

การพัฒนาแนวปฏิบัติที่ดีและต้นแบบระบบตรวจสอบการพัฒนาซอฟต์แวร์  
และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110  
ระดับ Basic Profile

นางสาวนิตยา สิงไทยสงค์

การศึกษาค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต  
แขนงวิชาเทคโนโลยีสารสนเทศและการสื่อสาร สาขาวิชาวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยสุโขทัยธรรมสาร

พ.ศ. 2557

**Development of Best Practice and a Prototype of Software Development Tracking  
System and Software Quality Control System Based on ISO/IEC 29110  
Standard, Basic Profile**

**Miss Nittaya Singthaisong**

A Independent Study Submitted in Partial Fulfillment of the Requirements for  
the Degree of Master of Science in Information and Communication Technology

School of Science and Technology  
Sukhothai Thammathirat Open University

2014

หัวข้อการศึกษาค้นคว้าอิสระ การพัฒนาแนวปฏิบัติที่ดีและต้นแบบระบบตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

ชื่อและนามสกุล นางสาวนิตยา สิงไทยสงค์

แขนงวิชา เทคโนโลยีสารสนเทศและการสื่อสาร

สาขาวิชา วิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยสุโขทัยธรรมาธิราช

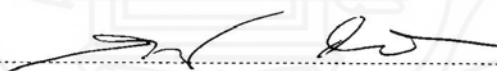
อาจารย์ที่ปรึกษา รองศาสตราจารย์ ดร. วิภา เจริญภัณฑารักษ์

การศึกษาค้นคว้าอิสระนี้ ได้รับความเห็นชอบให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรระดับปริญญาโท เมื่อวันที่ 28 กุมภาพันธ์ 2558

คณะกรรมการสอบการศึกษาค้นคว้าอิสระ



..... ประธานกรรมการ  
(รองศาสตราจารย์ ดร.วิภา เจริญภัณฑารักษ์)



..... กรรมการ  
(อาจารย์ ดร.สันติพัฒน์ อรุณธารี)



.....  
(รองศาสตราจารย์ ัญญพร พิมพาน)

ประธานกรรมการประจำสาขาวิชาวิทยาศาสตร์และเทคโนโลยี

**ชื่อการศึกษา** **ค้นคว้าอิสระ** การพัฒนาแนวปฏิบัติที่ดีและต้นแบบระบบตรวจสอบการพัฒนาซอฟต์แวร์ และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110  
ระดับ Basic Profile

**ชื่อและนามสกุล** นางสาวนิตยา สิงไทยสงค์ **รหัสนักศึกษา** 2559600321 **ปริญญา** วิทยาศาสตร์มหาบัณฑิต  
(เทคโนโลยีสารสนเทศและการสื่อสาร) **อาจารย์ที่ปรึกษา** รองศาสตราจารย์ ดร. วิภา เจริญกัณฑ์วารักษ์  
**ปีการศึกษา** 2557

### บทคัดย่อ

ในการพัฒนาซอฟต์แวร์จำเป็นต้องมีกระบวนการติดตามและควบคุมคุณภาพของซอฟต์แวร์ ดังนั้นมาตรฐาน ISO/IEC 29110 จึงเป็นเครื่องมือที่สำคัญในการดำเนินโครงการพัฒนาซอฟต์แวร์เพื่อให้มีคุณภาพ โดยโครงการนี้มีวัตถุประสงค์ คือ (1) เพื่อพัฒนาแนวปฏิบัติที่ดีด้านการพัฒนาซอฟต์แวร์ในการรองรับการตรวจสอบและควบคุมคุณภาพตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับองค์กรที่เป็นผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยที่มีบุคลากรด้านการพัฒนาซอฟต์แวร์ไม่เกิน 25 คน (2) เพื่อพัฒนาต้นแบบระบบตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามแนวมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

การดำเนินงานประกอบด้วย (1) ศึกษาเกี่ยวกับคุณลักษณะของระบบมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile (2) รวบรวมข้อมูลจากองค์กรตัวอย่างซึ่งเป็นผู้ให้บริการพัฒนาซอฟต์แวร์ที่ผ่านการประเมินและได้รับการรับรองมาตรฐาน ISO/IEC 29110(3) พัฒนาแนวปฏิบัติที่ดีด้านการตรวจสอบและควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับองค์กรผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทย (4) พัฒนาต้นแบบระบบตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile (5) ประเมินผลการพัฒนาแนวปฏิบัติที่ดีและต้นแบบระบบ ฯ ที่พัฒนาขึ้น (6) สรุปผลที่ได้จากการดำเนินงานและจัดทำข้อเสนอแนะ

ผลที่ได้รับจากการดำเนินงานคือ (1) แนวปฏิบัติที่ดีด้านการตรวจสอบการพัฒนาซอฟต์แวร์และควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile ในรูปแบบ Checklist สำหรับการประเมินตนเอง (Self-Assessment) ตามกรอบมาตรฐาน ISO/IEC 29110 เพื่อจะเป็นประโยชน์สำหรับองค์กรผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยหรือผู้ที่สนใจจะสามารถใช้แนวปฏิบัติฯ ที่จัดทำขึ้นเป็นแนวทางในการปรับปรุงกระบวนการดำเนินงานด้านการตรวจสอบการพัฒนาซอฟต์แวร์และควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ได้อย่างมีประสิทธิภาพและมีประสิทธิผลยิ่งขึ้น (2) ต้นแบบระบบตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ที่พัฒนาขึ้นนั้น องค์กรผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยสามารถนำไปใช้เป็นเครื่องมือสำหรับควบคุมคุณภาพในการดำเนินงานพัฒนาซอฟต์แวร์ให้ได้ตามมาตรฐาน อีกทั้งยังเป็นการตอบสนองนโยบายของสภาอุตสาหกรรมแห่งประเทศไทยในการยกระดับคุณภาพกระบวนการพัฒนาซอฟต์แวร์ของผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยให้มีมาตรฐานสู่ระดับสากล

**คำสำคัญ** ISO/IEC 29110 ระบบตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ ผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทย



**Independent Study title:** Development of Best Practice and a Prototype of Software Development Tracking System and Software Quality Control System Based on ISO/IEC 29110 Standard, Basic Profile

**Author:** Miss Nittaya Singthaisong; **ID:** 2559600321; **Degree:** Master of Science (Information and Communication Technology);

**Independent Study advisor:** Dr. Vipa Jaroenpuntaruk, Associate Professor;

**Academic year:** 2014

### Abstract

In software development processes, tracking system and software quality control are major concerns to achieve software development quality. Therefore, ISO/IEC 29110 is an important tool for software development processes improvement. Objectives of this project were as follows: (1) To develop the best practice guidelines for software development tracking and software quality control based on ISO/IEC 29110 standard, Basic Profile for Small and Medium Enterprise (SME) of Thai software firm, which had no more than 25 staffs in the firm (2) To develop a prototype of software development tracking system and software quality control system based on ISO/IEC 29110 standard, Basic Profile.

The project methodology consisted of the followings: (1) To review the literature of ISO/IEC 29110 standard, Basic Profile (2) To gather information from executive of a company in Thailand that had been assessed and certified ISO/IEC 29110 standard, Basic Profile (3) To develop best practice guidelines for software development tracking and software quality control based on ISO/IEC 29110 standard, Basic Profile for SME of Thai software firm (4) To develop a prototype of software development tracking system and software quality control system based on ISO/IEC 29110 standard, Basic Profile (5) To evaluate the developed best practice guidelines and the prototype of software development tracking system and software quality control system based on ISO/IEC 29110 standard, Basic Profile from the SME of Thai software firm that had been assessed and certified ISO/IEC 29110 standard, basic profile (6) To summarize result of the project and gave recommendations.

The results of the project were: (1) Best practice guidelines for software development tracking and software quality control based on ISO/IEC 29110 standard, Basic Profile as checklist for self-assessment for SME of Thai software firm to improve their software development processes and quality based on ISO/IEC 29110 standard more efficiently and effectively (2) A prototype of software development tracking system and software quality control system based on ISO/IEC 29110 standard, Basic Profile, which could be served as a tool for SME of Thai software firm during software development process to track and control quality of software based on ISO/IEC 29110 Standard. In addition, results of the project will support Thai Software Industry Policy for Thai SME software firm quality to meet global standard.

**Keywords:** ISO/IEC 29110, Software Development Tracking System and Software Quality Control System, SME of Thai Software Firm

## กิตติกรรมประกาศ

รายงานการศึกษาค้นคว้าอิสระนี้สำเร็จได้ด้วยความอนุเคราะห์เป็นอย่างดีจากผู้ทรงคุณวุฒิหลายท่าน โดยเฉพาะอย่างยิ่ง รองศาสตราจารย์ ดร. วิภา เจริญกัญฑ์ทาร์กษ์ อาจารย์ที่ปรึกษาที่สละเวลาและทุ่มเทให้คำแนะนำตลอดการจัดทำรายงานนี้ และอาจารย์ ดร.สันติพัฒน์ อรุณชาติ ที่สละเวลาในการให้คำปรึกษาและข้อเสนอแนะต่าง ๆ เพิ่มเติมเพื่อให้รายงานมีความถูกต้องเหมาะสมอย่างสมบูรณ์ ผู้ศึกษาขอขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ขอขอบพระคุณผู้บริหารบริษัท โพรเฟสชั่นแนล เซอร์วิส โพรไวเดอร์ จำกัด ที่ให้ความกรุณาในการให้สัมภาษณ์ รวมทั้งให้ความอนุเคราะห์ข้อมูลต่าง ๆ เพื่อใช้ในการศึกษาค้นคว้าครั้งนี้

นอกจากนี้ผู้ศึกษาขอขอบพระคุณคณาจารย์ประจำสาขาวิชาวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยสุโขทัยธรรมราชา ที่ได้มอบความรู้และให้คำปรึกษาด้วยดีตลอดมา รวมถึงเพื่อนนักศึกษาที่คอยให้กำลังใจจนกระทั่งรายงานฉบับนี้เสร็จสมบูรณ์ด้วยดี และสิ่งที่สำคัญอย่างยิ่ง ผู้ศึกษาขอขอบพระคุณ บิดา มารดา และครอบครัว ที่ให้กำลังใจและให้ความห่วงใยตลอดมา

นิตยา สิงไทยสงค์

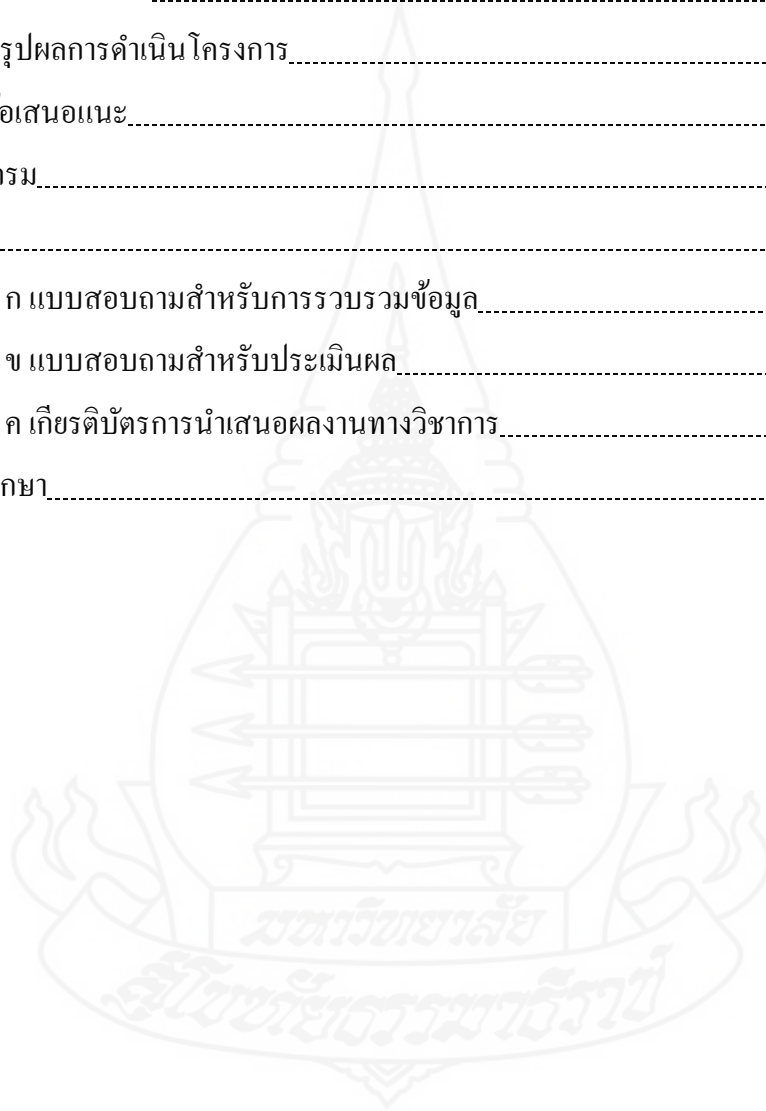
มีนาคม 2558

## สารบัญ

	หน้า
บทคัดย่อภาษาไทย .....	ง
บทคัดย่อภาษาอังกฤษ .....	จ
กิตติกรรมประกาศ .....	ฉ
สารบัญตาราง .....	ฅ
สารบัญภาพ.....	ฉ
บทที่ 1 บทนำ .....	1
ความเป็นมาและความสำคัญของปัญหา.....	1
วัตถุประสงค์.....	3
ขอบเขตของการดำเนินโครงการ.....	3
ขั้นตอนการดำเนินโครงการ.....	4
ประโยชน์ที่ได้จากการวิจัย.....	5
นิยามคำศัพท์.....	5
บทที่ 2 ทฤษฎี และงานวิจัยที่เกี่ยวข้อง.....	9
ความสำคัญของซอฟต์แวร์, กระบวนการพัฒนาซอฟต์แวร์, และมาตรฐานการพัฒนา ซอฟต์แวร์.....	9
ประวัติและความเป็นมาของระบบมาตรฐาน ISO/IEC 29110.....	12
ความรู้เบื้องต้นเกี่ยวกับมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile.....	20
บทที่ 3 วิธีการดำเนินโครงการ.....	27
ขั้นตอนการดำเนินโครงการ.....	27
วิธีการดำเนินโครงการ.....	30
บทที่ 4 ผลที่ได้จากการดำเนินโครงการ.....	33
ข้อมูลเบื้องต้นของบริษัทตัวอย่าง.....	33
ผลจากการศึกษา, บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC29110 ระดับ Basic Profile สำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทย.....	38
ผลลัพธ์จากการทำเหมืองข้อมูล.....	58

## สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปผลและข้อเสนอแนะ.....	174
การประเมินผล.....	174
สรุปผลการดำเนินโครงการ.....	175
ข้อเสนอแนะ.....	189
บรรณานุกรม.....	191
ภาคผนวก.....	193
ก แบบสอบถามสำหรับการรวบรวมข้อมูล.....	194
ข แบบสอบถามสำหรับประเมินผล.....	208
ค เกียรติบัตรการนำเสนอผลงานทางวิชาการ.....	211
ประวัติผู้ศึกษา.....	213



สารบัญตาราง

	หน้า
ตารางที่ 2.1	รายละเอียดกลุ่มทำงาน Sub-committee 7 (SC7)..... 13
ตารางที่ 2.2	รายละเอียดกลุ่มทำงาน Joint Technical Committee 1 (JTC1)..... 14
ตารางที่ 2.3	รายละเอียด The International Organization for Standardization..... 15
ตารางที่ 2.4	สมาชิก The International Organization for Standardization เต็มรูปแบบ..... 16
ตารางที่ 2.5	สมาชิก The International Organization for Standardization ที่มีการร่วมมือกัน... 18
ตารางที่ 2.6	ชุดองค์ประกอบของ ISO/IEC 29110..... 19
ตารางที่ 4.1	แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity Planning (การวางแผนโครงการ)..... 40
ตารางที่ 4.2	แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity PM.2: Project Plan Execution (การปฏิบัติตามแผนโครงการ)..... 60
ตารางที่ 4.3	แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity PM.3: Project Assessment and Control (การประเมินและควบคุมโครงการ)..... 73
ตารางที่ 4.4	แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity PM.4: Project Closure (การส่งมอบโครงการ)..... 77
ตารางที่ 4.5	แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.1 Software Implementation Initiation (การเริ่มการพัฒนา/ติดตั้งซอฟต์แวร์)... 83
ตารางที่ 4.6	แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.2 Software Requirements Analysis (การวิเคราะห์ความต้องการซอฟต์แวร์)... 86
ตารางที่ 4.7	แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้างสถาปัตยกรรมและรายละเอียดซอฟต์แวร์)..... 97
ตารางที่ 4.8	แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.4: Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์)..... 122

## สารบัญตาราง (ต่อ)

	หน้า
ตารางที่ 4.9 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.5 Software Integration and Tests (การทดสอบและการบูรณาการซอฟต์แวร์) 131	
ตารางที่ 4.10 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ซอฟต์แวร์).....	162
ตารางที่ 5.1 GAP Analysis ก่อน-หลังการปรับปรุงกระบวนการดำเนินงานโครงการพัฒนาซอฟต์แวร์ ตามมาตรฐาน ISO/IEC 29110 ของบริษัทตัวอย่าง.....	180



## สารบัญภาพ

	หน้า
ภาพที่ 2.1	กระบวนการภาพรวมของมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile..... 21
ภาพที่ 2.2	4Activities ของ Project Management (PM) Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์)..... 23
ภาพที่ 2.3	6Activities ของ Software Implementation (SI) Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์)..... 26
ภาพที่ 3.1	Standard Software Development Life Cycle..... 28
ภาพที่ 4.1	โครงสร้างองค์กรของบริษัท โปรเฟสชันแนล เซอร์วิส โพรไวเดอร์ จำกัด..... 34
ภาพที่ 4.2	โครงสร้างองค์กรของหน่วยงาน Business Software Development Service ของบริษัท โปรเฟสชันแนล เซอร์วิส โพรไวเดอร์ จำกัด..... 34
ภาพที่ 4.3	ตัวอย่างเอกสาร Project Plan ส่วนที่ 1..... 44
ภาพที่ 4.4	ตัวอย่างเอกสาร Project Plan ส่วนที่ 2..... 44
ภาพที่ 4.5	ตัวอย่างเอกสาร Project Plan ส่วนที่ 3..... 45
ภาพที่ 4.6	ตัวอย่างเอกสาร Project Plan ส่วนที่ 4..... 46
ภาพที่ 4.7	ตัวอย่างเอกสาร Project Plan ส่วนที่ 5..... 47
ภาพที่ 4.8	ตัวอย่างเอกสาร Project Plan ส่วนที่ 6..... 48
ภาพที่ 4.9	ตัวอย่างเอกสาร Project Plan ส่วนที่ 7..... 49
ภาพที่ 4.10	ตัวอย่างเอกสาร Project Plan ส่วนที่ 8..... 50
ภาพที่ 4.11	ตัวอย่างเอกสาร Project Plan ส่วนที่ 9..... 50
ภาพที่ 4.12	ตัวอย่างเอกสาร Project Plan ส่วนที่ 10..... 51
ภาพที่ 4.13	ตัวอย่างเอกสาร Project Plan ส่วนที่ 11..... 52
ภาพที่ 4.14	ตัวอย่างเอกสาร Project Plan ส่วนที่ 12..... 53
ภาพที่ 4.15	ตัวอย่างเอกสาร Project Plan ส่วนที่ 13..... 54
ภาพที่ 4.16	ตัวอย่างเอกสาร Verification & Validation Results ส่วนที่ 1..... 55
ภาพที่ 4.17	ตัวอย่างเอกสาร Verification & Validation Results ส่วนที่ 2..... 55
ภาพที่ 4.18	ตัวอย่างเอกสาร Meeting Record ส่วนที่ 1..... 56

สารบัญภาพ (ต่อ)

	หน้า
ภาพที่ 4.19 ตัวอย่างเอกสาร Meeting Record ส่วนที่ 2.....	56
ภาพที่ 4.20 ตัวอย่างเอกสาร Project Timeframe (Project Status Record) ส่วนที่ 1.....	67
ภาพที่ 4.21 ตัวอย่างเอกสาร Project Timeframe (Project Status Record) ส่วนที่ 2.....	68
ภาพที่ 4.22 ตัวอย่างเอกสาร Project Timeframe (Project Status Record) ส่วนที่ 3.....	68
ภาพที่ 4.23 ตัวอย่างเอกสาร Change Request ส่วนที่ 1.....	69
ภาพที่ 4.24 ตัวอย่างเอกสาร Change Request ส่วนที่ 2.....	69
ภาพที่ 4.25 ตัวอย่างเอกสาร Change Request ส่วนที่ 3.....	70
ภาพที่ 4.26 ตัวอย่างเอกสาร Correction Register ส่วนที่ 1.....	70
ภาพที่ 4.27 ตัวอย่างเอกสาร Correction Register ส่วนที่ 2.....	71
ภาพที่ 4.2 ตัวอย่างเอกสารจดหมายส่งมอบปิดโครงการ (Acceptance Record) ส่วนที่ 1.....	81
ภาพที่ 4.29 ตัวอย่างเอกสารจดหมายส่งมอบปิดโครงการ (Acceptance Record) ส่วนที่ 2.....	81
ภาพที่ 4.30 ตัวอย่างเอกสาร Software Requirements Specification ส่วนที่ 1.....	90
ภาพที่ 4.31 ตัวอย่างเอกสาร Software Requirements Specification ส่วนที่ 2.....	91
ภาพที่ 4.32 ตัวอย่างเอกสาร Software Requirements Specification ส่วนที่ 3.....	91
ภาพที่ 4.33 ตัวอย่างเอกสาร Software Requirements Specification ส่วนที่ 4.....	92
ภาพที่ 4.34 ตัวอย่างเอกสาร Software Requirements Specification ส่วนที่ 5.....	92
ภาพที่ 4.35 ตัวอย่างเอกสาร Software Design ส่วนที่ 1.....	111
ภาพที่ 4.36 ตัวอย่างเอกสาร Software Design ส่วนที่ 2.....	112
ภาพที่ 4.37 ตัวอย่างเอกสาร Software Design ส่วนที่ 3.....	112
ภาพที่ 4.38 ตัวอย่างเอกสารย่อย Program Structure ในชุดเอกสาร Software Design ส่วนที่ 1.....	113
ภาพที่ 4.39 ตัวอย่างเอกสารย่อย Program Structure ในชุดเอกสาร Software Design ส่วนที่ 2.....	114
ภาพที่ 4.40 ตัวอย่างเอกสาร Traceability Record ส่วนที่ 1.....	115
ภาพที่ 4.41 ตัวอย่างเอกสาร Traceability Record ส่วนที่ 2.....	116

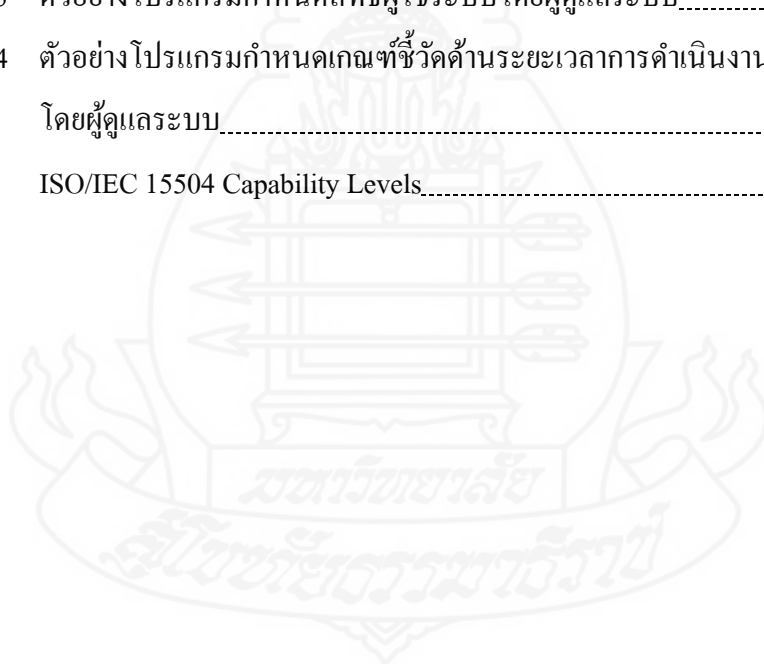


## สารบัญภาพ (ต่อ)

	หน้า
ภาพที่ 4.42 ตัวอย่างเอกสาร Test Cases and Test Procedures ส่วนที่ 1.....	117
ภาพที่ 4.43 ตัวอย่างเอกสาร Test Cases and Test Procedures ส่วนที่ 2.....	118
ภาพที่ 4.44 ตัวอย่างเอกสาร Test Report ส่วนที่ 1.....	118
ภาพที่ 4.45 ตัวอย่างเอกสาร Test Report ส่วนที่ 2 .....	119
ภาพที่ 4.46 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 1.....	150
ภาพที่ 4.47 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 2.....	151
ภาพที่ 4.48 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 3.....	152
ภาพที่ 4.49 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 4.....	153
ภาพที่ 4.50 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 5.....	154
ภาพที่ 4.51 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 6.....	155
ภาพที่ 4.52 ตัวอย่างเอกสาร User Manual (Software User Documentation) ส่วนที่ 1.....	156
ภาพที่ 4.53 ตัวอย่างเอกสาร User Manual (Software User Documentation) ส่วนที่ 2.....	157
ภาพที่ 4.54 ตัวอย่างเอกสาร User Manual (Software User Documentation) ส่วนที่ 3.....	158
ภาพที่ 4.55 ตัวอย่างเอกสาร User Manual (Software User Documentation) ส่วนที่ 4.....	159
ภาพที่ 4.56 ต้นแบบหน้าจอสำหรับ Login เข้าระบบเพื่อการพิสูจน์ตัวตนและรักษาความลับ.	166
ภาพที่ 4.57 ต้นแบบหน้าจอสำหรับการติดตามสถานะโครงการ (Project Status Tracking)....	167
ภาพที่ 4.58 ตัวอย่างฟอร์มบันทึกข้อมูลโครงการแยกตามความรับผิดชอบโดย Project Manager.....	168

## สารบัญภาพ (ต่อ)

	หน้า
ภาพที่ 4.59 ตัวอย่างฟอร์มบันทึกข้อมูลโครงการแยกตามความรับผิดชอบโดย Team Leader/System Analyst ส่วนที่ 1.....	169
ภาพที่ 4.60 ตัวอย่างฟอร์มบันทึกข้อมูลโครงการแยกตามความรับผิดชอบโดย Team Leader/System Analyst ส่วนที่ 2.....	170
ภาพที่ 4.61 ตัวอย่างฟอร์มบันทึกข้อมูลโครงการแยกตามความรับผิดชอบโดย Programmer ส่วนที่ 1.....	171
ภาพที่ 4.62 ตัวอย่างฟอร์มบันทึกข้อมูลโครงการแยกตามความรับผิดชอบโดย Programmer ส่วนที่ 2.....	172
ภาพที่ 4.63 ตัวอย่างโปรแกรมกำหนดสิทธิ์ผู้ใช้ระบบโดยผู้ดูแลระบบ.....	173
ภาพที่ 4.64 ตัวอย่างโปรแกรมกำหนดเกณฑ์จัดด้านระยะเวลาการดำเนินงานแต่ละกิจกรรม โดยผู้ดูแลระบบ.....	173
ภาพที่ 5.1 ISO/IEC 15504 Capability Levels.....	175



# บทที่ 1

## บทนำ

### 1. ความเป็นมาและความสำคัญของปัญหา

จากการที่ในปัจจุบันองค์กรต่างๆ ได้เล็งเห็นถึงประโยชน์ในการนำซอฟต์แวร์เข้ามาประยุกต์ใช้เพื่อปรับปรุงกระบวนการดำเนินงานทางธุรกิจให้มีประสิทธิภาพดียิ่งขึ้นอย่างเป็นที่แพร่หลาย ไม่ว่าจะเป็นการจัดซื้อซอฟต์แวร์สำเร็จรูป หรือการว่าจ้างบริการการพัฒนาซอฟต์แวร์ตามความต้องการทางธุรกิจของแต่ละองค์กรก็ตาม

ขณะเดียวกันผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยนับว่ามีส่วนขับเคลื่อนอุตสาหกรรมซอฟต์แวร์ในประเทศเป็นอย่างมาก จากการที่จำนวนสมาชิกของสมาคมอุตสาหกรรมซอฟต์แวร์ไทย (The Association of Thai Software Industry) มีมากขึ้นอย่างต่อเนื่องในปัจจุบัน ขณะที่ในปี 2558 ก็จะเป็นการก้าวสู่ประชาคมอาเซียน ซึ่งเป็นการเปิดเสรีภาคบริการอาเซียน ดังนั้นผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยควรหันมาให้ความสำคัญกับการยกระดับมาตรฐานคุณภาพซอฟต์แวร์และการให้บริการขององค์กรสู่ระดับสากล เพื่อสร้างความเชื่อมั่นด้านการให้บริการพัฒนาซอฟต์แวร์ขององค์กรทั้งภายในและต่างประเทศ ซึ่งจะเป็นการเปิดโอกาสการขยายตลาดการให้บริการสู่ประเทศในกลุ่มสมาชิกอาเซียน

ในส่วนของภาครัฐก็ได้ให้การสนับสนุนอุตสาหกรรมซอฟต์แวร์ในประเทศไทยจากการที่ในปี 2554 กระทรวงไอซีทีได้จับมือกับซีป้า (สำนักงานส่งเสริมอุตสาหกรรมซอฟต์แวร์แห่งชาติ (องค์การมหาชน)), สมอ. (สำนักงานมาตรฐานผลิตภัณฑ์อุตสาหกรรม), และมูลนิธิสถาบันเพื่อพัฒนานวัตกรรม เพื่อพัฒนาศักยภาพการแข่งขันซอฟต์แวร์ไทย สำหรับองค์กรขนาดเล็กสู่มาตรฐานระดับโลก ISO/IEC 29110 โดยมีนโยบายให้การสนับสนุนผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยให้ก้าวสู่มาตรฐานระดับสากล โดยมีเป้าหมายให้ประเทศไทยเป็นประเทศแรกในระดับภูมิภาคที่มีผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยผ่านการรับรองมาตรฐานใหม่ เพื่อตอกย้ำความเชื่อมั่นอุตสาหกรรมซอฟต์แวร์ไทยกับต่างชาติ

จากความเป็นมาข้างต้นประกอบกับการที่องค์กรที่ไม่มีบุคลากรมีความรู้ด้านมาตรฐาน ISO หรือองค์กรที่มีแต่บุคลากรทำงานเฉพาะด้าน Programmer เป็นส่วนมากซึ่งไม่มีความถนัดในงานด้านเอกสารอาจจะเป็นการยากในการที่จะเริ่มปรับปรุงกระบวนการดำเนินงานให้เป็นไปตาม

มาตรฐานได้ ดังนั้นผู้ดำเนินโครงการจึงมีแนวคิดที่จะศึกษาคุณลักษณะของมาตรฐาน ISO/IEC 29110 และพัฒนาแนวปฏิบัติตามมาตรฐาน ISO/IEC 29110 สำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยเพื่อให้เป็นการง่ายต่อการเริ่มต้นปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 สำหรับองค์กรพัฒนาซอฟต์แวร์ที่เป็น SME เพื่อจะสามารถใช้รายงานนี้เป็นแนวทางในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์เพื่อการขอรับรองมาตรฐาน ISO/IEC 29110 ได้อย่างสะดวกและรวดเร็วยิ่งขึ้นพร้อมกันนี้ยังสามารถใช้ข้อมูลต้นแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ในการพัฒนาเป็นเครื่องมือสำหรับใช้ในการควบคุมกระบวนการทำงานขององค์กรได้อีกด้วย

รายงานนี้เป็นการศึกษาและพัฒนาแนวปฏิบัติตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile เนื่องจากองค์กรตัวอย่างได้รับการสนับสนุนให้ผ่านการรับรอง ISO/IEC 29110 ระดับ Basic Profile จากทางซีป้า (สำนักงานส่งเสริมอุตสาหกรรมซอฟต์แวร์แห่งชาติ (องค์การมหาชน)) และสภาอุตสาหกรรมแห่งประเทศไทยและโดยเหตุที่เมื่อองค์กรตัวอย่างได้ผ่านการรับรอง ISO/IEC 29110 ระดับ Basic Profile แล้ว แต่ยังไม่มีการจัดทำข้อมูล Checklist สำหรับใช้ควบคุมการดำเนินงาน จึงเป็นอีกเหตุผลหนึ่งที่ทำให้ผู้ดำเนินงานมีแนวคิดในการพัฒนาแนวปฏิบัติตามมาตรฐาน ISO/IEC 29110 โดยการดำเนินงานครั้งนี้ผู้ดำเนินงานได้จัดทำรายงานขึ้นจากมุมมองของการเป็นส่วนหนึ่งในองค์กรที่ได้รับการประเมินและผ่านการรับรอง ISO/IEC 29110 รวมทั้งได้เคยผ่านการเข้าอบรมการประเมินมาตรฐาน ISO/IEC 29110 แล้ว

มาตรฐาน ISO/IEC 29110 เป็นมาตรฐานระหว่างประเทศทางด้านวิศวกรรมซอฟต์แวร์ ที่องค์การระหว่างประเทศว่าด้วยการมาตรฐานหรือ ISO ได้ประกาศใช้เมื่อวันที่ 1 กันยายน 2554 เป็นมาตรฐานฉบับแรกที่มีมุ่งเน้นให้ผู้พัฒนาซอฟต์แวร์ขนาดเล็ก นำไปใช้เพื่อยกระดับคุณภาพผลผลิตด้านซอฟต์แวร์และบริการให้เป็นที่ยอมรับในระดับสากลและมีวงจรในการพัฒนาปรับปรุงอย่างต่อเนื่อง ซึ่งเป็นมาตรฐานที่มีความซับซ้อนและคุณลักษณะต่างๆ น้อยกว่ามาตรฐาน CMM จึงส่งผลให้การปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 สามารถใช้ต้นทุนและระยะเวลาในการดำเนินการให้สำเร็จได้ น้อยกว่าการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐานมาตรฐาน CMM โดยมาตรฐาน ISO/IEC 29110 นั้นครอบคลุมกระบวนการหลัก 2 กระบวนการ ดังนี้

1. Project Management Process: เป็นกระบวนการที่มีวัตถุประสงค์ให้สามารถบรรลุถึงการดำเนินงานอย่างเป็นระบบในด้านการบริหาร โครงการพัฒนาซอฟต์แวร์ ซึ่งจะต้องสามารถบริหาร โครงการให้สำเร็จได้ตามขอบข่ายงาน, คุณภาพ, ระยะเวลา, และค่าใช้จ่ายที่ได้วางแผนไว้

2. Software Implementation Process: กระบวนการที่มีวัตถุประสงค์ให้สามารถบรรลุถึงการดำเนินงานอย่างเป็นระบบในด้านการวิเคราะห์, การออกแบบ, การพัฒนา, การทดสอบ ซอฟต์แวร์, และการจัดทำเอกสารคู่มือซอฟต์แวร์ให้ถูกต้องตรงตามความต้องการที่ได้กำหนดไว้

## 2. วัตถุประสงค์

การดำเนิน โครงการครั้งนี้เป็นการศึกษาเกี่ยวกับระบบมาตรฐาน ISO/IEC 29110 ด้านการพัฒนาซอฟต์แวร์สำหรับทีมพัฒนาซอฟต์แวร์ไม่เกิน 25 คน (A Very Small Entity (VSE) is an enterprise, organization, department or project having up to 25 people) ระดับ Basic Profile เพื่อเป็นข้อมูลสำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยจะสามารถใช้เป็นแนวทางในการพัฒนากระบวนการพัฒนาซอฟต์แวร์อย่างมีคุณภาพได้สะดวกและรวดเร็วยิ่งขึ้นเพื่อให้ได้มาซึ่ง วัตถุประสงค์ของการดำเนิน โครงการในครั้งนี้คือ

1. เพื่อพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile เพื่อเป็นข้อมูลแนวทางสำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยในการปรับปรุง กระบวนการดำเนินงานด้านการตรวจสอบและควบคุมคุณภาพการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110

2. เพื่อพัฒนาค้นแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุม คุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับผู้ที่มีความสนใจจะ สามารถใช้เป็นข้อมูลแนวทางการพัฒนาเป็นระบบสำหรับใช้ตรวจสอบและควบคุม โครงการพัฒนาซอฟต์แวร์ขององค์กรให้เป็นไปตามมาตรฐาน 29110

## 3. ขอบเขตของการดำเนินโครงการ

1. ศึกษารายละเอียดคุณลักษณะของมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile ด้านการพัฒนาซอฟต์แวร์สำหรับทีมพัฒนาซอฟต์แวร์ไม่เกิน 25 คน (A Very Small Entity (VSE) is an enterprise, organization, department or project having up to 25 people)

2. ศึกษากระบวนการพัฒนาซอฟต์แวร์จากข้อมูลการดำเนิน โครงการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 โดยการสัมภาษณ์องค์กรผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยที่ผ่านการรับรองมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile และรวบรวมข้อมูลความต้องการระบบสำหรับใช้ในการตรวจสอบและควบคุมคุณภาพการปฏิบัติตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

3. พัฒนาแนวปฏิบัติที่ดีอ้างอิงตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทย

4. พัฒนาค้นแบบระบบตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

#### 4. ขั้นตอนการดำเนินโครงการ

4.1 ศึกษารายละเอียดคุณลักษณะของมาตรฐาน ISO/IEC 29110 ด้านการพัฒนาซอฟต์แวร์สำหรับทีมพัฒนาซอฟต์แวร์ไม่เกิน 25 คน (A Very Small Entity (VSE) is an enterprise, organization, department or project having up to 25 people)ระดับ Basic Profile

4.2 รวบรวมข้อมูลการดำเนิน โครงการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile และรวบรวมข้อมูลความต้องการระบบสำหรับใช้ในการตรวจสอบและควบคุมคุณภาพการปฏิบัติตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile จากผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยที่ผ่านการรับรองมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile โดยการกำหนดคำถามและออกแบบแบบสอบถามสำหรับการสัมภาษณ์

4.3 พัฒนาแนวปฏิบัติที่ดีอ้างอิงตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทย

4.4 พัฒนาค้นแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

4.5 ประเมินผลการพัฒนาแนวปฏิบัติที่ดีและต้นแบบระบบ โดยนำแนวปฏิบัติที่ดีและต้นแบบระบบที่พัฒนาขึ้นไปนำเสนอต่อองค์กรผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยที่ผ่านการรับรองมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile แล้วทำการสรุปข้อคิดเห็นและข้อเสนอแนะเพิ่มเติม

4.6 สรุปผลที่ได้จากการดำเนินงานและศึกษาข้อมูล เสนอแนะ

4.7 จัดทำรายงานการศึกษาค้นคว้าอิสระ



## 5. ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่คาดว่าจะได้รับจากการดำเนิน โครงการนี้แบ่งเป็น 3 ส่วนดังนี้

### 5.1 ผู้ดำเนินโครงการ

5.1.1 ได้รับความรู้เกี่ยวกับรายละเอียดคุณลักษณะของมาตรฐาน ISO/IEC 29110

5.1.2 ได้รับความรู้ถึงแนวทางในการนำรายละเอียดคุณลักษณะของมาตรฐาน ISO/IEC 29110 มาประยุกต์ใช้ในการดำเนิน โครงการพัฒนาซอฟต์แวร์

### 5.2 องค์กรผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทย

5.2.1 องค์กรพัฒนาซอฟต์แวร์ที่ยังไม่ได้ปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ และยังไม่ได้รับการรับรองมาตรฐาน ISO/IEC 29110 สามารถนำผลงานนี้มาใช้เป็นแนวทางในการปรับปรุงการดำเนิน โครงการพัฒนาซอฟต์แวร์เพื่อการตรวจสอบและควบคุมคุณภาพการพัฒนาซอฟต์แวร์ขององค์กรได้อย่างสะดวกและรวดเร็วยิ่งขึ้น

5.2.2 องค์กรที่มีความสนใจในการพัฒนาระบบสำหรับใช้ประกอบการตรวจสอบ และควบคุมคุณภาพการดำเนิน โครงการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 สามารถนำข้อมูลต้นแบบระบบตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตาม มาตรฐาน ISO/IEC 29110 ในรายงานนี้ ไปใช้เป็นข้อมูลแนวทางประกอบการพัฒนาเป็นระบบ สำหรับใช้เป็นเครื่องมือในการตรวจสอบและควบคุมโครงการพัฒนาซอฟต์แวร์ขององค์กรให้ เป็นไปตามมาตรฐาน ISO/IEC 29110 ได้

### 5.3 วงการการศึกษา

5.3.1 เป็นการนำความรู้เชิงทฤษฎีมาประยุกต์ให้เป็นกรอบแนวทางการปฏิบัติงาน ได้จริง

### 5.4 ประเทศไทย

5.4.1 ช่วยส่งเสริมให้องค์กรผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยได้รับการ ขอมรับในระดับสากลได้อย่างสะดวกและรวดเร็วยิ่งขึ้น

## 6. นิยามศัพท์เฉพาะ

6.1 มาตรฐาน ISO/IEC 29110 ระดับ **Basic Profile** หมายถึง มาตรฐานด้านการ พัฒนาซอฟต์แวร์ระดับพื้นฐาน สำหรับองค์กรขนาดเล็กหรือองค์กรขนาดเล็กภายใต้องค์กรขนาด

ใหญ่ก็ได้ ซึ่งมีจำนวนนักพัฒนาซอฟต์แวร์ไม่เกิน 25 คน เป็นมาตรฐานที่ถูกกำหนดให้อยู่ในกลุ่ม Life Cycle Model

**6.2 System** หมายถึง ระบบหรือโปรแกรมซอฟต์แวร์ที่พัฒนาขึ้น

**6.3 Process** หมายถึง กระบวนการที่ประกอบด้วยชุดของกิจกรรมต่างๆ ที่มีความสัมพันธ์หรือมีผลกระทบซึ่งกัน และกันซึ่งทำหน้าที่แปลงข้อมูลนำเข้าไปสู่ข้อมูลผลลัพธ์ (ในที่นี้หมายถึง โปรแกรมซอฟต์แวร์ที่พัฒนาขึ้น) ตามที่ต้องการ

**6.4 Activity** หมายถึง ชุดของรายการดำเนินงาน (Task) ต่างๆ ของ Process

**6.5 Task** หมายถึง รายการดำเนินงานแต่ละงาน ที่กำหนดความต้องการในการกระทำอย่างชัดเจน

**6.6 Verification** หมายถึง การยืนยันตามแนวทางแห่งข้อกำหนดของวัตถุประสงค์ในการพัฒนาระบบที่มีหลักฐานตรวจสอบได้เพื่อทำให้ตรงกับความต้องการระบบที่ได้กำหนดไว้

**หมายเหตุ:** Verification ในวงจรการพัฒนาระบบ คือ ชุดของกิจกรรมซึ่งเป็นการเปรียบเทียบระบบที่อยู่ระหว่างวงจรการพัฒนา กับคุณลักษณะของมาตรฐานที่ต้องการ

**6.7 Validation** หมายถึง การตรวจสอบและการรับรองผลการตรวจสอบตามแนวทางแห่งข้อกำหนดของวัตถุประสงค์ในการพัฒนาระบบที่มีหลักฐานตรวจสอบได้ เพื่อให้คุณลักษณะของระบบที่พัฒนาขึ้นนั้นตรงกับความต้องการใช้งานตามความต้องการระบบ/ซอฟต์แวร์

**หมายเหตุ:** Validation ในวงจรการพัฒนาระบบ คือ ชุดของกิจกรรมเพื่อทำให้มั่นใจและไว้วางใจได้ว่า ระบบที่พัฒนาขึ้นจะสามารถสำเร็จได้ตรงตามความต้องการใช้งานที่กำหนดไว้, มีคุณลักษณะเป็นไปตามความมุ่งหวังและวัตถุประสงค์ที่กำหนดไว้

**6.8 Requirements analysis** หมายถึง กระบวนการการศึกษาความต้องการของผู้ใช้ระบบในด้านที่เกี่ยวข้องกับระบบ, ฮาร์ดแวร์, หรือความต้องการระบบ/ซอฟต์แวร์

**6.9 Requirements document** หมายถึง เอกสารที่ประกอบด้วยคำแนะนำ, ความต้องการ, หรือข้อกำหนด ที่เป็นรายละเอียดความต้องการระบบ/ซอฟต์แวร์

**6.10 Requirements phase** หมายถึง ช่วงเวลาในวงจรการพัฒนาซอฟต์แวร์ที่เป็นการกำหนดและจัดทำเอกสารความต้องการระบบ/ซอฟต์แวร์

**6.11 Software Requirements Specifications (SRS)** หมายถึง ข้อกำหนดเฉพาะของผลิตภัณฑ์ซอฟต์แวร์, โปรแกรม, หรือชุดโปรแกรม ที่แสดงให้เห็นถึงฟังก์ชันที่ชัดเจนในสภาวะแวดล้อมที่เฉพาะเจาะจง SRS อาจได้รับการจัดทำโดยผู้ให้บริการ 1 หรือหลายราย, จัดทำโดยลูกค้าผู้รับบริการ 1 หรือหลายราย หรือจัดทำร่วมกันทั้งสองฝ่ายก็ได้เอกสาร SRS ประกอบด้วยทั้งความต้องการส่วนฟังก์ชัน (Functional Requirement) และความต้องการส่วนที่ไม่ใช่ฟังก์ชัน (Non-



Functional Requirement)SRS สามารถได้รับการจัดทำในรูปแบบ Word File ที่มีเนื้อหาสาระเป็นเอกสาร 1 ชุด แต่ก็สามารถจัดการเก็บลงในฐานข้อมูลหรือจัดทำในรูปแบบ Excel File ก็ได้

**6.12 Requirement** หมายถึง 1. การบรรยายที่แสดงให้เห็นถึงการที่ผลิตภัณฑ์ซอฟต์แวร์หรือกระบวนการทางธุรกิจจะต้องบรรลุถึงการผลิตที่มีคุณสมบัติ และ/หรือ ผลลัพธ์ตามความต้องการระบบ/ซอฟต์แวร์ (ที่มา: IEEE 1220-2005 IEEE Standard for the Application and Management of the Systems Engineering Process) 2. ความต้องการระบบหรือความต้องการซอฟต์แวร์ที่มีการระบุฟังก์ชันซึ่งระบบ/ระบบซอฟต์แวร์ หรือส่วนประกอบของระบบ/ระบบซอฟต์แวร์จะต้องสามารถทำงานได้ตามความต้องการนั้น (ที่มา: ISO/IEC 24765, Systems and Software Engineering Vocabulary) 3. ความต้องการที่มีการระบุฟังก์ชันซึ่งระบบ หรือส่วนประกอบของระบบจะต้องสามารถทำงานได้ตามความต้องการนั้น (ที่มา: ISO/IEC24765)

**6.13 Non Functional Requirement** หมายถึง ความต้องการระบบ/ซอฟต์แวร์ที่อธิบายถึงสิ่งที่ซอฟต์แวร์ไม่ต้องกระทำ แต่ซอฟต์แวร์ควรมีคุณสมบัติอย่างไรเพื่อให้บรรลุความต้องการส่วนนี้ได้ (ที่มา: ISO/IEC24765 ISO/IEC 24765, Systems and Software Engineering Vocabulary. Syn. design constraints, non-functional requirement) ตัวอย่างเช่น ความต้องการด้านประสิทธิภาพความเร็วในการใช้งานซอฟต์แวร์, ความต้องการด้านการเชื่อมต่อกับซอฟต์แวร์อื่น, เงื่อนไขข้อกำหนดการออกแบบซอฟต์แวร์, และคุณลักษณะด้านคุณภาพซอฟต์แวร์บางครั้ง Non-Functional Requirement ก็เป็นสิ่งที่ยากต่อการทดสอบดังนั้นมักจะได้รับการประเมินทางด้านจิตใจในแง่ของความพึงพอใจ (ที่มา: ISO/IEC24765)

**6.14 Baseline** หมายถึง เอกสารแต่ละชนิดหรือระบบที่ได้รับการทบทวนและเห็นชอบร่วมกันแล้วซึ่งมีเนื้อหาหรือคุณลักษณะเพียงพอต่อการใช้เป็นจุดเริ่มต้นในการดำเนินงานได้ อย่างไรก็ตามสามารถเปลี่ยนแปลงภายหลังได้ตามระเบียบการควบคุมการเปลี่ยนแปลงอย่างเป็นทางการ

**6.15 Prototype** หมายถึง แบบจำลองระบบหรือซอฟต์แวร์ชุดทดลอง ที่มีฟังก์ชันตามข้อกำหนดเฉพาะของผลิตภัณฑ์ซอฟต์แวร์, โปรแกรม, หรือชุดโปรแกรมหรือฟังก์ชันบางส่วนของระบบ (ตามเอกสาร SRS)

**6.16 Traceable** หมายถึง การที่ส่วนประกอบต่างๆ ของระบบ สามารถได้รับการตรวจสอบที่มาของความต้องการระบบ หรือที่มาของการพัฒนาระบบได้

**6.17 Traceability matrix** หมายถึง เอกสารที่บันทึกข้อมูลความสัมพันธ์ระหว่างส่วนประกอบต่างๆ ของกระบวนการพัฒนาระบบ

**6.18 PM** หมายถึง กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์ (Project Management Process) โดยใช้ขอบข่ายงานตามความต้องการระบบของลูกค้าเป็นจุดเริ่มต้นในการจัดทำเอกสาร Project Plan และเป็นการควบคุมกระบวนการพัฒนาซอฟต์แวร์ให้เสร็จได้ภายในระยะเวลาที่กำหนดไว้และเป็นซอฟต์แวร์ที่มีคุณลักษณะตรงตามความต้องการใช้งาน

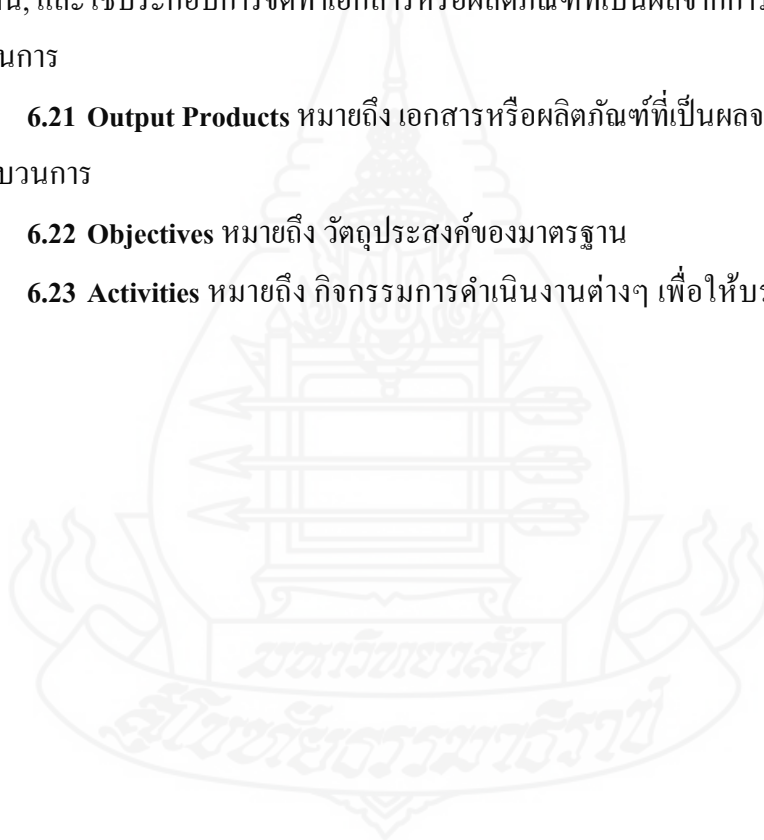
**6.19 SI** หมายถึง กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์ (Software Implementation Process) ซึ่งถูกขับเคลื่อนโดยเอกสารแผนโครงการ (Project Plan) โดยเริ่มตั้งแต่กิจกรรมการวิเคราะห์ความต้องการระบบ, การออกแบบโครงสร้างทางสถาปัตยกรรมและรายละเอียดของระบบ, การพัฒนาระบบ, การบูรณาการและทดสอบระบบ, และการส่งมอบระบบ

**6.20 Input Products** หมายถึง เอกสารหรือผลิตภัณฑ์นำเข้า ที่ใช้สำหรับการทบทวน, วางแผนงาน, และใช้ประกอบการจัดทำเอกสารหรือผลิตภัณฑ์ที่เป็นผลจากการจัดทำขึ้นสำหรับแต่ละกระบวนการ

**6.21 Output Products** หมายถึง เอกสารหรือผลิตภัณฑ์ที่เป็นผลจากการดำเนินงานในแต่ละกระบวนการ

**6.22 Objectives** หมายถึง วัตถุประสงค์ของมาตรฐาน

**6.23 Activities** หมายถึง กิจกรรมการดำเนินงานต่างๆ เพื่อให้บรรลุ Objectives ของมาตรฐาน



## บทที่ 2

### ทฤษฎี และงานวิจัยที่เกี่ยวข้อง

#### 1. ความสำคัญของซอฟต์แวร์, กระบวนการพัฒนาซอฟต์แวร์, และมาตรฐานการพัฒนาซอฟต์แวร์

ในช่วง 2-3 ทศวรรษที่ผ่านมา ซอฟต์แวร์ได้เข้ามามีบทบาทกับชีวิตของมนุษย์มากขึ้น ซอฟต์แวร์ไม่ได้เป็นผลิตภัณฑ์ที่หาซื้อตามอำเภอใจหรือซื้อเพียงเพราะความต้องการชั่วคราว และไม่ได้เป็นเพียงการดำเนินกิจกรรมการพัฒนาโปรแกรมเท่านั้น ผลิตภัณฑ์ซอฟต์แวร์ในปัจจุบันได้รับการจัดการในเชิงวิศวกรรมภายใต้เทคนิคทางกระบวนการที่ได้ถูกเลือกขึ้นเพื่อการปรับปรุงคุณภาพของผลงานการพัฒนาซอฟต์แวร์ที่ทำสำเร็จขึ้น นี่คือข้อสมมุติฐานพื้นฐานในการจัดซื้อได้แข็งและการสนับสนุนประสบการณ์ที่ต้องใช้ความอดทนของนักพัฒนาซอฟต์แวร์ในอดีต เพื่อให้ได้มาซึ่งระเบียบแบบแผนในการที่จะช่วยให้ผลงานการพัฒนาซอฟต์แวร์นั้นถูกพบข้อบกพร่องน้อยลง, ใช้เวลาพัฒนาและส่งมอบซอฟต์แวร์ได้สั้นลง, และซอฟต์แวร์มีคุณค่าเพิ่มขึ้น ความสำคัญของการเลือกแนวปฏิบัติที่เป็นทางการสำหรับการพัฒนาซอฟต์แวร์ คือการกำหนดกฎเกณฑ์ในการส่งมอบซอฟต์แวร์ที่มีคุณภาพที่เหมาะสมเพื่อช่วยให้ธุรกิจสามารถประสบผลสำเร็จได้ และหลีกเลี่ยงการสิ้นเปลืองเวลา, การได้ผลงานที่ไม่มีคุณภาพดีพอ, การโยนความผิดไปให้กับนักพัฒนาซอฟต์แวร์ เป็นต้น

##### 1.1 การพัฒนาซอฟต์แวร์: งานด้านวิศวกรรมที่มีความสำคัญอย่างมาก

การเป็นไปอย่างซ้ำๆ และแน่นอน ก็คือ การที่ระบบคอมพิวเตอร์และซอฟต์แวร์กำลังเข้าควบคุมบทบาทหน้าที่ต่างๆ ที่มีผลกระทบต่อการดำรงชีวิตของเราอย่างมากและยังกลายเป็นส่วนหนึ่งที่มีความสำคัญต่อชีวิตของเรา ในปัจจุบันระบบคอมพิวเตอร์และซอฟต์แวร์กำลังทำหน้าที่ในการเฝ้าติดตามและควบคุมข้อมูลการกระทำธุรกรรมทางด้านต่างๆ ทั้งหมด ไม่ว่าจะเป็นการเงิน, การขนส่ง, การสื่อสาร, ระบบป้องกันภัย, ระบบการผลิต เป็นต้น รวมถึงการทำธุรกรรมต่างๆ จากการใช้งานในบ้านของเรา การยอมรับถึงความสำคัญและความซับซ้อนในโครงสร้างและบทบาทของซอฟต์แวร์ว่าไม่ได้เป็นเพียงผลิตภัณฑ์สำหรับการฝึกฝนใช้งานตามความต้องการของผู้พัฒนา, มีความไม่แน่นอนในการใช้งาน, หรือเป็นเพียงกิจกรรมทางด้านการ

พัฒนาโปรแกรมเท่านั้น ซึ่งซอฟต์แวร์ไม่ใช่เพียงโปรแกรมใดโปรแกรมหนึ่งที่ปฏิบัติการกระทำ งานใดงานหนึ่งเท่านั้น แต่ในปัจจุบันการทำงานกับกับโปรแกรมต่างๆ, โครงสร้างข้อมูล, และการ จัดทำเอกสารเป็น

โครงสร้างที่มีความซับซ้อนต่อการที่จะพัฒนาขึ้นเป็นระบบ, การทดสอบ, และการ บำรุงรักษา ผลิตภัณฑ์ซอฟต์แวร์สมัยใหม่จึงได้รับการจัดการภายใต้เทคนิคอย่างเป็นทางการที่ได้รับการ เลือกสรรแล้วในการที่จะสามารถปรับปรุงคุณภาพของผลิตภัณฑ์แห่งความพยายามในการที่จะ พัฒนาซอฟต์แวร์ขึ้น

