

การใช้แนวคิด Agile Scrum ในการพัฒนาระบบสารสนเทศของมหาวิทยาลัยเชียงใหม่ The use of Agile Scrum methodology in the development of information systems at Chiang Mai University

ชื่อ-นามสกุลผู้แต่ง^{1*} จริยา ลิ้มจิระจรัส¹ และ วรินทิพย์ วิริยะนราทิพย์²

¹ชื่อหน่วยงานของผู้แต่งคนที่ 1 และ คนที่ 2 สำนักบริการเทคโนโลยีสารสนเทศ มหาวิทยาลัยเชียงใหม่

*ที่อยู่ E-mail jariya.lim@cmu.ac.th

1. แนวปฏิบัติที่ดีในด้าน

การพัฒนาซอฟต์แวร์ ที่เน้นความยืดหยุ่นในการทำงานร่วมกัน (cross-functional teams) และการส่งมอบงานที่มุ่งเน้นลูกค้า ช่วยให้ทีมสามารถจัดการกับโปรเจกต์ที่ซับซ้อน ปรับตัวตามข้อกำหนดที่เปลี่ยนแปลงได้รวดเร็ว สามารถส่งมอบซอฟต์แวร์คุณภาพสูงได้อย่างทันท่วงที

2. บทนำ/ที่มาและความสำคัญ

ฝ่ายพัฒนาระบบสารสนเทศ สำนักบริการเทคโนโลยีสารสนเทศ มหาวิทยาลัยเชียงใหม่ มีภารกิจหลักในการพัฒนาระบบสารสนเทศที่ใช้ในการบริหารจัดการมหาวิทยาลัย ภายใต้ระบบ “CMU MIS” ประกอบด้วย ระบบการเงิน ระบบแผนและงบประมาณ ระบบบริหารทรัพยากรบุคคล ระบบหลักสูตร ระบบบริหารงานวิจัย ระบบนักศึกษา ระบบประกันคุณภาพ เป็นต้น ได้เริ่มพัฒนาขึ้นตั้งแต่ปีงบประมาณ 2549 จวบจนปัจจุบัน รวมทั้งสิ้นมากกว่า 120 ระบบงาน โดยทีมนักพัฒนาระบบที่มีจำนวนจำกัด แต่เดิมการพัฒนาระบบใช้วิธีในรูปแบบ Waterfall ซึ่งมีข้อจำกัดหลายประการที่อาจส่งผลกระทบต่อประสิทธิภาพของโครงการ เช่น มีความยืดหยุ่นต่ำ การเปลี่ยนแปลงความต้องการในระหว่างกระบวนการพัฒนาทำได้ยากและมีค่าใช้จ่ายสูง เมื่อทีมพัฒนาส่งมอบงานไปแล้วก็จะไม่สามารถกลับไปแก้ไขหรือตอบสนองต่อข้อกำหนดใหม่ๆ ได้อย่างรวดเร็วง่ายดาย อีกทั้งการทดสอบระบบจะเกิดขึ้นในช่วงท้ายของกระบวนการพัฒนา ทำให้ข้อบกพร่องหรือปัญหาที่เกิดขึ้นในระหว่างการพัฒนาไม่ได้รับการแก้ไขอย่างทันท่วงที นำไปสู่การล่าช้าในการส่งมอบงาน การมีส่วนร่วมของเจ้าของระบบมีน้อย อาจมีเพียงช่วงต้นของกระบวนการ ซึ่งทำให้ผลิตภัณฑ์สุดท้ายไม่ตรงกับความต้องการหรือคาดหวัง ส่งผลให้เกิดความไม่พอใจในระยะยาว ด้วยข้อจำกัดเหล่านี้ โมเดล Waterfall อาจไม่เหมาะสมกับสภาพแวดล้อมที่มีการเปลี่ยนแปลงอย่างรวดเร็วหรือมีความไม่แน่นอนสูง ซึ่งเป็นเหตุผลที่ทำให้ฝ่ายพัฒนาระบบฯ มีการปรับเปลี่ยนรูปแบบการทำงานใหม่ให้สอดคล้องกับแนวทางการพัฒนาซอฟต์แวร์ที่ยืดหยุ่นมากขึ้น

