

การพัฒนาระบบจัดการข้อมูลและคำนวณระยะเวลา

ดำเนินงานของโครงการ

Project Management and Estimate Project Time System

ธนาดี กระทุ้ง

580510599

การค้นคว้าอิสระนี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาบริการคอมพิวเตอร์
คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่
ปีการศึกษา 2561

การพัฒนาระบบจัดการข้อมูลและคำนวณระยะเวลา
ดำเนินงานของโครงการ

Project Management and Estimate Project Time System

รับดี กระทุ้

580510599

การค้นคว้าอิสระนี้ได้รับการพิจารณาอนุมัติให้นับเป็นส่วนหนึ่งของการศึกษา^๑
ตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่
ปีการศึกษา 2561

คณะกรรมการสอบการค้นคว้าอิสระ

..... ประธานกรรมการ

(อ.ดร.จักริน ชาชาติ)

..... กรรมการ

(อ.ดร.ประภาพร เตชะอังกูร)

วันที่.....เดือน..... พ.ศ.....

กิตติกรรมประกาศ

การค้นคว้าอิสระเล่มนี้สำเร็จลุล่วงได้ด้วยความกรุณาจากบุคคลหลายท่าน ขอกราบขอบพระคุณอาจารย์ ดร.จักริน ชาชาติ อาจารย์ที่ปรึกษาการค้นคว้าอิสระ ผู้ซึ่งกรุณาให้ความรู้ คำแนะนำ คำปรึกษา รวมทั้งสละเวลาตรวจสอบแก้ไขข้อบกพร่องของเนื้อหาและสำนวนภาษา จนการค้นคว้านี้เสร็จสมบูรณ์

ขอกราบขอบพระคุณอาจารย์ ดร.ประภาพร เตชอังกูร กรรมการสอบการค้นคว้าอิสระ ซึ่งได้ให้คำแนะนำเป็นอย่างดีมาโดยตลอด ขอขอบคุณคณาจารย์ที่ได้ให้การสนับสนุนการทำงาน ต่าง ๆ ขอขอบคุณเจ้าหน้าที่ภาควิชาวิทยาการคอมพิวเตอร์ ที่ได้ให้ความสะดวกและให้ความช่วยเหลือในด้านต่าง ๆ เป็นอย่างดี

ขอขอบคุณพี่ ๆ จากบริษัท Banana Coding จำกัด ที่ได้ให้คำแนะนำในการทำงาน รวมไปถึงให้คำปรึกษาในเรื่องแนวคิดการทำงาน การวางแผนของการทำงานต่าง ๆ โดยเฉพาะอย่างยิ่ง ขอขอบคุณ พี่ปียก จิระพงษ์ นันตา ประธานบริษัท Banana Coding จำกัด ที่ได้ช่วยเหลือด้านข้อมูล และให้คำแนะนำในด้านทฤษฎีที่เกี่ยวข้องกับหัวข้อการค้นคว้าอิสระนี้

นอกจากนี้ ขอขอบคุณทุกความช่วยเหลือ ทั้งทางตรงและทางอ้อมที่มิได้กล่าวไว้ ณ ที่นี่ ซึ่งท่านเหล่านั้นได้กรุณาให้คำแนะนำในการทำการค้นคว้าอิสระให้สำเร็จลุล่วงไปด้วยดี

ชนบดี กระทุ้
580510599

หัวข้อการค้นคว้าอิสระ	การพัฒนาระบบจัดการข้อมูลและคำนวณระยะเวลาดำเนินงานของโครงการ	
ชื่อเจ้าของโครงการ	นายธนบดี กระทุ้น	รหัสประจำตัว 580510599
วิทยาศาสตรบัณฑิต	สาขาวิทยาการคอมพิวเตอร์	
อาจารย์ที่ปรึกษา	ดร.จักริน ชวชาติ	

บทคัดย่อ

การพัฒนาระบบสำหรับจัดการโครงการและคาดการณ์ระยะเวลาโครงการ มีวัตถุประสงค์ใน การศึกษาและพัฒนาระบบที่สามารถอ่านวิเคราะห์ความลึกซึ้งของข้อมูลที่ได้รับ ให้กับการจัดการโครงการภายในองค์กร ต่าง ๆ และมีระบบคำนวณระยะเวลาที่ใช้ในการทำงานของแต่ละโครงการที่ช่วยให้ผู้จัดการโครงการ สามารถติดตามและตรวจสอบการทำงานได้โดยตลอด ในการวางแผนงานมากขึ้น

การทำงานของเว็บไซต์ประกอบด้วยการทำงาน 3 ส่วน ได้แก่ การจัดการโครงการ การจัดการสมาชิกในองค์กร สรุประยุทธ์งานผลประสิทธิภาพของบุคลากรในองค์กร ซึ่งผู้ใช้งานระบบนี้ แบ่งเป็น 3 ประเภท ได้แก่ พนักงานทั่วไป ผู้จัดการโครงการ และเจ้าของบัญชีองค์กร โดยพนักงาน ทั่วไปจะทำการบันทึกเวลาการทำงานในแต่ละโครงการเข้าสู่ระบบ ผู้จัดการโครงการจะสามารถเข้ามาดูรายงานผลประสิทธิภาพของบุคลากรในองค์กร และสามารถเข้าจัดการโครงการแต่ละโครงการได้ ซึ่งในระบบการจัดการโครงการ จะมีระบบคาดการณ์ระยะเวลาที่ใช้ในการทำงานของโครงการแต่ละ โครงการ โดยใช้กระบวนการวิเคราะห์ความถดถอยอย่างง่ายร่วมกับการคำนวณประสิทธิภาพของ พนักงาน เพื่อคำนวณหาระยะเวลาของแต่ละโครงการได้ เจ้าของบัญชีองค์กรสามารถเพิ่มรายชื่อ พนักงานในองค์กรเข้ามาใช้งานระบบ และเจ้าของบัญชีองค์กรสามารถเข้าตั้งค่าต่าง ๆ ที่ใช้ภายใน ระบบจัดการโครงการได้

ระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ ทำให้บริษัทสามารถใช้ ระบบเพื่อจัดการข้อมูลโครงการ รวมทั้งสามารถใช้ระบบเพื่อคาดการณ์ระยะเวลาที่เหมาะสมของ โครงการได้ ช่วยให้ผู้จัดการโครงการและบริษัท สามารถวางแผนการดำเนินงานได้สะดวกยิ่งขึ้น

Independent Study Title	Project Management and Estimate Project Time System	
Author	Mr. Thanabodee Krathu	Student ID 580510599
Bachelor of Science		Computer Science
Supervisor	Jakin Chawachat	

Abstract

The objective of Project Management and Estimate Project Time System is to study and develop a project management system to facilitate for organizations and create a function estimate work time for each project. The function helps project managers more easily planning project.

This project consists of 3 parts: Project management, Managing members in the organization, and Personal performance reports in the organization. The users of this system can be divided into 3 types: Employees, Project Managers, and Organization Account Holders. Employees record their working time in each project into the system. Project managers can see the performance reports of each employee in the organization and can manage each project information. In the project management system, there is a system for estimate work time of each project by using simple linear regression analysis with employee performance to calculate the estimated time of the project. Account holders can add employees in the organization to use the system and can access various settings used within the project management system.

Project Management and Estimate Project Time System provides the organization to manage and estimate work time for each project and also helps project managers and organization to plan workload more easily.

สารบัญ

หัวข้อ	หน้า
กิตติกรรมประกาศ	ก
บทคัดย่อ	ข
Abstract.....	ค
สารบัญ	ง
สารบัญตาราง	ฉ
สารบัญภาพ	ช
บทที่ 1 บทนำ	1
1.1 หลักการและเหตุผล	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ประโยชน์ที่จะได้รับจากการศึกษาเชิงประยุกต์	1
1.4 ขอบเขตของโครงการ	2
1.5 แผนการดำเนินงานและระยะเวลาดำเนินงาน	3
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง	4
2.1 การประมาณเวลาการทำงาน.....	4
2.2 การวิเคราะห์การถดถอยเชิงเส้นอย่างง่าย	6
2.3 แนวคิดการทำงานแบบ Scrum.....	7
2.4 Scrum Burndown Chart	10
2.5 เทคโนโลยีที่เกี่ยวข้องกับระบบ	10
บทที่ 3 การวิเคราะห์และออกแบบระบบ	13
3.1 การคาดการณ์ระยะเวลาของโปรเจค.....	13
3.2 ประสิทธิภาพการทำงานของพนักงาน	16
3.3 แผนภาพยุสเคส	20
3.4 แผนภาพกิจกรรม	23
3.5 แผนภาพซีเควนซ์โดยแกรม	26
บทที่ 4 การออกแบบฐานข้อมูล.....	30
4.1 ลักษณะของตารางความสัมพันธ์ของฐานข้อมูล	30
4.2 รายละเอียดข้อมูลของตาราง	31
บทที่ 5 การออกแบบส่วนต่อประสานกับผู้ใช้	35
5.1 สิทธิ์ในการเข้าถึงในแต่ละหน้าจอ	35
5.2 การออกแบบส่วนต่อประสานกับผู้ใช้.....	35

สารบัญ (ต่อ)

หัวข้อ	หน้า
บทที่ 6 การทดสอบโปรแกรม	40
6.1 การทดสอบระบบส่วนจัดการข้อมูลโปรเจค.....	40
6.2 การทดสอบระบบคาดการณ์ระยะเวลาของโครงการ.....	44
บทที่ 7 สรุปผลการดำเนินการ	49
7.1 สรุปผลการดำเนินงาน	49
7.2 ข้อจำกัดของระบบ.....	49
7.3 แนวทางในการพัฒนาต่อ	49
เอกสารอ้างอิง	51
คู่มือการใช้งานระบบ.....	53
คู่มือการติดตั้งระบบ.....	60

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 1.1 ขั้นตอนการดำเนินงานและระยะเวลาการดำเนินงาน	3
ตารางที่ 3.1 ตัวอย่างข้อมูลการทำงานของแต่ละฟีเจอร์ในบริษัทแห่งหนึ่ง.....	14
ตารางที่ 3.2 ตัวอย่างข้อมูลประสิทธิภาพการทำงานของพนักงานในบริษัทแห่งหนึ่ง	17
ตารางที่ 3.3 ตัวอย่างข้อมูลการทำงานของพนักงานในฟีเจอร์.....	18
ตารางที่ 3.4 ผลลัพธ์การปรับค่า Performance ของสมาชิก.....	20
ตารางที่ 3.5 สัญลักษณ์และความหมายของแผนภาพสูตร.....	20
ตารางที่ 3.6 รายละเอียด Use Case “Manage Member”	21
ตารางที่ 3.7 รายละเอียด Use Case “Manage project”	22
ตารางที่ 3.8 รายละเอียด Use Case “Create Project”	22
ตารางที่ 3.9 รายละเอียด Use Case “Edit Project”	22
ตารางที่ 3.10 รายละเอียด Use Case “Estimate Time For Feature And Project”	23
ตารางที่ 3.11 รายละเอียด Use Case “Report”	23
ตารางที่ 3.12 สัญลักษณ์และความหมายของแผนภาพกิจกรรม.....	23
ตารางที่ 3.13 สัญลักษณ์และความหมายของแผนภาพชีวิตประจำวัน	26
ตารางที่ 4.1 ความหมายและสัญลักษณ์ของตารางความสัมพันธ์ของฐานข้อมูล	30
ตารางที่ 4.2 รายละเอียดตารางข้อมูลผู้ใช้งาน.....	31
ตารางที่ 4.3 รายละเอียดตารางบริษัท.....	31
ตารางที่ 4.4 รายละเอียดตารางประเภทของผู้ใช้งาน	32
ตารางที่ 4.5 รายละเอียดตารางโปรเจค.....	32
ตารางที่ 4.6 รายละเอียดตารางสมาชิกในโปรเจค	32
ตารางที่ 4.7 รายละเอียดตารางฟีเจอร์	33
ตารางที่ 4.8 รายละเอียดตารางสมาชิกในฟีเจอร์	33
ตารางที่ 4.9 รายละเอียดตารางงานนับเวลา	34
ตารางที่ 5.1 การเข้าถึงหน้าจอของผู้ใช้แต่ละประเภท.....	35
ตารางที่ 6.1 การทดสอบการสร้างโปรเจค.....	40
ตารางที่ 6.2 การทดสอบการแก้ไขโปรเจค	41
ตารางที่ 6.3 การทดสอบการลบโปรเจค.....	42
ตารางที่ 6.4 การทดสอบการแสดงรายงานของระบบ	43
ตารางที่ 6.5 การทดสอบการตั้งค่าของระบบ	43
ตารางที่ 6.6 ตัวอย่างข้อมูลบันทึกการนับเวลาของแต่ละโปรเจคในระบบ.....	44

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
ตารางที่ 6.7 ข้อมูลการสร้างโปรเจคสำหรับทดสอบระบบ	46
ตารางที่ 6.8 ข้อมูลสมาชิกในฟีเจอร์ทดสอบระบบ	46
ตารางที่ 6.9 ข้อมูลการทำงานของสมาชิกในโปรเจค Project Black Hat 1.0	47
ตารางที่ 6.10 ตารางเปรียบเทียบข้อมูลที่ได้จากการคาดการณ์ระยะเวลา	48

สารบัญภาพ

ภาพที่	หน้า
ภาพที่ 2.1 ตัวอย่าง Simple Regression Line.....	7
ภาพที่ 2.2 ตัวอย่างกราฟ Product Burndown.....	10
ภาพที่ 3.1 ตัวอย่างกราฟการลดต้อยของบริษัท 1 บริษัท	14
ภาพที่ 3.2 แผนภาพยุสเคสระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ	21
ภาพที่ 3.3 แผนภาพกิจกรรมระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ .	25
ภาพที่ 3.4 ชีวนะ์โดยแกรมการจัดการข้อมูลโปรเจค	27
ภาพที่ 3.5 ชีวนะ์โดยแกรมการสร้างโปรเจค.....	28
ภาพที่ 3.6 ชีวนะ์โดยแกรมการแก้ไขข้อมูลโปรเจค.....	29
ภาพที่ 4.1 ตารางความสัมพันธ์ของฐานข้อมูล.....	30
ภาพที่ 5.1 หน้าจอสำหรับจัดการข้อมูลโปรเจคในระบบ	36
ภาพที่ 5.2 หน้าจอการสร้างโปรเจคในขั้นตอนแรก	36
ภาพที่ 5.3 ก หน้าจอการสร้างโปรเจคในขั้นตอนที่สอง	37
ภาพที่ 5.3 ข หน้าจอการสร้างโปรเจคในขั้นตอนที่สอง.....	37
ภาพที่ 5.4 ก หน้าจอแสดงรายงานของระบบ	38
ภาพที่ 5.4 ข หน้าจอแสดงรายงานของระบบ	38
ภาพที่ 5.5 หน้าจอจัดการข้อมูลสมาชิก.....	39
ภาพที่ 5.6 หน้าจอตั้งค่าข้อมูลบริษัท.....	39
ภาพที่ 6.1 การสร้างโปรเจคเพื่อทดสอบระบบ	45
ภาพที่ 6.2 เลือกสมาชิกในแต่ละไฟเจอร์ของการทดสอบระบบ	46
ภาพที่ 6.3 ข้อมูลภายในไฟเจอร์ และปุ่มเสร็จสิ้นของแต่ไฟเจอร์ในโปรเจค	47
ภาพที่ 6.4 ผลลัพธ์การสร้างโปรเจคด้วยข้อมูลชุดเดิม	48
ภาพที่ ก.1 หน้าจอสร้างบัญชีเจ้าของบริษัท	53
ภาพที่ ก.2 หน้าจอเข้าสู่ระบบ	53
ภาพที่ ก.3 แบบเมนูของ Account Holder.....	54
ภาพที่ ก.4 แบบเมนูของ Project Manager และ Employee	54
ภาพที่ ก.5 หน้าจอจัดการสมาชิก	54
ภาพที่ ก.6 หน้าจอกองข้อมูลเพื่อเพิ่มสมาชิกเข้าสู่ระบบ	55
ภาพที่ ก.7 หน้าจอจัดการข้อมูลโปรเจค	55
ภาพที่ ก.8 หน้าจอสร้างโปรเจคและไฟเจอร์	56
ภาพที่ ก.9 หน้าจอสำหรับจัดการสมาชิกในไฟเจอร์	57

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
ภาพที่ ก.10 หน้าจอจัดการข้อมูลโครงการ หลังจากการสร้างโครงการ.....	58
ภาพที่ ก.11 หน้าจอแสดงรายงานของระบบ	59
ภาพที่ ก.12 หน้าจอการตั้งค่าระบบ	59

บทที่ 1

บทนำ

การพัฒนาระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ ในบทนี้จะกล่าวถึงหลักการและเหตุผล วัตถุประสงค์ของโครงการ ประโยชน์ที่จะได้รับจากการศึกษาเชิงประยุกต์ และขอบเขตของโครงการ รวมไปถึงแผนการดำเนินงาน และระยะเวลาการดำเนินงาน เพื่อให้เข้าใจระบบงานโดยรวม

1.1 หลักการและเหตุผล

การทำงานภายใต้บริบทหลาย ๆ แห่ง ได้ให้พนักงานทำการจับเวลาการทำงานในแต่ละวัน และแต่ละชั้นงาน เพื่อตรวจสอบว่าในแต่ละวันพนักงานนั้นทำงานอะไรบ้าง และแต่ละชั้นงานที่ได้รับมอบหมายพนักงานใช้เวลาทำเท่าใด และใช้เวลาตรงตามเป้าหมายหรือไม่ ซึ่งแต่ละบริษัทก็จะต้องมีการตั้งเป้าหมายไว้ว่าแต่ละชั้นงาน จะต้องใช้ระยะเวลาการทำเท่าใดจึงจะเหมาะสมที่สุด

อย่างไรก็ตาม การตั้งเป้าหมายของแต่ละบริษัทนั้น จะใช้ประสบการณ์ของผู้จัดการโครงการ เพื่อทำการประเมินและตั้งระยะเวลาดำเนินการขึ้นมา แต่ก็ยังพบว่าระยะเวลาที่ตั้งมาบ้านอาจไม่สอดคล้องกับประสิทธิภาพของพนักงาน หรืออาจยังมีความคลาดเคลื่อนอยู่บ้าง อีกทั้งข้อมูลระยะเวลาการทำงานที่ถูกบันทึกไว้ ยังนำมาใช้งานเพียงแค่การอกรายงานบันทึกผลเท่านั้น ซึ่งยังไม่ได้ใช้ประโยชน์จากข้อมูลตรงนี้มากนัก

ด้วยปัญหาเหล่านี้ ผู้จัดทำจึงได้ศึกษาและพัฒนาระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ โดยพัฒนาไฟล์เจอร์การคาดการณ์ระยะเวลาที่เหมาะสมกับการทำงานแต่ละชั้น โดยใช้การวิเคราะห์การถดถอย (Regression Analysis) ร่วมกับการวัดประสิทธิภาพของพนักงาน (Performance) เพื่อเพิ่มความสะดวกในการวางแผนการดำเนินงานของการทำงาน ให้สามารถคาดการณ์ระยะเวลาที่เหมาะสมสำหรับการทำงานได้่ายั่งขึ้น

1.2 วัตถุประสงค์ของโครงการ

เพื่อพัฒนาระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ ให้มีประสิทธิภาพ สามารถใช้คำนวณระยะเวลาการทำงานที่เหมาะสมของแต่ละโครงการได้อย่างแม่นยำ

1.3 ประโยชน์ที่จะได้รับจากการศึกษาเชิงประยุกต์

- 1) ได้นำข้อมูลการนับเวลา มาประยุกต์ใช้กับการวางแผนงาน
- 2) ได้ระบบที่ทำให้การวางแผนการดำเนินงานมีความสะดวกมากขึ้น สามารถคาดการณ์ระยะเวลาที่เหมาะสมกับการทำงานได้

3) ทำให้ผู้พัฒนาได้รู้จักการเรียนรู้ และนำเทคโนโลยีต่าง ๆ มาใช้ได้หลากหลายมากยิ่งขึ้น

1.4 ขอบเขตของโครงการ

ขอบเขตของโครงการประกอบด้วย ขอบเขตทางสถาปัตยกรรม ขอบเขตของระบบงาน และ ขอบเขตของข้อมูล มีรายละเอียดดังนี้

1) ขอบเขตทางสถาปัตยกรรม

ระบบที่ได้ทำการพัฒนาขึ้นนี้ เป็นเว็บแอปพลิเคชัน (Web Application)

1.1) ฮาร์ดแวร์ (Hardware) ที่ใช้ในการพัฒนาระบบ มีรายละเอียดดังนี้

- คอมพิวเตอร์โน๊ตบุ๊ครุ่น MacBook Pro Early 2011
- หน่วยประมวลผลกลาง (CPU) รุ่น Intel® Core™ i5 Processors ความเร็ว 2.3 GHz
- หน่วยความจำหลัก (Ram) ขนาด 6.00 GB
- ฮาร์ดดิสก์ (Hard Disk) ขนาดความจุ 120 GB

1.2) ซอฟต์แวร์ (Software) ที่ใช้ในการพัฒนาระบบ มีรายละเอียดดังนี้

- ระบบปฏิบัติการ MacOS High Sierra เวอร์ชัน 10.13
- ระบบจัดการฐานข้อมูลโพสต์เกรสคิวแอล (PostgreSQL)
- รูบีอ่อนเรลส์เฟรมเวิร์ค (Ruby on Rails Framework) ใช้ในการพัฒนาเว็บไซต์ ร่วมกับภาษารูบี (Ruby) ภาษาเชซทีเอ็มแอล (HTML) และใช้เครื่องมือซีอสเซส (CSS Framework) ในการปรับแต่งส่วนการแสดงผล โดยในระบบนี้ใช้ Ruby on Rails Framework เวอร์ชัน 5.2.0
- เครื่องมือพัฒนาโปรแกรม Sublime Text เวอร์ชัน 3.1.1

2) ขอบเขตของระบบงาน

ระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ ได้แบ่งงานของระบบออกเป็น 2 ส่วน ดังนี้

2.1) การคาดการณ์ระยะเวลาของโปรเจค

ในแต่ละโปรเจค จะมีการวางแผนการดำเนินงานไว้ว่างานนั้นควรจะใช้ระยะเวลาเท่าใดในการทำงาน ซึ่งโดยทั่วไปจะใช้การคาดการณ์จากประสบการณ์ของผู้ทำงานนั้น ๆ เอง ซึ่งในแต่ละโปรเจคนั้น จะมีการแบ่งงานออกเป็นงานย่อย ๆ หรือเรียกว่าฟีเจอร์ (Feature) โดยแต่ละฟีเจอร์ย่อย ๆ นั้น จะถูกนำระบบการคาดการณ์ระยะเวลาการทำงานที่เหมาะสมของโปรเจคมาใช้งานด้วย

การทำงานของระบบคาดการณ์ระยะเวลาของโปรเจคนั้น จะช่วยให้ผู้ใช้งานสามารถคาดการณ์ระยะเวลาการทำงานให้ถูกต้องขึ้น โดยจะทำงานเมื่อผู้ใช้งานทำการสร้างโปรเจคขึ้นมาใหม่ โดยผู้ใช้จะต้องเลือกสมาชิกในโปรเจค จากนั้นให้ผู้ใช้สร้างฟีเจอร์อย่างภายในโปรเจคขึ้นมา ในแต่ละ