ปัญหาหลักและสิ่งที่น่าจะเป็นงานที่สำคัญมากในการพัฒนาซอฟต์แวร์ ก็คือ“การรู้ว่าที่ ไหนควรจะเป็นจุดเริ่มต้น” แม้ว่าสถานการณ์ที่เลวร้ายที่สุดจะเกิดขึ้นเมื่อโครงการเริ่มต้นขึ้นด้วย บุคลากรที่เพิ่งเข้ามาใหม่ในทีม, เทคโนโลยีใหม่ล่าสุดที่ยังไม่ได้ผ่านการทดสอบอย่างถี่ถ้วน, ขอบเขตทางธุรกิจที่ยังไม่ชัดเจน สิ่งเหล่านี้ล้วนมีความทำลายต่อผู้ดำเนินโครงการด้วยกำหนดวันที่ ที่จะต้องส่งมอบงาน เมื่อสิ่งเหล่านี้ทั้งหมดกระทบต่อแผนการพัฒนาซอฟต์แวร์จะส่งผลให้การ จัดการตกอยู่ในวิกฤตแห่งความเสี่ยง ผลิตภัณฑ์ที่จะเสร็จได้นั้นก็จะเป็นการประเมินค่าใช้จ่ายที่ ผิดพลาดและระยะเวลาดำเนินงานก็ผิดไปจากแผนที่วางไว้และจะส่งผลให้คุณภาพของผลิตภัณฑ์ ซอฟต์แวร์ที่พัฒนาขึ้นนั้น ไม่สามารถใช้ได้ โดยเฉพาะอย่างยิ่งการที่ไม่สามารถรวบรวมรายละเอียด ขอบเขตความต้องการระบบของลูกค้าได้อย่างครบถ้วน ซึ่งในท้ายที่สุดจะส่งผลให้เกิดปัญหาต่อ ธุรกิจตามมาได้

ความท้าทายด้านอื่นๆ ก่อนความพยายามด้านการพัฒนาซอฟต์แวร์ ก็คือ การประเมิน ขอบเขตงานที่ยังคงคลุมเครือและไม่ชัดเจน, การประเมินกระบวนการดำเนินงานเพื่อให้ผลงานมี คุณภาพ, และการจัดการความเสี่ยง แม้ว่าการเจริญเติบโตทางเศรษฐกิจที่มีความเปลี่ยนแปลงอย่าง มากและการได้มาซึ่งประสิทธิภาพในการผลิตและการดำเนินงานนั้นจะสามารถเกิดขึ้นได้จาก ซอฟต์แวร์ แต่การถูกตำหนิเกี่ยวกับปัญหาด้านคุณภาพของซอฟต์แวร์ก็ยังคงมีอยู่ ทั้งหมดที่กล่าวมา นี้ส่งผลให้การพัฒนาซอฟต์แวร์เป็นการปฏิบัติงานที่มีความสำคัญขั้นวิกฤตด้านหนึ่งเลยทีเดียว

## 1.2 กระบวนการพัฒนาซอฟต์แวร์ (Process)

คล้ายๆ กับการจัดการด้านวิศวกรรมเพื่อการผลิตผลิตภัณฑ์ต่างๆ ผลิตภัณฑ์ ซอฟต์แวร์นั้นขึ้นอยู่กับความต้องการของลูกค้า การจัดการด้านวิศวกรรมซอฟต์แวร์นั้นประกอบด้วย กระบวนการที่มีโครงสร้างและนโยบายควบคุมการพัฒนาซอฟต์แวร์ การจัดทำเอกสารด้าน นโยบาย, เงื่อนไข, และขั้นตอนการดำเนินงานที่ทีมงานหรือองค์กรพัฒนาซอฟต์แวร์จะต้องปฏิบัติ ตามเพื่อการที่จะสามารถฝึกฝนด้านวิศวกรรมซอฟต์แวร์ได้นั้นถูกเรียกว่า “หลักเกณฑ์การพัฒนา ซอฟต์แวร์ (Software Development Methodology (SDM))” หรือ “วงจการพัฒนาซอฟต์แวร์

(System Development Life Cycle (SDLC))” หลักเกณฑ์การพัฒนาซอฟต์แวร์ (Software Development Methodology (SDM)) ล่าสุด คือ การที่มีการจัดระบบโครงสร้างการดำเนินงานกิจกรรมตามลำดับขั้นตอนที่ต่อเนื่องกัน (Sequential) และการดำเนินงานกิจกรรมต่างๆ ที่พร้อมกัน (Parallel) ได้ถูก

รวมเข้าด้วยกันสำหรับการดำเนินงานพัฒนาซอฟต์แวร์ แท้ที่จริงแล้ว “กระบวนการพัฒนาซอฟต์แวร์ (Process)” ก็คือ รายการงานต่างๆ ที่ได้รับการกำหนดขึ้น, สามารถดำเนินการซ้ำได้, และสามารถวัดได้ เพื่อนำไปสู่ผลลัพธ์ที่ก่อให้เกิดประโยชน์ต่อการพัฒนาซอฟต์แวร์ ซึ่งประโยชน์ที่จะได้รับจากการกำหนด “กระบวนการพัฒนาซอฟต์แวร์ (Process)” ได้ดีนั้นมีอยู่มากมาย

การกำหนด “กระบวนการพัฒนาซอฟต์แวร์ (Process)” จำเป็นจะต้องมีขั้นตอนดังต่อไปนี้

- การระบุช่วงระยะเวลา (Phase) การพัฒนาและรายการดำเนินงานต่างๆ ในแต่ละช่วงระยะเวลา (Phase)
- การออกแบบการเปลี่ยนผ่านระหว่างการดำเนินงานภายในทีมงานพัฒนาซอฟต์แวร์กับการดำเนินงานร่วมกับบุคคลอื่นภายนอกทีมงาน เช่น ผู้ใช้ระบบ, ผู้ทดสอบระบบ
- การเลือกใช้เทคนิคในการลงมือปฏิบัติรายการงานต่าง ๆ
- การตรวจสอบและการพิสูจน์ความถูกต้องของการดำเนินงานแต่ละรายการงาน และผลลัพธ์ที่ได้รับ
- การฝึกฝนทักษะด้านการจัดการโครงการและ “กระบวนการพัฒนาซอฟต์แวร์ (Process)”

### 1.3 ทำไมเราจึงจำเป็นต้องใช้มาตรฐานการพัฒนาซอฟต์แวร์

การกำหนดแบบจำลองวิธีการในอดีตยังอยู่ในรูปแบบที่ยังไม่เป็นทางการมากนัก คำนิยามมักจะออกมาจากสัญชาตญาณและยังมีช่องโหว่อย่างมาก ผู้คนผู้ที่มีภูมิหลังเกี่ยวกับวิธีการที่เป็นทางการมักจะวิจารณ์วิธีการเหล่านี้ในการดำเนินโครงการพัฒนาซอฟต์แวร์ พวกเขาโต้แย้งว่าขาดความหมายที่ชัดเจนและคลุมเครือมากเกินไป แต่ในความเป็นจริงแล้วผู้คนส่วนมากพบว่าแม้ว่าวิธีการลักษณะนี้ยังขาดความเป็นทางการ แต่วิธีการที่ไม่เป็นทางการนี้ให้ประโยชน์มากกว่าและให้ผลลัพธ์ที่เร็วกว่าวิธีการที่เป็นทางการ ในทางปฏิบัติแล้วดูเหมือนว่าวิธีการที่ไม่เป็นทางการนั้นจะมีข้อได้เปรียบที่ดีกว่า

แต่สำหรับโครงการพัฒนาซอฟต์แวร์ที่มีความซับซ้อนอย่างมากและมีขนาดใหญ่ นั้นวิธีการที่ไม่เป็นทางการได้พิสูจน์แล้วว่าเป็นสาเหตุหลักที่ทำให้โครงการล้มเหลว เมื่องานได้ถูกกระจายออกไปยังหลายๆ บุคคลหรือหลายทีมเพื่อช่วยกันทำงาน พวกเขาไม่สามารถทำงานที่มีความสัมพันธ์กันและบูรณาการงานของแต่ละบุคคลแต่ละทีมเข้าด้วยกันได้ เพราะข้อสมมุติฐาน

และพื้นฐานด้านการพัฒนาซอฟต์แวร์ที่ไม่เป็นปกติตามมาตรฐาน การไม่ปฏิบัติตามมาตรฐานการทำงานและแนวทางที่เป็นกลยุทธ์สำหรับการทำงานจะทำให้พวกเขาต้องพบกับวิกฤตและความล้มเหลวในการทำงาน อุตสาหกรรมด้านการพัฒนาซอฟต์แวร์เข้าใจถึงปัญหานี้ที่จำเป็นจะต้องมีกระบวนการที่เป็นทางการทางด้านการพัฒนาซอฟต์แวร์พวกเขาจึงได้พัฒนากระบวนการเหล่านี้จากประสบการณ์การทำงานทางด้านซอฟต์แวร์ของพวกเขา

ข้อมูลตามข้างต้นแสดงให้เห็นถึงความจำเป็นอย่างเด่นชัดที่จะต้องมีมาตรฐานสำหรับการพัฒนาและการวัดได้ทางด้านการพัฒนาซอฟต์แวร์ซึ่งขึ้นอยู่กับขนาด, ความซับซ้อน, และคุณลักษณะด้านคุณภาพของซอฟต์แวร์ การกำหนดมาตรฐานอย่างเป็นทางการยังคงต้องการการควบคุมกระบวนการการพัฒนาซอฟต์แวร์อีกด้วย ดังนั้นหากทีมงานใดฝ่าฝืนการปฏิบัติตามมาตรฐานแล้ว การส่งมอบงานหรือการเผยแพร่ผลิตภัณฑ์ซอฟต์แวร์ที่เสร็จแล้วอาจจะสามารถได้รับการเปลี่ยนไปให้เป็นการดำเนินงานของทีมงานอื่นที่สามารถปฏิบัติตามมาตรฐานแทนได้

## 2. ประวัติและความเป็นมาของระบบมาตรฐาน ISO/IEC 29110

ISO/IEC 29110: คือ มาตรฐานของวงจรการพัฒนาซอฟต์แวร์และระบบ และแนวปฏิบัติสำหรับองค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (Very Small Entities (VSEs)), มาตรฐานระดับนานาชาติ (International Standards (IS)), และรายงานทางเทคนิค (Technical Reports (TR)) ได้รับการกำหนดเป้าหมายให้เป็นมาตรฐานสำหรับองค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (Very Small Entities (VSEs)) โดยแต่ละองค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (Very Small Entities (VSEs)) ได้แก่ องค์กรขนาดใหญ่, ขนาดกลาง, แพลน, หรือโครงการ ที่มีสมาชิกในการดำเนินงานพัฒนาซอฟต์แวร์สูงสุดไม่เกิน 25 คน ISO/IEC 29110 คือ ชุดของมาตรฐานใหม่ระดับนานาชาติ ที่มีการตั้งชื่อว่า “วิศวกรรมซอฟต์แวร์และระบบ – โครงร่างวงจรการพัฒนาซอฟต์แวร์สำหรับองค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (Systems and Software Engineering — Lifecycle Profiles for Very Small Entities (VSEs))” มาตรฐานนี้ได้รับการพัฒนาโดยกลุ่มทำงาน 24 (WG24) ประกอบด้วย Sub-committee 7 (SC7) of Joint Technical Committee 1 (JTC1) of the International Organization for Standardization และ the International Electrotechnical Commission

**2.1 Sub-committee 7 (SC7) of Joint Technical Committee 1 (JTC1) of the International Organization for Standardization** ประกอบด้วยกลุ่มทำงานที่มีคุณลักษณะ ดังนี้



### 2.1.1 Sub-committee 7 (SC7)

ตารางที่ 2.1 รายละเอียดกลุ่มทำงาน Sub-committee 7 (SC7)

ISO/IEC Joint Technical Committee 1, Subcommittee 7, Software and systems engineering	
ตัวอักษรย่อ	ISO/IEC JTC1/SC7
การก่อตั้ง	ค.ศ. 1987; 27 ปีที่ผ่านมา
การรวมตัวกัน ระหว่าง หน่วยงาน	ISO/TC 97 และ IEC/TC 83
ประเภท	องค์กรด้านการกำหนดมาตรฐาน (Standards organization)
วัตถุประสงค์	การกำหนดมาตรฐานด้านกระบวนการ, การสนับสนุนเครื่องมือ และการสนับสนุนเทคโนโลยีสำหรับวิศวกรรมผลิตภัณฑ์ ซอฟต์แวร์และระบบ
สำนักงานใหญ่	องค์กรมาตรฐานแห่งแคนาดา
ขอบเขตการให้บริการ	ทั่วโลก
สมาชิก	องค์กรมาตรฐานระดับชาติ
ภาษาที่เป็นทางการ	อังกฤษ, ฝรั่งเศส
การขับเคลื่อนหลัก	การประชุมที่ครอบงำประชุมอย่างสมบูรณ์
องค์กรที่กำกับดูแลองค์กรนี้	ISO/IEC Joint Technical Committee 1
เว็บไซต์	<a href="http://www.jtc1-sc7.org">www.jtc1-sc7.org</a>
ชื่อก่อนหน้า	ISO/IEC JTC1/SC7 Software engineering

### 2.2.2 Joint Technical Committee 1 (JTC1)

ตารางที่ 2.2 รายละเอียดคํากลุ่มทํางาน Joint Technical Committee 1 (JTC1)

<b>ISO/IEC JTC 1 — Information Technology</b>	
<b>การก่อตั้ง</b>	ค.ศ. 1987
<b>ประเภท</b>	องค์กรด้านการกำหนดมาตรฐาน (Standards organization)
<b>วัตถุประสงค์</b>	พัฒนามาตรฐานด้านเทคโนโลยีการสื่อสารและระบบสารสนเทศระดับโลก (Worldwide Information And Communications Technology (ICT)) สำหรับแอปพลิเคชันทางธุรกิจและผู้บริโภค
<b>สถานที่</b>	นิวยอร์ก
<b>ขอบเขตการให้บริการ</b>	ทั่วโลก
<b>องค์กรที่กำกับดูแลองค์กรนี้</b>	International Organization for Standardization (ISO) และ International Electrotechnical Commission (IEC)
<b>เว็บไซต์</b>	<a href="http://www.iso.org/iso/jtc1_home.html">http://www.iso.org/iso/jtc1_home.html</a>



### 2.2.3 The International Organization for Standardization

ตารางที่ 2.3 รายละเอียด The International Organization for Standardization

International Organization for Standardization <sup>[1]</sup>	
	
ตัวอักษรย่อ	ISO
การก่อตั้ง	23 กุมภาพันธ์ ค.ศ. 1947
ประเภท	องค์กรที่ไม่ใช่หน่วยงานของรัฐบาล
วัตถุประสงค์	กำหนดมาตรฐานระดับโลก
สำนักงานใหญ่	เจนีวา ประเทศสวิตเซอร์แลนด์
สมาชิก	163 สมาชิก
ภาษาที่เป็นทางการ	อังกฤษ, ฝรั่งเศส, รัสเซีย
เว็บไซต์	iso.org

**2.2 The International Electrotechnical Commission (IEC)** เป็นองค์กรที่ไม่มุ่งหวังผลกำไร ไม่ใช่องค์กรมาตรฐานระดับนานาชาติของรัฐบาล มีหน้าที่ในการจัดเตรียมและเผยแพร่มาตรฐานระดับนานาชาติทางด้านอิเล็กทรอนิกส์และด้านเทคโนโลยีทุกประเภท มาตรฐาน IEC ครอบคลุมขอบเขตด้านเทคโนโลยีที่กว้างขวางมากตั้งแต่ระบบให้กำเนิดพลังงาน, อุปกรณ์กึ่งตัวนำ, ไฟเบอร์ ออปติกส์, แบตเตอรี่, พลังงานแสงอาทิตย์, นาโนเทคโนโลยี, พลังงานด้านการเดินเรือ และอื่นๆ อีกมากมาย IEC ทำหน้าที่จัดการระบบการประเมิน 3 ระบบมีความสอดคล้องกันทั่วโลก ซึ่งมีการรับรองว่าอุปกรณ์, ระบบ, หรือส่วนประกอบ มีความสอดคล้องกับมาตรฐานระดับนานาชาติ

### 2.2.1 สมาชิกเต็มรูปแบบ มีดังนี้

ตารางที่ 2.4 สมาชิก The International Organization for Standardization เต็มรูปแบบ

• <u>Algeria</u>	• <u>India - Bureau of Indian Standards (BIS)</u>	• <u>Saudi Arabia</u>
• <u>Argentina - Instituto Argentino de Normalización y Certificación (IRAM)</u>	• <u>Indonesia</u>	• <u>Serbia</u>
• <u>Australia- Standards Australia</u>	• <u>Iran</u>	• <u>Singapore</u>
• <u>Austria - Österreichischer Verband für Elektrotechnik (ÖVE)</u>	• <u>Iraq</u>	• <u>Slovakia</u>
• <u>Belarus</u>	• <u>Ireland</u>	• <u>Slovenia</u>
• <u>Belgium</u>	• <u>Israel</u>	• <u>South Korea - Korean Agency for Standards (KATS)</u>
• <u>Brazil - Comitê Brasileiro de Eletricidade, Eletrônica, Iluminação e Telecomunicações (Cobei)</u>	• <u>Italy - Comitato Eletrotecnico Italiano (CEI)</u>	• <u>South Africa - South African Bureau of Standards (SABS)</u>
• <u>Bulgaria</u>	• <u>Japan - Japanese Industrial Standards Committee</u>	• <u>Spain - Asociación Española de Normalización y Certificación (AENOR)</u>
• <u>Canada - Standards Council of Canada</u>	• <u>Libya</u>	
• <u>Colombia</u>	• <u>Luxembourg</u>	
• <u>Chile</u>	• <u>Malaysia</u>	
• <u>China - Standardization Administration of China (SAC)</u>	• <u>Mexico</u>	
• <u>Croatia</u>	• <u>Netherlands</u>	
• <u>Czech Republic</u>	• <u>New Zealand</u>	

## ตารางที่ 2.4 (ต่อ)

<ul style="list-style-type: none"> <li>• <u>Denmark</u></li> <li>• <u>Egypt</u></li> <li>• <u>Finland</u></li> <li>• <u>France</u> - <u>Union technique de l'électricité et de la communication (UTE)</u></li> <li>• <u>Germany</u> - <u>Deutsche Kommission Elektrotechnik Elektronik Informationstechnik DIN und VDE (DKE)</u></li> <li>• <u>Greece</u></li> </ul> <p><u>Hungary</u></p>	<ul style="list-style-type: none"> <li>• <u>Norway</u></li> <li>• <u>Oman</u></li> <li>• <u>Pakistan</u></li> <li>• <u>Philippines</u></li> <li>• <u>Poland</u></li> <li>• <u>Portugal</u></li> <li>• <u>Qatar</u></li> <li>• <u>Romania</u></li> </ul> <p><u>Russia</u> - Federal Agency for Technical Regulation and Metrology (<u>Rostekhnregulirovaniye</u>)</p>	<ul style="list-style-type: none"> <li>• <u>Sweden</u> - Swedish Electrical Standard (SEK)</li> <li>• <u>Switzerland</u> - Swiss Electrotechnical Committee (CES)</li> <li>• <u>Thailand</u></li> <li>• <u>Turkey</u></li> <li>• <u>Ukraine</u></li> <li>• <u>United Arab Emirates</u></li> <li>• <u>United Kingdom</u> - <u>British Electrotechnical Committee (BEC)</u>, part of the British Standards Institution (BSI)</li> </ul> <p><u>United States</u> - <u>American National Standards Institute (ANSI) (USNC/IEC)</u>; <u>NEMA</u> also helps to develop and promote IEC standards</p>
--	--	---

2.2.2 สมาชิกร่วมมือกัน มีดังนี้ (จำกัดการลงคะแนนเสียงและสิทธิด้านการจัดการ)

ตารางที่ 2.5 สมาชิกร่วมมือกัน The International Organization for Standardization

• <u>Albania</u>	• <u>Jordan</u>	• <u>Moldova</u>
• <u>Bahrain</u>	• <u>Kazakhstan</u>	• <u>Montenegro</u>
• <u>Bosnia &amp; Herzegovina</u>	• <u>Kenya</u>	• <u>Morocco - COMELEC</u>
• <u>Cuba</u>	• <u>Latvia</u>	• <u>Nigeria</u>
• <u>Cyprus</u>	• <u>Lithuania</u>	• <u>Sri Lanka</u>
• <u>Estonia</u>	• <u>Republic of Macedonia</u>	• <u>Tunisia</u>
• <u>Georgia</u>	• <u>Malta</u>	• <u>Vietnam</u> - Vietnamese National Committee Directorate for Standards and Quality (STAMEQ)
• <u>Iceland</u> - Icelandic Standards (IST)		

ชุดของ ISO/IEC 29110 มีเป้าหมายสำหรับผู้เข้าร่วมเพื่อการพัฒนาไปสู่การปรับปรุงผลิตภัณฑ์และ/หรือคุณภาพการบริการ และการเพิ่มประสิทธิภาพของกระบวนการ ดังตารางด้านล่างซึ่ง ISO/IEC 29110 ไม่ได้มีวัตถุประสงค์ที่จะการจัดการใช้วงจรการดำเนินงานที่แตกต่างกันระหว่างแบบจำลอง Waterfall, Iterative, Incremental, Evolutionary หรือ Agile

ตารางที่ 2.6 ชุดองค์ประกอบของ ISO/IEC 29110

ISO/IEC 29110	ชื่อเรื่อง	กลุ่มผู้เข้าร่วมที่เป็นเป้าหมาย
Part 1	Overview	องค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (VSEs), ผู้ประเมิน (assessors), ผู้กำหนดมาตรฐาน (standards producers), ผู้จำหน่ายเครื่องมือ (tool vendors), และ ผู้จำหน่ายวิธีการ (methodology vendors)
Part2	Framework and taxonomy	ผู้กำหนดมาตรฐาน (standards producers), ผู้จำหน่ายเครื่องมือ (tool vendors), และ ผู้จำหน่ายวิธีการ (methodology vendors) ไม่รวมองค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (VSEs)
Part3	Assessment guide	ผู้ประเมิน (assessors) และองค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (VSEs)
Part4	Profile specifications	ผู้กำหนดมาตรฐาน (standards producers), ผู้จำหน่ายเครื่องมือ (tool vendors), และ ผู้จำหน่ายวิธีการ (methodology vendors) ไม่รวมองค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (VSEs)
Part5	Management and engineering guide	องค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (VSEs)

### 3. ความรู้เบื้องต้นเกี่ยวกับมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

รายละเอียดคุณลักษณะของมาตรฐาน ISO/IEC 29110 สำหรับองค์กรพัฒนาซอฟต์แวร์ไม่เกิน 25 คน (VSEs) ระดับ Basic Profile มีดังนี้

#### 3.1 มาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

##### 3.1.1 พื้นฐานเหตุผลแห่งระดับ Basic Profile

1) การกำหนดแนวทางการจัดการโครงการและการพัฒนาซอฟต์แวร์สำหรับกระบวนการย่อยของกระบวนการและผลลัพธ์ของผลิตภัณฑ์ ISO/IEC 12207 และ ISO/IEC 15289 เพื่อความเหมาะสมสำหรับคุณลักษณะเฉพาะและความต้องการขององค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (VSEs)

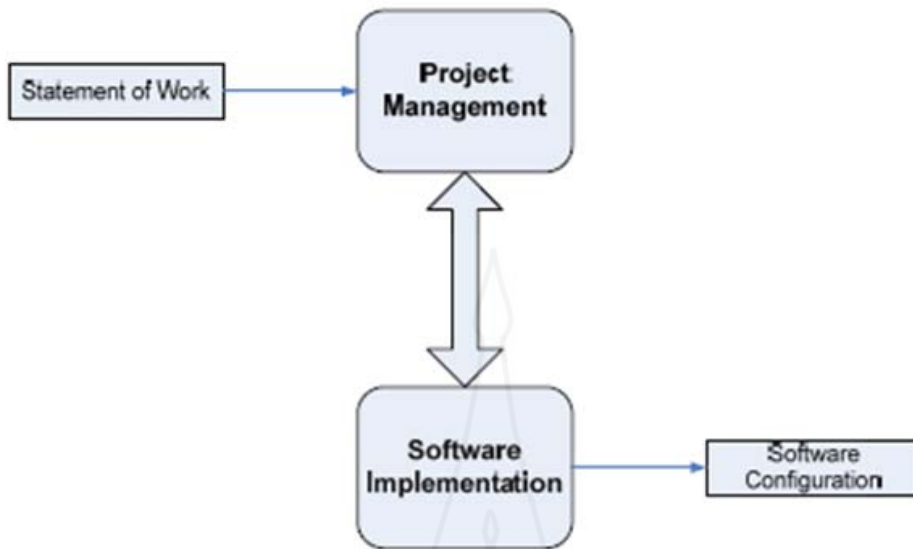
2) เหตุผลในการรวมการจัดการโครงการ ก็คือการทำธุรกิจหลักขององค์กรพัฒนาซอฟต์แวร์ขนาดเล็ก (VSEs) คือการพัฒนาซอฟต์แวร์ และความสำเร็จทางการเงินขององค์กรจะขึ้นอยู่กับผลกำไรของโครงการ

##### 3.1.2 ความเหมาะสม

1) การบรรยายถึงความเหมาะสมของการพัฒนาซอฟต์แวร์สำหรับ 1 โครงการ ซึ่งดำเนินการโดย 1 กลุ่มทำงานโครงการ โดยที่ไม่มีปัจจัยเกี่ยวกับสถานการณ์หรือความเสี่ยงพิเศษใดๆ

2) โครงการอาจจะเป็นการดำเนินการให้สำเร็จสำหรับสัญญาภายในหรือภายนอกองค์กรก็ได้

#### 3.2 กระบวนการที่แนะนำระบบมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile



ภาพที่ 2.1 กระบวนการภาพรวมของมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

ที่มา: Software Development using ISO/IEC 29110 TR – Engineering and Management Guide

Project Management (PM) Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์) ใช้ขอบข่ายงานของลูกค้า(Statement of Work) ในการบรรยายเพื่อจัดทำแผนโครงการ (Project Plan) งานด้านการประเมินและควบคุมโครงการของ Project Management (PM) Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์) เป็นการเปรียบเทียบความคืบหน้าของการดำเนินโครงการจริงกับแผนโครงการ (Project Plan) และการกระทำต่างๆ ที่กระทำเพื่อขจัดความคลาดเคลื่อนหรือการเปลี่ยนแปลงที่เกิดขึ้นกับแผนโครงการ (Project Plan) ส่วนของกิจกรรมการปิดโครงการ (PM Project Closure) เป็นการส่งมอบองค์ประกอบของซอฟต์แวร์(Software Configuration)ที่ดำเนินการโดย Software Implementation (SI) Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์) และได้รับการยอมรับจากลูกค้าอย่างเป็นทางการในการที่สามารถจบโครงการได้ ส่วนที่จัดเก็บโครงการ (Project Repository) ได้รับการกำหนดไว้ให้เป็นที่จัดเก็บผลิตภัณฑ์ที่เป็นผลจากการดำเนินงานและใช้สำหรับการควบคุมเวอร์ชันระหว่างการดำเนินโครงการ

Software Implementation (SI) Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์) ได้รับการขับเคลื่อนโดยแผนโครงการ (Project Plan) Software Implementation (SI) Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์) เริ่มต้นด้วยการริเริ่มกิจกรรมการทบทวนแผนโครงการ (Project Plan) ซึ่งแผนโครงการ (Project Plan) จะชี้นำการดำเนินการด้านการวิเคราะห์ความ

ต้องการซอฟต์แวร์, การออกแบบรายละเอียดและโครงสร้างสถาปัตยกรรมซอฟต์แวร์, การก่อสร้างซอฟต์แวร์, การบูรณาการและทดสอบซอฟต์แวร์, และกิจกรรมการส่งมอบผลิตภัณฑ์ซอฟต์แวร์

### 3.3 Project Management (PM) Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์)

#### 3.3.1 วัตถุประสงค์หลัก

1) เพื่อให้บรรลุและสามารถจัดการงานตามแนวทางที่เป็นระบบสำหรับโครงการพัฒนา/ติดตั้งซอฟต์แวร์ ซึ่งยินยอมให้เป็นไปตามวัตถุประสงค์ของโครงการในการคาดหวังถึงคุณภาพ, ระยะเวลา, และค่าใช้จ่าย

#### 3.3.2 วัตถุประสงค์ของกระบวนการจัดการโครงการพัฒนาซอฟต์แวร์

1) PM.O1: Project Plan (แผนโครงการ) สำหรับการดำเนินโครงการได้รับการจัดทำขึ้นอ้างอิงตามขอบข่ายงาน และได้รับการตรวจสอบจากลูกค้าแล้ว รายการงานและทรัพยากรต่างๆ ที่มีความจำเป็นต่อการดำเนินงานให้เสร็จสิ้นได้รับการประเมินไว้แล้ว

2) PM.O2: Progress (ความคืบหน้าของการดำเนินโครงการ) ได้รับการติดตามอ้างอิง Project Plan (แผนโครงการ) และได้รับการบันทึกไว้ในเอกสาร Progress Status Record

3) PM.O3: Change Request ได้รับการจัดเตรียมไว้สำหรับการวิเคราะห์และรับไว้พิจารณาแก้ไข การเปลี่ยนแปลงความต้องการซอฟต์แวร์จะได้รับการประเมินผลกระทบด้านค่าใช้จ่าย, ด้านกำหนดการดำเนินงาน, และด้านเทคนิค

4) PM.O4: Review Meetings (การประชุมเพื่อทบทวน) การประชุมเพื่อทบทวนร่วมกับทีมงานและลูกค้าได้มีการจัดขึ้นข้อตกลงต่างๆ ได้รับการบันทึกไว้และติดตาม

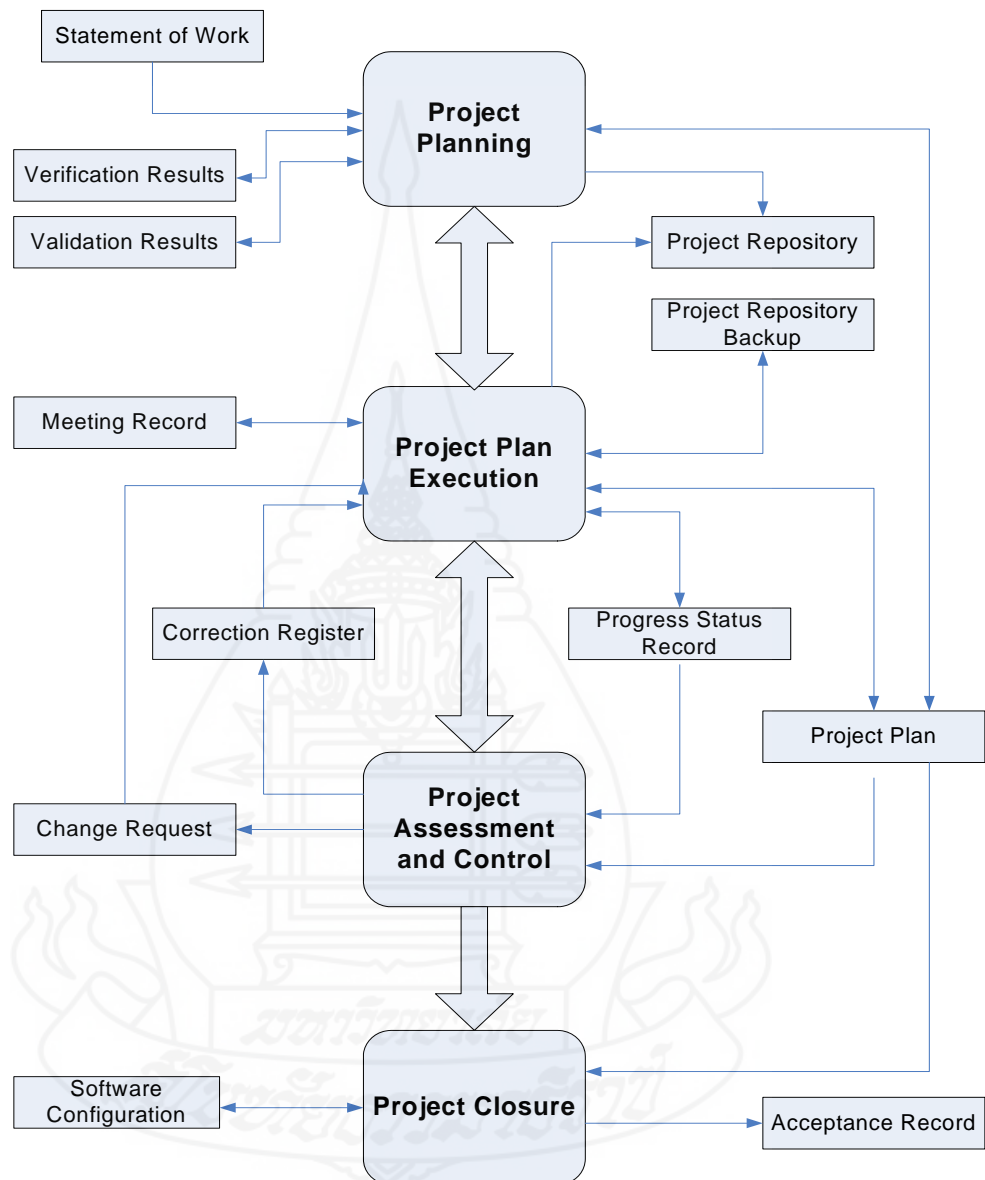
5) PM.O5: Risk are identified (ความเสี่ยงได้รับการระบุไว้) ระหว่างการจัดทำProject Plan (แผนโครงการ) และระหว่างการจัดการโครงการ

6) PM.O6: A software Version Control Strategy is developed. (กลยุทธ์การควบคุมเวอร์ชันของซอฟต์แวร์ได้รับการกำหนดขึ้น) รายการองค์ประกอบของซอฟต์แวร์ได้รับการระบุ, กำหนด, และวางพื้นฐานไว้ การแก้ไขและนำซอฟต์แวร์ออกติดตั้งได้รับการควบคุมและทำให้พร้อมใช้งานสำหรับทั้งลูกค้าและทีมงานทั้งทางด้านการจัดการสถานที่จัดเก็บ และการจัดการด้านการส่งมอบ

7) PM.O7: Software Quality Assurance is performed. (การรับประกันคุณภาพซอฟต์แวร์ได้รับการดำเนินการ) เพื่อเป็นการรับประกันว่าผลิตภัณฑ์ซอฟต์แวร์และกระบวนการมีความสอดคล้องกับ Project Plan (แผนโครงการ) และข้อกำหนดเฉพาะของความต้องการซอฟต์แวร์ (Requirements Specification)



3.3.3 4Activities ของ Project Management (PM) Process (กระบวนการจัดการ  
โครงการพัฒนาซอฟต์แวร์)



ภาพที่ 2.2 4Activities ของ Project Management (PM) Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์)

### 3.4 Software Implementation (SI) Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์)

#### 3.4.1 วัตถุประสงค์หลัก

1) วัตถุประสงค์ของกระบวนการพัฒนา/ติดตั้งซอฟต์แวร์คือ การทำให้บรรลุผลสำเร็จที่เป็นระบบของกิจกรรมการวิเคราะห์, การออกแบบ, การพัฒนาหรือสร้าง, การบูรณาการ, และการทดสอบสำหรับผลิตภัณฑ์ซอฟต์แวร์ใหม่หรือที่แก้ไขเพื่อให้เป็นไปตามความต้องการซอฟต์แวร์ที่มีความเฉพาะเจาะจง

#### 3.4.2 7 วัตถุประสงค์ของกระบวนการพัฒนาซอฟต์แวร์

1) SI.O1: Tasks of the activities are performed through the accomplishment of the current Project Plan. (รายการงานของกิจกรรมต่างๆ ได้รับการกระทำให้เป็นผลสำเร็จตามที่กำหนดไว้ใน Project Plan (แผนโครงการ))

2) SI.O2: Software Requirements ความต้องการซอฟต์แวร์ได้รับการกำหนด, วิเคราะห์ ว่ามีความถูกต้องและสามารถทดสอบได้ โดยได้รับการอนุมัติจากทางลูกค้าเพื่อใช้เป็นพื้นฐานในการยอมรับและสื่อสารร่วมกัน

3) SI.O3: Software Architectural and Detailed Design การออกแบบ โครงสร้างทางสถาปัตยกรรมและรายละเอียดซอฟต์แวร์ได้รับการพัฒนาจัดทำขึ้นและกำหนดเป็นพื้นฐานในการยอมรับซึ่งใช้ในการอธิบายรายการองค์ประกอบของซอฟต์แวร์, การเชื่อมต่อกันระหว่างองค์ประกอบภายในของซอฟต์แวร์, และการเชื่อมต่อระหว่างระบบอื่นภายนอกซอฟต์แวร์ความถูกต้องและการตรวจสอบย้อนกลับอ้างอิงตามความต้องการซอฟต์แวร์สามารถพิสูจน์ให้เห็นได้

4) SI.O4: Software Components องค์ประกอบของซอฟต์แวร์ได้รับการกำหนดขึ้นในขั้นตอนการออกแบบได้รับการพัฒนาจัดทำขึ้น การทดสอบหน่วยย่อย (Unit Test) ได้รับการกำหนดและดำเนินการเพื่อยืนยันความถูกต้องในความสอดคล้องกับความต้องการซอฟต์แวร์และการออกแบบการตรวจสอบย้อนกลับไปยังความต้องการซอฟต์แวร์และการออกแบบสามารถพิสูจน์ให้เห็นได้

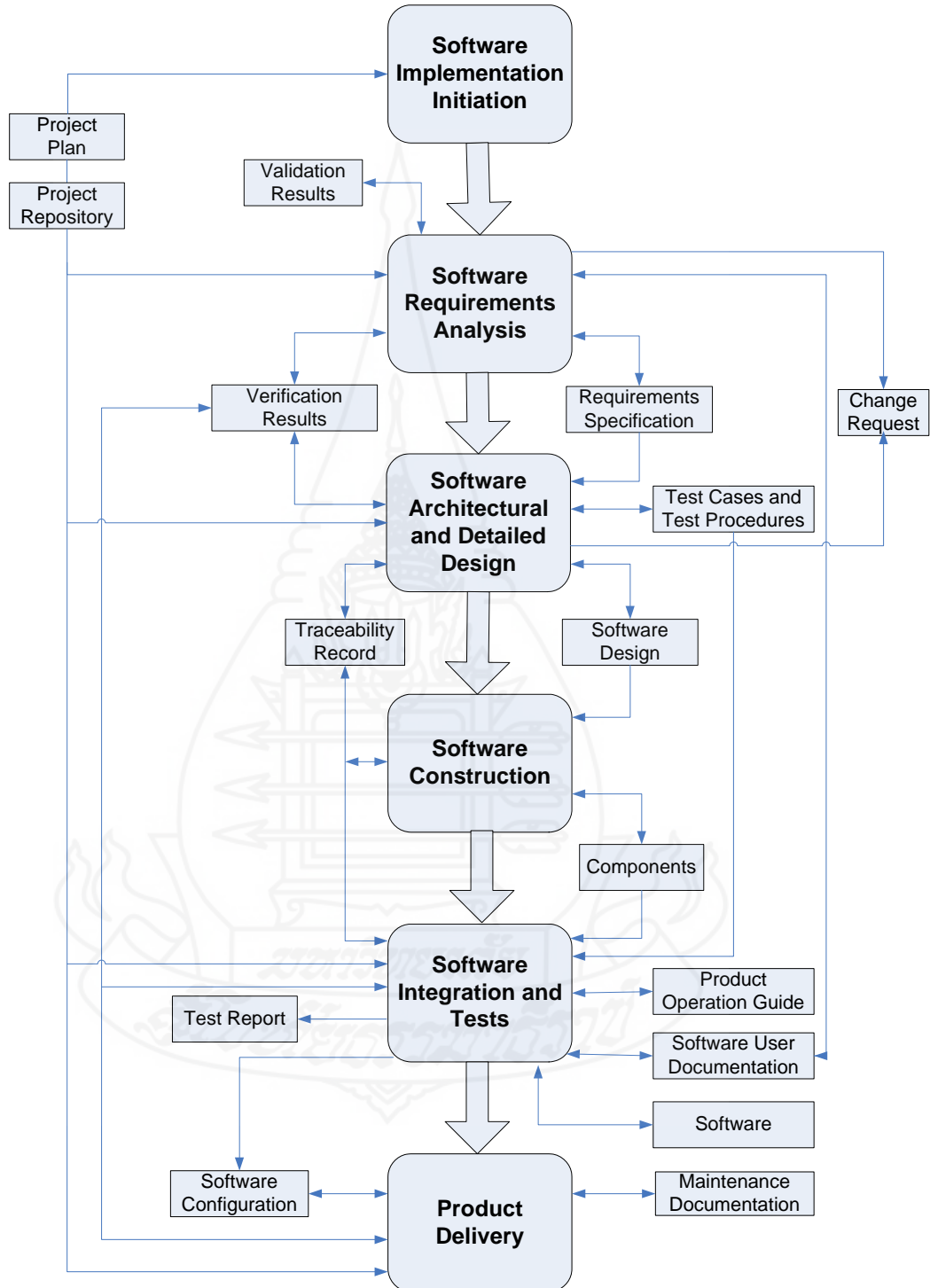
5) SI.O5: Software is produced performing integration of software components and verified using Test Cases and Test Procedures. ซอฟต์แวร์ได้รับการผลิตโดยการบูรณาการองค์ประกอบของซอฟต์แวร์และได้รับการยืนยันความถูกต้องโดยการทดสอบกรณี (Test Case) และการทดสอบการดำเนินงาน (Test Procedure) ผลการทดสอบได้รับการบันทึกในเอกสารรายงานการทดสอบ (Test Report) ข้อผิดพลาดได้รับการแก้ไขให้ถูกต้อง, ความสอดคล้องกับความต้องการซอฟต์แวร์และการออกแบบ, และการตรวจสอบย้อนกลับไปยังความต้องการซอฟต์แวร์และการออกแบบได้รับการพิสูจน์ให้เห็นได้

6) SI.O6: Software Configuration ซอฟต์แวร์ที่พัฒนาเสร็จแล้วซึ่งตรงตามข้อกำหนดเฉพาะของความต้องการซอฟต์แวร์ตามที่ได้รับการยืนยันจากลูกค้า ซึ่งประกอบด้วยเอกสารสำหรับผู้ใช้ระบบ, เอกสารด้านการปฏิบัติการและการบำรุงรักษาระบบ ได้รับการบูรณาการ, ได้รับการกำหนดเป็นพื้นฐานในการยอมรับ, และได้รับการจัดเก็บในProject Repository (ที่เก็บข้อมูลโครงการ) แล้ว ในส่วนของความต้องการเปลี่ยนแปลงซอฟต์แวร์ที่เสร็จแล้วนั้นจะได้รับการตรวจสอบว่ามีขึ้นหรือไม่ และเอกสารChange Request (การร้องขอเปลี่ยนแปลง) จะได้รับการริเริ่มจัดทำขึ้น

7) SI.O7: Verification and Validation tasks of all required work products are performed งานด้านการตรวจสอบและยืนยันความถูกต้องของ Work Products (ผลิตภัณฑ์การทำงาน) ทั้งหมด ได้รับการดำเนินการตามเงื่อนไขที่กำหนดไว้ตามมาตรฐานเพื่อให้บรรลุถึงความถูกต้องเหมาะสมระหว่าง Input Products (ผลิตภัณฑ์นำเข้า) และ Output Products (ผลิตภัณฑ์ที่ได้รับการผลิตออกมา) ของแต่ละกิจกรรมข้อผิดพลาดได้รับการแสดงให้เห็น, ได้รับการแก้ไขให้ถูกต้อง, และได้รับการบันทึกไว้ในเอกสารผลการยืนยันความถูกต้อง (Verification Result) และเอกสารผลการตรวจสอบตามแนวทางแห่งข้อกำหนด (Validation Result)



3.4.2 6 Activities ของ Software Implementation (SI) Process (กระบวนการพัฒนา / ติดตั้งซอฟต์แวร์)



ภาพที่ 2.3 6Activities ของ Software Implementation (SI) Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์)

# บทที่ 3

## วิธีการดำเนินโครงการ

### 1. ขั้นตอนการดำเนินโครงการ

1.1 ศึกษารายละเอียดคุณลักษณะของมาตรฐาน ISO/IEC 29110 ด้านการพัฒนาซอฟต์แวร์สำหรับทีมพัฒนาซอฟต์แวร์ไม่เกิน 25 คน (A Very Small Entity (VSE) is an enterprise, organization, department or project having up to 25 people) ระดับ Basic Profile

โดยศึกษาถึงรายละเอียดของระบบมาตรฐาน ISO/IEC 29110 ในด้านต่างๆ ดังนี้

1.1.1 ที่มาของระบบมาตรฐาน ISO/IEC 29110

1.1.2 รายละเอียดคุณลักษณะของมาตรฐาน ISO/IEC 29110 ด้านการพัฒนาซอฟต์แวร์สำหรับทีมพัฒนาซอฟต์แวร์ไม่เกิน 25 คน (A Very Small Entity (VSE) is an enterprise, organization, department or project having up to 25 people)

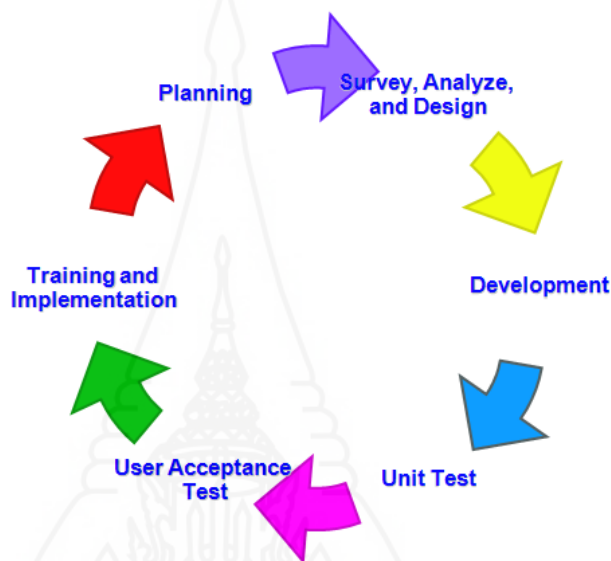
1.2 รวบรวมข้อมูลการดำเนินโครงการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile และรวบรวมข้อมูลความต้องการระบบสำหรับใช้ในการตรวจสอบและควบคุมคุณภาพการปฏิบัติตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile จากผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยที่ผ่านการรับรองมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile โดยการกำหนดคำถามและออกแบบแบบสอบถามสำหรับการสัมภาษณ์

ดำเนินการรวบรวมข้อมูลจากองค์กรผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยที่ผ่านการรับรองมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile โดยได้เลือกบริษัท โปรเฟสชั่นแนล เซอร์วิส โพรไวเดอร์ จำกัด (Professional Service Provider Co., Ltd.) เป็นบริษัทตัวอย่างที่ให้ข้อมูลรายละเอียดการดำเนินโครงการพัฒนาซอฟต์แวร์ที่สามารถผ่านการรับรองมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile ได้

บริษัท โปรเฟสชั่นแนล เซอร์วิส โพรไวเดอร์ จำกัด ดำเนินโครงการพัฒนาซอฟต์แวร์ในลักษณะ Custom Software Development กล่าวคือเป็นการพัฒนาซอฟต์แวร์ตามความต้องการที่เฉพาะเจาะจงของลูกค้าแต่ละราย ประกอบด้วยโครงการขนาดกลางที่มีทีมงานดำเนินโครงการไม่เกิน 5 คน ไปจนถึงโครงการขนาดใหญ่ที่มีทีมงานดำเนินโครงการประมาณ 8-10 คน โดยมีการดำเนินโครงการพร้อมกันประมาณ 3-4 โครงการในแต่ละช่วงเวลาในส่วนของขั้นตอนการดำเนินงาน

แต่ละโครงการนั้นครอบคลุมขั้นตอนการพัฒนาซอฟต์แวร์ตาม Standard Software Development Life Cycle ดังภาพที่ 3.1 แต่ยังไม่ได้มีการจัดทำเอกสารและการดำเนินงานเพิ่มเติมบางส่วนได้ครอบคลุมตามวัตถุประสงค์ของมาตรฐาน ISO/IEC 29110 อย่างครบถ้วน

## Development Life Cycle



ภาพที่ 3.1 Standard Software Development Life Cycle

สิ่งที่สนใจในการศึกษาและวิเคราะห์บริษัทตัวอย่างมีดังนี้

### 1.2.1 ข้อมูลทั่วไป ประกอบด้วย

- 1) ข้อมูลทั่วไปขององค์กร
- 2) วัตถุประสงค์ในการพัฒนาองค์กร
- 3) Development Methodology ที่ใช้ในการพัฒนาซอฟต์แวร์
- 4) ปัจจัยที่ทำให้องค์กรตัดสินใจปรับปรุงกระบวนการพัฒนาซอฟต์แวร์

โดยเลือกใช้มาตรฐาน ISO/IEC 29110

5) ปัญหาที่พบระหว่างการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ขององค์กร

6) ปัจจัยที่มีผลต่อความสำเร็จในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ขององค์กร

### 1.2.2 ข้อมูลกระบวนการจัดการโครงการพัฒนาซอฟต์แวร์ขององค์กรตามมาตรฐาน ISO/IEC 29110 ประกอบด้วย

1) ขั้นตอนการดำเนินงานและเอกสารที่เกี่ยวข้องขององค์กร ที่สามารถบรรลุวัตถุประสงค์ของกระบวนการจัดการโครงการพัฒนาซอฟต์แวร์ (Project Management Process) ตามมาตรฐาน ISO/IEC 29110

2) ขั้นตอนการดำเนินงานและเอกสารที่เกี่ยวข้องขององค์กร ที่บรรลุวัตถุประสงค์ของกระบวนการพัฒนา/ติดตั้งซอฟต์แวร์ (Software Implementation Process) ตามมาตรฐาน ISO/IEC 29110

### 1.2.3 ข้อมูลความต้องการคุณลักษณะของระบบที่จะสามารถนำมาใช้ประกอบการตรวจสอบและควบคุมคุณภาพโครงการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110

1.3 พัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทย

1.4 พัฒนาดัชนีแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

1.5 ประเมินผลการพัฒนาแนวปฏิบัติที่ดีและดัชนีแบบระบบ โดยนำแนวปฏิบัติที่ดีและดัชนีแบบระบบที่พัฒนาขึ้นไปนำเสนอต่อองค์กรผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยที่ผ่านการรับรองมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile แล้วทำการสรุปข้อคิดเห็นและข้อเสนอแนะเพิ่มเติม

1.6 สรุปผลที่ได้จากการดำเนินงานและศึกษาข้อมูล เสนอแนะ

สรุปผลการดำเนินงานในด้านต่างๆ ดังนี้

1) วิเคราะห์ข้อมูลที่ได้ทำการศึกษาและรวบรวม แล้วทำการสรุปผลโดยการพัฒนาเป็นแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

2) วิเคราะห์ข้อมูลที่ได้ทำการศึกษาและรวบรวม แล้วทำการสรุปผลโดยการพัฒนาดัชนีแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

3) สรุปผลการประเมินแนวปฏิบัติที่ดีและดัชนีแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile จากองค์กรตัวอย่าง แล้วทำการสรุปข้อคิดเห็นและข้อเสนอแนะเพิ่มเติม



## 2. วิธีการดำเนินโครงการ

### 2.1 ศึกษาบริษัทตัวอย่าง

ดำเนินการศึกษาบริษัทตัวอย่าง คือ บริษัท โพรเฟสชั่นแนล เซอร์วิส โพรไวเดอร์ จำกัด (Professional Service Provider Co., Ltd.) โดยทำการสัมภาษณ์ผู้บริหาร และผู้จัดการ โครงการพัฒนาซอฟต์แวร์เกี่ยวกับข้อมูลทั่วไปของบริษัทและกระบวนการจัดการ โครงการพัฒนาซอฟต์แวร์ขององค์กรตามมาตรฐาน ISO/IEC 29110

### 2.2 กำหนดคำถามและออกแบบแบบสอบถาม

เป็นการจัดทำแบบสอบถามเพื่อใช้เป็นเครื่องมือช่วยในการสัมภาษณ์ผู้บริหารขององค์กรได้อย่างครบถ้วน

#### 2.2.1 โครงสร้างของแบบสอบถาม

##### 1) ส่วนที่ 1: ข้อมูลทั่วไป ประกอบด้วย

- (1) ข้อมูลทั่วไปขององค์กร
- (2) วัตถุประสงค์ในการพัฒนาองค์กร
- (3) Development Methodology ที่ใช้ในการพัฒนาซอฟต์แวร์
- (4) ปัจจัยที่ทำให้องค์กรตัดสินใจปรับปรุงกระบวนการพัฒนาซอฟต์แวร์

โดยเลือกใช้มาตรฐาน ISO/IEC 29110

(5) ปัญหาที่พบระหว่างการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ขององค์กร

(6) ปัจจัยที่มีผลต่อความสำเร็จในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ขององค์กร

##### 2) ส่วนที่ 2: ข้อมูลกระบวนการจัดการ โครงการพัฒนาซอฟต์แวร์ขององค์กร

ตามมาตรฐาน ISO/IEC 29110 ประกอบด้วย

(1) ขั้นตอนการดำเนินงานและเอกสารที่เกี่ยวข้องขององค์กร ที่สามารถบรรลุวัตถุประสงค์ของกระบวนการจัดการ โครงการพัฒนาซอฟต์แวร์ (Project Management Process) ตามมาตรฐาน ISO/IEC 29110

(2) ขั้นตอนการดำเนินงานและเอกสารที่เกี่ยวข้องขององค์กร ที่บรรลุวัตถุประสงค์ของกระบวนการพัฒนา/ติดตั้งซอฟต์แวร์ (Software Implementation Process) ตามมาตรฐาน ISO/IEC 29110

3) ส่วนที่ 3: ข้อมูลความต้องการคุณลักษณะของระบบที่จะสามารถนำมาใช้ประกอบการตรวจสอบและควบคุมคุณภาพโครงการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110

### 2.2.2 รูปแบบของคำถาม

ชนิดของแบบสอบถาม (Questionnaire Types) ที่ใช้ในการดำเนินโครงการครั้งนี้ มีการเลือกรูปแบบของคำถามเป็นคำถามแบบปลายเปิด (Open-ended Response Question) ซึ่งเป็นคำถามที่ให้อิสระต่อผู้ตอบสามารถตอบคำถามโดยใช้แนวความคิดของตนเอง โดยรูปแบบของคำถามนั้นมีรายละเอียดตามที่จัดไว้ในภาคผนวก ก

## 2.3 แหล่งข้อมูล

การดำเนินโครงการครั้งนี้เป็นการจัดทำรายงานเชิงพรรณนา (Descriptive Research) โดยการศึกษาหาข้อมูลทางด้านทฤษฎี การวิเคราะห์ การพัฒนาแนวปฏิบัติที่ดีและต้นแบบระบบ และทำการสรุปผลพร้อมข้อเสนอแนะ โดยแหล่งข้อมูลที่ศึกษาประกอบด้วย

2.3.1 แหล่งข้อมูลปฐมภูมิ (Primary Data) ได้มาจากการสัมภาษณ์ผู้บริหารของบริษัทตัวอย่าง ซึ่งในที่นี้ คือ บริษัท โปรเฟสชั่นแนล เซอร์วิส โพรไวเดอร์ จำกัด (Professional Service Provider Co., Ltd.)