ในปัจจุบันการใช้ Agile Scrum ในการพัฒนาซอฟต์แวร์ ช่วยแก้ไขปัญหที่เกิดขึ้นในกระบวนการพัฒนา โดยเฉพาะเรื่องการเปลี่ยนแปลงความต้องการของลูกค้า ซึ่งมักจะเกิดขึ้นบ่อยครั้ง ด้วยการจัดระเบียบการทำงานในรูปแบบ Sprint และการประชุมประจำวัน ทำให้ทีมพัฒนาระบบสามารถปรับตัวให้เข้ากับความต้องการใหม่ได้อย่างรวดเร็ว นอกจากนี้ การสื่อสารที่ชัดเจนระหว่างสมาชิกในทีมและเจ้าของงาน ช่วยลดปัญหาความเข้าใจผิดและทำให้ทุกฝ่ายมีความเข้าใจตรงกัน ช่วยลดระยะเวลาการส่งมอบระบบงาน สามารถส่งมอบฟีเจอร์ใหม่ในช่วงเวลาอันสั้น ทีมจึงสามารถให้ผลตอบกลับจากลูกค้าได้เร็วขึ้น อีกทั้งการทำงานร่วมกันในทีมช่วยเพิ่มความร่วมมือและลดความขัดแย้ง นำไปสู่การพัฒนาซอฟต์แวร์ที่มีคุณภาพสูงและตอบสนองความต้องการของเจ้าของงานได้ดียิ่งขึ้น

3. วัตถุประสงค์

1. เพื่อเพิ่มประสิทธิภาพในการพัฒนาระบบ ให้มีความยืดหยุ่นสามารถปรับตัวตามความต้องการของผู้ใช้ได้อย่างรวดเร็ว
2. เพื่อสร้างการสื่อสารที่ชัดเจนและมีประสิทธิภาพระหว่างทีมพัฒนาและผู้ใช้ระบบ
3. เพื่อส่งเสริมการทำงานร่วมกันและการแบ่งปันความรู้ระหว่างสมาชิกในทีม
4. เพื่อให้สามารถทดสอบและปรับปรุงระบบได้อย่างต่อเนื่อง ส่งผลให้มีคุณภาพที่ดียิ่งขึ้น

4. ทฤษฎีและงานวิจัยที่เกี่ยวข้อง (ถ้ามี)

4.1 แนวคิดของ Agile

1. การทำงานร่วมกับลูกค้า : การมีส่วนร่วมของลูกค้าตลอดกระบวนการพัฒนา เพื่อให้มั่นใจว่าผลิตภัณฑ์ตรงตามความต้องการของลูกค้า
2. ความยืดหยุ่นและการปรับตัว : การตอบสนองต่อการเปลี่ยนแปลงของข้อกำหนด แม้ว่าจะเกิดขึ้นในช่วงปลายของกระบวนการพัฒนา
3. การส่งมอบแบบค่อยเป็นค่อยไป : การส่งมอบและทดสอบได้บ่อยๆ ซึ่งช่วยให้สามารถตรวจพบปัญหาได้เร็วและปรับปรุงอย่างต่อเนื่อง
4. ทีมที่จัดการตัวเอง : การมอบอำนาจให้ทีมในการจัดการงานของตนเอง ตัดสินใจ และขับเคลื่อนการทำงานโดยไม่ต้องมีการควบคุมมากเกินไป

4.2 Agile Manifesto

1. บุคคลและการสื่อสาร มากกว่ากระบวนการและเครื่องมือ
2. ซอฟต์แวร์ที่ใช้งานได้จริง มากกว่าการจัดทำเอกสารอย่างละเอียด
3. การทำงานร่วมกับลูกค้า มากกว่าการเจรจาตามสัญญา
4. การตอบสนองต่อการเปลี่ยนแปลง มากกว่าการทำตามแผน

4.3 Scrum Framework

Scrum เป็นเฟรมเวิร์กที่นำหลักการ Agile มาประยุกต์ใช้ มีโครงสร้างที่ยืดหยุ่นและเป็นระบบสำหรับทีมในการทำงานร่วมกัน โดยจัดการงานในรูปแบบรอบหรือ Sprint ใช้เวลา 2-4 สัปดาห์

4.4 Key Roles in Scrum

1. Product Owner : ผู้รับผิดชอบในความต้องการของลูกค้า โดยมีหน้าที่กำหนดและจัดลำดับความสำคัญของงานใน Product Backlog
2. Scrum Master : ผู้ดูแลกระบวนการ Scrum ให้เป็นไปตามที่กำหนด และช่วยขจัดอุปสรรคที่อาจเกิดขึ้นระหว่างการทำงานของทีม
3. Development Team : ทีมพัฒนาที่ประกอบด้วยสมาชิกที่มีทักษะข้ามสายงาน รับผิดชอบในการส่งมอบผลิตภัณฑ์ตามข้อกำหนดที่ได้รับมอบหมาย

4.5 องค์ประกอบของ Scrum

1. Product Backlog : รายการของฟีเจอร์ ถูกจัดลำดับความสำคัญโดย Product Owner
2. Sprint Backlog : รายการ Product Backlog ที่เลือกมาทำใน Sprint
3. Increment : ผลลัพธ์ของงานที่ส่งมอบในแต่ละ Sprint

