

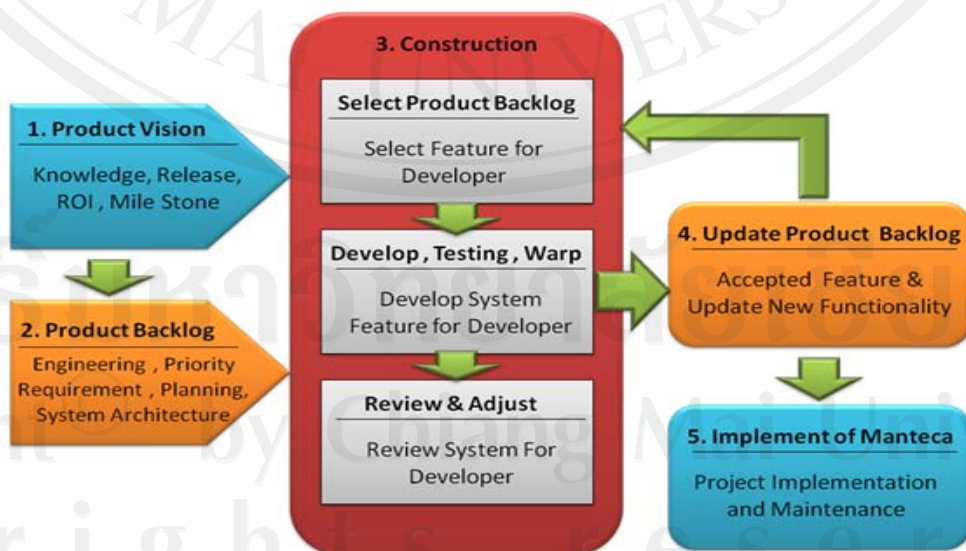
บทที่ 3

วิธีการศึกษา

วิธีการศึกษาการพัฒนาซอฟต์แวร์ตามงานวิจัยด้วยการใช้สถาปัตยกรรมการขับเคลื่อนด้วยแบบจำลองจะแบ่งออกเป็น 7 ขั้นตอน ดังต่อไปนี้

- 1) ศึกษาวิธีการพัฒนาซอฟต์แวร์ด้วยการใช้สถาปัตยกรรมการขับเคลื่อนด้วยแบบจำลอง
- 2) ศึกษาเครื่องมือที่ใช้ ในการพัฒนาซอฟต์แวร์ด้วยการใช้สถาปัตยกรรมการขับเคลื่อนด้วยแบบจำลอง
- 3) ศึกษาทำความเข้าใจระบบติดตามงานวิจัยและกำหนดวิสัยทัศน์ของการพัฒนาระบบ
- 4) เก็บรวบรวมและวิเคราะห์ความต้องการของระบบ
- 5) พัฒนาระบบ
- 6) รวบรวมรายละเอียดงานสำหรับพัฒนาเพิ่มเติม
- 7) นำระบบงานไปใช้และการบำรุงรักษา

สำหรับแนวทางการวิจัยเพื่อให้การค้นคว้าเป็นไปอย่างมีระเบียบหลักการ ผู้ค้นคว้าจึงเลือกพัฒนาซอฟต์แวร์ตามหลักการของเอจายล์(Agile) โดยเลือกใช้วิธีการพัฒนาซอฟต์แวร์แบบสครัม (Scrum Methodology) มาเป็นแนวทางในการพัฒนาอย่างเป็นแบบแผน ซึ่งเป็นกระบวนการในข้อที่ 3 – 7 ที่ได้กล่าวมาแล้วข้างต้น และแสดงให้เห็นดังรูปที่ 3.1