2.3.2 แหล่งข้อมูลทุติยภูมิ (Secondary Data) ได้มาจากเว็บไซต์ และบทความ เป็นต้น

## 2.4 การรวบรวมข้อมูล

ในการดำเนินโครงการครั้งนี้มีการรวบรวมข้อมูลทั้งทางด้านวิชาการ และข้อมูลจากบริษัทตัวอย่าง โดยมีการเก็บรวบรวมข้อมูลดังนี้

2.4.1 ทำการรวบรวมทฤษฎี และเรื่องที่เกี่ยวข้องกับการพัฒนาซอฟต์แวร์ และระบบมาตรฐาน ISO/IEC 29110 จากเว็บไซต์ และบทความวิชาการ

2.4.2 จัดเตรียมแบบสอบถามเพื่อใช้ในการเก็บรวบรวมข้อมูลจากบริษัทตัวอย่าง

2.4.3 ดำเนินการเก็บรวบรวมข้อมูลจากบริษัทตัวอย่าง โดยใช้แบบสอบถามในการสัมภาษณ์ผู้บริหารขององค์กรเพื่อนำมาใช้ในการวิเคราะห์ การพัฒนาแนวปฏิบัติที่ดีและต้นแบบระบบ และทำการสรุปผลพร้อมข้อเสนอแนะ

## 2.5 การวิเคราะห์และสรุปผลการดำเนินโครงการ

2.5.1 วิเคราะห์ข้อมูลกระบวนการจัดการโครงการพัฒนาซอฟต์แวร์ขององค์กรตัวอย่างตามมาตรฐาน ISO/IEC 29110 แล้วทำการพัฒนาเป็นแนวปฏิบัติที่ดี และทำการสรุปผลพร้อมข้อเสนอแนะเพื่อเป็นข้อมูลแนวทางสำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยในการปรับปรุงกระบวนการดำเนินงานด้านการตรวจสอบและควบคุมคุณภาพการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110

2.5.2 วิเคราะห์ข้อมูลความต้องการคุณลักษณะของระบบที่จะสามารถนำมาใช้ประกอบการตรวจสอบและควบคุมคุณภาพโครงการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 จากองค์กรตัวอย่าง แล้วทำการพัฒนาต้นแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับผู้ที่มีความสนใจจะสามารถใช้เป็นข้อมูลแนวทางประกอบการพัฒนาเป็นระบบสำหรับใช้ตรวจสอบและควบคุมโครงการพัฒนาซอฟต์แวร์ขององค์กรให้เป็นไปตามมาตรฐาน 29110

2.5.3 สรุปผลการประเมินแนวปฏิบัติที่ดีและต้นแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile จากองค์กรตัวอย่าง แล้วทำการสรุปข้อคิดเห็นและข้อเสนอแนะเพิ่มเติม



## บทที่ 4

# ผลที่ได้จากการดำเนินโครงการ

### 1. ข้อมูลเบื้องต้นของบริษัทตัวอย่าง

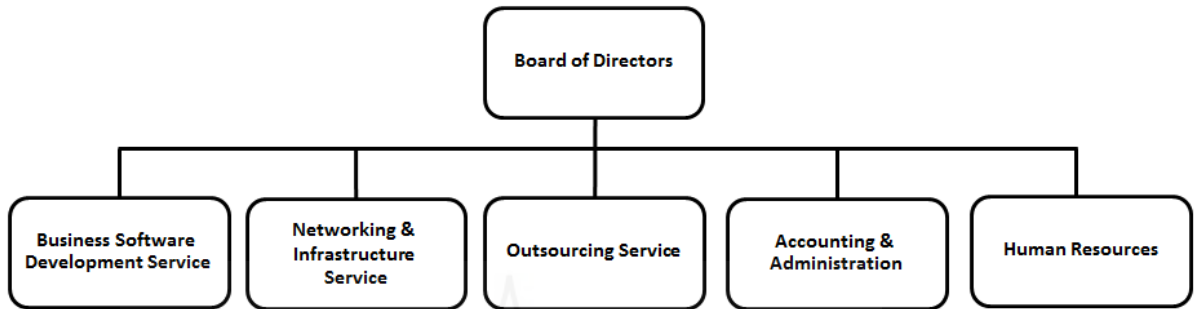
#### 1.1 ประวัติขององค์กร

บริษัท โพรเฟสชันแนล เซอร์วิส โพรไวเดอร์ จำกัด (Professional Service Provider Co., Ltd.) ก่อตั้งขึ้นในปีพ.ศ. 2547 โดยการรวมกลุ่มกันของผู้มีประสบการณ์ด้านให้คำปรึกษา ระบบเทคโนโลยีสารสนเทศ, ผู้เชี่ยวชาญวางระบบเทคโนโลยีสารสนเทศ, และผู้อบรมด้านระบบเทคโนโลยีสารสนเทศ ซึ่งเป็นบริษัทที่ได้รับการลงทุนและบริหารงานโดยคนไทย 100%

บริษัทฯ ดำเนินธุรกิจด้านระบบเทคโนโลยีสารสนเทศครบวงจร โดยให้บริการให้คำปรึกษา, วางระบบเครือข่าย (Network Infrastructure Implementation), พัฒนาซอฟต์แวร์ทางธุรกิจ (Business Software Development), บริการ Outsourcing โดยการนำเสนอ Solution แบบครบวงจรทั้งในส่วนของการให้บริการ และการจัดจำหน่ายผลิตภัณฑ์เครื่องคอมพิวเตอร์, อุปกรณ์ระบบเครือข่าย, และซอฟต์แวร์ลิขสิทธิ์ด้วยแนวความคิดการดำเนินธุรกิจที่มุ่งเน้นถึงความสำเร็จของลูกค้าการนำเสนอแนวทาง วิธีการ และผลิตภัณฑ์ที่เหมาะสม ที่ผ่านการเลือกสรร และทดสอบจากทีมงานผู้เชี่ยวชาญของบริษัทฯ แล้วว่าดีมีคุณภาพ

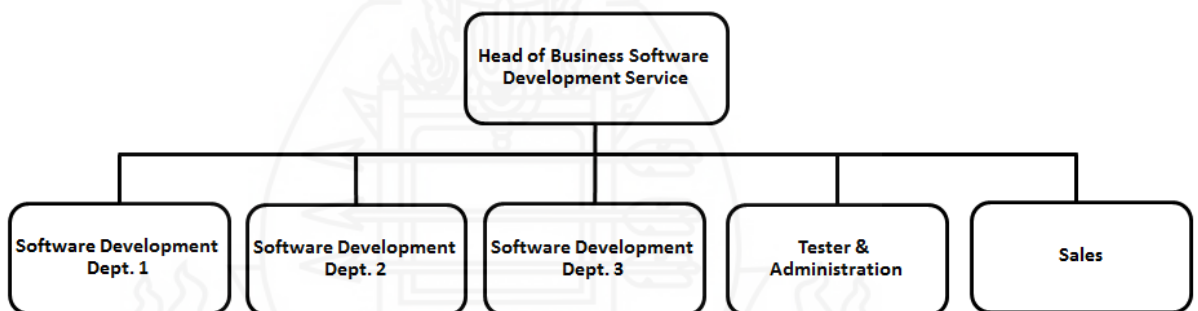
#### 1.2 โครงสร้างขององค์กร

โครงสร้างองค์กรของบริษัท โพรเฟสชันแนล เซอร์วิส โพรไวเดอร์ จำกัด ประกอบด้วยหน่วยงานต่างๆ ดังนี้



ภาพที่ 4.1 โครงสร้างองค์กรของบริษัท โพรเฟสชันแนล เซอร์วิส โพรไวเดอร์ จำกัด

โครงสร้างองค์กรของหน่วยงาน Business Software Development Service ของบริษัท โพรเฟสชันแนล เซอร์วิส โพรไวเดอร์ จำกัด มีการทำงานเป็นแบบ Project Base คือ เมื่อมีโครงการใหม่เข้ามาจะมีการมอบหมายงานให้บุคลากรทำงานร่วมกันเป็นทีมโครงการ โดยมีโครงสร้างหน่วยงาน ดังนี้



ภาพที่ 4.2 โครงสร้างองค์กรของหน่วยงาน Business Software Development Service ของบริษัท โพรเฟสชันแนล เซอร์วิส โพรไวเดอร์ จำกัด

เมื่อมีโครงการใหม่เข้ามาจะมีการมอบหมายงานให้บุคลากรทำงานร่วมกันเป็นทีมโครงการ ประกอบด้วยการดำเนินงานในบทบาทต่างๆ ส่วนที่เกี่ยวข้องกับการดำเนินโครงการพัฒนาซอฟต์แวร์ ดังนี้

1. Project Manager ทำหน้าที่เป็นผู้นำในการวางแผนและจัดการโครงการ, มอบหมายงานให้กับบุคลากร, ควบคุมโครงการให้สำเร็จได้ตามขอบข่ายงาน, ระยะเวลา, และต้นทุนที่กำหนดไว้

2. Team Leader/System Analyst ทำหน้าที่เป็นผู้นำในการวิเคราะห์ความต้องการซอฟต์แวร์และจัดทำเอกสารคุณลักษณะความต้องการซอฟต์แวร์, ออกแบบและจัดทำเอกสารโครงสร้างฐานข้อมูลและซอฟต์แวร์, ควบคุมงานทางด้านเทคนิคในส่วนของ การติดตั้งและบำรุงรักษาและจัดทำคู่มือระบบทางเทคนิค (System Manual) สำหรับซอฟต์แวร์ที่พัฒนาเสร็จแล้ว และผ่านการทดสอบเรียบร้อยแล้ว

3. Programmer ทำหน้าที่พัฒนาซอฟต์แวร์ตามที่ได้รับมอบหมาย

4. Tester ทำหน้าที่ทดสอบซอฟต์แวร์และบันทึกผลการทดสอบ

5. Administration Staff ทำหน้าที่จัดทำคู่มือผู้ใช้ระบบ (User Manual) สำหรับซอฟต์แวร์ที่พัฒนาเสร็จแล้วและผ่านการทดสอบจากลูกค้าเรียบร้อยแล้ว

### 1.3 Development Methodology ที่องค์กรใช้ในการพัฒนาซอฟต์แวร์

องค์กรใช้ Agile Development Methodology ในการพัฒนาซอฟต์แวร์

1.4 ปัจจัยที่ทำให้องค์กรตัดสินใจปรับปรุงกระบวนการพัฒนาซอฟต์แวร์โดยเลือกใช้มาตรฐาน ISO/IEC 29110

#### 1.4.1 ปัจจัยภายใน

1) เนื่องจากเป็นนโยบายของผู้บริหารที่ต้องการยกระดับความน่าเชื่อถือองค์กรในด้านการให้บริการพัฒนาซอฟต์แวร์ในตลาดอุตสาหกรรมซอฟต์แวร์ในประเทศไทย

2) เพื่อลดต้นทุนในด้านการจัดการโครงการพัฒนาซอฟต์แวร์เพราะในแต่ละขั้นตอนการดำเนินโครงการที่สำคัญจะมีการจัดทำและจัดเก็บเอกสารไว้อย่างเป็นระบบ

3) เพื่อลดระยะเวลาในการดำเนินโครงการพัฒนาซอฟต์แวร์เพราะผู้เกี่ยวข้องกับการดำเนินโครงการสามารถทำงานร่วมกันได้อย่างต่อเนื่องโดยการศึกษาจากเอกสารระบบที่ได้รับการจัดทำขึ้นได้อย่างสะดวกรวดเร็ว

4) เพื่อเพิ่มขีดความสามารถในการบำรุงรักษาและพัฒนาซอฟต์แวร์ต่อ ยอดให้กับลูกค้าได้อย่างต่อเนื่อง เพราะในแต่ละขั้นตอนการดำเนินโครงการที่สำคัญจะมีการจัดทำและจัดเก็บเอกสารไว้อย่างเป็นระบบ

5) เพื่อยกระดับคุณภาพของซอฟต์แวร์ที่ได้รับการพัฒนาขึ้น ซึ่งก็จะช่วยเป็นการเพิ่มความเชื่อมั่นให้กับลูกค้าด้วยเช่นกัน

#### 1.4.2 ปัจจัยภายนอก

1) จากสภาวะการแข่งขันทางการให้บริการพัฒนาซอฟต์แวร์ทั้งในประเทศและต่างประเทศ จึงส่งผลให้องค์กรจะต้องยกระดับคุณภาพการให้บริการพัฒนาซอฟต์แวร์ให้เป็นที่น่าเชื่อถือและยอมรับได้ทั้งภายในประเทศและระดับสากล

### 1.5 ระยะเวลาในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ขององค์กรตามมาตรฐาน ISO/IEC 29110 จนกระทั่งผ่านการรับรอง

บริษัท โพรเฟสชันแนล เซอร์วิส โพรไวเดอร์ จำกัด ใช้ระยะเวลาในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ขององค์กรตามมาตรฐาน ISO/IEC 29110 จนกระทั่งผ่านการรับรองเป็นระยะเวลาประมาณ 4 เดือน

### 1.6 ระดับมาตรฐาน ISO/IEC 29110 ที่บริษัทผ่านการรับรอง

บริษัท โพรเฟสชันแนล เซอร์วิส โพรไวเดอร์ จำกัด ผ่านการรับรองด้านการให้บริการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

### 1.7 ปัญหาที่พบระหว่างการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ขององค์กร

1.7.1 ปัญหาในช่วงระยะเวลาที่กำลังปรับปรุงกระบวนการดำเนินงานให้บรรลุ Objectives ตามคุณลักษณะตามมาตรฐาน ISO/IEC 29110 ซึ่งพนักงานยังไม่มี ความคุ้นเคยในการจัดทำเอกสารเพิ่มเติมอย่างครบถ้วน เช่น ในขั้นตอนการจัดทำเอกสารต่างๆ ยังไม่มีการบันทึกผลการตรวจสอบความครบถ้วนและความถูกต้องของเอกสารที่จัดทำไว้ในเอกสาร Verification & Validation Results เป็นต้น

1.7.2 ปัญหาด้านความรู้ความเข้าใจในคุณลักษณะตามมาตรฐาน ISO/IEC 29110 ของพนักงาน เช่น การจัดทำเอกสารในบางขั้นตอนการทำงานมีรายละเอียดยังไม่ครบถ้วนครอบคลุม Objectives ของมาตรฐาน ได้แก่ การจัดทำเอกสาร Project Plan ยังมีรายละเอียดเนื้อหาไม่ครบถ้วนครอบคลุม Objectives ของมาตรฐาน

### 1.8 ปัจจัยที่มีผลต่อความสำเร็จในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ขององค์กร

1.8.1 ได้รับการสนับสนุนจากผู้บริหารในด้านต่างๆ เช่น งบประมาณ บุคลากร

1.8.2 Project Manager/Business Analyst เป็นผู้มีประสบการณ์ในด้านมาตรฐาน ISO มาก่อน จึงสามารถนำทักษะความรู้มาประยุกต์ใช้ในการปรับปรุงกระบวนการดำเนินโครงการพัฒนาซอฟต์แวร์ให้บรรลุวัตถุประสงค์ตามคุณลักษณะตามมาตรฐาน ISO/IEC 29110 ได้อย่างสะดวกยิ่งขึ้น

### 1.9 ประโยชน์ที่องค์กรได้รับเมื่อผ่านการรับรองมาตรฐาน ISO/IEC 29110

1.9.1 ได้รับความเชื่อถือในด้านความสามารถและคุณภาพในการดำเนินโครงการพัฒนาซอฟต์แวร์จากลูกค้าที่มีอยู่เดิมเพิ่มมากขึ้น



1.9.2 เพิ่มโอกาสในการเพิ่มลูกค้าใหม่จากความน่าเชื่อถือในใบรับรองมาตรฐาน ISO/IEC 29110

1.9.3 สามารถเพิ่มอัตราค่าบริการจากการที่มีใบรับรองมาตรฐาน ISO/IEC 29110 ส่งผลให้รายได้ขององค์กรเพิ่มขึ้น

1.9.4 การดำเนินโครงการพัฒนาซอฟต์แวร์มีการทำงานเป็นระบบ, มีเอกสารอ้างอิงในทุกขั้นตอนการทำงาน, และช่วยให้ลดระยะเวลาในการดำเนินโครงการ

1.9.5 ช่วยเพิ่มประสิทธิภาพในการส่งมอบงานแต่ละ Phase การทำงานได้เร็วขึ้น

1.9.6 เพิ่มคุณภาพของซอฟต์แวร์และลดระยะเวลาในการแก้ไขปัญหาซอฟต์แวร์ เพราะมีการบันทึกสถานะการทดสอบอย่างละเอียดสำหรับใช้ประกอบการแก้ไขซอฟต์แวร์ได้ตรงจุดและเร็วขึ้น

1.9.7 สามารถมอบหมายงานหรือโอนงานให้พนักงานใหม่หรือพนักงานคนอื่น ดำเนินงานแทนพนักงานคนเก่าที่ลาออกได้อย่างสะดวกเนื่องจากมีการจัดทำเอกสารอ้างอิงไว้อย่างละเอียด

### บทวิเคราะห์

การจัดโครงสร้างองค์กรของหน่วยงาน Business Software Development Service ของบริษัท โพรเฟสชั่นแนล เซอร์วิส โพรไวเดอร์ จำกัด มีการทำงานเป็นแบบ Project Base จึงทำให้มีความยืดหยุ่นในการมอบหมายงาน โครงการให้บุคลากร โดยบุคลากรในแต่ละส่วนงานสามารถทำงานร่วมกันได้กับส่วนงานอื่นตามการมอบหมายงาน ซึ่งจะส่งผลให้บุคลากรไม่เบื่อหน่ายที่จะต้องทำงานซ้ำแบบเดิมตลอดเวลา เพราะจะได้รับมอบหมายงาน โครงการใหม่และการได้ร่วมงานกันกับบุคลากรในส่วนงานอื่นได้อย่างหลากหลาย และจะช่วยส่งผลให้เกิดการแลกเปลี่ยนประสบการณ์และทักษะความรู้ร่วมกัน ได้ดียิ่งขึ้น

ปัจจัยแห่งความสำเร็จในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ขององค์กรนี้ที่สำคัญที่สุด 2 ปัจจัย คือ การได้รับการสนับสนุนจากผู้บริหารอย่างเต็มที่ และการที่มีบุคลากรที่ทำหน้าที่เป็นผู้นำโครงการ ในบทบาท Project Manager/Business Analyst ที่มีประสบการณ์ เป็นส่วนสำคัญที่สามารถช่วยองค์กรนี้ประสบความสำเร็จในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ได้ ดังนั้นอย่างน้อยที่สุด 2 ปัจจัยนี้ก็น่าจะเป็นปัจจัยที่สำคัญสำหรับองค์กรอื่นๆ เช่นกันที่จะสามารถขับเคลื่อนทีมงานโครงการพัฒนาซอฟต์แวร์ให้สามารถปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ได้อย่างสะดวกและรวดเร็วยิ่งขึ้น

จะเห็นได้ว่าประโยชน์ที่องค์กรได้รับนั้น นอกจากการผ่านการรับรองมาตรฐาน ISO/IEC 29110 จะส่งผลให้การทำงานในทีมเป็นระบบมากยิ่งขึ้นและมีเอกสารอ้างอิงในทุกขั้นตอนการทำงานเพื่อจะสามารถมอบหมายงานหรือโอนงานให้พนักงานคนอื่นหรือพนักงานใหม่ทำงานแทนพนักงานคนเก่าได้อย่างสะดวกยิ่งขึ้นแล้ว ใรับรองมาตรฐาน ISO/IEC 29110 ยังช่วยสร้างความน่าเชื่อถือต่อองค์กรให้กับทั้งลูกค้าที่มีอยู่เดิมและเพิ่ม โอกาสในการเพิ่มฐานลูกค้าใหม่ และยังมีส่วนเพิ่มรายได้ให้กับองค์กรอีกด้วย

## 2. ผลจากการศึกษา, บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทย

### 2.1 Activities การดำเนินงานและเอกสารที่เกี่ยวข้องขององค์กรที่สามารถบรรลุ Objective ของ Project Management Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์) ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile มีดังนี้

#### 2.1.1 PM.1: Project Planning (การวางแผนโครงการ)

1) ผู้รับผิดชอบในโครงการสำหรับ Activity นี้และบทบาทความเกี่ยวข้องมี ดังนี้

(1) Project Manager ทำหน้าที่ทบทวนวัตถุประสงค์และขอบข่ายงานร่วมกับลูกค้า และวางแผนการดำเนินโครงการร่วมกับ Team Leader/System Analyst โดยการกำหนดข้อมูล ดังนี้

ซอฟต์แวร์

ก. การจัดเตรียมบุคลากรและทรัพยากรต่างๆ สำหรับการพัฒนา

ข. การกำหนดรายการดำเนินงานและระยะเวลาดำเนินงาน

ค. การประเมินต้นทุนที่เกี่ยวข้องกับการดำเนินโครงการ ได้แก่ ค่าจ้างบุคลากร, ค่าอุปกรณ์สำหรับใช้ประกอบการพัฒนาซอฟต์แวร์

ง. การประเมินความเสี่ยงในการดำเนินโครงการพัฒนาซอฟต์แวร์

จ. การกำหนดกลยุทธ์ในการควบคุมรูปแบบและเวอร์ชันของเอกสารที่มีการเปลี่ยนแปลง

ฉ. การกำหนดรายละเอียดการส่งมอบงาน

ช. การกำหนดสถานที่จัดเก็บข้อมูลโครงการ และสถานที่สำรองข้อมูลโครงการ

ซ. จัดทำเอกสาร Project Plan

(2) Team Leader/System Analyst ทำหน้าที่ทบทวนวัตถุประสงค์และขอบข่ายงานและวางแผนการดำเนินโครงการร่วมกับ Project Manager

(3) Project Manager นำเสนอเอกสาร Project Plan ร่วมกับลูกค้า (Customer) เพื่อให้ลูกค้าลงนามลงนามยอมรับเอกสาร Project Plan

2) เอกสารที่เกี่ยวข้องกับ Activity นี้มี ดังนี้

(1) Input Products

ก. สัญญาหรือเอกสารยืนยันการว่าจ้างจากลูกค้า

ข. เอกสาร Project Timeframe (ตัวอย่างดังภาพที่ 4.20-4.22)

ค. เอกสารการวางแผนงานภายในองค์กร

(2) Output Products

ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15)

ข. เอกสาร Verification & Validation Results (ตัวอย่างดังภาพที่ 4.16-4.17)

ค. เอกสาร Meeting Record (บันทึกถายงานการประชุม) (ตัวอย่างดังภาพที่ 4.18-4.19)

3) รายละเอียดเพิ่มเติมมี ดังนี้

(1) ควรให้ลูกค้าลงนามรับรองเอกสาร Project Plan เพื่อเป็นการยืนยันการยอมรับอย่างเป็นทางการเป็นลายลักษณ์อักษร

**บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity PM.1: Project Planning (การวางแผนโครงการ)**

Activity PM.1: Project Planning (การวางแผนโครงการ) เป็นการดำเนินงานด้านการทบทวนขอบข่ายงาน วางแผนงาน และจัดทำเอกสาร Project Plan ขึ้น โดยเอกสาร Project Plan จะต้องได้รับการทบทวนและยอมรับจากลูกค้าก่อนเริ่มดำเนินโครงการ ซึ่งบทบาทผู้รับผิดชอบโครงการที่สำคัญประกอบด้วย Project Manager และ Team Leader/System Analyst โดยแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 สำหรับ Activity PM.1: Project Planning (การวางแผนโครงการ) และตัวอย่างเอกสารที่เป็นผลจากการดำเนินงาน Activity นี้ (Output Products) มีรายละเอียด ดังนี้

ตารางที่ 4.1 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ

Activity PM.1: Project Planning (การวางแผน โครงการ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p><b>1. จัดทำเอกสาร Project Plan</b> ซึ่งเป็นข้อมูลสรุปภาพรวมของโครงการสำหรับการยืนยันขอบข่ายงาน, รายการดำเนินงาน, ระยะเวลาดำเนินงาน, การวางแผนทรัพยากรต่างๆ, และข้อมูลที่เกี่ยวข้องต่อการดำเนินโครงการพัฒนาซอฟต์แวร์ โดยเอกสาร Project Plan ควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</p> <ul style="list-style-type: none"> <li>○ ชื่อโครงการ, ชื่อโปรแกรม/ชื่อระบบ, ชื่อลูกค้า, รหัสเอกสาร</li> <li>○ วัตถุประสงค์ (Purpose) ของโครงการ</li> <li>○ ขอบข่ายงานการพัฒนา (Scope of work)</li> <li>○ ข้อกำหนดในการพัฒนาระบบ</li> <li>○ Stand Software Development Life Cycle Model ที่ใช้ในการพัฒนาระบบ</li> <li>○ กลยุทธ์/รูปแบบการควบคุมการเปลี่ยนแปลงเวอร์ชันของเอกสารโครงการ (Version Control Strategy)</li> <li>○ การกำหนดสถานที่จัดเก็บข้อมูลโครงการ (Project Repository)</li> <li>○ การกำหนดสถานที่สำรองข้อมูลโครงการ (Project Repository)</li> </ul>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- สัญญาหรือเอกสารยืนยันการว่าจ้างจากลูกค้า</li> <li>- เอกสาร Project Timeframe</li> <li>- เอกสารการวางแผนงานภายในองค์กร</li> <li>- เอกสาร Project Plan ฉบับร่าง</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan เวอร์ชัน Baseline ที่ได้รับการลงนาม</li> <li>- ยืนยันจากลูกค้า</li> <li>- เอกสาร Verification &amp; Validation Results</li> <li>- เอกสารรายงานการประชุม (Meeting Record)</li> </ul>	<ul style="list-style-type: none"> <li>- Project Manager ทบทวนขอบข่ายงานร่วมกับลูกค้า</li> <li>- Project Manager เป็นผู้จัดทำเอกสาร Project Plan โดยวางแผนโครงการและกำหนดรายละเอียดในเอกสาร Project Plan ร่วมกับ Team Leader/System Analyst</li> <li>- Team Leader/System Analyst บันทึกผลการ Verify เพื่อตรวจสอบเอกสาร Project Plan ในเอกสาร Verification &amp; Validation Results</li> <li>- Project Manager นำเสนอเอกสาร Project Plan ต่อ</li> </ul>

## ตารางที่ 4.1 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<ul style="list-style-type: none"> <li>○ แผนการดำเนินงานที่ประกอบด้วย รายการดำเนินงานและระยะเวลา ดำเนินงาน (Project Timeframe)</li> <li>○ ทีมงานดำเนินโครงการ (Project Resource)</li> <li>○ การส่งมอบงาน (Delivery Instruction)</li> <li>○ การประเมินต้นทุนในการดำเนิน โครงการ (Estimated Effort and Cost)</li> <li>○ การประเมินความเสี่ยงในการดำเนิน โครงการ (Identification of Project Risks)</li> <li>○ มีการบันทึกประวัติการแก้ไข (Revision History)</li> <li>○ ส่วนลงนามยืนยันจากลูกค้า</li> </ul>		<ul style="list-style-type: none"> <li>- ลูกค้า</li> <li>- Project Manager หรือ Team Leader/System Analystบันทึกการ ประชุมใน เอกสารรายงานการ ประชุม (Meeting Record)</li> <li>- Project Manager แก้ไขเอกสาร Project Plan ตามที่มี การบันทึกรายงาน การประชุมร่วมกับ ลูกค้า Project Managerนำ เอกสาร Project Plan เวอร์ชันที่ได้รับความ เห็นชอบจากลูกค้าให้ ลูกค้าลงนามยืนยัน และกำหนดเป็นเวอร์ ชัน Baseline</li> </ul>

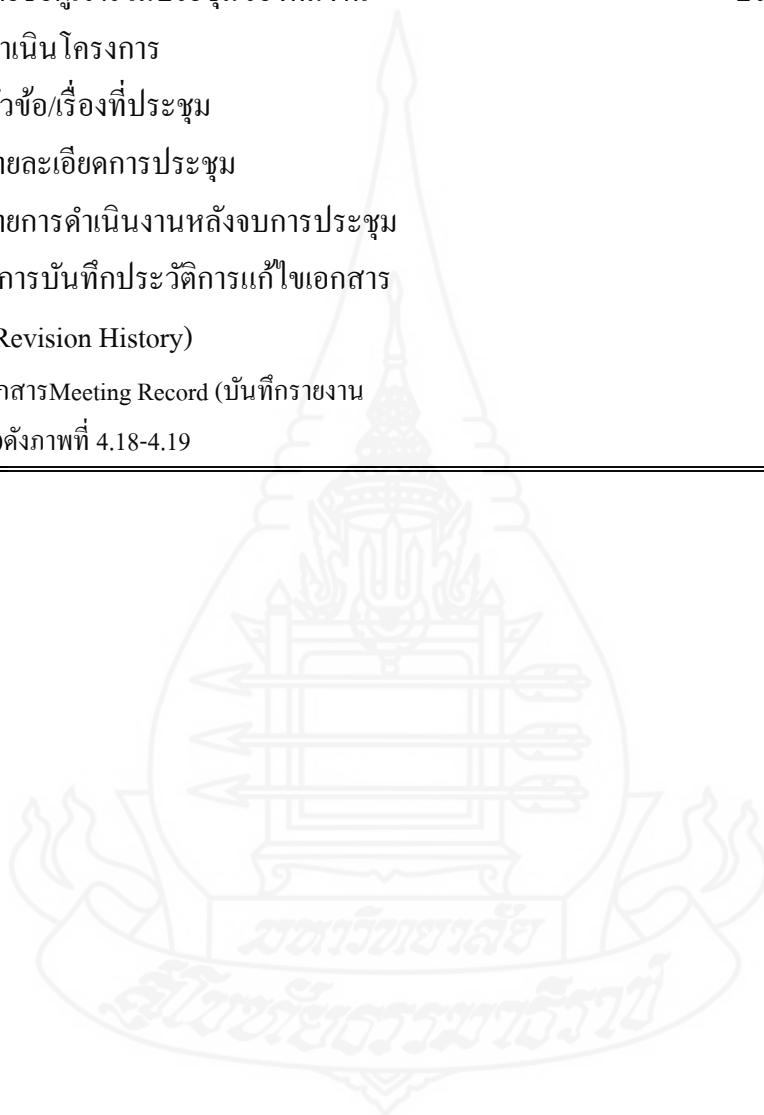
\*ตัวอย่างเอกสารProject Planดังภาพที่ 4.3-4.15

ตารางที่ 4.1 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p><b>2. จัดทำเอกสาร Verification &amp; Validation Results</b>(ผลการตรวจสอบความถูกต้อง) เพื่อการบันทึกและยืนยันผลการตรวจสอบความถูกต้องในการจัดทำเอกสารProject Planซึ่งเป็นการตรวจสอบภายในทีม ดำเนินโครงการ โดยเอกสาร Verification Results ควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</p> <ul style="list-style-type: none"> <li>○ ชื่อโครงการ, ชื่อลูกค้า, รหัสเอกสาร</li> <li>○ ชื่อผู้ดำเนินการตรวจสอบความถูกต้องในแต่ละขั้นตอน</li> <li>○ ชื่อขั้นตอนที่ทำการตรวจสอบ</li> <li>○ รายการดำเนินงานของขั้นตอนที่ตรวจสอบ</li> <li>○ ผลการตรวจสอบ</li> <li>○ มีการบันทึกประวัติการแก้ไขเอกสาร (Revision History)</li> </ul>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan ฉบับร่าง</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Verification &amp; Validation Results</li> </ul>	<ul style="list-style-type: none"> <li>- Project Manager เป็นผู้จัดทำเอกสาร Verification &amp; Validation Results</li> <li>- Team Leader/System Analystบันทึกผลการ Verify เพื่อตรวจสอบเอกสาร Project Plan ในเอกสารVerification &amp; Validation Results</li> </ul>
<p>*ตัวอย่างเอกสาร Verification &amp; Validation Results ดังภาพที่ 4.16-4.17</p>		
<p><b>3. จัดทำเอกสาร Meeting Record (บันทึก</b> <b>รายงานการประชุม)</b>สำหรับบันทึกผลการประชุมในการนำเสนอเอกสาร Project Plan ต่อลูกค้าโดยเอกสาร Meeting Record ควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</p> <ul style="list-style-type: none"> <li>○ รหัสเอกสาร</li> <li>○ รายชื่อผู้เข้าร่วมประชุมของลูกค้า</li> </ul>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan ฉบับร่าง</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Meeting Record (บันทึก รายงานการประชุม)</li> </ul>	<ul style="list-style-type: none"> <li>- Project Manager หรือ Team Leader/System Analystเป็นผู้จัดทำเอกสาร Meeting Record (บันทึกรายงานการ</li> </ul>

## ตารางที่ 4.1 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<ul style="list-style-type: none"> <li>○ รายชื่อผู้เข้าร่วมประชุมของทีมงาน     ดำเนินโครงการ</li> <li>○ หัวข้อ/เรื่องที่ประชุม</li> <li>○ รายละเอียดการประชุม</li> <li>○ รายการดำเนินงานหลังจบการประชุม</li> <li>○ มีการบันทึกประวัติการแก้ไขเอกสาร     (Revision History)</li> </ul>		ประชุม)
<p>*ตัวอย่างเอกสาร Meeting Record (บันทึกรายงานการประชุม) ดังภาพที่ 4.18-4.19</p>		





รหัสเอกสาร  
PP-TMS-BPP-54-03A-290854

# PROJECT PLAN

ชื่อโครงการ, ชื่อโปรแกรม/ระบบ

PROJECT NAME: \_\_\_\_\_

ชื่อลูกค้า

**Customer Name:** \_\_\_\_\_

PROJECT ID: TMS-BPP-54

Version: 3.0    Date: 29 สิงหาคม 2554

รหัสโครงการ

เวอร์ชันเอกสาร

ตารางสำหรับการ  
ลงนาม

	Full Name	Role	Date	Signature
Created by:	คุณ _____	ProjectManager	29 สิงหาคม 2554	
Reviewed by:	คุณ _____	Team Leader/SA	29 สิงหาคม 2554	
Approved by:	คุณ _____	Customer	29 สิงหาคม 2554	

ภาพที่ 4.3 ตัวอย่างเอกสาร Project Plan ส่วนที่ 1

ส่วนบันทึกประวัติการแก้ไขและ  
ควบคุมเวอร์ชันของเอกสาร

Revision History:

Version	Date	Author	Reviewer	Description
0.1	10 กรกฎาคม 2554	คุณ _____		Draft
1.0	14 กรกฎาคม 2554	คุณ _____	คุณ _____	Initial
2.0	18 กรกฎาคม 2554	คุณ _____	คุณ _____	Base line ( <a href="#">MR-TMS-BPP-54-01A-140754</a> )
3.0	29 สิงหาคม 2554	คุณ _____	คุณ _____	Base line ( <a href="#">MR-TMS-BPP-54-03A-260854</a> )

ภาพที่ 4.4 ตัวอย่างเอกสาร Project Plan ส่วนที่ 2

วัตถุประสงค์

## 1. วัตถุประสงค์ (Purpose)

ทางบริษัท ..... มีความประสงค์พัฒนาโปรแกรม ..... ดังนี้

- .....
- .....

ขอบข่ายงาน

## 2. ขอบข่ายงานการพัฒนาระบบ (Scope of work)

โปรแกรม ..... ประกอบด้วย Module การทำงาน, รายงาน, และคุณลักษณะดังนี้

1 Module ระบบการบริหารจัดการผู้ใช้, การสืบค้น และติดตามอุปกรณ์/เครื่องมือ ประกอบด้วยคุณลักษณะดังนี้

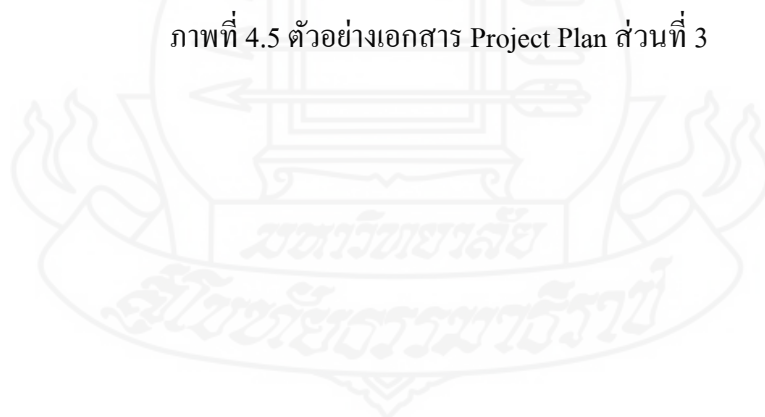
### 1.1 การทำงานของ Role Normal User

- 1.1.1 ค้นหาเอกสาร.....
- 1.1.2 พิมพ์.....
- 1.1.3 ค้นหา.....

### 1.2 การทำงานของ Role Power User

- 1.2.1 การบันทึกรับอุปกรณ์/เครื่องมือเข้าคลัง (Stock In) มีคุณลักษณะ ดังนี้
  - 1.2.1.1 ค้นหาข้อมูลรายชื่ออุปกรณ์/เครื่องมือเดิม หรือเพิ่มอุปกรณ์/เครื่องมือใหม่เข้าระบบ
  - 1.2.1.2 กรอกจำนวนรับอุปกรณ์/เครื่องมือ เข้าคลัง
  - 1.2.1.3 ในฟอร์มการกรอกรับอุปกรณ์/เครื่องมือ เข้าคลัง ควรประกอบด้วยข้อมูล ดังนี้
    - 1.2.1.3.1 รหัสอุปกรณ์/เครื่องมือ
    - 1.2.1.3.2 ชื่ออุปกรณ์/เครื่องมือ
    - 1.2.1.3.3 ระบบแสดงจำนวนคงเหลือล่าสุดก่อนรับ โดยอัตโนมัติ

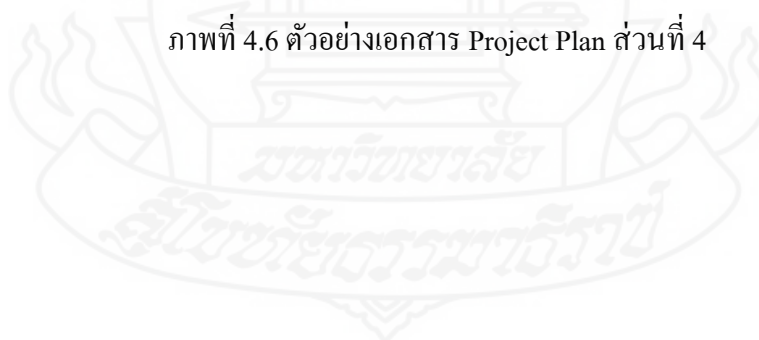
ภาพที่ 4.5 ตัวอย่างเอกสาร Project Plan ส่วนที่ 3

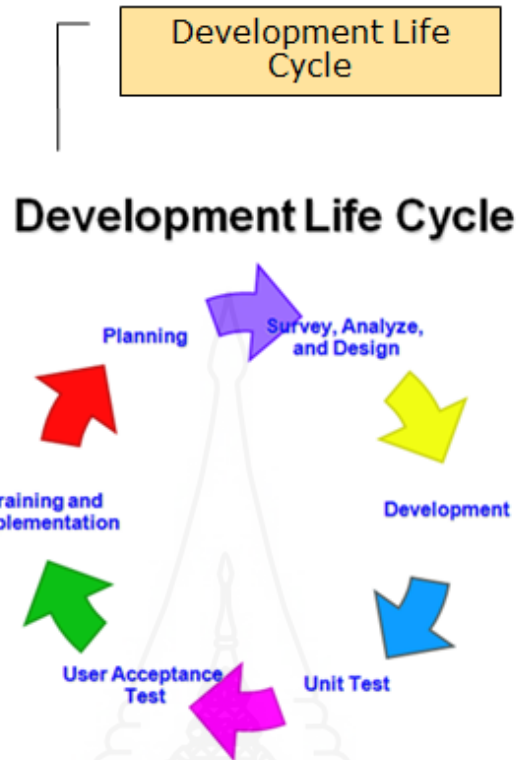


## 3 มาตรฐานการพัฒนาระบบ

- มาตรฐานในการออกแบบและพัฒนาระบบ เป็นรูปแบบสถาปัตยกรรม 3-Tier ประกอบด้วย 3 ส่วน ดังนี้
  - Presentation Tier ออกแบบรูปแบบ Web-based Application โดยแยกส่วนการแสดงผล User Interface และ Business Logic โปรแกรมออกจากกัน
  - Business Logic Tier ออกแบบโดยอ้างอิงหลักการออกแบบเชิงวัตถุซึ่งอาจใช้รูปแบบของ Class ช่วยในการออกแบบให้เป็นไปในแนวทางเพื่อนำไปสู่การนำกลับมาใช้ในงานใหม่ได้
  - Data Tier ออกแบบโดยมีข้อกำหนดดังนี้
    - ออกแบบฐานข้อมูลแบบ ODBMS หรือ ORDBMS หรือ RDBMS
    - ออกแบบทั้งใน Conceptual Data Model และ Physical Data Model
- ชุดเอกสารอ้างอิงระบบ
  - เอกสารข้อกำหนดความต้องการของผู้ใช้ระบบ
    - ขอบเขตการทำงานของระบบงาน (Requirements Checklist)
    - กิจกรรมขั้นตอนการทำงาน (Activity Diagram/Workflow)
    - แบบตัวอย่างหน้าจอ (Screen Layout)
    - แบบตัวอย่างรายงาน (Report Layout)
    - แผนภาพ Deployment Diagram
    - ข้อกำหนดเสริมของระบบ
  - เอกสารระบบงาน (System Document)
    - แบบตัวอย่างหน้าจอ (Screen Layout)

ภาพที่ 4.6 ตัวอย่างเอกสาร Project Plan ส่วนที่ 4





ภาพที่ 4.7 ตัวอย่างเอกสาร Project Plan ส่วนที่ 5



กลยุทธ์/รูปแบบการควบคุมการเปลี่ยนแปลงเวอร์ชันของเอกสาร  
โครงการ (Version Control Strategy)

4 Version Control Strategy

การจัดเก็บเอกสารโดยวิธีการกำหนดมาตรฐานการตั้งชื่อของเอกสาร โดยมีหลักการตั้งชื่อเอกสาร ดังนี้

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
X	X	X	-	X	X	X	-	X	X	X	-	X	X	X	-	X	X	X	X	X

- ความหมาย**
- หลักที่ 1-3 หมายถึง ชื่อเอกสาร  
กรณีชื่อย่อซ้ำกันจะสามารถเพิ่มหลักที่ 3 (optional)
  - หลักที่ 4-6 หมายถึง ชื่อระบบ
  - หลักที่ 7-9 หมายถึง ชื่อของบริษัทลูกค้า
  - หลักที่ 10-11 หมายถึง ปีพ.ศ. ที่เริ่มงาน
  - หลักที่ 12-13 หมายถึง Running ครั้งที่ต่อเนื่องจนจบโครงการ
  - หลักที่ 14 หมายถึง D หมายถึง Draft, A หมายถึง Actual (เอกสารชุดเป็นทางการ)
  - หลักที่ 15-20 หมายถึง วัน/เดือน/ปีพ.ศ. ที่บันทึกเอกสาร

ตัวอย่างชื่อไฟล์	ชื่อเอกสาร	ชื่อระบบ	ชื่อบริษัทลูกค้า	ปีพ.ศ.ที่เริ่มงาน	Running ครั้งที่ต่อเนื่องจนจบโครงการ	D (Draft), A (Actual)	วัน/เดือน/ปีพ.ศ. ที่บันทึกเอกสาร
PP-TMS-BPP-54-01A-140754	PP (Project Plan)	TMS (Tools)	BPP (Customer)	2554	01	A	140754

ภาพที่ 4.8 ตัวอย่างเอกสาร Project Plan ส่วนที่ 6

สถานที่จัดเก็บข้อมูลโครงการ  
(Project Repository)

- การจัดเก็บเอกสาร จะจัดเก็บที่เครื่อง Server ส่วนกลาง โดยผู้ใช้ระบบสามารถเข้าใช้งานได้ที่

<\\172.16.0.2\Biz2010\Sales\Customers\ CustomerName\ProjectName\ProjectSub Folders>

โดยมี folder ที่จัดเก็บเอกสารแบ่งตามประเภทของเอกสารดังนี้

- 1. PO
- 2. Project Plan Document
- 3. Change Request
- 4. Sign off Letter
- 5. Minute of Meeting
- 6. Correction Register
- 7. Project Timeframe
- 8. Verification & Validation Results
- 9. SRS
- 10\_16. Manual
- 11. Design Document
- 12. Traceability Record
- 13.-15. Test Document
- CD
- Graphic Design
- Info for Customer
- Info from Customer
- Internal
- Kick off Meeting
- Quotation\_Proposal
- Source Code & Database
- Waiting Information

ชื่อ Folder	การจัดเก็บข้อมูล
1. PO	เก็บข้อมูล/เอกสารใบสั่งซื้อ หรือเอกสารยืนยันการว่าจ้างจากลูกค้า
2. Project Plan Document	เก็บข้อมูล/เอกสารการ Project Plan Document
3. Change Request	เก็บข้อมูล/เอกสารการเปลี่ยนแปลง Requirement
4. Sign off Letter	เก็บข้อมูล/เอกสารการส่งมอบงานแต่ละงวด ทุกงวด (Acceptance Record)
5. Minute of Meeting	เก็บเอกสารรายงานการประชุม (Meeting Record) การประชุมกับลูกค้าทุกครั้ง
6. Correction Register	เก็บข้อมูล/เอกสารการเปลี่ยนแปลง Correction Register
7. Project Timeframe	เก็บข้อมูล/เอกสารการ Project Timeframe
8. Verification & Validation Result	เก็บข้อมูล/เอกสารการ Verification & Validation

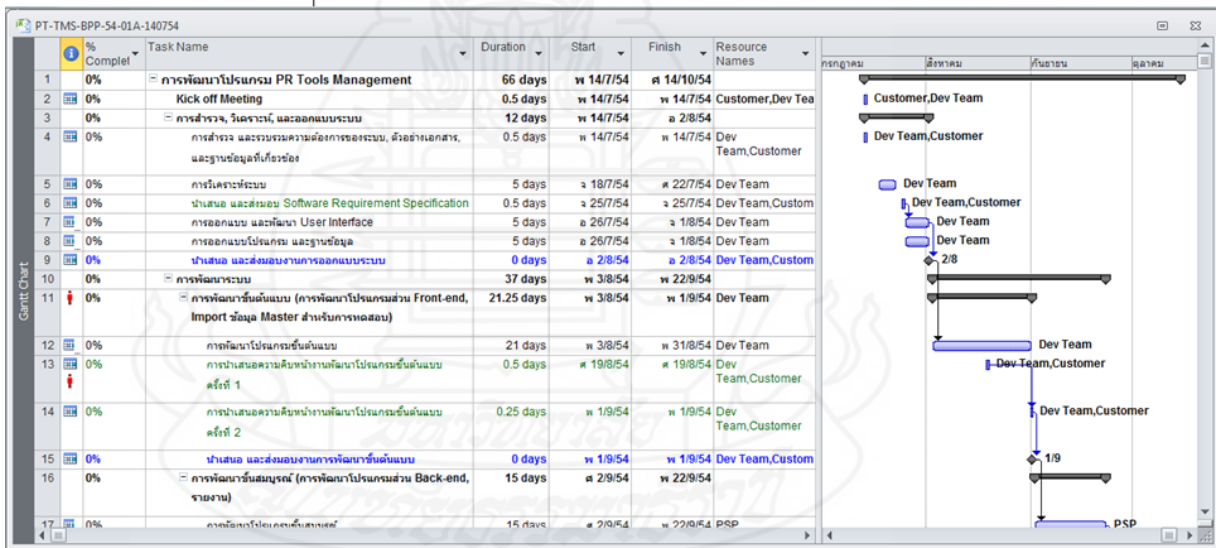
ภาพที่ 4.9 ตัวอย่างเอกสาร Project Plan ส่วนที่ 7

สถานที่สำรองข้อมูลโครงการ  
(Project Repository Backup)

- การ Backup ข้อมูล (Backup Strategy)
  - รูปแบบการ Backup
    - เจ้าหน้าที่ดำเนินการ Backup ข้อมูลจาก Server จัดเก็บใน [\\172.16.0.1\Backup\CustomerName\ProjectName\Project Sub Folders](#)
  - ความถี่ในการ Backup
    - ทุกสิ้นวัน
  - การ Recover ข้อมูล
    - โอนถ่ายจาก [\\172.16.0.1\Backup\CustomerName\ProjectName\Project Sub Folders](#) ที่ได้ Backup ข้อมูลไว้

ภาพที่ 4.10 ตัวอย่างเอกสาร Project Plan ส่วนที่ 8

รายการดำเนินงานและระยะเวลาดำเนินงาน  
(Project Timeframe)



ภาพที่ 4.11 ตัวอย่างเอกสาร Project Plan ส่วนที่ 9

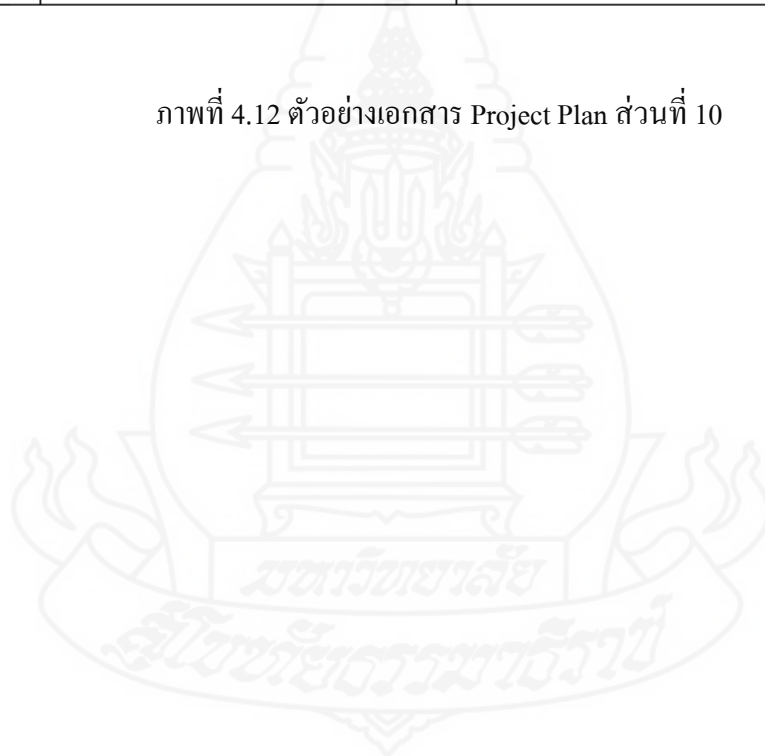


ทีมงานดำเนินโครงการ  
(Project Member)

■ Project Member (ผู้รับผิดชอบโครงการ)

ลำดับ	ชื่อ/สกุล	ตำแหน่งงาน
1	คุณ_____	ProjectManager
2	คุณ_____	Team Leader/System Analyst
3	คุณ_____	Programmer
4	คุณ_____	Tester
5	คุณ_____	Administration

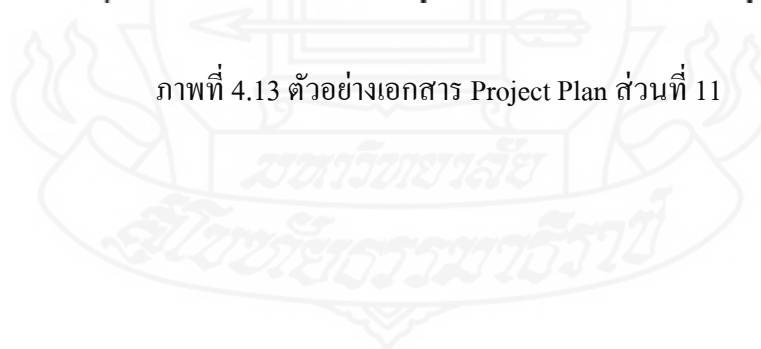
ภาพที่ 4.12 ตัวอย่างเอกสาร Project Plan ส่วนที่ 10



การส่งมอบงาน  
(Delivery Instruction)

สิ่งที่จะต้องส่งมอบ			
ขั้นตอนการดำเนินงาน	สิ่งที่ส่งมอบ	วิธีการส่งมอบ	วันที่กำหนดส่งมอบ
การสำรวจ, การวิเคราะห์, และการออกแบบระบบ	เอกสารการออกแบบระบบ ประกอบด้วยส่วนต่างๆ ดังนี้ <ul style="list-style-type: none"> <li>- System Architecture</li> <li>- Software Requirement Specification (SRS)</li> <li>- ER Diagram</li> <li>- Data Dictionary</li> </ul>	-ส่งมอบโดยการจัดทำเป็น Hard Copy จำนวน 1 ชุด	26 สิงหาคม 2554
	- User Interface Design	-ส่งมอบโดยการสาธิตตัวอย่างหน้าจอโปรแกรม โดยใช้เครื่องคอมพิวเตอร์ของผู้พัฒนาระบบ -จัดทำเป็น Hard Copy จำนวน 1 ชุด	26 สิงหาคม 2554
การพัฒนาารระบบ	- นำเสนอและสาธิตโปรแกรมขั้นสมบูรณ์	-ส่งมอบโดยการสาธิตโปรแกรม โดยใช้เครื่องคอมพิวเตอร์ของผู้พัฒนาระบบ	10 ตุลาคม 2554
การทดสอบระบบ	- ดำเนินการ Unit Test	-ดำเนินการทดสอบโดยเจ้าหน้าที่ของ	

ภาพที่ 4.13 ตัวอย่างเอกสาร Project Plan ส่วนที่ 11



การประเมินต้นทุนในการดำเนินโครงการ  
(Estimated Effort and Cost)

▪ ระยะเวลาโครงการ

- วันที่เริ่มโครงการ: 14 กรกฎาคม 2554
- วันที่สิ้นสุดโครงการ: 7 ธันวาคม 2554
- ระยะเวลาทำงานทั้งหมด: 97 วันทำการ

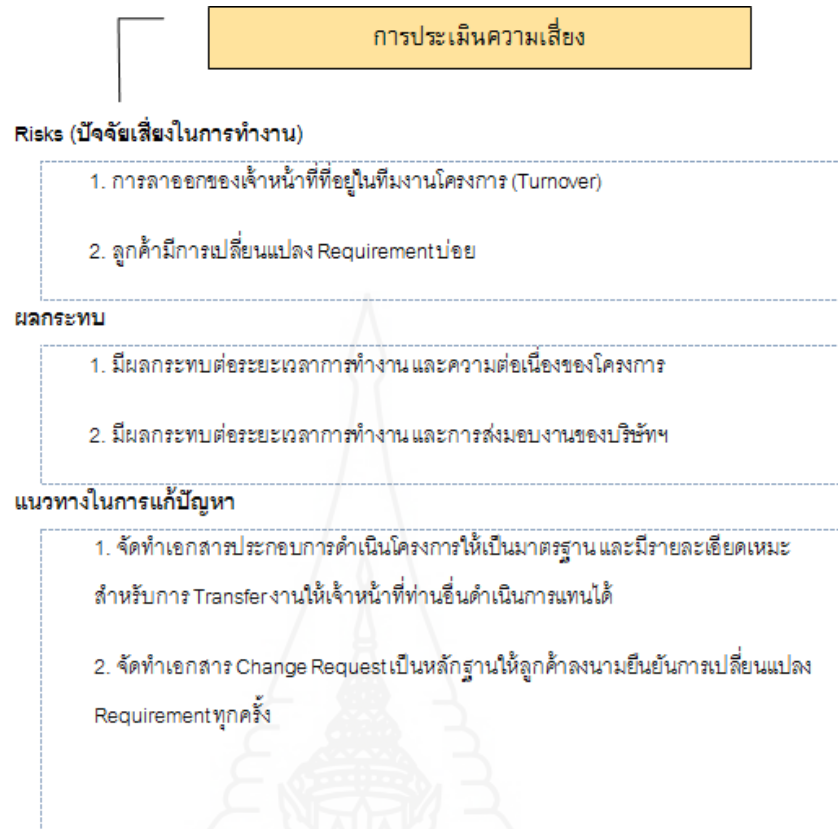
▪ บุคลากรที่ใช้

ลำดับ	ชื่อ/สกุล	ตำแหน่ง	เวลาที่ใช้ (คิดเป็น ชม.)	อัตรา/ชม. (Man Month)	ค่าใช้จ่าย (Cost)
1	คุณ _____	Project Manager	-	-	-
2	คุณ _____	Team Leader/SA	-	-	-
3	คุณ _____	Programmer	-	-	-
4	คุณ _____	Tester, Admin.	-	-	-

▪ ค่าใช้จ่ายในการพัฒนาระบบ

ลำดับ	รายการ	ค่าใช้จ่าย (Cost)
1	อุปกรณ์สำนักงาน Office Operation	-
2	จัดทำคู่มือเอกสาร	-

ภาพที่ 4.14 ตัวอย่างเอกสาร Project Plan ส่วนที่ 12



ภาพที่ 4.15 ตัวอย่างเอกสาร Project Plan ส่วนที่ 13



รหัสเอกสาร

## Verification & Validation Results

ตารางสำหรับการ  
ลงนาม
VFD-TMS-BPP-54-01A-300754

	Role	ชื่อ - นามสกุล	ลายเซ็น
Created by:	Project Manager	คุณ _____	
Reviewed by:	Team Leader/System Analyst	คุณ _____	
Approved by:	Project Manager	คุณ _____	

# Page 1

ประวัติการเปลี่ยนแปลงเอกสาร  
(REVISION HISTORY)

ลำดับ	วันเดือนปี	เวอร์ชัน	อ้างอิงเอกสาร	เพิ่ม /	รายละเอียดการแก้ไข	ผู้รับผิดชอบ
1	30 ก.ค. 2554	1	-	-	Initial	คุณ _____

ส่วนบันทึกประวัติ  
การแก้ไขและ  
ควบคุมเวอร์ชันของ  
เอกสาร

ภาพที่ 4.16 ตัวอย่างเอกสาร Verification & Validation Results ส่วนที่ 1

ชื่อโครงการ  
ชื่อผู้ดำเนินการตรวจสอบ

## Verification & Validation Result

ชื่อขั้นตอนที่ตรวจสอบ

Project Name:	การพัฒนา และ IMPLEMENT โปรแกรม TOOLS MANAGEMENT	Project Code:	TMS-BPP-54	Effort (MH):	Activity:
QA:	คุณ _____	Audit Phase:	Planning	Document:	
Audit Type:	Audit No.: 1	Round:		Audit Date:	30/7/2554

Reference Document: \_\_\_\_\_

รายการดำเนินงานของขั้นตอนที่ตรวจสอบ

No.	Checklist Details	Pass	Fail	Pending	Defect	Remark
<b>Business Model/Management Phase</b>						
<b>Project Planning/Cost (From Project Schedule)</b>						
1	Project Manager สร้างเอกสาร Project Timeframe (Progress Status Record) โดยใช้ Standard Template ที่ถูกต้อง	✓				VF
2	Project Manager ตรวจสอบเอกสาร Project Timeframe (Progress Status Record) ไว้ใน Project Library ของโครงการ	✓				VF
3	Project Manager สร้างเอกสาร Project Timeframe (Progress Status Record) โดยมีรายละเอียดครบถ้วน	✓				VD
4	Project Manager สร้างเอกสาร Project Plan โดยใช้ Standard Template ที่ถูกต้อง	✓				VF
5	Project Manager ตรวจสอบเอกสาร Project Plan ไว้ใน Project Library ของโครงการ	✓				VF
6	Project Manager สร้างเอกสาร Project Plan โดยมีรายละเอียดครบถ้วน	✓				VD
<b>Business Requirement Phase</b>						
<b>User Requirement</b>						
1	Project Manager สร้างเอกสาร Software Requirement Specification (SRS) โดยใช้ Standard Template ที่ถูกต้อง					
2	Project Manager ตรวจสอบเอกสาร Software Requirement Specification (SRS) ไว้ใน Project Library ของโครงการ					