4.6 การประชุมใน Scrum

1. Sprint Planning : การประชุมวางแผนโดยทีมพัฒนาและ Product Owner จะร่วมกันเลือกงานจาก Product Backlog มาทำงานเพื่อให้เสร็จสิ้นภายใน Sprint
2. Daily Scrum : การประชุมประจำวัน (ปกติไม่เกิน 15 นาที) ซึ่งสมาชิกทีมจะรายงานถึงสิ่งที่ทำไปแล้ว สิ่งที่จะทำต่อไป และอุปสรรคที่พบเจอ
3. Sprint Review : การประชุมเพื่อแสดงผลงานที่เสร็จสิ้น เพื่อรับฟังความคิดเห็นและข้อเสนอแนะ
4. Sprint Retrospective : การประชุมทบทวนการทำงานของทีม เพื่อหาหรือถึงสิ่งที่ดีได้ สิ่งที่ต้องปรับปรุง และวิธีการทำงานใน Sprint ถัดไปให้มีประสิทธิภาพมากขึ้น

5. วิธีการ/กระบวนการ

ฝ่ายพัฒนาระบบสารสนเทศ สำนักบริการเทคโนโลยีสารสนเทศ ได้ปรับวิธีการทำงาน โดยอ้างอิงจากแนวคิดการทำงานแบบ Agile Scrum และมีการประยุกต์ใช้ให้เหมาะสมกับการทำงานของหน่วยงาน โดยมีการดำเนินการ ดังนี้

5.1 ศึกษาวิธีการทำงาน

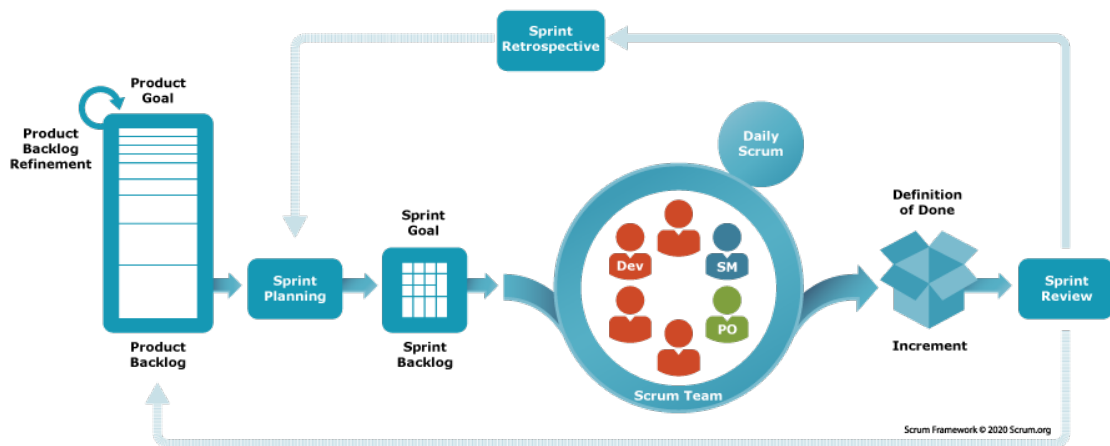
ศึกษาวิธีการทำงานแบบ Agile Scrum จากทฤษฎีต่างๆ อบรมและได้รับความรู้จากวิทยากรที่มีความเชี่ยวชาญ รวมถึงมีที่ปรึกษาเป็นผู้ที่มีความเชี่ยวชาญในการจัดการทีมได้อย่างมีประสิทธิภาพ เพื่อให้สามารถวิเคราะห์ปัญหาในการพัฒนาระบบ และสามารถนำวิธีการทำงานมาปรับใช้ให้เหมาะสมกับทีม

