบบ รหัส FR0122

รหัสความต้องการ : FR0122	ชื่อหน้าที่ : เพิ่มข้อมูลกำหนดวิธีการบำรุงรักษา
คำอธิบายหน้าที่ : เพิ่มข้อมูลกำหนดวิธีการบำรุงรักษาสำหรับกระบวนการการบำรุงรักษาซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการกำหนดวิธีการบำรุงรักษา	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.23 ความต้องการด้านหน้าที่ของระบบ รหัส FR0123

รหัสความต้องการ : FR0123	ชื่อหน้าที่ : เพิ่มข้อมูลสร้างจุดมุ่งหมายของการวัดผล
คำอธิบายหน้าที่ : เพิ่มข้อมูลสร้างจุดมุ่งหมายของการวัดผลสำหรับกระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการสร้างจุดมุ่งหมายของการวัดผล	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.24 ความต้องการด้านหน้าที่ของระบบ รหัส FR0124

รหัสความต้องการ : FR0124	ชื่อหน้าที่ : เพิ่มข้อมูลวางแผนกระบวนการงานวัดผล
คำอธิบายหน้าที่ : เพิ่มข้อมูลวางแผนกระบวนการงานวัดผลสำหรับกระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 	
ข้อมูลนำออก : <ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการวางแผนกระบวนการงานวัดผล 	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.25 ความต้องการด้านหน้าที่ของระบบ รหัส FR0125

รหัสความต้องการ : FR0125	ชื่อหน้าที่ : เพิ่มข้อมูลวัดผลการพัฒนาซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลวัดผลการพัฒนาซอฟต์แวร์สำหรับกระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 	
ข้อมูลนำออก : <ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการวัดผลการพัฒนาซอฟต์แวร์ 	
ผู้มีสิทธิ์ใช้งาน : ผู้จัดการโครงการ	

ตารางที่ ง.26 ความต้องการด้านหน้าที่ของระบบ รหัส FR0126

รหัสความต้องการ : FR0126	ชื่อหน้าที่ : เพิ่มข้อมูลวิเคราะห์และจัดเก็บผลลัพธ์
คำอธิบายหน้าที่ : เพิ่มข้อมูลวิเคราะห์และจัดเก็บผลลัพธ์สำหรับกระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ :	<ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ
ข้อมูลนำออก :	<ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการวิเคราะห์และจัดเก็บผลลัพธ์
ผู้มีสิทธิ์ใช้งาน :	ผู้จัดการโครงการ

ตารางที่ ง.27 ความต้องการด้านหน้าที่ของระบบ รหัส FR0127

รหัสความต้องการ : FR0127	ชื่อหน้าที่ : เพิ่มข้อมูลรายงานการวัดผลและแนวทางสำหรับการปรับปรุงกระบวนการ
คำอธิบายหน้าที่ : เพิ่มข้อมูลรายงานการวัดผลและแนวทางสำหรับการปรับปรุงกระบวนการสำหรับกระบวนการสนับสนุนซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ :	<ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ
ข้อมูลนำออก :	<ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการรายงานการวัดผลและแนวทางสำหรับการปรับปรุงกระบวนการ
ผู้มีสิทธิ์ใช้งาน :	ผู้จัดการโครงการ

ตารางที่ ง.28 ความต้องการด้านหน้าที่ของระบบ รหัส FR0128

รหัสความต้องการ : FR0128	ชื่อหน้าที่ : เพิ่มข้อมูลพัฒนาส่วนประกอบ
คำอธิบายหน้าที่ : เพิ่มข้อมูลพัฒนาส่วนประกอบสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 	
ข้อมูลนำออก : <ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการพัฒนาส่วนประกอบ 	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.29 ความต้องการด้านหน้าที่ของระบบ รหัส FR0129

รหัสความต้องการ : FR0129	ชื่อหน้าที่ : เพิ่มข้อมูลบูรณาการกับระบบที่มีอยู่
คำอธิบายหน้าที่ : เพิ่มข้อมูลบูรณาการกับระบบที่มีอยู่สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 	
ข้อมูลนำออก : <ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการบูรณาการกับระบบที่มีอยู่ 	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.30 ความต้องการด้านหน้าที่ของระบบ รหัส FR0130

