ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การประยุกต์ใช้ CMM ระดับที่ 2 กับโครงการขนาดเล็ก กรณีศึกษา: ระบบบริหารคลังวัตถุดิบ

Applying CMM Level 2 in small Projects

Case Study: Inventory System

H002000

โดย

นางสาวดวงสมร พันธ์พิกุล

รหัส 44067088

วัน เดือน ปี 22 ส.ค. 2550
เลขทะเบียน 02000
เลขเรียกหนังสือ วพ ด2311 ก 2545
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."

อาจารย์ที่ปรึกษา

ผศ.ดร.ประจวบ วานิชชัชวาล

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 2 ปีการศึกษา 2545
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	การประยุกต์ใช้ CMM ระดับที่ 2 กับโครงการขนาดเล็ก กรณีศึกษา: ระบบบริหารคลังวัตถุดิบ
นักศึกษา	นางสาวดวงสมร พันธ์พิกุล
อาจารย์ที่ปรึกษา	ผศ.ดร.ประจวบ วานิชชัชวาล
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2545

บทคัดย่อ

CMM (Capability Maturity Model) เป็นเครื่องมือที่พัฒนาขึ้นมาเพื่อตอบสนองความต้องการในการพัฒนาซอฟต์แวร์อย่างมีประสิทธิภาพ ซึ่งออกแบบมาให้ใช้ได้ดีกับองค์กรขนาดใหญ่ ที่มีทีมพัฒนาขนาดใหญ่ มีระบบขนาดใหญ่และมีซอฟต์แวร์ที่มีองค์ประกอบหลากหลาย แต่กิจกรรมหลายกิจกรรมที่ดูเหมือนจะเป็นสิ่งสำคัญในองค์กรขนาดใหญ่ อาจจะไม่ได้เป็นสิ่งจำเป็นในองค์กรหรือโครงการขนาดเล็กที่มีความยุ่งยากซับซ้อนน้อยกว่า มีทีมพัฒนาขนาดเล็กกว่าและใช้เวลาในการพัฒนาที่จำกัดกว่า การนำ CMM มาประยุกต์ใช้ให้ได้ผลดีกับองค์กรหรือโครงการขนาดเล็กจึงเป็นเรื่องที่น่าศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Applying CMM Level 2 in Small Projects .
	Case study: Inventory System
Student	Miss. Duangsamorn Panpikul
Advisor	Asst. Prof. Prachuab Vanitchatchavan, Ph.D.
Level of Study	Master of Science in Information Technology
Major	Information Science
Academic Year	2002

ABSTRACT

Capability Maturity Model (CMM) was developed in order to improve software process. It focus on large organization that has so many developers , large systems and variety of software components. Some activities in large organization need not have to do in small organization which has smaller team, smaller system and the schedule is shorter. Using CMM in small organization is valuable to study.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานฉบับนี้ สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างดียิ่งของ ผศ.ดร.ประจวบ วานิชชัชวาล อาจารย์ที่ปรึกษาโครงการ ซึ่งได้ให้คำแนะนำและข้อคิดเห็นต่างๆ ในการวิจัยเสมอมา

ขอขอบคุณพี่ๆ และ เพื่อนๆ จากศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ ที่ได้ให้คำแนะนำ ช่วยเหลือ เป็นอย่างดี และให้กำลังใจมาโดยตลอด

ท้ายนี้ ผู้เขียนใคร่ขอกราบขอบพระคุณบิดา มารดา และบุคคลในครอบครัว ซึ่งคอยให้กำลังใจแก่ผู้เขียนเสมอจนสำเร็จการศึกษา

ดวงสมร พันธ์พิกุล

มีนาคม 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญภาพ VI

บทที่ 1 1

1.1 ความเป็นมาของปัญหา 1

1.2 วัตถุประสงค์ของโครงการ 2

1.3 ขอบเขตของการดำเนินงาน 2

1.4 ขั้นตอนและวิธีการดำเนินงานโครงการ 3

1.5 ประโยชน์ที่คาดว่าจะได้รับ 3

บทที่ 2 4

2.1 ภาพรวมของแบบจำลองวุฒิภาวะความสามารถ 4

2.2 แบบจำลองวุฒิภาวะความสามารถ ระดับที่ 2 7

2.3 การประยุกต์ใช้ CMM ระดับที่ 2 กับองค์กรหรือโครงการขนาดเล็ก 19

บทที่ 3 32

3.1 เอกสารนโยบายเกี่ยวกับ RM, SPP, SPTO, SQA และ SCM 33

3.2 เอกสารกระบวนการในการบริหารจัดการความต้องการของผู้ใช้ 35

3.3 เอกสารกระบวนการในการวางแผนโครงการ 37

3.4 กระบวนการในเรื่องการดูแลติดตามโครงการ 38

3.5 เอกสารกระบวนการในเรื่องการประกันคุณภาพซอฟต์แวร์ 40

3.6 กระบวนการในการจัดการองค์ประกอบซอฟต์แวร์ 41

3.7 เอกสารแนวทางการดำเนินงานด้านการจัดการองค์ประกอบซอฟต์แวร์ 43

3.8 เอกสารความต้องการของระบบ 48

3.9 แผนการพัฒนาซอฟต์แวร์และแผนการติดตามงาน 50

3.10 แผนการประกันคุณภาพ 52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.11 แผนการจัดการองค์ประกอบซอฟต์แวร์ 53
3.12 เอกสารการขอเปลี่ยนแปลงความต้องการ 55
3.13 เอกสารการขอเปลี่ยนแปลง baseline 56
3.14 เอกสารการขอเปลี่ยนแปลงซอฟต์แวร์ 57
3.15 รายงานการวัดผลการดำเนินงานกิจกรรม 58
บทที่ 4 59
4.1 ข้อกำหนดความต้องการ 61
4.2 แผนงานโครงการและการติดตามงาน 63
4.3 แผนการประกันคุณภาพซอฟต์แวร์ 68
4.4 แผนการจัดการองค์ประกอบซอฟต์แวร์ 70
4.5 เอกสารการยืนยันหน้าจอของระบบ 74
4.6 เอกสารการยืนยันรายงานของระบบ 75
4.7 เอกสาร Program Specification 76
4.8 ตัวอย่างการใช้งานแบบฟอร์มการขอเปลี่ยนแปลงความต้องการ 77
4.9 ตัวอย่างการใช้งานแบบฟอร์มการขอเปลี่ยนแปลง baseline 79
4.10 ตัวอย่างการใช้งานแบบฟอร์มการขอเปลี่ยนแปลง software 80
4.11 ตัวอย่างรายงานสรุปสถานะความต้องการแต่ละตัว 81
บทที่ 5 82

บรรณานุกรม 84

ภาคผนวก ก ตัวอย่างหน้าจอและโครงสร้างข้อมูลระบบบริหารคลังวัตถุดิบ 85
ภาคผนวก ข รายการตรวจสอบการเข้าสู่ CMM ระดับที่ 2 99

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

หน้า

รูปที่ 2.1 แสดง 5 ระดับของวุฒิภาวะ 4
รูปที่ 2.2 โครงสร้างของ CMM 6
รูปที่ 2.3 แสดงบทบาทในองค์กรขนาดเล็กมากมาก (XXS) 20
รูปที่ 2.4 แสดงบทบาทในองค์กรขนาดเล็กมาก (XS) 21
รูปที่ 2.5 แสดงบทบาทในองค์กรขนาดเล็ก (S) 22
รูปที่ 2.6 แสดงกิจกรรมของ RM 25
รูปที่ 2.7 แสดงกิจกรรมของ SPP 26
รูปที่ 2.8 แสดงกิจกรรมของ SPTO 27
รูปที่ 2.9 แสดงกิจกรรมของ SSM 28
รูปที่ 2.10 แสดงกิจกรรมของ SQA 29
รูปที่ 2.11 แสดงกิจกรรมของ SCM 30
รูปที่ 3.1 ตัวอย่างนโยบายใน CMM ระดับที่ 2 35
รูปที่ 3.2 ตัวอย่างเอกสารกระบวนการในการจัดการความต้องการของผู้ใช้ 36
รูปที่ 3.3 ตัวอย่างเอกสารกระบวนการในการวางแผน โครงการ 38
รูปที่ 3.3 ตัวอย่างเอกสารกระบวนการในการดูแลติดตามโครงการ 39
รูปที่ 3.5 ตัวอย่างเอกสารกระบวนการในการประกันคุณภาพซอฟต์แวร์ 41
รูปที่ 3.6 ตัวอย่างเอกสารกระบวนการในการจัดการองค์ประกอบซอฟต์แวร์ 42
รูปที่ 3.7 ตัวอย่างเอกสารแนวทางการดำเนินงานด้านการจัดการองค์ประกอบซอฟต์แวร์ 48
รูปที่ 3.8 ตัวอย่างเอกสารแนวทางการดำเนินงานด้านการจัดการองค์ประกอบซอฟต์แวร์ 50
รูปที่ 3.9 ตัวอย่างเอกสารแนวทางการดำเนินงานด้านการจัดการองค์ประกอบซอฟต์แวร์ 52
รูปที่ 3.10 ตัวอย่างรูปแบบเอกสารแผนการรับประกันคุณภาพ 53
รูปที่ 3.11 ตัวอย่างเอกสารรูปแบบแผนการจัดการองค์ประกอบซอฟต์แวร์ 54
รูปที่ 3.12 ตัวอย่างเอกสารแบบฟอร์มการขอเปลี่ยนแปลงความต้องการ 55
รูปที่ 3.13 ตัวอย่างเอกสารแบบฟอร์มการขอเปลี่ยนแปลง baseline 56
รูปที่ 3.14 ตัวอย่างเอกสารแบบฟอร์มการขอเปลี่ยนแปลงซอฟต์แวร์ 57
รูปที่ 4.1 แสดงบทบาทในโครงการขนาดเล็กมากๆ (XXS) 59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.2 กระบวนการในการทำงานตามแนวทางของ CMM 60
รูปที่ 4.3 ตัวอย่างการประยุกต์ใช้ข้อกำหนดความต้องการ 63
รูปที่ 4.4 ตัวอย่างการประยุกต์ใช้แผนการพัฒนาซอฟต์แวร์ 67
รูปที่ 4.5 ตัวอย่างการประยุกต์ใช้แผนการรับประกันระบบ 70
รูปที่ 4.6 ตัวอย่างการประยุกต์ใช้แผนการจัดการองค์ประกอบซอฟต์แวร์ 74
รูปที่ 4.7 ตัวอย่างการประยุกต์ใช้เอกสารการยืนยันหน้าจอของระบบ 75
รูปที่ 4.8 ตัวอย่างการประยุกต์ใช้ข้อกำหนดความต้องการ 75
รูปที่ 4.9 ตัวอย่างการประยุกต์ใช้ Program Specification 76
รูปที่ 4.10 ตัวอย่างการประยุกต์ใช้แบบฟอร์มการขอเปลี่ยนแปลงความต้องการ 77
(กรณีมีการเปลี่ยนแปลงมีผลกระทบไม่มาก) 77
รูปที่ 4.11 ตัวอย่างการประยุกต์ใช้แบบฟอร์มการขอเปลี่ยนแปลงความต้องการ 78
(กรณีการเปลี่ยนแปลงมีผลกระทบในหลายส่วน) 78
รูปที่ 4.12 ตัวอย่างการประยุกต์ใช้แบบฟอร์มการขอเปลี่ยนแปลง baseline 79
รูปที่ 4.13 ตัวอย่างการประยุกต์ใช้แบบฟอร์มการขอเปลี่ยนแปลง software 80
รูปที่ 4.14 ตัวอย่างรายงานสรุปสถานะความต้องการแต่ละตัว 81

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของปัญหา

บนโลกที่เต็มไปด้วยการแข่งขันในปัจจุบัน องค์กรซอฟต์แวร์จะสามารถอยู่รอดได้ต้องอาศัยความพอใจของลูกค้าเป็นหลัก แต่ในขณะเดียวกัน คุณภาพของซอฟต์แวร์ดูเหมือนว่าจะถดถอยลง แม้ว่าซอฟต์แวร์ได้เข้ามายุ่งเกี่ยวกับชีวิตประจำวันของเรามากขึ้น แต่วิกฤติซอฟต์แวร์ที่เกิดขึ้นใน 2-3 ทศวรรษที่ผ่านมา เริ่มที่จะวิกฤติขึ้นเรื่อยๆ

คำว่า "วิกฤตซอฟต์แวร์" อาจจะฟังดูเกินจริงไป ซอฟต์แวร์แพร่หลายได้รวดเร็วเพราะมันมีความสามารถหลากหลาย และเทคโนโลยีของซอฟต์แวร์ก็ก้าวหน้าอย่างรวดเร็ว แต่ดูเหมือนว่าความซับซ้อนของปัญหาจะเติบโตเร็วว่าศักยภาพของเราในการที่จะพัฒนาและบำรุงรักษาซอฟต์แวร์ นอกจากนี้ความสามารถในการพัฒนาและส่งมอบซอฟต์แวร์ให้อยู่ในงบประมาณและตรงตามเวลา เป็นปัญหาขององค์กรซอฟต์แวร์ในปัจจุบัน

ได้มีการการค้นหาแนวทางเป็นเวลาหลายปี จนปัจจุบันองค์กรซอฟต์แวร์ได้ตระหนักว่าปัญหาพื้นฐานก็คือการที่*ไม่สามารถบริหารจัดการกระบวนการพัฒนาซอฟต์แวร์อย่างมีประสิทธิภาพได้*

Capability Maturity Model (CMM) (แบบจำลองวุฒิภาวะความสามารถ) ถูกพัฒนาขึ้นมาโดย SEI (สถาบันวิศวกรรมซอฟต์แวร์) ของมหาวิทยาลัยคาร์เนกี เมลลอน ซึ่งเป็นแบบจำลองสำหรับวัดว่าหน่วยงานที่ทำหน้าที่พัฒนาซอฟต์แวร์นั้น มีความสามารถและได้บรรลุวุฒิภาวะในการทำงานมากน้อยเพียงใดโดยที่ CMM จะอธิบายถึงการปรับปรุงองค์กรซอฟต์แวร์จากองค์กรที่ไม่เติบโตมาเป็นองค์กรที่เติบโตและมีวินัยในตัวเอง โดยจะแบ่งระดับการเติบโตออกเป็น 5 ระดับ ซึ่งจะได้อธิบายต่อไป CMM จะแนะแนวทางให้องค์กรซอฟต์แวร์ที่ต้องการเพิ่มการควบคุมกระบวนการพัฒนาและบำรุงรักษาซอฟต์แวร์ โดยมีจุดประสงค์ก็คือแนะให้องค์กรเลือกกลยุทธ์ที่จะพัฒนากระบวนการโดยตัดสินใจจากระดับวุฒิภาวะที่เป็นอยู่ในปัจจุบันและสิ่งที่วิกฤติมากที่สุด ซึ่งการให้ความสนใจเฉพาะในประเด็นที่สำคัญบางเรื่องนี้ทำให้องค์กรสามารถปรับปรุงกระบวนการได้อย่างต่อเนื่องและมีประสิทธิภาพ

ถึงแม้ว่า CMM version 1.1 ที่พัฒนาขึ้นมานั้นจะมุ่งเน้นในการประยุกต์ใช้กับองค์กรขนาดใหญ่ที่มีโครงการขนาดใหญ่ แต่ในความเป็นจริงแล้ว CMM ถูกพัฒนาขึ้นเพื่อใช้เป็นแนวทางและ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างในการบริหารจัดการโครงการทุกขนาด การบริหารจัดการโครงการให้ตรงตามเวลา อยู่ในงบประมาณ และเป็นไปตามความต้องการของผู้ใช้นั้น เป็นสิ่งสำคัญทั้งในองค์กรขนาดเล็กและขนาดใหญ่ ผู้ใช้ CMM ควรนำ CMM มาประยุกต์ใช้โดยอาศัยประสบการณ์และความรู้ในเรื่องของวิศวกรรมซอฟต์แวร์และการบริหารจัดการ โครงการรวมทั้งสภาพแวดล้อมขององค์กรมาช่วยในการตัดสินใจ

ในปัจจุบัน องค์กรซอฟต์แวร์ประมาณ 2 ใน 3 ถูกจัดอยู่ใน CMM ระดับที่ 1 ซึ่งเป็นลักษณะขององค์กรที่มีความวุ่นวาย ไม่มีการบริหารจัดการกระบวนการซอฟต์แวร์ที่ดี ดังนั้นจึงเป็นการดีที่จะนำ CMM ระดับที่ 2 มาช่วยปรับปรุงคุณภาพของกระบวนการซอฟต์แวร์ และการเริ่มต้นจะง่ายขึ้นถ้าเริ่มในขณะที่องค์กรยังเป็นองค์กรขนาดเล็ก มีทีมงานจำนวนไม่มาก แต่ตามที่ได้กล่าวไว้ตอนต้นว่า CMM มุ่งเน้นสำหรับองค์กรขนาดใหญ่ ตัวอย่างที่เห็นได้ชัดเช่น ในระดับ 2 CMM มีบทบาทหรือกลุ่มที่ต้องรับผิดชอบแตกต่างกันมากกว่า 25 บทบาท ซึ่งในองค์กรขนาดเล็กไม่มีทีมงานที่มากพอจะทำได้ทุกบทบาท ดังนั้น CMM จะต้องถูกลดขนาดลงให้สามารถใช้ได้อย่างเหมาะสมกับองค์กร และเมื่อองค์กรโตขึ้น CMM ก็จะต้องมีบทบาทเพิ่มขึ้น

1.2 วัตถุประสงค์ของโครงการ

การดำเนินงานโครงการนี้ มีวัตถุประสงค์เพื่อ

1) นำเสนอแนวทางในการพัฒนาซอฟต์แวร์อย่างมีคุณภาพ
2) นำแนวทางที่ได้จากการศึกษามาประยุกต์ใช้กับโครงการพัฒนาซอฟต์แวร์ได้จริง
3) สามารถประยุกต์ใช้ CMM กับองค์กรหรือโครงการขนาดเล็ก และดำเนินการพัฒนาซอฟต์แวร์ได้บรรลุตามวัตถุประสงค์ของ CMM ระดับที่ 2
4) ได้ทราบถึงปัญหา อุปสรรคในการนำ CMM ระดับที่ 2 มาใช้ในการพัฒนาซอฟต์แวร์ เพื่อเป็นแนวทางในการนำไปประยุกต์ใช้ต่อไป

1.3 ขอบเขตของการดำเนินงาน

การดำเนินงานโครงการ จะทำการศึกษา CMM ระดับที่ 2 และนำมาเป็นแนวทางในการจัดทำเอกสารมาตรฐานต่างๆ ที่จำเป็น และนำเอกสารที่ได้ไปประยุกต์ใช้งานจริงกับโครงการพัฒนาระบบงานในกรณีศึกษา คือ ระบบบริหารคลังวัตถุดิบ เพื่อให้ได้ทราบถึงปัญหา อุปสรรค และแนวทางการนำ CMM ระดับที่ 2 ไปใช้กับโครงการพัฒนาซอฟต์แวร์ได้จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขั้นตอนและวิธีการดำเนินงานโครงการ

ในการดำเนินการศึกษาวิจัยนั้น ผู้ดำเนินงานมีขั้นตอนในการดำเนินงานโครงการดังนี้คือ

1) ศึกษาค้นคว้าข้อมูลเกี่ยวกับแบบจำลองวุฒิภาวะความสามารถ (CMM) โดยเน้นศึกษาใน CMM ระดับที่ 2 จากหนังสือ วารสาร และอินเตอร์เน็ต
2) ศึกษาค้นคว้าในเรื่องการบริหารจัดการโครงการซอฟต์แวร์ รวมถึงเอกสารและมาตรฐานต่างๆ ที่จำเป็นในการพัฒนาซอฟต์แวร์
3) รวมรวมข้อมูลต่างๆ ที่ได้ศึกษา มาประยุกต์เข้าด้วยกันกับ key process area ของ CMM ระดับที่ 2
4) นำแนวทางที่ได้มาประยุกต์ใช้กับกรณีศึกษา คือการจัดทำระบบบริหารคลังวัตถุดิบ (Inventory System)
5) สรุปผลการดำเนินงานโครงการ และข้อเสนอแนะ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

เมื่อดำเนินงานโครงการเป็นผลสำเร็จแล้ว คาดว่าจะได้รับประโยชน์ดังนี้คือ

1) ได้ความรู้เกี่ยวกับ CMM ซึ่งเป็นแนวทางในการพัฒนาซอฟต์แวร์อย่างมีคุณภาพ
2) สามารถนำ CMM ไปประยุกต์ใช้กับโครงการพัฒนาซอฟต์แวร์ในองค์กรหรือโครงการขนาดเล็กได้จริง
3) สามารถนำแนวทางที่ได้พัฒนาไปใช้กับโครงการในลักษณะที่คล้ายคลึงกัน และทำให้โครงการซอฟต์แวร์นั้นมีกระบวนการพัฒนาซอฟต์แวร์อย่างมีคุณภาพ ตามเป้าหมายของ CMM ระดับที่ 2 คือระดับที่มีการทำซ้ำได้ (Repeatable)
4) เป็นแนวทางในการศึกษาวิจัยทางด้านวิศวกรรมซอฟต์แวร์ต่อไปในอนาคต

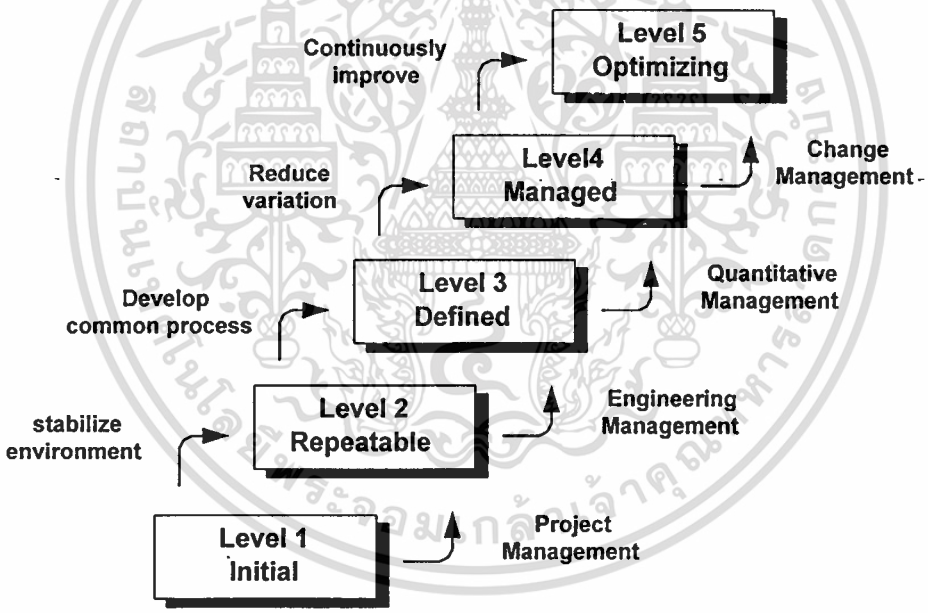
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

แบบจำลองวุฒิภาวะความสามารถ (Capability Maturity Model)

2.1 ภาพรวมของแบบจำลองวุฒิภาวะความสามารถ

Capability Maturity Model (CMM) (แบบจำลองวุฒิภาวะความสามารถ) ถูกพัฒนาขึ้นมาโดย SEI (สถาบันวิศวกรรมซอฟต์แวร์) ของมหาวิทยาลัยคาร์เนกี เมลอน ซึ่งเป็นแบบจำลองสำหรับวัดว่าหน่วยงานที่ทำหน้าที่พัฒนาซอฟต์แวร์นั้น มีความสามารถและได้บรรลุวุฒิภาวะในการทำงานมากน้อยเพียงใดโดยที่ CMM จะอธิบายถึงการปรับปรุงองค์กรซอฟต์แวร์จากองค์กรที่ไม่เติบโตมาเป็นองค์กรที่เติบโตและมีวินัยในตัวเอง โดยจะแบ่งระดับการเติบโตออกเป็น 5 ระดับ ดังนี้ คือ



รูปที่ 2.1 แสดง 5 ระดับของวุฒิภาวะ

2.1.1 ระดับของวุฒิภาวะ

CMM ได้จัดแบ่งวุฒิภาวะออกเป็น 5 ระดับ ซึ่งแต่ละระดับจะเป็นพื้นฐานให้กับระดับที่สูงขึ้นต่อเนื่องกันไปเรื่อยๆ โดยระดับเหล่านี้จะบอกถึงขนาดของการวัดการเติบโตของกระบวนการทางซอฟต์แวร์ขององค์กร และประเมินศักยภาพของกระบวนการทางซอฟต์แวร์ อีกทั้งยังช่วยให้องค์กรรู้ว่าควรจะปรับปรุงอะไรก่อนเป็นลำดับก่อนหลังด้วย โดยทั้ง 5 ระดับมีดังต่อไปนี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระดับที่ 1 ระดับเริ่มต้น (Initial Level)

กระบวนการซอฟต์แวร์จะมีลักษณะไม่ได้วางแผนไว้ล่วงหน้า และมักจะยุ่งเหยิงสับสน และความสำเร็จของโครงการจะขึ้นอยู่กับความสามารถของแต่ละบุคคล

ระดับที่ 2 ระดับทำซ้ำได้ (Repeatable Level)

เริ่มมีกระบวนการบริหารจัดการโครงการขั้นพื้นฐาน เพื่อติดตามเรื่องต้นทุน ตารางการทำงาน และการปฏิบัติงาน ระเบียบปฏิบัติของกระบวนการที่จำเป็นถูกนำมาใช้กับโครงการที่มีลักษณะคล้ายๆ กับที่เคยทำมาก่อน

ระดับที่ 3 ระดับที่มีกรรมวิธีชัดเจน (Defined Level)

ขั้นตอนกระบวนการซอฟต์แวร์สำหรับการจัดการและกิจกรรมทางวิศวกรรมซอฟต์แวร์ถูกจัดทำเป็นเอกสารมาตรฐาน และรวบรวมไปเป็นขั้นตอนกระบวนการซอฟต์แวร์มาตรฐานขององค์กร ทุกโครงการใช้ขั้นตอนการะบวนการซอฟต์แวร์มาตรฐานที่ผ่านการอนุมัติและจัดทำขึ้นโดยเฉพาะ เพื่อพัฒนาและบำรุงรักษาซอฟต์แวร์

ระดับที่ 4 ระดับจัดการ (Managed Level)

มีการเก็บรวบรวมรายละเอียดมาตรฐานการวัดกระบวนการซอฟต์แวร์และคุณภาพของผลิตภัณฑ์เป็นอย่างดี ขั้นตอนซอฟต์แวร์และผลิตภัณฑ์เป็นที่เข้าใจและสามารถควบคุมได้ .

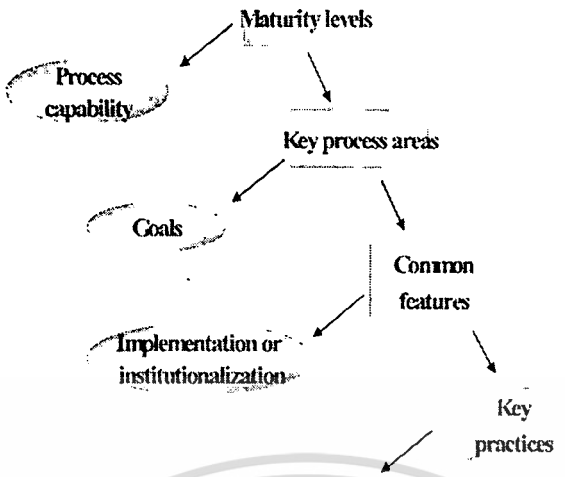
ระดับที่ 5 ระดับเหมาะที่สุด (Optimizing Level)

การปรับปรุงกระบวนการเป็นไปอย่างต่อเนื่องโดยได้รับการตอบรับจากกระบวนการและจากแนวคิดที่เปลี่ยนแปลงและเทคโนโลยีต่างๆ

2.1.2 โครงสร้างของ CMM

วุฒิภาวะในแต่ละระดับ จะประกอบไปด้วย กลุ่มกระบวนการหลัก (key process area) หลายๆ ตัว ซึ่งแต่ละตัวจะถูกจัดแบ่งออกเป็น 5 ส่วนเรียกว่า ลักษณะร่วม (common feature) และในแต่ละ commom feature ก็จะประกอบด้วย วิธีปฏิบัติหลัก (key practices) ซึ่งเป็นข้อกำหนดในระดับของการทำงานจริงๆ ซึ่งทั้งหมดนี้เมื่อทำงานร่วมกันแล้วก็จะบรรลุเป้าหมายของกลุ่มกระบวนการหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 โครงสร้างของ CMM

โดยโครงสร้างแต่ละตัวมีรายละเอียดดังนี้คือ

ระดับวุฒิภาวะ (Maturity Levels)

วุฒิภาวะ ทั้ง 5 ระดับ จะเป็นโครงสร้างที่อยู่บนสุดของ CMM โดยแต่ละระดับจะบอกถึงระดับของความสามารถของกระบวนการ

กลุ่มกระบวนการหลัก (Key Process Areas)

ในแต่ละระดับวุฒิภาวะ (ยกเว้นระดับที่ 1) จะประกอบด้วย กลุ่มกระบวนการหลัก หลายตัว ซึ่งจะเป็นตัวที่บอกว่าองค์กรควรจะมุ่งเป้าหมายไปที่ใดในการที่จะพัฒนากระบวนการซอฟต์แวร์ กลุ่มกระบวนการหลักจะเป็นตัวบอกว่าควรทำ อย่างไรเพื่อไปถึงระดับวุฒิภาวะที่ต้องการ

วิธีปฏิบัติหลัก (Key Practices)

กลุ่มกระบวนการหลักแต่ละตัว จะอธิบายในเทอมของ วิธีการปฏิบัติหลัก โดยที่วิธีการปฏิบัติหลักจะอธิบายถึงกิจกรรมและโครงสร้างพื้นฐานที่จะดำเนินการเพื่อให้กลุ่มกระบวนการหลักเป็นไปอย่างมีประสิทธิภาพ

ลักษณะร่วม (Common feature)

เพื่อให้เข้าใจได้ง่าย วิธีปฏิบัติที่อธิบายกลุ่มกระบวนการหลัก จะถูกจัดวางโครงสร้างโดยลักษณะร่วม 5 อย่าง ดังนี้คือ

พันธกิจในการดำเนินงาน (commitment to perform) ใช้ระบุสิ่งที่บริษัทหรือหน่วยงานจะต้องทำเพื่อให้การพัฒนาซอฟต์แวร์ก้าวไปสู่วุฒิภาวะที่สูงขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสามารถในการดำเนินงาน (Ability to Perform) อธิบายเงื่อนไขที่ต้องมีก่อนที่จะเริ่มโครงการหรือจะเริ่มทำกระบวนการซอฟต์แวร์ องค์กรมีความพร้อมหรือความสามารถแค่ไหนในการดำเนินการ

กิจกรรมที่ต้องดำเนินการ (Activities Performed) อธิบายกิจกรรม บทบาท และกระบวนการที่จำเป็นในการบรรลุเป้าหมายของกลุ่มกระบวนการหลัก

การวัดและการวิเคราะห์ (Measurement and Analysis) เป็นการวัดผลขั้นพื้นฐานซึ่งผลที่ได้จะนำมาวิเคราะห์เพื่อใช้ในการตัดสินใจเกี่ยวกับสถานะการดำเนินงานในกระบวนการ

การตรวจสอบการดำเนินการ (Verifying Implementation) ขั้นตอนเพื่อที่จะแน่ใจได้ว่ากิจกรรมนั้นได้ถูกทำตามกระบวนการที่ได้กำหนดไว้

แบบจำลองวุฒิภาวะความสามารถ ระดับที่ 2

องค์กรทั่วไปได้ถูกจัดระดับวุฒิภาวะอยู่ในระดับที่ 1 คือระดับที่ยังไม่มีกระบวนการที่ชัดเจนในการพัฒนาซอฟต์แวร์ การนำ CMM มาประยุกต์ใช้งานจะต้องผ่าน CMM ระดับที่ 2 ก่อน ซึ่งถือว่าเป็นระดับที่สำคัญที่สุดเพราะเป็นระดับเริ่มต้น โดยจะเน้นในเรื่องการบริหารจัดการโครงการขั้นพื้นฐาน ซึ่งจะประกอบไปด้วย การติดตามการใช้ต้นทุน การจัดตารางเวลาและหน้าที่ การจัดทำกระบวนการดังกล่าวอย่างมีขั้นตอนเป็นสิ่งที่จำเป็น เมื่อมีการพัฒนาโปรแกรมในลักษณะที่คล้ายกันจะได้นำกระบวนการนั้นมาทำซ้ำได้ โดยกลุ่มกระบวนการหลักของ CMM Level 2 ประกอบด้วย

RM: การบริหารจัดการความต้องการ (Requirement Management)

SPP: การวางแผนกระบวนการซอฟต์แวร์ (Software Project Planning)

SPTO: การติดตามดูแลโครงการ (Software Project Tracking and Oversight)

SSM: การจัดการผู้รับงานรายย่อย (Software Subcontract Management)

SQA: การประกันคุณภาพซอฟต์แวร์ (Software Quality Assurance)

SCM: การจัดการ configuration ของซอฟต์แวร์ (Software Configuration Management)

โดย KPA แต่ละตัวมีรายละเอียดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 RM: Requirement Management

เป็นกระบวนการสำคัญในการควบคุมดูแลความต้องการของผู้ใช้ไม่ให้ขยายหรือมีการเปลี่ยนแปลงมากเกินไป นอกจากนั้นยังดูแลให้การดำเนินการพัฒนาซอฟต์แวร์สอดคล้องกับข้อกำหนดความต้องการด้วย

เป้าหมาย

- **มีการควบคุมข้อกำหนดด้านซอฟต์แวร์เพื่อให้เกิดกิจกรรมทางด้านวิศวกรรมซอฟต์แวร์ และการจัดการซอฟต์แวร์ได้**
- **มีการดูแลให้แผนซอฟต์แวร์ ผลิตภัณฑ์ และ กิจกรรมต่าง ๆ สอดคล้องกับข้อกำหนดความต้องการ**

พันธกิจการดำเนินงาน

- **(C1) โครงการจะต้องดำเนินงานตามนโยบายที่เขียนไว้ ในเรื่องเกี่ยวกับการจัดการกับความต้องการของระบบในการพัฒนาซอฟต์แวร์**

ความสามารถในการดำเนินงาน

- **(AB1) ในแต่ละโครงการ จะต้องรับผิดชอบในเรื่องของการวิเคราะห์ความต้องการของระบบ รวมทั้งเรื่องของ ฮาร์ดแวร์ ซอฟต์แวร์ และส่วนประกอบอื่นๆ ด้วย**
- **(AB2) ความต้องการของระบบจะต้องถูกจัดทำเป็นเอกสาร**
- **(AB3) การจัดการเกี่ยวกับความต้องการจะต้องมีทรัพยากรและเงินทุนที่เพียงพอ**
- **(AB4) สมาชิกของกลุ่มวิศวกรรมซอฟต์แวร์และกลุ่มอื่นๆ ที่เกี่ยวข้องจะต้องได้รับการอบรมเพื่อที่จะทำกิจกรรมของการจัดการในเรื่องของความต้องการได้**

กิจกรรมที่ต้องดำเนินการ

- **(AC1)กลุ่มวิศวกรรมซอฟต์แวร์จะต้องทบทวนความต้องการก่อนที่รวบรวมเข้าไว้ในโครงการ**
- **(AC2)กลุ่มวิศวกรรมซอฟต์แวร์จะใช้ความต้องการนั้นเป็นพื้นฐานในการจัดทำแผนการพัฒนา ผลผลิต และกิจกรรมต่างๆ**
- **(AC3)การเปลี่ยนแปลงความต้องการใดๆ จะถูกทบทวนและรวบรวมเข้าไว้ในโครงการ**

การวัดและการวิเคราะห์

- **(M1) ต้องมีการวัดผลเพื่อใช้ในการตัดสินใจสถานะของกิจกรรมในการจัดการความต้องการ**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจสอบการดำเนินการ

- (V1) กิจกรรมสำหรับจัดการความต้องการจะถูกทบทวนพร้อมๆ กับผู้จัดการอาวุโส โดยมีระยะเวลาที่กำหนดไว้ชัดเจน
- (V2) กิจกรรมสำหรับจัดการความต้องการจะถูกทบทวนกับผู้จัดการโครงการ โดยมีระยะเวลาที่กำหนดไว้ชัดเจนและเมื่อมีเหตุการณ์ใดๆ เกิดขึ้น
- (V3) กลุ่มประกันคุณภาพซอฟต์แวร์จะต้องทบทวนและ/หรือตรวจสอบกิจกรรมและผลของการจัดการความต้องการและจัดทำรายงานผลการตรวจสอบ

2.2.2 SPP: Software Project Planning

เป็นการสร้างแผนงานที่เหมาะสมสำหรับการดำเนินการในเรื่องวิศวกรรมซอฟต์แวร์และการจัดการโครงการซอฟต์แวร์

เป้าหมาย

- มีการจัดทำเอกสารในเรื่องการประมาณการเวลาในการพัฒนาซอฟต์แวร์เพื่อใช้ในการวางแผนและติดตามโครงการ
- มีการวางแผนและจัดทำเอกสารในเรื่องของกิจกรรมของโครงการซอฟต์แวร์และพันธกิจต่างๆ
- พันธกิจทุกเรื่องของโครงการ จะต้องได้รับความเห็นชอบจากกลุ่มและบุคคลที่เกี่ยวข้อง

พันธกิจในการดำเนินงาน

- (C1) ผู้จัดการโครงการจะต้องรับผิดชอบในการเจรจาตกลงในพันธกิจต่างๆ และเป็นผู้จัดทำแผนการพัฒนาโครงการ
- (C2) โครงการจะต้องดำเนินงานตามนโยบายที่ได้เขียนไว้ในเรื่องการวางแผนโครงการซอฟต์แวร์

ความสามารถในการดำเนินงาน

- (AB1) พันธกิจที่ได้จัดทำเป็นเอกสารและผ่านการอนุมัติแล้วจะต้องมีปรากฏอยู่ในโครงการซอฟต์แวร์
- (AB2) ต้องมีการมอบหมายผู้รับผิดชอบในการจัดทำแผนการพัฒนาซอฟต์แวร์
- (AB3) ต้องมีทรัพยากรและเงินทุนที่เพียงพอในการดำเนินการวางแผนโครงการซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- (AB4) ผู้จัดการโครงการ วิศวกรซอฟต์แวร์ และทุกๆ คนที่มีส่วนเกี่ยวข้องในการวางแผนพัฒนาซอฟต์แวร์จะต้องได้รับการอบรมในเรื่องการประมาณการและกระบวนการวางแผนในด้านที่แต่ละคนต้องรับผิดชอบ

กิจกรรมที่ต้องดำเนินการ

- (AC1) กลุ่มวิศวกรรมซอฟต์แวร์ต้องมีส่วนร่วมในการจัดทำ proposal ของโครงการ
- (AC2) การวางแผนการพัฒนาซอฟต์แวร์จะต้องเริ่มต้นและทำคู่ขนานไปพร้อมกับการวางแผนในภาพรวมของโครงการ
- (AC3) กลุ่มวิศวกรรมซอฟต์แวร์ต้องเข้าร่วมกับกลุ่มอื่นๆ ที่มีส่วนเกี่ยวข้องในการวางแผนโครงการในภาพรวมจนกว่าจะจบโครงการ
- (AC4) พันธกิจต่างๆของโครงการซอฟต์แวร์ที่ทำกับบุคคลหรือกลุ่มใดๆ ภายนอกองค์กร ต้องถูกทบทวนกับผู้จัดการอาวุโสตามกระบวนการที่กำหนดไว้
- (AC5) ต้องมีการระบุวัฏจักรของซอฟต์แวร์ที่กำหนดขอบเขตขั้นตอนของการจัดการไว้ก่อน
- (AC6) แผนการพัฒนาซอฟต์แวร์ของโครงการถูกพัฒนาตามกระบวนการที่กำหนดไว้
- (AC7) แผนของโครงการซอฟต์แวร์ต้องถูกจัดทำเป็นเอกสาร
- (AC8) ต้องมีการกำหนดผลผลิตของซอฟต์แวร์ที่จำเป็นในการสร้างและบำรุงรักษาการควบคุม
- (AC9) การประมาณการขนาดของผลผลิตซอฟต์แวร์ (หรือการเปลี่ยนแปลงขนาดของผลผลิตซอฟต์แวร์) จะถูก อ้างอิงจากกระบวนการที่กำหนดไว้
- (AC10) การประมาณการสำหรับแรงงานและค่าใช้จ่ายในโครงการซอฟต์แวร์จะถูกอ้างอิงจากกระบวนการที่กำหนดไว้
- (AC11) การประมาณการสำหรับวิกฤตทรัพยากรทางคอมพิวเตอร์จะถูกอ้างอิงจากกระบวนการที่กำหนดไว้
- (AC12) ตารางเวลาของโครงการซอฟต์แวร์จะถูกอ้างอิงจากกระบวนการที่กำหนดไว้
- (AC13) ต้องมีการระบุและจัดทำเป็นเอกสารในเรื่องของความเสี่ยงที่อาจจะเกิดขึ้นจากต้นทุน ทรัพยากร ตารางเวลา และมุมมองทางด้านเทคนิคของโครงการ
- (AC14) ต้องมีการจัดเตรียมแผนสำหรับการอำนวยความสะดวกและเครื่องมือต่างๆ สำหรับการดำเนินการวิศวกรรมซอฟต์แวร์
- (AC15) ข้อมูลของการวางแผนโครงการต้องถูกบันทึกไว้ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวัดและการวิเคราะห์

- (M1) ต้องมีการวัดผลเพื่อใช้ในการตัดสินใจเกี่ยวกับสถานะของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์

การตรวจสอบการดำเนินการ

- (V1) กิจกรรมสำหรับการวางแผนโครงการจะต้องถูกทบทวนกับผู้จัดการอาวุโสตามช่วงเวลาที่กำหนด
- (V2) กิจกรรมสำหรับการวางแผนโครงการจะต้องถูกทบทวนกับผู้จัดการโครงการตามช่วงเวลาที่กำหนดและเมื่อมีเหตุการณ์ใดๆ
- (V3) กลุ่มประกันคุณภาพซอฟต์แวร์จะต้องทบทวนและ/หรือตรวจสอบกิจกรรมและผลงานที่ได้สำหรับการวางแผนโครงการและรายงานผลการตรวจสอบ

2.2.3 SPTO: Software Project Tracking and Oversight

เป็นการติดตามดูแลให้การดำเนินงานตามโครงการได้ผลดี ผู้จัดการสามารถดำเนินการใดๆ ก็ได้เมื่อผลงานถึงระดับใดระดับหนึ่งตามแผนงานโครงการ

เป้าหมาย

- มีการติดตามผลลัพธ์และประสิทธิภาพของงานให้เป็นไปตามแผนงานโครงการที่ได้วางไว้
- เมื่อพบว่าผลที่ออกมามีข้อผิดพลาดไปจากแผนงานที่วางไว้ ต้องมีการจัดการและแก้ไขให้ถูกต้อง
- การเปลี่ยนแปลงใดๆ ของพันธกิจจะต้องได้รับความเห็นชอบจากผู้เกี่ยวข้องทุกคน

