



ภาคผนวก ก แบบสอบถาม เรื่องการเปรียบเทียบประสิทธิภาพเทคนิคการพัฒนาระบบ  
โดย Agile and Non-Agile

แบบสอบถามฉบับนี้เพื่อศึกษากระบวนการเกี่ยวกับการพัฒนาระบบภายในองค์กรของท่าน โดยมีวัตถุประสงค์เพื่อเปรียบเทียบประสิทธิภาพเทคนิคการพัฒนาระบบโดย Agile and Non-Agile โดยคำตอบของท่านจะเป็นประโยชน์อย่างยิ่งในการพัฒนาองค์ความรู้ด้านเทคโนโลยีสารสนเทศ จึงใคร่ขอความร่วมมือให้จากท่านในการกรอกแบบสอบถามนี้เพื่อประโยชน์ทางการศึกษา และขอขอบพระคุณอย่างยิ่งในความร่วมมือของท่าน

แบบสอบถามแบ่งออกเป็น 5 ตอน

ตอนที่ 1 ข้อมูลเกี่ยวกับบริบทโครงการ

ตอนที่ 2 ข้อมูลเกี่ยวกับปัจจัยเบื้องต้นของโครงการ

ตอนที่ 3 ข้อมูลเกี่ยวกับผลผลิตของโครงการต่อประสิทธิภาพการประยุกต์ใช้เทคนิค หรือวิธีการพัฒนาระบบสำหรับโครงการพัฒนาซอฟต์แวร์

ตอนที่ 4 ข้อมูลเกี่ยวกับกระบวนการประยุกต์ใช้เทคนิค หรือวิธีการพัฒนาระบบสำหรับโครงการพัฒนาซอฟต์แวร์

ตอนที่ 5 ข้อมูลเกี่ยวกับทัศนคติที่มีต่อข้อบกพร่องการประยุกต์ใช้เทคนิค หรือวิธีการพัฒนาระบบสำหรับโครงการพัฒนาซอฟต์แวร์

Questionnaire ID .....

Questionnaire Name.....

Questionnaire Location .....

ผู้วิจัยนางสาวพิมพ์พรรณ ทิพยแสง

นักศึกษาระดับปริญญาโท สาขาเทคโนโลยีสารสนเทศ

มหาวิทยาลัยนเรศวร

### แบบสอบถามประสิทธิภาพระเบียบวิธีการพัฒนาระบบ

คำชี้แจง โปรดทำเครื่องหมายลงใน ☺ หรือ ☑ หรือเติมข้อความตามความเป็นจริง  
เกี่ยวกับการพัฒนาระบบภายในองค์กรของท่าน

#### ตอนที่ 1 ข้อมูลเกี่ยวกับบริษัทโครงการ

1. ประเภทของธุรกิจ/ องค์กร/ หน่วยงานที่คุณปฏิบัติงานอยู่

- เทคโนโลยีสารสนเทศ (Information technology)
- โทรคมนาคม (Telecommunication)
- หน่วยงานของภาครัฐ (Government)
- โรงงานอุตสาหกรรม (Factory)
- อื่นๆ โปรดระบุ

2. ระยะเวลาการประกอบกิจการขององค์กร/ หน่วยงานที่คุณปฏิบัติงาน

- 1-6 ปี
- 7-12 ปี
- 13-18 ปี
- 19-24 ปี
- $\geq 25$  ปีขึ้นไป

3. จำนวนผู้เชี่ยวชาญ หรือผู้ปฏิบัติงานประจำด้านการพัฒนาซอฟต์แวร์ภายในองค์กร/  
หน่วยงานที่คุณปฏิบัติงาน

- ผู้เชี่ยวชาญ หรือผู้ปฏิบัติงานประจำด้านการพัฒนาซอฟต์แวร์ น้อยกว่า 10 คน
- ผู้เชี่ยวชาญ หรือผู้ปฏิบัติงานประจำด้านการพัฒนาซอฟต์แวร์ 11-25 คน
- ผู้เชี่ยวชาญ หรือผู้ปฏิบัติงานประจำด้านการพัฒนาซอฟต์แวร์ 26-50 คน
- ผู้เชี่ยวชาญ หรือผู้ปฏิบัติงานประจำด้านการพัฒนาซอฟต์แวร์ 51-100 คน
- มากกว่า 100 คนขึ้นไป โปรดระบุ

4. สิ่งแวดล้อมทางกายภาพ (Physical environment) ของทีมพัฒนากับสถานที่ติดตั้ง  
ซอฟต์แวร์ที่พัฒนาขึ้น โดยมีความเกี่ยวข้อง เช่น ต้องมีการแลกเปลี่ยนข้อมูลร่วมกัน

- การพัฒนาที่มีความเกี่ยวข้องระหว่างประเทศ (International)
- การพัฒนาที่มีความเกี่ยวข้องระหว่างเมือง และ ระหว่างบริษัท (Multi-city and Multi Company)
- การพัฒนาที่มีความเกี่ยวข้องภายในเมือง หรือบริเวณเดียวกัน (Same-city or metro area)
- การพัฒนาที่มีความเกี่ยวข้องภายในอาคารเดียวกัน (Same building or complex)
- การพัฒนาที่มีความเกี่ยวข้องภายในตำแหน่งที่ตั้งเดียวกัน (Fully Collocated)

5. สถานะ หรือลักษณะของโครงการพัฒนาซอฟต์แวร์

- พัฒนาระบบงานภายในองค์กร (In-house Development)
- ผู้รับว่าจ้างในการพัฒนาระบบ (Outsource Development)
- ผู้ร่วมทุนในการพัฒนาระบบ (Partnership Development)
- ผู้พัฒนาระบบงานสำหรับซอฟต์แวร์สำเร็จรูป (Commercial Product)
- อื่นๆ โปรดระบุ

6. รูปแบบวัฒนธรรมองค์กร (Business culture) ในการยอมรับเทคโนโลยี (Technology) หรือวิธีการ (Method) ใหม่เกิดขึ้น....

- เป็นผู้นำด้านเทคโนโลยี และแนวคิดใหม่ โดย ลงมதியอมรับ และปรับใช้เทคโนโลยี หรือวิธีการใหม่
- มีความพอใจที่จะลงมதியอมรับ และปรับใช้เทคโนโลยี หรือวิธีการใหม่ ตามผู้นำด้านเทคโนโลยี
- มีความพอใจที่จะลงมதியอมรับ และปรับใช้เทคโนโลยี หรือวิธีการใหม่ก็ต่อเมื่อเทคโนโลยีนั้นได้รับการพิสูจน์แล้ว
- มีความพอใจที่จะลงมதியอมรับ และปรับเทคโนโลยี หรือแนวคิดใหม่เพียงบางส่วนที่ไม่ขัดต่อรูปแบบการพัฒนาที่เคยทำมา
- มีความพอใจที่จะพัฒนาตามรูปแบบที่เคยทำมา ไม่ยอมรับเทคโนโลยี หรือแนวคิดใหม่

7. จำนวนผู้ปฏิบัติงานร่วมกับซอฟต์แวร์เมื่อได้ผลิตภัณฑ์ขั้นสุดท้าย (End User) จากโครงการพัฒนาซอฟต์แวร์นี้

- ผู้ปฏิบัติงานร่วมกับซอฟต์แวร์ น้อยกว่า 10 คน
- ผู้ปฏิบัติงานร่วมกับซอฟต์แวร์ 11-25 คน
- ผู้ปฏิบัติงานร่วมกับซอฟต์แวร์ 26-50 คน
- ผู้ปฏิบัติงานร่วมกับซอฟต์แวร์ 51-100 คน
- ผู้ปฏิบัติงานร่วมกับซอฟต์แวร์ มากกว่า 100 คนขึ้นไป โปรดระบุ

8. มาตรฐานขององค์กร CMMI, ISO, Software Quality Assurance, Verification and Validation

- มีการกำหนดมาตรฐานเกี่ยวกับการพัฒนาระบบในองค์กร
- ไม่มีการกำหนดมาตรฐานเกี่ยวกับการพัฒนาระบบในองค์กร

9. โครงการพัฒนาซอฟต์แวร์นี้มีความกดดันในเรื่องของงบประมาณอย่างไร

- ไม่มีความกดดันเรื่องงบประมาณ
- มีความกดดันเรื่องงบประมาณน้อยมาก
- มีความกดดันเรื่องงบประมาณปานกลาง
- มีความกดดันเรื่องงบประมาณสูง
- มีความกดดันเรื่องงบประมาณสูงมาก

10. โครงการพัฒนาซอฟต์แวร์นี้มีความกดดันในเรื่องของระยะเวลาอย่างไร

- ไม่มีความกดดันเรื่องระยะเวลา
- มีความกดดันเรื่องระยเวลา น้อยมาก
- มีความกดดันเรื่องระยะเวลาปานกลาง
- มีความกดดันเรื่องระยะเวลาสูง
- มีความกดดันเรื่องระยะเวลาสูงมาก

## 11. ความต้องการให้ซอฟต์แวร์มีความแน่นอน

- ความต้องการให้ซอฟต์แวร์มีความแน่นอนน้อยมากความผิดพลาดใดใดสามารถแก้ไขได้
- ความต้องการให้ซอฟต์แวร์มีความแน่นอนน้อย
- ความต้องการให้ซอฟต์แวร์มีความแน่นอนปานกลาง
- ความต้องการให้ซอฟต์แวร์มีความแน่นอนสูง เกี่ยวข้องกับข้อมูลด้านการเงิน
- ความต้องการให้ซอฟต์แวร์มีความแน่นอนสูงหากผิดพลาดอาจส่งผลกระทบต่อชีวิต



12. องค์การของคุณกำหนดแนวทางของโครงการพัฒนาซอฟต์แวร์ที่ประสบความสำเร็จโดยเน้นสิ่งใดเป็นหลักที่สำคัญ

การพัฒนาทรัพยากรมนุษย์, การทำงานเป็นทีม, การมอบหมาย ความไว้วางใจแก่พนักงาน	น้อยที่สุด C 1	น้อย C 2	ปานกลาง C 3	มาก C 4	มากที่สุด C 5
ความเป็นเอกลักษณ์, ทัศนียภาพของผลิตภัณฑ์ โดยผลิตภัณฑ์ให้เป็นผู้นำ ด้านนวัตกรรม และผลิตภัณฑ์	C 1	C 2	C 3	C 4	C 5
ประสบความสำเร็จในด้านส่วนแบ่งการตลาด เป็นผู้นำด้านการ แข่งขันเหนือคู่แข่ง	C 1	C 2	C 3	C 4	C 5
ผลิตภัณฑ์มีประสิทธิภาพ, งบประมาณต่ำ, สามารถบริหารจัดการ ตารางการทำงานได้ดี, มีความน่าเชื่อถือด้านการส่งมอบผลิตภัณฑ์	C 1	C 2	C 3	C 4	C 5

13. การบริหารจัดการการทีมให้ความสำเร็จกับเรื่องดังต่อไปนี้ในระดับใด

การทำงานเป็นทีม, ความสอดคล้องกันในเรื่องของความคิด, การมี ส่วนร่วมของทุกคน	น้อยที่สุด C 1	น้อย C 2	ปานกลาง C 3	มาก C 4	มากที่สุด C 5
การแลกเปลี่ยนความรู้, นวัตกรรม, เทคโนโลยีใหม่	C 1	C 2	C 3	C 4	C 5
ผลิตภัณฑ์เกิดความกระตือรือร้น การทำงานที่มีประสิทธิภาพ	C 1	C 2	C 3	C 4	C 5



## ตอนที่ 2 ข้อมูลเกี่ยวกับปัจจัยเบื้องต้นของโครงการ

แบบสอบถามฉบับนี้ เพื่อศึกษากระบวนการเกี่ยวกับการพัฒนาระบบภายในองค์กรของท่าน โปรดกรอกรายละเอียดเกี่ยวกับโครงการที่มีการพัฒนาอยู่ หรือพัฒนาเสร็จเรียบร้อยแล้ว เพียง 1 โครงการ

14. ชื่อโครงการ \_\_\_\_\_

15. คำอธิบายโดยย่อเกี่ยวกับโครงการ \_\_\_\_\_

16. รูปแบบการจัดทำเอกสารสัญญา (Contract) หรือเอกสารนำเสนอโครงการ (Proposal) สำหรับการดำเนินการพัฒนาซอฟต์แวร์ให้ความสำคัญกับเรื่องใดมากที่สุด

- ข้อตกลงให้ความสำคัญกับเรื่องการกำหนดมูลค่าซอฟต์แวร์แบบตายตัว (Fixed Price Contract)
- ข้อตกลงให้ความสำคัญกับเรื่องการกำหนดเวลา และความเหมาะสมของซอฟต์แวร์ (Time and Material Contract)
- ข้อตกลงให้ความสำคัญกับการกำหนดเป้าหมายด้านมูลค่าซอฟต์แวร์ (Target Cost Contract)
- ข้อตกลงให้ความสำคัญกับการกำหนดเป้าหมายด้านตารางเวลาการซอฟต์แวร์ (Target Schedule Contract)
- ข้อตกลงให้ความสำคัญกับการมีส่วนร่วมในผลกำไรจากการพัฒนาซอฟต์แวร์สำหรับขายร่วมกัน (Share-Benefit Contract)

17. ระดับความคล้ายคลึงกับโครงการเดิม หรือมีแบบอย่างมาก่อน เมื่อเทียบกับโครงการปัจจุบัน

- ไม่มีแบบอย่าง หรือตัวอย่างก่อนหน้าโดยสิ้นเชิง (Thoroughly Unprecedented)
- ส่วนใหญ่ไม่มีแบบอย่าง หรือตัวอย่างก่อนหน้า (Largely Unprecedented)
- บางส่วนไม่มีแบบอย่าง หรือตัวอย่างก่อนหน้า (Somewhat Unprecedented)
- โดยภาพรวมมีความคุ้นเคยจากโครงการก่อนหน้า (Generally Familiar)
- ส่วนใหญ่มีความคุ้นเคยจากโครงการก่อนหน้า (Largely Familiar)

18. กรุณาเลือกตัวบ่งชี้ความซับซ้อนโครงการพัฒนาซอฟต์แวร์

- มีการเลือกใช้เทคโนโลยีใหม่ๆ มากกว่า 2 แบบ โดยที่ทีมพัฒนาไม่คุ้นเคยมาก่อน
- โครงการมีความเกี่ยวข้องกับขอบเขตทางธุรกิจ (Business Domain) โดยที่ทีมพัฒนาไม่คุ้นเคย
- โครงการมีการบูรณาการ (Integration) มากกว่า 2 โปรแกรม
- โครงการเผชิญกับปัญหา ซึ่งทีมไม่มีความคุ้นเคย และไม่มีวิธีการรูปแบบในการแก้ปัญหา
- ใช้ปัจจัยอื่นๆ ที่ตัวบ่งชี้ความซับซ้อนในการโครงการพัฒนาซอฟต์แวร์  
กรุณาอธิบาย

19. เป้าหมายของการพัฒนาซอฟต์แวร์

- พัฒนาระบบใหม่ (new system)
- พัฒนาเพื่อปรับปรุงระบบเดิมที่มีอยู่ (enhancement to existing system)
- พัฒนาเพื่อแทนที่ระบบเดิมที่มีอยู่ (Replacement of existing system)
- พัฒนาเพื่อรักษาสภาพของระบบ (Maintenance)
- อื่นๆ โปรดระบุ (Other, Please specify).....

20. โครงการพัฒนาซอฟต์แวร์ถูกพัฒนาขึ้นเพื่อประยุกต์ใช้ในการแก้ปัญหาเป้าหมายทางธุรกิจแบบใด เลือกคำตอบได้มากกว่า 1 ข้อ

- การบัญชี
- การธนาคาร
- ด้านความบันเทิง
- เกมส์
- คลังสินค้า
- การตลาด
- การจัดการสารสนเทศ
- สำนักงานอัตโนมัติ
- การจัดซื้อ
- การขาย
- ซอฟต์แวร์ระบบ
- วิทยาศาสตร์
- วิศวกรรม
- อื่นๆ, โปรดระบุ.....

21. โครงการพัฒนาซอฟต์แวร์มีเป้าหมายสำหรับการทำงานรูปแบบใด เลือกคำตอบได้มากกว่า 1 ข้อ

- ระบบการสนับสนุนและตัดสินใจ (decision support system)
- ระบบสารสนเทศเพื่อการจัดการ (management information system)
- ระบบสำนักงานอัตโนมัติ (office automation system)
- การประมวลผลด้วยรายการเปลี่ยนแปลง (transaction processing)
- ซอฟต์แวร์ระบบ (systems software)
- ระบบอินทราเน็ต (intranet system)
- ระบบอินเทอร์เน็ต (internet system)
- ระบบปฏิบัติการ (operating system)
- ระบบเอ็กซ์ทราเน็ต (extranet system)
- อื่นๆ, โปรดระบุ.....