ฟีเจอร์ จะให้ผู้ใช้เลือกสมาชิกในฟีเจอร์ และให้ระบุขนาดของงานในระดับ 1-10 เพื่อนำข้อมูลที่ผู้ใช้เลือกไปคำนวณช่วงเวลาการทำงานให้ว่า หากใช้สมาชิกทีมนี้ และฟีเจอร์มีขนาดของงานเท่านี้ จะใช้เวลาการดำเนินงานเท่าใด และแสดงผลลัพธ์ในแต่ละฟีเจอร์ และนำมาแสดงผลรวมเวลาของแต่ละฟีเจอร์อย่างให้ผู้ใช้ได้ทราบอีกรอบ เพื่อนำไปประกอบการพิจารณาวางแผนโครงการต่อไปได้

2.2) การหาค่าอัตราการทำงานสำเร็จของพนักงาน

พนักงานแต่ละคนที่อยู่ภายใต้ระบบ จะมีการเก็บค่าอัตราการทำงานสำเร็จตามเป้าหมาย เพื่อนำค่านี้ไปใช้คำนวณเวลาที่มีการสร้างฟีเจอร์ขึ้นมาใหม่ในระบบ โดยจะช่วยให้การคาดการณ์ระยะเวลาของโปรเจคนั้นมีความแม่นยำมากยิ่งขึ้น

ค่าอัตราการทำงานสำเร็จของพนักงานแต่ละคนนั้น จะถูกตั้งค่าเริ่มต้นให้ที่ 1.00 โดยจะมีการปรับค่าทุกรอบ เมื่อพนักงานแต่ละคนทำงานในฟีเจอร์ของตนเองเสร็จสิ้น เมื่อคำนวณแล้วจะได้ผลลัพธ์เป็นค่าอัตราการทำงานสำเร็จที่ถูกปรับเปลี่ยนไปตามการทำงานของฟีเจอร์ที่ผ่านมา

3) ขอบเขตของข้อมูล

ข้อมูลบันทึกการนับเวลาของผู้ใช้งานในแต่ละชิ้นงาน (Task) จะถูกจำลองขึ้นโดยอ้างอิงหลักการทำงานมาจาก บริษัท Banana Coding จำกัด

1.5 แผนการดำเนินงานและระยะเวลาดำเนินงาน

การศึกษานี้เริ่มดำเนินงานตั้งแต่เดือน สิงหาคม พ.ศ. 2561 สิ้นสุดเดือน เมษายน พ.ศ. 2562 แสดงรายละเอียดการดำเนินงาน ซึ่งมีขั้นตอนดังตารางที่ 1.1

ตารางที่ 1.1 ขั้นตอนการดำเนินงานและระยะเวลาการดำเนินงาน

ขั้นตอนการดำเนินการ	ระยะเวลาดำเนินการ	พ.ศ.2561					พ.ศ.2562			
		ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.
1. วิเคราะห์ขอบเขตระบบงาน										
2. ศึกษาทฤษฎีและเอกสารที่เกี่ยวข้อง										
3. ออกแบบระบบ										
4. พัฒนาระบบ										
5. ทดสอบและแก้ไขระบบงาน										
6. สรุปผลและจัดทำเอกสาร และคู่มือการใช้งาน										

บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

ระบบจัดการข้อมูลและการดำเนินงานของโครงการ ได้ทำการศึกษาข้อมูลจากหลักการและทฤษฎีในเรื่องต่อไปนี้

- 2.1 การประมาณเวลาการทำงาน
- 2.2 การวิเคราะห์การลดด้อยเชิงเส้นอย่างง่าย
- 2.3 แนวคิดการทำงานแบบ Scrum
- 2.4 Scrum Burndown Chart
- 2.5 เทคโนโลยีที่เกี่ยวข้องกับระบบจัดการข้อมูลและการดำเนินงานของโครงการ

2.1 การประมาณเวลาการทำงาน

การวางแผนโครงการ (Project Planning) [1,2] เป็นแนวทางที่จำเป็นเพื่อให้ผู้ที่เกี่ยวข้องในระหว่างการพัฒนาระบบงานได้รับทราบว่าจะต้องทำอะไรบ้าง เป็นงานหลักที่ต้องกระทำเมื่อเริ่มงานโดยปัจจัยสำคัญของการวางแผนโครงการ ซึ่งการวางแผนโครงการต้องมีการประมาณการในปัจจัย 3 ด้าน คือ ขอบข่ายงาน (Scope), ทรัพยากร (Resource) และ เวลา (Time) ปัจจัยทั้ง 3 ประการนี้มีความเกี่ยวพันธ์กัน เมื่อจะเพิ่มปัจจัยฝั่งใดฝั่งหนึ่งจะมีความสัมพันธ์ที่ทำให้ต้องเพิ่มปัจจัยในฝั่งอื่น ๆ ด้วย เช่น จะเพิ่มขอบข่ายของงาน ก็ต้องเพิ่มระยะเวลาในการทำงานและต้องเพิ่มทรัพยากรที่ใช้ด้วย

การประมาณเวลาการทำงาน เป็นส่วนหนึ่งในการประมาณการปัจจัยที่เกี่ยวข้องกับการวางแผนโครงการ ซึ่งมีความสำคัญเป็นอย่างยิ่งต่อการพัฒนาระบบ ซึ่งการประมาณเวลาการทำงานนั้นจะสอดคล้องกับการประมาณการปัจจัยด้านอื่น ๆ ด้วย โดยวิธีการประมาณการเวลาการทำงาน และปัจจัยอื่น ๆ นั้นมีหลากหลายวิธี ตัวอย่างเช่น

- 1) การใช้สูตรสมการคิดโดยมีค่าถ่วงน้ำหนัก เช่น

1.1) วิธีการเดลฟี (Delphi method)

เป็นวิธีการทางสถิติ ที่จะใช้กับงานที่ไม่เคยมีตัวเลขหรือข้อมูลมาก่อน สามารถใช้ในการประมาณการในเรื่องต่าง ๆ ของโครงการ เช่น ระยะเวลา ทรัพยากร ความเสี่ยง ต้นทุน ผลประโยชน์ เป็นต้น โดยจะใช้การประมาณการจากหมู่คณะที่มีความรู้ ความเข้าใจ ประสบการณ์ในสาขาที่แตกต่างกัน เข้ามาร่วมกันในการทำงานโครงการเดียวกัน โดยจะต้องมีสมาชิกอย่างน้อย 10 คน ร่วมกันให้ความคิดเห็นเกี่ยวกับเรื่องที่จะพิจารณา

1.2) วิธีการพีเอ็มไอ (PMI)

เป็นวิธีการประเมินแบบ 3 ประسان คือพิจารณาจากผู้ที่เกี่ยวข้องกับโครงการที่มีความคิดเห็นแยกเป็น 3 กลุ่ม เพื่อให้การประมาณการเวลาไม่คาดเดาได้มากยิ่งขึ้น ประกอบด้วย ผู้ที่คิดว่าไม่มีปัญหา ผู้ที่คิดว่าสามารถดำเนินการได้ และผู้ที่คิดว่าอาจจะมีปัญหา ซึ่งคนทั้ง 3 กลุ่มนี้นั้น ล้วนเป็นผู้มีประสบการณ์ในการทำงานในระบบงานนั้น

1.3) วิธีการจัดกลุ่มข้อมูล

มีการหาค่าระหว่างช่วง เพื่อนำมากำหนดขอบเขตของกลุ่ม ซึ่งแบ่งกลุ่มข้อมูลออกเป็น 3 กลุ่ม คือ P (มีปัญหา), M (ปานกลาง) และ O (ดีมาก) จากนั้นนำค่ามาหาค่าเฉลี่ยตามกลุ่มข้อมูล เพื่อประมาณการเวลาต่อไป

1.4) วิธีการใช้ค่าเบี่ยงเบนมาตรฐานของกลุ่มข้อมูล

1.5) วิธีการอื่น ๆ ที่คิดค้น และเหมาะสมกับโครงการขององค์กรนั้น ๆ

2) การแบ่งโครงการให้มีขนาดที่สามารถจัดการได้ เป็นการแยกระบบออกเป็นส่วนย่อยตามขนาดที่เหมาะสม ที่จะทำให้โครงการประสบความสำเร็จได้มากยิ่งขึ้น โดยวิธีแบ่งงานออกเป็น ส่วนย่อยมีหลายวิธี ดังนี้

2.1) ใช้กฎ 80 ชั่วโมง คือ แบ่งงานที่ทำออกให้เสร็จภายใน 80 ชั่วโมง หรือ 10 วัน หรือ 2 สัปดาห์

2.2) การประมาณการแบบ Timebox จะต้อง

- จัดลำดับความยากง่ายและความเสี่ยงของแต่ละงาน
- ในแต่ละงานก็จัดลำดับของแต่ละส่วนเข่นกัน
- สร้างตัวอย่างที่มีสภาพการทำงานเหมือนจริงจากลำดับที่จัดไว้
- จัดแต่ละส่วนงานเข้าไปใน Timebox ซึ่งใช้เวลา 10-20 วัน
- Timebox ประกอบด้วยขอบข่ายงานซึ่งสามารถควบคุมได้ ไม่ให้มีการเพิ่มเติม ปริมาณงานและสามารถจัดการกับความเสี่ยงได้

นอกจากนี้ การประมาณการปัจจัยที่เกี่ยวข้องกับโครงการนั้น ยังต้องคำนึงถึงอิทธิพลที่มีต่อ การประมาณการ ซึ่งสามารถแยกการพิจารณาได้เป็น 3 เรื่อง คือ

- 1) พิจารณาจากผลลัพธ์ที่ดำเนินการได้ เช่น พิจารณาจากขนาดของงาน ความซับซ้อน ของงาน ข้อจำกัดของงาน การจัดทำเอกสาร การตอบรับของระบบงาน เป็นต้น
- 2) พิจารณาระหว่างการพัฒนาระบบงาน เช่น การเปลี่ยนแปลงหน้าที่และงานที่ทำของ คนในโครงการ การเริ่มต้นโครงการและฝึกอบรม ช่วงเวลาการเรียนรู้ สิ่งที่จำเป็นในโครงการ การทดสอบ การเปลี่ยนแปลงและแก้ไขระบบงาน เป็นต้น

3) พิจารณาเรื่องสิ่งแวดล้อม เช่น เวลาแห่งปี วันหยุดพักผ่อน เวลาที่ใช้ในการทำแต่ละวัน เวลาในการตัดสินใจ การออกแบบ การใช้จัดลำดับก่อนหลังของความต้องการ การควบคุมการเปลี่ยนแปลงในกระบวนการ เป็นต้น

2.2 การวิเคราะห์การถดถอยเชิงเส้นอย่างง่าย

การวิเคราะห์การถดถอยเชิงเส้นอย่างง่าย (Simple Linear Regression Analysis) [3,4] เป็นการวิเคราะห์ความสัมพันธ์ระหว่างตัวแปร 2 ตัวคือ ตัวแปรอิสระ 1 ตัว และตัวแปรตาม 1 ตัว โดยใช้ข้อมูลในอดีตมาวิเคราะห์ เพื่อนำความสัมพันธ์ที่ได้จากการวิเคราะห์มาใช้ทำนาย หรือพยากรณ์เหตุการณ์ในอนาคตได้ ซึ่งทั้งสองตัวแปรมีความสัมพันธ์เชิงเส้นตรงซึ่งกันและกัน

$$Y = \beta_0 + \beta_1 X$$

โดยที่ Y คือ ตัวแปรตาม

X คือ ตัวแปรอิสระ

β_0 และ β_1 เป็นพารามิเตอร์ที่ไม่ทราบค่า โดยที่

β_0 คือ ค่าจากระยะตัดแกน Y (ค่าของ Y เมื่อ X มีค่าเป็นศูนย์)

β_1 คือ ค่าความชันของเส้นสมการถดถอย

สาเหตุที่ทำให้เกิดค่า e ซึ่งเป็นค่าความคลาดเคลื่อนแบบสุ่มนั้น เนื่องจากความสัมพันธ์ระหว่างสองตัวแปรอาจไม่เป็นไปตามสมการความสัมพันธ์เชิงเส้นเสมอไป เช่น อาจไม่ได้รวมตัวแปรบางชนิดไว้ในฟังก์ชัน หรือเกิดความคลาดเคลื่อนจากการรวมข้อมูล หรือเกิดความคลาดเคลื่อนจากการวัดค่าตัวแปร หรือการเกิดพฤติกรรมเชิงสุ่มของข้อมูล เป็นต้น

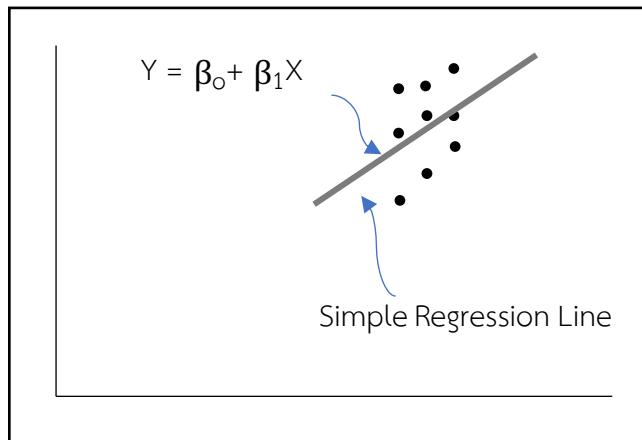
ค่าพารามิเตอร์ β_0 และ β_1 ซึ่งพารามิเตอร์ที่ไม่ทราบค่านั้น เรียกว่าสัมประสิทธิ์การถดถอย (Regression Coefficient) สามารถหาค่าของพารามิเตอร์ทั้งสองนี้ได้ด้วยวิธีต่าง ๆ เช่น วิธีการประมาณค่าแบบกำลังสองน้อยที่สุด วิธีการประมาณค่าที่ไม่มีความเออนเอียงเชิงเส้นดีที่สุด วิธีการประมาณค่าจากภาระน้ำจะเป็นสูงสุด เป็นต้น โดยในที่นี้ จะกล่าวถึงวิธีการประมาณค่าแบบกำลังสองน้อยที่สุดเท่านั้น

การประมาณค่าพารามิเตอร์ β_0 และ β_1 ด้วยวิธีการประมาณค่าแบบกำลังสองน้อยที่สุดนั้น เป็นวิธีการที่ได้รับความนิยมมากกว่าวิธีอื่น ๆ เนื่องจากสามารถทำความเข้าใจได้ง่ายกว่า และมีคุณสมบัติเป็นไปตามทฤษฎีของ Markov คือ (1) เป็นตัวประมาณค่าที่ไม่มีความเออนเอียง (Unbiased) (2) มีลักษณะเชิงเส้น (Linear) และ (3) มีความแปรปรวนน้อยที่สุดในบรรดาตัวประมาณค่าทั้งหมดที่มีคุณสมบัติเชิงเส้นและไม่เออนเอียง

หลักการของวิธีกำลังสองน้อยที่สุด คือ จะประมาณค่า β_0 และ β_1 ที่ทำให้เส้นทดแทนของตัวอย่าง (Simple Regression Line) ลากผ่านหรือใกล้เคียงกับค่าสังเกตที่รวบรวมมาได้มากที่สุดโดยหาค่าได้จากการสมการ

$$\beta_0 = \bar{y} - \beta_1(\bar{x})$$

$$\beta_1 = \frac{\sum xy - \frac{\sum x \sum y}{n}}{\sum x^2 - n(\bar{x})^2}$$



ภาพที่ 2.1 ตัวอย่าง Simple Regression Line

2.3 แนวคิดการทำงานแบบ Scrum

Scrum [5,6] เป็นหนึ่งในแนวคิดในการพัฒนาซอฟต์แวร์ที่เป็นที่รู้จัก และเป็นที่นิยมมากที่สุดในกลุ่มวิธีการพัฒนาซอฟต์แวร์แบบเจาจายล์ (Agile Software Development) โดยมุ่งเน้นไปที่การบริหารโครงการพัฒนาซอฟต์แวร์เป็นหลัก เพื่อเพิ่มอัตราความสำเร็จในการพัฒนาซอฟต์แวร์ให้มากขึ้น สามารถส่งมอบผลิตภัณฑ์ได้รวดเร็วขึ้น โดยมีรายละเอียดที่เกี่ยวข้องดังนี้

1) ทฤษฎีสครัม (Scrum Theory)

ทฤษฎีสครัม (Scrum Theory) นั้น เน้นการนำความรู้จากประสบการณ์เฉพาะที่เคลื่อนเมื่อทำจริง (Empiricism) มาใช้ในการพัฒนาการดำเนินงานในปัจจุบันให้ดียิ่งขึ้น ประกอบด้วย 3 ส่วนคือ

- ความโปร่งใส (Transparency) คือ ในการทำงานเป็นทีมมั่น สมาชิกในทีมจะต้องเห็นภาพชัดเจนและเข้าใจตรงกัน ให้เป็นมาตรฐานเดียวกัน ไม่มีความหมายต่างกัน เช่น นิยามของคำว่างานเสร็จ หมายถึง การผลิตเสร็จ หรือ ผลิตและทดสอบเสร็จ หรือ ได้รับการเชิญรับรอง หรือ ส่งมอบให้ผู้ใช้แล้ว ต้องนิยามและตกลงให้เข้าใจตรงกัน เป็นต้น

- การตรวจสอบ (Inspection) คือการนำผลลัพธ์การดำเนินกิจกรรมต่าง ๆ ของสครัม (Scrum Artifact) มาตรวจสอบและวัดผลว่าบรรลุตามเป้าหมายที่กำหนดไว้หรือไม่

- การปรับเปลี่ยน (Adoption) คือหากผลิตภัณฑ์ไม่เป็นไปตามที่กำหนด จะต้องปรับเปลี่ยนการดำเนินงาน หรือจำนวนทรัพยากรที่ใช้ เพื่อให้บรรลุผลตามที่กำหนดหรือใกล้เคียงได้มากที่สุด

2) Product Backlog

คืองานทั้งหมดที่ต้องทำเพื่อพัฒนาผลิตภัณฑ์ ส่วนใหญ่บัญมีขึ้นในรูปแบบของ User Story โดยมีรายละเอียดของงาน เกณฑ์การทดสอบงาน การประเมินความซับซ้อนและเวลาที่ต้องใช้ในการพัฒนาด้วย เมื่อพัฒนาเสร็จและส่งมอบแล้ว อาจนำผลตอบรับจากผู้ใช้งานทบทวนและปรับปรุงผลิตภัณฑ์เพิ่มเติม โดยเขียนเป็น User Story ใหม่

3) การประเมินเวลาที่ใช้ในการพัฒนา

เมื่อได้ Product Backlog แล้ว จะต้องมีการประเมินขนาดของงาน ว่างานแต่ละชิ้นใน Product Backlog นั้นจะมีขนาดเท่าใด ควรใช้เวลาเท่าใดในการดำเนินงาน ซึ่งในการประเมินแต่ละงานนั้น จะมีค่าที่เกี่ยวข้องอยู่ 2 ส่วน ได้แก่

3.1) ค่า Story Point คือ ขนาดของงาน บ่งบอกถึงความซับซ้อนในการพัฒนาของงาน โดยทั่วไปแล้วจะกำหนดค่าอยู่ในช่วง 1-10

3.2) ค่า Estimate Time คือ เวลาที่ต้องใช้ในการพัฒนางาน โดยในแต่ละงานจะมีเวลาที่ต้องใช้ในการพัฒนางานแตกต่างกันไปตามความซับซ้อนของงาน โดยทั่วไปจะนับเวลาเป็นหน่วยชั่วโมง

ค่าประเมินทั้งสองจะมีผลต่อการจัดเรียงความสำคัญของงานใน Product Backlog เพราะ Product Owner จะทราบว่างานแต่ละชิ้นต้องใช้ทรัพยากรในการพัฒนาเท่าไร คุ้มกับมูลค่าที่จะได้เมื่องานเสร็จหรือไม่

4) ตำแหน่งในทีม Scrum

ในแนวคิดการทำงานแบบ Scrum นั้นหมายสำหรับทีมขนาดเล็กที่พร้อมปรับตัว พัฒนาและเปลี่ยนแปลงอยู่ตลอดเวลา สมาชิกในทีมจะต้องสามารถบริหารและดำเนินงานกันเองได้ด้วยสมาชิกภายในทีม สามารถหาแก้ไขปัญหาได้เอง โดยไม่ต้องรอความช่วยเหลือจากนอกทีม โดยประกอบด้วย

- Product Owner

คือผู้ที่รับความต้องการ (Requirement) ของลูกค้า และนำมาเปลี่ยนเป็น User Story และต้องทำให้ทุกฝ่ายของทีมพัฒนาเห็นภาพของ User Story ตรงกัน และมีหน้าที่ปฏิเสธ Requirement บางอย่างที่อาจทำให้การพัฒนามี Backlog ค้างมากเกินไป

นอกจากนี้ Product Owner ยังต้องบริหารจัดการ Product Backlog ให้ชัดเจน ต้องเป็นผู้วางแผนการพัฒนาระบบทั้งในระยะสั้นและระยะยาว โดยที่ต้องรักษาสมดุลระหว่างการพัฒนาฟีเจอร์ใหม่ การแก้ไขปัญหาเดิม และการอัพเกรดรูปแบบเดิมให้ดีขึ้น เพื่อลดความเสี่ยงทางธุรกิจ เทคโนโลยี ต้นทุนในการพัฒนา และเวลาในการพัฒนา

- Developer

มีสมาชิกจำนวน 3–9 คน ไม่มีการแบ่งทีมย่อยภายในตัวทีมพัฒนาอีก เช่น ทีมออกแบบ ทีมทดสอบ เพื่อให้สามารถรับปริมาณงานได้ไม่น้อยเกินไป และไม่เสียเวลาในการประสานงานมากเกินไป โดยมีหน้าที่พัฒนาตามที่วางแผนเอาไว้ โดยสมาชิกในทีมแต่ละคนจะมีความสามารถเฉพาะทางของตนเอง เช่น สมาชิกอาจความสามารถในการออกแบบ แต่ความรับผิดชอบงานจะเป็นของทีมทั้งทีม หากงานออกแบบไม่เสร็จทั้งทีมต้องมาช่วยกันรับผิดชอบ

5) Scrum Events

การทำกิจกรรม Scrum เพื่อให้การดำเนินงานเป็นขั้นตอนชัดเจน ตรวจสอบ วัดผลได้ และลดการประชุมที่ไม่จำเป็น การทำ Scrum จะประกอบไปด้วยกิจกรรมต่าง ๆ ดังนี้

5.1) วางแผนสปริงท์ (Sprint Planning)

Product Owner จะเป็นผู้กำหนดเป้าหมายของ Sprint ว่าจะทำอะไรบ้าง เพราะอะไร และเพื่ออะไร โดยยึดตามงานที่มีใน Product Backlog และให้ Developer เลือกงานต่าง ๆ ใน Sprint นี้ไปเข้าสู่ Sprint Backlog เพื่อนำไปพัฒนางานต่อไป

5.2) การประชุมประจำวัน (Daily Meeting)

หรืออาจจะเรียกว่า Standup Meeting เป็นการล้อมวงประชุมกันในเวลาสั้น ๆ ใช้เวลาไม่เกิน 15 นาที โดยให้ Developer มาแจ้งความคืบหน้าในการพัฒนางานแก่ทีม โดยให้แต่ละคนแจ้งให้ทีมทราบว่า ทำอะไรไปในเมื่อวาน วันนี้จะทำอะไรเพิ่มเติม ปัญหาที่เกิดขึ้นในการพัฒนานี้อะไรบ้าง โดยจุดประสงค์ในการประชุมนี้ก็เพื่อ

- ตรวจสอบและแจ้งความคืบหน้าของงานใน Sprint
- เพื่อเป็นการวางแผนการทำงานในแต่ละวัน
- แจ้งให้ทีมทราบหากมีงานอื่นแทรกเข้ามา
- หากพบปัญหาที่ส่งผลกระทบต่อการทำงานให้บรรลุเป้าหมายของ Sprint จะได้มีการประชุม

5.3) ตรวจสอบผลลัพธ์ของสปริงท์ (Sprint Review)

