

ซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม (AppDO:SI)

ภคพล พงษ์ประดิษฐ์ ณัฏพล โลหิตรัตน์ ปิยะบุตร น้อมประสิทธิ์ อภิสิทธิ์ แสงใส และ นวลศรี เด่นวัฒนา

สาขาวิชาวิศวกรรมซอฟต์แวร์ คณะวิทยาการสารสนเทศ
มหาวิทยาลัยบูรพา จังหวัดชลบุรี

Emails : , pakkaphon.phongpadit@gmail.com, nutchapon.lohitrut@gmail.com, samsoikul_@hotmail.com,
a55160022@gmail.com, nuansri@buu.ac.th

บทคัดย่อ

กระบวนการการพัฒนาเชิงบุคคลเป็นรายวิชาในสาขาวิชาวิศวกรรมซอฟต์แวร์ โดยประกอบไปด้วยเรื่องของวิธีการพัฒนาแบบฟอร์ม และการประมาณการการวางแผน การวัดและการจัดการโครงการ ดังนั้นจึงจำเป็นต้องมีระบบสารสนเทศเข้ามาช่วยเหลือ บทความนี้จึงนำเสนอเครื่องมือที่มีชื่อว่าซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม Application software DevelOpment : Self Improvement (AppDO:SI) ซึ่งสนับสนุนกระบวนการทางการพัฒนาเชิงบุคคลในการบันทึกข้อมูลตั้งแต่ระดับ PSP0 การนับบรรทัดของโปรแกรมซึ่งคือ PSP0.1 จนกระทั่งถึง PSP 2.0 ซึ่งคือการทบทวนการเขียนโปรแกรมและการทบทวนการออกแบบ โดยซอฟต์แวร์นี้ช่วยอำนวยความสะดวกในการบันทึกข้อมูลเวลาและจำนวนข้อบกพร่องในระหว่างพัฒนาซอฟต์แวร์ ตลอดจนช่วยวิเคราะห์ข้อมูลทางกระบวนการการพัฒนาเชิงบุคคล นอกจากนี้ซอฟต์แวร์ยังสามารถนำเสนอข้อมูลการทำงานรายบุคคลในรูปแบบกระดานข้อมูลอิเล็กทรอนิกส์

ABSTRACT

Abstract The Personal Software Process (PSP) is a subject in software engineering program. It provides engineers with a disciplined personal framework for doing software work. The PSP process consists of a set of methods, forms, and scripts that show software engineers how to plan, measure, and manage their work. There is a need for automated tool support for PSP, since the method is laborious if used manually. This paper presents a PSP tool called Application software DevelOpment: Self Improvement (AppDO:SI) that supports PSP0, a Line-Of-Code (LOC) counter for PSP0.1 until PSP2.0, time consuming for design and code reviews. It will help a student conveniently

practice PSP by reducing data gathering and analysis overhead. It will assist PSP training in universities and industries. The AppDO:SI also provide personal profile dashboard in the form of chart, table, and card.

คำสำคัญ—กระบวนการการพัฒนาเชิงบุคคล;

1. บทนำ

ในปัจจุบันตามแนวพระราชบัญญัติการศึกษาตลอดชีวิต[3] เป็นการศึกษาตลอดชีวิตสำหรับประชาชนเพื่อพัฒนาขีดความสามารถของตนเองอย่างต่อเนื่อง โดยการเรียนรู้ตลอดชีวิตนั้นเกิดจากการเปลี่ยนแปลงและพัฒนาตนเอง จากการที่ได้รับความรู้ ทักษะ หรือประสบการณ์จากการศึกษา หรือการกระทำกิจกรรมต่าง ๆ

ตามหลักการกระบวนการการพัฒนาเชิงบุคคลนั้น เป็นกระบวนการที่ช่วยส่งเสริมให้นักพัฒนาซอฟต์แวร์สามารถพัฒนาคุณภาพของ กระบวนการพัฒนาซอฟต์แวร์ของตนเอง โดยเน้นการจัดการข้อผิดพลาด และพัฒนาทักษะการประมาณและวางแผนให้อย่างถูกต้อง แม่นยำ และมีคุณภาพ