22. โครงการพัฒนาซอฟต์แวร์นี้มีรูปแบบการใช้งานโปรแกรมที่เชื่อมต่อกับผู้ใช้รูปแบบใด เลือกคำตอบได้มากกว่า 1 ข้อ

- Gaming software engine
- Interface (console)
- Interface (web)
- Interface (windows)
- Middleware
- Mobile device
- Object database
- Relational database

- └ Web development (dynamic)
- └ Web development (Static)
- └ อื่นๆ, โปรดระบุ.....

23. กรุณาอธิบายขั้นตอน (Phase) การพัฒนาระบบที่ใช้ในโครงการนี้

.....

.....

.....

.....

24. กรุณาใส่ชื่อเครื่องมือหรือเทคนิคที่ใช้ในโครงการนี้

Case/modeling tools

.....

Unit testing tools

.....

System testing tools

.....

Programming languages

.....

Other tools

.....

25. กรุณาระบุหน้าที่รับผิดชอบภายในโครงการนี้ เลือกคำตอบได้มากกว่า 1 ข้อ

- พัฒนาโปรแกรม (Programmer / Developer)
- วิเคราะห์ระบบ (Analyst)
- นักออกแบบซอฟต์แวร์ (Software Architect)
- วิศวกรซอฟต์แวร์ (Software Engineering)
- ที่ปรึกษาโครงการ (Consultant)
- ผู้บริหารโครงการ (Project Manager)
- อื่นๆ, โปรดระบุ.....

26. จำนวนสมาชิกในทีมของโครงการพัฒนาซอฟต์แวร์นี้

- $\leq 6$  คน
- 7-12 คน
- 13-18 คน
- 19-24 คน
- $\geq 25$  คนขึ้นไป

27. ประสบการณ์การทำงานโดยเฉลี่ยของทีมที่เกี่ยวข้องกับขอบเขตทางธุรกิจ (Business Domain) ในรูปแบบเดียวกับโครงการพัฒนาซอฟต์แวร์นี้มานานเท่าไร ตัวอย่าง ระบบการบัญชี (Accounting System)

- $\geq 2$  เดือน
- 6 เดือน
- 1-2 ปี
- 3-4 ปี
- $> 5$  ปีขึ้นไป

28. ประสบการณ์การทำงานโดยเฉลี่ยของทีมที่เกี่ยวข้องกับการพัฒนาซอฟต์แวร์ (Software development) มานานเท่าไร

- 1-3 ปี
- 4-6 ปี
- 7-9 ปี
- 10-12 ปี
- > 12 ปีขึ้นไป

29. ประสบการณ์การทำงานและความคุ้นเคยกับภาษา

- $\geq 2$  เดือน
- 6 เดือน
- 1-2 ปี
- 3-4 ปี
- > 5 ปีขึ้นไป

30. ประสบการณ์การทำงานที่มีต่อระบบการทำงาน (Platform)

- $\geq 2$  เดือน
- 6 เดือน
- 1-2 ปี
- 3-4 ปี
- > 5 ปีขึ้นไป

31. ประสบการณ์โดยเฉลี่ยของทีมในการทำงานกับระเบียบวิธีการพัฒนาระบบ

- $\geq 2$  เดือน
- 6 เดือน
- 1-2 ปี
- 3-4 ปี
- > 5 ปีขึ้นไป

32. รูปแบบการติดต่อสื่อสารภายในทีมพัฒนาซอฟต์แวร์

- การติดต่อสื่อสารทั้งหมดโดยเอกสาร การประชุม หรือ เอกสารอิเล็กทรอนิกส์ที่มีรูปแบบตามกฎเกณฑ์
- การติดต่อสื่อสารโดยเอกสาร การประชุม หรือ เอกสารอิเล็กทรอนิกส์ที่มีรูปแบบตามกฎเกณฑ์เป็นส่วนใหญ่ โดยบางครั้งมีรูปแบบไม่เป็นทางการ มีความสมดุลระหว่างการติดต่อสื่อสารแบบตามกฎเกณฑ์ และไม่เป็นทางการ
- การติดต่อสื่อสารมีรูปแบบไม่เป็นทางการเป็นส่วนใหญ่ โดยบางครั้งมีรูปแบบตามกฎเกณฑ์
- การติดต่อสื่อสารเป็นแบบไม่เป็นทางการ แบบ face-to-face เน้นการทำงานร่วมกัน เปิดเผยและตรงไปตรงมา

33. รูปแบบการติดต่อสื่อสารระหว่างทีม และผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์

- การติดต่อสื่อสารทั้งหมดโดยเอกสาร การประชุม หรือ เอกสารอิเล็กทรอนิกส์ที่มีรูปแบบตามกฎเกณฑ์
- การติดต่อสื่อสารโดยเอกสาร การประชุม หรือ เอกสารอิเล็กทรอนิกส์ที่มีรูปแบบตามกฎเกณฑ์เป็นส่วนใหญ่ โดยบางครั้งมีรูปแบบไม่เป็นทางการ มีความสมดุลระหว่างการติดต่อสื่อสารแบบตามกฎเกณฑ์ และไม่เป็นทางการ
- การติดต่อสื่อสารมีรูปแบบไม่เป็นทางการเป็นส่วนใหญ่ โดยบางครั้งมีรูปแบบตามกฎเกณฑ์
- การติดต่อสื่อสารเป็นแบบไม่เป็นทางการ แบบ face-to-face เน้นการทำงานร่วมกัน เปิดเผยและตรงไปตรงมา

34. ช่วงเวลาการทำงานแบบวนรอบ (Iterative) เพื่อจัดส่งซอฟต์แวร์ให้กับลูกค้าเพื่อแสดงความก้าวหน้าของโครงการ

- ไม่มีการทำงานแบบวนรอบ
- 1 สัปดาห์
- 2 สัปดาห์
- 1 เดือน
- ช่วงเวลาวนรอบมีความยืดหยุ่น



ตอนที่ 3 ข้อมูลเกี่ยวกับกระบวนการประยุกต์ใช้เทคนิค หรือวิธีการพัฒนาระบบสำหรับ  
โครงการพัฒนาซอฟต์แวร์

คำนิยามความหมายของคุณลักษณะของประสิทธิภาพกระบวนการพัฒนาระบบ	
คุณลักษณะ (Features)	คำอธิบาย (Description)
1	<p>วงจรเวลาในการพัฒนา (cycle time)</p> <p>มิติด้านวงจรในการพัฒนา เทคนิค หรือวิธีปฏิบัติโดยลดระยะเวลาในการพัฒนาสำหรับ</p> <ul style="list-style-type: none"> <li>- ลดวงจรการพัฒนาโดยรวม (Reduce cycle time)</li> <li>- ลดระยะเวลาการส่งมอบ (Reduce delivery time)</li> </ul>
2	<p>งบประมาณ (Cost and benefit Effective)</p> <p>มิติด้านการบริหารจัดการงบประมาณ เทคนิค หรือวิธีปฏิบัติช่วยให้สามารถใช้งบประมาณได้อย่างมีประสิทธิภาพ สำหรับกระบวนการพัฒนาซอฟต์แวร์</p>
3	<p>คุณภาพของกระบวนการ (quality of process)</p> <p>มิติด้านคุณภาพของกระบวนการ เทคนิค หรือวิธีปฏิบัติโดยสนับสนุนให้สามารถปรับปรุงระดับคุณภาพของกระบวนการทั้งด้าน</p> <ul style="list-style-type: none"> <li>- การบรรลุเป้าหมายของโครงการ (Effectiveness)</li> <li>- ความพึงพอใจของโครงการ และผู้มีส่วนได้ส่วนเสียของโครงการ</li> <li>- ความอยู่รอดของโครงการด้านระยะเวลา และค่าใช้จ่าย (Sustainability)</li> <li>- การกระจายงานในแต่ละขั้นตอน (Equity)</li> <li>- ผลผลิตเปรียบเทียบกับเป้าหมายรวมกิจกรรมที่ทำแล้วเสร็จ ทรัพยากร และเวลาที่ใช้ไปได้ตรงตามเป้าหมาย (Progress)</li> <li>- ความพอเพียงของทรัพยากร (Adequacy)</li> <li>- ลดความสูญเสียที่เกิดขึ้นสำหรับโครงการพัฒนาซอฟต์แวร์</li> </ul>

คำนิยามความหมายของคุณลักษณะของประสิทธิภาพกระบวนการพัฒนาระบบ	
คุณลักษณะ (Features)	คำอธิบาย (Description)
4 การทำงานของผู้ที่เกี่ยวข้อง (inter-supplier performance)	<p>มิติด้านการทำงานของผู้ที่เกี่ยวข้อง เทคนิค หรือวิธีปฏิบัติใดช่วยเพิ่มประสิทธิภาพการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์</p> <ul style="list-style-type: none"> <li>- ความร่วมมือกัน (Collaborative)</li> <li>- การทำหน้าที่เป็นตัวแทน (Representative)</li> <li>- การมอบหมายอำนาจ (Authorized)</li> <li>- การมอบหมายความไว้วางใจ (Committed)</li> <li>- การสร้างความรู้ความเข้าใจในระบบงาน (Knowledgeable)</li> </ul>

35. คุณคิดว่าระเบียบวิธีการพัฒนาระบบที่เลือกใช้ส่งผลต่อกระบวนการควบคุมการทำงานให้เป็นไปตามเงื่อนไขของระยะเวลาที่กำหนดไว้ (วงจรการพัฒนารวมถึง ระยะเวลาในการเพิ่มลักษณะพิเศษสำหรับการทำงาน, ปรับให้เข้ากับเทคโนโลยีใหม่, ระยะเวลาการส่งมอบ (Reduce delivery time) อย่างไร

Project	Unit	Plan		Schedule
		Date of Start	Date of Complete	
	Days			
		Actual		
		Date of Start	Date of Complete	
	Days			
% of Schedule Deviation				

36. คุณคิดว่าระเบียบวิธีการพัฒนาระบบที่เลือกใช้ส่งผลกระทบต่อการบริหารจัดการงบประมาณอย่างไร

Project	Unit	Plan	Actual	% of Cost Deviation
	Bath			

ทัศนคติที่มีต่อผลการประยุกต์ใช้เทคนิค หรือวิธีการพัฒนาระบบสำหรับโครงการพัฒนาซอฟต์แวร์

๑ 1	ไม่ครอบคลุม (No Effect (1)) บ่งชี้ว่าระเบียบวิธีการพัฒนาระบบที่เลือกใช้ไม่ส่งผลกระทบต่อประสิทธิภาพของการพัฒนาระบบ
๑ 2	สนับสนุนน้อย (Moderate Support (2)) บ่งชี้ว่าระเบียบวิธีการพัฒนาระบบที่เลือกใช้ส่งผลกระทบต่อปรับปรุงประสิทธิภาพของการพัฒนาระบบน้อย
๑ 3	สนับสนุนปานกลาง (Support (3)) บ่งชี้ว่าระเบียบวิธีการพัฒนาระบบที่เลือกใช้ส่งผลกระทบต่อปรับปรุงประสิทธิภาพของการพัฒนาระบบปานกลาง
๑ 4	สนับสนุนมาก (Significant Support (4)) บ่งชี้ว่าระเบียบวิธีการพัฒนาระบบที่เลือกใช้ส่งผลกระทบต่อปรับปรุงประสิทธิภาพของการพัฒนาระบบมาก
๑ 5	สนับสนุนมากที่สุด (Strong Support (5)) บ่งชี้ว่าระเบียบวิธีการพัฒนาระบบที่เลือกใช้ส่งผลกระทบต่อปรับปรุงประสิทธิภาพของการพัฒนาระบบมากที่สุด

คำชี้แจง โปรดทำเครื่องหมาย ☺ เพื่อแสดงทัศนคติที่มีต่อผลการประยุกต์ใช้เทคนิค  
หรือวิธีการพัฒนาระบบสำหรับโครงการพัฒนาซอฟต์แวร์

37. คุณคิดว่าระเบียบวิธีการพัฒนาระบบที่เลือกใช้ส่งผลกระทบต่อระดับคุณภาพของ  
กระบวนการพัฒนาซอฟต์แวร์ในระดับใด

ปัจจัยด้านคุณภาพของกระบวนการพัฒนาซอฟต์แวร์					
การบรรลุเป้าหมายของโครงการ (Effectiveness)	๑	๒	๓	๔	๕
ความพึงพอใจของโครงการ และผู้มีส่วนได้ส่วนเสียของโครงการ	๑	๒	๓	๔	๕
ความอยู่รอดของโครงการด้านระยะเวลา และค่าใช้จ่าย (Sustainability)	๑	๒	๓	๔	๕
การกระจายงานในแต่ละขั้นตอน (Equity)	๑	๒	๓	๔	๕
ผลผลิตเปรียบเทียบกับเป้าหมายรวมกิจกรรมที่ทำแล้วเสร็จ ทรัพยากร และเวลาที่ใช้ไปได้ตรงตามเป้าหมาย (Progress)	๑	๒	๓	๔	๕
ความพอเพียงของทรัพยากร (Adequacy)	๑	๒	๓	๔	๕
ลดความสูญเสียที่เกิดขึ้นสำหรับโครงการพัฒนาซอฟต์แวร์	๑	๒	๓	๔	๕

38. คุณคิดว่าระเบียบวิธีการพัฒนาระบบที่เลือกใช้ส่งผลต่อกระทบต่อประสิทธิภาพการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ในระดับใด

ปัจจัยการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์					
ความร่วมมือกัน และช่วยสนับสนุนความสามารถในการบรรลุข้อตกลงของโครงการ (Collaborative)	C 1	C 2	C 3	C 4	C 5
การทำหน้าที่เป็นตัวแทน สร้างองค์กรเสมือนจริง ด้วยการสร้างแม่แบบ ประกอบด้วยมุมมองที่สามารถแลกเปลี่ยนแนวคิดกันระหว่างกลุ่ม (Representative)	C 1	C 2	C 3	C 4	C 5
การมีสิทธิตัดสินใจเฉพาะหน้าในเรื่องการเปลี่ยนแปลงความต้องการ (Authorized)	C 1	C 2	C 3	C 4	C 5
การให้ความสำคัญกับสมาชิกในทีมแต่ละคนเท่าๆ กัน โดยการมอบหมายความดูแล หรือความรับผิดชอบสำหรับการพัฒนา (Committed)	C 1	C 2	C 3	C 4	C 5
สร้างความรู้ความเข้าใจในระบบงาน และช่วยสนับสนุนให้เกิดการแลกเปลี่ยนความรู้เกี่ยวกับระบบงานที่จะพัฒนาของสมาชิกในทีม (Knowledgeable)	C 1	C 2	C 3	C 4	C 5

**ตอนที่ 4 ข้อมูลเกี่ยวกับผลผลิตของโครงการต่อประสิทธิภาพการประยุกต์ใช้เทคนิค  
หรือวิธีการพัฒนาระบบสำหรับโครงการพัฒนาซอฟต์แวร์**

คำชี้แจง โปรดทำเครื่องหมาย ☺ เพื่อแสดงทัศนคติที่มีต่อประสิทธิภาพการประยุกต์ใช้เทคนิค  
หรือวิธีการพัฒนาระบบสำหรับโครงการพัฒนาซอฟต์แวร์

☺ No	หากโครงการของคุณไม่มีการประยุกต์ใช้ เทคนิค และวิธีปฏิบัติ หรือ
☺ Yes	หากโครงการของคุณมีการประยุกต์ใช้เทคนิค และวิธีปฏิบัติ กรุณาแสดงความคิดเห็นว่า เทคนิค และวิธีปฏิบัติที่เลือกใช้ส่งผลกระทบต่อประสิทธิภาพของโครงการพัฒนาระบบในระดับใด ระดับความคิดเห็นแบ่งเป็น 5 ระดับดังนี้ (* เฉพาะเทคนิค และวิธีปฏิบัติที่โครงการของคุณมีการประยุกต์ใช้)
☺ ___	ขัดแย้ง (Largely Conflicting (----)) บ่งชี้ว่าเทคนิคหรือวิธีปฏิบัติส่งผลในทางตรงกันข้ามต่อการปรับปรุงประสิทธิภาพของการพัฒนาระบบมากที่สุด
☺ --	ขัดแย้ง (Conflicting (--)) บ่งชี้ว่าเทคนิคหรือวิธีปฏิบัติส่งผลในทางตรงกันข้ามต่อการปรับปรุงประสิทธิภาพของการพัฒนาระบบมาก
☺ -	ขัดแย้ง (Partially Conflicting (-)) บ่งชี้ว่าเทคนิคหรือวิธีปฏิบัติส่งผลในทางตรงกันข้ามต่อการปรับปรุงประสิทธิภาพของการพัฒนาระบบบางส่วน
☺ 0	ไม่ครอบคลุม (No Effect (0)) บ่งชี้ว่าการเลือกเทคนิคหรือวิธีปฏิบัติไม่ส่งผลกระทบต่อประสิทธิภาพของการพัฒนาระบบ
☺ +	สนับสนุนบางส่วน (Partially Support (+)) บ่งชี้ว่าการเลือกเทคนิคหรือวิธีปฏิบัติส่งผลกระทบต่อ การปรับปรุงประสิทธิภาพของการพัฒนาระบบบางส่วน
☺ ++	สนับสนุนอย่างมาก (Support (++) ) บ่งชี้ว่าการเลือกเทคนิคหรือวิธีปฏิบัติส่งผลกระทบต่อ การปรับปรุงประสิทธิภาพของการพัฒนาระบบอย่างมาก
☺ +++	สนับสนุนมากที่สุด (Largely Support (+++)) บ่งชี้ว่าการเลือกเทคนิคหรือวิธีปฏิบัติส่งผลกระทบต่อ การปรับปรุงประสิทธิภาพของการพัฒนาระบบมากที่สุด