ตรวจสอบงานที่พัฒนาและแสดงผลลัพธ์ของงานใน Sprint ให้แก่ผู้เกี่ยวข้องกับระบบ (Stakeholders) เพื่อรับฟังข้อเสนอแนะ และทบทวน Product Backlog ที่จะทำต่อไปให้สอดคล้องกับโอกาสและสถานการณ์ของตลาดในปัจจุบัน

5.4) ตรวจสอบการดำเนินงานของสปริงท์ (Sprint Retrospective)

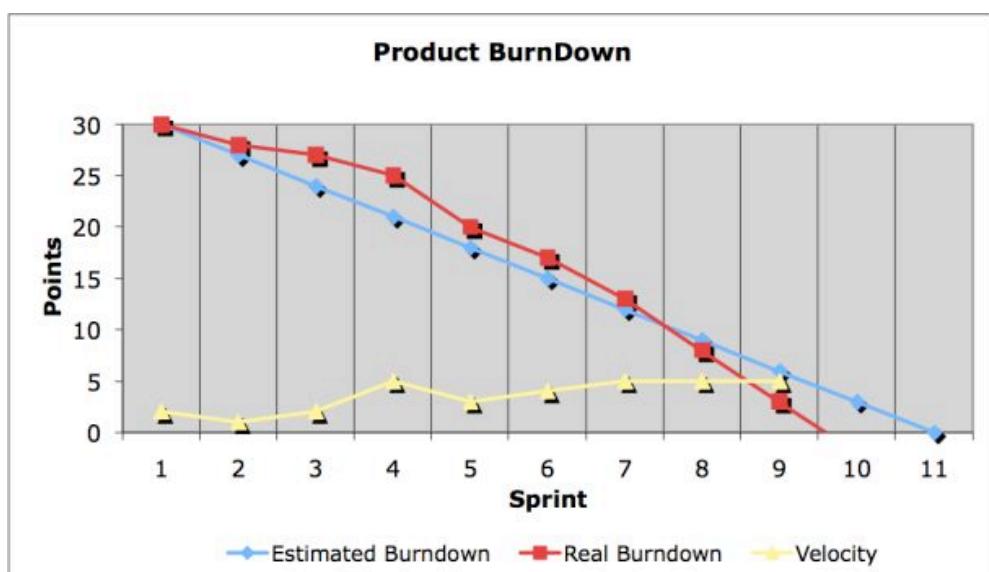
เมื่อเสร็จสิ้น 1 Sprint จะต้องมีการตรวจสอบการดำเนินงานภายใน Sprint ที่จบลงโดยต้องมีการวิเคราะห์ถึงความรู้ เครื่องมือที่ใช้ ความสัมพันธ์ภายในทีม สภาพแวดล้อมในการทำงาน รวมไปถึงการประเมินผลการทำงาน เพื่อให้ทราบว่าการทำงานใน Sprint ที่เสร็จสิ้นไปนั้นเรียบร้อยดี

หรือไม่ โดยในบางครั้งอาจใช้ Burndown Chart ในการประเมินว่าการทำงานในแต่ละ Sprint นั้นมีความรวดเร็วหรือล่าช้ามากน้อยเพียงใด เพื่อปรับปรุงและแก้ไขให้การทำงานดีขึ้นต่อไป

2.4 Scrum Burndown Chart

Scrum Burndown Chart [7,8,9] เป็นกราฟที่ใช้สำหรับการประเมินผลการทำงาน และใช้ในการคาดการณ์ระยะเวลาของการทำงานของโครงการ โดยในกราฟจะแสดงข้อมูลต่าง ๆ ดังนี้

- เส้นสีแดง : จำนวนของ Story Point ที่เหลือของโครงการในแต่ละ Sprint
- เส้นสีเหลือง : ค่าอัตราการทำงาน (Velocity) ของทีม โดยใน 1 Sprint จะมีการทำงานได้ทั้งหมดกี่ Story Point
- เส้นสีฟ้า : ค่าคาดการณ์จำนวน Sprint ที่จะใช้ทั้งหมดเพื่อให้ Story Point = 0 (งานเสร็จสิ้นทั้งหมด)



ภาพที่ 2.2 ตัวอย่างกราฟ Product Burndown

ที่มา: https://www.scrum-institute.org/Burndown_Chart.php

2.5 เทคโนโลยีที่เกี่ยวข้องกับระบบ

1) HTML5

HTML5 [10] คือ ภาษาที่ใช้ในการเขียนเว็บ โดยใช้ Tag ในการกำหนดการแสดงผล ซึ่ง HTML5 นั้นก้าวพัฒนาขึ้นมาจาก HTML ยุคก่อนหน้านี้ เพื่อนำมาใช้ในการสร้างเว็บให้เข้ากับการใช้งานในยุคปัจจุบันมากขึ้น HTML5 ยังช่วยให้ผู้พัฒนาสามารถทำได้ง่ายตามฟีเจอร์ที่เพิ่มขึ้น โดยข้อดีที่ถูกเพิ่มขึ้นมาจาก HTML ยุคก่อนหน้านี้คือ

- รองรับการใช้งานทางด้านมัลติมีเดียมากขึ้น เช่น Video, Audio ต่าง ๆ ที่จากแต่ก่อนต้องใช้ส่วนเสริม (Plug-in) ในการใช้งาน แต่ปัจจุบันก็ใช้งานผ่าน HTML5 ที่รองรับได้เลย

- เพิ่มเติมฟีเจอร์ที่หลากหลาย เช่น ฟีเจอร์ Local Storage ที่เก็บข้อมูลไว้ในเครื่องแบบอффไลน์, Canvas ที่ทำให้สามารถวาดภาพลงบนเว็บเพจได้โดยตรง, การรับข้อมูลเข้ารูปแบบใหม่ เช่น Datepicker, Email, Url, Number เป็นต้น

- Tag ใหม่ ๆ ใน HTML5 มีการสื่อความหมายในการใช้งานมากขึ้น เช่น section, article, aside เป็นต้น

2) Syntactically Awesome Style Sheets

Syntactically Awesome Stylesheets (SASS) หรือ SCSS [11] เป็นเครื่องมือรูปแบบหนึ่งที่เป็น CSS Preprocessor คือเมื่อเขียน SCSS เสร็จแล้วเมื่อนำไปใช้งานจะถูกแปลงกลับไปเป็น CSS ก่อนถึงจะนำไปใช้งานได้ โดย SCSS มีความแตกต่างจาก CSS ปกติคือ

- สามารถสร้างตัวแปรเพื่อใช้งานในส่วนต่าง ๆ ได้
- สามารถเขียน Selector แบบชั้น ๆ กัน (Nesting) ได้
- สามารถสร้าง Mixin เก็บไว้ใช้งานในส่วนอื่น ๆ ได้
- สามารถใช้สัญลักษณ์ทางคณิตศาสตร์ในการคำนวณค่าต่าง ๆ ได้

SCSS นั้น ใช้สำหรับตกแต่ง หรือปรับปรุงลักษณะโครงสร้างต่าง ๆ ของสิ่งที่เขียนจากภาษา HTML เพื่อให้เกิดความสวยงาม หรือตรงตามความต้องการมากขึ้น

3) Ruby

Ruby [12] คือภาษาเขียนโปรแกรมแบบด้านมิกส์ที่สามารถปรับเปลี่ยนโครงสร้างของมันเองได้ในขณะที่โปรแกรมทำงาน (Run time) นอกจากนี้ ยังเป็นภาษาสำหรับเขียนโปรแกรมเชิงวัตถุ (Object-oriented) มาตั้งแต่กำเนิด และเป็นภาษาสำหรับการเขียนโปรแกรมเพื่อวัตถุประสงค์ที่ว่าไป ถูกออกแบบและพัฒนาขึ้นในช่วงกลางของช่วงปี 1990 โดย Yukihiro Matsumoto ในประเทศญี่ปุ่น

ภาษา Ruby นั้นเป็นภาษาสำหรับการเขียนโปรแกรมเพื่อวัตถุประสงค์ที่ร่วม ดังนั้นมันสามารถใช้เขียนโปรแกรมได้หลากหลายรูปแบบ ยกตัวอย่างเช่น การเขียนแอพพลิเคชันแบบ Console การเขียนแอพพลิเคชันแบบ GUI หรือแม้กระทั่งการนำไปพัฒนาเว็บไซต์พร้อมกับเฟรมเวิร์คอย่าง Ruby On Rails โดยการใช้ร่วมกับเว็บเซิฟเวอร์อย่าง Mongrel ที่เขียนขึ้นโดยภาษา Ruby เอง

4) Ruby on Rails Framework

Ruby on Rails Framework [13] เป็น Open Source Web Application Framework ที่ถูกพัฒนาด้วยภาษา Ruby โดยที่ Rails นั้นจะมีโครงสร้างการจัดการแบบ Model View Controller ซึ่งมีข้อดีคือ

- มีการออกแบบที่ชัดเจน มองภาพรวมได้ง่าย
- ปรับปรุง/แก้ไข ระบบได้โดยง่าย เพราะแต่ละส่วนแยกกันอย่างชัดเจน

- Maintain ระบบได้ง่าย เพราะมีโครงสร้างชัดเจน จึงทำให้ยืดหยุ่นต่อการเพิ่มเติมรวมไปถึงการนำส่วนต่าง ๆ ไปใช้ซ้ำได้ด้วย

อีกทั้ง Rails ยังถูกพัฒนาขึ้นมาเพื่อให้แอพพลิเคชันถูกพัฒนาด้วยโค้ดที่น้อยกว่า Framework อื่นอย่างมาก และมี Configuration ที่น้อยที่สุด เพื่อให้ง่ายต่อการนำไปประยุกต์ใช้

5) JavaScript

JavaScript [14] คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ต JavaScript เป็น ภาษาสคริปต์เชิงวัตถุ ใช่วร์มกับภาษา HTML เพื่อให้เว็บไซต์ของเรามีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "เปลี่ยนและดำเนินงานไปทีละคำสั่ง" (Interpret) หรือเรียกว่า อ็อบเจกโอลีตเต็ด (Object Oriented Programming) ที่มีเป้าหมายในการ ออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต โดยข้อดีของการใช้ JavaScript คือ

- ทำให้สามารถใช้เขียนโปรแกรมแบบง่าย ๆ ได้ โดยไม่ต้องพึ่งภาษาอื่น
- มีคำสั่งที่ตอบสนองกับผู้ใช้งาน เช่น เมื่อผู้ใช้คลิกที่ปุ่ม หรือ Checkbox ก็สามารถสั่งให้เปิดหน้าใหม่ได้ ทำให้เว็บไซต์ของเรา มีปฏิสัมพันธ์กับผู้ใช้งานมากขึ้น
- สามารถเขียนหรือเปลี่ยนแปลง HTML Element ได้ สามารถเปลี่ยนแปลงรูปแบบการแสดงผลของเว็บไซต์ได้
- สามารถใช้ตรวจสอบข้อมูลได้ เช่น ตรวจสอบช่อง Email เมื่อเรากรอกข้อมูลผิดจะมีหน้าต่างฟ้องขึ้นมาว่าเรากรอกผิด หรือลืมกรอกอะไรบางอย่าง เป็นต้น
- JavaScript สร้าง Cookies ได้

6) Sublime Text 3

Sublime Text 3 [15] คือซอฟต์แวร์สำหรับใช้เขียนและพัฒนาโปรแกรม (Text Editor) ซึ่งรองรับการเขียนได้หลากหลายภาษา สามารถปรับแต่งเพิ่มเติมตัวเสริม (Plug-in) หรือลงแพคเกจ (Package) สำหรับการใช้งานในการเขียนภาษาต่าง ๆ เพื่อเพิ่มความสามารถให้กับการเขียนโปรแกรม ได้ สามารถใช้งานคีย์ลัดต่าง ๆ (Hotkeys) ได้

บทที่ 3

การวิเคราะห์และออกแบบระบบ

การวิเคราะห์และออกแบบระบบ เป็นส่วนสำคัญในการพัฒนาระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ เพื่อแสดงให้เห็นถึงขั้นตอนการทำงานต่าง ๆ ของระบบ ที่เกิดขึ้น เพื่อเป็นแนวทางในการพัฒนาระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ ให้ได้ตามความต้องการ โดยในที่นี้จะกล่าวถึงการออกแบบใน 5 รูปแบบคือ

- 3.1 การคาดการณ์ระยะเวลาของโปรเจค
- 3.2 อัตราการทำงานสำเร็จของพนักงาน
- 3.3 แผนภาพยุสเคส
- 3.4 แผนภาพกิจกรรม
- 3.5 แผนภาพซีเควนซ์โดยแกรม

3.1 การคาดการณ์ระยะเวลาของโปรเจค

ในแต่ละโปรเจค จะมีการแบ่งงานออกเป็นงานย่อย ๆ หรือเรียกว่าฟีเจอร์ (Feature) โดยแต่ละฟีเจอร์ย่อย ๆ นั้น จะต้องมีการคาดการณ์ระยะเวลาของฟีเจอร์ เพื่อหาผลรวมเวลาที่ใช้งานในแต่ละฟีเจอร์ออกมาเป็นระยะเวลาที่ใช้ของโปรเจค

การคาดการณ์ระยะเวลาการทำงาน (Estimate Time) ของระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ จะทำงานเมื่อผู้ใช้สร้างโปรเจคใหม่และสร้างฟีเจอร์ใหม่ขึ้นมาในระบบ และทำการเลือกสมาชิกในทีมโปรเจคและทีมฟีเจอร์ โดยการคาดการณ์จะใช้ข้อมูลการนับเวลาของฟีเจอร์ที่เสร็จสิ้นแล้ว มาคำนวณกับขนาดของงาน (Story Point) ที่ผู้ใช้กรอกข้อมูลเข้ามาในแต่ละฟีเจอร์ เพื่อให้ได้ผลลัพธ์เป็นเวลาที่จะใช้ในการทำงานของแต่ละฟีเจอร์ภายในโปรเจคนั้น

การคาดการณ์ระยะเวลาโดยใช้ทฤษฎีการวิเคราะห์การลดถอยเชิงเส้นอย่างง่าย (Simple Linear Regression Analysis) นั้น มีหลักการทั่วไปคือ จะต้องสร้างสมการการลดถอยเชิงเส้นของบริษัทที่ผู้ใช้งานสังกัดอยู่ขึ้นมา โดยนำข้อมูลการทำงานของแต่ละฟีเจอร์ที่เสร็จสิ้นแล้วมาคำนวณจากนั้นนำมาสมการนี้ไปใช้กับฟีเจอร์ที่ถูกสร้างขึ้นมาใหม่ เพื่อหาระยะเวลาที่ใช้ในแต่ละฟีเจอร์ออกมา

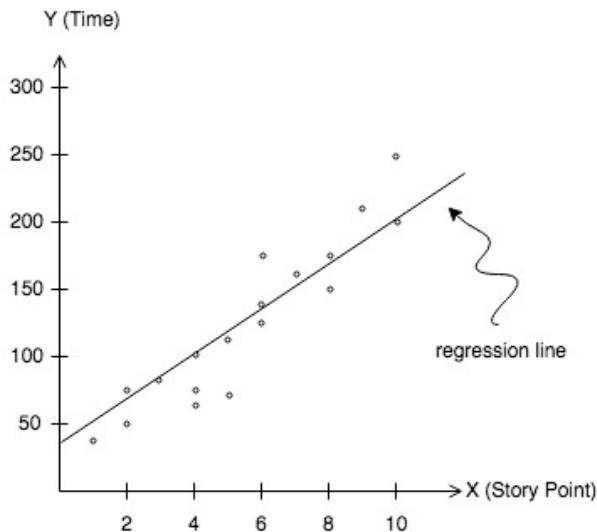
$$Estimate_time_{feature\ i} = \beta_0 + (\beta_1 * Story_point)$$

โดยที่ $Estimate_time_{feature\ i}$ คือ ระยะเวลาที่คาดการณ์ได้ของฟีเจอร์ที่สนใจ

$Story_point$ คือ ขนาดของงานในฟีเจอร์ที่สนใจ

β_0 คือ ค่าจากระยะตัดแกน Y (ค่าของ Y เมื่อ X มีค่าเป็นศูนย์)

β_1 คือ ค่าความชันของเส้นสมการ Regression



ภาพที่ 3.1 ตัวอย่างกราฟการถดถอยของบริษัท 1 บริษัท

ตัวอย่างการคำนวณ

กำหนดให้ข้อมูลการทำงานของแต่ละพีเจอร์ของบริษัทแห่งหนึ่ง มีค่าดังตารางที่ 3.1

ตารางที่ 3.1 ตัวอย่างข้อมูลการทำงานของแต่ละพีเจอร์ในบริษัทแห่งหนึ่ง

Project Name	Feature Name	Story Point	Time (Hours)	Total (Hours)
Project 3	Feature 3.1	1	8.00	550.34
	Feature 3.2	4	55.00	
	Feature 3.3	8	487.34	
Project 5	Feature 5.1	4	40.00	906.33
	Feature 5.2	5	75.00	
	Feature 5.3	8	791.33	
Project 1	Feature 1.1	5	97.5	1044.79
	Feature 1.2	3	26.17	
	Feature 1.3	4	61.67	
	Feature 1.4	8	640	
	Feature 1.5	7	219.45	
Project 8	Feature 8.1	4	90	267.33
	Feature 8.2	3	60	
	Feature 8.3	6	117.33	

$$Estimate_time_{feature\ i} = \beta_0 + (\beta_1 * Story_point)$$

เมื่อต้องการหาสมการ Regression ของบริษัทแห่งนี้ จะต้องหาค่า β_0 และ β_1 เพื่อนำไปประกอบเป็นสมการที่ต้องการ โดยมีวิธีการหาคือ

1) หาค่า β_1 จากสมการ

$$\beta_1 = \frac{\sum xy - \frac{\sum x \sum y}{n}}{\sum x^2 - n(\bar{x})^2}$$

โดยกำหนดให้ ค่า X ของสมการ คือ ค่าขนาดของงาน (Story Point) และค่า Y ของสมการ คือ ค่าเวลา (Time) ของแต่ละฟีเจอร์ในบริษัท ซึ่งสามารถหาค่าต่าง ๆ จากข้อมูลของบริษัท ในตารางที่ 3.1 ได้ค่าดังนี้

$$\sum x = 70 \mid \sum y = 2768.79 \mid \sum xy = 19705.18 \mid \sum x^2 = 410 \mid n = 14 \mid \bar{x} = 5$$

เมื่อแทนค่าลงในสมการ จะได้

$$\beta_1 = \frac{19705.18 - \frac{70*2768.79}{14}}{410 - (14*5^2)}$$

$$\beta_1 = \frac{19705.18 - 13843.95}{410 - 350}$$

$$\beta_1 = 97.69$$

2) หาค่า β_0 จากสมการ

$$\beta_0 = \bar{y} - \beta_1(\bar{x})$$

โดยมีค่าต่าง ๆ คือ

$$\bar{y} = 197.77 \mid \beta_1 = 97.69 \mid \bar{x} = 5$$

เมื่อแทนค่าลงในสมการ จะได้

$$\beta_0 = 197.77 - (97.69 * 5)$$

$$\beta_0 = -290.68$$

ดังนั้น จึงได้สมการ Regression ของบริษัทแห่งนี้คือ

$$Estimate_time_{feature\ i} = -290.68 + (97.69 * Story_point)$$

3.2 ประสิทธิภาพการทำงานของพนักงาน

เนื่องจากสมาชิกแต่ละคนมีความสามารถที่แตกต่างกันไปในการทำงาน เมื่อเลือกสมาชิกในทีมฟีเจอร์ต่าง ๆ ย่อมส่งผลให้เวลาการทำงานภาพรวมของฟีเจอร์นั้นมีการเปลี่ยนแปลง ดังนั้นจึงต้องมีวิธีในการเก็บข้อมูลสำหรับสมาชิกแต่ละคน เพื่อให้สามารถนำไปคำนวณกับการคาดการณ์การนับเวลาได้ว่า หากเลือกสมาชิกแต่ละคนแล้วจะใช้เวลาเท่าใด

ในระบบจัดการข้อมูลและการนับระยะเวลาดำเนินงานของโครงการนั้น จึงได้มีการเก็บค่า Performance ของสมาชิกแต่ละคนไว้ ซึ่งอ้างอิงมาจากค่า Burndown ใน Burndown Chart [7,8,9] ของการทำงานในแนวคิดแบบ Scrum โดย Performance เอาไว้ใช้สำหรับเก็บประสิทธิภาพการทำงานของสมาชิกในทีม โดยมีค่าเริ่มต้นของ Performance ที่ 1.00 จากนั้นค่าจะมีการเปลี่ยนแปลงในทุก ๆ ครั้งที่สมาชิกทำงานเสร็จสิ้น 1 ฟีเจอร์

1) การนำค่า Performance ไปใช้งาน

การนำค่า Performance ไปใช้นั้น จะใช่ว่าร่วมกับผลลัพธ์จากการคำนวณของ Simple Linear Regression Analysis เพื่อให้ได้ผลลัพธ์ของการนับเวลา ที่มีสมาชิกในทีมที่ถูกเลือกเข้ามาเกี่ยวข้อง

$$Time_{feature\ i} = \frac{\sum_{user \in member_{feature\ i}} (Performance_{user} * Estimate_time_{feature\ i})}{|member_{feature\ i}|}$$

โดยที่ $Time_{feature_i}$ คือ ระยะเวลาที่คาดว่าจะใช้ของฟีเจอร์หลังจากเลือกสมาชิก (ข้อมูล)

$user \in member_{feature_i}$ คือ สมาชิกของระบบ ที่เป็นสมาชิกในทีมของฟีเจอร์

$Estimate_time_{feature_i}$ คือ เวลาที่คาดการณ์มาจากการ Regression ของฟีเจอร์ (ชั่วโมง)

ระยะเวลาที่คาดว่าจะใช้ของฟีเจอร์หลังจากเลือกสมาชิก จะมีค่ามากขึ้น หรือน้อยลงจากเวลาที่คาดการณ์มาจากการ Simple Linear Regression Analysis นั้น ก็ขึ้นอยู่กับ Performance ของสมาชิกแต่ละคนว่า มีประสิทธิภาพในการใช้เวลาการทำงานมากน้อยเพียงใด ยิ่ง Performance มีค่ามาก นั่นหมายถึงจะยิ่งใช้เวลามากขึ้นกว่าที่กำหนดไว้ แต่ถ้า Performance มีค่าน้อย นั่นหมายถึงสมาชิกนี้จะสามารถงานได้เสร็จก่อนเวลาที่กำหนดไว้ได้

ตัวอย่างการคำนวณ

กำหนดให้ข้อมูลประสิทธิภาพการทำงานของพนักงานในบริษัทแห่งหนึ่ง ที่เป็นสมาชิกของ ฟีเจอร์ A นีค่าดังตารางที่ 3.2

ตารางที่ 3.2 ตัวอย่างข้อมูลประสิทธิภาพการทำงานของพนักงานในบริษัทแห่งหนึ่ง

พนักงาน	ประสิทธิภาพ (Performance)
User A	0.9706
User B	0.6789
User C	1.0714

กำหนดให้ Story Point ของฟีเจอร์ A คือ 7 และสมการ Regression ของบริษัทแห่งนี้คือ

$$Estimate_time_{feature\ i} = -290.68 + (97.69 * Story_point)$$

จะได้

$$Estimate_time_{feature\ i} = -290.68 + (97.69 * 7)$$

$$Estimate_time_{feature\ i} = 393.15$$

เมื่อได้ค่า Estimate Time ของฟีเจอร์ A แล้ว จะต้องมีการนำค่ามาปรับตาม User Performance ของสมาชิกในทีม จากสูตร

$$Time_{feature\ i} = \frac{\sum_{user \in member_{feature\ i}} (Performance_{user} * Estimate_time_{feature\ i})}{|member_{feature\ i}|}$$