รหัสความต้องการ : FR0130	ชื่อหน้าที่ : เพิ่มข้อมูลทดสอบสายผลิตภัณฑ์
คำอธิบายหน้าที่ : เพิ่มข้อมูลทดสอบสายผลิตภัณฑ์สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการทดสอบสายผลิตภัณฑ์	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.31 ความต้องการด้านหน้าที่ของระบบ รหัส FR0131

รหัสความต้องการ : FR0131	ชื่อหน้าที่ : เพิ่มข้อมูลจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป
คำอธิบายหน้าที่ : เพิ่มข้อมูลจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูปสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.32 ความต้องการด้านหน้าที่ของระบบ รหัส FR0132

รหัสความต้องการ : FR0132	ชื่อหน้าที่ : เพิ่มข้อมูลปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บสายผลิตภัณฑ์
คำอธิบายหน้าที่ : เพิ่มข้อมูลปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บสายผลิตภัณฑ์สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บสายผลิตภัณฑ์	
ผู้มีสิทธิใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.33 ความต้องการด้านหน้าที่ของระบบ รหัส FR0133

รหัสความต้องการ : FR0133	ชื่อหน้าที่ : เพิ่มข้อมูลวางแผนการติดตั้ง
คำอธิบายหน้าที่ : เพิ่มข้อมูลวางแผนการติดตั้งสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการวางแผนการติดตั้ง	
ผู้มีสิทธิใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.34 ความต้องการด้านหน้าที่ของระบบ รหัส FR0134

รหัสความต้องการ : FR0134	ชื่อหน้าที่ : เพิ่มข้อมูลส่งมอบผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลส่งมอบผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.35 ความต้องการด้านหน้าที่ของระบบ รหัส FR0135

รหัสความต้องการ : FR0135	ชื่อหน้าที่ : เพิ่มข้อมูลพัฒนาส่วนประกอบ
คำอธิบายหน้าที่ : เพิ่มข้อมูลพัฒนาส่วนประกอบสำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการพัฒนาส่วนประกอบ	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.36 ความต้องการด้านหน้าที่ของระบบ รหัส FR0136

รหัสความต้องการ : FR0136	ชื่อหน้าที่ : เพิ่มข้อมูลบูรณาการระบบและซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลบูรณาการระบบและซอฟต์แวร์สำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการบูรณาการระบบและซอฟต์แวร์	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.37 ความต้องการด้านหน้าที่ของระบบ รหัส FR0137

รหัสความต้องการ : FR0137	ชื่อหน้าที่ : เพิ่มข้อมูลทดสอบสายผลิตภัณฑ์
คำอธิบายหน้าที่ : เพิ่มข้อมูลทดสอบสายผลิตภัณฑ์สำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการทดสอบสายผลิตภัณฑ์	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.38 ความต้องการด้านหน้าที่ของระบบ รหัส FR0138

รหัสความต้องการ : FR0138	ชื่อหน้าที่ : เพิ่มข้อมูลจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป
คำอธิบายหน้าที่ : เพิ่มข้อมูลจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูปสำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการจัดการเวอร์ชันของซอฟต์แวร์สำเร็จรูป	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.39 ความต้องการด้านหน้าที่ของระบบ รหัส FR0139

รหัสความต้องการ : FR0132	ชื่อหน้าที่ : เพิ่มข้อมูลปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บสายผลิตภัณฑ์
คำอธิบายหน้าที่ : เพิ่มข้อมูลปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บสายผลิตภัณฑ์สำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการปรับปรุงซอฟต์แวร์สำเร็จรูปในแหล่งเก็บสายผลิตภัณฑ์	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.40 ความต้องการด้านหน้าที่ของระบบ รหัส FR0140

รหัสความต้องการ : FR0140	ชื่อหน้าที่ : เพิ่มข้อมูลวางแผนการติดตั้ง
คำอธิบายหน้าที่ : เพิ่มข้อมูลวางแผนการติดตั้งสำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการวางแผนการติดตั้ง	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.41 ความต้องการด้านหน้าที่ของระบบ รหัส FR0141

รหัสความต้องการ : FR0141	ชื่อหน้าที่ : เพิ่มข้อมูลส่งมอบผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลส่งมอบผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ	
ข้อมูลนำออก : 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการส่งมอบผลิตภัณฑ์ซอฟต์แวร์	
ผู้มีสิทธิ์ใช้งาน : ทีมผู้พัฒนา	