5.2 กำหนดบทบาทในการทำงาน เพื่อให้มีหน้าที่ในการทำงานอย่างชัดเจน ดังนี้

1. Business Analyst (BA) คือ นักวิเคราะห์ธุรกิจ เป็นผู้เก็บข้อมูลและทำความเข้าใจความต้องการของเจ้าของระบบ เพื่อนำมาวิเคราะห์หรือปรับปรุงวิธีการทำงานหรือกระบวนการขององค์กร และตอบโจทย์การแก้ไขปัญหาของเจ้าของระบบ โดยจะเป็นตัวกลางในการสื่อสารระหว่างทีมพัฒนาระบบและเจ้าของระบบ และออกแบบหน้าจอการใช้งานของเว็บไซต์หรือแอปพลิเคชัน โดยคำนึงถึงประสบการณ์ของผู้ใช้งาน และ ส่วนการเชื่อมประสานระหว่างผู้ใช้งานกับระบบ
2. ผู้พัฒนาระบบส่วนหน้าบ้าน (front end developer) เป็นผู้พัฒนาระบบส่วนการติดต่อกับผู้ใช้งาน ที่ผู้ใช้งานสามารถมองเห็นได้ รวมถึงความสวยงามของเว็บไซต์หรือแอปพลิเคชัน การใช้งานง่าย และความเหมาะสมของอุปกรณ์ที่ใช้งาน
3. ผู้พัฒนาระบบส่วนหลังบ้าน (back end developer) เป็นผู้พัฒนาระบบส่วนเบื้องหลังของเว็บไซต์หรือแอปพลิเคชัน ในการติดต่อกับฐานข้อมูล การประมวลผล ซึ่งมีการทำงานร่วมกันกับส่วนหน้าบ้าน

5.3 ปรับวิธีการทำงานของการพัฒนาระบบ

ปรับการทำงานเป็นแบบมีกรอบเวลา หรือ sprint โดยมีระยะเวลา 2 สัปดาห์ หรือ 10 วันทำการ เพื่อให้ได้ผลลัพธ์ของการพัฒนาระบบในระยะเวลาสั้น



ภาพที่ 1 แสดงกระบวนการวิธีการทำงานแบบ Scrum

1. ก่อนการทำงานใน sprint

- 1.1 เมื่อได้รับโครงการสำหรับการพัฒนาระบบ BA จะเป็นผู้เก็บข้อมูลและวิเคราะห์ความต้องการของระบบ ประเมินขอบเขตการทำงาน กระบวนการทำงาน และฟีเจอร์ของระบบงาน โดยทำงานร่วมกับเจ้าของระบบ
- 1.2 BA กำหนด เรื่องราวของผู้ใช้งาน หรือ user story ในแต่ละฟีเจอร์ของระบบงาน โดย 1 user story จะต้องพัฒนาเสร็จภายใน 1-2 วัน
- 1.3 BA ออกแบบหน้าจอและขั้นตอนการใช้งานของระบบเบื้องต้นก่อนนำเข้าสู่ sprint
- 1.4 กำหนดทีมในการพัฒนาระบบ ประกอบด้วย BA ประจำฟีเจอร์ของระบบงาน ผู้พัฒนาระบบส่วนหน้าบ้าน (front end developer) และ ผู้พัฒนาระบบส่วนหลังบ้าน (back end developer) โดยผู้พัฒนาระบบรวมกันทีมละไม่เกิน 5 คน และ กำหนด user story ที่จะพัฒนาระบบในแต่ละ sprint โดยคำนึงถึงจำนวนผู้พัฒนาระบบ และระยะเวลาการทำงานที่เพียงพอต่อเวลาในการพัฒนาของแต่ละทีม

2. การทำงานภายใน Sprint

- 2.1 วันที่ 1 ของ sprint เป็นวันเปิด sprint คือ การอธิบายภาพรวมของงานภายใน sprint ร่วมกัน วิเคราะห์งานและหน้าจอของระบบ กำหนดระยะเวลาในการพัฒนาระบบให้เพียงพอต่อเวลาใน Sprint ของแต่ละทีม โดยกำหนดงานย่อยในแต่ละ user story ตามการทำงาน ผู้พัฒนาระบบส่วนหน้าบ้าน (front end developer) และ ผู้พัฒนาระบบส่วนหลังบ้าน (back end developer) ภายในทีม
- 2.2 วันที่ 2 – 9 ของ sprint เป็นช่วงของการพัฒนาระบบ โดยจะมีการติดตามงานภายใน sprint และใช้เครื่องมือในการติดตามงานและการติดต่อสื่อสารกันระหว่างการทำงาน ซึ่งประกอบด้วย การออกแบบส่วนการพัฒนา การพัฒนาระบบ การทดสอบระบบ
- 2.3 วันที่ 8 ของ sprint ได้กำหนดร่วมกันให้มีการนำขึ้นระบบสำหรับทดสอบหรือ deploy งาน หรือ ระหว่างการพัฒนาระบบ หากผู้พัฒนาระบบมีการพัฒนาเสร็จ และนำขึ้นระบบสำหรับทดสอบ