โดยที่ $member_{feature\ i} = \{User\ A, User\ B, User\ C\}$

จะได้

$$Time_{feature\ i} = \frac{(393.15 * 0.9706) + (393.15 * 0.6789) + (393.15 * 1.0714)}{|3|}$$

$$Time_{feature\ i} = 356.3739$$

ดังนั้น คาดการณ์ระยะเวลาของฟีเจอร์ A เมื่อเลือกสมาชิกในทีมแล้วคือ 356.3739

2) การปรับค่า Performance ของสมาชิก

เมื่อฟีเจอร์มีการสิ้นสุดลง นี่คือสมาชิกแต่ละคนทำงานในฟีเจอร์นั้นเสร็จสิ้นแล้ว ในช่วงนี้ จะมีการคำนวณค่า Performance ของสมาชิกแต่ละคนในทีมฟีเจอร์ที่เสร็จสิ้นใหม่ เพื่อนำผลลัพธ์ ของการทำงานในฟีเจอร์ที่เสร็จสิ้นไปใช้ประโยชน์ให้มากขึ้น

ขั้นตอนการปรับค่า Performance ของสมาชิกในทีมฟีเจอร์ มีดังนี้

2.1) หากค่าความแตกต่างระหว่างระยะเวลาที่คาดการณ์ของฟีเจอร์ (Estimate Time) กับระยะเวลาที่ใช้จริงของทีม (Used Time) โดยคำนวณออกมาเป็นค่าร้อยละ ผลลัพธ์ที่ได้จะมีค่า + หรือ - ขึ้นอยู่กับว่าเวลาที่ใช้จริงของทีม มีค่ามากกว่า หรือน้อยกว่า Estimate Time ของฟีเจอร์

$$Different (\%) = \left(\frac{used_time}{estimate_time} * 100 \right) - 100$$

2.2) เมื่อได้ค่าความแตกต่างระหว่างเวลาตามแล้ว จะต้องหาค่าอัตราส่วนของการปรับ Rate ที่เกิดจากการทำงานของสมาชิกในทีมแต่ละคน เพื่อหาว่าสมาชิกแต่ละคน จะต้องมีการปรับ Performance ด้วยค่าเท่าใด โดยนำเวลาที่สมาชิกทำงานอยู่ในฟีเจอร์นั้น ๆ มาคิดเป็นร้อยละการทำงานภายในฟีเจอร์ว่าทำงานได้ร้อยละเท่าใด จากนั้นนำไปคำนวณกับค่าความแตกต่าง และหารด้วย 100^2 เพื่อให้ได้ค่าผลลัพธ์ในหน่วยเดียวกันกับ Performance ของสมาชิก

$$Rate_{user\ i} = \frac{(Different) * (\frac{used_time_{user\ i}}{used_time} * 100)}{100^2}$$

2.3) นำ Rate ของ User ไปปรับค่ากับค่า Performance ของ User แต่ละคน

$$Performance_{user\ i} += Rate_{user\ i}$$

ตัวอย่างการคำนวณ

กำหนดให้ ข้อมูลการทำงานของพนักงานในฟีเจอร์ A ซึ่งมีสถานะสิ้นสุดแล้ว ของบริษัทแห่งหนึ่ง มีค่าดังตารางที่ 3.3

ตารางที่ 3.3 ตัวอย่างข้อมูลการทำงานของพนักงานในฟีเจอร์

พนักงาน	เวลาที่ทำงานในฟีเจอร์
User A	100 Hours
User B	140 Hours
User C	160 Hours
Total	400 Hours

ขั้นตอนในการปรับค่า Performance ของสมาชิกในฟีเจอร์ มีดังนี้

1) หากค่าความแตกต่างระหว่างระยะเวลาที่คาดการณ์ของฟีเจอร์ (Estimate Time) กับระยะเวลาที่ใช้จริงของทีม (Used Time) จากสูตร

$$Different (\%) = \left(\frac{used_time}{estimate_time} * 100 \right) - 100$$

โดย used_time คือ 400 และ estimate_time ที่หาได้ของฟีเจอร์ A คือ 356.3739

จะได้

$$Different (\%) = \left(\frac{400}{356.3739} * 100 \right) - 100$$

$$Different (\%) = 12.2417$$

2) เมื่อได้ค่าความแตกต่างระหว่างเวลา (Different) มาแล้ว จะต้องหาค่าอัตราส่วนของการปรับ Rate ที่เกิดจากการทำงานของสมาชิกในทีมแต่ละคน เพื่อหาว่าสมาชิกแต่ละคน จะต้องมีการปรับ Performance ด้วยค่าเท่าใด

จากสูตร

$$Rate_{user i} = \frac{(Different) * \left(\frac{used_time_{user i}}{used_time} * 100 \right)}{100^2}$$

หา Rate ของ User A จะได้

$$Rate_{user A} = \frac{(Different) * \left(\frac{100}{400} * 100 \right)}{100^2}$$

$$Rate_{user A} = \frac{12.2417 * 25}{100^2}$$

$$Rate_{user A} = 0.0306$$

หา Rate ของ User B จะได้

$$Rate_{user B} = \frac{(Different) * \left(\frac{140}{400} * 100 \right)}{100^2}$$

$$Rate_{user B} = \frac{12.2417 * 35}{100^2}$$

$$Rate_{user B} = 0.0428$$

หา Rate ของ User C จะได้

$$Rate_{user C} = \frac{(Different) * \left(\frac{160}{400} * 100 \right)}{100^2}$$

$$Rate_{user C} = \frac{12.2417 * 40}{100^2}$$

$$Rate_{user C} = 0.0489$$

3) นำ Rate ของ User ไปปรับค่ากับค่า Performance ของ User แต่ละคนด้วยสูตร

$$Performance_{user\ i} += Rate_{user\ i}$$

ซึ่งจะได้ผลลัพธ์การปรับค่าดังตารางที่ 3.4

ตารางที่ 3.4 ผลลัพธ์การปรับค่า Performance ของสมาชิก

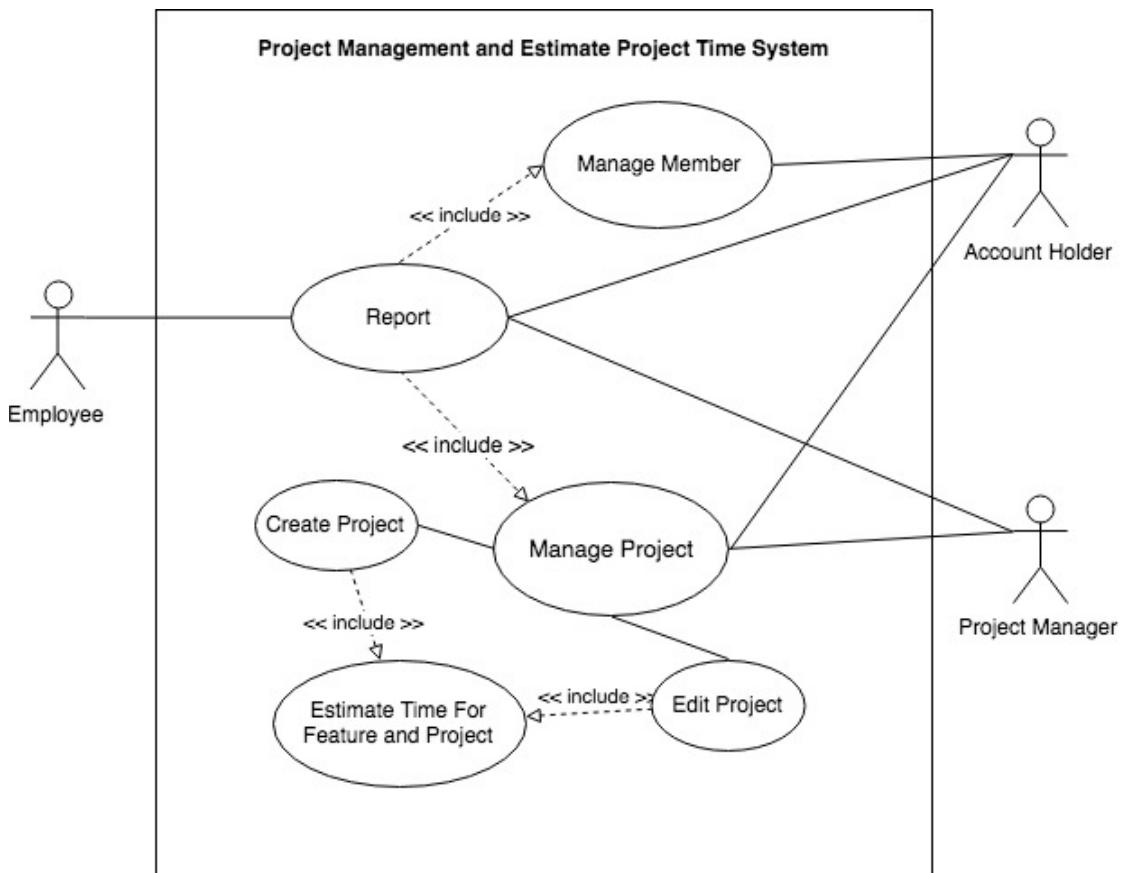
พนักงาน	Performance ก่อนปรับ	Performance หลังปรับ
User A	0.9706	1.0012
User B	0.6789	0.7217
User C	1.0714	1.1203

3.3 แผนภาพยูสเคส

แผนภาพยูสเคส (Use Case Diagram) ใช้สำหรับแสดงผู้ที่เกี่ยวข้องกับระบบ และแสดงพีเจอร์ต่าง ๆ ของระบบ โดยมีสัญลักษณ์ที่แสดงในแผนภาพ ดังตารางที่ 3.5

ตารางที่ 3.5 สัญลักษณ์และความหมายของแผนภาพยูสเคส

ชื่อสัญลักษณ์	คำอธิบาย	รูปสัญลักษณ์
Use Case Name	สิ่งต่าง ๆ ที่ระบบสามารถทำได้	
Actor Name	ผู้ที่เกี่ยวข้องกับระบบ	
Relationship	แสดงความสัมพันธ์ระหว่าง Actor กับ Use case	
Generalization / Specialization	แสดงความสัมพันธ์ในเชิงการจำแนก แยกแยะประเภทของ Use case	
Include Relationship	ความสัมพันธ์แบบเรียกใช้ คือมี Base use case หนึ่งไปเรียกใช้กิจกรรม ของอีก Use case หนึ่งมา เพื่อให้ตัว Base use case นั้นเกิดขึ้นได้	



ภาพที่ 3.2 แผนภาพยุสเคสระบบจัดการข้อมูลและการจัดการระยะเวลาดำเนินงานของโครงการ
จากแผนภาพยุสเคสแสดงกล่าว แสดงให้เห็นถึงผู้เกี่ยวข้องกับระบบ ซึ่งประกอบไปด้วยพนักงาน (Employee) ผู้จัดการโครงการ (Project Manager) และเจ้าของบัญชีบริษัท (Account Holder)
และแสดงสิ่งที่ระบบต้องดำเนินการ โดยสามารถอธิบายรายละเอียดของแต่ละ Use case ได้ดังตาราง
ต่อไปนี้

ตารางที่ 3.6 รายละเอียด Use Case “Manage Member”

ID	UC01
Name	Manage Member
Actor	Account Holder
Description	ผู้ใช้สามารถจัดการสมาชิกภายในระบบได้
Pre-Condition	-
Flow of Events	1) ผู้ใช้เข้าสู่หน้าจัดการข้อมูลสมาชิก 2) สามารถเลือกเมนู เพิ่มสมาชิก / ลบสมาชิก ได้

ตารางที่ 3.7 รายละเอียด Use Case “Manage project”

ID	UC02
Name	Manage project
Actor	Account Holder, Project Manager
Description	ผู้ใช้สามารถจัดการข้อมูลโครงการได้
Pre-Condition	ผู้ใช้ต้อง Login เข้าสู่ระบบ ด้วยประเภทสมาชิกที่กำหนดเท่านั้น
Flow of Events	<ul style="list-style-type: none"> 1) ผู้ใช้เข้าสู่หน้าแสดงโครงการ 2) สามารถเลือกเมนูสร้างโครงการ / แก้ไขโครงการ / ลบโครงการ ได้

ตารางที่ 3.8 รายละเอียด Use Case “Create Project”

ID	UC03
Name	Create Project
Actor	Account Holder, Project Manager
Description	ผู้ใช้สามารถสร้างโครงการและพีเจอร์ได้
Pre-Condition	-
Flow of Events	<ul style="list-style-type: none"> 1) ผู้ใช้เข้าสู่หน้าสร้างโครงการ 2) กรอกข้อมูลโครงการ เลือกสมาชิกในโครงการ 3) สร้างพีเจอร์ย่อย และเลือกสมาชิกในพีเจอร์ 4) กดตกลง เพื่อสร้างโครงการ

ตารางที่ 3.9 รายละเอียด Use Case “Edit Project”

ID	UC04
Name	Edit Project
Actor	Account Holder, Project Manager
Description	ผู้ใช้สามารถแก้ไขข้อมูลโครงการและพีเจอร์ได้
Pre-Condition	-
Flow of Events	<ul style="list-style-type: none"> 1) ผู้ใช้เข้าสู่หน้าแก้ไขข้อมูลโครงการ 2) แก้ไขข้อมูลโครงการ แก้ไขสมาชิกในโครงการ 3) แก้ไขพีเจอร์ย่อย และแก้ไขสมาชิกในพีเจอร์ 4) กดตกลง เพื่อบันทึกโครงการ

ตารางที่ 3.10 รายละเอียด Use Case “Estimate Time For Feature And Project”

ID	UC05
Name	Estimate Time For Feature And Project
Actor	Account Holder, Project Manager
Description	เมื่อผู้ใช้สร้างหรือแก้ไขโปรเจค และทำการเลือกสมาชิกในทีมฟีเจอร์ใหม่ และกรอกข้อมูลขนาดของฟีเจอร์ ระบบจะคาดการณ์ระยะเวลาที่เหมาะสมสำหรับการทำฟีเจอร์นั้น ๆ จากข้อมูลดังกล่าว
Pre-Condition	-
Flow of Events	<ul style="list-style-type: none"> 1) ผู้ใช้เข้าสู่หน้าสร้าง / แก้ไขโปรเจค 2) เลือกสมาชิกในทีมฟีเจอร์ 3) กรอกลักษณะฟีเจอร์ที่จะทำ และขนาดของงาน 4) ผู้ใช้ดูผลลัพธ์ของการคาดการณ์ระยะเวลาการทำงาน

ตารางที่ 3.11 รายละเอียด Use Case “Report”

ID	UC06
Name	Report
Actor	Employee, Account Holder, Project Manager
Description	ผู้ใช้สามารถดูรายงานสรุปการนับเวลาได้
Pre-Condition	ต้องมีการ Manage project ก่อน เพื่อให้มีข้อมูลสำหรับการขอรายงาน และต้องมีการ Manage Member ก่อนด้วย
Flow of Events	<ul style="list-style-type: none"> 1) ผู้ใช้เข้าสู่หน้าแสดงรายงาน

3.4 แผนภาพกิจกรรม

แผนภาพกิจกรรม (Activity Diagram) ใช้แสดงภาพรวมของระบบ มีสัญลักษณ์ที่แสดงในแผนภาพ ดังตารางที่ 3.12

ตารางที่ 3.12 สัญลักษณ์และความหมายของแผนภาพกิจกรรม

ชื่อสัญลักษณ์	คำอธิบาย	รูปสัญลักษณ์
Activity	กิจกรรม	
Start	จุดเริ่มต้นกิจกรรม	
End	จุดสิ้นสุดกิจกรรม	

ตารางที่ 3.12 สัญลักษณ์และความหมายของแผนภาพกิจกรรม (ต่อ)

ชื่อสัญลักษณ์	คำอธิบาย	รูปสัญลักษณ์
Condition	เงื่อนไขของกิจกรรม	<pre> graph TD A(()) -- "[x<0]" --> B A -- "[x=0]" --> C A -- "[x>0]" --> D </pre>
Swim Lanes	การแบ่งช่องของกิจกรรมตามผู้ที่เกี่ยวข้องในแต่ละกิจกรรม	<pre> graph LR Lane1[ผู้ที่เกี่ยวข้อง 1] Lane2[ผู้ที่เกี่ยวข้อง 2] </pre>

แผนภาพกิจกรรมของระบบจัดการข้อมูลและการคาดการณ์ระยะเวลาดำเนินงานของโครงการแสดงภาพรวมของระบบ ดังภาพที่ 3.3 ซึ่งอธิบายได้ 3 ลักษณะการใช้งานคือ

1) ผู้ใช้สามารถเลือกดูรายงานการนับเวลา (Report) ได้ โดยในหน้ารายงานจะประกอบไปด้วย

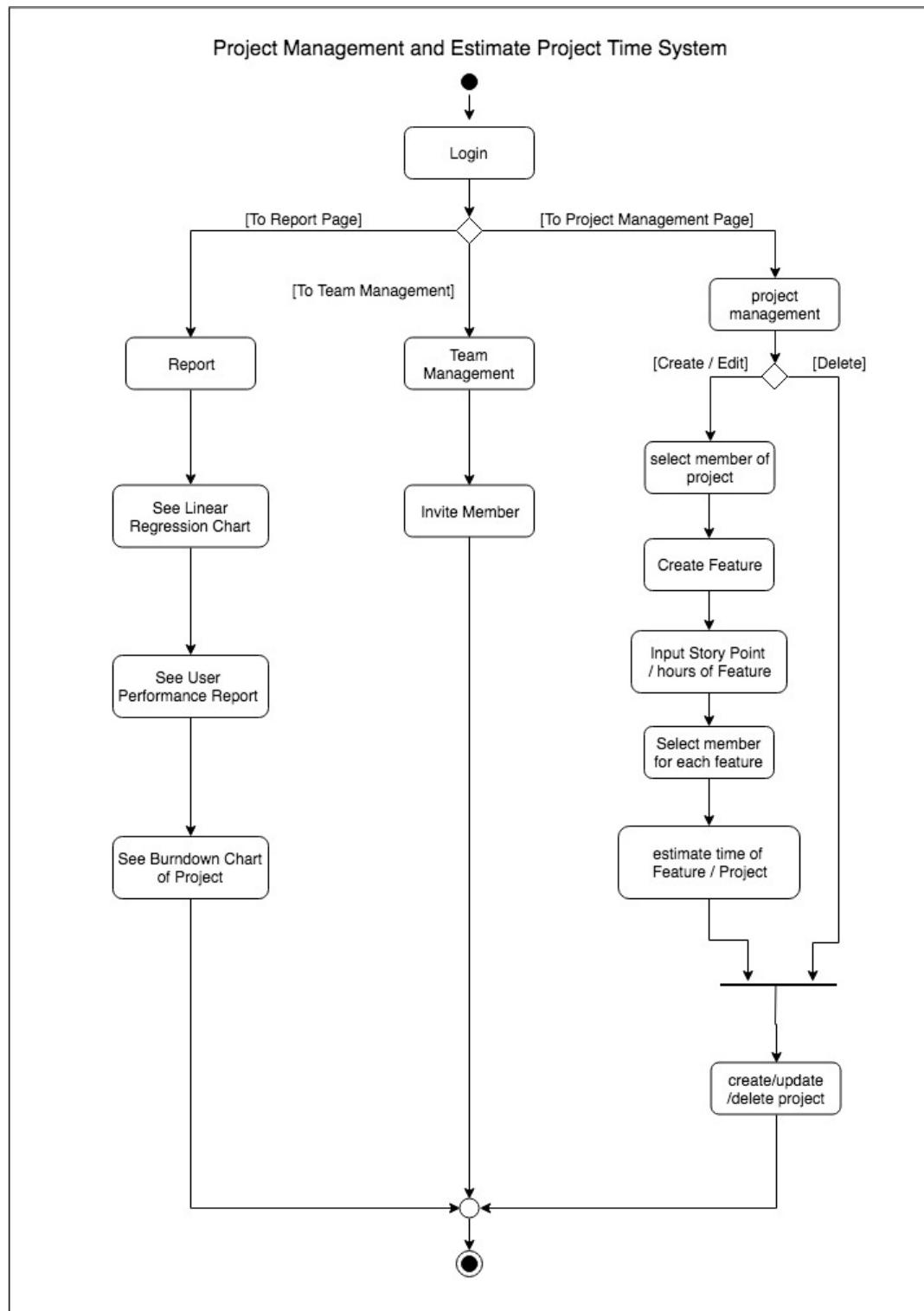
- กราฟแสดงสมการ Simple Linear Regression ของบริษัทของผู้ใช้งาน
- ตารางแสดงรายชื่อสมาชิกของบริษัท และแสดง Performance ของสมาชิกแต่ละคน
- กราฟแสดงอัตราการทำงานของแต่ละโปรเจค โดยแสดงผลในรูปแบบของ Burndown Chart โดยใช้หน่วยนับเวลาเป็นหน่วยชั่วโมงแทนหน่วยนับเวลาเป็นสัปดาห์

2) ผู้ใช้สามารถจัดการสมาชิกแต่ละคนในระบบได้ โดยสามารถเพิ่มผู้ใช้งานใหม่เข้าสู่ระบบ

เว็บไซต์ โดยการกรอกข้อมูลและอีเมล์ของผู้ใช้ใหม่ เพื่อเพิ่มผู้ใช้ใหม่เข้ามาใช้งานระบบได้

3) ผู้ใช้สามารถจัดการข้อมูลโปรเจคได้ โดยที่สามารถเพิ่ม ลบ หรือแก้ไขข้อมูลโปรเจคได้ ซึ่งในการจัดการข้อมูลโปรเจคนี้จะมีพิงก์ชันการคาดการณ์ระยะเวลาการดำเนินการที่เหมาะสมของโปรเจค

อยู่ด้วย การทำงานคือ ผู้ใช้จะต้องสร้างโปรเจคขึ้นมาใหม่ โดยเลือกสมาชิกในโปรเจคนั้น ๆ จากนั้นทำการสร้างฟีเจอร์อย่างของโปรเจค และทำการกำหนดขนาดของฟีเจอร์ (Story Point) กำหนดระยะเวลาของฟีเจอร์ (Estimate Time) ให้แต่ละฟีเจอร์ด้วย เมื่อเสร็จแล้วให้เลือกสมาชิกในแต่ละฟีเจอร์ จากนั้นระบบจะทำการคำนวณว่า ฟีเจอร์นั้น ๆ จะใช้ระยะเวลาทำงานเท่าใด โดยดูจากข้อมูลขนาดของฟีเจอร์ นำมาหารร่วมกับสมการ Simple Linear Regression ของบริษัท และร่วมกับการคำนวณโดยใช้ Performance ของสมาชิกในทีมฟีเจอร์



ภาพที่ 3.3 แผนภาพกิจกรรมระบบจัดการข้อมูลและการคาดการณ์ระยะเวลาดำเนินงานของโครงการ