ผลการตรวจสอบ  
 การตรวจสอบภาพรวม  
 โดยการ Verify

ภาพที่ 4.17 ตัวอย่างเอกสาร Verification & Validation Results ส่วนที่ 2

รหัสเอกสาร  
MR-BPP-TMS-64-01A-140764

**รายงานการประชุม**  
วันที่ 14 กรกฎาคม 2554  
เวลา 09.30 – 11.30 น.

ณ ห้องประชุม \_\_\_\_\_

ส่วนบันทึกประวัติการแก้ไขและ  
ควบคุมเวอร์ชันของเอกสาร

Revision History:

Version	Date	Author	Reviewer	Description
1.0	14 กรกฎาคม 2554	คุณ _____	คุณ _____	นำเสนอ Project Plan เตรียม ความพร้อมเริ่มโครงการ และ รวบรวม Requirement

รายชื่อผู้เข้าร่วมประชุม:

- ลูกค้า
  - คุณ \_\_\_\_\_
  - คุณ \_\_\_\_\_
  - คุณ \_\_\_\_\_
- ทีมงานดำเนินโครงการ
  - คุณ \_\_\_\_\_ / Project Manager
  - คุณ \_\_\_\_\_ / Team Leader/System Analyst
  - คุณ \_\_\_\_\_ / Programmer

รายชื่อผู้เข้าร่วมประชุมของลูกค้า

รายชื่อผู้เข้าร่วมประชุมของทีมงานดำเนินโครงการ

เรื่องที่ประชุม

เรื่องที่ประชุม:

- นำเสนอเอกสาร Project Plan สำหรับงานพัฒนาโปรแกรม Tools Management
- รวบรวม Requirement สำหรับการพัฒนาโปรแกรม Tools Management

รายละเอียดการประชุม

รายละเอียดการประชุม:

ทางทีมงานดำเนินการดำเนินการนำเสนอเอกสาร Project Plan สำหรับงานพัฒนาโปรแกรม Tools Management ให้ผู้เกี่ยวข้องกับโครงการของทางลูกค้าได้ทบทวน และรับทราบรายละเอียดเกี่ยวกับการดำเนินโครงการ

ทางทีมงานดำเนินการรวบรวมรายละเอียด Requirement สำหรับการพัฒนาโปรแกรม Tools Management โดยทางลูกค้ามีความประสงค์เพิ่ม Requirement ดังนี้

- ให้ออกแบบวางโครงสร้างโปรแกรมให้รองรับระบบการฝาก และขอรับเครื่องมือคืนบนโครงสร้างเดียวกับโปรแกรม Tools Management

รายการดำเนินงานหลังจบการประชุม

Next Step:

- ทางทีมงานดำเนินการดำเนินการดังนี้
  - ดำเนินงานออกแบบโปรแกรม Tools Management ตามขอบข่ายงานที่นำเสนอ Project Plan และประเมินระยะเวลาที่ต้องใช้ในการดำเนินงานเพิ่มเติมแจ้งกลับมายังทางลูกค้า

ภาพที่ 4.19 ตัวอย่างเอกสาร Meeting Record ส่วนที่ 2

### 2.1.2 PM.2: Project Plan Execution (การปฏิบัติตามแผนโครงการ)

1) ผู้รับผิดชอบในโครงการสำหรับ Activity นี้และบทบาทความเกี่ยวข้องมีดังนี้

(1) Project Manager ทำหน้าที่ติดตามการดำเนินงานตามรายละเอียดในเอกสาร Project Plan ร่วมกับ Team Leader/System Analyst และทีมงานดำเนินโครงการ และบันทึกสถานะการดำเนินโครงการในเอกสาร Project Timeframe (Progress Status Record) รายสัปดาห์

กรณีที่พบปัญหาหรือความเสี่ยงเพิ่มเติมระหว่างการดำเนินโครงการ ให้มีการบันทึกรายการปัญหาหรือความเสี่ยงไว้ในเอกสาร Correction Register เพื่อสำหรับติดตามสถานะและแก้ไขปัญหาให้จบลงได้

ก. กรณีที่ลูกค้ามีการแจ้งความต้องการในการเปลี่ยนแปลงระบบแตกต่างไปจากขอบข่ายงานที่กำหนดไว้ในเอกสาร Project Plan ทาง Project Manager และ Team Leader/System Analyst เป็นผู้ดำเนินการวิเคราะห์และประเมินผลกระทบกับการดำเนินโครงการในด้านต่างๆ ได้แก่ ผลกระทบด้านค่าใช้จ่าย, ผลกระทบด้านระยะเวลาและจัดทำเอกสาร Change Request เพื่อให้ลูกค้าลงนามยืนยันการเปลี่ยนแปลง และทำการปรับปรุงเอกสาร Project Plan ในส่วนที่เกี่ยวข้องกับการเปลี่ยนแปลง

(2) Project Manager ทำหน้าที่จัดการประชุมร่วมกับ Team Leader/System Analyst และทีมงานดำเนินโครงการ เพื่อ Update สถานะการดำเนินโครงการให้ทางลูกค้ารับทราบตามกำหนดการที่มีการนัดหมายร่วมกันกับลูกค้า โดยเนื้อหาการประชุมจะอ้างอิงตามรายละเอียดในเอกสาร Project Plan, เอกสาร Project Timeframe (Progress Status Record), เอกสาร Change Request (ถ้ามีการจัดทำขึ้น), และเอกสาร Meeting Record (บันทึกรายงานการประชุม) ครึ่งก่อนหน้า

หากมีการเปลี่ยนแปลงความต้องการระบบเพิ่มเติมภายหลังการประชุม ให้ดำเนินการปรับปรุงเอกสาร Project Plan, เอกสาร Change Request (ถ้ามีการจัดทำขึ้น), และเอกสาร Meeting Record (บันทึกรายงานการประชุม) แล้วนำเอกสาร Change Request ให้ลูกค้าลงนามยืนยัน

(3) Project Manager ทำหน้าที่จัดเก็บเอกสารและข้อมูลโครงการไว้ในสถานที่จัดเก็บข้อมูลโครงการ (Project Repository) และสถานที่สำรองข้อมูลโครงการ (Project Repository Backup) ทุกครั้งที่มีการเปลี่ยนแปลงเอกสารหรือมีข้อมูลเพิ่มเติม

2) เอกสารที่เกี่ยวข้องกับ Activity นี้มี ดังนี้

(1) Input Products

ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15)



ข. เอกสาร Project Timeframe (Progress Status Record) (ตัวอย่าง  
 ดังภาพที่ 4.20-4.22)

ค. เอกสาร Meeting Record (บันทึกถายงานการประชุม) ครั้งก่อน  
 หน้า (ตัวอย่างดังภาพที่ 4.18-4.19)

ง. เอกสาร Change Request ที่ได้รับการประเมินและจัดทำขึ้น โดย  
 Project Manager (ตัวอย่างดังภาพที่ 4.24-4.25)

จ. ข้อมูลสถานที่จัดเก็บข้อมูล โครงการ (Project Repository) ตามที่  
 กำหนดไว้ในเอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.9)

ฉ. ข้อมูลสถานที่สำรองข้อมูลโครงการ (Project Repository Backup)  
 ตามที่กำหนดไว้ในเอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.10)

## (2) Output Products

ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15) ที่ได้รับการปรับปรุง  
 ข้อมูลเป็นเวอร์ชันล่าสุด

ข. เอกสาร Project Timeframe (Progress Status Record) (ตัวอย่าง  
 ดังภาพที่ 4.20-4.22) ที่ได้รับการปรับปรุงสถานะการดำเนินงานจริงครั้งล่าสุดเป็นเวอร์ชันล่าสุด

ค. เอกสาร Meeting Record (บันทึกถายงานการประชุม) (ตัวอย่างดัง  
 ภาพที่ 4.18-4.19) ที่ได้รับการปรับปรุงข้อมูลล่าสุดเป็นเวอร์ชันล่าสุด

ง. เอกสาร Change Request (ตัวอย่างดังภาพที่ 4.24-4.25) ที่ได้รับ  
 การลงนามยืนยันจากลูกค้า

จ. เอกสาร Correction Register (ตัวอย่างดังภาพที่ 4.26-4.27) ที่ได้รับ  
 การจัดทำขึ้นและบันทึกปัญหาสำหรับการติดตามและแก้ไขสำหรับใช้ภายในทีมงานดำเนิน  
 โครงการ

ฉ. สถานที่จัดเก็บข้อมูลโครงการ (Project Repository) (ตัวอย่างดัง  
 ภาพที่ 4.9) ตามที่กำหนดไว้ในเอกสาร Project Plan ได้รับการจัดเตรียมขึ้นและได้รับการจัดเก็บ  
 ข้อมูล

ช. ข้อมูลสถานที่สำรองข้อมูลโครงการ (Project Repository Backup)  
 (ตัวอย่างดังภาพที่ 4.10) ตามที่กำหนดไว้ในเอกสาร Project Plan ได้รับการจัดเตรียมขึ้นและได้รับ  
 การสำรองข้อมูล

3) รายละเอียดเพิ่มเติมมี ดังนี้

(1) ควรให้ลูกค้านามรับรองเอกสาร Change Request เพื่อเป็นการยืนยันยอมรับการเปลี่ยนแปลงระบบอย่างเป็นทางการ

**บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ ActivityPM.2: Project Plan Execution (การปฏิบัติตามแผนโครงการ)**

ActivityPM.2:Project Plan Execution (การปฏิบัติตามแผนโครงการ) เป็นการดำเนินงานตามเอกสาร Project Plan และเอกสาร Project Timeframe (Progress Status Record) ที่ได้วางแผนโครงการไว้ โดยหากพบปัญหาเกิดขึ้นระหว่างการดำเนินงานก็ให้ทำการบันทึกรวบรวมไว้ในเอกสาร Correction Register เพื่อจะใช้สำหรับติดตามสถานะร่วมกันภายในทีมงานดำเนินโครงการและจัดการแก้ไขปัญหาให้จบลง แต่หากทางลูกค้ามีการแจ้งเปลี่ยนแปลงความต้องการระบบต่างไปจากขอบข่ายงานก็ให้จัดทำเอกสาร Change Request ขึ้นโดยมีการประเมินผลกระทบร่วมกันระหว่าง Project Manager และ Team Leader/System Analyst แล้วจึงนำไปให้ลูกค้าลงนามยืนยันการเปลี่ยนแปลงก่อนเริ่มแก้ไขระบบ ซึ่งบทบาทผู้รับผิดชอบโครงการที่สำคัญประกอบด้วย Project Manager, Team Leader/System Analyst, และทีมงานดำเนินโครงการ โดยแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 สำหรับ ActivityPM.2:Project Plan Execution (การปฏิบัติตามแผนโครงการ) และตัวอย่างเอกสารที่เป็นผลจากการดำเนินงาน Activity นี้ (Output Products) มีรายละเอียด ดังนี้

ตารางที่ 4.2 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ

Activity PM.2: Project Plan Execution (การปฏิบัติตามแผน โครงการ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบในโครงการ
<p><b>1. ทบทวนเอกสาร Project Timeframe</b> เพื่อใช้ในการติดตามและบันทึกสถานะการดำเนินโครงการเป็นเอกสาร Project Timeframe(Project Status Record) ควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</p> <ul style="list-style-type: none"> <li>○ ชื่อโครงการ, ชื่อโปรแกรม/ชื่อระบบ, ชื่อลูกค้า, รหัสเอกสาร</li> <li>○ รายการงาน (Tasks) ที่เป็นรายละเอียดขั้นตอนการดำเนินงาน</li> <li>○ ระยะเวลาเริ่มต้น-สิ้นสุด ของการดำเนินงานของแต่ละรายการงาน (Tasks) ที่กำหนดไว้เป็นแผนงาน Baseline ตามที่ระบุไว้ใน Project Plan</li> <li>○ ประมาณการณ้ค่าใช้จ่าย (Cost) ของการดำเนินงานของแต่ละรายการงาน (Tasks) ที่กำหนดเป็น Baseline</li> <li>○ รายชื่อทีมงานดำเนินโครงการ (Project Resources) ของการดำเนินงานของแต่ละรายการงาน (Tasks)</li> <li>○ ระยะเวลาเริ่มต้น-สิ้นสุด ของการดำเนินงานของแต่ละรายการงาน (Tasks) ตามที่ดำเนินงานจริง</li> <li>○ ประมาณการณ้ค่าใช้จ่าย (Cost) ของการดำเนินงานของแต่ละรายการงาน</li> </ul>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan เวอร์ชัน Baseline ที่ได้รับการลงนาม ยืนยันจากลูกค้า</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Timeframe(Project Status Record) เวอร์ชัน Baseline</li> </ul>	<p>- <b>Project Manager</b> ทำหน้าที่ติดตามการดำเนินงานตามรายละเอียดในเอกสาร Project Plan ร่วมกับ <b>Team Leader/System Analyst</b> และทีมงาน ดำเนินโครงการ และบันทึกสถานะการดำเนินโครงการในเอกสาร Project Timeframe (Progress Status Record) รายสัปดาห์</p>

## ตารางที่ 4.2 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<ul style="list-style-type: none"> <li>○ (Tasks) ตามที่ดำเนินงานจริง</li> <li>○ เลขที่เอกสาร Change Request และ/หรือข้อความหมายเหตุ ที่มีผลกระทบต่อการทำงานให้ต้องขยายเวลาการดำเนินงานแต่ละรายการงาน (Tasks)</li> <li>○ มีการบันทึกประวัติการแก้ไข (Revision History)</li> </ul> <p>*ตัวอย่างเอกสารProject Timeframe(Project Status Record)ดังภาพที่ 4.20-4.22</p>		
<p><b>2. จัดทำเอกสาร Change Request</b>กรณีที่ถูกค้ามีการแจ้งความต้องการในการเปลี่ยนแปลงระบบแตกต่างไปจากขอบข่ายงานที่กำหนดไว้ในเอกสาร Project Planเพื่อใช้บันทึกการขอเปลี่ยนแปลง/แก้ไขของลูกค้า ควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</p> <ul style="list-style-type: none"> <li>○ ชื่อโครงการ, ชื่อโปรแกรม/ชื่อระบบ, ชื่อลูกค้า</li> <li>○ ชื่อผู้ขอเปลี่ยนแปลง</li> <li>○ รายชื่อทีมงานดำเนินโครงการ (Project Member)</li> </ul>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan เวอร์ชัน Baseline ที่ได้รับการลงนาม ยืนยันจากลูกค้า</li> <li>- เอกสาร Change Request ฉบับร่าง</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan ที่ได้รับการปรับปรุง ข้อมูลเป็นเวอร์ชันล่าสุด</li> </ul>	<p><b>Project Manager</b></p> <p>จัดทำเอกสาร Change Request ฉบับร่างและดำเนินการวิเคราะห์ และประเมินผลกระทบกับการดำเนินโครงการในด้านต่างๆ ได้แก่ ผลกระทบด้านค่าใช้จ่าย, ผลกระทบด้านระยะเวลาพร้อมกับ</p>

## ตารางที่ 4.2 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<ul style="list-style-type: none"> <li>○ ระดับความสำคัญ</li> <li>○ ระดับผลกระทบ</li> <li>○ รายละเอียดการเปลี่ยนแปลง</li> <li>○ ระยะเวลาและค่าใช้จ่ายที่ใช้ในการเปลี่ยนแปลง</li> <li>○ มีการบันทึกประวัติการแก้ไข (Revision History)</li> <li>○ ส่วนลงนามยืนยันจากลูกค้า</li> </ul>	<ul style="list-style-type: none"> <li>- เอกสาร Change Request เวอร์ชันที่ได้รับการลงนามยืนยันจากลูกค้า</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Team Leader/System Analyst</b></li> <li>- <b>Project Manager</b> นำเอกสาร Change Request ให้ลูกค้าลงนามยืนยันการเปลี่ยนแปลง</li> <li>- <b>Project Manager</b> ทำการปรับปรุงเอกสาร Project Plan เป็นเวอร์ชันล่าสุด</li> </ul>

\*ตัวอย่างเอกสาร Change Request ดังภาพที่ 4.24-4.25

## ตารางที่ 4.2 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบในโครงการ
<p><b>3. จัดทำเอกสาร Meeting Record (บันทึกการประชุม)</b>สำหรับบันทึกผลการประชุมในการ Update สถานะการดำเนินโครงการให้ทางลูกค้ารับทราบตามกำหนดการที่มีการนัดหมายร่วมกันกับลูกค้าโดยเอกสาร Meeting Record (บันทึกรายงานการประชุม)ควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</p> <ul style="list-style-type: none"> <li>○ รหัสเอกสาร</li> <li>○ รายชื่อผู้เข้าร่วมประชุมของลูกค้า</li> <li>○ รายชื่อผู้เข้าร่วมประชุมของทีมงาน     ดำเนินโครงการ</li> <li>○ หัวข้อ/เรื่องที่ประชุม</li> <li>○ รายละเอียดการประชุม</li> <li>○ รายการดำเนินงานหลังจบการประชุม</li> <li>○ มีการบันทึกประวัติการแก้ไขเอกสาร (Revision History)</li> </ul>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan เวอร์ชัน Baseline ที่ได้รับการลงนามยืนยันจากลูกค้า</li> <li>- เอกสาร Project Timeframe(Project Status Record) เวอร์ชัน Baseline</li> <li>- เอกสาร Change Request ฉบับร่าง</li> <li>- เอกสาร Meeting Record (บันทึกการประชุม)เวอร์ชันก่อนหน้า</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Meeting Record (บันทึกการประชุม) ที่ได้รับการปรับปรุงข้อมูลเป็นเวอร์ชันล่าสุด</li> </ul>	<p>- <b>Project Manager</b> หรือ <b>Team Leader/System Analyst</b> เป็นผู้บันทึกเอกสาร Meeting Record (บันทึกการประชุม)</p>
<p>*ตัวอย่างเอกสาร Meeting Record (บันทึกการประชุม) ดังภาพที่ 4.18-4.19</p>		

## ตารางที่ 4.2 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p><b>4. จัดทำเอกสาร Correction Register เพื่อ</b> บันทึกปัญหา, วิธีการแก้ไข, และสถานะการ แก้ไข ที่มีการประชุมร่วมกันภายในทีมงาน ดำเนินโครงการ ควรประกอบด้วยข้อมูลที่ สำคัญดังต่อไปนี้</p> <ul style="list-style-type: none"> <li>○ ชื่อโครงการ, ชื่อโปรแกรม/ชื่อระบบ, ชื่อลูกค้า</li> <li>○ ประเด็นปัญหา</li> <li>○ วิธีการแก้ไข</li> <li>○ ผู้รับผิดชอบ</li> <li>○ สถานะการแก้ไข</li> <li>○ มีการบันทึกประวัติการแก้ไข (Revision History)</li> </ul>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan เวอร์ชันล่าสุด</li> <li>- เอกสาร Project Timeframe(Project Status Record) เวอร์ ชันล่าสุด</li> <li>- เอกสาร Meeting Record (บันทึกการ ประชุม)เวอร์ชัน ล่าสุด</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสารCorrection Register เวอร์ชัน Baseline</li> </ul>	<p>- <b>Project Manager</b> หรือ <b>Team Leader/System Analyst</b> เป็นผู้จัดทำ และบันทึกเอกสาร Correction Register และกำหนดเป็น เวอร์ชัน Baseline</p>
*ตัวอย่างเอกสาร Correction Register ดังภาพที่ 4.26-27		



## ตารางที่ 4.2 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>5. จัดเตรียมสถานที่จัดเก็บข้อมูลโครงการ (Project Repository) ตามที่กำหนดไว้ในเอกสาร Project Plan และจัดเก็บข้อมูลทุกครั้งที่มีการเปลี่ยนแปลงเอกสารหรือมีข้อมูลเพิ่มเติม</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- ข้อมูลสถานที่จัดเก็บข้อมูลโครงการ (Project Repository) ตามที่กำหนดไว้ในเอกสาร Project Plan เวอร์ชัน Baseline ที่ได้รับการลงนามยืนยันจากลูกค้า</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- สถานที่จัดเก็บข้อมูลโครงการ (Project Repository) ที่ได้รับการจัดเตรียมขึ้นและได้รับการจัดเก็บข้อมูลทุกครั้งที่มีการเปลี่ยนแปลงเอกสารหรือมีข้อมูลเพิ่มเติม</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Project Manager</b> เป็นผู้จัดเก็บข้อมูลโครงการทุกครั้งที่มีการเปลี่ยนแปลงเอกสารหรือมีข้อมูลเพิ่มเติม</li> <li>- <b>ทีมงานดำเนินโครงการ</b> สามารถเข้าถึงสถานที่จัดเก็บข้อมูลโครงการ (Project Repository) แต่ไม่สามารถเปลี่ยนแปลงข้อมูลในสถานที่จัดเก็บข้อมูลโครงการ (Project Repository) ได้</li> </ul>

\*ตัวอย่างสถานที่จัดเก็บข้อมูลโครงการ (Project Repository) ดังภาพที่ 4.9

ตารางที่ 4.2 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>6. จัดเตรียมสถานที่สำรองข้อมูลโครงการ (Project Repository Backup) ตามที่กำหนดไว้ในเอกสาร Project Plan และสำรองข้อมูลทุกครั้งที่มีการจัดเก็บข้อมูลในสถานที่จัดเก็บข้อมูลโครงการ (Project Repository)</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- ข้อมูลสถานที่สำรองข้อมูลโครงการ (Project Repository Backup) ตามที่กำหนดไว้ในเอกสาร Project Plan เวอร์ชัน Baseline ที่ได้รับการลงนามยืนยันจากลูกค้า</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- สถานที่สำรองข้อมูลโครงการ (Project Repository Backup) ที่ได้รับการจัดเตรียมขึ้นและได้รับการสำรองข้อมูลทุกครั้งที่มีการจัดเก็บข้อมูลในสถานที่จัดเก็บข้อมูลโครงการ (Project Repository)</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Project Manager</b> ทำการสำรองข้อมูลทุกครั้งที่มีการจัดเก็บข้อมูลในสถานที่จัดเก็บข้อมูลโครงการ (Project Repository)</li> </ul>
<p>*ตัวอย่างสถานที่สำรองข้อมูลโครงการ (Project Repository Backup) ดังภาพที่ 4.10</p>		

รหัสเอกสาร

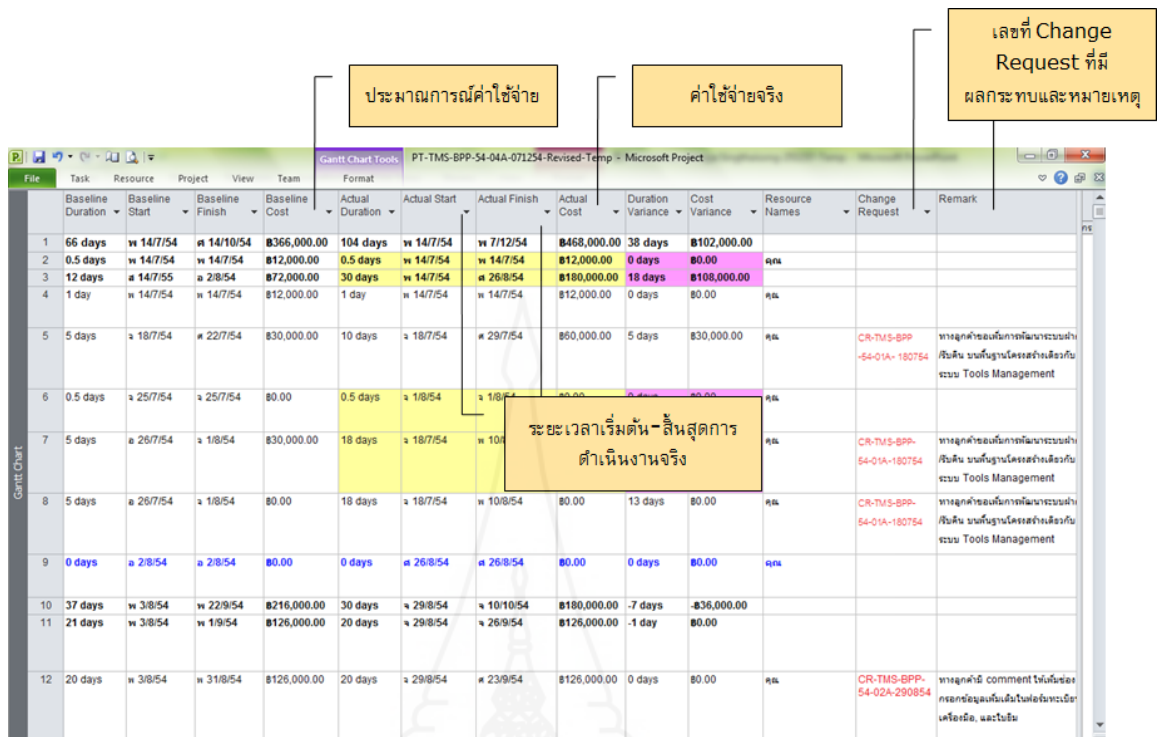
รายการดำเนินงาน

ระยะเวลาเริ่มต้น-สิ้นสุดการดำเนินงานที่เป็น  
แผนงาน Baseline

	% Cpl	Task Name	Baseline Duration	Baseline Start	Baseline Finish	Baseline Cost	Actual Duration	Actual Start	Actual Finish	Actual Cost	Duration Variance	Cost Variance
1	100%	การพัฒนาโปรแกรม Tools Management	66 days	พ 14/7/54	ศ 14/10/54	฿366,000.00	104 days	พ 14/7/54	พ 7/12/54	฿468,000.00	38 days	฿102,000.00
2	100%	Kick off Meeting	0.5 days	พ 14/7/54	พ 14/7/54	฿12,000.00	0.5 days	พ 14/7/54	พ 14/7/54	฿12,000.00	0 days	฿0.00
3	100%	การสำรวจ, วิเคราะห์, และออกแบบระบบ	12 days	อ 14/7/55	อ 2/8/54	฿72,000.00	30 days	พ 14/7/54	ศ 26/8/54	฿180,000.00	18 days	฿108,000.00
4	100%	การสำรวจ และรวบรวมความต้องการของระบบ, ศึกษาเอกสาร, และฐานข้อมูลที่เกี่ยวข้อง	1 day	พ 14/7/54	พ 14/7/54	฿12,000.00	1 day	พ 14/7/54	พ 14/7/54	฿12,000.00	0 days	฿0.00
5	100%	การวิเคราะห์ระบบ	5 days	จ 18/7/54	ศ 22/7/54	฿30,000.00	10 days	จ 18/7/54	ศ 29/7/54	฿60,000.00	5 days	฿30,000.00
6	100%	งานเขียน และพัฒนา Software Requirement Specification	0.5 days	จ 25/7/54	จ 25/7/54	฿0.00	0.5 days	จ 18/54	จ 18/54	฿0.00	0 days	฿0.00
7	100%	การออกแบบโปรแกรม และฐานข้อมูล	5 days	อ 26/7/54	จ 1/8/54	฿30,000.00	18 days	จ 18/7/54	พ 10/8/54	฿108,000.00	13 days	฿78,000.00
8	100%	การออกแบบ และพัฒนา User Interface	5 days	อ 26/7/54	จ 1/8/54	฿0.00	18 days	จ 18/7/54	พ 10/8/54	฿0.00	13 days	฿0.00
9	100%	นำเสนอ และส่งมอบงานการออกแบบระบบ	0 days	อ 2/8/54	อ 2/8/54	฿0.00	0 days	ศ 26/8/54	ศ 26/8/54	฿0.00	0 days	฿0.00
10	100%	การพัฒนาการระบบ	37 days	พ 3/8/54	พ 22/9/54	฿216,000.00	30 days	จ 29/8/54	จ 10/10/54	฿180,000.00	-7 days	฿36,000.00
11	100%	การพัฒนาขั้นต้นแบบ (การพัฒนาโปรแกรมส่วน Front-end, Import ข้อมูล Master ส่วนเงินการไหลสวน)	21 days	พ 3/8/54	พ 1/9/54	฿126,000.00	20 days	จ 29/8/54	จ 26/9/54	฿126,000.00	-1 day	฿0.00
12	100%	การพัฒนาโปรแกรมขั้นต้นแบบ	20 days	พ 3/8/54	พ 31/8/54	฿126,000.00	20 days	จ 29/8/54	ศ 23/9/54	฿126,000.00	0 days	฿0.00

ภาพที่ 4.20 ตัวอย่างเอกสาร Project Timeframe (Project Status Record) ส่วนที่ 1





ภาพที่ 4.21 ตัวอย่างเอกสาร Project Timeframe (Project Status Record) ส่วนที่ 2



ภาพที่ 4.22 ตัวอย่างเอกสาร Project Timeframe (Project Status Record) ส่วนที่ 3

<b>Change Request</b>				
ชื่อโครงการ, ชื่อโปรแกรม/ระบบ				รหัสเอกสาร
Project: โครงการพัฒนาโปรแกรม Tools Management				
Application: โปรแกรม Tools Management				
Customer Company: บริษัท				
Customer Contact: คุณ				ทีมงานดำเนินโครงการ
Tel/Fax No.:				
Project Manager:				
Project Member:				
อ้างอิง Software Requirement Specification: V.1			วันที่:	ส่วนบันทึกประวัติการแก้ไขและควบคุมเวอร์ชันของเอกสาร
ครั้งที่เปลี่ยนแปลง:		วันที่เปลี่ยนแปลง:		
Revision History:				
Version	Date	Author	Reviewer	Description
1.0	18 กรกฎาคม 2554	คุณ _____	คุณ _____	เพิ่มโปรแกรมระบบจัดการการฝากรับคืน
2.0	29 สิงหาคม 2554	คุณ _____	คุณ _____	แก้ไข Minor Issue

ภาพที่ 4.23 ตัวอย่างเอกสาร Change Request ส่วนที่ 1

ชื่อผู้ขอเปลี่ยนแปลง	
ผู้ขอเปลี่ยนแปลง: คุณ _____	หน่วยงาน: _____
ระดับความสำคัญ:	<input type="checkbox"/> สำคัญยิ่งยวด <input type="checkbox"/> เสง่ด่วน <input checked="" type="checkbox"/> ปรับแก้ตามปกติ
หน้าจอที่มีผลกระทบ:	
รายงานที่มีผลกระทบ:	
ส่วนประกอบอื่น ๆ ที่มีผลกระทบ:	ระดับความสำคัญ
คำอธิบายการเปลี่ยนแปลง:	
วัตถุประสงค์:	
ผลกระทบต่อระบบที่กำลังพัฒนา:	
เอกสารที่มีผลกระทบต่อการทำงาน:	
วันเริ่ม:	
วันสิ้นสุด:	

ภาพที่ 4.24 ตัวอย่างเอกสาร Change Request ส่วนที่ 2

ลำดับ	รายละเอียด	ผลกระทบต่อระบบ		จำนวนวันที่เพิ่ม (man-days)	ค่าใช้จ่ายที่เพิ่ม (บาท)	สถานะ	หมายเหตุ
		Major	Minor				
1	รายละเอียดการเปลี่ยนแปลง		✓	0.5	-	<input type="checkbox"/> Accept <input checked="" type="checkbox"/> Accept & Close <input type="checkbox"/> Pending <input type="checkbox"/> Reject <input type="checkbox"/> Reject & Close	

	ชื่อ	ลงนาม	วันที่
ผู้แจ้ง:	คุณ _____		26 สิงหาคม 2554
หัวหน้างานผู้แจ้ง:	คุณ _____	ตารางสำหรับการลงนาม	26 สิงหาคม 2554
ผู้รับแจ้ง:	คุณ _____		26 สิงหาคม 2554
ผู้ดำเนินการแก้ไข:	คุณ _____		29 สิงหาคม 2554

ภาพที่ 4.25 ตัวอย่างเอกสาร Change Request ส่วนที่ 3

## Correction Register

รหัสเอกสาร: CRR-TMS-BPP-54-02A-270854

บทยกหน้าที่รับผิดชอบ	ชื่อ - นามสกุล	ลายเซ็น
ผู้จัดทำ: Project Manager	คุณ _____	
ผู้ตรวจทาน: Team Leader/System Analyst	คุณ _____	
ผู้อนุมัติ: Project Manager	คุณ _____	

**Page 1**  
ประวัติการเปลี่ยนแปลงเอกสาร  
(REVISION HISTORY)

ลำดับ	วันเดือนปี	เวอร์ชัน	อ้างอิงเอกสาร	เพิ่ม /	รายละเอียดการแก้ไข	ผู้รับผิดชอบ
1	15 ก.ค. 2554	1	-	-	ประกาศใช้ครั้งแรก	คุณ _____
2	27 ส.ค. 2554	2	-	เพิ่ม	ประเด็นปัญหา	คุณ _____

ส่วนบันทึกประวัติการแก้ไขและควบคุมเวอร์ชันของเอกสาร

ภาพที่ 4.26 ตัวอย่างเอกสาร Correction Register ส่วนที่ 1

ชื่อโครงการ, ชื่อโปรแกรม/ ระบบ, ชื่อลูกค้า					Correction Register					สถานะการแก้ปัญหา	
ชื่อโครงการ: การพัฒนาโปรแกรม TOOLS MANAGEMENT			รหัสโครงการ: TMS-BPP-54		วันที่: 27 ส.ค. 2554						
ชื่อลูกค้า:			ผู้จัดการโครงการ: คุณ _____		นักวิเคราะห์ระบบ: คุณ _____		โปรแกรมเมอร์: คุณ _____				
ลำดับ	No.	Date	Problem Source	Problem Type	ประเด็นปัญหา	การแก้ไข	รายละเอียดการดำเนินการ	ผู้แก้ไข	สถานะ	วันที่ดำเนินการ	
1	CRR-001	14 ก.ค. 54	Internal	Resource	การที่ลูกค้ามี Requirement ตาม CR-TMS-BPP-54-01A-180754	- ต้องการแก้ไขโครงสร้างข้อมูล, ต้องออกตามรายละเอียดเวลา	มอบหมายโปรแกรมเมอร์ทั้งหมด 1 คน สำหรับดำเนินการแก้ไข	คุณ _____	เสร็จสิ้น	14 ก.ค. 54	
2	CRR-002	27 ส.ค. 54	Internal	Resource	การที่ลูกค้ามี Requirement ตาม CR-TMS-BPP-54-02A-290854	- ต้องการแก้ไขโครงสร้างข้อมูล, ต้องออกตามรายละเอียดเวลา	มอบหมายโปรแกรมเมอร์ทั้งหมด 1 คน สำหรับดำเนินการแก้ไข	คุณ _____	เสร็จสิ้น	27 ส.ค. 54	

ประเด็นปัญหา      วิธีการแก้ปัญหา      ผู้รับผิดชอบ

ภาพที่ 4.27 ตัวอย่างเอกสาร Correction Register ส่วนที่ 2

### 2.1.3 PM.3: Project Assessment and Control (การประเมินและควบคุมโครงการ)

1) ผู้รับผิดชอบในโครงการสำหรับ Activity นี้และบทบาทความเกี่ยวข้องมี

ดังนี้

(1) Project Manager เป็นผู้จัดการประชุมร่วมกับ Team Leader/System Analyst และทีมงานดำเนินโครงการ รายสัปดาห์เพื่อทบทวนสถานะการดำเนินโครงการร่วมกันในด้านต่างๆ ดังนี้

ก. ประเมินการดำเนินงานโครงการตามรายละเอียดในเอกสาร Project Plan เวอร์ชันล่าสุด โดยทำการประเมินงานในด้านต่างๆ ดังนี้

ก) สถานะการดำเนินงานจริง เปรียบเทียบกับรายการดำเนินงานที่วางแผนไว้

ข) ทรัพยากรและบุคลากรที่ใช้ในการดำเนินงานจริง เปรียบเทียบกับที่วางแผนไว้

ค) ค่าใช้จ่ายที่ใช้ในการดำเนินงานจริง เปรียบเทียบกับที่วางแผนไว้

ง) ระยะเวลาที่ใช้ในการดำเนินงานจริง เปรียบเทียบกับที่วางแผนไว้

จ) ความเสี่ยงที่ประเมินไว้เกิดขึ้นจริงหรือไม่และสถานะการแก้ไข

ฉ) ทบทวนปัญหาที่เกิดขึ้นระหว่างการดำเนินโครงการรายสัปดาห์

และทบทวนแนวทางแก้ไขและสถานะการแก้ไขปัญหาให้ครบถ้วน

ช) ทบทวนการเปลี่ยนแปลงความต้องการระบบของลูกค้าแล้ว ประเมินผลกระทบและจัดทำเอกสาร Change Request ให้ทางลูกค้ายืนยันก่อนเริ่มการแก้ไข และ



ทบทวนเอกสาร Change Request ที่เคยได้รับการยืนยันแล้วให้มีการดำเนินงานแก้ไขให้เสร็จสิ้นอย่างครบถ้วน

2) เอกสารที่เกี่ยวข้องกับ Activity นี้มี ดังนี้

(1) Input Products

- ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15) เวอร์ชันล่าสุด
- ข. เอกสาร Project Timeframe (Progress Status Record) (ตัวอย่างดังภาพที่ 4.20-4.22) เวอร์ชันล่าสุด
- ค. เอกสาร Change Request (ตัวอย่างดังภาพที่ 4.24-4.25) ที่เคยได้รับการยืนยันจากลูกค้าแล้ว
- ง. เอกสาร Correction Register (ตัวอย่างดังภาพที่ 4.26-4.27) เวอร์ชันล่าสุด

(2) Output Products

- ก. เอกสาร Project Timeframe (Progress Status Record) (ตัวอย่างดังภาพที่ 4.20-4.22) ที่ได้รับการปรับปรุงสถานะการดำเนินงานจริงครั้งล่าสุดเป็นเวอร์ชันล่าสุด
- ข. เอกสาร Change Request (ตัวอย่างดังภาพที่ 4.24-4.25) ที่จัดทำขึ้นใหม่สำหรับเสนอให้ลูกค้าลงนามยืนยัน
- ค. เอกสาร Change Request (ตัวอย่างดังภาพที่ 4.24-4.25) ที่ได้รับการลงนามยืนยันจากลูกค้าแล้ว
- ง. เอกสาร Correction Register (ตัวอย่างดังภาพที่ 4.26-4.27) ที่ได้รับการปรับปรุงครั้งล่าสุดเป็นเวอร์ชันล่าสุด

3) รายละเอียดเพิ่มเติมมี ดังนี้

(1) ควรให้ลูกค้าลงนามรับรองเอกสาร Change Request เพื่อเป็นการยืนยันยอมรับการเปลี่ยนแปลงระบบอย่างเป็นทางการ

**บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ ActivityPM.3: Project Assessment and Control (การประเมินและควบคุมโครงการ)**

ActivityPM.3: Project Assessment and Control (การประเมินและควบคุมโครงการ) เป็นการดำเนินงานภายในทีมงานดำเนินโครงการ โดยการทบทวนสถานะการดำเนินงานโครงการอ้างอิงเอกสาร Project Plan, เอกสาร Project Timeframe (Progress Status Record), และเอกสาร Change Request ที่ได้รับการยืนยันการแก้ไขจากลูกค้าและกำหนดให้มีการดำเนินงานอย่าง

ต่อเนื่อง รวมทั้งทบทวนปัญหาที่เกิดขึ้นระหว่างการดำเนินโครงการและกำหนดแนวทางแก้ไข ปัญหาและทบทวนสถานะการแก้ไขปัญหาอ้างอิงเอกสาร Correction Register เพื่อเป็นการประเมิน สถานะการดำเนินโครงการและจะสามารถควบคุมโครงการให้ดำเนินต่อไปได้อย่างเหมาะสม ซึ่งบทบาทผู้รับผิดชอบโครงการที่สำคัญสำหรับ Activity นี้ประกอบด้วย Project Manager, Team Leader/System Analyst, และทีมงานดำเนินโครงการโดยแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 สำหรับ Activity PM.3: Project Assessment and Control (การประเมินและควบคุมโครงการ) และตัวอย่างเอกสารที่เป็นผลจากการดำเนินงาน Activity นี้ (Output Products) มีรายละเอียดดังนี้

ตารางที่ 4.3 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

สำหรับ Activity PM.3: Project Assessment and Control (การประเมินและควบคุมโครงการ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบในโครงการ
<b>1. ประเมินการดำเนินงานโครงการตาม</b>	<b>Input Products</b>	- <b>Project Manager</b>
รายละเอียดในเอกสาร <b>Project Plan</b> เวอร์ชัน	- เอกสาร Project	เป็นผู้นำในการ
ล่าสุด, เอกสาร <b>Project Timeframe</b>	Plan เวอร์ชันล่าสุด	ประชุม
( <b>Progress Status Record</b> ) เวอร์ชันล่าสุด,	- เอกสาร Project	- <b>Team</b>
และเอกสาร <b>Change Request</b> ที่เคยได้รับ	Timeframe	<b>Leader/System</b>
การยืนยันจากลูกค้าแล้ว โดยการประชุม	(Progress Status	<b>Analyst</b> เข้าร่วม
ร่วมกันรายสัปดาห์เพื่อทบทวนสถานะการ	Record) เวอร์ชัน	ประชุม
ดำเนินโครงการ เพื่อประเมินงานในด้าน	ล่าสุด	- <b>ทีมงานดำเนิน</b>
ต่างๆ ดังนี้	- เอกสาร Change	<b>โครงการ</b> เข้าร่วม
○ สถานะการดำเนินงานจริง เปรียบเทียบ	Request ที่เคยได้รับ	ประชุม
กับรายการดำเนินงานที่วางแผนไว้	การยืนยันจากลูกค้า	
○ ทรัพยากรและบุคลากรที่ใช้ในการ	แล้ว	
ดำเนินงานจริง เปรียบเทียบกับที่	<b>Output Products</b>	
วางแผนไว้	- เอกสาร Project	

## ตารางที่ 4.3 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<ul style="list-style-type: none"> <li>○ ค่าใช้จ่ายที่ใช้ในการดำเนินงานจริง เปรียบเทียบกับที่วางแผนไว้</li> <li>○ ระยะเวลาที่ใช้ในการดำเนินงานจริง เปรียบเทียบกับที่วางแผนไว้</li> <li>○ ความเสี่ยงที่ประเมินไว้เกิดขึ้นจริง หรือไม่และสถานะการแก้ไข</li> </ul> <p>*ตัวอย่างเอกสาร Project Plan ดังภาพที่ 4.3-4.15</p> <p>*ตัวอย่างเอกสาร Project Timeframe (Progress Status Record) ดังภาพที่ 4.20-4.22</p> <p>*ตัวอย่างเอกสาร Change Request ดังภาพที่ 4.24-4.25</p>	<p>Timeframe (Progress Status Record) ที่ได้รับการปรับปรุงสถานะการดำเนินงานจริงครั้งล่าสุดเป็นเวอร์ชันล่าสุด</p>	
<p><b>2. ทบทวนปัญหาที่เกิดขึ้นระหว่างการดำเนินโครงการ</b> โดยการประชุมร่วมกันรายสัปดาห์ แล้วบันทึกรายการปัญหา, แนวทางแก้ไข, และสถานะการแก้ไขในเอกสาร Correction Register</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Correction Register เวอร์ชันล่าสุด</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Correction Register ที่ได้รับการปรับปรุงครั้งล่าสุดเป็นเวอร์ชันล่าสุด</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Project Manager</b> เป็นผู้นำในการประชุม</li> <li>- <b>Team Leader/System Analyst</b> เข้าร่วมประชุม</li> <li>- <b>ทีมงานดำเนินโครงการ</b> เข้าร่วมประชุม</li> </ul>
<p>*ตัวอย่างเอกสาร Correction Register ดังภาพที่ 4.26-4.27</p>		

ตารางที่ 4.3 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>3. ทบทวนการเปลี่ยนแปลงความต้องการระบบของลูกค้าแล้วประเมินผลกระทบและจัดทำเอกสาร Change Request ให้ทางลูกค้ายืนยันก่อนเริ่มการแก้ไข และทบทวนเอกสาร Change Request ที่เคยได้รับการยืนยันแล้วให้มีการดำเนินงานแก้ไขให้เสร็จสิ้นอย่างครบถ้วนโดยการประชุมร่วมกันรายสัปดาห์แล้วจัดทำเอกสาร Change Request เพิ่มเติม และ update สถานะการแก้ไขระบบตามเอกสาร Change Request ที่เคยได้รับการยืนยันแล้ว</p> <p>*ตัวอย่างเอกสาร Change Request ดังภาพที่ 4.24-4.25</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Change Request ที่เคยได้รับการยืนยันจากลูกค้าแล้ว</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Change Request ที่จัดทำขึ้นใหม่สำหรับเสนอให้ลูกค้าลงนามยืนยัน</li> <li>- เอกสาร Change Request ที่ได้รับการ update สถานะการดำเนินงานครั้งล่าสุดเป็นเวอร์ชันล่าสุด</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Project Manager</b> เป็นผู้นำในการประชุม</li> <li>- <b>Team Leader/System Analyst</b> เข้าร่วมประชุม</li> <li>- <b>ทีมงานดำเนินโครงการ</b> เข้าร่วมประชุม</li> </ul>

#### 2.1.4 PM.4: Project Closure (การส่งมอบโครงการ)

1) ผู้รับผิดชอบในโครงการสำหรับ Activity นี้และบทบาทความเกี่ยวข้องมีดังนี้

(1) Project Manager จัดเตรียมเอกสารและข้อมูลสำหรับการส่งมอบงานอย่างเป็นทางการตามข้อมูลการส่งมอบงาน (Delivery Instruction) ที่กำหนดไว้ในเอกสาร Project Plan และจัดเตรียมจดหมายส่งมอบปิดโครงการเพื่อนำเสนอและส่งมอบให้กับลูกค้าลงนามยืนยันรับมอบโครงการ

(2) Project Manager จัดเก็บเอกสารและข้อมูลโครงการที่ส่งมอบให้กับลูกค้าในสถานที่จัดเก็บข้อมูลโครงการ (Project Repository) และสถานที่สำรองข้อมูลโครงการ (Project Repository Backup)

2) เอกสารที่เกี่ยวข้องกับ Activity นี้มี ดังนี้

(1) Input Products

ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15) เวอร์ชันล่าสุด

ข. ข้อมูลและเอกสารในชุด Software Configuration ชุดที่ส่งมอบให้ลูกค้า ซึ่งข้อมูล Software Configuration ประกอบด้วย

ก) เอกสาร Software Requirements Specification\*

ข) เอกสาร Software Design\*

ค) เอกสาร Traceability Record\*

ง) Software ที่พัฒนาเสร็จแล้ว\*

จ) เอกสาร Test Cases and Test Procedures\*

ฉ) เอกสาร Test Report\*

ช) เอกสาร System Manual (Product Operation Guide & Maintenance Documentation)\*

ซ) เอกสาร User Manual (Software User Documentation)\*

**หมายเหตุ:**\*เอกสารและข้อมูลในชุด Software Configuration จะได้รับการกล่าวถึงอีกครั้งในกระบวนการ Software Implementation

ค. สถานที่จัดเก็บข้อมูลโครงการ (Project Repository) (ตัวอย่างดังภาพที่ 4.9)

ง. สถานที่สำรองข้อมูลโครงการ (Project Repository Backup) (ตัวอย่างดังภาพที่ 4.10)

(2) Output Products

ก. ข้อมูล Software Configuration ชุดที่ได้รับการยอมรับจากลูกค้า

**หมายเหตุ:** เอกสารและข้อมูลในชุด Software Configuration จะได้รับการกล่าวถึงในกระบวนการ Software Implementation

ข. จดหมายส่งมอบงานปิดโครงการ (Acceptance Record) (ตัวอย่างดังภาพที่ 4.28-4.29) ที่ได้รับการลงนามจากลูกค้า

ค. สถานที่จัดเก็บข้อมูลโครงการ (Project Repository) (ตัวอย่าง  
 ดังภาพที่ 4.9) ที่ได้รับการ update การจัดเก็บเอกสารและข้อมูลชุดที่ได้รับการยอมรับจากลูกค้า

ง. สถานที่สำรองข้อมูลโครงการ (Project Repository Backup)  
 (ตัวอย่างดังภาพที่ 4.10) ที่ได้รับการ update การจัดเก็บเอกสารและข้อมูลชุดที่ได้รับการยอมรับจาก  
 ลูกค้า

### 3) รายละเอียดเพิ่มเติมมี ดังนี้

(1) Project Manager นำจดหมายส่งมอบงานปิดโครงการ (Acceptance  
 Record) ให้ลูกค้าลงนามยืนยันยอมรับการส่งมอบโครงการอย่างเป็นทางการเป็นลายลักษณ์อักษร

#### บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110

##### ระดับ Basic Profile สำหรับ Activity PM.4: Project Closure (การส่งมอบโครงการ)

PM.4: Project Closure (การส่งมอบโครงการ) เป็นการดำเนินงานโดย  
**Project Manager** เป็นผู้ที่ทำหน้าที่ตรวจทาน, รวบรวม, และจัดเตรียมข้อมูลและเอกสารต่างๆ  
 สำหรับการส่งมอบงานเพื่อปิดโครงการตามข้อมูลการส่งมอบงาน (Delivery Instruction) ที่กำหนด  
 ไว้ในเอกสาร Project Plan เพื่อนำเสนอต่อลูกค้าและทำการ update การจัดเก็บข้อมูลและเอกสาร  
 โครงการทั้งหมดที่ผ่านการยอมรับจากลูกค้าแล้วในสถานที่จัดเก็บข้อมูลโครงการ (Project  
 Repository) และสถานที่สำรองข้อมูลโครงการ (Project Repository Backup) โดยมีรายละเอียดแนว  
 ปฏิบัติที่ดี ดังนี้

ตารางที่ 4.4 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

สำหรับ Activity PM.4: Project Closure (การส่งมอบโครงการ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
1. จัดเตรียมข้อมูล Software Configuration ซึ่งคือเอกสารและข้อมูลสำหรับการส่งมอบ งานอย่างเป็นทางการตามข้อมูลการส่งมอบ งาน (Delivery Instruction) ที่กำหนดไว้ใน เอกสาร Project Plan	<b>Input Products</b> - เอกสาร Project Plan เวอร์ชันล่าสุด - ข้อมูล Software Configuration ชุดที่ ส่งมอบให้ลูกค้า ซึ่ง ข้อมูล Software	- <b>Project Manager</b> เป็นผู้ดำเนินการ จัดเตรียมข้อมูล Software Configuration

## ตารางที่ 4.4 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>*ตัวอย่างเอกสารในชุด Software Configuration มีรายละเอียดในกระบวนการ Software Implementation</p>	<p>Configuration ประกอบด้วย</p> <ul style="list-style-type: none"> <li>- เอกสาร Software Requirements Specification</li> <li>- เอกสาร Software Design</li> <li>- เอกสาร Traceability Record</li> <li>- เอกสาร Traceability Record</li> <li>- Software Components และ Software ที่พัฒนาเสร็จแล้ว</li> <li>- เอกสาร Test Cases and Test Procedures</li> <li>- เอกสาร Test Report</li> <li>- เอกสาร System Manual (Product Operation Guide&amp; Maintenance Documentation)</li> <li>- เอกสาร User Manual (Software User Documentation)</li> </ul> <p><b>Output Products</b></p> <p>ข้อมูล Software Configuration ชุดที่จัดเตรียมสำหรับการส่งมอบงานเรียบร้อยแล้ว</p>	



## ตารางที่ 4.4 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p><b>2. จัดทำเอกสารจดหมายส่งมอบงานปิดโครงการ (Acceptance Record) เพื่อ</b> นำส่งให้ลูกค้าลงนามรับมอบงาน โดยเอกสารควรประกอบด้วยข้อมูลที่สำคัญ ดังนี้</p> <ul style="list-style-type: none"> <li>○ ชื่อโครงการ</li> <li>○ ชื่อผู้มีอำนาจรับมอบงานของลูกค้า</li> <li>○ ข้อมูลเอกสารเสนอราคาที่ได้รับการยืนยันว่าจ้างงาน</li> <li>○ รายการส่งมอบโครงการตามข้อมูล Software Configuration</li> <li>○ ระยะเวลาประกัน (ถ้ามี)</li> <li>○ ส่วนลงนามรับมอบงานปิดโครงการ จากลูกค้ารับมอบงานปิดโครงการ จากลูกค้า</li> </ul> <p>*ตัวอย่างเอกสารในชุด Software Configuration มีรายละเอียดในกระบวนการ Software Implementation</p> <p>*ตัวอย่างเอกสารจดหมายส่งมอบปิดโครงการ (Acceptance Record) ดังภาพที่ 4.28-4.29</p>	<p><b><u>Input Products</u></b></p> <ul style="list-style-type: none"> <li>- ข้อมูล Software Configuration ชุดที่จัดเตรียมสำหรับการส่งมอบงานเรียบร้อยแล้ว</li> </ul> <p><b><u>Output Products</u></b></p> <ul style="list-style-type: none"> <li>- ข้อมูล Software Configuration เวอร์ชันที่ได้รับการรับมอบงานปิดโครงการจากลูกค้าแล้ว</li> <li>- เอกสารจดหมายส่งมอบงานปิดโครงการ (Acceptance Record) ที่รับการลงนามรับมอบงานปิดโครงการจากลูกค้าแล้ว</li> </ul>	<p>- <b>Project Manager</b> เป็นผู้ดำเนินการส่งมอบข้อมูล Software Configuration และเอกสารจดหมายส่งมอบงานปิดโครงการ (Acceptance Record) เพื่อเสนอให้ลูกค้า ลงนามรับมอบงานปิดโครงการ</p>

ตารางที่ 4.4 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>3. จัดเก็บเอกสารและข้อมูลโครงการที่ส่งมอบให้กับลูกค้าในสถานที่จัดเก็บข้อมูลโครงการ (Project Repository) และสถานที่สำรองข้อมูลโครงการ (Project Repository Backup)</p> <p>*ตัวอย่างโครงสร้างสถานที่จัดเก็บข้อมูลโครงการ (Project Repository) และสถานที่สำรองข้อมูลโครงการ (Project Repository Backup) ดังภาพที่ 4.9-4.10</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- ข้อมูล Software Configuration ชุดที่ได้รับการยอมรับจากลูกค้า</li> <li>- สถานที่จัดเก็บข้อมูลโครงการ (Project Repository)</li> <li>- สถานที่สำรองข้อมูลโครงการ (Project Repository Backup)</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- สถานที่จัดเก็บข้อมูลโครงการ (Project Repository) ที่ได้รับการ update การจัดเก็บเอกสารและข้อมูลชุดที่ได้รับการยอมรับจากลูกค้า</li> <li>- สถานที่สำรองข้อมูลโครงการ (Project Repository Backup) ที่ได้รับการ update การจัดเก็บเอกสารและข้อมูลชุดที่ได้รับการยอมรับจากลูกค้า</li> </ul>	<p>- <b>Project Manager</b> เป็นผู้ดำเนินการจัดเก็บข้อมูลโครงการ</p>

ชื่อโครงการที่ส่งมอบ \_\_\_\_\_ วันที่ \_\_\_\_\_ วันที่ส่งมอบโครงการ \_\_\_\_\_

เรื่อง: ส่งมอบงานโครงการ \_\_\_\_\_  
เรียน: คุณ \_\_\_\_\_ ผู้มีอำนาจรับมอบงานของลูกค้า  
บริษัท \_\_\_\_\_

อ้างถึง: ใบเสนอราคาเลขที่ \_\_\_\_\_ ลงวันที่ \_\_\_\_\_ อ้างอิงเอกสารที่ยืนยันการว่าจ้าง

ชื่อโครงการที่ส่งมอบ \_\_\_\_\_  
ตามที่บริษัทฯ ได้ดำเนินการโครงการ \_\_\_\_\_ เรียบร้อยแล้ว โดยบริษัทฯ ได้ดำเนินการ  
และส่งมอบงานดังนี้ \_\_\_\_\_ รายการส่งมอบโครงการตาม  
ข้อมูล Software Configuration

ชื่อโครงการที่ส่งมอบ \_\_\_\_\_ บริษัทฯ จึงขอเรียนให้ท่านลงนามในส่วนท้ายของเอกสารฉบับนี้เพื่อเป็นการยืนยันการรับมอบงาน  
โครงการ \_\_\_\_\_ โดยที่ระยะเวลาประกันจะเริ่มตั้งแต่วันที่ \_\_\_\_\_ ถึง \_\_\_\_\_  
วันที่ \_\_\_\_\_ จึงเรียนมาเพื่อพิจารณา ระยะเวลาประกัน

ขอแสดงความนับถือ \_\_\_\_\_  
ผู้จัดการโครงการ  
บริษัท \_\_\_\_\_

ส่วนลงนามของผู้จัดการโครงการของบริษัท

ภาพที่ 4.28 ตัวอย่างเอกสารจดหมายส่งมอบปิดโครงการ (Acceptance Record) ส่วนที่ 1

ชื่อโครงการที่ส่งมอบ \_\_\_\_\_

.....

ลงนามรับมอบงานโครงการ \_\_\_\_\_ โดย \_\_\_\_\_

.....

ส่วนลงนามรับมอบงานของผู้มีอำนาจรับมอบงานของลูกค้า และวันที่รับมอบงาน

(คุณ \_\_\_\_\_)  
บริษัท \_\_\_\_\_

วันที่.....

