

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

เครื่องมือบริหารจัดการโครงการพัฒนาซอฟต์แวร์

SOFTWARE PROJECT MANAGEMENT TOOLS



H001207

โดย

สุรเชษฐ์ เกลิมผล

SURACHET CHALERMPHOL

อาจารย์ที่ปรึกษา

รศ.ดร.เอื้อน ปิ่นเงิน

วัน เดือน ปี..... 04 ส.ค. 2550.....

เลขทะเบียน..... H001207.....

เลขเรียกหนังสือ..... อท. ล 848 ค 254ค.....

"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."

๖1๘๗๙๗๖

11๗14๐๗๕๘

รายงานนี้เป็นส่วนหนึ่งของ วิชาโครงการศึกษากรณีพิเศษ
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปะเสียงเนื้อหา และต้องขออนุญาตเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคเรียนที่ 2 ปีการศึกษา 2549

SOFTWARE PROJECT MANAGEMENT TOOLS



A SPECIAL STUDY PROJECT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY
FACULTY OF INFORMATION TECNOLOGY
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2/ 2006



COPYRIGHT 2007

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาเท่านั้น ห้ามเผยแพร่หรือใช้เพื่อการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	เครื่องมือบริหารจัดการ โครงการพัฒนาซอฟต์แวร์
นักศึกษา	ศุรเชษฐ์ เกลิมผล
รหัสนักศึกษา	48066639
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขา	วิชาเทคโนโลยีสารสนเทศ
แขนงวิชา	การจัดการเทคโนโลยีสารสนเทศ
ปีการศึกษา	2549
อาจารย์ที่ปรึกษา	รศ.ดร.เอื้อน ปิ่นเงิน

บทคัดย่อ

โครงการพัฒนาซอฟต์แวร์เป็นหนึ่งในโครงการพัฒนาที่มีความซับซ้อนและมีการบริหารโครงการที่เป็นลักษณะเฉพาะตัว มีปัจจัยความสำเร็จและตัวแปรต่างๆในการบริหารงาน โครงการที่แตกต่างจากโครงการพัฒนาอื่นๆ ทำให้การบริหารจัดการโครงการพัฒนาซอฟต์แวร์จำเป็นต้องมีการกำหนดขั้นตอนของการพัฒนาให้สอดคล้องกับลักษณะของโครงการ อีกทั้งยังต้องการเครื่องมือที่ช่วยในการบริหารงานในแต่ละขั้นตอนเพื่อให้สามารถบริหารโครงการได้อย่างมีประสิทธิภาพ

การศึกษาและจัดทำเครื่องมือบริหารจัดการโครงการพัฒนาซอฟต์แวร์นี้มีวัตถุประสงค์เพื่อสร้างเครื่องมือที่มีประสิทธิภาพในการบริหารงานโครงการและลดความผิดพลาดในการบริหารงานโครงการ โดยได้ทำการศึกษาทฤษฎีการบริหารโครงการเสตปไวส์ (Step Wise) และทฤษฎีอื่นๆที่เกี่ยวข้อง เพื่อทำการออกแบบโครงสร้างขั้นตอนการบริหารงานโครงการพัฒนาซอฟต์แวร์และออกแบบเครื่องมือการบริหารงานโครงการพัฒนาซอฟต์แวร์ตามการวิเคราะห์ระบบ

จากการศึกษาพบว่า โครงการพัฒนาซอฟต์แวร์จำเป็นต้องใช้ระบบสารสนเทศที่มีประสิทธิภาพเข้ามาช่วยในการบริหารจัดการโครงการ เครื่องมือบริหารจัดการโครงการพัฒนาซอฟต์แวร์ที่จัดทำขึ้นนี้เป็นเครื่องมือที่ถูกพัฒนาขึ้นภายใต้การทำงานแบบเว็บแอปพลิเคชัน โดยซอฟต์แวร์นี้ต้องสามารถสร้างและจัดเก็บข้อมูลของโครงการในทุกขั้นตอนรวมถึงต้องสามารถตรวจสอบสถานะของโครงการ ในระหว่างการดำเนินโครงการเพื่อเป็นข้อมูลในการประเมินโครงการและแก้ไขปัญหาต่างๆที่เกิดขึ้นกับโครงการได้อย่างมีประสิทธิภาพ

Title	Software Project Management Tools
Student	Mr. Surachet Chalermphol
Student ID.	48066639
Degree	Master of Science
Program	Information Technology Management
Academic Year	2006
Advisor	Assoc.Prof.Dr. Ouen Pimngern

ABSTRACT

Software project management is a complicated project and unique among general project management. It is also controlled by many factors which are different from other project managements. At this stage, the software project management is necessary to draw the process by considering the project type and then to assess the conditions that might be required for particular instruments to be effective component of management.

The study and design of the software project management tools aims to fabricate a productive implement for project management and overcome the problem of operation by analyzing Step Wise theory and related papers. In order to create the structure and the software project management tools follows the way of system analysis.

Consistent with a number of studies, it suggests that information system plays an important role in the software project management. This software project management tools was improved from Web application to decide on producing and collecting data through the process. In addition, it is important to examine the project status during project performing, evaluate the situation and solve the problem to achieve the efficient management.

กิตติกรรมประกาศ

ความสำเร็จของการพัฒนาการ โครงการศึกษาระดับปริญญาโทสำเร็จขึ้นได้จากความช่วยเหลือของบุคคลหลายๆ ท่าน ข้าพเจ้าขอขอบคุณทุกท่านที่มีส่วนช่วยเหลือในด้านต่างๆ ด้วยความจริงใจ หากขาดบุคคลที่จะกล่าวถึงต่อไปนี้ก็จะส่งผลกระทบต่อความสำเร็จของโครงการศึกษาระดับปริญญาโท ข้าพเจ้าจึงขอขอบคุณทุกท่านในความช่วยเหลือด้วยความจริงใจของทุกท่าน

ขอขอบพระคุณ รศ.ดร.เอื้อน ปิ่นเงิน ที่กรุณารับเป็นอาจารย์ที่ปรึกษา ให้คำแนะนำตลอดจนให้ความช่วยเหลือในการตรวจสอบและแก้ไขข้อบกพร่องของการศึกษาค้นคว้าในการดำเนินโครงการฉบับนี้

ขอขอบพระคุณ คณาจารย์ทุกท่านที่ได้ประสาทวิชาความรู้ซึ่งช่วยให้ข้าพเจ้าได้รับความรู้ทั้งหลายทั้งมวลและ ได้นำความรู้ที่ได้รับ ไปใช้ประโยชน์ที่ก่อให้เกิดความเจริญก้าวหน้าต่อไป

ขอขอบคุณเพื่อนๆ ทุกคนในสถาบันที่ให้ความช่วยเหลือในด้านต่างๆ ทั้งการพัฒนาโครงการฯ ให้คำปรึกษาและเป็นกำลังใจมาโดยตลอด โดยเฉพาะผู้ช่วยคนสำคัญที่ให้ความรับผิดชอบต่างๆ ในฐานะประธานรุ่นเป็นไปด้วยความเรียบร้อยตลอดระยะเวลาการศึกษา จึงทำให้มีเวลาและสมาธิในการทำโครงการศึกษาระดับปริญญาโทได้อย่างเต็มที่

ขอขอบคุณข้อมูลและคำปรึกษาจากพี่น้องๆบริษัท เอ็ม เอช อี ซี จำกัด (มหาชน) ที่ช่วยให้ข้อมูลของโครงการศึกษาระดับปริญญาโทใกล้เคียงการปฏิบัติงานจริงมากที่สุด

สุดท้ายนี้ ขอกราบขอบพระคุณ บิดา มารดา ที่ให้การสนับสนุนและเป็นกำลังใจในการศึกษามาโดยตลอด

สุรเชษฐ์ เฉลิมผล

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญรูป	IX
บทที่ 1 บทนำ.....	1
1.1 วัตถุประสงค์ในการศึกษา	2
1.2 ขอบเขตการศึกษา	2
1.3 ขั้นตอนการศึกษา	2
1.4 ประโยชน์ที่คาดว่าจะได้รับจากการพัฒนาระบบ	3
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	4
2.1 วงจรการพัฒนาระบบ (System Development Life Cycle: SDLC).....	4
2.2 ทฤษฎีสเตปไวส์ (Step Wise).....	10
2.3 ภาษาจาวา.....	12
2.4 การวิเคราะห์และออกแบบเชิงวัตถุ.....	13
บทที่ 3 ออกแบบโครงสร้างขั้นตอนการบริหาร โครงการ.....	17
3.1 เครื่องมือที่ใช้ในขั้นตอนต่างๆและขอบเขตการทำงานของเครื่องมือ.....	17
3.2 รายละเอียดการออกแบบเครื่องมือ.....	20
บทที่ 4 การวิเคราะห์ระบบงาน	
4.1 การวิเคราะห์ระบบงาน.....	30
4.2 รายละเอียดคุณสมบัติของแต่ละระบบงาน.....	31
บทที่ 5 การออกแบบระบบงาน.....	62
5.1 การออกแบบยูสเคสของระบบงาน.....	62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา **IV**ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
5.2 การออกแบบ Class และ Method ของระบบ.....	63
5.3 Problem Domain & Package Diagram.....	64
5.4 CRC Card และ Method Specification.....	66
5.5 State Diagram.....	95
บทที่ 6 การออกแบบฐานข้อมูล.....	96
6.1 การออกแบบฐานข้อมูล.....	96
6.2 พจนานุกรมข้อมูล.....	98
บทที่ 7 การออกแบบโครงสร้างระบบและ โครงสร้างหน้าจอของเว็บแอปพลิเคชัน.....	105
7.1 การออกแบบ โครงสร้างระบบ.....	105
7.2 การออกแบบ โครงสร้างหน้าจอของเว็บแอปพลิเคชัน.....	106
บทที่ 8 การทดสอบและสรุปผล	113
8.1 การทดสอบระบบ.....	113
8.2 สรุปผล.....	113
8.3 ปัญหา ข้อจำกัด และข้อเสนอแนะ.....	114
บรรณานุกรม	115
ประวัติผู้เขียน	116

สารบัญตาราง

ตารางที่	หน้า
2.1 ทฤษฎีเสตปไวส์ (Step Wise).....	10
3.1 ตารางแสดงขั้นตอนการบริหาร โครงการและเครื่องมือ	17
3.2 ตารางแสดงค่าความซับซ้อน.....	22
3.3 ตารางแสดงค่าตัวแปรความซับซ้อน	23
3.4 ตารางแสดงค่าจำนวนบรรทัดของแต่ละโปรแกรม	24
3.5 ตารางแสดงข้อมูลเปรียบเทียบการเลือกวิธีการดำเนิน โครงการ	25
3.6 ตารางแสดงระยะเวลาการดำเนิน โครงการ.....	26
3.7 ตารางแสดงความเสี่ยง.....	27
3.8 ตารางแสดงจำนวนวันแต่ละกิจกรรม	28
4.1 รายละเอียดชุดเคส Create New Project.....	32
4.2 รายละเอียดชุดเคส Update Project Information.....	33
4.3 รายละเอียดชุดเคส Insert Cost-Benefit Information.....	36
4.4 รายละเอียดชุดเคส Feasibility Calculation.....	38
4.5 รายละเอียดชุดเคส Project scope management.....	40
4.6 รายละเอียดชุดเคส Project Infrastructure Management.....	42
4.7 รายละเอียดชุดเคส Project Estimation.....	44
4.8 รายละเอียดชุดเคส Implementation Methodology.....	46
4.9 รายละเอียดชุดเคส Project Planning.....	48
4.10 รายละเอียดชุดเคส Risk Management.....	50
4.11 รายละเอียดชุดเคส Update Risk Parameter on Task.....	52
4.12 รายละเอียดชุดเคส Resource Allocation.....	55
4.13 รายละเอียดชุดเคส Assign Resource.....	57
4.14 รายละเอียดชุดเคส Project Monitoring and Control.....	60
5.1 CRC Card Person.....	66
5.2 CRC Card Project Manager	67
5.3 CRC Card Project Admin	68
5.4 CRC Card Project Information	69
5.5 Method Specification Project Information	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และ VI ึ่งอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
5.6 CRC Card Project Evaluation	72
5.7 Method Specification Project Evaluation	73
5.8 CRC Card Project Scope	83
5.9 Method Specification Project Scope	84
5.10 CRC Card Project Infrastructure	85
5.11 Method Specification Project Infrastructure	86
5.12 CRC Card Project Estimation	87
5.13 Method Specification Project Estimation	88
5.14 CRC Card Function Point	90
5.15 CRC Card Project Implementation Methodology	91
5.16 Method Specification Project Implementation Methodology	92
5.17 CRC Card Project Planning	93
5.18 CRC Card Project Risk Management	94
5.19 CRC Card Resource Allocation	95
6.1 พจนานุกรมตารางข้อมูล BENEFIT.....	98
6.2 พจนานุกรมตารางข้อมูล COST.....	98
6.3 พจนานุกรมตารางข้อมูล COST_UNIT.....	98
6.4 พจนานุกรมตารางข้อมูล DEVELOP_METHOD.....	99
6.5 พจนานุกรมตารางข้อมูล FUNCTION_POINT_PARAM.....	99
6.6 พจนานุกรมตารางข้อมูล FUNCTION_POINT_SYSTEM_COMPLEXITY.....	99
6.7 พจนานุกรมตารางข้อมูล FUNCTION_POINT_SYSTEM_COMPONENT.....	99
6.8 พจนานุกรมตารางข้อมูล METHODOLOGY.....	100
6.9 พจนานุกรมตารางข้อมูล PERSON_ROLE.....	100
6.10 พจนานุกรมตารางข้อมูล PERSON.....	100
6.11 พจนานุกรมตารางข้อมูล PROJECT_STATUS.....	100
6.12 พจนานุกรมตารางข้อมูล PROJECT.....	101
6.13 พจนานุกรมตารางข้อมูล RESOURCE_ALLOCATION.....	102
6.14 พจนานุกรมตารางข้อมูล RESOURCE_CATEGORY.....	102

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา **VII** ให้อ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
6.15 พจนานุกรมตารางข้อมูล RESOURCE_TYPE.....	102
6.16 พจนานุกรมตารางข้อมูล RISK.....	103
6.17 พจนานุกรมตารางข้อมูล RISK_CATEGORY.....	103
6.18 พจนานุกรมตารางข้อมูล RISK_STATUS.....	103
6.19 พจนานุกรมตารางข้อมูล ROLE.....	103
6.20 พจนานุกรมตารางข้อมูล WBS.....	104



สารบัญรูป

รูปที่	หน้า
3.1 แสดงแบบจำลองการทำงานของโปรแกรม	22
4.1 ยูสเคสไดอะแกรมของระบบบริหาร โครงการ	30
4.2 ยูสเคสไดอะแกรมของระบบ Project Information Management.....	31
4.3 แอคติวิตีไดอะแกรมของ Update Project Information.....	34
4.4 ยูสเคสไดอะแกรมของระบบ Project Cost-Benefit Evaluation.....	35
4.5 แอคติวิตีไดอะแกรมของ Insert Cost-Benefit Information.....	37
4.6 แอคติวิตีไดอะแกรมของ Feasibility Calculation.....	39
4.7 ยูสเคสไดอะแกรมของระบบ Project scope management.....	39
4.8 แอคติวิตีไดอะแกรมของ Project scope management.....	41
4.9 ยูสเคสไดอะแกรมของระบบ Project Infrastructure Management.....	41
4.10 แอคติวิตีไดอะแกรมของ Project Infrastructure Management.....	43
4.11 ยูสเคสไดอะแกรมของระบบ Project Estimation.....	43
4.12 แอคติวิตีไดอะแกรมของ Project Estimation.....	45
4.13 ยูสเคสไดอะแกรมของระบบ Implementation Methodology.....	46
4.14 แอคติวิตีไดอะแกรมของ Implementation Methodology.....	47
4.15 ยูสเคสไดอะแกรมของระบบ Project Planning.....	47
4.16 แอคติวิตีไดอะแกรมของ Project Planning.....	49
4.17 ยูสเคสไดอะแกรมของระบบ Risk Management.....	49
4.18 แอคติวิตีไดอะแกรมของ Risk Management.....	51
4.19 แอคติวิตีไดอะแกรมของ Update Risk Parameter on Task.....	53
4.20 ยูสเคสไดอะแกรมของระบบ Resource Allocation.....	54
4.21 แอคติวิตีไดอะแกรมของ Resource Allocation.....	56
4.22 แอคติวิตีไดอะแกรมของ Assign Resource.....	58
4.23 ยูสเคสไดอะแกรมของระบบ Project Monitoring and Reporting.....	59
4.24 แอคติวิตีไดอะแกรมของ Project Monitoring and Control.....	61
5.1 ยูสเคสไดอะแกรมของภาพรวมทั้งระบบ.....	62
5.2 คลาสไดอะแกรม.....	63

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.3 คลาสไดอะแกรมและ Package Diagram.....	65
5.4 สเตทไดอะแกรม.....	95
6.1 แผนภาพความสัมพันธ์ระหว่างเอนทิตี.....	61
7.1 แผนภาพโครงสร้างระบบ.....	105
7.2 แผนภาพโครงสร้างหน้าจอของเว็บแอปพลิเคชัน.....	107
7.3 หน้าจอ New Project.....	108
7.4 หน้าจอ Project Information.....	109
7.5 หน้าจอ Project Evaluation.....	110
7.6 หน้าจอ Project Estimation.....	111
7.7 หน้าจอ Project Planning.....	112
7.8 หน้าจอ Project Risk.....	112

บทที่ 1

บทนำ

1. บทนำ

เนื่องด้วยปัจจุบันการบริหาร โครงการเป็นส่วนสำคัญสำหรับความสำเร็จของ โครงการต่างๆ โดยเฉพาะ โครงการพัฒนาซอฟต์แวร์ ซึ่งมีลักษณะเฉพาะในการบริหารที่แตกต่างไปจากการบริหาร โครงการโดยทั่วไป โครงการพัฒนาซอฟต์แวร์นั้นมีปัจจัยที่เกี่ยวข้องกับความสำเร็จของโครงการ อยู่หลายด้านด้วยกันเช่น ด้านบุคลากร เทคโนโลยีการออกแบบ เทคโนโลยีการพัฒนา ด้านการบริหารโครงการ ความเสี่ยงที่เกี่ยวข้องกับความสำเร็จของโครงการ เป็นต้น ทำให้การบริหารโครงการจำเป็นต้องทำอย่างเป็นระบบ

ปัจจัยดังกล่าวสามารถสร้างปัญหาให้กับโครงการจนถึงระดับที่ทำให้โครงการล้มเหลวได้เช่น โครงการที่เกิดความล้มเหลวเนื่องจากไม่สามารถตอบสนองความต้องการหรือความคาดหวังของลูกค้า หรือไม่สามารถส่งมอบซอฟต์แวร์ภายในเวลาที่กำหนดหรือใช้งบประมาณมากกว่าที่กำหนด

การบริหารโครงการที่ดีช่วยทำให้สามารถวางแผนงานการพัฒนา ระบุความเสี่ยงในการดำเนินโครงการ กำหนดบุคลากรที่จะใช้ในการพัฒนา รวมถึงการตรวจสอบสถานะของโครงการ ในระหว่างการพัฒนาเพื่อควบคุมให้เป็นไปตามแผนงานที่วางไว้ สามารถรับรู้และแก้ไขปัญหาที่จะเกิดขึ้นในระหว่างการพัฒนาโครงการได้ รวมถึงสามารถประเมินประสิทธิภาพของโครงการเพื่อนำไปปรับปรุงแก้ไขในการทำโครงการอื่นๆต่อไป

การที่จะสามารถบริหารโครงการได้อย่างมีประสิทธิภาพนั้นจำเป็นต้องอาศัยระบบสารสนเทศเข้ามาช่วยในการบริหารโครงการ ซึ่งหมายถึงความต้องการซอฟต์แวร์ที่ควรสามารถทำให้ผู้บริหารโครงการจัดเก็บข้อมูลต่างๆที่เกี่ยวข้องกับโครงการอย่างมีระบบ ช่วยในการกำหนดและบริหารแผนงานได้ นอกจากนั้นต้องสามารถตรวจสอบสถานะของโครงการในด้านต่างๆได้เช่น ความก้าวหน้าของโครงการ จำนวนคนและเวลาที่ใช้ในการพัฒนาโครงการในขณะใดๆ

โครงการนี้เป็นการศึกษาทฤษฎีที่เกี่ยวข้องกับการบริหารงานโครงการและนำความรู้ที่ได้รับมาทำการออกแบบซอฟต์แวร์ที่เป็นเครื่องมือในการช่วยบริหารงานโครงการ ซึ่งครอบคลุมขั้นตอนหลักที่สำคัญในการบริหารงานโครงการ เพื่อให้การบริหารโครงการเป็นไปอย่างมีประสิทธิภาพ สามารถใช้ข้อมูลต่างๆเกี่ยวกับโครงการทั้งหมดมาแสดงเพื่อให้เป็นภาพรวมของการบริหารโครงการและสามารถทราบถึงความเสี่ยงที่อาจทำให้ปัญหาต่อการดำเนินโครงการได้

อย่างทันถ่วงทีเพื่อกำหนดแนวทางการแก้ไขที่ถูกต้องทำให้การดำเนินโครงการเป็นไปตามวัตถุประสงค์และระยะเวลาที่กำหนดในแผนการดำเนินงาน

1.1 วัตถุประสงค์ในการศึกษา

1. เพื่อศึกษาขั้นตอนการบริหารโครงการพัฒนาซอฟต์แวร์ที่สำคัญ
2. เพื่อกำหนดเครื่องมือที่เหมาะสมกับการบริหารโครงการพัฒนาซอฟต์แวร์ในแต่ละขั้นตอน
3. เพื่อศึกษา วิเคราะห์และออกแบบระบบ โดยนำระบบสารสนเทศมาช่วยในการบริหารจัดการโครงการพัฒนาซอฟต์แวร์ให้มีประสิทธิภาพ
4. เพื่อนำข้อมูลในขั้นตอนต่างๆของการบริหารโครงการพัฒนาซอฟต์แวร์มาใช้อย่างสอดคล้องและมีประสิทธิภาพ

1.2 ขอบเขตการศึกษา

1. ศึกษาทฤษฎีบริหารโครงการเพื่อกำหนดขั้นตอนที่สำคัญในการบริหารโครงการ
2. ศึกษาเครื่องมือที่สำคัญในการบริหารงานโครงการ
3. กำหนดขั้นตอนการบริหารงานโครงการและเครื่องมือที่ใช้ในขั้นตอนต่างๆ
4. วิเคราะห์และออกแบบซอฟต์แวร์บริหารโครงการตามขั้นตอนการบริหารโครงการที่กำหนด

1.3 ขั้นตอนการศึกษา

ขั้นตอนการศึกษาประกอบด้วย 3 ส่วนหลักคือ

1. ศึกษาทฤษฎีที่เกี่ยวข้อง
 - 1.1. ศึกษาทฤษฎีการบริหารงานโครงการ สเตปไวส์ (Step Wise)
 - 1.2. ศึกษาเครื่องมือที่ใช้ในการบริหารงานโครงการ
2. ออกแบบโครงสร้างขั้นตอนการบริหารโครงการ
 - 2.1. พิจารณาขั้นตอนการบริหารโครงการตามทฤษฎี สเตปไวส์ (Step Wise)
 - 2.2. เลือกขั้นตอนการบริหารงานโครงการที่สำคัญ
 - 2.3. พิจารณาเครื่องมือที่ใช้ในการบริหารงานโครงการในแต่ละขั้นตอน
 - 2.4. เลือกเครื่องมือที่จะใช้ในแต่ละขั้นตอนในการบริหารงานโครงการ
3. ออกแบบเครื่องมือบริหารจัดการโครงการพัฒนาซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 3.1. กำหนดความต้องการของระบบ
- 3.2. วิเคราะห์ความต้องการของระบบ
- 3.3. ออกแบบระบบตามการวิเคราะห์ระบบ

1.4 ประโยชน์ที่คาดว่าจะได้รับจากการพัฒนาระบบ

1. ทำให้การบริหาร โครงการมีขั้นตอนที่ชัดเจนและเป็นไปอย่างมีระบบ
2. มีจัดเก็บข้อมูลต่างๆของการบริหารจัดการ โครงการอย่างมีประสิทธิภาพ
3. ทำให้ใช้ข้อมูลต่างๆในการบริหาร โครงการมีความต่อเนื่องและสัมพันธ์กัน
4. ทำให้สามารถตรวจสอบสถานะของ โครงการได้
5. ทำให้สามารถบริหารจัดการ โครงการ ได้อย่างมีประสิทธิภาพ



บทที่ 2

ทฤษฎีที่เกี่ยวข้องกับการพัฒนาระบบ

2.1 วงจรการพัฒนาระบบ (System Development Life Cycle: SDLC)

ระบบสารสนเทศทั้งหลายมีวงจรชีวิตที่เหมือนกันตั้งแต่เกิดจนตายวงจรนี้จะเป็นขั้นตอนที่เป็นลำดับตั้งแต่ต้นจนเสร็จเรียบร้อย เป็นระบบที่ใช้งานได้ ซึ่งนักวิเคราะห์ระบบต้องทำความเข้าใจให้ได้ว่าในแต่ละขั้นตอนจะต้องทำอะไร และทำอย่างไร ขั้นตอนการพัฒนาระบบมีอยู่ด้วยกัน 7 ขั้นตอนด้วยกัน คือ

1. เข้าใจปัญหา (Problem Recognition)
2. ศึกษาความเป็นไปได้ (Feasibility Study)
3. วิเคราะห์ (Analysis)
4. ออกแบบ (Design)
5. สร้างหรือพัฒนาระบบ (Construction)
6. การปรับเปลี่ยน (Conversion)
7. บำรุงรักษา (Maintenance)

ขั้นที่ 1: เข้าใจปัญหา (Problem Recognition)

ระบบสารสนเทศจะเกิดขึ้นได้ก็ต่อเมื่อผู้บริหารหรือผู้ใช้ตระหนักว่าต้องการระบบสารสนเทศหรือระบบจัดการเดิม ได้แก่ระบบเอกสารในผู้เอกสาร ไม่มีประสิทธิภาพเพียงพอที่ตอบสนองความต้องการในปัจจุบัน

ปัจจุบันผู้บริหารตื่นตัวกันมากที่จะให้มีการพัฒนาระบบสารสนเทศมาใช้ในหน่วยงานของตน ในงานธุรกิจ อุตสาหกรรม หรือใช้ในการผลิต ตัวอย่างเช่น บริษัทของเรา จำกัด ติดต่อซื้อสินค้าจากผู้ขายหลายบริษัท ซึ่งบริษัทของเรามีระบบ MIS ที่เก็บข้อมูลเกี่ยวกับหนี้สินที่บริษัทของเราคิดค้างผู้ขายอยู่ แต่ระบบเก็บข้อมูลผู้ขายได้เพียง 1,000 รายเท่านั้น แต่ปัจจุบันผู้ขายมีระบบเก็บข้อมูลถึง 900 ราย และอนาคตอันใกล้นี้จะเกิน 1,000 ราย ดังนั้นฝ่ายบริหารจึงเรียกนักวิเคราะห์ระบบเข้ามาศึกษา แก้ไขระบบงาน

ปัญหาที่สำคัญของระบบสารสนเทศในปัจจุบัน คือ ระบบเขียนมานานแล้ว ส่วนใหญ่เขียนมาเพื่อติดตามเรื่องการเงิน ไม่ได้มีจุดประสงค์เพื่อให้ข้อมูลข่าวสารในการตัดสินใจ แต่ปัจจุบันฝ่ายบริหารต้องการดูสถิติการขายเพื่อใช้ในการคาดคะเนในอนาคต หรือความต้องการอื่นๆ เช่น สินค้าที่มียอดขายสูง หรือสินค้าที่ลูกค้าต้องการสูง หรือการแยกประเภทสินค้าต่างๆ ที่ทำได้ไม่ง่ายนัก การที่จะแก้ไขระบบเดิมที่มีอยู่แล้ว ไม่ใช่เรื่องที่ย่ายนั้ก หรือแม้แต่การสร้างระบบใหม่ ดังนั้นควรจะมีไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การศึกษาเสียก่อนว่าความต้องการของเราเพียงพอที่เป็นไปได้หรือไม่ ได้แก่ "การศึกษาความเป็นไปได้" (Feasibility Study)

สรุป ขั้นตอนที่ 1: เข้าใจปัญหา

หน้าที่ : ตระหนักว่ามีปัญหาในระบบ

ผลลัพธ์ : อนุมัติการศึกษาความเป็นไปได้

เครื่องมือ : ไม่มี

บุคลากรและหน้าที่ความรับผิดชอบ : ผู้ใช้หรือผู้บริหารชี้แจงปัญหาต่อนักวิเคราะห์ระบบ

ขั้นตอนที่ 2: ศึกษาความเป็นไปได้ (Feasibility Study)

จุดประสงค์ของการศึกษาความเป็นไปได้อีกคือ การกำหนดว่าปัญหาคืออะไรและตัดสินใจว่าการพัฒนาสร้างระบบสารสนเทศ หรือการแก้ไขระบบสารสนเทศเดิมมีความเป็นไปได้หรือไม่ โดยเสียค่าใช้จ่ายและเวลาน้อยที่สุด และได้ผลเป็นที่น่าพอใจ

ปัญหาต่อไปคือ นักวิเคราะห์ระบบจะต้องกำหนดให้ได้ว่า การแก้ไขปัญหาดังกล่าวมีความเป็นไปได้ทางเทคนิคและบุคลากร ปัญหาทางเทคนิคก็จะเกี่ยวข้องกับเรื่องคอมพิวเตอร์ และเครื่องมือต่างๆ ถ้ามี รวมทั้งเรื่องคอมพิวเตอร์ซอฟต์แวร์ด้วย ตัวอย่างคือ คอมพิวเตอร์ที่ใช้อยู่ในบริษัทเพียงพอหรือไม่ คอมพิวเตอร์อาจจะมีเนื้อที่ของฮาร์ดดิสก์ไม่เพียงพอ รวมทั้งซอฟต์แวร์ว่า อาจจะต้องซื้อใหม่ หรือพัฒนาขึ้นใหม่ เป็นต้น ความเป็นไปได้ทางด้านบุคลากร คือ บริษัทมีบุคคลที่เหมาะสมที่จะพัฒนาและติดตั้งระบบเพียงพอหรือไม่ ถ้าไม่มีจะหาได้หรือไม่ จากที่ใด เป็นต้น นอกจากนั้นควรจะให้ความสนใจว่าผู้ใช้ระบบมีความคิดเห็นอย่างไรกับการเปลี่ยนแปลง รวมทั้งความเห็นของผู้บริหารด้วย

