

ผลการศึกษาการใช้กระบวนการ agile ในวิชาโครงงาน

กรณีศึกษาหลักสูตรวิศวกรรมซอฟต์แวร์ มหาวิทยาลัยวลัยลักษณ์

The Study Results of Agile Process in Senior Project

Case Study: Software Engineering Program in Walailak University

กรัณรัตน์ ธรรมรักษ์ (Karanrat Thammarak)*

บทคัดย่อ

งานวิจัยนี้นำเสนอผลการศึกษาการใช้กระบวนการ agile ในรายวิชาโครงงาน ของหลักสูตรวิศวกรรมซอฟต์แวร์ มหาวิทยาลัยวลัยลักษณ์ เพื่อ 1) ศึกษาผลการใช้กระบวนการ agile ต่อผลผลิตที่ได้ในการทำโครงงานของนักศึกษา 2) ศึกษาผลการใช้กระบวนการ agile ต่อพัฒนาการด้านทักษะการทำงานของนักศึกษา 3) เพื่อหาปัจจัยที่เกี่ยวข้องกับความสำเร็จและล้มเหลวของโครงงานที่ใช้กระบวนการ agile และ 4) เพื่อหาแนวปฏิบัติที่ดีที่สามารถนำมาใช้ในการปรับปรุงกระบวนการให้ดีขึ้น จากการดำเนินงาน พบว่า 1) ผลผลิตที่ได้ออกมาเป็นชิ้นงานมากขึ้นและผ่านการตรวจรับจากผู้ได้รวดเร็วขึ้น 2) ทักษะทางด้านการเขียนโปรแกรม การทดสอบ และความเข้าใจในกระบวนการพัฒนาซอฟต์แวร์สูงขึ้นอย่างชัดเจน 3) ปัจจัยเรื่องการจัดตั้งและบริหารทีมงาน และกระบวนการพัฒนาทักษะด้านการเขียนโปรแกรม เป็นปัจจัยที่มีผลต่อความสำเร็จและความล้มเหลวของโครงงาน และ 4) การทำการหมุนเวียนงานร่วมกับการทำการจับคู่เขียนโปรแกรมจะช่วยให้มีความเร็วรอบในการส่งงานได้

คำสำคัญ: กระบวนการ agile วิชาโครงงานทางคอมพิวเตอร์ หลักสูตรวิศวกรรมซอฟต์แวร์ มหาวิทยาลัยวลัยลักษณ์

Abstract

This paper presents the study results of agile process in the senior project based on case study of Software Engineering Program in Walailak University. The objectives of this research were: 1) to study and effective of Agile process to

student productivity in senior project, 2) to study and effective of Agile process to student's skill improvement, 3) to find an success and failure factors of agile projects and 4) to find a good practice that can improve the process. The research findings found that: 1) agile process can improves the student productivity, 2) agile process can improves a programming skill, testing skill and software process skill of student, 3) team construction and management is an importance factor to define success and failure of project and finally 4) a combination between job rotation and pair programming can reduce delivery time in this project.

Keywords: Agile Process, Senior Project, Software Engineering Program, Walailak University.

1. บทนำ

รายวิชาโครงงานทางคอมพิวเตอร์เป็นรายวิชาสำคัญในการพัฒนาผู้เรียนให้มีความสามารถในการบูรณาการองค์ความรู้ที่เรียนมาตลอดหลักสูตรเพื่อใช้ในการออกแบบและพัฒนาซอฟต์แวร์ให้เกิดขึ้นและใช้งานได้จริง โดยทั่วไปการกำหนดโครงสร้างและเนื้อหาวิชาในรายวิชาในวิชาโครงงานมักกำหนดให้มีรายวิชาโครงงาน 2 รายวิชาต่อเนื่องกันและมักจัดรูปแบบรายวิชาแบบโครงงานเป็นฐาน [1-4] สำหรับหลักสูตรวิศวกรรมซอฟต์แวร์ มหาวิทยาลัยวลัยลักษณ์ก็เช่นเดียวกัน มีการกำหนดรายวิชาโครงงานทางวิศวกรรมซอฟต์แวร์ขึ้นเพื่อให้ให้นักศึกษาชั้นปีที่ 4 ได้ใช้ทักษะความรู้ทั้งหมดที่เรียนมากับการแก้โจทย์ปัญหาจริง อย่างไรก็ตามการพัฒนาวิชาโครงงานให้มีความเหมาะสมและบรรลุ

* สำนักวิชาสารสนเทศศาสตร์ มหาวิทยาลัยวลัยลักษณ์

* School of Informatics, Walailak University.