ตาราง 71 แสดงแนวปฏิบัติหลักที่ทำงานประยุกต์ใช้สำหรับโครงการพัฒนาซอฟต์แวร์

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การนำเสนอผลการพัฒนาทีละน้อยอย่างต่อเนื่อง โดยพัฒนาเป็นวงรอบสั้นๆ เพื่อให้ผู้ที่เกี่ยวข้องเห็นภาพของระบบที่กำลังดำเนินการพัฒนา (Small Releases of Software Product)	Yes No	++ ++	++ ++	++ ++	++ ++
สร้างวิสัยทัศน์การออกแบบร่วมกัน เพื่อให้ผู้ใช้ที่เกี่ยวข้องเข้าใจการทำงานของระบบ (System metaphor development)	Yes No	++ ++	++ ++	++ ++	++ ++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การออกแบบด้วยโครงสร้าง ง่ายๆ ไม่ซับซ้อนสามารถแก้ไข หรือเปลี่ยนแปลงง่าย และ คล่องตัว (Design is kept as simple as possible)	<p>Yes C -- C - C 0 C + C - C - C 0 C - C - C - C - C 0</p> <p>No C + C ++ C +++ C + C ++ C +++ C + C ++ C +++</p>				
การปรับเปลี่ยนโครงสร้างของ โคช (code) เพื่อช่วยให้ สามารถรับมือกับการ เปลี่ยนแปลงได้ง่ายขึ้น มีความ ยืดหยุ่น (Refactoring of code)	<p>Yes C -- C - C 0 C + C - C - C 0 C - C - C - C - C 0</p> <p>No C + C ++ C +++ C + C ++ C +++ C + C ++ C +++</p>				



ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การพัฒนาโปรแกรมเป็นคู่ สลับหน้าทีกันระหว่างการพัฒนาโปรแกรม และการตรวจสอบ					
<input type="checkbox"/> Yes		0	0	0	0
<input type="checkbox"/> No		++	+++	++	+++
พัฒนาขึ้นมีคุณภาพ เนื่องจากมีการตรวจสอบอยู่ตลอดเวลา (pair programming)					
<input type="checkbox"/> Yes		0	0	0	0
<input type="checkbox"/> No		++	+++	++	+++
การสร้างควมมีส่วนร่วมในทุกส่วนของโปรแกรม ทีมพัฒนาทุกคนสามารถเข้าไปศึกษา (Collective ownership of code)					
<input type="checkbox"/> Yes		0	0	0	0
<input type="checkbox"/> No		++	+++	++	+++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การรวมการทำงาน ซอฟต์แวร์อย่างต่อเนื่อง เพื่อ ตรวจสอบความสามารถของ การทำงานของซอฟต์แวร์ (Continue Integration)	Yes No	++ ++	-- ++	-- ++	-- ++
การยินยอมให้ลูกค้าเข้ามา มีส่วนเกี่ยวข้องกับการพัฒนา ระบบ (Customer on-site)	Yes No	-- ++	-- ++	-- ++	-- ++
การทำงาน 40 ชั่วโมงต่อ สัปดาห์ (40-h week)	Yes No	-- ++	-- ++	-- ++	-- ++
การนำการทดสอบมาตั้งต้น การพัฒนาในวงรอบถัดไป (Test first Development)	Yes No	-- ++	-- ++	-- ++	-- ++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การจัดทำมาตรฐานการเขียนโปรแกรม เพื่อให้ทีมพัฒนา มีความเข้าใจร่วมกันในการพัฒนาโปรแกรม (Coding to an agree standard)	Yes No	++ +++	++ +++	++ +++	++ +++
การระดมความคิดในเชิงการพัฒนาซอฟต์แวร์ เพื่อให้ผู้ที่เกี่ยวข้องมีส่วนร่วมในการทำงาน และเกิดการสื่อสารระหว่างทีม และเกิดการกระจายความรู้ในทีม (Scrum Team)	Yes No	++ +++	++ +++	++ +++	++ +++



ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การทบทวนการดำเนินงานในแต่ละวงรอบ (Sprint review)	Yes	C -	C -	C -	C -
	No	C +	C ++	C ++	C ++
กิจกรรมที่จำลองรูปแบบธุรกิจ					
โดยค้นหา และอธิบายขอบเขตของปัญหา ด้วยแบบจำลองการปฏิบัติการเชิงวัตถุ (Domain object modeling)	Yes	C -	C -	C -	C -
	No	C +	C ++	C ++	C ++
การพัฒนาจากคุณลักษณะ (Developing by Feature)	Yes	C -	C -	C -	C -
	No	C +	C ++	C ++	C ++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การแบ่งระดับการดูแล และสิทธิใน					
การเข้าถึงได้ (Code) เพื่อให้การพัฒนาเป็นไปอย่างคล่องตัว (Individual class ownership)	Yes No	++ +++	-- +++	-- +++	-- +++
การจัดตั้งทีมที่มีขนาดเล็ก มีระดับความสามารถสูง ความคล่องตัวสูง (Feature teams) <sup>1</sup>	Yes No	-- +++	-- +++	-- +++	-- +++
การพัฒนาซอฟต์แวร์ที่สามารถทำงานได้อย่างสม่ำเสมอ เพื่อลดการทำงานให้กับลูกค้า และช่วยให้พบข้อบกพร่องของการพัฒนาอย่างรวดเร็ว (Regular Builds)	Yes No	-- +++	-- +++	-- +++	-- +++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การพัฒนาซอฟต์แวร์ที่สามารถทำงานได้อย่างสม่ำเสมอ เพื่อ					
•  สถิติการทำงานให้กับลูกค้า	•  Yes	•  -	•  -	•  -	•  -
•  และช่วยให้พบข้อบกพร่องของการพัฒนาอย่างรวดเร็ว	•  No	•  ++	•  +++	•  ++	•  +++
(Regular Builds)					
•  ขั้นตอนวางแผนการพัฒนาโดยใช้ข้อมูลพื้นฐานสำหรับการกำหนดภารกิจของโครงการเป็นตัวขับเคลื่อนการพัฒนา (The project mission development) <sup>2</sup>					
•  Yes	•  -	•  -	•  -	•  -	•  -
•  No	•  +	•  +++	•  +++	•  ++	•  +++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การพัฒนาจากคอมพิวเตอร์ (Component Development)	Yes	++	++	++	++
	No	++	++	++	++
การสร้างความร่วมมือร่วมกับทุกฝ่ายที่ร่วมงานจะเรียนรู้ไปพร้อมๆ กับความก้าวหน้าไปสู่อะไรก็ตาม (Postmortems) เป็นต้น (Collaborative teams) <sup>3</sup>	Yes	++	++	++	++
	No	++	++	++	++



ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การพัฒนาแบบมีส่วนร่วม					
ระหว่าง ผู้ใช้ ผู้พัฒนาระบบ และผู้บริหาร (Joint application development: JAD)	<p>Yes C -- C - C 0 C ++ C - C 0 C -- C - C 0 C - C - C 0</p> <p>No C + C ++ C +++ C + C ++ C +++ C + C ++ C +++</p>				
การตรวจสอบข้อผิดพลาดของซอฟต์แวร์ โดยมุ่งไปที่ข้อผิดพลาดของการทำงานแต่ละส่วน (Software inspections)	<p>Yes C -- C - C 0 C -- C - C 0 C -- C - C 0 C -- C - C 0</p> <p>No C + C ++ C +++ C + C ++ C +++ C + C ++ C +++</p>				

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
พัฒนาเฉพาะส่วนที่เกี่ยวข้องกับการทำงานของผู้ใช้ (Active user involvement) <sup>4</sup>	Yes	C -	C -	C -	C -
	No	C ++	C ++	C ++	C ++
การจัดตั้งทีมมีการจัดระเบียบและร่วมมือกันสูง สนับสนุนการพัฒนาแบบไม่หยุดนิ่ง ซึ่งการเปลี่ยนแปลงอาจเกิดขึ้นได้ตลอดเวลา (Empowered teams)	Yes	C -	C -	C -	C -
	No	C ++	C ++	C ++	C ++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การส่งมอบซอฟต์แวร์ที่สามารถทำงานได้อย่างสม่ำเสมอ	Yes	++	++	++	++
เพื่อนำเสนอความก้าวหน้าของซอฟต์แวร์ (Frequent product delivery)	No	+++	+++	+++	+++
การพัฒนาเป็นวงรอบ โดยอาศัยการเพิ่มส่วนงานเข้าไปในงานเดิมเรื่อยๆ เพื่อให้มีความสมบูรณ์มากขึ้น (Iterative and incremental development)	Yes	++	++	++	++
	No	+++	+++	+++	+++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การยินยอมให้มีการเปลี่ยนแปลงได้ตลอดเวลา					
ขณะที่พัฒนา โดยการพัฒนา	Yes	- - -	- - -	- - -	- - -
แบบยอมรับการเปลี่ยนแปลง	No	+ + +	+ + +	+ + +	+ + +
สนับสนุนวงจรการพัฒนาแบบ					
ปรับตัว (Reversible changes)					
การทดสอบการทำงานของกลุ่ม					
โปรแกรมหรือส่วนประกอบย่อย					
ที่ถูกประสานเข้าด้วยกัน โดย	Yes	- - -	- - -	- - -	- - -
ทำงานหน้าที่เดิมน้ำที่หนึ่ง	No	+ + +	+ + +	+ + +	+ + +
ร่วมกัน เพื่อค้นหาข้อผิดพลาดที่					
อาจเกิดขึ้นได้(Integrated testing)					

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การจัดเตรียมรายการความ					
ต้องการที่มีความคงที่สูง	Yes	- - - - -	- - - - -	- - - - -	- - - - -
(Requirements are baseline at a high level)	No	+ + + + +	+ + + + +	+ + + + +	+ + + + +
เน้นการให้ความร่วมมือ และ					
ทำงานร่วมกับระหว่างผู้ที่มีส่วน	Yes	- - - - -	- - - - -	- - - - -	- - - - -
เกี่ยวข้องกับโครงการ	No	+ + + + +	+ + + + +	+ + + + +	+ + + + +
(Collaboration and cooperation among stakeholders)					

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การพัฒนาด้วยวิธีการจัดระบบตามลำดับขั้น (Staging)	Yes	C - C 0	C - C 0	C - C 0	C - C 0
	No	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++
การจัดตั้งทีมแบบองค์รวม (Holistic Diversity and Strategy) <sup>5</sup>	Yes	C - C 0	C - C 0	C - C 0	C - C 0
	No	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++
การพัฒนาแบบขนาน โดยมี					
การจัดลำดับชุดเหตุการณ์จากสถานะหนึ่ง ไปอีกสถานะหนึ่ง	Yes	C - C 0	C - C 0	C - C 0	C - C 0
แสดงให้เห็นสถานะของโครงการเป็นโครงข่ายกิจกรรม (Parallelism and Flux)	No	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การยินยอมให้ผู้ใช้เข้ามามีส่วนในการทบทวนการทำงานของบริษัท	Yes	---	---	---	---
แบบจำลองเพื่อพยายามส่งมอบซอฟต์แวร์ที่ตามมา	No	+++	+++	+++	+++
ทำงานได้ (User Viewings)					
การพัฒนาที่เน้นความสามารถในการนำร่องเพื่อพยายามส่งมอบซอฟต์แวร์ที่ตามมา	Yes	---	---	---	---
ทำงานได้ โดยยินยอมให้มีการทบทวนและปรับปรุงแก้ไข (Revision and review)	No	+++	+++	+++	+++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
วางแผนร่วมกับระหว่างผู้ที่เกี่ยวข้องกับการพัฒนาซอฟต์แวร์ก่อนจะเริ่มการวนรอบ โดยมีกำหนดรายละเอียด และขอบเขตการทำงานแต่ละวนรอบ (Planning Game)	Yes No	++ +++	-- +++	-- +++	-- +++
ผู้ที่ทำหน้าที่ดูแลทีมงาน เป็นโค้ชของทีมงาน เป็นคนที่รับผิดชอบคุณภาพของผลงาน จัดลำดับความสำคัญของงาน (Scrum master)	Yes No	-- ++	-- ++	-- ++	-- ++



ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
ประชุมวางแผนการมอบถึงเป้าหมาย และการพัฒนาเพิ่ม (Sprint Planning meeting)	Yes No	0 +++	0 +++	0 +++	0 +++
จัดประชุมทีมทุกวัน เพื่อติดตามความคืบหน้า และช่วยสำหรับการวางแผน และปรับปรุงกระบวนการพัฒนา (Daily scrum meeting)	Yes No	0 +++	0 +++	0 +++	0 +++
การรายงานความก้าวหน้าของซอฟต์แวร์จะถูกสร้างพร้อมกับผลลัพธ์ที่ได้ (Reporting/ visibility of results)	Yes No	0 +++	0 +++	0 +++	0 +++
การวางแผนมีการปรับตัวตามปัญหาเฉพาะหน้า (Adaptive cycle planning)	Yes No	0 +++	0 +++	0 +++	0 +++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
รูปแบบการบริหารจัดการโครงการ พัฒนามีการปรับตัว (Adaptive management model)	Yes	++	++	++	++
	No	++	++	++	++
กระบวนการพัฒนาซอฟต์แวร์ (Monitoring of a progress)	Yes	++	++	++	++
	No	++	++	++	++
การวางแผนจัดทำต้นแบบของ ซอฟต์แวร์ (Prototype planning)	Yes	++	++	++	++
	No	++	++	++	++
การบริหารจัดการเกี่ยวกับ กำหนดคุณสมบัติของคอมพิวเตอร์ อุปกรณ์หรือโปรแกรมใดๆ ที่จะ นำมาใช้กับคอมพิวเตอร์ เพื่อให้ ทำงานมีประสิทธิภาพ เหมาะสม กับงานที่ต้องการ (Configuration management)	Yes	++	++	++	++
	No	++	++	++	++

ตาราง 71 (ต่อ)

Technique / Practice	Usage	Decreased cycle time	Improved cost effective	Improved Quality of Process	Improved inter-supplier performance
การบริหารจัดการความเสี่ยง (Risk management)	Yes	C - C - C - C 0	C - C - C - C 0	C - C - C - C 0	C - C - C - C 0
	No	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++
การตรวจสอบโครงการภายหลัง (Project post mortem)	Yes	C - C - C 0	C - C - C - C 0	C - C - C - C 0	C - C - C - C 0
	No	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++
ปรับปรุงคุณภาพ					
กระบวนการพัฒนา จากการเรียนรู้ และองค์ความรู้ที่ได้จาก		C - C - C 0	C - C - C - C 0	C - C - C - C 0	C - C - C - C 0
การพัฒนาเพิ่มเติมแต่ละรอบ (Reflection workshops Methodology tuning)		C + C ++ C +++	C + C ++ C +++	C + C ++ C +++	C + C ++ C +++