สุดท้ายนักวิเคราะห์ระบบต้องวิเคราะห์ได้ว่า ความเป็นไปได้อีกเรื่องค่าใช้จ่าย รวมทั้งเวลาที่ใช้ในการพัฒนาระบบ และที่สำคัญคือ ผลประโยชน์ที่จะได้รับ เรื่องเวลาเป็นสิ่งสำคัญ เช่น การเปลี่ยนแปลงระบบเพื่อรองรับผู้ขายให้ได้มากกว่า 1,000 บริษัทนั้น ควรใช้เวลาไม่เกิน 1 ปี ตั้งแต่เริ่มต้นจนถึงงานได้ ค่าใช้จ่ายเริ่มตั้งแต่พัฒนาจนถึงใช้งานได้จริง ได้แก่ เงินเดือน เครื่องมือ อุปกรณ์ต่างๆ เป็นต้น พุทธิถึงเรื่องผลประโยชน์ที่ได้รับอาจมองเห็นได้ไม่่ง่ายนัก แต่นักวิเคราะห์ระบบควรมองและตีออกมาในรูปเงินให้ได้ เช่น เมื่อนำระบบใหม่เข้ามาใช้จะทำให้ ค่าใช้จ่ายบุคลากรลดลง หรือกำไรเพิ่มมากขึ้น เช่น ทำให้ยอดขายเพิ่มมากขึ้น เนื่องจากผู้บริหารมีข้อมูลพร้อมที่จะช่วยในการตัดสินใจที่ดีขึ้น

สรุปขั้นตอนที่ 2: การศึกษาความเป็นไปได้ (Feasibility Study)

หน้าที่ : กำหนดปัญหา และศึกษาว่าความเป็นไปได้อาจจะเปลี่ยนแปลงระบบ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับอาจารย์ผู้สอนเท่านั้น ไม่ให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลลัพธ์ : รายงานความเป็นไปได้

เครื่องมือ : เก็บรวบรวมข้อมูลของระบบและคาดคะเนความต้องการของระบบ

บุคลากรและหน้าที่ความรับผิดชอบ : ผู้ใช้จะมีบทบาทสำคัญในการศึกษา

1. นักวิเคราะห์ระบบจะเก็บรวบรวมข้อมูลทั้งหมดที่จำเป็นทั้งหมดเกี่ยวกับปัญหา
2. นักวิเคราะห์ระบบคาดคะเนความต้องการของระบบและแนวทางการแก้ปัญหา
3. นักวิเคราะห์ระบบ กำหนดความต้องการที่แน่ชัดซึ่งจะใช้สำหรับขั้นตอนการวิเคราะห์ต่อไป
4. ผู้บริหารตัดสินใจว่าจะดำเนินโครงการต่อไปหรือไม่

ขั้นตอนที่ 3 การวิเคราะห์ (Analysis)

เริ่มเข้าสู่การวิเคราะห์ระบบ การวิเคราะห์ระบบเริ่มตั้งแต่การศึกษาระบบการทำงานของธุรกิจนั้น ในกรณีที่ระบบเรศึกษานั้นเป็นระบบสารสนเทศอยู่แล้วจะต้องศึกษาว่าทำงานอย่างไร เพราะเป็นการยากที่จะออกแบบระบบใหม่โดยที่ไม่ทราบว่าจะระบบเดิมทำงานอย่างไร หรือธุรกิจดำเนินการอย่างไร หลังจากนั้นกำหนดความต้องการของระบบใหม่ ซึ่งนักวิเคราะห์ระบบจะต้องใช้เทคนิคในการเก็บข้อมูล (Fact-Gathering Techniques) ดังรูป ได้แก่ ศึกษาเอกสารที่มีอยู่ ตรวจสอบวิธีการทำงานในปัจจุบัน สัมภาษณ์ผู้ใช้และผู้จัดการที่มีส่วนเกี่ยวข้องกับระบบ เอกสารที่มีอยู่ได้แก่ คู่มือการใช้งาน แผนผังใช้งานขององค์กร รายงานต่างๆที่หมุนเวียนในระบบการศึกษาวิธีการทำงานในปัจจุบันจะทำให้นักวิเคราะห์ระบบรู้ว่าระบบจริงๆทำงานอย่างไร ซึ่งบางครั้งค้นพบข้อผิดพลาดได้ ตัวอย่าง เช่น เมื่อบริษัทได้รับใบเรียกเก็บเงินจะมีขั้นตอนอย่างไรในการจ่ายเงิน ขั้นตอนที่เสมือนใบเรียกเก็บเงินอย่างไร เฝ้าสังเกตการทำงานของผู้เกี่ยวข้อง เพื่อให้เข้าใจและเห็นจริงๆ ว่าขั้นตอนการทำงานเป็นอย่างไร ซึ่งจะทำให้นักวิเคราะห์ระบบค้นพบจุดสำคัญของระบบว่าอยู่ที่ใด

การสัมภาษณ์เป็นศิลปะอย่างหนึ่งที่นักวิเคราะห์ระบบควรจะต้องมีเพื่อเข้ากับผู้ใช้ได้ง่าย และสามารถดึงสิ่งที่ต้องการจากผู้ใช้ได้ เพราะว่าความต้องการของระบบคือ สิ่งสำคัญที่จะใช้ในการออกแบบต่อไป ถ้าเราสามารถกำหนดความต้องการได้ถูกต้อง การพัฒนาระบบในขั้นตอนต่อไปก็จะง่ายขึ้น เมื่อเก็บรวบรวมข้อมูลแล้วจะนำมาเขียนรวมเป็นรายงานการทำงานของระบบซึ่งควรแสดงหรือเขียนออกมาเป็นรูปแทนที่จะบรรยายออกมาเป็นตัวหนังสือ การแสดงแผนภาพจะทำให้เราเข้าใจได้ดีและง่ายขึ้น หลังจากนั้นนักวิเคราะห์ระบบ อาจจะนำข้อมูลที่รวบรวมได้นำมาเขียนเป็น "แบบทดลอง" (Prototype) หรือตัวต้นแบบ แบบทดลองจะเขียนขึ้นด้วยภาษาคอมพิวเตอร์ต่างๆ และที่ช่วยให้ง่ายขึ้นได้แก่ ภาษายุคที่ 4 (Fourth Generation Language) เป็นการสร้างโปรแกรมคอมพิวเตอร์ขึ้นมาเพื่อใช้งานตามที่เราต้องการได้ ดังนั้นแบบทดลองจึงช่วยลดข้อผิดพลาดที่อาจจะเกิดขึ้นได้ เมื่อจบขั้นตอนการวิเคราะห์แล้ว นักวิเคราะห์ระบบจะต้องเขียน

เอกสารสรุปลงมาเป็น ข้อมูลเฉพาะของปัญหา (Problem Specification) ซึ่งมีรายละเอียดดังนี้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของระบบเดิม ซึ่งควรเขียนมาเป็นรูปภาพแสดงการทำงานของระบบ พร้อมคำบรรยาย, กำหนดความต้องการของระบบใหม่รวมทั้งรูปภาพแสดงการทำงานของระบบพร้อมคำบรรยาย, ข้อมูลและไฟล์ที่จำเป็น, คำอธิบายวิธีการทำงาน และสิ่งที่จะต้องแก้ไข. รายงานข้อมูลเฉพาะของปัญหาของระบบขนาดกลางควรจะมีขนาดไม่เกิน 100-200 หน้ากระดาษ

สรุป ขั้นตอนที่3: การวิเคราะห์ (Analysis)

หน้าที่ : กำหนดความต้องการของระบบใหม่ (ระบบใหม่ทั้งหมดหรือแก้ไขระบบเดิม)

ผลลัพธ์ : รายงานข้อมูลเฉพาะของปัญหา

เครื่องมือ : เทคนิคการเก็บรวบรวมข้อมูล, Data Dictionary, Data Flow Diagram, Process

Specification, Data Model, System Model, Prototype, system Flowcharts

บุคลากรและหน้าที่รับผิดชอบ : ผู้ใช้จะต้องให้ความร่วมมือเป็นอย่างดี

1. วิเคราะห์ระบบ ศึกษาเอกสารที่มีอยู่ และศึกษาระบบเดิมเพื่อให้เข้าใจถึงขั้นตอนการทำงานและทราบว่าจุดสำคัญของระบบอยู่ที่ไหน
2. นักวิเคราะห์ระบบ เตรียมรายงานความต้องการของระบบใหม่
3. นักวิเคราะห์ระบบ เขียนแผนภาพการทำงาน (Diagram) ของระบบใหม่ โดยไม่ต้องบอกว่าหน้าที่ใหม่ในระบบจะพัฒนาขึ้นมาได้อย่างไร
4. นักวิเคราะห์ระบบ เขียนสรุปรายงานข้อมูลเฉพาะของปัญหา
5. ถ้าเป็นไปได้นักวิเคราะห์ระบบอาจจะเตรียมแบบทดลองด้วย

ขั้นตอนที่4: การออกแบบ (Design)

ในระยะแรกของการออกแบบ นักวิเคราะห์ระบบจะนำการตัดสินใจของฝ่ายบริหารที่ได้จากขั้นตอนการวิเคราะห์การเลือกซื้อคอมพิวเตอร์ ฮาร์ดแวร์และซอฟต์แวร์ด้วย (ถ้ามีหรือเป็นไปได้) หลังจากนั้นนักวิเคราะห์ระบบจะนำแผนภาพต่างๆ ที่เขียนขึ้นในขั้นตอนการวิเคราะห์มาแปลงเป็นแผนภาพลำดับขั้น (แบบต้นไม้) ดังรูปข้างล่าง เพื่อให้มองเห็นภาพลักษณะที่แน่นอนของโปรแกรมว่ามีความสัมพันธ์กันอย่างไร และโปรแกรมอะไรบ้างที่จะต้องเขียนในระบบ หลังจากนั้นก็เริ่มตัดสินใจว่าจะจัดโครงสร้างจากโปรแกรมอย่างไร การเชื่อมระหว่างโปรแกรมควรจะทำอย่างไร ในขั้นตอนการวิเคราะห์นักวิเคราะห์ระบบต้องหว่า "จะต้องทำอะไร (What)" แต่ในขั้นตอนการออกแบบต้องรู้ว่า "จะต้องทำอะไร (How)"

ในการออกแบบโปรแกรมต้องคำนึงถึงความปลอดภัย (Security) ของระบบด้วย เพื่อป้องกันการผิดพลาดที่อาจจะเกิดขึ้น เช่น "รหัส" สำหรับผู้ใช้ที่มีสิทธิ์สำรองไฟล์ข้อมูลทั้งหมด เป็นต้น นักวิเคราะห์ระบบจะต้องออกแบบฟอร์มสำหรับข้อมูลขาเข้า (Input Format) ออกแบบรายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Report Format) และการแสดงผลบนจอภาพ (Screen Format) หลักการการออกแบบฟอร์มข้อมูลขาเข้าคือ ง่ายต่อการใช้งาน และป้องกันข้อผิดพลาดที่อาจจะเกิดขึ้น

ถัดมาระบบจะต้องออกแบบวิธีการใช้งาน เช่น กำหนดว่าการป้อนข้อมูลจะต้องทำอย่างไร จำนวนบุคลากรที่ต้องการในหน้าที่ต่างๆ แต่ถ้านักวิเคราะห์ระบบตัดสินใจว่าการซื้อซอฟต์แวร์ดีกว่าการเขียนโปรแกรม ขั้นตอนการออกแบบก็ไม่จำเป็นเลย เพราะสามารถนำซอฟต์แวร์สำเร็จรูปมาใช้งานได้ทันที สิ่งที่นักวิเคราะห์ระบบออกแบบมาทั้งหมดในขั้นตอนที่กล่าวมาทั้งหมดจะนำมาเขียนรวมเป็นเอกสารชุดหนึ่งเรียกว่า “ข้อมูลเฉพาะของการออกแบบระบบ” (System Design Specification) เมื่อสำเร็จแล้วโปรแกรมเมอร์สามารถใช้เป็นแบบในการเขียนโปรแกรมได้ทันทีที่สำคัญก่อนที่จะส่งถึงมือ โปรแกรมเมอร์เราควรตรวจสอบกับผู้ใช้ว่าพอใจหรือไม่ และตรวจสอบกับทุกคนในทีมว่าถูกต้องสมบูรณ์หรือไม่ และแน่นอนที่สุดต้องส่งให้ฝ่ายบริหารเพื่อตัดสินใจว่าจะดำเนินการ ต่อไปหรือไม่ ถ้านุมัติก็ผ่านเข้าสู่ขั้นตอนการสร้างหรือพัฒนาระบบ (Construction)

สรุปขั้นตอนที่ 4: การออกแบบ (Design)

หน้าที่ : ออกแบบระบบใหม่เพื่อให้สอดคล้องกับความต้องการของผู้ใช้และฝ่ายบริหาร

ผลลัพธ์ : ข้อมูลเฉพาะของการออกแบบ(System Design Specification)

เครื่องมือ : พจนานุกรมข้อมูล Data Dictionary, แผนภาพการไหลของข้อมูล (Data Flow Diagram), ข้อมูลเฉพาะการประมวลผล (Process Specification), รูปแบบข้อมูล (Data Model), รูปแบบระบบ (System Model), พังงานระบบ (System Flow Charts), พังงาน โครงสร้าง (Structure Charts), พังงาน HIPO (HIPO Chart), แบบฟอร์มข้อมูลขาเข้าและรายงาน

บุคลากรและหน้าที่ :

1. นักวิเคราะห์ระบบ ตัดสินใจเลือกคอมพิวเตอร์ฮาร์ดแวร์และซอฟต์แวร์ (ถ้าใช้)
2. นักวิเคราะห์ระบบ เปลี่ยนแผนภาพทั้งหลายที่ได้จากขั้นตอนการวิเคราะห์มาเป็นแผนภาพลำดับขั้น
3. นักวิเคราะห์ระบบ ออกแบบความปลอดภัยของระบบ
4. นักวิเคราะห์ระบบ ออกแบบฟอร์มข้อมูลขาเข้า รายงาน และการแสดงผลบนจอ
5. นักวิเคราะห์ระบบ กำหนดจำนวนบุคลากรในหน้าที่ต่างๆและการทำงานของระบบ
6. ผู้ใช้ ฝ่ายบริหาร และนักวิเคราะห์ระบบ ทบทวน เอกสารข้อมูลเฉพาะของการออกแบบเพื่อความถูกต้องและสมบูรณ์แบบของระบบ

ขั้นตอนที่ 5: การพัฒนาระบบ (Construction)

ในขั้นตอนนี้โปรแกรมเมอร์จะเริ่มเขียนและทดสอบ โปรแกรมว่าทำงานถูกต้องหรือไม่ต้อง

มีการทดสอบกับข้อมูลจริงที่เลือกแล้ว ถ้าทุกอย่างเรียบร้อย เราจะได้โปรแกรมที่พร้อมที่จะนำไปใช้
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานจริงต่อไป หลังจากนั้นต้องเตรียมคู่มือการใช้และการฝึกอบรมผู้ใช้งานจริงของระบบ ระยะแรกในขั้นตอนนี้ นักวิเคราะห์ระบบต้องเตรียมสถานที่สำหรับเครื่องคอมพิวเตอร์แล้วจะต้องตรวจสอบว่าคอมพิวเตอร์ทำงานเรียบร้อยดี

โปรแกรมเมอร์เขียน โปรแกรมตามข้อมูลที่ได้จากเอกสารข้อมูลเฉพาะของการออกแบบ (Design Specification) ปกติแล้วนักวิเคราะห์ระบบไม่มีหน้าที่เกี่ยวข้องในการเขียน โปรแกรม แต่ถ้าโปรแกรมเมอร์คิดว่าการเขียนอย่างอื่นดีกว่าจะต้องปรึกษานักวิเคราะห์ระบบเสียก่อน เพื่อที่ว่านักวิเคราะห์จะบอกได้ว่า โปรแกรมที่จะแก้ไขนั้นมีผลกระทบกับระบบทั้งหมดหรือไม่ โปรแกรมเมอร์เขียนเสร็จแล้วต้องมีการทบทวนกับนักวิเคราะห์ระบบและผู้ใช้งาน เพื่อค้นหาข้อผิดพลาด วิธีการนี้เรียกว่า "Structure Walkthrough" การทดสอบ โปรแกรมจะต้องทดสอบกับข้อมูลที่เลือกแล้วชุดหนึ่ง ซึ่งอาจจะเลือกโดยผู้ใช้ การทดสอบเป็นหน้าที่ของโปรแกรมเมอร์ แต่นักวิเคราะห์ระบบต้องแน่ใจว่า โปรแกรมทั้งหมดจะต้องไม่มีข้อผิดพลาด

หลังจากนั้นต้องควบคุมดูแลการเขียนคู่มือซึ่งประกอบด้วยข้อมูลการใช้งานสารบัญการอ้างอิง "Help" บนจอภาพ เป็นต้น นอกจากข้อมูลการใช้งานแล้ว ต้องมีการฝึกอบรมพนักงานที่จะเป็นผู้ใช้งานจริงของระบบเพื่อให้เข้าใจและทำงานได้โดยไม่มีปัญหาอาจจะอบรมตัวต่อตัวหรือเป็นกลุ่มก็ได้

สรุปขั้นตอนที่ 5: การพัฒนาระบบ (Construction)

หน้าที่ : เขียนและทดสอบ โปรแกรม

ผลลัพธ์ : โปรแกรมที่ทดสอบเรียบร้อยแล้ว เอกสารคู่มือการใช้ และการฝึกอบรม

เครื่องมือ : เครื่องมือของ โปรแกรมเมอร์ทั้งหลาย Editor, compiler, Structure Walkthrough, วิธีการทดสอบโปรแกรม การเขียนเอกสารประกอบการใช้งาน

บุคลากรและหน้าที่ :

1. นักวิเคราะห์ระบบ ดูแลการเตรียมสถานที่และติดตั้งเครื่องคอมพิวเตอร์ (ถ้าซื้อใหม่)
2. นักวิเคราะห์ระบบ วางแผนและดูแลการเขียน โปรแกรม ทดสอบ โปรแกรม
3. โปรแกรมเมอร์เขียนและทดสอบ โปรแกรม หรือแก้ไข โปรแกรม ถ้าซื้อ โปรแกรมสำเร็จรูป
4. นักวิเคราะห์ระบบ วางแผนทดสอบ โปรแกรม
5. ทีมที่ทำงานร่วมกันทดสอบ โปรแกรม
6. ผู้ใช้ตรวจสอบให้แน่ใจว่า โปรแกรมทำงานตามต้องการ
7. นักวิเคราะห์ระบบ ดูแลการเขียนคู่มือการใช้งานและการฝึกอบรม

ขั้นตอนที่ 6: การปรับเปลี่ยน (Construction)

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการดำเนินงานเท่านั้น การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารเป็นการฝ่าฝืนเงื่อนไขการใช้งาน และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ป้อนข้อมูลต้องทำให้เรียบร้อย และในที่สุดบริษัทเริ่มต้นใช้งานระบบใหม่นี้ได้

การนำระบบเข้ามาควรจะทำอย่างค่อยเป็นค่อยไปที่ละน้อย ที่ดีที่สุดคือ ใช้ระบบใหม่ควบคู่ไปกับระบบเก่าไปสักระยะหนึ่ง โดยใช้ข้อมูลชุดเดียวกันแล้วเปรียบเทียบผลลัพธ์ว่าตรงกันหรือไม่ ถ้าเรียบร้อยก็เอาระบบเก่าออกได้ แล้วใช้ระบบใหม่ต่อไป

ขั้นตอนที่ 7: บำรุงรักษา (Maintenance)

การบำรุงรักษาได้แก่ การแก้ไข โปรแกรมหลังจากการใช้งานแล้ว สาเหตุที่ต้องแก้ไข โปรแกรมหลังจากใช้งานแล้ว สาเหตุที่ต้องแก้ไขระบบส่วนใหญ่มี 2 ข้อ คือ

1. มีปัญหาในโปรแกรม (Bug) และ
2. การดำเนินงานในองค์กรหรือธุรกิจเปลี่ยนไป

จากสถิติของระบบที่พัฒนาแล้วทั้งหมดประมาณ 40% ของค่าใช้จ่ายในการแก้ไข โปรแกรม เนื่องจากมี "Bug" ดังนั้นนักวิเคราะห์ระบบควรให้ความสำคัญกับการบำรุงรักษา ซึ่งปกติจะคิดว่าไม่มีความสำคัญมากนัก เมื่อธุรกิจขยายตัวมากขึ้น ความต้องการของระบบอาจจะเพิ่มมากขึ้น เช่น ต้องการรายงานเพิ่มขึ้น ระบบที่คิดควรจะแก้ไขเพิ่มเติมถึงที่ความต้องการได้

การบำรุงรักษาระบบ ควรจะอยู่ภายใต้การดูแลของนักวิเคราะห์ระบบ เมื่อผู้บริหารต้องการแก้ไขส่วนใดนักวิเคราะห์ระบบต้องเตรียมแผนภาพต่าง ๆ และศึกษาผลกระทบต่อระบบ และให้ผู้บริหารตัดสินใจต่อไปว่าควรจะแก้ไขหรือไม่

2.2 ทฤษฎีเสตปไวส์ (Step Wise)

ตารางที่ 2.1 ทฤษฎีเสตปไวส์ (Step Wise)

Step	Activities with in step
0	Select project
1	Identify project scope and objectives <ol style="list-style-type: none"> 1.1 Identify objective and measures of effectiveness in meeting them 1.2 Establish a project authority 1.3 Identify stakeholders 1.4 Modify objective in the light of stakeholder analysis 1.5 Establish method of communication with all project

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 (ต่อ)

Step	Activities with in step
2	Identify project infrastructure 2.1 Establish relationship between project and strategic planning 2.2 Identify installation standard and procedures 2.3 Identify project team organization
3	Analyses project characteristics 3.1 Distinguish the project as either objective or product driven 3.2 Analyses other project characteristic 3.3 Identify high-level project risk 3.4 Take into account user requirement concerning implementation 3.5 Select general life cycle approach 3.6 Review overall resource estimation
4	Identify project product and activities 4.1 Identify and describe project product (Including quality criteria) 4.2 Document generic product flows 4.3 Recognize product instance 4.4 Produce ideal activity network 4.5 Modify ideal to take into account need for stages and checkpoint
5	Estimate effort each activity 5.1 Carry out bottom-up estimation 5.2 Revise plan to create controllable activity
6	Identify activity risk 6.1 Identify and quantity activity-based risks 6.2 Plan risk reduction and contingency measures where appropriate 6.3 Adjust plans and estimates to take account of risk
7	Allocate resources 7.1 Identify and allocate resource 7.2 Revise plans and estimates to account for resource constraints
8	Review/publicize plan 8.1 Review quality aspects of project plan 8.2 Document plans and obtain agreement

ตารางที่ 2.2 (ต่อ)

Step	Activities with in step
9/10	Execute plan/lower levels of planning This may requirement the reiteration of the planning progress at a lower level

ใน Step 0-3 เป็นส่วนของการศึกษาโครงการในด้านต่างๆซึ่งเป็นส่วนสำคัญในการเลือกโครงการที่จะทำการพัฒนาในด้านความคุ้มค่าทางธุรกิจ และประเมินความเป็นไปได้ของโครงการ

ใน Step 4-8 เป็นส่วนของการกำหนดรายละเอียดเกี่ยวกับ ขั้นตอนการพัฒนา จำนวนคนที่ต้องใช้ในการพัฒนาในส่วนต่างๆ ความเสี่ยงที่ต้องคำนึงถึงในการพัฒนา

ใน Step 9-10 เป็นส่วนของการนำแผนการทำงานไปใช้งานจริงและทำการตรวจสอบสถานะของโครงการในช่วงเวลาระหว่างพัฒนาโครงการเพื่อให้สามารถแก้ไขปัญหาที่จะเกิดขึ้นกับโครงการในด้านต่างๆได้อย่างทันกับสถานะการณ์

2.3 ภาษาจาวา

ภาษาจาวา (Java programming language) เป็นภาษาโปรแกรมเชิงวัตถุ พัฒนาโดย เจมส์ กอสลิง และวิศวกรคนอื่นๆ ที่ ซัน ไมโครซิสเต็มส์ ภาษาจาวาถูกพัฒนาขึ้นในปี พ.ศ. 2534 (ค.ศ. 1991) โดยเป็นส่วนหนึ่งของ โครงการกรีน (the Green Project) และสำเร็จออกสู่สาธารณะในปี พ.ศ. 2538 (ค.ศ. 1995) ซึ่งภาษานี้มีจุดประสงค์เพื่อใช้แทนภาษาซีพลัสพลัส (C++) โดยรูปแบบที่เพิ่มเติมขึ้นคล้ายกับภาษาอ็อบเจกต์ทีฟซี (Objective-C) แต่เดิมภาษานี้เรียกว่า ภาษาโอ๊ก (Oak) ซึ่งตั้งชื่อตามต้นไม้โอ๊กใกล้ที่ทำงานของ เจมส์ กอสลิง แต่ว่ามีปัญหาทางลิขสิทธิ์ จึงเปลี่ยนไปใช้ชื่อ “จาวา” ซึ่งเป็นชื่อกาแฟแทน และแม้ว่าจะมีชื่อคล้ายกัน แต่ภาษาจาวา ไม่มีความเกี่ยวข้องกับใด ๆ กับภาษาจาวาสคริปต์ (JavaScript) ปัจจุบันมาตรฐานของภาษาจาวาดูแลโดย Java Community Process ซึ่งเป็นกระบวนการอย่างเป็นทางการ ที่อนุญาตให้ผู้ที่สนใจเข้าร่วมกำหนดความสามารถในจาวาแพลตฟอร์มได้ จุดมุ่งหมายหลัก 4 ประการ ในการพัฒนาจาวา คือ

1. ใช้ภาษาโปรแกรมเชิงวัตถุ
2. ไม่ขึ้นกับแพลตฟอร์ม (สถาปัตยกรรม และ ระบบปฏิบัติการ)
3. เหมาะกับการใช้ในระบบเครือข่าย พร้อมมีไลบรารีสนับสนุน
4. เรียกใช้งานจากระยะไกลได้อย่างปลอดภัย

จาวาแพลตฟอร์ม และ ภาษาจาวา เนื่องจากชื่อที่เหมือนกัน และการเรียกขานที่มักจะถูกพูดถึงพร้อมกันบ่อยๆ ทำให้คนทั่วไป มักสับสนว่า ภาษาจาวา และ จาวาแพลตฟอร์ม เป็นสิ่งเดียวกัน ในความเป็นจริงนั้น ทั้งสองสิ่ง แม้จะทำงานเสริมกัน แต่ก็ก็เป็นสิ่งที่แยกออกจากกัน โดย ภาษาจาวานั้นคือภาษาโปรแกรมอย่างหนึ่ง ดังที่ได้อธิบายไปข้างต้น ส่วน จาวาแพลตฟอร์มนั้น คือสภาพแวดล้อม

สำหรับการใช้งานโปรแกรมจาวา โดยมีองค์ประกอบหลักคือ จาวาเวอร์ชวลแมชีน (Java virtual machine) และ ไลบรารีมาตรฐานจาวา (Java standard library) โปรแกรมที่ทำงานบนจาวา แพลตฟอร์มนั้น ไม่จำเป็นจะต้องสร้างด้วยภาษาจาวา เช่น อาจจะใช้ ภาษาไพธอน (Python) หรือ ภาษาอื่นๆ ก็ได้ ส่วนภาษาจาวานั้น ก็สามารถนำไปใช้พัฒนาโปรแกรมสำหรับแพลตฟอร์มอื่นได้ เช่นเดียวกัน เช่น คอมไพเลอร์ gcj สามารถคอมไพล์โปรแกรมที่เขียนด้วยภาษาจาวา ให้ทำงานได้ โดยไม่ต้องใช้ จาวาเวอร์ชวลแมชีน

การพัฒนาซอฟต์แวร์ด้วยการเขียนโปรแกรมในแนวตั้งเดิม (Structured Programming) เริ่มมีความนิยมลดลง เนื่องจากการพัฒนาซอฟต์แวร์ในลักษณะนี้ ไม่สามารถรองรับความซับซ้อนของ โปรแกรมที่เพิ่มขึ้นทุกวัน ๆ ได้ จึงได้มีการพัฒนาการเขียนโปรแกรมเชิงวัตถุขึ้น (Object Orientation) ที่มีลักษณะสำคัญคือ การมองข้อมูล เหตุการณ์ หรือปัญหาต่าง ๆ ให้เป็นวัตถุ (Object) ซึ่งได้มีการแยกแยะวัตถุเหล่านี้ให้เป็นหมวดหมู่ (Class) เพื่อพัฒนาเป็น โปรแกรมต่อไป ทำให้สามารถนำมาใช้ในการพัฒนาซอฟต์แวร์สำหรับระบบงานที่มีขนาดใหญ่ได้ดีขึ้น

2.4 การวิเคราะห์และออกแบบเชิงวัตถุ

2.4.1 แนวคิดพื้นฐานเชิงวัตถุ (Object Oriented Paradigm)

การวิเคราะห์และออกแบบเชิงวัตถุ เป็นวิธีการหนึ่งในการวิเคราะห์และออกแบบระบบที่ แสดงให้เห็นระบบจากมุมมองของตัวเอง โดยใช้แบบจำลองเชิงวัตถุ (Object Model) ซึ่งเป็น แนวคิดหรือระเบียบวิธีคิดของการสร้างหรือพัฒนาระบบงานหนึ่ง ๆ โดยจะมองระบบเป็นกลุ่มของ วัตถุที่มีปฏิสัมพันธ์กัน โดยรวมข้อมูลและฟังก์ชันการทำงานเข้าไว้ด้วยกันในวัตถุ และกำหนด วิธีการติดต่อกันระหว่างวัตถุ (กิตติ ภัคดีวัฒนะกุล และกิตติพงษ์ กลมกล่อม, 2544; สมพร เรือง อ่อน, 2547)

