

2110725 Software Engineering Process and Improvement

Workshop 1: Software development life cycle selection and recommendation for adaptation

1. หน่วยงานหนึ่งมีความรับผิดชอบในหนึ่งโครงการ กำหนดให้ลักษณะของโครงการมีวัตถุประสงค์เพื่อต้องการ **พัฒนา web-based development สำหรับ 2 ระบบงาน**คือ “1. ระบบการนัดหมายผู้ป่วยนอก และ 2. ระบบการบันทึกการรักษาโดยแพทย์” ให้กับโรงพยาบาลเอกชนขนาดเล็กแห่งหนึ่ง มีระยะเวลาในการพัฒนาระบบ 6 เดือน
2. ข้อมูลเกี่ยวกับหน่วยงาน
 - 2.1 หน่วยงานนี้เคยมีประสบการณ์ดังนี้
 - 1) ในการพัฒนาด้าน web-based application มา ประมาณ 5 ปี จำนวน 10 แอปพลิเคชัน
 - 2) ทีมงานมีประสบการณ์ในการพัฒนาและการใช้เทคโนโลยีเพื่อการพัฒนา web-based application ในระดับปานกลางถึงค่อนข้างมาก
 - 3) หน่วยงานนี้มี **software development life cycle model ที่หลากหลาย** ให้หัวหน้าโครงการเป็นผู้เลือกได้เองตามความเหมาะสมของ project characteristic
 - 2.2 ปัญหาที่ผ่านมาของหน่วยงานนี้ที่มีก่ประสบคือ
 - 1) **ด้านความต้องการจากผู้ใช้** ผู้ใช้งานมักจะเปลี่ยนแปลงรายละเอียดของความต้องการ และไม่สามารถส่งมอบงานได้ตามกำหนด (มักจะส่งงานล่าช้าไปประมาณ 15% ของระยะเวลาที่ประมาณการณไว้)
 - 2) **ด้านคุณภาพของงาน** มีบางส่วนไม่ตรงกับความคาดหวังของลูกค้าในการใช้งานในสภาพแวดล้อมของการใช้งานจริง ทำให้ต้องเสียเวลาและ effort ในปรับแก้ไข
3. สำหรับโครงการนี้กำหนดให้มีทีมงานประกอบด้วย senior software engineer 2 คน, junior software engineer 4 คน และ project manager 1 คน

งตอบคำถามต่อไปนี้

1. จากข้อมูลของหน่วยงาน ประเภทของ application ที่ต้องการพัฒนา และปัญหาที่ผ่านมาของหน่วยงานนี้ที่มีก่ประสบ จงระบุ Software development life cycle model (SDLCM) มา 1 process model ที่ทีมคิดว่าเหมาะสมกับโครงการนี้ พร้อมทั้งระบุเหตุผลที่เลือกมาอย่างน้อย 2 รายการ **ทั้งนี้ניתสามารถกำหนดข้อสมมุติฐานเพิ่มเติมได้**

| Software development life cycle model (SDLCM) ที่เลือก | เหตุผล |
|--|--|
| Agile (Scrum) | 1. รองรับการเปลี่ยนแปลงของความต้องการผู้ใช้ได้ดี: จากปัญหาที่หน่วยงานเคยประสบ เนื่องจากผู้ใช้เปลี่ยนแปลงความต้องการบ่อย Scrum เน้นการพัฒนาเป็น Sprint และมีการทำ Sprint Review รับ Feedback เป็นระยะ ๆ ในระหว่างการ Implementation ทำให้สามารถปรับเปลี่ยนฟีเจอร์หรือปรับปรุงฟังก์ชันได้ทันที |
| | 2. ลดความเสี่ยงเรื่องคุณภาพของซอฟต์แวร์: Scrum มีแนวทางการทำงานที่ส่งมอบซอฟต์แวร์เป็นระยะ (Incremental development) ซึ่งช่วยให้สามารถตรวจสอบคุณภาพและทดสอบระบบในสภาพแวดล้อมจริงได้บ่อยครั้ง ป้องกันปัญหาที่เกิดขึ้นหลังจากการพัฒนาเสร็จสิ้น |