ตารางที่ ง.42 ความต้องการด้านหน้าที่ของระบบ รหัส FR0142

รหัสความต้องการ : FR0142	ชื่อหน้าที่ : เพิ่มข้อมูลประเมินข้อกำหนดความต้องการ
คำอธิบายหน้าที่ : เพิ่มข้อมูลประเมินข้อกำหนดความต้องการสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ :	<ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 5. คะแนนผลการประเมิน
ข้อมูลนำออก :	<ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการประเมินข้อกำหนดความต้องการ 3. เอกสารสนับสนุนการประเมิน
ผู้มีสิทธิใช้งาน : ทีมผู้ประเมิน	

ตารางที่ ง.43 ความต้องการด้านหน้าที่ของระบบ รหัส FR0143

รหัสความต้องการ : FR0143	ชื่อหน้าที่ : เพิ่มข้อมูลประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ :	<ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 5. คะแนนผลการประเมิน
ข้อมูลนำออก :	<ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ 3. เอกสารสนับสนุนการประเมิน
ผู้มีสิทธิใช้งาน : ทีมผู้ประเมิน	

ตารางที่ ง.44 ความต้องการด้านหน้าที่ของระบบ รหัส FR0144

รหัสความต้องการ : FR0144	ชื่อหน้าที่ : เพิ่มข้อมูลประเมินการพัฒนาส่วนประกอบ
คำอธิบายหน้าที่ : เพิ่มข้อมูลประเมินการพัฒนาส่วนประกอบสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ :	<ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 5. คะแนนผลการประเมิน
ข้อมูลนำออก :	<ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการประเมินการพัฒนาส่วนประกอบ 3. เอกสารสนับสนุนการประเมิน
ผู้มีสิทธิใช้งาน : ทีมผู้ประเมิน	

ตารางที่ ง.45 ความต้องการด้านหน้าที่ของระบบ รหัส FR0145

รหัสความต้องการ : FR0145	ชื่อหน้าที่ : เพิ่มข้อมูลประเมินการบูรณาการกับระบบที่มีอยู่
คำอธิบายหน้าที่ : เพิ่มข้อมูลประเมินการบูรณาการกับระบบที่มีอยู่สำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ :	<ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 5. คะแนนผลการประเมิน
ข้อมูลนำออก :	<ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการประเมินการบูรณาการกับระบบที่มีอยู่ 3. เอกสารสนับสนุนการประเมิน
ผู้มีสิทธิใช้งาน : ทีมผู้ประเมิน	

ตารางที่ ง.46 ความต้องการด้านหน้าที่ของระบบ รหัส FR0146

รหัสความต้องการ : FR0146	ชื่อหน้าที่ : เพิ่มข้อมูลประเมินแผนการติดตั้ง
คำอธิบายหน้าที่ : เพิ่มข้อมูลประเมินแผนการติดตั้งสำหรับกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ :	<ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 5. คะแนนผลการประเมิน
ข้อมูลนำออก :	<ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการประเมินแผนการติดตั้ง 3. เอกสารสนับสนุนการประเมิน
ผู้มีสิทธิใช้งาน : ทีมผู้ประเมิน	

ตารางที่ ง.47 ความต้องการด้านหน้าที่ของระบบ รหัส FR0147

รหัสความต้องการ : FR0147	ชื่อหน้าที่ : เพิ่มข้อมูลประเมินข้อกำหนดความต้องการ
คำอธิบายหน้าที่ : เพิ่มข้อมูลประเมินข้อกำหนดความต้องการสำหรับกระบวนการการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ :	<ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 5. คะแนนผลการประเมิน
ข้อมูลนำออก :	<ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการประเมินข้อกำหนดความต้องการ 3. เอกสารสนับสนุนการประเมิน
ผู้มีสิทธิใช้งาน : ทีมผู้ประเมิน	

ตารางที่ ง.48 ความต้องการด้านหน้าที่ของระบบ รหัส FR0148

รหัสความต้องการ : FR0148	ชื่อหน้าที่ : เพิ่มข้อมูลประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์สำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 5. คะแนนผลการประเมิน 	
ข้อมูลนำออก : <ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการประเมินสถาปัตยกรรมสายผลิตภัณฑ์ซอฟต์แวร์ 3. เอกสารสนับสนุนการประเมิน 	
ผู้มีสิทธิใช้งาน : ทีมผู้ประเมิน	