พันธกิจในการดำเนินการ

- (C1) ผู้จัดการโครงการต้องรับผิดชอบต่อกิจกรรมและผลลัพธ์ที่ได้ของโครงการ
- (C2) โครงการจะต้องดำเนินไปตามนโยบายขององค์กรที่ได้เขียนไว้ในเรื่องของการติดตามดูแลโครงการ

ความสามารถในการดำเนินการ

- (AB1) แผนการพัฒนาโครงการซอฟต์แวร์ต้องถูกจัดทำเป็นเอกสารและต้องผ่านการอนุมัติ
- (AB2) ผู้จัดการโครงการต้องถูกมอบหมายอย่างชัดเจนให้รับผิดชอบต่อผลผลิตซอฟต์แวร์และกิจกรรมต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- (AB3) ต้องมีทรัพยากรและเงินทุนที่เพียงพอสำหรับการติดตามผลการดำเนินงานของโครงการซอฟต์แวร์
- (AB4) ผู้จัดการโครงการจะต้องได้รับการอบรมในเรื่องของการจัดการในด้านเทคนิคและด้านบุคคล ในแง่ที่เกี่ยวกับโครงการซอฟต์แวร์
- (AB5) ผู้จัดการโครงการที่มีชื่อในเบรรทัดแรกต้องเป็นผู้ควบคุมดูแลด้านเทคนิคของโครงการซอฟต์แวร์

กิจกรรมที่ต้องดำเนินงาน

- (AC1) แผนการพัฒนาซอฟต์แวร์ที่ได้จัดทำเป็นเอกสารแล้ว จะถูกนำมาใช้ในการติดตามผลการดำเนินกิจกรรมและสถานะภาพในการติดต่อสื่อสาร
- (AC2) แผนการพัฒนาซอฟต์แวร์จะถูกปรับปรุงให้เหมาะสมตามกระบวนการที่กำหนดไว้
- (AC3) ข้อตกลงและการเปลี่ยนแปลงข้อตกลงใดๆ ของโครงการที่ทำกับบุคคลหรือกลุ่มภายนอกองค์กรจะต้องถูกนำมาพิจารณาร่วมกับผู้จัดการอาวุโสตามกระบวนการที่กำหนดไว้
- (AC4) การอนุมัติการเปลี่ยนแปลงพันธกิจใดๆ ที่มีผลต่อโครงการจะต้องแจ้งให้สมาชิกทุกคนในกลุ่มวิศวกรรมซอฟต์แวร์หรือกลุ่มอื่นๆ ที่เกี่ยวข้องรับรู้
- (AC5) ขนาดของผลงานซอฟต์แวร์ (หรือขนาดของการเปลี่ยนแปลงผลงานซอฟต์แวร์) จะต้องถูกติดตามและแก้ไขได้ถ้ามีความจำเป็น
- (AC6) แรงงานและต้นทุนในโครงการซอฟต์แวร์จะต้องถูกติดตามและแก้ไขได้ถ้ามีความจำเป็น
- (AC7) ทรัพยากรทางด้านคอมพิวเตอร์ที่เป็นวิกฤตจะต้องถูกติดตามและแก้ไขได้ถ้ามีความจำเป็น
- (AC8) ตารางการทำงานจะต้องถูกติดตามและแก้ไขได้ถ้ามีความจำเป็น
- (AC9) กิจกรรมทางเทคนิคด้านวิศวกรรมซอฟต์แวร์จะต้องถูกติดตามและแก้ไขได้ถ้ามีความจำเป็น
- (AC10) ความเสี่ยงของซอฟต์แวร์ในด้านต้นทุน ทรัพยากร ตารางเวลาและทางด้านเทคนิคจะต้องถูกติดตาม
- (AC11) ข้อมูลการวัดผลที่เป็นจริงและข้อมูลการวางแผนใหม่ของโครงการซอฟต์แวร์จะต้องถูกบันทึกไว้ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- (AC12) กลุ่มวิศวกรรมซอฟต์แวร์เป็นผู้ควบคุมการพิจารณาภายในตามช่วงเวลาที่กำหนดในการติดตามความก้าวหน้าในด้านเทคนิค แผนงาน ประสิทธิภาพงาน และเรื่องที่เกี่ยวกับแผนงานโครงการ
- (AC13) การพิจารณาอย่างเป็นทางการในเรื่องผลสำเร็จและผลที่ได้รับของโครงการถูกควบคุมโดย milestone ที่เลือกตามกระบวนการที่ได้กำหนดไว้

การวัดและการวิเคราะห์

- (M1) ต้องมีการวัดผลเพื่อใช้ในการตัดสินใจสถานะของการติดตามดูแลและกิจกรรมต่างๆ

การตรวจสอบการดำเนินงาน

- (V1) กิจกรรมสำหรับการติดตามดูแลต้องนำมาพิจารณาร่วมกับผู้จัดการอาวุโสในช่วงเวลาที่กำหนด
- (V2) กิจกรรมสำหรับการติดตามดูแลต้องนำมาพิจารณาร่วมกับผู้จัดการโครงการในช่วงเวลาที่กำหนดและเมื่อเกิดเหตุการณ์ใดๆ
- (V3) กลุ่มประกันคุณภาพซอฟต์แวร์จะต้องทบทวนและ/หรือตรวจสอบกิจกรรมและผลงานของการติดตามดูแลโครงการ และรายงานผลการตรวจสอบ

2.2.4 Software Subcontract Management

เป็นการเลือกผู้รับงานรายย่อยที่มีคุณภาพและสามารถจัดการตรวจสอบผู้รับงานรายย่อยได้อย่างมีประสิทธิภาพ

เป้าหมาย

- ผู้รับงานหลักและผู้รับงานย่อยตกลงในพันธกิจ
- ผู้รับงานหลักติดตามผลลัพธ์การดำเนินงานของผู้รับงานย่อยว่าเป็นไปตามพันธกิจ
- ผู้รับงานหลักและผู้รับงานย่อยมีการสื่อสารติดต่อกันอย่างต่อเนื่อง ไม่ขาดหาย
- ผู้รับงานหลักติดตามสมรรถนะของผู้รับงานรายย่อยว่าเป็นไปตามพันธกิจ

พันธกิจในการดำเนินงาน

- (C1) โครงการจะต้องดำเนินงานตามนโยบายที่เขียนไว้ในเรื่องขงการจัดการกับผู้รับงานรายย่อย
- (C2) ต้องมีมอบหมายให้ผู้จัดการการรับงานย่อยเป็นผู้รับผิดชอบในเรื่องของการจัดทำและการจัดการในเรื่องของผู้รับงานย่อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสามารถในการดำเนินงาน

- (AB1) ต้องมีทรัพยากรและเงินทุนที่เพียงพอในการคัดเลือกผู้รับงานรายย่อยและการจัดการเกี่ยวกับการดำเนินงานผู้รับงานรายย่อย
- (AB2) ผู้จัดการโครงการและผู้ที่มีส่วนเกี่ยวข้องในเรื่องของการจัดทำและการจัดการผู้รับงานรายย่อยจะต้องได้รับการอบรม
- (AB3) ผู้จัดการโครงการและผู้ที่มีส่วนเกี่ยวข้องในเรื่องของการจัดการกับผู้รับงานรายย่อยจะต้องรับผิดชอบในด้านเทคนิคของผู้รับงานรายย่อยด้วย

กิจกรรมที่ต้องดำเนินการ

- (AC1) ต้องมีการระบุและวางแผนเกี่ยวกับงานที่จะส่งมอบให้ผู้รับงานรายย่อยตามกระบวนการที่กำหนดไว้
- (AC2) การคัดเลือกผู้รับงานรายย่อยจะต้องอยู่ในพื้นฐานของการประมูลงานที่ผู้รับงานสามารถทำได้ตามกระบวนการที่กำหนดไว้
- (AC3) พันธกิจต่างๆ ระหว่างผู้รับงานหลักและผู้รับงานรายย่อยจะใช้เป็นพื้นฐานในการจัดการผู้รับงานรายย่อย
- (AC4) แผนการพัฒนาซอฟต์แวร์ของผู้รับงานย่อยที่ได้จัดทำเป็นเอกสารไว้ จะต้องถูกพิจารณาและอนุมัติโดยผู้รับงานหลัก
- (AC5) แผนการพัฒนาซอฟต์แวร์ของผู้รับงานรายย่อยที่ได้จัดทำเป็นเอกสารและผ่านการอนุมัติแล้วจะถูกใช้ในการติดตามการดำเนินงานของกิจกรรมทางซอฟต์แวร์และสถานะในการติดต่อสื่อสาร
- (AC6) การเปลี่ยนแปลงใดๆ ที่เกิดขึ้นกับข้อความในการดำเนินงาน เงื่อนไขต่างๆ และพันธกิจจะต้องได้รับการตัดสินใจตามกระบวนการที่กำหนดไว้
- (AC7) การจัดการของผู้รับงานหลักจะควบคุมการพิจารณาสถานะและการประสานงานตามช่วงเวลาที่กำหนด
- (AC8) ต้องมีการพิจารณาในด้านเทคนิคและการเปลี่ยนแปลงภายในต่างๆ กับผู้รับงานรายย่อยตามช่วงเวลาที่ได้กำหนดไว้
- (AC9) การพิจารณาอย่างเป็นทางการเพื่อระบุถึงผลสำเร็จและผลงานของผู้รับงานรายย่อยจะถูกควบคุมโดย milestone ที่ได้เลือกไว้ตามกระบวนการที่กำหนด
- (AC10) กลุ่มประกันคุณภาพของผู้รับงานหลักจะต้องตรวจสอบคุณภาพของซอฟต์แวร์ของผู้รับงานรายย่อยตามกระบวนการที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- (AC11) กลุ่มการจัดการ configuration ของซอฟต์แวร์ของผู้รับงานหลักจะต้องตรวจสอบกิจกรรมด้านการจัดการ configuration ของซอฟต์แวร์ของผู้รับงานรายย่อยตามกระบวนการที่กำหนดไว้
- (AC12) ผู้รับงานหลักควบคุมการตรวจรับโดยเป็นส่วนหนึ่งของการส่งมอบงานของผู้รับงานรายย่อยตามกระบวนการที่กำหนดไว้
- (AC13) ผลการดำเนินงานของผู้รับงานรายย่อยจะต้องถูกประเมินตามช่วงเวลาและผลการประเมินจะต้องถูกนำมาพิจารณาร่วมกับผู้รับงานรายย่อย

การวัดและการวิเคราะห์

- (M1) การวัดผลจะถูกจัดทำเพื่อใช้ในการตัดสินสถานะของกิจกรรมสำหรับการจัดการผู้รับงานรายย่อย

การตรวจสอบการดำเนินการ

- (V1) กิจกรรมในการจัดการกับผู้รับงานรายย่อยจะต้องถูกพิจารณาร่วมกับผู้จัดการอาวุโสในช่วงเวลาที่กำหนด
- (V2) กิจกรรมในการจัดการกับผู้รับงานรายย่อยจะต้องถูกพิจารณาร่วมกับผู้จัดการโครงการในช่วงเวลาที่กำหนดและเมื่อเกิดเหตุการณ์ใดๆ
- (V3) กลุ่มประกันคุณภาพของซอฟต์แวร์จะต้องพิจารณาและ/หรือ ตรวจสอบกิจกรรมและผลงานในการจัดการกับผู้รับงานรายย่อยและรายงานผลการตรวจสอบ

2.2.5 SQA: Software Quality Assurance

เป็นการตรวจสอบคุณภาพของผลผลิตทั้งในระหว่างการพัฒนาและก่อนการส่งผลผลิตให้ลูกค้า

เป้าหมาย

- มีการวางแผนกิจกรรมการประกันคุณภาพของซอฟต์แวร์
- มีการตรวจสอบว่ากิจกรรมและ ผลผลิตซอฟต์แวร์นั้นเป็นไปตามข้อกำหนดมาตรฐาน วิธีดำเนินการ และ ข้อกำหนดความต้องการ
- กลุ่มหรือบุคคลที่เกี่ยวข้องต้องได้รับข้อมูลกิจกรรมและผลลัพธ์เกี่ยวกับการประกันคุณภาพซอฟต์แวร์
- กิจกรรมที่ไม่เป็นไปตามที่กำหนดไว้และไม่สามารถตัดสินใจได้ภายในโครงการซอฟต์แวร์จะถูกตัดสินโดยผู้จัดการอาวุโส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พันธกิจในการดำเนินงาน

- (C1)โครงการจะดำเนินงานตามนโยบายขององค์กรที่เขียนไว้ในเรื่องที่เกี่ยวกับการประกันคุณภาพซอฟต์แวร์

ความสามารถในการดำเนินงาน

- (AB1) ต้องมีกลุ่มที่รับผิดชอบในการประสานงานและดำเนินการด้าน SQA ในโครงการ
- (AB2) ต้องมีทรัพยากรและเงินทุนที่เพียงพอในการดำเนินงานกิจกรรมที่เกี่ยวข้องกับ SQA
- (AB3) สมาชิกในกลุ่ม SQA จะต้องได้รับการอบรมในกิจกรรมที่เกี่ยวข้องกับ SQA
- (AB4) สมาชิกในโครงการซอฟต์แวร์จะมีบทบาท มีความรับผิดชอบ และได้รับสิทธิ์และมีคุณค่าต่อกลุ่มของ SQA

กิจกรรมที่ต้องดำเนินการ

- (AC1) ต้องมีการจัดเตรียมแผนงาน SQA สำหรับโครงการซอฟต์แวร์ตามกระบวนการที่กำหนด
- (AC2) กิจกรรมของกลุ่ม SQA จะต้องทำตามแผนงาน SQA
- (AC3) กลุ่ม SQA จะมีส่วนร่วมในการจัดเตรียมและพิจารณาแผนการพัฒนาซอฟต์แวร์ทั้งเรื่องมาตรฐาน และกระบวนการ
- (AC4) กลุ่ม SQA ต้องทบทวนกิจกรรมวิศวกรรมซอฟต์แวร์เพื่อตรวจสอบให้เป็นไปตามมาตรฐานที่กำหนดไว้
- (AC5) กลุ่ม SQA ต้องตรวจสอบผลผลิตงานซอฟต์แวร์ที่ได้เพื่อให้เป็นไปตามมาตรฐานที่กำหนด
- (AC6) กลุ่ม SQA จัดทำรายงานผลของกิจกรรมของกลุ่มวิศวกรรมซอฟต์แวร์ตามช่วงเวลาที่กำหนด
- (AC7) การกระทำใดที่ไม่เป็นไปตามกิจกรรมหรืองานที่กำหนดไว้ในเอกสาร จะต้องถูกจัดการตามกระบวนการที่กำหนดไว้
- (AC8) กลุ่ม SQA ควบคุมการพิจารณากิจกรรมและค้นหากับบุคลากร SQA ของลูกค้าตามความเหมาะสม

การวัดและการวิเคราะห์

- (M1) จะต้องมีการวัดผลเพื่อใช้ในการตัดสินใจเกี่ยวกับต้นทุนและตารางเวลาทำงานของกิจกรรม SQA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจสอบการดำเนินการ

- (V1) กิจกรรม SQA ถูกพิจารณาร่วมกับผู้จัดการอาวุโสตามช่วงเวลาที่กำหนด
- (V2) กิจกรรม SQA ถูกพิจารณาร่วมกับผู้จัดการโครงการตามช่วงเวลาที่กำหนด หรือเมื่อเกิดเหตุการณ์ใดๆ
- (V3) ผู้เชี่ยวชาญที่ไม่ได้อยู่ในกลุ่ม SQA ต้องทำการพิจารณากิจกรรมและผลงานของซอฟต์แวร์ของกลุ่ม SQA ของโครงการตามช่วงเวลาที่กำหนด

2.2.6 SCM: Software Configuration Management

การควบคุมจัดการความเข้ากันได้ของผลผลิตของโครงการซอฟต์แวร์ ตลอดวัฏจักรชีวิตของซอฟต์แวร์

เป้าหมาย

- มีการวางแผนกิจกรรมการจัดการ configuration ของซอฟต์แวร์
- มีการกำหนด ควบคุม ผลผลิตซอฟต์แวร์ที่เลือกไว้ อีกทั้งซอฟต์แวร์นั้นจะต้องมีให้ใช้งานได้
- มีการควบคุมการเปลี่ยนแปลงผลผลิตซอฟต์แวร์
- กลุ่มและบุคคลที่เกี่ยวข้องได้รับการแจ้งสถานภาพและเนื้อหาของ software baseline โดยตลอด

พันธกิจในการดำเนินงาน

- (C1) โครงการดำเนินงานตามนโยบายขององค์กรที่ได้เขียนไว้ในเรื่องของการจัดการ configuration ของซอฟต์แวร์

ความสามารถในการดำเนินงาน

- (AB1) ต้องมีคณะผู้บริหารที่มีสิทธิ์ในการจัดการ baseline ของโครงการซอฟต์แวร์ (ตัวอย่างเช่น SCCB – Software Configuration Control Board)
- (AB2) ต้องมีกลุ่มที่รับผิดชอบในการประสานงานและจัดทำ SCM ในโครงการ
- (AB3) ต้องมีทรัพยากรและเงินทุนที่เพียงพอในการดำเนินการจัดทำกิจกรรมของ SCM
- (AB4) สมาชิกในกลุ่ม SCM ต้องได้รับการอบรมในเรื่องของจุดประสงค์ กระบวนการและกรรมวิธีในการดำเนินงานกิจกรรมของ SCM
- (AB5) สมาชิกของกลุ่มวิศวกรรมซอฟต์แวร์และกลุ่มอื่นๆ ที่เกี่ยวข้องต้องได้รับการอบรมเพื่อสามารถดำเนินการในกิจกรรมของ SCM

กิจกรรมที่ต้องดำเนินการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- (AC1) มีการจัดเตรียมแผนงาน SCM ในแต่ละโครงการซอฟต์แวร์ตามกระบวนการที่กำหนด
- (AC2) แผนงานที่ได้จัดทำเป็นเอกสารและผ่านการอนุมัติแล้วจะใช้เป็นพื้นฐานในการดำเนินงานของกิจกรรม SCM
- (AC3) ระบบห้องสมุดการจัดการ configuration ต้องถูกสร้างขึ้นมาเพื่อใช้เป็นที่เก็บของ software baseline
- (AC4) ผลิตผลซอฟต์แวร์ที่จะถูกแทนที่ภายใต้การจัดการ configuration จะต้องถูกกำหนดไว้
- (AC5) การเปลี่ยนแปลงคำขอใดๆ และปัญหาที่พบจาก configuration item ทุกตัวจะต้องถูกระบุ ถูกบันทึก ถูกพิจารณา ได้รับการอนุมัติ และติดตามผลงานตามกระบวนการที่กำหนด
- (AC6) การเปลี่ยนแปลง baseline จะถูกควบคุมตามขั้นตอนที่กำหนดไว้ในเอกสาร
- (AC7) ผลผลิตจาก software baseline ถูกสร้างและ release จะถูกควบคุมตามกระบวนการที่กำหนดไว้
- (AC8) สถานะของ configuration item จะถูกบันทึกตามกระบวนการที่กำหนดไว้
- (AC9) มีการพัฒนาและจัดทำเอกสารที่เป็นมาตรฐานของกิจกรรม SCM และเนื้อหาของ software baseline สำหรับกลุ่มหรือบุคคลที่เกี่ยวข้อง
- (AC10) การตรวจสอบ software baseline จะถูกควบคุมตามกระบวนที่กำหนดไว้ในเอกสาร

การวัดและการวิเคราะห์

- (M1) จะต้องมีการวัดผลเพื่อใช้ในการตัดสินใจเกี่ยวกับสถานะของกิจกรรม SCM

การตรวจสอบการดำเนินการ

- (V1) กิจกรรม SCM ต้องได้รับการพิจารณาร่วมกับผู้จัดการอาวุโสตามเวลาที่กำหนด
- (V2) กิจกรรม SCM ต้องได้รับการพิจารณาร่วมกับผู้จัดการโครงการตามเวลาที่กำหนดและเมื่อเกิดเหตุการณ์ใดๆ
- (V3) กลุ่ม SCM จะทำการตรวจสอบ software baseline เป็นระยะๆ เพื่อที่จะแน่ใจได้ว่าได้ดำเนินการตามกระบวนการที่กำหนด
- (V4) กลุ่มประกันคุณภาพซอฟต์แวร์พิจารณา และ/หรือตรวจสอบกิจกรรมและผลที่ได้ของ SCM และรายงานผลการตรวจสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การประยุกต์ใช้ CMM ระดับที่ 2 กับองค์กรหรือโครงการขนาดเล็ก

2.3.1 ความหมายและคุณลักษณะขององค์กรหรือโครงการขนาดเล็ก

องค์กร: *"หน่วยหนึ่งๆ ในบริษัทที่ซึ่งมีการจัดการโครงการหลายๆ โครงการ โดยทุกโครงการในองค์กรมีผู้จัดการสูงสุดคนเดียวกันและใช้นโยบายร่วมกัน"*

โครงการ: *"แรงงานที่เน้นในเรื่องการพัฒนาและ/หรือบำรุงรักษาผลิตภัณฑ์หนึ่งๆ โดยทั่วไปแล้วโครงการจะมีเงินทุนของตนเอง มีค่าใช้จ่ายเอง และมีกำหนดเวลาส่งงาน"*

2.3.2 ลักษณะของโครงการและองค์กรขนาดเล็ก

โครงการและองค์กรขนาดเล็ก จะเป็นโครงการที่มีระยะเวลาทำงานสั้น มีทีมงานจำนวนไม่มาก และมีทรัพยากรที่ใช้ในการพัฒนาอย่างจำกัด คนหนึ่งคนอาจจะรับผิดชอบหลายหน้าที่ เช่น อาจจะเป็นทั้งผู้จัดการโครงการและผู้พัฒนาไปด้วย องค์กรลักษณะนี้จะมีโครงสร้างองค์กรที่ไม่มีลำดับขั้นมากนัก (Flat Structure) สามารถแบ่งความรับผิดชอบของแต่ละคนได้อย่างชัดเจน และเนื่องจากไม่มีลำดับขั้นจึงสามารถติดต่อสื่อสารกันได้ง่าย วัฏจักรการพัฒนาจะเป็นไปในลักษณะเขียนโปรแกรมและทดสอบ ส่งเวอร์ชันใหม่ วนอยู่อย่างนี้เรื่อยๆ แทบจะไม่ได้ทำตามแบบจำลองแบบ Waterfall เลย และนอกจากนี้ยังมีการเปลี่ยนแปลงบ่อยครั้งในเรื่องของตลาด สายการผลิต และที่สำคัญคือความอยู่รอดขององค์กรขึ้นอยู่กับความสำเร็จของโครงการ

2.3.3 แบบจำลององค์กรขนาดเล็ก

ในเอกสารโครงการฉบับนี้ จะแบ่งองค์กรขนาดเล็กออกเป็นสามประเภท ตามจำนวนพนักงานและจำนวนผลิตภัณฑ์/โครงการ ที่กำลังพัฒนา

ตารางที่ 2.1 การแบ่งองค์กรขนาดเล็ก

	1-2 คน	3-15 คน	> 15 คน
1 โครงการ	XXS	XS	S
2-5 โครงการ	-	XS	S
> 6 โครงการ	-	S	S

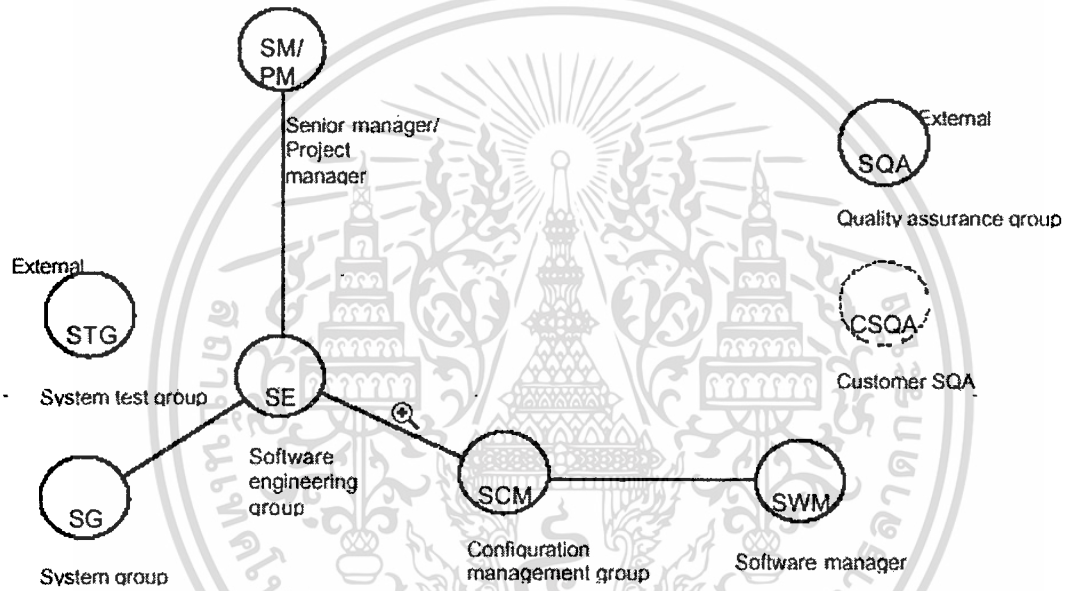
แบบจำลองที่จะกล่าวถึงนี้เหมาะสมกับองค์กรขนาดเล็กที่กำลังเจริญเติบโต เมื่อองค์กรโตขึ้นเราสามารถเพิ่มบทบาทและหน้าที่โดยไม่กระทบกับงานเดิมได้ และสำหรับองค์กรขนาดเล็กที่สุดอาจจะตัดบทบาทหรืองานบางอย่างทิ้งไปได้เลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) องค์กรขนาดเล็กมากมาก (eXtra eXtra Small)

องค์กรขนาดนี้จะมีพนักงานเพียง 1-2 คน และจะมีเพียงโครงการเดียว บทบาทที่เป็นไปได้เป็นไปตามรูปที่ 2.3

เนื่องจากมีพนักงานจำนวนน้อยจึงไม่มีปัญหาในเรื่องงานของแต่ละคน การบริหารจัดการโครงการอาจจะไม่ต้องทำอย่างเป็นทางการมากนัก แต่ไม่ควรจะละเลยในเรื่องของการวางแผนโครงการและการติดตามงาน ซึ่งเป็นบทบาทของผู้จัดการอาวุโสและผู้จัดการโครงการ (SM/PM) ซึ่งในกรณีนี้จะใช้เป็นคนเดียวกัน และอาจจะเป็นวิศวกรซอฟต์แวร์ด้วย (กรณีมีพนักงานคนเดียว)



รูปที่ 2.3 แสดงบทบาทในองค์กรขนาดเล็กมากมาก (XXS)

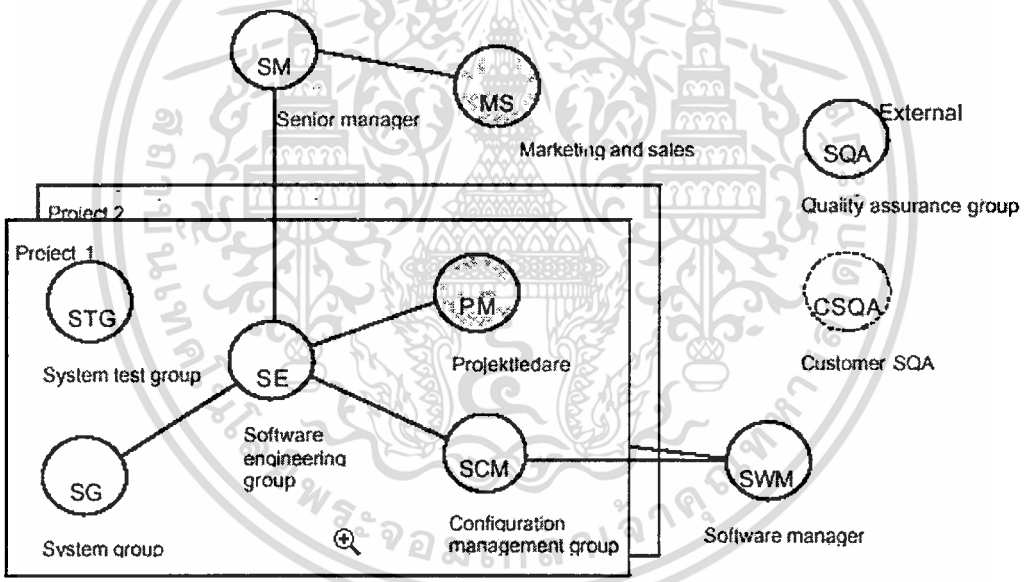
บทบาทอื่นที่จำเป็นต้องมีคือกลุ่มของวิศวกรระบบ (SEG) ที่รับผิดชอบในเรื่องของฮาร์ดแวร์ และซอฟต์แวร์ที่จำเป็นในการพัฒนาระบบ โดยผู้ที่รับบทบาท SE อาจจะเป็นคนดูแลเรื่องนี้ได้ บทบาทของกลุ่มทดสอบระบบ (STG) ก็เป็นอีกบทบาทที่จำเป็น SE อาจจะทดสอบระบบในช่วงของการพัฒนา แต่ STG จะดูแลการทดสอบในภาพรวมของทั้งระบบ ซึ่งเป็นส่วนที่สำคัญมากเพื่อให้ได้ระบบที่มีคุณภาพ ซึ่งในองค์กรขนาดเล็กมากๆ นี้อาจจะต้องอาศัยกลุ่มคนภายนอกช่วยในการทดสอบ อีกบทบาทที่สำคัญคือผู้จัดการซอฟต์แวร์และผู้จัดการองค์ประกอบซอฟต์แวร์ (SWM และ SCM) ซึ่งรับผิดชอบดูแลสภาพแวดล้อมในการพัฒนาระบบ การพัฒนาระบบประจำวันและองค์ประกอบต่างๆ ของการพัฒนาระบบ ซึ่งในองค์กรขนาดนี้อาจจะเป็นคนเดียวกับ SE และบทบาทสุดท้ายที่สำคัญมากคือกลุ่มประกันคุณภาพซอฟต์แวร์ (SQA) ซึ่งไม่ควรเป็นกลุ่มคนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อยู่ในทีมพัฒนา แต่ก็อาจจะเป็นการยากที่จะหาคนมารับผิดชอบในส่วนนี้ถ้าเป็นองค์กรขนาดเล็กมากมาก ดังนั้น อาจจะต้องจ้างจากภายนอก หรืออาจจะให้ลูกค้าเป็นคนรับผิดชอบในส่วนนี้ ส่วนในเรื่องของการจัดการเอกสารนั้น SE จะเป็นผู้รับผิดชอบเอกสารที่เกี่ยวกับโปรแกรม และเอกสารอื่นๆ นอกเหนือจากนั้นอยู่ในความรับผิดชอบของ SM

2) องค์กรขนาดเล็กมาก (eXtra Small)

องค์กรขนาดนี้จะมีพนักงานประมาณ 3-15 คน และมีโครงการ 2-3 โครงการ บริษัทขนาดเล็กมากมาก (XXS) ที่มีการจัดการที่ยุ่งยากขึ้นและมีโครงการรับผิดชอบมากขึ้น อาจจะต้องปรับเปลี่ยนเป็นบริษัทขนาดเล็กมาก (XS) และอาจจะต้องจ้างพนักงานเพิ่มขึ้นเพื่อรองรับบทบาทที่ต้องเพิ่มขึ้น ในองค์กรขนาดนี้จะมีบทบาทใหม่เพิ่มขึ้นมาจากขนาดแรกหรืออาจจะเพิ่มบทบาทให้กับบางบทบาท ดังรูปที่ 2.4

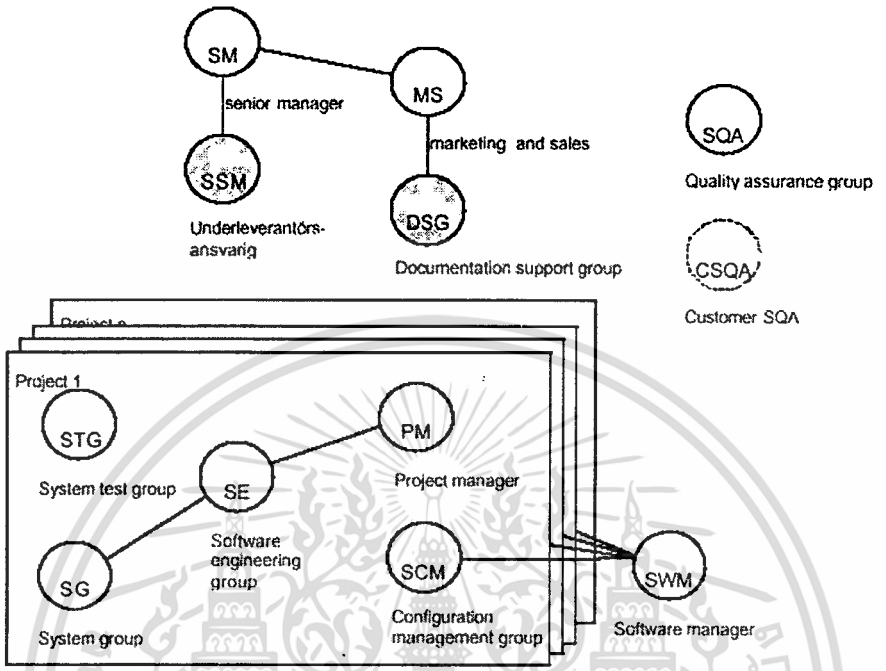


รูปที่ 2.4 แสดงบทบาทในองค์กรขนาดเล็กมาก (XS)

ในระดับนี้ผู้จัดการโครงการ (PM) จะมีบทบาทสำคัญมาก ต้องมี PM ของแต่ละโครงการ ผู้จัดการอาวุโส (SM) ไม่ต้องรับหน้าที่เป็น PM บทบาทของกลุ่มการตลาดและการขาย (MS) เป็นกลุ่มหนึ่งที่ควรเพิ่มขึ้นมาเมื่อองค์กรเติบโตขึ้น และกลุ่มการทดสอบระบบ (STG) ที่เป็นกลุ่มภายนอกในองค์กรระดับที่แล้วก็อาจจำเป็นต้องเป็นกลุ่มคนภายในแต่ละโครงการที่จะต้องตั้งขึ้นมา กลุ่มของผู้จัดการซอฟต์แวร์ (SWM) อาจจะเป็นส่วนหนึ่งของกลุ่ม SCM ในแต่ละโครงการ ส่วนกลุ่ม SQA ที่ดูแลเรื่องกิจกรรมการควบคุมคุณภาพ ก็ยังคงเป็นบทบาทที่ต้องแยกไว้ภายนอกโครงการจะเหมาะสมกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) องค์กรขนาดเล็ก (Small)



รูปที่ 2.5 แสดงบทบาทในองค์กรขนาดเล็ก (S)

องค์กรขนาดนี้จะมีพนักงานประมาณ 16-50 คนและมีผลิตภัณฑ์หลายตัวที่อยู่ในระหว่างการพัฒนา รวมถึงมีหลายเวอร์ชันด้วย ดังรูปที่ 2.5

ในองค์กรขนาดนี้ การจัดทำเอกสารถือว่าเป็นเรื่องสำคัญมาก จึงต้องมีกลุ่มสนับสนุนการทำเอกสาร (DSG) ที่ต้องจัดทำเอกสารทุกประเภทภายในองค์กร รวมถึงเอกสารที่ใช้เป็นมาตรฐานหรือใช้เป็นแม่แบบด้วย และในองค์กรขนาดนี้อาจจะมีกลุ่มการจัดการผู้รับงานรายย่อย (SSM) เพิ่มขึ้นมาด้วย และอาจจะต้องมีบทบาทของ SQA ภายในของแต่ละโครงการด้วย อีกบทบาทที่อาจจะต้องเพิ่มขึ้นมาคือกลุ่ม SCCB ที่อาจจะต้องมีในทุกๆ โครงการเพื่อใช้ในการทบทวนและอนุมัติการเปลี่ยนแปลงทั้งหมดของ baseline แต่ก็อาจจะใช้ SCCB กลุ่มเดียวกันรับผิดชอบหลายโครงการก็ได้

2.3.4 การประยุกต์ใช้ CMM กับองค์กรขนาดเล็ก

การนำ CMM มาประยุกต์ใช้ในองค์กรหรือโครงการขนาดเล็กนั้น มีปัจจัยที่ต้องคำถึง คือ

1) ขอบเขตของการปรับปรุงกระบวนการ

CMM มี 5 ระดับ มีเป้าหมาย 52 เป้าหมาย มีกลุ่มกระบวนการหลัก 18 กลุ่ม และมีกิจกรรม 316 กิจกรรม รวมถึงบทบาทหน้าที่ต่างๆ ที่หลากหลายในแต่ละกิจกรรม จะเห็นได้ชัดเจนว่าในองค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรขนาดเล็ก เราไม่สามารถทำได้ทั้งหมด ดังนั้นเราต้องมีการจำกัดขอบเขตและพิจารณาว่าอะไรบ้างที่เป็นสิ่งจำเป็นต้องทำ อะไรเป็นสิ่งที่สามารถตัดทิ้งไปได้ โดยแนวทางในการแก้ไขปัญหานี้เช่นอาจจะมีการจัดทำ workshop หรือการประเมินเล็กๆ แทนที่จะทำการประเมินอย่างเป็นทางการ หรืออาจจะต้องมองให้เห็นถึงปัญหาที่เราต้องการจะแก้ไขจริงๆ ซึ่งอาจจะเป็นปัญหาในเรื่องของคุณภาพ เรื่องของเวลา ต้นทุน หรืออาจจะเป็นเป้าหมายว่าเราต้องการจะผ่าน CMM ระดับที่สองไปสู่ระดับสูงกว่าเป็นต้น และนอกจากนี้เราไม่ควรจะกังวลเกี่ยวกับระดับความเติบโตของ CMM หรือกลุ่มกระบวนการหลักให้มากเกินไป เพราะจุดประสงค์ในการทำของเราคือปรับปรุงให้กระบวนการซอฟต์แวร์มีคุณภาพยิ่งขึ้น และในการเริ่มต้นควรจะเริ่มจากสิ่งง่ายๆ ก่อนเช่นเริ่มจากปัญหาที่เรารู้ว่าจะแก้ไขอย่างไร เริ่มจากสิ่งที่มีคนเคยทำมาก่อนนี้แล้ว หรือจะใช้ข้อเท็จจริงจากการวัดผลที่เคยมีก่อนหน้านี้มาเป็นตัวช่วย สุดท้ายที่สำคัญคือเราควรจะคิดไปถึงระยะยาวด้วย ไม่ควรมองแต่ให้แก้ปัญหาเฉพาะหน้าอย่างเดียว

2) การจัดทำเอกสาร

ในแต่ละระดับของ CMM จะมีการจัดทำเอกสารเกี่ยวกับนโยบาย แผนงาน และกระบวนการทำงานต่างๆ ตัวอย่างเช่นใน CMM ระดับ 2 เราจะต้องจัดทำนโยบาย 6 ตัว มีการจัดเอกสารแผนงาน 5 แผน และมีเอกสารกระบวนการทำงานอีก 24 กระบวนการ องค์กรขนาดใหญ่อาจจะมีคนเพียงพอในการจัดทำเอกสารทั้งหมดนี้ แต่ในองค์กรขนาดเล็ก การจัดทำเอกสารจำนวนมากเกินไปอาจจะไม่มีประโยชน์มากนัก ทั้งยังจะเป็นการเพิ่มงานและทำให้งานที่ควรจะเสร็จเร็วกลับเสร็จช้าลงไปอีก แนวทางการจัดทำเอกสารในองค์กรขนาดเล็กควรจะเป็นการรวมเอกสาร เราสามารถรวมเอกสารที่มีความคล้ายคลึงกันเข้าด้วยกัน ไม่จำเป็นต้องจัดทำนโยบายของทุกๆ กลุ่มกระบวนการหลักก็ได้ หรือในกรณีการจัดทำแผนกระบวนการซอฟต์แวร์ ก็อาจจะจัดทำเพียงแผนเดียวให้ครอบคลุมหมดทั้งโครงการได้ อีกเรื่องที่ต้องคำนึงถึงคือการจัดทำเอกสารให้เหมาะสมกับขนาดขององค์กร (right- size) เราไม่จำเป็นต้องทำนโยบายหนาเกิน 1-2 หน้าก็ได้ กระบวนการทำงานก็ไม่ต้องละเอียดมากเพราะเราไม่ได้จัดทำคู่มือการใช้งาน หรือการทำเอกสาร SCM สำหรับทีมงานเพียงแค่ 3 คน ก็ไม่ต้องทำให้หนาเกินไปก็ได้ และควรมีการใช้แม่แบบ (template) ในเอกสารที่มีการใช้งานบ่อย นอกจากนี้เราควรจัดทำเอกสารตามลำดับความสำคัญ ควรคำนึงว่าเอกสารตัวไหนที่มีแล้วจะช่วยให้โครงการสำเร็จได้ จัดทำเอกสารที่เราต้องใช้จริงๆ ไม่ใช่เอกสารที่ควรจะมี และนโยบายเอาไว้เขียนหลังสุดก็ได้

3) การสรรหาทรัพยากร

วัฏจักรการนำ CMM เข้ามาใช้จะเป็นไปในลักษณะเริ่มต้นจาก การได้งานมา แล้ววางแผนการพัฒนา จัดตั้งทีมงาน จัดทำเอกสารที่ระบุไว้ใน CMM ตรวจสอบเป็นระยะ รับงานใหม่ แล้วเริ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรงวางแผนการพัฒนาอีกครั้ง ทำไปเรื่อยๆ จนกระทั่งถึงระดับที่ 2 ซึ่งจะเห็นว่าเป็นแบบจำลองที่ดีตัวหนึ่ง แต่ข้อเสียคือว่าใช้ทรัพยากรจำนวนมาก เราอาจจะต้องใช้ทรัพยากร 5-20% หมดไปกับการปรับปรุงกระบวนการ และอาจจะใช้เวลามากถึง 25 เดือนเพื่อให้ไปถึง CMM ระดับที่ 2 และยังต้องจัดตั้งกลุ่มใหม่ๆ เช่น SQA หรือ SEPG นอกจากนี้เราอาจจะไม่มีเงิน เวลา หรือข้อมูลมากพอที่จะนำเครื่องมือมาช่วยได้ แนวทางในการจัดการทรัพยากรในองค์กรขนาดเล็กอาจจะมีแนวทางดังนี้คือ เรียนรู้จากความสำเร็จหรือความล้มเหลวของคนอื่นก่อน แล้วค่อยกำหนดกระบวนการของตัวเอง จัดตารางเวลาระหว่างโครงการเพื่อทำงาน และไม่ต้องใช้คนมากแต่ใช้คนที่ดีที่สุด กำหนดแนวทางและข้อจำกัด และลองใช้เครื่องมือง่ายๆ ที่เรามีอยู่แทนการใช้เครื่องมือแพงๆ ที่ต้องซื้อมา ในแต่ละโครงการควรจะมีการแบ่งปันเครื่องมือ อุปกรณ์และคน รวมทั้งเทคนิควิธีการต่างๆ และควรใช้คนให้ได้หลากหลายหน้าที คนหนึ่งคนอาจจะทำได้หลายบทบาท หลายกิจกรรม