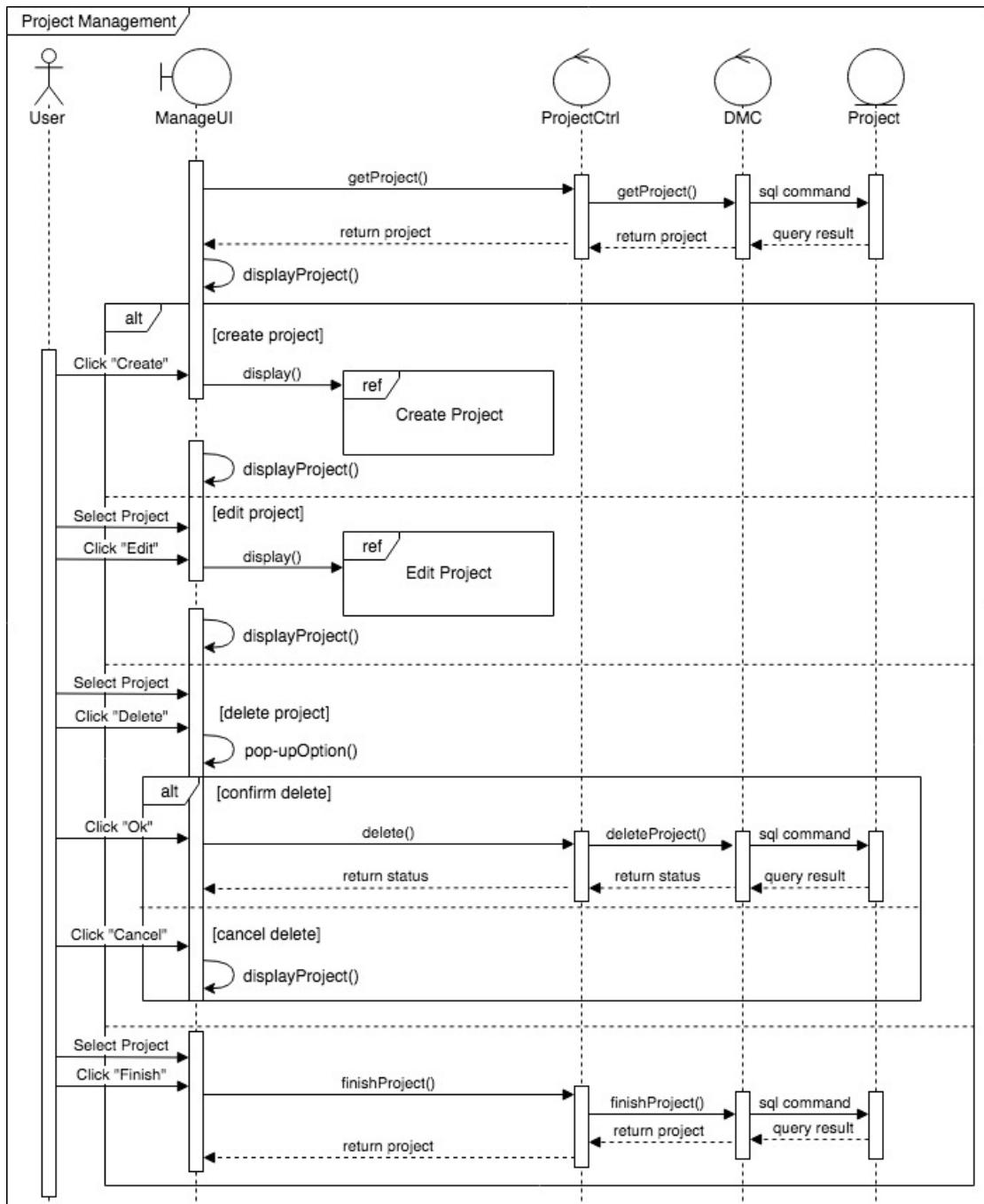
3.5 แผนภาพชีวเคนซ์ไดอะแกรม

แผนภาพชีวเคนซ์ไดอะแกรม (Sequence Diagram) ใช้สำหรับการแสดงรายการและอีดการทำงานของกระบวนการต่าง ๆ มีสัญลักษณ์ที่ใช้แสดงในแผนภาพ ดังตารางที่ 3.13

ตารางที่ 3.13 สัญลักษณ์และความหมายของแผนภาพชีวเ肯ซ์ไดอะแกรม

ชื่อสัญลักษณ์	คำอธิบาย	รูปสัญลักษณ์
Actor	ผู้ที่เกี่ยวข้องกับระบบ	
Object	อ็อบเจกต์ หรือคลาสที่มีการตอบสนองต่อผู้ใช้งาน	
Life line	เส้นแสดงช่วงชีวิตของ อ็อบเจกต์ หรือคลาส	
Focus of Control / Activation	จุดเริ่มต้น และจุดสิ้นสุดของแต่ละกิจกรรมในระหว่างนั้น	
Message	การส่งข้อความ คำสั่ง หรือ พิงก์ชันไปยังอ็อบเจกต์หนึ่ง (เส้นทึบ) และอ็อบเจกต์ที่รับข้อความ สามารถส่งคืนค่ากลับมาได้ (เส้นประ)	
Callback or Self Delegation	การประมวลผล และคืนค่าที่ทำภัยในอ็อบเจกต์เดียวกัน	
Loop	การวนซ้ำ	
Condition	แสดงเงื่อนไขการตัดสินใจ	

ระบบจัดการข้อมูลและการจัดการณ์ระยะเวลาดำเนินงานของโครงการ มีการออกแบบฟีเจอร์หลักของการทำงานด้วยชีเควนซ์โดยแกรมดังต่อไปนี้

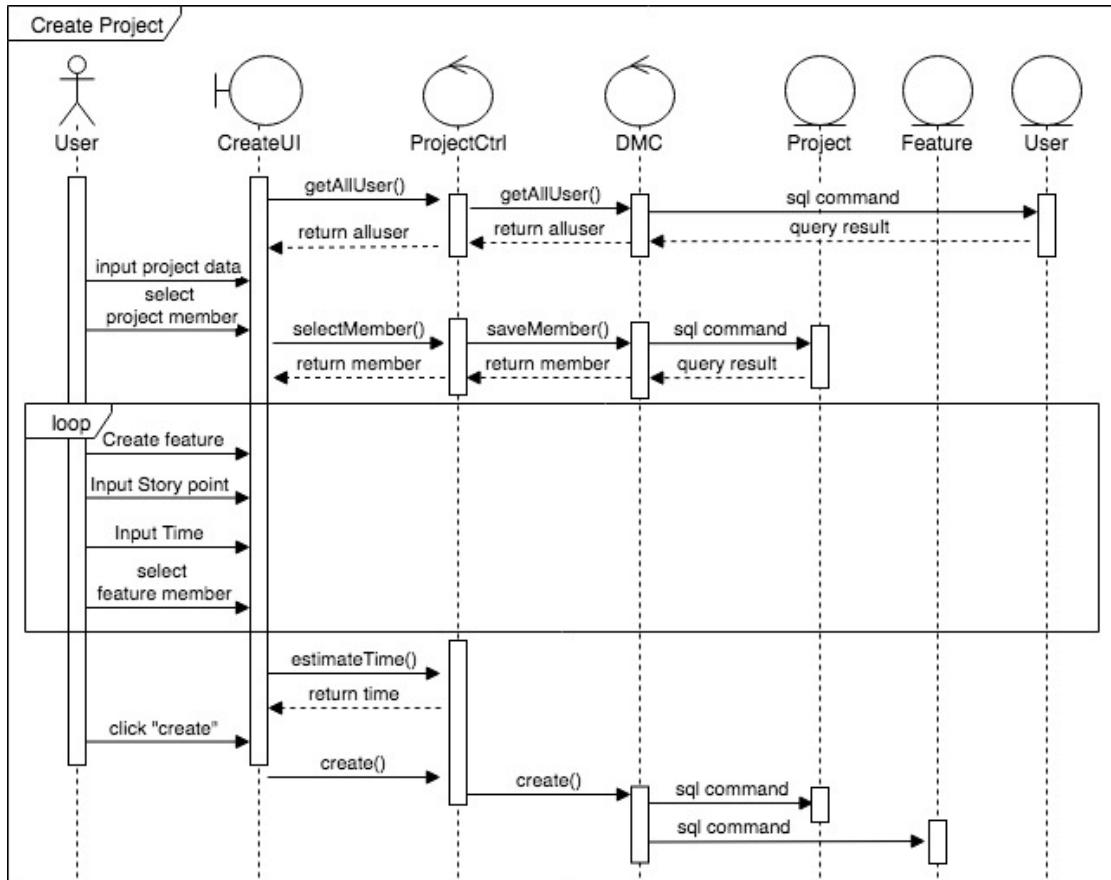


ภาพที่ 3.4 ชีเควนซ์โดยแกรมการจัดการข้อมูลโปรเจค

จากการที่ 3.4 การทำงานของระบบจัดการข้อมูลโปรเจคนั้น ผู้ใช้สามารถดำเนินการได้ 4 อย่างคือ สร้างโปรเจค แก้ไขโปรเจค ลบโปรเจค และเสร็จสิ้นโปรเจค

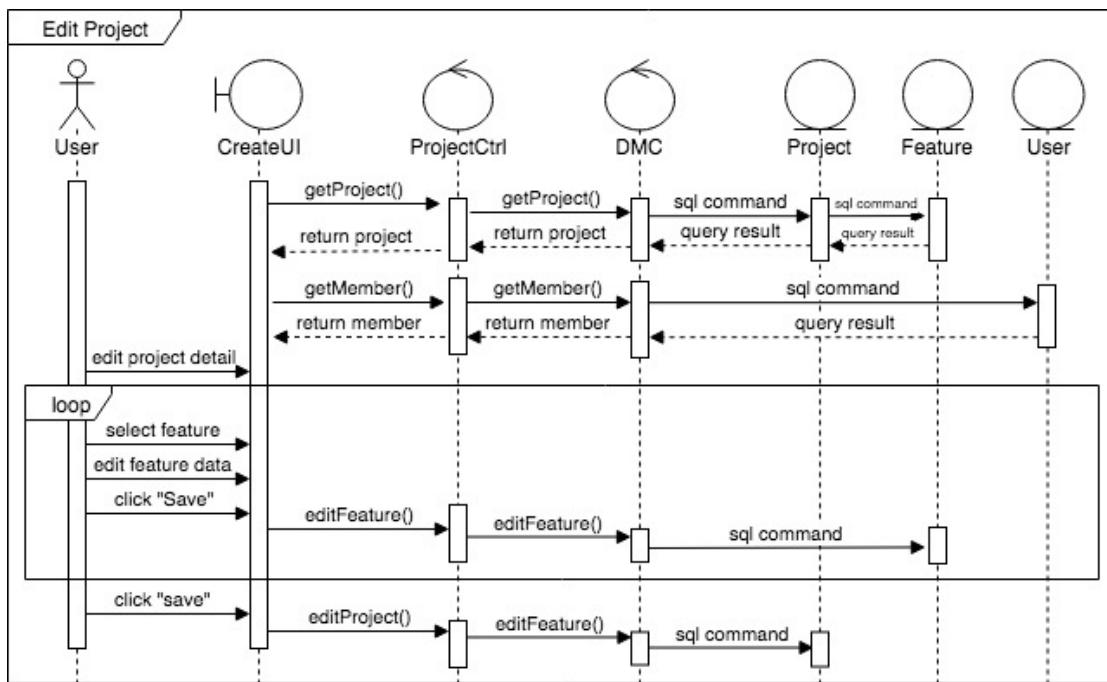
โดยในการลบโปรเจค ผู้ใช้ต้องเลือกโปรเจคที่ต้องการ จากนั้นจะกดปุ่ม “Delete” จากนั้นจะมี Popup เพื่อให้ยืนยันการลบ เมื่อยืนยันแล้วระบบจะทำการลบโปรเจคออก

นอกจากนี้ ในการเสร็จสิ้นโปรเจคนั้น ผู้ใช้ต้องเลือกโปรเจคที่ต้องการ จากนั้นจะกดปุ่ม Finish เพื่อเสร็จสิ้นโปรเจค



ภาพที่ 3.5 ชีวนิชีของโปรแกรมการสร้างโปรเจค

จากภาพที่ 3.5 คือขั้นตอนการสร้างโปรเจคใหม่ โดยผู้ใช้จะใส่ข้อมูลโปรเจค จากนั้นจะเลือกสมาชิกในโปรเจค แล้วมาสร้างฟีเจอร์ โดยเริ่มจากใส่ชื่อฟีเจอร์ ใส่ค่า Story Point และใส่เวลาที่จะใช้จากนั้นเลือกสมาชิกในฟีเจอร์ จนครบทุกฟีเจอร์ที่จะสร้าง จากนั้นระบบจะประมวลผลเวลาอ กมา และผู้ใช้จะกดสร้างโปรเจค เป็นอันเสร็จสมบูรณ์



ภาพที่ 3.6 ชีวเอนซ์โดยการแก้ไขข้อมูลโปรเจค

จากภาพที่ 3.6 คือขั้นตอนการแก้ไขข้อมูลโปรเจค โดยผู้ใช้จะแก้ไขข้อมูลโปรเจค จากนั้นจะแก้ไขข้อมูลของฟีเจอร์แต่ละอัน เมื่อแก้ไขเสร็จในแต่ละฟีเจอร์แล้วจะกด Save ของฟีเจอร์ เพื่อให้ระบบบันทึกการแก้ไข จากนั้นเมื่อเสร็จสิ้นทุกอย่างแล้ว ผู้ใช้จะกดปุ่ม Save ของโปรเจค เพื่อบันทึกการเปลี่ยนแปลงทั้งหมดในโปรเจคอีกรอบหนึ่ง

บทที่ 4

การออกแบบฐานข้อมูล

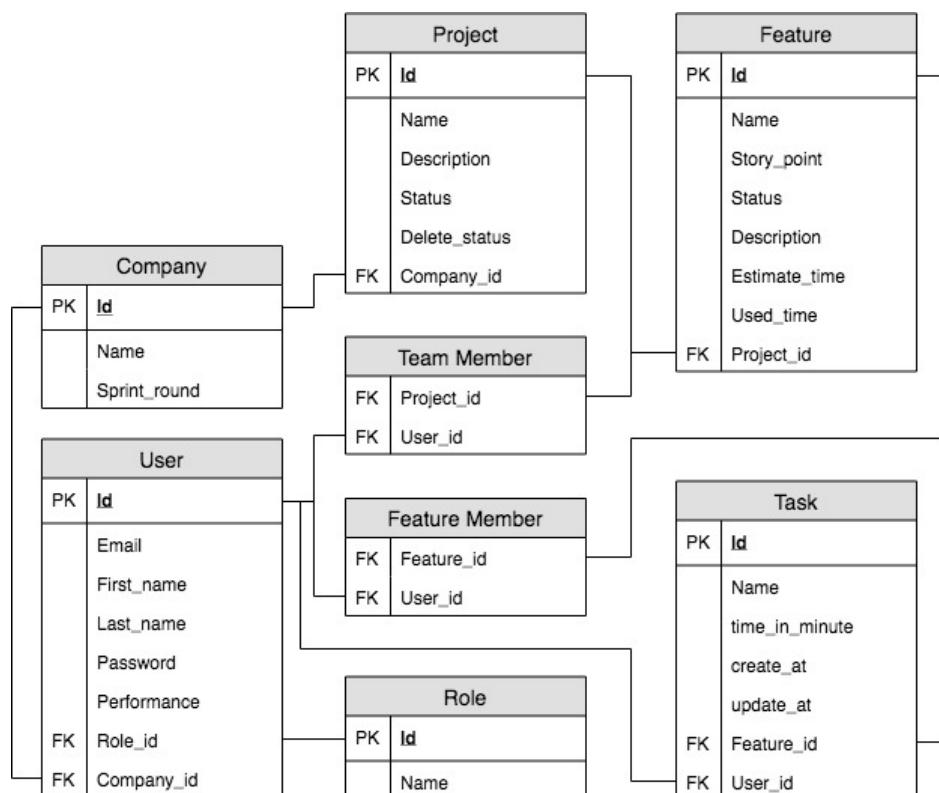
ในบทนี้จะกล่าวถึงการออกแบบฐานข้อมูลของระบบจัดการข้อมูลและการณ์ระยะเวลาดำเนินงานของโครงการ จุดประสงค์เพื่อเรียกใช้ฐานข้อมูลได้สะดวก และเกิดประโยชน์สูงสุด รายละเอียดต่อไปนี้จะกล่าวถึงการออกแบบฐานข้อมูลของระบบ ซึ่งมีรายละเอียดดังนี้

4.1 ลักษณะของตารางความสัมพันธ์ของฐานข้อมูล

ตารางความสัมพันธ์ของฐานข้อมูล สามารถแสดงความสัมพันธ์ระหว่างข้อมูลในตารางได้ดังนี้

ตารางที่ 4.1 ความหมายและสัญลักษณ์ของตารางความสัมพันธ์ของฐานข้อมูล

สัญลักษณ์	ความหมาย
	ตารางข้อมูล เป็นที่จัดเก็บข้อมูล
PK	คีย์หลัก
FK	คีย์รอง



ภาพที่ 4.1 ตารางความสัมพันธ์ของฐานข้อมูล

4.2 รายละเอียดข้อมูลของตาราง

จากตารางความสัมพันธ์ของฐานข้อมูล มีรายละเอียดของโครงสร้างข้อมูลในแต่ละตาราง ประกอบด้วยชื่อตาราง ชนิดของตาราง คำอธิบายตาราง คีย์หลัก คีย์นอก ชื่อข้อมูล ชนิดข้อมูล คำอธิบาย ตัวอย่างข้อมูล โดยแสดงรายละเอียดดังตารางที่ 4.2 ถึง 4.9 ดังนี้

ตารางที่ 4.2 รายละเอียดตารางข้อมูลผู้ใช้งาน

ชื่อตาราง : User				
รายละเอียด : เก็บข้อมูลผู้ใช้งานของระบบ				
คีย์หลัก : Id				
คีย์นอก : role_id จากตาราง Role, company_id จากตาราง Company				
ชื่อข้อมูล	ชนิดข้อมูล	ขนาด(ไบต์)	คำอธิบาย	ตัวอย่าง
Id	Integer	8	รหัสประจำตัวผู้ใช้งาน	153
email	Varchar	255	อีเมล์ของผู้ใช้งาน	chanatip@email.com
first_name	Varchar	255	ชื่อของผู้ใช้งาน	นนทการ
last_name	Varchar	255	นามสกุลของผู้ใช้งาน	ซัยบادல
password	Varchar	255	รหัสของผู้ใช้งาน	123456
performance	Float	8	ประสิทธิภาพการทำงาน ทำงานของสมาชิก	0.937266
role_id	Integer	8	รหัสประเภทผู้ใช้งาน	4
company_id	Integer	8	รหัสบริษัทของผู้ใช้งาน	1

ตารางที่ 4.3 รายละเอียดตารางบริษัท

ชื่อตาราง : Company				
รายละเอียด : เก็บข้อมูลรายชื่อบริษัทต่าง ๆ ที่ใช้งานระบบนี้				
คีย์หลัก : id				
ชื่อข้อมูล	ชนิดข้อมูล	ขนาด(ไบต์)	คำอธิบาย	ตัวอย่าง
id	Integer	8	รหัสบริษัท	1
name	Varchar	255	ชื่อของบริษัท	Chiang Mai Company
sprint_round	Integer	8	เวลาการตัดรอบการ นับเวลาการทำงาน (สัปดาห์)	2

ตารางที่ 4.4 รายละเอียดตารางประเภทของผู้ใช้งาน

ชื่อตาราง : Role				
รายละเอียด : เก็บข้อมูลประเภทของผู้ใช้งานในระบบ				
คีย์หลัก : id				
ชื่อข้อมูล	ชนิดข้อมูล	ขนาด(ไบต์)	คำอธิบาย	ตัวอย่าง
id	Integer	8	รหัสของประเภท	3
name	Varchar	255	ชื่อประเภทของผู้ใช้	Project Manager

ตารางที่ 4.5 รายละเอียดตารางโปรเจค

ชื่อตาราง : Project				
รายละเอียด : เก็บข้อมูลโปรเจคในระบบ				
คีย์หลัก : id				
คืนออก : company_id จากตาราง Company				
ชื่อข้อมูล	ชนิดข้อมูล	ขนาด(ไบต์)	คำอธิบาย	ตัวอย่าง
id	Integer	8	รหัสโปรเจค	142
Name	Varchar	255	ชื่อโปรเจค	Sandclock
description	Varchar	255	คำอธิบายโปรเจค	เว็บไซต์นี้เป็นระบบจัดการ.....
status	Boolean	1	สถานการณ์เสร็จสิ้น	False
delete_status	Boolean	1	สถานการณ์ลบ	True
company_id	Integer	8	รหัสบริษัท	1

ตารางที่ 4.6 รายละเอียดตารางสมาชิกในโปรเจค

ชื่อตาราง : TeamMember				
รายละเอียด : เก็บข้อมูลสมาชิกที่อยู่ในโปรเจค				
คีย์หลัก : project_id ร่วมกับ user_id				
คืนออก : project_id จากตาราง Project, user_id จากตาราง User				
ชื่อข้อมูล	ชนิดข้อมูล	ขนาด(ไบต์)	คำอธิบาย	ตัวอย่าง
project_id	Integer	8	รหัสโปรเจค	15
user_id	Integer	8	รหัสประจำตัวผู้ใช้งาน	92

ตารางที่ 4.7 รายละเอียดตารางฟีเจอร์

ข้อมูล : Feature				
รายละเอียด : เก็บข้อมูลฟีเจอร์ของแต่ละโปรเจค				
คีย์หลัก : id				
คืนออก : project_id จากตาราง Project				
ชื่อข้อมูล	ชนิดข้อมูล	ขนาด(ไบต์)	คำอธิบาย	ตัวอย่าง
id	Integer	8	รหัสประจำงาน	2
name	Varchar	255	ชื่อประเภทของงาน	Development
story_point	Integer	8	ขนาดของงาน	8
status	Boolean	1	สถานะเสร็จสิ้น	False
description	Varchar	8	คำอธิบายฟีเจอร์	Mobile Application in importamt
estimate_time	Float	8	ระยะเวลาที่ถูกคาดการณ์ของฟีเจอร์	240.58888888
used_time	Float	8	ระยะเวลาที่ถูกใช้งานจริง ๆ ของฟีเจอร์	237.28937008
project_id	Integer	8	รหัสโปรเจค	1

ตารางที่ 4.8 รายละเอียดตารางสมาชิกในฟีเจอร์

ข้อมูล : TeamMember				
รายละเอียด : เก็บข้อมูลสมาชิกที่อยู่ในฟีเจอร์ต่าง ๆ				
คีย์หลัก : feature_id ร่วมกับ user_id				
คืนออก : feature_id จากตาราง Feature, user_id จากตาราง User				
ชื่อข้อมูล	ชนิดข้อมูล	ขนาด(ไบต์)	คำอธิบาย	ตัวอย่าง
feature_id	Integer	8	รหัสฟีเจอร์	15
user_id	Integer	8	รหัสประจำตัวผู้ใช้งาน	92

ตารางที่ 4.9 รายละเอียดตารางงานนับเวลา

ชื่อตาราง : Task				
รายละเอียด : เก็บข้อมูลรายละเอียดของงานนับเวลา				
คีย์หลัก : id				
คีย์นอก : feature_id จากตาราง Feature, user_id จากตาราง User				
ชื่อข้อมูล	ชนิดข้อมูล	ขนาด(ไบต์)	คำอธิบาย	ตัวอย่าง
id	uuid	32	รหัสของงานนับเวลา	2f2969ce-07e2-49c3-9df3-68d36ef07e6c
name	Varchar	255	ชื่องานนับเวลา	Create Task name
created_at	Datetime	1	เวลาที่สร้างงาน	2018-11-14 04:52:23.749881
updated_at	Datetime	1	เวลาที่อัพเดทงาน	2018-11-21 09:11:45.818308
time_in_minute	Float	8	จำนวนเวลาที่นับได้	182
feature_id	Integer	8	รหัสของฟีเจอร์	3
user_id	Integer	8	รหัสผู้ใช้งาน	15

บทที่ 5

การออกแบบส่วนต่อประสานกับผู้ใช้

การออกแบบส่วนต่อประสานระหว่างระบบและผู้ใช้ที่ดี เป็นสิ่งสำคัญที่ทำให้ผู้ใช้งานสามารถใช้งานระบบได้อย่างมีประสิทธิภาพ ดังนั้น จึงต้องมีการกำหนดองค์ประกอบต่าง ๆ ของระบบ เพื่อความสะดวกในการใช้งานของผู้ใช้