- หมายเหตุ:** <sup>1</sup> Feature teams การจัดตั้งทีมที่มีขนาดเล็ก มีระดับความสามารถสูง ความคล่องตัวสูง และเต็มไปด้วยความคิดสร้างสรรค์ สมาชิกในทีมอาจจะต้องทำหน้าที่อื่นนอกเหนือจากทักษะที่ตนเองชำนาญ เช่น นักวิเคราะห์ระบบอาจจะต้องทำหน้าที่เป็นผู้ทดสอบ สถาปนิกที่รับผิดชอบด้วยคุณลักษณะของการที่มีรายละเอียดค่อนข้างมากกว่าโครงการอื่นๆ ที่จะมีมุ่งพัฒนา คุณลักษณะที่ตามมาเรื่อยๆ 2 สัปดาห์ เมื่อโครงการมีขนาดและความซับซ้อนมากขึ้น นักพัฒนา ผู้บริหารโครงการ และลูกค้าจำเป็นต้อง เข้าใจสถานการณ์ของคุณลักษณะของซอฟต์แวร์ทั้งหมด
- <sup>2</sup> The project mission development ขั้นตอนวางแผนการพัฒนาโดยใช้ข้อมูลเบื้องต้นของเขาของพัฒนาและข้อมูลพื้นฐานสำหรับการกำหนดภารกิจของโครงการเป็นตัวขับเคลื่อนการพัฒนา ซึ่งจะหาการพัฒนาจะเกิดความร่วมมือกัน และการเรียนรู้ส่งผลให้ภารกิจอาจมีการปรับเปลี่ยนได้ตลอดระยะเวลาที่ดำเนินการพัฒนา
- <sup>3</sup> Collaborative teams การทำงานด้วยความร่วมมือกันกับทุกฝ่ายที่มุ่งเน้นไปกับการเรียนรู้ไปพร้อมกับความก้าวหน้าไปสู่การเสร็จสิ้นงานทั้ง กลุ่มเฉพาะทาง (Focus Groups) การพบพบนทางเทคนิคอย่างเป็นทางการ (Formal Technical Review) และการตรวจสอบภายหลัง (Postmortems) เป็นต้น
- <sup>4</sup> Active user involvement พัฒนาเฉพาะส่วนที่เกี่ยวข้องกับการทำงานของผู้ใช้ โดยศึกษาความเป็นไปได้เกี่ยวกับโครงการ สิ่งที่สำคัญที่สุดคือปัญหาขององค์กร และผู้ปฏิบัติงาน และวิเคราะห์คุณลักษณะของธุรกิจ และเทคโนโลยี ทั้งนี้การรวบรวมข้อมูลทั้งหมดต้องไม่ประมาณความเข้าใจของตัวเองด้านเทคโนโลยี หรือด้านอื่นๆ เป็นจริง โดยการคิดประชุมเชิงปฏิบัติการ เพื่อรวบรวมความคิดเห็นในแง่มุมมองต่างๆ จากผู้เชี่ยวชาญของผู้ใช้ การพัฒนายอมให้มีการขอความเห็นส่วนของผู้ใช้ที่พัฒนาขึ้น และปรับปรุงคุณภาพ
- <sup>5</sup> การจัดตั้งทีมแบบองค์รวมที่มีความหลากหลาย ที่มีทักษะที่แตกต่างกันที่ทำงานร่วมกัน และมีผู้ดูแลตั้งแต่ในจุดกระบวนการวางแผน, กระบวนการทำงาน, ทดสอบ และพัฒนา ร่วมกัน เพื่อบรรลุเป้าหมายที่ข้อตกลงร่วมกัน

**ตอนที่ 5 ข้อมูลเกี่ยวกับทัศนคติที่มีต่อข้อบกพร่องการประยุกต์ใช้เทคนิค หรือ  
วิธีการพัฒนาระบบสำหรับโครงการพัฒนาซอฟต์แวร์เลือกคำตอบได้มากกว่า  
1 ข้อ**

ใช้เวลาสำหรับการวางแผนงานล่วงหน้ามากเกินไปจนขอบเขต (Comprehensive Upfront Planning)

ไม่ยอมรับต่อการเปลี่ยนแปลง (Not prone to change)

การลดเอกสารเกี่ยวกับการออกแบบ ส่งผลให้เกิดปัญหา

Customer on site หรือการยอมให้ลูกค้าเข้ามามีส่วนร่วม เป็นสิ่งที่ยากที่จะทำให้โครงการสำเร็จ

ความพยายามที่จะบริหารจัดการโดยนำเทคนิคของ Agile หรือ เทคโนโลยีใหม่ เข้ามาใช้ เป็นสิ่งที่ยาก

สมาชิกในทีมไม่สามารถยอมรับบางเทคนิคของ Agile

การควบคุมการวนรอบ (Iterative) ทำได้ยาก

การเปลี่ยนแปลงความต้องการยากต่อการควบคุม

ลูกค้าต้องการให้โครงการพัฒนามีการกำหนดงบประมาณตามที่ตกลงในสัญญา

ลูกค้าไม่สามารถจัดลำดับความสำคัญของความต้องการ

สิ่งแวดล้อมของการพัฒนามีรูปแบบกระจาย (Distributed Development Environment)

อื่นๆ โปรดระบุ.....

.....

.....

.....

คุณต้องการรับข้อสรุปของโครงการวิจัยนี้หรือไม่

- No
- Yes

E-mail address \_\_\_\_\_

**ขอขอบพระคุณทุกท่านที่ให้ความร่วมมือ  
พิมพ์พรณ ทิพย์แสง**

ภาคผนวก ข การวิเคราะห์แนวปฏิบัติหลักที่กรณีสึกษาเลือกใช้ในโครงการพัฒนาซอฟต์แวร์

การวิเคราะห์แนวปฏิบัติหลักที่กรณีสึกษาเลือกใช้ในโครงการพัฒนาซอฟต์แวร์ กรณีศึกษา 1 Alpha

ตาราง 72 แสดงการวิเคราะห์ระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีสึกษา Alpha

Phases	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Pre-Game					
Development					
Post-Game					
Practices					
Scrum master	0	0	0	0	0
Scrum teams	1	0	0	0	0
Product backlog	0	0	1	0	0
Sprint	0	1	0	0	0
Sprint planning meeting	1	0	1	1	0
Daily scrum meeting	1	0	0	0	1
Sprint review	0	1	0	1	1

ตาราง 72 (ต่อ)

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Short release	0	1	0	0	0
Metaphor	0	0	0	1	0
Simple design	0	0	0	1	0
Test frits Development	0	1	0	0	0
Refactoring	0	0	0	1	0
Collective ownership	1	0	0	0	0
Continuous integration	0	1	0	0	0
40-h week	0	0	0	0	0
On-site customer	1	0	1	0	0
Coding standards	0	0	0	1	0
Feature teams	1	0	0	0	0
<b>Total</b>	<b>5</b>	<b>5</b>	<b>3</b>	<b>6</b>	<b>2</b>
<b>Proportion</b>	<b>0.350</b>	<b>0.290</b>	<b>0.180</b>	<b>0.350</b>	<b>0.120</b>
<b>Value of Agility</b>	<b>35.290</b>	<b>29.410</b>	<b>17.650</b>	<b>35.290</b>	<b>11.760</b>



การวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 2 Beta

ตาราง 73 แสดงข้อมูลรายละเอียดการวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 2 Beta

Phases	Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Project initiation						
Development						
Testing						
Practices						
System concept		0	0	0	0	0
Analysis		0	0	0	0	0
Simple		0	0	0	0	0
Developing by components		0	1	0	0	0
Testing		0	0	0	0	0
Total		0	1	0	0	0
Proportion		0.000	0.200	0.000	0.000	0.000
Value of Agility		0.000	20.000	0.000	0.000	0.000

การวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 3 Delta

ตาราง 74 แสดงข้อมูลรายละเอียดการวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 3 Delta

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Phases					
Pre-Game					
Development					
Post-Game					
Practices					
Scrum master	0	0	0	0	0
Scrum teams	1	0	0	0	0
Product backlog	0	0	1	0	0
Sprint	0	1	0	0	0
Sprint planning meeting	1	0	1	1	0
Daily scrum meeting	1	0	0	0	1
Sprint review	0	1	0	1	1
Short release	0	1	0	0	0

ตาราง 74 (ต่อ)

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Metaphor	0	0	0	1	0
Simple design	0	0	0	1	0
Test frits Development	0	1	0	0	0
Refactoring	0	0	0	1	0
Pair programming	1	0	0	0	0
Collective ownership	1	0	0	0	0
Continuous integration	0	1	0	0	0
40-h week	0	0	0	0	0
On-site customer	1	0	1	0	0
Coding standards	0	0	0	1	0
Feature teams	1	0	0	0	0
<b>Total</b>	<b>7</b>	<b>5</b>	<b>3</b>	<b>6</b>	<b>2</b>
<b>Proportion</b>	<b>0.368</b>	<b>0.263</b>	<b>0.158</b>	<b>0.316</b>	<b>0.105</b>
<b>Value of Agility</b>	<b>36.840</b>	<b>26.320</b>	<b>15.790</b>	<b>31.580</b>	<b>10.530</b>

การวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 4 Zeta

ตาราง 75 แสดงข้อมูลรายละเอียดการวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 4 Zeta

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Phases					
Inception Phases					
Elaboration Phases					
Construction Phases					
Transition Phases					
Practices					
Develop Software Iteratively and Incremental	0	1	0	0	0
Manage Requirement	0	0	0	0	0
Use Component-Base Architectures	0	1	0	0	0
Visually Model Software	0	0	0	0	0
Verify Software Quality	0	1	0	0	0

ตาราง 75 (ต่อ)

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Control Changes to					
Software	0	0	0	0	0
Collective ownership	1	0	0	0	0
On-site customer	1	0	1	0	0
<b>Total</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>0</b>	<b>0</b>
<b>Proportion</b>	<b>0.286</b>	<b>0.429</b>	<b>0.143</b>	<b>0.000</b>	<b>0.000</b>
<b>Value of Agility</b>	<b>28.570</b>	<b>42.860</b>	<b>14.290</b>	<b>0.000</b>	<b>0.000</b>

การวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 5 Theta

ตาราง 76 แสดงข้อมูลรายละเอียดการวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 5 Theta

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Phases					
Exploration					
Planning					
Productionizing					
Maintenance					
Practices					
The planning game	1	0	1	0	0
Short release	0	1	0	0	0
Metaphor	0	0	0	1	0
Simple design	0	0	0	1	0
Refactoring	0	0	0	1	0

ตาราง 76 (ต่อ)

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Pair programming	1	0	0	0	0
Collective ownership	1	0	0	0	0
40-h week	0	0	0	0	0
On-site customer	1	0	1	0	0
Product backlog	0	0	1	0	0
Sprint	0	1	0	0	0
Sprint planning meeting	1	0	1	1	0
Sprint review	0	1	0	1	1
<b>Total</b>	<b>5</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>1</b>
<b>Proportion</b>	<b>0.385</b>	<b>0.231</b>	<b>0.308</b>	<b>0.385</b>	<b>0.077</b>
<b>Value of Agility</b>	<b>38.46</b>	<b>23.08</b>	<b>30.77</b>	<b>38.46</b>	<b>7.69</b>

การวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 6 lota

ตาราง 77 แสดงข้อมูลรายละเอียดการวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 6 lota

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Phases					
Project initiation					
Feasibility study					
Business study					
Development					
Testing					
Practices					
System concept	0	0	0	0	0
Analysis	0	0	0	0	0
Simple design	0	0	0	1	0
Requirements are baselined at High Level	0	0	1	0	0



ตาราง 77 (ต่อ)

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Developing by components	0	1	0	0	0
Testing	0	0	0	0	0
<b>Total</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>0</b>
<b>Proportion</b>	<b>0.000</b>	<b>0.200</b>	<b>0.200</b>	<b>0.200</b>	<b>0.000</b>
<b>Value of Agility</b>	<b>0.000</b>	<b>20.000</b>	<b>20.000</b>	<b>20.000</b>	<b>0.000</b>

การวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 7 Rho

ตาราง 78 แสดงข้อมูลรายละเอียดการวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 7 Rho

Phases	Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Exploration						
Productionizing						
Maintenance						
Practices						
The planning game	1	0	1	0	0	0
Short release	0	1	0	0	0	0
Metaphor	0	0	0	0	1	0
Simple design	0	0	0	0	1	0
Test frits Development	0	1	0	0	0	0
Refactoring	0	0	0	0	1	0
Collective ownership	1	0	0	0	0	0

ตาราง 78 (ต่อ)

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Continuous integration	0	1	0	0	0
40-h week	0	0	0	0	0
On-site customer	1	0	1	0	0
Coding standards	0	0	0	1	0
Feature teams	1	0	0	0	0
<b>Total</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>4</b>	<b>0</b>
<b>Proportion</b>	<b>0.333</b>	<b>0.250</b>	<b>0.167</b>	<b>0.333</b>	<b>0.000</b>
<b>Value of Agility</b>	<b>33.330</b>	<b>25.000</b>	<b>16.670</b>	<b>33.330</b>	<b>0.000</b>

การวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 8 Tau

ตาราง 79 แสดงข้อมูลรายละเอียดการวิเคราะห์แยกกลุ่มระเบียบวิธีการพัฒนาซอฟต์แวร์กรณีศึกษา 8 Tau

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Phases					
The project initiation					
Elaboration Phases					
Construction Phases					
Quality review					
Transition Phases					
Practices					
Develop Software Iteratively and Incremental	0	1	0	0	0
Manage Requirement	0	0	0	0	0
Use Component-Base Architectures	0	1	0	0	0

ตาราง 79 (ต่อ)

Practices	Interactions and collaboration	Working software	Customer-Oriented	Responding to change	Keeping the process agile
Visually Model Software	0	0	0	0	0
Verify Software Quality/ Software inspect	0	1	0	0	0
Control Changes to Software	0	0	0	0	0
Individual class ownership	1	0	0	0	0
Feature teams	1	0	0	0	0
Configuration management	0	0	0	1	0
<b>Total</b>	<b>2</b>	<b>3</b>	<b>0</b>	<b>1</b>	<b>0</b>
<b>Proportion</b>	<b>0.222</b>	<b>0.222</b>	<b>0.000</b>	<b>0.111</b>	<b>0.000</b>
<b>Value of Agility</b>	<b>22.220</b>	<b>22.220</b>	<b>0.000</b>	<b>11.110</b>	<b>0.000</b>

ภาคผนวก ค แผนการดำเนินงานของกรณีศึกษา

แผนการดำเนินงานของกรณีศึกษา Alpha

ตาราง 80 แสดงข้อมูลรายละเอียดของแผนการดำเนินงานของกรณีศึกษา Alpha

วงรอบการ พัฒนา	แผนการดำเนินงานจากการประเมิน			แผนการดำเนินงานจริง			% ค่า ความคลาด เคลื่อน	% ค่าความคลาดเคลื่อน แผนงานโดยรวม
	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา		
Pre-Game	11/2/2008	13/2/2008	3	11/2/2008	14/2/2008	3.5	16.667%	1.190%
Development	14/2/2008	10/3/2008	17	14/1/2008	12/3/2008	18.5	8.824%	3.571%
Post-Game	11/3/2008	14/3/2008	4	13/3/2008	18/3/2008	4	0.000%	0.000%
<b>รวม ระยะเวลาวางรอบวันแรก</b>			<b>24</b>			<b>26</b>	<b>8.333%</b>	<b>4.762%</b>
Pre-Game	19/3/2008	21/3/2008	3	19/3/2008	21/3/2008	3	0.000%	0.000%
Development	24/3/2008	8/4/2008	12	24/3/2008	11/4/2008	15	25.000%	7.143%
Post-Game	9/4/2008	11/4/2008	3	16/4/2008	19/4/2008	4	33.333%	2.381%
<b>รวม ระยะเวลาวางรอบวันที่สอง</b>			<b>18</b>			<b>22</b>	<b>22.222%</b>	<b>9.524%</b>
<b>รวม</b>			<b>42</b>			<b>48</b>	<b>14.286%</b>	

จากตาราง 80 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการวงรอบ (Iteration Plan) ครั้งที่ 1 ของกรณีศึกษา 1 Alpha พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการดังนี้ ขั้นตอนการวางแผน (Pre-game) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 16.667% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 1.190% ขั้นตอนการพัฒนา (Development) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 8.824% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 3.571% และขั้นตอนสุดท้าย (Post-Game) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% โดยการวงรอบครั้งที่ 1 มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 8.333% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 4.762%

ข้อมูลรายละเอียดของกิจกรรมการวางแผนการวงรอบ (Iteration Plan) ครั้งที่ 2 ของกรณีศึกษา 1 Alpha พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการ ดังนี้ ขั้นตอนการวางแผน (Pre-game) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% ขั้นตอนการพัฒนา (Development) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 25.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 7.143% และขั้นตอนสุดท้าย (Post-Game) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 33.333% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 2.381% โดยการวงรอบครั้งที่ 2 มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 22.222% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 9.524% การวงรอบทั้ง 2 ครั้งมีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 14.286%

แผนการดำเนินงานของกรณีศึกษา Beta

ตาราง 81 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์กรณีศึกษา 2 Beta

วงรอบการพัฒนา	แผนการดำเนินงานจากกรอบประเมิน				แผนการดำเนินงานจริง				% ค่า ความคลาด เคลื่อน	% ค่าความคลาดเคลื่อน แผนงานโดยรวม
	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา	ค่า เคลื่อน			
Project initiation	23/6/2008	11/7/2008	15	23/6/2008	16/7/2008	18	20.000%	3.448%		
Development	14/7/2008	13/10/2008	65	21/7/2008	31/10/2008	90	38.462%	28.736%		
Deployment	14/10/2008	22/10/2008	7	31/11/2008	21/11/2008	15	114.286%	9.195%		
<b>รวม</b>			<b>87</b>			<b>123</b>	<b>41.379%</b>			