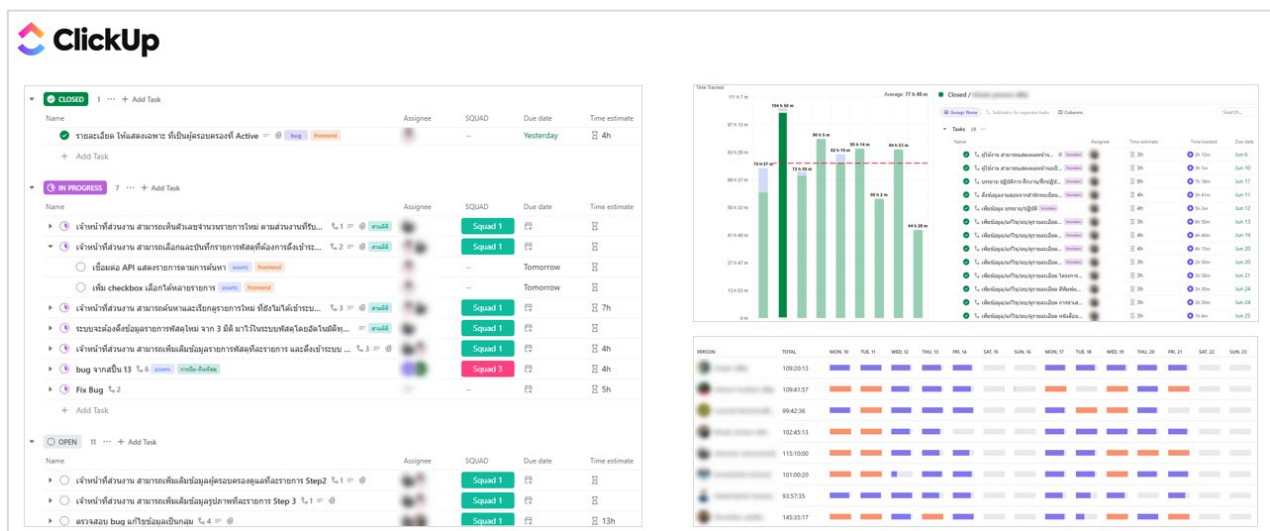
แล้ว ผู้พัฒนาระบบจะมีการแจ้ง BA ผ่านเครื่องมือในการติดตามงาน เพื่อให้ BA ได้ตรวจสอบตามฟังก์ชันและเงื่อนไขการทำงานของระบบให้ตรงตามความต้องการของลูกค้า

2.4 วันที่ 10 ของ sprint เป็นวันปิด sprint โดยจะมี 2 กิจกรรม คือ

- กิจกรรมที่ 1 Sprint Review คือ การนำเสนอผลงานที่พัฒนาเสร็จสิ้น โดยผู้พัฒนาระบบในแต่ละทีม จะมีการนำเสนองานที่ได้มีการพัฒนาระบบภายใน sprint และมีการให้ข้อคิดเห็นข้อเสนอแนะถึงการพัฒนาระบบ และแนวคิดในการออกแบบต่างๆ รวมถึงเทคนิคในการพัฒนาเพื่อให้ผู้พัฒนาในแต่ละทีมสามารถนำไปปรับใช้ในงานของตนเองได้
- กิจกรรมที่ 2 Sprint Retrospective คือ การประชุมทบทวนการทำงานของทีมงาน หรือสิ่งที่เกิดขึ้นภายในทีมทั้งสิ่งที่ทำได้ดี และสิ่งที่ควรปรับปรุง เพื่อให้มีการปรับปรุงการทำงานใน sprint ถัดไป

3. การนำเสนองานให้กับเจ้าของระบบ

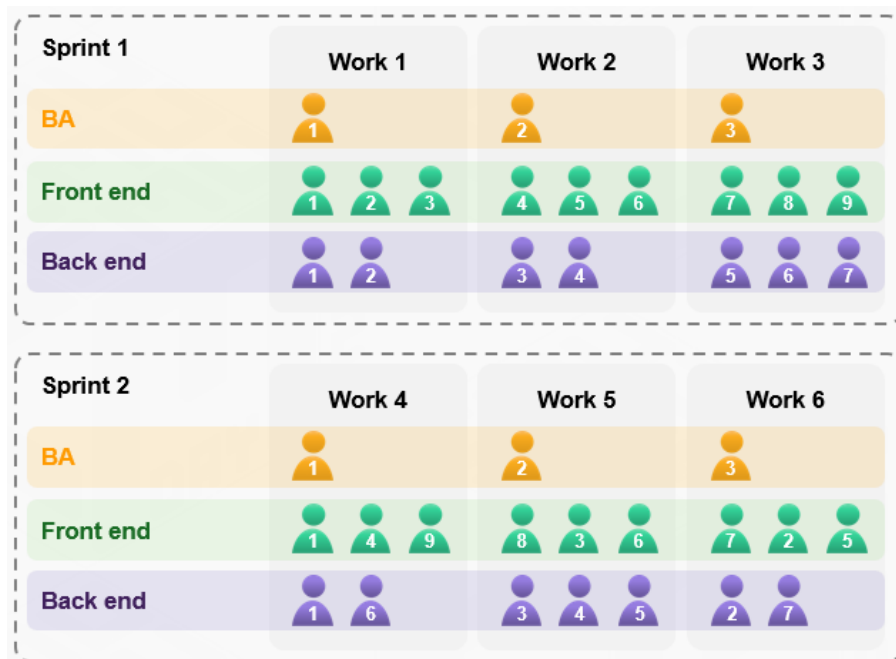
มีการนำเสนองานให้กับเจ้าของระบบ โดยรับฟังข้อคิดเห็น หากมีการปรับเปลี่ยน สามารถวิเคราะห์และพัฒนาระบบใน sprint ถัดไปได้ เพื่อให้ได้ระบบที่ตรงกับความต้องการมากที่สุด