หลักการสำคัญของแนวคิดเชิงวัตถุ (ชาลี วรรกุลพิพัฒน์ และเทพฤทธิ์ บันจิตวัฒนาวงศ์, 2544 : 15-18) มีดังนี้

1. ออบเจกต์ (Object) หมายถึง วัตถุหรือสิ่งที่ประกอบด้วยสถานะหรือข้อมูล และฟังก์ชัน การทำงานหรือพฤติกรรม หรือสิ่งที่เราสนใจซึ่งอาจจับต้อง ได้หรือจับต้องไม่ได้ โดยเป็นสิ่งที่แทน คน สถานที่ เหตุการณ์ หรือทรานเซคชันก็ได้

2. คลาส (Class) หมายถึง กลุ่มของออบเจกต์ที่มีโครงสร้างพื้นฐานพฤติกรรมเดียวกัน ดังนั้นออบเจกต์ที่มีคุณสมบัติลักษณะเดียวกันนี้จะรวมกลุ่มอยู่ในคลาสเดียวกัน จึงกล่าวได้ว่า คลาสคือต้นแบบข้อมูลที่มีไว้เพื่อสร้างออบเจกต์ ประกอบด้วย ชื่อของคลาส แอตทริบิวต์ (Attribute) และ โอเปอเรชัน (Operation)

3. เอ็นแคปซูลชัน (Encapsulation) หมายถึง การจัดกลุ่มองค์ความคิดที่คล้ายคลึงกันเข้า เป็นหน่วยเดียวกัน เพื่อทำการอ้างถึงด้วยชื่อเดียวกัน ในเชิงซอฟต์แวร์โปรแกรมเมอร์ได้มีแนวคิด

ไม่ว่าการณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการจัดรูปแบบการซ้ำดังกล่าวเข้ามาอยู่ในสิ่งเดียวกัน เพื่อความง่ายในการทำความเข้าใจตัวโปรแกรมและช่วยประหยัดพื้นที่หน่วยความจำด้วย

4. อินสแตนซ์ (Instance) สำหรับออบเจกต์ที่ถูกสร้างขึ้นจากคลาส เช่น คลาส A จะเรียกออบเจกต์ดังกล่าวว่า เป็นอินสแตนซ์ของคลาส A

5. แอตทริบิวต์ (Attribute) หมายถึง คุณสมบัติของออบเจกต์ หรืออาจใช้แสดงสถานะของออบเจกต์ ณ เวลาใดเวลาหนึ่ง

6. โอเปอเรชัน (Operation) หรือ เมธอด (Method) หมายถึง ฟังก์ชันพฤติกรรม (Behavior) หรือบริการที่ออบเจกต์สามารถกระทำให้ได้

7. ซิกเนเจอร์ (Signature) ประกอบด้วยชื่อของโอเปอเรชัน พารามิเตอร์ของโอเปอเรชัน และชนิดของข้อมูลที่ถูกส่งคืนจากโอเปอเรชัน

8. เมสเสจ (Message) ประกอบด้วยชื่อของโอเปอเรชันและค่าพารามิเตอร์ต่างๆ ของโอเปอเรชัน โดยส่วนใหญ่จะเป็นการเรียกใช้งานโอเปอเรชันของออบเจกต์

9. อินเทอร์เฟซ (Interface) หมายถึง ชุดของลายเซ็นทั้งหมดของคลาสใดคลาสหนึ่ง ซึ่งจะแสดงถึงสิ่งที่ออบเจกต์ของคลาสดังกล่าวสามารถตอบสนองได้

10. อินเฮอริทหรือการสืบทอดคุณสมบัติ (Inheritance) คือ การสืบทอดคุณสมบัติจากวัตถุหนึ่งไปยังวัตถุหนึ่ง ทำให้มีโครงสร้างที่เป็นระบบระเบียบ ปรับเปลี่ยนได้ง่าย และสามารถนำซอฟต์แวร์บางส่วนที่มีอยู่เดิมกลับมาใช้ใหม่ได้ ทั้งยังสามารถเพิ่มคุณสมบัติที่ผู้พัฒนาต้องการเข้าไป ทำให้การพัฒนาซอฟต์แวร์ใหม่ๆ ทำได้รวดเร็วขึ้น

หลักของการสืบทอดคุณสมบัติจะทำเป็นลำดับชั้น ทำให้ความสัมพันธ์ระหว่างวัตถุมีความชัดเจนยิ่งขึ้น ซึ่งการสืบทอดคุณสมบัตินี้จะทำให้เกิดซบคลาส (Sub class) โดยคลาสที่ให้กำเนิดเรียกว่า “พารেন্টคลาส” (Parent class) หรือ “ซูเปอร์คลาส” (Super class) และการสืบทอดคุณสมบัตินี้สามารถสืบทอดคุณสมบัติจากวัตถุมากกว่าหนึ่งวัตถุได้ เรียกว่า “มัลติเพิล อินเฮอริทเทนซ์” (Multiple inheritance)

11. ความสัมพันธ์ระหว่างคลาสหรือออบเจกต์ (Relationships) แบ่งได้เป็น 3 ประเภทคือ

- ความสัมพันธ์แบบพึ่งพา (Dependency) เมื่อฝ่ายหนึ่งถูกพึ่งพิงมีการเปลี่ยนแปลงจะก่อให้เกิดผลกระทบกับอีกฝ่ายหนึ่งที่เป็นฝ่ายพึ่งพิง
- ความสัมพันธ์แบบสืบทอดคุณสมบัติ (Inheritance) เช่น คลาสแม่กับคลาสลูก
- ความสัมพันธ์แบบเกี่ยวพัน (Association) จะมีการกำหนดถึงบทบาทของแต่ละฝ่ายที่ร่วมความสัมพันธ์กัน

2.4.2 ภาษายูเอ็มแอล (Unified Modeling Language)

ยูเอ็มแอลเป็นภาษาสัญลักษณ์ที่ใช้ในการอธิบายแบบจำลองของระบบ มีเครื่องมือสร้าง

รูปภาพกราฟิกและเทคนิคให้เห็นอย่างชัดเจน สามารถแบ่งโคแอมแกรมหลักๆ ที่สำคัญได้ดังนี้ การค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ยูสเคสไดอะแกรม (Use Case Diagram)

ยูสเคสไดอะแกรมเป็นเทคนิคในการ โมเดลความต้องการของผู้ใช้ รวมถึงการบรรยายความสามารถของระบบ ยูสเคสไดอะแกรมเป็นไออะแกรมมาตรฐานที่สามารถทำความเข้าใจได้ง่าย จึงส่งผลดีต่อการสื่อสารระหว่างผู้พัฒนากับผู้ใช้งานระบบ และกับพัฒนาด้วยตนเอง ยูสเคสไดอะแกรมได้รับความสนใจเป็นจำนวนมากจากผู้พัฒนาโปรแกรมประเภทเชิงวัตถุ อีกทั้งยังอำนวยความสะดวกในการพัฒนาโปรแกรม จะทำให้เห็นภาพชัดเจนของปฏิริยาระหว่างผู้ใช้ระบบงานกับระบบสารสนเทศ เป็นแผนผังที่แสดงความต้องการของระบบว่าในระบบนั้นมีการดำเนินงานอะไรบ้าง ระบบหรือแผนผังยูสเคสไดอะแกรมจะประกอบด้วย

- แอคเตอร์ (Actor) จะใช้สัญลักษณ์เป็นรูปคน โดยแอกเตอร์จะหมายถึง บุคคลหรือสิ่งที่ยอยู่นอกระบบ จะแสดงถึงผู้ใช้งานในระบบ โดยมีวัตถุประสงค์และเป้าหมายเฉพาะในการติดต่อกับระบบอย่างใดอย่างหนึ่ง โดยแอกเตอร์จะมีการแลกเปลี่ยนข่าวสาร (Message) กับระบบ
- ยูสเคส (Use Case) จะใช้สัญลักษณ์เป็นรูปวงรี โดยยูสเคสจะหมายถึง กิจกรรมหลักๆ ที่เกิดขึ้นภายในระบบซึ่งอาจจะเป็นกิจกรรมการกระทำ หรือเหตุการณ์ที่เกิดขึ้นระหว่างผู้ใช้กับระบบ
- ความสัมพันธ์ (Relationships) เป็นความสัมพันธ์ระหว่างแอกเตอร์กับยูสเคส หรือระหว่างยูสเคสด้วยกันเอง

2. คลาสไดอะแกรม (Class Diagram)

เป็นแผนภาพที่ใช้แสดงคลาสดและความสัมพันธ์ในแง่ต่างๆ ระหว่างคลาสเหล่านั้น สัญลักษณ์ที่ใช้แทนคลาสดิอะแกรมมีลักษณะเป็นสี่เหลี่ยมผืนผ้า โดยมีองค์ประกอบของคลาสดิอะแกรม 3 ส่วน คือ ชื่อของคลาสดิอะแกรม, แอตทริบิวต์ของคลาสดิอะแกรม และ โอเปอเรชันของคลาสดิอะแกรม ในคลาสดิอะแกรมจะมีการกำหนดตัวเลขความสัมพันธ์ โดยระบุเป็นตัวเลขหรือช่วงของตัวเลขในรูปแบบค่าต่ำสุดและค่าสูงสุดไว้ที่ด้านปลายของเส้นแสดงความสัมพันธ์ ถ้าเป็นตัวเลขจำนวนเดียว หมายถึง ค่าที่แน่นอน ถ้าเป็นช่วง หมายถึง ค่าที่เป็นไปได้ ถ้าหมายถึงจำนวนใดๆ จะใช้สัญลักษณ์ดอกจัน (*)

3. ซีควเอนซ์ไดอะแกรม (Sequence Diagram)

เป็นแผนภาพที่แสดงลำดับเหตุการณ์หรือการทำงานระหว่างออบเจกต์ต่างๆ ที่เกิดขึ้นระหว่างวัตถุหนึ่งกับอีกวัตถุหนึ่ง นักวิเคราะห์ระบบอาจใช้ซีควเอนซ์ไดอะแกรมแสดงผลลัพธ์ที่เป็นไปได้ทั้งหมด หรือมุ่งเน้นในแต่ละเหตุการณ์ ซีควเอนซ์ไดอะแกรมประกอบด้วย

- คลาส (Class) ใช้สัญลักษณ์สี่เหลี่ยมผืนผ้าซึ่งมีชื่ออยู่ข้างใน การรับหรือส่งคำสั่งจะแสดงอยู่ที่ส่วนบนสุดของซีควเอนซ์ไดอะแกรม
- ไลฟไลน์ (Lifeline) สัญลักษณ์เป็นเส้นไข่ปลา หมายถึง ระยะเวลาซึ่งออบเจกต์ที่อยู่ข้างบนมาปฏิสัมพันธ์กับออบเจกต์อีกอันหนึ่ง ซึ่งอยู่ภายใต้ยูสเคสเดียวกัน

- เมสเซจ (Message) สัญลักษณ์ของคำสั่งเป็นเส้นซึ่งชี้อยู่ระหว่างออบเจกต์ทั้งสอง โดยแต่ละคำสั่งจะใช้สัญลักษณ์เป็นเส้นมีหัวลูกศรซึ่งมีข้อความเพื่ออธิบายคำสั่งนั้นๆ

4. แอ็กทิวิตีไดอะแกรม (Activity Diagram)

แสดงขั้นตอนการทำงานของยูสเคสเช่นเดียวกับซีควเอนซ์ไดอะแกรม แต่จะเน้นไปที่งานย่อยของวัตถุ ซึ่งอาจเป็นการเจาะจงไปที่งานๆ ของวัตถุนั้น เป็นลักษณะแสดงถึงการออกคำสั่งให้เกิดการกระทำต่างๆ แอ็กทิวิตีไดอะแกรมจะเปลี่ยนสถานะได้โดยไม่ต้องมีเหตุการณ์ที่กำหนดไว้ในไดอะแกรมมาก่อน แต่จะเปลี่ยนสถานะเองตามกระบวนการทำงานคล้ายกับผังงาน (Flow Chart) แอ็กทิวิตีไดอะแกรมประกอบด้วย

- จุดเริ่มต้น (Start) เป็นจุดเริ่มต้นของกิจกรรม แสดงด้วยวงกลมทึบ
- กิจกรรม (Activity) ใช้สัญลักษณ์สี่เหลี่ยมมุมมน เพื่อแสดงลำดับการทำกิจกรรมโดยอธิบายกิจกรรมไว้ในสี่เหลี่ยมนั้นๆ แต่หากกิจกรรมที่ต้องการตัดสินใจจะแสดงด้วยสี่เหลี่ยมข้าวหลามตัด
- เส้นลูกศรหรือทริกเกอร์ (Trigger) ใช้เชื่อมกิจกรรมตามลำดับ
- จุดจบ (End) เป็นจุดจบของกิจกรรม แสดงด้วยวงกลมโปร่งล้อมรอบวงกลมทึบข้างใน

บทที่ 3

คุณลักษณะและความต้องการของระบบ

3.1 เครื่องมือที่ใช้ในขั้นตอนต่างๆ และขอบเขตการทำงานของเครื่องมือ

ตารางที่ 3.1 ตารางแสดงขั้นตอนการบริหาร โครงการและเครื่องมือ

No.	Module	Step Wise Mapping	Scope of work	Tools
I	Project Selection	Step 0 Select project	<p>ระบบสามารถเก็บข้อมูลพื้นฐานของโครงการเพื่อให้ทราบถึงจุดมุ่งหมายและรายละเอียดต่างๆของการดำเนินโครงการ โดยมีข้อมูลต่างๆเช่น</p> <ol style="list-style-type: none"> 1. Project code 2. Project name 3. Project description 4. Project category 5. Stakeholder and information <ol style="list-style-type: none"> 5.1. Customer Project Owner 5.2. Customer Project Manager 5.3. Customer information 5.4. Project Manager 6. Project status 7. Additional information (Attach File) <p>นอกจากข้อมูลต่างๆที่เกี่ยวข้องกับโครงการแล้ว ระบบสามารถประเมินความเป็นไปได้ทางธุรกิจต่างๆเพื่อเป็นข้อประกอบในการพิจารณาเลือกดำเนินโครงการเช่น</p>	1. Project Information Management

คือมูลค่าของเงินในปัจจุบันซึ่งจะต้องมีการร่วมกับอัตราเงินเฟ้อในแต่ละปีเพื่อให้ทราบมูลค่าที่แท้จริงของโครงการ

$$PV = \frac{\text{Amount}}{(1 + \text{Interest rate})^n}, n = \text{number of year in the future} \quad (3.1)$$

3. มูลค่าปัจจุบันสุทธิ (Net present value (NPV))

คือกำไรหรือขาดทุนหลังจากหักหรือรวมรายการพิเศษแล้ว ซึ่งคำนวณจากมูลค่าจริงในปัจจุบัน

$$NPV = PV \text{ Benefit} - PV \text{ Cost}$$

4. ผลตอบแทนจากการลงทุน (Return on investment (ROI))

คือผลตอบแทนที่ได้จากการดำเนินโครงการ

$$ROI = \frac{\text{Total Benefit NPV} - \text{Total Cost NPV}}{\text{Total Cost NPV}} \quad (3.2)$$

5. จุดคุ้มทุน (Break-Event Point (BEP))

คือระยะเวลาเป็นจำนวนปีที่ NPV เป็นค่าบวกครั้งแรก

$$BEP = BEP \text{ Year} + \frac{\text{Yearly NPV}^* - \text{Cumulative NPV}}{\text{Yearly NPV}^*} \quad (3.3)$$

3.3.3 Project scope management

คือเครื่องมือที่ใช้บริหารขอบเขตความต้องการของโครงการเช่น การกำหนดขอบเขตความต้องการของงาน ผู้ที่เกี่ยวข้องกับการดำเนินงาน

1. Project objective
2. Project scope
 - 2.1 Nonfunctional requirement
 - 2.2 Function Requirement

3.3.4 Project Infrastructure Management

คือเครื่องมือที่ใช้ในแสดงถึงแผนการดำเนินโครงการในระดับสูงเช่น แผนการดำเนิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.9 Resource Allocation

1. Resource requirement list

คือการกำหนดทรัพยากรต่างๆ ที่จะต้องใช้ในแต่ละกิจกรรมของโครงการ ประกอบด้วย

1. คนที่ทำงานในแต่ละกิจกรรม
ซึ่งแสดงด้วย Resource Histogram
2. เครื่องมือที่ใช้ในแต่ละกิจกรรมเช่น
 - คอมพิวเตอร์และอุปกรณ์ต่อพ่วงต่างๆ
 - โปรแกรมต่างๆที่เกี่ยวข้องกับการพัฒนา
3. อุปกรณ์สิ้นเปลืองต่างๆเช่น
 - กระดาษ
 - แผ่นซีดีสำหรับบันทึกข้อมูล
4. พื้นที่ในการทำงานซึ่งต้องจัดเตรียมเพื่อให้การทำงานเป็นไปอย่างราบรื่น
5. บริการที่เกี่ยวข้องกับการทำงานเช่น
 - ระบบการสเคลื่อนย้ายขององค์กรที่ใช้ในการทำงาน

3.3.10 Project Monitoring and Control

1. Time Sheet

เป็นการให้ผู้ที่ทำหน้าที่ในการดำเนินโครงการ ในขั้นตอนหรือกิจกรรมต่างๆเมื่อได้ทำกิจกรรมนั้นแล้ว ดำเนินการบันทึกการทำกิจกรรมนั้นๆตามการทำงานจริงเพื่อการติดตามการดำเนินโครงการอย่างมีประสิทธิภาพ

2. Progress report

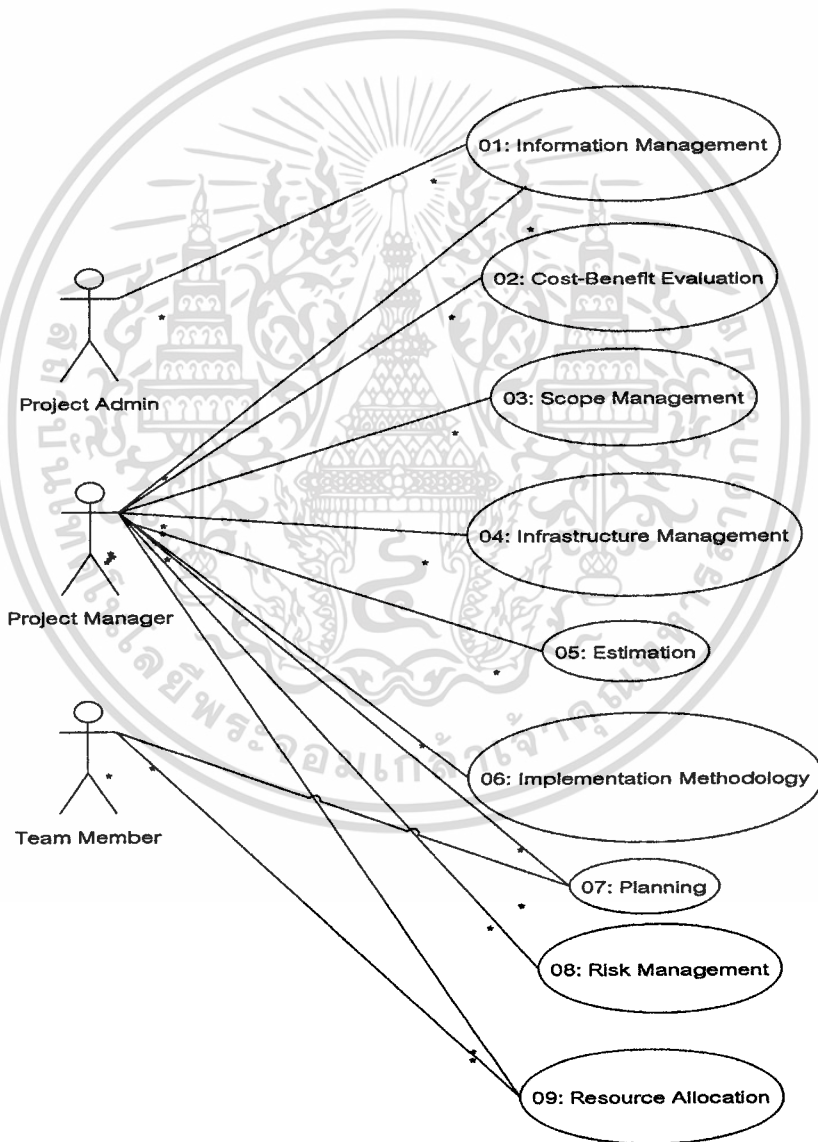
เป็นการให้ผู้ที่ทำหน้าที่ในการดำเนินโครงการ ในขั้นตอนต่างๆหรือกิจกรรมต่างๆ บันทึกความก้าวหน้าในการดำเนินการในกิจกรรมที่ตนเองรับผิดชอบเพื่อการติดตามการดำเนินโครงการอย่างมีประสิทธิภาพ

บทที่ 4

การวิเคราะห์ระบบงาน

4.1 การวิเคราะห์ระบบงาน

จากการวิเคราะห์ขั้นตอนบริหาร โครงการสามารถสร้างยูสเคสไดอะแกรมเพื่อแสดงให้เห็นภาพรวมของระบบที่สร้างขึ้น เป็นไดอะแกรมที่ใช้อธิบายเครื่องมือทั้งหมดของการบริหารโครงการ



รูปที่ 4.1 ยูสเคสไดอะแกรมของระบบบริหารโครงการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

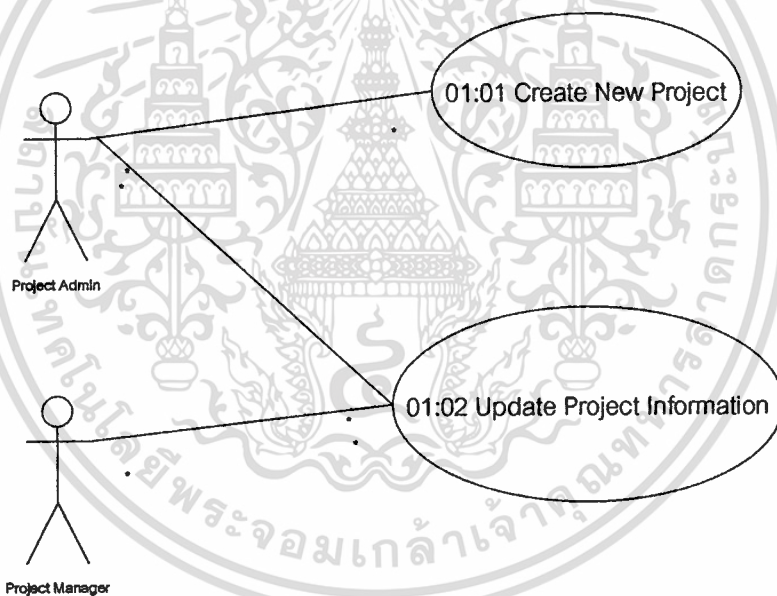
4.1.1 คำอธิบายความหมายแอกเตอร์ของระบบ

Project Admin คือ ผู้ที่ดูแลข้อมูลของระบบมีหน้าที่ในการสร้างโครงการใหม่และกำหนดให้ผู้ใดมีสิทธิ์บริหารโครงการนั้น Project Manager คือผู้บริการโครงการทำหน้าที่ให้เครื่องมือในการบริหารโครงการเพื่อประโยชน์ในการบริหารโครงการนั้นๆ Team Member คือผู้ที่เกี่ยวข้องในโครงการเช่น นักวิเคราะห์ระบบ ทีมพัฒนาระบบ เป็นต้น

4.2 รายละเอียดยูสเคสของแต่ละระบบงาน

4.2.1 Project Information Management

การบริหารประกอบด้วย 2 ยูสเคสทำหน้าที่หลักในการบริการข้อมูลเบื้องต้นของโครงการและถูกใช้งานโดย Project Admin และ Project Manager



รูปที่ 4.2 ยูสเคสไดอะแกรมของระบบ Project Information Management

ตารางที่ 4.1 รายละเอียดยูสเคส Create New Project

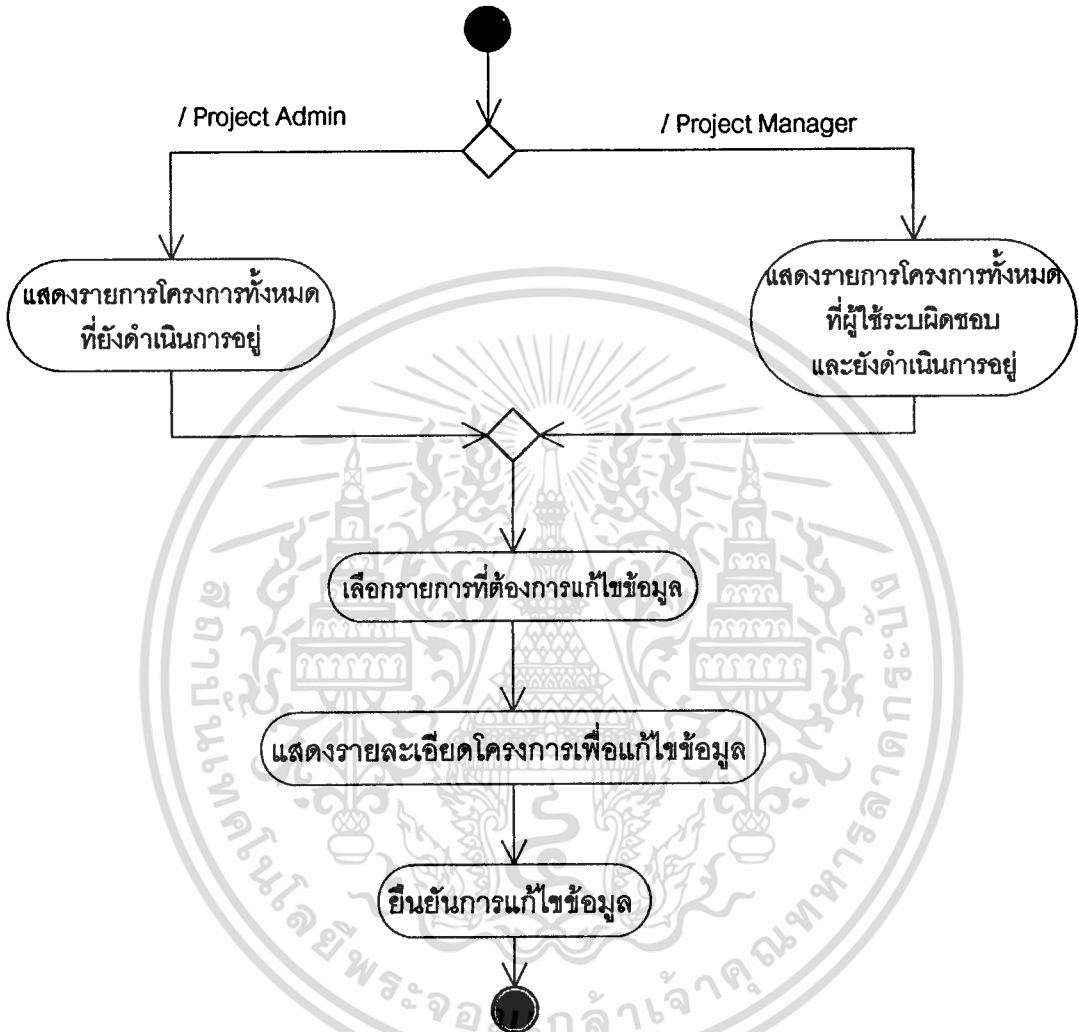
Use Case Name : Create New Project	ID : 01:01	Importance Level : Height
Primary Actor : Project Admin	Use Case Type : Essential	
Stakeholders and Interests :		
Project Admin: ทำการสร้างโครงการใหม่ในระบบเพื่อให้ผู้บริหารโครงการนำไปใช้บริหารโครงการ		
Brief Description :		
- เมื่อต้องการสร้างโครงการใหม่ Project Admin จะเป็นผู้สร้างโครงใหม่เพื่อกำหนด Project Code ,ชื่อ โครงการ และผู้บริหารโครงการเพื่อให้ผู้บริหารโครงการนำไปใช้บริหารโครงการต่อไป		
Trigger : เพื่อสร้าง โครงการใหม่		
Type :		
Relationships :		
Association : Project Admin		
Include :		
Extend :		
Generalization :		
Normal Flow of Events :		
<ol style="list-style-type: none"> 1. สร้างโครงการใหม่ 2. กรอกรายละเอียดข้อมูลของ โครงการ Project Code และ ชื่อ โครงการ 3. ยืนยันการสร้างโครงการ 		
Sub flows :		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 รายละเอียดยูสเคส Update Project Information

Use Case Name : Update Project Information	ID : 01:02	Importance Level : Height
Primary Actor : Project Admin, Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Admin: ทำการแก้ไขข้อมูลโครงการในส่วนของชื่อโครงการ Project Manager: ทำการแก้ไขข้อมูลต่างๆของโครงการที่ตนรับผิดชอบการยกเว้นชื่อโครงการ		
Brief Description : - เมื่อผ่านขั้นตอนการสร้างโครงการใหม่ Project Manager จะทำการเพิ่มเติมแก้ไขข้อมูลโครงการต่างๆเพื่อให้ข้อมูลโครงการสมบูรณ์		
Trigger : เพื่อแก้ไขข้อมูลโครงการ Type :		
Relationships : Association : Project Admin, Project Manager Include : Extend : Generalization :		
Normal Flow of Events : 1. ตรวจสอบชนิดผู้ใช้งาน S-1 2. แสดงรายการ โครงการที่ยังคงดำเนินการอยู่ 3. เลือกโครงการที่ต้องการแก้ไขข้อมูล 4. แสดงรายละเอียดโครงการเพื่อแก้ไขข้อมูล 5. ยืนยันการแก้ไขข้อมูล		
Sub flows : S-1: 1. ถ้าผู้ใช้เป็น Project Admin ให้แสดงรายการ โครงการทั้งหมดที่ยังคงดำเนินการอยู่ 2. ถ้าผู้ใช้เป็น Project Manager ให้แสดงรายการที่ผู้ใช้มีสิทธิ์แก้ไขข้อมูล		
Alternate/Exceptional Flows :		

เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Update Project Information ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทวิตีไดอะแกรมดังนี้



รูปที่ 4.3 เอกทวิตีไดอะแกรมของ Update Project Information

4.2.2 Project Cost-Benefit Evaluation

การวิเคราะห์ประกอบด้วย 6 ยูสเคสทำหน้าที่หลักในการบริการข้อมูลเบื้องต้นของโครงการและถูกใช้งานโดย Project Manager

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

คุณลักษณะและความต้องการของระบบ

3.1 เครื่องมือที่ใช้ในขั้นตอนต่างๆ และขอบเขตการทำงานของเครื่องมือ

ตารางที่ 3.1 ตารางแสดงขั้นตอนการบริหาร โครงการและเครื่องมือ

No.	Module	Step Wise Mapping	Scope of work	Tools
I	Project Selection	Step 0 Select project	<p>ระบบสามารถเก็บข้อมูลพื้นฐานของโครงการเพื่อให้ทราบถึงจุดมุ่งหมายและรายละเอียดต่างๆของการดำเนินโครงการ โดยมีข้อมูลต่างๆเช่น</p> <ol style="list-style-type: none"> 1. Project code 2. Project name 3. Project description 4. Project category 5. Stakeholder and information <ol style="list-style-type: none"> 5.1. Customer Project Owner 5.2. Customer Project Manager 5.3. Customer information 5.4. Project Manager 6. Project status 7. Additional information (Attach File) <p>นอกจากข้อมูลต่างๆที่เกี่ยวข้องกับโครงการแล้ว ระบบสามารถประเมินความเป็นไปได้ทางธุรกิจต่างๆเพื่อเป็นข้อประกอบในการพิจารณาเลือกดำเนินโครงการเช่น</p>	1. Project Information Management

ตารางที่ 3.1 (ต่อ)

No.	Module	Step Wise Mapping	Scope of work	Tools
			1 Cost-benefits Evaluation 1.1 Net profit (NP) 1.2 Payback period 1.3 Return on investment (ROI) 1.4 Net present value (NPV)	2. Project Cost-Benefit Evaluation
		Step 1	ระบบสามารถเก็บข้อมูลเกี่ยวกับวัตถุประสงค์และขอบเขตของการดำเนินงานรวมถึงผู้ที่มีส่วนได้ส่วนเสียในการดำเนินโครงการ โดยมีข้อมูลต่างๆเช่น 1. Project scope 2. Project objective 3. Stakeholder	3. Project scope management
		Step 2	ระบบสามารถเก็บข้อมูลเกี่ยวกับโครงสร้างพื้นฐานที่เกี่ยวข้องกับโครงการเช่น แผนธุรกิจในการดำเนินโครงการ แผนด้านการใช้เทคโนโลยี รวมถึงมาตรฐานต่างๆในการดำเนินโครงการและโครงสร้างของผู้ดำเนินการพัฒนาโครงการ โดยมีข้อมูลเช่น 1. Business strategic planning 2. Technology planning 3. Project Organization	2. Project Infrastructure Management

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 (ต่อ)

No.	Module	Step Wise Mapping	Scope of work	Tools
		Step 3	ระบบสามารถประมาณการขนาดและระยะเวลาในการดำเนินโครงการจากข้อมูลในขั้นตอนที่ 0-2	1. Project Estimation 1.1 Function Point
			ระบบสามารถบันทึกการเลือกวิธีการดำเนินโครงการที่เหมาะสมเพื่อให้ได้ประสิทธิภาพสูงสุดในการดำเนินโครงการ	2. Implementation Methodology 2.1 Project implementation methodology
II	Project Planning and Execution	Step 4-8	ระบบสามารถ	1 Project Planning 1.1 Work Breakdown Structure (WBS) 1.2 Activity on arrow chart 2 Risk Management 2.1 Risk exposure assessment 2.2 Expect time and standard deviation 2.3 PERT chart 3 Resource Allocation 3.1 Resource requirement list 3.2 Bar chart Resource Histogram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 (ต่อ)

No.	Module	Step Wise Mapping	Scope of work	Tools
III	Project Monitoring and Control	Step 9-10		1. Time Sheet 2. Progress report

3.2 รายละเอียดการออกแบบเครื่องมือ

3.2.1 Project Information Management

เครื่องมือนี้เป็นเครื่องมือที่เก็บข้อมูลเริ่มต้นของ โครงการ ซึ่งเป็นข้อมูลพื้นฐานเช่น

1. รหัสโครงการ (Project code)
2. ชื่อโครงการ (Project name)
3. รายละเอียดโครงการ (Project description)
4. ประเภทโครงการ (Project category) เช่น โครงการพัฒนาซอฟต์แวร์ตามความต้องการ โครงการพัฒนาซอฟต์แวร์จากซอฟต์แวร์แพ็คเกจ
5. ผู้เกี่ยวข้องในโครงการ (Stakeholder and information)
 - 5.1 ลูกค้าผู้รับผิดชอบโครงการ (Customer Project Owner)
 - 5.2 ลูกค้าผู้บริหารโครงการ (Customer Project Manager)
 - 5.3 ข้อมูลลูกค้า (Customer information)
 - 5.4 ผู้บริหารโครงการ (Project Manager)
6. สถานะภาพโครงการ (Project status) เช่น ช่วงการนำเสนอข้อมูลโครงการ ช่วงการดำเนินโครงการ ช่วงเสร็จสิ้นการดำเนินโครงการ
7. ข้อมูลเพิ่มเติม (Additional information) เช่น ข้อมูลรายละเอียดความต้องการของลูกค้า

3.2.2 Project Cost-Benefit Evaluation

1. กำไรสุทธิ (Net profit (NP))

คือกำไรหรือขาดทุนหลังจากหักหรือรวมรายการพิเศษแล้ว ซึ่งคำนวณจาก

$$NP = \text{รายได้ของโครงการทั้งหมด} - \text{เงินลงทุนทั้งหมด}$$

2. มูลค่าเงินในปัจจุบัน (Present value (PV))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือมูลค่าของเงินในปัจจุบันซึ่งจะต้องมีการร่วมกับอัตราเงินเฟ้อในแต่ละปีเพื่อให้ทราบมูลค่าที่แท้จริงของโครงการ

$$PV = \frac{\text{Amount}}{(1 + \text{Interest rate})^n}, n = \text{number of year in the future} \quad (3.1)$$

3. มูลค่าปัจจุบันสุทธิ (Net present value (NPV))

คือกำไรหรือขาดทุนหลังจากหักหรือรวมรายการพิเศษแล้ว ซึ่งคำนวณจากมูลค่าจริงในปัจจุบัน

$$NPV = PV \text{ Benefit} - PV \text{ Cost}$$

4. ผลตอบแทนจากการลงทุน (Return on investment (ROI))

คือผลตอบแทนที่ได้จากการดำเนินโครงการ

$$ROI = \frac{\text{Total Benefit NPV} - \text{Total Cost NPV}}{\text{Total Cost NPV}} \quad (3.2)$$

5. จุดคุ้มทุน (Break-Event Point (BEP))

คือระยะเวลาเป็นจำนวนปีที่ NPV เป็นค่าบวกครั้งแรก

$$BEP = BEP \text{ Year} + \frac{\text{Yearly NPV}^* - \text{Cumulative NPV}}{\text{Yearly NPV}^*} \quad (3.3)$$

3.3.3 Project scope management

คือเครื่องมือที่ใช้บริหารขอบเขตความต้องการของโครงการเช่น การกำหนดขอบเขตความต้องการของงาน ผู้ที่เกี่ยวข้องกับการดำเนินงาน

1. Project objective
2. Project scope
 - 2.1 Nonfunctional requirement
 - 2.2 Function Requirement

3.3.4 Project Infrastructure Management

คือเครื่องมือที่ใช้ในแสดงถึงแผนการดำเนินโครงการในระดับสูงเช่น แผนการดำเนิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

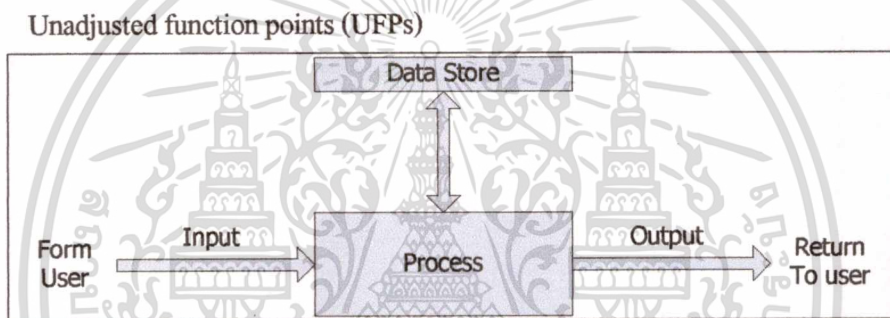
โครงการให้สอดคล้องกับการดำเนินธุรกิจ แผนการใช้เทคโนโลยี กระบวนการการพัฒนาโครงการ รวมถึง โครงสร้างของทีมงานในการพัฒนาโครงการ

1. Business strategic planning
2. Technology planning
3. Project Organization

3.3.5 Project Estimation

1. Function Point

การประเมินขนาดของโครงการด้วยวิธี Function Point ถูกแนะนำโดย CCTA (Central Computer and Telecommunication Agency) ซึ่งถูกใช้เป็นมาตรฐานใช้กับโครงการต่างๆ ของรัฐบาลอังกฤษ ซึ่งการประเมินด้วยวิธีนี้มีขั้นตอนการทำงานดังนี้



รูปที่ 3.1 แสดงแบบจำลองการทำงานของโปรแกรม

Step1: Total Unadjusted Function Point (TUFP)

ในขั้นตอนนี้ทำการคำนวณ Input, Output ในแต่ละขั้นตอนการทำงานเพื่อหาผลรวมของ TUFP ทั้งหมด โดยมีค่าความซับซ้อนของการทำงานต่างๆ (Complexity) ดังนี้

ตารางที่ 3.2 ตารางแสดงค่าความซับซ้อน

Function Type	Complexity		
	Low	Medium	High
Input	3	4	6
Outputs	4	5	7
Queries	3	4	6
Files	7	10	15
Program Interfaces	5	7	10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Step2: Total Processing Complexity (PC)

คือการกำหนดค่าความซับซ้อนของการพัฒนาเช่น Data Communication, Transaction rate, End-user efficiency ซึ่งส่วนต่างๆเหล่านี้ทำให้การพัฒนาใช้เวลาเพิ่มมากขึ้นและเป็นตัวแปรสำคัญของความสำเร็จของโครงการ โดยตัวแปรแต่ละตัวสามารถให้ค่าความซับซ้อนได้ตั้งแต่ 1-5 และมีได้ไม่เกิน 20 ตัวแปรกล่าวคือผลรวมของค่าความซับซ้อนเมื่อกำหนดให้แต่ละตัวแปรมีค่ามากที่สุดต้องไม่เกิน 100 ซึ่งสามารถยกตัวอย่างตัวแปรได้ดังนี้

ตารางที่ 3.3 ตารางแสดงค่าตัวแปรความซับซ้อน

No.	Complexity Parameter
1	Data Communication
2	Transaction rate
3	Heavy use configuration
4	End-user efficiency
5	Complex processing
6	Installation ease
7	Multiple sites
8	Performance
9	Distributed functions
10	Online data entry
11	Online update
12	Reusability
13	Operational ease
14	Extensibility

Step3: Adjusted Processing Complexity (APC)

$$APC = 0.65 + (0.01 \times [PC]) \quad (3.4)$$

Step4: Total Adjusted Function Point (TAFP)

$$TAFP = APC \times TAFP \quad (3.5)$$

Step5: Converting from Function Point to Lines of Code

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในขั้นตอนนี้เป็นการคำนวณ เพื่อเปลี่ยน TAFP เป็นจำนวนบรรทัดในการพัฒนาโปรแกรมตามชนิดของภาษาที่ใช้ตั้งข้อมูลข้างดังนี้

ตารางที่ 3.4 ตารางแสดงค่าจำนวนบรรทัดของแต่ละโปรแกรม

Language	Approximate Number of Lines of Code per Function Point
C	130
COBOL	110
Java	55
C++	50
C# .NET	30
Power Builder	15
HTML	15
Packages (e.g., Access, Excel)	10-40
Source: Capers Jones, Software Productivity Research, http://www.spr.com	

Step6: Find Effort by COCOMO

ในขั้นตอนการแปลงจำนวนบรรทัดของโปรแกรมที่ได้จากขั้นตอนที่ 5 มาเป็นจำนวนเดือนที่ใช้ในการพัฒนาโดยใช้คนหนึ่งคนโดยมีการคำนวณดังนี้

$$\text{Effort (in person-months)} = 1.4 \times \text{thousands of lines of code} \quad (3.6)$$

Step7: Estimate Times Required by COCOMO

ในขั้นตอนนี้เป็นการประมาณการเวลาในการพัฒนาที่ใช้จริง โดยมีการคำนวณดังนี้

$$\text{Schedule time (months)} = 3.0 \times \text{person-months}^{1/3} \quad (3.7)$$

3.3.6 Implementation Methodology

1. Project implementation methodology

โดยอ้างอิงข้อมูลตามตารางดังนี้

ตารางที่ 3.5 ตารางแสดงข้อมูลเปรียบเทียบการเลือกวิธีการดำเนิน โครงการ

Ability to Develop Systems	Structured Methodologies			RAD Methodologies		Agile Methodologies
	Waterfall	Parallel	Phased	Prototyping	Throwaway Prototyping	XP
with Unclear User Requirements	Poor	Poor	Good	Excellent	Excellent	Excellent
with Unfamiliar Technology	Poor	Poor	Good	Poor	Excellent	Poor
that are Complex	Good	Good	Good	Poor	Excellent	Poor
that are Reliable	Good	Good	Good	Poor	Excellent	Good
with a Short Time Schedule	Poor	Good	Excellent	Excellent	Good	Excellent
with Schedule Visibility	Poor	Poor	Excellent	Excellent	Good	Good

3.3.7 Project Planning

7.1 Work Breakdown Structure (WBS)

เป็นการแบ่งกิจกรรมต่างๆ ของโครงการออกเป็นส่วนย่อยๆ ซึ่งเป็นการวางแผนงานนั่นเอง การวางแผนงานนี้เป็นส่วนสำคัญอย่างมากต่อการดำเนิน โครงการถือเป็นกระบวนการที่เป็นหัวใจของโครงการ โดยการแบ่งกิจกรรมต่างๆ ออกเป็นส่วนย่อยตามขั้นตอนการดำเนิน โครงการ เช่น

7.2 ขั้นตอนการเตรียมโครงการ

7.3 ขั้นตอนการเก็บข้อมูลความต้องการของระบบ

7.4 ขั้นตอนการวิเคราะห์ระบบ

7.5 ขั้นตอนการพัฒนาาระบบ

7.6 ขั้นตอนการทดสอบระบบ

7.7 ขั้นตอนการติดตั้งระบบ

7.8 ขั้นตอนการนำระบบมาใช้งานจริง

7.9 ขั้นตอนการทำเอกสารของระบบ

7.10 ขั้นตอนการฝึกอบรมผู้ใช้งานและดูแลระบบ

ซึ่งในแต่ละกิจกรรมควรจะแบ่งย่อยลงให้ถึงระดับที่ในกิจกรรมหนึ่งๆ มีผู้รับผิดชอบเพียงคนเดียวซึ่งจะทำให้การติดตามการดำเนินงาน ไม่เกิดความยุ่งยาก ซึ่งประกอบด้วยรายละเอียดต่างๆ ที่จำเป็นเช่น ชื่อกิจกรรม ระยะเวลาเริ่มต้น ระยะเวลาสิ้นสุด ระยะเวลาดำเนินการ ผู้รับผิดชอบ กิจกรรม

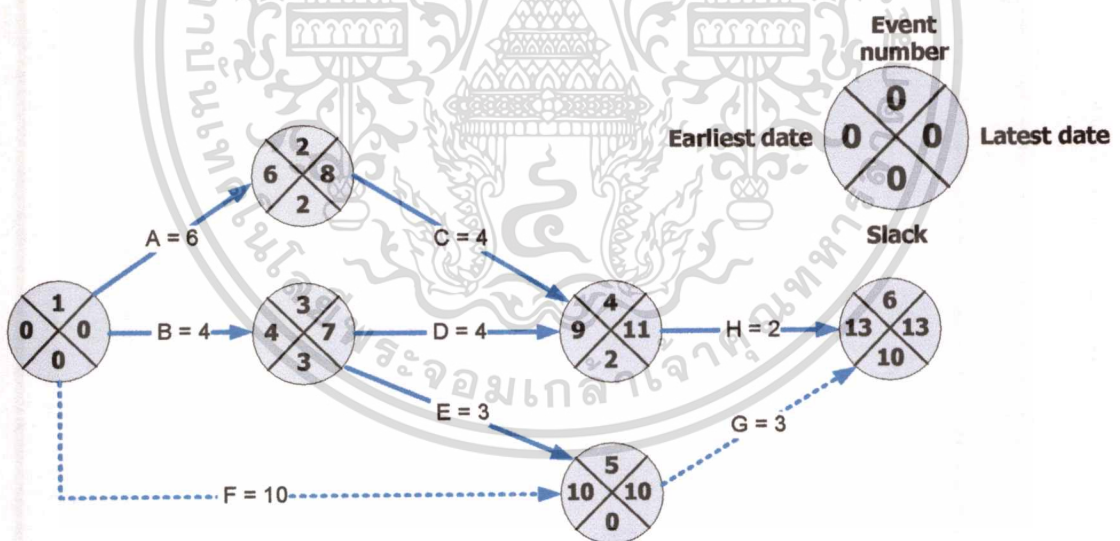
7.11 Activity on arrow chart

เป็นวิธีหนึ่งในการหาเส้นทางของกิจกรรมที่ใช้ระยะเวลาในการดำเนินการนานที่สุด เพื่อให้ทราบถึงกิจกรรมที่มีผลกระทบต่อระยะเวลาของโครงการซึ่งหมายความว่าถ้ากิจกรรมที่อยู่เบื้องหลังนั้นล่าช้าลงจะส่งผลให้ระยะเวลาของโครงการเพิ่มขึ้นหรือไม่ขึ้นอยู่กับระยะเวลาของกิจกรรมนั้นไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในเส้นทางที่ใช้ระยะเวลานานที่สุดกิจกรรมใดกิจกรรมหนึ่งไม่เป็นไปตามแผนงานที่วางไว้จะมีผลต่อระยะเวลาในการดำเนินโครงการ โดย Activity on arrow มีตัวอย่างดังนี้

ตารางที่ 3.6 ตารางแสดงระยะเวลาการดำเนินโครงการ

Activity	Duration (Weeks)	Earliest Start date	Latest Start date	Earliest Finish date	Latest Finish date	Total Float
A	6	0	2	6	8	2
B	4	0	3	4	7	3
C	3	6	8	9	11	2
D	4	4	7	8	11	2
E	3	4	7	7	10	3
F	10	0	0	10	10	0
G	3	10	10	13	13	0
H	2	9	11	11	13	2



Event number = ลำดับของกิจกรรม

Earliest date = ระยะเวลาที่สามารถเริ่มต้นกิจกรรมได้เร็วที่สุดมีหน่วยเป็นหน่วยเดียวดับ Duration

Latest date = ระยะเวลาที่ช้าที่สุดที่จะสิ้นสุดกิจกรรมมีหน่วยเป็นหน่วยเดียวดับ Duration

Slack = Latest Finish date – Earliest Start date (3.8)

Float = Latest Finish date – Earliest Finish date (3.9)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.8 Risk Management

1. Risk exposure assessment

เป็นการกำหนดความเสี่ยงโดยการการระบุนรายละเอียดต่างๆของความเสี่ยงดังแสดง

ตัวอย่างในตาราง

ตารางที่ 3.7 ตารางแสดงความเสี่ยง

Risk (describe as a real event)	Category	Priority	Probability [1,...,9]	Severity [0,...,5]	Risk Exposure	Impact on Project (describe in words)	Fighting Actions	Preventing Actions	When	Who	Status
A	Technical	10	5	3	15						

2. Expect time and standard deviation

เป็นประเมินระยะเวลาในการดำเนินโครงการ โดยประเมินความเสี่ยงของโครงการที่มีผลต่อกิจกรรมต่างๆของโครงการซึ่งประกอบด้วยรายละเอียดดังตาราง

2.1 Activity คือชื่อกิจกรรม

2.2 Optimistic คือระยะเวลาที่สั้นที่สุดการดำเนินกิจกรรม

2.3 Most likely คือระยะเวลาตามแผนงานที่วางไว้

2.4 Pessimistic คือระยะเวลายาวนานที่สุดในการดำเนินกิจกรรม

2.5 Expected คือระยะที่เป็นไปได้มากที่สุดในการดำเนินกิจกรรม

$$\text{Expected } (t_e): t_e = \frac{a + 4m + b}{6} \quad (3.10)$$

2.6 Standard deviation คือค่าเบี่ยงเบนมาตรฐานของการประเมินระยะเวลา

$$\text{Standard deviation } (s): s = \frac{b - a}{6} \quad (3.11)$$

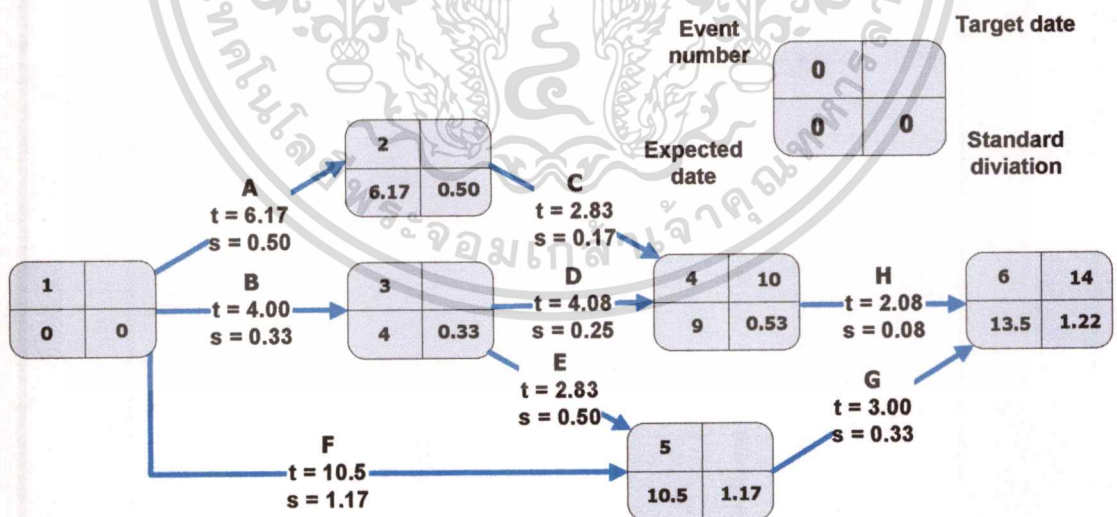
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.8 ตารางแสดงจำนวนวันแต่ละกิจกรรม

Activity	Activity duration (days or weeks)				
	Optimistic (a)	Most likely (m)	Pessimistic (b)	Expected (t_e)	Standard deviation (s)
A	5	6	8	6.17	0.50
B	3	4	5	4.00	0.33
C	2	3	3	2.83	0.17
D	3.5	4	5	4.08	0.25
E	1	3	4	2.83	0.50
F	8	10	15	10.50	1.17
G	2	3	4	3.00	0.33
H	2	2	2.5	2.08	0.08

3. PERT chart

คือการผลจากการคำนวณ Expected time and standard deviation เพื่อทำการหาเส้นทางที่ยาวที่สุดในกรณีการดำเนินโครงการ โดยหาจาก Expected date ที่มีการคิดคำนวณด้านความเสี่ยงของโครงการควบคู่ไปด้วยดังตัวอย่าง

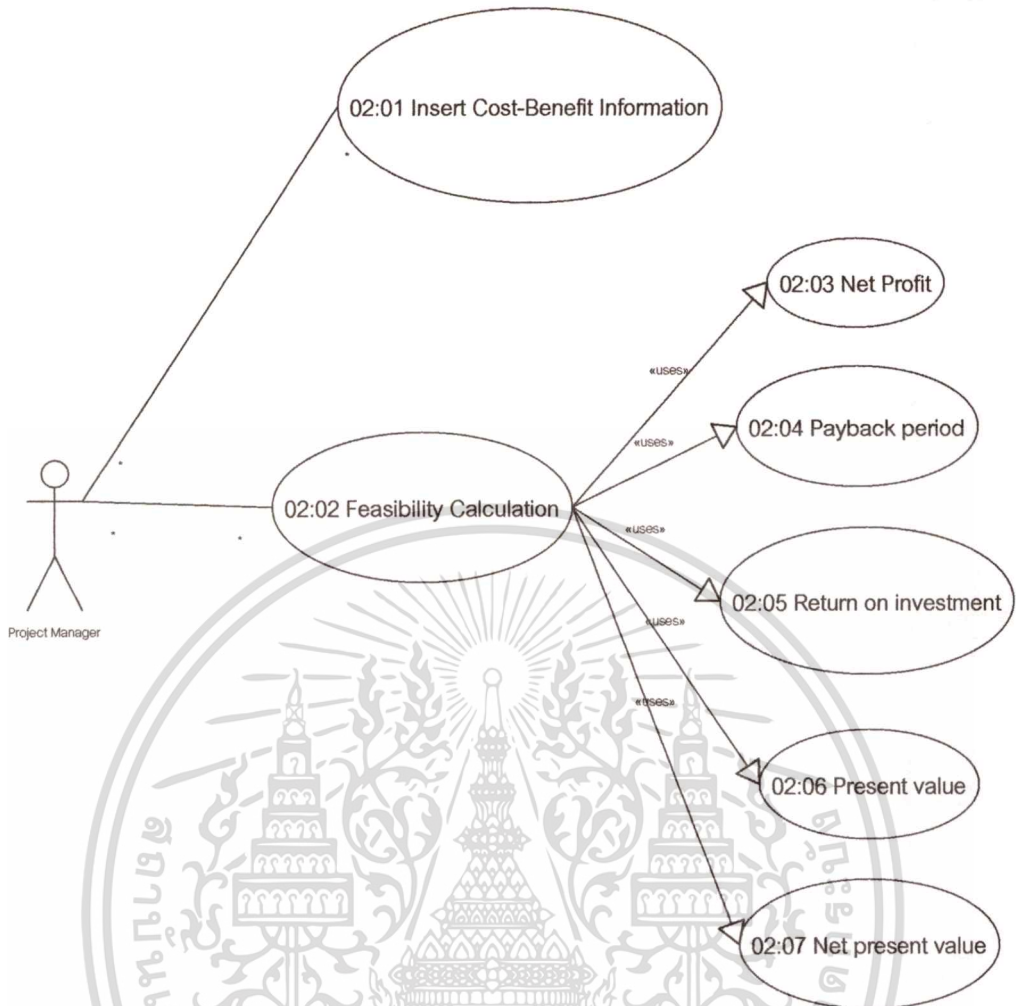


ตัวอย่างการคำนวณ S ของกิจกรรมที่ 4 เป็นดังนี้

$$s = \sqrt{0.33^2 + 0.25^2} = 0.53$$

(3.12)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 ยูสเคสไดอะแกรมของระบบ Project Cost-Benefit Evaluation

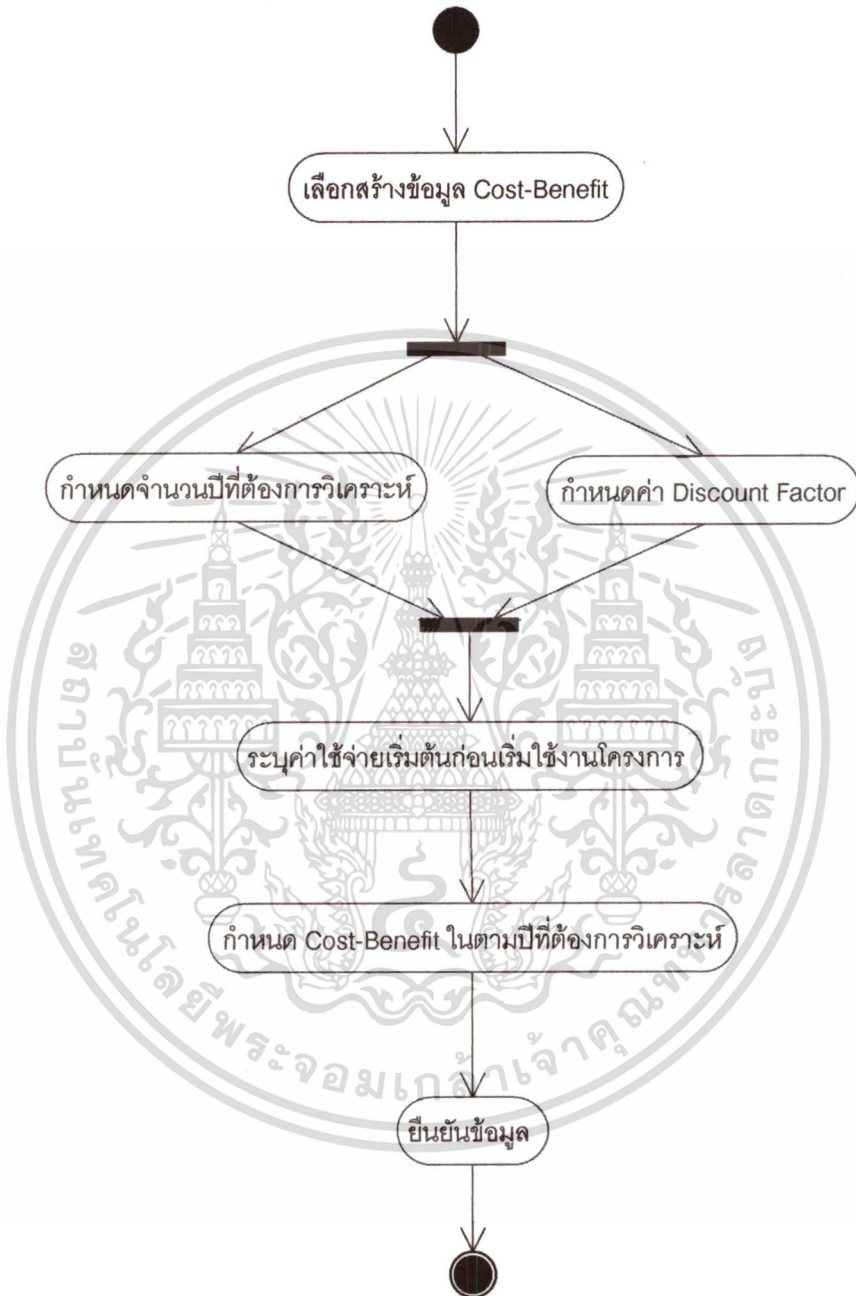
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 รายละเอียดยูสเคส Insert Cost-Benefit Information