5.1 สิทธิ์ในการเข้าถึงในแต่ละหน้าจอ

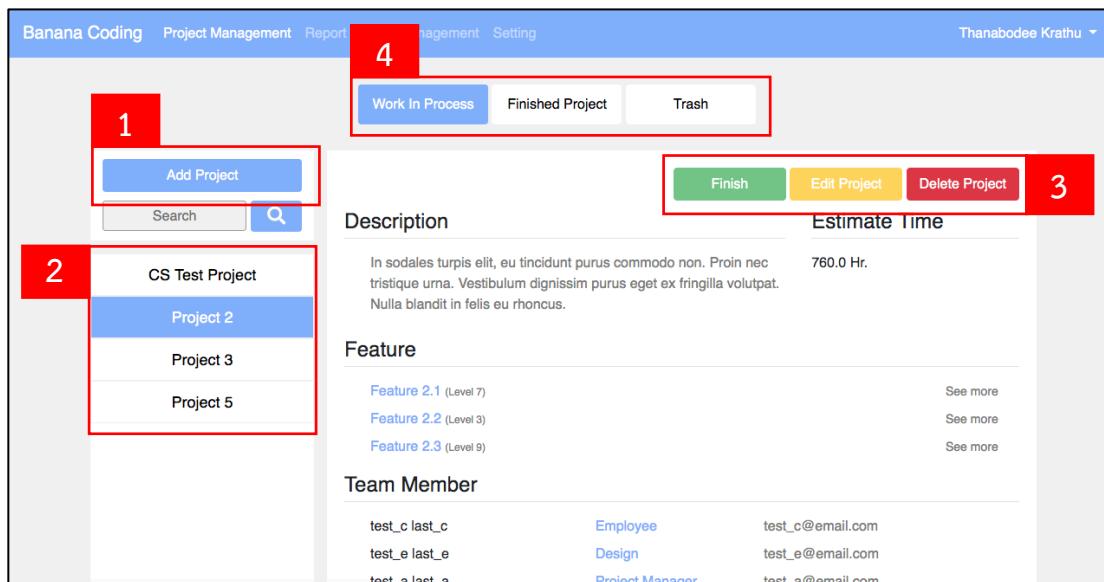
การออกแบบส่วนต่อประสานในระบบจัดการข้อมูลและการณ์ระยะเวลาดำเนินงานของโครงการ มีการแยกประเภทของผู้เข้าใช้งาน ซึ่งในแต่ละส่วนผู้ใช้งานแต่ละประเภทจะมีสิทธิ์เข้าถึงได้ต่างกันดังแสดงในตารางที่ 5.1

ตารางที่ 5.1 การเข้าถึงหน้าจอของผู้ใช้แต่ละประเภท

ชื่อหน้าจอ	ประเภทผู้ใช้	Account Holder	Project Manager	Employee
จัดการข้อมูลโปรเจคในระบบ (ดูข้อมูลโปรเจค)	✓	✓	✓	
จัดการข้อมูลโปรเจคในระบบ (เพิ่ม / ลบ / แก้ไข โปรเจค)	✓	✓		
รายงานของระบบ	✓	✓	✓	
จัดการข้อมูลสมาชิกในระบบ	✓			
การตั้งค่าระบบ	✓			

5.2 การออกแบบส่วนต่อประสานกับผู้ใช้

การออกแบบส่วนต่อประสานกับผู้ใช้ มีหน้าจอหลัก 6 หน้าจอดังแสดงในภาพที่ 5.1 ถึงภาพที่ 5.6



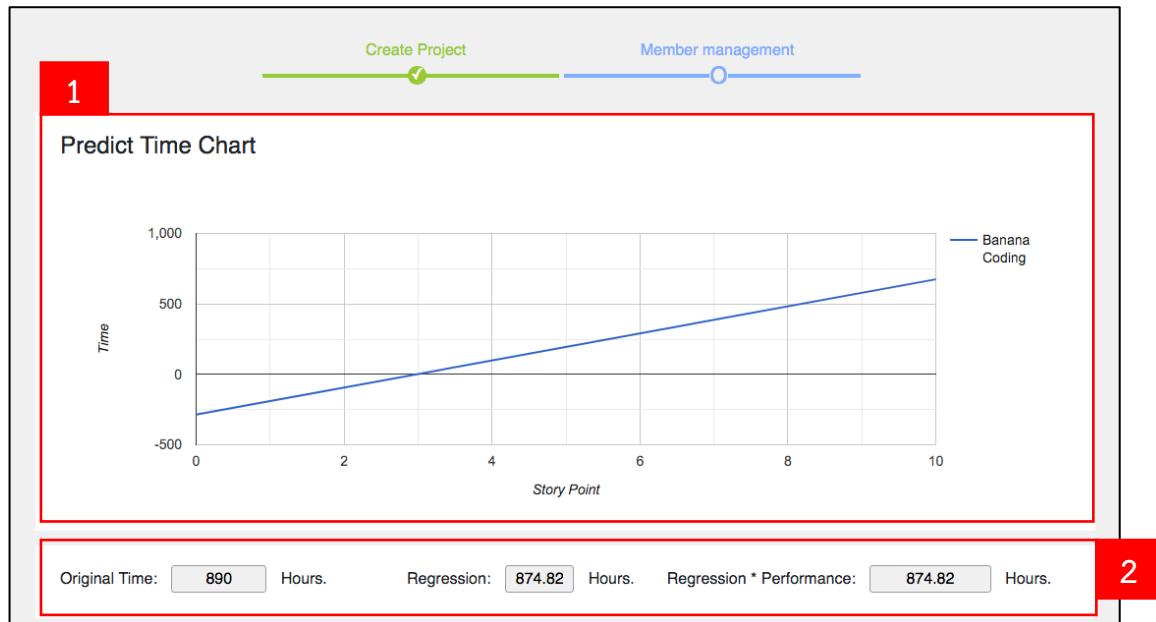
ภาพที่ 5.1 หน้าจอสำหรับจัดการข้อมูลโครงการในระบบ

จากภาพที่ 5.1 คือหน้าจอสำหรับจัดการข้อมูลโครงการ โดยสามารถกดปุ่มเพิ่มโครงการ (หมายเลข 1) ได้ หรือ สามารถเลือกโครงการ (หมายเลข 2) และกดแก้ไขโครงการ และลบโครงการได้ (หมายเลข 3) และมีเมนูสำหรับกดเพื่อดูโครงการที่มีสถานะเสร็จสิ้นแล้ว และกดเพื่อดูโครงการที่มีสถานะกลับแล้ว (หมายเลข 4)

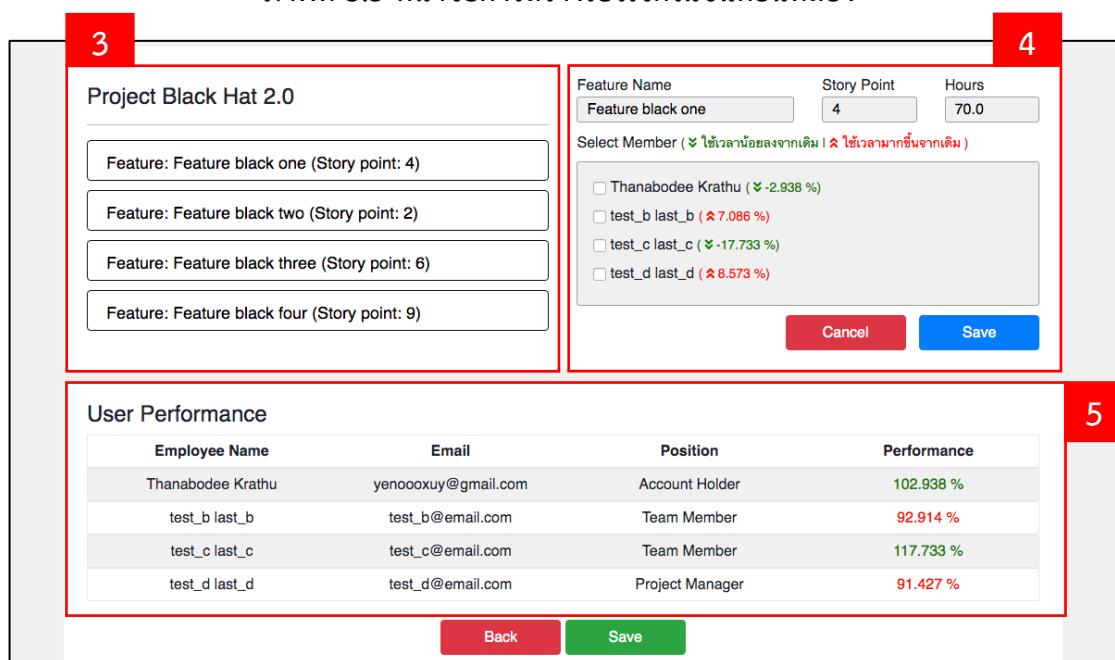
ภาพที่ 5.2 หน้าจอการสร้างโครงการในขั้นตอนแรก

จากภาพที่ 5.2 คือหน้าจอสำหรับการสร้างโครงการในขั้นตอนแรก โดยผู้ใช้สามารถกรอกข้อมูล ชื่อโครงการ คำอธิบายโครงการ และเลือกสมาชิกในโครงการได้ (หมายเลข 1) จากนั้นผู้ใช้สามารถสร้างฟีเจอร์อย่างของโครงการได้ในหน้าเดียวกัน (หมายเลข 2) โดยสามารถกรอกชื่อฟีเจอร์ ขนาดของ

ฟีเจอร์ เวลาที่คาดว่าจะใช้ของฟีเจอร์ รวมไปถึงคำอธิบายของฟีเจอร์ได้ โดยมีส่วนแสดงผลฟีเจอร์ที่ถูกสร้างแล้วอยู่ทางด้านล่าง (หมายเลข 3)



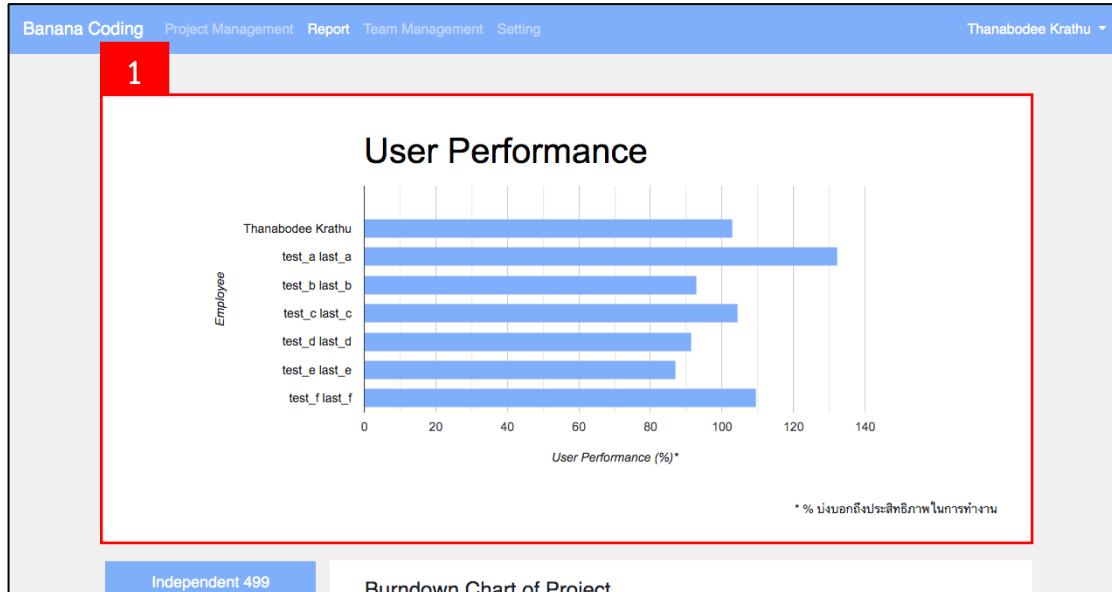
ภาพที่ 5.3 หน้าจອกรสร้างโปรเจคในขั้นตอนที่สอง



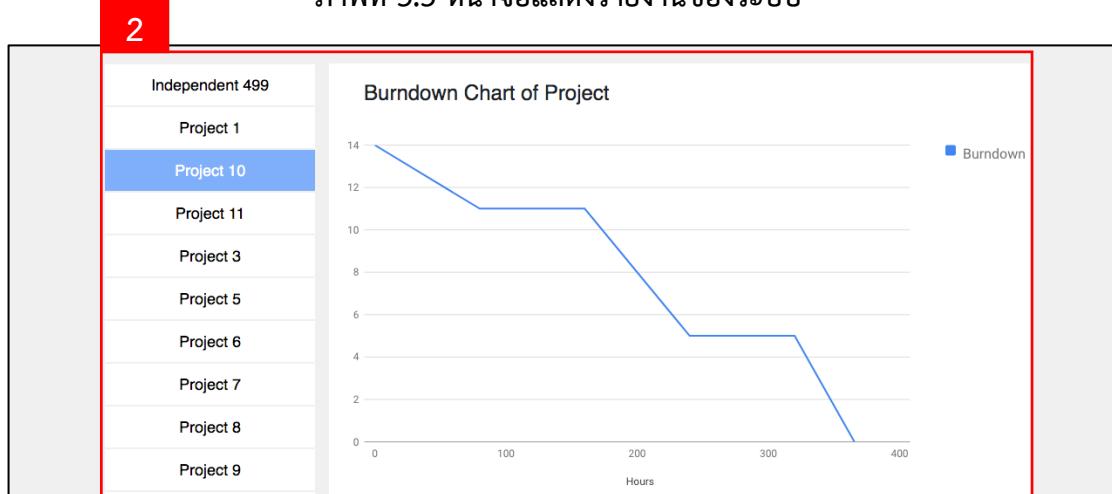
ภาพที่ 5.4 หน้าจອกรสร้างโปรเจคในขั้นตอนที่สอง

จากภาพที่ 5.3 และภาพที่ 5.4 คือหน้าจอสำหรับการสร้างโปรเจคในขั้นตอนที่สอง จะแสดงข้อมูลที่มาจากการน้ำขั้นตอนแรก โดยในภาพที่ 5.3 จะมีการแสดงผลกราฟ Predict Time Chart ที่ใช้ประเมินเวลาการทำงานของบริษัท (หมายเลข 1) และมีการแสดงผลการคาดการณ์ระยะเวลาที่เหมาะสมของโปรเจคจากการเลือกสมาชิกในฟีเจอร์ของผู้ใช้ (หมายเลข 2) และในภาพที่ 5.4 จะมีส่วนสำหรับให้ผู้ใช้เลือกฟีเจอร์ที่ต้องการ (หมายเลข 3) เพื่อกดเลือกสมาชิกทีมของฟีเจอร์ต่าง ๆ ได้

รวมไปถึงสามารถแก้ไขข้อมูลฟีเจอร์ที่เคยสร้างไปก่อนหน้านี้ได้ (หมายเลข 4) นอกจากนี้ ยังมีการแสดงรายชื่อสมาชิกในโปรเจค พร้อมทั้งแสดง Performance ของสมาชิกแต่ละคน (หมายเลข 5) เพื่อให้ผู้ใช้งานดูข้อมูลดังกล่าวประกอบการตัดสินใจสร้างโปรเจค



ภาพที่ 5.5 หน้าจอแสดงรายงานของระบบ



ภาพที่ 5.6 หน้าจอแสดงรายงานของระบบ

จากภาพที่ 5.5 และภาพที่ 5.6 คือหน้าจอแสดงรายงานของระบบ โดยจะแสดงกราฟประสิทธิภาพของพนักงานอุปกรณ์ในรูปของกราฟแท่ง (หมายเลข 1) และแสดงผล Burndown Chart ของแต่ละโปรเจค (หมายเลข 2) โดยกราฟจะบอกว่า ในแต่ละรอบการทำงาน (Sprint Round) นั้นมีการทำงานสำเร็จเท่าใด และเมื่อถูกราฟ จะได้ประเมินได้ว่าแนวโน้มของโปรเจคแต่ละชิ้นเป็นอย่างไร

Team Management		
Thanabodee Krathu	yenooooxuy@gmail.com	Account holder
test_a last_a	test_a@email.com	Project Manager
test_b last_b	test_b@email.com	Employee
test_c last_c	test_c@email.com	Employee
test_d last_d	test_d@email.com	Project Manager
test_e last_e	test_e@email.com	Design
test_f last_f	test_f@email.com	Employee
test_g last_g	test_g@email.com	Employee

ภาพที่ 5.7 หน้าจอจัดการข้อมูลสมาชิก

จากภาพที่ 5.7 คือหน้าจอสำหรับจัดการข้อมูลสมาชิก โดยจะแสดงข้อมูลรายชื่อ อีเมล์ ตำแหน่งของสมาชิก สามารถกดลบสมาชิกออกจากระบบได้ (หมายเลข 1) และสามารถกดปุ่มเพื่อเชิญ สมาชิกใหม่เข้าใช้งานระบบได้ (หมายเลข 2)

Setting	
Company Name	<input type="text" value="Banana Coding"/>
Sprint Round	<input type="text" value="2"/> Weeks
<input type="button" value="Save"/> <input type="button" value="Cancel"/>	

ภาพที่ 5.8 หน้าจอตั้งค่าข้อมูลบริษัท

จากภาพที่ 5.6 คือหน้าจอตั้งค่าข้อมูลบริษัท โดยจะมีการแสดงข้อมูลของบริษัทต่าง ๆ เช่น ข้อมูลชื่อบริษัท เวลาการตัดรอบการนับเวลาการทำงานของระบบ เป็นต้น

บทที่ 6

การทดสอบโปรแกรม

การทดสอบโปรแกรมเป็นขั้นตอนที่สำคัญ ซึ่งการทดสอบระบบจะทำให้รู้ว่า การทำงานถูกต้องตามความต้องการ หรือเกิดข้อผิดพลาด และจะสามารถติดตามและแก้ไขก่อนที่จะนำระบบไปใช้งานจริง หากการดำเนินงานการพัฒนาโปรแกรม ระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ ผู้พัฒนาได้ทดสอบระบบตั้งแต่ล่างโดยมีรายละเอียดดังนี้

6.1 การทดสอบระบบส่วนจัดการข้อมูลโครงการ

การทดสอบระบบส่วนจัดการข้อมูลโครงการ แบ่งเป็น 3 ส่วน ดังต่อไปนี้

6.1.1 การทดสอบระบบการจัดการข้อมูลโครงการ

การทดสอบระบบการจัดการข้อมูลโครงการ แบ่งการทดสอบออกเป็น 3 ส่วนย่อยคือ

- 1) การทดสอบการสร้างโครงการ แสดงดังตารางที่ 6.1

ตารางที่ 6.1 การทดสอบการสร้างโครงการ

ข้อกรณฑ์ทดสอบ: การสร้างโครงการ		Expected Result	Test Result	
No	Test Case		Pass	No
1	กรอกข้อมูลโครงการและพีเจอร์ ถูกต้อง - กรอกชื่อโครงการ - กรอกคำอธิบายโครงการ - เลือกสมาชิกในโครงการอย่างน้อย <ol style="list-style-type: none">1 คน - สร้างพีเจอร์ 1 พีเจอร์	สามารถกดยืนยันได้	✓	

ตารางที่ 6.1 การทดสอบการสร้างโปรเจค (ต่อ)

No	Test Case	Expected Result	Test Result	
			Pass	No
2	กรอกข้อมูลเพื่อสร้างฟีเจอร์ถูกต้อง - กรอกชื่อฟีเจอร์ - กรอกขนาดของฟีเจอร์ - กรอกระยะเวลาที่คาดการณ์ของโปรเจค - กดสร้างฟีเจอร์	สามารถกดสร้างฟีเจอร์ได้	✓	
3	กรอกข้อมูลโปรเจคไม่ถูกต้อง - ไม่กรอกชื่อโปรเจค - ไม่กรอกคำอธิบายโปรเจค	ไม่สามารถกดยืนยันได้	✓	
4	ไม่สร้างฟีเจอร์ย่อย	ไม่สามารถกดยืนยันได้	✓	

2) การทดสอบการแก้ไขโปรเจค แสดงดังตารางที่ 6.2

ตารางที่ 6.2 การทดสอบการแก้ไขโปรเจค

ชื่อกรณีทดสอบ: การแก้ไขโปรเจค

เงื่อนไข: เข้าสู่ระบบโดยเป็น Account Holder หรือ Project Manager

ขั้นตอน:

- ไปที่หน้า Project Management
- เลือกโปรเจคที่ต้องการแก้ไข
- กด Edit Project เพื่อแก้ไขโปรเจค
- แก้ไขข้อมูลโปรเจค หรือ ฟีเจอร์
- กดยืนยัน

No	Test Case	Expected Result	Test Result	
			Pass	No
1	แก้ไขแล้วข้อมูลโปรเจคถูกต้อง - กรอกชื่อโปรเจค - กรอกคำอธิบายโปรเจค - เลือกสมาชิกในโปรเจคอย่างน้อย 1 คน	สามารถกดยืนยันได้	✓	

ตารางที่ 6.2 การทดสอบการแก้ไขโปรเจค (ต่อ)

No	Test Case	Expected Result	Test Result	
			Pass	No
2	แก้ไขแล้วข้อมูลโปรเจคไม่ถูกต้อง - ไม่กรอกชื่อโปรเจค - ไม่กรอกคำอธิบายโปรเจค - ไม่เลือกสมาชิกในโปรเจค	ไม่สามารถกดยืนยันได้	✓	
3	แก้ไขแล้วข้อมูลฟีเจอร์ถูกต้อง - กรอกชื่อฟีเจอร์ - กรอกขนาดของฟีเจอร์ - กรอกระยะเวลาที่คาดการณ์ของโปรเจค	สามารถกดยืนยันได้	✓	
4	แก้ไขแล้วข้อมูลฟีเจอร์ถูกต้อง - ไม่กรอกชื่อฟีเจอร์ - ไม่กรอกขนาดของฟีเจอร์ - ไม่กรอกระยะเวลาที่คาดการณ์ของโปรเจค	ไม่สามารถกดยืนยันได้	✓	

3) การทดสอบการลบโปรเจค แสดงดังตารางที่ 6.3

ตารางที่ 6.3 การทดสอบการลบโปรเจค

ชื่อกรณีทดสอบ: การลบโปรเจค

เงื่อนไข: เข้าสู่ระบบโดยเป็น Account Holder หรือ Project Manager

ขั้นตอน:

1. ไปที่หน้า Project Management
2. เลือกโปรเจคที่ต้องการแก้ไข
3. กด Delete Project เพื่อลบโปรเจค
4. กดยืนยัน

No	Test Case	Expected Result	Test Result	
			Pass	No
1	กดยืนยันการลบ	ข้อมูลถูกลบ	✓	
2	กดยกเลิกการลบ	ข้อมูลไม่ถูกลบ	✓	

6.1.2 การทดสอบการแสดงรายงานของระบบ

การทดสอบการแสดงรายงานของระบบ แสดงดังตารางที่ 6.4

ตารางที่ 6.4 การทดสอบการแสดงรายงานของระบบ

No	Test Case	Expected Result	Test Result	
			Pass	No
1	แสดงกราฟ Regression ของบริษัท	แสดงกราฟ Linear Regression ของบริษัทได้	✓	
2	แสดงรายชื่อของสมาชิกบริษัท	แสดงรายชื่อของสมาชิก พร้อมทั้งแสดงค่า Performance ของสมาชิกได้	✓	
3	แสดง Burndown Chart ของโปรเจค	แสดงกราฟ Burndown Chart ของโปรเจคได้	✓	