4) นำ peer review เข้ามาใช้

การจัดทำ peer review เป็นสิ่งที่มีประโยชน์มาก เพราะมันสามารถช่วยในการปรับปรุงคุณภาพให้ดียิ่งขึ้น และสามารถลดการทำงานซ้ำซ้อนได้ แต่องค์กรส่วนใหญ่ไม่ค่อยเห็นความสำคัญ อาจจะเป็นเพราะไม่มีเวลา ไม่มีทุนในการจัดอบรม หรือไม่มีคนเพียงพอที่จะมาทำหน้าที่นี้ บางองค์กรอาจจะเคยทำแต่ไม่สำเร็จก็เป็นเหตุผลให้ไม่ยอมทำอีก หรืออาจจะมีทัศนคติที่ไม่ดีเนื่องจากเคยได้รับประสบการณ์ที่ไม่ดีจากการทำ peer review สำหรับองค์กรขนาดเล็ก เราสามารถนำ peer review มาประยุกต์ใช้ได้เช่น ใช้การทำแบบ peer deskcheck หรือในกรณีมีคนน้อยก็อาจจะใช้ทีมงานแค่ 2 คน ทำในบทบาทสำคัญคือ author และ reviewer การดึงคนจากโครงการอื่นมาทำก็เป็นแนวทางหนึ่ง และในเรื่องของเวลาอาจจะใช้เครื่องมือเข้ามาช่วยเพื่อให้การตรวจสอบเป็นไปได้ง่ายขึ้น ใช้เวลาน้อยลง ในกรณีทีมพัฒนามีประสบการณ์ที่ไม่ดีกับการทำ peer review มาก่อนก็อาจจะใช้การอบรมหรือการสอน มาช่วยให้มีทัศนคติที่ดีขึ้น

5) การจัดอบรม

ในการปรับปรุงกระบวนการซอฟต์แวร์โดยใช้ CMM นั้น ทุกคนในองค์กรจะต้องมีความรู้และเข้าใจในเรื่องของ CMM ร่วมกัน และเข้าใจในแนวทางเดียวกัน การจัดการอบรมเป็นวิธีที่จะช่วยให้เราบรรลุถึงวัตถุประสงค์นั้น แต่ในองค์ขนาดเล็กอาจจะมีปัญหาที่เกี่ยวข้องกับการฝึกอบรมได้มากมาย เช่นการขาดเงินทุน ขาดเวลา การไล่ตามเทคโนโลยี หรืออาจจะมีมุมมองที่คิดว่าการอบรมจะต้องทำอย่างเป็นทางการอยู่ในห้องอบรม การมีความคิดว่าการอบรมไม่ให้ประโยชน์อะไรก็เป็นอุปสรรคอย่างหนึ่ง นอกจากนี้อาจจะเป็นเพราะกลัวว่าเมื่อพนักงานมีความรู้ความเข้าใจแล้วอาจจะลาออกไป หรือปัญหาง่ายๆ ที่อาจเกิดขึ้นได้คือองค์กรไม่มีเครื่องมือในการจัดฝึกอบรม แนวทางแก้ไขในเรื่องนี้ทำได้หลายทางเช่น การเน้นให้ 'เรียนรู้' มากกว่าจะทำการอบรม หรือใช้การจัด

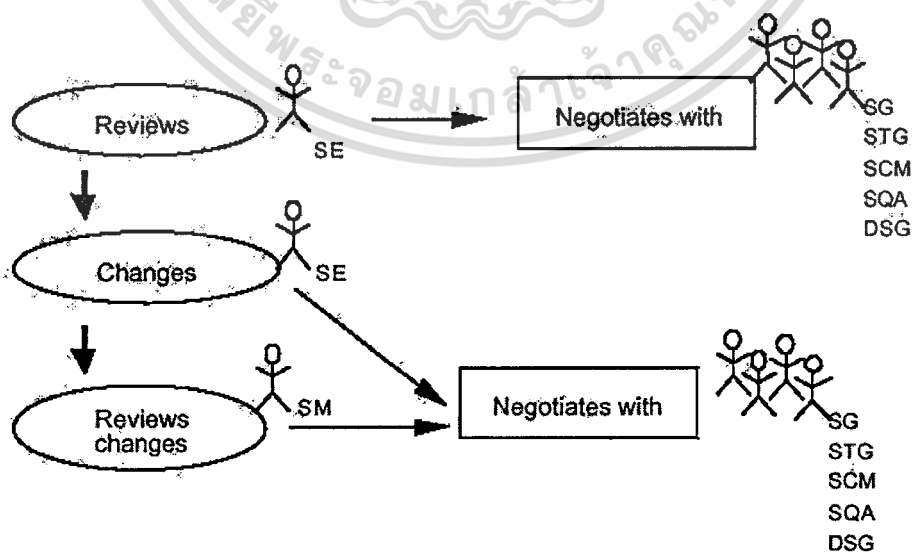
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำ workshop แทนการทำสัมมนา การเรียนรู้ในงบประมาณที่จำกัดอาจจะทำได้โดยใช้วิธี on-the-job training การศึกษาด้วยตนเอง การศึกษาจากองค์กรใหญ่ๆ หรือการเข้าร่วมการสัมมนาต่างๆ ที่มีการจัด และถ้ามีการอบรม เข้าจะต้องตระหนักว่าเราจะอบรมไปเพื่ออะไร ต้องมีการวางแผนการอบรมที่จำเป็น พิจารณาทางเลือกต่างๆ และอบรมตามความเชี่ยวชาญของแต่ละคน เพื่อให้การอบรมมีประสิทธิภาพและไม่สิ้นเปลืองงบประมาณเกินไป

2.3.5 การประยุกต์ใช้งานตามกลุ่มกระบวนการหลักของ CMM ระดับที่ 2

จากระดับการเจริญเติบโตทั้ง 5 ระดับของ CMM คือระดับเริ่มต้น ระดับทำซ้ำได้ ระดับที่มีกรรมวิธีชัดเจน ระดับจัดการ และระดับเหมาะที่สุด นั้น ทุกระดับต่างก็มีความสำคัญ แต่ CMM ระดับที่สองถือว่าเป็นระดับที่ค่อนข้างสำคัญมาก เพราะเป็นระดับเริ่มต้น ซึ่งโดยทั่วไปอาจจะใช้เวลาถึง 2 ปี ในการปรับเปลี่ยนจากองค์กรที่สับสนวุ่นวาย มาเป็นองค์กรที่มีการบริหารจัดการโครงการที่สามารถควบคุมความก้าวหน้าของโครงการและสามารถนำเทคนิควิธีการและประสบการณ์จากโครงการที่ทำเสร็จแล้วมาประยุกต์ใช้กับโครงการใหม่ได้ เพราะ CMM ระดับที่สองจะมีการวัดผลที่มีประสิทธิภาพ ทำให้สามารถประเมินต้นทุนและเวลาที่จะใช้ในการทำโครงการในอนาคตได้ ใน CMM ระดับที่สองนี้ จะเน้นในเรื่องของการบริหารจัดการโครงการซอฟต์แวร์เป็นหลัก การบริหารจัดการที่ไม่มีคุณภาพอาจนำไปสู่ปัญหาได้มากมาย การที่จะผ่าน CMM ระดับนี้ไปในขั้นที่สูงกว่าจะต้องมีการบริหารจัดการตามกลุ่มกระบวนการหลัก 6 ตัวดังนี้คือ

1) RM: การจัดการความต้องการ (Requirement Management)



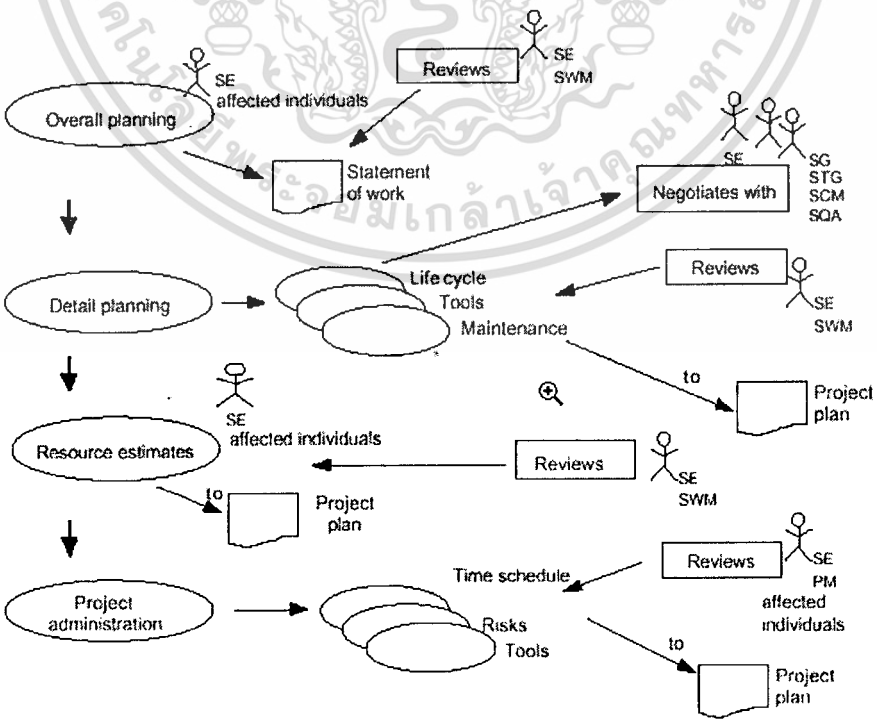
รูปที่ 2.6 แสดงกิจกรรมของ RM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องมีการจัดทำเอกสารเกี่ยวกับข้อตกลงและข้อกำหนดความต้องการต่างๆ ของผู้ใช้ ในทุกๆ เรื่องที่จะทำให้มีผลต่อโครงการ โดยเอกสารเหล่านี้จะเป็นสิ่งสำคัญในการแยกแยะและตัดสินใจในเรื่องความขัดแย้งต่างๆ เมื่อโครงการได้ดำเนินการต่อไป การติดต่อประสานงานกับผู้ใช้ทุกๆ เรื่องถือว่าเป็นเรื่องสำคัญและต้องจัดทำเป็นเอกสารถึงแม้เอกสารที่จัดทำจะเป็นเพียงแค่เอกสารง่ายๆ แผ่นเดียวก็ตาม

กลุ่มที่จะต้องเข้ามาเกี่ยวข้องในเรื่องการจัดการความต้องการได้แก่ ผู้จัดการโครงการ (PM) กุล่มควบคุมคุณภาพ (SQA) และผู้พัฒนา (SE) โดย PM จะต้องวิเคราะห์ความต้องการด้านฮาร์ดแวร์และซอฟต์แวร์ รวมถึงองค์ประกอบต่างๆ ที่จะต้องใช้ในระบบ ต้องควบคุมเอกสารความต้องการและควบคุมการเปลี่ยนแปลงของความต้องการต่างๆ SQA จะเข้ามาทบทวนกิจกรรมการจัดการความต้องการเป็นระยะ เพื่อให้แน่ใจว่ามีการดำเนินการในเรื่องความต้องการของผู้ใช้ และเมื่อมีการเปลี่ยนแปลง จะต้องมีการแก้ไขในแผนหรือกระบวนการอื่นๆ หรือไม่ ส่วน SE จะดูแลในเรื่องความต้องการที่ขาดหายไป หรือความต้องการที่ไม่ได้เขียนไว้ในเอกสาร และควบคุมไม่ให้มีการขัดแย้งกันในเรื่องของความต้องการ รวมทั้งประสานงานกับกลุ่มที่ได้รับผลกระทบจากการเปลี่ยนแปลงความต้องการนั้นด้วย

2) SPP : การวางแผนกระบวนการซอฟต์แวร์ (Software Project Planning)



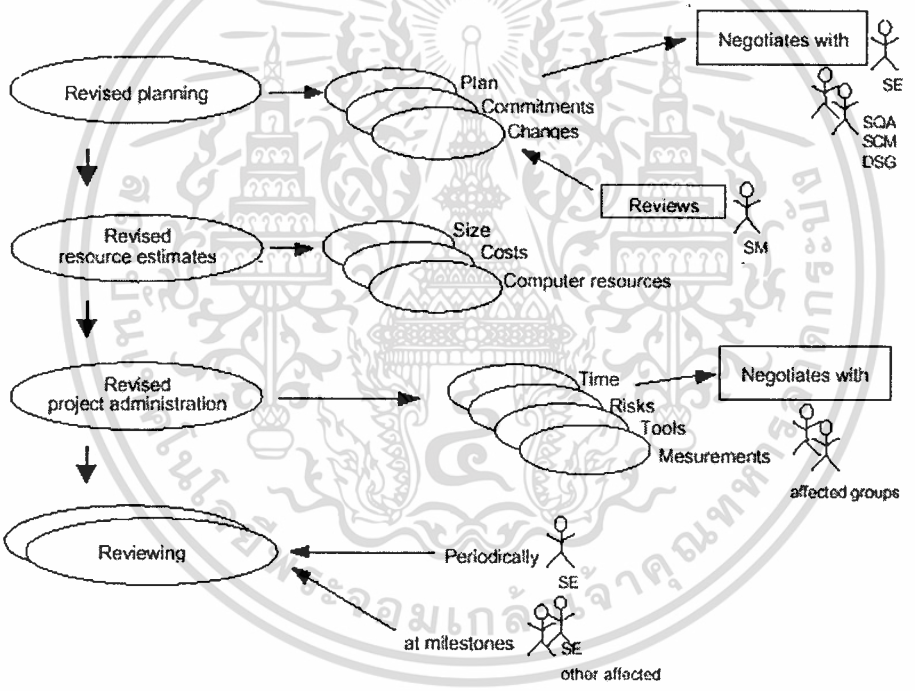
รูปที่ 2.7 แสดงกิจกรรมของ SPP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวางแผนเป็นสิ่งที่จำเป็นในทุกกระบวนการซอฟต์แวร์ แต่แผนนั้นจะต้องอยู่ในขอบเขตที่ได้พิจารณาไว้แล้ว

ผู้ที่ต้องรับผิดชอบการวางแผนกระบวนการซอฟต์แวร์คือผู้จัดการโครงการ (PM) ซึ่งทำหน้าที่ประสานงานในการวางแผน ดำเนินการวางแผนกิจกรรม ผลิตภัณฑ์ และกำหนดคนลงในแต่ละงาน นอกจากนี้ ยังต้องมี SQA ทำการทบทวนกิจกรรมการวางแผนโครงการในเรื่องเกี่ยวกับกิจกรรมในการประเมินและวางแผน พันธกิจ มาตรฐาน และเนื้อหาของแผนงาน และยังต้องมีการจัดทำแผนการควบคุมคุณภาพและแผนการจัดการองค์ประกอบซอฟต์แวร์ควบคู่ไปด้วย

3) SPTO : การติดตามดูแลโครงการ (Software Project Tracking and Oversight)



รูปที่ 2.8 แสดงกิจกรรมของ SPTO

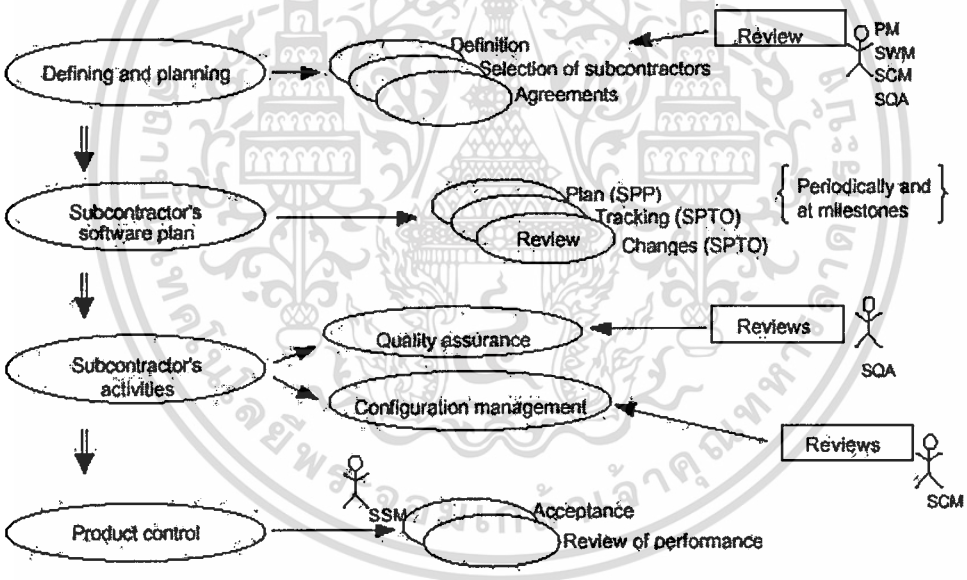
การเปรียบเทียบสิ่งที่เราได้ทำไปกับแผนที่ได้วางไว้เป็นสิ่งสำคัญ การใช้ gantt chart เทียบกับผลงานที่ได้เป็นสิ่งที่นิยมใช้ในการติดตามงาน การติดตามงานเป็นระยะทีละเล็กละน้อย ประมาณ 2-3 ครั้งต่อสัปดาห์ และ 2-3 งานต่อสัปดาห์ เป็นสิ่งที่มีประโยชน์ งานที่ได้รับมอบหมายควรจำแนกเป็นสองประเภทคือเสร็จกับไม่เสร็จ การใช้เปอร์เซนต์ไม่ค่อยจะได้ผลนัก และในการติดตามงานต้องคำนึงถึงสิ่งที่จะเกิดขึ้นถ้างานนั้นทำไม่สำเร็จด้วย นอกจากนี้การบริหารจัดการที่ดีควรมีการวัดผลที่ดีด้วย ข้อมูลที่จะใช้ในการวัดผลต้องเป็นข้อมูลที่มีประโยชน์ องค์กรขนาดเล็กไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควรใช้เวลาในการเก็บรวบรวมข้อมูลมากเกินไป มิฉะนั้นแทนที่จะช่วยในการทำงานกลับจะกลายเป็น overhead ได้

ผู้จัดการโครงการ (PM) มีหน้าที่รับผิดชอบในกลุ่มกระบวนการนี้ โดยจะดูแลในเรื่องของผลิตภัณฑ์และผลงานที่พัฒนาหรือการบริการต่างๆ แรงงานและต้นทุนที่ใช้ในกิจกรรม ตารางเวลาและงบประมาณ นอกจากนี้ยังต้องมี SQA ที่จะต้องดำเนินการทบทวนในเรื่องของพันธกิจแผนงานโครงการ เนื้อหาของแผนงานโครงการกิจกรรมที่เกี่ยวข้องกับต้นทุน เวลา ความเสี่ยง ข้อจำกัดด้านเทคนิค ฟังก์ชันการทำงานและประสิทธิภาพของผลงาน และในเรื่องของตารางเวลาทำงานก็จะต้องมีการติดตามและแก้ไขในเรื่องความสมบูรณ์ของกิจกรรม ไมล์สโตน และพันธกิจอื่นๆ เปรียบเทียบกับแผนงานโครงการ ส่วนเรื่องที่มีความเสี่ยงมากๆ จะให้ SM เป็นคนตัดสินใจ

4) SSM: การจัดการผู้รับงานรายย่อย (Software Subcontract Management)



รูปที่ 2.9 แสดงกิจกรรมของ SSM

องค์กรขนาดเล็กมากมาก (XXS) หรือองค์กรขนาดเล็กมาก (XS) อาจไม่จำเป็นต้องใช้กลุ่มกระบวนการหลักนี้ แต่องค์กรขนาดเล็ก (X) บางครั้งอาจจะต้องมีการส่งงานให้กับผู้รับงานรายย่อย

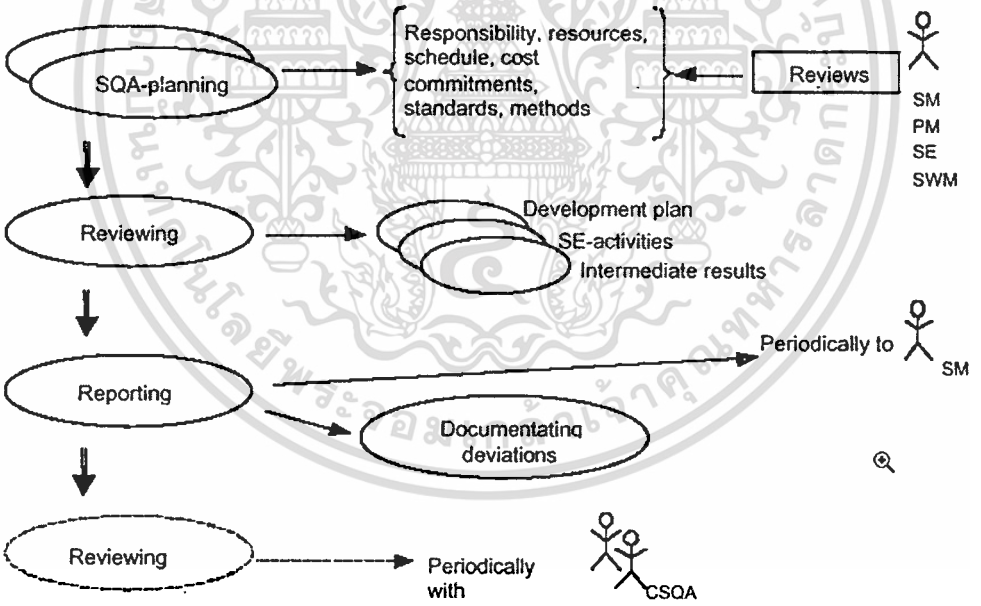
การดำเนินการกับผู้รับงานรายย่อยนั้น จะต้องมีกลุ่ม SQA เพื่อพิจารณาทบทวนในการเลือกผู้รับงานรายย่อย การทำสัญญากับผู้รับงานรายย่อย การประสานงานในเรื่องการจัดการองค์ประกอบของผู้รับงานหลักและผู้รับงานรายย่อย รวมทั้งแผนการทบทวนของผู้รับงานรายย่อยและ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบการยอมรับของซอฟต์แวร์ของผู้รับงานรายย่อย นอกจากนี้จะต้องมีการพิจารณาและติดต่อกับผู้รับงานรายย่อยเป็นระยะในเรื่องงความต้องการและความจ าเป็นของลูกค้า การควบคุมกิจกรรมทางด้านเทคนิค การตรวจสอบการด าเนินการของผู้รับงานรายย่อยให้เป็นไปตามข้อก าหนดความต้องการ และการตรวจสอบว่าได้ท าตามสัญญาครบทุกเงื่อนไข เมื่อใดก็ตามที่มีการเปลี่ยนแปลงเงื่อนไขใดๆ กับผู้รับงานรายย่อย จะต้องแจ้งให้ฝ่ายที่เกี่ยวข้องรับทราบด้วย

5) SQA : การประกันคุณภาพซอฟต์แวร์ (Software Quality Assurance)

องค์กรขนาดเล็กไม่จำเป็นต้องมีกลุ่ม SQA แยกอิสระ แต่การตรวจสอบว่าความต้องการผู้ใช้ และมาตรฐานหรือกระบวนการต่างๆ เป็นที่น่าพอใจก็ยังเป็นสิ่งสำคัญ ดังนั้นสำหรับองค์กรขนาดเล็กเราสามารถรวมหน้าที่ของการประกันคุณภาพซอฟต์แวร์ไว้ในกระบวนการได้เลย อาจจะใช้เป็น checklist หรือเครื่องมืออื่นๆ ถึงแม้ว่าจะมีกลุ่มของ SQA อยู่แล้ว การทำอย่างนี้ก็ยังมีประโยชน์



รูปที่ 2.10 แสดงกิจกรรมของ SQA

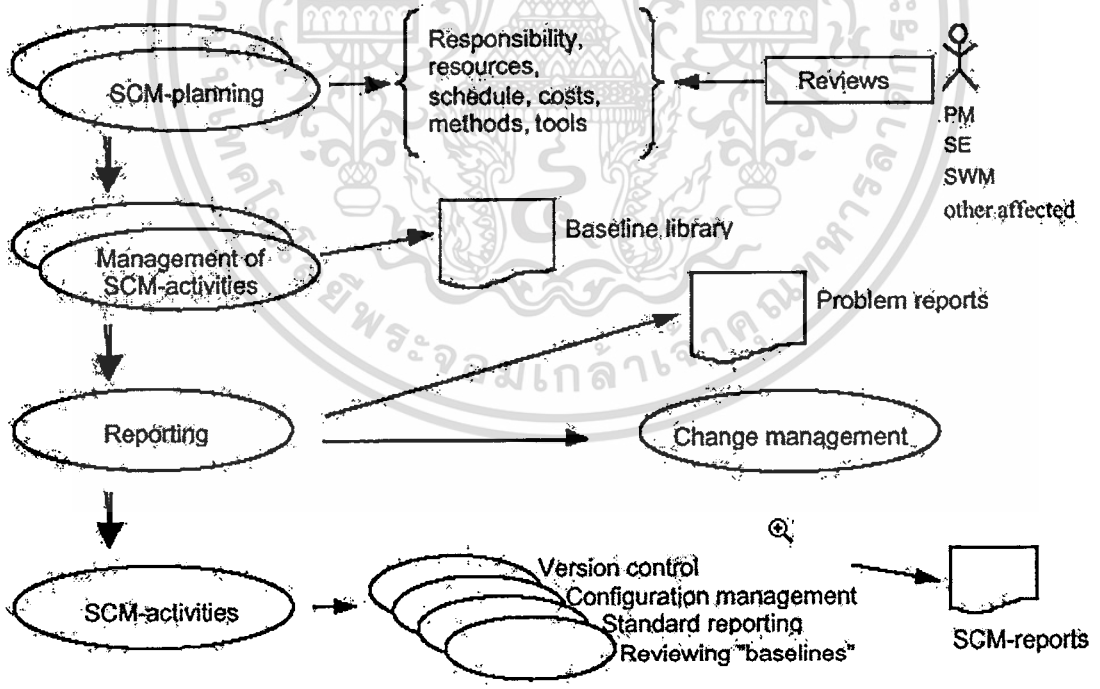
กลุ่ม SQA ต้องประสานงานกับ PM ตั้งแต่การวางแผน การกำหนดมาตรฐาน และกระบวนการต่างๆ ให้เป็นไปตามนโยบายของการประกันคุณภาพ และเพื่อวางแผนการทบทวนคุณภาพตลอดโครงการ นอกจากนี้จะต้องรับผิดชอบให้มีการกำหนดและจัดทำการวัดผลการด าเนินงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนการประกันคุณภาพจะต้องถูกจัดเตรียมตามกระบวนการที่ได้จัดทำเป็นเอกสารไว้แล้วและต้องทำคู่ขนานไปกับการวางแผนกระบวนการซอฟต์แวร์ กลุ่ม SQA ต้องมีส่วนร่วมในการทำวัตถุประสงค์โครงการ การทบทวนและการควบคุมที่เกี่ยวข้องกับแผนตามนโยบายองค์กร แผนตามมาตรฐานและข้อกำหนดความต้องการ มาตรฐานที่พอเพียงในโครงการ และควบคุมให้แผนงานมีวัตถุประสงค์ที่ชัดเจน นอกจากนี้ยังต้องทบทวนกิจกรรมให้เป็นไปตามแผนงานและมาตรฐานกระบวนการต่างๆ สิ่งที่แตกต่างออกมาต้องถูกกำหนด ถูกจัดทำเป็นเอกสารและถูกติดตามอย่างใกล้ชิด มีการตรวจสอบความถูกต้องและอะไรที่เพิ่มเติมขึ้นมาจากเดิมจะต้องถูกจัดทำเป็นเอกสารและเอกสารเหล่านั้นจะต้องมีการควบคุมอย่างดี

6) SCM: การจัดการองค์ประกอบซอฟต์แวร์ (Software Configuration Management)

องค์กรขนาดเล็กไม่จำเป็นต้องมีกลุ่มของ SCM หรือกลุ่มของ SCCB ก็ได้ แต่ถึงอย่างไรงานทางด้านการจัดการองค์ประกอบซอฟต์แวร์และการควบคุมการเปลี่ยนแปลงต่างๆ ก็ยังเป็นเรื่องที่จำเป็นต้องมี เราต้องทำการกำหนด baseline และควบคุมการเปลี่ยนแปลง baseline อยู่เสมอ



รูปที่ 2.11 แสดงกิจกรรมของ SCM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในองค์กรขนาดที่มีพนักงานน้อยกว่า 16 คน กลุ่มของ SCCB และ SCM อาจจะแฝงตัวอยู่ในกระบวนการ โดย SCCB จะมีหน้าครอบคลุมในเรื่องของการกำหนดสิทธิ์การสร้าง baseline และกำหนด configuration item การพิจารณาและอนุญาตให้มีการแก้ไข baseline หรืออนุญาตให้สร้างผลผลิตจาก baseline library ส่วนกลุ่ม SCM มีหน้าที่ในการสร้างและจัดการ baseline library การพัฒนา ดูแลรักษา และแจกจ่ายแผน SCM มาตรฐานและกระบวนการต่างๆ การกำหนดงานที่จะอยู่ภายใต้การควบคุมองค์ประกอบ การจัดการการเข้าถึง library และ update ข้อมูลใน library การสร้างผลิตภัณฑ์จาก baseline library การบันทึกกิจกรรมของ SCM และจัดทำเอกสารแจกจ่ายให้กลุ่มที่เกี่ยวข้อง นอกจากนี้ควรมีการจัดอบรมกิจกรรม SCM ให้กับผู้เกี่ยวข้อง เช่น กลุ่มของ SE SQA และ DSG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบเอกสารมาตรฐานตามแนวทางของ CMM ระดับที่ 2

การพัฒนาซอฟต์แวร์โดยใช้แนวทางของ CMM จะต้องข้องเกี่ยวกับเอกสารจำนวนมาก ทุกๆ อย่างที่เกิดขึ้นในโครงการจะต้องทำเป็นเอกสารเพื่อสามารถใช้อ้างอิงได้ แต่ในโครงการขนาดเล็กนั้น เป็นเรื่องยุ่งยากและแทบจะเป็นไปไม่ได้ที่จะจัดการกับเอกสารมากมายขนาดนั้น เราต้องจำกัดจำนวนเอกสารเฉพาะเอกสารที่จำเป็น กิจกรรมหลายอย่างที่จำเป็นจะต้องถูกจัดทำเป็นเอกสารอาจจะรวมไว้ในเอกสารเดียวกัน ซึ่งจะช่วยลดจำนวนเอกสารและเวลาในการจัดการกับเอกสารนั้นได้

เอกสารใน CMM แบ่งได้เป็น 3 ประเภทหลักๆ คือ เอกสารเกี่ยวกับนโยบาย เอกสารเกี่ยวกับกระบวนการทำงาน และเอกสารที่เป็นผลผลิตจากกระบวนการพัฒนาซอฟต์แวร์ (software work products) เอกสารที่เป็นนโยบายและกระบวนการทำงานนั้น เป็นเอกสารที่เขียนขึ้นครั้งเดียวและสามารถใช้ได้ทั้งองค์กร ส่วนเอกสารที่เป็นผลผลิตจากกระบวนการพัฒนาซอฟต์แวร์นั้น จะเกิดขึ้นตลอดวัฏจักรของกระบวนการพัฒนาซอฟต์แวร์

ตารางที่ 2 แสดงเอกสารมาตรฐานทั้งหมดตามแนวทางของ CMM ระดับที่ 2 ที่ควรจะมีในโครงการขนาดเล็ก ช่องซ้ายมือเป็นชื่อเอกสาร และช่องขวามือเป็นการอ้างอิงในแต่ละกลุ่มกระบวนการหลัก (KPA) โดยแบ่งย่อยถึงพันธกิจแต่ละตัวด้วย นอกจากนี้ โครงงานฉบับนี้จะเน้นการศึกษาในเรื่องโครงการขนาดเล็กมากๆ (XSS) ดังนั้นจะตัด KPA ในเรื่องการจัดการผู้รับงานรายย่อย (Software Subcontract Management) ออกไป เพื่อลดจำนวนเอกสาร

ตารางที่ 2 แสดงเอกสารมาตรฐานตามแนวทางของ CMM ระดับที่ 2 สำหรับโครงการขนาดเล็ก

ชื่อเอกสาร	KPA
เอกสารเกี่ยวกับนโยบาย	
1. เอกสารนโยบายเกี่ยวกับ RM, SPP, SPTO, SQA และ SCM	RM/C.1, SPP/C.2, SPTO/C.2, SSM/C.1, SQA/C.1, SCM/C.1
เอกสารเกี่ยวกับกระบวนการทำงาน	
2. กระบวนการในการบริหารจัดการความต้องการของผู้ใช้	RM/AB.2, RM/AC.1,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อเอกสาร	KPA
	RM/AC.3
3. กระบวนการในการวางแผนการพัฒนาซอฟต์แวร์	SPP/AC.4, SPP/AC.6, SPP/AC.9, SPP/AC.10, SPP/AC.11, SPP/AC.12
4. กระบวนการในการติดตามดูแลโครงการ	SPTO/AC.2, SPTO/AC.3, SPTO/AC.13
5. กระบวนการในเรื่องการประกันคุณภาพซอฟต์แวร์	SQA/AC.1, SQA/AC.7
6. กระบวนการในการจัดการองค์ประกอบซอฟต์แวร์	SCM/AC.1, SCM/AC.5, SCM/AC.6, SCM/AC.7, SCM/AC.8, SCM/AC.10
เอกสารที่เป็นผลผลิตจากกระบวนการซอฟต์แวร์	
7. เอกสารข้อกำหนดความต้องการของระบบ	RM/AB.1, RM/AB.2
8. เอกสารแผนงาน • แผนการพัฒนาซอฟต์แวร์และแผนการติดตามงาน • แผนการประกันคุณภาพซอฟต์แวร์ • แผนการจัดการองค์ประกอบซอฟต์แวร์	 AC.7, SPTO/AC.8 SQA/AC.1 SCM/AC.2
9. เอกสารการขอเปลี่ยนแปลง • เอกสารการเปลี่ยนแปลงความต้องการ • เอกสารการขอเปลี่ยนแปลง baseline • เอกสารการขอเปลี่ยนแปลงซอฟต์แวร์	 RM/AC.3 SCM/AC.3, SCM/AC.4 SCM/AC.3, SCM/AC.4
10. เอกสารการวัดผล (รายงานต่างๆ)	RM/V.1, SPP/V.1, SPTO/V.1, SQA/V.1, SCM/V.1

โดยเอกสารแต่ละตัวมีรายละเอียดดังนี้คือ

3.1 เอกสารนโยบายเกี่ยวกับ RM, SPP, SPTO, SQA และ SCM

เอกสารเกี่ยวกับนโยบาย เป็นเอกสารที่จัดทำขึ้นเพื่อเป็นแนวทางหลักในการบริหารจัดการโครงการซอฟต์แวร์ โดยจัดทำเพียงครั้งเดียว และสามารถนำไปใช้ได้ในทุกโครงการ และองค์กรจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องมีนโยบายตามกลุ่มกระบวนการหลัก ทั้ง 6 ตัว ในองค์กรขนาดเล็กนั้น เราจะจัดทำเอกสารนโยบายรวมอยู่ในเอกสารฉบับเดียวกันเพื่อลดจำนวนเอกสาร โดยนโยบายจะเป็นเอกสารที่ต้องจัดเก็บไว้เป็น CI ตัวหนึ่งในโครงการ ในกรณีมีการเปลี่ยนแปลงนโยบายจะสามารถนำไปอ้างอิงได้ เอกสารนโยบายจะนำไปอ้างอิงถึง KPA ได้ดังนี้คือ RM/C.1, SPP/C.2, SPTO/C.2, SSM/C.1, SQA/C.1 และ SCM/C.1

Policy_v1.0

วันที่ 26 มกราคม พ.ศ. 2546

นโยบายในการบริหารจัดการโครงการซอฟต์แวร์

บริษัท ABC จำกัด

<u>วัตถุประสงค์</u>

เอกสารนโยบายฉบับนี้ จัดทำขึ้นเพื่อให้พนักงานทุกคนของบริษัทฯ มีแนวทางหลักในการบริหารจัดการโครงการซอฟต์แวร์ โดยพนักงานจะต้องดำเนินการพัฒนาซอฟต์แวร์ตามนโยบายที่กำหนดไว้อย่างเคร่งครัด เพื่อให้ได้ผลิตภัณฑ์ที่ตรงตามความต้องการของผู้ใช้ ส่งถึงลูกค้าได้ทันเวลา และอยู่ในงบประมาณที่กำหนดไว้

<u>นโยบายด้านการจัดการความต้องการของผู้ใช้</u>

1. ความต้องการของผู้ใช้จะต้องจัดทำเป็นเอกสาร
2. ความต้องการของผู้ใช้จะต้องได้รับการทบทวนตรวจทานโดยผู้จัดการโครงการและบุคคลที่เกี่ยวข้อง
3. เมื่อมีการเปลี่ยนแปลงความต้องการของผู้ใช้ จะต้องมีการปรับแก้แผนงานโครงการ รวมถึงผลกระทบที่อาจจะเกิดขึ้นกับกิจกรรมอื่นๆ

<u>นโยบายด้านการวางแผนโครงการ</u>

1. ต้องใช้ข้อมูลความต้องการของผู้ใช้และข้อมูลการประมาณการมาเป็นพื้นฐานในการวางแผนโครงการ
2. ข้อตกลงต่างๆ จะต้องได้รับความเห็นชอบจากผู้จัดการโครงการ ผู้อำนวยการ และหัวหน้ากลุ่มอื่นๆ ที่เกี่ยวข้อง
3. การทำงานที่ขึ้นกับหน่วยงานอื่นๆ จะต้องมีการหารือและจัดทำเป็นเอกสาร
4. กลุ่มบุคคลที่เกี่ยวข้องจะต้องตรวจทานแผนงานโครงการและการประมาณการต่างๆ
5. ผู้อำนวยการต้องทำการตรวจทานข้อตกลงทุกข้อที่ทำร่วมกับหน่วยงานภายนอก
6. มีการจัดทำเอกสารแผนงานโครงการ และเอกสารนี้ต้องได้รับการควบคุมและจัดการเป็นอย่างดี (แผนงานโครงการเป็น CI ตัวหนึ่ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นโยบายด้านการติดตามดูแลโครงการ

1. ต้องใช้แผนงานโครงการเป็นพื้นฐานในการติดตามงาน และต้องปรับแก้ให้แผนงานนั้นทันสมัยอยู่เสมอ
2. ผู้จัดการโครงการจะต้องได้รับรายงานสถานะ และปัญหาที่เกิดขึ้นกับโครงการอย่างสม่ำเสมอ
3. ต้องมีการจัดการอย่างหนึ่งอย่างใดเมื่อโครงการมีแนวโน้มว่าจะไม่สำเร็จตามเป้าหมาย ไม่ว่าจะเป็นการปรับประสิทธิภาพของระบบหรือการปรับแผนงานโครงการ
4. เมื่อข้อตกลงใดๆ มีการเปลี่ยนแปลงต้องมีการหารือร่วมกันระหว่างกลุ่มบุคคลที่เกี่ยวข้อง

นโยบายด้านการประกันคุณภาพ

1. ต้องมีบทบาทของ SQA เข้าไปเกี่ยวข้องในทุกโครงการ
2. กิจกรรมต่างๆ ของ SQA จะต้องแยกออกจากการบริหารจัดการโครงการ
3. ผู้อำนวยการจะต้องตรวจทานกิจกรรมและผลงานของSQA อย่างสม่ำเสมอ

นโยบายด้านการจัดการองค์ประกอบซอฟต์แวร์

1. ต้องมีการมอบหมายหน้าที่ของ SCM ให้เห็นอย่างเด่นชัด
2. กิจกรรม SCM จะต้องทำตลอดวงจรชีวิตของการพัฒนาซอฟต์แวร์
3. กิจกรรมของ SCM จะต้องดำเนินการกับผลงานทุกอย่างที่เป็นผลิตผลของการพัฒนาซอฟต์แวร์ ทั้งที่จะส่งออกภายนอกหรือส่งภายในหน่วยงาน
4. โครงการจะต้องมีพื้นที่จัดเก็บองค์ประกอบต่างๆ ของ SCM
5. มีการทบทวนตรวจทาน baseline และกิจกรรมต่างๆ ของ SCM อย่างสม่ำเสมอ

ลงชื่อ .....

(.....)