ปรัชญาและปณิธานการผลิตบัณฑิตของหลักสูตรวิศวกรรมซอฟต์แวร์ ตลอดจนมีความสอดคล้องกับกรอบมาตรฐาน ACM –SWEBOK และมาตรฐานคุณวุฒิระดับปริญญาตรี สาขาคอมพิวเตอร์ (มคอ.1) นั้นเป็นความท้าทายอย่างยิ่งในการออกแบบรายวิชาโครงการให้มีความสามารถตอบสนองต่อประเด็นการผลิตบัณฑิตข้างต้น ประเด็นความท้าทายหนึ่งคือปัจจัยการคัดเลือกกระบวนการพัฒนาซอฟต์แวร์ที่เหมาะสมมาใช้ในการทำงานของนักศึกษา ซึ่งในปัจจุบันนี้มีกระบวนการพัฒนาซอฟต์แวร์ทั้งเก่าและใหม่จำนวนมากไม่ยอมให้เลือก เช่น กระบวนการพัฒนาซอฟต์แวร์แบบน้ำตก (Waterfall) สไปรัล (Spiral) อาร์ยูพี (RUP) และอไจล์ (Agile) เป็นต้น สำหรับงานวิจัยนี้ได้ทำการศึกษาการทดลองใช้กระบวนการอไจล์ ที่ใช้วิธีการพัฒนาแบบเอ็กซ์ตรีมโปรแกรมมิ่ง (Extreme Programming) [6] ซึ่งเป็นกระบวนการพัฒนาซอฟต์แวร์สมัยใหม่ที่ใช้กันอย่างแพร่หลายในกลุ่มนักพัฒนาซอฟต์แวร์ มาใช้ในการจัดการเรียนการสอนในรายวิชาโครงการทางวิศวกรรมซอฟต์แวร์โดยมีวัตถุประสงค์งานวิจัยเพื่อ 1) ศึกษาผลการใช้งานกระบวนการ อไจล์ต่อผลผลิตที่ได้ในการทำโครงการของนักศึกษา 2) ศึกษาผลการใช้งานกระบวนการอไจล์ต่อพัฒนาการด้านทักษะการทำงานของนักศึกษา 3) เพื่อหาปัจจัยที่เกี่ยวข้องกับความสำเร็จและล้มเหลวของโครงการที่ใช้กระบวนการอไจล์ 4) เพื่อหาแนวปฏิบัติที่ดีที่สามารถนำมาใช้ในการปรับปรุงกระบวนการให้ดีขึ้น

2. ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 การเรียนรู้แบบโครงการเป็นฐาน (Project-based Learning)

การเรียนรู้แบบโครงการเป็นฐาน เป็นหนึ่งในรูปแบบการจัดการเรียนรู้ที่เน้นผู้เรียนเป็นสำคัญ โดยใช้โครงการเป็นเครื่องมือในการขับเคลื่อนการเรียนรู้และพัฒนาทักษะของผู้เรียน ซึ่งในปัจจุบันมีงานวิจัยจำนวนมากทั้งในและต่างประเทศที่ใช้รูปแบบการเรียนรู้แบบโครงการเป็นฐานนี้ เช่น งานวิจัยของสันติ หุตะมาน และพลศักดิ์ โกษียาภรณ์ [1] นำเสนอการพัฒนาแบบการจัดการเรียนรู้ตามแนวคอนสตรัคติวิสต์โดยใช้โครงการเป็นฐานร่วมกับระบบพี่เลี้ยงในรายวิชาระบบควบคุมแบบคลาสสิก งานวิจัยของนิลาวัลย์ และศิรินทร [2] นำเสนอผลของการใช้วิธีการสอนแบบ

โครงการที่มีต่อพัฒนาการด้านทักษะภาษาอังกฤษ งานวิจัยของจิตติยาและคณะ[3] นำเสนอการเรียนรู้ร่วมกันแบบผสมผสานและใช้โครงการเป็นฐานที่ส่งผลต่อการสร้างความรู้และผลสัมฤทธิ์ทางการเรียน และงานวิจัยของ José A. Macías [4] นำเสนอวิธีการเพิ่มประสิทธิภาพการเรียนรู้แบบโครงการเป็นฐานในการเรียนการสอนปฏิบัติการทางวิศวกรรมซอฟต์แวร์ และ งานวิจัยของ Andrea Prencipe และคณะ [5] นำเสนอการใช้งานวิธีการเอ็มบีทีไอเพื่อประเมินความสำเร็จของทีมทางวิศวกรรมในการเรียนรู้แบบโครงการเป็นฐาน เป็นต้น

2.2 กระบวนการพัฒนาซอฟต์แวร์แบบอไจล์

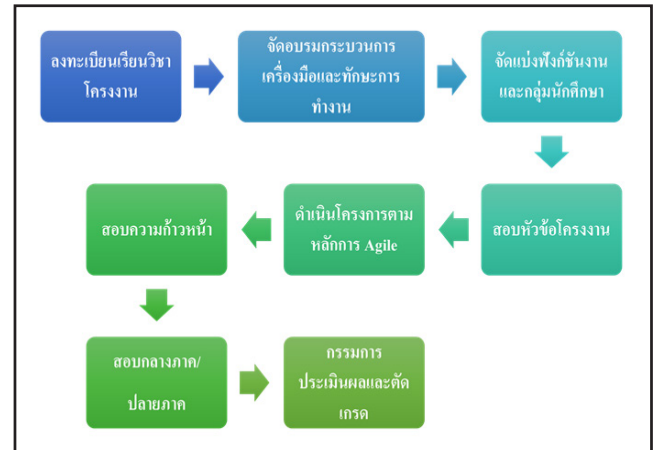
กระบวนการพัฒนาซอฟต์แวร์แบบอไจล์ ประกอบด้วยหลักการทำงานทั้งสิ้น 12 ข้อ [6], [7] ดังนี้ 1) การจับคู่เขียนโปรแกรม (Pair Programming) 2) การวางแผนแบบเกม (Planning Game) 3) การขับเคลื่อนการพัฒนาโดยการทดสอบ (Test Driven Development) 4) การมีส่วนร่วมทั้งทีม (Whole Team) 5) การต่อประสานอย่างต่อเนื่อง (Continuous Integration) 6) การพัฒนาการออกแบบ (Design Improvement) 7) การปล่อยออกทีละน้อย (Small Release) 8) การวางมาตรฐานการเขียนโค้ด (Coding Standard) 9) การอนุญาตให้ผู้อื่นสามารถแก้ไขในส่วนของตนเองได้ (Corrective Code Ownership) 10) ออกแบบให้ง่าย (Simple Design) 11) มีการทำ System Metaphor และ 12) การให้งานไม่ควรเกิน 40 ชม./สัปดาห์ (Sustainable Pace) ในปัจจุบันการพัฒนาตามวิถีทางแห่งอไจล์มีหลายวิธี เช่น สครัม (Scrum) คัมบัง (Kanban) ลีน (Lean) และ เอ็กซ์ตรีมโปรแกรมมิ่ง เป็นต้น ซึ่งในที่นี้เน้นวิธีการแบบเอ็กซ์ตรีมโปรแกรมมิ่ง ดังจะเห็นได้จากงานวิจัยต่างๆ ที่เกิดขึ้น เช่น การวิเคราะห์เปรียบเทียบความสามารถในการพัฒนาและบำรุงรักษาซอฟต์แวร์ระหว่าง กระบวนการพัฒนาแบบ Agile และ Non-Agile [7] และวิธีการพัฒนาซอฟต์แวร์ด้วยวิธีผสมผสานระหว่างสครัมและเอ็กซ์ตรีมโปรแกรมมิ่ง [8] เป็นต้น นอกจากนี้ในขอบเขตงานด้านการศึกษาก็ยังพบงานวิจัยที่เกี่ยวข้องกับการประยุกต์ใช้กระบวนการนี้จำนวนมาก เช่น งานวิจัยของ Michael E. Caspersen และ Jens Bennedsen [9], [10] ได้นำเสนอวิธีการสอนอไจล์ โดยใช้หลักการจดจำการเรียนรู้ของมนุษย์และนำเสนอวิธีการใช้งานอไจล์ที่ปรับปรุงใหม่เพื่อให้เหมาะสมกับความต้องการของนักเรียน เป็นต้น