ตารางที่ ง.49 ความต้องการด้านหน้าที่ของระบบ รหัส FR0149

รหัสความต้องการ : FR0149	ชื่อหน้าที่ : เพิ่มข้อมูลประเมินการพัฒนาส่วนประกอบ
คำอธิบายหน้าที่ : เพิ่มข้อมูลประเมินการพัฒนาส่วนประกอบสำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 5. คะแนนผลการประเมิน 	
ข้อมูลนำออก : <ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการประเมินการพัฒนาส่วนประกอบ 3. เอกสารสนับสนุนการประเมิน 	
ผู้มีสิทธิใช้งาน : ทีมผู้ประเมิน	

ตารางที่ ง.50 ความต้องการด้านหน้าที่ของระบบ รหัส FR0150

รหัสความต้องการ : FR0150	ชื่อหน้าที่ : เพิ่มข้อมูลประเมินการบูรณาการระบบและซอฟต์แวร์
คำอธิบายหน้าที่ : เพิ่มข้อมูลประเมินการบูรณาการระบบและซอฟต์แวร์สำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 5. คะแนนผลการประเมิน 	
ข้อมูลนำออก : <ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการประเมินการบูรณาการระบบและซอฟต์แวร์ 3. เอกสารสนับสนุนการประเมิน 	
ผู้มีสิทธิใช้งาน : ทีมผู้ประเมิน	

ตารางที่ ง.51 ความต้องการด้านหน้าที่ของระบบ รหัส FR0151

รหัสความต้องการ : FR0151	ชื่อหน้าที่ : เพิ่มข้อมูลประเมินแผนการติดตั้ง
คำอธิบายหน้าที่ : เพิ่มข้อมูลประเมินแผนการติดตั้งสำหรับกระบวนการบูรณาการซอฟต์แวร์สำเร็จรูป	
ข้อมูลนำเข้าระบบ : <ol style="list-style-type: none"> 1. ชื่อภารกิจ 2. ทรัพยากรที่ใช้ 3. วัน/เดือน/ปี ที่เริ่มดำเนินการ 4. วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ 5. คะแนนผลการประเมิน 	
ข้อมูลนำออก : <ol style="list-style-type: none"> 1. เอกสารผลผลิตงาน 2. เอกสารสนับสนุนกระบวนการประเมินแผนการติดตั้ง 3. เอกสารสนับสนุนการประเมิน 	
ผู้มีสิทธิใช้งาน : ทีมผู้ประเมิน	