ประธานกรรมการ

รูปที่ 3.1 ตัวอย่างนโยบายใน CMM ระดับที่ 2

3.2 เอกสารกระบวนการในการบริหารจัดการความต้องการของผู้ใช้

เอกสารกระบวนการในการบริหารจัดการความต้องการของผู้ใช้ แสดงถึงขั้นตอนในการจัดการกับความต้องการของผู้ใช้ที่อาจจะเกิดขึ้นตลอดช่วงเวลาที่พัฒนาซอฟต์แวร์ โดยจะจัดทำขึ้นเพียงครั้งเดียวภายในองค์กรและสามารถนำไปใช้ในทุกโครงการ และเอกสารนี้จะเป็น CI ตัวหนึ่งที่ต้องจัดเก็บใน library ของโครงการ เอกสารนี้จะนำไปอ้างอิงถึง KPA ที่ RM/AB.2, RM/AC.1, RM/AC.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RM_Process_v1.0

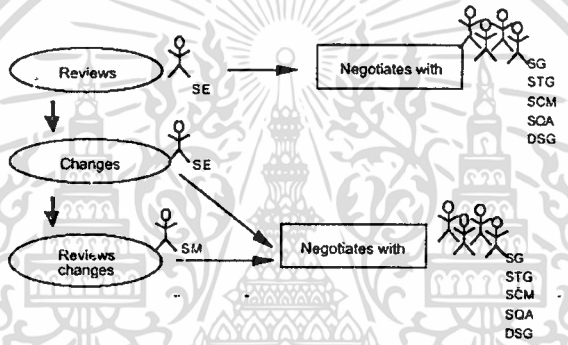
วันที่ 26 มกราคม 2546

กระบวนการในการบริหารจัดการความต้องการของผู้ใช้

บริษัท ABC จำกัด

<u>วัตถุประสงค์</u>

เอกสารกระบวนการฉบับนี้จัดทำขึ้นเพื่อเป็นแนวทางในการบริหารจัดการความต้องการของผู้ใช้ เพื่อให้ได้ผลผลิตที่ตรงตามความต้องการของผู้ใช้มากที่สุด และลดความขัดแย้งในเรื่องการสื่อสารที่อาจจะเข้าใจไม่ตรงกัน



กิจกรรมในการบริหารจัดการความต้องการของผู้ใช้

Input ความต้องการของผู้ใช้

Output เอกสารกำหนดความต้องการ

1. SE รวมรวมความต้องการและจัดทำเอกสารกำหนดความต้องการ ทั้งในด้านฮาร์ดแวร์ ซอฟต์แวร์ และองค์ประกอบอื่นๆ ที่เกี่ยวข้อง
2. กลุ่มบุคคลที่เกี่ยวข้องทบทวนตรวจทานเอกสารกำหนดความต้องการ ก่อนจะนำไปจัดเก็บไว้ใน library
3. เมื่อมีการเปลี่ยนแปลงความต้องการ ให้ SE ทบทวนการเปลี่ยนแปลงนั้นร่วมกับ SM และผู้เกี่ยวข้อง ว่ามีผลกระทบอย่างไร และมีแนวทางการดำเนินงานอย่างไร จากนั้นให้กรอกแบบฟอร์มการขอเปลี่ยนแปลงความต้องการ ก่อนที่จะมีการปรับปรุงเอกสารกำหนดความต้องการ
4. การเปลี่ยนแปลงที่มีผลกระทบกับสัญญา ต้องได้รับการหารือร่วมกันระหว่างผู้เกี่ยวข้องทุกฝ่าย
5. PM ต้องทำการประเมินผลการดำเนินงานกิจกรรม RM เป็นระยะ และจัดทำรายงานสรุป

รูปที่ 3.2 ตัวอย่างเอกสารกระบวนการในการจัดการความต้องการของผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 เอกสารกระบวนการในการวางแผนโครงการ

เอกสารกระบวนการในการวางแผนโครงการ แสดงถึงขั้นตอนในการวางแผนงานโครงการ รวมทั้งกิจกรรมที่เกี่ยวข้องเกิดขึ้นตลอดช่วงเวลาที่พัฒนาซอฟต์แวร์ โดยจะจัดทำขึ้นเพียงครั้งเดียวภายในองค์กรและสามารถนำไปใช้ในทุกโครงการ และเอกสารนี้จะเป็น CI ตัวหนึ่งที่ต้องจัดเก็บใน library ของโครงการ โดยสามารถนำไปอ้างอิงถึง KPA ที่ SPP/AC.4, SPP/AC.6, SPP/AC.9, SPP/AC.10, SPP/AC.11 และ SPP/AC.12

SPP_Process_v1 0

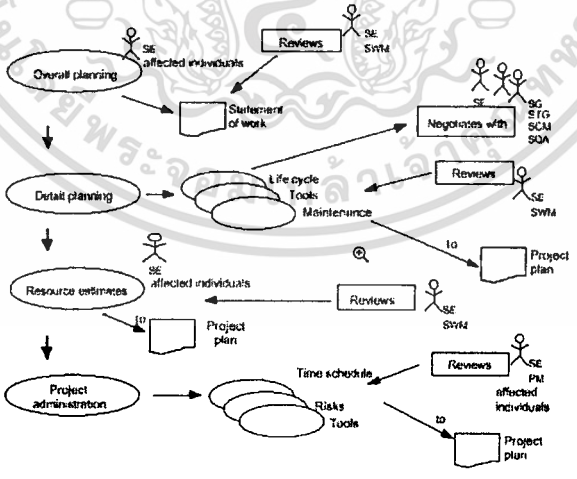
26 มกราคม 2546

กระบวนการในการวางแผนโครงการซอฟต์แวร์

บริษัท ABC จำกัด

<u>จุดประสงค์</u>

เอกสารฉบับนี้จัดทำขึ้นเพื่อแสดงขั้นตอนและกระบวนการในการวางแผนโครงการซอฟต์แวร์ รวมถึงกิจกรรมเกี่ยวกับแผนงานโครงการที่เกิดขึ้นตลอดช่วงชีวิตการพัฒนาซอฟต์แวร์ เพื่อให้การพัฒนาซอฟต์แวร์เป็นไปตามแผนการดำเนินงานที่วางไว้



กิจกรรมในกระบวนการวางแผนโครงการซอฟต์แวร์

Input เอกสารกำหนดความต้องการ

Output statement of work, แผนงานโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPP_Process_v1.0

1. PM ร่วมกับ SE วางแผนโครงการในภาพรวม และดำเนินการจัดทำ statement of work โดยใช้เอกสารกำหนดความต้องการเป็นข้อมูลหลัก และ statement of work ที่จัดทำขึ้นนี้ ต้องได้รับการทบทวนตรวจทานจากผู้เกี่ยวข้องก่อนที่จะนำไปจัดเก็บใน library
2. PM วางแผนโครงการในรายละเอียด ในเรื่องของแบบจำลองการพัฒนาซอฟต์แวร์ที่จะใช้ เครื่องมือ และการบำรุงรักษาซอฟต์แวร์ โดยหารือกับกลุ่มผู้เกี่ยวข้อง แล้วให้ SE และ SWM ตรวจทาน และจัดทำเป็นแผนงานโครงการ
3. PM จัดทำการประมาณการทรัพยากรที่ใช้ ในเรื่องของต้นทุนและแรงงาน ที่จะใช้ในการพัฒนาระบบ โดยใช้ข้อมูลจากโครงการเดิมที่คล้ายคลึงกันมาเป็นข้อมูลประกอบในการประมาณการ และต้องให้ SE และ SWM ตรวจทาน และทำการรวบรวมไว้เป็นหัวข้อหนึ่งในแผนงานโครงการ
4. PM จัดทำตารางเวลาในการทำงาน เครื่องมือที่จะใช้ช่วยในการดำเนินงาน รวมถึงประมาณการความเสี่ยงที่อาจจะเกิดขึ้น โดยให้ SE และ PM ที่เกี่ยวข้องตรวจทาน และรวมรวมไว้เป็นหัวข้อหนึ่งในแผนงานโครงการ
5. นำแผนงานโครงการที่ได้รับการทบทวนตรวจทานจากผู้เกี่ยวข้องเรียบร้อยแล้วจัดเก็บไว้ใน library
6. PM ต้องตรวจสอบผลการดำเนินกิจกรรมการวางแผนซอฟต์แวร์เป็นระยะ และจัดทำรายงานสรุป

รูปที่ 3.3 ตัวอย่างเอกสารกระบวนการในการวางแผนโครงการ

3.4 กระบวนการในเรื่องการดูแลติดตามโครงการ

เอกสารกระบวนการในการดูแลติดตามโครงการ แสดงถึงกระบวนการในการติดตามดูแลโครงการให้เป็นไปตามแผนที่วางไว้ ซึ่งจะต้องทำตลอดช่วงเวลาที่พัฒนาซอฟต์แวร์ โดยเอกสารนี้จะจัดทำขึ้นเพียงครั้งเดียวภายในองค์กรและสามารถนำไปใช้ในทุกโครงการ และเป็น CI ตัวหนึ่งที่ต้องจัดเก็บใน library ของโครงการ โดยเอกสารนี้สามารถนำไปอ้างอิงถึง KPA ที่ SPTO/AC.2, SPTO/AC.3 และ SPTO/AC.13

SPTO_Process_v1.0

26 มกราคม 2546

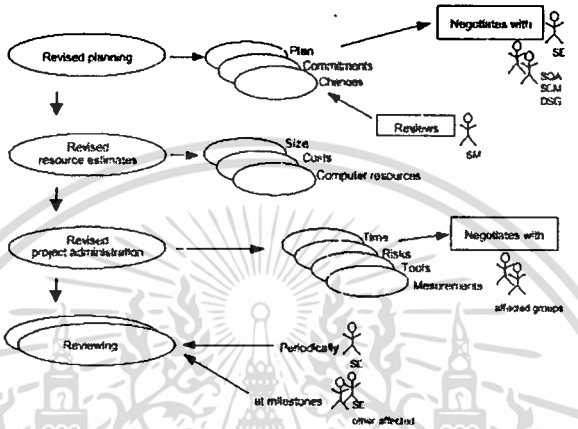
กระบวนการติดตามดูแลโครงการ

บริษัท ABC จำกัด

<u>วัตถุประสงค์</u>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นแนวทางในการติดตามดูแลโครงการซอฟต์แวร์ เพื่อให้การดำเนินการเป็นไปตามแผนงานที่กำหนดไว้ และสามารถทราบสถานะการทำงานของโครงการตลอดเวลา เมื่อมีปัญหาจะสามารถแก้ไขได้ทันเวลา



กิจกรรมในกระบวนการดูแลติดตามโครงการ

Input แผนงานโครงการ

Output แผนงานโครงการที่ได้รับการปรับปรุงแล้ว และรายงานที่เกี่ยวข้อง

1. PM ทำการติดตามการพัฒนาซอฟต์แวร์ ทั้งในเรื่องของผลผลิตที่ต้องได้รับตามช่วงเวลา ต้นทุนและแรงงานที่ใช้ไป และรายจ่ายที่ใช้ไป เปรียบเทียบกับแผนงานโครงการที่จัดทำไว้แล้ว
2. เมื่อมีการเปลี่ยนแปลงใดๆ กับแผนงานโครงการ ให้ PM หารือกับผู้เกี่ยวข้อง จากนั้นให้กรอกแบบฟอร์มการขอเปลี่ยนแปลง baseline พร้อมทั้งทำการปรับปรุงแผนงานโครงการ แล้วให้ SM ตรวจทานก่อนนำเก็บเข้า library ต่อไป
3. SE ต้องจัดทำแผนการตรวจทานแผนงานโครงการเปรียบเทียบกับผลงานที่ได้จริงเป็นระยะ หรือเมื่อถึง milestone และต้องมีการหารือร่วมกับผู้เกี่ยวข้องเมื่อผลการดำเนินงานไม่เป็นไปตามแผนงานโครงการ หรือเห็นควรที่จะปรับปรุงแผนงานโครงการ
4. มีการวัดผลการดำเนินงานเกี่ยวกับการติดตามงาน โดยสรุปเป็นรายงาน เช่นรายงานต้นทุนและทรัพยากรที่ใช้ในการติดตามงาน หรือสรุปแผนงานโครงการที่มีการเปลี่ยนแปลง (ทั้งในด้านขนาด ต้นทุน ทรัพยากรอื่นๆ และเวลาทำงาน)
5. ข้อมูลต่างๆ ที่สามารถวัดได้ จะถูกจัดเก็บและติดตาม เพื่อนำไปใช้ในโครงการปัจจุบันและเป็นแนวทางให้โครงการใหม่ที่จะเกิดขึ้นในอนาคต

รูปที่ 3.3 ตัวอย่างเอกสารกระบวนการในการดูแลติดตามโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 เอกสารกระบวนการในเรื่องการประกันคุณภาพซอฟต์แวร์

เอกสารกระบวนการในเรื่องของการประกันคุณภาพซอฟต์แวร์ แสดงถึงกระบวนการในการประกันคุณภาพซอฟต์แวร์ให้เป็นไปตามกระบวนการและขั้นตอนที่กำ าหนดไว้ ซึ่งกิจกรรมนี้ต้องทำตลอดช่วงเวลาที่พัฒนาซอฟต์แวร์ เอกสารนี้จะจัดทำขึ้นเพียงครั้งเดียวภายในองค์กรและสามารถนำไปใช้ในทุกโครงการ และเป็น CI ตัวหนึ่งที่ต้องจัดเก็บใน library ของโครงการ โดยเอกสารนี้สามารถนำไปอ้างอิงถึง KPA ที่ SQA/AC.1 และ SQA/AC.7

SQA_Process_v1.0

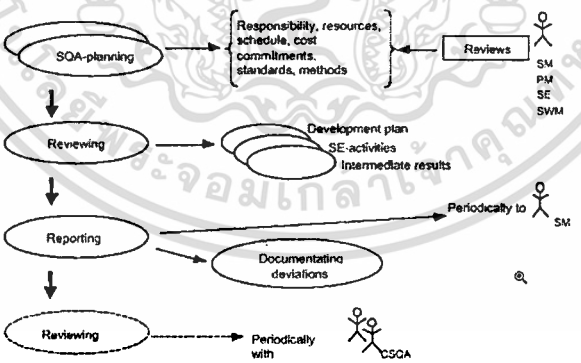
26 มกราคม 2546

กระบวนการประกันคุณภาพซอฟต์แวร์

บริษัท ABC จำกัด

<u>วัตถุประสงค์</u>

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นแนวทางในการดำเนินกิจกรรมการประกันคุณภาพซอฟต์แวร์ เพื่อรับประกันว่าการพัฒนาซอฟต์แวร์ได้ดำเนินการตามนโยบายและกระบวนการที่ได้กำหนดไว้อย่างเคร่งครัด เพื่อให้ได้ผลผลิตที่มีคุณภาพตรงตามความต้องการของผู้ใช้



กิจกรรมในเรื่องการประกันคุณภาพซอฟต์แวร์

Input เงื่อนไข น โยบาย กระบวนการและมาตรฐานต่างๆ ที่ได้กำหนดไว้

Output แผนการรับประกันคุณภาพและรายงานที่เกี่ยวข้อง

1. SQA จัดทำแผนการประกันคุณภาพในภาพรวม ในเรื่องของหน้าที่ความรับผิดชอบ ตารางเวลาทำงาน ต้นทุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SQA_Process_v1.0

มาตรฐานและวิธีการที่จะใช้ โดยจะจัดทำไปพร้อมๆ กับการจัดทำแผนงานโครงการ เพื่อให้แผนการดำเนินงานมีความสอดคล้องกัน

2. ผู้เกี่ยวข้องตรวจทานแผนการประกันคุณภาพ แล้วจัดเก็บแผนนั้นใน library
3. เมื่อมีการเปลี่ยนแปลงใดๆ ที่มีผลกระทบกับแผนการประกันคุณภาพ จะต้องหารือกับผู้เกี่ยวข้อง แล้วกรอกแบบฟอร์มการขอเปลี่ยนแปลง baseline เพื่อทำการปรับปรุงแผนฯ ให้เป็นไปตามข้อตกลงนั้น
4. SQA วางแผนการตรวจสอบให้สอดคล้องกับขั้นตอนในการพัฒนาซอฟต์แวร์ โดยเปรียบเทียบกับมาตรฐานและกระบวนการที่กำหนดไว้
5. เมื่อพบกิจกรรมที่ไม่เป็นไปตามมาตรฐานหรือกระบวนการที่กำหนดไว้ SQA จะต้องรายงานให้ SM ทราบและจัดทำเอกสารการเปลี่ยนแปลงนั้น
6. PM วัดผลการดำเนินงานกิจกรรม SQA และจัดทำเป็นรายงาน

รูปที่ 3.5 ตัวอย่างเอกสารกระบวนการในการประกันคุณภาพซอฟต์แวร์

3.6 กระบวนการในการจัดการองค์ประกอบซอฟต์แวร์

เอกสารกระบวนการในการจัดการองค์ประกอบซอฟต์แวร์จะแสดงถึงกระบวนการในการดำเนินกิจกรรมที่เกี่ยวข้องกับการจัดการองค์ประกอบซอฟต์แวร์ เช่นการจัดการ library การจัดเก็บเอกสาร ซึ่งเป็นกิจกรรมที่ต้องทำตลอดช่วงเวลาที่พัฒนาซอฟต์แวร์ โดยเอกสารนี้จะจัดทำขึ้นเพียงครั้งเดียวภายในองค์กรและสามารถนำไปใช้ในทุกโครงการ และเป็น CI ตัวหนึ่งที่ต้องจัดเก็บใน library ของโครงการ โดยเอกสารนี้สามารถอ้างอิงได้ถึง KPA ที่ SCM/AC.1, SCM/AC.5, SCM/AC.6, SCM/AC.7, SCM/AC.8 และ SCM/AC.10

SCM_Process_v1.0

26 มกราคม 2546

กระบวนการจัดการองค์ประกอบซอฟต์แวร์

บริษัท ABC จำกัด

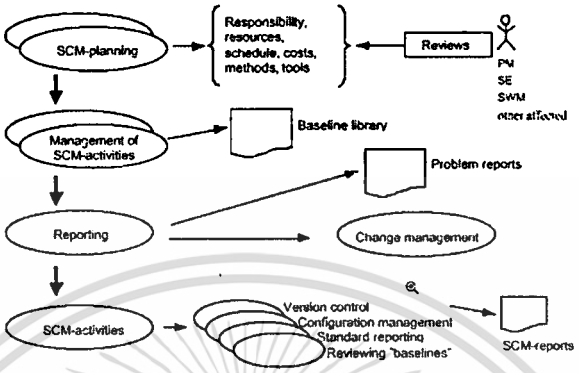
<u>วัตถุประสงค์</u>

เอกสารฉบับนี้จัดทำขึ้นเพื่อเป็นแนวทางในการจัดการองค์ประกอบซอฟต์แวร์ของโครงการ เพื่อให้การดำเนินการพัฒนาซอฟต์แวร์เป็นไปได้อย่างเรียบร้อย ไม่มีความสับสน มีการจัดเก็บเอกสารและ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCM_Process_v1.0

โปรแกรมอย่างเป็นระบบ



กิจกรรมในกระบวนการจัดการองค์ประกอบซอฟต์แวร์

Input นโยบายด้านการจัดการองค์ประกอบซอฟต์แวร์

Output ข้อกำหนดต่างๆ , รายการ CI ที่ต้องการจัดเก็บ, baseline และรายงานที่เกี่ยวข้อง

1. SCM จัดทำแผนการจัดการองค์ประกอบซอฟต์แวร์ โดยจะครอบคลุมในเรื่องหน้าที่ความรับผิดชอบ ทรัพยากรที่ต้องใช้ ตารางการทำงาน วิธีการและเครื่องมือต่างๆ ที่จะนำมาช่วยในการใช้งาน โดยแผนฯ นี้จะจัดทำควบคู่ไปกับแผนงานโครงการ เพื่อให้การดำเนินงานมีความสอดคล้องกัน
2. PM, SE , SWM และผู้ที่ได้รับผลกระทบ ตรวจทานแผนการจัดการองค์ประกอบซอฟต์แวร์ ก่อนที่จะนำแผนฯ นั้นจัดเก็บไว้ใน library
3. เมื่อมีการเปลี่ยนแปลงใดๆ เกิดขึ้นกับแผนการจัดการองค์ประกอบซอฟต์แวร์ ให้หารือร่วมกับผู้เกี่ยวข้องแล้วกรอกแบบฟอร์มการขอเปลี่ยนแปลง baseline เพื่อขออนุญาตเปลี่ยนแปลงแผนฯ นั้น
4. SCM จัดทำ baseline library ซึ่งจะประกอบด้วย CI ทุกตัวที่จะเก็บไว้ใน library นี้ รวมทั้งกำหนดกระบวนการในการเข้าถึง CI กระบวนการในการเปลี่ยนแปลง CI และการกำหนดสิทธิ์การเข้าถึง CI แต่ละตัว
5. SCM จะต้องจัดเก็บประวัติการเปลี่ยนแปลงของ CI แต่ละตัว เพื่อนำไปใช้อ้างอิงได้
6. SCM จะต้องกำหนดแนวทางในเรื่องต่อไปนี้คือ การควบคุมเวอร์ชันของ baseline การจัดการองค์ประกอบ รายงานมาตรฐาน และการตรวจทาน baseline
7. SCM มีการตรวจทาน baseline อย่างสม่ำเสมอ และจัดทำรายงานที่เกี่ยวข้อง
8. เมื่อพบปัญหาในการดำเนินกิจกรรมการจัดการองค์ประกอบซอฟต์แวร์ ให้รายงานปัญหาให้ผู้เกี่ยวข้องรับทราบ
9. PM วัดผลการดำเนินงานกิจกรรม SCM เป็นระยะ และจัดทำรายงานสรุป

รูปที่ 3.6 ตัวอย่างเอกสารกระบวนการในการจัดการองค์ประกอบซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 เอกสารแนวทางการดำเนินงานด้านการจัดการองค์ประกอบซอฟต์แวร์

เอกสารแนวทางการดำเนินงานด้านการจัดการองค์ประกอบซอฟต์แวร์นั้น เป็นกระบวนการทำงานย่อยที่แยกมาจากกระบวนการจัดการองค์ประกอบซอฟต์แวร์ โดยจะเป็นแนวทางมาตรฐานที่กำหนดขึ้นเพื่อให้การดำเนินการพัฒนาซอฟต์แวร์เป็นไปในแนวทางเดียวกัน ทุกคนรับรู้และเข้าใจร่วมกัน โดยเอกสารนี้จะจัดทำขึ้นเพียงครั้งเดียวภายในองค์กร และสามารถนำไปใช้ได้ในทุกโครงการ และเป็น CI ตัวหนึ่งที่ต้องจัดเก็บใน Library

SCM_guideline_v1.0

26 มกราคม 2546

แนวทางการดำเนินงานด้านการจัดการองค์ประกอบซอฟต์แวร์

บริษัท ABC จำกัด

<u>วัตถุประสงค์</u>

เอกสารฉบับนี้จัดทำขึ้นเพื่อใช้เป็นแนวทางในการควบคุมซอฟต์แวร์หรือผลิตภัณฑ์ รวมทั้งเอกสารและรายงานต่างๆ ที่ได้จากการดำเนินงานโครงการ เพื่อเพิ่มประสิทธิภาพของงานด้านต่างๆ เช่นการติดตามข้อมูล (Tracking) หรือผลิตภัณฑ์ได้อย่างรวดเร็วและถูกต้อง การนำข้อมูลที่ถูกต้องหรือทันสมัยมาดำเนินการขั้นต่อไป รวมทั้งเพื่อให้การพัฒนาซอฟต์แวร์เป็นไปในมาตรฐานเดียวกันและพนักงานทุกคนเข้าใจร่วมกัน

1. <u>การจัดตั้ง Library ของโครงการ และการจัดเก็บไฟล์</u>

1) SCM ผู้ได้รับมอบหมายโครงการเปิด directory ชื่อโครงการนั้นๆ โดยมีการ share และให้สิทธิการใช้งานแก่เจ้าหน้าที่ที่เกี่ยวข้อง พร้อมทั้งประกาศให้ผู้เกี่ยวข้องรับทราบต่อไป

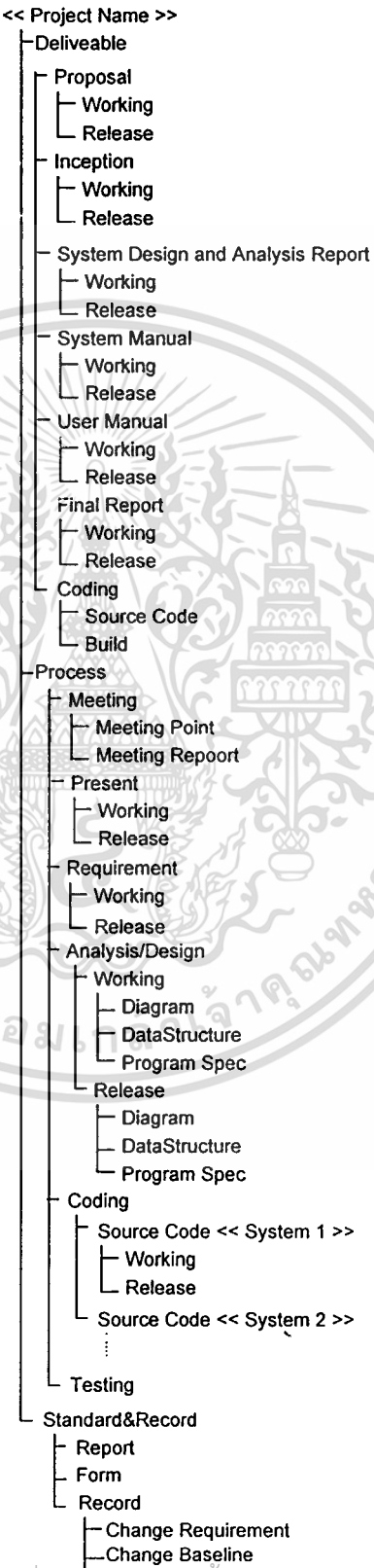
2) เปิด folder งานต่างๆ ดังนี้

- ตั้ง folder ชื่อ process ซึ่งจะแยกย่อยตามงานหลักๆ ของโครงการ เช่น ตามด้านของงานที่ต้องดำเนินการ เป็นต้น ถ้าโครงการใดไม่มีการแบ่งงานหลัก ก็ให้ดำเนินการเปิด Floder สำหรับ Deliverable
- ตั้ง folder ชื่อ deliverable ซึ่งจะแยกย่อยตามงานที่ต้องส่งให้กับลูกค้าต่อไป เช่น proposal, inception, existing, progress, master_plan, user_manual, tech_manual, code เป็นต้น

3) เปิด folder ย่อยของแต่ละ folder หลัก ดังนี้

- folder: process ตั้ง folder ย่อยตามขั้นตอนของการพัฒนา เช่น req_capture, design, coding, testing, maintain, quality, conceptual_work เป็นต้น) ซึ่งแต่ละส่วนงานอาจมีการแยกย่อยตาม section ที่สำคัญเช่น meeting, process, product เป็นต้น) เพื่อง่ายต่อการติดตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Inception Report) บทที่ 1 Version 1.0 จะตั้งชื่อเป็น I_ch1Pic1_V1.0.jpg

3) ไฟล์โปรแกรม

- กำหนดให้ขึ้นด้วยชื่อระบบ ตามด้วยตัวเลข Running No. และ Version ดังนี้ <<***System NameXXX _Version***>>

4. ข้อกำหนดเรื่องรูปแบบ ใน source code

1) การเขียน comment

ในส่วนเริ่มต้นของฟังก์ชันให้ประกอบด้วยข้อมูลดังนี้ คือ

- กรณีเป็นการเริ่มเขียนฟังก์ชันนั้นครั้งแรก
  - ชื่อฟังก์ชัน
  - วันที่สร้าง
  - ชื่อผู้สร้าง
  - หน้าที่การทำงานหลัก
  - Input/Output
- กรณีเป็นการเพิ่มเติม/แก้ไขฟังก์ชันที่มีอยู่
  - ครั้งที่
  - ชื่อผู้ทำการเพิ่มเติม/แก้ไข
  - วัตถุประสงค์ที่ทำการแก้ไข
  - ให้ทำการ comment บรรทัดที่ทำการเพิ่มเติมแก้ไขนั้นด้วยว่าเป็นการแก้ไขครั้งที่เท่าใด
- นอกจากส่วนหัวของฟังก์ชันแล้วแล้ว ควรทำการ comment ตัว source code เป็นระยะ เพื่อให้ผู้ที่จะทำการแก้ไขต่อไปสามารถอ่านแล้วเข้าใจได้

2) การตั้งชื่อตัวแปรและ control ในระบบ ให้ตั้งชื่อตามความหมายข้อมูลที่เชื่อมโยงกับตัวแปรนั้น ไม่ควรตั้งชื่อเป็นตัวเลขที่ไม่สามารถสื่อความหมายได้ มีแนวทางการตั้งชื่อดังนี้คือ

การตั้งชื่อตัวแปร

Character แรก หมายถึง Scope ของตัวแปร

G – Global Variable

I – Instance Variable

S – Share Variable

L – Local Variable

A - Argument Variable

Character ที่สองหมายถึง Type ของตัวแปร

S – String

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D – Date

N - Integer , Float , Double , Single , Long , Byte

B – Boolean

T – Transaction Object

ตัวอย่าง Global int GN_mun1

การตั้งชื่อคอนโทรล

ให้ใช้ตัวย่อที่บอกถึงชนิดของ control นำหน้า โดยตัวย่อต่างๆจะตั้งตามชนิดของ control ดังนี้

Control Type	Control Name
Window	W_xxxx
Datawindow control	Dw_xxxx
Datawindow object	D_xxxx
Single line edit	Sle_xxx
Command button	Cb_xxxx
Dropdown listbox	Ddlb_xxxx
Listbox	Lb_xxxx
Picture button	Pb_xxx
Checkbox	Cbx_xxxx
Radiobutton	Rb_xxxx
Statictext	1.1.1.1 St_xxxx
Picture	P_xxxx
Gropbbox	Gb_xxxx
Line	Ln_xxxx
Editmask	Em_xxxx
Multi line edit	Mle_xxxx
Rich text edit	Rte_xxxx
Horizontal Scroll bar	Hsb_xxxx
Vertical scroll bar	Vsb_xxxx
Tab page	Tabpage_xxxx
Tab	Tab_xxxx
User object	U_xxxx

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCM_guideline_v1.0

5. ข้อกำหนดเรื่องรูปแบบเอกสาร

1) เนื้อหาในเอกสารใช้ font AngsanaUPC, 16, Regular

2) เนื้อหาในส่วนที่เป็นตาราง Hearder ของตาราง ใช้ font AngsanaUPC, 16, Bold ส่วนเนื้อหาใช้ font AngsanaUPC, 14, Regular

3) ชื่อเรื่องหรือชื่อบท (title) ใช้ font AngsanaUPC, 20, Bold มีกรอบล่าง

4) หัวข้อให้มีหัวข้อได้ 2 ระดับ

- ระดับแรก ใช้ font AngsanaUPC, 18, Bold space before = 8 space after = 8
- ระดับที่ 2 ใช้ font AngsanaUPC, 16, Bold space before = 6 space after = 6

5) ในส่วนของ Footer ใช้ font EucrosiaUPC, 12, Regular มีกรอบบน

6) กำหนด Page Margin ดังนี้

- ขอบบน = 1"
- ขอบล่าง = 1"
- ขอบซ้าย = 1.25"
- ขอบขวา = 1"
- โดยมี header และ footer = 0.5"

6. การกำหนดสิทธิการใช้ข้อมูลและการสำรองข้อมูล

1) จัดกลุ่มและกำหนดสิทธิของผู้ใช้ข้อมูลของโครงการในแต่ละ Folder โดย ให้มีการกำหนดผู้รับผิดชอบของโครงการในการดำเนินการนี้ และแจ้งต่อผู้เกี่ยวข้องในโครงการ

2) กำหนดระยะเวลา วิธีการและผู้รับผิดชอบเพื่อการสำรองข้อมูลในแต่ละส่วนของ Baseline Library ของโครงการ

รูปที่ 3.7 ตัวอย่างเอกสารแนวทางการดำเนินงานด้านการจัดการองค์ประกอบซอฟต์แวร์

3.8 เอกสารความต้องการของระบบ

เอกสารความต้องการของระบบ เป็นเอกสารที่ได้จากกระบวนการบริหารจัดการความต้องการของผู้ใช้ ซึ่งต้องได้รับการตรวจทานจากกลุ่มบุคคลที่เกี่ยวข้องก่อนที่จะจัดเก็บไว้ใน Library เอกสารจะระบุถึงความต้องการทั้งที่เป็นความต้องการด้านเทคนิคของซอฟต์แวร์และความต้องการที่ไม่ได้เป็นเรื่องของเทคนิค ซึ่งอาจจะมีผลกระทบและเป็นตัวกำหนดกิจกรรมที่จะต้องมีในโครงการ และนอกจากนี้ยังรวมถึงเงื่อนไขการตรวจรับระบบ ซึ่งจะใช้ในการตรวจสอบว่าผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภัณฑ์ที่ได้ตรงกับความต้องการหรือไม่ด้วย โดยเอกสารนี้สามารถอ้างอิงได้กับ KPA ที่ RM/AB.1 และ RM/AB.2

เอกสารความต้องการของระบบควรมีรูปแบบดังนี้คือ

RM_SRS_v1.0

ข้อกำหนดความต้องการ

ระบบ << *ชื่อระบบ* >>

1. บทนำ
   กล่าวถึงภาพรวมของโครงการโดยย่อ ความเป็นมาของระบบและเหตุผลความต้องการมีระบบ
2. หน้าที่หลักของระบบ
   กล่าวถึงหน้าที่การทำงานหลักๆ ของระบบที่ระบบต้องสามารถทำงานได้ รวมทั้งหน้าจอและรายงานที่ต้องการ
3. External interface requirement
   กล่าวถึงความต้องการในการเชื่อมโยงกับระบบอื่นๆ (ถ้ามี)
4. ความต้องการด้านทรัพยากร สภาพแวดล้อมในการพัฒนาระบบ
   กล่าวถึงความต้องการในด้าน ฮาร์ดแวร์ ซอฟต์แวร์ และอื่นๆ เพื่อใช้ในการพัฒนาระบบ
5. ความต้องการด้านประสิทธิภาพของระบบ
   กล่าวถึงความต้องการในเรื่องของประสิทธิภาพการทำงานของระบบ
6. ความต้องการด้านมาตรฐาน
   กล่าวถึงความต้องการในเรื่องของมาตรฐานที่จะใช้ในการพัฒนาหรือมาตรฐานของผู้ใช้ (ถ้ามี)
7. ความต้องการเพิ่มเติมอื่นๆ
   กล่าวถึงความต้องการเพิ่มเติมที่นอกเหนือจากหน้าที่การทำงานหลัก (ถ้ามี) เช่น
   - ความต้องการในด้านความปลอดภัยของระบบ
   - ความต้องการในด้านความน่าเชื่อถือของระบบ
   - ความต้องการในด้านการรองรับข้อมูลจำนวนมาก
   - ความต้องการในด้านการสำรองข้อมูล
   - ความต้องการในเรื่องการกระจายฐานข้อมูล
   - ความต้องการในโอนถ่ายข้อมูล
   - ความต้องการในเรื่องการอบรม
   - ความต้องการในเรื่องของคู่มือการใช้งาน
8. เงื่อนไขการตรวจรับ
   กล่าวถึงเงื่อนไขต่างๆ ที่จะใช้ในการตรวจรับระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RM_SRS_v1.0

9. ข้อจำกัด

กล่าวถึงข้อจำกัดในการพัฒนาระบบ

10. prototype

กรณีมีการใช้ prototype เป็นเครื่องมือช่วยในการรวบรวมความต้องการ ให้ระบุถึง prototype ด้วย

ลงชื่อ ..... วันที่ .....

(SE)

ลงชื่อ ..... วันที่ .....

(PM)

ลงชื่อ ..... วันที่ .....

(SM)

ลงชื่อ ..... วันที่ .....

(ผู้ใช้ระบบ)

รูปที่ 3.8 ตัวอย่างเอกสารแนวทางการดำเนินงานด้านการจัดการองค์ประกอบซอฟต์แวร์

การจัดทำเอกสารนี้ควรจะมีเวอร์ชัน เพื่ออ้างอิงได้ในกรณีที่มีการเปลี่ยนแปลง รวมถึงควรมีลายเซ็นของผู้เกี่ยวข้องเพื่อยืนยันการตรวจทานและยืนยันความต้องการกับผู้ใช้ด้วย ทั้งนี้ในการจัดทำเอกสารข้อกำหนดความต้องการตามรูปแบบที่กำหนดไว้นั้น หัวข้อใดที่ไม่จำเป็นหรือไม่มีความต้องการตามนั้น สามารถตัดออกได้ และควรจะทำการตรวจทานทั้งจากผู้เกี่ยวข้องและผู้ใช้ระบบให้เรียบร้อยก่อนที่จะจัดเก็บไว้ใน Library

3.9 แผนการพัฒนาซอฟต์แวร์และแผนการติดตามงาน

เอกสารแผนการพัฒนาซอฟต์แวร์และแผนการติดตามงาน เป็นเอกสารที่เป็นผลลัพธ์จากกระบวนการวางแผนโครงการซอฟต์แวร์ จัดทำขึ้นเพื่อใช้ในการดำเนินงานโครงการและติดตามงานตั้งแต่เริ่มต้นจนสิ้นสุดการพัฒนาซอฟต์แวร์ เอกสารฉบับนี้จะต้องได้รับการตรวจทานจากกลุ่มบุคคลที่เกี่ยวข้องก่อนที่จะนำไปจัดเก็บไว้ใน Library และสามารถอ้างอิงได้ถึง KPA ที่ SPP/AC.7 และ SPTO/AC.8

เอกสารแผนการพัฒนาซอฟต์แวร์ ควรจะประกอบไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPP_Plan_v1.0

แผนการพัฒนาซอฟต์แวร์

ระบบ <<ชื่อระบบ>>

1. บทนำ
   - 1.1 วัตถุประสงค์ของแผนงานโครงการ
   - 1.2 ขอบเขตการดำเนินงาน
2. การประมาณการโครงการ
   - 2.1 เทคนิคที่ใช้ในการประมาณการ
   - 2.2 การประมาณการขนาดของซอฟต์แวร์
   - 2.3 การประมาณการด้านแรงงาน และเวลา
3. การบริหารความเสี่ยง
   - 3.1 รายการความเสี่ยง
   - 3.2 ความเสี่ยงที่ต้องดำเนินการจัดการ
   - 3.3 แผนการดำเนินการกับความเสี่ยง
4. แผนการดำเนินงาน
   - 4.1 วิธีพัฒนาซอฟต์แวร์ที่เลือกใช้
   - 4.2 Gantt Chart และ MileStone
5. ทรัพยากรที่จำเป็นต้องใช้
   - 5.1 บุคคล
   - 5.2 ฮาร์ดแวร์และซอฟต์แวร์
   - 5.3 ทรัพยากรอื่นๆ
6. ทีมงาน
   - 6.1 โครงสร้างทีมงาน
   - 6.2 รายชื่อและหน้าที่ความรับผิดชอบของทีมงาน
7. กระบวนการควบคุมและติดตามงาน
   - 7.1 การควบคุมและการประกันคุณภาพ
   - 7.2 การควบคุมและจัดการการเปลี่ยนแปลง
   - 7.3 การควบคุมและติดตามงาน

ภาคผนวก

ลงชื่อ ..... วันที่ .....

(SE)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPP_Plan_v1.0

ลงชื่อ ..... วันที่ .....

(PM)

รูปที่ 3.9 ตัวอย่างเอกสารแนวทางการดำเนินงานด้านการจัดการองค์ประกอบซอฟต์แวร์

การจัดทำเอกสารนี้ควรจะมีเวอร์ชัน เพื่ออ้างอิงได้ในกรณีที่มีการเปลี่ยนแปลง รวมถึงควรมีลายเซ็นของผู้เกี่ยวข้องเพื่อยืนยันการตรวจทานก่อนนำไปใช้งานจริง และจัดเก็บไว้ใน Library ทั้งนี้ในการจัดแผนงานโครงการตามรูปแบบที่กำหนดไว้นั้น หัวข้อใดที่ไม่จำเป็นหรือไม่มีกระบวนการตามนั้น สามารถตัดออกได้

3.10 แผนการประกันคุณภาพ

เอกสารแผนการประกันคุณภาพ เป็นเอกสารที่เป็นผลลัพธ์จากกระบวนการดำเนินการด้านการรับประกันคุณภาพ จัดทำขึ้นเพื่อใช้ในการดำเนินงานโครงการตั้งแต่เริ่มต้นจนสิ้นสุดการพัฒนาซอฟต์แวร์ เอกสารฉบับนี้จะต้องได้รับการตรวจทานจากกลุ่มบุคคลที่เกี่ยวข้องก่อนที่จะนำไปจัดเก็บไว้ใน Library และการจัดทำแผนฯ นี้สามารถนำไปอ้างอิงถึง KPA ที่ SQA/AC.1

เอกสารแผนการประกันคุณภาพ ควรจะประกอบไปด้วย

SQA_Plan_v1.0

แผนการประกันคุณภาพ

ระบบ << ชื่อระบบ >>

1. วัตถุประสงค์ของแผน
2. เอกสารอ้างอิง
3. การจัดการ
   - 3.1 ทรัพยากรที่ใช้
   - 3.2 กิจกรรมที่เกี่ยวข้องกับ SQA
   - 3.3 แผนการดำเนินงาน
   - 3.4 เอกสารจากกิจกรรม SQA
4. มาตรฐานและการวัดผล
5. การตรวจทานและการตรวจสอบ
6. การทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SQA_Plan_v1.0

7. การรายงานผลของ SQA
8. เครื่องมือ วิธีการ ที่ใช้
9. การควบคุม code และการควบคุมสื่อ
10. การควบคุม supplier
11. การเก็บ บำรุงรักษา การบันทึกต่างๆ
12. การอบรม
13. การบริหารจัดการความเสี่ยง

ลงชื่อ ..... วันที่ .....

(SE)

ลงชื่อ ..... วันที่ .....

(PM)

รูปที่ 3.10 ตัวอย่างรูปแบบเอกสารแผนการรับประกันคุณภาพ

การจัดทำเอกสารนี้ควรจะมีเวอร์ชัน เพื่ออ้างอิงได้ในกรณีที่มีการเปลี่ยนแปลง รวมถึงควรมีลายเซ็นของผู้เกี่ยวข้องเพื่อยืนยันการตรวจทานก่อนนำไปใช้งานจริงและจัดเก็บใน Library ทั้งนี้ในการจัดแผนการประกันคุณภาพตามรูปแบบที่กำหนดไว้นั้น หัวข้อใดที่ไม่จำเป็นหรือไม่มีกระบวนการตามนั้น สามารถตัดออกได้

3.11 แผนการจัดการองค์ประกอบซอฟต์แวร์

เอกสารแผนการจัดการองค์ประกอบซอฟต์แวร์ เป็นเอกสารที่เป็นผลลัพธ์จากกระบวนการดำเนินการด้านการจัดการองค์ประกอบซอฟต์แวร์ จัดทำขึ้นเพื่อใช้ในการดำเนินงานโครงการตั้งแต่เริ่มต้นจนสิ้นสุดการพัฒนาซอฟต์แวร์ เอกสารฉบับนี้จะต้องได้รับการตรวจทานจากกลุ่มบุคคลที่เกี่ยวข้องก่อนที่จะนำไปจัดเก็บไว้ใน Library และการจัดทำแผนฯ นี้สามารถนำไปอ้างอิงถึง KPA ที่ SCM/AC.2

เอกสารแผนการจัดการองค์ประกอบซอฟต์แวร์ ควรจะประกอบไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCM_Plan_v1.0

แผนการจัดการองค์ประกอบซอฟต์แวร์

ระบบ << ชื่อระบบ >>

1. บทนำ
2. หน้าที่ความรับผิดชอบของ SCM
3. กิจกรรมของ SCM
   - 3.1 การจัดการข้อมูล
   - 3.2 การระบุองค์ประกอบซอฟต์แวร์
   - 3.3 การควบคุมรายการองค์ประกอบซอฟต์แวร์
   - 3.4 การบันทึกและรายงานสถานะการเปลี่ยนแปลงองค์ประกอบซอฟต์แวร์
   - 3.5 การตรวจสอบองค์ประกอบซอฟต์แวร์ (Configuration Audit)
   - 3.6 การดำเนินงานโครงการ
4. แผนการดำเนินงาน
5. การกำหนดรายการองค์ประกอบซอฟต์แวร์
6. แนวทางการควบคุมองค์ประกอบซอฟต์แวร์

ลงชื่อ ..... วันที่ .....

(SE)

ลงชื่อ ..... วันที่ .....