2.3 รายวิชาโครงการ หลักสูตรวิศวกรรมซอฟต์แวร์ มหาวิทยาลัยวลัยลักษณ์

หลักสูตรวิศวกรรมซอฟต์แวร์ มหาวิทยาลัยวลัยลักษณ์ กำหนดให้นักศึกษาชั้นปีที่ 4 ทุกคนลงทะเบียนเรียนวิชาโครงการจำนวน 2 รายวิชาต่อเนื่องกัน คือรายวิชา SWE-494 โครงการทางวิศวกรรมซอฟต์แวร์ 1 (Senior Project in Software Engineering1) โดยมีจำนวนหน่วยวิชาทั้งสิ้น 0.5 (0-6-0) หน่วยวิชา แบ่งเป็นการฝึกปฏิบัติ จำนวน 6 ชั่วโมงต่อสัปดาห์ การตัดเกรด แบ่งเป็นช่วง A-F และรายวิชา SWE-495 โครงการทางวิศวกรรมซอฟต์แวร์ 2 (Senior Project in Software Engineering2) โดยมีจำนวนหน่วยวิชาทั้งสิ้น 1 (0-12-0) หน่วยวิชา แบ่งเป็นการฝึกปฏิบัติ จำนวน 12 ชั่วโมงต่อสัปดาห์ เพื่อฝึกการค้นคว้าของนักศึกษาตลอดจนถึงการศึกษาค้นคว้า วิเคราะห์วางแผนโครงการและการทำโครงการทางด้านวิศวกรรมซอฟต์แวร์

การออกแบบเนื้อหาวิชาดังกล่าว เน้นการทำงานเสมือนจริง ที่ต้องทำงานเป็นทีม สมาชิกในทีมมีบทบาทหน้าที่แตกต่างกัน เช่น นักวิเคราะห์ระบบ (System Analyst) นักออกแบบระบบ (System Designer) นักพัฒนา (Developer) และ นักทดสอบ (Tester) เป็นต้น กำหนดขนาดของกลุ่มอยู่ที่ 4-6 คนขึ้นไป

การวัดผลจะแบ่งเป็นการนำเสนอข้อเสนอโครงการ การร่วมทีมงาน และประสิทธิภาพทีมงาน นอกจากนี้โครงการของหลักสูตรได้รับการสนับสนุนโจทย์งานจริงจากบริษัท และเข้าร่วมบริหารโครงการ และประเมินผลงานด้วยตนเอง ร่วมกับอาจารย์ที่ปรึกษาตลอดระยะเวลาดำเนินงาน โดยระบบที่เป็นกรณีศึกษาในครั้งนี้คือระบบจัดการร้านอาหาร แบ่งเป็นระบบย่อยจำนวน 3 ระบบ คือ ระบบจัดการหน้าร้าน หรือ POS ระบบบัญชี/การเงิน คลังวัตถุดิบ วิเคราะห์การตลาด ทรัพย์สิน หรือ AmFin และระบบดำเนินการวิเคราะห์การตลาดและจัดการโปรโมชั่น หรือ M2P ด้านอุปกรณ์และเครื่องมือใช้ในการทำงานในโครงการนี้ จะอิงตามเครื่องมือที่สถานประกอบการใช้ทำงานจริง เช่น 1) Bitbucket 2) MongoDB 3) Atom Text Editor 4) Ruby on Rails 5) Rubygems 6) HTML 7) CSS 8) Javascript 9) Angular.js 10) jQuery 11) Rspec 12) Capybara และ 13) Pivotal Tracker เป็นต้น ประกอบด้วยกระบวนการหลัก 8 กระบวนการ ดังภาพที่ 1