ดังนั้นซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม Application Software Development : Self Improvement (AppDO:SI) ที่ใช้หลักการกระบวนการการพัฒนาเชิงบุคคล นั้นจึงถูกพัฒนาขึ้นมา และนำไปใช้ในการเรียนการสอนของนิสิตในระดับปริญญาตรี สาขาวิชาวิศวกรรมซอฟต์แวร์ มหาวิทยาลัยบูรพา เพื่อใช้ในการบันทึกข้อมูลในการพัฒนาของนิสิต และนำข้อมูลมาวิเคราะห์เพื่อสร้างรายงานของนักพัฒนาสำหรับนำไปวิเคราะห์เพื่อพัฒนาตนเอง

การจัดเรียงเนื้อหาในบทความนี้ บทนำจะอธิบายในส่วนที่ 1 ในส่วนที่ 2 จะอธิบายแนวคิดและทฤษฎีที่เกี่ยวข้อง จากนั้นในส่วนที่ 3 จะกล่าวถึงซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม ตลอดจนรายละเอียดของการทำงานของซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม

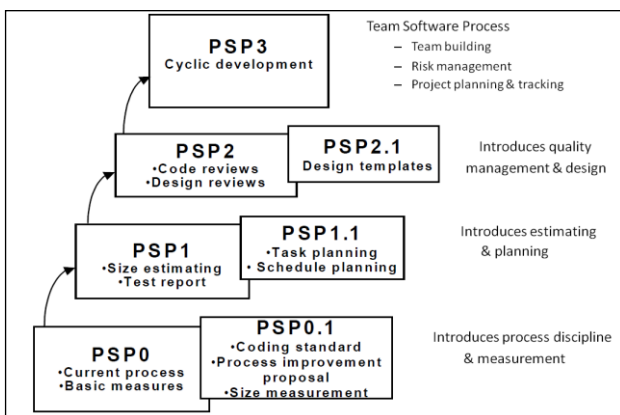
และสถาปัตยกรรมการเชื่อมโยงระหว่างข้อมูลของซอฟต์แวร์ ที่พัฒนาและการนำไปใช้งานจริง จะอธิบายในส่วนที่ 4 และในส่วนสรุปผลและงานในอนาคต จะกล่าวถึงในส่วนสุดท้าย

2. แนวคิดและทฤษฎีที่เกี่ยวข้อง

สำหรับในส่วนนี้จะเริ่มต้นด้วยการอธิบายหลักการของ PSP (Personal Software Process) จากนั้นจะอธิบายโดยสรุปของขอบเขตการทำงานของระบบสารสนเทศที่เกี่ยวข้องกับซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม

2.1. Personal Software Process (PSP)

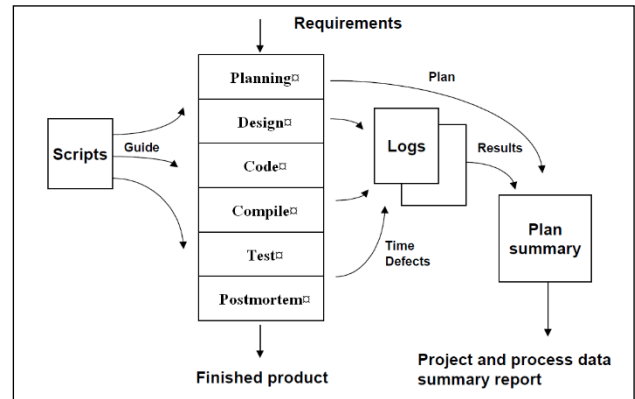
กระบวนการพัฒนาซอฟต์แวร์เชิงบุคคล (Personal Software Process : PSP) [2] เป็นบวนการที่ช่วยส่งเสริมให้นักพัฒนาซอฟต์แวร์สามารถพัฒนาคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ของตนเอง โดยเน้นการจัดการข้อผิดพลาด และพัฒนาทักษะการประมาณการและวางแผนให้อย่างถูกต้อง แม่นยำ และมีคุณภาพ กระบวนการพัฒนาซอฟต์แวร์ส่วนบุคคลถูกคิดค้นและพัฒนาเมื่อปี ค.ศ. 1989 โดย Watt S. Humphrey และ Software Engineering Institute (SEI) Carnegie Mellon University ประเทศสหรัฐอเมริกา กระบวนการพัฒนาซอฟต์แวร์ส่วนบุคคลจะใช้วิธีการเก็บรวบรวมข้อมูลของการพัฒนาซอฟต์แวร์โดยละเอียด เช่นข้อมูลเวลาในการพัฒนา ความผิดพลาดที่เกิดขึ้น และจำนวนบรรทัดของชิ้นงาน และนำข้อมูลที่ได้เหล่านั้นมาคำนวณ วิเคราะห์ ประมาณการ และสรุปผลซึ่งจะช่วยให้วิศวกรสามารถประมาณการขนาดและเวลาในการพัฒนาซอฟต์แวร์ได้ถูกต้องและแม่นยำมากขึ้น อีกทั้งยังช่วยให้วิศวกรสามารถป้องกันจุดบกพร่องที่เกิดขึ้นต่อตนเองและซอฟต์แวร์ให้มีประสิทธิภาพสูงสุด ซึ่งกระบวนการพัฒนาซอฟต์แวร์เชิงบุคคลจะมีระดับของกระบวนการ ดังรูปที่ 1



รูปที่ 1. ระดับของกระบวนการพัฒนาซอฟต์แวร์เชิงบุคคล

PSP0 : The Baseline Personal Process

เป็นกระบวนการรายบุคคลสำหรับใช้อ้างอิง เปรียบเทียบกับกระบวนการที่ได้รับการปรับปรุง ฝึกปฏิบัติแล้วในภายหลัง แนะนำให้รู้จักวัฏจักร 6 ขั้นตอน (Phase) ในการพัฒนาซอฟต์แวร์ ได้แก่ Planning Design Code Compile Test และ Postmortem โดยจะแสดงในรูปที่ 2



รูปที่ 2 PSP Process Flow

PSP0.1 : The Baseline Personal Process

ในขั้นตอน PSP0.1 เป็นการแนะนำแบบฟอร์มเอกสาร Process Improvement Proposals (PIP) ที่ให้บันทึกแนวความคิดที่ต้องการใช้เพื่อพัฒนากระบวนการ

PSP1 : Personal Project Management

เป็นการเน้นเทคนิคการจัดการโครงการรายบุคคล โดยแนะนำการประมาณขนาดของโปรแกรมและแรงงาน (Effort) ที่คาดว่าจะใช้ การวางแผนงาน (Task planning) การวางแผนตารางเวลา (Schedule planning) และการติดตาม (Tracking) การประมาณขนาดของโปรแกรม และแรงงาน โดยวิธี PROBE (Proxy-Based Estimating)

PSP2 : Personal Quality Management

ในขั้นตอน PSP2 มีการเพิ่มการออกแบบรายบุคคล (Personal design) การทบทวน Code (Code review) เพื่อช่วยให้วิศวกรสามารถพบข้อบกพร่องได้เร็วขึ้นในกระบวนการ ทำให้การแก้ปัญหาได้เร็วและประหยัดเวลา มีการวิเคราะห์ข้อบกพร่องที่พบ และนำมาใช้เป็นข้อมูลในการสร้างรายการตรวจสอบ (Review checklists)

PSP2.1 เป็นการชี้ให้เห็นถึงความจำเป็นของการออกแบบ

โดยเพิ่มส่วนของ Design notation มีการกำหนดต้นแบบ (Template) ของการออกแบบ 4 แบบ และวิธีการทบทวนการออกแบบ ข้อกำหนดทางการปฏิบัติ (The Operation Specification Template :OST) ข้อกำหนดทางฟังก์ชัน (The