รูปที่ 3.1 แสดงลำดับขั้นตอนการค้นคว้าและพัฒนาตามหลักการเอจายล์ด้วยวิธีสครัม

วิธีการพัฒนาซอฟต์แวร์แบบสครัมได้มุ่งเน้นให้มีการวางแผนการทำงานการพัฒนาและพูดคุยกับผู้ใช้งานอย่างต่อเนื่องเพื่อให้มีการปรับแก้ระบบ ก่อนส่งมอบผลงานโดยเน้นความพึงพอใจของลูกค้าเป็นสำคัญ ทั้งนี้ได้กำหนดให้มีการพัฒนาระบบติดตามงานวิจัยโดยใช้สถาปัตยกรรมการขับเคลื่อนด้วยแบบจำลอง มีวิธีการทำงานดังต่อไปนี้

3.1 ศึกษาวิธีการพัฒนาซอฟต์แวร์ด้วยการใช้สถาปัตยกรรมการขับเคลื่อนด้วยแบบจำลอง

ศึกษาวิธีการพัฒนาซอฟต์แวร์ด้วยสถาปัตยกรรมการขับเคลื่อนด้วยแบบจำลองหรือเอ็มดีเอ จำเป็นต้องทราบถึงขั้นตอนและวิธีการพัฒนาต่างๆ ซึ่งประกอบไปด้วยขั้นตอนดังนี้

- 1) ขั้นตอนการเก็บความต้องการของระบบ (Requirement)
- 2) ขั้นตอนการวิเคราะห์ระบบ (Analysis)
- 3) ขั้นตอนการออกแบบระบบ (Design)
- 4) ขั้นตอนการพัฒนาระบบ (Construction)
- 5) ขั้นตอนการทดสอบระบบ (Test)

3.2 ศึกษาเครื่องมือที่ใช้ในการพัฒนาซอฟต์แวร์โดยใช้สถาปัตยกรรมขับเคลื่อนด้วยแบบจำลอง

การพัฒนาซอฟต์แวร์ด้วยการใช้สถาปัตยกรรมการขับเคลื่อนด้วยแบบจำลอง เพื่อให้ได้ประสิทธิภาพมากที่สุด จำเป็นต้องใช้เครื่องมือช่วยเพื่อแปลงรูปของโมเดลไปยังโมเดล หรือจากโมเดลไปเป็นโค้ด โดยในปัจจุบันมีซอฟต์แวร์ช่วยมากมาย แต่สำหรับการวิจัยนี้ได้เลือกเครื่องมืออีซีโอ 6.0(ECO 6.0)เป็นเครื่องมือสำหรับการศึกษาค้นคว้า ประกอบไปด้วยสิ่งต่างๆ ดังนี้

- 1) ศึกษาการออกแบบโดเมนโมเดล (Domain Modeling)
- 2) ศึกษาการออกแบบโครงสร้างโมเดล (Architecture Modeling)
- 3) ศึกษาการออกแบบซอฟต์แวร์ประยุกต์โมเดล (Application Modeling)
- 4) ศึกษาโมเดลของโค้ดโปรแกรม (Code Modeling)

3.3 กำหนดวิสัยทัศน์ของการพัฒนาระบบ

ขั้นตอนการกำหนดวิสัยทัศน์ของการพัฒนาระบบจะเป็นขั้นตอนการกำหนดแนวทางและเป้าหมายการพัฒนาระบบติดตามงานวิจัยโดยใช้สถาปัตยกรรมการขับเคลื่อนด้วยแบบจำลอง ทั้งนี้เพื่อให้ทีมงานพัฒนาที่มีความเข้าใจและปฏิบัติตามกระบวนการพัฒนาได้อย่างถูกต้องและเหมาะสม ได้มีการกำหนดเป้าหมายการพัฒนาระบบ (Mile Stone)เพื่อให้ทราบถึงขอบเขตโดยรวมด้านต่าง ๆ

เช่น ขอบเขตด้านเวลา ต้นทุนการพัฒนา และทรัพยากรที่จำเป็นในการพัฒนา ศึกษาข้อมูลที่เกี่ยวข้องกับการพัฒนาเพื่อทำความเข้าใจในลักษณะต่อสิ่งที่เกี่ยวข้องต่อระบบทั้งหมด สรุปความเป็นไปได้ในการพัฒนาระบบ โดยที่สำคัญคือการศึกษาขั้นตอนการทำงานการติดตามงานวิจัยที่ได้ปฏิบัติอยู่ในขณะนี้ เพื่อนำไปใช้ในขั้นตอนการออกแบบ