6.1.3 การทดสอบการตั้งค่าของระบบ

การทดสอบการตั้งค่าของระบบ แสดงดังตารางที่ 6.5

ตารางที่ 6.5 การทดสอบการตั้งค่าของระบบ

No	Test Case	Expected Result	Test Result	
			Pass	No
1	แก้ไขและล้วนข้อมูลบริษัทถูกต้อง - กรอกชื่อบริษัท - เลือกวิธีการในการคาดการณ์	สามารถดึงข้อมูลได้	✓	
2	แก้ไขและล้วนข้อมูลบริษัทไม่ถูกต้อง - ไม่กรอกชื่อบริษัท	ไม่สามารถดึงข้อมูลได้	✓	

6.2 การทดสอบระบบคาดการณ์ระยะเวลาของโครงการ

ในระบบจัดการข้อมูลและการคาดการณ์ระยะเวลาดำเนินงานของโครงการนั้น มีการเก็บบันทึกข้อมูลการนับเวลาของโปรเจคต่าง ๆ โดยมีตัวอย่างข้อมูลดังตารางที่ 6.6

ตารางที่ 6.6 ตัวอย่างข้อมูลบันทึกการนับเวลาของแต่ละโปรเจคในระบบ

Project Name	Feature Name	User Name	Time (Hours)	Total (Hours)
Project 3	Feature 3.1	test_b	8.00	550.34
	Feature 3.2	test_c	55.00	
	Feature 3.3	test_f	293.67	
		test_e	193.67	
Project 5	Feature 5.1	test_a	40.00	906.33
	Feature 5.2	test_f	75.00	
	Feature 5.3	Thanabodee	382.00	
		test_f	409.33	
Project 1	Feature 1.1	Thanabodee	97.5	1044.79
	Feature 1.2	test_a	26.17	
	Feature 1.3	test_b	61.67	
	Feature 1.4	test_c	640	
	Feature 1.5	Thanabodee	45	
		test_9	30	
		test_6	81.25	
		test_c	63.20	
Project 8	Feature 8.1	test_b	90	267.33
	Feature 8.2	test_e	60	
	Feature 8.3	test_d	28	
		test_e	38	
		test_f	22	
		test_b	29.33	

ตารางที่ 6.6 ตัวอย่างข้อมูลบันทึกการนับเวลาของแต่ละโปรเจคในระบบ (ต่อ)

Project Name	Feature Name	User Name	Time (Hours)	Total (Hours)
Project 10	Feature 10.1	test_a	15	359.58
		test_b	19	
	Feature 10.2	Thanabodee	86.17	
		test_a	90	
	Feature 10.3	test_d	79.83	
		test_b	69.58	

เมื่อมีข้อมูลการบันทึกการนับเวลาภายในระบบแล้ว ในการทดสอบระบบคาดการณ์ระยะเวลาของโครงการ จะเริ่มต้นจากการสร้างโปรเจคใหม่สำหรับทดสอบระบบขึ้นมา 1 โปรเจค ดังภาพที่ 6.1

ภาพที่ 6.1 การสร้างโปรเจคเพื่อทดสอบระบบ
โดยในการทดสอบนี้จะใช้ข้อมูลการสร้างโปรเจคสำหรับทดสอบระบบ แสดงดังตารางที่ 6.7

ตารางที่ 6.7 ข้อมูลการสร้างโปรเจคสำหรับทดสอบระบบ

ชื่อโปรเจค: Project Black Hat 1.0

รายละเอียด: Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s

เลือกสมาชิก: yenoooxuy@gmail.com, test_b@email.com, test_c@email.com, test_d@email.com

No	Feature Name	Story Point	Estimate Hours
1	Feature black one	4	70
2	Feature black two	2	20
3	Feature black three	6	150
4	Feature black four	9	650

โดยในแต่ละฟีเจอร์ จะมีสมาชิกของแต่ละฟีเจอร์ ดังตารางที่ 6.8

ตารางที่ 6.8 ข้อมูลสมาชิกในฟีเจอร์ทดสอบระบบ

Feature Name	Member Name			
	Thanabodee	test_b	test_c	test_d
Feature black one	✓			
Feature black two		✓		
Feature black three			✓	
Feature black four				✓

Original Time: 890 Hours.	Regression: 854.56 Hours.	Regression * Performance: 868.08 Hours.																
<p>Project Black Hat 1.0</p> <table border="1"> <tr> <td>Feature: Feature black one (Story point: 4)</td> <td>Feature Name Feature black one</td> <td>Story Point 4</td> <td>Hours 70.0</td> </tr> <tr> <td>Feature: Feature black two (Story point: 2)</td> <td>Select Member (ใช้เวลาต้นอย่างมากจากเดิม ใช้เวลามากขึ้นจากเดิม)</td> <td colspan="2"> <input checked="" type="checkbox"/> Thanabodee Krathu (↓-17.223 %) <input type="checkbox"/> test_b last_b (↓-17.914 %) <input type="checkbox"/> test_c last_c (↑ 8.934 %) <input type="checkbox"/> test_d last_d (↓-2.196 %) </td> </tr> <tr> <td>Feature: Feature black three (Story point: 6)</td> <td colspan="3"></td> </tr> <tr> <td>Feature: Feature black four (Story point: 9)</td> <td colspan="3"></td> </tr> </table>			Feature: Feature black one (Story point: 4)	Feature Name Feature black one	Story Point 4	Hours 70.0	Feature: Feature black two (Story point: 2)	Select Member (ใช้เวลาต้นอย่างมากจากเดิม ใช้เวลามากขึ้นจากเดิม)	<input checked="" type="checkbox"/> Thanabodee Krathu (↓-17.223 %) <input type="checkbox"/> test_b last_b (↓-17.914 %) <input type="checkbox"/> test_c last_c (↑ 8.934 %) <input type="checkbox"/> test_d last_d (↓-2.196 %)		Feature: Feature black three (Story point: 6)				Feature: Feature black four (Story point: 9)			
Feature: Feature black one (Story point: 4)	Feature Name Feature black one	Story Point 4	Hours 70.0															
Feature: Feature black two (Story point: 2)	Select Member (ใช้เวลาต้นอย่างมากจากเดิม ใช้เวลามากขึ้นจากเดิม)	<input checked="" type="checkbox"/> Thanabodee Krathu (↓-17.223 %) <input type="checkbox"/> test_b last_b (↓-17.914 %) <input type="checkbox"/> test_c last_c (↑ 8.934 %) <input type="checkbox"/> test_d last_d (↓-2.196 %)																
Feature: Feature black three (Story point: 6)																		
Feature: Feature black four (Story point: 9)																		
<p><input type="button" value="Cancel"/> <input type="button" value="Save"/></p>																		

ภาพที่ 6.2 เลือกสมาชิกในแต่ละฟีเจอร์ของการทดสอบระบบ

จากภาพที่ 6.2 จะเห็นว่า เมื่อมีการสร้างฟีเจอร์ขึ้นมาแล้ว ระบบจะมีคำนวนเวลาของทุกฟีเจอร์อุปกรณ์เป็นเวลารวมของโปรเจค โดยข้อมูลเวลานั้นจะแบ่งเป็น 3 ค่าเวลา คือ

- Total Original Time คือ เวลารวมของโปรเจค ซึ่งเกิดจากผลรวมเวลาของแต่ละฟีเจอร์ที่ผู้ใช้กรอกตอนสร้างฟีเจอร์

- Regression คือ ระยะเวลาที่คาดการณ์ด้วยการใช้สมการ Simple Linear Regression โดยหาสมการมาจากข้อมูลฟีเจอร์ของบริษัท

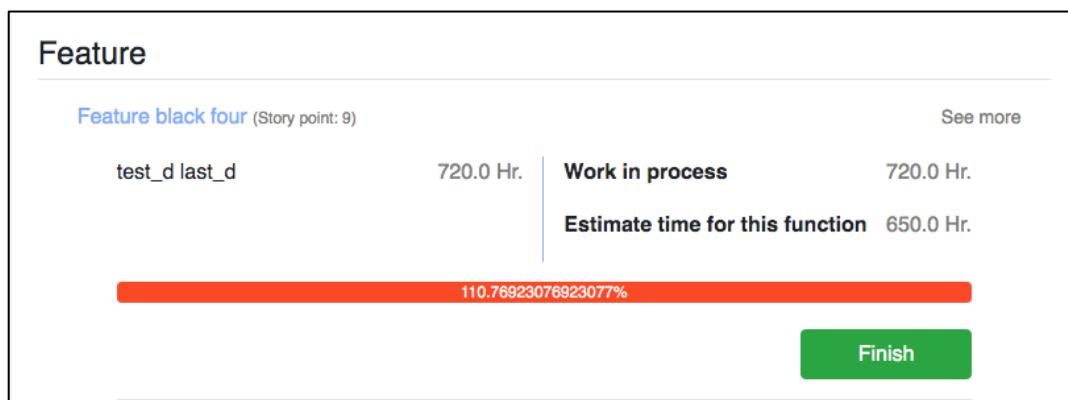
- Regression * Performance คือ ระยะเวลาที่คาดการณ์ด้วยการใช้สมการ Simple Linear Regression ร่วมกับการใช้ค่า Performance ของสมาชิกของฟีเจอร์ต่าง ๆ เพื่อให้การคาดการณ์เวลา มีการเปลี่ยนแปลง ตามการเลือกสมาชิกของแต่ละฟีเจอร์

เมื่อสร้างโปรเจคเสร็จแล้ว ให้นำเข้าข้อมูลการทำงานของสมาชิกแต่ละคนในโปรเจค เพื่อให้สามารถกดเสริจสิ้นฟีเจอร์แต่ละชิ้นได้ โดยข้อมูลที่ถูกนำเข้า มีค่าดังตารางที่ 6.9

ตารางที่ 6.9 ข้อมูลการทำงานของสมาชิกในโปรเจค Project Black Hat 1.0

Feature Name	Member Name	Use Time
Feature black one	Thanabodee	80 Hr.
Feature black two	test_b	25 Hr.
Feature black three	test_c	130 Hr.
Feature black four	test_d	720 Hr.
ผลรวมเวลาที่ใช้ของโปรเจค		955 Hr.

เมื่อเพิ่มข้อมูลการทำงานของสมาชิกในโปรเจคแล้ว ให้กดเสริจสิ้นฟีเจอร์ ดังปุ่มในภาพที่ 6.3 ในทุกฟีเจอร์ของโปรเจค Project Black Hat 1.0 และกดสิ้นสุดโปรเจค จากนั้นจึงทดสอบระบบอีกครั้ง โดยการสร้างโปรเจคขึ้นมาใหม่ ชื่อโปรเจคว่า “Project Black Hat 2.0” ซึ่งข้อมูลของโปรเจค จะใช้ข้อมูลชุดเดียวกันจากตารางที่ 6.7 ถึงตารางที่ 6.8 จะได้ผลลัพธ์ดังภาพที่ 6.4



ภาพที่ 6.3 ข้อมูลภายในฟีเจอร์ และปุ่มเสริจสิ้นของแต่ฟีเจอร์ในโปรเจค

Original Time:	890	Hours.	Regression:	874.82	Hours.	Regression * Performance:	902.13	Hours.
Project Black Hat 2.0				Feature Name <input type="text" value="Feature black one"/> Story Point <input type="text" value="4"/> Hours <input type="text" value="70.0"/> Select Member (+ เวลาส่วนน้อยจากเดิม - เวลาส่วนมากจากเดิม)				
Feature: Feature black one (Story point: 4) Feature: Feature black two (Story point: 2) Feature: Feature black three (Story point: 6) Feature: Feature black four (Story point: 9)				<input checked="" type="checkbox"/> Thanabodee Krathu (+2.938 %) <input type="checkbox"/> test_b last_b (+7.086 %) <input type="checkbox"/> test_c last_c (+4.399 %) <input type="checkbox"/> test_d last_d (+8.573 %)				
				<input type="button" value="Cancel"/> <input type="button" value="Save"/>				

ภาพที่ 6.4 ผลลัพธ์การสร้างโปรเจคด้วยข้อมูลชุดเดิม

จากภาพที่ 6.2 และภาพที่ 6.4 เมื่อดูที่ผลลัพธ์ที่เกิดขึ้นจากการคาดการณ์ระยะเวลาของแต่ละฟีเจอร์แล้ว สามารถสรุปได้ดังตารางที่ 6.10

ตารางที่ 6.10 ตารางเปรียบเทียบข้อมูลที่ได้จากการคาดการณ์ระยะเวลา

ชื่อโปรเจค	Project Black Hat 1.0	เวลาที่ใช้จริงของ Project Black Hat 1.0	Project Black Hat 2.0
คำนวณด้วยวิธี			
Total Original Time	890.00 Hr.		890.00 Hr.
Regression	854.56 Hr.	955.00 Hr.	874.82 Hr.
Regression * Performance	868.08 Hr.		902.13 Hr.

จากตารางที่ 6.10 พบร่วมกันว่า ค่าที่ได้จากการสร้างโปรเจค Project Black Hat 2.0 นั้น ค่าวремาของการคาดการณ์ระยะเวลา มีค่าเพิ่มขึ้นจากการสร้างโปรเจคในครั้งก่อนหน้านี้ เนื่องจาก ข้อมูลการนับเวลาของสมาชิกใหม่ที่มีโปรเจค Project Black Hat 1.0 ที่ถูกเพิ่มเข้ามาในระบบ มีการใช้เวลาการทำงานรวมทั้งสิ้น 955.00 ชั่วโมง ซึ่งใช้เวลามากกว่าเวลาที่ระบบคาดการณ์ไว้ทุกวิธี จึงทำให้ค่าวремาที่คาดการณ์สำหรับโปรเจค Project Black Hat 2.0 มีค่าเพิ่มขึ้น

โดยระบบคาดการณ์ระยะเวลา นั้น จะมีการปรับปรุงสมการ Simple Linear Regression และปรับปรุงค่า Performance ของสมาชิกทุกครั้งที่มีการกดเสร็จสิ้นฟีเจอร์ จากการทดลองของโปรเจค Project Black Hat 1.0 เมื่อมีการกดเสร็จสิ้นในแต่ละฟีเจอร์ของโปรเจค Project Black Hat 1.0 จึงทำให้ค่าของสมการ Simple Linear Regression และค่า Performance ของสมาชิกมีการเปลี่ยนแปลงไปตามเวลาที่ใช้งานจริง ซึ่งสังเกตผลลัพธ์ของการเปลี่ยนแปลงนี้ได้ จากรายงานที่ 6.2 และ ภาพที่ 6.4 ที่ด้านหลังซึ่งของสมาชิกแต่ละคน ค่าเบอร์เซ็นต์ Performance ของสมาชิกแต่ละคนมีการเปลี่ยนแปลงไป และสังเกตจากเวลาที่ระบบคาดการณ์ในตารางที่ 6.10

บทที่ 7

สรุปผลการดำเนินการ

จากการพัฒนาระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ ซึ่งพัฒนาขึ้นเพื่อเพิ่มความสะดวกในการวางแผนการดำเนินงานของโครงการให้แก่ผู้จัดการโครงการ ให้สามารถคาดการณ์ระยะเวลาที่เหมาะสมสำหรับการทำงานได้ง่ายและสะดวกขึ้น ซึ่งในบทนี้จะกล่าวถึงผลการพัฒนาระบบ ปัญหาที่พบ และแนวทางการพัฒนาระบบตั้งกล่าวต่อไปในอนาคต โดยมีรายละเอียดดังต่อไปนี้

7.1 สรุปผลการดำเนินงาน

จากการค้นคว้าอิสระในครั้งนี้ ได้พัฒนาระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ ซึ่งถูกพัฒนาด้วยรูปแบบเรลส์เฟรมเวิร์ค (Ruby on Rails Framework) และจัดเก็บข้อมูลของระบบด้วยระบบจัดการฐานข้อมูลโพสต์เกรสคิวเอล (PostgreSQL)

จากการพัฒนาระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ ผู้จัดการโครงการ สามารถใช้ระบบเพื่อจัดการข้อมูลโครงการ รวมทั้งสามารถใช้ระบบเพื่อคาดการณ์ระยะเวลาที่เหมาะสมของโครงการได้ ซึ่งช่วยให้ผู้จัดการโครงการ สามารถวางแผนการดำเนินงานได้สะดวกยิ่งขึ้น

7.2 ข้อจำกัดของระบบ

- 1) ไม่รองรับการใช้งานเว็บไซต์แบบ Responsive Website
- 2) ไม่รองรับการซิงค์ข้อมูลการนับเวลา จากโปรแกรมนับเวลาภายนอก ทำให้ต้องทำการใส่ข้อมูลการนับเวลาเข้าไปในหน้าเว็บเท่านั้น
- 3) ไม่สามารถส่งอกรายงาน หรือแชร์รายงานของระบบได้
- 4) ไม่สามารถเชิญสมาชิกใหม่ด้วยการส่งอีเมล์เชิญผู้ใช้งานได้

7.3 แนวทางในการพัฒนาต่อ

- 1) ปรับปรุงระบบ ให้รองรับการใช้งานเว็บไซต์แบบ Responsive Website
- 2) เพิ่มการซิงค์ข้อมูลการนับเวลา และข้อมูลโปรเจก จากโปรแกรมนับเวลาภายนอก
- 3) เพิ่มฟังก์ชัน Export File รายงานของระบบ
- 4) ปรับปรุงระบบเพิ่มสมาชิก ให้สามารถเชิญสมาชิกใหม่ด้วยการส่งอีเมล์เชิญผู้ใช้งานได้
- 5) อาจใช้วิธีอื่นในการคาดการณ์ระยะเวลาของโครงการได้ เช่น การนำลักษณะของข้อมูลเวลาที่ใช้ทำงาน (Hours) และขนาดของฟีเจอร์ (Story Point) ที่นำมาคำนวณกับ Simple Linear

Regression นั้นมีการเติบโตของข้อมูลที่ไม่สอดคล้องกับสมการเส้นตรง โดยข้อมูล Hours นั้นมีการเติบโตอย่างก้าวกระโดดในทุก ๆ Story Point ซึ่งมีลักษณะการเพิ่มขึ้นของข้อมูลคล้ายกับรูปแบบของ Exponential Function

เอกสารอ้างอิง

- [1] ศาสตราจารย์ ดร. จรัญ จันทลักษณ์. สติ๊ติ: การวิเคราะห์และการวางแผนงานวิจัย. สำนักพิมพ์มหาวิทยาลัย เกษตรศาสตร์ ปี พ.ศ. 2549 หน้า 71-81 ISBN 978-974-53-7820-9
- [2] ผู้ช่วยศาสตราจารย์สุพจน์ โกสิยะจินดา. การบริหารโครงการในระบบงานไอที. บริษัทวิทยพัฒน์ จำกัด ปี พ.ศ. 2550 หน้า 131-200 ISBN 978-974-11-0273-0
- [3] ผู้ช่วยศาสตราจารย์สุธรรม นาบศักดิ์. สติ๊ติเศรษฐศาสตร์. มหาวิทยาลัยทักษิณ. แหล่งที่มา : http://www2.tsu.ac.th/ecba/UserFiles/0703221_document.pdf. สืบค้นเมื่อ 25 ตุลาคม, 2561
- [4] CORALINE CO. LTD. สติ๊ติเบื้องต้นง่ายๆ ที่จะทำให้คุณเข้าใจการวิเคราะห์มากขึ้น (ตอนที่ 2). Medium. แหล่งที่มา : https://medium.com/@info_46914/สติ๊ติเบื้องต้นง่ายๆ-ที่จะทำให้คุณเข้าใจการวิเคราะห์มากขึ้น-ตอนที่-2-1f94b6664ede. สืบค้นเมื่อ 25 ตุลาคม, 2561
- [5] Thanyavuth Akarasomcheep. Scrum คืออะไร เริ่มใช้งานอย่างไร. แหล่งที่มา: <https://medium.com/fastwork-engineering/scrum-คืออะไร-เริ่มใช้งานอย่างไร-2483e761a47e>. สืบค้นเมื่อ 4 เมษายน, 2562
- [6] Thanyavuth Akarasomcheep. Agile คืออะไร เริ่มใช้งานอย่างไร. แหล่งที่มา: <https://medium.com/fastwork-engineering/agile-คืออะไร-เริ่มใช้งานอย่างไร-ab749306d96e>. สืบค้นเมื่อ 4 เมษายน, 2562
- [7] (ไม่ระบุผู้แต่ง). Scrum Burndown Chart. แหล่งที่มา: https://www.scrum-institute.org/Burndown_Chart.php. สืบค้นเมื่อ 4 เมษายน, 2562
- [8] Be Tiranasak. การ Estimate Size จาก Story Point. แหล่งที่มา: <https://medium.com/@betiranasak/การ-estimate-size-จาก-story-point-ba1ac2351746>. สืบค้นเมื่อ 4 เมษายน, 2562
- [9] Piyrot. Stop Starting, Start Finishing!!!!!! . แหล่งที่มา: <https://medium.com/agile-development-in-thai/stop-starting-start-finishing-83b4f99de57e>. สืบค้นเมื่อ 4 เมษายน, 2562
- [10] Siwa Khongsuphap. สิ่งที่คุณควรรู้เกี่ยวกับ HTML5 (ตอนที่ 1). Medium. แหล่งที่มา : <https://medium.com/open-source-technology/html5-basic-1-b27b0f6464c1>. สืบค้นเมื่อ 21 กันยายน, 2561
- [11] Chatchapong Mekgamol. Sass คืออะไร? เขียนยังไง? ใช้ยังไง?. Medium. แหล่งที่มา : <https://medium.com/imkrz/sass-howto-3856a33e15de>. สืบค้นเมื่อ 21 กันยายน, 2561

- [12] (ไม่ระบุผู้แต่ง). แนะนำภาษา Ruby. แหล่งที่มา : <http://marcuscode.com/lang/ruby/introduction>. สืบค้นเมื่อ 21 กันยายน, 2561
- [13] (ไม่ระบุผู้แต่ง). แนะนำ Ruby On Rails. แหล่งที่มา: <https://swiftlet.co.th/learn/intro-to-rails/>. สืบค้นเมื่อ 21 กันยายน, 2561
- [14] (ไม่ระบุผู้แต่ง). JavaScript คืออะไร. แหล่งที่มา: <https://www.mindphp.com/คูมือ/73-คืออะไร/2187-java-javascript-คืออะไร.html>. สืบค้นเมื่อ 21 กันยายน, 2561
- [15] Sublime HQ Pty Ltd. Sublime Text 3. แหล่งที่มา : <https://www.sublimetext.com/3>. สืบค้นเมื่อ 17 พฤษภาคม, 2561

ภาคผนวก ก

คู่มือการใช้งานระบบ

ระบบจัดการข้อมูลและการณ์ระยะเวลาดำเนินงานของโครงการ มีคู่มือการใช้งานระบบดังนี้

การสร้างบัญชีเจ้าของบริษัท และการเข้าสู่ระบบ

ผู้ใช้ต้องสมัครบัญชีหลักของบริษัท เพื่อสร้างข้อมูลของบริษัทขึ้นมา และเพื่อให้สามารถเขียน
สมาชิกคนอื่นเข้ามาบันทึกข้อมูลภายใต้บัญชีของบริษัทได้ โดยแสดงการสมัครดังภาพที่ ก.1

The form is titled 'Create New Company'. It has two main sections. The first section, 'Company Name', contains a text input field and a dropdown menu for 'Sprint Round' set to '1 Weeks'. The second section, 'Set Account Holder', contains fields for 'First Name' and 'Last Name' (each with its own input field), 'Email' (one input field), 'Password' and 'Confirm Password' (each with its own input field). At the bottom are 'Save' and 'Cancel' buttons.