Functional Specification Template : FST) ข้อกำหนดทางสถานะ (The State Specification Template : SST) ข้อกำหนดทางตรรกะ (The Logic Specification Template : LST)

PSP3 : Cyclic Personal Process

โดย PSP3 จะแนะนำกลยุทธ์ในการพัฒนาเป็นวงรอบ นั่นหมายถึงโปรแกรมขนาดใหญ่จะถูกแบ่งเป็นส่วนงานย่อย ๆ ในแต่ละวงรอบ ในการพัฒนาตามความสามารถของตนเอง และทำส่วนที่เหลือในวงรอบถัดไป PSP3 มีการแนะนำ High-level design High-level design review Cycle planning Development cycles

2.2 CodeIgniter Framework

CodeIgniter [1] คือ framework ที่ถูกพัฒนาขึ้นด้วย ภาษา PHP สำหรับ CodeIgniter Framework เป็นเครื่องมือสำหรับช่วยพัฒนาเว็บไซต์และเว็บแอปพลิเคชัน โดยมีโครงสร้างการพัฒนาโปรแกรมอย่างเป็นระบบและรวมคำสั่งต่างๆ ที่จำเป็นต่อการพัฒนาเว็บไซต์ด้วย PHP ไว้ภายใน CodeIgniter พัฒนาโดย Rick Ellis ซึ่งมีลิขสิทธิ์เป็น Open Source สามารถดาวน์โหลดและใช้ได้ฟรี ถูกพัฒนาขึ้นตั้งแต่ปี 2006 และยังมีการพัฒนามาเรื่อยๆ จนถึงปัจจุบัน

2.3.ระบบสารสนเทศที่เกี่ยวข้อง

ในส่วนนี้จะเป็นการอธิบายระบบสารสนเทศที่มีลักษณะใกล้เคียงกับซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม โดยในหัวเรื่องนี้จะกล่าวถึง The Software Process Dashboard Project

2.3.1 The Software Process Dashboard Project

เป็นระบบ Open Source ที่นำแนวคิดของกระบวนการพัฒนาซอฟต์แวร์เชิงบุคคล มาสร้างเป็นเครื่องมือที่ช่วยในเรื่องของการพัฒนากระบวนการพัฒนาซอฟต์แวร์ของนักพัฒนาซอฟต์แวร์ โดยมีฟังก์ชันการทำงานที่ประกอบไปด้วย เวลา ข้อผิดพลาด และขนาดซอฟต์แวร์ที่เป็นทั้งการประมาณการและข้อมูลที่เป็นจริง โดยทำในส่วนฟังก์ชันในการวางแผน การติดตามสมรรถนะของการพัฒนาซอฟต์แวร์ การวิเคราะห์ข้อมูล และการส่งออกรายงานผลการพัฒนา

3. ซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม

เนื้อหาส่วนนี้จะกล่าวถึงซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม รวมไปถึงการแสดงผลรายละเอียดในส่วนของการติดตามสมรรถนะการพัฒนาซอฟต์แวร์

3.1 มอดูลหรือฟังก์ชันการทำงานหลักของซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม

ฟังก์ชันการทำงานหลักของซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม ซึ่งประกอบไปด้วย การจัดการโครงการ การจัดการโปรแกรม การจัดการเกี่ยวกับเวลา การจัดการแสดงผลรายงาน

3.1.1 การจัดการโครงการ

ในส่วนนี้เป็นส่วนของการสร้างหรือกำหนดโปรเจกต์หรืองานที่ต้องการที่จะทำการพัฒนาโดยจะสามารถสร้างและเลือกชนิดของภาษาในโปรเจกต์.