3.4 เก็บรวบรวมและวิเคราะห์ความต้องการของระบบ

การเก็บรวบรวมและวิเคราะห์ความต้องการของระบบ จะดำเนินการตามกระบวนการวิศวกรรมซอฟต์แวร์ (Software Engineering) ในส่วนของการจัดเก็บความต้องการ การวิเคราะห์ความต้องการ และการวางแผนการพัฒนาระบบ ได้มีขั้นตอนการทำงานดังต่อไปนี้

1) รวบรวมความต้องการของระบบ (Requirement Elicitation)

จะครอบคลุมกระบวนการวางแผนเพื่อจัดเก็บข้อมูล และวิธีการบันทึกจัดเก็บข้อมูลซึ่งแบ่งรูปข้อมูลในการจัดเก็บแบ่งเป็น ข้อมูลที่ได้จากการสัมภาษณ์ ข้อมูลในรูปแบบเอกสารการบันทึกข้อมูล แบบฟอร์มสอบถามข้อมูล โดยมีกลุ่มผู้ที่เกี่ยวข้องกับระบบงานออกเป็นพนักงานระดับปฏิบัติงาน ผู้บริหาร และคณะทำงานติดตามงานวิจัย โดยมีการวางแผนการสัมภาษณ์ถึงกระบวนการทำงานในปัจจุบัน รวมไปถึงรายงานสารสนเทศต่างๆ ที่ผู้บริหารต้องการ

2) วิเคราะห์ความต้องการ (Requirement Analysis)

เมื่อรวบรวมข้อมูลความต้องการของระบบมาแล้ว ก็ทำการวิเคราะห์ความต้องการจากการศึกษาและการเก็บข้อมูลความต้องการ โดยให้สอดคล้องกับการใช้งานจริงของระบบการบริหารงานพัฒนาโปรแกรมเพื่อตรวจสอบความถูกต้องและจำแนกความต้องการออกเป็นส่วนๆ เพื่อเชื่อมโยงความสัมพันธ์ระหว่างความต้องการของระบบที่ใช้ขั้นตอนกลไกหรือออกแบบระบบ เพื่อวิเคราะห์หาสิ่งต่างๆ ดังนี้

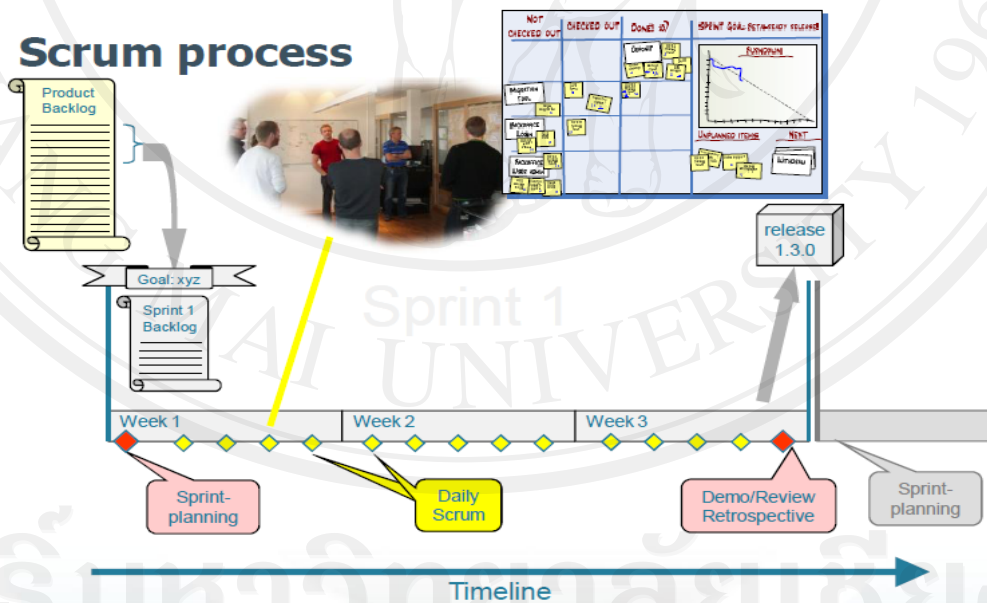
- วิเคราะห์ฟังก์ชันงานที่จำเป็นต่อผู้ใช้และกำหนดลำดับความสำคัญของความต้องการ (Priority) จากการวิเคราะห์ความต้องการที่ได้
- วิเคราะห์ระบบงานย่อย (Sub System) เพื่อแตกย่อยสำหรับการพัฒนาแบบโมดูล (Module)