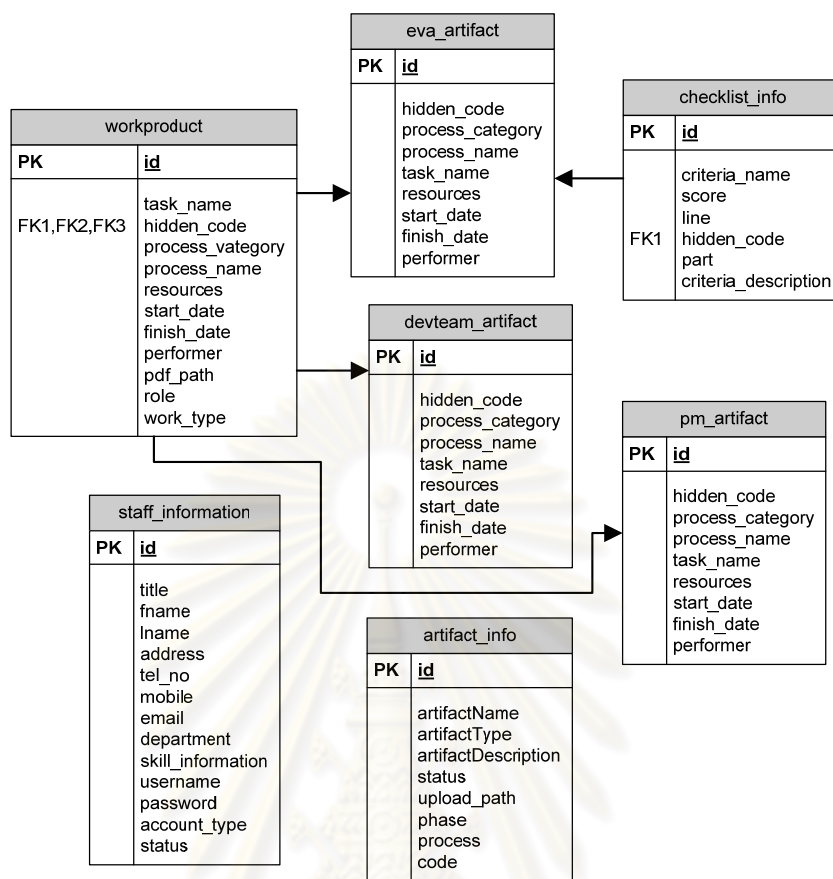
ภาคผนวก จ โครงสร้างตารางข้อมูล

ตารางของเครื่องมือสนับสนุนกระบวนการการพัฒนาซอฟต์แวร์สำเร็จรูปที่ใช้
สายผลิตภัณฑ์ซอฟต์แวร์เป็นฐาน มีทั้งหมด 7 ตาราง ซึ่งสามารถสรุปได้ดังตารางที่ จ.1

ตารางที่ จ.1 สรุปตารางของเครื่องมือสนับสนุน

ลำดับ	ชื่อตาราง	คำอธิบายตาราง
1	artifact_info	ข้อมูลเอกสารสนับสนุนกระบวนการ
2	staff_information	ข้อมูลผู้ใช้งานระบบ
3	pm_artifact	ข้อมูลเอกสารผลลัพธ์การดำเนินการของ ผู้จัดการโครงการ
4	devteam_artifact	ข้อมูลเอกสารผลลัพธ์การดำเนินการของ ทีมผู้พัฒนา
5	eva_artifact	ข้อมูลเอกสารผลลัพธ์การดำเนินการของ ทีมผู้ประเมิน
6	checklist_info	ข้อมูลการประเมินเกณฑ์การใช้อำนาจของรายการ ตรวจสอบ
7	workproduct	ข้อมูลผลผลิตงาน

โดยที่ตารางข้อมูลทั้งหมดจะถูกแสดงในรูปแบบของแบบจำลองข้อมูลเชิงกายภาพ
ดังรูปที่ จ.1 ทั้งนี้โครงสร้างตารางข้อมูลทั้งหมด แสดงได้ดังตารางที่ จ.2-จ.8



รูปที่ ๑.1 โครงสร้างฐานข้อมูลเชิงสัมพันธ์

ตารางที่ ๑.2 โครงสร้างตารางข้อมูลของข้อมูลเอกสารสนับสนุนกระบวนการ

ชื่อสดมภ์	ประเภทข้อมูล	คำอธิบาย
id	INT(10)	หมายเลขเอกสารสนับสนุน
artifactName	VARCHAR(300)	ชื่อเอกสารสนับสนุน
artifactType	VARCHAR(20)	ชนิดของเอกสารสนับสนุน
artifactDescription	VARCHAR(700)	คำอธิบายของเอกสารสนับสนุน
status	VARCHAR(10)	สถานะของเอกสารสนับสนุน
upload_path	VARCHAR(200)	ที่เก็บไฟล์ข้อมูลเอกสารสนับสนุน
phase	VARCHAR(70)	ประเภทของกลุ่มกระบวนการ
process	VARCHAR(100)	ชื่อกระบวนการที่ใช้เอกสารสนับสนุน
code	VARCHAR(10)	รหัสของเอกสารสนับสนุน

ตารางที่ ๑.3 โครงสร้างตารางข้อมูลของข้อมูลผู้ใช้งานระบบ

ชื่อสดมภ์	ประเภทข้อมูล	คำอธิบาย
id	INT(10)	หมายเลขบุคลากร
title	VARCHAR(10)	คำนำหน้าชื่อ
fname	VARCHAR(40)	ชื่อ
lname	VARCHAR(40)	ชื่อสกุล
address	VARCHAR(500)	ที่อยู่
tel_no	VARCHAR(60)	เบอร์โทรศัพท์
mobile	VARCHAR(60)	เบอร์โทรศัพท์เคลื่อนที่
email	VARCHAR(100)	ไปรษณีย์อิเล็กทรอนิกส์
department	VARCHAR(300)	ชื่อหน่วยงาน
skill_information	VARCHAR(700)	ข้อมูลทักษะหรือความชำนาญ
username	VARCHAR(20)	รหัสผู้ใช้งาน
password	VARCHAR(25)	รหัสผ่าน
account_type	VARCHAR(30)	ประเภทผู้ใช้งาน
status	VARCHAR(10)	สถานะการจ้างงาน