3.1.2 การจัดการโปรแกรม

เกี่ยวข้องกับโปรแกรมที่จะทำการพัฒนาในโปรเจกต์ และเกี่ยวกับการเลือกประเภทของ กระบวนการพัฒนาซอฟต์แวร์ (PSP)

3.1.3 การจัดการเกี่ยวกับเวลา

ในส่วนนี้ผู้ใช้สามารถเลือกโปรแกรมที่ต้องการเพื่อนำไปจับเวลาการทำงานในเฟสต่าง ๆ และบันทึกข้อมูล สามารถบันทึกข้อมูลความผิดพลาดได้อีกด้วย

3.1.4 การจัดการแสดงผลรายงาน

ในส่วนนี้ผู้ใช้สามารถดูกราฟผลรวมของงาน กราฟแสดงข้อผิดพลาด กราฟการแสดงผล PROBE กราฟแสดงผลค่าที่ใช้ในการคำนวณ PROBE และกราฟแสดงเวลาที่ใช้ไปได้

3.2 การติดตามสมรรถนะการพัฒนาซอฟต์แวร์

ซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรมได้มีการพัฒนาการติดตามสมรรถนะการพัฒนาซอฟต์แวร์ เพื่ออำนวยความสะดวกให้ผู้ใช้ที่มีส่วนเกี่ยวข้องสามารถควบคุม และติดตามผลของบุคคลที่พัฒนาซอฟต์แวร์

โดยซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม จะทำการแสดงผลภาพรวมของเวลาที่ใช้ในการพัฒนาโปรเจกต์ ภาพรวมของจำนวนบรรทัดของโปรแกรม (LOC) ในโปรแกรมที่พัฒนาเปรียบเทียบกับกัน แสดงความถูกต้องในการประมาณการเวลาที่ใช้ในการพัฒนาซอฟต์แวร์ กับร้อยละของข้อผิดพลาด แสดงข้อบกพร่องที่เกิดขึ้นและถูกกำจัดของในแต่ละโปรแกรม แสดงจำนวนของข้อบกพร่องที่เกิดขึ้นต่อจำนวนบรรทัด 1000 บรรทัด ในเฟสคอมไพล์ และเฟสเทส

4. ผลการดำเนินโครงการ

ในการดำเนินโครงการพัฒนาซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม ได้ผลการดำเนินโครงการ ซึ่งประกอบไปด้วยลักษณะการพัฒนาเครื่องมือที่ใช้ในการพัฒนา รวมไปถึงผลลัพธ์ของการพัฒนา และผลการนำไปใช้จริง โดยมีรายละเอียดดังต่อไปนี้

4.1 ผลการพัฒนาซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม

ลักษณะการพัฒนาและเครื่องมือที่ใช้ รวมไปถึงผลลัพธ์ของการพัฒนาซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม มีรายละเอียดดังต่อไปนี้

ตารางที่ 1 ลักษณะเครื่องมือที่ใช้ในการพัฒนา

ลักษณะ/เครื่องมือ	ผลการดำเนินงาน
รูปแบบการพัฒนา	Web-base application
ภาษา	PHP 5.6
ฐานข้อมูล	MySQL 14.14
ระบบปฏิบัติการ	Debian 8.6

จากตารางที่ 1 จะเห็นได้ว่าการพัฒนาซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรมในรูปแบบของ Web-base application โดยมีภาษาที่ใช้ในการพัฒนาคือ ภาษา PHP เวอร์ชัน 5.4 ฐานข้อมูลที่ใช้คือ MySQL เวอร์ชัน 5.5 และทำการพัฒนาซอฟต์แวร์ ฯ บนระบบปฏิบัติการ Debian เวอร์ชัน 8.6

โปรแกรมของอัน				
Project Non-Project				
ลำดับ	ชื่อโครงการ	วันที่สร้าง	ภาษา	ดำเนินการ
1	คลังสินค้า 581607/1	06 กันยายน 2559	C++	ทำเสร็จแล้ว

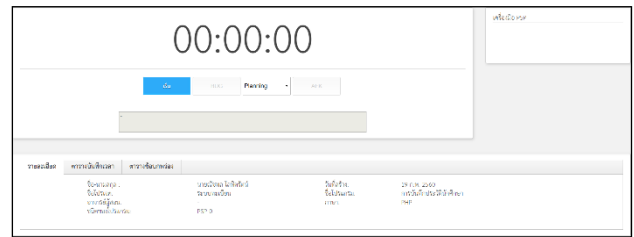
รูปที่ 3 หน้าจอหลักการจัดการโครงการ

รูปที่ 3 หน้าจอหลักการจัดการโครงการ แสดงตัวอย่างข้อมูลโครงการของนักพัฒนาซอฟต์แวร์