จากตาราง 81 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการพัฒนาซอฟต์แวร์ของ  
กรณีศึกษา 2 Beta พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัด  
ค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการดังนี้ ขั้นตอนการเริ่มต้น (Project  
initiation) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 20.000% ส่งผลกระทบต่อค่า  
เบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 3.448% ขั้นตอนการพัฒนา (Development)  
มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 38.462% ส่งผลกระทบต่อค่าเบี่ยงเบนของ  
ระยะเวลาจากแผนงานโดยรวมคิดเป็น 28.736% และขั้นตอนการใช้งาน (Deployment) มีความ  
คาดเคลื่อนไปจากการประมาณการคิดเป็น 114.286% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลา  
จากแผนงานโดยรวมคิดเป็น 9.195%



แผนการดำเนินงานของกรณีศึกษา Delta

ตาราง 82 แสดงข้อมูลรายละเอียดกิจกรรมการวางแผนพัฒนาซอฟต์แวร์ของกรณีศึกษา 3 Delta

วงรอบการพัฒนา	แผนการดำเนินงานจากกรอบประเมิน				แผนการดำเนินงานจริง				% ค่าความคลาดเคลื่อน แผนงานโดยรวม
	วันที่เริ่มโครงการ	วันที่เสร็จสิ้นโครงการ	ระยะเวลา	วันที่เริ่มโครงการ	วันที่เสร็จสิ้นโครงการ	ระยะเวลา	% ค่าความคลาดเคลื่อน		
Pre-Game	8/1/2008	10/1/2008	3	8/1/2008	10/1/2008	3	0.000%	0.000%	
Development	11/1/2008	4/2/2008	17	11/1/2008	4/2/2008	17	0.000%	0.000%	
Post-Game	5/2/2008	11/2/2008	4	5/2/2008	6/2/2008	2	-50.000%	-4.762%	
<b>รวมระยะเวลากรอบแรก</b>			<b>24</b>			<b>22</b>	<b>-8.333%</b>	<b>-4.762%</b>	
12/2/2008	3	8/2/2008	12/2/2008	3	0.000%	0.000%	0.000%		
6/3/2008	16	13/2/2008	6/3/2008	16	0.000%	0.000%	0.000%		
11/3/2008	3	7/3/2008	11/3/2008	3	0.000%	0.000%	0.000%		
<b>รวมระยะเวลากรอบที่สอง</b>			<b>22</b>			<b>22</b>	<b>0.000%</b>	<b>0.000%</b>	

ตาราง 82 (ต่อ)

วงรอบการ พัฒนา	แผนการดำเนินงานจากกรอบประเมิน				แผนการดำเนินงานจริง			% ค่า ความคลาด เคลื่อน	% ค่าความคลาดเคลื่อน แผนงานโดยรวม
	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา			
Pre-Game	11/3/2008	13/3/2008	3	11/3/2008	13/3/2008	3	0.000%	0.000%	
Development	14/3/2008	31/3/2008	12	14/3/2008	31/3/2008	12	0.000%	0.000%	
Post-Game	1/4/2008	8/4/2008	6	1/4/2008	8/4/2008	6	0.000%	0.000%	
<b>รวมระยะเวลากรอบที่สาม</b>			<b>21</b>			<b>21</b>	<b>0.000%</b>	<b>0.000%</b>	
<b>รวม</b>			<b>67</b>			<b>66</b>	<b>-2.985%</b>		

จากตาราง 82 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการวงรอบ (Iteration Plan) ครั้งที่ 1 ของกรณีศึกษา 3 Delta พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการดังนี้ ขั้นตอนการวางแผน (Pre-game) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 10.000% ขั้นตอนการพัฒนา (Development) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% และขั้นตอนสุดท้าย (Post-Game) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -50.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -4.762% โดยการวงรอบครั้งที่ 1 มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -8.333% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -4.762%

ข้อมูลรายละเอียดของกิจกรรมการวางแผนการวงรอบ (Iteration Plan) ครั้งที่ 2 ของกรณีศึกษา 3 Delta พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการ ดังนี้ ขั้นตอนการวางแผน (Pre-game) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% ขั้นตอนการพัฒนา (Development) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% และขั้นตอนสุดท้าย (Post-Game) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% โดยการวงรอบครั้งที่ 2 มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000%

ข้อมูลรายละเอียดของกิจกรรมการวางแผนการวงรอบ (Iteration Plan) ครั้งที่ 3 ของกรณีศึกษา 3 Delta พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการ ดังนี้ ขั้นตอนการวางแผน (Pre-game) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% ขั้นตอนการพัฒนา (Development) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% และขั้นตอนสุดท้าย (Post-Game) มีความคาดเคลื่อนไปจากการประมาณ

การคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% โดยการวงรอบครั้งที่ 3 มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% การวงรอบทั้ง 3 ครั้ง มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -2.985%

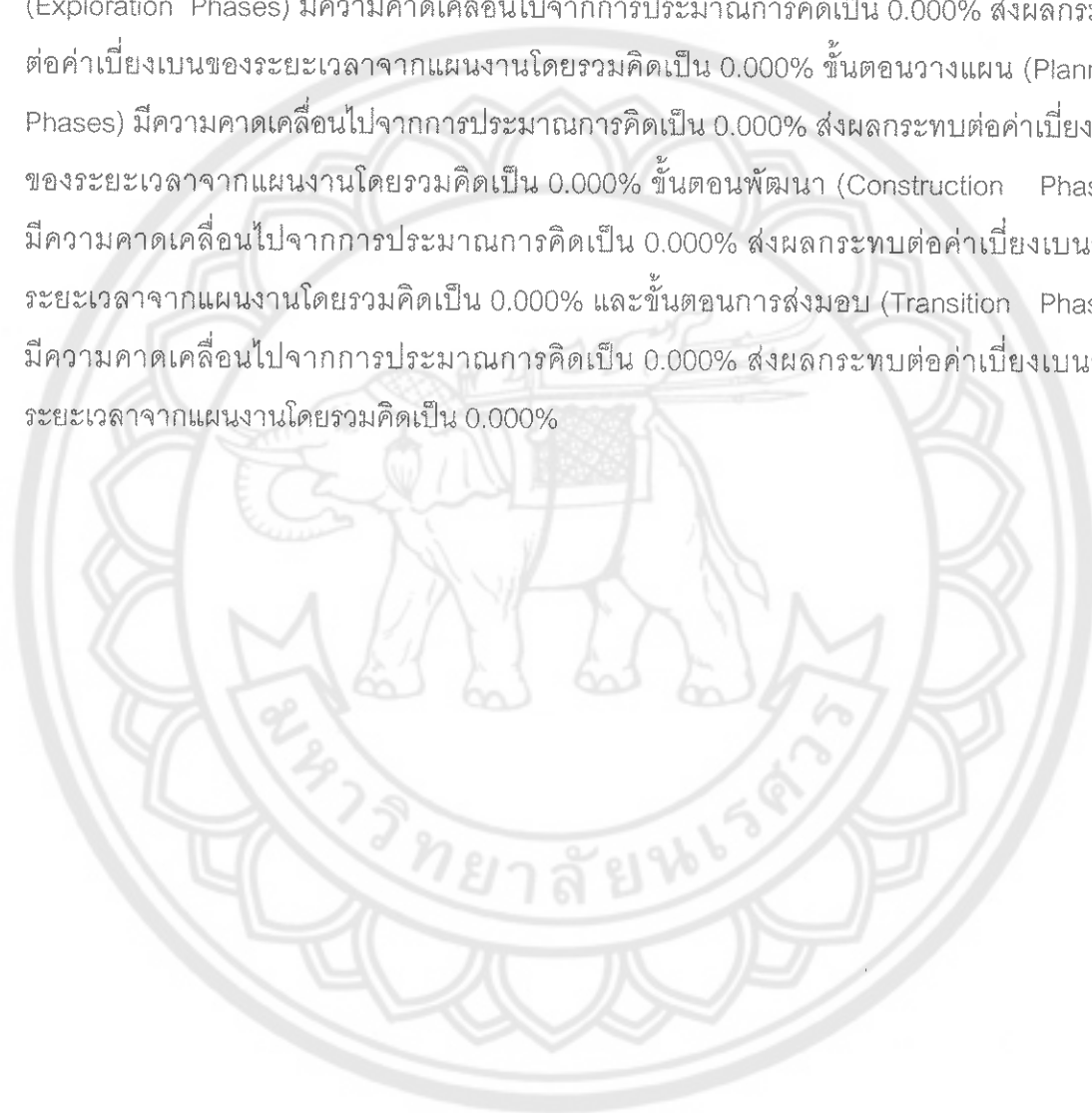


แผนการดำเนินงานของกรณีศึกษา Zeta

ตาราง 83 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์กรณีศึกษา กรณีศึกษา 4 Zeta

วงรอบการพัฒนา	แผนการดำเนินงานจากการประเมิน				แผนการดำเนินงานจริง				% ค่า ความคลาด เคลื่อน	% ค่าความคลาดเคลื่อน แผนงานโดยรวม
	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา	วันที่เริ่ม โครงการ	วันที่เริ่ม โครงการ	วันที่เสร็จ สิ้นโครงการ	ระยะเวลา	ความคลาด เคลื่อน		
Exploration Phases	6/3/2007	18/4/2007	30	6/3/2007	18/4/2007	30	0.000%	0.000%		
Planning Phases	19/4/2007	25/6/2007	45	19/4/2007	25/6/2007	45	0.000%	0.000%		
Construction Phases	26/6/2007	1/11/2007	90	26/6/2007	1/11/2007	90	0.000%	0.000%		
Transition Phases	2/11/2007	17/12/2007	30	2/11/2007	17/12/2007	30	0.000%	0.000%		
<b>รวม</b>			<b>195</b>			<b>195</b>	<b>0.000%</b>	<b>0.000%</b>		

ตาราง 83 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์ ของกรณีศึกษา 4 Zeta พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่า เบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการ ดังนี้ ขั้นตอนการสำรวจและค้นหา (Exploration Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% ขั้นตอนวางแผน (Planning Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% ขั้นตอนพัฒนา (Construction Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% และขั้นตอนการส่งมอบ (Transition Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000%



แผนการดำเนินงานของกรรณศึกษา Theta

ตาราง 84 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์กรรณศึกษา 5 Theta

วงรอบการพัฒนา	แผนการดำเนินงานจากการประเมิน				แผนการดำเนินงานจริง				% ค่า ความคลาด เคลื่อน	% ค่าความคลาดเคลื่อน แผนงานโดยรวม
	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา	วันที่เริ่ม โครงการ	วันที่เสร็จ สิ้นโครงการ	ระยะเวลา	ความคลาด เคลื่อน			
Exploration	1/4/2005	31/7/2005	122	13/4/2005	31/7/2005	110	-9.836%	-2.824%		
Planning	1/5/2005	31/7/2005	92	1/5/2005	24/4/2006	361	292.391%	63.294%		
Iteration to release	1/7/2005	30/11/2005	153	1/7/2005	8/8/2006	373	143.791%	51.765%		
Maintenance	1/11/2005	28/12/2005	58	1/11/2005	22/9/2006	334	475.862%	64.941%		
<b>รวม</b>			<b>425</b>			<b>1178</b>	<b>177.176%</b>			



ตาราง 84 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์ ของกรณีศึกษา 5 Theta พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการ ดังนี้ ขั้นตอนการสำรวจและค้นหา (Exploration Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -9.836% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -2.824% ขั้นตอนวางแผน (Planning Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 292.391% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 63.294% ขั้นตอนพัฒนา (Product ionizing Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 143.791% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 51.765% และขั้นตอนการทดสอบและการบำรุงรักษา (Maintenance) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 475.862% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 64.941% การพัฒนาซอฟต์แวร์ทั้งกระบวนการมีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 177.176%

แผนการดำเนินงานของกรณีสึกษา Iota

ตาราง 85 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์กรณีสึกษา Iota

วงรอบการพัฒนา	แผนการดำเนินงานจากการประเมิน				แผนการดำเนินงานจริง				% ค่าความคลาดเคลื่อน แผนงานโดยรวม
	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา	วันที่เริ่ม โครงการ	วันที่เสร็จ สิ้นโครงการ	ระยะเวลา	ความคลาด เคลื่อน		
Analysis (Business study Feasibility study)	23/3/2005	30/4/2005	39	1/3/2006	31/3/2006	31	-20.513%	-2.909%	
Design	23/4/2005	30/5/2005	38	1/4/2006	30/4/2006	30	-21.053%	-2.909%	
Development	23/5/2005	30/11/2005	160	1/5/2006	5/7/2006	66	-58.750%	-34.182%	
Testing and Integrate	23/11/2005	30/12/2005	38	1/7/2006	31/12/2007	540	1321.053%	182.545%	
<b>รวม</b>			<b>275</b>			<b>667</b>	<b>142.545%</b>		

ตาราง 85 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์ ของกรณีศึกษา 6 lota พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการ ดังนี้ ขั้นตอนการวิเคราะห์ระบบ (Analysis) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -20.513% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -2.909% ขั้นตอนออกแบบ (Design) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -21.053% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -2.909% ขั้นตอนพัฒนา (Development) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -58.750% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -34.182% และขั้นตอนการทดสอบและการบำรุงรักษา (Testing and Integrate) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 1321.053% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 182.545% การพัฒนาซอฟต์แวร์ทั้งกระบวนการมีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 142.545%

แผนการดำเนินงานของกรณีศึกษา Rho

ตาราง 86 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์กรณีศึกษา กรณีศึกษา 7 Rho

วงรอบการพัฒนา	แผนการดำเนินงานจากการประเมิน				แผนการดำเนินงานจริง			% ค่า ความคลาด เคลื่อน	% ค่าความคลาดเคลื่อน แผนงานโดยรวม
	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา	วันที่เริ่ม โครงการ	วันที่เสร็จ สิ้นโครงการ	ระยะเวลา	ค่า		
Exploration	3/3/2008	5/3/2008	3	3/3/2008	5/3/2008	3	0.000%	0.000%	
Product ionizing	6/3/2008	26/3/2008	15	6/3/2008	26/3/2008	15	0.000%	0.000%	
Maintenance	27/3/2008	31/3/2008	3	27/3/2008	31/3/2008	3	0.000%	0.000%	
<b>รวมระยะเวลาจรอบแรก</b>			<b>21</b>			<b>21</b>	<b>0.000%</b>	<b>0.000%</b>	
Exploration	1/4/2008	3/4/2008	3	1/4/2008	2/4/2008	2	-33.3333%	-2.381%	
Product ionizing	4/4/2008	25/4/2008	14	3/4/2008	25/4/2008	15	7.143%	2.381%	
Maintenance	28/4/2008	30/4/2008	3	28/4/2008	30/4/2008	3	0.000%	0.000%	
<b>รวมระยะเวลาจรอบที่สอง</b>			<b>20</b>			<b>20</b>	<b>0.000%</b>	<b>0.000%</b>	
Exploration	2/5/2008	7/5/2008	3	2/5/2008	6/5/2008	2	-33.3333%	-2.381%	

ตาราง 86 (ต่อ)

วงรอบการพัฒนา	แผนการดำเนินงานจากการประเมิน		แผนการดำเนินงานจริง		% ค่า ความคลาด เคลื่อน	% ค่าความคลาดเคลื่อน แผนงานโดยรวม
	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	วันที่เริ่ม โครงการ	วันที่เสร็จ สิ้นโครงการ		
			ระยะเวลา	ระยะเวลา		
Product ionizing	8/5/2008	30/5/2008	16	17	6.250%	2.381%
Maintenance	2/6/2008	13/6/2008	10	9	-10.000%	-2.381%
<b>รวมระยะเวลาวงรอบที่สาม</b>			29	28	-3.448%	-2.381%
<b>รวม</b>			70	69	-1.429%	

จากตาราง 86 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการวงรอบ (Iteration Plan) ครั้งที่ 1 ของกรณีศึกษา 7 Rho พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการดังนี้ ขั้นตอนการสำรวจและค้นหา (Exploration) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% ขั้นตอนการพัฒนา (Product ionizing) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% และขั้นตอนสุดท้าย (Maintenance) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -0.000% โดยการวงรอบครั้งที่ 1 มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000%

ข้อมูลรายละเอียดของกิจกรรมการวางแผนการวงรอบ (Iteration Plan) ครั้งที่ 2 ของกรณีศึกษา 7 Rho พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการ ดังนี้ ขั้นตอนการสำรวจและค้นหา (Exploration) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -33.333% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -2.381% ขั้นตอนการพัฒนา (Product ionizing) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 7.143% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 2.381% และขั้นตอนสุดท้าย (Maintenance) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% โดยการวงรอบครั้งที่ 2 มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000%

ข้อมูลรายละเอียดของกิจกรรมการวางแผนการวงรอบ (Iteration Plan) ครั้งที่ 3 ของกรณีศึกษา 7 Rho พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการ ดังนี้ ขั้นตอนการสำรวจและค้นหา (Exploration) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -33.333% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -2.381% ขั้นตอนการพัฒนา (Product ionizing) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 6.250% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 2.381% และขั้นตอนสุดท้าย (Maintenance) มีความ