ตารางที่ ๑.4 โครงสร้างตารางข้อมูลของข้อมูลเอกสารผลลัพธ์การดำเนินการของผู้จัดการโครงการ

ชื่อสดมภ์	ประเภทข้อมูล	คำอธิบาย
id	INT(10)	หมายเลขเอกสารผลลัพธ์
hidden_code	VARCHAR(70)	รหัสเอกสารผลลัพธ์
process_category	VARCHAR(100)	ประเภทของกลุ่มกระบวนการ
process_name	VARCHAR(120)	ชื่อกระบวนการ
task_name	VARCHAR(400)	ชื่อภารกิจ
resources	VARCHAR(700)	ทรัพยากรที่ใช้ในการดำเนินการ
start_date	VARCHAR(20)	วัน/เดือน/ปี ที่เริ่มดำเนินการ
finish_date	VARCHAR(20)	วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ
performer	VARCHAR(40)	ชื่อผู้ดำเนินการ

ตารางที่ ๑.5 โครงสร้างตารางข้อมูลของข้อมูลเอกสารผลลัพธ์การดำเนินการของทีมผู้พัฒนา

ชื่อสดมภ์	ประเภทข้อมูล	คำอธิบาย
id	INT(10)	หมายเลขเอกสารผลลัพธ์
hidden_code	VARCHAR(70)	รหัสเอกสารผลลัพธ์
process_category	VARCHAR(100)	ประเภทของกลุ่มกระบวนการ
process_name	VARCHAR(120)	ชื่อกระบวนการ
task_name	VARCHAR(400)	ชื่อภารกิจ
resources	VARCHAR(700)	ทรัพยากรที่ใช้ในการดำเนินการ
start_date	VARCHAR(20)	วัน/เดือน/ปี ที่เริ่มดำเนินการ
finish_date	VARCHAR(20)	วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ
performer	VARCHAR(40)	ชื่อผู้ดำเนินการ

ตารางที่ ๑.6 โครงสร้างตารางข้อมูลของข้อมูลเอกสารผลลัพธ์การดำเนินการของทีมผู้ประเมิน

ชื่อสดมภ์	ประเภทข้อมูล	คำอธิบาย
id	INT(10)	หมายเลขเอกสารผลลัพธ์
hidden_code	VARCHAR(70)	รหัสเอกสารผลลัพธ์
process_category	VARCHAR(100)	ประเภทของกลุ่มกระบวนการ
process_name	VARCHAR(120)	ชื่อกระบวนการ
task_name	VARCHAR(400)	ชื่อภารกิจ
resources	VARCHAR(700)	ทรัพยากรที่ใช้ในการดำเนินการ
start_date	VARCHAR(20)	วัน/เดือน/ปี ที่เริ่มดำเนินการ
finish_date	VARCHAR(20)	วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ
performer	VARCHAR(40)	ชื่อผู้ดำเนินการ

ตารางที่ ๑.7 โครงสร้างตารางข้อมูลของข้อมูลการประเมินเกณฑ์การใช้ซ้ำของรายการตรวจสอบ

ชื่อสดมภ์	ประเภทข้อมูล	คำอธิบาย
Id	INT(10)	หมายเลขข้อมูลการประเมินเกณฑ์การใช้ซ้ำของรายการตรวจสอบ
criteria_name	VARCHAR(50)	ชื่อเกณฑ์การใช้ซ้ำ
score	VARCHAR(15)	คะแนนผลลัพท์การประเมิน
line	VARCHAR(15)	คะแนนมาตรฐานสำหรับการประเมิน
hidden_code	VARCHAR(70)	รหัสของข้อมูลการประเมินเกณฑ์การใช้ซ้ำของรายการตรวจสอบ
part	VARCHAR(20)	ส่วนเกณฑ์การใช้ซ้ำของรายการตรวจสอบ
criteria_description	VARCHAR(500)	คำอธิบายเกณฑ์การใช้ซ้ำของรายการตรวจสอบ