โปรแกรมของโปรแกรม คลังสินค้า				
ลำดับ	ชื่อโครงการ	วันที่ส่งมอบ	รูปแบบโครงการ	สถานะโครงการ
1	คลังสินค้า 581607/1	13 กันยายน 2559	PSP 0	เสร็จสิ้น
2	คลังสินค้า 581607/2	20 กันยายน 2559	PSP 0.1	เสร็จสิ้น
3	คลังสินค้า 581607/3	18 ตุลาคม 2559	PSP 1.0	เสร็จสิ้น
4	คลังสินค้า 581607/4	25 ตุลาคม 2559	PSP 1.1	เสร็จสิ้น
5	คลังสินค้า 581607/5	1 พฤศจิกายน 2559	PSP 2.0	เสร็จสิ้น
6	คลังสินค้า 581607/6	8 พฤศจิกายน 2559	PSP 2.1	เสร็จสิ้น

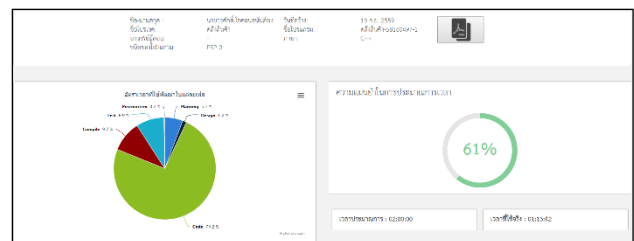
รูปที่ 4 หน้าจอหลักการจัดการโปรแกรม

รูปที่ 4 หน้าจอหลักการจัดการโปรแกรม แสดงข้อมูลโปรแกรมที่นักพัฒนาซอฟต์แวร์ที่ดำเนินการพัฒนาทั้งหมดที่ได้สร้างไว้ในโครงการ



รูปที่ 5 หน้าจอสำหรับกรบันทึกข้อมูลการพัฒนาซอฟต์แวร์

รูปที่ 5 จะแสดงรายละเอียดของข้อมูลในการพัฒนาซอฟต์แวร์ ได้แก่ ตารางบันทึกเวลา ตารางข้อผิดพลาด เครื่องมือ PSP



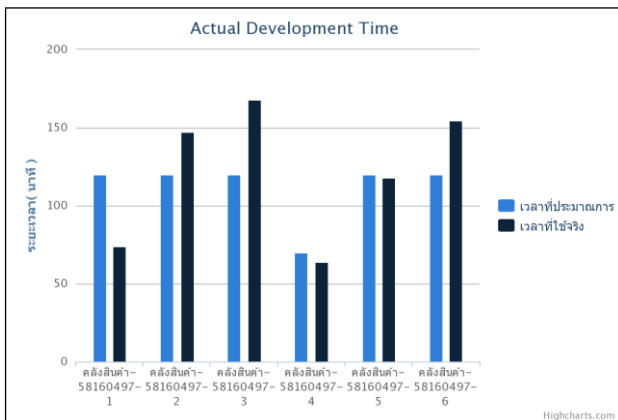
รูปที่ 6 หน้าจอแสดงการจัดการแสดงผลรายงาน

รูปที่ 6 จะแสดงรายงานผลการพัฒนาซอฟต์แวร์ ได้แก่ อัตราเวลาที่ใช้พัฒนาในแต่ละเฟส ความแม่นยำในการประมาณการเวลา เวลาประมาณการ เวลาที่ใช้จริง ขนาดของโปรแกรม