คาดเคลื่อนไปจากการประมาณการคิดเป็น -10.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -2.381% โดยการวงรอบครั้งที่ 3 มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -3.448% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น -2.381% การวงรอบทั้ง 3 ครั้งมีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -1.429%



แผนการดำเนินงานของกรณีศึกษา Tau

ตาราง 87 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์กรณีศึกษา กรณีศึกษา 8 Tau

วงรอบการพัฒนา	แผนการดำเนินงานจากการประเมิน				แผนการดำเนินงานจริง				% ค่า ความคลาด เคลื่อน	% ค่ารวมคลาดเคลื่อน แผนงานโดยรวม
	วันที่เริ่ม โครงการ	วันที่เสร็จสิ้น โครงการ	ระยะเวลา	วันที่เริ่ม โครงการ	วันที่เริ่ม โครงการ	วันที่เสร็จ สิ้นโครงการ	ระยะเวลา	ความคลาด เคลื่อน		
Exploration	4/5/2007	22/5/2007	15	4/5/2007	23/5/2007	16	6.667%	0.690%		
Planning	23/5/2007	21/7/2007	40	24/5/2007	22/7/2007	40	0.000%	0.000%		
Construction	22/7/2007	6/12/2007	75	23/7/2007	7/12/2007	75	0.000%	0.000%		
Transition	7/12/2007	27/12/2007	15	8/12/2007	29/12/2007	16	6.667%	0.690%		
<b>รวม</b>			<b>145</b>			<b>147</b>	<b>1.379%</b>			



ตาราง 87 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนการพัฒนาซอฟต์แวร์ ของกรณีศึกษา 8 Tau พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารตารางเวลาจากการประมาณการ ดังนี้ ขั้นตอนการสำรวจและค้นหา (Exploration Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 6.667% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.690% ขั้นตอนวางแผน (Planning Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% ขั้นตอนพัฒนา (Construction Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.000% และขั้นตอนการส่งมอบ (Transition Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 6.667% - ส่งผลกระทบต่อค่าเบี่ยงเบนของระยะเวลาจากแผนงานโดยรวมคิดเป็น 0.690%



ภาคผนวก ง แผนงบประมาณของกรณีศึกษา

ตาราง 88 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 1 Alpha

วงรอบ การพัฒนา	แผนงบประมาณ		% ค่าคลาดเคลื่อน ของงบประมาณ ของงบประมาณ	% ค่าคลาดเคลื่อน ของงบประมาณจาก แผนงานโดยรวม
	ประมาณการ	ค่าใช้จ่ายจริง		
Iteration #1	48000	52312	8.983%	5.133%
Iteration #2	36000	44264	22.956%	9.838%
Total	84,000	96,576	14.971%	

จากตาราง 88 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 1 Alpha พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารงบประมาณจากการประมาณการดังนี้ การวงรอบ ครั้งที่ 1 (Iteration # 1) มีความคลาดเคลื่อนไปจากการประมาณการคิดเป็น 8.983% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 5.133% และการวงรอบ ครั้งที่ 2 (Iteration # 2) มีความคลาดเคลื่อนไปจากการประมาณการคิดเป็น 22.956% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 9.838%

การวงรอบทั้ง 2 ครั้งส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 14.971%

ตาราง 89 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา  
2 Beta

วงรอบ การพัฒนา	แผนงบประมาณ		% ค่าคลาดเคลื่อน ของงบประมาณ	% ค่าคลาดเคลื่อน ของงบประมาณจาก แผนงานโดยรวม
	ประมาณการ	ค่าใช้จ่ายจริง		
Iteration #1	150,000	184,500	23.000%	23.000%
ค่าปรับ	0	18,000		
รวม	150,000	202,500	35.000%	

จากตาราง 89 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 2 Beta พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีที่การวัดค่าเพียงเบนของการบริหารงบประมาณจากการประมาณการ มีความคลาดเคลื่อนไปจากการประมาณการคิดเป็น 23.000% ส่งผลกระทบต่อค่าเพียงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 23.000% และเสียค่าปรับเนื่องจากการส่งมอบงานล่าช้าส่งผลให้ค่าเพียงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 35.000%

ตาราง 90 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา  
3 Delta

วงรอบ การพัฒนา	แผนงบประมาณ		% ค่าคลาดเคลื่อน ของงบประมาณ	% ค่าคลาดเคลื่อน ของงบประมาณจาก แผนงานโดยรวม
	ประมาณการ	ค่าใช้จ่ายจริง		
Iteration #1	48000	44198	-7.921%	-2.837%
Iteration #2	44000	44000	0.000%	0.000%
Iteration #3	42000	42000	0.000%	0.000%
Total	134,000	130,198	-2.837%	

จากตาราง 90 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 3 Delta พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีที่การวัดค่าเบี่ยงเบนของการบริหารงบประมาณจากการประมาณการดังนี้ การวงรอบ ครั้งที่ 1 (Iteration # 1) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -7.921% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น -2.837%

การวงรอบ ครั้งที่ 2 (Iteration # 2) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 0.000%

การวงรอบ ครั้งที่ 3 (Iteration # 3) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 0.000%

การวงรอบทั้ง 3 ครั้งมีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -2.837%

ตาราง 91 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 4 Zeta

วงรอบ การพัฒนา	แผนงบประมาณ		% ค่าคลาดเคลื่อน ของงบประมาณ	% ค่าคลาดเคลื่อน ของงบประมาณจาก แผนงานโดยรวม
	ประมาณการ	ค่าใช้จ่ายจริง		
Exploration Phases	142,600	141,568	-0.724%	-0.069%
Planning Phases	184,000	176,200	-4.239%	-0.520%
Construction Phases	441,600	439,040	-0.580%	-0.171%
Transition Phases	151,800	136,192	-10.282%	-1.041%
Outsource	580,000	580,000	0.000%	0.000%
<b>รวม</b>	<b>1,500,000</b>	<b>1,473,000</b>	<b>-1.800%</b>	

จากตาราง 91 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 4 Zeta พบว่าการดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารงบประมาณจากการประมาณการดังนี้ ขั้นตอนการสำรวจและค้นหา (Exploration Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -0.724% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น -0.069% ขั้นตอนวางแผน (Planning Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -4.239% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น -0.520% ขั้นตอนพัฒนา (Construction Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -0.580% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น -0.171% และขั้นตอนการส่งมอบ (Transition Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -10.282% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น -1.041% ค่าใช้จ่ายสำหรับทีมภายนอก (Outsource) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 0.000% การพัฒนาซอฟต์แวร์ทั้งกระบวนการมีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -1.800%

ตาราง 92 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 5 Theta

วงรอบ การพัฒนา	แผนงบประมาณ		% ค่าคลาดเคลื่อน ของงบประมาณ	% ค่าคลาดเคลื่อน ของงบประมาณจาก แผนงานโดยรวม
	ประมาณการ	ค่าใช้จ่ายจริง		
Exploration	10,000	60,000	500.000%	62.500%
Planning	10,000	45,000	350.000%	43.750%
Iteration to release	20,000	330,000	1500.000%	387.500%
Maintenance	40,000	297,000	642.500%	321.250%
<b>รวม</b>	<b>80,000</b>	<b>732,000</b>	<b>815.000%</b>	

จากตาราง 92 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 5 Theta พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารงบประมาณจากการประมาณการดังนี้ ขั้นตอนการสำรวจและค้นหา (Exploration Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 500.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 62.500% ขั้นตอนวางแผน (Planning Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 350.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 43.750% ขั้นตอนพัฒนา (Product ionizing Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 1500.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 387.500% และขั้นตอนการส่งมอบ (Maintenance) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 642.500% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 321.250% การพัฒนาซอฟต์แวร์ทั้งกระบวนการมีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 815.000%

ตาราง 93 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 6 Iota

วงรอบ การพัฒนา	แผนงบประมาณ		% ค่าคลาดเคลื่อน ของงบประมาณ	% ค่าคลาดเคลื่อน ของงบประมาณจาก แผนงานโดยรวม
	ประมาณการ	ค่าใช้จ่ายจริง		
Analysis	7,580	30,000	295.778%	11.210%
Design	9,096	20,000	119.877%	5.452%
Development	83,324	100,000	20.013%	8.338%
Testing and Integrate	100,000	270,000	170.000%	85.000%
<b>รวม</b>	<b>200,000</b>	<b>420,000</b>	<b>110.000%</b>	

จากตาราง 93 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 6 lota พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของต้นทุนที่การวัดค่าเบี่ยงเบนของการบริหารงบประมาณจากการประมาณการดังนี้ ขั้นตอนการวิเคราะห์ระบบ (Analysis) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 295.778% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 11.210% ขั้นตอนออกแบบ (Design) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 119.877% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 5.452% ขั้นตอนพัฒนา (Development) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 20.013% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 8.338% และขั้นตอนการทดสอบและบำรุงรักษา (Testing and Integrate) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 170.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 85.000% การพัฒนาซอฟต์แวร์ทั้งกระบวนการมีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 110.000%

ตาราง 94 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 7 Rho

วงรอบ การพัฒนา	แผนงบประมาณ		% ค่าคลาดเคลื่อน ของงบประมาณ	% ค่าคลาดเคลื่อน ของงบประมาณจาก แผนงานโดยรวม
	ประมาณการ	ค่าใช้จ่ายจริง		
Iteration #1	81690	81690	0.000%	0.000%
Iteration #2	79800	79800	0.000%	0.000%
Iteration #3	118610	114520	-3.448%	-3.052%
Total	280,100	276,010	-1.460%	

จากตาราง 94 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 7 Rho พบว่า การดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีที่การวัดค่าเบี่ยงเบนของการบริหารงบประมาณจากการประมาณการดังนี้ การวงรอบ ครั้งที่ 1 (Iteration # 1) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 0.000%

การวงรอบ ครั้งที่ 2 (Iteration # 2) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 0.000%

การวงรอบ ครั้งที่ 3 (Iteration # 3) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -3.448% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น -3.052%

การวงรอบทั้ง 3 ครั้งมีความคาดเคลื่อนไปจากการประมาณการคิดเป็น -1.460%

ตาราง 95 แสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 8 Tau

วงรอบ การพัฒนา	แผนงบประมาณ		% ค่าคลาดเคลื่อน ของงบประมาณ	% ค่าคลาดเคลื่อน ของงบประมาณจาก แผนงานโดยรวม
	ประมาณการ	ค่าใช้จ่ายจริง		
Exploration Phases	57,750	61,600	0.067%	0.008%
Iteration to release Planning Phases	138,000	141,600	0.026%	0.007%
Construction Phases	258,750	258,750	0.000%	0.000%
Transition Phases	31,050	31,050	0.000%	0.000%
<b>รวม</b>	<b>485,550</b>	<b>493,000</b>	<b>1.534%</b>	



จากตาราง 95 ซึ่งแสดงข้อมูลรายละเอียดของกิจกรรมการวางแผนงบประมาณของกรณีศึกษา 8 Tau พบว่าการดำเนินกิจกรรมการพัฒนาซอฟต์แวร์มีเปอร์เซ็นต์ของดัชนีชี้การวัดค่าเบี่ยงเบนของการบริหารงบประมาณจากการประมาณการดังนี้ ขั้นตอนการสำรวจและค้นหา (Exploration Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.067% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 0.008% ขั้นตอนวางแผน (Planning Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.026% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 0.007% ขั้นตอนพัฒนา (Construction Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.00 0% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 0.000% และขั้นตอนการส่งมอบ (Transition Phases) มีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 0.000% ส่งผลกระทบต่อค่าเบี่ยงเบนของงบประมาณจากแผนงานโดยรวมคิดเป็น 0.000% การพัฒนาซอฟต์แวร์ทั้งกระบวนการมีความคาดเคลื่อนไปจากการประมาณการคิดเป็น 1.534%

## ภาคผนวก จ คุณภาพของกระบวนการพัฒนาซอฟต์แวร์

ตาราง 96 แสดงข้อมูลรายละเอียดของคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ของ  
กรณีศึกษา 1 Alpha

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
การบรรลุเป้าหมายของโครงการ (Effectiveness)	5
ความพึงพอใจของโครงการ และผู้มีส่วนได้ส่วนเสียของโครงการ	5
ความอยู่รอดของโครงการด้านระยะเวลา และค่าใช้จ่าย (Sustainability)	4
การกระจายงานในแต่ละขั้นตอน (Equity)	5
ผลผลิตเปรียบเทียบกับเป้าหมายรวมกิจกรรมที่ทำแล้วเสร็จ ทรัพยากร และเวลาที่ใช้ไป ได้ตรงตามเป้าหมาย (Progress)	4
ลดความสูญเสียที่เกิดขึ้นสำหรับโครงการพัฒนาซอฟต์แวร์	4
ค่าเฉลี่ย	4.50

จากตาราง 96 แปลความหมายได้ว่าเกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์  
ของกรณีศึกษา 1 Alpha มีค่าเฉลี่ย 4.50 หมายถึง ระดับความพึงพอใจมากที่สุดด้านคุณภาพของ  
กระบวนการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการ อายิลล์  
(Agile)

ตาราง 97 แสดงข้อมูลรายละเอียดของคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ของ  
กรณีศึกษา 2 Beta

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
การบรรลุเป้าหมายของโครงการ (Effectiveness)	2
ความพึงพอใจของโครงการ และผู้มีส่วนได้ส่วนเสียของโครงการ	2
ความอยู่รอดของโครงการด้านระยะเวลา และค่าใช้จ่าย ( Sustainability)	2
การกระจายงานในแต่ละขั้นตอน (Equity)	2
ผลผลิตเปรียบเทียบกับเป้าหมายรวมกิจกรรมที่ทำแล้วเสร็จ ทรัพยากร และเวลาที่ใช้ไป ได้ตรงตามเป้าหมาย (Progress)	2
ลดความสูญเสียที่เกิดขึ้นสำหรับโครงการพัฒนาซอฟต์แวร์	1
ค่าเฉลี่ย	1.83

จากตาราง 97 แปลความหมายได้ว่าเกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์  
ของกรณีศึกษา 2 Beta มีค่าเฉลี่ย 1.83 หมายถึง ระดับความพึงพอใจน้อยด้านคุณภาพของ  
กระบวนการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการ แบบจำลอง  
น้ำตก (Waterfall Model)

ตาราง 98 แสดงข้อมูลรายละเอียดของคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ของ  
กรณีศึกษา 3 Delta

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
การบรรลุเป้าหมายของโครงการ (Effectiveness)	5
ความพึงพอใจของโครงการ และผู้มีส่วนได้ส่วนเสียของโครงการ	5
ความอยู่รอดของโครงการด้านระยะเวลา และค่าใช้จ่าย ( Sustainability)	5
การกระจายงานในแต่ละขั้นตอน (Equity)	5
ผลผลิตเปรียบเทียบกับเป้าหมายรวมกิจกรรมที่ทำแล้วเสร็จ ทรัพยากร และเวลาที่ใช้ไปได้ตรงตามเป้าหมาย (Progress)	5
ลดความสูญเสียที่เกิดขึ้นสำหรับโครงการพัฒนาซอฟต์แวร์	5
<b>ค่าเฉลี่ย</b>	<b>5.00</b>

จากตาราง 98 แปลความหมายได้ว่าเกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์ของกรณีศึกษา 3 Delta มีค่าเฉลี่ย 5.00 หมายถึง ระดับความพึงพอใจมากที่สุดด้านคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการ อัจฉริยะ (Agile)

ตาราง 99 แสดงข้อมูลรายละเอียดของคุณภาพของกระบวนการพัฒนาซอฟต์แวร์  
ของกรณีศึกษา 4 Zeta

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
การบรรลุเป้าหมายของโครงการ (Effectiveness)	4
ความพึงพอใจของโครงการ และผู้มีส่วนได้ส่วนเสียของโครงการ	4
ความอยู่รอดของโครงการด้านระยะเวลา และค่าใช้จ่าย ( Sustainability)	4
การกระจายงานในแต่ละขั้นตอน (Equity)	3
ผลผลิตเปรียบเทียบกับเป้าหมายรวมกิจกรรมที่ทำแล้วเสร็จ ทรัพยากร และเวลาที่ใช้ไปได้ตรงตามเป้าหมาย (Progress)	4
ลดความสูญเสียที่เกิดขึ้นสำหรับโครงการพัฒนาซอฟต์แวร์	4
<b>ค่าเฉลี่ย</b>	<b>3.83</b>

จากตาราง 99 แปลความหมายได้ว่าเกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์ของกรณีศึกษา 4 Zeta มีค่าเฉลี่ย 3.83 หมายถึง ระดับความพึงพอใจมากด้านคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการ อาใจล์แบบไม่สมบูรณ (Ad-Hoc)

ตาราง 100 แสดงข้อมูลรายละเอียดของคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ของกรณีศึกษา 5 Theta