Use Case Name : Insert Cost-Benefit Information	ID : 02:01	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: ทำการเพิ่มข้อมูลค่าใช้จ่ายทั้งหมดที่ใช้ในโครงการเพื่อนำมาเป็นต้นทุนในการคำนวณ		
Brief Description : - เพื่อการวิเคราะห์ความคุ้มค่าในการทำโครงการ จัดให้มีการเก็บข้อมูลด้านค่าใช้จ่ายและผลจากการใช้งานโครงการหลังพัฒนาเสร็จสิ้นแล้วในรูปตัวเงิน		
Trigger : เพื่อเก็บข้อมูลการวิเคราะห์ความคุ้มค่าของโครงการ Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : <ol style="list-style-type: none"> 1. เลือกสร้างข้อมูล Cost-Benefit 2. กำหนดจำนวนปีที่ต้องการวิเคราะห์ 3. กำหนดค่า Discount Factor 4. ระบุค่าใช้จ่ายเริ่มต้นก่อนเริ่มใช้งานโครงการ 5. กำหนด Cost-Benefit ในตามปีที่ต้องการวิเคราะห์ 6. ยืนยันข้อมูล 		
Sub flows :		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยุทธศาสตร์ Insert Cost-Benefit Information ทำงานตามที่ได้อธิบายในรายละเอียดของยุทธศาสตร์ชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทวิตีไดอะแกรมดังนี้



รูปที่ 4.5 เอกทวิตีไดอะแกรมของ Insert Cost-Benefit Information

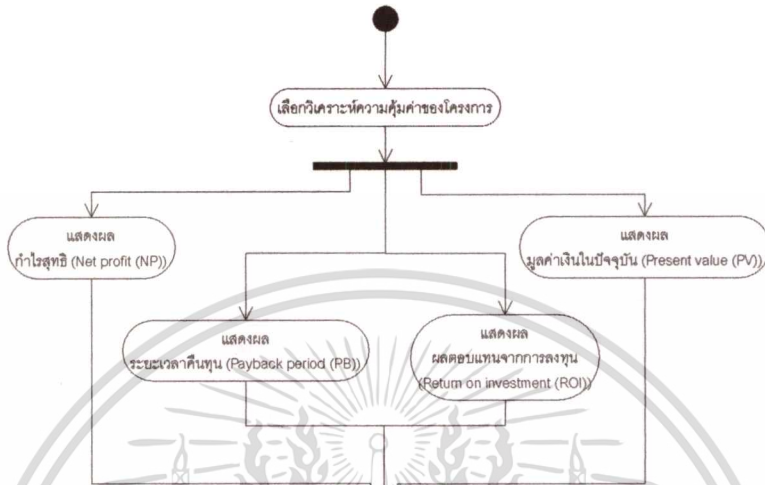
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 รายละเอียดคุณลักษณะ Feasibility Calculation

Use Case Name : Feasibility Calculation	ID : 02:02	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: ทำการคำนวณค่าต่างๆเพื่อวิเคราะห์ความเป็นไปได้ของโครงการ		
Brief Description : - การวิเคราะห์ความคุ้มค่าในการทำโครงการ		
Trigger : วิเคราะห์ความคุ้มค่าของโครงการ Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : <ol style="list-style-type: none"> 1. เลือกวินิจฉัยความคุ้มค่าของโครงการ 2. แสดงผลการวิเคราะห์ S-1 		
Sub flows : S-1: <ol style="list-style-type: none"> 1. แสดงผล กำไรสุทธิ (Net profit (NP)) 2. แสดงผล ระยะเวลาคืนทุน (Payback period (PB)) 3. แสดงผล ผลตอบแทนจากการลงทุน (Return on investment (ROI)) 4. แสดงผล มูลค่าเงินในปัจจุบัน (Present value (PV)) 		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

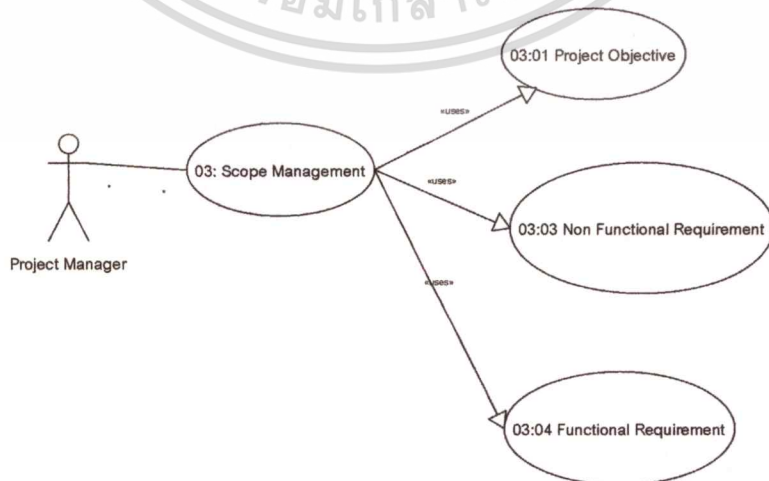
เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Feasibility Calculation ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทวิตีไดอะแกรม ดังนี้



รูปที่ 4.6 เอกทวิตีไดอะแกรมของ Feasibility Calculation

4.2.3 Project scope management

การวิเคราะห์ประกอบด้วย 5 ยูสเคสทำหน้าที่หลักในการบริการข้อมูลเบื้องต้นของโครงการและถูกใช้งานโดย Project Manager



รูปที่ 4.7 ยูสเคสไดอะแกรมของระบบ Project scope management

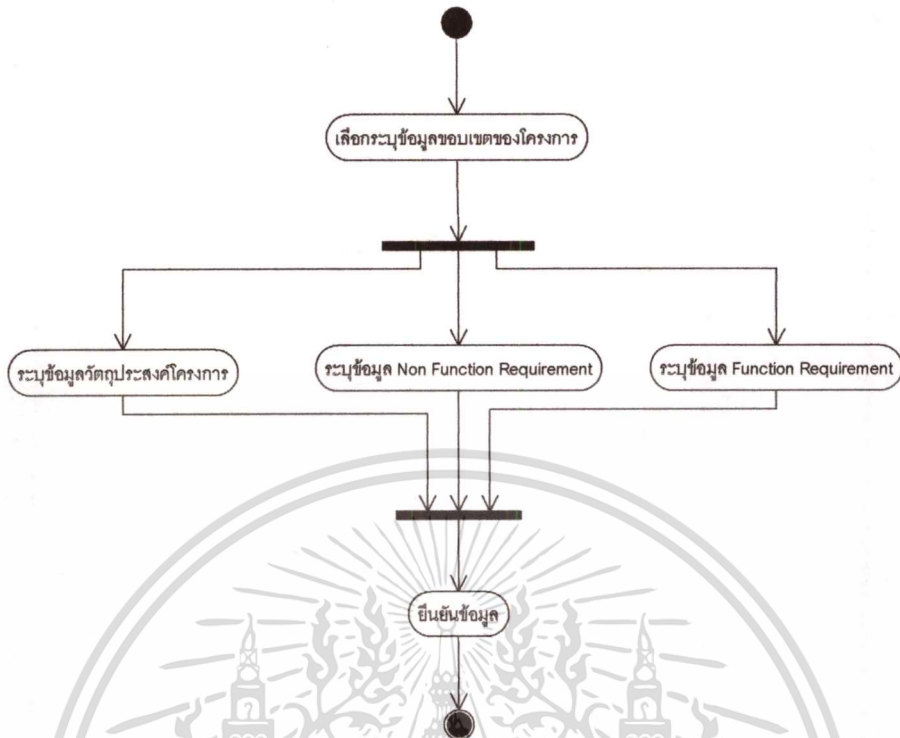
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 รายละเอียดยูสเคส Project scope management

Use Case Name : Project scope management	ID : 03:	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: ระบุข้อมูลจุดประสงค์ของโครงการและขอบเขตของโครงการ		
Brief Description : - ระบุข้อมูลจุดประสงค์ของโครงการและขอบเขตของโครงการ		
Trigger : ระบุข้อมูลจุดประสงค์ของโครงการและขอบเขตของโครงการ Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : 1. เลือกระบุข้อมูลขอบเขตของโครงการ S-1 2. ยืนยันข้อมูล		
Sub flows : S-1: 1. ระบุข้อมูลวัตถุประสงค์โครงการ 2. ระบุข้อมูล Non Function Requirement 3. ระบุข้อมูล Function Requirement		
Alternate/Exceptional Flows :		

เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Project scope management ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทิวทัศน์

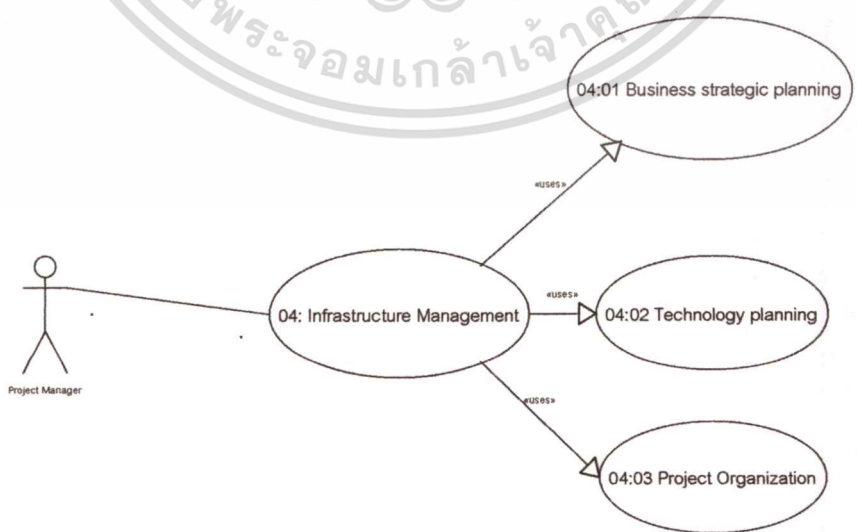
ไดอะแกรมดังนี้
เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 แอ็กทิวตี้ไดอะแกรมของ Project scope management

4.2.4 Project Infrastructure Management

การวิเคราะห์ที่ประกอบด้วย 4 ยุคศาสตร์ทำหน้าที่หลักในการบริการข้อมูลเบื้องต้นของโครงการและถูกใช้งานโดย Project Manager



รูปที่ 4.9 ยุคศาสตร์ไดอะแกรมของระบบ Project Infrastructure Management

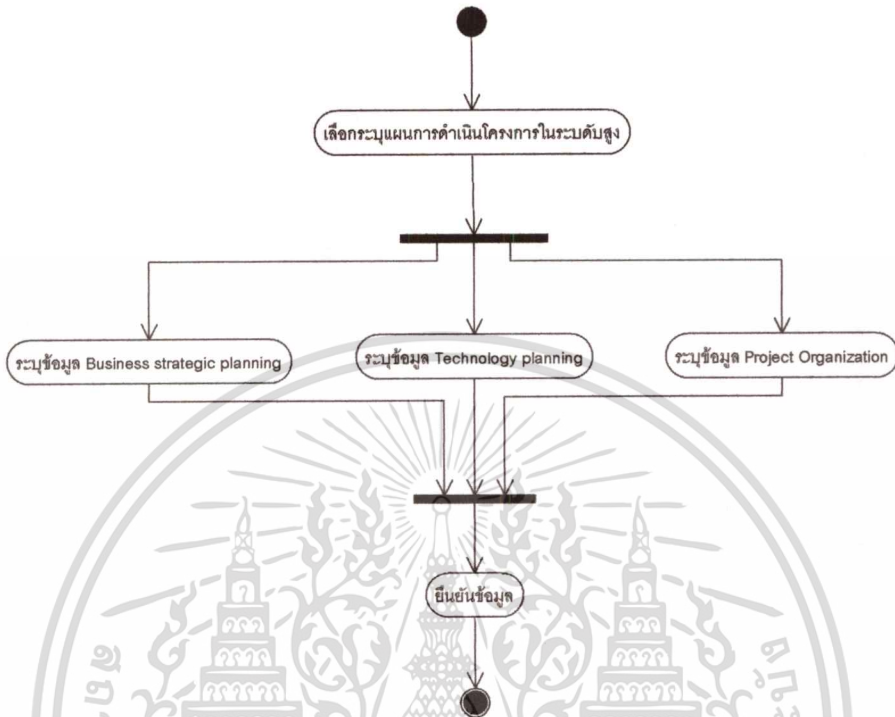
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 รายละเอียดชุดสเคส Project Infrastructure Management

Use Case Name : Project Infrastructure Management	ID : 04:	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: แผนการดำเนินโครงการในระดับสูง		
Brief Description : - เครื่องมือที่ใช้ในแสดงถึงแผนการดำเนินโครงการในระดับสูงเช่น แผนการดำเนินธุรกิจ แผนการใช้เทคโนโลยี รวมถึงโครงสร้างของทีมงานในการพัฒนาโครงการ		
Trigger : ระบุแผนการดำเนินโครงการในระดับสูง Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : <ol style="list-style-type: none"> 1. เลือกระบุแผนการดำเนินโครงการในระดับสูง S-1 2. ยืนยันข้อมูล 		
Sub flows : S-1: <ol style="list-style-type: none"> 1. ระบุข้อมูล Business strategic planning 2. ระบุข้อมูล Technology planning 3. ระบุข้อมูล Project Organization 		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

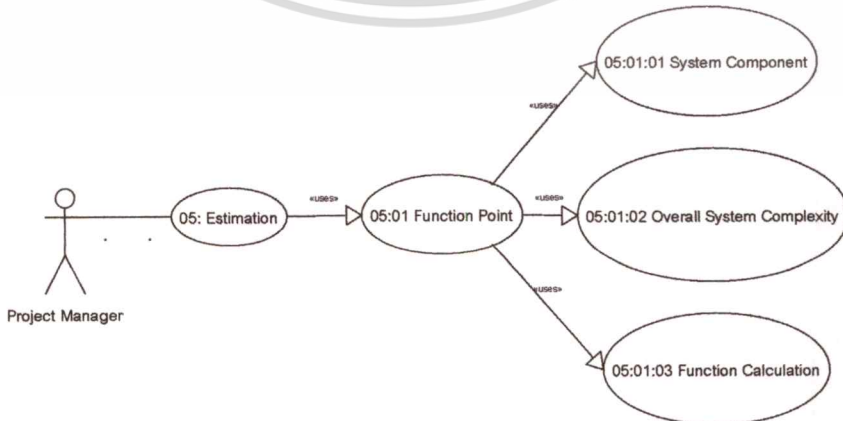
เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Project Infrastructure Management ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทิวทัศน์ไดอะแกรมดังนี้



รูปที่ 4.10 เอกทิวทัศน์ไดอะแกรมของ Project Infrastructure Management

4.2.5 Project Estimation

การวิเคราะห์ประกอบด้วย 5 ยูสเคสทำหน้าที่หลักในการบริการข้อมูลเบื้องต้นของโครงการและถูกใช้งานโดย Project Manager



รูปที่ 4.11 ยูสเคสไดอะแกรมของระบบ Project Estimation

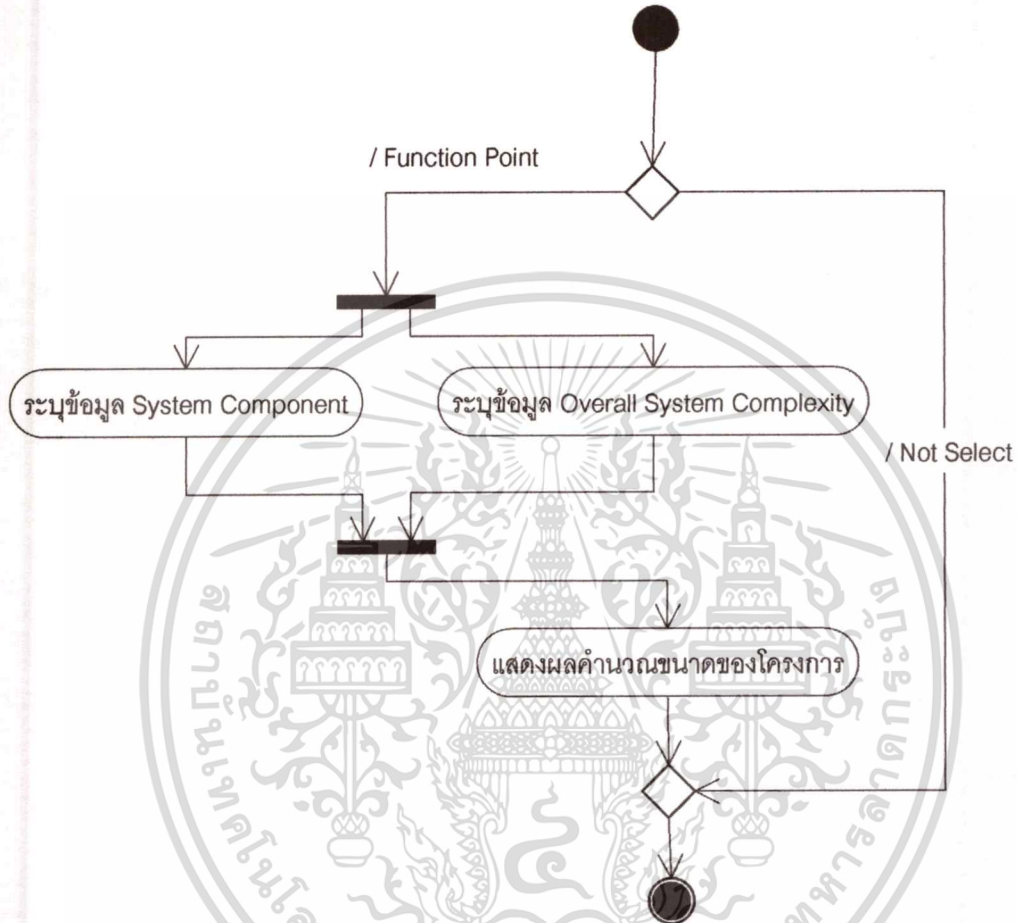
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.7 รายละเอียดยูสเคส Project Estimation

Use Case Name : Project Estimation	ID : 05:	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests :		
Project Manager: การประเมินขนาดของโครงการ		
Brief Description :		
<ul style="list-style-type: none"> - การประเมินขนาดของโครงการเพื่อให้สามารถประเมิน Resource ที่จะใช้ในการดำเนินโครงการและระยะเวลาในการดำเนินโครงการโดยประมาณได้ 		
Trigger : การประเมินขนาดของโครงการ		
Type :		
Relationships :		
Association : Project Manager		
Include :		
Extend :		
Generalization :		
Normal Flow of Events :		
<ol style="list-style-type: none"> 1. เลือกวิธีการประเมินขนาดของโครงการ S-1 2. แสดงผลคำนวณขนาดของโครงการ 		
Sub flows :		
S-1:		
<ol style="list-style-type: none"> 1. เลือกวิธี Function Point <ol style="list-style-type: none"> a. ระบุข้อมูล System Component b. ระบุข้อมูล Overall System Complexity 		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Project Infrastructure Management ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทวิตรีโคอะแกรมดังนี้

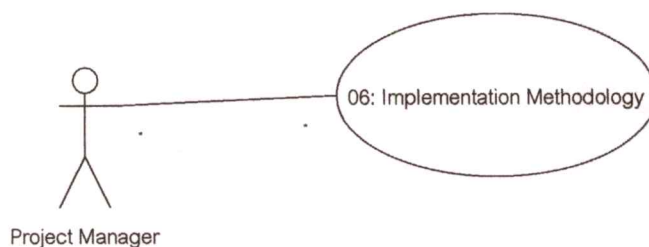


รูปที่ 4.12 เอกทวิตรีโคอะแกรมของ Project Estimation

4.3 Implementation Methodology

การวิเคราะห์ประกอบด้วย 1 ยูสเคสทำหน้าที่หลักในการบริการข้อมูลเบื้องต้นของโครงการและถูกใช้งาน โดย Project Manager

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



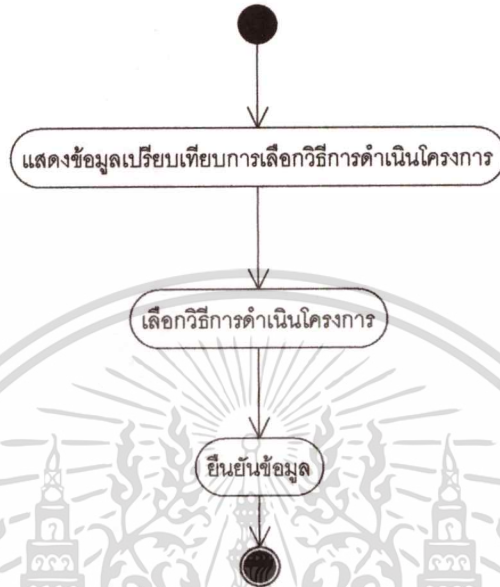
รูปที่ 4.13 ยูสเคสไคอะแกรมของระบบ Implementation Methodology

ตารางที่ 4.8 รายละเอียดยูสเคส Implementation Methodology

Use Case Name : Implementation Methodology	ID : 06:	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: เลือกกระบวนการพัฒนาโครงการ		
Brief Description : - เลือกกระบวนการพัฒนาโครงการเพื่อให้สอดคล้องกับสภาพของโครงการทั้งทางด้านเวลาและขนาดของโครงการ		
Trigger : เลือกกระบวนการพัฒนาโครงการ		
Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : 1. แสดงข้อมูลเปรียบเทียบการเลือกวิธีการดำเนินโครงการ 2. เลือกวิธีการดำเนินโครงการ 3. ยืนยันข้อมูล		
Sub flows :		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

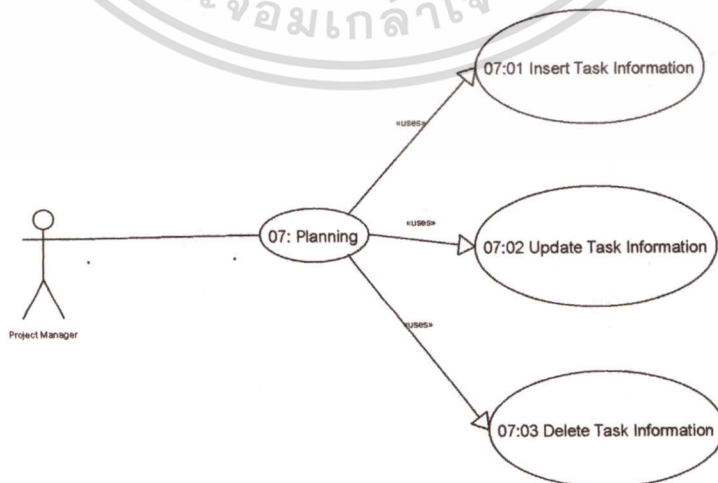
เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Implementation Methodology ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทิวทัศน์ไคอะแกรมดังนี้



รูปที่ 4.14 เอกทิวทัศน์ไคอะแกรมของ Implementation Methodology

4.3 Project Planning

การวิเคราะห์ประกอบด้วย 4 ยูสเคสทำหน้าที่หลักในการบริการข้อมูลเบื้องต้นของโครงการและถูกใช้งานโดย Project Manager



รูปที่ 4.15 ยูสเคสไคอะแกรมของระบบ Project Planning

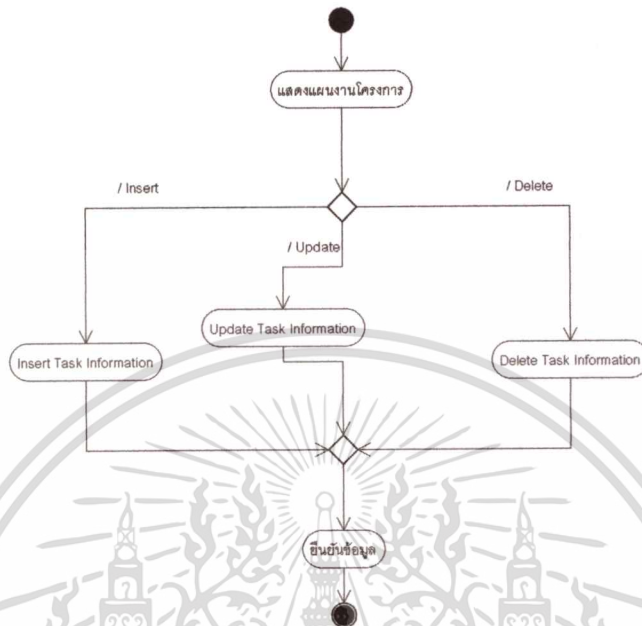
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.9 รายละเอียดยูสเคส Project Planning

Use Case Name : Project Planning	ID : 07:	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: สร้างกิจกรรมต่างๆของโครงการเป็นส่วนย่อยๆเพื่อวางแผนงาน		
Brief Description : - การแบ่งกิจกรรมต่างๆของโครงการออกเป็นส่วนย่อยๆและทำการสร้างเป็นแผนงานเพื่อใช้ในการบริหารงาน โครงการ		
Trigger : สร้างกิจกรรมต่างๆของโครงการเป็นส่วนย่อยๆเพื่อวางแผนงาน Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : 1. แสดงแผนงานโครงการ 2. เลือกขั้นตอนการจัดการขั้นตอน โครงการ S-2 3. ยืนยันข้อมูล		
Sub flows : S-2 1. Insert Task Information 2. Update Task Information 3. Delete Task Information		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

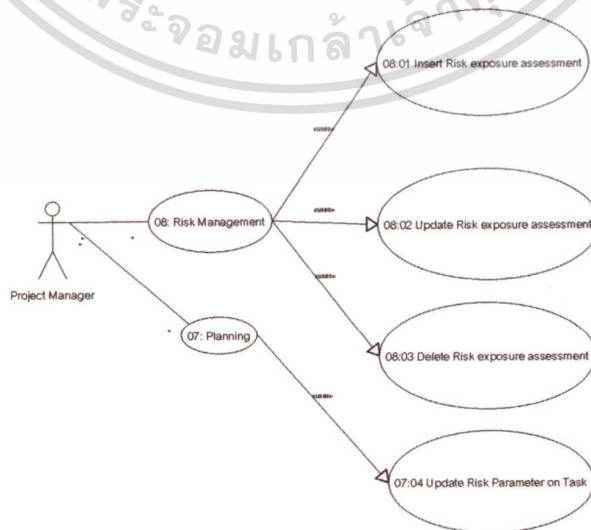
เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Project Planning ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยแอกทิวิตี้ไดอะแกรม ดังนี้



รูปที่ 4.16 แอกทิวิตี้ไดอะแกรมของ Project Planning

4.5 Risk Management

การวิเคราะห์ประกอบด้วย 4 ยูสเคสทำหน้าที่หลักในการบริการข้อมูลเบื้องต้นของโครงการและถูกใช้งาน โดย Project Manager



รูปที่ 4.17 ยูสเคสไดอะแกรมของระบบ Risk Management

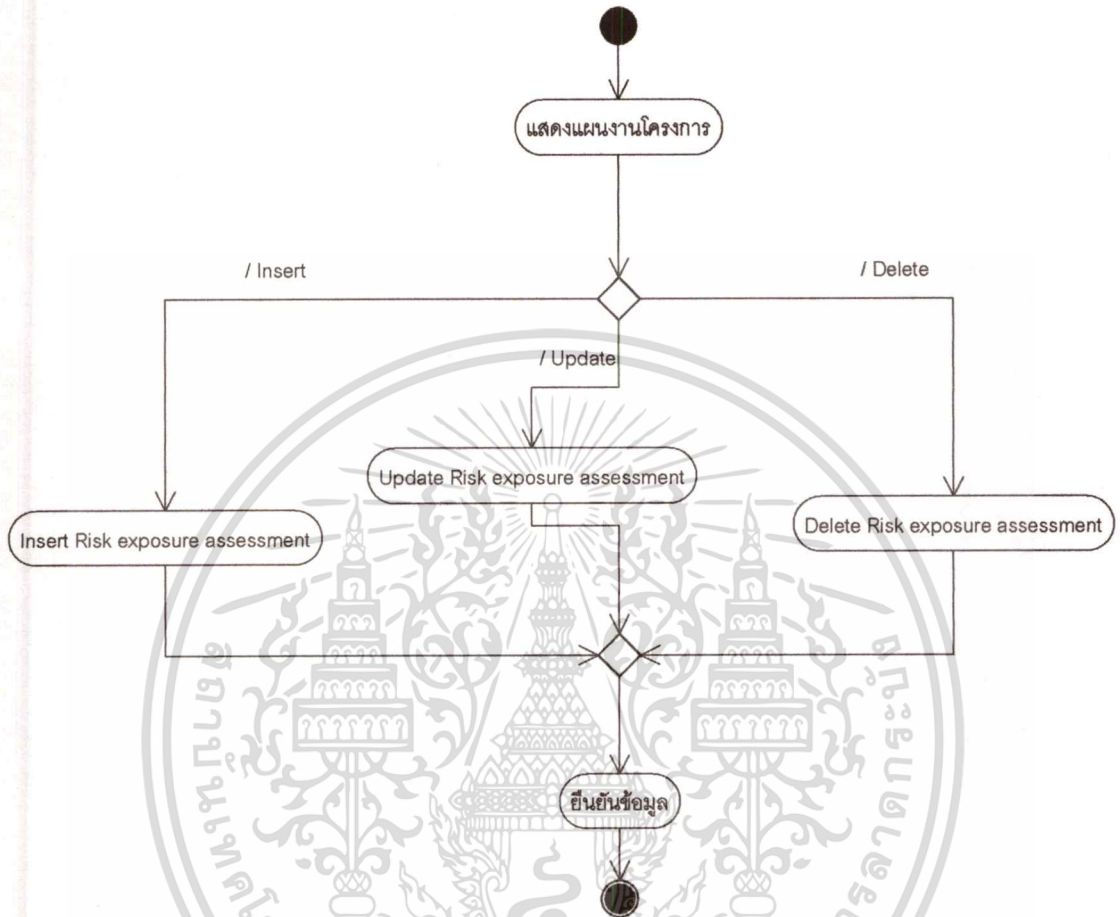
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.10 รายละเอียดชุดเคส Risk Management