(SM)

รูปที่ 3.11 ตัวอย่างเอกสารรูปแบบแผนการจัดการองค์ประกอบซอฟต์แวร์

การจัดทำเอกสารนี้ควรจะมีเวอร์ชัน เพื่ออ้างอิงได้ในกรณีที่มีการเปลี่ยนแปลง รวมถึงควรมีลายเซ็นของผู้เกี่ยวข้องเพื่อยืนยันการตรวจทานก่อนนำไปใช้งานจริงและจัดเก็บใน Library ทั้งนี้ในการจัดทำแผนการจัดการองค์ประกอบซอฟต์แวร์ตามรูปแบบที่กำหนดไว้นั้น หัวข้อใดที่ไม่จำเป็นหรือไม่มีกระบวนการตามนั้น สามารถตัดออกได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.12 เอกสารการขอเปลี่ยนแปลงความต้องการ

เอกสารการขอเปลี่ยนแปลงความต้องการ จัดทำขึ้นตามกระบวนการบริหารจัดการความต้องการของผู้ใช้และการจัดการองค์ประกอบซอฟต์แวร์ เพื่อให้การดำเนินการเป็นไปตามกระบวนการที่ได้กำหนดและมีเอกสารอ้างอิงในการประเมินผลงานต่อไป โดยแม่แบบของเอกสารนี้จะเป็น CI ตัวหนึ่งที่ต้องจัดเก็บใน Library และการจัดทำเอกสารนี้สามารถอ้างอิงได้ถึง KPA ที่ RM/AC.3

เอกสารการขอเปลี่ยนแปลงความต้องการ มีรูปแบบดังนี้

RM_ChForm_v1.0 **แบบฟอร์มการขอเปลี่ยนแปลงความต้องการ**	
โครงการ :	หมายเลข
ผู้ขอเปลี่ยนแปลง :	วันที่ :
ความต้องการที่ขอ :	
ผู้วิเคราะห์ :	วันที่วิเคราะห์ :
การประเมินผลกระทบ 1. 2.	
ความเร่งด่วน : (ด่วนที่สุด, ด่วน, ไม่เร่งด่วน)	จำนวนแรงงานที่ใช้ :
ความเห็นของ PM :	
ผลการดำเนินการ :	
วันที่ดำเนินการ :	
หมายเหตุ :	

รูปที่ 3.12 ตัวอย่างเอกสารแบบฟอร์มการขอเปลี่ยนแปลงความต้องการ

เมื่อมีความต้องการเพิ่มเติมหรือเปลี่ยนแปลง SE วิเคราะห์และนำมาหารือกับผู้เกี่ยวข้อง จากนั้นเขียนแบบฟอร์มข้างต้นเพื่อให้ PM ทำการอนุมัติและดำเนินการแก้ไข จากนั้นส่งให้ SCM รวบรวมจัดเก็บไว้ใน Library ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.13 เอกสารการขอเปลี่ยนแปลง baseline

เอกสารการขอเปลี่ยนแปลง baseline จัดทำขึ้นตามกระบวนการจัดการองค์ประกอบซอฟต์แวร์ เพื่อให้การดำเนินการเป็นไปตามกระบวนการที่ได้กำหนดและมีเอกสารอ้างอิงในการดำเนินงานต่อไป โดยแม่แบบของเอกสารนี้จะเป็น CI ตัวหนึ่งที่ต้องจัดเก็บใน Library และการจัดทำเอกสารนี้สามารถอ้างอิงได้ถึง KPA ที่ SCM/AC.3 และ SCM/AC.4

เอกสารการขอเปลี่ยนแปลง baseline มีรูปแบบดังนี้

SCM_ChBLForm_v1.0 **แบบฟอร์มการขอเปลี่ยนแปลง baseline**	
โครงการ :	หมายเลข
ผู้ขอเปลี่ยนแปลง :	วันที่ :
Baseline ที่ขอเปลี่ยนแปลง :	
เหตุผลในการขอเปลี่ยนแปลง : อ้างอิงจาก :	
ความเห็นของ SCM :	
ผลการดำเนินการ :	
วันที่ดำเนินการ :	
หมายเหตุ :	

รูปที่ 3.13 ตัวอย่างเอกสารแบบฟอร์มการขอเปลี่ยนแปลง baseline

เมื่อมีการเปลี่ยนแปลงใดๆ ที่ทำให้เกิดผลกระทบต่อ baseline ให้ผู้เกี่ยวข้องหารือร่วมกัน จากนั้นเขียนแบบฟอร์มข้างต้นส่งให้ SCM เพื่อขออนุญาตแก้ไขเปลี่ยนแปลง baseline และรวบรวมจัดเก็บไว้ใน Library เพื่อใช้ในการอ้างอิงต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.14 เอกสารการขอเปลี่ยนแปลงซอฟต์แวร์

เอกสารการขอเปลี่ยนแปลงซอฟต์แวร์ จัดทำขึ้นตามกระบวนการจัดการองค์ประกอบซอฟต์แวร์ เพื่อให้การดำเนินการเป็นไปตามกระบวนการที่ได้กำหนดและมีเอกสารอ้างอิงในการดำเนินงานต่อไป โดยแม่แบบของเอกสารนี้จะเป็น CI ตัวหนึ่งที่ต้องจัดเก็บใน Library และการจัดทำเอกสารนี้สามารถอ้างอิงได้ถึง KPA ที่ SCM/AC.3 และ SCM/AC.4

เอกสารการขอเปลี่ยนแปลงซอฟต์แวร์ มีรูปแบบดังนี้

SCM_ChSWForm_v1.0 **แบบฟอร์มการขอเปลี่ยนแปลงซอฟต์แวร์**	
โครงการ :	หมายเลข
ผู้ขอเปลี่ยนแปลง :	วันที่ :
Baseline ที่ต้องการแก้ไข :	
Module ที่ต้องการแก้ไข :	
เหตุผลที่ขอแก้ไข :	
ผู้วิเคราะห์ :	วันที่วิเคราะห์ :
ความเร่งด่วน : (ด่วนที่สุด, ด่วน, ไม่เร่งด่วน)	จำนวนแรงงานที่ใช้ :
ความเห็นของ SCM :	
ผลการดำเนินการ :	
วันที่ดำเนินการ :	
หมายเหตุ :	

รูปที่ 3.14 ตัวอย่างเอกสารแบบฟอร์มการขอเปลี่ยนแปลงซอฟต์แวร์

แบบฟอร์มนี้จะใช้ในกรณีการขอเปลี่ยนแปลง baseline เฉพาะเมื่อ baseline นั้นเป็น software ถ้า baseline เป็นเอกสาร จะใช้แบบฟอร์มก่อนหน้านี้แทน โดยเมื่อมีการเปลี่ยนแปลงที่มีผลกระทบต่อ software ที่เป็น baseline ให้ SE วิเคราะห์และนำมาหารือกับผู้เกี่ยวข้อง จากนั้นเขียนแบบฟอร์มข้างต้นเพื่อขออนุญาตแก้ไข software แล้วส่งให้ SCM รวบรวมจัดเก็บไว้ใน Library เพื่อใช้อ้างอิงต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.15 รายงานการวัดผลการดำเนินงานกิจกรรม

การวัดผลการดำเนินงานกิจกรรมในแต่ละกลุ่มกระบวนการหลักนั้น เป็น key practice ตัวหนึ่งใน common feature ซึ่งจะเป็นรายงานที่สามารถใช้ประเมินผลการดำเนินการในแต่ละกิจกรรมว่ามีสถานะเป็นอย่างไร และผลลัพธ์ที่ได้ตรงตามที่ต้องการหรือไม่ ตัวอย่างรายงานการวัดผลการดำเนินงานของแต่ละกิจกรรม เช่น

RM

- รายงานสรุปสถานะความต้องการแต่ละตัว
- รายงานสรุปจำนวนการเปลี่ยนแปลงความต้องการ

SPP

- รายงานสรุปแผนการทำงานเปรียบเทียบกับ milestone
- รายงานสรุปแผนการทำงานเปรียบเทียบกับต้นทุนและแรงงานที่ใช้ไปจริง

SPTO

- รายงานสรุปผลการติดตามงาน และผลกระทบต่อแผนงานโครงการ
- รายงานสรุปต้นทุนและแรงงานที่ใช้ในการติดตามงาน

SQA

- รายงานสรุปผลการตรวจสอบกิจกรรม SQA
- รายงานสรุปผลการดำเนินการของกิจกรรม SQA เปรียบเทียบกับแผนที่วางไว้

SCM

- รายงานสรุปจำนวนการขอเปลี่ยนแปลง CI
- รายงานสรุปผลการดำเนินงานของกิจกรรม SCM เปรียบเทียบกับแผนที่วางไว้

รายงานเหล่านี้จะจัดทำเมื่อถึงเวลาที่กำหนด หรือเมื่อต้องการสรุปผลการดำเนินงานทั้งหมด โดยจะนำเสนอตัวอย่างของรายงานนี้บางตัวในบทต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

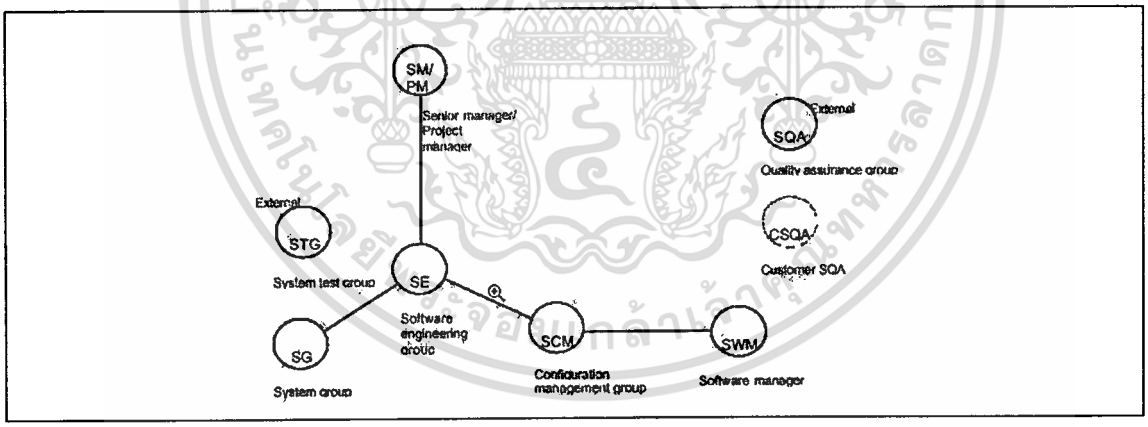
บทที่ 4

การประยุกต์ใช้ CMM ระดับที่ 2 กับกรณีศึกษา: ระบบบริหารคลังวัตถุดิบ

ในกรณีศึกษานี้ จะสมมติให้บริษัทที่รับจ้างพัฒนาซอฟต์แวร์เป็นบริษัทเล็กๆ ที่มีพนักงาน 1-2 คน และซอฟต์แวร์ที่จ้างพัฒนาเป็นซอฟต์แวร์ในการบริหารคลังวัตถุดิบของโรงพิมพ์ขนาดเล็ก

	1-2 คน	3-15 คน	> 15 คน
1 โครงการ	XXS	XS	S
2-5 โครงการ	-	XS	S
> 6 โครงการ	-	S	S

เมื่อเปรียบเทียบจำนวนพนักงานและโครงการในตารางแล้ว จะเห็นว่าโครงการพัฒนาซอฟต์แวร์ตัวนี้จัดเป็นโครงการขนาดเล็กมากๆ (XXS) ซึ่งมีบทบาทในการทำงานดังรูป



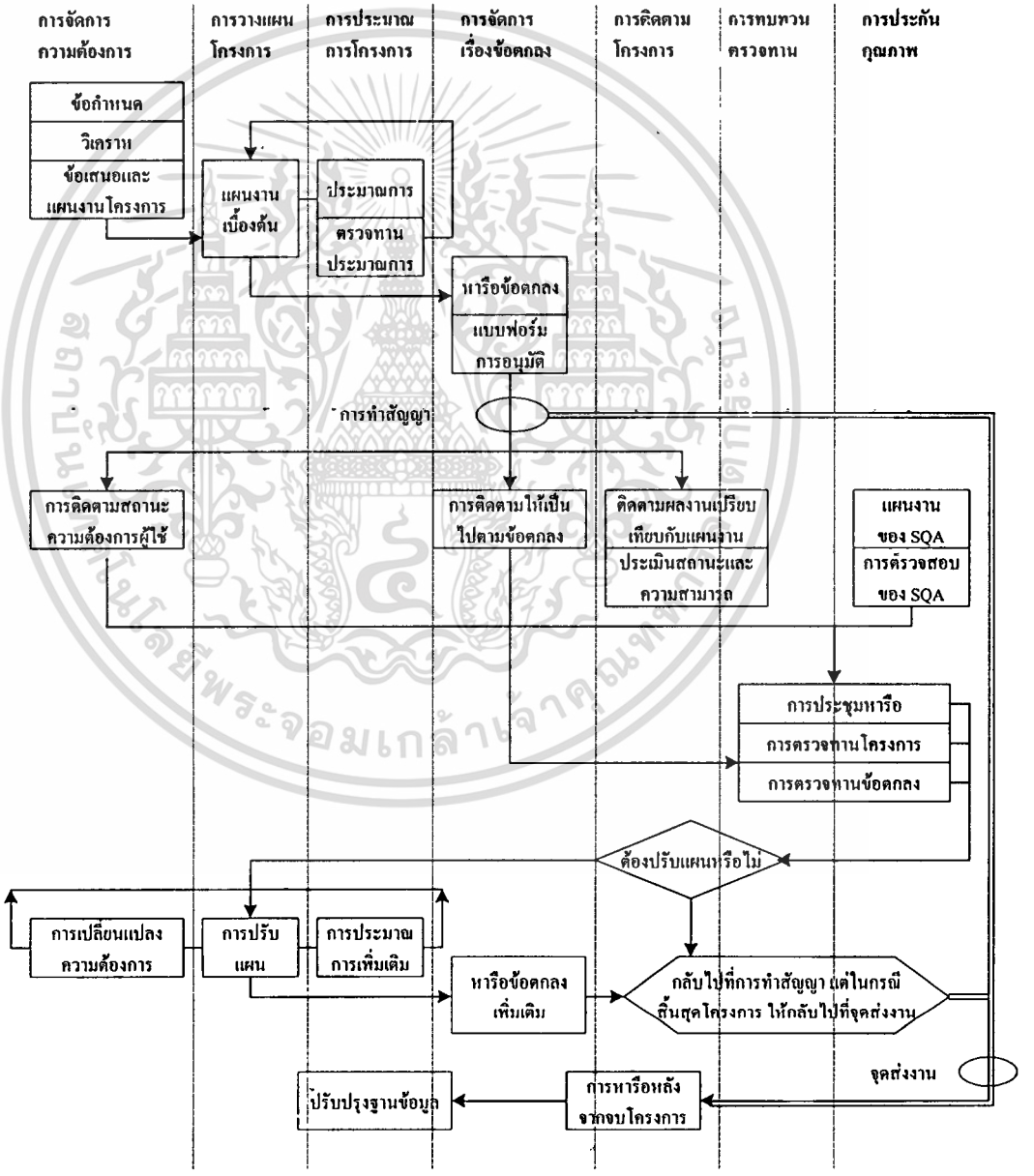
รูปที่ 4.1 แสดงบทบาทในโครงการขนาดเล็กมากๆ (XXS)

การบริหารจัดการโครงการในลักษณะนี้อาจจะไม่ต้องใช้การบริหารในรูปแบบที่เป็นทางการมาก คนหนึ่งคนสามารถรับผิดชอบได้หลายบทบาท หรืออาจจะรับผิดชอบในทุกบทบาทได้ แต่อย่างไรก็ตาม ก็ยังคงต้องมีกระบวนการในการบริหารจัดการที่ขาดไม่ได้ เช่น การจัดทำแผนงานโครงการและการติดตามงาน การจัดการองค์ประกอบซอฟต์แวร์อาจจะเป็นเพียงการพูดตกลงทำความเข้าใจ และแผนงานประกันคุณภาพอาจจะเป็นเพียงแค่ check list ในการตรวจสอบเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในงานวิจัยฉบับนี้ จะประยุกต์ใช้เอกสารที่กำหนดไว้ในบทที่ 3 กับการบริหารจัดการโครงการพัฒนาระบบบริหารคลังวัตถุดิบนี้ ตั้งแต่เริ่มต้นการพัฒนาจนสิ้นสุดการพัฒนาโครงการ โดยที่เอกสารนโยบายและเอกสารกระบวนการ จะถือว่าได้จัดทำไว้แล้ว และเอกสารรายงานการวัดผลต่างๆ จะแสดงตัวอย่างเพียงบางรายงานเท่านั้น

<u>กระบวนการทำงานตามแนวทางของ CMM ระดับที่ 2</u>



รูปที่ 4.2 กระบวนการในการทำงานตามแนวทางของ CMM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปกระบวนการทำงานตามแนวทางของ CMM ระดับที่ 2 นั้น จะสังเกตได้ว่าจะเน้นในเรื่องของการตรวจทานและการหารือร่วมกันเป็นหลัก จะมีการตรวจทานในทุกขั้นตอน ซึ่งเอกสารต่างๆ ที่ออกแบบไว้ในบทที่ 3 นั้น จะนำมาประยุกต์ใช้ได้โดยให้มีการลงชื่อทุกครั้งเมื่อทำการตรวจทานเอกสารและกระบวนการต่างๆ เรียบร้อยแล้ว

การนำเอกสารที่ออกแบบไว้มาประยุกต์ใช้กับระบบงานจริง จะนำมาแสดงเฉพาะเอกสารที่เป็นผลผลิตจากการพัฒนาซอฟต์แวร์เท่านั้น (Software work products) ส่วนเอกสารนโยบายและเอกสารกระบวนการ ได้จัดทำในบทที่ 3 ไว้แล้ว โดยเอกสารที่นำมาประยุกต์ใช้มีดังนี้คือ

4.1 ข้อกำหนดความต้องการ

InentRM_SRS_v1.0

ข้อกำหนดความต้องการ

ระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์

1. บทนำ

โรงพิมพ์สันติภาพ พริ้นท์ เป็นโรงพิมพ์ขนาดเล็ก มีพนักงานไม่เกิน 15 คน รับพิมพ์งานทุกชนิด เช่น หนังสือ วารสาร การ์ดแต่งงาน กล่องกระดาษขนาดบาง ด้วยระบบพิมพ์ออฟเซ็ทแบบครบวงจร ปัจจุบันโรงพิมพ์ฯ มีปัญหาในเรื่องการบริหารคลังวัตถุดิบ ไม่สามารถควบคุมการเบิกจ่ายวัตถุดิบ ไม่ทราบแน่ชัดถึงจำนวนคงเหลือของวัตถุดิบในคลัง ไม่ทราบเมื่อถึงจุดต้องสั่งซื้อ เป็นเหตุให้บางครั้งไม่สามารถสั่งซื้อวัตถุดิบได้ทันเวลา และทำให้ส่งงานล่าช้า ดังนั้น ทางโรงพิมพ์ฯ จึงต้องการระบบบริหารคลังวัตถุดิบเพื่อนำไปช่วยในการบริหารการเบิกจ่ายและรายงานเมื่อถึงจุดสั่งซื้อได้

เอกสารข้อกำหนดความต้องการฉบับนี้ จัดทำขึ้นเพื่ออธิบายถึงความต้องการของระบบการบริหารคลังวัตถุดิบโดยละเอียด โดยจากนี้จะเรียกว่า "ระบบ"

2. ความต้องการด้านหน้าที่การทำงาน

<u>ความต้องการของระบบ</u>

1) ระบบสามารถจัดเก็บบันทึกการรับวัตถุดิบจาก supplier ในแต่ละวันได้
2) ระบบสามารถจัดเก็บการเบิกจ่ายประจำวันได้
3) ระบบสามารถเรียกดูจำนวนคงเหลือของวัตถุดิบแต่ละประเภทได้
4) ระบบสามารถจัดทำรายงานตามวัน สัปดาห์ เดือน และปี ได้
5) ระบบสามารถจัดทำรายงานวัตถุดิบที่ถึงจุดสั่งซื้อได้

<u>ความต้องการหน้าจอ</u>

1) หน้าจอ login

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) หน้าจอการบันทึกข้อมูล

3) หน้าจอการเรียกดูจำนวนคงเหลือ

4) หน้าจอเลือกเงื่อนไขเพื่อแสดงรายงาน

5) หน้าจอเกี่ยวกับการจัดการตารางข้อมูลหลัก

ความต้องการรายงาน

1) รายงานการเบิกจ่ายและการรับวัตถุดิบ ประจำวัน สัปดาห์ เดือน และปี

2) รายงานการเบิกจ่ายรายบุคคล

3) รายงานการรับวัตถุดิบของ supplier แต่ละราย

4) รายงานวัตถุดิบที่ถึงจุดสั่งซื้อ

3. ความต้องการด้านทรัพยากรและสภาพแวดล้อม

ฮาร์ดแวร์

คอมพิวเตอร์ความเร็วอย่างน้อย Pentium-133 ,มีหน่วยความจำ 32 Mb และเนื้อที่ในฮาร์ดดิสก์ 50 MB

ซอฟต์แวร์

ระบบปฏิบัติการ window98 ขึ้นไป และควรมีฐานข้อมูล MS Access

4. ความต้องการด้าน performance

- เวลาในการเปิดหน้าจอแต่ละหน้าจอในครั้งแรก ไม่ควรเกิน 5 วินาที
- การบันทึกข้อมูลแต่ละหน้าจอไม่ควรใช้เวลาในการบันทึกเกิน 5 วินาที

5. ความต้องการเพิ่มเติมอื่นๆ

ความปลอดภัยและการกำหนดสิทธิ์

1) การเข้าสู่การใช้งานจะต้องใส่ชื่อผู้ใช้และรหัสผ่านที่ถูกต้อง

2) ผู้ใช้สามารถเปลี่ยนแปลงชื่อและรหัสผ่านได้เอง

3) สามารถตรวจสอบได้ว่าแต่ละ record ใครเป็นผู้เปลี่ยนแปลงแก้ไข

การสำรองและกู้คืนข้อมูล

1) ต้องมีหน้าจอให้ผู้ใช้สามารถทำการสำรองข้อมูลได้บ่อยเท่าที่ต้องการและนำข้อมูลที่สำรองกลับมาใช้ใหม่ได้

คู่มือการใช้งานและระบบช่วยเหลือ

1) เมื่อติดตั้งระบบในครั้งแรก จะต้องมีการอบรมการใช้งานให้กับผู้ใช้จำนวน 2 คน

2) ต้องจัดทำคู่มือการติดตั้งระบบและคู่มือการใช้งานระบบ

6. เงื่อนไขการตรวจรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

InentRM_SRS_v1.0

จะต้องมีการพัฒนาระบบให้ตรงกับความต้องการทุกอย่างที่กล่าวถึงในเอกสารนี้ ทั้งในเรื่องของความต้องการทางด้านหน้าที่การทำงาน ความต้องการทางด้านประสิทธิภาพการทำงาน และความต้องการอื่นๆ ทั้งหมด

7. prototype

ความต้องการในระดับรายละเอียดจะต้องดูจาก User Interface ดังนั้นการใช้ prototype เป็นวิธีการหนึ่งที่จะทำให้ผู้ใช้บอกถึงความต้องการในรายละเอียดได้ โดยเมื่อได้ prototype แล้ว ผู้ใช้จะต้อง

1) ตรวจสอบแต่ละหน้าจอและเพิ่มรายละเอียดที่ต้องการ
2) ตรวจสอบความต่อเนื่องของแต่ละหน้าจอที่เชื่อมโยงกัน
3) เซ็นชื่อเป็นหลักฐานการอนุมัติและการตรวจสอบหน้าจอแต่ละครั้ง

ผู้จัดทำ ลงชื่อ ..... วันที่ .....
(นางสาวดวงสมร พันธ์พิกุล)

ลงชื่อ ..... วันที่ .....
(ผู้ใช้ระบบ)

รูปที่ 4.3 ตัวอย่างการประยุกต์ใช้ข้อกำหนดความต้องการ

4.2 แผนงานโครงการและการติดตามงาน

InventSPP_Plan_v1.0

แผนงานโครงการ
ระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์

1. บทนำ

แผนงานโครงการซอฟต์แวร์ฉบับนี้จัดทำขึ้นเพื่อเป็นแนวทางในการพัฒนาระบบบริหารคลังวัตถุดิบของโรงพิมพ์สันติภาพ พริ้นท์ โดยจะครอบคลุมทั้งในเรื่องของการประมาณการแรงงาน ต้นทุน ตารางเวลาทำงาน การควบคุมและการติดตามงาน และความเสี่ยงที่อาจจะเกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การประมาณการโครงการ

ในการประมาณการโครงการ จะใช้เอกสารข้อกำหนดความต้องการเป็นหลัก

2.1 เทคนิคที่ใช้ในการประมาณการ

โครงการมีข้อกำหนดในเรื่องความซับซ้อนของระบบเป็น 3 ระดับคือ

- ระบบที่ไม่ซับซ้อน (1) การทำงานไม่ยุ่งยาก ไม่มีเงื่อนไขมาก และยุ่งเกี่ยวไม่เกิน 2-3 table ใช้เวลาในการพัฒนาไม่เกิน 1 วันต่อ 1 คน
- ระบบที่ซับซ้อนปานกลาง (2) มีเงื่อนไขเยอะขึ้น และมี table ที่ต้องเกี่ยวข้อง 2-4 table ใช้เวลาในการพัฒนาไม่เกิน 3 วันต่อ 1 คน
- ระบบที่ซับซ้อนมาก (3) มีเงื่อนไขที่ยุ่งยากและต้องดึงข้อมูลจาก table มากกว่า 4 table ขึ้นไป
- ใช้เวลาในการพัฒนาไม่เกิน 5 วันต่อ 1 คน

2.2 การประมาณการขนาดของซอฟต์แวร์

หน้าจอ	จำนวน module	ความซับซ้อน	จำนวนวัน
บันทึกข้อมูล	2 (x 5 module ย่อย)	2	3
เรียกดูข้อมูล	1 (x 3 module ย่อย)	2	1
เงื่อนไขรายงาน	4	2	3
บันทึกข้อมูลพื้นฐาน	6 (x 5 module ย่อย)	1	2

สรุปได้ว่าระบบมีขนาดไม่ใหญ่มากนักและมีความซับซ้อนไม่สูงมาก

2.3 การประมาณการแรงงานและเวลาที่ต้องใช้

การประมาณการจะใช้ข้อมูลจากโครงการที่เคยทำมาแล้วประกอบการตัดสินใจ

กิจกรรม	จำนวนวันต่อคน
วางแผนการดำเนินงาน	5
ขั้นตอนการรวบรวมความต้องการ	2
ขั้นตอนการวิเคราะห์ความต้องการ	3
ขั้นตอนการออกแบบระบบ	8
ขั้นตอนการ Coding และ unit test	9
ขั้นตอนการทดสอบ	3

ใช้เวลาในการพัฒนา 30 วันทำการ โดยประมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การบริหารความเสี่ยง

3.1 รายการความเสี่ยง

ความเสี่ยงที่อาจจะเกิดขึ้นได้ในการพัฒนาระบบนี้คือ

1) ผู้ใช้ไม่มีเวลาเพียงพอในการตรวจสอบเอกสารและ prototype ที่นำเสนอไป อาจจะทำให้เกิดความล่าช้าในการพัฒนาระบบ
2) อาจจะมีความต้องการใหม่ๆ และความต้องการที่เปลี่ยนแปลง ซึ่งอาจจะมีผลกระทบถึงโครงสร้างระบบและทำให้ระบบไม่สามารถพัฒนาระบบได้เสร็จทันเวลาที่กำหนด

3.2 แผนการดำเนินการกับความเสี่ยง

1) อธิบายให้ผู้ใช้เข้าใจถึงความจำเป็นและความเร่งด่วนในการอนุมัติเอกสารและ prototype ที่นำเสนอ เพื่อจะได้พัฒนาระบบได้อย่างต่อเนื่อง
2) ถ้ามีการเปลี่ยนแปลงความต้องการมาก อาจต้องหารือร่วมกับผู้ใช้ขอขยายระยะเวลาส่งมอบงาน

4. แผนการดำเนินงาน

4.1 การเลือกใช้วิธีการพัฒนาซอฟต์แวร์

ระบบบริหารคลังวัตถุดิบนี้ เป็นระบบขนาดเล็ก ไม่ซับซ้อนมากนัก ดังนั้นการใช้วิธีการพัฒนาโดยใช้แบบจำลองแบบ Waterfall Diagram ร่วมกับการใช้ prototype เป็นวิธีการพัฒนาที่เหมาะสม โดยประกอบด้วยขั้นตอนต่างๆ ดังนี้คือ

- การรวบรวมความต้องการ
- การวิเคราะห์ความต้องการ
- การออกแบบระบบ
- การพัฒนาระบบ
- การทดสอบระบบ

โดยอาจจะมีการทำงานควบคู่กันไปในบางช่วง เช่น ระหว่างที่พัฒนาระบบ อาจจะทำการออกแบบ Test case ไปด้วย เป็นต้น

4.2 Gantt Chart

การพัฒนาระบบบริหารคลังวัตถุดิบ มีระยะเวลาในการดำเนินงานพัฒนาประมาณ 1.5 เดือน โดยมีแผนการดำเนินงานตามรูป และ milestone เมื่อสิ้นสุดแต่ละขั้นตอนของการพัฒนามีดังนี้คือ

<u>ขั้นตอนการรวบรวมความต้องการ</u>

เอกสารข้อกำหนดความต้องการ

<u>ขั้นตอนการวิเคราะห์และการออกแบบระบบงาน</u>

รายงานการวิเคราะห์และออกแบบระบบ

<u>ขั้นตอนการพัฒนาระบบ</u>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบบริหารคลังวัตถุดิบ

ขั้นตอนการทดสอบระบบ

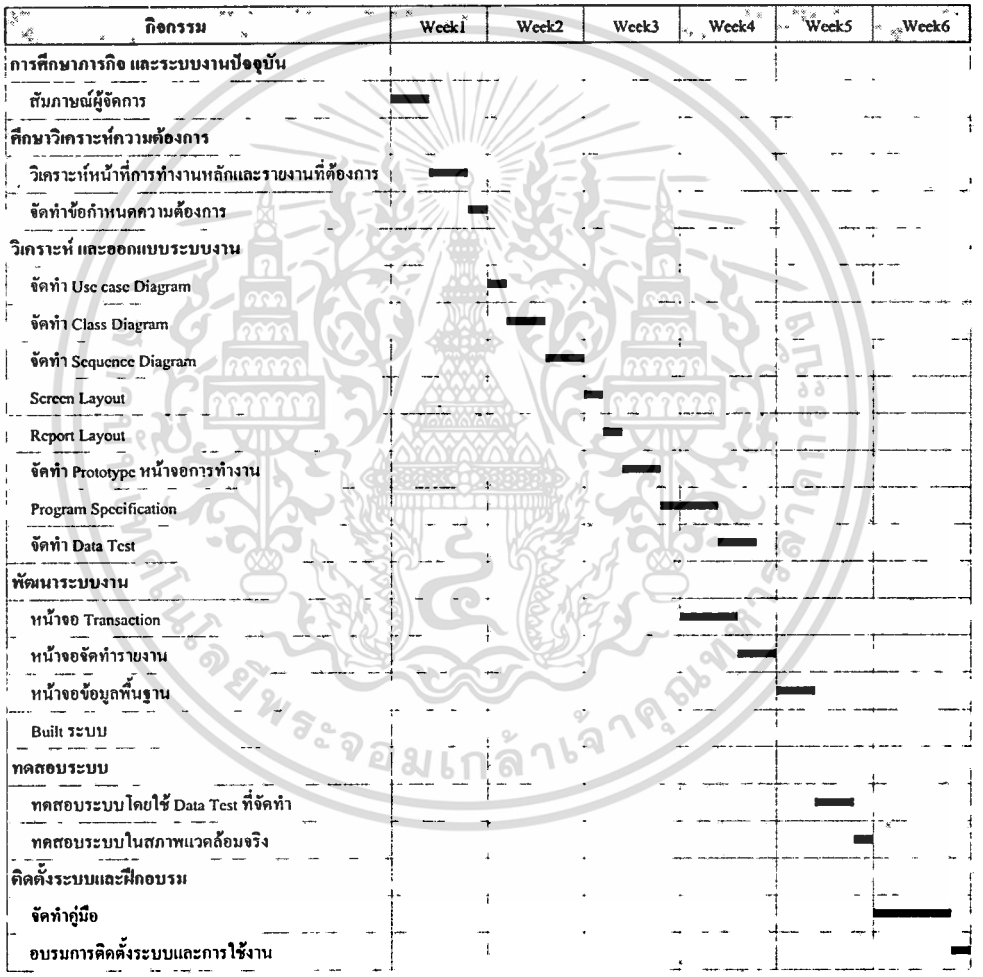
รายงานการทดสอบระบบ

ขั้นตอนการติดตั้งระบบ

โปรแกรมการติดตั้งระบบบริหารคลังวัตถุดิบ

คู่มือการใช้งาน

คู่มือการพัฒนาระบบ



5. ทรัพยากรที่จำเป็นต้องใช้

5.1 บุคคล

ระบบนี้ใช้ผู้พัฒนา 2 คน

5.2 ฮาร์ดแวร์และซอฟต์แวร์

ฮาร์ดแวร์ที่ใช้พัฒนาเป็น Celeron 500 MHz ระบบปฏิบัติการ Windows98

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องมือที่ใช้พัฒนา PowerBuilder 8.0 บนฐานข้อมูล MS Access 97

6. ทีมงาน

ระบบนี้มีทีมงานพัฒนา 2 คน ทำหน้าที่เป็นนักวิเคราะห์ระบบ 1 คนและผู้พัฒนาระบบ 1 คน มีรายชื่อดังนี้คือ

นักวิเคราะห์ระบบ นางสาวดวงสมร พันธ์พิกุล

โปรแกรมเมอร์ นางสาว.....

7. กระบวนการควบคุมและติดตามงาน

7.1 การควบคุมและการประกันคุณภาพ

มีการดำเนินการด้านการควบคุมและการประกันคุณภาพ โดยมีแผนการประกันคุณภาพตามเอกสาร InventSQA_Plan_v1.0

7.2 การควบคุมและจัดการการเปลี่ยนแปลง

มีการดำเนินการด้านการควบคุมและจัดการองค์ประกอบซอฟต์แวร์ โดยมีแผนการจัดการองค์ประกอบซอฟต์แวร์ ตามเอกสาร InventSCM_Plan_v1.0

7.3 กระบวนการบริหารและติดตามงาน

ในการบริหารและติดตามแผนการดำเนินงานโครงการ จะกำหนดแผนดังนี้คือ

1) ติดตามการทำงานจริงเปรียบเทียบกับแผนงานโครงการทุกวันศุกร์บ่าย และจัดทำรายงานสรุปผล
2) ติดตามผลลัพธ์ของ milestone ที่ได้กับแผนงานโครงการเมื่อถึงกำหนด milestone และจัดทำรายงานสรุปผล
3) ติดตามในเรื่องของต้นทุน แรงงาน และค่าใช้จ่าย เปรียบเทียบกับแผนงานโครงการที่ได้ประมาณการไว้ เมื่อปิดโครงการ

ผู้จัดทำ ..... วันที่ .....

(PM)

ตรวจทาน ..... วันที่ .....

(SM)

รูปที่ 4.4 ตัวอย่างการประยุกต์ใช้แผนการพัฒนาซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 แผนการประกันคุณภาพซอฟต์แวร์

InventSQA_Plan_v1.0

แผนการประกันคุณภาพซอฟต์แวร์
ระบบบริหารคลังวัตถุดิบโรงพิมพ์สันติภาพ พริ้นท์

1. วัตถุประสงค์

แผนการประกันคุณภาพฉบับนี้ จัดทำขึ้นเพื่อเป็นแนวทางในการควบคุมคุณภาพการพัฒนาซอฟต์แวร์ระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์ โดยจะระบุถึงกิจกรรมและงานด้าน SQA ที่เกี่ยวข้องกับโครงการ มาตรฐาน กระบวนการที่จะนำมาใช้ รวมถึงเครื่องมือที่จะนำมาใช้ดำเนินการ

2. เอกสารอ้างอิง

- เอกสารนโยบายในการพัฒนาโครงการซอฟต์แวร์ (Policy_v1.0)
- เอกสารกระบวนการทำงาน (SPP_Process_v1.0, RM_Process_v1.0, SPTO_Process_v1.0, SQA_Process_v1.0, SCM_Proces_v1.0)
- เอกสารแผนงาน (SPP_Plan_v1.0, SCM_Plan_v1.0, SCM_Guideline_v1.0)
- หนังสือเกี่ยวกับ CMM, SQA และ Software Engineering

3. การจัดการ

ทรัพยากรที่ใช้

- อุปกรณ์และสิ่งอำนวยความสะดวก เช่น คอมพิวเตอร์ เอกสารอ้างอิง (ตามข้อ 2) แบบฟอร์มต่างๆ และซอฟต์แวร์ที่ช่วยในการปฏิบัติงานด้าน SQA
- บุคลากรที่รับผิดชอบในการดำเนินกิจกรรม SQA ได้แก่ นางสาว ดวงสมร พันธ์พิกุล

กิจกรรมของ SQA

- กิจกรรมเกี่ยวข้องกับกระบวนการพัฒนา ทำการประเมินผลกิจกรรมทุกอย่างที่เกี่ยวข้องกับการพัฒนาซอฟต์แวร์ เช่นการวางแผนโครงการ การวิเคราะห์ความต้องการ ประเมินผลการออกแบบ เป็นต้น
- กิจกรรมเกี่ยวข้องกับผลิตภัณฑ์ ทำการทบทวน ผลิตภัณฑ์ทุกอย่างทั้งที่เป็นระบบและเป็นเอกสาร และประเมินผลเครื่องมือที่นำมาใช้ในโครงการ
- กิจกรรมเกี่ยวข้องกับการบริหารงาน SQA เช่นการจัดทำแผนการประกันคุณภาพ การจัดหาทรัพยากร จัดทำรายงานเกี่ยวกับ SQA เป็นต้น

แผนการดำเนินงาน

- ให้ทำ Process Audit ในแต่ละขั้นตอนของการพัฒนาซอฟต์แวร์
- ให้ทำ Spot Check คือการตรวจสอบโดยไม่บอกล่วงหน้า เป็นระยะ
- ให้ทำ Product Review คือประเมินผลผลิตภัณฑ์ที่ได้ระหว่างการพัฒนาซอฟต์แวร์ ตาม milestone เมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สิ้นสุดแต่ละกระบวนการ

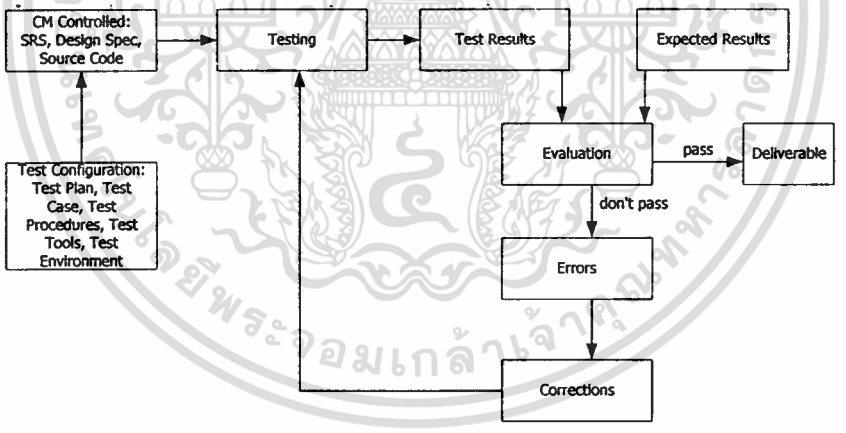
<u>เอกสารจากกิจกรรม SQA</u>

ในแต่ละขั้นตอนของการดำเนินกิจกรรม SQA จะได้เอกสารที่เกี่ยวข้องคือ

- แผนงานประกันคุณภาพซอฟต์แวร์
- รายงานการประเมินผลกระบวนการ
- รายงานการประเมินผลผลิตภัณฑ์
- แนวทางในการดำเนินงานต่อไป

4. มาตรฐานและการวัดผล
   - ใช้มาตรฐานตามที่กำหนดไว้ในองค์กร เช่น นโยบาย กระบวนการ และแนวทางต่างๆ และมาตรฐานจากภายนอก เช่นมาตรฐานของ CMM
   - การวัดผลจะพิจารณาจากขนาด ต้นทุน แรงงาน และตารางการทำงานของกิจกรรม SQA และรายงานผลต่อ PM

5. กระบวนการทดสอบ



6. การรายงานผลของ SQA

เมื่อ SQA ทำการตรวจสอบตามกระบวนการที่กำหนดไว้แล้ว ให้จัดทำรายงานในส่วนที่เกี่ยวข้องกับ

- ผลิตภัณฑ์
- กระบวนการ
- การวัดผลต่างๆ ทางด้านคุณภาพของซอฟต์แวร์ เช่น จำนวนข้อผิดพลาดที่พบ
- การวัดผลต่างๆ ทางด้านกระบวนการทำงาน เช่น เวลาที่ใช้ในการดำเนินการ เวลาที่ใช้ในการประเมินผล เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

InventSQA_Plan_v1.0

7. เครื่องมือ เทคนิค วิธีการที่ใช้

เนื่องจากในโครงการขนาดเล็กมีจำนวนบุคลากรไม่เพียงพอ ดังนั้นการตรวจสอบกระบวนการต่างๆ จะใช้วิธีทำ checklist เพื่อให้สามารถตรวจสอบได้อย่างรวดเร็ว ไม่เสียเวลามากนัก

8. การควบคุม Code และ media

วิธีการควบคุม Source และสื่อต่างๆ อธิบายไว้ในแนวทางการจัดการองค์ประกอบซอฟต์แวร์ (SCM_Guideline_v1.0) ซึ่ง SQA จะประเมินผลการควบคุมเพื่อให้แน่ใจว่าเป็นไปตามที่กำหนดไว้

9. การรวบรวมและบำรุงรักษาข้อมูล

กิจกรรม SQA ถูกบันทึกและทำรายงานตลอด SDLC ข้อมูลการวัดผลการดำเนินการจะถูกนำมาตรวจทานและเป็นแนวทางในการปรับปรุงกระบวนการ และจะรวบรวมข้อมูลทั้งหมดไว้ใน Library จนกว่าจะหมดช่วงของการบำรุงรักษาซอฟต์แวร์

ผู้จัดทำ ..... วันที่ .....

(SQA)

ผู้ตรวจทาน .....วันที่ .....