3. จงระบุข้อดีและข้อเสีย (หรือข้อจำกัด) ของ SDLCM ที่เลือกมา **ที่ส่งผลต่อหน่วยงานและโครงการนี้** มาอย่างละ 2 รายการ

| | |
|------------------------|--|
| ข้อดี | 1. ลดความเสี่ยงเรื่องคุณภาพของซอฟต์แวร์: เนื่องจากเป็นการพัฒนาในรูปแบบ Incremental development ซึ่งมีการตรวจสอบความต้องการ ความคาดหวังของลูกค้า และทดสอบในสภาพแวดล้อมเสมือนจริงเป็นระยะ จึงช่วยให้ช่วยลด Effort ในการปรับแก้ไข และลดปัญหาที่อาจจะเกิดขึ้นภายหลังการพัฒนาเสร็จสิ้นลงไปได้ |
| | 2. ส่งมอบซอฟต์แวร์ที่ใช้งานได้เร็วขึ้น: ทีมสามารถพัฒนาและส่งมอบฟังก์ชันที่จำเป็นได้ตั้งแต่ระยะแรกของโครงการ ทำให้ผู้ใช้สามารถเริ่มต้นทดลองใช้งานระบบบางส่วนได้ก่อนที่ระบบจะเสร็จสมบูรณ์ทั้งหมด |
| ข้อเสีย (หรือข้อจำกัด) | 1. อาจมีปัญหาในการคุมระยะเวลาการพัฒนา: เนื่องจาก Agile อาศัยการปรับเปลี่ยนตาม Feedback ของผู้ใช้ อาจทำให้ขอบเขตงานขยายตัวออกไปเรื่อย ๆ (Scope creep) จนโครงการใช้เวลานานกว่าที่คาดการณ์ไว้ |
| | 2. ต้องมีการทำงานร่วมกันอย่างใกล้ชิด: Agile ต้องอาศัยการสื่อสารและการทำงานร่วมกันของทีมและผู้ใช้งานอย่างต่อเนื่อง หากทีมไม่สามารถประชุมหรือผู้ใช้งานไม่สามารถให้ Feedback ได้ตรงเวลา อาจส่งผลกระทบต่อแผนงาน |

3. จากข้อเสีย (หรือข้อจำกัด) ของ SDLCM ให้ระบุคำแนะนำในการปรับปรุงข้อเสียหรือข้อจำกัดให้ดีขึ้น เพื่อจะสามารถนำ SDLCM ไปใช้ในการกำหนดขั้นตอนต่าง ๆ (process adaptation) ในการพัฒนาระบบ “ระบบการนัดหมายผู้ป่วยนอกและระบบการบันทึกการรักษาโดยแพทย์” ด้วยการประยุกต์ใช้ process reference model (12207) หรือ ความรู้อื่น ๆ ที่เกี่ยวข้อง มาอย่างน้อย 4 รายการ

| |
|--|
| 1. Software Documentation: กำหนดมาตรฐานเอกสารที่จำเป็น (เช่น เอกสาร Requirements, เอกสาร System design, คู่มือผู้ใช้) รวมทั้งกระบวนการในการสร้างและปรับปรุงเอกสารให้เป็นปัจจุบันอยู่เสมอ แม้จะใช้ Agile ก็ตาม เพื่อป้องกันปัญหาในการบำรุงรักษาและการส่งมอบงานในระยะยาว |
| 2. Verification & Validation Processes: กำหนด DoD (Definition of Done) ที่ครอบคลุม รวมถึง Non-functional requirements เช่น ความปลอดภัย, ประสิทธิภาพ, และการทดสอบ เพื่อป้องกันขอบเขตงานขยายตัวออกไปเรื่อย ๆ |
| 3. Measurement Process: ใช้ Burndown chart หรือ Dashboard อื่น ๆ เพื่อให้เห็นภาพรวมและสามารถติดตามความก้าวหน้าของการพัฒนาระบบได้ (Progress Visualization) |
| 4. Requirements Elicitation Process: สร้างตัวอย่างระบบ (Prototype) หรือภาพจำลอง (Mockup) เพื่อให้ผู้ใช้งานเห็นภาพที่ชัดเจนและสามารถให้ Feedback ได้ตั้งแต่ช่วงเริ่มต้นหรือระยะแรก ๆ ของการพัฒนา |

รายชื่อสมาชิก

- Chaiwat Kasetsinchai 6770228121
- Chanya-sean Phongthana-oran 6772018021
- Setsiri Matewin 6772097621
- Supakrit Chuchattwannakul 6770285921