Use Case Name : Risk Management	ID : 08:	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: ระบุข้อมูลความเสี่ยงของโครงการ		
Brief Description : - ประเมินความเสี่ยงของโครงการและจัดการบริหารความเสี่ยงของโครงการ		
Trigger : ระบุข้อมูลความเสี่ยงของโครงการ Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : <ol style="list-style-type: none"> 1. ระบุข้อมูล Risk exposure assessment 2. เลือกขั้นตอนการจัดการขั้นตอนความเสี่ยง S-2 3. ยืนยันข้อมูล 		
Sub flows : <ol style="list-style-type: none"> 1. Insert Risk exposure assessment 2. Update Risk exposure assessment 3. Delete Risk exposure assessment 		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Risk Management ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทวิตรีโคอะแกรม ดังนี้



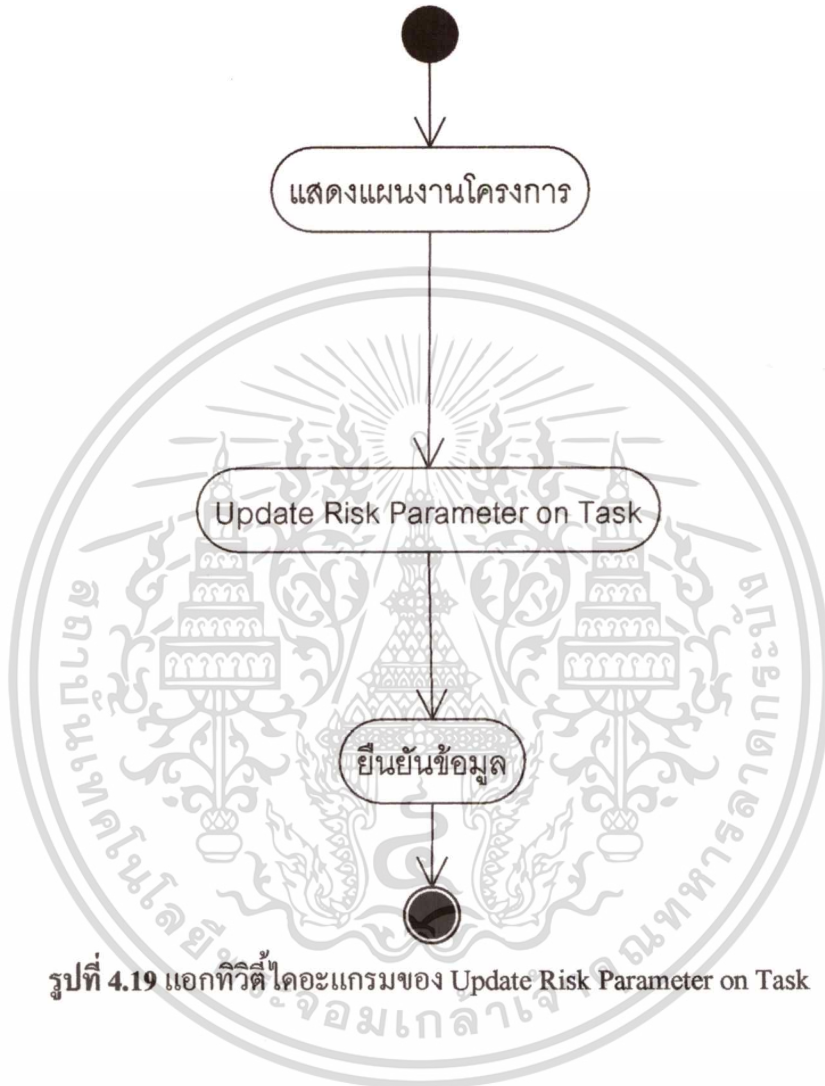
รูปที่ 4.18 เอกทวิตรีโคอะแกรมของ Risk Management

ตารางที่ 4.11 รายละเอียดยูสเคส Update Risk Parameter on Task

Use Case Name : Update Risk Parameter on Task	ID : 07:04	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: เพิ่ม/แก้ไขข้อมูลความเสี่ยงของโครงการในขั้นตอนโครงการ		
Brief Description : - เพิ่มเติมข้อมูลความเสี่ยงของโครงการเพื่อคำนวณหาข้อมูลต่างๆด้านความเสี่ยง		
Trigger : เพิ่ม/แก้ไขข้อมูลความเสี่ยงของโครงการในขั้นตอนโครงการ Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : 1. แสดงแผนงานโครงการ 2. เลือกขั้นตอนการจัดการขั้นตอนโครงการ S-2 3. ยืนยันข้อมูล		
Sub flows : 1. Update Risk Parameter on Task		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

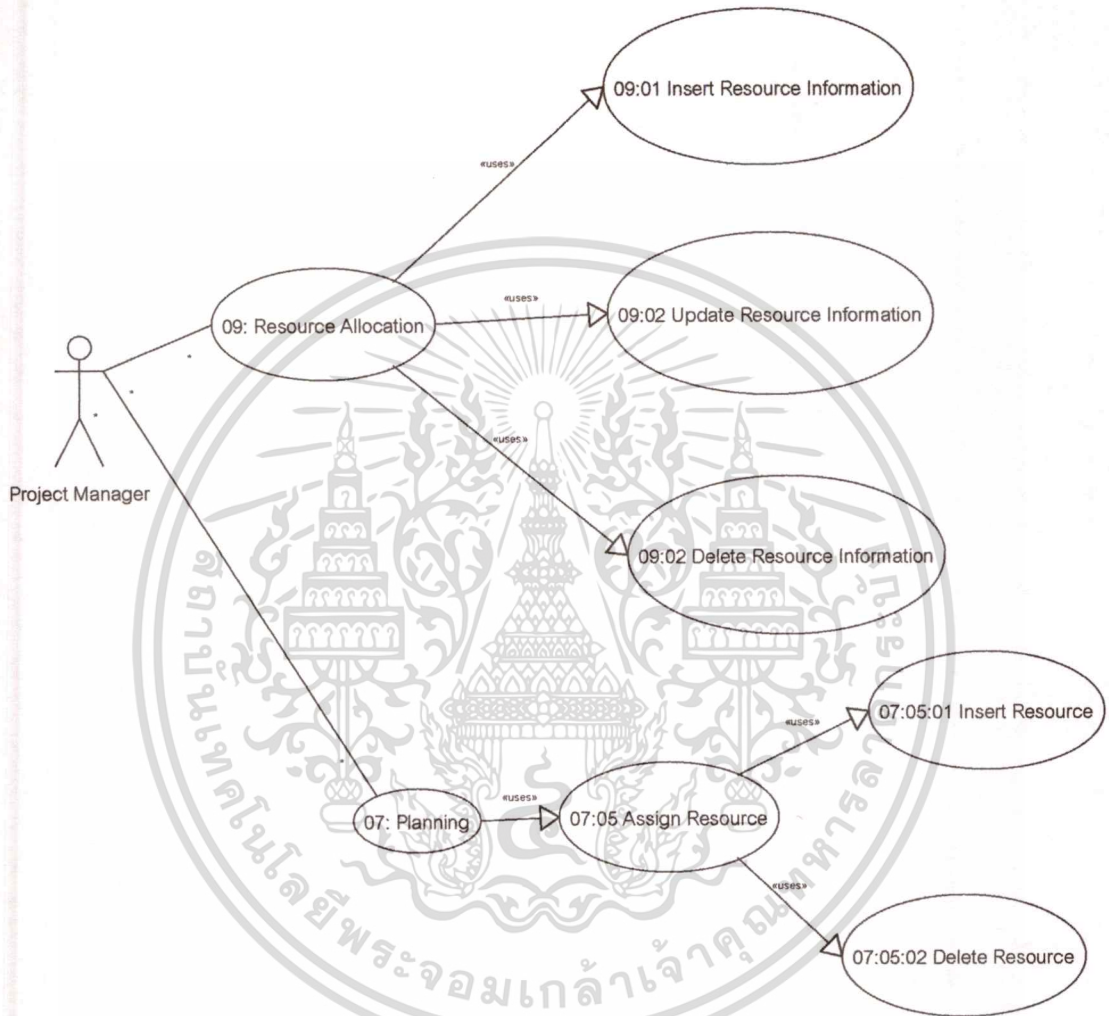
เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Update Risk Parameter on Task ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทวิตรีโคอะแกรมดังนี้



รูปที่ 4.19 เอกทวิตรีโคอะแกรมของ Update Risk Parameter on Task

4.6 Resource Allocation

การวิเคราะห์ประกอบด้วย 8 ยูสเคสทำหน้าที่หลักในการบริการข้อมูลเบื้องต้นของโครงการและถูกใช้งานโดย Project Manager



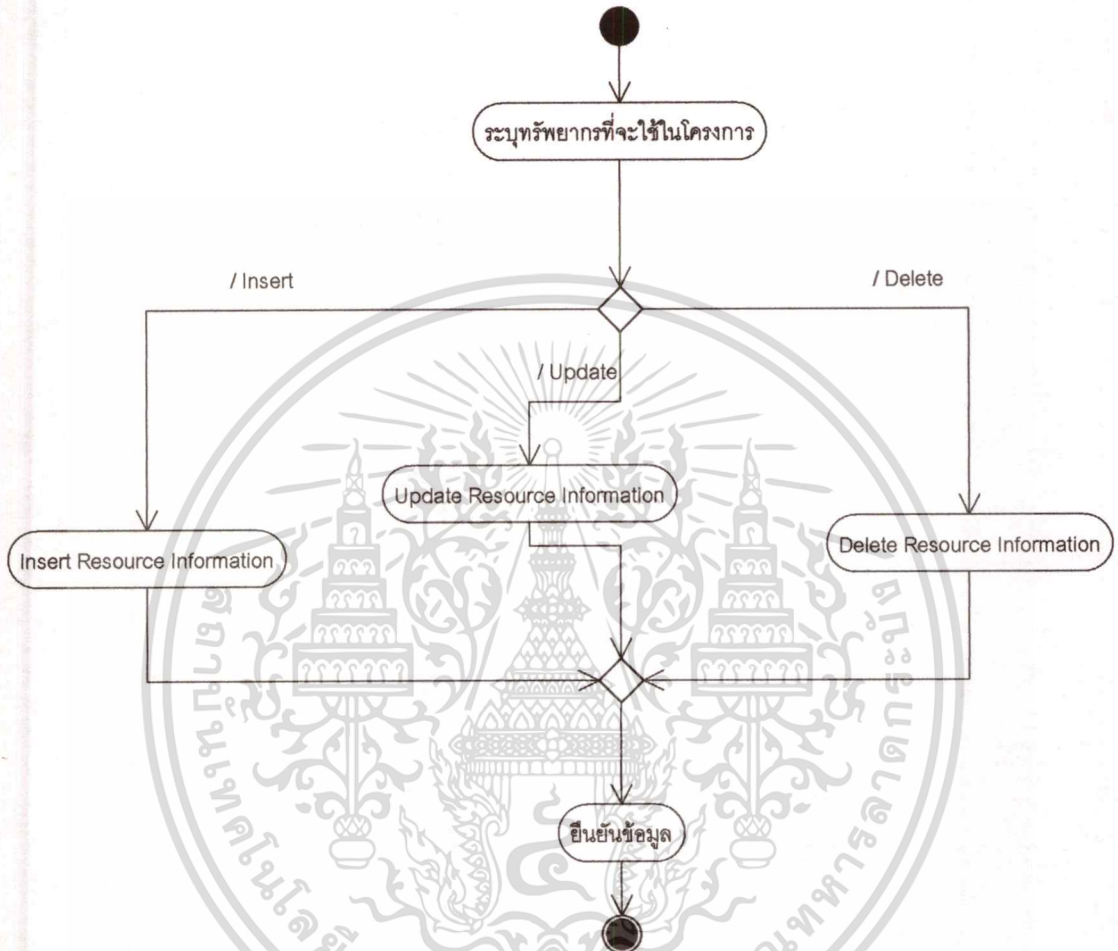
รูปที่ 4.20 ยูสเคสไดอะแกรมของระบบ Resource Allocation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.12 รายละเอียดคุณสมบัติ Resource Allocation

Use Case Name : Resource Allocation	ID : 09:	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: ระบุทรัพยากรที่จะใช้ในโครงการ		
Brief Description : - กำหนดทรัพยากรที่ต้องใช้ในโครงการเพื่อนำไปกำหนดการใช้ทรัพยากรในกิจกรรมต่างๆของโครงการต่อไป		
Trigger : ระบุทรัพยากรที่จะใช้ในโครงการ Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : <ol style="list-style-type: none"> 1. ระบุทรัพยากรที่จะใช้ในโครงการ 2. เลือกขั้นตอนการจัดการทรัพยากร S-2 3. ยืนยันข้อมูล 		
Sub flows : <ol style="list-style-type: none"> 1. Insert Resource Information 2. Update Resource Information 3. Delete Resource Information 		
Alternate/Exceptional Flows :		

เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Resource Allocation ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชุดเงินยี่งั้น จะอธิบายด้วยแอกทิวตี้ไดอะแกรม ดังนี้



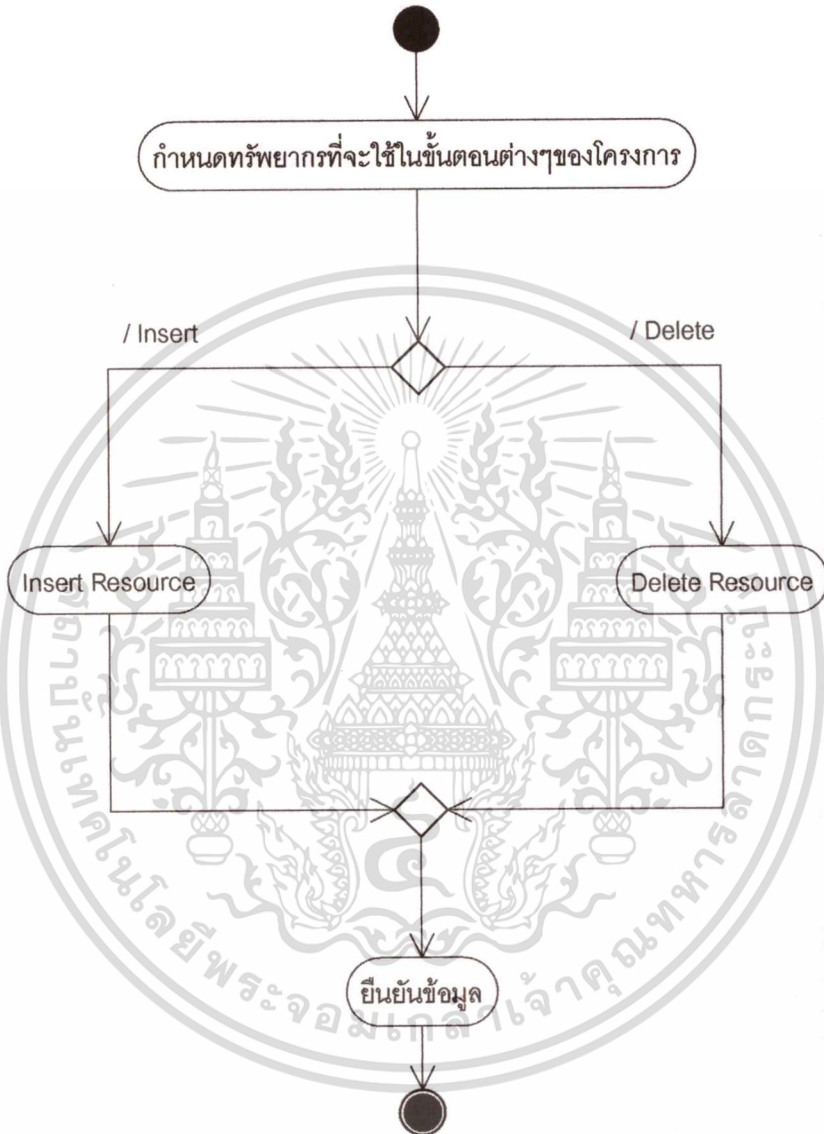
รูปที่ 4.21 แอกทิวตี้ไดอะแกรมของ Resource Allocation

ตารางที่ 4.13 รายละเอียดยูสเคส Assign Resource

Use Case Name : Resource Allocation	ID : 07:05	Importance Level : Height
Primary Actor : Project Manager	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: กำหนดทรัพยากรที่จะใช้ในกิจกรรมต่างๆ โครงการ		
Brief Description : - กำหนดทรัพยากรที่ต้องใช้ในกิจกรรมต่างๆ โครงการ		
Trigger : กำหนดทรัพยากรที่จะใช้ในกิจกรรมต่างๆ โครงการ Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : 1. กำหนดทรัพยากรที่จะใช้ในขั้นตอนต่างๆของโครงการ 2. เลือกขั้นตอนการกำหนดทรัพยากร S-2 3. ยืนยันข้อมูล		
Sub flows : 1. Insert Resource 2. Delete Resource		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Assign Resource ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทวิตรีโคอะแกรม ดังนี้

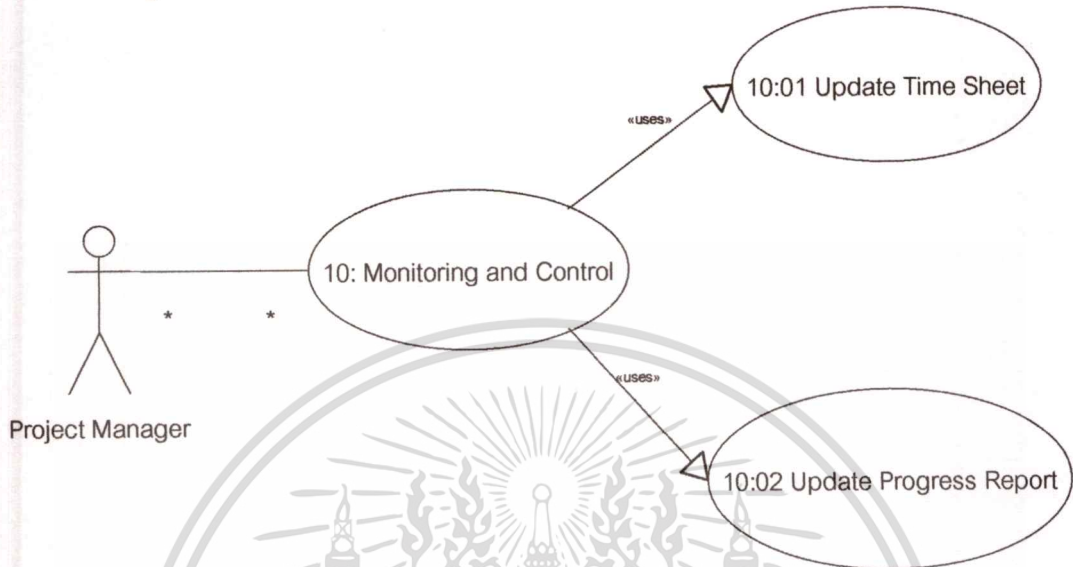


รูปที่ 4.22 เอกทวิตรีโคอะแกรมของ Assign Resource

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7 Project Monitoring and Reporting

การวิเคราะห์ประกอบด้วย 8 ยูสเคสทำหน้าที่หลักในการบริการข้อมูลเบื้องต้นของโครงการและถูกใช้งานโดย Project Manager



รูปที่ 4.23 ยูสเคส โคอะแกรมของระบบ Project Monitoring and Reporting

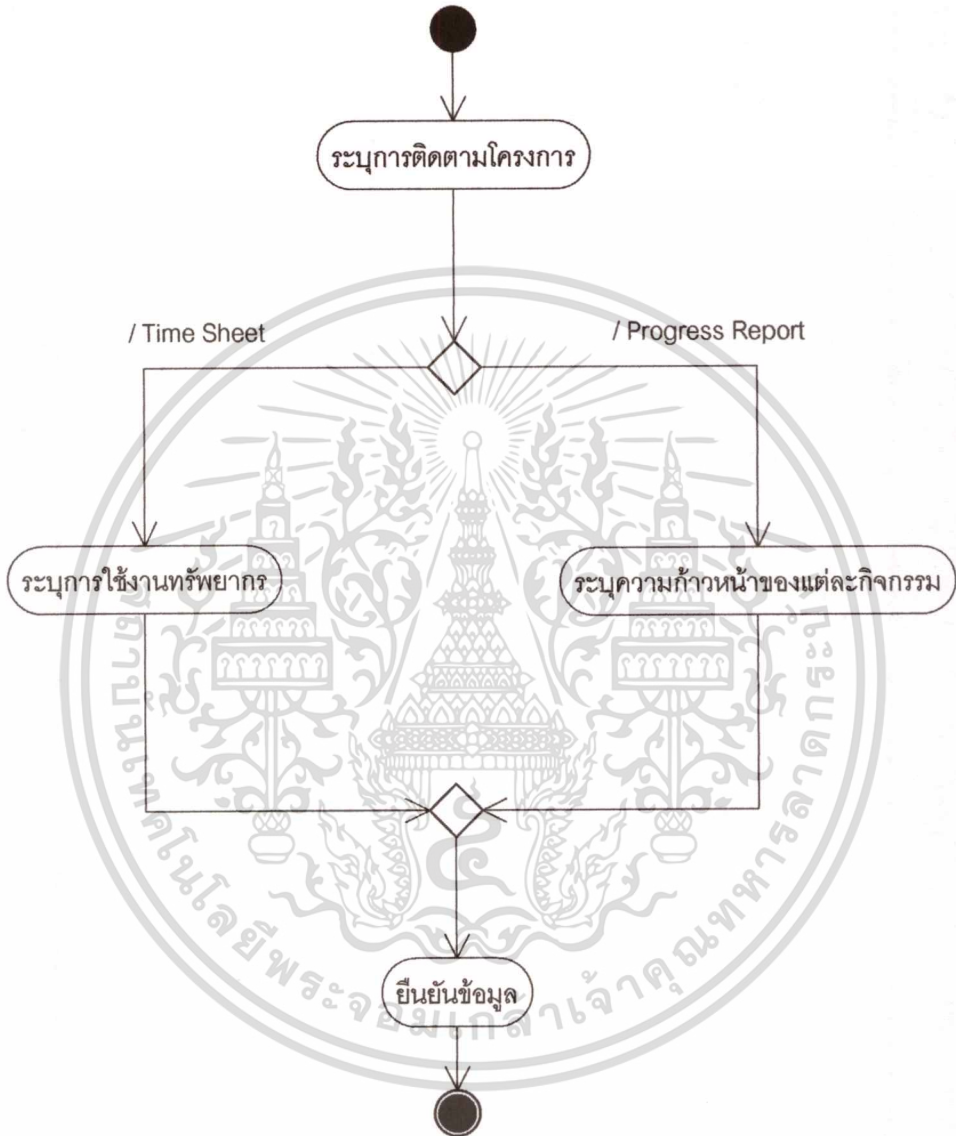
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.14 รายละเอียดยูสเคส Project Monitoring and Control

Use Case Name : Resource Allocation	ID : 06:	Importance Level : Height
Primary Actor : Project Monitoring and Control	Use Case Type : Essential	
Stakeholders and Interests : Project Manager: ระบุการใช้งานทรัพยากรและความก้าวหน้าของแต่ละกิจกรรม		
Brief Description : - เพื่อเป็นการติดตามการดำเนินโครงการ Project Manager จะทำการระบุการใช้งานทรัพยากรและความก้าวหน้าของแต่ละกิจกรรม		
Trigger : ระบุการใช้งานทรัพยากรและความก้าวหน้าของแต่ละกิจกรรม		
Type :		
Relationships : Association : Project Manager Include : Extend : Generalization :		
Normal Flow of Events : 1. ระบุการติดตามโครงการ 2. เลือกขั้นตอนการระบุการติดตามโครงการ S-2 3. ยืนยันข้อมูล		
Sub flows : 1. ระบุการใช้งานทรัพยากร 2. ระบุความก้าวหน้าของแต่ละกิจกรรม		
Alternate/Exceptional Flows :		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เกิดความเข้าใจในขั้นตอนการทำงานของเหตุการณ์ที่ยูสเคส Project Monitoring and Control ทำงานตามที่ได้อธิบายในรายละเอียดของยูสเคสชัดเจนยิ่งขึ้น จะอธิบายด้วยเอกทวิตีไดอะแกรมดังนี้



รูปที่ 4.24 เอกทวิตีไดอะแกรมของ Project Monitoring and Control

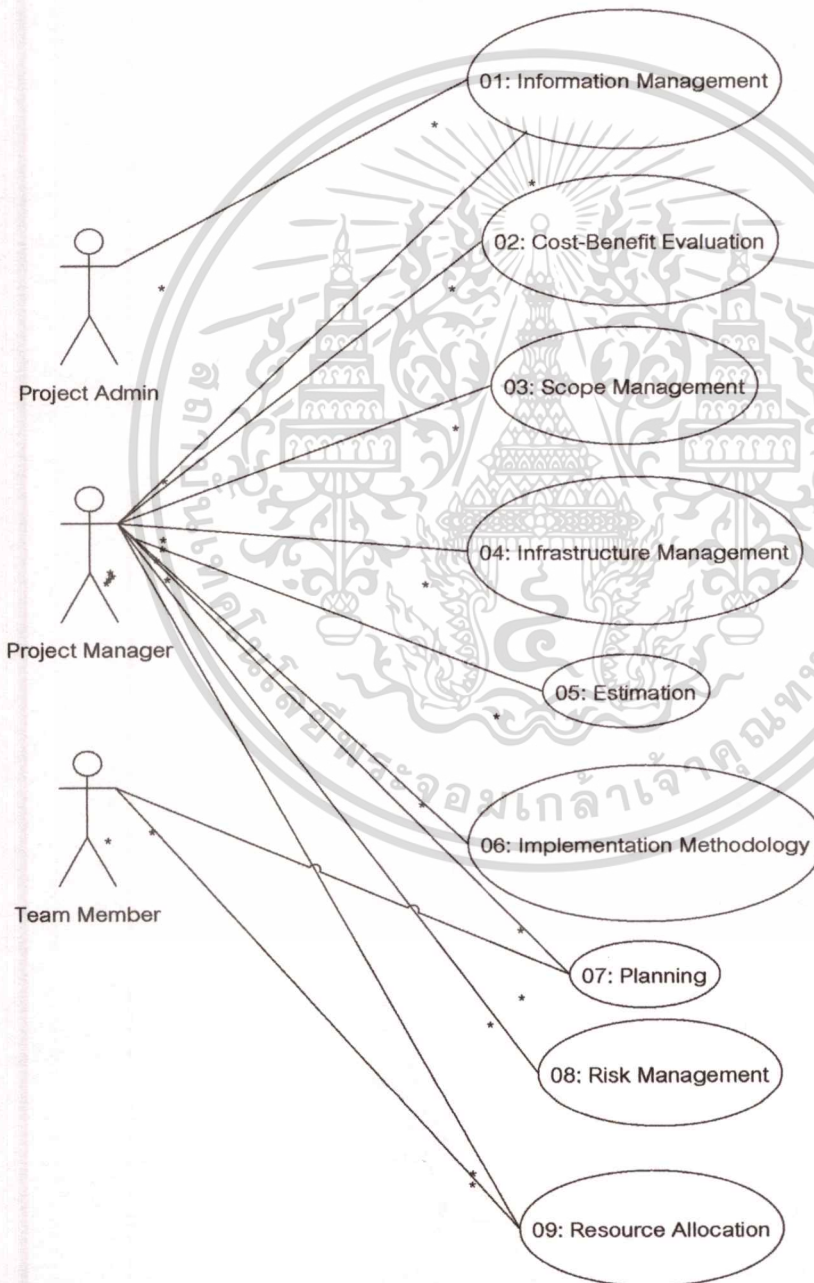
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การออกแบบระบบงาน

5.1 การออกแบบยูสเคสของระบบงาน

จากการวิเคราะห์ระบบงานสามารถสร้างเป็นยูสเคสโดยภาพรวมได้ดังนี้



รูปที่ 5.1 ยูสเคสไดอะแกรมของภาพรวมทั้งระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.1 Problem Domain & Package Diagram

จากการทบทวน Diagram ดังกล่าวได้ดำเนินการแก้ไขรายละเอียดของแต่ละ Class ให้สอดคล้องกับความต้องการของระบบมากที่สุด และสามารถนำไปใช้ได้จริง โดยได้ปรับปรุง Attribute Method และความสัมพันธ์ของแต่ละ Class ใหม่ พร้อมจัดกลุ่ม Class ที่เกี่ยวข้องกันไว้ใน Package เดียวกัน ดังแผนภาพต่อไป นี้ เพื่อให้ง่ายต่อการเขียนโปรแกรมและนำไปพัฒนาต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.2 CRC Card และ Method Specification

ส่วนรายละเอียดของแต่ละ Class ข้างต้น ไม่ว่าจะเป็น Attribute Method และ Relationship สามารถอธิบายได้ด้วย CRC Card และ Method Specification ดังต่อไปนี้

5.2.2.1 CRC Card and Method Specification ของ Person Package

ตารางที่ 5.1 CRC Card Person

Front:

Class Name: Person	ID: 01:01	Type: Concrete, Domain
Description: ข้อมูลบุคคล	Associated Use Case: -	
Responsibilities	Collaborators	
#Create		
#Update		
#Delete		