(PM)

รูปที่ 4.5 ตัวอย่างการประยุกต์ใช้แผนการรับประกันระบบ

4.4 แผนการจัดการองค์ประกอบซอฟต์แวร์

InventSCM_Plan_v1.0

แผนงานการจัดการองค์ประกอบซอฟต์แวร์
ระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์

1. บทนำ

แผนการจัดการองค์ประกอบซอฟต์แวร์นี้ อธิบายถึงโครงสร้างของ SCM และการปฏิบัติเกี่ยวกับการจัดการเกี่ยวกับ Configuraion ต่างๆ (CIs) ในโครงการพัฒนาระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์ รวมถึงระบุหน้าที่ของ SCM ทรัพยากรที่ต้องใช้ดำเนินการ กำหนดขบวนการ เพื่อใช้ในการดำเนินการพัฒนาและบำรุงรักษาซอฟต์แวร์ โดยมีกำหนดระยะเวลาที่เป็นไปตามแผนการดำเนินงานโครงการ(InventSPP_Plan_V1.0)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หน้าที่ความรับผิดชอบ

Software Configuration Manager ได้แก่นางสาวดวงสมร พันธ์พิกุล
มีหน้าที่ควบคุมการใช้ Baseline Library ของโครงการ ควบคุมการเข้าออกของเอกสาร Source Code โปรแกรม การกำหนดสิทธิผู้เข้าใช้ข้อมูลใน ใน Baseline Library รวมทั้งจัดทำรายงานสถานะการขอเปลี่ยนแปลงเอกสาร Source Code โปรแกรม ใน Baseline Library

3. กิจกรรมในเรื่องการจัดการองค์ประกอบซอฟต์แวร์

การจัดการข้อมูล (Data Management)

ต้องควบคุมขั้นตอนในการจัดการข้อมูล และ Configuration ของโครงการ การDistribution การเข้าถึงข้อมูลการจัดเก็บ และการจัดระบบความปลอดภัยของข้อมูล ได้แก่

- การใช้และการเข้าถึงข้อมูล กำหนดให้ SCM มีหน้าที่ในการใช้และการเข้าถึงข้อมูลโดยเป็นไปตามขั้นตอนการปฏิบัติของการจัดการการข้อมูลโครงการที่กำหนด
- ขั้นตอนการจัดทำข้อมูล ให้จัดทำข้อมูลโดยเป็นไปตามขั้นตอนการปฏิบัติของการจัดการการข้อมูลโครงการที่กำหนด
- การรายงานสถานะของข้อมูล ให้ SCM จัดทำรายงานสถานะของข้อมูลเสนอต่อผู้จัดการโครงการทุกวันศุกร์บ่าย
- การรักษาความปลอดภัยของข้อมูล กำหนดสิทธิให้ผู้ที่มีหน้าที่ในโครงการเท่านั้นที่สามารถเข้าใช้ข้อมูลของโครงการได้
- การบำรุงรักษาข้อมูล และการสำรองข้อมูล กำหนดให้ SCM ดำเนินการสำรองข้อมูลทุกวัน นำข้อมูลที่สำรองเก็บไว้ใน library ที่กำหนดไว้

การระบุองค์ประกอบซอฟต์แวร์ (Configuration Identification)

เป็นขบวนการในการกำหนด Configuration ของโครงการ ได้แก่

- การจัดการ Library เพื่อจัดเก็บ Configuration และการจัดทำ Baseline ของโครงการ ให้ SCM จัดทำ Baseline เมื่อเริ่มดำเนินการโครงการตามขั้นตอนการจัดเก็บไฟล์ข้อมูล และแจ้งให้ผู้เกี่ยวข้องในโครงการได้ทราบ
- วิธีการกำหนด CI และข้อมูลที่เกี่ยวข้อง CI ของโครงการเป็นไปตามที่กำหนดไว้และต้องมีการแจ้งให้ผู้เกี่ยวข้องในโครงการได้ทราบ
- ขั้นตอนการแก้ไข CI ของโครงการ การแก้ไข CI นั้นให้ดำเนินการตามขั้นตอนการปฏิบัติการขอเปลี่ยนแปลง

การควบคุมรายการองค์ประกอบซอฟต์แวร์ (Configuration Control)

เป็นการกำหนดขั้นตอนการเปลี่ยนแปลง Baseline ของโครงการ โดยให้กำหนดตามขั้นตอนการปฏิบัติการเปลี่ยนแปลง Baseline ตามเอกสาร SCM_guideline_v1.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การบันทึกและรายงานสถานะการเปลี่ยนแปลงองค์ประกอบซอฟต์แวร์ (Configuration Status Accounting)

เป็นการกำหนดขั้นตอนและผู้มีหน้าที่ในการบันทึกสถานะและการเปลี่ยนแปลงขององค์ประกอบซอฟต์แวร์

- การบันทึกจัดทำสถานะของ Configuration ของโครงการ ให้ SCM บันทึกการเปลี่ยนแปลงของ Configuration รายการต่างๆที่เกิดขึ้นในเอกสารแบบฟอร์มการขอเปลี่ยนแปลง
- การรายงานสถานะของ Configuration ของโครงการ ให้ SCM สรุปผลการเปลี่ยนแปลงของ Configuration รายการต่างๆที่เกิดขึ้นของโครงการ

การตรวจสอบองค์ประกอบซอฟต์แวร์ (Configuration Audit)

ให้ SQA เป็นผู้ดำเนินการตรวจสอบงานดังต่อไปนี้และจัดทำรายงานเสนอต่อผู้จัดการโครงการและผู้อำนวยการโครงการ

- การตรวจสอบ Functional Configuration Audit
- การตรวจสอบ Physical Configuration Audit
- การตรวจสอบผู้ทำหน้าที่ SCM

การดำเนินงานโครงการ

การจัดการองค์ประกอบซอฟต์แวร์มีมาตรฐานที่ใช้ในการดำเนินโครงการ ได้แก่

- การตั้งชื่อเอกสาร
- การตั้งชื่อไฟล์การจัดการเอกสาร
- การจัดการสื่อบันทึกข้อมูล
- การควบคุมเวอร์ชันของชิ้นงานที่เกิดขึ้นตลอดโครงการ
- การจัดการการเปลี่ยนแปลงที่เกิดขึ้น
- การกำหนดตัวบุคคลที่ทำหน้าที่ในการจัดเก็บข้อมูล
- การกำหนดกระบวนการสำรองข้อมูล

ซึ่งได้กำหนดไว้แล้วในเอกสารแนวทางการจัดการองค์ประกอบซอฟต์แวร์ (SCM_guideline_v1.0)

4. แผนการดำเนินงาน

ในการดำเนินกิจกรรม SCM นั้น จะต้องทำอย่างต่อเนื่องตลอดทั้งช่วงชีวิตของการพัฒนาซอฟต์แวร์ โดยแผนการดำเนินการ สามารถกำหนดได้ดังนี้คือ

- กำหนดที่เก็บ (library) ของโครงการ พร้อมทั้งกำหนดสิทธิ์การเข้าถึงตามแนวทางที่กำหนดไว้ (SCM_guideline_v1.0) ก่อนที่จะเริ่มกระบวนการทำงานทั้งหมด
- จัดทำแผนการจัดการองค์ประกอบซอฟต์แวร์ ต้องจัดทำไปพร้อมกับๆ การจัดทำแผนงานโครงการ ก่อนที่จะเริ่มการพัฒนาระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในช่วงของการพัฒนาระบบ กิจกรรม SCM จะต้องดำเนินไปอยู่ตลอดเวลา ในเรื่องของการจัดเก็บ baseline และการขอเปลี่ยนแปลง baseline ต่างๆ
- ให้ SCM จัดทำรายงานสรุปทุกๆ วันศุกร์บ่าย เพื่อให้ PM ทราบสถานะของ baseline แต่ละตัว และความถี่ในการขอเปลี่ยนแปลง

5. การกำหนดรายการองค์ประกอบซอฟต์แวร์

ในแต่ละขั้นตอน จะมี milestone ที่ต้องจัดเก็บไว้เป็น CI ใน library ดังนี้คือ

ขั้นตอนเตรียมแผนการดำเนินงานโครงการ

แผนการดำเนินงานโครงการ (SP_CI_01)

แผนการประกันคุณภาพซอฟต์แวร์ (SP_CI_02)

แผนงานการจัดการองค์ประกอบซอฟต์แวร์ (SP_CI_03)

ขั้นตอนการศึกษาและรวบรวมความต้องการของผู้ใช้

ข้อกำหนดความต้องการ (SP_CI_4)

ขั้นตอนการวิเคราะห์และการออกแบบระบบงาน

รายงานการวิเคราะห์และออกแบบระบบ (SP_CI_08)

UML DIAGRAM (SP_CI_09)

PROGRAM SPECIFICATION (SP_CI_10)

SCREEN LAYOUT (SP_CI_11)

REPORT LAYOUT (SP_CI_12)

DATA STRUCTURE (SP_CI_13)

ขั้นตอนการพัฒนาระบบ

ระบบบริหารคลังวัตถุดิบ (SP_CI_14)

Soucre Code ระบบบริหารคลังวัตถุดิบ (SP_CI_15)

ขั้นตอนการทดสอบระบบ

แผนการทดสอบระบบ (SP_CI_16)

รายงานการทดสอบระบบ (SP_CI_17)

ขั้นตอนการติดตั้งระบบ

โปรแกรมการติดตั้งระบบบริหารคลังวัตถุดิบ (SP_CI_18)

คู่มือการใช้งาน (SP_CI_19)

คู่มือการพัฒนาระบบ (SP_CI_20)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

InventSCM_Plan_v1.0

จัดทำโดย ..... วันที่ .....

(SCM)

รับรองโดย ..... วันที่ .....

(PM)

รูปที่ 4.6 ตัวอย่างการประยุกต์ใช้แผนการจัดการองค์ประกอบซอฟต์แวร์

4.5 เอกสารการยืนยันหน้าจอของระบบ

InventSP_ScreenLayout_v1.0

รูปแบบหน้าจอ

ระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์

1. . หน้าจอการรับวัตถุดิบ

<u>รายละเอียดหน้าจอ</u> : ทำการเพิ่ม ลบ แก้ไข ข้อมูลการรับวัตถุดิบตามใบส่งของของ supplier

<u>ความต้องการเพิ่มเติม/แก้ไข</u> :

.....

.....

.....

ลงชื่อ ..... ผู้จัดทำ

( )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลงชื่อ .....ผู้ตรวจสอบ

( )

รูปที่ 4.7 ตัวอย่างการประยุกต์ใช้เอกสารการยืนยันหน้าจอของระบบ

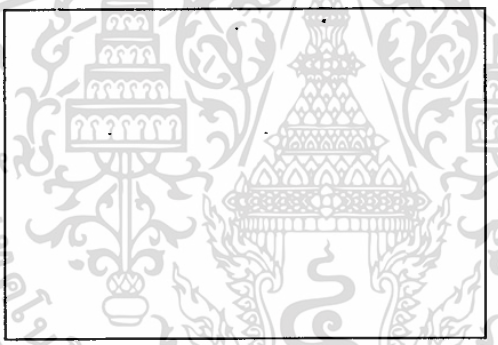
4.6 เอกสารการยืนยันรายงานของระบบ

InventSP_RepLayout_v1.0

รูปแบบรายงาน

ระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์

1. รายงานการรับวัตถุดิบ



<u>รายละเอียดรายงาน</u> : รายงานการรับวัตถุดิบ แบ่งกลุ่มตาม supplier และแสดงผลรวมของ supplier แต่ละราย โดยแสดงผลตามช่วงเวลาตามเงื่อนไขที่กำหนด และเรียงลำดับตามวันที่รับวัตถุดิบ

<u>ความต้องการเพิ่มเติม/แก้ไข</u> :

.....

.....

.....

ลงชื่อ ..... ผู้จัดทำ

( )

ลงชื่อ .....ผู้ตรวจสอบ

( )

รูปที่ 4.8 ตัวอย่างการประยุกต์ใช้ข้อกำหนดความต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7 เอกสาร Program Specification

InventSP_PgSpec_v1.0

หน้าที่ 1/xx

Program Specification

ระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์

ชื่อ Class : UpdPaperScreen	คำอธิบาย : หน้าจอบันทึกข้อมูลวัตถุดิบ (กระดาษ)
Use Case อ้างอิง	1. บันทึกข้อมูลวัตถุดิบ

Method	รายละเอียดการทำงาน
Open()	• Set menu enabled = m_clear, m_save, m_search • Set Material group = '01' (paper group)
Clear()	• Clear all value in screen • Set mode = add
SelectType()	• Generate material_id (group code + type code + running no. 3 digit) • Put material_id value into screen
Save()	• Get values from screen • If mode = add then Insert into table Material Else update table material where material_id = material_id from screen • Do method clear()
Search()	• Open w_search_material and send parameter material group
Retrieve()	• Select values from table Material where material_id = argument material_id • Insert all values in screen • Set Mode = edit
Delete()	• If mode = edit then - Show messagebox to confirm deleting - delete record from table Material where material_id = material_id from screen Else show messagebox to warn that the record has not been saved.
Close()	Close this screen

วันที่ส่งมอบ Spec .....

ลงชื่อ ..... ผู้จัดทำ (SE)

ลงชื่อ ..... ผู้รับมอบ (PG)

รูปที่ 4.9 ตัวอย่างการประยุกต์ใช้ Program Specification

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.8 ตัวอย่างการใช้งานแบบฟอร์มการขอเปลี่ยนแปลงความต้องการ

กรณีมีการเพิ่มเติมที่ไม่กระทบการทำงานอื่นมากนัก

InventSP_ChForm_v1.0 **แบบฟอร์มการขอเปลี่ยนแปลงความต้องการ**	
โครงการ : พัฒนาระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์	หมายเลข : 1
ผู้ขอเปลี่ยนแปลง : ดวงสมร พันธ์พิกุล (SE)	วันที่ : 10 มกราคม 2546
ความต้องการที่ขอ : สามารถตรวจสอบยอดคงเหลือจากหน้าจอการเบิกวัตถุดิบได้	
ขั้นตอนในการพัฒนา : วิเคราะห์และออกแบบ	
ผู้วิเคราะห์ : ดวงสมร	วันที่วิเคราะห์ : 10 มกราคม 2546
การประเมินผลกระทบ 1. เพิ่มปุ่มเชื่อมโยงให้เปิดหน้าจอตรวจสอบวัตถุดิบคงเหลือ ทำให้ต้องแก้ไข Screen Layout (SP_CI_11) 2. ในด้านการทำงานของระบบไม่กระทบกับการทำงานอื่น	
ความเร่งด่วน : ไม่เร่งด่วน	จำนวนแรงงานที่ใช้ : 2 ชั่วโมง
ความเห็นของ PM : เห็นควรให้เพิ่มเติม	
ผลการดำเนินการ : เพิ่มเติมเรียบร้อยแล้ว	
วันที่ดำเนินการ : 12 มกราคม 2546	
หมายเหตุ :	

รูปที่ 4.10 ตัวอย่างการประยุกต์ใช้แบบฟอร์มการขอเปลี่ยนแปลงความต้องการ (กรณีมีการเปลี่ยนแปลงมีผลกระทบไม่มาก)

กรณีมีความเพิ่มเติมที่ส่งผลกระทบหลายอย่าง

InventSP_ChForm_v1.0 **แบบฟอร์มการขอเปลี่ยนแปลงความต้องการ**	
โครงการ : พัฒนาระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์	หมายเลข : 2
ผู้ขอเปลี่ยนแปลง : ดวงสมร พันธ์พิกุล (SE)	วันที่ : 20 มกราคม 2546
ความต้องการที่ขอ : สามารถเรียกดูราคาวัตถุดิบครั้งล่าสุดได้	
ขั้นตอนในการพัฒนา : Testing	
ผู้วิเคราะห์ : ดวงสมร	วันที่วิเคราะห์ : 20 มกราคม 2546
การประเมินผลกระทบ	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ต้องปรับแก้โครงสร้างข้อมูลและ Screen Layout 6 หน้าจอ 2: CI ที่ต้องปรับแก้ได้แก่ ข้อกำหนดความต้องการ (SP_CI_04) รายงานการวิเคราะห์และออกแบบระบบ (SP_CI_08) UML DIAGRAM (SP_CI_09) PROGRAM SPECIFICATION (SP_CI_10) SCREEN LAYOUT (SP_CI_11) DATA STRUCTURE (SP_CI_13) ระบบบริหารคลังวัตถุดิบ (SP_CI_14) Soucre Code ระบบบริหารคลังวัตถุดิบ (SP_CI_15)
ความเร่งด่วน : ด่วน | จำนวนแรงงานที่ใช้ : 3 วัน
ความเห็นของ PM : เห็นควรให้เพิ่มเติม
ผลการดำเนินการ : 1. ปรับแก้โครงสร้างข้อมูลและ Screen Layout 2. ปรับแก้ Source Code ในหน้าจอที่มีการเพิ่มเติม 3. ปรับแก้เอกสาร CI ที่เกี่ยวข้อง
วันที่ดำเนินการ : 21 มกราคม 2546 – 23 มกราคม 2546
หมายเหตุ : ควรหารือเรื่องความต้องการที่ยังเหลือกับผู้ใช้อีกครั้ง ก่อนที่จะนำระบบไปให้ทดสอบ

รูปที่ 4.11 ตัวอย่างการประยุกต์ใช้แบบฟอร์มการขอเปลี่ยนแปลงความต้องการ (กรณีการเปลี่ยนแปลงมีผลกระทบในหลายส่วน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.9 ตัวอย่างการใช้งานแบบฟอร์มการขอเปลี่ยนแปลง baseline

InventSP_ChBLForm_v1.0 แบบฟอร์มการขอเปลี่ยนแปลง baseline	
โครงการ : พัฒนาระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์	หมายเลข : 1
ผู้ขอเปลี่ยนแปลง : ดวงสมร	วันที่ : 10 มกราคม 2546
Baseline ที่ขอเปลี่ยนแปลง : Screen Layout (SP_PI_11)	
เหตุผลในการขอเปลี่ยนแปลง : มีความต้องการเพิ่มเติม อ้างอิงจาก : InventSP_chForm_v1.0 หมายเลข 1	
ความเห็นของ SCM : อนุมัติให้เปลี่ยนแปลง	
ผลการดำเนินการ : ปรับแก้หน้าจอที่เปลี่ยนแปลง 1 หน้าจอ และเพิ่มเวอร์ชันของ Screen Layout เป็น 2.0	
วันที่ดำเนินการ :	
หมายเหตุ :	

InventSP_ChBLForm_v1.0 แบบฟอร์มการขอเปลี่ยนแปลง baseline	
โครงการ : พัฒนาระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์	หมายเลข : 2
ผู้ขอเปลี่ยนแปลง : ดวงสมร	วันที่ : 21 มกราคม 2546
Baseline ที่ขอเปลี่ยนแปลง : DATA STRUCTURE (SP_CI_13)	
เหตุผลในการขอเปลี่ยนแปลง : มีความต้องการเพิ่มเติม อ้างอิงจาก : InventSP_chForm_v1.0 หมายเลข 2	
ความเห็นของ SCM : อนุมัติให้เปลี่ยนแปลง	
ผลการดำเนินการ : เพิ่ม field (Price [currency], Price_unit[text 2] , Price_Date[date time]) ใน Table Material และเพิ่มเวอร์ชันของ Data Structure เป็น 2.0	
วันที่ดำเนินการ : 23 มกราคม 2546	
หมายเหตุ :	

รูปที่ 4.12 ตัวอย่างการประยุกต์ใช้แบบฟอร์มการขอเปลี่ยนแปลง baseline

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.10 ตัวอย่างการใช้งานแบบฟอร์มการขอเปลี่ยนแปลง software

<table>
<tr><td colspan="2">SCM_ChSWForm_v1.0
แบบฟอร์มการขอเปลี่ยนแปลงซอฟต์แวร์</td></tr>
<tr><td>โครงการ : พัฒนาระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์</td><td>หมายเลข : 1</td></tr>
<tr><td>ผู้ขอเปลี่ยนแปลง : ดวงสมร</td><td>วันที่ : 21 มกราคม 2546</td></tr>
<tr><td colspan="2">Baseline ที่ต้องการแก้ไข : Soucre Code ระบบบริหารคลังวัตถุดิบ (SP_CI_15)</td></tr>
<tr><td colspan="2">Module ที่ต้องการแก้ไข :
• function wf_save(), wf_retrieve(), wf_clrscr(), wf_edit() ของหน้าจอ w_upd_paper, w_upd_ink, w_upd_card, w_upd_env, w_upd_misc
• function wf_retrieve() ,wf_clearscr()ของหน้าจอ w_search_balance</td></tr>
<tr><td colspan="2">เหตุผลที่ขอแก้ไข : มีความต้องการเพิ่มเติม
อ้างอิง : InventSP_chForm_v1.0 หมายเลข 2</td></tr>
<tr><td>ผู้วิเคราะห์ : ดวงสมร</td><td>วันที่วิเคราะห์ : 21 มกราคม 2546</td></tr>
<tr><td>ความเร่งด่วน : ด่วน</td><td>จำนวนแรงงานที่ใช้ : 1.5 วัน</td></tr>
<tr><td colspan="2">ผลการดำเนินการ : ดำเนินการแก้ไขเรียบร้อยแล้ว</td></tr>
<tr><td colspan="2">วันที่ดำเนินการ : 22 มกราคม 2546</td></tr>
<tr><td colspan="2">หมายเหตุ :</td></tr>
</table>

รูปที่ 4.13 ตัวอย่างการประยุกต์ใช้แบบฟอร์มการขอเปลี่ยนแปลง software

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.11 ตัวอย่างรายงานสรุปสถานะความต้องการแต่ละตัว

รายงานสรุปสถานะความต้องการแต่ละตัว						
ระบบบริหารคลังวัตถุดิบ โรงพิมพ์สันติภาพ พริ้นท์					วันที่ 20/02/46	
ความต้องการ	Defined	Approved	Committed	Designed	Implemented	Complete
1. หน้าจอบันทึกใบส่งของ	X	X	X	X	X	X
2. หน้าจอบันทึกใบเบิก	X	X	X	X	X	-
3. หน้าจอเรียกดูยอดคงเหลือ	X	X	X	-	-	-
4. หน้าจอการเลือกเงื่อนไขรายงานจุดสั่งซื้อ	X	X	X	-	-	-
5.	X	-	-	-	-	-
หมายเหตุ :						
ลงชื่อ ..... ผู้รายงานผล (น.ส.ดวงสมร พันธ์พิกุล)						

รูปที่ 4.14 ตัวอย่างรายงานสรุปสถานะความต้องการแต่ละตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการดำเนินงานและข้อเสนอแนะ

CMM ระดับที่ 2 จะเน้นในเรื่องของการบริหารจัดการโครงการเป็นหลัก เพราะการบริหารจัดการที่ไม่มีประสิทธิภาพเป็นต้นเหตุให้โครงการไม่ประสบความสำเร็จ ในโครงการขนาดเล็กส่วนใหญ่ ไม่ได้ให้ความสำคัญในเรื่องการบริหารจัดการ จะเน้นไปในการพัฒนาโปรแกรมเป็นหลัก ซึ่งส่งผลให้ผลผลิตที่ได้รับอาจจะไม่ตรงตามความต้องการของผู้ใช้ หรืออาจจะส่งระบบได้ไม่ทันตามเวลาที่กำหนด จากการนำ CMM ระดับที่ 2 มาประยุกต์ใช้กับโครงการขนาดเล็ก มีข้อดีและข้อเสียดังนี้คือ

ข้อดีของการนำ CMM ระดับที่ 2 มาประยุกต์ใช้กับโครงการขนาดเล็ก

1. ผู้จัดการโครงการสามารถติดตามงานตามแผนการที่กำหนดไว้ได้ รู้สถานะโดยรวมว่าระบบพัฒนาไปถึงขั้นตอนใด และสามารถเห็นปัญหาที่อาจจะเกิดขึ้นและหาทางแก้ไขได้อย่างรวดเร็วและมีประสิทธิภาพ
2. มีการยืนยันความต้องการร่วมกับผู้ใช้อย่างเป็นระบบ ทำให้ผลิตภัณฑ์ในขั้นตอนสุดท้ายตรงตามความต้องการของผู้ใช้ และลดข้อขัดแย้งในการสื่อสารกับผู้ใช้ได้เป็นอย่างดี
3. กรณีมีการพัฒนาระบบร่วมกัน จะมีแนวทางในการจัดการองค์ประกอบที่ทำให้การทำงานไม่ซ้ำซ้อนและตรวจสอบเวอร์ชันของเอกสารและระบบได้ตลอดเวลา
4. ทำให้การทำงานโดยรวมเป็นไปอย่างมีระบบ มีมาตรฐาน สามารถตรวจสอบได้ และเมื่อองค์กรมีขนาดใหญ่ขึ้นจะทำให้การทำงานง่ายขึ้นและทุกคนทำงานไปในทิศทางเดียวกัน

ข้อเสียของการนำ CMM ระดับที่ 2 มาประยุกต์ใช้กับโครงการขนาดเล็ก

1. ในการเริ่มต้นทำงานโครงการโดยใช้ CMM เป็นครั้งแรกนั้น อาจจะต้องใช้เวลา และแรงงานในการศึกษาและกำหนดนโยบายรวมถึงขั้นตอนกระบวนการในการทำงานค่อนข้างมาก เพื่อกำหนดเป็นมาตรฐานและแนวทางให้กับโครงการอื่นต่อไป
2. การตรวจทานกิจกรรมและทำรายงานสรุปผลเป็นระยะนั้น อาจจะเป็นการเพิ่มภาระงานในกรณีที่โครงการนั้นมีทีมงานจำนวนน้อยมากเพียง 1 หรือ 2 คน
3. ในกรณีไม่มีระบบมาช่วย อาจจะต้องจัดเก็บเอกสารที่เป็นกระดาษเพิ่มขึ้นจำนวนมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการวิจัยและนำ CMM ระดับที่ 2 มาประยุกต์ใช้กับกรณีศึกษา สามารถสรุปผลได้ว่า CMM ระดับที่ 2 จะเห็นผลได้ชัดเจนกว่าถ้านำไปประยุกต์ใช้ในโครงการที่มีคนมากกว่า 2 คนขึ้นไป แต่ทั้งนี้ทั้งนั้นจากการวิจัยจะแสดงให้เห็นว่าถึงแม้เราจะไม่สามารถบรรลุ KPAของ CMM ระดับที่ 2 ได้ครบถ้วนในทุก common feature แต่เราก็สามารถเห็นแนวทางว่าในระบบต่อไปที่คล้ายคลึงกัน สามารถนำนโยบายและกระบวนการที่กำหนดไว้แล้วไปใช้ได้ ซึ่งตรงกับจุดประสงค์ของ CMM ระดับที่สองคือ เป็นระดับที่ทำซ้ำได้ และไม่เสียเวลามาก ในช่วงแรกอาจจะมีความลำบากในการพัฒนา แต่จะให้ผลลัพธ์ที่คุ้มค่า ปัญหาที่พบในการพัฒนาซอฟต์แวร์จะลดลงและได้ผลิตภัณฑ์ที่มีคุณภาพ ตรงตามความต้องการของผู้ใช้

<u>ปัญหาอุปสรรคและข้อเสนอแนะ</u>

จากการดำเนินการวิจัยการพัฒนาระบบบริหารคลังวัตถุดิบโดยนำ CMM ระดับที่ 2 มาประยุกต์ใช้นั้น พบปัญหาและอุปสรรค ดังนี้คือ

1. ผู้วิจัยไม่มีประสบการณ์ทางด้านการบริหารจัดการโครงการ ทำให้ต้องใช้เวลาในการศึกษาเอกสารในเรื่องนี้ เพื่อใช้ประกอบในการศึกษาวิจัย
2. การคัดเลือกและรวบรวมเอกสารที่จำเป็นจะต้องใช้จริงๆ กับโครงการขนาดเล็กค่อนข้างจะเลือกยาก เนื่องจากเอกสาร CMM ระดับที่ 2 จะต้องยุ่งเกี่ยวกับเอกสารจำนวนมาก
3. เวลาที่จำกัดในการวิจัย ทำให้ไม่สามารถนำ CMM ระดับที่ 2 ไปประยุกต์ใช้ได้จนถึงขั้นตอนของการบำรุงรักษาระบบ

จากปัญหาและอุปสรรคที่ได้ ทำให้สามารถสรุปเป็นข้อเสนอแนะได้ดังนี้

1. การนำ CMM ระดับที่ 2 มาใช้งานนั้น ควรจะมีระบบมาช่วย เพื่อลดการจัดเก็บเอกสารที่เป็นกระดาษให้ลดลง เช่นเอกสารแบบฟอร์มต่างๆ หรือเอกสารที่ต้องมีการตรวจทานจากผู้เกี่ยวข้อง
2. การวิจัยเป็นการทำวิจัยด้วยตนเอง และนำมาประยุกต์ใช้เอง ไม่ได้นำไปให้ผู้อื่นทดลองใช้ ซึ่งอาจจะเป็นมุมมองเป็นการมองจากคนเพียงคนเดียว ด้านเดียว ถ้ามีการนำไปประยุกต์ใช้ในระบบอื่นๆ ที่ผู้วิจัยไม่ได้พัฒนาเอง อาจจะมีข้อเสนอแนะใหม่ๆ และเป็นแนวทางในการประยุกต์ใช้ CMM ระดับที่ 2 กับโครงการขนาดเล็กได้มีประสิทธิภาพยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ครรชิต มาลัยวงศ์ 2543, "การบริหารโครงการซอฟต์แวร์และ CMM" เอกสารประกอบคำบรรยาย ศูนย์บริหารสารสนเทศทางเทคโนโลยี สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ.

ครรชิต มาลัยวงศ์ 2543. พจนานุกรม CMM [Online]. Available : Http://www.drkanchit.com.

จิตรา วัฒนรัตน์ 2540. "การออกแบบการพัฒนาองค์กรซอฟต์แวร์เข้าสู่ระบบซีเอ็มเอ็ม" วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ บัณฑิตวิทยาลัย. จุฬาลงกรณ์มหาวิทยาลัย

Carnegie Melon University 1994. **The Capability Maturity Model : Guidelines for Improving the Software Process** : Software Engineering Institute , Carnegie Melon University. : Addison- Wesley Longman Inc.

Henrietta Foster 2001. Applying the CMM in Small Companies, Organizations, and Projects. [Online]. Available : Http://www.ewh.ieee.org/r1/rochester/comsoc/2001/hfoster_slide/

Kim Caputo. 2001. **CMM Implementation guide : choreographing software process improvement.** 6th. Boston : Addison-Wesley.

Mark C. Paulk. 1998. "Using the Software CMM in Small Organizations". Carnegie Melon University

Mark C. Paulk. 1999 "Using the Software CMM with Good Judgement" ASQ Software Quality Professional, vol1, No.3, June 1999 pp19-29

Roger S. Pressman. 1997. **Software Engineering A Practitioner's Approach.** 5th ed. McGraw-Hill International Editions.

Terttu Orci and Astrid Laryd 2000. CMM for Small Organisations Level 2 UMINF- 00.20. Department of Computing Science. Umea University .SWEDEN

Terttu Orci, Astrid Laryd, Jurgen borstler 2000. CMM for Small Organisations, Level 2 The supplements.UMINF 01.19 ISSN-0348-0542

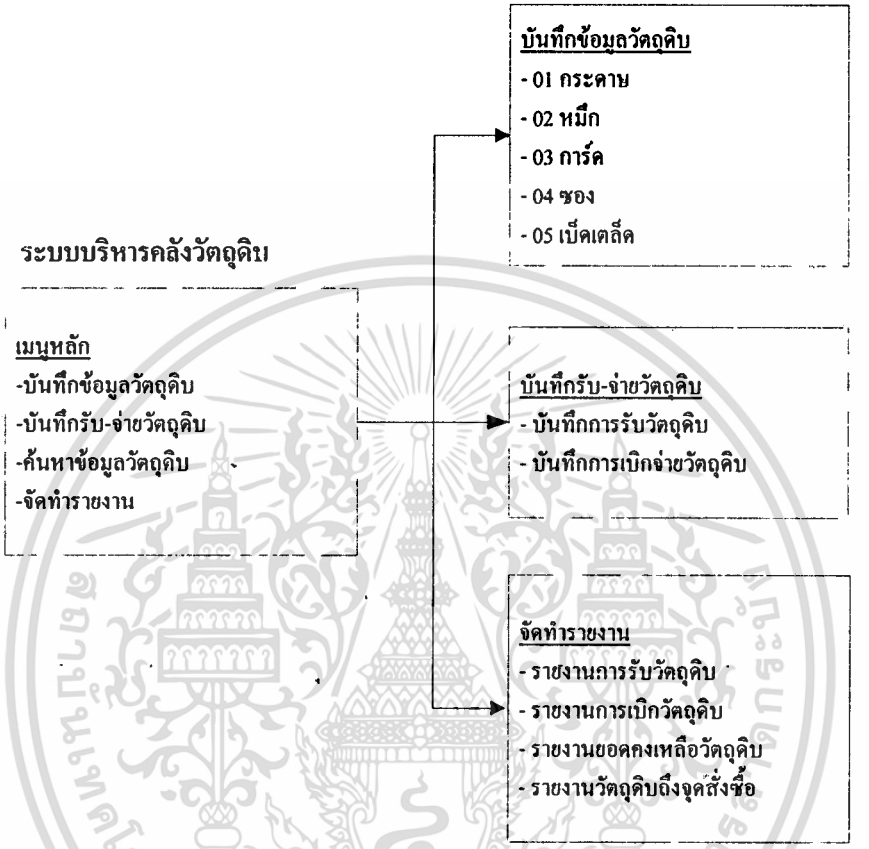
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
ตัวอย่างหน้าจอและโครงสร้างข้อมูลระบบบริหารคลังวัตถุดิบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างระบบ

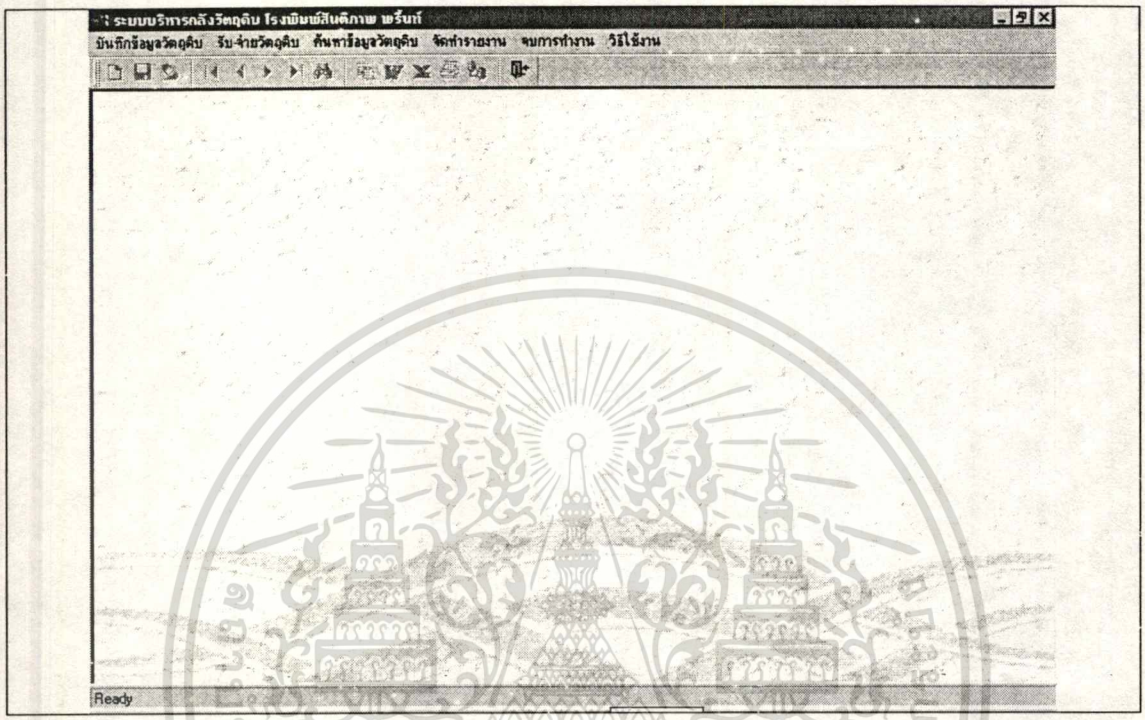
ระบบบริหารคลังวัตถุดิบ



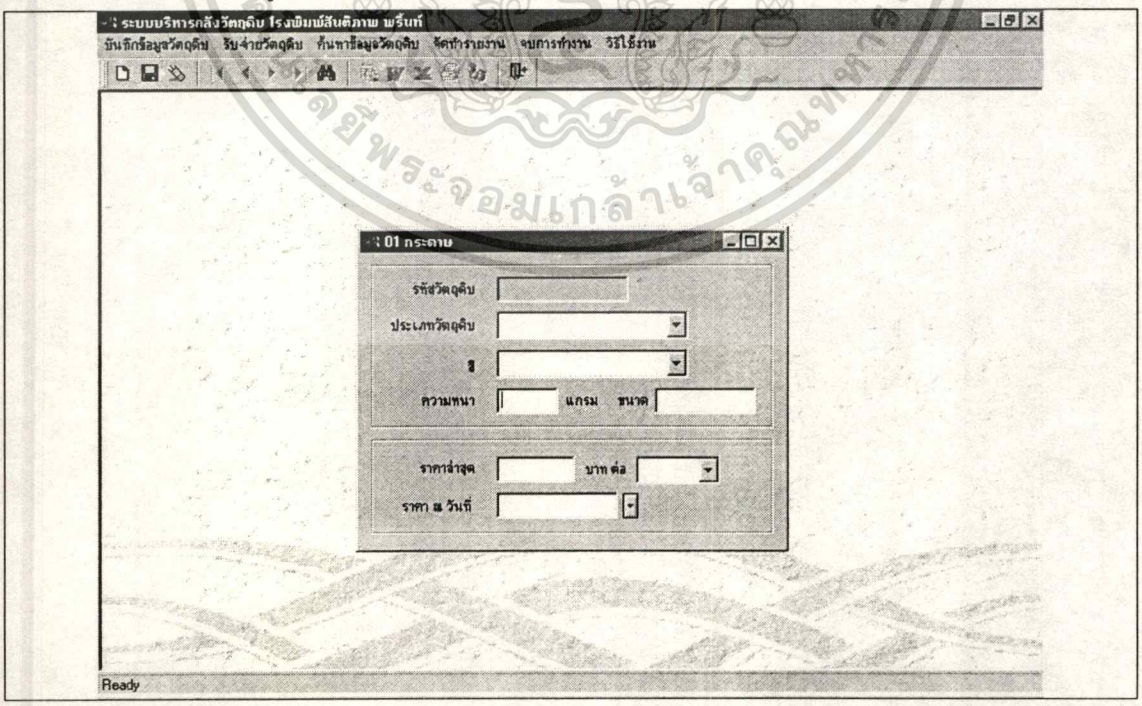
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบหน้าจอ ระบบบริหารคลังวัตถุดิบ