ภาพที่ 4.29 ตัวอย่างเอกสารจดหมายส่งมอบปิดโครงการ (Acceptance Record) ส่วนที่ 2

**2.2 Activities การดำเนินงานและเอกสารที่เกี่ยวข้องขององค์กร ที่บรรลุ Objective ของ Software Implementation Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์) ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile มี ดังนี้**

**2.2.1 SI.1 Software Implementation Initiation (การเริ่มการพัฒนา/ติดตั้งซอฟต์แวร์)**

1) ผู้รับผิดชอบในโครงการสำหรับกิจกรรมนี้และบทบาทความเกี่ยวข้องมี ดังนี้

(1) Project Manager เป็นผู้จัดการประชุมร่วมกับ Team Leader/System Analyst และทีมงานดำเนินโครงการ เพื่อทบทวนรายละเอียดโครงการตามเอกสาร Project Plan เวอร์ชันล่าสุด เพื่อให้ทุกคนที่เกี่ยวข้องรับทราบรายการงานและระยะเวลาดำเนินงานของตนเองเพื่อ การเริ่มดำเนินงานในส่วนของการพัฒนา/ติดตั้งซอฟต์แวร์

(2) Team Leader/System Analyst เป็นผู้จัดการประชุมร่วมกับทีมงาน ดำเนินโครงการเพิ่มเติมเพื่อมอบหมายงานการพัฒนา/ติดตั้งซอฟต์แวร์, กำหนดรายละเอียดทางเทคนิคในด้านการพัฒนาซอฟต์แวร์และการกำหนดสถานะแวดล้อมในการพัฒนาซอฟต์แวร์ของ ทีมงานให้เป็นไปในทิศทางเดียวกันตามเอกสาร Project Plan เวอร์ชันล่าสุด

2) เอกสารที่เกี่ยวข้องกับ Activity นี้มี ดังนี้

(1) Input Products

ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15) เวอร์ชันล่าสุด

(2) Output Products

ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15) เวอร์ชันล่าสุดที่

ได้รับการทบทวนแล้ว

3) รายละเอียดเพิ่มเติมมีดังนี้

(1) ควรมอบหมายงานที่แต่ละคนต้องรับผิดชอบให้ชัดเจน

**บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.1 Software Implementation Initiation (การเริ่มการพัฒนา/ติดตั้งซอฟต์แวร์)**

SI.1 Software Implementation Initiation (การเริ่มการพัฒนา/ติดตั้งซอฟต์แวร์) เป็นการทบทวนรายละเอียดโครงการและรายการงานที่แต่ละคนที่เกี่ยวข้องจะต้องรับผิดชอบตาม เอกสาร Project Plan เวอร์ชันล่าสุด โดย Project Manager เป็นผู้ทบทวนรายละเอียดโครงการและ มอบหมายงานในภาพรวมให้กับ **Team Leader/System Analyst** และทีมงานดำเนินโครงการ ส่วน **Team Leader/System Analyst** เป็นผู้มอบหมายงานและกำหนดรายละเอียดทางเทคนิคในด้านการ

พัฒนาซอฟต์แวร์และสถานะแวดล้อมในการพัฒนาซอฟต์แวร์ของทีมงานให้เป็นไปได้ในทิศทางเดียวกัน โดยมีรายละเอียดแนวปฏิบัติที่ดีดังนี้

ตารางที่ 4.5 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity  
SI.1 Software Implementation Initiation (การเริ่มการพัฒนา/ติดตั้งซอฟต์แวร์)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>1. ทบทวนรายละเอียดโครงการตามเอกสาร <b>Project Plan</b> เวอร์ชันล่าสุด เพื่อให้ทุกคนที่เกี่ยวข้องรับทราบรายการงานและระยะเวลาดำเนินงานของตนเองเพื่อการเริ่มดำเนินงานในส่วนของการพัฒนา/ติดตั้งซอฟต์แวร์</p>	<p><b>Input Products</b></p> <p>- เอกสาร Project Plan เวอร์ชันล่าสุด</p> <p><b>Output Products</b></p> <p>- เอกสาร Project Plan เวอร์ชันล่าสุด ที่ได้รับการทบทวนแล้ว</p>	<p>- <b>Project Manager</b> เป็นผู้จัดการประชุมร่วมกับ <b>Team Leader/System Analyst</b> และทีมงานดำเนินโครงการ</p>
<p>*ตัวอย่างเอกสาร Project Plan ดังภาพที่ 4.3-4.15</p>		
<p>2. มอบหมายงานการพัฒนา/ติดตั้งซอฟต์แวร์, กำหนดรายละเอียดทางเทคนิคในด้านการพัฒนาซอฟต์แวร์และการกำหนดสถานะแวดล้อมในการพัฒนาซอฟต์แวร์ของทีมงานตามเอกสาร <b>Project Plan</b> เวอร์ชันล่าสุด</p>	<p><b>Input Products</b></p> <p>- เอกสาร Project Plan เวอร์ชันล่าสุด</p> <p><b>Output Products</b></p> <p>- เอกสาร Project Plan เวอร์ชันล่าสุด ที่ได้รับการทบทวนแล้ว</p>	<p>- <b>Team Leader/System Analyst</b> เป็นผู้จัดการประชุมร่วมกับ <b>ทีมงาน</b> ดำเนินโครงการ</p>
<p>*ตัวอย่างเอกสาร Project Plan ดังภาพที่ 4.3-4.15</p>		

### 2.2.2 *SI.2 Software Requirements Analysis (การวิเคราะห์ความต้องการซอฟต์แวร์)*

1) ผู้รับผิดชอบในโครงการสำหรับ Activity นี้และบทบาทความเกี่ยวข้องมีดังนี้

(1) Team Leader/System Analyst เป็นผู้จัดการประชุมร่วมกับลูกค้าเพื่อรวบรวมรายละเอียดความต้องการซอฟต์แวร์ตามเอกสาร Project Plan เวอร์ชันล่าสุด โดยให้คำปรึกษาและแนะนำเกี่ยวกับแนวทางการพัฒนาซอฟต์แวร์เพื่อให้สอดคล้องตามความต้องการซอฟต์แวร์ของลูกค้า, ดำเนินการวิเคราะห์ความต้องการซอฟต์แวร์ของลูกค้า, แล้วทำการออกแบบตัวอย่าง User Interface และจัดทำเป็นเอกสาร Software Requirements Specification โดยข้อมูลที่สำคัญที่ควรรวบรวมจากลูกค้ามี ดังนี้

ก. บทบาทการทำงานของ User กับซอฟต์แวร์ว่าประกอบด้วยบทบาทใดบ้าง แต่ละบทบาทมีหน้าที่ทำงานกับซอฟต์แวร์อย่างไรบ้าง

ข. กระบวนการทำงานในปัจจุบันที่ต้องการนำมาพัฒนาขึ้นเป็นซอฟต์แวร์

ค. ตัวอย่างเอกสารที่เป็น Hard Copy และ/หรือ Soft Copy ที่ใช้งานอยู่ในปัจจุบันตามที่ต้องการนำมาพัฒนาขึ้นเป็นซอฟต์แวร์

ง. ข้อกำหนดหรือกฎเกณฑ์ต่างๆ เกี่ยวกับการทำงานของซอฟต์แวร์

จ. ฐานข้อมูลจากแหล่งต่างๆ ที่มีความเกี่ยวข้องกับการพัฒนาซอฟต์แวร์

ฉ. ความต้องการเชื่อมโยงข้อมูลระหว่างซอฟต์แวร์ที่จะพัฒนาขึ้นกับ

ระบบอื่น

ช. ตัวอย่างรูปแบบรายงาน

(2) Team Leader/System Analyst ทบทวนรายละเอียดในเอกสาร Software Requirements Specification เปรียบเทียบกับรายละเอียดในเอกสาร Project Plan เวอร์ชันล่าสุด แล้วปรับปรุงแก้ไขให้ถูกต้องตรงกับเอกสาร Project Plan หากมีความต้องการซอฟต์แวร์ส่วนใดที่เพิ่มเติมจากเอกสาร Project Plan เวอร์ชันล่าสุด ให้จัดทำเอกสาร Change Request แล้วบันทึกสถานะการ Verify การจัดทำเอกสาร Software Requirements Specification ไว้ในเอกสาร Verification & Validation Results

(3) Team Leader/System Analyst นำเสนอเอกสาร Software Requirements Specification เพื่อให้ลูกค้าทบทวนรายละเอียดตามเอกสาร Project Plan เวอร์ชันล่าสุด และพิจารณารูปแบบ User Interface ให้มีความเหมาะสมกับความต้องการใช้งานซอฟต์แวร์ เมื่อลูกค้าเห็นชอบในรายละเอียดของเอกสาร Software Requirements Specification แล้วให้ลูกค้าลงนามยืนยันในเอกสาร Software Requirements Specification และ Team Leader/System Analyst บันทึก

สถานะเอกสาร Software Requirements Specification ผ่านการ Validate แล้วไว้ในเอกสาร Verification & Validation Results

กรณีที่มีความต้องการซอฟต์แวร์เพิ่มเติมจากเอกสาร Project Plan เวอร์ชันล่าสุด ให้จัดทำเอกสาร Change Request และนำเสนอให้ลูกค้าลงนามยืนยันในเอกสาร Change Request เพิ่มเติม

2) เอกสารที่เกี่ยวข้องกับ Activity นี้มี ดังนี้

(1) Input Products

- ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15) เวอร์ชันล่าสุด
- ข. เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34) ฉบับร่าง
- ค. เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34) ที่ผ่านการ Verify แล้ว
- ง. เอกสาร Change Request (ตัวอย่างดังภาพที่ 4.24-4.25) (ถ้ามี)

(2) Output Products

- ก. เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34) ที่ผ่านการ Validate แล้วได้รับการลงนามยืนยันจากลูกค้าแล้ว
- ข. เอกสาร Change Request (ตัวอย่างดังภาพที่ 4.24-4.25) ที่ได้รับการลงนามยืนยันจากลูกค้าแล้ว (ถ้ามี)
- ค. เอกสาร Verification & Validation Results (ตัวอย่างดังภาพที่ 4.16-4.17) ที่บันทึกสถานะการ Verify และ Validate เอกสาร Software Requirements Specification แล้ว

3) รายละเอียดเพิ่มเติมมี ดังนี้

(1) ควรให้ลูกค้าลงนามรับรองเอกสาร Software Requirements Specification และเอกสาร Change Request (ถ้ามี) เพื่อเป็นการยืนยันอย่างเป็นทางการเป็นลายลักษณ์อักษร

**บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.2 Software Requirements Analysis (การวิเคราะห์ความต้องการซอฟต์แวร์)**

SI.2 Software Requirements Analysis (การวิเคราะห์ความต้องการซอฟต์แวร์) เป็นการดำเนินงานโดยบทบาท **Team Leader/System Analyst** เพื่อทำการรวบรวมความต้องการซอฟต์แวร์จากลูกค้า โดยให้คำปรึกษาและแนะนำแนวทางการพัฒนาซอฟต์แวร์อย่างเหมาะสมระหว่างการประชุมร่วมกับลูกค้าได้ แล้วนำข้อมูลที่ได้รวบรวมนั้นมาทำการวิเคราะห์ เพื่อออกแบบ



ตัวอย่าง User Interface และเรียบเรียงข้อมูลความต้องการซอฟต์แวร์อย่างเป็นระบบเพื่อจัดทำเป็นเอกสาร Software Requirements Specification ซึ่งกิจกรรมนี้นับว่าเป็นกิจกรรมที่มีความสำคัญมาก กิจกรรมหนึ่งเพราะหากผู้ดำเนินกิจกรรมนี้มีทักษะในการการออกแบบและจัดทำ เอกสาร Software Requirements Specification ได้ครอบคลุมความต้องการซอฟต์แวร์ก็จะช่วยให้ทีมงานดำเนินโครงการโดยเฉพาะบทบาท Programmer จะสามารถพัฒนาซอฟต์แวร์ได้อย่างรวดเร็วและถูกต้องตรงตามความต้องการซอฟต์แวร์ได้ โดยมีรายละเอียดแนวปฏิบัติที่ดีและเอกสารที่เกี่ยวข้องกับ Activity นี้ ดังนี้

ตารางที่ 4.6 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.2 Software Requirements Analysis (การวิเคราะห์ความต้องการซอฟต์แวร์)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<b>1. จัดทำเอกสาร Software Requirements Specification ฉบับร่าง</b> โดยการประชุมร่วมกับลูกค้าเพื่อรวบรวมรายละเอียดความต้องการซอฟต์แวร์ตามเอกสาร Project Plan เวอร์ชันล่าสุด โดยให้คำปรึกษาและแนะนำเกี่ยวกับแนวทางการพัฒนาซอฟต์แวร์เพื่อให้สอดคล้องตามความต้องการซอฟต์แวร์ของลูกค้า, ดำเนินการวิเคราะห์ความต้องการซอฟต์แวร์ของลูกค้า, แล้วทำการออกแบบตัวอย่าง User Interface และจัดทำเป็นเอกสาร Software Requirements Specification โดยข้อมูลที่สำคัญที่ควรรวบรวมจากลูกค้ามีดังนี้	<b>Input Products</b> - เอกสาร Project Plan เวอร์ชันล่าสุด <b>Output Products</b> - เอกสาร Software Requirements Specification ฉบับร่าง	- Team <b>Leader/System Analyst</b> เป็นผู้ดำเนินการจัดประชุมกับลูกค้า และจัดทำเอกสาร Software Requirements Specification ฉบับร่าง
<ul style="list-style-type: none"> <li>○ บทบาทการทำงานของ User กับซอฟต์แวร์ว่าประกอบด้วยบทบาทใดบ้าง แต่ละบทบาทมีหน้าที่ทำงานกับซอฟต์แวร์อย่างไรบ้าง</li> </ul>		

## ตารางที่ 4.6 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบในโครงการ
<ul style="list-style-type: none"> <li>○ กระบวนการทำงานในปัจจุบันที่ ต้องการนำมาพัฒนาขึ้นเป็น ซอฟต์แวร์</li> <li>○ ตัวอย่างเอกสารที่เป็น Hard Copy และ/หรือ Soft Copy ที่ใช้งานอยู่ใน ปัจจุบันตามที่ต้องการนำมา พัฒนาขึ้นเป็นซอฟต์แวร์</li> <li>○ ข้อกำหนดหรือกฎเกณฑ์ต่างๆ เกี่ยวกับการทำงานของซอฟต์แวร์</li> <li>○ ฐานข้อมูลจากแหล่งต่างๆ ที่มีความ เกี่ยวข้องกับการพัฒนาซอฟต์แวร์</li> <li>○ ความต้องการเชื่อมโยงข้อมูล ระหว่างซอฟต์แวร์ที่จะพัฒนาขึ้น กับระบบอื่น</li> <li>○ ตัวอย่างรูปแบบรายงานเอกสาร Software Requirements Specification ควรประกอบด้วย ข้อมูลที่สำคัญดังต่อไปนี้</li> <li>○ ชื่อ โครงการ, ชื่อ โปรแกรม/ชื่อ ระบบ, ชื่อลูกค้า</li> <li>○ รายชื่อทีมงานดำเนิน โครงการ (Project Member)</li> <li>○ รายการ Requirement แยกตามแต่ ละบทบาท (Role) การทำงานกับ โปรแกรม (ระบบ) หรือแยกตามแต่ ละ Function การทำงานของโปรแกรม</li> </ul>		

## ตารางที่ 4.6 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบในโครงการ
<ul style="list-style-type: none"> <li>○ (ระบบ)</li> <li>○ ตัวอย่าง User Interface</li> <li>○ มีการบันทึกประวัติการแก้ไข (Revision History)</li> <li>○ ส่วนลงนามยืนยันจากลูกค้า</li> </ul> <p>*ตัวอย่างเอกสาร Software Requirements Specification ดังภาพที่ 4.30-4.34</p> <p>*ตัวอย่างเอกสาร Project Plan ดังภาพที่ 4.3-4.15</p>		
<p><b>2. ทบทวนรายละเอียดในเอกสาร Software Requirements Specification เปรียบเทียบกับรายละเอียดในเอกสาร Project Plan เวอร์ชันล่าสุด แล้วปรับปรุงแก้ไขให้ถูกต้องตรงกับเอกสาร Project Plan และบันทึกสถานะการ Verify การจัดทำเอกสาร Software Requirements Specification ไว้ในเอกสาร Verification &amp; Validation Results</b></p> <p><b>หมายเหตุ:</b>หากมีความต้องซอฟต์แวร์ส่วนใดที่เพิ่มเติมจากเอกสาร Project Plan เวอร์ชันล่าสุด ให้จัดทำเอกสาร Change Request แล้วนำเสนอเพื่อให้ลูกค้าลงนามยืนยัน</p>	<p><b><u>Input Products</u></b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan เวอร์ชันล่าสุด</li> <li>- เอกสาร Software Requirements Specification ฉบับร่าง</li> </ul> <p><b><u>Output Products</u></b></p> <ul style="list-style-type: none"> <li>- เอกสาร Software Requirements Specification เวอร์ชันที่ผ่านการ Verify แล้ว</li> <li>- เอกสาร Verification &amp; Validation Results ที่ได้รับการบันทึก</li> </ul>	<p><b>- Team Leader/System Analyst</b> เป็นผู้ดำเนินการขั้นตอนนี้</p>

## ตารางที่ 4.6 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
*ตัวอย่างเอกสาร Software Requirements Specification ดังภาพที่ 4.30-4.34	- สถานะการ Verity เอกสาร Software Requirements Specificationแล้ว	
*ตัวอย่างเอกสาร Project Plan ดังภาพที่ 4.3-4.15		
*ตัวอย่างเอกสาร Verification & Validation Results ดังภาพที่ 4.16-4.17	- เอกสาร Change Request ฉบับร่าง (ถ้ามี)	
*ตัวอย่างเอกสาร Change Request ดังภาพที่ 4.24-4.25		
<b>3. นำเสนอเอกสาร Software Requirements Specification เพื่อให้ลูกค้าทบทวนรายละเอียดตามเอกสาร Project Plan เวอร์ชันล่าสุด และพิจารณารูปแบบ User</b>	<b>Input Products</b> - เอกสาร Project Plan เวอร์ชันล่าสุด - เอกสาร Software	- <b>Team Leader/System Analyst</b> เป็นผู้ดำเนินการขั้นตอนนี้
<b>4. Interface ให้มีความเหมาะสมกับความต้องการใช้งานซอฟต์แวร์ เมื่อลูกค้าเห็นชอบในรายละเอียดของเอกสาร Software Requirements Specification แล้วให้ลูกค้าลงนามยืนยันในเอกสาร Software Requirements Specification และบันทึกสถานะเอกสาร Software Requirements Specification ผ่านการ Validate แล้วไว้ในเอกสาร Verification &amp; Validation Results</b>	- Requirements Specification เวอร์ชันที่ผ่านการ Verify แล้ว <b>Output Products</b> - เอกสาร Software Requirements Specification เวอร์ชันที่ได้รับการลงนามยืนยันจากลูกค้าและผ่านการ Validate แล้วและกำหนดเป็นเวอร์ชัน Baseline - เอกสาร Verification & Validation Results ที่ได้รับการบันทึกสถานะการ	

ตารางที่ 4.6 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
*ตัวอย่างเอกสาร Software Requirements Specification ดังภาพที่ 4.30-4.34	- Validate เอกสาร Software Requirements Specification แล้ว	
*ตัวอย่างเอกสาร Project Plan ดังภาพที่ 4.3-4.15		
*ตัวอย่างเอกสาร Verification & Validation Results ดังภาพที่ 4.16-4.17	เอกสาร Change Request	
*ตัวอย่างเอกสาร Change Request ดังภาพที่ 4.24-4.25	ที่ได้รับการลงนามยืนยันจากลูกค้าแล้ว (ถ้ามี)	

ชื่อโครงการ, ชื่อโปรแกรม/ระบบ

รหัสเอกสาร

Software Requirements Specification (SRS)

SRS-TMS-BPP-54-03A-290854

ชื่อลูกค้า

ทีมงานดำเนินโครงการ

ส่วนบันทึกประวัติการแก้ไขและควบคุมเวอร์ชันของเอกสาร

Revision History:

Version	Date	Author	Reviewer	Description
1.0	1 สิงหาคม 2554	คุณ _____	คุณ _____	Initial
2.0	2 สิงหาคม 2554	คุณ _____	คุณ _____	Baseline
3.0	29 สิงหาคม 2554	คุณ _____	คุณ _____	เพิ่มเงื่อนไขข้อกำหนด

ภาพที่ 4.30 ตัวอย่างเอกสาร Software Requirements Specification ส่วนที่ 1

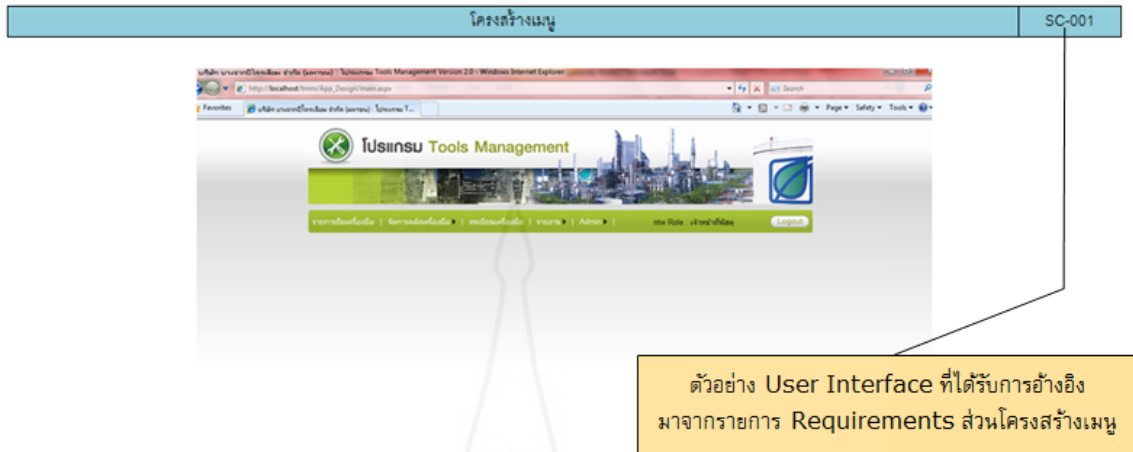
รายการ Requirements ส่วน โครงสร้างเมนู	
โปรแกรม Tools Management	
1	โครงสร้างเมนู
	<p>โปรแกรม Tools Management ประกอบด้วยโครงสร้างเมนูดังนี้</p> <p><b>ส่วน Front-end</b></p> <p>1.1. <b>รายการพิมพ์เครื่องมือ</b></p> <p>1.2. <b>จัดการคลังเครื่องมือ</b></p> <p>1.2.1. <b>รับเครื่องมือเข้าคลัง (Tools Receive)</b></p> <p>1.2.2.1. มี Function ค้นหา</p> <p>1.2.2. <b>รายการพิมพ์เครื่องมือ (ยื่นรับจ่ายเครื่องมือ)</b></p> <p>1.2.2.1. มี Function ค้นหา</p> <p>1.2.3. <b>รายการรับคืนเครื่องมือ</b></p> <p>1.2.3.1. มี Function ค้นหา</p> <p>1.2.4. <b>รายการปรับปรุงคลังเครื่องมือ (Stock Adjustment)</b></p> <p>1.2.4.1. มี Function ค้นหา</p> <p>1.3. <b>ทะเบียนเครื่องมือ ประกอบด้วย Function ดังนี้</b></p> <p>1.3.1. ค้นหา</p> <p>1.3.2. เพิ่มทะเบียนเครื่องมือ</p> <p>1.3.3. แก้ไขทะเบียนเครื่องมือ</p> <p>1.3.4. Disableทะเบียนเครื่องมือ</p> <p>1.4. <b>รายงาน</b></p> <p>1.4.1. <b>รายงาน Bin Card (Physical Count)</b></p> <p>1.4.2.1. <b>ฟอร์ม Physical Count ประกอบด้วย Function ดังนี้</b></p> <p>1.4.2.1.1. สร้างฟอร์ม</p> <p>1.4.2.1.2. แก้ไขฟอร์ม</p>
	การระบุ หมายเลขอ้างอิง รูปภาพ User Interface
	SC-001

ภาพที่ 4.31 ตัวอย่างเอกสาร Software Requirements Specification ส่วนที่ 2

รายการ Requirements ส่วนบทบาท การทำงาน	
Role การเข้าใช้งาน และสิทธิการเข้าใช้งาน	
2	Role การเข้าใช้งาน และสิทธิการเข้าใช้งาน
	<p>2.1. <b>Role เจ้าหน้าที่พัสดุ</b> ผู้ใช้ระบบสามารถทำงานตามเมนู และ Function ดังนี้</p> <p><b>ส่วน Front-end</b></p> <p>2.1.1. <b>รายการพิมพ์เครื่องมือ</b></p> <p>2.1.1.1. มองเห็นรายการพิมพ์เครื่องมือได้ทั้งระบบ</p> <p>2.1.1.2. สามารถยืนยันการจ่ายเครื่องมือไปยังไปยัง</p> <p>2.1.1.3. ยกเลิกใบยืมฯ ที่ยังไม่ได้รับการยืนยันจ่ายเครื่องมือ</p> <p>2.1.2. <b>จัดการคลังเครื่องมือ</b></p> <p>2.1.2.1. <b>รับเครื่องมือเข้าคลัง (Tools Receive)</b></p> <p>2.1.2.1.1. มี Function ค้นหา</p> <p>2.1.2.2. <b>รายการพิมพ์เครื่องมือ (ยื่นรับจ่ายเครื่องมือ)</b></p> <p>2.1.2.2.1. มี Function ค้นหา</p> <p>2.1.2.2.2. ลักษณะการใช้งานเหมือนเมนู "รายการพิมพ์เครื่องมือ"</p> <p>2.1.2.2.3. ยกเลิกใบยืมฯ ที่ยังไม่ได้รับการยืนยันจ่ายเครื่องมือ</p> <p>2.1.2.3. <b>รายการรับคืนเครื่องมือ</b></p> <p>2.1.2.3.1. มี Function ค้นหา</p> <p>2.1.2.4. <b>รายการปรับปรุงคลังเครื่องมือ (Stock Adjustment)</b></p> <p>2.1.2.4.1. มี Function ค้นหา</p> <p>2.1.3. <b>ทะเบียนเครื่องมือ ประกอบด้วย Function ดังนี้</b></p> <p>2.1.3.1. ค้นหา</p>
	การระบุ หมายเลขอ้างอิง รูปภาพ User Interface
	SC-002

ภาพที่ 4.32 ตัวอย่างเอกสาร Software Requirements Specification ส่วนที่ 3

## User Interface Design



ภาพที่ 4.33 ตัวอย่างเอกสาร Software Requirements Specification ส่วนที่ 4

ตารางสำหรับการลงนาม				
	Full Name	Role	Date	Signature
Prepare by:	คุณ _____	Team Leader/System Analyst	29 สิงหาคม 2554	
Review by:	คุณ _____	Team Leader/System Analyst	29 สิงหาคม 2554	
Approve by:	คุณ _____	Customer	29 สิงหาคม 2554	

ภาพที่ 4.34 ตัวอย่างเอกสาร Software Requirements Specification ส่วนที่ 5

### 2.2.3 SI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้าง

#### สถาปัตยกรรมและรายละเอียดซอฟต์แวร์)

1) ผู้รับผิดชอบในโครงการสำหรับ Activity นี้และบทบาทความเกี่ยวข้อง มีดังนี้

(1) Team Leader/System Analyst ทบทวนและทำความเข้าใจในรายละเอียดของเอกสาร Software Requirements Specification เวอร์ชัน Baseline แล้วทำการออกแบบและจัดทำเอกสาร Software Design (เอกสารออกแบบระบบ) โดยเอกสารออกแบบระบบควรครอบคลุมข้อมูลหรือเอกสารย่อยที่สำคัญ ดังนี้



- ก. โครงสร้างสถาปัตยกรรมระบบ
- ข. Process Flow Diagram
- ค. Data Interface Diagram (กรณีมีการเชื่อมโยงข้อมูลกับระบบอื่น)
- ง. ER Diagram
- จ. Data Dictionary
- ฉ. Program Structure

**หมายเหตุ:** เอกสาร Program Structure เป็นข้อมูลที่สามารถแสดงให้เห็นถึงการแบ่งแยกซอฟต์แวร์ออกเป็นซอฟต์แวร์ Module ย่อย (Software Components) เพื่อสามารถนำไปเชื่อมโยงความสัมพันธ์กับข้อมูลในเอกสารข้อเสนอขอบข่ายงาน, เอกสาร Software Requirements Specification, และเอกสาร Test Cases and Test Procedures ได้โดยการจัดทำเป็นเอกสาร Traceability Record สำหรับการตรวจสอบกระบวนการพัฒนา/ติดตั้งซอฟต์แวร์ย้อนกลับได้อย่างมีประสิทธิภาพได้จึงขอแนะนำเฉพาะตัวอย่างเอกสาร Program Structure ส่วนข้อมูลหรือเอกสารย่อยอื่นๆ ในชุดของเอกสาร Software Design เป็นการเอกสารทั่วไปที่ต้องจัดทำประกาศออกแบบซอฟต์แวร์อยู่แล้วจึงไม่ขอแนะนำตัวอย่างในรายงานฉบับนี้

(2) Team Leader/System Analyst จัดทำเอกสาร Traceability Record โดยการเชื่อมโยงความสัมพันธ์ระหว่างรายการขอบข่ายงานในเอกสารเสนอราคา, รายการซอฟต์แวร์ Module ย่อย (Software Components) ตามรายละเอียดที่กำหนดไว้ในเอกสาร Program Structure, และรายการความต้องการซอฟต์แวร์ในเอกสาร Software Requirements Specification ให้สอดคล้องตรงกัน

(3) Team Leader/System Analyst ทบทวนรายละเอียดเอกสาร ดังนี้

ก. ทบทวนและปรับปรุงรายละเอียดเอกสารย่อยต่างๆ ในชุดของเอกสาร Software Design ให้ครบถ้วนตามรายละเอียดในเอกสาร Software Requirements Specification เวอร์ชัน Baseline

ข. ทบทวนและปรับปรุงเอกสาร Traceability Record ให้ครบถ้วนตามรายการขอบข่ายงานในเอกสารเสนอราคา, เอกสาร Program Structure ในชุดของเอกสาร Software Design, และเอกสาร Software Requirements Specification เวอร์ชัน Baseline

**หมายเหตุ:** หากมีความต้องการซอฟต์แวร์ส่วนใดที่เพิ่มเติมจากเอกสาร Software Requirements Specification เวอร์ชัน Baseline ให้จัดทำเอกสาร Change Request

เมื่อ **Team Leader/System Analyst** ทบทวนรายละเอียดและปรับปรุงเอกสาร Traceability Record เรียบร้อยแล้ว ให้ทำการบันทึกสถานะการ Verify การจัดทำ

เอกสาร Software Design และเอกสาร Traceability Record ไว้ในเอกสาร Verification & Validation Results

(4) Tester จัดทำเอกสาร Test Cases and Test Procedures และเอกสาร Test Report อ้างอิงตามรายละเอียดของเอกสาร Software Requirements Specification

(5) Tester ทบทวนรายละเอียดในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report ร่วมกับ Team Leader/System Analyst อ้างอิงตามรายละเอียดของเอกสาร Software Requirements บันทึกสถานะการ Verify และ Validate การจัดทำเอกสาร Test Cases and Test Procedures ไว้ในเอกสาร Verification & Validation Results

(6) Tester ปรับปรุงเอกสาร Traceability Record โดยเพิ่มรหัสรายการทดสอบระบบตามรายละเอียดในเอกสาร Test Cases and Test Procedures ตามที่มีความสัมพันธ์กับรายการหัวข้อ Requirements ในเอกสาร Software Requirements Specification เวอร์ชัน Baseline

(7) Team Leader/System Analyst ทบทวนรายละเอียดในเอกสาร Software Design และเอกสาร Traceability Record แล้วนำเอกสาร Software Design นำเสนอและให้ลูกค้าลงนามยืนยันและกำหนดเวอร์ชัน Baseline และกำหนดเอกสาร Traceability Record เป็นเวอร์ชัน Baseline

(8) Team Leader/System Analyst เมื่อเอกสาร Test Cases and Test Procedures และ Test Report ได้รับการบันทึกผลผ่านการทดสอบโดย Tester และผ่านการแก้ไขโดย Programmer เรียบร้อยแล้ว Team Leader/System Analyst จะนำเอกสาร Test Cases and Test Procedures และ Test Report เวอร์ชันที่ผ่านการ Validate แล้ว (ก่อนการบันทึกผลการทดสอบโดย Tester ของทีมงานดำเนินโครงการ) มาจัดทำเป็นเอกสารสำหรับการดำเนินการ User Acceptance Test ของลูกค้า และนำเสนอให้ลูกค้าลงนามยืนยันผลการทดสอบระบบเมื่อระบบผ่านการ User Acceptance Test ของลูกค้าเรียบร้อยแล้ว

## 2) เอกสารที่เกี่ยวข้องกับ Activity นี้มี ดังนี้

### (1) Input Products

ก. เอกสารเสนอราคาที่มีการระบุขอบข่ายงานที่ได้รับการยืนยันการว่าจ้างจากลูกค้า

ข. เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34) เวอร์ชัน Baseline

ค. เอกสาร Software Design ฉบับร่าง (ตัวอย่างดังภาพที่ 4.35-4.37)

ง. เอกสารย่อย Program Structure ในชุดของเอกสาร Software Design (ตัวอย่างผังภาพที่ 4.38-4.39)

จ. เอกสาร Traceability Record ฉบับร่าง (ตัวอย่างผังภาพที่ 4.40-4.41)

ฉ. เอกสาร Change Request (ตัวอย่างผังภาพที่ 4.24-4.25) (ถ้ามี)

(2) Output Products

ก. เอกสาร Software Design (ตัวอย่างผังภาพที่ 4.35-4.37) ที่ได้รับการลงนามยืนยันจากลูกค้าแล้ว และกำหนดเป็นเวอร์ชัน Baseline

ข. เอกสาร Traceability Record (ตัวอย่างผังภาพที่ 4.40-4.41) เวอร์ชัน Baseline

ค. เอกสาร Test Cases and Test Procedures (ตัวอย่างผังภาพที่ 4.42-4.43) ที่ได้รับการ Validate แล้ว

ง. เอกสาร Test Report (ตัวอย่างผังภาพที่ 4.44-4.45) ที่ได้รับการ Validate แล้ว

จ. เอกสาร Test Cases and Test Procedures (ตัวอย่างผังภาพที่ 4.42-4.43) ที่ได้รับการลงนามยืนยันผลการทดสอบ User Acceptance Test จากลูกค้าแล้ว

ฉ. เอกสาร Test Report (ตัวอย่างผังภาพที่ 4.44-4.45) ที่ได้รับการลงนามยืนยันผลการทดสอบ User Acceptance Test จากลูกค้าแล้ว

ช. เอกสาร Change Request (ตัวอย่างผังภาพที่ 4.24-4.25) ที่ได้รับการลงนามยืนยันจากลูกค้าแล้ว (ถ้ามี)

ซ. เอกสาร Verification & Validation Results (ตัวอย่างผังภาพที่ 4.16-4.17) ที่บันทึกสถานะการ Verify เอกสาร Software Design, เอกสาร Traceability Record, และเอกสาร Test Cases and Test Procedures

3) รายละเอียดเพิ่มเติมมี ดังนี้

(1) ควรให้ลูกค้าลงนามรับรองเอกสาร Software Design และเอกสาร Change Request (ถ้ามี) เพื่อเป็นการยืนยันอย่างเป็นทางการเป็นลายลักษณ์อักษร

**บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ ActivitySI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้างสถาปัตยกรรมและรายละเอียดซอฟต์แวร์)**

SI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้างสถาปัตยกรรมและรายละเอียดซอฟต์แวร์) เป็นการดำเนินงาน โดยบทบาท **Team Leader/System**

**Analyst** ในส่วนของการทบทวนและวิเคราะห์เอกสาร Software Requirements Specification เวอร์ชัน Baseline เพื่อทำการออกแบบและจัดทำเอกสาร Software Design เป็นลำดับถัดมา ซึ่งเอกสารย่อยในชุดของเอกสาร Software Design ที่สำคัญสำหรับการนำมาประกอบในการใช้จัดทำเอกสาร Traceability Record คือ เอกสาร Program Structure ซึ่งควรที่จะแสดงให้เห็นถึงองค์ประกอบของ Software ในลักษณะซอฟต์แวร์ Module ย่อย (Software Components) ได้ หลังจากนั้น **Team Leader/System Analyst** จะเป็นผู้เริ่มจัดทำเอกสาร Traceability Record โดยการเชื่อมโยงความสัมพันธ์ของรายการ

ขอบข่ายงานในเอกสารเสนอราคา, เอกสาร Program Structure ในชุดของเอกสาร Software Design, และเอกสาร Software Requirements Specification เวอร์ชัน Baseline เพื่อจะสามารถใช้เอกสาร

Traceability Record ในการตรวจสอบย้อนกลับกระบวนการพัฒนาซอฟต์แวร์ ได้ว่าสามารถดำเนินการพัฒนาระบบและทดสอบระบบได้ครอบคลุมตามความต้องการซอฟต์แวร์ตามที่ได้รับการว่าจ้างงานอย่างครบถ้วนแล้วหรือไม่

หลังจาก **Team Leader/System Analyst** ออกแบบระบบและจัดทำเอกสาร Software Design และเอกสาร Traceability Record ฉบับร่างเสร็จเรียบร้อยแล้ว **Tester** จะดำเนินการจัดทำเอกสาร Test Cases and Test Procedures อ้างอิงตามรายละเอียดของเอกสาร Software Requirements Specification เป็นลำดับต่อมา หลังจากนั้น **Tester** จะเป็นผู้ทำการ Update เอกสาร Traceability Record โดยเพิ่มรหัสหัวข้อการทดสอบตามที่กำหนดไว้ในเอกสาร Test Cases and Test Procedures ให้สอดคล้องกับรายการขอบข่ายงานในเอกสารเสนอราคา, เอกสาร Program Structure ในชุดของเอกสาร Software Design, และเอกสาร Software Requirements Specification เวอร์ชัน Baseline

หลังจากนั้น **Team Leader/System Analyst** จะเป็นผู้ทบทวนรายละเอียดในเอกสาร Software Design และเอกสาร Traceability Record แล้วนำเอกสาร Software Design นำเสนอและให้ลูกค้าลงนามยืนยันและกำหนดเวอร์ชัน Baseline และกำหนดเอกสาร Traceability Record เป็นเวอร์ชัน Baseline

นอกจากนี้เมื่อเอกสาร Test Cases and Test Procedures และ Test Report ได้รับการบันทึกผลผ่านการทดสอบโดย Tester และผ่านการแก้ไขโดย Programmer เรียบร้อยแล้ว **Team Leader/System Analyst** จะนำเอกสาร Test Cases and Test Procedures และ Test Report เวอร์ชันที่ผ่านการ Validate แล้ว (ก่อนการบันทึกผลการทดสอบโดย Tester ของทีมงานดำเนินโครงการ) มาจัดทำเป็นสำเนาเอกสารสำหรับการดำเนินการ User Acceptance Test ของลูกค้า และ

นำเสนอให้ลูกค้าลงนามยืนยันผลการทดสอบระบบเมื่อระบบผ่านการ User Acceptance Test ของลูกค้าเรียบร้อยแล้ว

จะเห็นได้ว่า Activity SI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้างสถาปัตยกรรมและรายละเอียดซอฟต์แวร์) เป็นขั้นตอนการดำเนินงานที่สำคัญมากอีกขั้นตอนหนึ่งในการออกแบบโครงสร้างซอฟต์แวร์, การจัดทำเอกสารสำหรับการทดสอบซอฟต์แวร์, และการเชื่อมโยงความสัมพันธ์ของเอกสารต่างๆ เข้าด้วยกันเพื่อให้มั่นใจได้ว่าซอฟต์แวร์ที่กำลังจะได้รับการพัฒนาขึ้นนั้นมีส่วนประกอบที่ครบถ้วนตามขอบข่ายงานที่ได้รับการว่าจ้างจากลูกค้า ซึ่งหากสามารถดำเนินงานขั้นตอนนี้ได้อย่างมีประสิทธิภาพจะช่วยส่งผลให้สามารถใช้ระยะเวลาในการพัฒนาซอฟต์แวร์ได้อย่างรวดเร็วยิ่งขึ้น

ตารางที่ 4.7 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้างสถาปัตยกรรมและรายละเอียดซอฟต์แวร์)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทที่ได้รับผิดชอบในโครงการ
<b>1. จัดทำเอกสาร Software Design (เอกสารออกแบบระบบ) โดยเอกสารควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</b> <ul style="list-style-type: none"> <li>○ ชื่อโครงการ, ชื่อโปรแกรม/ชื่อระบบ, ชื่อลูกค้า</li> <li>○ ชื่อผู้ควบคุมและตรวจสอบเอกสาร Software Design</li> <li>○ การออกแบบโครงสร้างสถาปัตยกรรมระบบ (ถ้ามีการระบุไว้ในเอกสารการเสนอราคา)</li> <li>○ การออกแบบ Process Flow Diagram (ถ้ามีการระบุไว้ในเอกสารการเสนอราคา)</li> <li>○ การออกแบบ ER Diagram (ถ้ามีการระบุไว้ในเอกสารการเสนอราคา)</li> <li>○ การออกแบบ Data Dictionary (ถ้ามีการระบุไว้ใน</li> </ul>	<b>Input Products</b> - เอกสาร Software Requirements Specification เวอร์ชัน Baseline  <b>Output Products</b> - เอกสาร Software Designฉบับร่าง	<b>Team Leader/System Analyst</b> เป็นผู้ดำเนินการออกแบบและจัดทำเอกสาร Software Design

## ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<ul style="list-style-type: none"> <li>○ เอกสารการเสนอราคา)</li> <li>○ การออกแบบ Program Structure</li> <li>○ การออกแบบเอกสารอื่นๆ ตามที่มีการระบุไว้ในเอกสารการเสนอราคา</li> <li>○ มีการบันทึกประวัติการแก้ไข (Revision History)</li> <li>○ ส่วนลงนามยืนยันจากลูกค้า</li> </ul> <p>*ตัวอย่างเอกสาร Software Design ดังภาพที่ 4.35-4.37</p> <p>*ตัวอย่างเอกสารย่อย Program Structure ในชุดตัวอย่างเอกสาร Software Design ดังภาพที่ 4.38-4.39</p>		
<p><b>2. จัดทำเอกสาร Traceability Record</b> โดยการเชื่อมโยงความสัมพันธ์ระหว่างรายการขอบข่ายงานในเอกสารเสนอราคา, รายการซอฟต์แวร์ Module ย่อย (Software Components) ตามรายละเอียดที่กำหนดไว้ในเอกสาร Program Structure, และรายการความต้องการซอฟต์แวร์ในเอกสาร Software Requirements Specification ให้สอดคล้องตรงกันโดยเอกสารควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</p> <ul style="list-style-type: none"> <li>○ ชื่อโครงการ, ชื่อโปรแกรม/ชื่อระบบ, ชื่อลูกค้า</li> <li>○ ชื่อผู้ดูแลเอกสาร</li> <li>○ ชื่อผู้ปรับปรุงเอกสาร</li> <li>○ ตาราง Requirements Traceability Matrix (RTM) ที่มีการเชื่อมโยงความสัมพันธ์ข้อมูลในเอกสารต่างๆ ที่เกี่ยวข้องกัน ควร</li> </ul>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Software Requirements Specification เวอร์ชัน Baseline</li> <li>- เอกสารเสนอราคาที่ได้รับกรยืนยันว่าจ้างงาน</li> <li>- เอกสารย่อย Program Structure ในชุดเอกสาร Software Design</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Traceability</li> </ul>	<p><b>Team</b></p> <p><b>Leader/System Analyst</b> เป็นผู้จัดทำเอกสาร Traceability Record ฉบับร่าง</p>



## ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<ul style="list-style-type: none"> <li>○ ประกอบด้วยข้อมูลที่สำคัญดังนี้               <ul style="list-style-type: none"> <li>- หัวข้อ Requirements อ้างอิงตามเอกสาร Software Requirements Specification</li> <li>- หัวข้อขอบข่ายงานตามเอกสารเสนอราคาที่ได้รับ การยืนยันว่าจ้างงาน</li> <li>- หัวข้อซอฟต์แวร์Module ย่อย (Software Components) ตาม รายละเอียดที่กำหนดไว้ในเอกสารย่อยProgram Structureในชุดเอกสาร Software Design</li> <li>- รหัสรายการทดสอบ ระบบตามรายละเอียดในเอกสาร Test Cases and Test Procedures</li> <li>- มีการบันทึกประวัติการแก้ไข (Revision History)</li> </ul> </li> </ul>	Record ฉบับร่าง	
*ตัวอย่างเอกสาร Traceability Record ดังภาพที่ 4.40-4.41		



ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<b>3. ทบทวนรายละเอียดในเอกสาร Software Design และเอกสาร Traceability Record ให้ถูกต้องครบถ้วนตรงกับเอกสาร Software Requirements Specification เวอร์ชัน Baseline และเอกสารเสนอราคาที่ได้รับการยืนยันว่าจ้างงานแล้วบันทึกสถานะการ Verify การจัดทำเอกสาร Software Design และเอกสาร Traceability Record ไว้ในเอกสาร Verification</b>	<p><b><u>Input Products</u></b></p> <ul style="list-style-type: none"> <li>- เอกสาร Software Design ฉบับร่าง</li> <li>- เอกสาร Traceability Record ฉบับร่าง</li> <li>- เอกสาร Software Requirements Specification เวอร์ชัน Baseline</li> <li>- เอกสารเสนอราคาที่ได้รับการยืนยันว่าจ้างงาน</li> </ul> <p><b><u>Output Products</u></b></p> <ul style="list-style-type: none"> <li>เอกสาร Software Design เวอร์ชันที่ผ่านการ Verify แล้ว</li> <li>- เอกสาร Traceability Record เวอร์ชันที่ผ่านการ Verify แล้ว</li> </ul>	<p><b>Team</b></p> <p><b>Leader/System Analyst</b> เป็นผู้ดำเนินการขั้นตอนนี้</p>

## ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>*ตัวอย่างเอกสาร Software Design ดังภาพที่ 4.35-4.37</p> <p>*ตัวอย่างเอกสาร Traceability Record ดังภาพที่ 4.40-4.41</p> <p>*ตัวอย่างเอกสาร Verification &amp; Validation Results ดังภาพที่ 4.16-4.17</p> <p>*ตัวอย่างเอกสาร Change Request ดังภาพที่ 4.24-4.25</p>	<p>- เอกสาร Verification &amp;Validation Results ที่ได้รับ การบันทึก สถานะการVerify เอกสาร Software Design และ เอกสาร Traceability Record</p> <p>- เอกสาร Change Request ฉบับร่าง (ถ้ามี)</p>	
<p><b>4. จัดทำเอกสาร Test Cases and Test Procedures และเอกสาร Test Report อังอิงตามรายละเอียดของเอกสาร Software Requirements Specification เพื่อใช้ประกอบการทดสอบซอฟต์แวร์</b></p> <p>○ เอกสาร Test Cases and Test Procedures ควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</p>	<p><b>Input Products</b></p> <p>- เอกสาร Software Requirements Specification เวอร์ชัน Baseline</p>	<p>- Tester เป็นผู้ดำเนินการขั้นตอนนี้</p>

## ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<b><u>Output Products</u></b>		
<ul style="list-style-type: none"> <li>- ชื่อโครงการ, ชื่อโปรแกรม/ชื่อระบบ, ชื่อลูกค้า</li> <li>- รายชื่อทีมงานดำเนินโครงการ (Project Member)</li> </ul>	<ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures ฉบับร่าง</li> <li>- เอกสาร Test Report ฉบับร่าง</li> </ul>	
<ul style="list-style-type: none"> <li>- ตารางบันทึกผลการทดสอบ ควรประกอบด้วยข้อมูลที่สำคัญ ดังนี้               <ul style="list-style-type: none"> <li>● รหัส (Test ID) และชื่อ Module ที่ทดสอบ</li> <li>● รหัส (Test Case ID) และวัตถุประสงค์ของการทดสอบแยกแต่ละกรณี</li> <li>● ระบุเลขที่หัวข้อของเอกสาร Software Requirements Specification ที่ตรงกับรหัส (Test Case ID) ที่ทำการทดสอบ</li> </ul> </li> </ul>		

## ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
	<ul style="list-style-type: none"> <li>● ขั้นตอนการทดสอบ (Test Procedure) แยกแต่ละ Test Case ID โดยระบุ Data Input และ Expected Result</li> <li>● สถานะการทดสอบแยกแต่ละขั้นตอนการทดสอบ (Test Procedure)</li> <li>- มีการบันทึกประวัติการแก้ไข (Revision History)</li> <li>- ส่วนลงนามยืนยันจากลูกค้า</li> </ul>	
<p>○ เอกสาร Test Report ควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</p> <ul style="list-style-type: none"> <li>- ชื่อโครงการ, ชื่อโปรแกรม/ชื่อระบบ, ชื่อลูกค้า</li> </ul>		

## ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<ul style="list-style-type: none"> <li>- รายชื่อทีมงานดำเนินโครงการ (Project Member)</li> <li>- ตารางสรุปผลการทดสอบ ควรประกอบด้วยข้อมูลที่ สำคัญดังนี้               <ul style="list-style-type: none"> <li>● รหัส (Test ID) และชื่อ Module ที่ทดสอบ</li> <li>● รหัส (Test Case ID) และ วัตถุประสงค์ของ การทดสอบแยก แต่ละกรณี</li> <li>● สถานะการ ทดสอบแยกแต่ ละ Test Case ID</li> </ul> </li> <li>- มีการบันทึกประวัติการ แก้ไข (Revision History)</li> <li>- ส่วนลงนามยืนยันจาก ลูกค้า</li> </ul>		
*ตัวอย่างเอกสาร Test Cases and Test Procedures ดังภาพที่ 4.42-4.43		
*ตัวอย่างเอกสาร Test Report ดังภาพที่ 4.44-4.45		

ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p><b>5. ทบทวนรายละเอียดในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report อ้างอิงตามรายละเอียดของเอกสาร Software Requirements Specification เวอร์ชัน Baseline แล้วบันทึกสถานะการ Verify และ Validate การจัดทำเอกสาร Test Cases and Test Procedures และ Test Report ไว้ในเอกสาร Verification &amp; Validation Results</b></p>	<p><b><u>Input Products</u></b></p> <ul style="list-style-type: none"> <li>- เอกสาร Software Requirements Specification เวอร์ชัน Baseline</li> <li>- เอกสาร Test Cases and Test Procedures ฉบับร่างเอกสาร Test Report ฉบับร่าง</li> </ul> <p><b><u>Output Products</u></b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ผ่านการ Verify และ Validate แล้ว</li> <li>- เอกสาร Test Report เวอร์ชันที่ผ่านการ Verify และ Validate แล้ว</li> </ul>	<p><b>Team</b></p> <p><b>Leader/System Analyst และ Tester</b> ทบทวนเอกสารร่วมกัน</p> <p><b>- Team Leader/System Analyst</b> เป็นผู้บันทึกสถานะการ Verify และ Validate เอกสาร Test Cases and Test Procedures และ เอกสาร Test Report ในเอกสาร Verification &amp; Validation Results</p>

## ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>*ตัวอย่างเอกสาร Test Cases and Test Procedures ดังภาพที่ 4.42-4.43</p> <p>*ตัวอย่างเอกสาร Test Report ดังภาพที่ 4.44-4.45</p> <p>*ตัวอย่างเอกสาร Verification &amp; Validation Results ดังภาพที่ 4.16-4.17</p>	<p>- เอกสาร Verification &amp; Validation Resultsที่ได้รับ การบันทึก สถานะการ Verify และ Validate เอกสาร Test Cases and Test Procedures และเอกสาร Test Report</p>	
<p><b>6. ปรับปรุงเอกสาร Traceability Record โดยเพิ่มรหัสรายการทดสอบระบบตามรายละเอียดในเอกสาร Test Cases and Test Procedures ตามที่มีความสัมพันธ์กับรายการหัวข้อ Requirements ในเอกสาร Software Requirements Specification เวอร์ชัน Baseline</b></p>	<p><b>Input Products</b></p> <p>- เอกสาร Software Requirements Specification เวอร์ชัน Baseline</p> <p>- เอกสาร Traceability Record ฉบับร่าง</p> <p>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ผ่านการ Verify และ Validate แล้ว</p>	<p>- Tester เป็นผู้ดำเนินการขั้นตอนนี้</p>



## ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
*ตัวอย่างเอกสาร Test Cases and Test Procedures ดังภาพที่ 4.42-4.43	<b>Output Products</b> - เอกสาร Traceability Record เวอร์ชันที่ ได้รับการเพิ่ม รหัสหัวข้อการ ทดสอบ	-
*ตัวอย่างเอกสาร Traceability Record ดังภาพที่ 4.40-4.41		
<b>7. ทบทวนเอกสาร Software Design และเอกสาร Traceability Record แล้วกำหนดเอกสาร Software Design และเอกสาร Traceability Record เป็นเวอร์ชัน Baseline แล้วนำเอกสาร Software Design ให้ลูกค้ายืนยันการออกแบบระบบ</b>	<b>Input Products</b> - เอกสาร Software Design เวอร์ชันที่ผ่านการ Verify แล้ว - เอกสาร Traceability Record เวอร์ชันที่ได้รับการปรับปรุงเพิ่มรหัสรายการทดสอบระบบตามรายละเอียดในเอกสาร Test Cases and Test Procedures ตามที่มีความสัมพันธ์กับรายการหัวข้อ Requirements ใน	- <b>Team Leader/System Analyst</b> เป็นผู้ดำเนินการ ชั้นตอนนี้

## ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
	เอกสาร Software Requirements Specification เวอร์ชัน Baseline แล้ว <b>Output Products</b> - เอกสาร Software Design เวอร์ชันที่ได้รับการลงนาม ยืนยันจากลูกค้า และกำหนดเป็น เวอร์ชัน Baseline - เอกสาร Traceability Record เวอร์ชันที่ได้รับการกำหนด เป็นเวอร์ชัน Baseline - เอกสาร Change Request ที่ได้รับการลงนาม ยืนยัน จากลูกค้าแล้ว (ถ้ามี)	
*ตัวอย่างเอกสาร Software Design ดังภาพที่ 4.35-4.37		
*ตัวอย่างเอกสาร Traceability Record ดังภาพที่ 4.40-4.41		

ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p><b>8. จัดทำเอกสารสำหรับการดำเนินการ User Acceptance Test ของลูกค้า โดยนำเอกสาร Test Cases and Test Procedures และ Test Report เวอร์ชันที่ผ่านการ Validate แล้ว (ก่อนการบันทึกผลการทดสอบโดย Tester ของทีมงานดำเนินโครงการ) มาจัดทำเป็นสำเนาเอกสารสำหรับการ User Acceptance Test และลงนามยืนยันผลการทดสอบระบบของลูกค้า</b></p>	<p><b><u>Input Products</u></b></p> <p>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ผ่านการทดสอบโดย Tester และผ่านการแก้ไขโดย Programmer เรียบร้อยแล้ว</p> <p>- เอกสาร Test Report เวอร์ชันที่ผ่านการทดสอบโดย Tester และผ่านการแก้ไขโดย Programmer เรียบร้อยแล้ว</p> <p><b><u>Output Products</u></b></p> <p>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ได้รับการเปลี่ยนแปลงเป็นเอกสารสำหรับการ</p>	<p>- <b>Team Leader/System Analyst</b> เป็นผู้ดำเนินการ</p>

## ตารางที่ 4.7 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>*ตัวอย่างเอกสาร Test Cases and Test Procedures ดังภาพที่ 4.42-4.43</p> <p>*ตัวอย่างเอกสาร Test Report ดังภาพที่ 4.44-4.45</p>	<p>- User Acceptance Test และได้รับการลงนามยืนยันผลการทดสอบระบบของลูกค้าแล้ว</p> <p>เอกสาร Test Report เวอร์ชันที่ได้รับการเปลี่ยนแปลงเป็นเอกสารสำหรับการ User Acceptance Test และได้รับการลงนามยืนยันผลการทดสอบระบบของลูกค้าแล้ว</p>	

รหัสเอกสาร

DD-TMS-BPP-54-02A-290854

## เอกสารการออกแบบระบบ

ชื่อโครงการ

ชื่อโครงการ, ชื่อโปรแกรม/ระบบ

สำหรับ

ชื่อลูกค้า

ชื่อลูกค้า

ภาพที่ 4.35 ตัวอย่างเอกสาร Software Design ส่วนที่ 1



ตารางสำหรับการลงนาม

	Full Name	Role	Date	Signature
Created by:	คุณ_____	ProjectManager	29 สิงหาคม 2554	
Reviewed by:	คุณ_____	Team Leader/System Analyst	29 สิงหาคม 2554	
Approved by:	คุณ_____	Customer	29 สิงหาคม 2554	

ส่วนบันทึกประวัติการแก้ไขและควบคุมเวอร์ชันของเอกสาร

Revision History:

Version	Date	Author	Reviewer	Description
1.0	23 สิงหาคม 2554	คุณ_____	คุณ_____	Initial
2.0	29 สิงหาคม 2554	คุณ_____	คุณ_____	Baseline

ภาพที่ 4.36 ตัวอย่างเอกสาร Software Design ส่วนที่ 2

รายการเอกสารย่อยในชุดเอกสาร ออกแบบระบบ

### สารบัญรายการเอกสารออกแบบระบบ

หัวข้อ	หน้า
System Architecture (โครงสร้างสถาปัตยกรรมระบบ).....	A
Process Flow Diagram.....	B
Software Requirement Specification.....	C
Program Structure.....	D
Database (ER) Diagram.....	E
Data Dictionary.....	F