Back:

Attributes:

personID	(String)
fristName	(String)
lastName	(String)
gender	(String)
dateOfBirth	(Date)
position	(String)
telephone	(String)
email	(String)
role	(String)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Relationships:**Generalization:** _____**Aggregation:** _____**Other Associations:** _____

ตารางที่ 5.2 CRC Card Project Manager

Front:

Class Name: Project Manager	ID: 01:02	Type: Concrete, Domain
Description: ผู้บริหารโครงการ	Associated Use Case: -	
Responsibilities	Collaborators	
_____	_____	
_____	_____	
_____	_____	

Back:**Attributes:****Relationships:****Generalization:** Person _____**Aggregation:** _____**Other Associations:** _____

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.3 CRC Card Project Admin

Front:

Class Name: Project Admin	ID: :01:03	Type: Concrete, Domain
Description: ผู้ดูแลระบบบริหารโครงการ		Associated Use Case: -
Responsibilities	Collaborators	
_____	_____	
_____	_____	
_____	_____	

Back:

Attributes:

Relationships:

Generalization: Person

Aggregation:

Other Associations:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.2.2 CRC Card and Method Specification ของ Project Management Package

ตารางที่ 5.4 CRC Card Project Information

Front:

Class Name: ProjectInformation	ID: 02:01	Type: Concrete, Domain
Description: ข้อมูลโครงการ		Associated Use Case: -
Responsibilities +CreateNewProject +UpdateProjectInformation	Collaborators	

Back:

Attributes:	
Relationships:	
Generalization:	
Aggregation:	
Other Associations:	ProjectAdmin {0...*}, ProjectManager {0...*}

ตารางที่ 5.5 Method Specification Project Information

Method Specification		
Method Name: CreateNewProject	Class Name: ProjectInformation	ID: 02:01:01
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic	<input type="checkbox"/> Smalltalk	<input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java
Triggers/Events: Create New Project		
Arguments Received:		
Data Type:	Notes:	
ProjectInformationVo	Create from ProjectInformationVo	
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
<ol style="list-style-type: none"> 1. Create Project Code 2. Insert Project Information to Database 3. Create New Report in relate table as below <ol style="list-style-type: none"> a. Project Scope b. Project Infrastructure c. Project Estimate d. Project Implementation Methodology 		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.5 (ต่อ)

Method Specification		
Method Name: UpdateProjectInformation	Class Name: ProjectInformation	ID: 02:01:01
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic <input type="checkbox"/> Smalltalk <input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java		
Triggers/Events: Update Project Information		
Arguments Received:		
Data Type: ProjectInformationVo	Notes: Create from ProjectInformationVo	
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type: Boolean	Notes: Success or Fail	
Algorithm Specification:		
4. Update Project Information to Database		
a. KEY: Project Code		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 Method Specification Project Evaluation

Method Specification		
Method Name: AddBenefit	Class Name: ProjectEvaluation	ID: 02:02:01
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic	<input type="checkbox"/> Smalltalk	<input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java
Triggers/Events: Add Project Benefit		
Arguments Received:		
Data Type:	Notes:	
ProjectBenefitVo	Create from ProjectBenefitVo	
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
<ol style="list-style-type: none"> 1. Generate Project Benefit ID 2. Add Project Benefit to Database 		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 (ต่อ)

Method Specification		
Method Name: DeleteBenefit	Class Name: ProjectEvaluation	ID: 02:02:02
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic	<input type="checkbox"/> Smalltalk	<input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java
Triggers/Events: Delete Project Benefit		
Arguments Received:		
Data Type:	Notes:	
ProjectBenefitVo	Create from ProjectBenefitVo	
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. Delete Project Benefit from Database		
a. KEY: Project Code, Benefit ID		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 (ต่อ)

Method Specification		
Method Name: AddCost	Class Name: ProjectEvaluation	ID: 02:02:03
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic	<input type="checkbox"/> Smalltalk	<input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java
Triggers/Events: Add Project Cost		
Arguments Received:		
Data Type:	Notes:	
ProjectBenefitVo	Create from ProjectBenefitVo	
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
<ol style="list-style-type: none"> 1. Generate Project Cost ID 2. Add Project Cost to Database 		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 (ต่อ)

Method Specification		
Method Name: DeleteCost	Class Name: ProjectEvaluation	ID: 02:02:04
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic <input type="checkbox"/> Smalltalk <input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java		
Triggers/Events: Delete Project Cost		
Arguments Received:		
Data Type:	Notes:	
ProjectBenefitVo	Create from ProjectBenefitVo	
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
3. Delete Project Cost from Database <ol style="list-style-type: none"> a. KEY: Project Code, Cost ID 		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 (ต่อ)

Method Specification		
Method Name: FeasibilityCalculate	Class Name: ProjectEvaluation	ID: 02:02:05
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic	<input type="checkbox"/> Smalltalk	<input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java
Triggers/Events: Calculate Feasibility Value		
Arguments Received:		
Data Type:	Notes:	
ProjectBenefitVo[]		
ProjectCostVo[]		
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
ProjectEvaluation.NP	ProjectBenefitVo[]	
	ProjectCostVo[]	
ProjectEvaluation.PV	ProjectBenefitVo[]	
	ProjectCostVo[]	
ProjectEvaluation.NPV	ProjectBenefitVo[]	
	ProjectCostVo[]	
ProjectEvaluation.ROT	ProjectBenefitVo[]	
	ProjectCostVo[]	
ProjectEvaluation.BEP	ProjectBenefitVo[]	
	ProjectCostVo[]	
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. Calculate All of Feasibility Value		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 (ต่อ)

Method Specification		
Method Name: NetProfit	Class Name: ProjectEvaluation	ID: 02:02:06
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic	<input type="checkbox"/> Smalltalk	<input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java
Triggers/Events: Calculate Net Profit		
Arguments Received:		
Data Type:	Notes:	
ProjectBenefitVo[]		
ProjectCostVo[]		
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. ผลรวมรายได้ทั้งโครงการ - ผลรวมค่าใช้จ่ายทั้งโครงการ SUM (ProjectBenefitVo.amount[]) - SUM (ProjectCostVo.amount[])		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 (ต่อ)

Method Specification		
Method Name: PV	Class Name: ProjectEvaluation	ID:02:02:07
Contract ID:	Programmer:	Date Due:
Programming Language: <input type="checkbox"/> Visual Basic <input type="checkbox"/> Smalltalk <input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java		
Triggers/Events: Calculate Present Value		
Arguments Received:		
Data Type:	Notes:	
ProjectBenefitVo[]		
ProjectCostVo[]		
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. คำนวณหาปีที่ค่า Present Value (PV) ในแต่ละปี For I = 0 ; I <= จำนวนปีของโครงการในวิธีการ Implement ที่เลือก ; I ++ Calculate PV Benefit Calculate PV Cost Update Year PV to database End For		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 (ต่อ)

Method Specification		
Method Name: NPV	Class Name: ProjectEvaluation	ID: 02:02:08
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic	<input type="checkbox"/> Smalltalk	<input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java
Triggers/Events: Calculate Net Present Value		
Arguments Received:		
Data Type:	Notes:	
ProjectBenefitVo[]		
ProjectCostVo[]		
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. คำนวณหาปีที่ค่า Net Present Value (NPV) ในแต่ละปี		
For I = 0 ; I <= จำนวนปีของโครงการในวิธีการ Implement ที่เลือก ; I ++		
Calculate Year NPV = Year PV Benefit – Year PV Cost		
Update Year NPV to database		
End For		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 (ต่อ)

Method Specification		
Method Name: ROI	Class Name: ProjectEvaluation	ID: 02:02:09
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic	<input type="checkbox"/> Smalltalk	<input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java
Triggers/Events: Calculate Return on Investment		
Arguments Received:		
Data Type:	Notes:	
ProjectBenefitVo[]		
ProjectCostVo[]		
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. Calculate ROI		
$ROI = \frac{\text{Total Benefit NPV} - \text{Total Cost NPV}}{\text{Total Cost NPV}}$		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.7 (ต่อ)

Method Specification		
Method Name: BEP	Class Name: ProjectEvaluation	ID:02:02:10
Contract ID:	Programmer:	Date Due:
Programming Language: <input type="checkbox"/> Visual Basic <input type="checkbox"/> Smalltalk <input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java		
Triggers/Events: Calculate Break-Event Period		
Arguments Received:		
Data Type:	Notes:	
ProjectBenefitVo[]		
ProjectCostVo[]		
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. คำนวณหาจุดคุ้มทุน For I = 0 ; I <= จำนวนปีของโครงการในวิธีการ Implement ที่เลือก ; I ++ หา Year NPV > 0 ในปีแรก <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> $BEP = BEP\ Year + \frac{Yearly\ NPV * - Cumulative\ NPV}{Yearly\ NPV *}$ </div> Update Break-Event Point to database End For		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.8 CRC Card Project Scope

Front:

Class Name: ProjectScope	ID: 02:03	Type: Concrete, Domain
Description: ข้อมูลขอบเขตโครงการ		Associated Use Case: -
Responsibilities +UpdateProjectScope _____ _____ _____	Collaborators _____ _____ _____	

Back:

Attributes: _____ _____	
Relationships:	
Generalization:	_____
Aggregation:	_____
Other Associations:	ProjectManager {0...*} _____

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.9 Method Specification Project Scope

Method Specification		
Method Name: UpdateProjectScope	Class Name: ProjectScope	ID: 02:03:01
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic	<input type="checkbox"/> Smalltalk	<input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java
Triggers/Events: Update Project Scope		
Arguments Received:		
Data Type:	Notes:	
ProjectScopeVo	Create from ProjectScopeVo	
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. Update Project Scope to Database		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.10 CRC Card Project Infrastructure

Front:

Class Name: ProjectInfrastructure	ID: 02:04	Type: Concrete, Domain
Description: Project Infrastructure		Associated Use Case: -
Responsibilities	Collaborators	
+UpdateProjectInfrastructure		

Back:

Attributes:	
Relationships:	
Generalization:	
Aggregation:	
Other Associations:	ProjectManager {0..*}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.11 Method Specification Project Infrastructure

Method Specification		
Method Name: UpdateProjectInfrastructure	Class Name: ProjectInfrastructure	ID: 02:04:01
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic <input type="checkbox"/> Smalltalk <input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java		
Triggers/Events: Update Project Infrastructure		
Arguments Received:		
Data Type:	Notes:	
ProjectInfrastructureVo	Create from ProjectInfrastructureVo	
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. Update Project Infrastructure to Database		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.12 CRC Card Project Estimation

Front:

Class Name: ProjectEstimation	ID: 02:05	Type: Concrete, Domain
Description: Project Estimation		Associated Use Case: -
Responsibilities +AddEstimateMethod <hr/> +DeleteEstimateMethod <hr/> <hr/>		Collaborators <hr/> <hr/> <hr/>

Back:

Attributes:	
<hr/>	
Relationships:	
Generalization:	<hr/>
Aggregation:	<hr/>
Other Associations:	ProjectManager {0...*} <hr/>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.13 Method Specification Project Estimation

Method Specification		
Method Name: AddEstimateMethod	Class Name: ProjectEstimation	ID: 02:05:01
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic <input type="checkbox"/> Smalltalk <input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java		
Triggers/Events: Add Estimate Method		
Arguments Received:		
Data Type:	Notes:	
estimateMethodID (String)		
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. Add Estimate Method to Database		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.13 (ต่อ) Method Specification Project Estimation

Method Specification		
Method Name: DeleteEstimateMethod	Class Name: ProjectEstimation	ID: 02:05:02
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic <input type="checkbox"/> Smalltalk <input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java		
Triggers/Events: Delete Estimate Method		
Arguments Received:		
Data Type:	Notes:	
estimateMethodID (String)		
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type:	Notes:	
Boolean	Success or Fail	
Algorithm Specification:		
1. Delete Estimate Method to Database		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.14 CRC Card Function Point

Front:

Class Name: FunctionPoint	ID: 02:06	Type: Concrete, Domain
Description: Function Point Estimation		Associated Use Case: -
Responsibilities	Collaborators	
+UpdateSystemComponent	_____	
+AddSystemCompexity	_____	
+DeleteSystemComplexity	_____	
+FunctionPointEstimate	_____	

Back:

Attributes:	

Relationships:	
Generalization:	_____
Aggregation:	_____
Other Associations: ProjectManager {0...*}	_____

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.15 CRC Card Project Implementation Methodology

Front:

Class Name: ProjectImplementationMethodology	ID: 02:07	Type: Concrete, Domain
Description: Project Implementation Methodology		Associated Use Case: -
Responsibilities + UpdateMethodology _____ _____ _____	Collaborators _____ _____ _____	

Back:

Attributes: _____ _____
Relationships: Generalization: _____ Aggregation: _____ Other Associations: ProjectManager {0..*} _____

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.16 Method Specification Project Implementation Methodology

Method Specification		
Method Name: UpdateMethodology	Class Name: ProjectImplementationMethodology	ID: 02:07:01
Contract ID:	Programmer:	Date Due:
Programming Language:		
<input type="checkbox"/> Visual Basic <input type="checkbox"/> Smalltalk <input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java		
Triggers/Events: Update Implementation Methodology		
Arguments Received:		
Data Type: ProjectImplementationMethodologyVo	Notes: Create from ProjectImplementationMethodologyVo	
Messages Sent & Arguments Passed:		
ClassName.MethodName:	Data Type:	Notes:
Argument Returned:		
Data Type: Boolean	Notes: Success or Fail	
Algorithm Specification:		
1. Update Project Implementation Methodology to Database		
Misc.Notes:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.17 CRC Card Project Planning

Front:

Class Name: ProjectPlanning	ID: 02:08	Type: Concrete, Domain
Description: Project Planning		Associated Use Case: -
Responsibilities	Collaborators	
+ InsertTask	_____	
+UpdateTask	_____	
+DeleteTask	_____	
+UpdateRiskParam	_____	
+UpdateEffortUsage	_____	
+UpdateTaskProgress	_____	

Back:

Attributes:

Relationships:
Generalization: _____
Aggregation: _____
Other Associations: ProjectManager {0...*}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.18 CRC Card Project Risk Management

Front:

Class Name: ProjectRiskManagement	ID: 02:09	Type: Concrete, Domain
Description: Project Risk Management		Associated Use Case: -
Responsibilities	Collaborators	
+ InsertRisk		
+UpdateRisk		
+DeleteRisk		

Back:

Attributes:	
Relationships:	
Generalization:	
Aggregation:	
Other Associations:	ProjectManager {0...*}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.19 CRC Card Resource Allocation

Front:

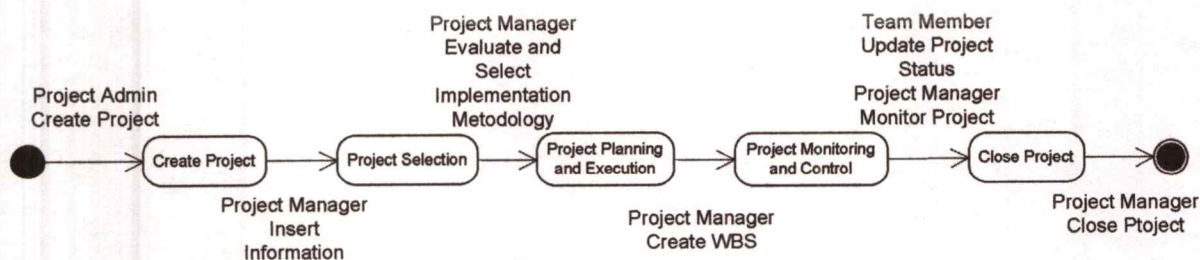
Class Name: ResourceAllocation	ID: 02:10	Type: Concrete, Domain
Description: Project Resource Allocation		Associated Use Case: -
Responsibilities	Collaborators	
+ InsertResource	_____	
+UpdateResource	_____	
+DeleteResource	_____	

Back:

Attributes:

Relationships:
Generalization:
Aggregation:
Other Associations: ProjectManager {0...}*}

5.2.3 State Diagram



รูปที่ 5.4 สเตทไออะแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การออกแบบฐานข้อมูล

6.1 การออกแบบฐานข้อมูล

จากการวิเคราะห์โครงสร้างและการออกแบบระบบงาน ทำให้สามารถออกแบบฐานข้อมูลที่เป็นฐานข้อมูลเชิงสัมพันธ์ แสดงโดยแผนภาพความสัมพันธ์ระหว่างเอนทิตี แบบ Crow's Foot ดังรูปที่ 6.1 ซึ่งประกอบด้วยตารางสำหรับจัดเก็บข้อมูลต่างๆดังนี้

6.1.1 **BENEFIT** เป็นตารางสำหรับเก็บข้อมูลรายได้จากโครงการ

6.1.2 **COST** เป็นตารางสำหรับเก็บข้อมูลรายจ่ายโครงการ

6.1.3 **COST_UNIT** เป็นตารางสำหรับเก็บข้อมูลชนิดของรายจ่ายของทรัพยากร

6.1.4 **DEVELOP_METHOD** เป็นตารางสำหรับเก็บข้อมูลชนิดของการพัฒนาโครงการ

6.1.5 **FUNCTION_POINT_PARAM** เป็นตารางสำหรับเก็บข้อมูลชนิดและค่าของตัวแปรในการคำนวณ Function Point

6.1.6 **FUNCTION_POINT_SYSTEM_COMPLEXITY** เป็นตารางสำหรับเก็บข้อมูลความซับซ้อนของโครงการในการคำนวณ Function Point

6.1.7 **FUNCTION_POINT_SYSTEM_COMPONENT** เป็นตารางสำหรับเก็บข้อมูลค่า INPUT/OUTPUT ต่างๆในการคำนวณ Function Point

6.1.8 **METHODOLOGY** เป็นตารางสำหรับเก็บข้อมูลกระบวนการพัฒนาโครงการ

6.1.9 **PERSON_ROLE** เป็นตารางสำหรับเก็บข้อมูลสิทธิ์ของใช้งาน

6.1.10 **PERSON** เป็นตารางสำหรับเก็บข้อมูลผู้ใช้

6.1.11 **PROJECT_STATUS** เป็นตารางสำหรับเก็บข้อมูลสถานะของโครงการ

6.1.12 **PROJECT** เป็นตารางสำหรับเก็บข้อมูลโครงการ

6.1.13 **RESOURCE_ALLOCATION** เป็นตารางสำหรับเก็บข้อมูลทรัพยากร

6.1.14 **RESOURCE_CATEGORY** เป็นตารางสำหรับเก็บข้อมูลกลุ่มของทรัพยากร

6.1.15 **RESOURCE_TYPE** เป็นตารางสำหรับเก็บข้อมูลชนิดของทรัพยากร

6.1.16 **RISK** เป็นตารางสำหรับเก็บข้อมูลความเสี่ยง

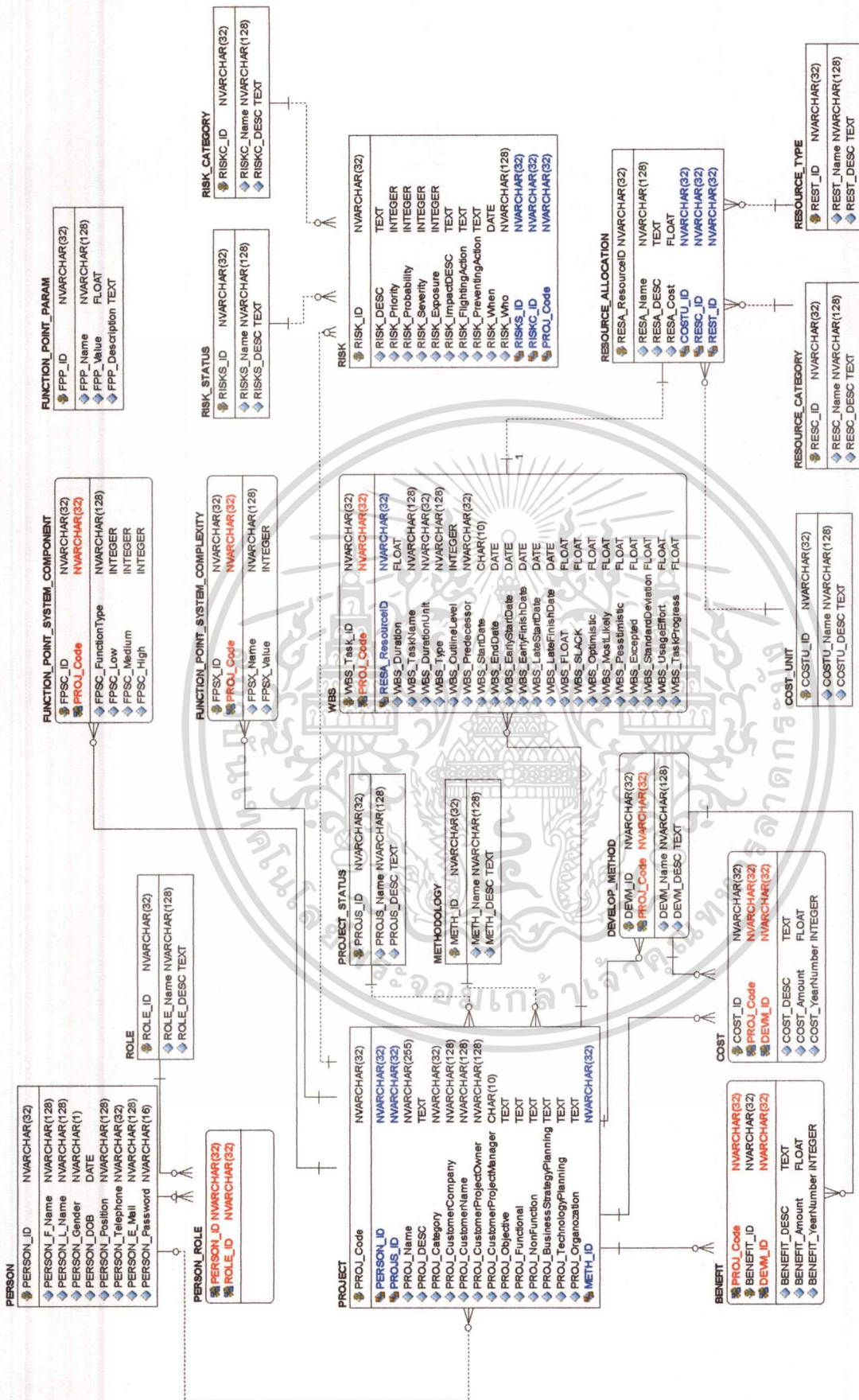
6.1.17 **RISK_CATEGORY** เป็นตารางสำหรับเก็บข้อมูลกลุ่มของความเสี่ยง

6.1.18 **RISK_STATUS** เป็นตารางสำหรับเก็บข้อมูลสถานะของความเสี่ยง

6.1.19 **ROLE** เป็นตารางสำหรับเก็บข้อมูลชนิดของสิทธิ์ของใช้งาน

เอกสารนี้เป็น 6.1.20 **WBS** เป็นตารางสำหรับเก็บข้อมูลกิจกรรมต่างๆในโครงการ นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.1 แผนภาพความสัมพันธ์ระหว่างเอนทิตี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 พจนานุกรมข้อมูล

รายละเอียดของตารางต่างๆแสดงได้ด้วยพจนานุกรมข้อมูลดังนี้

ตารางที่ 6.1 พจนานุกรมตารางข้อมูล BENEFIT

Attribute Name	Description	Data type	PK/FK	Table (FK)
PROJ_Code	รหัสโครงการ	NVARCHAR(32)	PK	
BENEFIT_ID	รหัสรายได้	NVARCHAR(32)	PK	
DEVM_ID	รหัสวิธีการพัฒนา	NVARCHAR(32)	PK	
BENEFIT_DESC	รายละเอียดรายได้	TEXT		
BENEFIT_Amount	รายได้	FLOAT		
BENEFIT_YearNumber	ปีที่รับรายได้	INTEGER		

ตารางที่ 6.2 พจนานุกรมตารางข้อมูล COST

Attribute Name	Description	Data type	PK/FK	Table (FK)
COST_ID	รหัสรายจ่าย	NVARCHAR(32)	PK	
PROJ_Code	รหัสโครงการ	NVARCHAR(32)	PK	
DEVM_ID	รหัสวิธีการพัฒนา	NVARCHAR(32)	PK	
COST_DESC	รายละเอียดรายจ่าย	TEXT		
COST_Amount	รายจ่าย	FLOAT		
COST_YearNumber	ปีที่จ่าย	INTEGER		

ตารางที่ 6.3 พจนานุกรมตารางข้อมูล COST_UNIT

Attribute Name	Description	Data type	PK/FK	Table (FK)
COSTU_ID	รหัสหน่วยรายจ่าย ของทรัพยากร	NVARCHAR(32)	PK	
COSTU_Name	ชื่อหน่วย	NVARCHAR(128)		
COSTU_DESC	รายละเอียดหน่วย	TEXT		

ตารางที่ 6.4 พจนานุกรมตารางข้อมูล DEVELOP_METHOD

Attribute Name	Description	Data type	PK/FK	Table (FK)
DEVM_ID	รหัสวิธีพัฒนา	NVARCHAR(32)	PK	
DEVM_Name	ชื่อวิธีพัฒนา	NVARCHAR(128)		
DEVM_DESC	รายละเอียด	TEXT		

ตารางที่ 6.5 พจนานุกรมตารางข้อมูล FUNCTION_POINT_PARAM

Attribute Name	Description	Data type	PK/FK	Table (FK)
FPP_ID	รหัสตัวแปร	NVARCHAR(32)	PK	
FPP_Name	ชื่อตัวแปร	NVARCHAR(128)		
FPP_Value	ค่าตัวแปร	FLOAT		
FPP_Description	รายละเอียด	TEXT		

ตารางที่ 6.6 พจนานุกรมตารางข้อมูล FUNCTION_POINT_SYSTEM_COMPLEXITY

Attribute Name	Description	Data type	PK/FK	Table (FK)
FPSX_ID	รหัสความซับซ้อน	NVARCHAR(32)	PK	
PROJ_Code	รหัสโครงการ	NVARCHAR(32)	PK	
FPSX_Name	ชื่อความซับซ้อน	NVARCHAR(128)		
FPSX_Value	ค่าความซับซ้อน	INTEGER		

ตารางที่ 6.7 พจนานุกรมตารางข้อมูล FUNCTION_POINT_SYSTEM_COMPONENT

Attribute Name	Description	Data type	PK/FK	Table (FK)
FPSC_ID	รหัส Input/output	NVARCHAR(32)	PK	
PROJ_Code	รหัสโครงการ	NVARCHAR(32)	PK	
FPSC_FunctionType	ชนิด Input/output	NVARCHAR(128)		
FPSC_Low	จำนวนชนิด Low	INTEGER		
FPSC_Medium	จำนวนชนิด Medium	INTEGER		
FPSC_High	จำนวนชนิด High	INTEGER		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.8 พจนานุกรมตารางข้อมูล METHODOLOGY

Attribute Name	Description	Data type	PK/FK	Table (FK)
METH_ID	รหัสกระบวนการ	NVARCHAR(32)	PK	
METH_Name	ชื่อกระบวนการ	NVARCHAR(128)		
METH_DESC	รายละเอียด	TEXT		

ตารางที่ 6.9 พจนานุกรมตารางข้อมูล PERSON_ROLE

Attribute Name	Description	Data type	PK/FK	Table (FK)
PERSON_ID	รหัสผู้ใช้งาน	NVARCHAR(32)	PK	
ROLE_ID	สิทธิ์ผู้ใช้งาน	NVARCHAR(32)	PK	

ตารางที่ 6.10 พจนานุกรมตารางข้อมูล PERSON

Attribute Name	Description	Data type	PK/FK	Table (FK)
PERSON_ID	รหัสผู้ใช้งาน	NVARCHAR(32)	PK	
PERSON_F_Name	ชื่อ	NVARCHAR(128)		
PERSON_L_Name	นามสกุล	NVARCHAR(128)		
PERSON_Gender	เพศ	NVARCHAR(1)		
PERSON_DOB	วันเกิด	DATE		
PERSON_Position	ตำแหน่ง	NVARCHAR(128)		
PERSON_Telephone	เบอร์โทรศัพท์	NVARCHAR(32)		
PERSON_E-Mail	e-Mail	NVARCHAR(128)		
PERSON_Password	รหัสผ่าน	NVARCHAR(16)		

ตารางที่ 6.11 พจนานุกรมตารางข้อมูล PROJECT_STATUS

Attribute Name	Description	Data type	PK/FK	Table (FK)
PROJS_ID	รหัสสถานะ	NVARCHAR(32)	PK	
PROJS_Name	ชื่อสถานะ	NVARCHAR(128)		
PROJS_DESC	รายละเอียด	TEXT		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.12 พจนานุกรมตารางข้อมูล PROJECT

Attribute Name	Description	Data type	PK/FK	Table (FK)
PROJ_Code	รหัสโครงการ	NVARCHAR(32)	PK	
PROJS_ID	รหัสสถานะ	NVARCHAR(32)	FK	
PERSON_ID	รหัสผู้ใช้งาน	NVARCHAR(32)	FK	
PROJ_Name	ชื่อโครงการ	NVARCHAR(255)		
PROJ_DESC	รายละเอียด	TEXT		
PROJ_Category	กลุ่ม	NVARCHAR(32)		
PROJ_CustomerCompany	ชื่อบริษัทลูกค้า	NVARCHAR(128)		
PROJ_CustomerName	ชื่อลูกค้า	NVARCHAR(128)		
PROJ_CustomerProjectOwner	ชื่อลูกค้าเจ้าของโครงการ	NVARCHAR(128)		
PROJ_CustomerProjectManager	ชื่อลูกค้าผู้บริหารโครงการ	CHAR(10)		
PROJ_Objective	จุดประสงค์โครงการ	TEXT		
PROJ_Functional	ความต้องการแบบ Functional	TEXT		
PROJ_NonFunction	ความต้องการแบบ NonFunctional	TEXT		
PROJ_BusinessStrategyPlanning	แผนธุรกิจ	TEXT		
PROJ_TechnologyPlanning	แผนเทคโนโลยี	TEXT		
PROJ_Organozation	โครงสร้างการบริหารโครงการ	TEXT		
METH_ID	รหัสวิธีการพัฒนา	NVARCHAR(32)		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.13 พจนานุกรมตารางข้อมูล RESOURCE_ALLOCATION