ภาพที่ 2 แสดงเครื่องมือในการติดตามงาน

5.4 การเวียนทีมและเวียนงานในการทำงาน

โดยเมื่อครบเวลาการทำงานใน 1-2 sprint จะมีการปรับทีม เปลี่ยนผู้พัฒนาระบบในแต่ละงาน เพื่อให้ผู้พัฒนาระบบทุกคน ได้มีส่วนร่วมในการพัฒนาระบบในทุกงาน แต่ BA จะไม่เวียนระบบ



ภาพที่ 3 แสดงการเวียนทีมและเวียนงาน

5.5 การจัดการกับงานเมื่อมีงานพัฒนาหรือปรับปรุงระบบเข้ามาแบบเร่งด่วน

เนื่องจากฝ่ายพัฒนาระบบสารสนเทศ สำนักบริการเทคโนโลยีสารสนเทศ มีหน้าที่รับผิดชอบพัฒนาระบบสารสนเทศของมหาวิทยาลัย และมีระบบที่เปิดใช้งานในปัจจุบันจำนวนมาก ทำให้ในระหว่างการทำงานใน sprint ของทีมพัฒนาระบบจะมีงานเกี่ยวกับการพัฒนาหรือปรับปรุงระบบแบบเร่งด่วน โดยได้มีการปรับวิธีการทำงานดังนี้

1. มีการคัดกรองงานโดย BA ประจำทีมของผู้พัฒนาระบบ เพื่อประเมินงาน
2. หากเป็นงานที่จะต้องพัฒนาหรือปรับปรุงโดยเร่งด่วน BA ประจำทีมและผู้พัฒนาระบบในทีมจะประเมินงานและจำนวนวันใน sprint ที่เหลือ และ ประเมิน user story ปรับคนในการทำงาน หรือนำ user story ออกจาก sprint และปรับการพัฒนาระบบใน sprint ถัดไป
3. หากเป็นงานที่ไม่เร่งด่วน BA และ ผู้พัฒนาระบบที่เกี่ยวข้อง ร่วมกันวิเคราะห์และประเมินงาน เพื่อทำใน sprint ถัดไป

6. ผลกระทบที่เป็นประโยชน์หรือสร้างคุณค่า

การปรับปรุงพัฒนางาน

1. สามารถรับข้อคิดเห็นจากเจ้าของระบบและปรับปรุงระบบได้อย่างรวดเร็ว ได้ทันท่วงที ไม่ต้องรอจนจบโครงการ
2. มีการปรับปรุงพัฒนาระบบในฟีเจอร์การทำงานเมื่อมีการเปลี่ยนแปลงได้อย่างรวดเร็ว

การมีส่วนร่วมในการพัฒนา

1. เจ้าของระบบมีส่วนร่วมในการพัฒนาระบบมากขึ้น เพื่อให้ได้ระบบที่ตรงกับความต้องการมากที่สุด
2. ผู้พัฒนาระบบทุกคนมีส่วนร่วมในการออกแบบและพัฒนาาร่วมกันในทุกฟีเจอร์การทำงาน

การพัฒนาบุคลากร

1. การเขียนทีมและเขียนงาน ทำให้มีการแลกเปลี่ยนเรียนรู้เทคนิคต่างๆ ซึ่งกันและกัน และสามารถนำไปปรับใช้ในงานของตนเองได้

2. ได้ฝึกพัฒนาทักษะในการประเมินงานและระยะเวลาการทำงานของทีม

การทำงานทดแทน การพัฒนาระบบไม่ยึดติดกับผู้พัฒนาระบบเพียงคนเดียวคนหนึ่ง สามารถทำงานแทนกันได้ทุกคน