3) ออกแบบระบบ (System Design)

ในขั้นต้นทำการออกแบบระบบเพื่อให้เห็นภาพการทำงาน และผู้ที่เกี่ยวข้องกับระบบโดยทำการออกแบบระบบตามการออกแบบซอฟต์แวร์เชิงวัตถุด้วยยูเอ็มแอล (UML) เฉพาะยูสเคสไดอะแกรม (Use Case Diagram)

4) วางแผนการพัฒนา (Project Planning)

การวางแผนหรือการกำหนดอนาคตล่วงหน้าเกี่ยวกับภารกิจหรือกิจกรรมต่างๆ ที่จะกระทำให้บรรลุวัตถุประสงค์ที่ต้องการ ซึ่งในวิธีการสครัมนี้จะเน้นเกี่ยวกับการวางแผนการพัฒนาตามกำหนดการนำระบบออกมาทดสอบ (Release) ด้วยการจัดลำดับความสำคัญของกิจกรรม ระยะเวลา และทรัพยากรที่ใช้ในการพัฒนาระบบ สำหรับขั้นตอนการวางแผนพัฒนาจะมีการปรับปรุงเป็นระยะตามหลักการพัฒนาแบบสครัม (Scrum Methodology) เพื่อปรับปรุงแผนการพัฒนาดังรูปที่ 3.3แสดงกระบวนการทำงานของการวางแผนการพัฒนาซอฟต์แวร์



รูปที่ 3.2 แสดงกระบวนการทำงานของการพัฒนาซอฟต์แวร์ด้วยวิธีสครัม

(<http://www.crisp.se/henrik.kniberg>)