1. หน้าจอหลัก

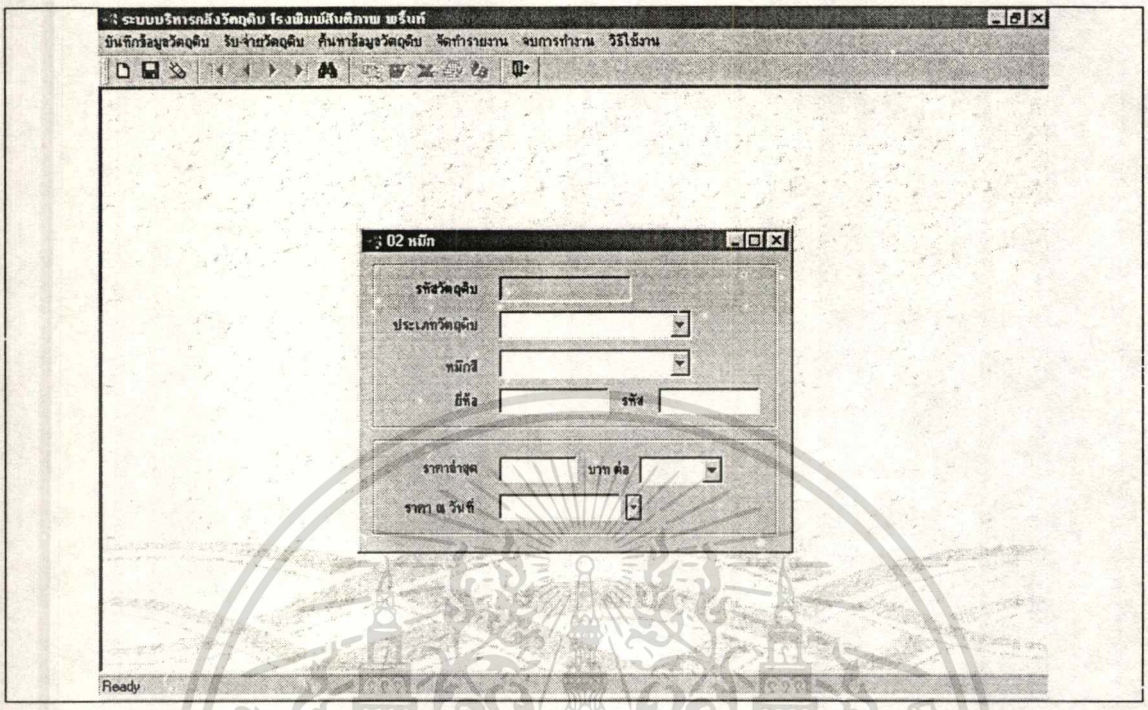


2. หน้าจอบันทึกข้อมูลกระดาษ

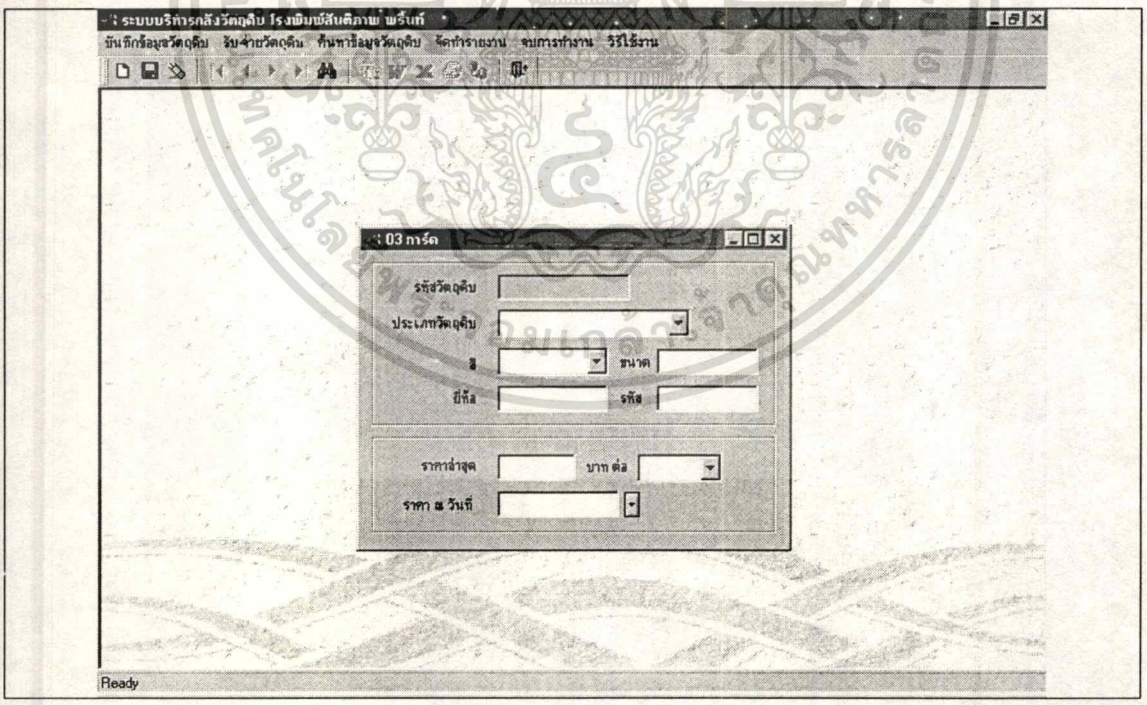


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. หน้าจอบันทึกข้อมูลหมึก

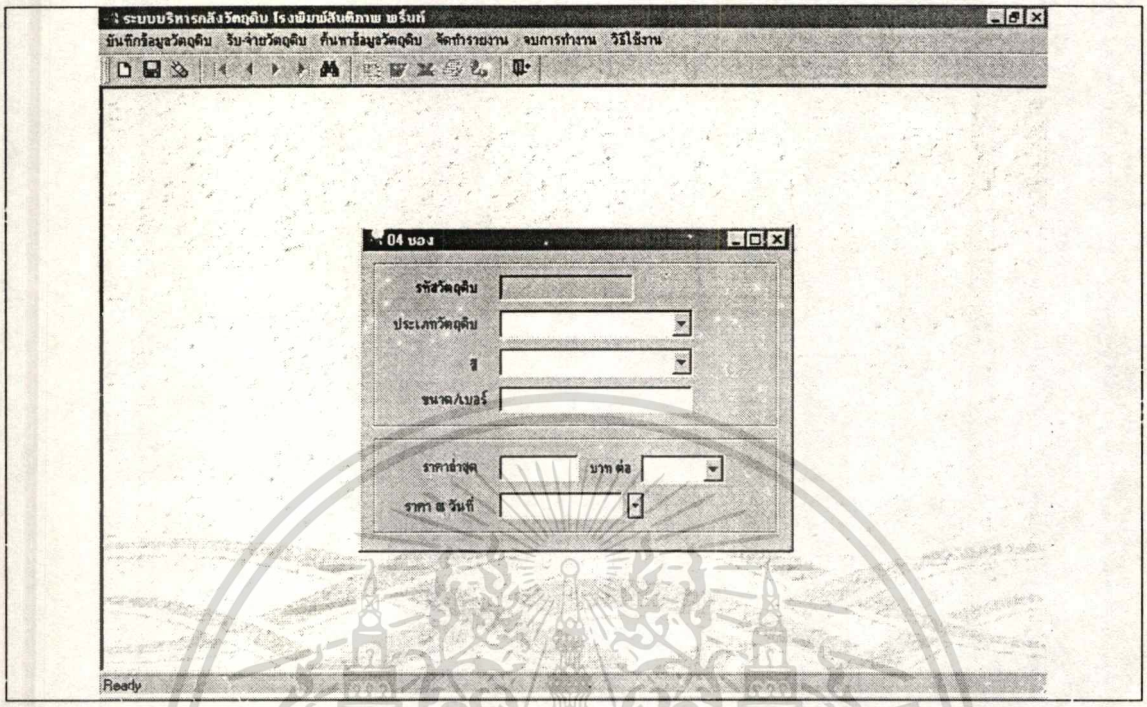


4. หน้าจอบันทึกข้อมูลการ์ด

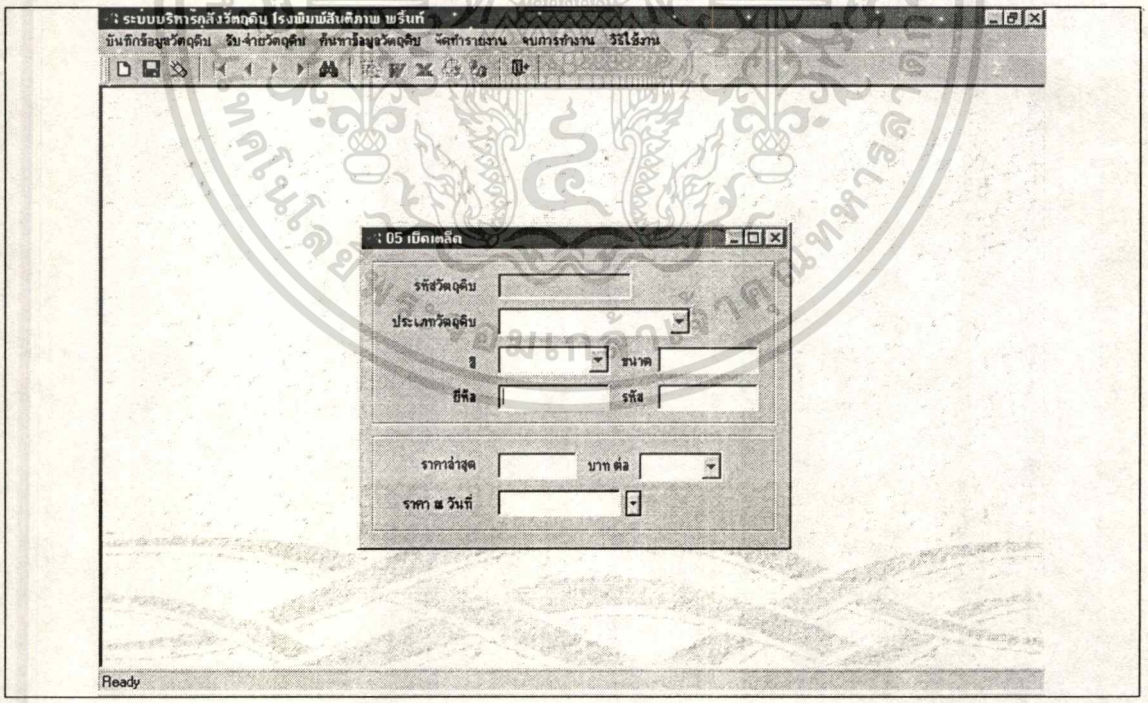


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. หน้าจอบันทึกข้อมูลซอง

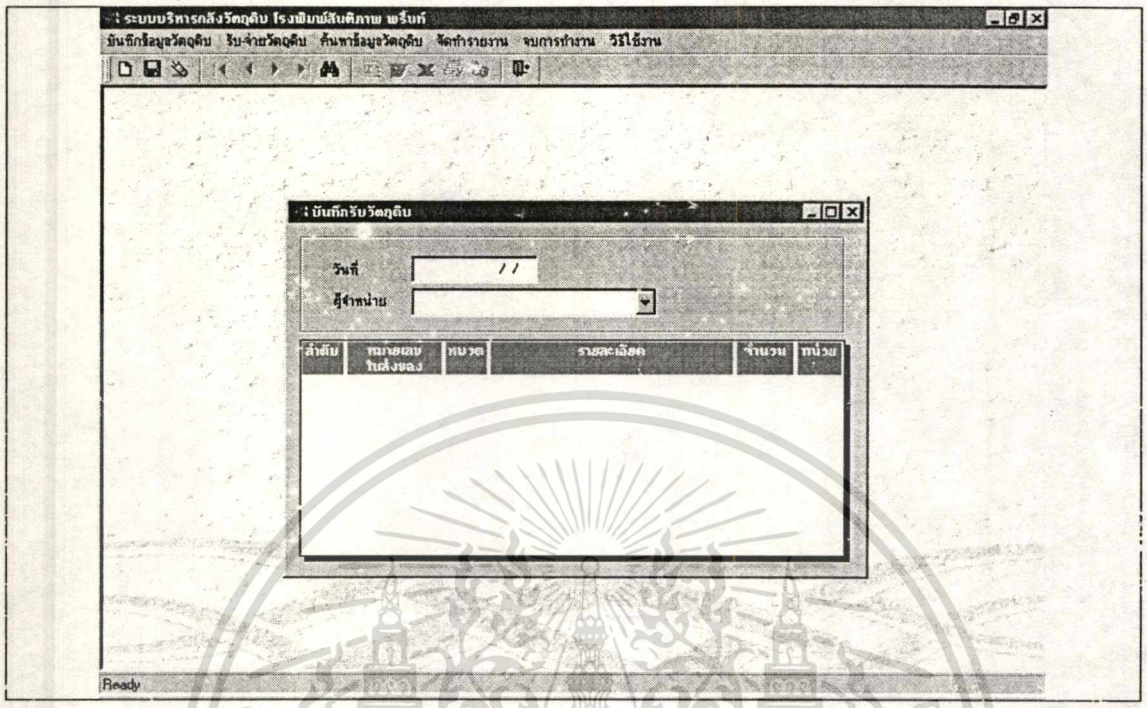


6. หน้าจอบันทึกข้อมูลเบ็ดเตล็ด

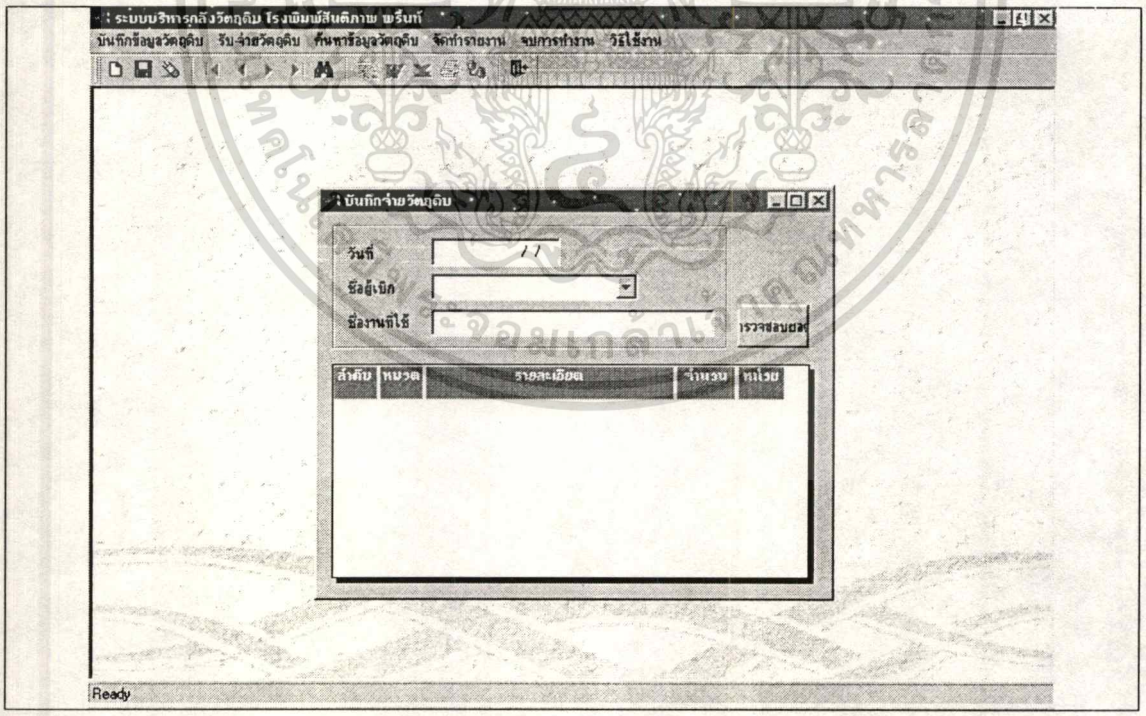


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. หน้าจอบันทึกข้อมูลการรับวัตถุดิบ

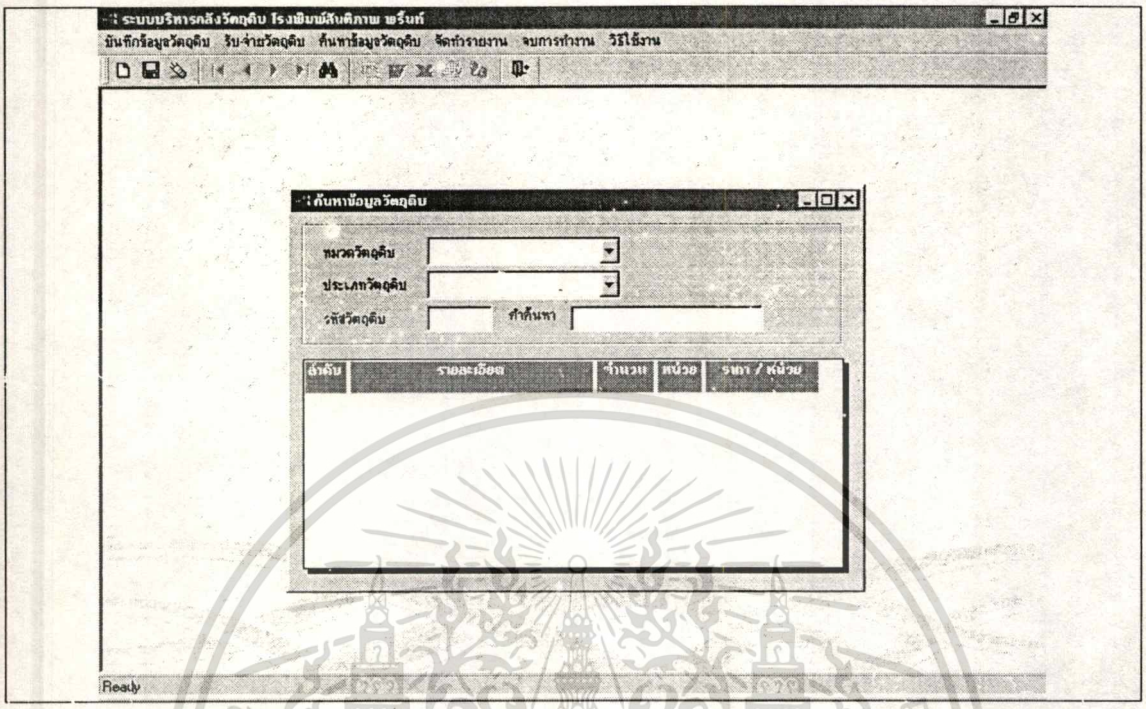


8. หน้าจอบันทึกข้อมูลการเบิกวัตถุดิบ

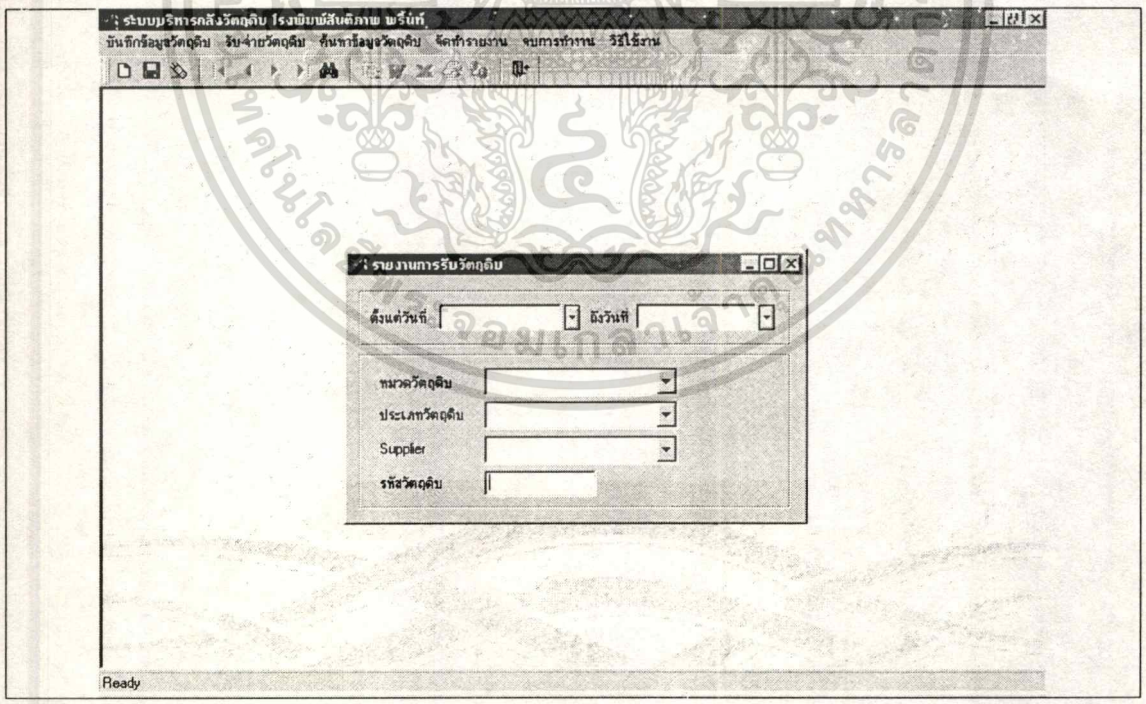


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. หน้าจอค้นหาข้อมูลวัตถุดิบ

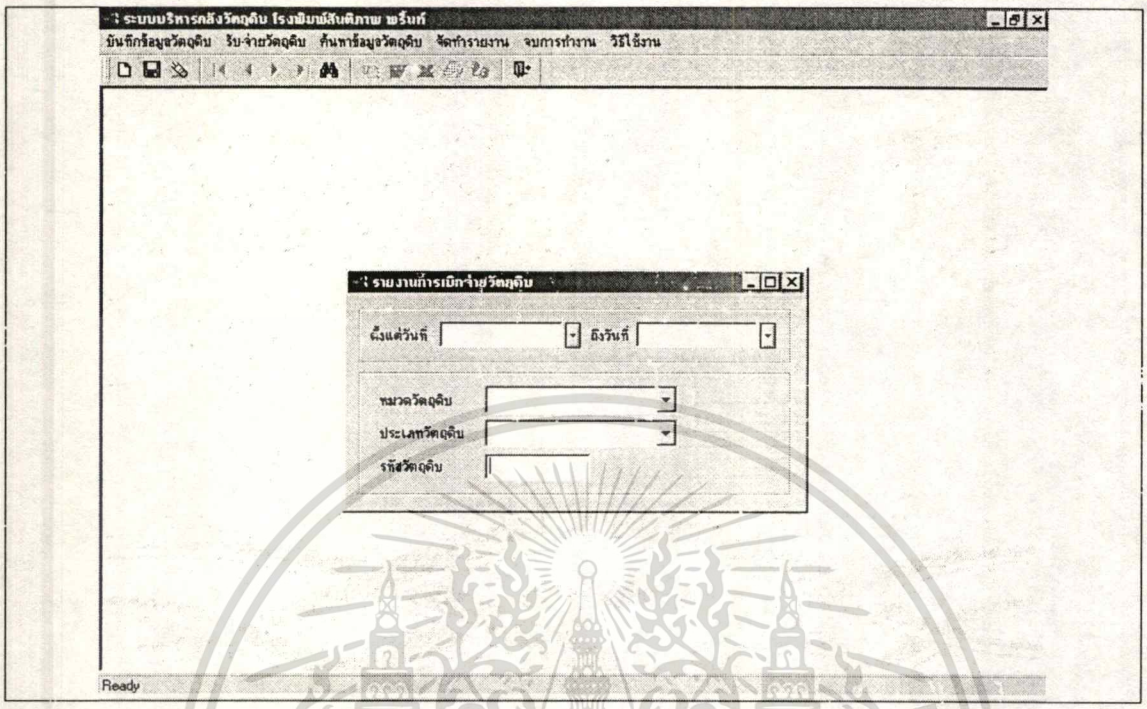


10. หน้าจอจัดทำรายงานการรับวัตถุดิบ

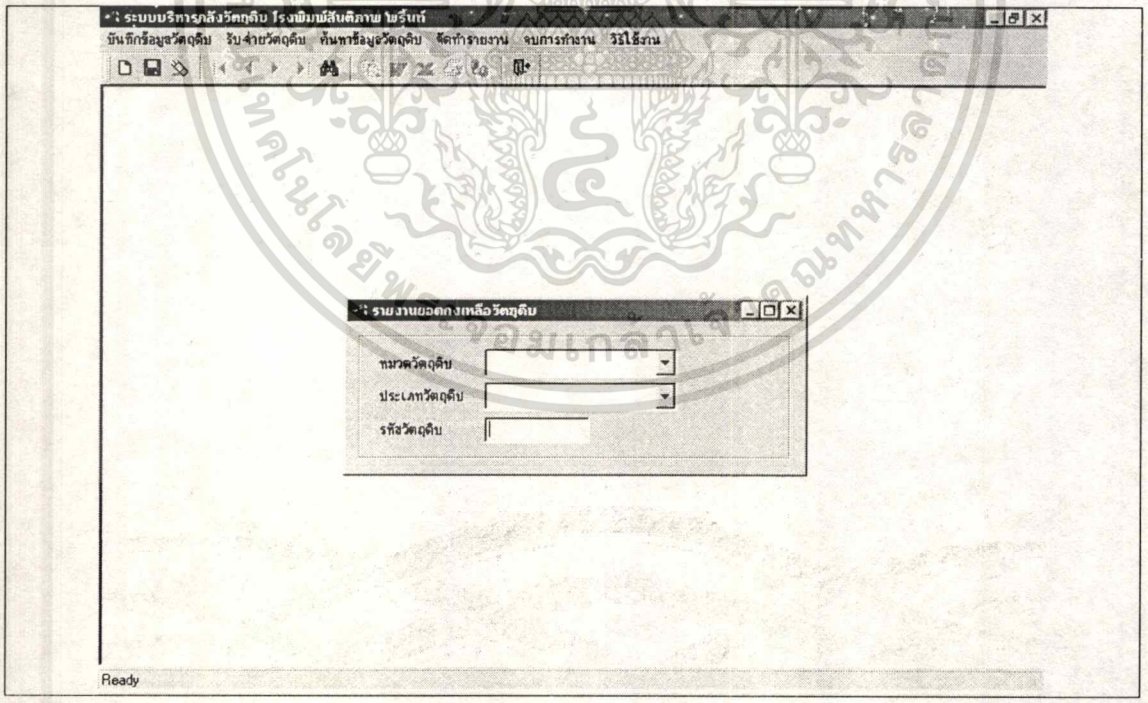


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. หน้าจอจัดทำรายงานการจ่ายวัตถุดิบ

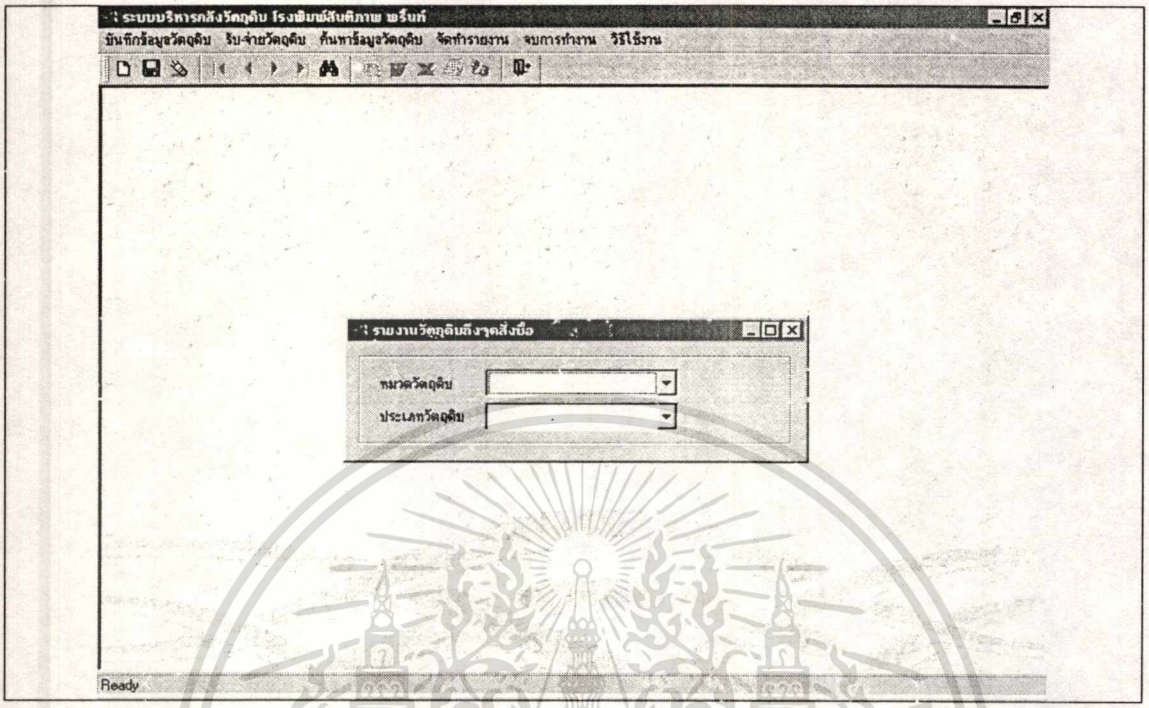


12. หน้าจอจัดทำรายงานยอดคงเหลือวัตถุดิบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. หน้าจอจัดทำรายงานวัตถุดิบถึงจุดสั่งซื้อ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างข้อมูล

Table: Balance

ลำดับที่	รายการข้อมูล	คำอธิบาย	ชนิด	KEY
1.	ma_Id	รหัสวัตถุดิบ	text(50)	PK
2.	bal_Total	จำนวนวัตถุดิบคงเหลือ	number(Long)	
3.	unit_Id	รหัสหน่วย	text(50)	

Table: Invoice

ลำดับที่	รายการข้อมูล	คำอธิบาย	ชนิด	KEY
1.	inv_Date	วันที่รับใบส่งของ	Date/Time	PK
2.	supp_Id	รหัส supplier	Text(2)	
3.	inv_No	ลำดับ	Number(Long)	
4.	inv_Code	หมายเลขใบส่งของ	Text(50)	
5.	ma_Id	รหัสวัตถุดิบ	Text(50)	
6.	inv_Total	จำนวนวัตถุดิบที่ส่ง	Number(Long)	
7.	unit_Id	หน่วย	Text(2)	

Table: MColor

ลำดับที่	รายการข้อมูล	คำอธิบาย	ชนิด	KEY
1.	grp_Id	รหัสหมวดวัตถุดิบ	Text(2)	PK
2.	col_Id	รหัสสี	Text(6)	
3.	col_Name	ชื่อสี	Text(50)	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table: MGroup

ลำดับที่	รายการข้อมูล	คำอธิบาย	ชนิด	KEY
1.	grp_Id	รหัสหมวดวัตถุดิบ	Text(2)	PK
2.	grp_Name	ชื่อหมวดวัตถุดิบ	Text(50)	

Table: Material

ลำดับที่	รายการข้อมูล	คำอธิบาย	ชนิด	KEY
1.	ma_Id	รหัสวัตถุดิบ	text(6)	PK
2.	ma_Name	รายละเอียดวัตถุดิบ	Text(50)	
3.	ma_Color	รหัสสี	text(50)	
4.	ma_Size1	ขนาดวัตถุดิบ (1)	Text	
5.	ma_Size2	ขนาดวัตถุดิบ (2)	Text	
6.	ma_Brand	ยี่ห้อวัตถุดิบ	Text	
7.	ma_BrandCode	รหัสตามยี่ห้อ	Text	
8.	ma_Price	ราคาล่าสุดวัตถุดิบต่อหน่วย	Currency	
9.	ma_PriceUnit	หน่วยของราคา	Text	
10.	ma_PriceDate	วันที่ของราคาล่าสุด	Date/Time	
11.	ma_buyPoint	จุดสั่งซื้อ	Number(Long)	
12.	ma_buyPointUnit	หน่วยของจุดสั่งซื้อ	Text	
13.	ma_Remark	หมายเหตุ	Text(255)	

Table: MType

ลำดับที่	รายการข้อมูล	คำอธิบาย	ชนิด	KEY
1.	grp_Id	รหัสหมวดวัตถุดิบ	Text(2)	PK
2.	type_Id	รหัสประเภทวัตถุดิบ	Text(2)	PK
3.	type_Name	ชื่อประเภทวัตุดิบ	Text(50)	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table: Request

ลำดับที่	รายการข้อมูล	คำอธิบาย	ชนิด	KEY
1.	req_Date	วันที่เบิก	Date/Time	PK
2.	emp_Code	รหัสพนักงานที่เบิก	Text(6)	
3.	req_No	ลำดับที่	Number(Long)	
4.	req_work	ชื่องานที่ต้องใช้วัตถุดิบ	Text(255)	
5.	ma_Id	รหัสวัตถุดิบ	Text(6)	
6.	req_Total	จำนวนเบิก	Currency	
7.	req_Unit	หน่วยที่เบิก	Text(2)	

Table: Unit

ลำดับที่	รายการข้อมูล	คำอธิบาย	ชนิด	KEY
1.	grp_Id	รหัสหมวดวัตถุดิบ	Text(2)	PK
2.	unit_Id	รหัสหน่วย	Text(2)	
3.	unit_Name	ชื่อหน่วย	Text(50)	
4.	unit_Multiply	จำนวนคูณเพื่อให้ได้หน่วยย่อยที่สุด	Number(Long)	
5.	unit_status	สถานะบอกว่าเป็นหน่วยย่อยหรือไม่ 1- เป็น 0- ไม่เป็น	Text(1)	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบรายงาน ระบบบริหารคลังวัตถุดิบ

1. รายงานการรับวัตถุดิบ

รายงานการรับวัตถุดิบ

ระหว่างวันที่ xx/xx/xxxx ถึงวันที่ xx/xx/xxxx

วันที่พิมพ์รายงาน xx/xx/xxxx

หน้าที่ xx/xx

ร้านค้า xxxxxxxxxxxxxxxxxxxx

ลำดับที่	หมายเลขใบส่งของ	รายละเอียด	จำนวน	หน่วย	ราคา (บาท)
				รวม	

2. รายงานการเบิกวัตถุดิบ

รายงานการเบิกวัตถุดิบ

ระหว่างวันที่ xx/xx/xxxx ถึงวันที่ xx/xx/xxxx

วันที่พิมพ์รายงาน xx/xx/xxxx

หน้าที่ xx/xx

หมวดวัตถุดิบ xxxxxxxxxxxxxxxxxxxx

ลำดับที่	รายละเอียด	จำนวน	หน่วย	ผู้เบิก	งานที่ใช้
ประเภทวัตถุดิบ xxxxxxxxx					
ประเภทวัตถุดิบ xxxxxxxxx					

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. รายงานยอดคงเหลือวัตถุดิบ

รายงานยอดคงเหลือวัตถุดิบ

วันที่ xx/xx/xxxx

วันที่พิมพ์รายงาน xx/xx/xxxx

หน้าที่ xx/xx

หมวดวัตถุดิบ xxxxxxxxxxxxxxxxxxxx

ลำดับที่	รายละเอียด	จำนวน	หน่วย
ประเภทวัตถุดิบ xxxxxxxxxxxxx			
ประเภทวัตถุดิบ xxxxxxxxxxxxx			

4. รายงานวัตถุดิบถึงจุดสั่งซื้อ

รายงานวัตถุดิบที่ถึงจุดสั่งซื้อ

วันที่ xx/xx/xxxx

วันที่พิมพ์รายงาน xx/xx/xxxx

หน้าที่ xx/xx

หมวดวัตถุดิบ xxxxxxxxxxxxxxxxxxxxx

ลำดับที่	รายละเอียด	จำนวน	หน่วย	ราคา/หน่วย
ประเภทวัตถุดิบ xxxxxxxxxxxxxxxxxx				
ประเภทวัตถุดิบ xxxxxxxxxxxxxxxxxx				

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
รายการตรวจสอบการเข้าสู่ CMM ระดับที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CMM for Small Organisations, Level 2

The Supplements

Terttu Orci, Astrid Laryd, and Jürgen Börstler

UMINF 01.19
ISSN-0348-0542

UMEÅ UNIVERSITY
Department of Computing Science
SE-901 87 UMEÅ
SWEDEN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Abbreviations

AB	ABility to perform (feature of key practice)
AC	ACtivity (feature of key practice)
C	Commitment (feature of key practice)
CMM	Capability Maturity Model
CSQA	Customer Software Quality Assurance (role)
DSG	Documentation Support Group (role)
G	Goal (feature of key practice)
KPA	Key Process Area
M	Measurement (feature of key practice)
MS	Marketing and Sales (role)
PM	Project Manager (role)
RM	Requirements Management (KPA)
S	Small enterprise
SCCB	Software Configuration Control Board (role)
SCM	Software Configuration Management (KPA)
SE	Software Engineer (role)
SEPG	Software Engineering Process Group
SM	Senior Manager (role)
SPP	Software Project Planning (KPA)
SPTO	Software Project Tracking and Oversight (KPA)
SWM	SoftWare Manager (role)
SQA	Software Quality Assurance (KPA)
SSM	Software Subcontract Management (KPA)
SG	System engineering Group (role)
STG	Software Test Group (role)
SW-CMM	Capability Maturity Model for SoftWare
V	Verification (feature of key practice)
XS	eXtra Small enterprise
XXS	eXtra eXtra Small enterprise

Cross-references in the supplements are in the format *<KPA>/<feature>.<number>*, i.e. each reference refers to number *number* of feature *feature* in key process area *KPA* in the original SW-CMM.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Supplement B—Questionnaire

Introduction

The purpose of this questionnaire is to form a basis for interviews and support the interviewers in internal assessments of the maturity level within an organisation regarding software engineering; to help the organisation to identify strengths and weaknesses in the software engineering process. The interviews should de conducted by a person well familiarised with this very model; CMM for Small Organisations.

The questionnaire is split up according to the key process areas that characterises CMM with questions significant for each maturity level. The answers can be "yes", "no" or "n.a." (not applicable/don't know). It is recommended to validate some of the "yes"-answers by asking for and looking into the documentation that underlies the answers.

The few questions that might not be relevant in the context should be answered with "n.a." (not applicable). An example to be mentioned is the SSM key process area (Software Subcontract Management), a key process area that normally does not concern the small organisations XXS and XS.

For some questions answered with "yes", additional information is required, such as, whom?, who?, etc in order to increase the confidence in the answer and to evaluate the depth of the respondents knowledge in the subject concerned. The name should not be written down since the questionnaire does not comprise individuals or products but the maturity level of the engineering process of the organisation. Check that the answer agrees with the documentation – if not, it could indicate that the management has failed in communicating information.

To be able to put a measure on the results of the assessment, a weight factor "w" is used. The measure that can be used to prioritise specific process areas as well as to evaluate improvements in the process by comparing results from other assessments made later and see if any improvements really have been made and to what extent. The weight factor is throughout this paper set to the value 1, since in our opinion all questions are of equal importance.

Usage

Use the same copy of the questionnaire for all the interviews in one assessment. There are enough space for this between the questions. It simplifies the analysis to have all information on the same paper. Use different coloured pens; one colour for each interview/group and you can easily keep the different interviews/groups apart. The questionnaire should under all circumstances be handled confidentially, and the relation between group and colour shall not be revealed.

The questionnaire is split up into sections corresponding to the key process areas of maturity level 2 of the SW-CMM.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Requirements Management

The purpose of the Requirements Management is to establish a common understanding between the customer and the software project of the customer's requirements that will be addressed by the software project.

Requirements management is a basis for parallel activities such as preparation for acceptance tests, documentation, and education.

Goal 1: *System requirements allocated to software are controlled to establish a baseline for software engineering and management use.*

Goal 2: *Software plans, products and activities are kept consistent with the requirements allocated to software.*

RM	Requirements Management	Yes	No	n.a.	w
1.	Is there someone assigned to be responsible for analysing the requirements and to allocate them to hardware, software and other system components?	❑	❑	❑	
2.	Are adequate resources and funding available for managing the allocated requirements. - *Staff?* - *Time?* - *Tools?* - *Methods?*	❑	❑	❑	
3.	Do you think that members of the software engineering group and other individuals responsible for requirements management have been trained to perform their requirement activities. - *Who would need that education?* - *Is there such education offered? – To whom?*	❑	❑	❑	
4.	Are the requirements/changes in the requirements documented? - *Who documents them? – Where?*	❑	❑	❑	
5.	Are the commitments made to individuals and groups internal as well as external to the organisation documented? - *Who documents them? – Where?*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RM	Requirements Management	Yes	No	n.a.	w

6.	Are the requirements reviewed before they are incorporated into the software project?
	- *Who reviews? The software engineering group?*
	- *Are the requirements used as a basis for*
	  • *the project plan?*
	  • *working products?*
	  • *activities*
	- *How do you know that they have been reviewed (protocol, minutes of meeting, signature)?*
		❑	❑	❑	

7.	Are changes in the requirements reviewed before they are incorporated into the software project? - *By whom?* - *How do you know that they have been reviewed (protocol, meeting minutes, signature)?*	❑	❑	❑	
8.	Do you know the status of the activities for managing the allocated requirements? - *How many requirements have been changed?* - *The total number of requirements?* - *And how do you know all this?*	❑	❑	❑	
9.	Are the activities for managing requirements reviewed with senior manager (on a periodic basis)? - *How often?* - *Is the result documented?*	❑	❑	❑	
10.	Are the activities for managing requirements reviewed with project manager (on both a periodic and event-driven basis)? - *How often?* - *When something happens?* - *Is the result documented?*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RM	Requirements Management	Yes	No	n.a.	w
11.	Are commitments resulting from the allocated requirements negotiated with the affected groups? - *Who are involved in the negotiations?*	❑	❑	❑	
12.	Are measurements recorded? • *Status for every allocated requirement* • *Change activities for the allocated requirements* • *Accumulated number of changes in the allocated requirements including,* - *total number of changes suggested,* - *total number of changes approved,* - *total number of changes included in base-lines.*	❑	❑	❑	
13.	Do you know if there is a written policy for requirements management? *A policy that specifies* • *The allocated requirements are documented.* • *The allocated requirements are reviewed by the project manager and others affected.* • *The software plans, work products, and activities are changed to be consistent with changes to the allocated requirements.* - *Do you know where you can find it?*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Project Planning

The purpose of the Software Project Planning is to establish reasonable plans for performing the software engineering and for managing the software project.

Goal 1: *Software estimates are documented for use in planning and tracking the software project.*

Goal 2: *Software project activities and commitments are planned and documented.*

Goal 3: *Affected groups and individuals agree to their commitments related to the software project.*

SPP	Software Project Planning	Yes	No	n.a.	w
1.	Do you know if documented procedures for software project planning exist? - *To develop the project plan?* - *To derive the project's software schedule?* - *To derive estimates for* • *size/changes in size of software work products,* • *effort and costs,* • *critical computer resources?*	❑	❑	❑	
2.	Is there a project manager assigned for every project? - *Who?* - *Where is it documented?*	❑	❑	❑	
3.	Do you know if there is a software manager (SWM) appointed? - *Who? The project manager?* • *Responsible for the software development and operational environment (HW/SW) at the home site as well as at the customer computer site.*	❑	❑	❑	
4.	Do you think there are adequate resources and funding available for managing the software project planning? - *Experienced individuals?* - *Facilities/tools*	❑	❑	❑	
5.	Are the allocated requirements documented before the project planning activities start?	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPP	Software Project Planning	Yes	No	n.a.	w
6.	Have the software engineering group participated on the project proposal team - *Where is it registered (e.g. minutes of. meeting)?*	❑	❑	❑	
7.	Have the software engineering group reviewed - *The project-level plans?* - *The project's proposed commitments?* - *Are the reviews documented? – Where?*	❑	❑	❑	
8.	Do you know if the senior manager is well informed about all the commitments made to external individuals/groups? - *Are the reviews with the senior manager documented?* - *Do you know where to find these review notes?*	❑	❑	❑	
9.	Has the software manager (SWM) been oriented in the technical aspects of the project? - *Where is that documented?*	❑	❑	❑	
10.	Have estimations been derived for: - *Size/changes in size of software work products?* - *Effort and costs?* - *Critical computer resources?* - *Staffing?* - *Where is the information that has been used as the basis for these estimations?*	❑	❑	❑	
11.	Is there a documented procedure for estimations? Are the estimates derived according to that procedure?	❑ ❑	❑ ❑	❑ ❑	
12.	Have the software risks been identified, assessed and documented? *Risks associated with* - *cost,* - *resources,* - *schedule, and* - *technical aspects?*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPP	Software Project Planning	Yes	No	n.a.	w
13.	What do you know about software engineering facilities? - *Is there a plan prepared for the use of these?* - *Is the plan reviewed? – By whom? (All affected groups?)*	❑	❑	❑	
14.	Is the project plan developed according to a documented procedure?	❑	❑	❑	
15.	Are measurements made and registered? *Measurement used to determine the status of the software planning activities. For example* • *number of completed milestones in the project planning compared to the planned,* • *work effort and costs for the planning compared the planned.*	❑	❑	❑	
16.	Are the software planning activities reviewed? - *Periodically with the senior manager?* - *Periodically and event-driven with the project manager?* - *Are the reviews documented (review report)?* - *Where can you find them?* - *What do they look like? - Action items proposed as a result of the each review?*	❑	❑	❑	
17.	Is the project plan documented? - Including: - *Plans for involvement of the software engineering group in activities of other software-related groups;* - *Plans for other software-related groups involved in the activities of the software engineering group;* - *Plans for development facilities and tools;* - *Schedules;* - *The software life cycle with predefined milestones;* - *Commitments;* - *Risks;* - *Work products.*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPP	Software Project Planning	Yes	No	n.a.	w

18. Do you know if there is a written policy for software project planning? ❑ ❑ ❑

A policy that specifies:

- *The system requirements allocated to software are used as the basis for planning the software project.*
- *The software project's commitments are negotiated between the project manager and the software manager (SWM) responsible for the development environment.*
- *Involvement of other engineering groups in software activities is negotiated with these groups and is documented.*
- *Affected groups review the software project's estimates and schedules.*
- *Senior management reviews all software project commitments made to individuals and groups external to the organisation.*
- *The project plan is managed and controlled.*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Project Tracking and Oversight

The purpose of Software Project Tracking and Oversight is to provide adequate visibility into actual progress so the management can take effective actions when the software project's performance deviates significantly from the software plans.

Goal 1: *Actual results and performance are tracked against the software plans.*

Goal 2: *Corrective actions are taken and managed to closure when actual results and performance deviate significantly from the software plans.*

Goal 3: *Changes to software commitments are agreed to by the affected groups and individuals.*

SPTO	Software Project Tracking and Oversight	Yes	No	n.a.	w
1.	Do you know if documented procedures for software project tracking and oversight exist? - *For revising the project plan?* - *For reviewing results compared to the project plan?*	❑	❑	❑	
2.	Do you think there are adequate resources and funding available for managing the software project tracking and oversight? - *Experienced individuals?* - *Supporting facilities/tools?*	❑	❑	❑	
3.	Are all suggested changes that concern the project recorded? - *Changes in commitments?* - *New commitments?* - *Changes in the project plan?* - *Motives for new estimations?*	❑	❑	❑	
4.	Do you know if the senior manager is familiar with changes in the commitments made to external individuals/groups? - *Are the reviews with the senior manager documented?* - *Do you know where to find these review notes?*	❑	❑	❑	
5.	Have changes in commitments been negotiated with the software development group? - *Where is it documented (e.g. meeting minutes)?*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPTO	Software Project Tracking and Oversight	Yes	No	n.a.	w

6. Have actual results been followed-up against the estimates? ❑ ❑ ❑

   - *size and extent of software work and software products*
   - *costs and work effort*
   - *staffing*
   - *computer capacity*

   - *Has actual consumption of resources been registered?*
   - *Have correcting/revising measures been taken when the results diverged significantly?*
   - *Has the project plan been revised accordingly?*

7. What do you know about the risks with the project? Is there a risk analysis for the project and how is that handled? ❑ ❑ ❑

   - *Has the analysis been followed-up in connection with the changes in the project plan?*
   - *Has a correction been made when the results have significantly diverged from the project plan?*
   - *How has this been documented? – Where?*

8. Are all corrections reviewed before they are incorporated in the project plan? ❑ ❑ ❑

   - *By whom?*
   - *Where are the reviews documented?*

9. Are you participating in the project tracking and oversight? ❑ ❑ ❑

   - *Who is reviewing the project situation compared to the project plan?*
     - *SE together with PM, customer and SWM*
     - *Technical progress, plans, performance, other matters*
   - *How are these reviews documented?*

10. Are the milestones evaluated in relation to the project plan? ❑ ❑ ❑

    - *Are the divergences noted?*
    - *Have the effects of activities completed too late or too early been evaluated according to the influence on future development work and milestones?*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPTO	Software Project Tracking and Oversight	Yes	No	n.a.	w
11.	Have metrics been registered? • *Work effort and other resource consumption for the project tracking and oversight activities* • *Changes in the project plan concerning costs, project size, resources, the time schedule*	❑	❑	❑	
12	Have the project tracking and oversight activities been reviewed? - *With the business executive at regular intervals?* - *With the project manager at regular intervals?* - *With the project manager every time something unpredictable occurs?* - *Has the reviews been documented?* - *Where can you find them? – What do the look like?* - *Has an activity plan been produced as a result of every review?*	❑	❑	❑	
13.	Has the project plan been revised according to documented routine? - *Has it been put under revision management?* - *Have the changes been reviewed and approved?* - *Where and how are the reviews documented?*	❑	❑	❑	
14.	Do you know if there is a written policy for software project tracking and oversight? *A policy that specifies* • *A documented software development plan is used and maintained as the basis for tracking the software project.* • *The project manager is kept informed of the software project's status and issues.* • *Corrective actions are taken when the project results significantly deviate from the project plan, either by adjusting performance or by adjusting the plans.* • *Changes to the software commitments are made with the involvement and agreement of the affected groups.* • *Senior management reviews all commitment changes and new software project commitments made to individuals and groups external to the organisation.*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Subcontract Management

(S-organisations only)

The purpose of Software Subcontract Management is to select qualified software subcontractors and manage them efficiently.