4.2 ผลการนำซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรมไปใช้จริง

การนำซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรมไปใช้ ได้มีการดำเนินการอบรมเชิงปฏิบัติการสารสนเทศให้นิสิตสาขาวิศวกรรมซอฟต์แวร์ชั้นปีที่ 2 ตัวแทนผู้เข้าอบรมเป็นนิสิต และคณาจารย์จำนวนมากกว่า 80 ท่าน โดยได้ทำการประเมินผลความพึงพอใจของผู้เข้าอบรม ที่มีต่อการใช้งานและประสิทธิภาพของซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม และผลสรุปจากตัวชี้วัดทั้งหมดอยู่ในเกณฑ์ ดี (4.2/5)

นอกจากนั้นยังมีผลการวิเคราะห์ข้อมูลของนิสิตตัวอย่างในเรื่องของการประมาณการเวลาในการพัฒนาซอฟต์แวร์ดังรูปที่ 7



รูปที่ 7 กราฟแสดงภาพรวมเรื่องของเวลา

รูปที่ 7 จะแสดงให้เห็นถึงเวลาในการประมาณการที่จะพัฒนาซอฟต์แวร์กับเวลาที่ใช้จริงในการพัฒนาซอฟต์แวร์

5. สรุปผลการนิพนธ์โครงการ

ผลการพัฒนาซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม ได้นำไปใช้ในการเรียนการสอนของคณะวิทยาการสารสนเทศ สาขาวิชาวิศวกรรมซอฟต์แวร์ ในรายวิชาการกระบวนการพัฒนาซอฟต์แวร์เชิงบุคคล ชั้นปีที่ 2 เห็นได้ชัดว่าซอฟต์แวร์ ๓ ได้เข้ามาช่วยในการบันทึกและจัดเก็บข้อมูลของรายวิชาการกระบวนการพัฒนาซอฟต์แวร์เชิงบุคคล เพื่อนำข้อมูลไปประมวลผลออกเป็นรายงาน เพื่อนำมาวิเคราะห์สำหรับการพัฒนาตนเอง โดยการพัฒนาซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม ได้เริ่มต้นพัฒนาโดยเป็นส่วนหนึ่งของรายวิชา กระบวนการพัฒนาซอฟต์แวร์เชิงกลุ่มงาน (Team Software Development Process) และได้พัฒนาต่อยอดในรายวิชาการการบริหารโครงการซอฟต์แวร์ (Software Project Management) ซึ่งในอนาคตของการพัฒนาซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม จะมีการปรับปรุงประสิทธิภาพในการทำงาน ได้แก่ การนับบรรทัดของโปรแกรมที่พัฒนาได้หลากหลายภาษามากยิ่งขึ้น และในส่วนของการติดตามสมรรถนะการพัฒนาซอฟต์แวร์ในภาพรวมได้อย่างมีประสิทธิภาพ

ในท้ายที่สุดนี้ขอขอบพระคุณทุกภาคส่วนที่เกี่ยวข้องกับการจัดทำโครงการในครั้งนี้ อันได้แก่ ทีมพัฒนาซอฟต์แวร์เพื่อการปรับปรุงตนเองเกี่ยวกับการเขียนโปรแกรม นักวิชาการคอมพิวเตอร์ห้องปฏิบัติการวิจัยวิศวกรรมระบบสารสนเทศ (Information System Engineering Research Laboratory :ISERL) เขตอุตสาหกรรมซอฟต์แวร์ภาคตะวันออก (Eastern Software Park : ESWP) และคณาจารย์ สาขาวิชาวิศวกรรมซอฟต์แวร์ คณะวิทยาการสารสนเทศ

เอกสารอ้างอิง

- [1] CodeIgniter คือ. สืบค้นเมื่อ 18 กุมภาพันธ์ 2560, จาก <http://www.codeigniter.in.th/>
- [2] Humphrey, Watts S. PSP: A Self-Improvement Process for Software Engineers. Boston, MA: Addison-Wesley. Publishers, 2005
- [3] Life Long Learning คือ. สืบค้นเมื่อ 18 กุมภาพันธ์ 2560, จาก <http://www.thailibrary.in.th/2015/04/01/lifelong-learning/>