ภาพที่ 1 กระบวนการดำเนินการในรายวิชาโครงการทางวิศวกรรมซอฟต์แวร์

จากภาพกระบวนการดำเนินงานมีรายละเอียดดังนี้
 1) นักศึกษาลงทะเบียนเรียนวิชาโครงการ 2) หลักสูตรและสถานประกอบการจัดอบรมกระบวนการ เครื่องมือและทักษะการทำงานให้กับนักศึกษา 3) อาจารย์ และสถานประกอบการจัดแบ่งฟังก์ชันงาน และกลุ่มนักศึกษา 4) นักศึกษาแต่ละกลุ่มเริ่มสอบหัวข้อโครงการ 5) เมื่อผ่านการอนุมัติหัวข้อโครงการ นักศึกษาแต่ละกลุ่มจะเริ่มดำเนินการออกแบบและพัฒนา ระบบแบบอไจล์ โดยมีกิจกรรมการทำงานดังนี้คือ 5ก) กำหนดรอบการทำงาน ทุกๆ 1 สัปดาห์ มีการประชุมร่วมกับสถานประกอบการผ่านระบบการประชุมทางไกล 5ข) ทุกกลุ่มมีการทำ Standup meeting ร่วมกันทุกวัน 5ค) ทุกกลุ่มมีการส่งมอบงานกับสถานประกอบการอย่างน้อย ทุกสัปดาห์ ทั้งนี้ยังสามารถนัดเพิ่มได้ 5ฅ) มีการจับคู่เขียนโปรแกรม (Pair Programming) เพื่อพัฒนานักศึกษาที่มีทักษะน้อยโดยอาศัยการทำงานร่วมกับนักศึกษาที่มีทักษะสูงกว่า 5ง) อาจารย์ที่ปรึกษาให้คำปรึกษาดูตามงาน และร่วมประชุมเป็นระยะ 5จ) ทุกสัปดาห์แต่ละกลุ่มจะต้องจัดทำเอกสารรายงานการดำเนินงานรายสัปดาห์ 6) สอบความก้าวหน้ากลางเทอม โดยนักศึกษาต้องนำเสนอฟังก์ชันงานที่ส่งมอบและเอกสารที่เกี่ยวข้องตามหลักการของอไจล์ 7) สอบปลายภาค 8) กรรมการประเมินผลและตัดเกรด แจ้งผลการศึกษาให้นักศึกษาทราบ

3. วิธีการดำเนินการวิจัย

งานวิจัยนี้ใช้รูปแบบการประเมินตามแบบจำลอง CIPP Model (Context-Input-Process-Product Model) [11] โดยมีรายละเอียดดังต่อไปนี้

- กลุ่มตัวอย่างที่ใช้คือ นักศึกษาที่ลงทะเบียนเรียนวิชา
โครงการ ทั้งสิ้น 13 คน แบ่งเป็น 3 กลุ่มตามฟังก์ชันงาน
คือ ระบบจัดการหน้าร้าน หรือ POS ระบบบัญชี/การเงิน คลัง
วัตถุดิบ วิเคราะห์การตลาด ทรัพย์สิน หรือ AmFin และระบบ
ดำเนินการวิเคราะห์การตลาดและจัดการโปรโมชั่น หรือ M2P

- ช่วงเวลาในการศึกษาคือ ภาคการศึกษาที่ 2/2559 ใน
ระบบไตรภาค ตั้งแต่ ธันวาคม 2558 - มีนาคม 2559

- เครื่องมือที่ใช้ในการเก็บรวบรวมข้อมูลประกอบด้วย
1) แบบสอบถาม 2) แบบวัดผลในรายวิชา 3) การสังเกตการณ์
และ 4) การประชุมกลุ่มย่อย

- การวิเคราะห์ข้อมูล ประกอบด้วย 1) ข้อมูลเชิงคุณภาพ
เป็นข้อมูลที่ได้จากการศึกษาวิเคราะห์เนื้อหาเอกสาร
ที่เกี่ยวข้อง การสังเกตการณ์ และการประชุมกลุ่มย่อย
2) ข้อมูลเชิงปริมาณ เป็นข้อมูลที่ได้จาก แบบสอบถาม แบบ
วัดผลในรายวิชา โดยใช้มาตราส่วนประมาณค่า ซึ่งกำหนด
เกณฑ์ในการให้คะแนนเป็น 5 ระดับตั้งแต่ 1-5 เรียงจากระดับ
ความคิดเห็นน้อยที่สุด ไปมากที่สุด ตามลำดับ

- ข้อมูลที่ได้มาจากคะแนนในแต่ละข้อของแบบสอบถาม
จะนำไปวิเคราะห์ค่าเฉลี่ย และส่วนเบี่ยงเบนมาตรฐาน
ของประชากรหรือกลุ่มตัวอย่าง โดยใช้โปรแกรมสำเร็จรูป
Microsoft Excel

4. ผลการประเมิน

ในส่วนนี้จะนำเสนอผลลัพธ์ที่ได้จากการประเมินตามแบบ
จำลอง CIPP ซึ่งประกอบด้วยผลการประเมินดังนี้

4.1 ผลการประเมินบริบท

จากการศึกษาเปรียบเทียบรายวิชา SWE-494 โครงการ
ทางวิศวกรรมซอฟต์แวร์ 1 กับ รายวิชา CSC-494 โครงการ
ทางวิทยาการคอมพิวเตอร์ 1 ซึ่งเป็นรายวิชาที่คล้ายคลึงกัน
ในหลักสูตรเดิมก่อนการปรับปรุงใหม่ พบข้อแตกต่าง
ดังตารางที่ 1