Goal 1: *The prime contractor selects qualified software subcontractors.*

Goal 2: *The prime contractor and the software subcontractor agree on their commitments to each other.*

Goal 3: *The prime contractor and the software subcontractor maintain ongoing communications.*

Goal 4: *The prime contractor tracks the software subcontractor's results and performance against its commitments.*

SSM	Software Subcontract Management	Yes	No	n.a.	w
1.	Is anyone designated to be responsible for the planning and reviewing of subcontracting activities? - *Who?* - *Are the responsibilities documented?*	❑	❑	❑	
2.	Do you think there are adequate resources and funding available for managing the software project tracking and oversight? - *Personnel?* - *Time?* - *Tools?* - *Methods?* - *Education?*	❑	❑	❑	
3.	Does everyone involved get a briefing of the technical aspects of the subcontractors?	❑	❑	❑	
4.	Are there documented procedures for subcontract management? *Routines for* • *planning the work that will be done by the subcontractor* • *defining criteria used for selecting subcontractor* • *managing changes in the contract etc* • *reviews of the work and results of the subcontractor* • *following-up the SCM-activities of the subcontractor* • *acceptance test of the SQA-activities*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SSM	Software Subcontract Management	Yes	No	n.a.	w
5.	Are these procedures followed?	❑	❑	❑	
6.	Are the requirements of the product to be contracted documented?	❑	❑	❑	
7.	Are the contract and the project plan of the subcontractor used as a base for the subcontract management? - *Technical reviews of products and achievements according to the milestones specified in the project plan?*	❑	❑	❑	
8.	Is the project plan of the subcontractor reviewed? - *By whom?* - *Who approves it?*	❑	❑	❑	
9.	How are changes in the subcontractor's undertakings, work assignment and contract conditions managed? - *According to a documented procedure?* - *How else?*	❑	❑	❑	
10.	Does the senior manager evaluate the work status and co-ordination together with the subcontractor management?	❑	❑	❑	
11.	Are technical reviews and communication with the subcontractor performed? - *How often?* - *By whom?* - *Where can you see that it has been done? (protocol, signature)*	❑	❑	❑	
12.	Is the performance of subcontractor validated? - *How often?* - *By whom?*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SSM	Software Subcontract Management	Yes	No	n.a.	w
13.	Is there any knowledge of the status of the subcontractor management? - *Delays?* - *Deliverance?* - *Quality?* - *Is the status documented at different points in time?*	❑	❑	❑	
14.	Are the activities concerning subcontractor management reviewed with the senior manager? - *By whom?* - *How often?* - *Where can you see that it has been done? (protocol, signature)*	❑	❑	❑	
15.	Are the activities concerning subcontractor management reviewed with the project manager? - *By whom?* - *How often?* - *Where can you see that it has been done? (protocol, signature)*	❑	❑	❑	
16.	Do both the prime contractor and the subcontractor review the acceptance criteria and the routines? - *Do both sign their approval?* - *Is there an activity plan for products that fall outside the acceptance criteria?*	❑	❑	❑	
17.	Do both the contractor and the subcontractor approve agreements and changes concerning the contract? - *Do both sign them?*	❑	❑	❑	
18.	Are the results of the acceptance test documented?	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SSM	Software Subcontract Management	Yes	No	n.a.	w

19. Have metrics been registered? ❑ ❑ ❑

- *Costs for managing subcontractors compared to the plan.*
- *Actual date of deliverance compared to the scheduled date from the subcontractor.*
- *Actual date of deliverance compared to the scheduled date to the subcontractor.*

20. Do you know if there is a written policy for software subcontract management? ❑ ❑ ❑

A policy that specifies

- *Documented standards and procedures are used for managing the software subcontractors and the software subcontracts.*
- *The contractual agreements form the basis for subcontract management.*
- *Changes to the subcontract are made with the involvement and of agreement of both the prime contractor and the subcontractor.*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Quality Assurance

The purpose of Software Quality Assurance is to provide management appropriate visibility into the process being used by the software project and of the products being built.

Goal 1: *Software quality assurance activities are planned.*

Goal 2: *Adherence of software products and activities to applicable standards, procedures and requirements is verified objectively.*

Goal 3: *Affected groups and individuals are informed of software quality assurance activities and results.*

Goal 4: *Non-compliance issues that cannot be resolved within the software project are addressed by the senior manager.*

SQA	Software Quality Assurance	Yes	No	n.a.	w
1.	What do you know about quality assurance? Do you know if there is a quality assurance group in the organisation (SQA-group)? - *Independent of the management and development group?* - *Is it the same SQA group for each project?*	❑	❑	❑	
2.	Have you been informed about the - *Benefits of quality assurance?* - *Role, responsibility and authority of the SQA group*	❑	❑	❑	
3.	Do you think there are adequate resources and funding available for managing the software quality assurance? - *Experienced individuals?* - *Supporting facilities/tools*	❑	❑	❑	
4.	Do you think that those who are responsible for SQA have adequate competence for their work? - *The person responsible for SQA in the project?* - *Is the senior manager familiar with SQA?* - *Are you aware of what you have to do to fulfil the demands related to SQA?*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SQA	Software Quality Assurance	Yes	No	n.a.	w

5. Do you know if there is an SQA plan for the project you work with? ❑ ❑ ❑

   - *Where can it be found?*

   The plan specifies:
   - *The responsibility and authorities of the SQA group*
   - *Time schedule and resources for SQA-activities*
   - *The evaluations that will be performed by the SQA-group*
   - *The reviews that will be performed by the SQA-group*
   - *Routines concerning documentation and deviation follow-up.*
   - *The documentation that the SQA-group should produce*
   - *The way the feedback is given to the developing-group and other SQA-related groups, and how frequently*
   - *Other activities concerning the development of the product*

6. Do you know if there are documented procedures for producing a quality plan? ❑ ❑ ❑

   Routines that specifies that
   - *the plan is produced in an early stage, parallel to the project plan,*
   - *the plan is reviewed by concerned groups and individuals,*
   - *the plan is checked and managed, at least configuration managed.*

7. Does the SQA-group participate in the preparations of the project? ❑ ❑ ❑

   - *A review of the project plan?*
   - *Reviews of standards, routines?*
   - *Reviews of development activities?*

8. Are the results of the reviews reported to the software engineering group? ❑ ❑ ❑

   - *How?*
   - *Where?*

9. Are the project plan, standards and routines reviewed periodically, in order to verify the correspondence with ❑ ❑ ❑

   - *software engineering activities?*
   - *software products?*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SQA	Software Quality Assurance	Yes	No	n.a.	w
10.	How are the deviations reported? - *Are there documented procedures for how to manage deviations in the project?* - *How is it reported?* - *What does the reports look like?* - *Where are they to be found?*	❑	❑	❑	
11.	Have metrics been registered? • *Completed SQA milestones compared to the planned.* • *The effort in work and the consumption of other resources concerning SQA activities compared to the plan.* • *Number of reviews of activities and products compared to the plan.*	❑	❑	❑	
12.	Have the SQA activities been reviewed with - *The senior manager periodically?* - *The project manager periodically?* - *The project manager every time something unforeseen happens?* - *Have the reviews been documented?* - *Where are they? - What do the look like?*	❑	❑	❑	
14.	Have each review resulted in an activity plan?	❑	❑	❑	
15.	*If applicable:* Do you know if the customer SQA-personnel makes any reviews of the SQA-group activities and achievements? - *Where can you find these reports?*	❑	❑	❑	
16.	How are unsolved conflicts within the project group managed (e.g. divergence in standards)? - *Who deals with this? - The senior manager?*	❑	❑	❑	
18.	Are there an independent and external audit appointed and planned for auditing the work of the SQA-group? - *Have this been documented?* - *Do you know where these reports can be found?*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SQA	Software Quality Assurance	Yes	No	n.a.	w
19.	If an SQA group exists, do the group review both activities and products? - *How often?* - *Event-driven?* - *Are the results documented? - How?* - *To whom?*	❑	❑	❑	
20.	Does the SQA group review activities and products from the following? - *Requirements management* - *Software project planning* - *Software project tracking and oversight* - *Subcontract management* - *The subcontractor's quality assurance activities* - *Software configuration management*	❑	❑	❑	
21.	Does the SQA-group report the results from the reviews?	❑	❑	❑	
22.	Do you know if there is a written policy for software quality assurance? *A policy that specifies* • *The SQA function is in place on all software projects.* • *The SQA group has a reporting channel to senior management that is independent of the project manger, the project's software engineering group, and the other software related groups.* • *Organisations must determine the organisational structure that will support activities that require independence..* • *Senior management periodically reviews the SQA activities and results.*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Configuration Management

The purpose of Software Configuration Management is to establish and maintain the integrity of the products of the software project throughout the project's software life cycle.

Goal 1: *Software configuration management activities are planned.*

Goal 2: *Selected software work products are identified, controlled, and available.*

Goal 3: *Changes to identified software work products are controlled.*

Goal 4: *Affected groups and individuals are informed of the status and contents of software baselines.*

SCM	Software Configuration Management	Yes	No	n.a.	w
1.	What do you know about configuration management? Do you know what it means, its purpose, …? Have you any kind of training/education? What is a baseline? - *What kind of training/education or information?*	❑	❑	❑	
2.	Do you know if there are documented procedures for configuration management • *To produce a SCM-plan?* • *For changes in the requirements and for reports concerning problems about the SCM-items?* • *The creation of products from the baseline library and for controlling the releases?* • *To control baseline changes?* • *To control status of SCM-items?* • *For baseline revisions*	❑	❑	❑	
3.	Do you know if there is anyone appointed responsible for configuration management? - *Software Configuration Management group (SCM)?* - *S organisations only: Software Configuration Control Group (SCCB)?* - *Who?* - *Where is this documented?*	❑	❑	❑	
4.	Do you find the persons responsible for SCM/ SCCB educated and competent in what they do?	❑	❑	❑	
	Do you know whether the personnel participating in the SCM-activities have any formal education in SCM?	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCM	Software Configuration Management	Yes	No	n.a.	w
5.	Do you know if there is a CM-plan for the project you work with? - *Reviewed by whom?* - *Approved by whom?* *The plan comprises:* • *SCM-activities, time schedule for these activities, a responsibility breakdown chart and requirements on resources, including personnel, tools and computer resources.* • *SCM-requirements and activities performed by the software development group and/or other software related groups.*	❑	❑	❑	
6.	Is this plan followed?	❑	❑	❑	
7.	Do you know whether every project has identified configuration items? - *Which are they? Name a few!* *Examples of CM-items are* • *requirements specifications, project plans, design specifications, data definitions, source code, and make files.*	❑	❑	❑	
8.	How are change requests and problem reports handled? - *I there a documented plan to follow?* - *Is it followed?* - *Are the propositions documented?* - *Are they reviewed? - Are they approved? - By whom?* - *Are they completed in a defined way? (introduced/rejected)*	❑	❑	❑	
9.	Do you know if there is a baseline library? - *Where is it?* - *Do you know how to get the information?* - *Is the library available to everyone within the project?*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCM	Software Configuration Management	Yes	No	n.a.	w

10. How are changes in the baselines managed? ❑ ❑ ❑

- *According to a documented routine?*
- *Are they sanctioned by SCM/SCCB?*
- *Reviewed and approved?*
- *Are the changes logged?*
- *Is the library updated after every approved change?*
- *Do all personnel in the project receive information of and get access to all changes made to the library? (is change information distributed?)*

11. Do you know if there are any reports made concerning the SCM-activities? ❑ ❑ ❑

- *Records from SCM/SCCB-meetings?*
- *Summaries and status on the changing proposals?*
- *Status of the SCM-items?*
- *Status concerning problem reports?*
- *Baseline revision history?*

12. Are the affected groups/individuals informed about the status and the contents of baselines? ❑ ❑ ❑

13. Have metrics been used? ❑ ❑ ❑

- *Which are they? Name a few!*

Examples:

- *Completed CM-milestones compared to the planed.*
- *The effort in work and the consumption of other resources concerning CM-activities compared to the plan.*
- *Number of reviewed and completed changing proposals per time unit.*

14. Have the SCM-activities been reviewed? ❑ ❑ ❑

- *Periodically with the senior manager?*
- *Periodically by the project manager?*
- *By the project manager every time something unforeseen happens (event-driven)?*
- *Have the reviews been documented?*
- *Where are they? - What do they look like?*
- *Have each review resulted in an activity plan?*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCM	Software Configuration Management	Yes	No	n.a.	w
15.	Is there a SQA-group appointed, or someone who formally have the responsibility for SQA concerning the configuration activities? - *Who (internal/external)?* - *Where is this documented?* - *Does the group identify adequate configuration items?*	❑	❑	❑	
16.	Does the SCM-group review the subcontractor's configuration management activities? - *How? - According to a documented procedure?*	❑	❑	❑	
17.	Do you know if there is a written policy for software configuration management? *A policy that specifies:* • *Responsibility for SCM for each project is explicitly assigned.* • *SCM is implemented throughout the project's life cycle.* • *SCM is implemented for software products to be delivered externally, designated internal software work products, and designated support tools used inside the project (e.g., compilers).* • *The projects establish or have access to a repository for storing items/units and the associated SCM records.* • *The contents of this repository are referred to as the "software baseline library" in these practices.* • *Work product that are placed under configuration management and treated as a single entity are referred to as configuration items.* • *The software baselines and SCM activities are reviewed on a periodic basis.*	❑	❑	❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Supplement C—Checklists

Introduction

This supplement contains a number of checklists to help small companies or organisations to make self-assessments of their software development processes; if policy, standards, processes, procedures, education, training, and tools are consistent with the model we have described in "CMM for Small Organisations" (UMINF00.20). The checklists are not to be regarded as a formal instrument for a CMM assessment, but to give an indication of the quality of the development processes, and to show the week points, where software process improvement might have the greatest effect.

The primary target group for this document are persons within the organisation responsible for software process improvement. The document also addresses persons involved in the software quality assurance process (SQA), and the management responsible for software planning, management, development, and maintenance.

We also recommend studying supplements A and D, which provide information on documentation and measurements. Supplement A contains guidance for the documentation and how it ought to be worked out. It provides examples for document contents, e.g. policies for specific key process areas. Supplement D discusses the how and why of measurement in general and appropriate measurements for the level 2 key process areas in particular.

The checklists cover all Key Process Areas (KPA) of the SW-CMM maturity level 2. For each KPA, we provide four different types of checklists:

- ***Roles*** involved in any activities within the KPA.
- ***Entry criteria*** provide guidance for the conditions that should be fulfilled before any activities in a certain KPA are started. It also lists all kinds of documentation required before start. Entry criteria have been subdivided into *General Entry Criteria* and *Project Specific Entry Criteria*.
- ***Activities*** recommended in the model.
- ***Exit criteria*** provide guidance for the conditions that should be fulfilled when the activities in a certain KPA are ended. It also lists the documentation required **before** the KPA is regarded as being properly accomplished.

Each checklist has also some spare space for adding organisation specific recommendations.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Requirements Management

The purpose of Requirements Management is to establish a common understanding between the customer and the software project of the customer's requirements that will be addressed by the software project.

Requirements Management involves establishing and maintaining an agreement with the customer on the requirements for the software project. This agreement is referred to as the "system requirements allocated to software." The agreement forms the basis for estimating, planning, performing, and track the software project's activities throughout the software life cycle. Whenever the system requirements allocated to software are changed, the affected plans, work products, and activities are adjusted to remain consistent with the updated requirements.

Roles

The table below lists the roles with responsibility for some activity/activities in the requirements management process.

- Senior Manager (SM),
- Project Manager (PM),
- Software Engineering group (SE),
- Responsible for analysing and allocating the system requirements, and
- Software Quality Assurance group (SQA).

Entry Criteria

The model recommends that the conditions described in the entry criteria tables below be satisfied before entering the requirements management process. The first table shows the general process entry criteria, and the second shows the project specific entry criteria.

General Entry Criteria - Requirements Management			
			Ref.
1	A Software Quality Assurance group (SQA) exists.	❑	
2	A written policy for requirements management exists.	❑	C2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Specific Entry Criteria - Requirements Management			
			Ref.
1	For each project, responsibility has been established for analysing the system requirements and allocating them to hardware, software and other components.	❑	AB1
2	Adequate resources and funding are available for managing the allocated requirements.	❑	AB3
3	Members of the software engineering group have been trained to perform their requirement activities.	❑	AB4
4	Allocated requirements, and/or changes to the allocated requirements are documented.	❑	C1.1, AB2
5	Tools to support the activities for managing requirements are available.	❑	AB3.2
		❑	
		❑	
		❑	
		❑	
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Activities

The table below lists the recommended activities for the requirements management process.

Activities - Requirements Management		
1	The software engineering group reviews allocated requirements before they are incorporated into th~ software project.	❑
2	Commitments resulting from the allocated requirements are negotiated with the affected groups.	❑
3	The software engineering group uses the allocated requirements as the basis for plans, work products, and activities.	❑
4	Changes to the allocated requirements are reviewed and incorporated into the software project.	❑
5	Measurements are made and used to determine the status of the activities for managing the allocated requirements.	❑
6	The activities for managing requirements are reviewed with senior manager on a periodic basis.	❑
7	The activities for managing requirements are reviewed with project manager on both a periodic and event-driven basis.	❑
8	The software quality assurance group reviews the activities and work products for managing the alloeated requirements and reports the result.	❑
		❑
		❑
		❑
		❑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Exit Criteria

The model recommends that outputs satisfy the states described in the table below to exit the requirements management process.

Exit Criteria - Requirements Management			
			Ref.
1	Allocated requirements, or changes to allocated requirements are documented.	❑	C1, AC1, AC3
2	Commitments, or changes to commitments are documented.	❑	AC1, AC3
3	Commitments have been negotiated with affected groups and documented (meetings, signatures....)	❑	AC3
4	Review reports exist from • activities reviews with the senior manager • activities reviews with the project manager	 ❑ ❑	 V1 V2
5	SQA reports exist, i.e., result reports from the SQA reviews.	❑	V3
6	Measurements have been made to determine the status of the activities for managing the allocated requirements. The measurements have been recorded.	❑	M1
7	Software statement of work, and work products exist.	❑	V3
8	Changes to plans, project activities, and work products are documented.	❑	V3
		❑	
		❑	
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Project Planning

The purpose of Software Project Planning is to establish reasonable plans for performing the software engineering and for managing the software project.

The software planning process is normally carried trough once per project. When the project plan, project estimates, work products, and/or project activities are changed the software project tracking and oversight process will take over. However, if the changes affect the allocated requirements, or if new requirements have been added to the project, the requirements management process is to be carried out again.

Roles

The table below lists the roles with responsibility for some activity/activities in the software project planning process.

- Senior Manager (SM),
- Project Manager (PM),
- Software Manager (SWM),
- Software Engineering group (SE),
- Software Quality Assurance group (SQA), and
- Software Configuration Management group (SCM).

Entry Criteria

The model recommends that the conditions described in the entry criteria tables below be satisfied before entering the software project planning process. The first table shows the general process entry criteria, and the second shows the project specific entry criteria.

General Entry Criteria - Software Project Planning			
			Ref.
1	Certain roles have been assigned: • Software Manager (SWM) • Software Quality Assurance group (SQA) • Software Configuration Management group (SCM)	❑ ❑ ❑	
2	A written policy for software project planning exists	❑	C2
3	A documented procedure for software project planning exists.	❑	AC6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Activities

The table below lists the recommended activities for the software project planning process. The software planning begins with a statement of the work to be performed and other constrains and goals that define and bound the software project (those established by the practices of the Requirements Management key process area).

Activities - Software Project Planning		
1	The software engineering group: • participates on the project proposal team, • reviews the project-level plans, • reviews the project's proposed commitments.	❑ ❑ ❑
2	Commitments made to individuals and groups external to the organisation are reviewed with senior manager.	❑
3	A software life cycle with predefined stages of manageable size is identified or defined	❑
4	Adequate milestones are defined	❑
5	Adequate configuration items/units are defined with the software configuration group.	❑
6	Estimates are derived according to a documented procedure for: • size/changes in size of software work products, • effort and costs, • critical computer resources.	❑ ❑ ❑
7	The project's software schedule is derived according to a documented procedure	❑
8	The software risks of the project are identified, assessed and documented Risks associated with: - cost, - resource, - schedule, and - technical aspects	❑
9	Plans for the project's software engineering facilities and support tools are • prepared • reviewed by all affected groups	❑ ❑
10	Software planning data are recorded. Information includes the estimates and the associated information needed to reconstruct the estimates and assess their reasonableness.	❑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Activities - Software Project Planning		
11	Measurements are made and used to determine the status of the software planning activities	❑
12	Software project planning activities are reviewed: • periodically with the senior manager, • both periodically and event-driven with the project manager	❑ ❑
13	The project plan is developed according to a documented procedure	❑
14	The plan for the software project is documented	❑
15	The software quality assurance group (SQA) • reviews the activities and work products, • reports the result.	❑ ❑
		❑
		❑
		❑
		❑
		❑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Exit Criteria

The model recommends that outputs satisfy the states described in the table below to exit the software project planning process.

Exit Criteria - Software Project Planning			
			Ref.
1	Involvement of the software engineering group in the software project planning activities is documented (e.g., minutes of meetings)	❑	C2.3
2	The reviews by the senior manager concerning commitments made to external individuals/groups are documented (review notes)	❑	C2.5
3	The project plan is documented, including: • plans for involvement of the software engineering group in activities of other software-related groups, • plans for other software-related groups involved in the activities of the software engineering group, • plans for development facilities and tools, • schedules, • the software life cycle with predefined milestones, • commitments, • risks, • work products.	❑ ❑ ❑ ❑ ❑ ❑ ❑ ❑	AC7
4	Estimates have been documented: • size or changes in size of software work products • critical computer resources • capacity requirements for facilities and tools • the software project's effort and costs.	❑ ❑ ❑ ❑	AC11 AC9 AC10 AC11
5	Assumptions are made in deriving • estimates for the software project's effort and costs • the project's software schedule.	❑ ❑	AC10 AC12
6	Measurement data to determine the status of the software planning activities are registered.	❑	M1
7	The reviews with the senior manager and the project manager are documented as: • review reports, • action items.	❑ ❑	V1, V2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Exit Criteria - Software Project Planning			
			Ref.
8	Results from the SQA group reviews of the activities and work products for software project planning.	❑	V3
		❑	
		❑	
		❑	
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Project Tracking and Oversight

The purpose of Software Project Tracking and Oversight is to provide adequate visibility into actual progress so that management can take effective actions when the software project's performance deviates significantly from the plans.

Roles

The table below lists the roles with responsibility for some activity/activities in the software project tracking and oversight process.

- Senior Manager (SM),
- Project Manager (PM),
- Software Manager (SWM),
- Software Engineering group (SE),
- Software Quality Assurance group (SQA), and
- Software Configuration Management group (SCM).

Entry Criteria

The model recommends that the conditions described in the entry criteria tables below be satisfied before entering the software project tracking and oversight process. The first table shows the general process entry criteria, and the second shows the project specific entry criteria.

General Entry Criteria - Software Tracking and Oversight			
			Ref.
1	Roles have been assigned: • Software Manager (SWM) • Software Quality Assurance group (SQA) • Software Configuration Management group (SCM)	❑ ❑ ❑	
2	A written policy for software project tracking and oversight exists	❑	C2
3	Documented procedures for software project tracking and oversight exist.	❑	AC2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Specific Entry Criteria - Tracking and Oversight			
			Ref.
1	A project manager has been assigned, responsible for the software project planning and software project tracking and oversight.	❑	C1
2	A documented project plan, reviewed, and approved exists	❑	AB1
3	Adequate resources and funding are available for managing software project tracking and overview. • project manager is assigned specific responsibilities, • tools to support software tracking.	❑ ❑	AB3
4	The project manager and the software manager have been trained in managing the technical and personnel aspects of the software project.	❑	AB4
5	The software manager has been oriented in the technical aspects of the software project.	❑	AB5
6	Changes in the software project have been recorded: • changes to commitments, • new commitments proposed, • changes to the project plan.	❑ ❑ ❑	AC2
7	Changes to planning data have been recorded. Information includes the estimates and the associated information needed to reconstruct the estimates and assess their realism.	❑	AC11
		❑	
		❑	
		❑	
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Activities

The table below lists the recommended activities for the software project tracking and oversight process.

Software activities are monitored by the management. Progress is primarily determined by comparing the actual software size, effort, cost, and schedule to the plan when selected software work products are completed and at selected milestones. When it is determined that the software project's plans are not being met, corrective actions are taken. These actions may include revising the project plan to reflect the actual accomplishments and replanning the remaining work or taking actions to improve the performance.

Activities - Software Tracking and Oversight		
1	Commitments or changes to commitments made to individuals and groups external to the organisation are reviewed with senior manager.	❑
2	Approved changes to commitments that affect the software project are communicated to the members of the software engineering group.	❑
3	Estimates are tracked and corrective actions are taken as necessary. Actual values are compared to the documented estimates for: • size of the software work products, • effort and costs, • critical computer resources.	❑ ❑ ❑
4	Tracking and corrective actions are taken for • the project's software schedule, • technical aspects.	❑ ❑
5	The software risks are tracked and corrective actions are taken. Risks associated with: - cost, - resource, - schedule, and - technical aspects	❑
6	The software engineering group conducts periodic reviews against the project plan. To track: - technical progress, - plans, - performance, and - issues	❑
7	Effects of early or late completion of software activities, milestones and other commitments are evaluated for impacts on future activities and milestones.	❑
8	Actual completion of software activities and milestones is compared against the project plan.	❑
9	Formal reviews to address the accomplishments and results of the	❑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Activities - Software Tracking and Oversight		
	software project are conducted at selected project milestones according to a documented procedure.	
10	Measurements are made and used to determine the status of the software tracking and oversight activities	❑
11	Software project tracking and oversight activities are reviewed: • periodically with the senior manager, • both periodically and event-driven with the project manager	❑ ❑
12	The project plan is documented and revised according to a documented, and managed and controlled. "Managed and controlled" implies that the version of the work product in use at a given time (past or present) is known (i.e., version control) and changes are incorporated in a controlled manner (i.e., change control).	❑
13	The quality assurance group (SQA) • reviews and revises the software tracking and oversight activities, • reviews and revises the products, and • reports the result.	❑ ❑ ❑
		❑
		❑
		❑
		❑
		❑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Exit Criteria

The model recommends that outputs satisfy the states described in the table below to exit the software tracking and oversight process.

Exit Criteria - Software Tracking and Oversight			
			Ref.
1	The reviews by the senior manager concerning changes to commitments made to external individuals/groups are documented (e.g., review notes, signatures…)	❑	AC3
2	Communication to the members of the software engineering group about changes to commitments, that affect the software project, are documented (e.g., minutes of meeting).	❑	AC4
3	Actual use of resources has been recorded: • size of the software work products, • effort and costs, • critical computer resources.	 ❑ ❑ ❑	 AC5 AC6 AC6
4	Corrective actions of estimates as a result of deviations compared to the project plan are documented.	❑	
5	Changes to estimates that affect software commitments are negotiated with the affected groups are documented, changes in • size of software work products, and • staffing.	 ❑ ❑	 AC5, AC6
6	Milestones are tracked against the project plan and the software schedule is revised and documented: • completed commitments, • completed activities, • evaluation of effects of early or late completion of software activities, milestones and other commitments, that have impacts on future activities and milestones.	❑	AC8
7	Problems identified in any of the software work products are documented.	❑	AC9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Exit Criteria - Software Tracking and Oversight			
			Ref.
8	Software risks have been analysed, updated, prioritised, and documented , risks associated with • cost aspects, • resource aspects, and • technical aspects.	❑	AC10
9	Revised project plan • is documented, • has been reviewed (review protocol notes), and • has been managed and controlled.	❑ ❑ ❑	AC2
10	Measurement data to determine the status of the software tracking and oversight activities have been recorded.	❑	M1
11	The reviews with the senior manager and the project manager are documented: • reports, and • action items.	❑ ❑	V1, V2
12	Performance has been reviewed against the project plan with the senior manager and the project manager, and documented: • effort and costs, • schedule, • staffing, and • technical aspects.	❑ ❑ ❑ ❑	V1, V2
13	Results from the SQA group reviews of the activities and work products for software project tracking and oversight.	❑	V3
		❑	
		❑	
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Subcontract Management

The purpose of Software Subcontract Management is to select qualified software subcontractors and manage them effectively.

Roles

The table below lists the roles with responsibility for some activity/activities in the software project tracking and oversight process.

- Responsible for Software Subcontract (SSM),
- Project Manager (PM),
- Subcontractor, and
- Software Configuration Management group (SCM).

Entry Criteria

The model recommends that the conditions described in the entry criteria tables below be satisfied before entering the software subcontract management process. The first table shows the general process entry criteria, and the second shows the project specific entry criteria.

General Entry Criteria - Software Subcontract Management			
			Ref.
1	A group responsible for Software Configuration Management (SCM) exists.	❑	
2	A written policy for software subcontract management exists	❑	C1
3	Documented procedures exist for • definition and planning of work to be subcontracted, • selection of subcontractor, • resolving changes to the software subcontractor's statement of work, subcontract terms and other commitments, • formal reviews to address the subcontractor's software engineering accomplishments and results, • monitoring the subcontractor's activities for software quality assurance, • monitoring the subcontractor's activities for software configuration management, and • acceptance testing as a part of the delivery of the subcontractor's software work products	 ❑ ❑ ❑ ❑ ❑ ❑	 AC1 AC2 AC6 AC9 AC10 AC11 AC12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Specific Entry Criteria - Subcontract Management			
			Ref.
1	A subcontract manager has been designated to be responsible for establishing and managing the software subcontract.	❑	C2
2	Adequate resources and funding are available for selecting the software subcontractor and managing the subcontract.	❑	AB1
3	The subcontract manager is knowledgeable and experienced in software engineering.	❑	C2
4	Individuals who are involved in establishing and managing the software subcontract have been trained to perform these activities.	❑	AB2
5	The project manager and other individuals involved in managing the software subcontract have received orientation in the technical aspects of the subcontract.	❑	AB3
6	The software requirements of the work to be subcontracted and the standards to be followed exists.	❑	AC1.2
		❑	
		❑	
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Activities

The table below lists the recommended activities for the software subcontract management process.

Activities - Software Subcontract Management		
1	The work to be subcontracted is defined and planned according to a documented procedure.	❑
2	The software subcontractor is selected, based on an evaluation of the ability to perform the work.	❑
3	A documented subcontractor's project plan is reviewed and approved.	❑
4	Changes to the software subcontractor's statement of work, subcontract terms, and other commitments are resolved.	❑
5	Senior manager reviews periodically status/coordination with the software subcontractor's management.	❑
6	Periodic technical reviews and interchanges are held with the software subcontractor.	❑
7	Formal reviews to address the subcontractor's software engineering accomplishments and results are conducted at selected milestones.	❑
8	The SQA group monitors the subcontractor's SQA activities according to a documented procedure.	❑
9	The subcontractor's technical, cost, and schedule performance is reviewed against the subcontractor's project plan	❑
10	Measurements are made to determine the status of the software subcontract management activities	❑
11	Software subcontract management activities are reviewed: • periodically with the senior manager, • both periodically and event-driven with the project manager	❑ ❑
12	Software subcontract management activities and work products are reviewed by the SQA group and the result is reported.	❑
13	The SCM group monitors the subcontractor's SCM activities according to a documented procedure.	❑
14	Acceptance test of the software subcontractors products are performed according to a documented procedure.	❑
		❑
		❑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Exit Criteria

The model recommends that outputs satisfy the states described in the table below to exit the software subcontract management process.

Exit Criteria - Software Subcontract Management			
			Ref.
1	Acceptance procedures and criteria for each product (defined, reviewed, approved, and documented).	❑	AC12
2	An action plan for any software product that does not pass its acceptance test.	❑	AC12
4	Action items for resulting from periodic status reviews.	❑	AC7
5	Contractual agreements.	❑	C1
6	Evaluation of the subcontract bidders' ability to perform the work.	❑	AC2
7	Changes to the subcontract, i.e.: - subcontract - subcontractor's statement of work - terms and conditions	❑	AC6
8	Results from the acceptance tests.	❑	AC12
9	Plan for selecting a subcontractor.	❑	AC1
10	Measurements (to determine the status of the software subcontract management activities).	❑	M1
11	The reviews with the senior manager and the project manager are documented: • reports, • action items.	❑	V1, V2
11	Results from the SQA group reviews of the activities and work products for software subcontract management.	❑	V3
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Quality Assurance

The purpose of Software Quality Assurance is to provide management with appropriate visibility into the process being used by the software project and of the products being built.

The software quality assurance group works with the software project during its early stages to establish plans, standards, and procedures that will add value to the software project and satisfy the constraints of the project and the organisation's policies. The software quality assurance group reviews project activities and reviews software work products throughout the life cycle and provides management with visibility as to whether the software project is adhering to its established plans, standards, and procedure

Roles

The table below lists the roles with responsibility for some activity/activities in the software project tracking and oversight process.

- Senior Manager (SM),
- Project Manager (PM),
- Software Engineering group (SE),
- Software Quality Assurance group (SQA),
- Experts independent of the SQA group, and
- Customer SQA representative (CSQA).

Entry Criteria

The model recommends that the conditions described in the entry criteria tables below be satisfied before entering the software quality assurance process. The first table shows the general process entry criteria, and the second shows the project specific entry criteria.

General Entry Criteria - Software Quality Assurance			
			Ref.
1	A written policy for software quality assurance exists.	❑	C1
2	Documented procedure for preparing a SQA plan exists.	❑	AC1
3	Documented procedures for management of the deviations in the software activities and software products exist.	❑	AC7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Specific Entry Criteria - Software Quality Assurance			
			Ref.
1	A group responsible for coordinating and implementing SQA for the project exists.	❑	AB1
2	A manager has been assigned specific responsibilities for the project's SQA activities.	❑	AB2
3	Adequate resources and funding are available for managing software quality assurance.	❑	AB2
4	Tools to support the SQA activities are available.	❑	AB2
5	Members of the SQA group have been trained to perform their SQA activities.	❑	AB3
6	All managers in the SQA reporting chain to the senior manager are knowledgeable in the SQA role responsibilities, and authority.	❑	AB2
7	The members of the software project have received orientation on the role, responsibilities, authority, and value of the SQA group.	❑	AB4
		❑	
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Activities

The table below lists the recommended activities for the software quality assurance process.

Activities - Software Quality Assurance		
1	A SQA plan is prepared for the software project in the early stages, and in parallel with the overall project planning. • the plan is reviewed by the affected groups and individuals, • the plan is managed and controlled - (version and change control).	❑ ❑
2	The SQA group's activities are performed in accordance with the SQA plan.	❑
3	The SQA group participates in the preparation and review of the project plan, standards, and procedures.	❑
4	The SQA group reviews the software engineering activities to verify compliance: • development activities, • selected software work products.	❑ ❑
5	The SQA group periodically reports the results of its activities to the software engineering group.	❑
6	Deviations identified in the software activities and software work products are documented and handled according to a documented procedure.	❑
7	Deviations in the project plan are reported to senior manager via a report chain independent of: - project manager, - software engineering group, and - other groups/individuals involved in the project development	❑
8	Software quality assurance activities are reviewed: • periodically with the senior manager, • both periodically and event-driven with the project manager	❑ ❑
9	Experts independent of the SQA group periodically review the SQA activities and products.	❑
10	The SQA group periodically review their activities and results with the customer SQA, when appropriate.	❑
11	Status of the SQA activities are measured.	❑
		❑
		❑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Exit Criteria

The model recommends that outputs satisfy the states described in the table below to exit the software quality assurance process.

Exit Criteria - Software Quality Assurance			
			Ref.
1	A SQA-plan for the project exists, reviewed, managed and controlled - (version and change control).	❑	AC1
2	Deviations between the project plan and the software engineering activities. Review report exists.	❑	AC4
3	Corrections to deviations between the software engineering activities and the • the designated software standards, • the project plan.	❑	AC4
4	SQA reports, i.e. reports from reviews of the software activities and work products.	❑	AC5
5	Measurement data have been recorded and used to determine the cost and schedule the status of the SQA activities..	❑	M1
6	The reviews with the senior manager and the project manager are documented: • reports, • action items	❑ ❑	V1, V2
7	Reports from reviews held by experts independent of the SQA group. - review of SQA activities, - review of products.	❑	V3
		❑	
		❑	
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software Configuration Management

The purpose of Software Configuration Management is to establish and maintain the integrity of the products of the software project throughout the project's software life cycle.

Roles

The table below lists the roles with responsibility for some activity/activities in the software project tracking and oversight process.

- Senior Manager (SM),
- Project Manager (PM),
- Software Manager (SWM),
- Software Engineering group (SE),
- Software Quality Assurance group (SQA),
- Software Configuration Management group (SCM), and
- Software Configuration Control Board (SCCB, S organisations only).

Entry Criteria

The model recommends that the conditions described in the entry criteria tables below be satisfied before entering the software configuration management process. The first table shows the general process entry criteria, and the second shows the project specific entry criteria.

General Entry Criteria - Software Configuration Management			
			Ref.
1	Roles have been assigned: • Software Manager (SWM), • Software Quality Assurance group (SQA), • Software Configuration Management group (SCM), • Software Configuration Control Board (SCCB).	❑ ❑ ❑ ❑	
2	A written policy for software configuration management exists	❑	C2
3	Documented procedures for managing software configuration exist.	❑	AC5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Specific Entry Criteria - Configuration Management			
			Ref.
1	A group responsible for coordinating and implementing SCM for the project exists. for XXS- and XS-organisation also responsible for baselines (SCCB)	❑	C1, AB3
2.	A board having the authority for managing the project's software baselines (SCCB) exists. (S organisation only)	❑	AB1
3	Adequate resources and funding are available for software configuration management.	❑	AB3, AB4
4	Tools to support the SCM activities are available.	❑	AB3
5	Members of the software engineering group and other software-related groups have been trained to perform their SCM activities.	❑	AB5
6	A documented and approved SCM plan exists.	❑	AC2
7	Identified software work products to be placed under configuration management (configuration item/units) exist.	❑	AC4
8	Change requests and/or problem reports exist. • change request for identified configuration item/units, • problem reports concerning identified configuration item/units, • baselines, and • changes in baselines.	 ❑ ❑ ❑ ❑	 AC5 AC5 AC6 AC6
		❑	
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Activities

The table below lists the recommended activities for the software configuration management process.

Activities - Software Configuration Management		
1	A SCM plan is prepared for each software project according to a documented procedure: • reviewed by affected groups, • documented and approved, and • managed and controlled (version and change control).	❑
2	A configuration management library system is established as a repository for the software baselines.	❑
3	The software work products to be placed under configuration management are identified. • the configuration items/units are selected based on documented criteria, • the configuration items/units are assigned unique identifiers, • the characteristics of each configuration item/unit are specified, • the software baselines to which each configuration item/unit belongs are specified, • the person responsible for each configuration item/unit (i.e., the owner, from a configuration management point of view) is identified.	❑ ❑ ❑ ❑ ❑
4	Change requests and problem reports for all configuration items/units are handled according to a documented procedure: • initiated, • recorded, • reviewed, • approved, and • tracked to closure.	❑ ❑ ❑ ❑ ❑
5	Changes in baselines are handled according to a documented procedure: • approved by SCCB, • reviewed, • tested (regression tests), • archived, and • a change log is created.	❑ ❑ ❑ ❑ ❑
6	Products from the software baseline library are created and their release is controlled according to a documented procedure.	❑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Activities - Software Configuration Management		
7	Standard reports documenting the SCM activities and the contents of the software baseline are developed and made available to affected groups and individuals.	❑
9	Measurements are made and used to determine the status of the SCM activities.	❑
10	Software configuration management activities are reviewed: • periodically with the senior manager, • both periodically and event-driven with the project manager	 ❑ ❑
11	The SCM group periodically reviews software baselines to verify that they conform to the documentation that defines them.	❑
12	The SQA-group reviews the activities and work products for SCM and reports the result.	❑
		❑
		❑
		❑
		❑

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Exit Criteria

The model recommends that outputs satisfy the states described in the table below to exit the software configuration management process.

Exit Criteria - Software Configuration Management			
			Ref.
1	The SCM-plan is • reviewed by affected groups, • placed under version control, • documented, and • approved.	 ❑ ❑ ❑ ❑	AC1
2	A baseline library, is available to affected groups/individuals.	❑	AC3
3	List of selected configuration items/units.	❑	AB1
4	Archive versions of configuration items/units.	❑	AC1
5	Changes to the software baselines.	❑	AB1
6	Current status and history (i.e. changes and other actions) of each configuration item/unit.	❑	AC8
7	Products from the software baseline library.	❑	AB1
8	Standard reports documenting the SCM activities and the contents of the software baseline.	❑	AC9
9	Results of the software baseline review.	❑	AC10
10	Measurement data (status of the SCM activities).	❑	M1
11	The reviews with the senior manager and the project manager are documented. • reports, and • action items.	 ❑ ❑	V1, V2
12	Revision reports of baselines.	❑	V3
13	Results from the SQA group reviews of the activities and work products for SCM.	❑	V4
		❑	
		❑	
		❑	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

นางสาวดวงสมร พันธ์พิกุล เกิดเมื่อวันที่ 28 กรกฎาคม พ.ศ.2518 จบการศึกษาในระดับปริญญาตรี จากคณะวิทยาศาสตร์ สาขาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยเชียงใหม่ ในปี พ.ศ. 2540 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต คณะเทคโนโลยีสารสนเทศ ภาควิชาเทคโนโลยีสารสนเทศ สาขาวิชาเทคโนโลยีสารสนเทศ แขนงวิชาวิทยาการสารสนเทศ เมื่อ พ.ศ.2544 ปัจจุบันทำงานในตำแหน่งเจ้าหน้าที่ระบบคอมพิวเตอร์ ฝ่ายประยุกต์เทคโนโลยี ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้