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
การบรรลุเป้าหมายของโครงการ (Effectiveness)	3
ความพึงพอใจของโครงการ และผู้มีส่วนได้ส่วนเสียของโครงการ	4
ความอยู่รอดของโครงการด้านระยะเวลา และค่าใช้จ่าย (Sustainability)	2
การกระจายงานในแต่ละขั้นตอน (Equity)	3
ผลผลิตเปรียบเทียบกับเป้าหมายรวมกิจกรรมที่ทำแล้วเสร็จ ทรัพยากร และเวลาที่ใช้ไป ได้ตรงตามเป้าหมาย (Progress)	2
ลดความสูญเสียที่เกิดขึ้นสำหรับโครงการพัฒนาซอฟต์แวร์	2
<b>ค่าเฉลี่ย</b>	<b>2.67</b>

จากตาราง 100 แปลความหมายได้ว่าเกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์ของกรณีศึกษา 5 Theta มีค่าเฉลี่ย 2.67 หมายถึง ระดับความพึงพอใจปานกลางด้านคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการ อาใจล์ (Agile)

ตาราง 101 แสดงข้อมูลรายละเอียดของคุณภาพของกระบวนการพัฒนาซอฟต์แวร์  
ของกรณีศึกษา 6 lota

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
การบรรลุเป้าหมายของโครงการ (Effectiveness)	4
ความพึงพอใจของโครงการ และผู้มีส่วนได้ส่วนเสียของโครงการ	3
ความยืดหยุ่นของโครงการด้านระยะเวลา และค่าใช้จ่าย (Sustainability)	2
การกระจายงานในแต่ละขั้นตอน (Equity)	3
ผลผลิตเปรียบเทียบกับเป้าหมายรวมกิจกรรมที่ทำแล้วเสร็จ ทรัพยากร และเวลาที่ใช้ไปได้ตรงตามเป้าหมาย (Progress)	2
ลดความสูญเสียที่เกิดขึ้นสำหรับโครงการพัฒนาซอฟต์แวร์	3
<b>ค่าเฉลี่ย</b>	<b>2.83</b>

จากตาราง 101 แปลความหมายได้ว่าเกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์ของกรณีศึกษา 6 lota มีค่าเฉลี่ย 2.83 หมายถึง ระดับความพึงพอใจปานกลางด้านคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการแบบจำลองน้ำตก (Waterfall Model)

ตาราง 102 แสดงข้อมูลรายละเอียดของคุณภาพของกระบวนการพัฒนาซอฟต์แวร์  
ของกรณีศึกษา 7 Rho

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
การบรรลุเป้าหมายของโครงการ (Effectiveness)	5
ความพึงพอใจของโครงการ และผู้มีส่วนได้ส่วนเสียของโครงการ	5
ความอยู่รอดของโครงการด้านระยะเวลา และค่าใช้จ่าย ( Sustainability)	4
การกระจายงานในแต่ละขั้นตอน (Equity)	4
ผลผลิตเปรียบเทียบกับเป้าหมายรวมกิจกรรมที่ทำแล้วเสร็จ ทรัพยากร และเวลาที่ใช้ไปได้ตรงตามเป้าหมาย (Progress)	5
ลดความสูญเสียที่เกิดขึ้นสำหรับโครงการพัฒนาซอฟต์แวร์	4
ค่าเฉลี่ย	4.50

จากตาราง 102 แปลความหมายได้ว่าเกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์ของกรณีศึกษา 7 Rho มีค่าเฉลี่ย 4.50 หมายถึง ระดับความพึงพอใจมากที่สุดด้านคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการ อาจิลล์ (Agile)

ตาราง 103 แสดงข้อมูลรายละเอียดของคุณภาพของกระบวนการพัฒนาซอฟต์แวร์  
ของกรณีศึกษา 8 Tau

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
การบรรลุเป้าหมายของโครงการ (Effectiveness)	5
ความพึงพอใจของโครงการ และผู้มีส่วนได้ส่วนเสียของโครงการ	4
ความอยู่รอดของโครงการด้านระยะเวลา และค่าใช้จ่าย ( Sustainability)	4
การกระจายงานในแต่ละขั้นตอน (Equity)	4
ผลผลิตเปรียบเทียบกับเป้าหมายรวมกิจกรรมที่ทำแล้วเสร็จ ทรัพยากร และเวลาที่ใช้ไป ได้ตรงตามเป้าหมาย (Progress)	4
ลดความสูญเสียที่เกิดขึ้นสำหรับโครงการพัฒนาซอฟต์แวร์	4
ค่าเฉลี่ย	4.17

จากตาราง 103 แปลความหมายได้ว่าเกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์  
ของกรณีศึกษา 8 Tau มีค่าเฉลี่ย 4.17 หมายถึง ระดับความพึงพอใจมากด้านคุณภาพของกระบวนการ  
พัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการ อาใจล์แบบไม่สมบูรณ์  
(Ad-Hoc)



ภาคผนวก จ การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์

ตาราง 104 แสดงข้อมูลรายละเอียดของการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ของกรณีศึกษา 1 Alpha

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
ความร่วมมือกัน และช่วยสนับสนุนความสามารถในการบรรลุข้อตกลงของโครงการ (Collaborative)	5
การทำหน้าที่เป็นตัวแทน สร้างองค์กรเสมือนจริง ด้วยการสร้างแม่แบบ ประกอบด้วยมุมมองที่สามารถแลกเปลี่ยนแนวคิดกันระหว่างกลุ่ม (Representative)	4
การมีสิทธิตัดสินใจเฉพาะหน้าในเรื่องการเปลี่ยนแปลงความต้องการ (Authorized)	5
การให้ความสำคัญกับสมาชิกในทีมแต่ละคนเท่าๆ กัน โดยการมอบหมายความดูแล หรือ ความรับผิดชอบสำหรับการพัฒนา (Committed)	5
สร้างความรู้ความเข้าใจในระบบงาน และช่วยสนับสนุนให้เกิดการแลกเปลี่ยนความรู้เกี่ยวกับระบบงานที่จะพัฒนาของสมาชิกในทีม (Knowledgeable)	5
ความร่วมมือกัน และช่วยสนับสนุนความสามารถในการบรรลุข้อตกลงของโครงการ (Collaborative)	5
การทำหน้าที่เป็นตัวแทน สร้างองค์กรเสมือนจริง ด้วยการสร้างแม่แบบ ประกอบด้วยมุมมองที่สามารถแลกเปลี่ยนแนวคิดกันระหว่างกลุ่ม (Representative)	4
การมีสิทธิตัดสินใจเฉพาะหน้าในเรื่องการเปลี่ยนแปลงความต้องการ (Authorized)	5
การให้ความสำคัญกับสมาชิกในทีมแต่ละคนเท่าๆกัน โดยการมอบหมายความดูแล หรือ ความรับผิดชอบสำหรับการพัฒนา (Committed)	5
สร้างความรู้ความเข้าใจในระบบงาน และช่วยสนับสนุนให้เกิดการแลกเปลี่ยนความรู้เกี่ยวกับระบบงานที่จะพัฒนาของสมาชิกในทีม (Knowledgeable)	5
<b>ค่าเฉลี่ย</b>	<b>4.8</b>

จากตาราง 104 แปลความหมายได้ว่าเกณฑ์การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ของกรณีศึกษา 1 Alpha มีค่าเฉลี่ย 4.80 หมายถึง ระดับความพึงพอใจมากที่สุด การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการ อาไจล์ (Agile)

ตาราง 105 แสดงข้อมูลรายละเอียดของการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ของกรณีศึกษา 2 Beta

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
ความร่วมมือกัน และช่วยสนับสนุนความสามารถในการบรรลุข้อตกลงของโครงการ (Collaborative)	2
การทำหน้าที่เป็นตัวแทน สร้างองค์กรเสมือนจริง ด้วยการสร้างแม่แบบ ประกอบด้วยมุมมองที่สามารถแลกเปลี่ยนแนวคิดกันระหว่างกลุ่ม (Representative)	2
การมีสิทธิตัดสินใจเฉพาะหน้าในเรื่องการเปลี่ยนแปลงความต้องการ (Authorized)	2
การให้ความสำคัญกับสมาชิกในทีมแต่ละคนเท่าๆกัน โดยการมอบหมายความดูแล หรือความรับผิดชอบสำหรับการพัฒนา (Committed)	2
สร้างความรู้ความเข้าใจในระบบงาน และช่วยสนับสนุนให้เกิดการแลกเปลี่ยนความรู้เกี่ยวกับระบบงานที่จะพัฒนาของสมาชิกในทีม (Knowledgeable)	2
ค่าเฉลี่ย	2

จากตาราง 105 แปลความหมายได้ว่าเกณฑ์การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ของกรณีศึกษา 2 Beta มีค่าเฉลี่ย 2.00 หมายถึง ระดับความพึงพอใจน้อยด้านการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการ แบบจำลองน้ำตก (Waterfall Model)

ตาราง 106 แสดงข้อมูลรายละเอียดของการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการ  
พัฒนาซอฟต์แวร์ของกรณีศึกษา 3 Delta

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
ความร่วมมือกัน และช่วยสนับสนุนความสามารถในการบรรลุข้อตกลงของโครงการ (Collaborative)	5
การทำหน้าที่เป็นตัวแทน สร้างองค์กรเสมือนจริง ด้วยการสร้างแม่แบบ ประกอบด้วยมุมมอง ที่สามารถแลกเปลี่ยนแนวคิดกันระหว่างกลุ่ม (Representative)	5
การมีสิทธิตัดสินใจเฉพาะหน้าในเรื่องการเปลี่ยนแปลงความต้องการ (Authorized)	5
การให้ความสำคัญกับสมาชิกในทีมแต่ละคนเท่าๆกัน โดยการมอบหมายความดูแล หรือ ความรับผิดชอบสำหรับการพัฒนา (Committed)	5
สร้างความรู้ความเข้าใจในระบบงาน และช่วยสนับสนุนให้เกิดการแลกเปลี่ยนความรู้เกี่ยวกับ ระบบงานที่จะพัฒนาของสมาชิกในทีม (Knowledgeable)	5
ค่าเฉลี่ย	5

จากตาราง 106 แปลความหมายได้ว่าเกณฑ์การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับ  
โครงการพัฒนาซอฟต์แวร์ของกรณีศึกษา 3 Delta มีค่าเฉลี่ย 5.00 ระดับความพึงพอใจมากที่สุด  
ด้านการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการ  
พัฒนาซอฟต์แวร์ตามหลักการ อาไจล์ (Agile)

ตาราง 107 แสดงข้อมูลรายละเอียดของการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการ  
พัฒนาซอฟต์แวร์ของกรณีศึกษา 4 Zeta

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
ความร่วมมือกัน และช่วยสนับสนุนความสามารถในการบรรลุข้อตกลงของโครงการ (Collaborative)	4
การทำหน้าที่เป็นตัวแทน สร้างองค์การเสมือนจริง ด้วยการสร้างแม่แบบ ประกอบด้วยมุมมองที่สามารถแลกเปลี่ยนแนวคิดกันระหว่างกลุ่ม (Representative)	3
การมีสิทธิตัดสินใจเฉพาะหน้าในเรื่องการเปลี่ยนแปลงความต้องการ (Authorized)	2
การให้ความสำคัญกับสมาชิกในทีมแต่ละคนเท่าๆกัน โดยการมอบหมายความดูแล หรือความรับผิดชอบสำหรับการพัฒนา (Committed)	3
สร้างความรู้ความเข้าใจในระบบงาน และช่วยสนับสนุนให้เกิดการแลกเปลี่ยนความรู้เกี่ยวกับระบบงานที่จะพัฒนาของสมาชิกในทีม (Knowledgeable)	3
ค่าเฉลี่ย	3

จากตาราง 107 แปลความหมายได้ว่าเกณฑ์การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ของกรณีศึกษา 4 Zeta มีค่าเฉลี่ย 3.00 ระดับความพึงพอใจปานกลางด้านการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนาซอฟต์แวร์ตามหลักการ อาไจล์แบบไม่สมบูรณ์ (Ad-Hoc)

ตาราง 108 แสดงข้อมูลรายละเอียดของการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการ  
พัฒนาซอฟต์แวร์ของกรณีศึกษา 5 Theta

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
ความร่วมมือกัน และช่วยสนับสนุนความสามารถในการบรรลุข้อตกลงของโครงการ (Collaborative)	4
การทำหน้าที่เป็นตัวแทน สร้างองค์กรเสมือนจริง ด้วยการสร้างแม่แบบ ประกอบด้วยมุมมอง ที่สามารถแลกเปลี่ยนแนวคิดกันระหว่างกลุ่ม (Representative)	3
การมีสิทธิตัดสินใจเฉพาะหน้าในเรื่องการเปลี่ยนแปลงความต้องการ (Authorized)	4
การให้ความสำคัญกับสมาชิกในทีมแต่ละคนเท่าๆกัน โดยการมอบหมายความดูแล หรือ ความรับผิดชอบสำหรับการพัฒนา (Committed)	4
สร้างความรู้ความเข้าใจในระบบงาน และช่วยสนับสนุนให้เกิดการแลกเปลี่ยนความรู้เกี่ยวกับ ระบบงานที่จะพัฒนาของสมาชิกในทีม (Knowledgeable)	3
ค่าเฉลี่ย	3.6

จากตาราง 108 แปลความหมายได้ว่าเกณฑ์การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับ  
โครงการพัฒนาซอฟต์แวร์ของกรณีศึกษา 5 Theta มีค่าเฉลี่ย 3.60 ระดับความพึงพอใจมากด้าน  
การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการ  
พัฒนาซอฟต์แวร์ตามหลักการ อาไจล์ (Agile)

ตาราง 109 แสดงข้อมูลรายละเอียดของการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการ  
พัฒนาซอฟต์แวร์ของกรณีศึกษา 6 Iota

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
ความร่วมมือกัน และช่วยสนับสนุนความสามารถในการบรรลุข้อตกลงของโครงการ (Collaborative)	4
การทำหน้าที่เป็นตัวแทน สร้างองค์กรเสมือนจริง ด้วยการสร้างแม่แบบ ประกอบด้วยมุมมอง ที่สามารถแลกเปลี่ยนแนวคิดกันระหว่างกลุ่ม (Representative)	3
การมีสิทธิตัดสินใจเฉพาะหน้าในเรื่องการเปลี่ยนแปลงความต้องการ (Authorized)	3
การให้ความสำคัญกับสมาชิกในทีมแต่ละคนเท่าๆกัน โดยการมอบหมายความดูแล หรือ ความรับผิดชอบสำหรับการพัฒนา (Committed)	4
สร้างความรู้ความเข้าใจในระบบงาน และช่วยสนับสนุนให้เกิดการแลกเปลี่ยนความรู้เกี่ยวกับ ระบบงานที่จะพัฒนาของสมาชิกในทีม (Knowledgeable)	2
ค่าเฉลี่ย	3.02

จากตาราง 109 แปลความหมายได้ว่าเกณฑ์การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับ  
โครงการพัฒนาซอฟต์แวร์ของกรณีศึกษา 6 Iota มีค่าเฉลี่ย 3.20 ระดับความพึงพอใจปานกลาง  
ด้านการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการ  
พัฒนาซอฟต์แวร์ตามหลักการ แบบจำลองน้ำตก (Waterfall Model)

ตาราง 110 แสดงข้อมูลรายละเอียดของการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการ  
พัฒนาซอฟต์แวร์ของกรณีศึกษา 7 Rho

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
ความร่วมมือกัน และช่วยสนับสนุนความสามารถในการบรรลุข้อตกลงของโครงการ (Collaborative)	5
การทำหน้าที่เป็นตัวแทน สร้างองค์การเสมือนจริง ด้วยการสร้างแม่แบบ ประกอบด้วยมุมมอง ที่สามารถแลกเปลี่ยนแนวคิดกันระหว่างกลุ่ม (Representative)	5
การมีสิทธิตัดสินใจเฉพาะหน้าในเรื่องการเปลี่ยนแปลงความต้องการ (Authorized)	4
การให้ความสำคัญกับสมาชิกในทีมแต่ละคนเท่าๆ กัน โดยการมอบหมายความดูแล หรือ ความรับผิดชอบสำหรับการพัฒนา (Committed)	4
สร้างความรู้ความเข้าใจในระบบงาน และช่วยสนับสนุนให้เกิดการแลกเปลี่ยนความรู้เกี่ยวกับ ระบบงานที่จะพัฒนาของสมาชิกในทีม (Knowledgeable)	5
ค่าเฉลี่ย	4.6

จากตาราง 110 แปลความหมายได้ว่าเกณฑ์การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการ  
พัฒนาซอฟต์แวร์ของกรณีศึกษา 7 Rho มีค่าเฉลี่ย 4.60 ระดับความพึงพอใจมากที่สุดด้านการ  
ทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการพัฒนา  
ซอฟต์แวร์ตามหลักการ อาไจล์ (Agile)

ตาราง 111 แสดงข้อมูลรายละเอียดของการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการ  
พัฒนาซอฟต์แวร์ของกรณีศึกษา 8 Tau