ภาพที่ ก.1 หน้าจอสร้างบัญชีเจ้าของบริษัท

โดยผู้ใช้จะต้องกรอกชื่อบริษัท และเลือกเวลาการติดรอบการนับเวลาการทำงาน (Sprint Round) และกรอกข้อมูลส่วนตัวของผู้ใช้ จากนั้นให้กด Save และล็อกอินเข้าสู่ระบบด้วยบัญชีที่ กรอกไว้ โดยการล็อกอินจะแสดงดังภาพที่ ก.2

The page is titled 'Log in'. It has two input fields: 'Email' and 'Password', both with placeholder text. Below them is a 'Log in' button. At the bottom is a link 'Sign Up New Company'.

ภาพที่ ก.2 หน้าจอเข้าสู่ระบบ

ในการเข้าสู่ระบบจะแบ่งออกเป็น 3 ระดับคือ เจ้าของบัญชีบริษัท (Account Holder) ผู้จัดการโครงการ (Project Manager) และพนักงาน (Employee) เมื่อเข้าสู่ระบบสำเร็จ แลบเมนูของระบบจะแสดงเมนูที่ผู้ใช้แต่ละระดับสามารถเข้าถึงได้ แสดงดังภาพที่ ก.3 ถึงภาพที่ ก.4

Chiang Main University Project Management Report Team Management Setting

ภาพที่ ก.3 แลบเมนูของ Account Holder

Chiang Main University Project Management Report

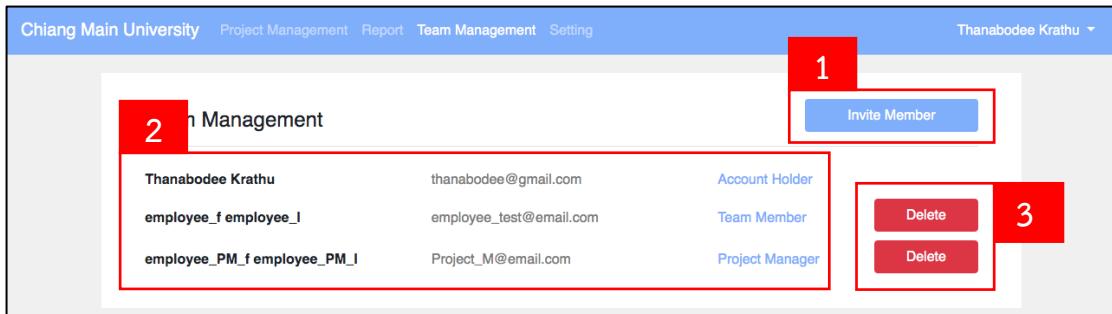
ภาพที่ ก.4 แลบเมนูของ Project Manager และ Employee

โดยในภาพที่ ก.4 นั้น เมนูของหัว Project Manager และ Employee จะมีเหมือนกัน แต่จะแตกต่างกันที่ Project Manager เมื่อเข้าสู่หน้า Project Management แล้วจะสามารถทำการเพิ่ม/ลบ/แก้ไข ข้อมูลต่าง ๆ ของโครงการได้ ในขณะที่ Employee นั้นจะทำได้เพียงแค่เข้าไปดูข้อมูลที่ว่าไปของโครงการเท่านั้น

การใช้งานในแต่ละฟังก์ชัน

1. หน้าจัดการสมาชิก (Team Management)

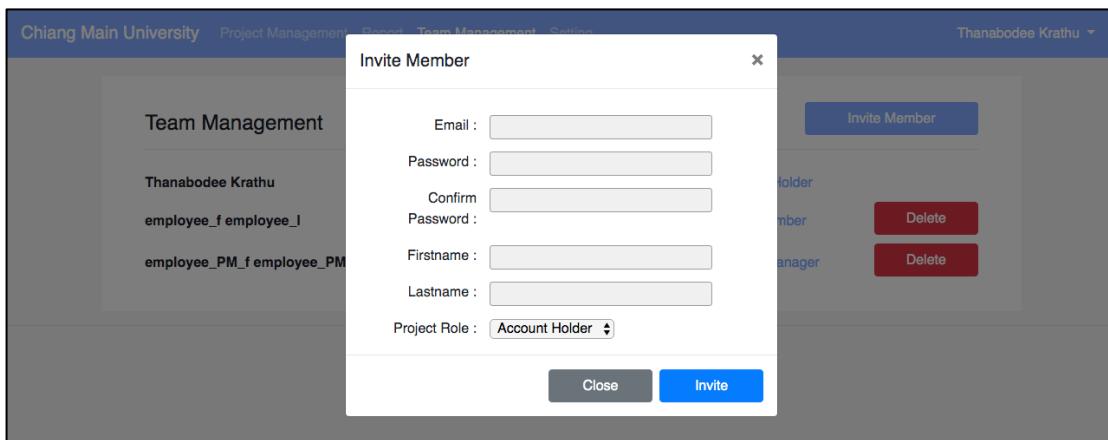
หน้าจัดการสมาชิกในบริษัท แสดงดังภาพที่ ก.5



ภาพที่ ก.5 หน้าจัดการสมาชิก

จากภาพที่ ก.5 หน้าจัดการสมาชิก เป็นหน้าสำหรับเพิ่มสมาชิก หรือเพิ่มพนักงานเข้าสู่การใช้งานภายใต้ชื่อบริษัทของบัญชี Account Holder โดยมีส่วนประกอบของหน้าจอดังต่อไปนี้

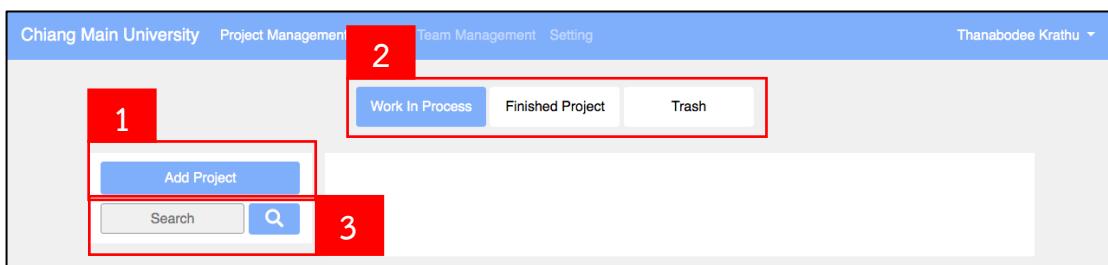
- 1) ปุ่มเพิ่มสมาชิกในบริษัท โดยเมื่อกดแล้วจะมีช่องให้กรอกข้อมูลดังภาพที่ ก.6
- 2) ส่วนแสดงผลสมาชิกในบริษัท โดยสมาชิกที่ถูกเพิ่มแล้วจะแสดงผลอยู่ในส่วนนี้
- 3) ส่วนจัดการสมาชิก โดยสามารถลบสมาชิกของบริษัทออกจากระบบได้ (ไม่สามารถลบ บัญชีของ Account Holder ได้)



ภาพที่ ก.6 หน้าจอกอรากข้อมูลเพื่อเพิ่มสมาชิกเข้าสู่ระบบ

2. หน้าจอจัดการข้อมูลโปรเจค (Project Management)

หน้าจอจัดการข้อมูลโปรเจค แสดงดังภาพที่ ก.7



ภาพที่ ก.7 หน้าจอจัดการข้อมูลโปรเจค

จากภาพที่ ก.7 หน้าจอจัดการข้อมูลโปรเจค มีส่วนประกอบดังต่อไปนี้

- 1) ปุ่มเพิ่มโปรเจค เมื่อกดปุ่มจะไปยังหน้าสร้างโปรเจคและไฟล์เอกสารดังภาพที่ ก.8
- 2) ปุ่มแสดงโปรเจค โดยแยกประเภทของโปรเจคออกเป็น โปรเจคที่กำลังดำเนินการ (Work In Process) โปรเจคที่เสร็จสิ้นแล้ว (Finished Project) และโปรเจคที่ถูกลบ ซึ่งเก็บอยู่ในถังขยะ (Trash)
- 3) ปุ่มค้นหาโปรเจค โดยสามารถกรอกชื่อโปรเจค แล้วกดเพื่อค้นหาได้

The screenshot shows the 'Create Project' and 'Create Feature' forms. The 'Select Team Member' section and the 'Add Feature' button are highlighted with red boxes and numbered 2 and 4 respectively. The 'List of Feature' section is highlighted with a red box and numbered 5. The 'Project Name' and 'Feature Name' fields are also highlighted with red boxes and numbered 1 and 3 respectively.

ภาพที่ ก.8 หน้าจอสร้างโปรเจกและฟีเจอร์

จากภาพที่ ก.8 หน้าจอสร้างโปรเจกและฟีเจอร์ จะมีส่วนประกอบดังต่อไปนี้

- 1) ส่วนกรอกข้อมูลทั่วไปของโปรเจก คือชื่อโปรเจก และคำอธิบายของโปรเจก
 - 2) ส่วนเลือกสมาชิกของโปรเจก โดยจะมีรายชื่อของสมาชิกที่อยู่ในบริษัท ซึ่งมาจาก การจัดการสมาชิกในภาพที่ ก.5
 - 3) ส่วนสร้างฟีเจอร์ของโปรเจก โดยให้กรอกข้อมูลของฟีเจอร์ คือ ชื่อฟีเจอร์ ขนาดของ ฟีเจอร์ (Story Point) จำนวนชั่วโมงที่จะใช้สำหรับทำงานฟีเจอร์นี้ (Estimate Hours) และคำอธิบาย ของฟีเจอร์
 - 4) ปุ่มสำหรับกดเพิ่มฟีเจอร์ โดยจะนำข้อมูลจากส่วนที่ 3) ไปเพิ่มเป็นฟีเจอร์ และ แสดงผลในส่วนที่ 5)
 - 5) ส่วนแสดงผลฟีเจอร์ที่ถูกสร้างขึ้นในโปรเจก
- เมื่อกรอกข้อมูลทุกอย่างครบถ้วนแล้ว จากนั้นกด Create Project เพื่อสร้างโปรเจก และเข้า สู่หน้าจัดการสมาชิกในฟีเจอร์ ซึ่งจะแสดงในภาพที่ ก.9

1 Predict Time Chart

2 Original Time: 230 Hours. Regression: 0 Hours. Regression * Performance: 0 Hours.

3 Project Black Car 1.0

- Feature: Black Car 1 (Story point: 2)
- Feature: Black Car 2 (Story point: 4)
- Feature: Black Car 3 (Story point: 6)

4 Feature Name: Black Car 2 Story Point: 4 Hours: 60.0
Select Member: Thanabodee Krathu (0.0%), employee_PM_I (0.0%), employee_f_employee_I (0.0%)
Cancel Save

5 User Performance

Employee Name	Email	Position	Performance
Thanabodee Krathu	thanabodee@gmail.com	Account Holder	100.0 %
employee_PM_I	Project_M@email.com	Project Manager	100.0 %
employee_f_employee_I	employee_test@email.com	Team Member	100.0 %

Back Save

ภาพที่ ก.9 หน้าจอสำหรับจัดการสมาชิกในฟีเจอร์

จากภาพที่ ก.9 หน้าจอสำหรับจัดการสมาชิกในฟีเจอร์ จะมีส่วนประกอบดังต่อไปนี้

1) ส่วนแสดงผล Predict Time Chart ซึ่งเป็นกราฟของสมการ Regression ของบริษัท โดยกราฟจะแสดงผลว่า หากใส่ขนาดงาน (Story Point) แล้วจะใช้เวลาในการทำงาน (Hours) เท่าใด (แต่ระบบจะยังไม่มีกราฟนี้ จนกว่าจะมีการทำงานไปสักระยะ แล่มีการกดเสร็จสิ้นของฟีเจอร์ และโปรเจค)

2) ส่วนแสดงผลการคาดการณ์ระยะเวลาของโปรเจค โดยจะมีค่า 3 ค่าดือ

- Total Original Time คือ เวลารวมของโปรเจค ซึ่งเกิดจากผลรวมเวลาของแต่ละฟีเจอร์ที่ผู้ใช้กรอกตอนสร้างฟีเจอร์

- Regression คือ ระยะเวลาที่คาดการณ์ด้วยการใช้สมการ Simple Linear Regression โดยหาสมการมาจากการข้อมูลฟีเจอร์ของบริษัท

- Regression * Performance คือ ระยะเวลาที่คาดการณ์ด้วยการใช้สมการ Simple Linear Regression ร่วมกับการใช้ค่า Performance ของสมาชิกของฟีเจอร์ต่าง ๆ เพื่อให้การคาดการณ์เวลาเมื่อมีการเปลี่ยนแปลง ตามการเลือกสมาชิกของแต่ละฟีเจอร์

ซึ่งจากค่าทั้ง 3 ค่า ผู้ใช้จะนำค่าทั้งหมดไปพิจารณา แล้วนำไปปรับค่าที่ส่วนที่ 4

3) ส่วนแสดงผลฟีเจอร์ในโปรเจค โดยสามารถกดเลือกฟีเจอร์ที่ต้องการ เพื่อจัดการสมาชิกในฟีเจอร์ โดยส่วนการจัดการสมาชิกในฟีเจอร์จะแสดงในส่วนที่ 4

4) ส่วนจัดการข้อมูลฟีเจอร์ โดยสามารถจัดการสมาชิกในฟีเจอร์ รวมไปถึงแก้ไขข้อมูล ต่าง ๆ ของฟีเจอร์ได้ โดยในส่วนของรายชื่อที่มีให้เลือกนั้น จะแสดงสมาชิกที่ถูกเลือกจากขั้นตอนการสร้างโปรเจค และด้านท้ายซึ่งจะแสดงเปอร์เซ็นต์ที่มีผลต่อการใช้เวลาการทำงานของฟีเจอร์ โดยถ้า

เป็นสีเขียว แสดงว่าหากเลือกสมาชิกคนนี้ จะทำให้เวลาร่วมในการทำงานของฟีเจอร์ลดลง แต่ถ้าเป็นสีแดงคือ เมื่อเลือกสมาชิกคนนี้แล้ว เวลาร่วมในการทำงานของฟีเจอร์จะเพิ่มขึ้น

5) ส่วนแสดงประสิทธิภาพของสมาชิกในโปรเจค โดยจะปรากฏรายชื่อของสมาชิกในโปรเจค พร้อมทั้งบอกข้อมูลประสิทธิภาพการทำงาน โดยถ้ายิ่งเปอร์เซ็นต์สูง ยิ่งหมายถึงสมาชิกสามารถทำงานได้ดี ใช้เวลาการทำงานน้อยกว่าคนที่มีประสิทธิภาพต่ำกว่า

เมื่อจัดการข้อมูลของฟีเจอร์เสร็จสิ้นแล้ว ให้กดสร้างโปรเจค โดยหลังจากกดสร้างแล้ว ข้อมูลโปรเจคจะแสดงผลดังภาพที่ ก.10

The screenshot shows the Chiang Mai University Project Management system. At the top, there's a navigation bar with 'Chiang Main University', 'Project Management', 'Report', 'Team Management', and 'Setting'. On the right, it says 'Thanabodee Krathu'. Below the navigation, there are three tabs: 'Work In Process' (selected), 'Finished Project', and 'Trash'. On the left, there's a sidebar with 'Add Project', 'Search', and a list for 'Project Black Car 1.0'. The main content area has a 'Description' section with placeholder text 'Lorem ipsum dolor sit amet.' and an 'Estimate Time' of '230.0 Hr.'. A red box highlights the top right of this section with buttons for 'Finish', 'Edit Project', and 'Delete Project'. A large red box surrounds the 'Feature' section for 'Black Car 1'. This section shows 'Story point: 2', 'Thanabodee Krathu' assigned, '0 Hr.' worked, 'Work in process', '0 Hr.' remaining, and an 'Estimate time for this function' of '20.0 Hr.'. It also includes a progress bar at 0.0% and a 'Finish' button. Below this, there are links for 'Black Car 2 (Story point: 4)' and 'Black Car 3 (Story point: 6)'. A red box highlights the 'Team Member' section at the bottom, which lists 'Thanabodee Krathu', 'Account Holder', and the email 'thanabodee@gmail.com'.

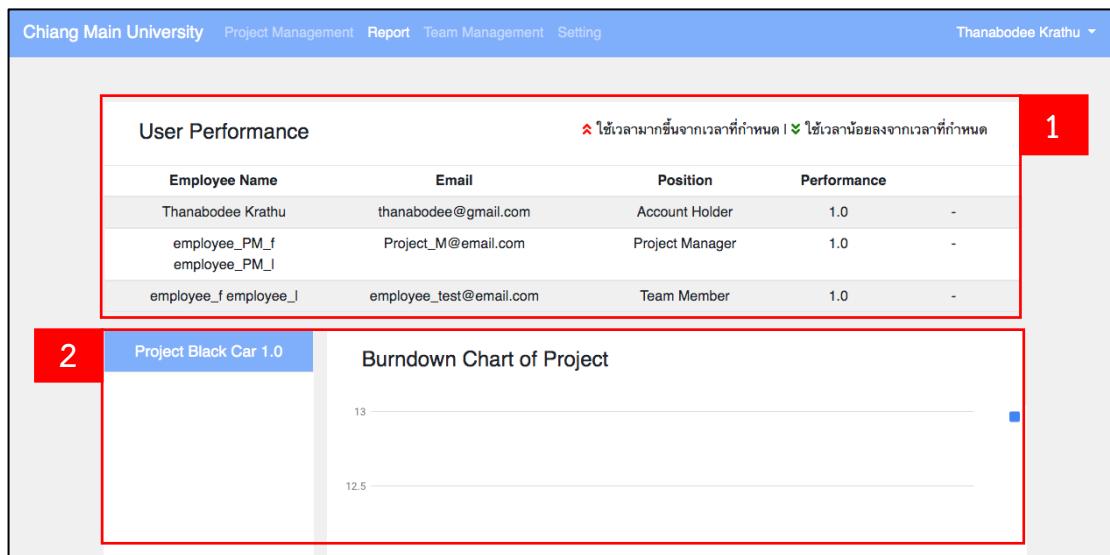
ภาพที่ ก.10 หน้าจอจัดการข้อมูลโปรเจค หลังจากการสร้างโปรเจค

จากภาพที่ ก.10 หน้าจอจัดการข้อมูลโปรเจค เมื่อมีโปรเจคแล้ว จะมีส่วนประกอบดังต่อไปนี้

- 1) ส่วนจัดการโปรเจค โดยสามารถกดปุ่มต่าง ๆ ได้ เช่น ปุ่ม Finish เพื่อกดเสร็จสิ้นโปรเจค, ปุ่ม Edit Project เพื่อแก้ไขข้อมูลโปรเจคและข้อมูลฟีเจอร์ในโปรเจค, ปุ่ม Delete Project เพื่อลบโปรเจคไปยังถังขยะ เป็นต้น
- 2) ส่วนจัดการฟีเจอร์ โดยสามารถดูข้อมูลการทำงานของสมาชิกต่าง ๆ ในฟีเจอร์ได้ อีกทั้งยังมีปุ่ม Finish ของแต่ละฟีเจอร์ให้กดเมื่อทำงานในแต่ละฟีเจอร์เสร็จสิ้นแล้ว

3. หน้าจอแสดงรายงานของระบบ

หน้าจอแสดงรายงานของระบบ แสดงดังภาพที่ ก.11



ภาพที่ ก.11 หน้าจอแสดงรายงานของระบบ

จากภาพที่ ก.11 หน้าจอแสดงรายงานของระบบ จะมีส่วนประกอบดังต่อไปนี้

- 1) ส่วนแสดงประสิทธิภาพของสมาชิกในโครงการ โดยจะปรากฏรายชื่อของสมาชิกในโครงการ พร้อมทั้งบอกข้อมูลประสิทธิภาพการทำงาน โดยถ่ายทอดเปอร์เซ็นต์สูง ยิ่งหมายถึงสมาชิกสามารถทำงานได้ดี ใช้เวลาการทำงานน้อยกว่าคนที่มีประสิทธิภาพต่ำกว่า
- 2) ส่วนแสดงผล Burndown Chart ของโครงการที่เลือก โดยเมื่อมีการทำงานของแต่ละโครงการแล้ว จะนำเวลาที่ใช้ทำงานรวมของทั้งโครงการ มาแสดงผลในรูปแบบของกราฟ ร่วมกับการแสดงผลว่า ในแต่ละรอบการนับเวลา (Sprint Round) นั้น มีการทำงานสำเร็จแล้วกี่ฟีเจอร์ ทำงานได้แล้วกี่ค่า Story Point

4. หน้าจอการตั้งค่าระบบ

หน้าจอการตั้งค่าระบบ แสดงดังภาพที่ ก.12

The screenshot shows the 'Setting' page of the system. It includes fields for Company Name (set to 'Chiang Main University') and Sprint Round (set to '2 Weeks'). At the bottom are 'Save' and 'Cancel' buttons.

ภาพที่ ก.12 หน้าจอการตั้งค่าระบบ

จากภาพที่ ก.12 หน้าจอการตั้งค่าระบบนั้น สามารถเข้ามาแก้ไขข้อมูลชื่อบริษัท และแก้ไขรอบการนับเวลา (Sprint Round) ได้

ภาคผนวก ข

คู่มือการติดตั้งระบบ

ระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ มีคู่มือการติดตั้งระบบดังนี้

ระบบปฏิบัติการที่ใช้

- ระบบปฏิบัติการแมคโออีส 10.13 (MacOS 10.13 High Sierra)

ลำดับการติดตั้ง

ในการติดตั้งเว็บไซต์ จะประกอบไปด้วย 5 ขั้นตอนดังนี้ 1) การติดตั้ง Homebrew 2) การติดตั้ง Ruby 3) การติดตั้ง Ruby On Rails Framework 4) การติดตั้ง PostgreSQL และ 5) การติดตั้งเว็บไซต์ระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ โดยมีรายละเอียดในการติดตั้งแต่ละขั้นตอนได้แก่

1) การติดตั้ง Homebrew

Homebrew จะช่วยให้สามารถติดตั้งซอฟต์แวร์แพคเกจต่าง ๆ ได้ง่ายขึ้น โดยมีคำสั่งติดตั้งง่าย ๆ และเมื่อกำลังติดตั้ง หากระบบแจ้งเตือนให้ติดตั้ง XCode CommandLine Tools ก็ให้ตอบตกลงเพื่อติดตั้งด้วย

```
$ ruby -e "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

2) การติดตั้ง Ruby

เราจะใช้ Homebrew ในการช่วยติดตั้ง rbenv เพื่อใช้ rbenv 在การติดตั้งและควบคุมเวอร์ชันของภาษา Ruby ในเครื่อง

```
$ brew install rbenv ruby-build  
  
# Add rbenv to bash so that it loads every time you open a terminal  
  
$ echo 'if which rbenv > /dev/null; then eval "$(rbenv init -)"; fi' >>  
~/.bash_profile  
  
$ source ~/.bash_profile  
  
# Install Ruby  
  
$ rbenv install 2.5.1  
  
$ rbenv global 2.5.1
```

```
$ ruby -v
```

3) ติดตั้ง Ruby On Rails Framework

```
$ gem install rails -v 5.2.2  
$ rbenv rehash  
$ rails -v  
# Rails 5.2.2
```

4) การติดตั้ง PostgreSQL

```
$ brew install postgresql  
  
# To have launchd start postgresql at login:  
$ brew services start postgresql
```

5) การใช้งานเว็บไซต์ระบบจัดการข้อมูลและคาดการณ์ระยะเวลาดำเนินงานของโครงการ

(1) Clone file โครงการจาก Github โดยใช้คำสั่ง

```
$ git clone https://github.com/thanabodeekrathu/Independent_499.git
```

(2) ไปยังโฟลเดอร์ /independent_499

(3) พิมพ์คำสั่ง rake db:create db:migrate db:seed สำหรับการสร้างฐานข้อมูลของระบบ

(4) พิมพ์คำสั่ง bundle install เพื่อติดตั้งแพคเกจที่สำคัญของระบบ

(5) พิมพ์คำสั่ง rails s เพื่อเริ่มการใช้งานระบบ