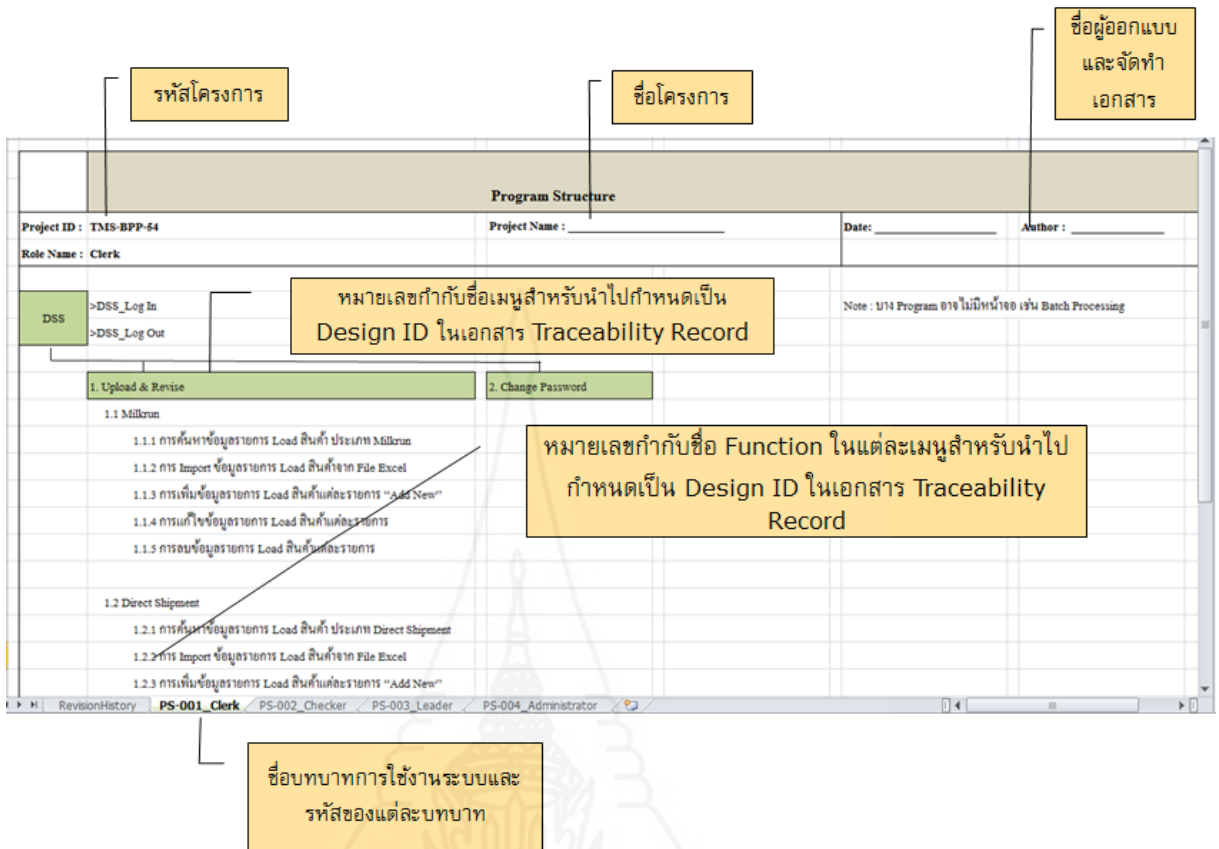
ภาพที่ 4.37 ตัวอย่างเอกสาร Software Design ส่วนที่ 3

Program Structure						
						รหัสเอกสาร
ตารางสำหรับการ ลงนาม						PG-TMS-BPP-54-01A-310754
		<b>Role</b>	<b>ชื่อ - นามสกุล</b>	<b>ลายเซ็น</b>		
<b>Created by:</b>		Team Leader/System Analyst	คุณ _____			
<b>Reviewed by:</b>		Team Leader/System Analyst	คุณ _____			
<b>Approved by:</b>		Team Leader/System Analyst	คุณ _____			
<b>ประวัติการเปลี่ยนแปลงเอกสาร</b> <b>(REVISION HISTORY)</b>						
ลำดับ	วันเดือนปี	เวอร์ชัน	อ้างอิงเอกสาร	เพิ่ม /	รายละเอียดการแก้ไข	ผู้รับผิดชอบ
1	13 ก.ค. 2554	1	-	-	Initial	คุณ _____

ส่วนบันทึกประวัติ  
การแก้ไขและ  
ควบคุมเวอร์ชันของ  
เอกสาร

ภาพที่ 4.38 ตัวอย่างเอกสารย่อย Program Structure ในชุดเอกสาร Software Design ส่วนที่ 1





ภาพที่ 4.39 ตัวอย่างเอกสารย่อย Program Structure ในชุดเอกสาร Software Design ส่วนที่ 2

TRACEABILITY RECORD						
ตารางสำหรับการ ลงนาม		รหัสเอกสาร TA-DSS-AGC-57-03A-150957				
	บทบาทหน้าที่รับผิดชอบ	ชื่อ - นามสกุล	ลายเซ็น			
ผู้จัดทำ :	Team Leader/System Analyst	คุณ _____				
ผู้ตรวจทาน :	Team Leader/System Analyst	คุณ _____				
ผู้อนุมัติ :	Team Leader/System Analyst	คุณ _____				
<b>ประวัติการเปลี่ยนแปลงเอกสาร</b> <b>(REVISION HISTORY)</b>						
ลำดับ	วันเดือนปี	เวอร์ชัน	อ้างอิงเอกสาร	เพิ่ม /	รายละเอียดการแก้ไข	ผู้รับผิดชอบ
1	21 พ.ค. 2557	1	-	-	Initial	คุณ _____
2	22 มิ.ย. 2557	2	Test Documents	-	เพิ่มรหัสการ Test	คุณ _____
3	15 ก.ย. 2557	3	Test Documents	-	เพิ่มรหัสการ Test	คุณ _____

ภาพที่ 4.40 ตัวอย่างเอกสาร Traceability Record ส่วนที่ 1

ชื่อโครงการ: การพัฒนาโปรแกรมระบบ Delivery Status System		รหัสโครงการ: DSS-AGC-57		วันที่จัดทำเอกสาร:				
ผู้ดูแลเอกสาร: คุณ _____		ผู้ปรับปรุงเอกสาร: คุณ _____		วันที่ปรับปรุงเอกสาร:				
การกระจายเอกสาร: PM ทุกรายงาน		Total Number of Requirements: 38						
Reqmts_ID (SRS)	Quotation/Proposal Requirements	Requirements Descriptions (SRS)	Priority	Design_ID	Design Descriptions	Test_ID	TestCase_ID	Remarks
RQ-4-4.1	1-1.1, 1-1.2	คุณสมบัติของโปรแกรมระบบสำหรับ Role Clerk/	Normal	PS-001-1.1.1	การตั้งค่าข้อมูลรายการ Load สินค้า ประเภท Mikrun โดย Role Clerk	UT-004	TC-001	NA
RQ-4-4.2	1-1.1, 1-1.2	Clerk/Upload & Revise: Mikrun/การ Import ข้อมูล	Normal	PS-001-1.1.2	การ Import ข้อมูลรายการ Load สินค้าจาก File Excel โดย Role Clerk	UT-004	TC-002	NA
RQ-4-4.3	1-1.1, 1-1.2	Clerk/Upload & Revise: Mikrun/การเพิ่มข้อมูล รายการ Load สินค้าต่อรายการ "Add New"	Normal	PS-001-1.1.3	"Add New" โดย Role Clerk	UT-004	TC-003	NA
RQ-4-4.4	1-1.2	คุณสมบัติของโปรแกรมระบบสำหรับ Role Clerk/Upload & Revise: Mikrun/การแก้ไขข้อมูล รายการ Load สินค้าต่อรายการ	Normal	PS-001-1.1.4	การแก้ไขรายการ			หัวข้อซอฟต์แวร์โมดูลย่อย (Software Components) ตามรายละเอียดที่กำหนดไว้ในเอกสารย่อย Program Structure ในชุดเอกสาร Software Design
	1-1.2	คุณสมบัติของโปรแกรมระบบสำหรับ Role Clerk/Upload & Revise: Mikrun/การลบข้อมูล	Normal	PS-001-1.1.5	การลบ			

ภาพที่ 4.41 ตัวอย่างเอกสาร Traceability Record ส่วนที่ 2



TEST RECORD						
ตารางสำหรับการ ลงนาม			รหัสเอกสาร UST-TMS-BPP-54-01A-270954			
		บทบาทหน้าที่รับผิดชอบ	ชื่อ - นามสกุล		ลายเซ็น	
ผู้จัดทำ :		Tester	คุณ _____			
ผู้ตรวจทาน :		Team Leader/System Analyst, Tester	คุณ _____			
ผู้อนุมัติ :		Customer	คุณ _____		ส่วนบันทึกประวัติ การแก้ไขและ ควบคุมเวอร์ชันของ เอกสาร	
<div style="font-size: 2em; font-weight: bold; opacity: 0.5;">Page 1</div> ประวัติการเปลี่ยนแปลงเอกสาร (REVISION HISTORY)						
ลำดับ	วันเดือนปี	เวอร์ชัน	อ้างอิงเอกสาร	เพิ่ม /	รายละเอียดการแก้ไข	ผู้รับผิดชอบ
1	27 ก.ย. 2554	1	-	-	Initial	คุณ _____

ภาพที่ 4.42 ตัวอย่างเอกสาร Test Cases and Test Procedures ส่วนที่ 1

ชื่อโครงการ, ชื่อโปรแกรม/ระบบ, ชื่อลูกค้า		รายชื่อทีมงานดำเนินโครงการ	
บันทึกการทดสอบระบบ (Test Record) TEST_ID: UT-002			
ชื่อโครงการ: การพัฒนา และ IMPLEMENT โปรแกรม TOOLS MANAGEMENT	รหัสโครงการ: TMS-BPP-54	ผู้ทำการทดสอบ: คุณ _____	วันที่ทดสอบ: 27 ก.ย. 2554
ชื่อผู้ทำ: _____	ผู้จัดการโครงการ: คุณ _____	นักวิเคราะห์ระบบ: คุณ _____	โปรแกรมเมอร์: คุณ _____
การกระจายเอกสาร: ส่งให้ SA และ PM เท่านั้น			
Test_ID: UT-002	Test Name: การตรวจสอบบทบาท (Role) การเข้าถึงงาน และสิทธิ์การใช้งาน	Version: 1.0	Number of Test Case: 5
Test Environment: เซิร์ฟเวอร์ (Server) ของ PSP		Total Test Time: 25 นาที	Total Defects Found: -
TestCase_ID: TC-001	Objective: การตรวจสอบโครงสร้างเมนูรายการสินค้าเครื่องใช้	Reqs_ID: RO-2-SC-002	Test Time: 5 นาที
Test Procedure		Data Input	Expected Result
1	Login Role เจ้าหน้าที่อาวุโส	Login Name: store Password: password	แสดงรายการเมนูของกรม SRB-03A
2	Login Role เจ้าหน้าที่ผู้รับ	Login Name: bomover Password: password	แสดงรายการเมนูของกรม SRB-03A
Actual Result		Defect Details	
Pass		Fail	
สถานะผลการทดสอบ			
Test_ID: TC-002	Objective: การตรวจสอบโครงสร้างเมนูจัดการสินค้าเครื่องใช้	Reqs_ID: RO-2-SC-002	Test Time: 5 นาที
Test Procedure		Data Input	Expected Result
	Login Role เจ้าหน้าที่อาวุโส	Login Name: store	แสดงรายการเมนูของกรม SRB3A
Actual Result		Defect Details	
Pass		Fail	
RevisionHistory			
คำแนบมา	UST001	UST002	UST003
	UST004	UST005	UST006
	UST007	UST008	UST009
	UST010	UST011	UST012
	UST013	UST014	UST015
ขั้นตอนการทดสอบ (Test Procedure), Data Input, และ Expected Result			หัวข้อ Requirement ในเอกสาร SRS ที่ทำการทดสอบ

ภาพที่ 4.43 ตัวอย่างเอกสาร Test Cases and Test Procedures ส่วนที่ 2

TEST REPORT						
ตารางสำหรับการลงนาม			รหัสเอกสาร			
			TR-TMS-BPP-54-01A-270954			
บทบาทหน้าที่รับผิดชอบ		ชื่อ - นามสกุล	ลายเซ็น			
ผู้จัดทำ :	Tester	คุณ _____				
ผู้ตรวจทาน :	Team Leader/SystemAnalyst , Tester	คุณ _____				
ผู้อนุมัติ :	Customer	คุณ _____	ส่วนบันทึกประวัติการแก้ไขและควบคุมเวอร์ชันของเอกสาร			
Page 1						
ประวัติการเปลี่ยนแปลงเอกสาร						
(REVISION HISTORY)						
ลำดับ	วันเดือนปี	เวอร์ชัน	อ้างอิงเอกสาร	เพิ่ม /	รายละเอียดการแก้ไข	ผู้รับผิดชอบ
1	27 ก.ย. 2554	1	-	-	Initial	คุณ _____

ภาพที่ 4.44 ตัวอย่างเอกสาร Test Report ส่วนที่ 1

ชื่อโครงการ, ชื่อโปรแกรม/ระบบ, ชื่อลูกค้า		รายชื่อทีมงานดำเนินโครงการ			
รายงานผลการทดสอบระบบ (TEST REPORT)					
ชื่อโครงการ:	การพัฒนา และ IMPLEMENT โปรแกรม TOOLS MANAGEMENT	รหัสโครงการ:	TMS-BPP-54	ผู้ทำการทดสอบ:	คุณ _____ วันที่ทดสอบ: 27 ก.ย. 2554
ชื่อลูกค้า:		ผู้จัดการโครงการ:	คุณ _____	นักวิเคราะห์ระบบ:	คุณ _____ โปรแกรมเมอร์: คุณ _____
อ้างอิงเอกสาร:	UST-TMS-BPP-54-01A-270954				
Test_ID	Test Case_ID	Test Name	ผลการทดสอบ		สถานะผลการทดสอบ
			ผ่าน	ไม่ผ่าน	
UT-001	TC-001	การตรวจสอบโครงสร้างเมนูรายการยื่นเรื่องเมื่อ	☑		
UT-001	TC-002	การตรวจสอบโครงสร้างเมนูจัดการคลังเรื่องเมื่อ	☑		
UT-001	TC-003	การตรวจสอบโครงสร้างเมนูทะเบียนเรื่องเมื่อ	☑		
UT-001	TC-004	การตรวจสอบโครงสร้างเมนูรายงาน	☑		
UT-001	TC-005	การตรวจสอบโครงสร้างเมนู Admin	☑		
UT-001	TC-001	การตรวจสอบโครงสร้างเมนูรายการยื่นเรื่องเมื่อ	☑		
UT-002	TC-002	การตรวจสอบโครงสร้างเมนูจัดการคลังเรื่องเมื่อ	☑		
UT-002	TC-003	การตรวจสอบโครงสร้างเมนูทะเบียนเรื่องเมื่อ	☑		
UT-002	TC-004	การตรวจสอบโครงสร้างเมนูรายงาน	☑		
UT-002	TC-005	การตรวจสอบโครงสร้างเมนู Admin	☑		
UT-003	TC-001	การตรวจสอบโครงสร้างเมนูรายการยื่นเรื่องเมื่อ	☑		
UT-003	TC-002	การตรวจสอบโครงสร้างเมนูจัดการคลังเรื่องเมื่อ	☑		

ภาพที่ 4.45 ตัวอย่างเอกสาร Test Report ส่วนที่ 2

2.2.4 SI.4: Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์)

ดังนี้

1) ผู้รับผิดชอบในโครงการสำหรับ Activity นี้และบทบาทความเกี่ยวข้องมี

(1) Team Leader/System Analyst มอบหมายงานพัฒนาซอฟต์แวร์ให้ Programmer ตามรายละเอียดในเอกสาร Project Plan เวอร์ชันล่าสุด

(2) Programmer ทบทวนรายละเอียดและทำความเข้าใจเอกสาร Software Design เวอร์ชัน Baseline

(3) Programmer ดำเนินการพัฒนาซอฟต์แวร์แยกเป็น Software Components (ซอฟต์แวร์ Module ย่อย) โดยดำเนินการอ้างอิงตามเอกสาร Software Design เวอร์ชัน Baseline และเอกสาร Traceability Record เวอร์ชัน Baseline

(4) Programmer ดำเนินการ Unit Test เพื่อทดสอบหน่วยย่อยสำหรับ Software Components (ซอฟต์แวร์ Module ย่อย) ส่วนที่ตนเองได้รับมอบหมายให้เป็นผู้พัฒนาและ

ทบทวนอ้างอิงตามเอกสาร Software Design เวอร์ชัน Baseline และเอกสาร Traceability Record เวอร์ชัน Baseline ว่าได้ดำเนินการงานที่ได้รับมอบหมายอย่างครบถ้วนแล้ว

(5) Programmer บันทึกผลการพัฒนาซอฟต์แวร์และผลการ Unit Test กำกับแต่ละ Software Components (ซอฟต์แวร์ Module ย่อย) ส่วนที่ตนเองได้รับมอบหมายในเอกสาร Traceability Record เวอร์ชันล่าสุด

(6) Team Leader/System Analyst ตรวจสอบและควบคุมผลการดำเนินงานของ Programmer ทั้งในส่วนของ Software Components (ซอฟต์แวร์ Module ย่อย) และการบันทึกผลการพัฒนาซอฟต์แวร์และผลการ Unit Test ในเอกสาร Traceability Record ให้ถูกต้องและครบถ้วน

## 2) เอกสารที่เกี่ยวข้องกับ Activity นี้มี ดังนี้

### (1) Input Products

ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15) เวอร์ชันล่าสุด

ข. เอกสาร Software Design (ตัวอย่างดังภาพที่ 4.35-4.37) เวอร์ชัน

Baseline

ค. เอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41) เวอร์ชัน

ล่าสุด

ง. Software Components (ซอฟต์แวร์ Module ย่อย) ที่ได้รับการ

พัฒนาขึ้น

### (2) Output Products

ก. Software Components (ซอฟต์แวร์ Module ย่อย) ที่ได้รับการพัฒนา, ได้รับการ Unit Test และแก้ไขเรียบร้อยแล้ว

ข. เอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41) เวอร์ชันล่าสุดที่ได้รับการบันทึกผลการพัฒนาซอฟต์แวร์และผลการ Unit Test ในเอกสาร Traceability Record ให้ถูกต้องและครบถ้วนแล้ว

**หมายเหตุ:** สำหรับรายการ Software Components (ซอฟต์แวร์ Module ย่อย) ที่ได้รับการพัฒนา, ได้รับการ การ Unit Test และแก้ไขเรียบร้อยแล้ว โดยไม่มีประเด็นปัญหาค้างจะได้รับการบันทึกสถานะ “N/A” ใน column “Remarks” ของเอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41)



### 3) รายละเอียดเพิ่มเติมมี ดังนี้

(1) Team Leader/System Analyst ควรตรวจสอบและควบคุมให้ Programmer แก้ไข Software Components (ซอฟต์แวร์ Module ย่อย) ให้ทำงานอย่างถูกต้องตามที่บันทึกผลการดำเนินงานไว้ในเอกสาร Traceability Record ก่อนดำเนินการใน Activity SI.5 Software Integration and Tests (การทดสอบและการบูรณาการซอฟต์แวร์)

**บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.4: Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์)**

SI.4: Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์) เป็นการดำเนินงานโดยบทบาท **Programmer** และ **Team Leader/System Analyst** โดย Programmer จะดำเนินงานพัฒนาซอฟต์แวร์ Module ย่อย (Software Components) ตามที่ได้รับมอบหมายงานจาก **Team Leader/System Analyst** โดยเป็นการดำเนินงานตามขอบเขตของงานในเอกสาร Project Plan และตามโครงสร้างการออกแบบระบบในเอกสาร Software Design เมื่อ **Programmer** พัฒนาซอฟต์แวร์ Module ย่อย (Software Components) แล้วจะต้องดำเนินการ Unit Test และแก้ไขให้ซอฟต์แวร์ทำงานถูกต้อง หลังจากจึงทำการบันทึกผลการพัฒนาซอฟต์แวร์และผลการ Unit Test กำกับแต่ละ Software Components (ซอฟต์แวร์ Module ย่อย) ส่วนที่ตนเองได้รับมอบหมายในเอกสาร Traceability Record

**Team Leader/System Analyst** เป็นผู้หนึ่งที่มีบทบาทสำคัญมากในการตรวจสอบและควบคุมการดำเนินงานของ Programmer เพื่อให้ Programmer แก้ไข Software Components (ซอฟต์แวร์ Module ย่อย) ให้ทำงานอย่างถูกต้องตามที่บันทึกผลการดำเนินงานไว้ในเอกสาร Traceability Record อย่างครบถ้วน ก่อนเริ่มดำเนินงานใน Activity SI.5 Software Integration and Tests (การทดสอบและการบูรณาการซอฟต์แวร์)

ตารางที่ 4.8 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

สำหรับ Activity SI.4: Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p><b>1. มอบหมายงานพัฒนาซอฟต์แวร์ให้ Programmer ตามรายละเอียดในเอกสาร Project Plan เวอร์ชันล่าสุด</b></p>	<p><b><u>Input Products</u></b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan เวอร์ชันล่าสุด</li> </ul> <p><b><u>Output Products</u></b></p>	<ul style="list-style-type: none"> <li>- <b>Team Leader/System Analyst</b> เป็นผู้ดำเนินการ</li> </ul>
*ตัวอย่างเอกสาร Project Plan ดังภาพที่ 4.3-4.15		
<p><b>2. ทบทวนรายละเอียดและทำความเข้าใจเอกสาร Software Design เวอร์ชัน Baseline</b></p>	<p><b><u>Input Products</u></b></p> <ul style="list-style-type: none"> <li>- เอกสาร Software Design เวอร์ชัน Baseline</li> </ul> <p><b><u>Output Products</u></b></p>	<ul style="list-style-type: none"> <li>- <b>Programmer</b> เป็นผู้ดำเนินการ</li> </ul>
*ตัวอย่างเอกสาร Software Design ดังภาพที่ 4.35-4.37		
<p><b>3. พัฒนาซอฟต์แวร์แยกเป็น Software Components (ซอฟต์แวร์ Module ย่อย) โดยดำเนินการอ้างอิงตามเอกสาร Software Design เวอร์ชัน Baseline และเอกสาร Traceability Record เวอร์ชัน Baseline</b></p>	<p><b><u>Input Products</u></b></p> <ul style="list-style-type: none"> <li>- เอกสาร Software Design เวอร์ชัน Baseline</li> <li>- เอกสาร Traceability Record เวอร์ชัน Baseline</li> </ul> <p><b><u>Output Products</u></b></p> <ul style="list-style-type: none"> <li>- Software Components (ซอฟต์แวร์ Module ย่อย)</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Programmer</b> เป็นผู้ดำเนินการ</li> </ul>
*ตัวอย่างเอกสาร Software Design ดังภาพที่ 4.35-4.37		
*ตัวอย่างเอกสาร Traceability Record ดังภาพที่ 4.40-4.41		

ตารางที่ 4.8 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p><b>4. ดำเนินการ Unit Test เพื่อทดสอบหน่วยย่อย สำหรับ Software Components (ซอฟต์แวร์ Module ย่อย) และบันทึกผลการทดสอบกับ รายการ Software Components (ซอฟต์แวร์ Module ย่อย) ในเอกสาร Traceability Record เวอร์ชันล่าสุด</b> ส่วนที่ตนเองได้รับมอบหมายให้ เป็นผู้พัฒนาและทบทวนการดำเนินงานอ้างอิง ตามเอกสาร Software Design เวอร์ชัน Baseline และเอกสาร Traceability Record เวอร์ชันล่าสุด ว่าได้ดำเนินการงานที่ได้รับมอบหมายอย่าง ครบถ้วนแล้ว</p> <p><b>หมายเหตุ:</b> สำหรับรายการ Software Components (ซอฟต์แวร์ Module ย่อย) ที่ ได้รับการพัฒนา, ได้รับการ การ Unit Test และแก้ไขเรียบร้อยแล้ว โดยไม่มีประเด็น ปัญหาค้างจะได้รับการบันทึกสถานะ “N/A” ใน column “Remarks” ของเอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41)</p> <p>*ตัวอย่างเอกสาร Software Design ดังภาพที่ 4.35-4.37</p> <p>*ตัวอย่างเอกสาร Traceability Record ดังภาพที่ 4.40-4.41</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Software Design เวอร์ชัน Baseline</li> <li>- เอกสาร Traceability Record เวอร์ชัน ล่าสุด</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- Software Components (ซอฟต์แวร์ Module ย่อย) ที่ ผ่านการ Unit Test แล้ว</li> <li>- เอกสาร Traceability Record เวอร์ชัน ล่าสุดที่ได้รับการ บันทึกผลการ พัฒนาและ Unit Test แล้ว</li> </ul>	<p>- <b>Programmer</b> เป็นผู้ดำเนินการ</p>

ตารางที่ 4.8 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>5. ตรวจสอบและควบคุมผลการดำเนินงานของ Programmer ทั้งในส่วนของ Software Components (ซอฟต์แวร์ Module ย่อย) และการบันทึกผลการพัฒนาซอฟต์แวร์และผลการ Unit Test ในเอกสาร Traceability Record ให้ถูกต้องและครบถ้วน</p>	<p><b><u>Input Products</u></b></p> <ul style="list-style-type: none"> <li>- Software Components (ซอฟต์แวร์ Module ย่อย) ที่ผ่านการ Unit Test แล้วโดย Programmer</li> <li>- เอกสาร Traceability Record เวอร์ชันล่าสุดที่ได้รับการบันทึกผลการพัฒนาและ Unit Test แล้วโดย Programmer</li> </ul> <p><b><u>Output Products</u></b></p> <ul style="list-style-type: none"> <li>- Software Components (ซอฟต์แวร์ Module ย่อย) ที่ได้รับการพัฒนา, ได้รับการ Unit Test และแก้ไขเรียบร้อยแล้ว</li> </ul>	<p>- <b>Team Leader/System Analyst</b> เป็นผู้ดำเนินการ</p>

ตารางที่ 4.8 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>*ตัวอย่างเอกสาร Traceability Record ดังภาพที่ 4.40-4.41</p>	<p>- เอกสาร Traceability Record เวอร์ชันล่าสุดที่ ได้รับการการ บันทึกผลการ พัฒนาซอฟต์แวร์ และผลการ Unit Test ในเอกสาร Traceability Record ให้ถูกต้องและ ครบถ้วนแล้ว</p>	

### 2.2.5 SI.5: Software Integration and Tests (การทดสอบและการบูรณาการซอฟต์แวร์)

- 1) ผู้รับผิดชอบในโครงการสำหรับ Activity นี้และบทบาทความเกี่ยวข้องมีดังนี้
- (1) Team Leader/System Analyst ทบทวนสถานะการดำเนินงานและมอบหมายงานพัฒนาซอฟต์แวร์ให้ Programmer ตามรายละเอียดในเอกสาร Project Plan เวอร์ชันล่าสุด
  - (2) Programmer ทบทวนรายละเอียดและทำความเข้าใจเอกสาร Test Cases and Test Procedures เวอร์ชันที่ผ่านการ Validate แล้ว และทำความเข้าใจสภาพแวดล้อมของระบบสำหรับการทดสอบของ Tester ที่เป็นทีมงานดำเนินโครงการ และสภาพแวดล้อมของระบบสำหรับการทดสอบในส่วน User Acceptance Test ของลูกค้า
  - (3) Team Leader/System Analyst และ Programmer ดำเนินการนำ Software Components (ซอฟต์แวร์ Module ย่อย) ทั้งหมดมาทำการบูรณาการเข้าด้วยกันเป็น Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)

(4) Tester ดำเนินการทดสอบ Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด) ตามรายละเอียดในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันที่ผ่านการ Validate แล้ว และบันทึกผลการทดสอบ

(5) Programmer จะทำการแก้ไข Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด) จนกระทั่งผลการทดสอบจาก Tester เป็น “ผ่าน” ทุกรายการ ได้รับการบันทึกในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันล่าสุด

(6) Team Leader/System Analyst นำเอกสาร Test Cases and Test Procedures และ Test Report เวอร์ชันที่ผ่านการ Validate แล้ว (ก่อนการบันทึกผลการทดสอบโดย Tester ของทีมงานดำเนินโครงการ) มาจัดทำเป็นสำเนาเอกสารสำหรับการ User Acceptance Test และนำไปให้ลูกค้าใช้ประกอบการบันทึกผลการทดสอบระบบ

(7) Programmer จะทำการแก้ไข Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด) จนกระทั่งผลการทดสอบ User Acceptance Test จากลูกค้าเป็น “ผ่าน” ทุกรายการ ได้รับการบันทึกในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันล่าสุดแล้ว

(8) Team Leader/System Analyst นำเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันล่าสุดที่มีผลการทดสอบ User Acceptance Test เป็น “ผ่าน” ทุกรายการแล้วนำเสนอให้ลูกค้าลงนามยืนยันผลการทดสอบระบบ User Acceptance Test

(9) Team Leader/System Analyst จัดทำเอกสาร System Manual (มีเนื้อหาครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation)

(10) Team Leader/System Analyst และ Programmer ทบทวนรายละเอียดของเอกสาร System Manual (มีเนื้อหาครอบคลุม Product Operation Guide และ Maintenance Documentation) ร่วมกัน แล้วบันทึกการ Verify ในเอกสาร Verification & Validation Results

(11) Administration Staff จัดทำเอกสาร User Manual (Software User Documentation) ตามคุณลักษณะของ Software ที่ผ่านการทดสอบเรียบร้อยแล้ว

(1) Team Leader/System Analyst นำเอกสาร User Manual (Software User Documentation) ให้ลูกค้าทบทวนเนื้อหา แล้วบันทึกการ Verify ในเอกสาร Verification & Validation Results

(12) Team Leader/System Analyst จัดเตรียมข้อมูลและเอกสารเป็นชุด Software Configuration เวอร์ชันฉบับร่างสำหรับเตรียมการส่งมอบปิดโครงการเสนอให้กับลูกค้า โดยข้อมูลและเอกสารในชุด Software Configuration ประกอบด้วย

ก. เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34)

ข. เอกสาร Software Design (ตัวอย่างดังภาพที่ 4.35-4.37)

ค. เอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41)

ง. Software ที่พัฒนาเสร็จแล้ว (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วน ตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)

จ. เอกสาร Test Cases and Test Procedures (ตัวอย่างดังภาพที่ 4.42-4.43)

ฉ. เอกสาร Test Report (ตัวอย่างดังภาพที่ 4.44-4.45)

ช. เอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) (ตัวอย่างดังภาพที่ 4.46-4.51)

ซ. เอกสาร User Manual (Software User Documentation) (ตัวอย่าง ดังภาพที่ 4.52-4.55)

2) เอกสารที่เกี่ยวข้องกับ Activity นี้มี ดังนี้

(1) Input Products

ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15) เวอร์ชันล่าสุด

ข. Software Components (ซอฟต์แวร์ Module ย่อย) ที่ได้รับการ พัฒนา, ได้รับการ การ Unit Test และแก้ไขเรียบร้อยแล้ว

ค. เอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41) เวอร์ชัน ล่าสุดที่ได้รับการบันทึกผลการพัฒนาซอฟต์แวร์และผลการ Unit Test ในเอกสาร Traceability Record ให้ถูกต้องและครบถ้วนแล้ว

ง. เอกสาร Test Cases and Test Procedures (ตัวอย่างดังภาพที่ 4.42-4.43) เวอร์ชันที่ผ่านการ Validate แล้ว

จ. เอกสาร Test Report (ตัวอย่างดังภาพที่ 4.44-4.45) เวอร์ชันที่ผ่านการ Validate แล้ว

ฉ. เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34) เวอร์ชันล่าสุด



ข. เอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) (ตัวอย่างดังภาพที่ 4.46-4.51) ฉบับร่าง

ค. เอกสาร User Manual (Software User Documentation) (ตัวอย่างดังภาพที่ 4.52-4.55) ฉบับร่าง

## (2) Output Products

ก. Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)

ข. เอกสาร Test Cases and Test Procedures (ตัวอย่างดังภาพที่ 4.42-4.43) เวอร์ชันที่ได้รับการบันทึกผลการทดสอบเป็น “ผ่าน” ทุกรายการจาก Tester แล้ว

ค. เอกสาร Test Report (ตัวอย่างดังภาพที่ 4.44-4.45) เวอร์ชันที่ได้รับการบันทึกผลการทดสอบเป็น “ผ่าน” ทุกรายการจาก Tester แล้ว

ง. เอกสาร Test Cases and Test Procedures (ตัวอย่างดังภาพที่ 4.42-4.43) เวอร์ชันที่ได้รับการบันทึกผลการทดสอบ User Acceptance Test เป็น “ผ่าน” ทุกรายการจากลูกค้าแล้ว

จ. เอกสาร Test Report (ตัวอย่างดังภาพที่ 4.44-4.45) เวอร์ชันที่ได้รับการบันทึกผลการทดสอบ User Acceptance Test เป็น “ผ่าน” ทุกรายการจากลูกค้าแล้ว

ฉ. เอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) (ตัวอย่างดังภาพที่ 4.46-4.51) เวอร์ชันที่ผ่านการ Verify แล้ว

ช. เอกสาร User Manual (Software User Documentation) (ตัวอย่างดังภาพที่ 4.52-4.55) เวอร์ชันที่ผ่านการ Verify แล้ว

ซ. ข้อมูลและเอกสารเป็นชุด Software Configuration เวอร์ชันฉบับร่างประกอบด้วย

ฌ. เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34)

ฎ. เอกสาร Software Design (ตัวอย่างดังภาพที่ 4.35-4.37)

ฏ. เอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41)

ฐ. Software ที่พัฒนาเสร็จแล้ว (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)

ฐ. เอกสาร Test Cases and Test Procedures (ตัวอย่างดังภาพที่ 4.42-4.43)

ท. เอกสาร Test Report (ตัวอย่างดังภาพที่ 4.44-4.45)

ฅ. เอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) (ตัวอย่างดังภาพที่ 4.46-4.51)

ณ . เอกสาร User Manual (Software User Documentation) (ตัวอย่างดังภาพที่ 4.52-4.55)

### 3) รายละเอียดเพิ่มเติมมี ดังนี้

(1) Team Leader/System Analyst ควรตรวจสอบข้อมูลและเอกสาร ในชุด Software Configuration ให้ครบถ้วนเพื่อจะสามารถเสนอการปิดโครงการกับลูกค้าได้อย่าง รวดเร็วยิ่งขึ้น

(2) Team Leader/System Analyst ควรนำเอกสาร Test Cases and Test Procedures และเอกสาร Test Report ให้ลูกค้าลงนามยืนยันผลการทดสอบ

**บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.5 Software Integration and Tests (การทดสอบและการบูรณาการซอฟต์แวร์)**

SI.5 Software Integration and Tests (การทดสอบและการบูรณาการซอฟต์แวร์) เป็นการดำเนินงานโดยบทบาท **Team Leader/System Analyst, Programmer, และ Tester** โดย **Team Leader/System Analyst และ Programmer** จะทำความเข้าใจในเอกสาร Test Cases and Test Procedures และสภาพแวดล้อมของระบบสำหรับการทดสอบของ Tester ที่เป็นทีมงานดำเนินโครงการและสภาพแวดล้อมของระบบสำหรับการทดสอบในส่วน User Acceptance Test ของลูกค้าเพื่อจะสามารถเตรียมการในการ support การแก้ไขปัญหาการทดสอบซอฟต์แวร์ได้อย่างเหมาะสมแล้วดำเนินการนำ Software Components (ซอฟต์แวร์ Module ย่อย) ทั้งหมดมาทำการบูรณาการเข้าด้วยกันเป็น Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)

หลังจากนั้น **Tester** จึงจะดำเนินการทดสอบระบบอ้างอิงตามเอกสาร Test Cases and Test Procedures และบันทึกผลการทดสอบไว้ในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report กรณีพบปัญหาของซอฟต์แวร์ **Programmer** จะให้การ support ในการแก้ไขปัญหาการทดสอบซอฟต์แวร์จนกระทั่งผลการทดสอบจาก **Tester** เป็น “ผ่าน” ทุกรายการตามที่บันทึกไว้ในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันล่าสุด

เมื่อ Software ผ่านการทดสอบโดย **Tester** ซึ่งเป็นทีมงานดำเนินโครงการ แล้ว **Team Leader/System Analyst** นำเอกสาร Test Cases and Test Procedures และ Test Report เวอร์ชันที่ผ่านการ Validate แล้ว (ก่อนการบันทึกผลการทดสอบโดย Tester ของทีมงานดำเนินโครงการ) มาจัดทำเป็นสำเนาเอกสารสำหรับการ User Acceptance Test และนำไปให้ลูกค้าใช้ประกอบการบันทึกผลการทดสอบระบบ โดย **Programmer** จะให้การ support ในการแก้ไขปัญหาการทดสอบซอฟต์แวร์จนกระทั่งผลการทดสอบ User Acceptance Test จากลูกค้าเป็น “ผ่าน” ทุกรายการตามที่บันทึกไว้ในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันล่าสุดแล้ว ขั้นตอนที่สำคัญมากอีกขั้นตอนหนึ่งคือการทำ **Team Leader/System Analyst** จะต้องนำเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันล่าสุดที่ผ่านการบันทึกผล User Acceptance Test จากลูกค้าเป็น “ผ่าน” ทุกรายการแล้วให้ลูกค้าลงนามยืนยันผลการทดสอบ เพื่อจะสามารถใช้เป็นหลักฐานยืนยันผลการดำเนินโครงการในขั้นตอนการพัฒนาซอฟต์แวร์ว่าทีมงานได้ดำเนินงานครบถ้วนตามขอบข่ายงานแล้ว

เมื่อ Software ผ่านการทดสอบทั้งโดย **Tester** ซึ่งเป็นทีมงานดำเนินโครงการ และผ่านทดสอบ User Acceptance Test จากลูกค้าแล้ว **Team Leader/System Analyst** จะจัดทำเอกสาร **System Manual** และ **Tester** จัดทำเอกสาร User Manual เป็นลำดับถัดไป

การดำเนินการขั้นตอนสุดท้ายของกิจกรรมนี้ คือการทำ **Team Leader/System Analyst** เป็นผู้รวบรวมข้อมูลและเอกสารทั้งหมดให้เป็นชุด Software Configuration เวอร์ชัน Baseline เพื่อเตรียมจะขอเสนอส่งมอบปิดโครงการให้ลูกค้าพิจารณาเป็นลำดับถัดไป ซึ่ง **Team Leader/System Analyst** ก็ยังคงมีบทบาทสำคัญในการดำเนินงานในขั้นตอนนี้เพื่อที่จะช่วยให้ทีมงานสามารถปิดโครงการได้อย่างรวดเร็วหาก **Team Leader/System Analyst** สามารถดำเนินงานได้อย่างละเอียดและครบถ้วนตามขอบเขตงานที่ได้ระบุไว้ในเอกสาร Project Plan

ตารางที่ 4.9 แนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity  
SI.5 Software Integration and Tests (การทดสอบและการบูรณาการซอฟต์แวร์)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p><b>1. ทบทวนสถานะการดำเนินงานและมอบหมายงานพัฒนาซอฟต์แวร์ให้ Programmer ตามรายละเอียดในเอกสาร Project Plan เวอร์ชันล่าสุดให้ครบถ้วน</b></p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Project Plan เวอร์ชันล่าสุด</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>-</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Team Leader/System Analyst</b> เป็นผู้ดำเนินการ</li> </ul>
*ตัวอย่างเอกสาร Project Plan ดังภาพที่ 4.3-4.15		
<p><b>2. ทบทวนรายละเอียดและทำความเข้าใจเอกสาร Test Cases and Test Procedures เวอร์ชันที่ผ่านการ Validate แล้ว และทำความเข้าใจสภาพแวดล้อมของระบบสำหรับการทดสอบของ Tester ที่เป็นทีมงานดำเนินโครงการ และสภาพแวดล้อมของระบบสำหรับการทดสอบในส่วน User Acceptance Test ของลูกค้า</b></p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ผ่านการ Validate แล้ว</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>-</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Team Leader/System Analyst และ Programmer</b> เป็นผู้ดำเนินการ</li> </ul>
*ตัวอย่างเอกสาร Test Cases and Test Procedures ดังภาพที่ 4.42-4.43		
<p><b>3. นำ Software Components (ซอฟต์แวร์ Module ย่อย) ทั้งหมดมาทำการบูรณาการเข้าด้วยกันเป็น Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)</b></p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- Software Components (ซอฟต์แวร์ Module ย่อย) ที่ได้รับการพัฒนา, ได้รับการ การ Unit Test และ</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Team Leader/System Analyst และ Programmer</b> เป็นผู้ดำเนินการ</li> </ul>

## ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาท ผู้รับผิดชอบใน โครงการ
<p data-bbox="288 1406 853 1491">*ตัวอย่างเอกสาร Traceability Record ดังภาพที่ 4.40-4.41</p>	<p data-bbox="871 591 1126 1137">แก้ไขเรียบร้อยแล้ว</p> <p data-bbox="871 656 1126 1137">- เอกสาร Traceability Record เวอร์ชันล่าสุดที่ได้รับการฉบับที่ผลการพัฒนาซอฟต์แวร์และผลการ Unit Test ในเอกสาร Traceability Record ให้ถูกต้องและครบถ้วนแล้ว</p> <p data-bbox="871 1167 1126 1200"><b>Output Products</b></p> <p data-bbox="871 1216 1126 1368">Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานใน</p> <p data-bbox="871 1397 1126 1480">เอกสาร Project Plan เวอร์ชันล่าสุด)</p>	

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบในโครงการ
<p>4. ดำเนินการทดสอบ Software (ซอฟต์แวร์) ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด) ตามรายละเอียดในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันที่ผ่านการ Validate แล้ว และบันทึกผลการทดสอบ</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ผ่านการ Validate แล้ว</li> <li>- เอกสาร Test Report เวอร์ชันที่ผ่านการ Validate แล้ว</li> <li>- Software (ซอฟต์แวร์) ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ได้รับการบันทึกผลการทดสอบทุกรายการ จาก Tester แล้ว</li> <li>- เอกสาร Test Report เวอร์ชันที่ได้รับการบันทึกผลการทดสอบทุกรายการ</li> </ul>	<p>- Tester เป็นผู้ดำเนินการ</p>

## ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทที่ได้รับผิดชอบในโครงการ
<p>*ตัวอย่างเอกสาร Test Cases and Test Procedures ดังภาพที่ 4.42-4.43</p> <p>*ตัวอย่างเอกสาร Test Reportดังภาพที่ 4.44-4.45</p>	<p>จาก Tester แล้ว</p> <p>Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)เวอร์ชันที่ได้รับการทดสอบจาก Tester แล้ว</p>	
<p><b>5. การแก้ไข Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด) จนกระทั่งผลการทดสอบจาก Tester เป็น “ผ่าน” ทูกรายการได้รับการบันทึกในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันล่าสุด</b></p>	<p><b><u>Input Products</u></b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ได้รับการบันทึกผลการทดสอบทูกรายการจาก Tester แล้ว</li> <li>- เอกสาร Test Report เวอร์ชันที่ได้รับการบันทึกผลการทดสอบทูกรายการจาก Tester แล้ว</li> <li>- Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan</li> </ul>	<p><b>- Programmer</b></p> <p>เป็นผู้ดำเนินการ</p>



ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทที่ได้รับ ผิดชอบ ในโครงการ
<p>*ตัวอย่างเอกสาร Test Cases and Test Procedures ดังภาพที่ 4.42-4.43</p> <p>*ตัวอย่างเอกสาร Test Report ดังภาพที่ 4.44-4.45</p>	<p>Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)เวอร์ชันที่มีสถานะการทดสอบเป็น “ผ่าน” ทุกรายการจาก Tester แล้ว</p>	
<p><b>6. นำเอกสาร Test Cases and Test Procedures และ Test Report เวอร์ชันที่ผ่านการ Validate แล้ว (ก่อนการบันทึกผลการทดสอบโดย Tester ของทีมงานดำเนินโครงการ) มาจัดทำเป็นสำเนาเอกสารสำหรับการ User Acceptance Test และนำไปให้ลูกค้าใช้ประกอบการบันทึกผลการทดสอบระบบ</b></p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ผ่านการ Validate แล้ว (ก่อนการบันทึกผลการทดสอบโดย Tester ของทีมงานดำเนินโครงการ)</li> <li>- เอกสาร Test Report</li> </ul>	<p><b>- Team Leader/System Analystเป็นผู้อำนวยการ</b></p>

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทที่ได้รับผิดชอบในโครงการ
<p>7. นำเอกสาร Test Cases and Test Procedures และ Test Report เวอร์ชันที่ผ่านการ Validate แล้ว (ก่อนการบันทึกผลการทดสอบโดย Tester ของทีมงานดำเนินโครงการ) มาจัดทำเป็นสำเนาเอกสารสำหรับการ User Acceptance Test และนำไปใช้ประกอบการบันทึกผลการทดสอบระบบ</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ผ่านการ Validate แล้ว (ก่อนการบันทึกผลการทดสอบโดย Tester ของทีมงานดำเนินโครงการ)</li> <li>- เอกสาร Test Report</li> </ul>	<p>- <b>Team Leader/System Analyst</b> เป็นผู้ดำเนินการ</p>
	<p>เวอร์ชันที่ผ่านการ Validate แล้ว (ก่อนการบันทึกผลการทดสอบโดย Tester ของทีมงานดำเนินโครงการ)</p> <p><b>Output Products</b></p> <p>เอกสาร Test Cases and Test Procedures เวอร์ชันสำหรับการ</p>	

## ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทที่ได้รับผิดชอบในโครงการ
<p>*ตัวอย่างเอกสาร Test Cases and Test Procedures ดังภาพที่ 4.42-4.43</p> <p>*ตัวอย่างเอกสาร Test Report ดังภาพที่ 4.44-4.45</p>	<ul style="list-style-type: none"> <li>- บันทึกผล User Acceptance Test และลงนามยืนยันผลการทดสอบของลูกค้า</li> <li>- เอกสาร Test Report เวอร์ชันสำหรับการบันทึกผล User Acceptance Test และลงนามยืนยันผลการทดสอบของลูกค้า</li> </ul>	
<p><b>8. ดำเนินการทดสอบ Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด) ตามรายละเอียดในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report สำหรับการบันทึกผล User Acceptance Test และลงนามยืนยันผลการทดสอบของลูกค้า และบันทึกผลการทดสอบ</b></p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันสำหรับการบันทึกผล User Acceptance Test และลงนามยืนยันผลการทดสอบของลูกค้า</li> <li>- เอกสาร Test Report เวอร์ชันสำหรับการบันทึกผล User Acceptance Test และลงนามยืนยันผลการทดสอบของลูกค้า</li> <li>- Software (ซอฟต์แวร์)</li> </ul>	<ul style="list-style-type: none"> <li>- ลูกค้า เป็นผู้ดำเนินการ</li> </ul>

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาท ผู้รับผิดชอบใน โครงการ
	<p>ที่มีคุณลักษณะ ครบถ้วนตามขอบข่าย งานในเอกสาร Project Plan) เวอร์ชันที่ผ่าน การทดสอบจาก Tester แล้ว</p> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ได้รับการ บันทึกผลการทดสอบ ทุกรายการจากลูกค้า แล้ว</li> <li>- เอกสาร Test Report เวอร์ชันที่ได้รับการ บันทึกผลการทดสอบ ทุกรายการจากลูกค้า แล้ว</li> <li>- Software (ซอฟต์แวร์ ที่มีคุณลักษณะ ครบถ้วนตามขอบข่าย งานในเอกสาร Project Plan) เวอร์ชันที่รับ การทดสอบจากลูกค้า แล้ว</li> </ul>	
*ตัวอย่างเอกสาร Test Cases and Test Procedures ดัง ภาพที่ 4.42-4.43		
*ตัวอย่างเอกสาร Test Report ดังภาพที่ 4.44-4.45		

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาท ผู้รับผิดชอบในโครงการ
<p>9. การแก้ไข Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด) จนกระทั่งผลการทดสอบจากลูกค้าเป็น “ผ่าน” ทุกรายการได้รับการบันทึกในเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันล่าสุดสำหรับการบันทึกผล User Acceptance Test แล้ว</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ได้รับการบันทึกผลการทดสอบทุกรายการจากลูกค้าแล้ว</li> <li>- เอกสาร Test Report เวอร์ชันที่ได้รับการบันทึกผลการทดสอบทุกรายการจากลูกค้าแล้ว</li> <li>- Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด) เวอร์ชันที่ได้รับการทดสอบจากลูกค้าแล้ว</li> </ul>	<p>- <b>Programmer</b> เป็นผู้ดำเนินการ</p>

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาท ผู้รับผิดชอบใน โครงการ
<p>*ตัวอย่างเอกสาร Test Cases and Test Procedures ดังภาพที่ 4.42-4.43</p> <p>*ตัวอย่างเอกสาร Test Report ดังภาพที่ 4.44-4.45</p>	<p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ได้รับการบันทึกผลการทดสอบเป็น “ผ่าน” ทูกรายการจากลูกค้าแล้ว</li> <li>- เอกสาร Test Report เวอร์ชันที่ได้รับการบันทึกผลการทดสอบเป็น “ผ่าน” ทูกรายการจากลูกค้าแล้ว</li> <li>- Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan) เวอร์ชันที่ “ผ่าน” การทดสอบ ทูกรายการจากลูกค้าแล้ว</li> </ul>	

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาท ผู้รับผิดชอบใน โครงการ
<p><b>10. นำเอกสาร Test Cases and Test Procedures และเอกสาร Test Report เวอร์ชันล่าสุดที่มีผลการทดสอบ User Acceptance Test เป็น “ผ่าน” ทุกรายการแล้วนำเสนอให้ลูกค้าลงนามยืนยันผลการทดสอบระบบ User Acceptance Test</b></p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ได้รับการบันทึกผลการทดสอบเป็น</li> </ul>	<ul style="list-style-type: none"> <li>- <b>Team Leader/System Analyst</b> เป็นผู้ดำเนินการ</li> </ul>
	<p>“ผ่าน” ทุกรายการ จากลูกค้าแล้ว</p> <ul style="list-style-type: none"> <li>- เอกสาร Test Report เวอร์ชันที่ได้รับการบันทึกผลการทดสอบเป็น “ผ่าน” ทุกรายการ จากลูกค้าแล้ว</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Test Cases and Test Procedures เวอร์ชันที่ได้รับการบันทึกผลการทดสอบเป็น “ผ่าน” ทุกรายการจากลูกค้า และได้รับการลงนามยืนยันจากลูกค้าแล้ว</li> </ul>	
<p>*ตัวอย่างเอกสาร Test Cases and Test Procedures ดังภาพที่ 4.42-4.43</p> <p>*ตัวอย่างเอกสาร Test Report ดังภาพที่ 4.44-4.45</p>		



ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาท ผู้รับผิดชอบ ในโครงการ
	<ul style="list-style-type: none"> <li>- เอกสาร Test Report เวอร์ชันที่ได้รับการบันทึกผลการทดสอบเป็น “ผ่าน” ทุกรายการและได้รับการลงนามยืนยันจากลูกค้าแล้ว</li> </ul>	
<p><b>11. จัดทำเอกสาร System Manual (มีเนื้อหาครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) โดยเอกสารควรประกอบด้วยข้อมูลที่สำคัญดังต่อไปนี้</b></p> <ul style="list-style-type: none"> <li>○ ชื่อโปรแกรม/ชื่อระบบ, ชื่อลูกค้า</li> <li>○ การเตรียมฮาร์ดแวร์และ Pre-requisite ซอฟต์แวร์</li> <li>○ ขั้นตอนการติดตั้งซอฟต์แวร์ที่ทีมงานดำเนินโครงการเป็นผู้พัฒนาขึ้น</li> <li>○ Development Tool ที่ใช้ในการพัฒนาซอฟต์แวร์</li> <li>○ คุณสมบัติของเครื่อง Client ที่รองรับการใช้งานซอฟต์แวร์</li> <li>○ วิธีการปรับเปลี่ยนค่า Configuration ของซอฟต์แวร์ (การบำรุงรักษาระบบ)</li> <li>○ มีการบันทึกประวัติการแก้ไข (Revision History)</li> </ul>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)</li> <li>เวอร์ชันที่มีสถานะการทดสอบเป็น “ผ่าน” ทุกรายการจากลูกค้าแล้ว</li> <li>- เอกสาร Software Design เวอร์ชันล่าสุด</li> </ul>	<ul style="list-style-type: none"> <li>- Team Leader/System Analyst เป็นผู้ดำเนินการ</li> </ul>

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาท ผู้รับผิดชอบใน โครงการ
<p>*ตัวอย่างเอกสาร System Manual ดังภาพที่ 4.46-4.51</p> <p>*ตัวอย่างเอกสาร Software Design ดังภาพที่ 4.35-4.37</p>	<p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร System Manual (มีเนื้อหาครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation)</li> <li>ฉบับร่าง</li> </ul>	
<p><b>12. ทบทวนรายละเอียดของเอกสาร System Manual (มีเนื้อหาครอบคลุม Product Operation Guide และ Maintenance Documentation) ร่วมกัน แล้วบันทึกการ Verify ในเอกสาร Verification &amp; Validation Results</b></p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร System Manual (มีเนื้อหาครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation)</li> <li>ฉบับร่าง</li> <li>- เอกสาร Verification &amp; Validation Result</li> <li>เวอร์ชันล่าสุด</li> </ul>	<p>- <b>Team Leader/System Analyst และ Programmer</b> เป็นผู้ดำเนินการ</p>

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาท ผู้รับผิดชอบ ในโครงการ
	<p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร System Manual (มีเนื้อหาครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) เวอร์ชันที่ผ่านการ Verify และกำหนด เป็นเวอร์ชัน Baseline แล้ว</li> <li>- เอกสาร Verification &amp; Validation Result เวอร์ชันที่ได้รับการ บันทึกผลการ Verify เอกสาร System Manual แล้ว</li> </ul>	
<p>13. จัดทำเอกสาร User Manual (Software User Documentation) ตามคุณลักษณะของ Software ที่ผ่านการทดสอบจากลูกค้าเรียบร้อยแล้ว โดยเอกสารควรประกอบด้วย ข้อมูลที่สำคัญดังต่อไปนี้</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- Software (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตาม</li> </ul>	<ul style="list-style-type: none"> <li>- Administration Staff เป็น ผู้ดำเนินการ</li> </ul>

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาท ผู้รับผิดชอบใน โครงการ
<ul style="list-style-type: none"> <li>○ ชื่อโปรแกรม/ชื่อระบบ, ชื่อลูกค้า</li> <li>○ Process Flow ภาพรวมของระบบซอฟต์แวร์</li> <li>○ รายการบทบาทหรือสิทธิ์การเข้าใช้งานระบบซอฟต์แวร์</li> <li>○ วิธีการใช้งานระบบซอฟต์แวร์โดยควรมีรูปภาพประกอบคำบรรยาย</li> <li>○ มีการบันทึกประวัติการแก้ไข (Revision History)</li> </ul>	<p>ขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)</p> <p>เวอร์ชันที่มีสถานะการทดสอบ</p> <p>เป็น “ผ่าน” ทุกรายการจากลูกค้าแล้ว</p> <p>- เอกสาร Software Requirements Specification เวอร์ชันล่าสุด</p>	
	<b>Output Products</b>	
*ตัวอย่างเอกสาร User Manual ดังภาพที่ 4.52-4.55	- เอกสาร User Manual (Software	
*ตัวอย่างเอกสาร Software Requirements Specification ดังภาพที่ 4.30-4.34	User User Documentation) ฉบับร่าง	

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาท ผู้รับผิดชอบในโครงการ
<p>14. นำเอกสาร User Manual (Software User Documentation) ให้ลูกค้าทบทวนเนื้อหาแล้วบันทึกการ Verify ในเอกสาร Verification &amp; Validation Results</p>	<p><b>Input Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร User Manual (Software User Documentation) ฉบับร่าง</li> <li>- เอกสาร Verification &amp; Validation Result เวอร์ชันล่าสุด</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- เอกสาร User Manual (Software User Documentation) เวอร์ชันที่ผ่านการ Verify และกำหนดเป็นเวอร์ชัน Baseline แล้ว</li> <li>- เอกสาร Verification &amp; Validation Result เวอร์ชันที่ได้รับการบันทึกผลการ Verify เอกสาร User Manual แล้ว</li> </ul>	<p>- <b>Team Leader/System Analyst</b> เป็นผู้ดำเนินการ</p>
<p>*ตัวอย่างเอกสาร User Manual ดังภาพที่ 4.52-4.55</p>		
<p>*ตัวอย่างเอกสาร Verification &amp; Validation Result ดังภาพที่ 4.16-4.17</p>		

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารเวอร์ชันที่ผ่านการที่เกี่ยวข้อง	บทบาทผู้รับ()ผิดชอบในโครงการ
<b>15. จัดเตรียมข้อมูลและเอกสารชุด Software Configuration เวอร์ชันฉบับร่างสำหรับเตรียมการส่งมอบปิดโครงการเสนอให้กับลูกค้า โดยข้อมูลและเอกสารในชุด Software Configuration ประกอบด้วย</b>	<b>Input Products</b>	<b>- Team Leader/System Analyst เป็นผู้ดำเนินการ</b>
○ เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34) เวอร์ชันล่าสุดที่ได้รับการลงนามยืนยันจากลูกค้าแล้ว	ได้รับการลงนามยืนยันจากลูกค้าแล้ว - เอกสาร Software Design เวอร์ชันล่าสุดที่ได้รับการลงนามยืนยันจากลูกค้าแล้ว	
○ เอกสาร Software Design (ตัวอย่างดังภาพที่ 4.35-4.37) เวอร์ชันล่าสุดที่ได้รับการลงนามยืนยันจากลูกค้าแล้ว	ได้รับการลงนามยืนยันจากลูกค้าแล้ว	
○ เอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41) เวอร์ชันล่าสุด	- เอกสาร Traceability Record เวอร์ชันล่าสุด	
○ Software ที่พัฒนาเสร็จแล้ว (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด) เวอร์ชันที่ผ่านการทดสอบและได้รับการลงนามผลการทดสอบจากลูกค้าแล้ว	- Software ที่พัฒนาเสร็จแล้ว (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)	
○ เอกสาร Test Cases and Test Procedures (ตัวอย่างดังภาพที่ 4.42-4.43) เวอร์ชันที่ผ่านการทดสอบและได้รับการลงนามผลการทดสอบจากลูกค้าแล้ว	ครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)	
○ เอกสาร Test Report (ตัวอย่างดังภาพที่ 4.44-4.45) เวอร์ชันที่ผ่านการทดสอบ	เวอร์ชันที่ผ่านการทดสอบและได้รับ	

## ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารเวอร์ชันที่ผ่านการที่เกี่ยวข้อง	บทบาทผู้รับ()ผิดชอบในโครงการ
และได้รับการลงนามผลการทดสอบจากลูกค้าแล้ว	การลงนามผลการทดสอบจากลูกค้าแล้ว	
○ เอกสาร System Manual (ครอบคลุม ส่วน Product Operation Guide และส่วน Maintenance Documentation) (ตัวอย่าง ค้างภาพที่ 4.46-4.51) เวอร์ชัน Baseline	แล้ว - เอกสาร Test Cases and Test Procedures เวอร์ชัน	
○ เอกสาร User Manual (Software User Documentation) (ตัวอย่าง ค้างภาพที่ 4.52-4.55) เวอร์ชัน Baseline	ที่ผ่านการทดสอบ และได้รับการลงนามผลการ	
	ทดสอบจากลูกค้าแล้ว - เอกสาร Test Report เวอร์ชันที่ผ่านการทดสอบและได้รับการลงนามผลการทดสอบจากลูกค้าแล้ว	
	- เอกสาร System Manual (ครอบคลุม ส่วน Product Operation Guide และส่วน Maintenance Documentation) เวอร์ชัน Baseline	

ตารางที่ 4.9 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารเวอร์ชันที่ผ่าน การที่เกี่ยวข้อง	บทบาทผู้รับ()ผิด ชอบในโครงการ
	<ul style="list-style-type: none"> <li>- เอกสาร User Manual (Software User Documentation) เวอร์ชัน Baseline</li> </ul> <p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- ข้อมูลและเอกสาร ชุด Software Configuration เวอร์ ชันฉบับร่าง</li> </ul>	





รหัสเอกสาร

SM-DSS-AGC-57-01A-080957

## เอกสาร System Manual

โปรแกรมระบบ \_\_\_\_\_

ชื่อโปรแกรม/ระบบ

สำหรับ

บริษัท \_\_\_\_\_

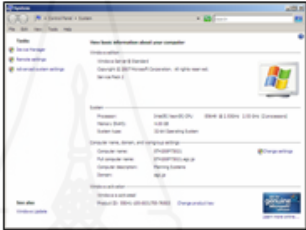
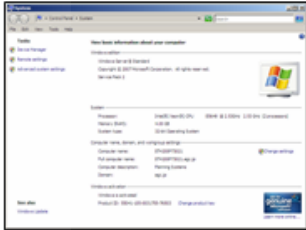
ชื่อลูกค้า

ส่วนบันทึกประวัติการแก้ไขและ  
ควบคุมเวอร์ชันของเอกสาร

## Revision History:

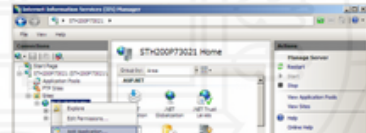
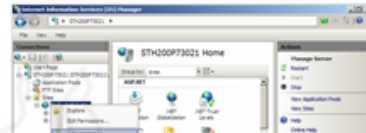
Version	Date	Author	Reviewer	Description
1	8 กันยายน 2557	คุณ _____	คุณ _____	Initial
2	10 กันยายน 2557	คุณ _____	คุณ _____	Baseline

ภาพที่ 4.46 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 1

1. การติดตั้งโปรแกรมระบบ		การเตรียมฮาร์ดแวร์และ Pre-requisite ซอฟต์แวร์
1.1. การเตรียมเครื่อง Server		
ก่อนการติดตั้งโปรแกรมฯ ควรมีการเตรียมเครื่อง Server ดังรายละเอียดต่อไปนี้		
Web Application Server	Production	Test
Server Name	STH200P73021	STH200P73021
IP Address	172.18.73.21	172.18.73.2
URL	http:// 172.18.73.21/dds	http:// 172.18.73.21/ddstest
Operating System	Microsoft Windows Server (R) Standard 	Microsoft Windows Server (R) Standard 
Hardware Spec	System Type : 32 bit CPU : Intel(R) Xeon(R) CPU E5649 @2.53GHz 2.53GHz (2 processors) RAM : 4 GB Hard Disk : Drive: C Space : 99.9 GB Hard Disk: Drive: D Space : 99.9 GB	System Type : 32 bit CPU : Intel(R) Xeon(R) CPU E5649 @2.53GHz 2.53GHz (2 processors) RAM : 4 GB Hard Disk: Drive: C Space : 99.9 GB Hard Disk: Drive: D Space : 99.9 GB
Software/Component	Microsoft.NET Framework 4.0	Microsoft Internet Information Services (IIS) 7.0 หรือ 7.5

ภาพที่ 4.47 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 2

ขั้นตอนการติดตั้งซอฟต์แวร์ที่ทีมงาน  
ดำเนินโครงการเป็นผู้พัฒนาขึ้น

การติดตั้งซอฟต์แวร์ Web Application ใน Web Server: IIS 7.0		
การติดตั้ง Web Application ใน Web Server: IIS 7.0	Production	Test
Web Application Name :	DDS	DDSTest
<u>ขั้นตอนดังนี้:</u>	<ul style="list-style-type: none"> <li>- ไปที่ Control Panel &gt; Administrative Tools</li> <li>- Double Click ที่</li> <li>- เลือกที่ชื่อ Server ในช่องด้านซ้าย &gt; Sites &gt; Default Web Site</li> <li>- Clickขวา &gt; Add Application</li> <li>- เลือกApplication Pools เป็นASP .NET v4.0, Alias: ชื่อ "DSS"</li> <li>- เลือก Browse &gt; Physical path ไปที่ "D:\DDS_WebProduction" (หรือ Folder ที่เก็บไฟล์โปรแกรมฯ ไว้)</li> <li>- เสร็จแล้วเลือก OK จะปรากฏ Virtual Directory "DSS" ทางด้านซ้าย สามารถทดสอบ run WebApplication ได้โดยการ click Virtual Directory "DSS" ทางด้านซ้าย แล้ว click "Browse *.80 (http) ทางด้านขวา</li> </ul> 	<ul style="list-style-type: none"> <li>- ไปที่ Control Panel &gt; Administrative Tools</li> <li>- Double Click ที่</li> <li>- เลือกที่ชื่อ Server ในช่องด้านซ้าย &gt; Sites &gt; Default Web Site</li> <li>- Clickขวา &gt; Add Application</li> <li>- เลือกApplication Pools เป็นASP .NET v4.0, Alias: ชื่อ "DSSTest"</li> <li>- เลือก Browse &gt; Physical path ไปที่ "D:\DDS_WebTest" (หรือ Folder ที่เก็บไฟล์โปรแกรมฯ ไว้)</li> <li>- เสร็จแล้วเลือก OK จะปรากฏ Virtual Directory "DDSTest" ทางด้านซ้าย สามารถทดสอบ run WebApplication ได้โดยการ click Virtual Directory "DDSTest" ทางด้านซ้าย แล้ว click "Browse *.80 (http) ทางด้านขวา</li> </ul> 

ภาพที่ 4.48 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 3

## 2. Development Tool

เครื่องคอมพิวเตอร์ที่จะใช้ในการพัฒนาแก้ไขโปรแกรมฯ ควรมีคุณลักษณะดังนี้

Development Tool	Specification
Hardware Specification	<p>System Type : 64 bit</p> <p>CPU : 1 x Core i5 2.40 GHz หรือสูงกว่า</p> <p>RAM : 4 GB หรือมากกว่า</p> <p>Hard Disk : Drive: C Space : 100 GB หรือสูงกว่า</p> <p>Hard Disk : Drive: D Space : 200 GB หรือสูงกว่า</p>
Operating System	<ul style="list-style-type: none"> <li>- Microsoft Windows 7</li> <li>- Microsoft Windows 8</li> </ul>
Software/Component	<ul style="list-style-type: none"> <li>- Microsoft Visual Studio.NET 2010: ASP.NET with VB.NET</li> <li>หรือ Microsoft Visual Studio.NET 2012</li> <li>- Microsoft .NET Framework 4.0</li> <li>- Microsoft SQL Server 2008 R2 Standard Edition</li> <li>- Microsoft SQL Server 2008 Management Studio</li> <li>- Microsoft IIS 7.0 หรือ 7.5</li> </ul>

ภาพที่ 4.49 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 4

### 3. คุณสมบัติเครื่อง Client

คุณสมบัติเครื่อง Client	Production
Hardware Specification	<b>System Type : 64 bit</b> <b>CPU : 1 x Core i5 2.40 GHz หรือสูงกว่า</b> <b>RAM : 1 GB หรือมากกว่า</b> <b>Hard Disk: Drive: C Space : 80 GB</b> <b>หรือสูงกว่า</b> <b>Hard Disk: Drive: D Space : 100 GB</b> <b>หรือสูงกว่า</b>
Operating System	- Microsoft Windows XP - Microsoft Windows 7 - Microsoft Windows 8
Software/Component	- Microsoft Internet Explorer V.8 ขึ้นไป - Microsoft Excel 2007 ขึ้นไป

ภาพที่ 4.50 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 5



การปรับเปลี่ยนค่า Configuration  
ของซอฟต์แวร์ (การบำรุงรักษาระบบ)

#### 4. วิธีการกำหนดค่า Configuration ของโปรแกรม

1.1. การแก้ไขค่า ConnectionString สำหรับ Web Application ในไฟล์ web.config โดยข้อมูลที่ "D:\DDS\_WebProduction" และ "D:\DDS\_WebTest" มีขั้นตอนดังนี้

การกำหนดค่า Configuration ของโปรแกรม	Production	Test
Web.config	D:\DDS_WebProduction	D:\DDS_WebTest

Connection String : สำหรับติดต่อ ฐานข้อมูล Microsoft SQL Server 2008	Production	Test
Connection String	<add name="PDSConnectionString" connectionString="Data Source=172.18.73.21;Initial Catalog=dss;UserID=xx; Password=password" providerName="System.Data.SqlClient" >	<add name="PDSConnectionString" connectionString="Data Source=172.18.73.21;Initial Catalog=dsstest;User ID=xx; Password=password" providerName="System.Data.SqlClient" >
Server	Database Server Name\Instance ame	Database Server Name\Instance ame
Uid	Database Login Name	Database Login Name
Pwd	Database Login Password	Database Login Password
Initial Catalog	Database Name	Database Name

ภาพที่ 4.51 ตัวอย่างเอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) ส่วนที่ 6

รหัสเอกสาร

UM-DSS-AGC-57-01A-080957

## เอกสาร User Manual

โปรแกรมระบบ \_\_\_\_\_

ชื่อโปรแกรม/ระบบ

สำหรับ

บริษัท \_\_\_\_\_

ชื่อลูกค้า

ส่วนบันทึกประวัติการแก้ไขและ  
ควบคุมเวอร์ชันของเอกสาร

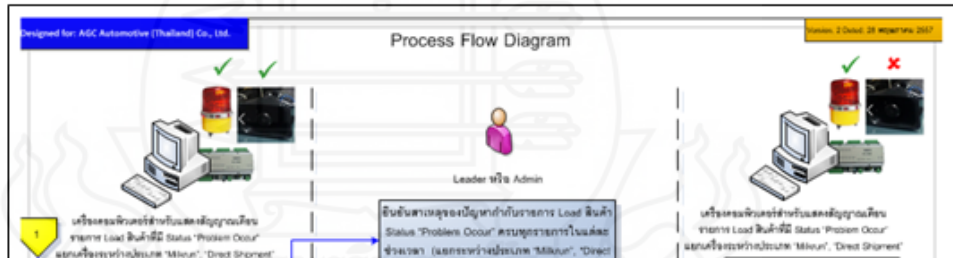
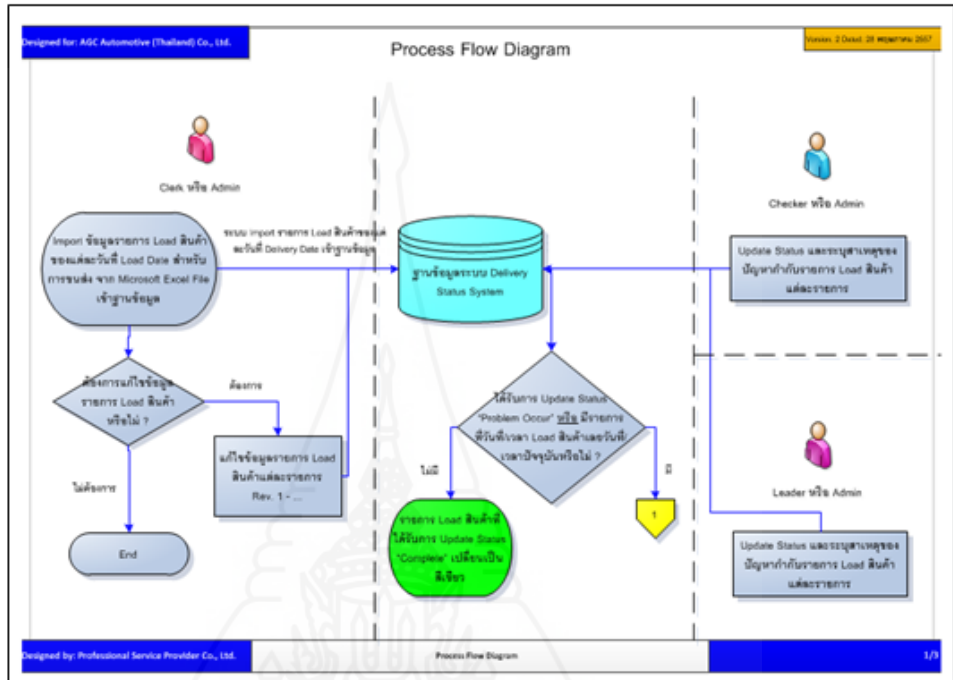
## Revision History:

Version	Date	Author	Reviewer	Description
1	8 กันยายน 2557	คุณ _____	คุณ _____	Initial
2	10 กันยายน 2557	คุณ _____	คุณ _____	Baseline

ภาพที่ 4.52 ตัวอย่างเอกสาร User Manual (Software User Documentation) ส่วนที่ 1

Process Flow ภาพรวมของระบบซอฟต์แวร์

1. Process Flow & System Architecture ภาพรวมของโปรแกรมระบบ



ภาพที่ 4.53 ตัวอย่างเอกสาร User Manual (Software User Documentation) ส่วนที่ 2



รายการบทบาทหรือสิทธิ์การเข้า  
ใช้งานระบบซอฟต์แวร์

## 2. บทบาทการเข้าใช้งานและสิทธิ์การใช้งาน

ลำดับ	ชื่อระบบ	ชื่อบทบาท (Role)	ประเภทของสิทธิ์การ ทำงาน กับเอกสาร	สิทธิ์การทำงาน	สิทธิ์การมองเห็นเอกสาร
<b>Role _____</b>					
1	—	Role _____ (Upload & Revise)	-	<ul style="list-style-type: none"> <li>ค้นหาข้อมูลรายการ Load สินค้า</li> <li>Import ข้อมูลวันที่ Delivery Date รายวัน</li> <li>แก้ไขข้อมูลรายการ Load สินค้าแต่ละรายการ</li> <li>เปลี่ยนพาสเวิร์ดได้</li> </ul>	
<b>Role _____</b>					
1	—	Role Checker (Update)	-	<ul style="list-style-type: none"> <li>ค้นหาข้อมูลรายการ Load สินค้า</li> <li>Update Status ข้อมูลรายการ Load สินค้า แต่ละรายการ ดังนี้ <ul style="list-style-type: none"> <li>- Wait for Loading</li> <li>- Problem Occur</li> <li>- Complete</li> </ul> </li> <li>ระบบสาเหตุของปัญหาสำหรับรายการ Load สินค้าที่มี Status "Problem Occur" แต่ละรายการ</li> <li>เปลี่ยนพาสเวิร์ดได้</li> </ul>	

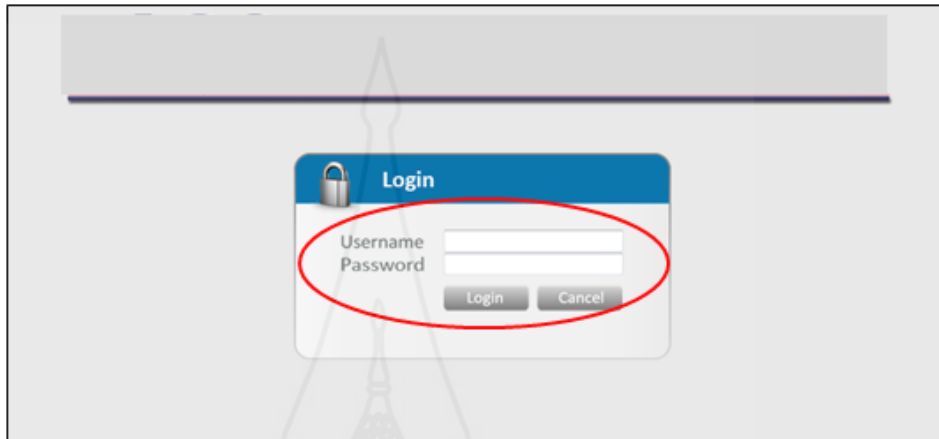
ภาพที่ 4.54 ตัวอย่างเอกสาร User Manual (Software User Documentation) ส่วนที่ 3

วิธีการใช้งานระบบซอฟต์แวร์โดย  
ควรมีรูปภาพประกอบคำบรรยาย

**4. คุณลักษณะของโปรแกรมระบบ Delivery Status System สำหรับ Role \_\_\_\_\_**

**4.1. การค้นหาข้อมูล**

4.1.1. ผู้ใช้ระบบ Login เข้าสู่ระบบ



4.1.2. เลือกเมนู "Upload & Revise"



ภาพที่ 4.55 ตัวอย่างเอกสาร User Manual (Software User Documentation) ส่วนที่ 4

**2.2.6 SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ซอฟต์แวร์)**

1) ผู้รับผิดชอบในโครงการสำหรับ Activity นี้และบทบาทความเกี่ยวข้องมี  
ดังนี้

(1) Team Leader/System Analyst ทบทวนสถานะโครงการร่วมกับทีมงาน  
ดำเนินโครงการและมอบหมายงานส่วนคงค้าง (ถ้ามี) ให้ดำเนินงานครบถ้วนตามเอกสาร Project  
Plan เวอร์ชันล่าสุด

(2) Team Leader/System Analyst ตรวจสอบและ Update ข้อมูลและเอกสารชุด Software Configuration เป็นเวอร์ชันล่าสุดสำหรับการส่งมอบปิดโครงการเสนอให้กับลูกค้า โดยข้อมูลและเอกสารในชุด Software Configuration ประกอบด้วย

ก. เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34)

ข. เอกสาร Software Design (ตัวอย่างดังภาพที่ 4.35-4.37)

ค. เอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41)

ง. Software ที่พัฒนาเสร็จแล้ว (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)

จ. เอกสาร Test Cases and Test Procedures (ตัวอย่างดังภาพที่ 4.42-4.43)

ฉ. เอกสาร Test Report (ตัวอย่างดังภาพที่ 4.44-4.45)

ช. เอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) (ตัวอย่างดังภาพที่ 4.46-4.51)

ซ. เอกสาร User Manual (Software User Documentation) (ตัวอย่างดังภาพที่ 4.52-4.55)

2) เอกสารที่เกี่ยวข้องกับ Activity นี้มี ดังนี้

(1) Input Products

ก. เอกสาร Project Plan (ตัวอย่างดังภาพที่ 4.3-4.15) เวอร์ชันล่าสุด  
ข. ข้อมูลและเอกสารชุด Software Configuration ฉบับร่างโดยข้อมูลและเอกสารในชุด Software Configuration ประกอบด้วย

ก) เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34)

ข) เอกสาร Software Design (ตัวอย่างดังภาพที่ 4.35-4.37)

ค) เอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41)

ง) Software ที่พัฒนาเสร็จแล้ว (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)

จ) เอกสาร Test Cases and Test Procedures (ตัวอย่างดังภาพที่ 4.42-4.43)

ฉ) เอกสาร Test Report (ตัวอย่างดังภาพที่ 4.44-4.45)

ช) เอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) (ตัวอย่างดังภาพที่ 4.46-4.51)

ซ) เอกสาร User Manual (Software User Documentation) (ตัวอย่างดังภาพที่ 4.52-4.55)

## (2) Output Products

ก. ข้อมูลและเอกสารชุด Software Configuration เวอร์ชันล่าสุด สำหรับการส่งมอบให้กับลูกค้าเพื่อปิดโครงการ โดยประกอบด้วย

ก) เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34)

ข) เอกสาร Software Design (ตัวอย่างดังภาพที่ 4.35-4.37)

ค) เอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41)

ง) Software ที่พัฒนาเสร็จแล้ว (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตามขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)

จ) เอกสาร Test Cases and Test Procedures (ตัวอย่างดังภาพที่ 4.42-4.43)

ฉ) เอกสาร Test Report (ตัวอย่างดังภาพที่ 4.44-4.45)

ช) เอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) (ตัวอย่างดังภาพที่ 4.46-4.51)

ซ) เอกสาร User Manual (Software User Documentation) (ตัวอย่างดังภาพที่ 4.52-4.55)

## 3) รายละเอียดเพิ่มเติมมี ดังนี้

(1) Team Leader/System Analyst ควรตรวจสอบข้อมูลและเอกสารในชุด Software Configuration ให้ครบถ้วนเพื่อจะสามารถเสนอการปิดโครงการกับลูกค้าได้อย่างรวดเร็วยิ่งขึ้น

**บทวิเคราะห์และการพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับ Activity SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ซอฟต์แวร์)**

SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ซอฟต์แวร์) เป็นการทบทวนสถานะการดำเนินโครงการช่วงสุดท้ายโดย **Team Leader/System Analyst** และทำการมอบหมายหมายงานให้ทีมงานดำเนินโครงการดำเนินงานคงค้างให้ครบถ้วนตามเอกสาร Project Plan เวอร์ชัน

ล่าสุด เพื่อ **Team Leader/System Analyst** จะสามารถจัดเตรียมข้อมูลและเอกสารชุด Software Configuration

สำหรับการส่งมอบงานปิดโครงการเสนอให้กับลูกค้าได้อย่างรวดเร็ว โดยเมื่อ **Team Leader/System Analyst** จัดเตรียมข้อมูลและเอกสารชุด Software Configuration เวอร์ชันล่าสุดที่พร้อมสำหรับการส่งมอบงานปิดโครงการเสนอให้กับลูกค้าแล้ว **Team Leader/System Analyst** ควรประสานงานร่วมกับ **Project Manager** เพื่อนำจดหมายส่งมอบงานปิดโครงการ (Acceptance Record) ส่งมอบพร้อมกับข้อมูลและเอกสารชุด Software Configuration เวอร์ชันล่าสุด และให้ลูกค้าลงนามรับมอบการปิดโครงการในจดหมายส่งมอบงานปิดโครงการ (Acceptance Record) จึงจะเป็นการปิดโครงการอย่างสมบูรณ์ได้

ตารางที่ 4.10 แนวปฏิบัติที่ดีที่ตามมาตราฐาน ISO/IEC 29110 ระดับ Basic Profile

สำหรับ Activity SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ซอฟต์แวร์)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบในโครงการ
<p>1. ทบทวนสถานะโครงการร่วมกับทีมงานดำเนินโครงการและมอบหมายงานส่วนคงค้าง (ถ้ามี) ให้ดำเนินงานครบถ้วนตามเอกสาร Project Plan เวอร์ชันล่าสุด</p> <p>*ตัวอย่างเอกสาร Project Plan ดังภาพที่ 4.3-4.15</p>	<p><b>Input Products</b></p> <p>- เอกสาร Project Plan เวอร์ชันล่าสุด</p> <p><b>Output Products</b></p>	<p>- Team Leader/System Analyst และทีมงานดำเนินโครงการเป็นผู้ดำเนินการ</p>
<p>2. ตรวจสอบและ Update ข้อมูลและเอกสารชุด Software Configuration เป็นเวอร์ชันล่าสุด สำหรับการส่งมอบปิดโครงการเสนอให้กับลูกค้า โดยข้อมูลและเอกสารในชุด Software Configuration ประกอบด้วย</p> <p>- เอกสาร Software Requirements Specification(ตัวอย่างดังภาพที่ 4.30-4.34)</p> <p>- เอกสาร Software Design (ตัวอย่างดังภาพที่ 4.35-4.37)</p>	<p><b>Input Products</b></p> <p>- ข้อมูลและเอกสารชุด Software Configuration ฉบับร่าง</p>	<p>- Team Leader/System Analyst เป็นผู้ดำเนินการ</p>

## ตารางที่ 4.10 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<ul style="list-style-type: none"> <li>- เอกสาร Traceability Record(ตัวอย่างดัง ภาพที่ 4.40-4.41)</li> <li>- Software ที่พัฒนาเสร็จแล้ว (ซอฟต์แวร์ที่มี คุณลักษณะครบถ้วนตามขอบข่ายงานใน เอกสาร Project Plan เวอร์ชันล่าสุด)</li> <li>- เอกสาร Test Cases and Test Procedures( ตัวอย่างดังภาพที่ 4.42-4.43)</li> <li>- เอกสาร Test Report(ตัวอย่างดังภาพที่ 4.44-4.45)</li> <li>- เอกสาร System Manual (ครอบคลุมส่วน Product Operation Guide และส่วน Maintenance Documentation) (ตัวอย่างดัง ภาพที่ 4.46-4.51)</li> <li>- เอกสาร User Manual (Software User Documentation) (ตัวอย่างดังภาพที่ 4.52- 4.55)</li> </ul>	<p><b>Output Products</b></p> <ul style="list-style-type: none"> <li>- ข้อมูลและ เอกสารชุด Software Configuration เวอร์ชันล่าสุด สำหรับการส่ง มอบให้กับลูกค้า เพื่อปิดโครงการ</li> </ul>	

ตารางที่ 4.10 (ต่อ)

แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p><b>3. นำจดหมายส่งมอบงานปิดโครงการ (Acceptance Record) ส่งมอบพร้อมทั้งข้อมูลและเอกสารชุด Software Configuration เวอร์ชันล่าสุดให้กับลูกค้า และให้ลูกค้าลงนามรับมอบการปิดโครงการ ในจดหมายส่งมอบงานปิดโครงการ (Acceptance Record) โดยข้อมูลและเอกสาร ในชุด Software Configuration ประกอบด้วย</b></p> <ul style="list-style-type: none"> <li>- เอกสาร Software Requirements Specification (ตัวอย่างดังภาพที่ 4.30-4.34)</li> <li>- เอกสาร Software Design (ตัวอย่างดังภาพที่ 4.35-4.37)</li> <li>- เอกสาร Traceability Record (ตัวอย่างดังภาพที่ 4.40-4.41)</li> <li>- Software ที่พัฒนาเสร็จแล้ว (ซอฟต์แวร์ที่มีคุณลักษณะครบถ้วนตาม ขอบข่ายงานในเอกสาร Project Plan เวอร์ชันล่าสุด)</li> <li>- เอกสาร Test Cases and Test Procedures (ตัวอย่างดังภาพที่ 4.42-4.43)</li> <li>- เอกสาร Test Report (ตัวอย่างดังภาพที่ 4.44-4.45)</li> </ul>	<p><b><u>Input Products</u></b></p> <ul style="list-style-type: none"> <li>- ข้อมูลและเอกสารชุด Software Configuration เวอร์ชันล่าสุดสำหรับการส่งมอบให้กับลูกค้าเพื่อปิดโครงการ</li> <li>- จดหมายส่งมอบงานปิดโครงการ (Acceptance Record)</li> </ul> <p><b><u>Output Products</u></b></p> <ul style="list-style-type: none"> <li>- จดหมายส่งมอบงานปิดโครงการ (Acceptance Record) ที่ได้รับการลงนามรับมอบงานการปิดโครงการจากลูกค้าเรียบร้อยแล้ว</li> </ul>	<p>- <b>Project Manager และ Team Leader/System Analyst</b> เป็นผู้ดำเนินการ</p>

## ตารางที่ 4.10 (ต่อ)

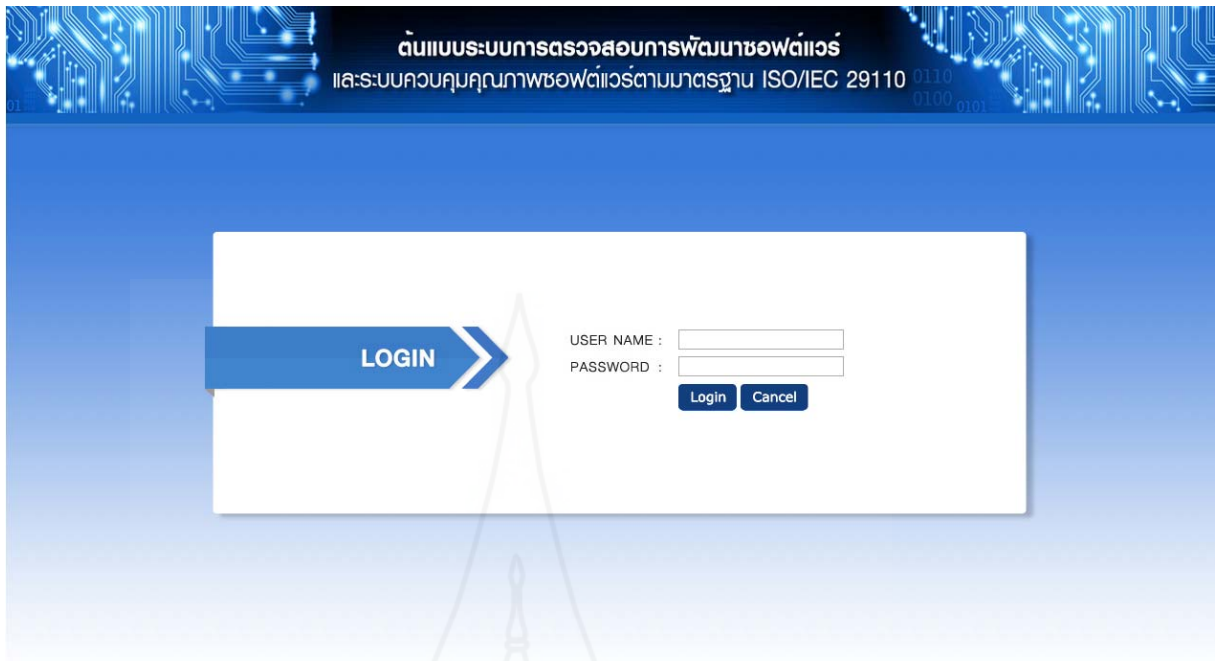
แนวปฏิบัติที่ดี	เอกสารที่เกี่ยวข้อง	บทบาทผู้รับผิดชอบ ในโครงการ
<p>- เอกสาร System Manual (ครอบคลุม ส่วน Product Operation Guide และ ส่วน Maintenance Documentation) (ตัวอย่างดังภาพที่ 4.46-4.51)</p> <p>- เอกสาร User Manual (Software User Documentation) (ตัวอย่างดังภาพที่ 4.52-4.55)</p> <p>*ตัวอย่างจดหมายส่งมอบงานปิดโครงการ (Acceptance Record) ดังภาพที่ 4.28-4.29</p>		-

### 3. ผลจากการศึกษา, วิเคราะห์และพัฒนาด้านแบบระบบตรวจสอบการพัฒนาซอฟต์แวร์ และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile

ต้นแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile ควรมีคุณลักษณะที่สำคัญดังนี้

3.1 การพิสูจน์ตัวตนและการรักษาความลับ: ระบบมีส่วนควบคุมการ Login เข้าระบบ โดยจะต้องกรอก User Name และ Password เพื่อการพิสูจน์ตัวตนก่อนการเข้าทำงานกับระบบ และระบบมีการควบคุมให้ผู้เกี่ยวข้องกับโครงการใดๆ สามารถค้นหาและทำงานกับมูลโครงการนั้นๆ ได้เท่านั้น ทั้งนี้ยกเว้นผู้ดูแลระบบสามารถค้นหาข้อมูลโครงการได้ทั้งหมดดังภาพที่ 4.56





ภาพที่ 4.56 ต้นแบบหน้าจอสำหรับ Login เข้าสู่ระบบเพื่อการพิสูจน์ตัวตนและรักษาความลับ

**3.2 การติดตามสถานะโครงการ (Project Status Tracking):** โดยการเลือกเมนู Project Status Tracking ผู้ใช้ระบบจะสามารถค้นหาข้อมูลโครงการ และติดตามภาพรวมสถานะของโครงการตาม Standard Process ของ ISO/IEC 29110 ได้อย่างสะดวกรวดเร็ว โดยระบบจะแสดงสัญลักษณ์แถบสีเขียวกำกับ Process ที่ดำเนินงานเสร็จและมีการจัดเก็บไฟล์เข้าระบบภายในระยะเวลาที่กำหนดไว้, แสดงสัญลักษณ์แถบสีเหลืองกำกับ Process ที่ดำเนินงานเสร็จและมีการจัดเก็บไฟล์เข้าระบบเกินระยะเวลาที่กำหนดไว้ไม่เกินจำนวนวันที่ยอมรับได้, และแสดงสัญลักษณ์แถบสีแดงกำกับ Process ที่ยังดำเนินงานไม่เสร็จหรือดำเนินงานเสร็จเกินจำนวนวันที่ไม่สามารถยอมรับได้ดังภาพที่ 4.57

**ต้นแบบระบบการตรวจสอบการพัฒนาซอฟต์แวร์**  
และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110

0110  
0100  
0101

**Project Status Tracking**    **Project Repository**    **Standard Information**    **Management**    User: FirstName LastName [Role:.....]    **Logout**

**Project Status Tracking**

ชื่อโครงการ :     ระยะเวลาโครงการ : ระหว่างวันที่  ถึงวันที่   
 ชื่อลูกค้า :     สถานะโครงการ :     **Search**

**New Project**

ลำดับ	ชื่อโครงการ	ชื่อลูกค้า	ระยะเวลาโครงการ	สถานะโครงการ	PM.01	PM.02	PM.03	PM.04	PM.05	PM.06	PM.07	SL.01	SL.02	SL.03	SL.04	SL.05	SL.06	SL.07
1	โครงการพัฒนาโปรแกรมบริหารความเสี่ยง	บริษัท Customer A จำกัด	1/2/1014 - 30/9/2014	In Progress	■	■	■	■	■	■	■	■	■	■	■	■	■	■
2	โครงการพัฒนาโปรแกรมบริหารลูกค้า	บริษัท Customer C จำกัด	1/10/1013 - 31/12/2013	Finished	■	■	■	■	■	■	■	■	■	■	■	■	■	■
3																		
4																		
5																		

[1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#)

■ หมายถึง กิจกรรมที่ดำเนินงานเสร็จภายในมาตรฐานระยะเวลาดำเนินงาน  
 ■ หมายถึง กิจกรรมที่ดำเนินงานเสร็จภายในระยะเวลาดำเนินงานเกินมาตรฐานในระดับที่ยอมรับได้  
 ■ หมายถึง กิจกรรมที่ดำเนินงานเสร็จภายในระยะเวลาดำเนินงานเกินมาตรฐานในระดับที่ยอมรับไม่ได้

ภาพที่ 4.57 ต้นแบบหน้าจอสำหรับการติดตามสถานะโครงการ (Project Status Tracking)

**3.3 φόρμบบันทึกข้อมูลโครงการแยกตามความรับผิดชอบ (Project Form):** เป็นฟอร์มสำหรับบันทึกรายละเอียดและจัดเก็บข้อมูลและเอกสารของโครงการสำหรับแต่ละกระบวนการตาม Standard Process ของ ISO/IEC 29110 โดย Project Manager สามารถกำหนดผู้รับผิดชอบสำหรับแต่ละกระบวนการ ซึ่งผู้รับผิดชอบแต่ละรายก็จะสามารถ Update ข้อมูลเข้าระบบได้เฉพาะกระบวนการที่รับผิดชอบตามที่ได้รับการกำหนดสิทธิ์ตามความรับผิดชอบเท่านั้นดังภาพที่ 4.58-4.62

**ต้นแบบระบบการตรวจสอบการพัฒนาซอฟต์แวร์**  
 และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110

Project Status Tracking
Project Repository
Standard Information
Management
User: FirstName LastName [Role.....]
Logout

Project Repository

ข้อมูลโครงการ

ชื่อโครงการ :

ชื่อลูกค้า :

ระยะเวลาโครงการ :  ถึง  สถานะโครงการ :

ข้อมูลโครงการโดยย่อ :

รายชื่อทีมงานส่วนในโครงการ (Project Member):

ลำดับ	ชื่อ	Project Role	Process ที่รับผิดชอบ	
1	คุณ Project Manager	Project Manager	PM.1, PM.2, PM.3, PM.4 SI.1, SI.2, SI.3, SI.4, SI.5, SI.6	✖
2	คุณ Team Leader/System Analyst	Team Leader/System Analyst	SI.1, SI.2, SI.3, SI.4, SI.5, SI.6	✖
3	คุณ Senior Programmer	Senior Programmer	SI.4, SI.5	✖
4	คุณ Programmer	Programmer	SI.4, SI.5	✖
5	คุณ Tester	Tester	SI.4, SI.5	✖
6	คุณ Admin.	Admin.	SI.5	✖

ISO/IEC 29110 Processes

Project Management Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์)

PM.1: Project Planning (การวางแผนโครงการ)

Checklist	Tasks	Input Products	Output Products	หมายเหตุ	ผู้รับผิดชอบ	วันที่กำหนดเสร็จงาน	วันที่เสร็จงานจริง
<input checked="" type="checkbox"/>	- จัดทำเอกสาร Project Plan	- สัญญาหรือเอกสารยืนยันการจ้างจากลูกค้า <input type="text" value="Contract.pdf"/>	- เอกสาร Project Plan เวอร์ชัน Baseline ที่ได้รับการลงนามยืนยันจากลูกค้า <input type="text" value="PP-TMS-BPP-56-01A-050256"/>		คุณ Project Manager	<input type="text" value="7/2/2014"/>	<input type="text" value="7/2/2014"/>
<input checked="" type="checkbox"/>	- จัดทำเอกสาร Verification & Validation Results	- เอกสาร Project Timeframe <input type="text"/>	- เอกสาร VF & VD Results <input type="text" value="VFD-TMS-BPP-56-01A-0502"/>				
<input checked="" type="checkbox"/>	- จัดทำเอกสาร Meeting Record (บันทึกข้อความการประชุม)	- เอกสารการวางแผนภายในองค์กร <input type="text"/>	- เอกสารรายงานการประชุม (Meeting Record) <input type="text" value="MR-TMS-BPP-56-01A-08025"/>				
		- เอกสาร Project Plan ฉบับร่าง <input type="text"/>	- เอกสาร Project Plan ฉบับร่าง <input type="text"/>				

PM.2: Project Plan Execution (การปฏิบัติตามแผนโครงการ)

PM.3: Project Assessment and Control (การประเมินและควบคุมโครงการ)

Software Implementation Process (กระบวนการพัฒนาผลิตภัณฑ์ซอฟต์แวร์)

SI.1 Software Implementation Initiation (การเริ่มการพัฒนาดังกล่าว)

SI.2 Software Requirements Analysis (การวิเคราะห์ความต้องการของผลิตภัณฑ์)

SI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้างสถาปัตยกรรมและรายละเอียดของผลิตภัณฑ์)

SI.4 Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์)

SI.5 Software Integration and Tests (การทดสอบและบูรณาการซอฟต์แวร์)

SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ซอฟต์แวร์)

Create By :                      คุณ Project Manager  
 Create date/time :           7/2/2014 10:00  
 Last update by :               คุณ Project Manager  
 Last update date / time :   7/2/2014 10:00

ภาพที่ 4.58 ตัวอย่างฟอร์มบันทึกข้อมูลโครงการแยกตามความรับผิดชอบโดย Project Manager

**ต้นแบบระบบการตรวจสอบการพัฒนาซอฟต์แวร์**  
 และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110

0110  
0100  
0100

---

**Project Status Tracking**   **Project Repository**   Standard Information   Management   User: FirstName LastName [Role:.....]   Logout

---

**Project Repository**

**Project Information**

ชื่อโครงการ : โครงการพัฒนาโปรแกรมระบบบริหารความเสี่ยง

ชื่อลูกค้า : บริษัท Customer A จำกัด

ระยะเวลาโครงการ : 1/2/1014   31/5/2014   สถานะโครงการ : In Progress

ข้อมูลโครงการโดยย่อ : โปรแกรม Web Application สำหรับการบริหารจัดการข้อมูลความเสี่ยงในการดำเนินงานของหน่วยงานต่าง ๆ ในองค์กร

**งานเขียนงานส่วนนี้โครงการ (Project Member):**

ลำดับชื่อ	ชื่อ	Project Role	Process ที่รับผิดชอบ
1	คุณ Project Manager	Project Manager	PM.1, PM.2, PM.3, PM.4 SI.1, SI.2, SI.3, SI.4, SI.5, SI.6
2	คุณ Team Leader/System Analyst	Team Leader/System Analyst	SI.1, SI.2, SI.3, SI.4, SI.5, SI.6
3	คุณ Senior Programmer	Senior Programmer	SI.4, SI.5
4	คุณ Programmer	Programmer	SI.4, SI.5
5	คุณ Tester	Tester	SI.4, SI.5
6	คุณ Admin.	Admin.	SI.5

**ISO/IEC 29110 Processes**

**Project Management Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์)**

**PM.1: Project Planning (การวางแผนโครงการ)**

Checklist	Tasks	Input Products	Output Products	หมายเหตุ	ผู้รับผิดชอบ	วันที่ดำเนินการเสร็จงาน	วันที่เสร็จงานจริง
✓	- จัดทำเอกสาร Project Plan	- สัญญาหรือเอกสารยืนยันการขาย ว่าจ้างจากลูกค้า ContractLpdf	- เอกสาร Project Plan เวอร์ชัน Baseline ที่ได้รับการลงนามยืนยันจากลูกค้า PP-TMS-BPP-56-01A-050256.docx		คุณ Project Manager	7/2/2014	7/2/2014
✓	- จัดทำเอกสาร Verification & Validation Results	- เอกสาร Project Timeframe	- เอกสาร VF & VD Results VFD-TMS-BPP-56-01A-050256.xlsx				
✓	- จัดทำเอกสาร Meeting Record (บันทึกทรงงานการประชุม)	- เอกสารตารางแผนงานภายในองค์กร - เอกสาร Project Plan ฉบับร่าง	- เอกสารรายงานการประชุม (Meeting Record) MR-TMS-BPP-56-01A-080256.docx - เอกสาร Project Plan ฉบับร่าง				

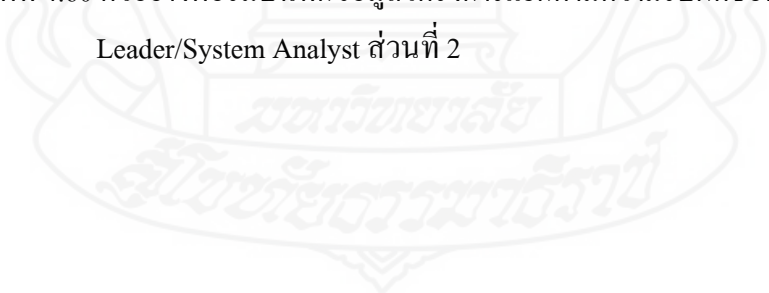
**PM.2: Project Plan Execution (การปฏิบัติตามแผนโครงการ)**

**PM.3: Project Assessment and Control (การประเมินและควบคุมโครงการ)**

ภาพที่ 4.59 ตัวอย่างฟอร์มบันทึกข้อมูลโครงการแยกตามความรับผิดชอบโดย Team Leader/System Analyst ส่วนที่ 1

Software Implementation Process (กระบวนการพัฒนาตั้งแต่เริ่มจนถึง)							
SI.1 Software Implementation Initiation (การเริ่มการพัฒนาดังข้อซอฟต์แวร์)							
SI.2 Software Requirements Analysis (การวิเคราะห์ความต้องการซอฟต์แวร์)							
Checklist	Tasks	Input Products	Output Products	หมายเหตุ	ผู้รับผิดชอบ	วันที่กำหนดเสร็จงาน	วันที่เสร็จงานจริง
<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/></li> <li><input checked="" type="checkbox"/></li> <li><input checked="" type="checkbox"/></li> <li><input checked="" type="checkbox"/></li> <li><input checked="" type="checkbox"/></li> </ul>	<ul style="list-style-type: none"> <li>- จัดทำเอกสาร Software Requirements Specification ฉบับร่าง</li> <li>- บทบาทของผู้ออกเอกสาร Software Requirements Specification เปลี่ยนที่ขึ้นกับรายละเอียดในเอกสาร Project Plan เสมือนในเอกสาร Project Plan เวอร์ชันล่าสุด</li> <li>- ชื่นกักสถานะเอกสาร Software Requirements Specification ผ่านการ Verify แล้ว</li> <li>- นำเสนอเอกสาร Software Requirements Specification เพื่อให้อุดหนุนงานและลงนาม</li> <li>- ชื่นกักสถานะเอกสาร Software Requirements Specification ผ่านการ Validate แล้วในเอกสาร Verification &amp; Validation Results</li> </ul>	<ul style="list-style-type: none"> <li>- เอกสาร Project Plan เวอร์ชันล่าสุด <a href="#">PP-TMS-BPP-56-01A-050256.docx</a></li> </ul>	<ul style="list-style-type: none"> <li>- เอกสาร Software Requirements Specification ฉบับร่าง SRS-TMS-BPP-56-01C</li> <li>- เอกสาร Software Requirements Specification ที่ผ่านการ Verify แล้ว (one time check only) SRS-TMS-BPP-56-01A</li> <li>- เอกสาร VF &amp; VD Results ที่บันทึกสถานะ SRS ผ่านการ Verify แล้ว VFD-TMS-BPP-56-02A</li> <li>- เอกสาร Software Requirements Specification ที่ถูกส่งนามแล้ว SRS-TMS-BPP-56-02A</li> <li>- เอกสาร VF &amp; VD Results ที่บันทึกสถานะ SRS ผ่านการ Validate แล้ว VFD-TMS-BPP-56-03A</li> </ul>	เนื่องจากทางลูกค้าขอใช้ระยะเวลาในการตรวจสอบเอกสาร SRS เพิ่มเติม 3 วันทำการจึงทำให้กำหนดการลงนามขึ้นเป็นเอกสาร SRS เสียออกไป	คุณ Team Leader/ System Analyst	15/3/2014	7/3/2014
SI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้างสถาปัตยกรรมและรายละเอียดซอฟต์แวร์)							
SI.4: Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์)							
SI.5: Software Integration and Tests (การทดสอบและการบูรณาการซอฟต์แวร์)							
SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ซอฟต์แวร์)							
<b>Save</b>				<b>Close</b>			
						Create By :	คุณ TL/SA
						Create date/time :	17/3/2014 10:00
						Last update by :	คุณ TL/SA
						Last update date / time :	17/3/2014 10:00

ภาพที่ 4.60 ตัวอย่างฟอร์มบันทึกข้อมูลโครงการแยกตามความรับผิดชอบโดย Team Leader/System Analyst ส่วนที่ 2





## ต้นแบบระบบการตรวจสอบการพัฒนาซอฟต์แวร์

ใช้ระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110

Project Status Tracking
Project Repository
Standard Information
Management
User: FirstName LastName [Role:.....]
Logout

### Project Repository

#### Project Information

ชื่อโครงการ :

ชื่อลูกค้า :

ระยะเวลาโครงการ :  -  สถานะโครงการ :

ข้อมูลโครงการโดยย่อ :

**งานเขียนที่งานดำเนินการ (Project Member):**

ลำดับชื่อ	ชื่อ	Project Role	Process ที่รับผิดชอบ
1	คุณ Project Manager	Project Manager	PM.1, PM.2, PM.3, PM.4 SI.1, SI.2, SI.3, SI.4, SI.5, SI.6
2	คุณ Team Leader/System Analyst	Team Leader/System Analyst	SI.1, SI.2, SI.3, SI.4, SI.5, SI.6
3	คุณ Senior Programmer	Senior Programmer	SI.4, SI.5
4	คุณ Programmer	Programmer	SI.4, SI.5
5	คุณ Tester	Tester	SI.4, SI.5
6	คุณ Admin.	Admin.	SI.5

**ISO/IEC 29110 Processes**

**Project Management Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์)**

PM.1: Project Planning (การวางแผนโครงการ)

PM.2: Project Plan Execution (การปฏิบัติตามแผนโครงการ)

PM.3: Project Assessment and Control (การประเมินและควบคุมโครงการ)

PM.4: Project Closure (การส่งมอบโครงการ)

**Software Implementation Process (กระบวนการพัฒนาผลิตภัณฑ์ซอฟต์แวร์)**

SI.1 Software Implementation Initiation (การเริ่มการพัฒนาดังกล่าว)

SI.2 Software Requirements Analysis (การวิเคราะห์ความต้องการซอฟต์แวร์)

Checklist	Tasks	Input Products	Output Products	หมายเหตุ	ผู้รับผิดชอบ	วันที่กำหนดเสร็จงาน	วันที่เสร็จงานจริง
✓	- จัดทำเอกสาร Software Requirements Specification ฉบับร่าง	- เอกสาร Project Plan เวอร์ชันล่าสุด <a href="#">PP-TMS-BPP-56-01A-050256.docx</a>	- เอกสาร Software Requirements Specification ฉบับร่าง <a href="#">SRS-TMS-BPP-56-01D-100356.docx</a>				
✓	- นำทวนรายละเอียดในเอกสาร Software Requirements Specification เปรียบเทียบกับรายละเอียดในเอกสาร Project Plan เวอร์ชันล่าสุด		- เอกสาร Software Requirements Specification ที่ผ่านการ Verify แล้ว <a href="#">SRS-TMS-BPP-56-01A-110356.docx</a>	เนื่องจากทางลูกค้าขอไประยะเวลาในการตรวจสอบเอกสาร SRS เพิ่มขึ้น 3 วันทำการจึงทำให้กำหนดการลงนามยืนยันเอกสาร SRS เสียออกไป	คุณ Team Leader/System Analyst	15/3/2014	17/3/2014
✓	- บันทึกสถานะเอกสาร Software Requirements Specification ผ่านการ Verify แล้ว		- เอกสาร VF & VD Results ที่บันทึกสถานะ SRS ผ่านการ Verify แล้ว <a href="#">VFD-TMS-BPP-56-02A-110356.xlsx</a>				
✓	- นำเสนอเอกสาร Software Requirements Specification เพื่อให้อุทิศำขบวนและลงนาม		- เอกสาร Software Requirements Specification ที่ลูกค้าลงนามแล้ว <a href="#">SRS-TMS-BPP-56-02A-170356.docx</a>				
✓	- บันทึกสถานะเอกสาร Software Requirements Specification ผ่านการ Validate แล้วไว้ในเอกสาร Verification & Validation Results (หลังจากลูกค้าลงนามแล้ว)		- เอกสาร VF & VD Results ที่บันทึกสถานะ SRS ผ่านการ Validate แล้ว <a href="#">VFD-TMS-BPP-56-03A-170356.xlsx</a>				

SI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้างสถาปัตยกรรมและรายละเอียดซอฟต์แวร์)

ภาพที่ 4.61 ตัวอย่างฟอร์มบันทึกข้อมูลโครงการแยกตามความรับผิดชอบโดย Programmer ส่วนที่ 1

SI.4: Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์)							
Checklist	Tasks	Input Products	Output Products	หมายเหตุ	ผู้รับผิดชอบ	วันที่กำหนดเสร็จงาน	วันที่เสร็จงานจริง
<input checked="" type="checkbox"/>	- มอบหมายงานพัฒนาซอฟต์แวร์ให้ Programmer ตามเอกสาร Project Plan เวอร์ชันล่าสุด	- เอกสาร Project Plan เวอร์ชันล่าสุด <a href="#">PP-TMS-BPP-56-01A-050256.docx</a>	- Software Components เวอร์ชันที่ผ่านการ Unit Test แล้ว		คุณ Team Leader/ System Analyst คุณ Programmer	14/2/2014	15/5/2014
<input checked="" type="checkbox"/>	- ยืนยันขอบเขตและกำหนดความเข้าใจในเอกสาร Software Design เวอร์ชัน Baseline	- เอกสาร Software Design เวอร์ชัน Baseline <a href="#">DD-TMS-BPP-56-01A-310356.docx</a>	- เอกสาร Traceability Record เวอร์ชันที่ได้นำมาใช้ในการพัฒนา Unit Test แล้ว				
<input checked="" type="checkbox"/>	- พัฒนาซอฟต์แวร์ Software Components ตามเอกสาร Software Design เวอร์ชัน Baseline และ เอกสาร Traceability Record เวอร์ชัน Baseline	- เอกสาร Traceability Record เวอร์ชัน Baseline <a href="#">TA-TMS-BPP-56-01A-310356.docx</a>	- Software Components เวอร์ชันที่ผ่านการตรวจและแล้ว				
<input checked="" type="checkbox"/>	- Unit Test Software Components และบันทึกผลการทดสอบในเอกสาร Traceability Record เวอร์ชันล่าสุด		- เอกสาร Traceability Record เวอร์ชันที่ผ่านการตรวจสอบสถานะ Unit Test แล้ว				
<input checked="" type="checkbox"/>	- ตรวจสอบและควบคุมผลการดำเนินงานของ Programmer ทั้งในส่วนของ การพัฒนา Software Components การ Unit Test ในเอกสาร Traceability Record ให้ถูกต้องและครบถ้วน		- เอกสาร Traceability Record เวอร์ชันที่ผ่านการตรวจสอบสถานะ Unit Test แล้ว <a href="#">TA-TMS-BPP-56-03A-1</a>				
SI.5: Software Integration and Tests (การทดสอบและการบูรณาการซอฟต์แวร์)							
SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ซอฟต์แวร์)							

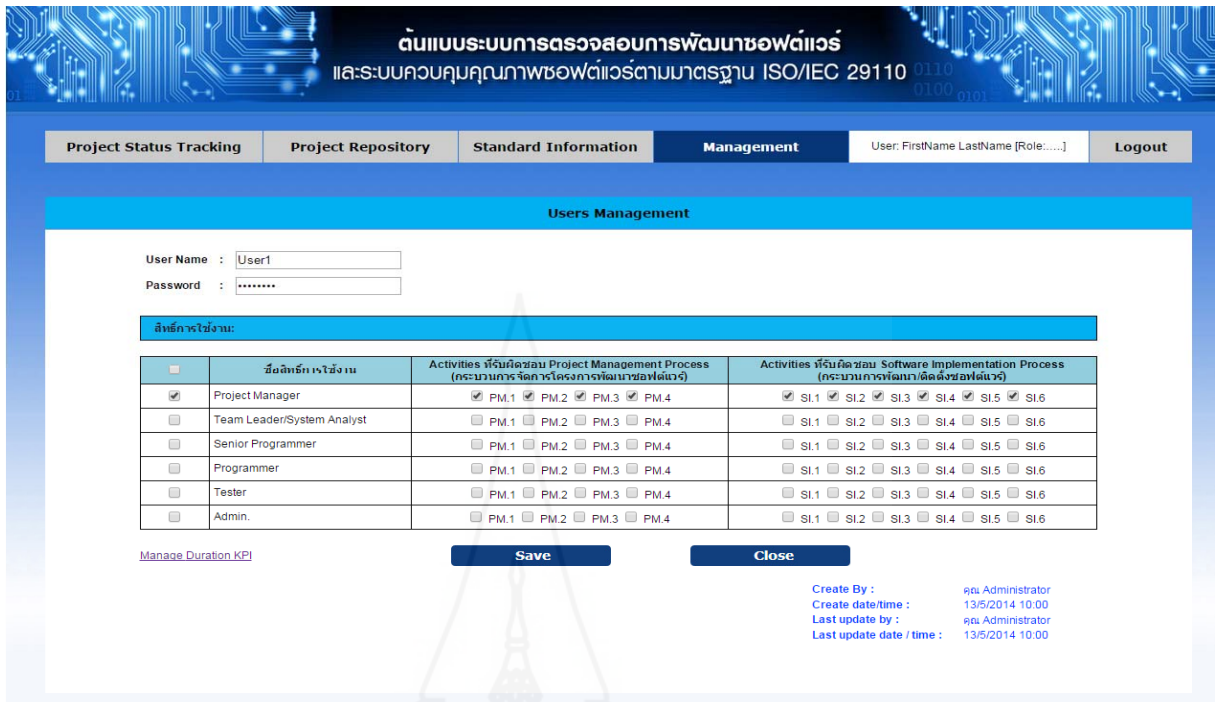
Save
Close

Create By : คุณ Programmer  
 Create datetime : 13/5/2014 10:00  
 Last update by : คุณ Programmer  
 Last update date / time : 13/5/2014 10:00

ภาพที่ 4.62 ตัวอย่างฟอร์มบันทึกข้อมูลโครงการแยกตามความรับผิดชอบโดย Programmer ส่วนที่ 2

### 3.4 การจัดการระบบโดยผู้ดูแลระบบ: ผู้ดูแลระบบสามารถกำหนดสิทธิ์ให้ผู้ใช้ระบบ

แต่ละบุคคลสามารถบันทึกข้อมูลเอกสาร, update สถานะโครงการ, และระบุวันที่ดำเนินงานเสร็จในแต่ละขั้นตอนการดำเนินงานที่รับผิดชอบได้ เพื่อเป็นการควบคุมการดำเนินงานให้เป็นไปตามบทบาทของทีมงานโครงการที่ได้วางแผนไว้ นอกจากนี้ผู้ดูแลระบบยังสามารถกำหนดค่าเกณฑ์ชี้วัดด้านระยะเวลาการดำเนินงานแต่ละกิจกรรมเพื่อเป็นมาตรฐานการดำเนินงานของทีมงานและระบบจะสามารถแสดงสัญลักษณ์สีเขียว เหลือง แดง เพื่อให้ทราบผลการดำเนินงานแต่ละกิจกรรมได้ทันทีอย่างสะดวกรวดเร็วเพื่อที่หาก Project Manager พบว่าการดำเนินงานกิจกรรมใดมีการแสดงสัญลักษณ์สีเขียว หรือสีแดง ก็จำเป็นต้องให้ความสนใจในการควบคุมและแก้ไขปัญหาให้คล่องโดยเร็วที่สุด ตัวอย่างโปรแกรมการจัดการระบบโดยผู้ดูแลระบบดังภาพที่ 4.63-4.64