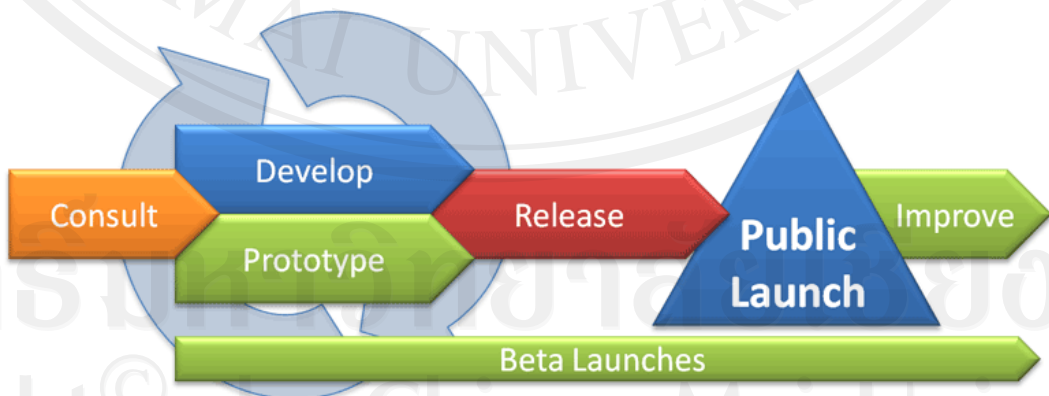
3.5 พัฒนาระบบ

ในการพัฒนาระบบติดตามงานวิจัย เพื่อให้สอดคล้องกับ วิธีการพัฒนาซอฟต์แวร์แบบ สกิร์ม (Scrum Methodology) ให้ได้ตามข้อกำหนดที่ได้ออกแบบไว้ โดยอ้างอิงจากการออกแบบ ยูเอ็มแอล และทดสอบความถูกต้องของระบบในระหว่างพัฒนาไปด้วยอย่างสม่ำเสมอและประกอบ ซอฟต์แวร์เข้าด้วยกัน เพื่อให้ได้ซอฟต์แวร์ที่สามารถใช้งานได้และสอดคล้องกับวัตถุประสงค์ จะ ออกแบบให้เป็นระบบย่อยและวางแผนตามระบบงานย่อย โดยแบ่งลำดับการทำงานได้ดังนี้

1. **ประชุมและเลือกฟังก์ชันงาน (Checkout Product Backlog)** เพื่อพัฒนาความสามารถในแต่ละรุ่นของระบบ (Release)

2. **พัฒนา ทดสอบ และจัดทำแพ็คเกจ(Development, Test)** ในการพัฒนาจะทำตาม ฟังก์ชันงานที่เลือกตามการออกแบบ และควบคุมคุณภาพการพัฒนาด้วยการทดสอบระบบ ซึ่งจะทำการทดสอบในระดับของการทดสอบหน่วยย่อย (Unit Test) และทำในระดับของการทดสอบรวม (Integration Test) ให้สามารถทำงานร่วมกันทั้งหมดได้

3. **นำเสนอผลงาน(Presentation)** สำหรับผู้ใช้งานระบบและปรับแก้ ร่วมกับผู้ใช้งานก่อนทำการยอมรับคุณสมบัติของระบบ (Accepted Features) ก่อนเข้าไปสู่การพัฒนาในระบบในวงรอบการพัฒนาต่อไป (Next Release) โดยเน้นให้มีการเห็นความคืบหน้าและหน้าตาของงาน กับ ความสัมพันธ์กับลูกค้าอยู่ตลอดเวลา จึงเน้นในส่วนของการแสดงผลของตัวอย่างงาน (Prototype) ให้กับผู้ใช้งานหรือผู้ช่วยเบหาวานได้ทดลองใช้งานเป็นระยะ ๆ พร้อมรองรับความเปลี่ยนแปลงที่ ผู้ใช้ต้องการได้ ตามหลักการพัฒนาซอฟต์แวร์ของเอจายล์ โดยมีวิธีการทำงานดังรูป 3.4



รูปที่ 3.3 วิธีการพัฒนาซอฟต์แวร์ด้วยวิธีสครัมของ Agile Software Development

<http://www.nochallenge.net/about.html>

3.6 รวบรวมรายละเอียดงานสำหรับพัฒนาระบบเพิ่มเติม (Update Product Backlog)

เนื่องจากวิธีการพัฒนาระบบอ้างอิงจาก วิธีการพัฒนาแบบสครัม (Scrum Methodology) ดังนั้นจะต้องมีการรวบรวมรายละเอียดงานสำหรับพัฒนาระบบเพิ่มเติมอย่างต่อเนื่อง (Update Product Backlog) จากการพัฒนาในขั้นตอนการพัฒนาระบบ (3.5) ซึ่งในกรณีที่ยังมีรายละเอียดงานคงเหลือ และจำเป็นต้องการทำงานของระบบ จะต้องย้อนกลับไปทำในขั้นตอนการพัฒนาระบบ (3.5) จนครบตามเป้าหมายโครงการ

3.7 นำระบบไปใช้งานและบำรุงรักษาระบบ (Implement and Maintenance)

เมื่อพัฒนาระบบครบตามคุณสมบัติที่กำหนดไว้จะเข้าสู่กระบวนการ นำระบบไปใช้งาน ร่วมกับข้อมูลจริงและการบำรุงรักษาระบบ

- 1) การนำระบบไปใช้งานร่วมกับข้อมูลจริง (Implementation)
- 2) จัดทำคู่มือ แผนการบำรุงรักษา และตรวจสอบระบบ เพื่อลดความผิดพลาดที่จะเกิดขึ้นกับระบบ