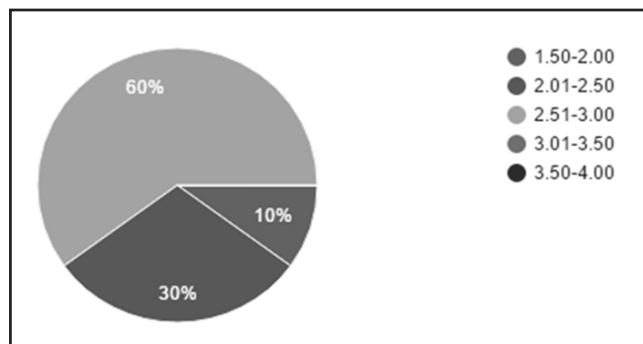
4.2 ผลการประเมินด้วยแบบสอบถามความคิดเห็น

ผลที่ได้จากการประเมินด้วยแบบสอบถามความคิดเห็น
ที่เกี่ยวข้องกับการประเมินปัจจัยป้อน กระบวนการ และ
ผลผลิต มีดังนี้

- กลุ่มนักศึกษาที่ลงทะเบียนเรียนวิชาโครงการประกอบ
ด้วยกลุ่มที่มีเกรดเฉลี่ย 1.50-2.00 คิดเป็น 10% เกรดเฉลี่ย

ตารางที่ 1 ผลการประเมินระบบให้คำแนะนำรายบุคคลเพื่อ
การวางแผนการทอ้งเที่ยว

ข้อแตกต่าง	CSC-494	SWE-494
1. จำนวนสมาชิก ในทีม	1-2 คน	4-6 คน
2. จำนวน ฟังก์ชันงาน	ตามขนาดของทีม	ตามขนาดของทีม
3. กระบวนการ	- มีอาจารย์ที่ปรึกษาโครงการ หรืออาจารย์ที่ปรึกษาร่วม 1-2 คน - มีการสอบข้อเสนอโครงการ กลางภาค และปลายภาค - มีการประเมินผลรายงานและผลผลิตที่ได้ แต่ไม่มีการประเมินผลกระบวนการ - ไม่มีการร่วมดำเนินการกับสถานประกอบการ	- มีทีมที่ปรึกษาโครงการประกอบด้วยอาจารย์และสถานประกอบการ - มีการสอบข้อเสนอโครงการ กลางภาค และปลายภาค - มีการประเมินผลรายงาน ผลผลิตที่ได้และกระบวนการดำเนินงาน - มีการร่วมดำเนินการกับสถานประกอบการตลอดการดำเนินงาน

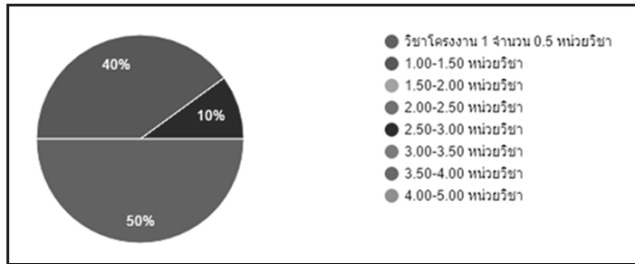


ภาพที่ 2 การจำแนกเกรดเฉลี่ยของนักศึกษาที่ลงทะเบียน
เรียนวิชาโครงการ

2.01-2.50 คิดเป็น 30% และ กลุ่มที่มีเกรดเฉลี่ย 2.51-3.00
คิดเป็น 60% ดังภาพที่ 2

- ในวิชาเทอม 2/2559 นักศึกษาลงทะเบียนเรียนเฉพาะ
วิชาโครงการ คิดเป็น 50 % มีวิชาเรียนที่ต้องลงทะเบียน
เรียนอื่นนอกเหนือจากวิชาโครงการจำนวน 1.00-1.50
หน่วยวิชา คิดเป็น 40% และ 2.5-3.00 หน่วยวิชา คิดเป็น
10% ดังภาพที่ 3

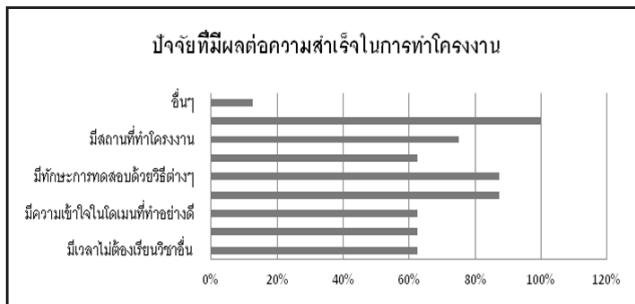
- มีทักษะความชำนาญด้านการเขียนโปรแกรมก่อน
การทำโครงการในระดับน้อยมาก น้อยและปานกลาง คิดเป็น