ตารางเปรียบเทียบการพัฒนาระบบแบบ Waterfall และ Agile

การทำงาน	การพัฒนาระบบแบบ Waterfall	การพัฒนาระบบตามแนวคิดแบบ Agile
1. การวิเคราะห์ความต้องการของระบบ	จะต้องวิเคราะห์ทั้งหมด จึงจะลงมือพัฒนาระบบได้	วิเคราะห์ภาพรวม และเจาะลึกในแต่ละเฟสเฟเจอร์การทำงานในแต่ละ sprint
2. การมีส่วนร่วมในการทำงาน	เจ้าของงานมีส่วนร่วมในการให้ข้อมูลความต้องการในระยะเริ่มต้น และตรวจรับงานเมื่อสิ้นสุดโครงการ	เจ้าของงานมีส่วนร่วมกับการพัฒนาในทุกขั้นตอนของการพัฒนา
3. การพัฒนาระบบ	จะต้องมีการพัฒนาระบบให้ครบ จึงจะทดสอบการใช้งาน	มีการออกแบบ พัฒนาระบบ และทดสอบการใช้งานในแต่ละเฟสเฟเจอร์การทำงานภายใน sprint

7. แผนการดำเนินงาน / การนำไปใช้

ฝ่ายพัฒนาระบบสารสนเทศ สำนักบริการเทคโนโลยีสารสนเทศ ได้นำแนวคิด Agile Scrum นี้มาใช้ในการกระบวนการทำงานตั้งแต่เดือนตุลาคม 2565 เป็นต้นมา ซึ่งในระยะแรก ได้มีการแบ่งบทบาทการทำงาน และฝึกอบรมบุคลากรให้เข้าใจพื้นฐานในการพัฒนาระบบให้เป็นรูปแบบเดียวกัน โดยระบบแรกที่น่าวิธีการทำงานใหม่นี้มาประยุกต์ใช้คือระบบ CMUIDP ของสายปฏิบัติการ ซึ่งทีมพัฒนาระบบประกอบด้วยนักพัฒนาระบบ 16 คน และนักวิเคราะห์เชิงธุรกิจ 4 คน โดยมีการเรียนรู้วิธีการทำงานควบคู่ไปกับการทำงาน ซึ่งต้องมีการจัดการแบ่งเวลาในการศึกษาการใช้งานเครื่องมือและเทคโนโลยี รวมถึงปรับทักษะของบุคลากรให้มีความสามารถใกล้เคียงกันด้วยวิธีการแชร์องค์ความรู้ร่วมกัน การจับคู่ในการเขียนโปรแกรม การเวียนกันตรวจสอบความถูกต้องของโค้ดและเทคนิคการเขียนโปรแกรม มีการปรึกษาผู้เชี่ยวชาญและผู้บริหารมีการให้การสนับสนุนอย่างมาก ไม่ว่าจะเป็นทั้งในด้านกระบวนการทำงาน ด้วยแนวคิด Agile Scrum มาใช้ในการทำงาน และนำมาปรับใช้ให้เหมาะสมกับธรรมชาติของฝ่ายพัฒนาระบบสารสนเทศ

หลังจากนั้น เมื่อทีมพัฒนาที่มีความคุ้นชินกับการพัฒนาในแนวคิด Agile Scrum แล้ว ทุกโครงการที่ได้รับมอบหมาย ไม่ว่าจะเป็นการปรับปรุงระบบเดิม หรือการพัฒนาระบบโครงการใหม่ จะมีการใช้แนวคิดนี้ในการทำงาน โดยจะมีวิธีการและกระบวนการทำงานดังกล่าวมาข้างต้น และมีการปรับใช้และเรียนรู้ร่วมกันมาโดยตลอด เช่น การปรับกระบวนการในการติดตามงาน การแชร์วิธีการพัฒนาระบบ เป็นต้น เพื่อให้ทีมสามารถปรับตัวเมื่อมีการเปลี่ยนแปลงทีมในการทำงานได้อยู่เสมอ

8. วิเคราะห์และสรุปสาระสำคัญ

ปัจจัยแห่งความสำเร็จในการประยุกต์ใช้แนวคิด Agile Scrum ที่จะช่วยให้การพัฒนาซอฟต์แวร์มีประสิทธิภาพ มีหลายประการที่ควรคำนึงถึงดังนี้

8.1 การสนับสนุนจากผู้บริหาร

ผู้บริหารมีอำนาจในการตัดสินใจและสั่งการอย่างมีประสิทธิภาพ พร้อมทั้งสนับสนุนให้ทีมสามารถขับเคลื่อนการทำงานตามแนวทาง Agile Scrum ได้อย่างเต็มที่ ไม่ว่าจะเป็นในด้านกระบวนการทำงาน วิธีการร่วมมือภายในทีม การสร้างสภาพแวดล้อมที่ยืดหยุ่นและเหมาะสม วัฒนธรรมการทำงานที่เน้นความร่วมมือและนวัตกรรม รวมถึงการจัดสรรงบประมาณสำหรับเครื่องมือและเทคโนโลยีที่จำเป็น อีกทั้งยังควรมีการจัดการฝึกอบรมและพัฒนาทักษะของบุคลากรให้สอดคล้องกับตำแหน่งงานอย่างต่อเนื่อง เพื่อให้ทีมสามารถทำงานได้อย่างมีประสิทธิภาพและตอบสนองความต้องการขององค์กรได้อย่างสูงสุด