ตารางที่ ๑.8 โครงสร้างตารางข้อมูลของข้อมูลผลผลิตงาน

ชื่อสดมภ์	ประเภทข้อมูล	คำอธิบาย
id	INT(10)	หมายเลขเอกสารผลผลิตงาน
task_name	VARCHAR(500)	ชื่อภารกิจ
hidden_code	VARCHAR(70)	รหัสเอกสารผลผลิตงาน
process_category	VARCHAR(100)	ประเภทของกลุ่มกระบวนการ
process_name	VARCHAR(200)	ชื่อกระบวนการ
resources	VARCHAR(700)	ทรัพยากรที่ใช้ในการดำเนินการ
start_date	VARCHAR(20)	วัน/เดือน/ปี ที่เริ่มดำเนินการ
finish_date	VARCHAR(20)	วัน/เดือน/ปี ที่สิ้นสุดการดำเนินการ
performer	VARCHAR(100)	ชื่อผู้ดำเนินการ
pdf_path	VARCHAR(250)	ที่เก็บไฟล์ข้อมูลเอกสารผลผลิตงาน
role	VARCHAR(50)	บทบาทหน้าที่ของผู้ดำเนินงาน
work_type	VARCHAR(70)	ประเภทของเอกสารผลผลิตงาน

ประวัติผู้เขียนวิทยานิพนธ์

นายยุทธนันท์ ก้นหาชน เกิดวันที่ 26 กันยายน พ.ศ. 2527 ที่จังหวัดขอนแก่น โดยมีประวัติการศึกษาดังต่อไปนี้

- สำเร็จการศึกษาระดับประถมศึกษาที่โรงเรียนบ้านไผ่ประถมศึกษา (ข.ก.10) อำเภอบ้านไผ่ จังหวัดขอนแก่น

- สำเร็จการศึกษาระดับมัธยมศึกษาตอนต้นและตอนปลายที่โรงเรียนบ้านไผ่ (ข.ก.5) อำเภอบ้านไผ่ จังหวัดขอนแก่น

- สำเร็จการศึกษาระดับปริญญาวิทยาศาสตรบัณฑิต จากภาควิชาคณิตศาสตร์ สถิติและคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยอุบลราชธานี อำเภอวารินชำราบ จังหวัดอุบลราชธานี

และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมซอฟต์แวร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย สำหรับการศึกษานี้ผู้วิจัยได้มีผลงานวิชาการร่วมกับอาจารย์ที่ปรึกษา ซึ่งเป็นบทความวิชาการระดับนานาชาติและระดับชาติ รวม 2 บทความ ได้แก่

1) บทความวิชาการเรื่อง “การออกแบบและพัฒนาแบบจำลองกระบวนการการปรับแต่งซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์” ซึ่งได้รับการคัดเลือกเพื่อนำเสนอและตีพิมพ์ในงาน “การประชุมวิชาการร่วมระดับนานาชาติด้านวิทยาการคอมพิวเตอร์และวิศวกรรมซอฟต์แวร์ ครั้งที่ 7 (The 7th International Joint Conference on Computer Science and Software Engineering : JCSSE 2010)” ระหว่างวันที่ 12-14 พฤษภาคม 2553 ณ มหาวิทยาลัยรามคำแหง กรุงเทพฯ ประเทศไทย

2) บทความวิชาการเรื่อง “การนิยามแบบจำลองกระบวนการและเครื่องมือสนับสนุนสำหรับการบูรณาการซอฟต์แวร์สำเร็จรูปที่ใช้สายผลิตภัณฑ์ซอฟต์แวร์” ซึ่งได้รับการคัดเลือกเพื่อนำเสนอและตีพิมพ์ในงาน “การประชุมวิชาการระดับชาติด้านวิทยาการคอมพิวเตอร์และวิศวกรรมคอมพิวเตอร์ ครั้งที่ 14 (The 14th National Computer Science and Engineering Conference : NCSEC 2010)” ระหว่างวันที่ 17-19 พฤศจิกายน 2553 ณ โรงแรมเซ็นทารา ดวงตะวัน เชียงใหม่ ประเทศไทย