Attribute Name	Description	Data type	PK/FK	Table (FK)
RESA_ResourceID	รหัสทรัพยากร	NVARCHAR(32)	PK	
RESA_Name	ชื่อทรัพยากร	NVARCHAR(128)		
RESA_DESC	รายละเอียด	TEXT		
RESA_Cost	ค่าใช้จ่าย	FLOAT		
COSTU_ID	รหัสหน่วยค่าใช้จ่าย	NVARCHAR(32)		
RESC_ID	รหัสกลุ่ม	NVARCHAR(32)		
REST_ID	รหัสชนิด	NVARCHAR(32)		

ตารางที่ 6.14 พจนานุกรมตารางข้อมูล RESOURCE_CATEGORY

Attribute Name	Description	Data type	PK/FK	Table (FK)
RESC_ID	รหัสกลุ่มทรัพยากร	NVARCHAR(32)	PK	
RESC_Name	ชื่อกลุ่มทรัพยากร	NVARCHAR(128)		
RESC_DESC	รายละเอียด	TEXT		

ตารางที่ 6.15 พจนานุกรมตารางข้อมูล RESOURCE_TYPE

Attribute Name	Description	Data type	PK/FK	Table (FK)
REST_ID	รหัสชนิดทรัพยากร	NVARCHAR(32)	PK	
REST_Name	ชื่อชนิดทรัพยากร	NVARCHAR(128)		
REST_DESC	รายละเอียด	TEXT		

ตารางที่ 6.16 พจนานุกรมตารางข้อมูล RISK

Attribute Name	Description	Data type	PK/FK	Table (FK)
RISK_ID	รหัสความเสี่ยง	NVARCHAR(32)	PK	
RISK_DESC	รายละเอียด	TEXT		
RISK_Priority	ลำดับความสำคัญ	INTEGER		
RISK_Probability	ค่าความบ่อย	INTEGER		
RISK_Severity	ค่าความรุนแรง	INTEGER		
RISK_Exposure	ค่าคุณความเสี่ยง	INTEGER		

เอกสารนี้เป็นทรัพย์สินทางปัญญาของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.16 (ต่อ) พจนานุกรมตารางข้อมูล RISK

Attribute Name	Description	Data type	PK/FK	Table (FK)
RISK_ImpactDESC	ผลกระทบ	TEXT		
RISK_FlightingAction	การแก้ไข	TEXT		
RISK_PreventingAction	การป้องกัน	TEXT		
RISK_When	เวลาดำเนินการ	DATE		
RISK_Who	ผู้ดำเนินการ	NVARCHAR(128)		
RISKS_ID	รหัสสถานะ	NVARCHAR(32)	FK	
RISKC_ID	รหัสกลุ่มความเสี่ยง	NVARCHAR(32)	FK	
PROJ_CODE	รหัสโครงการ	NVARCHAR(32)	FK	

ตารางที่ 6.17 พจนานุกรมตารางข้อมูล RISK_CATEGORY

Attribute Name	Description	Data type	PK/FK	Table (FK)
RISKC_ID	รหัสกลุ่มความเสี่ยง	NVARCHAR(32)	PK	
RISKC_Name	ชื่อกลุ่มความเสี่ยง	NVARCHAR(128)		
RISKC_DESC	รายละเอียด	TEXT		

ตารางที่ 6.18 พจนานุกรมตารางข้อมูล RISK_STATUS

Attribute Name	Description	Data type	PK/FK	Table (FK)
RISKS_ID	รหัสสถานะความเสี่ยง	NVARCHAR(32)	PK	
RISKS_Name	ชื่อสถานะความเสี่ยง	NVARCHAR(128)		
RISKS_DESC	รายละเอียด	TEXT		

ตารางที่ 6.19 พจนานุกรมตารางข้อมูล ROLE

Attribute Name	Description	Data type	PK/FK	Table (FK)
ROLE_ID	รหัสสิทธิ์ผู้ใช้	NVARCHAR(32)	PK	
ROLE_Name	ชื่อสิทธิ์ผู้ใช้	NVARCHAR(128)		
ROLE_DESC	รายละเอียด	TEXT		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.13 พจนานุกรมตารางข้อมูล RESOURCE_ALLOCATION

Attribute Name	Description	Data type	PK/FK	Table (FK)
RESA_ResourceID	รหัสทรัพยากร	NVARCHAR(32)	PK	
RESA_Name	ชื่อทรัพยากร	NVARCHAR(128)		
RESA_DESC	รายละเอียด	TEXT		
RESA_Cost	ค่าใช้จ่าย	FLOAT		
COSTU_ID	รหัสหน่วยค่าใช้จ่าย	NVARCHAR(32)		
RESC_ID	รหัสกลุ่ม	NVARCHAR(32)		
REST_ID	รหัสชนิด	NVARCHAR(32)		

ตารางที่ 6.14 พจนานุกรมตารางข้อมูล RESOURCE_CATEGORY

Attribute Name	Description	Data type	PK/FK	Table (FK)
RESC_ID	รหัสกลุ่มทรัพยากร	NVARCHAR(32)	PK	

ตารางที่ 6.20 พจนานุกรมตารางข้อมูล WBS

Attribute Name	Description	Data type	PK/FK	Table (FK)
WBS_Task_ID	รหัสกิจกรรม	NVARCHAR(32)	PK	
PROJ_Code	รหัสโครงการ	NVARCHAR(32)	PK	
RESA_ResourceID	รหัสทรัพยากร	NVARCHAR(32)	PK	
WBS_Duration	ระยะเวลา	FLOAT		
WBS_TaskName	ชื่อกิจกรรม	NVARCHAR(128)		
WBS_DurationUnit	หน่วยเวลา	NVARCHAR(32)		
WBS_Type	ชนิดกิจกรรม	NVARCHAR(128)		
WBS_OutlineLevel	ลำดับในกลุ่ม	INTEGER		
WBS_Predecessor	กิจกรรมที่ทำก่อน	NVARCHAR(32)		
WBS_StartDate	วันเริ่มต้น	CHAR(10)		
WBS_EndDate	วันสิ้นสุด	DATE		
WBS_EarlyStartDate	วันที่เริ่มได้เร็วที่สุด	DATE		
WBS_EarlyFinishDate	วันที่เสร็จเร็วที่สุด	DATE		
WBS_LateStartDate	วันเริ่มต้นที่ช้าที่สุด	DATE		
WBS_LateFinishDate	วันที่เสร็จช้าที่สุด	DATE		
WBS_FLOAT	ค่า Float	FLOAT		

ตารางที่ 6.16 (ต่อ)

Attribute Name	Description	Data type	PK/FK	Table (FK)
RISK_ImpactDESC	ผลกระทบ	TEXT		
RISK_FlightingAction	การแก้ไข	TEXT		
RISK_PreventingAction	การป้องกัน	TEXT		
RISK_When	เวลาดำเนินการ	DATE		
RISK_Who	ผู้ดำเนินการ	NVARCHAR(128)		
RISKS_ID	รหัสสถานะ	NVARCHAR(32)	FK	
RISKC_ID	รหัสกลุ่มความเสี่ยง	NVARCHAR(32)	FK	
PROJ_CODE	รหัสโครงการ	NVARCHAR(32)	FK	

ตารางที่ 6.17 พจนานุกรมตารางข้อมูล RISK_CATEGORY

Attribute Name	Description	Data type	PK/FK	Table (FK)
RISKC_ID	รหัสกลุ่มความเสี่ยง	NVARCHAR(32)	PK	
RISKC_Name	ชื่อกลุ่มความเสี่ยง	NVARCHAR(128)		
RISKC_DESC	รายละเอียด	TEXT		

ตารางที่ 6.18 พจนานุกรมตารางข้อมูล RISK_STATUS

Attribute Name	Description	Data type	PK/FK	Table (FK)
RISKS_ID	รหัสสถานะความเสี่ยง	NVARCHAR(32)	PK	
RISKS_Name	ชื่อสถานะความเสี่ยง	NVARCHAR(128)		
RISKS_DESC	รายละเอียด	TEXT		

ตารางที่ 6.19 พจนานุกรมตารางข้อมูล ROLE

Attribute Name	Description	Data type	PK/FK	Table (FK)
ROLE_ID	รหัสสิทธิ์ผู้ใช้	NVARCHAR(32)	PK	
ROLE_Name	ชื่อสิทธิ์ผู้ใช้	NVARCHAR(128)		
ROLE_DESC	รายละเอียด	TEXT		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.20 พจนานุกรมตารางข้อมูล WBS

Attribute Name	Description	Data type	PK/FK	Table (FK)
WBS_Task_ID	รหัสกิจกรรม	NVARCHAR(32)	PK	
PROJ_Code	รหัสโครงการ	NVARCHAR(32)	PK	
RESA_ResourceID	รหัสทรัพยากร	NVARCHAR(32)	PK	
WBS_Duration	ระยะเวลา	FLOAT		
WBS_TaskName	ชื่อกิจกรรม	NVARCHAR(128)		
WBS_DurationUnit	หน่วยเวลา	NVARCHAR(32)		
WBS_Type	ชนิดกิจกรรม	NVARCHAR(128)		
WBS_OutlineLevel	ลำดับในกลุ่ม	INTEGER		
WBS_Predecessor	กิจกรรมที่ทำก่อน	NVARCHAR(32)		
WBS_StartDate	วันเริ่มต้น	CHAR(10)		
WBS_EndDate	วันสิ้นสุด	DATE		
WBS_EarlyStartDate	วันที่เริ่มได้เร็วที่สุด	DATE		
WBS_EarlyFinishDate	วันที่เสร็จเร็วที่สุด	DATE		
WBS_LateStartDate	วันเริ่มต้นที่ช้าที่สุด	DATE		
WBS_LateFinishDate	วันที่เสร็จช้าที่สุด	DATE		
WBS_FLOAT	ค่า Float	FLOAT		
WBS_SLACK	ค่า Slack	FLOAT		
WBS_Optimistic	ค่า Optimistic	FLOAT		
WBS_MostLikely	ค่า MostLikely	FLOAT		
WBS_Pesstimistic	ค่า Pesstimistic	FLOAT		
WBS_Excepted	ค่า Excepted	FLOAT		
WBS_StandardDeviation	ค่า Standard Deviation	FLOAT		
WBS_UsageEffort	ค่า Usage Effort	FLOAT		
WBS_TaskProgress	ค่า Task Progress	FLOAT		

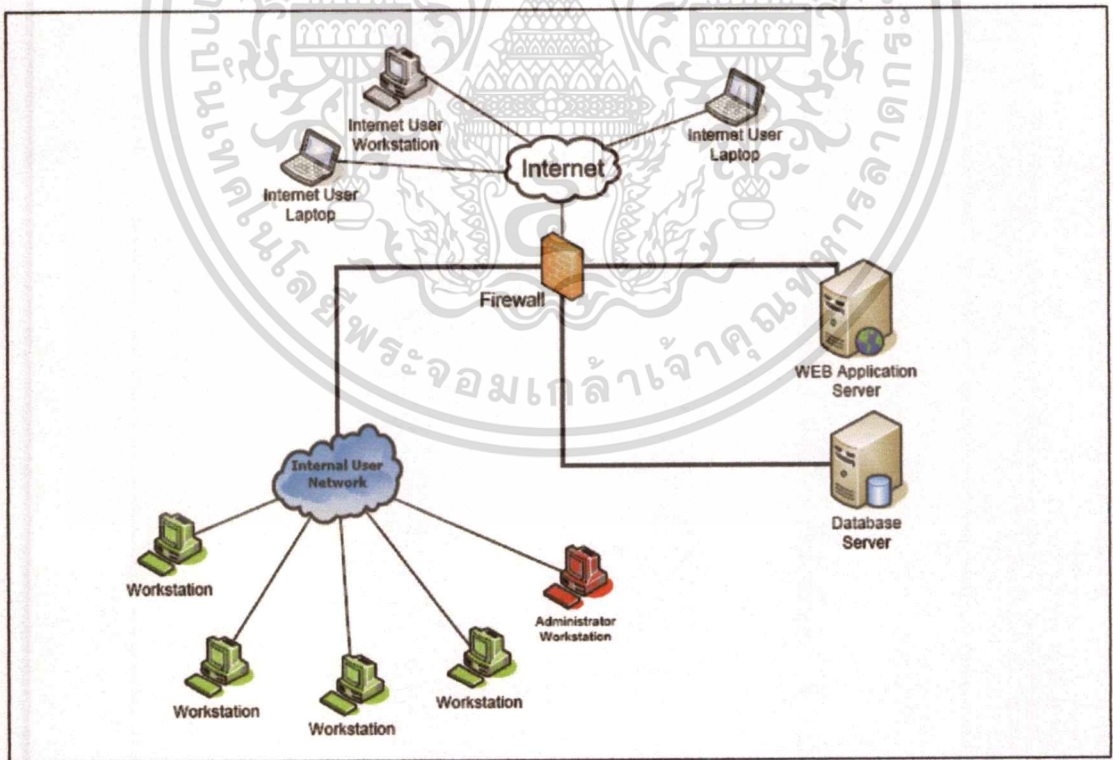
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

การออกแบบโครงสร้างระบบและโครงสร้างหน้าจอของ เว็บแอปพลิเคชัน

7.1 การออกแบบโครงสร้างระบบ

เนื่องจากเครื่องมือบริหารจัดการ โครงการพัฒนาซอฟต์แวร์เป็นระบบที่พัฒนาเป็นแบบเว็บแอปพลิเคชันสามารถใช้งานผ่านอินเทอร์เน็ต และอินเทอร์เน็ตได้ ดังนั้น สำหรับในบทนี้ นอกจากจะกล่าวถึงรายละเอียดในการออกแบบเครือข่ายแล้วยังกล่าวถึงส่วนประสานการใช้งานระหว่างผู้ใช้กับระบบ โดยการออกแบบหน้าจอของเว็บแอปพลิเคชัน โดยมุ่งเน้นไปที่การออกแบบหน้าจอของระบบ ที่ผู้ใช้ระบบสามารถเรียนรู้และเข้าใจการทำงานของระบบได้โดยง่ายและราบรื่น (User Friendly) แผนภาพแสดงด้านล่างเป็นการแสดงการสื่อสารข้อมูลของระบบบนเครือข่ายเพื่อแสดงให้เห็นถึงภาพรวมของระบบดังนี้



รูปที่ 7.1 แผนภาพ โครงสร้างระบบ

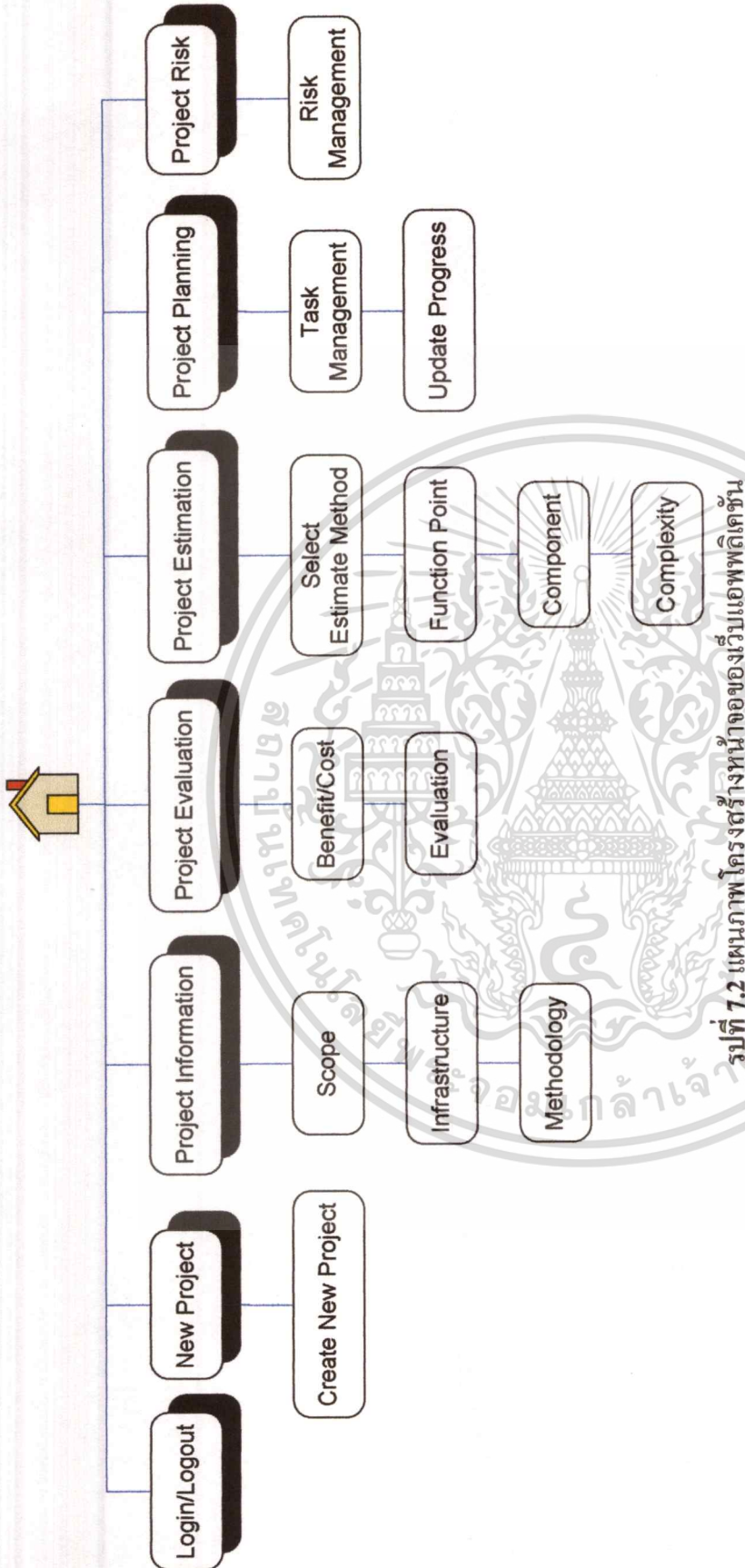
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2 การออกแบบโครงสร้างหน้าจอของเว็บแอปพลิเคชัน

7.2.1 โครงสร้างหน้าจอของเว็บแอปพลิเคชัน

เครื่องมือบริหารจัดการ โครงการพัฒนาซอฟต์แวร์ได้รับการออกแบบให้มีการทำงานแบ่งเป็น 7 ส่วนดังนี้

1. ส่วนการเข้าสู่ระบบ
2. ส่วนการสร้างโครงการใหม่และกำหนดผู้บริหารโครงการ
3. ส่วนการประเมินโครงการ
4. ส่วนการกำหนดรายละเอียดโครงการประกอบด้วย
 - 4.1. กำหนดขอบเขตโครงการ
 - 4.2. กำหนดโครงสร้างโครงการ
 - 4.3. กำหนดกระบวนการพัฒนา
5. ส่วนการประเมินขนาดของโครงการ
6. ส่วนการกำหนดความเสี่ยงของโครงการ
7. ส่วนการกำหนดแผนการดำเนินโครงการประกอบด้วย
 - 7.1. การกำหนดกิจกรรมในโครงการ
 - 7.2. กำหนดค่าต่างๆของกิจกรรม
 - 7.3. กำหนดทรัพยากรที่ใช้ในกิจกรรม
 - 7.4. กำหนดความก้าวหน้าของกิจกรรมและทรัพยากรที่ใช้



รูปที่ 7.2 แผนภาพโครงสร้างหน้าของเว็บไซต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.2 หน้าจอของเว็บแอปพลิเคชัน

7.2.2.1 หน้าจอ New Project

Project Management Tools

[Login](#)
[New Project](#)
[Project Information](#)
[Project Evaluation](#)
[Project Estimation](#)
[Project Planning](#)
[Project Risk](#)

Create New Project

Project Code:

Project Name:

Project Description:

Project Category:

Customer Company:

Customer Name:

Customer Project Owner:

Customer Project manager:

Project Status:

รูปที่ 7.3 หน้าจอ New Project

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.2.2 หน้าจอ Project Information

The screenshot shows a web-based form for entering project information. At the top, there is a navigation bar with links: Login, New Project, Project Information, Project Evaluation, Project Estimation, Project Planning, and Project Risk. The main form area is titled 'Project Information' and contains the following fields:

- Project Code:** 0001
- Project Name:** Project Management Tools
- Project Scope:** A large empty text area.
- Project Infrastructure:** A large empty text area.
- Implementation Methodology:** A dropdown menu currently showing 'Water Fall'.

At the bottom of the form, there are two buttons: 'Update' and 'Cancel'.

รูปที่ 7.4 หน้าจอ Project Information

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.2.3 Project Evaluation

Project Management Tools

[Login](#) : [New Project](#) : [Project Information](#) : [Project Evaluation](#) : [Project Estimation](#) : [Project Planning](#) : [Project Risk](#)

Project Evaluation

Project Code: 0001

Project Name: Project Management Tools

Discount Factor : %

[+ Add Benefit] [- Delete Benefit]

Benefit Information	Year Number	Amount (Baht)
<input type="checkbox"/> Revenue 1	1	34,825.00
<input type="checkbox"/> Revenue 2	2	10,945.00
<input type="checkbox"/> Revenue 3	3	1,236,525.00

Project Benefit:

[+ Add Cost] [- Delete Cost]

Cost Information	Year Number	Amount (Baht)
<input type="checkbox"/> Software Licenses	1	34,825.00
<input type="checkbox"/> Server Software	0	10,945.00
<input type="checkbox"/> Development Labor	0	1,236,525.00

Project Cost:

Evaluation Information	Value
Net Profit	700,201.00
Percent Value	690,201.00
Net Present Value	629,421.00
Return On Investment	24.44
Break-Event Point	3.63

Evaluation Value:

รูปที่ 7.5 หน้าจอ Project Evaluation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.2.4 Project Estimation

Project Estimation

Project Code: 0001

Project Name: Project Management Tools

Estimation Type: Function Point

Development Language: JSP

Description	Total Number	Complexity			Total
		Low	Medium	High	
Inputs	6	3	2	1	23
Outputs	19	4	10	5	101
Queries	10	7	0	3	39
Files	15	0	15	0	150
Program Interface	3	1	0	2	25
Total Unadjusted Function Point (TUFP):					338

System Component :

[+ Add Complexity] [- Delete Complexity]

Complexity Information	Value
<input type="checkbox"/> Heavy use configuration	3
<input type="checkbox"/> Transaction rate	2
<input type="checkbox"/> End-User efficiency	2
<input type="checkbox"/> Multiple site	1
<input type="checkbox"/> Performance	4
Total Processing Complexity:	12

Overall System Complexity :

Estimation Information	Value
Adjusted Processing Complexity	0.72
Total Adjusted Function Points	243
Line Of Code	14,000
Effort (in person-months)	5.3
Schedule time (months)	3.2

Estimation Value :

Estimate Cancel

รูปที่ 7.6 หน้าจอ Project Estimation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.2.5 Project Planning

Project Management Tools

Project Planning

Project Code: 0001

Project Name: Project Management Tools

[+ Add Task] [- Delete Task]

No.	Task	Duration	Duration Unit	Start Date	End Date	Predecessor
<input type="checkbox"/> 1	Scope	5	Day	12/3/2007	16/3/2007	
<input type="checkbox"/> 2	Analysis	10	Day	17/3/2007	26/3/2007	1
<input type="checkbox"/> 3	Design	15	Day	27/3/2007	10/4/2007	2
<input type="checkbox"/> 4	Develop	45	Day	17/4/2007	31/5/2007	3
<input type="checkbox"/> 5	Testing	10	Day	1/6/2007	10/6/2007	4

Update Cancel

รูปที่ 7.7 หน้าจอ Project Planning

7.2.2.6 Project Risk

Project Management Tools

Project Risk

Project Code: 0001

Project Name: Project Management Tools

No.	Risk	Category	Priority	Probability [1,..,9]	Severity [0,..,5]	Risk Exposure
<input type="checkbox"/> 1	A	Technical	10	5	3	15
<input type="checkbox"/> 2	B	Business	20	4	2	8

Update Cancel

รูปที่ 7.8 หน้าจอ Project Risk

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8

การทดสอบและสรุปผล

8.1 การทดสอบระบบ

หลังจากออกแบบระบบเป็นที่เรียบร้อยแล้ว ขั้นตอนต่อไปจะเป็นการทดสอบการทำงานของระบบ โดยตรวจสอบความถูกต้อง และหาข้อผิดพลาดที่เกิดขึ้นอันเนื่องมาจากการเขียนโปรแกรม (Debugging) โดยทำไปทีละ Module (Unit Testing) จากนั้นจะทำการรวบรวม Module มาประกอบรวมกันทดสอบ แล้วทำการทดสอบการทำงานทั้งระบบอีกครั้ง (Integration & System Testing) ถัดไปจะเป็นการป้อนข้อมูลสมมติเข้าสู่ระบบเพื่อดูผลลัพธ์ที่เกิดขึ้น (Alpha Testing) รวมถึงให้พนักงานในองค์กรที่มีส่วนเกี่ยวข้องทั้ง Front-end และ Back-end ได้ร่วมทดลองใช้ระบบในสภาพแวดล้อมจริง (Beta Testing) เพื่อยืนยันความสมบูรณ์ของระบบ และความพึงพอใจในการใช้งาน หากผู้ใช้ระบบมีข้อสงสัยหรือต้องการเพิ่มเติม Requirement คณะผู้พัฒนาจะทำการจดบันทึกข้อเรียกร้องนั้นไว้ แล้วนำไปวางแผนเพื่อทำการพัฒนาระบบต่อไป

8.2 สรุปผล

โครงการศึกษากรณีพิเศษฉบับนี้ เป็นการศึกษาขั้นตอนการบริหารโครงการพัฒนาซอฟต์แวร์และออกแบบเครื่องมือบริหารจัดการโครงการพัฒนาซอฟต์แวร์ เป็นการออกแบบระบบเว็บแอปพลิเคชันทำงานผ่านระบบเครือข่ายอินเทอร์เน็ตและอินเทอร์เน็ต ในการศึกษาใช้ทฤษฎีการบริหารโครงการสเตปไวส์ (Step Wise) เป็นหลักและศึกษาทฤษฎีอื่นๆเพิ่มเติม ในการออกแบบใช้หลักคิดเชิงวัตถุ โดยนำภาษายูเอ็มแอลมาใช้ในการวิเคราะห์และออกแบบระบบสารสนเทศและจัดทำแผนภาพเชิงสัมพันธ์ระหว่างเอนทิตีในการออกแบบฐานข้อมูลเชิงสัมพันธ์ ซึ่งเครื่องมือที่ใช้ได้แก่ อีเอสทูดีโอ 6.6.1 (ER Studio 6.6.1)

จากการศึกษาและออกแบบพบว่าเครื่องมือบริหารจัดการโครงการพัฒนาซอฟต์แวร์สามารถใช้ประโยชน์ได้ครอบคลุมในการบริหารงานโครงการพัฒนาซอฟต์แวร์ได้เป็นอย่างดี อีกทั้งยังสามารถช่วยให้ผู้บริหารโครงการสามารถประเมินค่าต่างๆที่เป็นประโยชน์ในการบริหารงานโครงการเช่น ค่าความเสี่ยง ได้อีกด้วย

8.3 ปัญหา ข้อจำกัด และข้อเสนอแนะ

เนื่องจากขั้นตอนในการบริหารงาน โครงการพัฒนาซอฟต์แวร์มีความซับซ้อนและมีวิธีการดำเนินการที่หลากหลายเช่นการประเมินขนาดของโครงการ ในเครื่องมือบริหารจัดการ โครงการพัฒนาซอฟต์แวร์ที่ออกแบบเลือกใช้วิธีการ ฟังก์ชันพอร์ต (Function Point) แต่ก็ออกแบบให้สามารถพัฒนาวิธีอื่นๆเพิ่มเติมภายหลังซึ่งจะเป็นประโยชน์อย่างมากต่อการประเมินขนาดของโครงการ

ในด้านการแสดงผลเนื่องจากการพัฒนาแบบเว็บแอปพลิเคชันจึงมีข้อจำกัดบางประการในการแสดงผลทางหน้าจอ อาจดีน้อยกว่าเครื่องมือที่มีใช้อยู่เช่น ไมโครซอฟต์โปรเจ็ค (Microsoft Project) ซึ่งเป็นเดสทอปแอปพลิเคชันแสดงผลได้สวยงามและสื่อความหมายได้ดีกว่า โดยเฉพาะการแสดงความสัมพันธ์ระหว่างกิจกรรมต่างๆ

ควรเพิ่มเติมการนำเข้าข้อมูลจากภายนอกเช่นการนำเข้าข้อมูลจากโปรแกรม ไมโครซอฟต์โปรเจ็ค (Microsoft Project) ซึ่งจะเป็นประโยชน์อย่างมากในการทำงานเนื่องจากโปรแกรม ไมโครซอฟต์โปรเจ็ค (Microsoft Project) เป็นโปรแกรมเดสทอปโปรแกรมสามารถทำงานโดยไม่ต้องออนไลน์ได้ และควรพัฒนาการนำเข้าข้อมูลออกตามมาตรฐานโปรแกรมบริหารงานโครงการที่เป็นที่นิยมเช่น ไมโครซอฟต์โปรเจ็ค (Microsoft Project) เพื่อให้สามารถใช้งานร่วมกันได้อย่างมีประสิทธิภาพ

บรรณานุกรม

Alan Dennis and Barbara Haley Wixon ET.AL. 2005. **System Analysis And Design With UML. Version 2.0 An Object-Oriented Approach. Second Edition CA: WILEY**



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อผู้เขียน	นายสุรเชษฐ์ เฉลิมผล
วันเดือนปีเกิด	9 เมษายน 2517
วุฒิการศึกษาระดับปริญญาตรี	วิศวกรรมไฟฟ้า โรงเรียนนายเรืออากาศ
การทำงาน	2541 – ปัจจุบัน อาจารย์ประจำภาควิชาวิศวกรรมโรงเรียนนายเรืออากาศ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้