ภาพที่ 3 การจำแนกเกรดเฉลี่ยของนักศึกษาที่ลงทะเบียนเรียนวิชาโครงการ

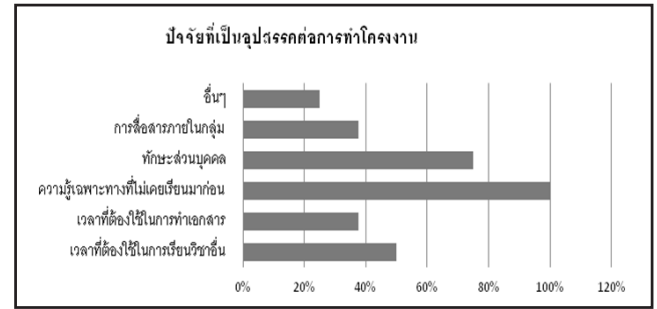
10% 60% และ 30% ตามลำดับ

- ทักษะความชำนาญด้านกระบวนการแบบอโรไลน์ก่อนการทำโครงการในระดับน้อยมาก น้อย และปานกลาง คิดเป็น 20% 50% และ 30% ตามลำดับ
- ทักษะความชำนาญด้านการทดสอบโปรแกรมก่อนการทำโครงการในระดับน้อยมาก น้อย และปานกลาง คิดเป็น 30% 30% และ 40% ตามลำดับ
- ความมั่นใจในการปฏิบัติงานเมื่อจบการศึกษา ก่อนการทำโครงการอยู่ในระดับน้อยมาก น้อย ปานกลาง และมาก คิดเป็น 40% 40% 10% และ 10% ตามลำดับ
- ปัจจัยที่มีผลต่อความสำเร็จในกระบวนการทำโครงการของนักศึกษาสามอันดับแรกได้แก่ มีทีมที่ทำงานร่วมกันได้ (100%) ทักษะการใช้เครื่องมือต่างๆ (88%) และมีความชำนาญในการทดสอบแบบต่างๆ (88%) เป็นต้น



ภาพที่ 4 ปัจจัยที่มีผลต่อความสำเร็จในการทำโครงการ

- ปัจจัยที่เป็นอุปสรรคในกระบวนการทำโครงการของนักศึกษาสูงสุดสามอันดับแรกคือ ความรู้เฉพาะทางที่ไม่เคยเรียนมาก่อน (100%) เช่น TDD และเครื่องมือที่ใช้ ทักษะความสามารถในการเขียนโปรแกรมส่วนบุคคล (75%) และเวลาที่ต้องใช้ในการเรียนวิชาอื่นๆ (50%) มีดังนี้
- ทักษะความชำนาญด้านการเขียนโปรแกรมภายหลังการทำโครงการเพิ่มขึ้นในระดับปานกลางและมาก คิดเป็น 50% เท่ากัน



ภาพที่ 5 ปัจจัยที่เป็นอุปสรรคในการทำโครงการ

- ทักษะความชำนาญด้านกระบวนการแบบอโรไลน์ภายหลังการทำโครงการเพิ่มขึ้นในระดับปานกลาง มาก และมากที่สุด คิดเป็น 40% 50% และ 10% ตามลำดับ
- ทักษะความชำนาญด้านการทดสอบโปรแกรมภายหลังการทำโครงการเพิ่มขึ้นในระดับปานกลาง มาก และมากที่สุด คิดเป็น 30% 60% และ 10% ตามลำดับ
- ความมั่นใจในการปฏิบัติงานเมื่อจบการศึกษาภายหลังการทำโครงการเพิ่มขึ้นในระดับน้อย ปานกลาง มาก และมากที่สุด คิดเป็น 10% 10% 40% และ 40% ตามลำดับ

4.3 ผลการสังเกตการณ์และการประชุมกลุ่มย่อย

ผลที่ได้จากการสังเกตการณ์และการประชุมกลุ่มย่อย มีดังนี้

- นักศึกษาพบปัญหาความล่าช้าในงานเนื่องจากการทำการจับคู่เขียนโปรแกรมเพราะต้องสอนงานให้กับเพื่อนที่มีทักษะน้อยกว่า จากปัญหาดังกล่าวมีการปรับปรุงกระบวนการโดยการทำการหมุนเวียนงาน (Job Rotation) ร่วมกับการทำการจับคู่เขียนโปรแกรม กล่าวคือ มีการกำหนดหน้าที่หลักให้กับแต่ละคนในทีมตามความชำนาญ และให้ย้ายคนจากงานอื่นเข้ามาทำงานที่ไม่ถนัดเพื่อเพิ่มทักษะที่ละน้อย เช่น ในการทำประกันคุณภาพ (Quality Assurance) จะมีคนทำงานหลักรับผิดชอบงานส่วนใหญ่ และคนที่ถูกย้ายงานจะรับงานประมาณ 10-15% ในช่วงแรกโดยมีคนที่ทำหน้าที่รับผิดชอบหลักคอยสอนงานให้และทำงานร่วมกันแบบจับคู่เขียนโปรแกรม เมื่อมีความชำนาญแล้วคนที่ย้ายงานจะสามารถรับงานที่ตนไม่ถนัดเพิ่มขึ้นได้ซึ่งช่วยแก้ปัญหาความล่าช้าในงานได้ และยังเพิ่มทักษะที่รอบด้านให้กับคนในทีมอีกด้วย

- สมาชิกในกลุ่มที่ยังมีวิชาเรียนนอกเหนือจากรายวิชาโครงการ ทำให้ประสิทธิภาพของกลุ่มลดลง เพราะต้องแบ่งเวลาไปเข้าชั้นเรียน และทำการบ้าน หรืองานที่ได้รับ

มอบหมายในรายวิชา ซึ่งแตกต่างจากกลุ่มที่ทำงานเต็มเวลา

- ควรมีการฝึกทำโครงการหรือทักษะการเขียนโปรแกรม การวางแผน การจอย และอื่นๆ เพื่อช่วยให้มีประสิทธิภาพในการทำโครงการจริงมากขึ้น

4.4 ผลการสอบวัดผลโครงการทั้งกลางภาคและปลายภาค

ผลที่ได้จากการสอบวัดผลโครงการทั้งกลางภาคและปลายภาค พบว่า กลุ่มที่ทำงานเต็มเวลา (POS) ซึ่งทำข้อตกลงส่งมอบงานจำนวน 4 ฟังก์ชัน ส่งมอบได้จริง 4 ฟังก์ชัน และกลุ่มที่สมาชิกในกลุ่มยังมีวิชาเรียนอยู่ คือกลุ่ม M2P และ AmFin ซึ่งทำข้อตกลงส่งมอบงานจำนวน 4 ฟังก์ชันเท่ากัน และส่งมอบได้จริง 2 ฟังก์ชันเท่านั้น