ภาพที่ 4.63 ตัวอย่างโปรแกรมกำหนดสิทธิ์ผู้ใช้ระบบ โดยผู้ดูแลระบบ



ภาพที่ 4.64 ตัวอย่างโปรแกรมกำหนดเกณฑ์วัดด้านระยะเวลาการดำเนินงานแต่ละกิจกรรม โดยผู้ดูแลระบบ



## บทที่ 5

### สรุปผลและข้อเสนอแนะ

#### 1. การประเมินผล

1.1 ความคิดเห็นเกี่ยวกับแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยจากบริษัทตัวอย่างที่เป็นผู้ให้ข้อมูล

1.1.1 มีการแยกขั้นตอนการดำเนินงานสำหรับแต่ละ Activity (กิจกรรมการดำเนินงาน) ให้เห็นเป็นขั้นตอนทำให้เข้าใจได้ง่าย และสามารถใช้เป็น checklist สำหรับองค์กรที่มีความสนใจเริ่มปรับปรุงกระบวนการพัฒนาและควบคุมคุณภาพซอฟต์แวร์ให้เป็นไปตามมาตรฐาน ISO/IEC 29110 ได้ง่ายขึ้น

1.1.2 มีการแสดงรูปภาพตัวอย่างเอกสารที่เกี่ยวข้องกับกระบวนการดำเนินโครงการพัฒนาซอฟต์แวร์ให้เห็นอย่างชัดเจนและเข้าใจได้ง่าย

1.2 ความคิดเห็นเกี่ยวกับต้นแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile จากบริษัทตัวอย่างที่เป็นผู้ให้ข้อมูล

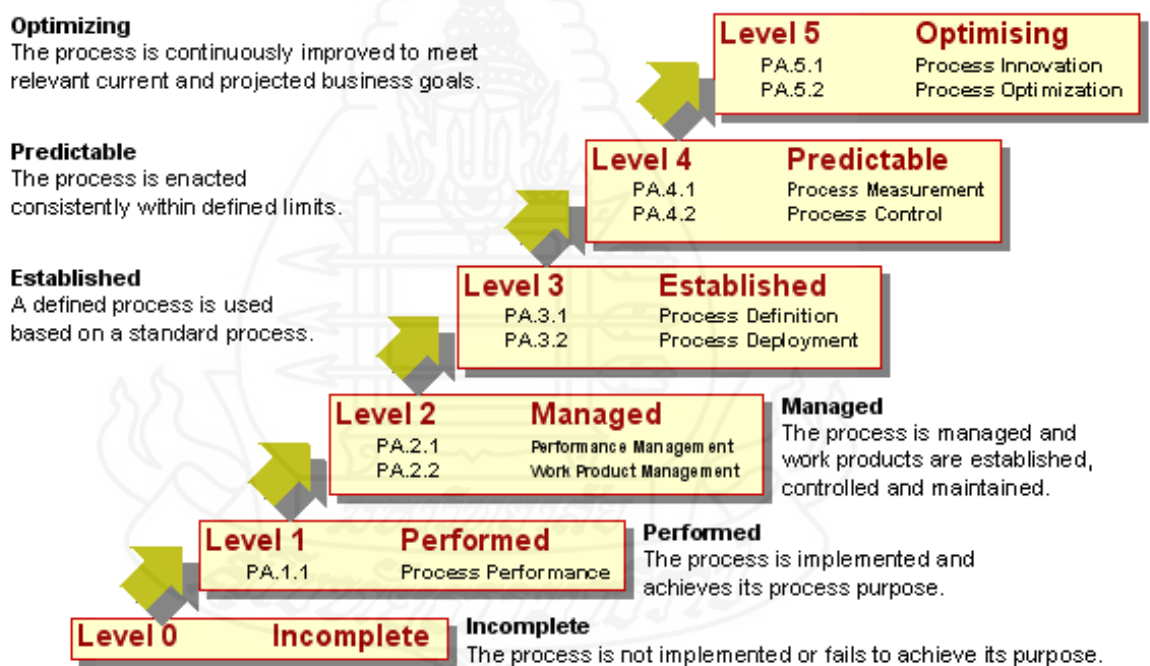
ในอนาคตอาจมีการพัฒนาต่อขอเพิ่มส่วนการแสดงผลข้อมูลโครงการในรูปแบบตาราง Timeline เพื่อให้เห็นภาพรวมระยะเวลาดำเนินโครงการได้ในอีกมุมมองหนึ่ง, เพิ่มส่วนสำหรับการเพิ่มรายการ Task ย่อยภายใต้แต่ละ Activity การดำเนินงาน, และเพิ่มการพัฒนาระบบให้รองรับการใช้งานผ่าน Mobile Devices

## 2. สรุปผลการดำเนินโครงการ

2.1 ผลที่ได้จากการศึกษารายละเอียดคุณลักษณะของมาตรฐาน ISO/IEC 29110 ด้านการพัฒนาซอฟต์แวร์สำหรับทีมพัฒนาซอฟต์แวร์ไม่เกิน 25 คน ระดับ Basic Profile มีดังนี้

2.1.1 มาตรฐาน ISO/IEC 29110 เป็นมาตรฐานด้านการพัฒนาซอฟต์แวร์สำหรับทีมพัฒนาซอฟต์แวร์ไม่เกิน 25 คน โดยแบ่งเป็น Capability Levels (ระดับความสามารถ) ในการดำเนินงานที่สามารถบรรลุวัตถุประสงค์ของแต่ละ Level ได้ ดังภาพที่ 5.1 สำหรับบทความนี้เป็นการศึกษาข้อมูลของระดับ Basic Profile ซึ่งเทียบเท่ากับ Level 1. Performed ตามมาตรฐาน ISO/IEC 15504ซึ่งใช้ในการประเมิน ISO/IEC 29110

### Capability Levels and Process Attributes



ภาพที่ 5.1 ISO/IEC 15504 Capability Levels

2.1.2 มาตรฐาน ISO/IEC 29110 ระดับ Basic Profile ประกอบด้วย Process (กระบวนการ) ที่สำคัญ 2 Process (กระบวนการ) ดังนี้

1) Project Management (PM) Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์)

2) Software Implementation (SI) Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์)

2.1.3 มาตรฐาน ISO/IEC 29110 ระดับ Basic Profile ประกอบด้วย Objectives (วัตถุประสงค์) ในการดำเนินโครงการและการดำเนินงานในแต่ละ Process (กระบวนการ) ที่องค์กรพัฒนาซอฟต์แวร์จะต้องสามารถทำให้บรรลุ Objectives (วัตถุประสงค์) เหล่านี้ได้อย่างครบถ้วน ดังนี้

1) Project Management (PM) Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์) ประกอบด้วย Objectives (วัตถุประสงค์) ในการดำเนินโครงการและการดำเนินงาน ดังนี้

- PM.O1: Project Plan (แผนโครงการ) สำหรับการดำเนินโครงการ ได้รับการจัดทำขึ้นอ้างอิงตามขอบข่ายงาน และได้รับการตรวจสอบจากลูกค้าแล้ว รายการงานและทรัพยากรต่าง ๆ ที่มีความจำเป็นต่อการดำเนินงานให้เสร็จสิ้นได้รับการประเมินไว้แล้ว

- PM.O2: Progress (ความคืบหน้าของการดำเนินโครงการ) ได้รับการติดตามอ้างอิง Project Plan (แผนโครงการ) และได้รับการบันทึกไว้ในเอกสาร Progress Status Record

- PM.O3: Change Request ได้รับการจัดเตรียมไว้สำหรับการวิเคราะห์ และรับไว้พิจารณาแก้ไข การเปลี่ยนแปลงความต้องการซอฟต์แวร์ได้รับการประเมินผลกระทบด้านค่าใช้จ่าย, ด้านกำหนดการดำเนินงาน, และด้านเทคนิค

- PM.O4: Review Meetings (การประชุมเพื่อทบทวน) การประชุมเพื่อทบทวนร่วมกับทีมงานและลูกค้าได้มีการจัดขึ้นข้อตกลงต่างๆ ได้รับการบันทึกไว้และติดตาม

- PM.O5: Risk are identified (ความเสี่ยงได้รับการระบุไว้) ระหว่างการจัดทำ Project Plan (แผนโครงการ) และระหว่างการจัดการโครงการ

- PM.O6: A software Version Control Strategy is developed. (กลยุทธ์การควบคุมเวอร์ชันของซอฟต์แวร์ได้รับการกำหนดขึ้น) รายการองค์ประกอบของซอฟต์แวร์ได้รับการระบุ, กำหนด, และวางพื้นฐานไว้ การแก้ไขและนำซอฟต์แวร์ออกติดตั้งได้รับการควบคุมและทำให้พร้อมใช้งานสำหรับทั้งลูกค้าและทีมงานทั้งทางด้านการจัดการสถานที่จัดเก็บ และการจัดการด้านการส่งมอบ

- PM.O7: Software Quality Assurance is performed. (การรับประกันคุณภาพซอฟต์แวร์ได้รับการดำเนินการ) เพื่อเป็นการรับประกันว่าผลิตภัณฑ์ซอฟต์แวร์และกระบวนการได้มีความสอดคล้องกับ Project Plan (แผนโครงการ) และข้อกำหนดเฉพาะของความต้องการซอฟต์แวร์ (Requirements Specification)

2) Software Implementation (SI) Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์) ประกอบด้วย Objectives (วัตถุประสงค์) ในการดำเนินโครงการและการดำเนินงาน ดังนี้

- SI.O1: Tasks of the activities are performed through the accomplishment of the current Project Plan. (รายการงานของกิจกรรมต่างๆ ได้รับการกระทำให้เป็นผลสำเร็จตามที่กำหนดไว้ใน Project Plan (แผนโครงการ))

- SI.O2: Software Requirements ความต้องการซอฟต์แวร์ได้รับการกำหนด, วิเคราะห์ ว่ามีความถูกต้องและสามารถทดสอบได้ โดยได้รับการอนุมัติจากทางลูกค้าเพื่อใช้เป็นพื้นฐานในการยอมรับและสื่อสารร่วมกัน

- SI.O3: Software Architectural and Detailed Design การออกแบบโครงสร้างทางสถาปัตยกรรมและรายละเอียดซอฟต์แวร์ได้รับการพัฒนาจัดทำขึ้นและกำหนดเป็นพื้นฐานในการยอมรับซึ่งใช้ในการอธิบายรายการองค์ประกอบของซอฟต์แวร์, การเชื่อมต่อกันระหว่างองค์ประกอบภายในของซอฟต์แวร์, และการเชื่อมต่อระหว่างระบบอื่นภายนอกซอฟต์แวร์ ความถูกต้องและการตรวจสอบย้อนกลับอ้างอิงตามความต้องการซอฟต์แวร์สามารถพิสูจน์ให้เห็นได้

- SI.O4: Software Components องค์ประกอบของซอฟต์แวร์ได้รับการกำหนดขึ้นในขั้นตอนการออกแบบได้รับการพัฒนาจัดทำขึ้น การทดสอบหน่วยย่อย (Unit Test) ได้รับการกำหนดและดำเนินการเพื่อยืนยันความถูกต้องในความสอดคล้องกับความต้องการซอฟต์แวร์และการออกแบบการตรวจสอบย้อนกลับไปยังความต้องการซอฟต์แวร์และการออกแบบสามารถพิสูจน์ให้เห็นได้

- SI.O5: Software is produced performing integration of software components and verified using Test Cases and Test Procedures. ซอฟต์แวร์ได้รับการผลิตโดยการบูรณาการองค์ประกอบของซอฟต์แวร์และได้รับการยืนยันความถูกต้องโดยการทดสอบกรณี (Test Case) และการทดสอบการดำเนินงาน (Test Procedure) ผลการทดสอบได้รับการบันทึกในเอกสารรายงานการทดสอบ (Test Report) ข้อผิดพลาดได้รับการแก้ไขให้ถูกต้อง, ความสอดคล้องกับความต้องการซอฟต์แวร์และการออกแบบ, และการตรวจสอบย้อนกลับไปยังความต้องการซอฟต์แวร์และการออกแบบได้รับการพิสูจน์ให้เห็นได้

- SI.O6: Software Configuration ซอฟต์แวร์ที่พัฒนาเสร็จแล้วซึ่งตรงตามข้อกำหนดเฉพาะของความต้องการซอฟต์แวร์ตามที่ได้รับการยืนยันจากลูกค้า ซึ่งประกอบด้วยเอกสารสำหรับผู้ใช้ระบบ, เอกสารด้านการปฏิบัติการและการบำรุงรักษาระบบ ได้รับการบูรณาการ, ได้รับการกำหนดเป็นพื้นฐานในการยอมรับ, และได้รับการจัดเก็บใน Project Repository (ที่เก็บข้อมูลโครงการ) แล้ว ในส่วนของความต้องการเปลี่ยนแปลงซอฟต์แวร์ที่เสร็จแล้วนั้นจะได้รับการตรวจสอบว่ามีขึ้นหรือไม่ และเอกสาร Change Request (การร้องขอเปลี่ยนแปลง) จะได้รับการริเริ่มจัดทำขึ้น

- SI.O7: Verification and Validation tasks of all required work products are performed งานด้านการตรวจสอบและยืนยันความถูกต้องของ Work Products (ผลิตภัณฑ์การทำงาน) ทั้งหมด ได้รับการดำเนินการตามเงื่อนไขที่กำหนดไว้ตามมาตรฐาน เพื่อให้บรรลุถึงความถูกต้องเหมาะสมระหว่าง Input Products (ผลิตภัณฑ์นำเข้า) และ Output Products (ผลิตภัณฑ์ที่ได้รับการผลิตออกมา) ของแต่ละกิจกรรม ข้อผิดพลาดได้รับการแสดงให้เห็น, ได้รับการแก้ไขให้ถูกต้อง, และได้รับการบันทึกไว้ในเอกสารผลการยืนยันความถูกต้อง (Verification Result) และเอกสารผลการตรวจสอบตามแนวทางแห่งข้อกำหนด (Validation Result)

2.1.3 มาตรฐาน ISO/IEC 29110 ระดับ Basic Profile ประกอบด้วย Activities (กิจกรรม) ในการดำเนินโครงการและการดำเนินงานในแต่ละ Process (กระบวนการ) ที่องค์กรพัฒนาซอฟต์แวร์ควรจะดำเนินการดำเนินงานได้ดังนี้

1) Project Management (PM) Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์) ประกอบด้วย Objectives (วัตถุประสงค์) ในการดำเนินโครงการและการดำเนินงาน ดังนี้

- PM.1: Project Planning (การวางแผนโครงการ)
- PM.2: Project Plan Execution (การปฏิบัติตามแผนโครงการ)
- PM.3: Project Assessment and Control (การประเมินและควบคุม

โครงการ)

- PM.4: Project Closure (การส่งมอบโครงการ)

2) Software Implementation (SI) Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์) ประกอบด้วย Activities (กิจกรรม) ในการดำเนินโครงการและการดำเนินงานในแต่ละ Process (กระบวนการ) ที่องค์กรพัฒนาซอฟต์แวร์ควรจะดำเนินการดำเนินงานได้ ดังนี้

- SI.1 Software Implementation Initiation (การเริ่มการพัฒนา/ติดตั้งซอฟต์แวร์)

- SI.2 Software Requirements Analysis (การวิเคราะห์ความต้องการซอฟต์แวร์)
- SI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้างสถาปัตยกรรมและรายละเอียดซอฟต์แวร์)
- SI.4: Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์)
- SI.5 Software Integration and Tests (การทดสอบและการบูรณาการซอฟต์แวร์)
- SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ซอฟต์แวร์)

## 2.2 ผลที่ได้จากการศึกษากระบวนการบริหารโครงการพัฒนาซอฟต์แวร์อย่างมีคุณภาพตามมาตรฐาน ISO/IEC 29110 จากบริษัทตัวอย่างที่ผ่านการรับรองมาตรฐาน ISO/IEC 29110 แล้วทำการวิเคราะห์และพัฒนาเป็นแนวปฏิบัติที่ดี มีดังนี้

การดำเนินโครงการในส่วนนี้เป็นการวิเคราะห์กระบวนการบริหารโครงการพัฒนาซอฟต์แวร์อย่างมีคุณภาพตามมาตรฐาน ISO/IEC 29110 จากบริษัทตัวอย่างที่ผ่านการรับรองมาตรฐาน ISO/IEC 29110 แล้ว คือบริษัท โปรเฟสชั่นแนล เซอร์วิส โพรไวเดอร์ จำกัด แล้วทำการพัฒนาเป็นแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile โดยทำการระบุเป็นขั้นตอนการดำเนินงานสำหรับแต่ละ Activity ในการทำงานที่สามารถบรรลุ Objectives ของมาตรฐาน ISO/IEC 29110 ได้โดยข้อมูลแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile เหมาะสำหรับผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยที่มีความสนใจในการปรับปรุงกระบวนการและคุณภาพการพัฒนาซอฟต์แวร์ให้มีประสิทธิภาพยิ่งขึ้น เพื่อเป็นข้อมูลช่วยให้สามารถทำความเข้าใจขั้นตอนการทำงานและรูปแบบการจัดทำเอกสารที่เหมาะสมประกอบการดำเนินโครงการพัฒนาซอฟต์แวร์ได้สะดวกและรวดเร็วยิ่งขึ้นและยังเป็นการยกระดับคุณภาพการพัฒนาซอฟต์แวร์ของไทยสู่ระดับสากลอีกด้วย

โดยบทบาทการทำงานที่สำคัญอย่างมากในทางที่จะสามารถประสบความสำเร็จในการดำเนินโครงการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ตามข้อมูลที่ได้รวบรวมจากบริษัทตัวอย่าง คือ Project Manager และ Team Leader/System Analyst โดย Project Manager จะเป็นผู้นำในการวางแผนโครงการ, มอบงานหมายงาน, ติดตาม/ควบคุมโครงการให้มีประสิทธิภาพด้านการควบคุมระยะเวลา, ควบคุมความเสี่ยงและปัญหาที่เกิดขึ้นระหว่างดำเนินโครงการ, และควบคุมค่าใช้จ่ายในการดำเนินโครงการ ส่วน Team Leader/System Analyst จะเป็นผู้นำในด้านการออกแบบระบบ, วางแผนงาน/มอบหมายงาน/และควบคุมงานการพัฒนาซอฟต์แวร์,



วางแผนงาน/มอบหมายงาน/และควบคุมงานการทดสอบซอฟต์แวร์, วางแผนงาน/มอบหมายงาน/และควบคุมงานการจัดทำคู่มือ User Manual และ System Manual, และเป็นผู้จัดเตรียมข้อมูลและเอกสารชุด Software Configuration สำหรับการส่งมอบงานปิดโครงการโดยประสานงานร่วมกับ Project Manager ในการส่งมอบข้อมูลและเอกสารชุด Software Configuration และนำจดหมายส่งมอบงานปิดโครงการ (Acceptance Record) ให้ลูกค้าลงนามรับมอบงานปิดโครงการได้อย่างสมบูรณ์ ดังนั้นบุคลากรขององค์กรที่รับผิดชอบใน 2 บทบาทหน้าที่นี้จึงควรเป็นผู้มีประสบการณ์และทักษะในการทำงานที่เหมาะสมจึงจะมีส่วนช่วยให้องค์กรสามารถประสบความสำเร็จในการดำเนินโครงการพัฒนาซอฟต์แวร์อย่างมีคุณภาพตามมาตรฐาน ISO/IEC 29110 ได้

การที่บริษัทตัวอย่างได้ปรับปรุงกระบวนการดำเนินการดำเนินโครงการพัฒนาซอฟต์แวร์ขององค์กรให้สามารถผ่านการรับรองมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile ได้นั้น ส่งผลให้องค์กรได้รับประโยชน์ในด้านต่างๆ ที่ดีขึ้นเพิ่มขึ้น ดังตารางที่ 5.1 GAP Analysis

ตารางที่ 5.1 GAP Analysis ก่อน-หลังการปรับปรุงกระบวนการดำเนินการดำเนินโครงการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ของบริษัทตัวอย่าง

Activities ตาม มาตรฐาน ISO/IEC 29110	ก่อนการปรับปรุง กระบวนการ		ประโยชน์ที่องค์กรได้รับ หลังการปรับปรุง กระบวนการ
		หลังการปรับปรุง กระบวนการ	
<b>Project Management Process (กระบวนการจัดการโครงการพัฒนาซอฟต์แวร์)</b>			
PM.1: Project Planning (การวางแผนโครงการ)	- ไม่มีการจัดทำเอกสาร Project Plan แต่มีการจัดทำเอกสารบางส่วนตามเนื้อหาในเอกสาร Project Plan	- จัดทำเอกสาร Project Plan ที่มีเนื้อหาครอบคลุมตาม Objectives ของมาตรฐาน	- การดำเนินโครงการพัฒนาซอฟต์แวร์มีการทำงานเป็นระบบ, มีเอกสารอ้างอิงในทุกขั้นตอนการทำงาน, และช่วยให้ลด
	- ไม่มีการจัดทำเอกสาร Verification & Validation Results	- จัดทำเอกสาร Verification & Validation Results เพื่อบันทึกสถานะการตรวจสอบเอกสารต่าง ๆ ที่จัดทำขึ้น	ระยะเวลาในการดำเนินโครงการ

## ตารางที่ 5.1 (ต่อ)

Activities ตาม มาตรฐาน ISO/IEC 29110	ก่อนการปรับปรุง กระบวนการ	หลังการปรับปรุง กระบวนการ	ประโยชน์ที่องค์กร ได้รับหลังการ ปรับปรุงกระบวนการ
PM.2: Project Plan Execution (การปฏิบัติ ตามแผนโครงการ)	- ไม่มีการจัดทำ เอกสาร Change Request - ไม่มีการจัดทำ เอกสาร Correction Register	- จัดทำเอกสาร Change Request เมื่อลูกค้ามีการ เปลี่ยนแปลง ขอข่ายงาน - จัดทำเอกสาร Correction Register เพื่อบันทึกปัญหา และสถานะการ แก้ไขปัญหา ระหว่างดำเนิน โครงการ	- การดำเนิน โครงการ พัฒนาซอฟต์แวร์มี การทำงานเป็น ระบบ, มี เอกสารอ้างอิงใน ทุกขั้นตอนการ ทำงาน, และช่วยให้ ลดระยะเวลาในการ ดำเนินโครงการ
PM.3: Project Assessment and Control (การประเมิน และควบคุมโครงการ)	- ไม่มีการจัดทำ เอกสาร Change Request - ไม่มีการจัดทำ เอกสาร Correction Register	- จัดทำเอกสาร Change Request เมื่อลูกค้ามีการ เปลี่ยนแปลง ขอข่ายงาน - จัดทำเอกสาร Correction Register เพื่อบันทึกปัญหา และสถานะการ แก้ไขปัญหา ระหว่างดำเนิน โครงการ	- การดำเนิน โครงการ พัฒนาซอฟต์แวร์มี การทำงานเป็น ระบบ, มี เอกสารอ้างอิงใน ทุกขั้นตอนการ ทำงาน, และช่วยให้ ลดระยะเวลาในการ ดำเนินโครงการ



ตารางที่ 5.1 (ต่อ)

Activities ตาม มาตรฐาน ISO/IEC 29110	ก่อนการปรับปรุง กระบวนการ	หลังการปรับปรุง กระบวนการ	ประโยชน์ที่องค์กร ได้รับหลังการ ปรับปรุงกระบวนการ
Software Implementation Process (กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์)			
SL.1 Software Implementation Initiation (การเริ่มการ พัฒนา/ติดตั้ง ซอฟต์แวร์)	- ไม่มีการจัดทำ เอกสาร Project Plan แต่มีการจัดทำ เอกสารบางส่วน ตามเนื้อหาใน เอกสาร Project Plan	- จัดทำเอกสาร Project Plan ที่มี เนื้อหาครอบคลุม ตาม Objectives ของมาตรฐาน	- การดำเนิน โครงการ พัฒนาซอฟต์แวร์มี การทำงานเป็น ระบบ, มี เอกสารอ้างอิงใน ทุกขั้นตอนการ ทำงาน, และช่วยให้ ลดระยะเวลาในการ ดำเนินโครงการ
SL.2 Software Requirements Analysis (การ วิเคราะห์ความต้องการ ซอฟต์แวร์)	- มีการจัดทำเอกสาร Software Requirements Specification อยู่ แล้ว แต่ไม่ได้มีการ นำรูปภาพ User Interface ใ้ ประกอบไว้	- จัดทำเอกสาร Software Requirements Specification โดย เพิ่มการนำรูปภาพ User Interface ใ้ ประกอบไว้	- ช่วยเพิ่ม ประสิทธิภาพใน การส่งมอบงานแต่ ละ Phase การ ทำงานได้เร็วขึ้น

ตารางที่ 5.1 (ต่อ)

Activities ตาม มาตรฐาน ISO/IEC 29110	ก่อนการปรับปรุง กระบวนการ	หลังการปรับปรุง กระบวนการ	ประโยชน์ที่องค์กร ได้รับหลังการ ปรับปรุงกระบวนการ
SI.3 Software Architectural and Detailed Design (การ ออกแบบโครงสร้าง สถาปัตยกรรมและ รายละเอียด ซอฟต์แวร์)	<ul style="list-style-type: none"> <li>- มีการจัดทำเอกสาร Software Design (เอกสารออกแบบ ระบบ)แต่ไม่มีการ จัดทำเอกสารย่อย ในส่วน Program Structure</li> <li>- ไม่มีการจัดทำ เอกสาร Traceability Record</li> <li>- มีการจัดทำเอกสาร ประกอบการ ทดสอบระบบแต่ ไม่มีรายละเอียด ตามคุณลักษณะ เหมือนเอกสาร Test Cases and Test Proceduresและ เอกสาร Test Report</li> <li>- ไม่มีการจัดทำ เอกสาร Verification &amp; Validation Results</li> </ul>	<ul style="list-style-type: none"> <li>- จัดทำเอกสาร Software Design (เอกสารออกแบบ ระบบ)โดยเพิ่มการ จัดทำเอกสารย่อย ในส่วน Program Structure</li> <li>- จัดทำเอกสาร Traceability Record และบันทึก รายละเอียดที่มี เนื้อหาครอบคลุม ตาม Objectives ของมาตรฐาน</li> <li>- จัดทำเอกสาร Test Cases and Test Procedures และ เอกสาร Test Report</li> <li>- บันทึกสถานะการ ตรวจสอบเอกสาร ต่าง ๆ ในเอกสาร Verification &amp; Validation Results</li> </ul>	<ul style="list-style-type: none"> <li>- การดำเนิน โครงการ พัฒนาซอฟต์แวร์มี การทำงานเป็น ระบบ, มี เอกสารอ้างอิงใน ทุกขั้นตอนการ ทำงาน, และช่วยให้ ลดระยะเวลาในการ ดำเนินโครงการ</li> <li>- ช่วยเพิ่ม ประสิทธิภาพใน การส่งมอบงาน แต่ละ Phase การ ทำงานได้เร็วขึ้น</li> <li>- เพิ่มคุณภาพของ ซอฟต์แวร์และลด ระยะเวลาในการ แก้ไขปัญหา ซอฟต์แวร์ เพราะมี การบันทึก สถานะการทดสอบ อย่างละเอียด สำหรับใช้ ประกอบการแก้ไข ซอฟต์แวร์ได้ตรง</li> </ul>

ตารางที่ 5.1 (ต่อ)

Activities ตาม มาตรฐาน ISO/IEC 29110	ก่อนการปรับปรุง กระบวนการ	หลังการปรับปรุง กระบวนการ	ประโยชน์ที่องค์กร ได้รับหลังการ ปรับปรุงกระบวนการ
			<ul style="list-style-type: none"> <li>- จุดและเร็วขึ้น</li> <li>สามารถมอบหมายงานหรือโอนงานให้พนักงานใหม่หรือพนักงานคนอื่น</li> <li>ดำเนินงานแทนพนักงานคนเก่าที่ลาออกได้อย่างสะดวก</li> <li>เนื่องจากการจัดทำเอกสารอ้างอิงไว้อย่างละเอียด</li> </ul>
SL4: Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์)	<ul style="list-style-type: none"> <li>- ไม่มีการจัดทำเอกสาร Project Plan แต่มีการจัดทำเอกสารบางส่วนตามเนื้อหาในเอกสาร Project Plan</li> <li>- มีการจัดทำเอกสาร Software Design (เอกสารออกแบบระบบ)แต่ไม่มีการจัดทำเอกสารย่อยในส่วน Program Structure</li> </ul>	<ul style="list-style-type: none"> <li>- จัดทำเอกสาร Project Plan ที่มีเนื้อหาครอบคลุมตาม Objectives ของมาตรฐาน</li> <li>- จัดทำเอกสาร Software Design (เอกสารออกแบบระบบ)โดยเพิ่มการจัดทำเอกสารย่อยในส่วน Program Structure</li> <li>- จัดทำเอกสาร Traceability Record</li> </ul>	<ul style="list-style-type: none"> <li>- การดำเนินโครงการพัฒนาซอฟต์แวร์มีการทำงานเป็นระบบ, มีเอกสารอ้างอิงในทุกขั้นตอนการทำงาน, และช่วยให้ลดระยะเวลาในการดำเนินโครงการ</li> <li>- ช่วยเพิ่มประสิทธิภาพในการส่งมอบงานแต่ละ Phase การทำงานได้เร็วขึ้น</li> </ul>

ตารางที่ 5.1 (ต่อ)

Activities ตาม มาตรฐาน ISO/IEC 29110	ก่อนการปรับปรุง กระบวนการ	หลังการปรับปรุง กระบวนการ	ประโยชน์ที่องค์กร ได้รับหลังการ ปรับปรุงกระบวนการ
	<ul style="list-style-type: none"> <li>- ไม่มีการจัดทำเอกสาร Traceability Record</li> <li>- มีการจัดทำเอกสารประกอบการทดสอบระบบแต่ไม่มีรายละเอียดตามคุณลักษณะเหมือนเอกสาร Test Cases and Test Procedures และเอกสาร Test Report</li> </ul>	<ul style="list-style-type: none"> <li>และบันทึกรายละเอียดที่มีเนื้อหาครอบคลุมตาม Objectives ของมาตรฐาน</li> <li>- จัดทำเอกสาร Test Cases and Test Procedures และเอกสาร Test Report ที่มีเนื้อหาครอบคลุมตาม Objectives ของมาตรฐาน</li> <li>บันทึกสถานะการตรวจสอบเอกสารต่าง ๆ ในเอกสาร Verification &amp; Validation Results</li> </ul>	<ul style="list-style-type: none"> <li>- เพิ่มคุณภาพของซอฟต์แวร์และลดระยะเวลาในการแก้ไขปัญหาซอฟต์แวร์ เพราะมีการบันทึกสถานะการทดสอบอย่างละเอียดสำหรับใช้ประกอบการแก้ไขซอฟต์แวร์ได้ตรงจุดและเร็วขึ้นสามารถมอบหมายงานหรือโอนงานให้พนักงานใหม่หรือพนักงานคนอื่นดำเนินงานแทนพนักงานคนเก่าที่ลาออกได้อย่างสะดวกเนื่องจากการจัดทำเอกสารอ้างอิงไว้อย่างละเอียด</li> </ul>

ตารางที่ 5.1 (ต่อ)

Activities ตาม มาตรฐาน ISO/IEC 29110	ก่อนการปรับปรุง กระบวนการ	หลังการปรับปรุง กระบวนการ	ประโยชน์ที่องค์กร ได้รับหลังการ ปรับปรุงกระบวนการ
SI.5 Software Integration and Tests (การทดสอบและการ บูรณาการซอฟต์แวร์)	<ul style="list-style-type: none"> <li>- ไม่มีการจัดทำเอกสาร Project Plan แต่มีการจัดทำเอกสารบางส่วนตามเนื้อหาในเอกสาร Project Plan</li> <li>- มีการจัดทำเอกสาร Software Design (เอกสารออกแบบระบบ)แต่ไม่มีการจัดทำเอกสารย่อยในส่วน Program Structure</li> <li>- ไม่มีการจัดทำเอกสาร Traceability Record</li> <li>- มีการจัดทำเอกสารประกอบการ User Acceptance Test ของลูกค้า แต่ไม่มีรายละเอียดตามคุณลักษณะเหมือนเอกสาร Test Cases and Test</li> </ul>	<ul style="list-style-type: none"> <li>- จัดทำเอกสาร Project Plan ที่มีเนื้อหาครอบคลุมตาม Objectives ของมาตรฐาน</li> <li>- จัดทำเอกสาร Software Design (เอกสารออกแบบระบบ)โดยเพิ่มการจัดทำเอกสารย่อยในส่วน Program Structure</li> <li>- จัดทำเอกสาร Traceability Record และบันทึก รายละเอียดที่มีเนื้อหาครอบคลุมตาม Objectives ของมาตรฐาน</li> <li>- จัดทำเอกสาร Test Cases and Test Procedures และเอกสาร Test Report ที่มีเนื้อหาครอบคลุมตาม</li> </ul>	<ul style="list-style-type: none"> <li>- การดำเนิน โครงการพัฒนาซอฟต์แวร์มีการทำงานเป็นระบบ, มีเอกสารอ้างอิงในทุกขั้นตอนการทำงาน, และช่วยให้ลดระยะเวลาในการดำเนินโครงการ</li> <li>- ช่วยเพิ่มประสิทธิภาพในการส่งมอบงานแต่ละ Phase การทำงานได้เร็วขึ้น</li> <li>- เพิ่มคุณภาพของซอฟต์แวร์และลดระยะเวลาในการแก้ไขปัญหาซอฟต์แวร์ เพราะมีการบันทึกสถานะการทดสอบอย่างละเอียดสำหรับใช้ประกอบการแก้ไขซอฟต์แวร์ได้ตรง</li> </ul>

ตารางที่ 5.1 (ต่อ)

Activities ตาม มาตรฐาน ISO/IEC 29110	ก่อนการปรับปรุง กระบวนการ	หลังการปรับปรุง กระบวนการ	ประโยชน์ที่องค์กร ได้รับหลังการ ปรับปรุงกระบวนการ
	Procedures และ เอกสาร Test Report	Objectives ของ มาตรฐานสำหรับ การ User Acceptance Test ของลูกค้า บันทึกสถานะการ ตรวจสอบเอกสาร ต่าง ๆ ในเอกสาร Verification & Validation Results	จุดและเร็วขึ้น - สามารถ มอบหมายงาน หรือโอนงานให้ พนักงานใหม่หรือ พนักงานคนอื่น ดำเนินงานแทน พนักงานคนเก่าที่ ลาออกได้อย่าง สะดวกเนื่องจากมี การจัดทำ เอกสารอ้างอิงไว้ อย่างละเอียด
SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ ซอฟต์แวร์)	- ไม่มีการจัดทำ เอกสาร Project Plan แต่มีการจัดทำ เอกสารบางส่วน ตามเนื้อหาใน เอกสาร Project Plan - ไม่มีการจัดเตรียม ข้อมูล Software Configuration สำหรับการส่งมอบ ปิดโครงการรวมไว้	- จัดทำเอกสาร Project Plan ที่มี เนื้อหาครอบคลุม ตาม Objectives ของมาตรฐาน - จัดเตรียมข้อมูล Software Configuration สำหรับการส่งมอบ ปิดโครงการรวมไว้ เป็นชุดเดียวกัน	- การดำเนิน โครงการ พัฒนาซอฟต์แวร์มี การทำงานเป็น ระบบ, มี เอกสารอ้างอิงใน ทุกขั้นตอนการ ทำงาน, และช่วยให้ ลดระยะเวลาในการ ดำเนินโครงการ - ช่วยเพิ่ม ประสิทธิภาพใน การส่งมอบงาน

ตารางที่ 5.1 (ต่อ)

Activities ตาม มาตรฐาน ISO/IEC 29110	ก่อนการปรับปรุง กระบวนการ	หลังการปรับปรุง กระบวนการ	ประโยชน์ที่องค์กร ได้รับหลังการ ปรับปรุงกระบวนการ
	เป็นชุดเดียวกัน - มีการจัดทำ จดหมายส่งมอบ งานปิดโครงการ (Acceptance Record) สำหรับ การลงนาม รัับ มอบการปิด โครงการของ ลูกค้าอยู่แล้ว		แต่ละ Phase การ ทำงานได้เร็วขึ้น
<b>การปรับปรุงกระบวนการดำเนินโครงการพัฒนาซอฟต์แวร์ขององค์กรให้ผ่านการรับรอง มาตรฐาน ISO/IEC 29110</b>			
PM.1-4 SI.1-6	องค์กรมีขั้นตอนการ ดำเนินโครงการพัฒนา ซอฟต์แวร์ที่เป็นระบบ อยู่แล้ว แต่ยังไม่บรรลุ Objectives ของ มาตรฐาน ISO/IEC 29110 อย่างครบถ้วน	องค์กรปรับปรุง ขั้นตอนการดำเนิน โครงการพัฒนา ซอฟต์แวร์เพิ่มเติม ให้ สามารถบรรลุ Objectives ของ มาตรฐาน ISO/IEC 29110 อย่างครบถ้วน	- ได้รับความเชื่อถือ ในด้าน ความสามารถและ คุณภาพในการ ดำเนินโครงการ พัฒนาซอฟต์แวร์ จากลูกค้าที่มีอยู่เดิม เพิ่มมากขึ้น - เพิ่มโอกาสในการ เพิ่มลูกค้าใหม่จาก ความน่าเชื่อถือใน ใบรับรองมาตรฐาน ISO/IEC 29110



ตารางที่ 5.1 (ต่อ)

Activities ตาม มาตรฐาน ISO/IEC 29110	ก่อนการปรับปรุง กระบวนการ	หลังการปรับปรุง กระบวนการ	ประโยชน์ที่องค์กร ได้รับหลังการ ปรับปรุงกระบวนการ
			- สามารถเพิ่มอัตรา ค่าบริการจากการ ที่มีใบรับรอง มาตรฐาน ISO/IEC 29110 ส่งผลให้รายได้ ขององค์กร เพิ่มขึ้น

**2.3 ผลที่ได้จากการรวบรวมข้อมูลความต้องการระบบสำหรับใช้เป็นเครื่องมือในการตรวจสอบการพัฒนาซอฟต์แวร์และควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile จากบริษัทตัวอย่าง มีดังนี้**

ข้อมูลความต้องการคุณลักษณะของระบบที่จะสามารถนำมาใช้ประกอบการตรวจสอบและควบคุมคุณภาพโครงการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 นั้น ก็ได้รับการนำมาวิเคราะห์และออกแบบเป็นต้นแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile สำหรับผู้ที่มีความสนใจจะสามารถใช้ต้นแบบชุดนี้เป็นข้อมูลประกอบการพัฒนาเป็นระบบต่อไปในอนาคตได้อย่างรวดเร็วยิ่งขึ้น

### 3. ข้อเสนอแนะ

ในการที่หน่วยงานภาครัฐอย่างเช่น ชีป้า (สำนักงานส่งเสริมอุตสาหกรรมซอฟต์แวร์แห่งชาติ (องค์การมหาชน)) และสภาอุตสาหกรรมแห่งประเทศไทยได้ให้การสนับสนุนและผลักดันผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยในการปรับปรุงกระบวนการบริหารโครงการพัฒนาซอฟต์แวร์และควบคุมคุณภาพซอฟต์แวร์ให้มีคุณภาพตามมาตรฐาน ISO/IEC 29110 นั้น ถือ

เป็นการช่วยยกระดับความน่าเชื่อถือของซอฟต์แวร์ไทยออกสู่ระดับสากล ดังนั้นผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยเองก็ควรหันมาให้ความสำคัญในการที่จะปรับปรุงกระบวนการบริหารโครงการพัฒนาซอฟต์แวร์และควบคุมคุณภาพซอฟต์แวร์ของตนเองให้มีคุณภาพตามมาตรฐาน ISO/IEC 29110 เช่นกัน เพราะนอกจากจะได้รับประโยชน์ในด้านการเพิ่มความน่าเชื่อถือขององค์กรสู่ระดับสากลแล้ว ยังเป็นการช่วยลดต้นทุนทั้งในด้านระยะเวลาและค่าใช้จ่ายในการดำเนินโครงการ และยังสามารถเพิ่มรายได้ให้กับองค์กรได้อีกทางหนึ่งด้วย

ผู้ดำเนินงานมีความมั่นใจว่าแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ที่ได้พัฒนาขึ้นในรายงานนี้ จะสามารถมีส่วนช่วยให้ผู้ให้บริการพัฒนาซอฟต์แวร์ SME ของไทยผ่านการประเมินและได้รับการรับรองมาตรฐาน ISO/IEC 29110 ได้ เนื่องจากผู้ดำเนินงานเป็นส่วนหนึ่งขององค์กรตัวอย่างที่ร่วมดำเนินงานในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตั้งแต่องค์กรตัวอย่างได้รับการประเมินและผ่านการรับรองมาตรฐาน ISO/IEC 29110 ครั้งแรกในปี พ.ศ. 2555 จนกระทั่งองค์กรตัวอย่างได้รับการประเมินและผ่านการรับรองการต่ออายุมาตรฐาน ISO/IEC 29110 ในปี 2558 จึงสามารถพัฒนาแนวปฏิบัติที่ดีตามมาตรฐาน ISO/IEC 29110 ในรูปแบบ checklist ได้อย่างมีประสิทธิภาพ

สำหรับผู้ที่มีความสนใจในการพัฒนาเครื่องมือสำหรับใช้ในการตรวจสอบและควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 สามารถนำข้อมูลต้นแบบของระบบการตรวจสอบการพัฒนาซอฟต์แวร์และระบบควบคุมคุณภาพซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ระดับ Basic Profile มาใช้เป็นตัวอย่าง User Interface ประกอบการพัฒนาซอฟต์แวร์ได้เร็วขึ้น โดยเพิ่มการออกแบบฐานข้อมูลและการออกแบบ Program Structure ในส่วนของการ coding เท่านั้น

บรรณานุกรม



## บรรณานุกรม

- ความหมายและความเป็นมา ISO/IEC 29110 สืบค้นจาก [http://en.wikipedia.org/wiki/ISO\\_29110](http://en.wikipedia.org/wiki/ISO_29110)
- Alena Buchalceva. (2009). Software Development using ISO/IEC 29110 TR – Engineering and Management Guide.
- Czech Republic: Member of ISO/IEC JTC 1 SC7 - Working Group 24Prague University of Economics, Czech Republic.
- Deepak Jain. (11 Jan 2007). Importance of Processes and Standards in Software Development สืบค้นจาก <http://www.codeproject.com/Articles/17121/Importance-of-Processes-and-Standards-in-Software>
- SIPA.(2554). ไอซีที ผนึก 4 หน่วยงาน พัฒนามาตรฐานวิศวกรรมซอฟต์แวร์ไทย. สืบค้นจาก [http://www.sipa.or.th/ewt\\_news.php?nid=882](http://www.sipa.or.th/ewt_news.php?nid=882)





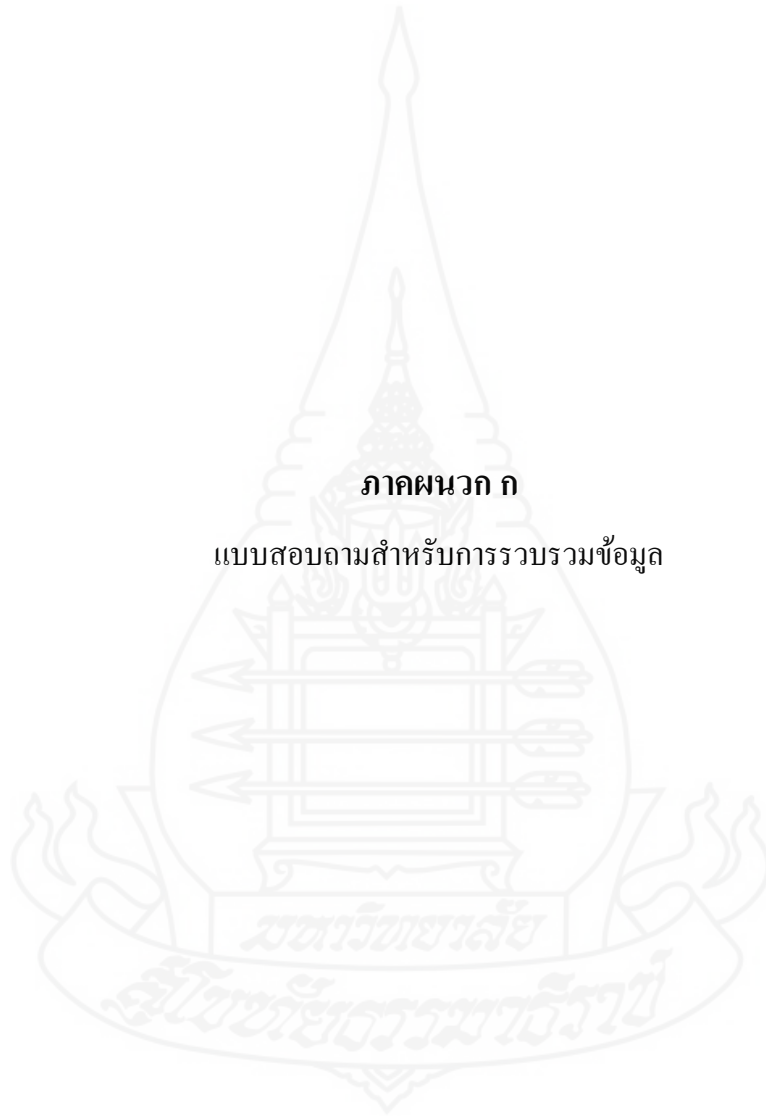
ภาคผนวก

มหาวิทยาลัยราชภัฏสกลนคร

สืบราชสันตติวงศ์

ภาคผนวก ก

แบบสอบถามสำหรับการรวบรวมข้อมูล



แบบสอบถามสำหรับการรวบรวมข้อมูล

แบบสอบถามผู้ประกอบการพัฒนาซอฟต์แวร์ที่ผ่านการประเมินมาตรฐานISO/IEC  
29110 ระดับ Basic Profile

ส่วนที่ 1: ข้อมูลทั่วไป

1.1 ชื่อสถานประกอบการ

.....  
.....  
.....  
.....

ที่อยู่

.....  
.....  
.....  
.....

1.2 ประวัติบริษัทฯ (ปีที่ก่อตั้ง, ทุนจดทะเบียน)

.....  
.....  
.....  
.....  
.....  
.....



1.3 โครงสร้างบริษัทฯ (การจัดโครงสร้างการบริหาร, จำนวนพนักงาน)

.....

.....

.....

.....

.....

.....

.....

1.4 Development Methodology ที่ใช้ในการพัฒนาซอฟต์แวร์

.....

.....

.....

.....

.....

.....

.....

1.5 ปัจจัยที่ทำให้องค์กรตัดสินใจปรับปรุงกระบวนการพัฒนาซอฟต์แวร์โดยเลือกใช้มาตรฐาน ISO/IEC 29110

.....

.....

.....

.....

.....

.....

.....

1.6 ระยะเวลาในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ขององค์กรตามมาตรฐาน ISO/IEC 29110 จนกระทั่งผ่านการรับรอง

.....

.....

.....

.....

.....

.....

.....

1.7 ระดับมาตรฐาน ISO/IEC 29110 ที่บริษัทผ่านการรับรอง

.....

.....

.....

.....

.....

.....

.....

1.8 หน่วยงาน/บริษัทที่ตรวจประเมิน และให้การรับรองมาตรฐาน ISO/IEC 29110

.....

.....

.....

.....

.....

.....

.....

1.9 ปัญหาที่พบระหว่างการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ขององค์กร

.....

.....

.....

.....

.....

.....

1.10 ปัจจัยที่มีผลต่อความสำเร็จในการปรับปรุงกระบวนการพัฒนาซอฟต์แวร์ตามมาตรฐาน ISO/IEC 29110 ขององค์กร

.....

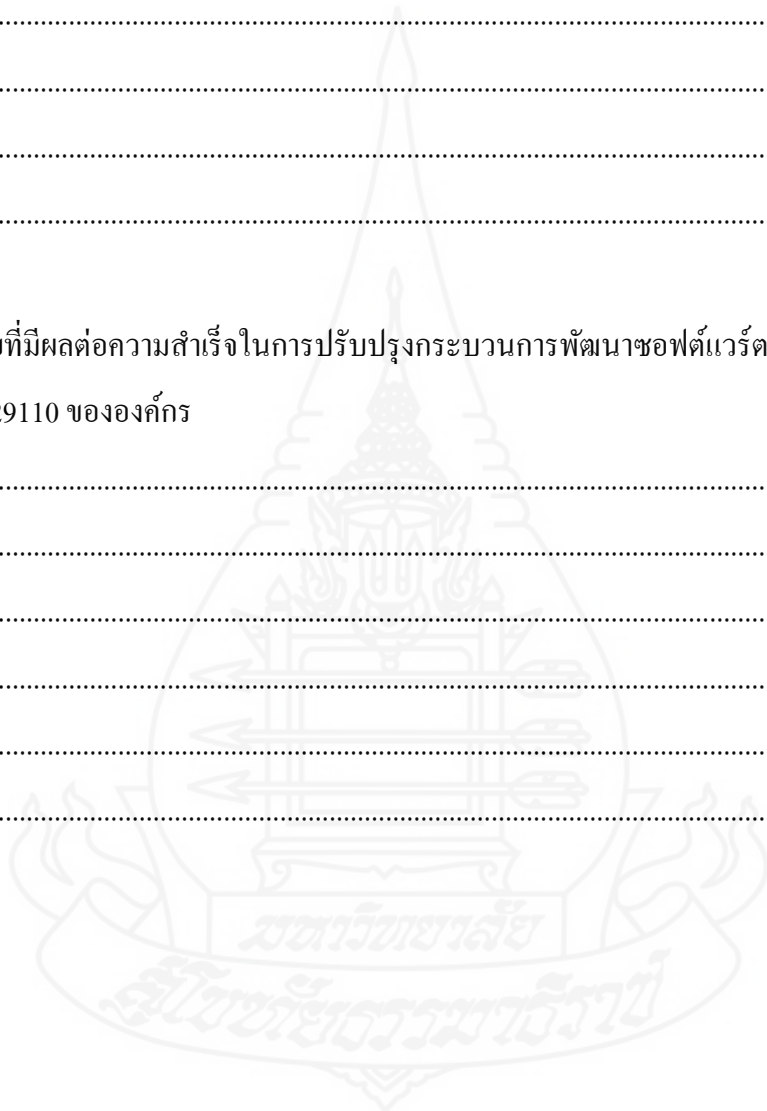
.....

.....

.....

.....

.....



**ส่วนที่ 2: ข้อมูลกระบวนการจัดการโครงการพัฒนาซอฟต์แวร์ขององค์กรตามมาตรฐาน ISO/IEC 29110**

**2.1 Activities การดำเนินงานและเอกสารที่เกี่ยวข้องขององค์กร ที่สามารถบรรลุวัตถุประสงค์ของกระบวนการจัดการโครงการพัฒนาซอฟต์แวร์ (Project Management Process) ตามมาตรฐาน ISO/IEC 29110**

**2.1.1 PM.1: Project Planning (การวางแผนโครงการ)**

ผู้รับผิดชอบในโครงการมีบทบาทใดบ้าง แต่ละบทบาทมีความเกี่ยวข้องกับกระบวนการนี้อย่างไร

.....

.....

.....

.....

.....

เอกสารที่เกี่ยวข้องกับกระบวนการนี้มีอะไรบ้าง

.....

.....

.....

.....

.....

รายละเอียดเพิ่มเติม

.....

.....

.....

.....

.....

**2.1.2PM.2: Project Plan Execution (การปฏิบัติตามแผนโครงการ)**

ผู้รับผิดชอบในโครงการมีบทบาทใดบ้าง แต่ละบทบาทมีความเกี่ยวข้องกับกระบวนการนี้อย่างไร

.....

.....

.....

.....

.....

เอกสารที่เกี่ยวข้องกับกระบวนการนี้มีอะไรบ้าง

.....

.....

.....

.....

.....

รายละเอียดเพิ่มเติม

.....

.....

.....

.....

.....

**2.1.3 PM.3: Project Assessment and Control (การประเมินและควบคุมโครงการ)**

ผู้รับผิดชอบในโครงการมีบทบาทใดบ้าง แต่ละบทบาทมีความเกี่ยวข้องกับกระบวนการนี้อย่างไร

.....

.....

.....

.....

เอกสารที่เกี่ยวข้องกับกระบวนการนี้มีอะไรบ้าง

.....

.....

.....

.....

.....

รายละเอียดเพิ่มเติม

.....

.....

.....

.....

.....

**2.1.4 PM.4: Project Closure (การส่งมอบโครงการ)**

ผู้รับผิดชอบในโครงการมีบทบาทใดบ้าง แต่ละบทบาทมีความเกี่ยวข้องกับกระบวนการนี้อย่างไร

.....

.....

.....

.....

.....

เอกสารที่เกี่ยวข้องกับกระบวนการนี้มีอะไรบ้าง

.....

.....

.....

.....

รายละเอียดเพิ่มเติม

.....  
.....  
.....  
.....  
.....

**2.2 Activities การดำเนินงานและเอกสารที่เกี่ยวข้องขององค์กร ที่บรรลุวัตถุประสงค์ของ  
กระบวนการพัฒนา/ติดตั้งซอฟต์แวร์ (Software Implementation Process) ตามมาตรฐาน  
ISO/IEC 29110**

**2.2.1 SI.1 Software Implementation Initiation (การเริ่มการพัฒนา/ติดตั้งซอฟต์แวร์)**

ผู้รับผิดชอบในโครงการมีบทบาทใดบ้าง แต่ละบทบาทมีความเกี่ยวข้องกับ  
กระบวนการนี้อย่างไร

.....  
.....  
.....  
.....  
.....

เอกสารที่เกี่ยวข้องกับกระบวนการนี้มีอะไรบ้าง

.....  
.....  
.....  
.....  
.....

รายละเอียดเพิ่มเติม

.....

.....

.....

.....

.....

**2.2.2SI.2 Software Requirements Analysis (การวิเคราะห์ความต้องการซอฟต์แวร์)**

ผู้รับผิดชอบในโครงการมีบทบาทใดบ้าง แต่ละบทบาทมีความเกี่ยวข้องกับกระบวนการนี้อย่างไร

.....

.....

.....

.....

.....

เอกสารที่เกี่ยวข้องกับกระบวนการนี้มีอะไรบ้าง

.....

.....

.....

.....

.....

รายละเอียดเพิ่มเติม

.....

.....

.....

.....



**2.2.3SI.3 Software Architectural and Detailed Design (การออกแบบโครงสร้าง**

**สถาปัตยกรรมและรายละเอียดซอฟต์แวร์)**

ผู้รับผิดชอบในโครงการมีบทบาทใดบ้าง แต่ละบทบาทมีความเกี่ยวข้องกับ  
กระบวนการนี้อย่างไร

.....

.....

.....

.....

.....

เอกสารที่เกี่ยวข้องกับกระบวนการนี้มีอะไรบ้าง

.....

.....

.....

.....

.....

รายละเอียดเพิ่มเติม

.....

.....

.....

.....

.....

**2.2.4SI.4: Software Construction (การก่อสร้างหรือพัฒนาซอฟต์แวร์)**

ผู้รับผิดชอบในโครงการมีบทบาทใดบ้าง แต่ละบทบาทมีความเกี่ยวข้องกับ  
กระบวนการนี้อย่างไร

.....

.....

.....

เอกสารที่เกี่ยวข้องกับกระบวนการนี้มีอะไรบ้าง

.....

.....

.....

.....

.....

รายละเอียดเพิ่มเติม

.....

.....

.....

.....

.....

**2.2.5SI.5: Software Integration and Tests(การทดสอบและการบูรณาการซอฟต์แวร์)**

ผู้รับผิดชอบในโครงการมีบทบาทใดบ้าง แต่ละบทบาทมีความเกี่ยวข้องกับกระบวนการนี้อย่างไร

.....

.....

.....

.....

.....

เอกสารที่เกี่ยวข้องกับกระบวนการนี้มีอะไรบ้าง

.....

.....

.....

.....

รายละเอียดเพิ่มเติม

.....

.....

.....

.....

.....

**2.2.6 SI.6 Product Delivery (การส่งมอบผลิตภัณฑ์ซอฟต์แวร์) .....**

ผู้รับผิดชอบในโครงการมีบทบาทใดบ้าง แต่ละบทบาทมีความเกี่ยวข้องกับกระบวนการนี้อย่างไร

.....

.....

.....

.....

.....

เอกสารที่เกี่ยวข้องกับกระบวนการนี้มีอะไรบ้าง

.....

.....

.....

.....

.....

รายละเอียดเพิ่มเติม

.....

.....

.....

.....





ภาคผนวก ข

แบบสอบถามสำหรับประเมินผล







ภาคผนวก ค

เกียรติบัตรการนำเสนอผลงานทางวิชาการ



## เกียรติบัตรการนำเสนอผลงานทางวิชาการ



## ประวัติผู้ศึกษา

ชื่อ	นางสาวนิตยา สิงไทยสงค์
วัน เดือน ปีเกิด	30 ตุลาคม 2515
สถานที่เกิด	จังหวัดนครราชสีมา
ประวัติการศึกษา	บริหารธุรกิจบัณฑิต (การจัดการทั่วไป) มหาวิทยาลัยสุโขทัยธรรมาราช
สถานที่ทำงาน	บริษัท โพรเฟสชั่นแนล เซอร์วิส โพรไวเดอร์ จำกัด จังหวัดกรุงเทพมหานคร
ตำแหน่ง	Director, Software Solution