8.2 การจัดการโครงการ

การจัดลำดับความสำคัญของงานเป็นสิ่งจำเป็น โดยต้องปรับให้สอดคล้องกับความต้องการของลูกค้า มีการประเมินภาระงานที่เหมาะสมกับกำลังคนที่มีอยู่ และสอดคล้องกับรอบระยะเวลาการพัฒนา นอกจากนี้ ควรมีการติดตามและตรวจสอบความคืบหน้าของงานอย่างสม่ำเสมอ เพื่อให้แน่ใจว่าทุกกระบวนการดำเนินไปตามแผนที่วางไว้ และสามารถปรับปรุงแก้ไขได้ทันเวลา

8.3 การสื่อสารและการทำงานร่วมกันเป็นทีม

การมอบอิสระให้ทีมสามารถจัดการตัวเองได้เป็นกุญแจสำคัญในการดำเนินงานอย่างมีประสิทธิภาพ การสร้างบรรยากาศที่เต็มไปด้วยความเชื่อมั่นและการสนับสนุนซึ่งกันและกัน จะช่วยให้สมาชิกทีมทำงานร่วมกันได้อย่างราบรื่น การสนับสนุนให้เกิดการสนทนาโดยตรง เช่น การประชุม Daily Standup Meeting การใช้ Task Board และการทบทวนงานผ่าน Retrospective จะส่งผลให้เกิดความเข้าใจที่ชัดเจนและสามารถแก้ไขปัญหาได้ทันที่ อีกทั้งการแลกเปลี่ยนความรู้เชิงเทคนิคอย่างต่อเนื่องภายในทีม ช่วยให้กระบวนการทำงานมีการปรับปรุงอย่างต่อเนื่องและเป็นไปในทิศทางที่สอดคล้องกับเป้าหมายของลูกค้า ทำให้สามารถส่งมอบผลงานที่มีคุณภาพได้อย่างมีประสิทธิภาพตลอดเวลา

8.4 การส่งมอบงานอย่างต่อเนื่อง

เพื่อให้การทำงานของทีมเป็นไปอย่างมีประสิทธิภาพและมั่นคง ควรมีการตรวจสอบผลการดำเนินงานอย่างต่อเนื่อง โดยการใช้เครื่องมือที่เหมาะสมในการประเมินคุณภาพการพัฒนาซอฟต์แวร์ เช่น การทดสอบ (Test) การตรวจสอบโค้ด (Code Review) และการตรวจหาช่องโหว่ (Vulnerability Scanning) เพื่อให้มั่นใจว่าระบบมีความปลอดภัยและพร้อมใช้งาน ก่อนที่จะส่งมอบให้กับลูกค้า โดยการส่งมอบงานควรส่งมอบในระยะเวลาที่สม่ำเสมอ จะช่วยลดความเสี่ยง และเปิดโอกาสให้ลูกค้าตรวจสอบระบบ พร้อมทั้งให้ข้อเสนอแนะได้ทันที ซึ่งส่งผลให้ทีมสามารถปรับปรุงงานและตอบสนองต่อการเปลี่ยนแปลงหรือความต้องการใหม่ๆ ของลูกค้าได้อย่างรวดเร็วและมีประสิทธิภาพ

9. เอกสารอ้างอิง

- ณัฐณิธิย์ ศรีมาเสริม และ ญาดา เอื้อสมิทธิ์. (2565). การประยุกต์ใช้แนวคิดการทำงานแบบ Agile เพื่อการพัฒนากระบวนการ: กรณีศึกษาสำนักงานผู้อำนวยการสถาบันชีววิทยาศาสตร์โมเลกุล มหาวิทยาลัยมหิดล. วารสารสังคมศาสตร์บูรณาการ มหาวิทยาลัยมหิดล , 9(2), 193–214.
- นิภาพร สุวัฒนนิช. (2562). แนวทางการใช้ Agile ในองค์กรให้ประสบความสำเร็จของธนาคารแห่งหนึ่ง. ค้นเมื่อวันที่ 28 ตุลาคม 2567, จาก <https://brandinside.asia/agile-and-scrum-for-new-business/https://archive.cm.mahidol.ac.th/bitstream/123456789/3066/1/TP%20MS.011%202562.pdf>