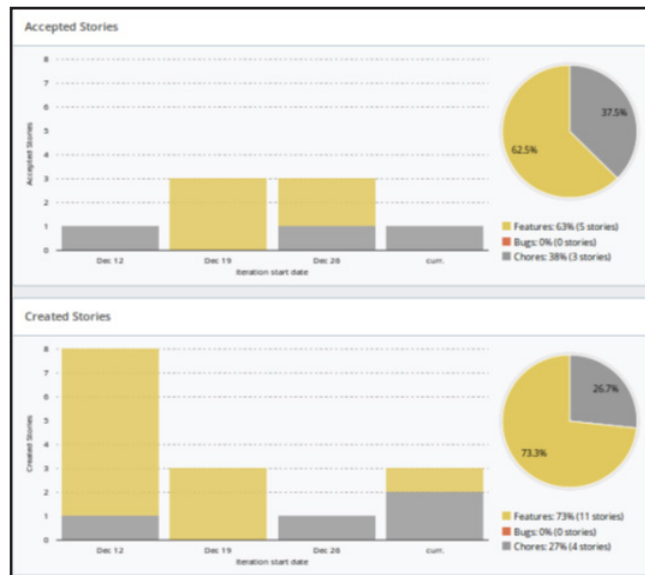
4.5 ผลการศึกษารายงานผลการดำเนินงานรายสัปดาห์

ผลที่ได้จากการศึกษารายงานผลการดำเนินงานรายสัปดาห์ ที่ประกอบด้วยหัวข้อต่างๆ เช่น กิจกรรมที่ทำในรอบสัปดาห์ จำนวนงานที่ได้รับผิดชอบ จำนวนงานที่ทำเสร็จ จำนวนงานที่ส่งมอบและปฏิเสธ ระยะเวลาการทํางานเขียนโปรแกรม และระยะเวลาในการศึกษาเพิ่มเติม ทั้งแบบรายบุคคลและทีม ในช่วงระหว่าง เดือนธันวาคม 2558 ถึง มีนาคม 2559 พบว่าประสิทธิภาพของงานเมื่อเปรียบเทียบกับแต่ละช่วงของรายงานเพิ่มขึ้นในทุกกลุ่ม เช่น คุณภาพของงานที่ส่งมอบ มีข้อผิดพลาดน้อยลง ปริมาณความสำเร็จของงานเพิ่มขึ้นในทุกกลุ่มซึ่งมีเปอร์เซ็นต์การงานที่ทำและสถานะความสำเร็จในงานที่เพิ่มขึ้น

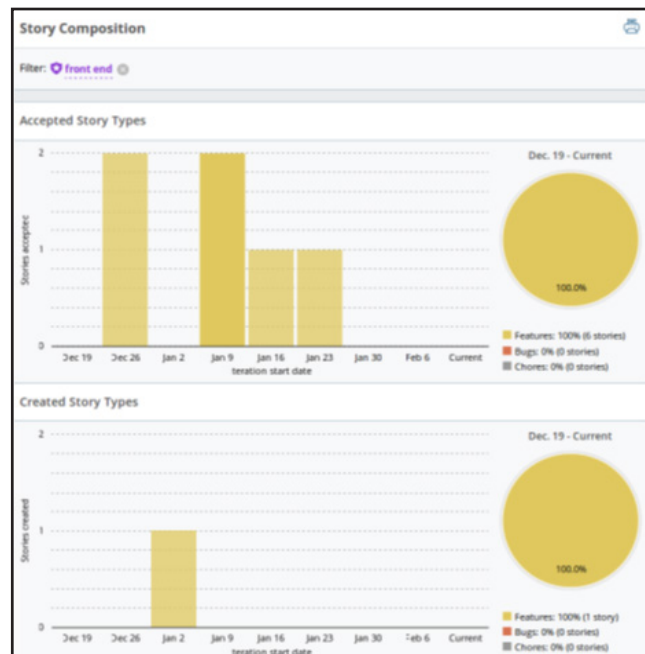
เมื่อพิจารณาความสามารถรายบุคคล นักศึกษาทุกคนในทีมสามารถรับงานได้เพิ่มขึ้นจากสัปดาห์ที่ 1 ถึงสัปดาห์ที่ 7 เฉลี่ยคนละ 4.79 งานต่อคน และมีงานที่ถูกปฏิเสธลดลงเฉลี่ย 0.25 งานต่อคน นอกจากนี้ยังมีค่าเฉลี่ยการทำการจับคู่เขียนโปรแกรมอยู่ที่ 17.16 ชั่วโมงต่อคน และค่าเฉลี่ยการศึกษาเพิ่มเติมอยู่ที่ 26.33 ชั่วโมงต่อคน ซึ่งในส่วนนี้ยังไม่มีผลที่แตกต่างอย่างมีนัยสำคัญ ดังจะเห็นได้ดังภาพที่ 6 และ 7

5. การอภิปรายและสรุปผล

งานวิจัยนี้เสนอผลการศึกษาการใช้งานกระบวนการจอยในรายวิชาโครงการ ของหลักสูตรวิศวกรรมซอฟต์แวร์ มหาวิทยาลัยวลัยลักษณ์ ทั้งนี้เพื่อ 1) ศึกษาผลการใช้งาน



ภาพที่ 6 Accepted Stories สัปดาห์ที่ 2



ภาพที่ 7 Accepted Stories สัปดาห์ที่ 4

กระบวนการจอยต่อผลผลิตที่ได้ในการทำโครงการของนักศึกษา 2) ศึกษาผลการใช้งานกระบวนการจอยต่อพัฒนาการด้านทักษะการทำงานของนักศึกษา 3) เพื่อหาปัจจัยที่เกี่ยวข้องกับความสำเร็จและล้มเหลวของโครงการที่ใช้กระบวนการจอย และ 4) เพื่อหาแนวปฏิบัติที่ดีที่สามารถนำมาใช้ในการปรับปรุงกระบวนการให้ดีขึ้น จากการดำเนินงานและประเมินผลด้วยแบบจำลอง CIPP พบข้อสรุป ดังนี้ 1) ผลผลิตที่ได้ออกมาเป็นชิ้นงานมากขึ้นและผ่านการตรวจรับจากผู้ใช้ได้รวดเร็วขึ้นเมื่อเทียบกับรายวิชา CSC-494 ซึ่งใช้

กระบวนการแบบจำลองน้ำตก ซึ่งผลลัพธ์ที่ได้เมื่อสิ้นสุดคือ การนำเสนอการวิเคราะห์และออกแบบระบบเท่านั้น 2) ทักษะทางด้านการเขียนโปรแกรม การทดสอบ และความเข้าใจในกระบวนการพัฒนาซอฟต์แวร์สูงขึ้นอย่างชัดเจน ซึ่งสอดคล้องกับผลที่ได้จากแบบสอบถามที่ดีขึ้นในทุกด้าน ภายหลังการทำโครงการ และผลการดำเนินงานรายสัปดาห์ ที่พบว่านักศึกษาทีมงานที่ถูกปฏิเสธลดลงเฉลี่ย 0.25 งาน ต่อคน 3) ปัจจัยที่เกี่ยวข้องกับความสำเร็จและล้มเหลวของโครงการ คือการจัดตั้งและบริหารทีมงาน และการพัฒนา ทักษะด้านการเขียนโปรแกรม โดยการจัดตั้งทีมควรพิจารณา เรื่อง เกียรติยศส่วนบุคคล และวิชาเรียนร่วมด้วย เนื่องจาก พบว่าทีมที่สมาชิกทุกคนไม่มีวิชาเรียนและมีระดับเกรด เฉลี่ยสูง (POS) ทำงานได้ผลผลิตที่สูงกว่าอีกสองทีมและ 4) การทำการหมุนเวียนงานร่วมกับการจับคู่เขียนโปรแกรม จะช่วยเพิ่มความไว้วางใจในการส่งงานได้

สำหรับงานวิจัยในอนาคต คือการศึกษาเพื่อหาแนวปฏิบัติที่ดีในการจัดตั้งและบริหารทีมงาน ซึ่งมีความเป็นไปได้ที่จะนำแนวทางการทำ PSP (Personal Software Process) และ TSP (Team Software Process) รวมทั้งวิธีการ MBTI มาใช้ในการแก้ปัญหาดังกล่าวต่อไป

6. เอกสารอ้างอิง

- [1] S. Hutamarn and P. Koseeyaporn. "The Development of Learning Management via Project-based Learning with Mentor under Constructivist Theory for Classical Control System Subject." *Journal of Education Naresuan University*, Vol 17, No 2, pp. 108-122, 2015.
- [2] N. Newprasit and S. Seepho. "The Effects Of A Project-Based Learning Approach On The Improvement Of English Language Skills." *Journal of Applied Language Studies and Communication*, Vol. 1, No. 1, pp. 16-51, January – June 2015
- [3] T. Natewong and B. Thamnanjit. "The Effect of Blended Collaborative Learning and Project-Based Approaches on Knowledge Construction and Learning Achievement." *SDU Research Journal Humanities and Social Sciences*, Vol 8, No 3, pp. 1-11, September - December 2012.
- [4] J. A. Macías. "Enhancing Project-Based Learning in Software Engineering Lab Teaching Through an E-Portfolio Approach." *IEEE Transactions on Education*, Vol. 55, No. 4, pp. 502-507, 2012.
- [5] V. R. Montequí'n, J. M. M. Fernández J. V. Balsera, and A. G. Nieto. "Using MBTI for the success assessment of engineering teams in project-based learning." *International Journal of Technology and Design Education*, pp.1127–1146, 2013.
- [6] K. Beck. *Extreme programming explained - embrace change*, Addison Wesley, 2000.
- [7] C. Klakhaeng, S. Rongviriyapanish, and T. Apiwattanapong. "Comparative analysis of software productivity and maintainability in Agile (Extreme programming) and Non-Agile (Waterfall-based) development." *The 12th National Computer Science and Engineering Conference (NCSEC 2008)*, pp. 421-428, 2008.
- [8] P. Maleekaew and N. Porrawatpreyakorn. "Software Development Using Hybrid Scrum/XP." *The 12th National Computer Science and Engineering Conference (NCSEC 2008)*, pp. 421-428, 2008.
- [9] M. E. Caspersen and J. Bennedsen. "Instructional Design of a Programming Course - A Learning Theoretic Approach." *ICER '07*, pp. 111-122, 2007.
- [10] M. E. Caspersen and J. Bennedsen. "Towards an Agile Approach to Adapting Dynamic Collaboration Support to Student Needs." *International Journal of Artificial Intelligence in Education*, pp. 92–124, 2014.
- [11] Y. R. Wibulsri. *Project concepts and practices*, Chulalongkorn University, Bangkok, 2542.