เกณฑ์คุณภาพของกระบวนการพัฒนาซอฟต์แวร์	
ความร่วมมือกัน และช่วยสนับสนุนความสามารถในการบรรลุข้อตกลงของโครงการ (Collaborative)	4
การทำหน้าที่เป็นตัวแทน สร้างองค์การเสมือนจริง ด้วยการสร้างแม่แบบ ประกอบด้วยมุมมอง ที่สามารถแลกเปลี่ยนแนวคิดกันระหว่างกลุ่ม (Representative)	4
การมีสิทธิตัดสินใจเฉพาะหน้าในเรื่องการเปลี่ยนแปลงความต้องการ (Authorized)	4
การให้ความสำคัญกับสมาชิกในทีมแต่ละคนเท่าๆกัน โดยการมอบหมายความดูแล หรือ ความรับผิดชอบสำหรับการพัฒนา (Committed)	5
สร้างความรู้ความเข้าใจในระบบงาน และช่วยสนับสนุนให้เกิดการแลกเปลี่ยนความรู้เกี่ยวกับ ระบบงานที่จะพัฒนาของสมาชิกในทีม (Knowledgeable)	5
ค่าเฉลี่ย	4.4

จากตาราง 111 แปลความหมายได้ว่าเกณฑ์การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการ  
พัฒนาซอฟต์แวร์ของกรณีศึกษา 8 Tau มีค่าเฉลี่ย 4.4 หมายถึง ระดับความพึงพอใจมากที่สุดด้าน  
การทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์ เมื่อได้ประยุกต์ใช้เทคนิคการ  
พัฒนาซอฟต์แวร์ตามหลักการ อัจฉริยะแบบไม่สมบูรณ (Ad-Hoc)



ภาคผนวก ข สรุปคุณลักษณะสิ่งแวดล้อมของโครงการพัฒนาซอฟต์แวร์ของกรณีศึกษา

ตาราง 112 แสดงคุณลักษณะสิ่งแวดล้อมของโครงการพัฒนาซอฟต์แวร์ของกรณีศึกษา

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
ประเภทของธุรกิจ/องค์กร/ หน่วยงาน	อุตสาหกรรม การผลิต (Factory)	เทคโนโลยี สารสนเทศ (Information technology)	อุตสาหกรรม การผลิต (Factory)	เทคโนโลยี สารสนเทศ (Information technology)	เทคโนโลยี สารสนเทศ (Information technology)	เทคโนโลยี สารสนเทศ (Information technology)	เทคโนโลยี สารสนเทศ (Information technology)	เทคโนโลยี สารสนเทศ (Information technology)
ระยะเวลาการประกอบการที่ หน่วยงานได้ดำเนินกิจการ ตั้งแต่เริ่มก่อตั้ง	≥ 25 ปีขึ้นไป	1-6 ปี	≥ 25 ปีขึ้นไป	7-12 ปี	13-18 ปี	13-18 ปี	13-18 ปี	19-24 ปี
จำนวนผู้เชี่ยวชาญ หรือ ผู้ปฏิบัติงานประจำด้านการ พัฒนาซอฟต์แวร์ภายใน องค์กร/ หน่วยงานที่คุ้น ปฏิบัติงาน	ผู้เชี่ยวชาญ ผู้ปฏิบัติงาน ด้านการพัฒนา ซอฟต์แวร์ 26- 50 คน	ผู้เชี่ยวชาญ ผู้ปฏิบัติงาน ด้านการพัฒนา ซอฟต์แวร์ 11- 25 คน	ผู้เชี่ยวชาญ ผู้ปฏิบัติงาน ด้านการพัฒนา ซอฟต์แวร์ 26- 50 คน	มากกว่า 100 คนขึ้นไป	ผู้เชี่ยวชาญ ผู้ปฏิบัติงาน ด้านการพัฒนา ซอฟต์แวร์ 11- 25 คน	ผู้เชี่ยวชาญ ผู้ปฏิบัติงาน ด้านการพัฒนา ซอฟต์แวร์ 11- 25 คน	ผู้เชี่ยวชาญ ผู้ปฏิบัติงาน ด้านการพัฒนา ซอฟต์แวร์ 11- 25 คน	ผู้เชี่ยวชาญ ผู้ปฏิบัติงาน ด้านการพัฒนา ซอฟต์แวร์ 51- 100 คน

ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau	
สิ่งแวดล้อมทางกายภาพ (Physical environment) ของพื้นที่พัฒนาที่ติดตั้งซอฟต์แวร์ที่พัฒนาขึ้น โดยมีซอฟต์แวร์ที่เกี่ยวข้อง เช่น ต้องมีการแลกเปลี่ยนข้อมูลร่วมกัน	การพัฒนา ระหว่างเมือง และ ระหว่าง บริษัท (Multi-city and Multi city and Multi Company)	การพัฒนา ภายในเมือง หรือบริเวณ เดียวกัน (Same-city or metro area)	การพัฒนา ระหว่างเมือง และ ระหว่าง บริษัท (Multi-city and Multi Company)	การพัฒนา ระหว่างเมือง และ ระหว่าง บริษัท (Multi-city and Multi Company)	การพัฒนา ความเกี่ยวข้อง ระหว่างเมือง และ ระหว่าง บริษัท (Multi-city and Multi Company)	การพัฒนา ความเกี่ยวข้อง ภายในเมือง หรือบริเวณ เดียวกัน (Same-city or metro area)	การพัฒนา ความเกี่ยวข้อง ภายในเมือง หรือบริเวณ เดียวกัน (Same-city or metro area)	การพัฒนา ความเกี่ยวข้อง ภายในเมือง หรือบริเวณ เดียวกัน (Same-city or metro area)	การพัฒนา ความเกี่ยวข้อง ระหว่าง ประเทศ (International)
ลักษณะของโครงสร้างทีมพัฒนาหรือลักษณะการทำงาน	พัฒนา ระบบงาน ภายในองค์กร (In-house Development)	ผู้รับจ้างใน การพัฒนา ซอฟต์แวร์ (Outsource Development)	พัฒนา ระบบงาน ภายในองค์กร (In-house Development)	ผู้รับจ้างใน การพัฒนา ซอฟต์แวร์ (Outsource Development)	ผู้รับจ้างใน การพัฒนา ซอฟต์แวร์ (Outsource Development)	ผู้รับจ้างใน การพัฒนา ซอฟต์แวร์ (Outsource Development)	ผู้ร่วมทุนในการ พัฒนา ซอฟต์แวร์ (Partnership Development)	ผู้รับจ้างใน การพัฒนา ซอฟต์แวร์ (Outsource Development)	

ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
รูปแบบวัฒนธรรมองค์กร (Business culture) ในการยอมรับเทคโนโลยี	มีความพอใจที่จะลงมือรับ และปรับใช้	มีความพอใจที่จะลงมือรับ และปรับใช้	มีความพอใจที่จะลงมือรับ และปรับใช้	มีความพอใจที่จะลงมือรับ และปรับใช้	มีความพอใจที่จะลงมือรับ และปรับใช้	มีความพอใจที่จะลงมือรับ และปรับใช้	มีความพอใจที่จะลงมือรับ และปรับใช้	มีความพอใจที่จะลงมือรับ และปรับใช้
(Technology) หรือวิธีการ (Method) ใหม่เกิดขึ้น	เทคโนโลยี หรือ วิธีการใหม่ ตาม ผู้นำด้าน เทคโนโลยี	เทคโนโลยี หรือ เทคโนโลยีใหม่ แนวคิดใหม่ เพียงบางส่วนที่ ไม่รัดกุมแบบ การพัฒนาที่เคย ทำมา	เทคโนโลยี หรือ เทคโนโลยีใหม่ ตาม ผู้นำด้าน เทคโนโลยี	เทคโนโลยี หรือ เทคโนโลยีใหม่ ตาม ผู้นำด้าน เทคโนโลยี	เทคโนโลยี หรือ เทคโนโลยีใหม่ ตาม ผู้นำด้าน เทคโนโลยี	เทคโนโลยี หรือ เทคโนโลยีใหม่ ตาม ผู้นำด้าน เทคโนโลยี	เทคโนโลยี หรือ เทคโนโลยีใหม่ ตาม ผู้นำด้าน เทคโนโลยี	เทคโนโลยี หรือ เทคโนโลยีใหม่ ตาม ผู้นำด้าน เทคโนโลยี
จำนวนผู้ปฏิบัติงาน ณ ฐานสุดท้าย (End User)	26-50 คน	26-50 คน	26-50 คน	51-100 คน	มากกว่า 100 คนขึ้นไป	11-25 คน	26-50 คน	26-50 คน

ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
มาตรฐานขององค์กร โดยส่งผลต่อการพัฒนาซอฟต์แวร์ ซึ่งช่วยให้การพัฒนาซอฟต์แวร์มีประสิทธิภาพขึ้น	ISO	Verification and Validation	ISO	CMMI	Software Quality Assurance	Software Quality Assurance	Software Quality Assurance, Verification	Software Quality Assurance, Verification
เงื่อนไขด้านงบประมาณที่ได้รับ	มีความกดดันเรื่อง	มีความกดดันเรื่อง	มีความกดดันเรื่อง	มีความกดดันเรื่อง	มีความกดดันเรื่อง	มีความกดดันเรื่อง	มีความกดดันเรื่อง	มีความกดดันเรื่อง
สำหรับโครงการพัฒนาซอฟต์แวร์	งบประมาณน้อยมาก	งบประมาณสูงมาก	งบประมาณน้อยมาก	งบประมาณปานกลาง	งบประมาณสูงมาก	งบประมาณสูงมาก	งบประมาณสูงมาก	งบประมาณปานกลาง
เงื่อนไขด้านความกดดันในเรื่องของระยะเวลา	มีความกดดันเรื่องระยะเวลา	มีความกดดันเรื่องระยะเวลา	มีความกดดันเรื่องระยะเวลา	มีความกดดันเรื่องระยะเวลา	มีความกดดันเรื่องระยะเวลา	มีความกดดันเรื่องระยะเวลา	มีความกดดันเรื่องระยะเวลา	มีความกดดันเรื่องระยะเวลา
	สูงมาก	สูงมาก	สูงมาก	สูง	สูงมาก	ปานกลาง	สูง	ปานกลาง

ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
เงื่อนไขด้านความต้องการให้	ความต้องการ	ความต้องการ	ความต้องการ	ความต้องการ	ความต้องการ	ความต้องการ	ความต้องการ	ความต้องการ
ซอฟต์แวร์มีความแน่นอน	ให้ซอฟต์แวร์มี	ให้ซอฟต์แวร์มี	ให้ซอฟต์แวร์มี	ให้ซอฟต์แวร์มี	ให้ซอฟต์แวร์มี	ให้ซอฟต์แวร์มี	ให้ซอฟต์แวร์มี	ให้ซอฟต์แวร์มี
	ความแน่นอนสูง	ความแน่นอนสูง	ความแน่นอนสูง	ความแน่นอนสูง	ความแน่นอนสูง	ความแน่นอนสูง	ความแน่นอนสูง	ความแน่นอนสูง
	เกี่ยวข้องกับ	เกี่ยวข้องกับ	เกี่ยวข้องกับ	เกี่ยวข้องกับ	เกี่ยวข้องกับ	เกี่ยวข้องกับ	เกี่ยวข้องกับ	เกี่ยวข้องกับ
	ข้อมูลด้าน	ข้อมูลด้าน	ข้อมูลด้าน	ข้อมูลด้าน	ข้อมูลด้าน	ข้อมูลด้าน	ข้อมูลด้าน	ข้อมูลด้าน
	การเงิน	การเงิน	การเงิน	การเงิน	การเงิน	การเงิน	การเงิน	การเงิน
การพัฒนาทรัพยากรมนุษย์,	มากที่สุด	ปานกลาง	มากที่สุด	มาก	มากที่สุด	มากที่สุด	มากที่สุด	มากที่สุด
การทำงานเป็นทีม, การ								
มอบหมายควมไว้วางใจแก่								
พนักงาน								
ความเป็นเอกลักษณ์, ทัศนคติ	มากที่สุด	ปานกลาง	มากที่สุด	มาก	มาก	มาก	มากที่สุด	มากที่สุด
ของผลิตภัณฑ์ โดยผลิตภัณฑ์ให้								
เป็นผู้นำด้านนวัตกรรม และ								
ผลิตภัณฑ์								



ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
การแลกเปลี่ยนความรู้, นวัตกรรม, เทคโนโลยีใหม่	มากที่สุด	น้อย	มากที่สุด	ปานกลาง	ปานกลาง	ปานกลาง	มากที่สุด	มาก
ผลักดันให้เกิดความกระตือรือร้น	มากที่สุด	มาก	มากที่สุด	มาก	มากที่สุด	มากที่สุด	มากที่สุด	มาก
การทำงานที่มีประสิทธิภาพ และ บรรลุผล	มากที่สุด	มากที่สุด	มากที่สุด	มากที่สุด	มากที่สุด	มากที่สุด	มากที่สุด	มากที่สุด
รูปแบบการจัดทำเอกสารสัญญา (Contract) หรือเอกสารนำเสนอ	ข้อตกลงให้	ข้อตกลงให้	ข้อตกลงให้	ข้อตกลงให้	ข้อตกลงให้	ข้อตกลงให้	ข้อตกลงให้	ข้อตกลงให้
โครงการ (Proposal) สำหรับ	ความสำคัญกับ	ความสำคัญกับ	ความสำคัญกับ	ความสำคัญกับ	ความสำคัญกับ	ความสำคัญกับ	ความสำคัญกับ	ความสำคัญกับ
ดำเนินการพัฒนาซอฟต์แวร์	กำหนดเวลา	เรื่องกำหนด	กำหนดเวลา	กำหนดมูลค่า	กำหนด	กำหนด	กำหนดเวลา	กำหนดเวลา
	และความ	มูลค่าซอฟต์แวร์	และความ	ซอฟต์แวร์แบบ	เป้าหมายด้าน	เป้าหมายด้าน	และความ	กำหนดเวลา
	เหมาะสมของ	แบบตายตัว	เหมาะสมของ	ตายตัว (Fixed	มูลค่าซอฟต์แวร์	มูลค่าซอฟต์แวร์	เหมาะสมของ	และความ
	ซอฟต์แวร์ (Time	(Fixed Price	ซอฟต์แวร์ (Time	Price	(Target Cost	(Target Cost	ซอฟต์แวร์	เหมาะสมของ
	and Material	Contract)	and Material	Contract)	Contract)	Contract)	(Time and	ซอฟต์แวร์
	Contract)		Contract)				Material	(Time and
							Contract)	Material
								Contract)
								Contract)

ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
ระดับความคล้ายคลึงกับโครงการเดิม หรือมีแบบอย่างมาก่อน เมื่อเทียบกับโครงการปัจจุบัน	ส่วนใหญ่ไม่มี หรือแบบอย่างก่อนหน้าโดยสิ้นเชิง (Largely Unprecedented)	ไม่มีแบบอย่าง หรือตัวอย่างก่อนหน้าโดยสิ้นเชิง (Thoroughly Unprecedented)	ส่วนใหญ่ไม่มี หรือตัวอย่างก่อนหน้าโดยสิ้นเชิง (Largely Unprecedented)	ไม่มีแบบอย่าง หรือตัวอย่างก่อนหน้าโดยสิ้นเชิง (Thoroughly Unprecedented)	ไม่มีแบบอย่าง หรือตัวอย่างก่อนหน้าโดยสิ้นเชิง (Thoroughly Unprecedented)	ส่วนใหญ่ไม่มี หรือแบบอย่างก่อนหน้าโดยสิ้นเชิง (Largely Unprecedented)	บางส่วนไม่มี หรือแบบอย่างก่อนหน้าโดยสิ้นเชิง (Somewhat Unprecedented)	บางส่วนไม่มี หรือแบบอย่างก่อนหน้าโดยสิ้นเชิง (Somewhat Unprecedented)
ตัวบ่งชี้ความสัมพันธ์โครงการพัฒนาซอฟต์แวร์	โครงการเผชิญกับปัญหาซึ่งทีมไม่มีความคุ้นเคย และไม่มีการใช้รูปแบบในการ	โครงการเกี่ยวข้องกับการบูรณาการ (Integration) มากกว่า 2 โปรแกรม (Business Domain) ที่ทีมไม่คุ้นเคย	มีการเลือกใช้เทคโนโลยีใหม่ๆ มากกว่า 2 โปรแกรมที่ทีมพัฒนาไม่เคยใช้	โครงการเผชิญกับปัญหาซึ่งทีมไม่มีความคุ้นเคย และไม่มีการใช้รูปแบบในการ	โครงการเผชิญกับปัญหาซึ่งทีมไม่มีความคุ้นเคย และไม่มีการใช้รูปแบบในการ	ใช้ปัจจัยอื่นๆ ที่เกี่ยวข้องซึ่งซับซ้อนในการโครงการพัฒนาซอฟต์แวร์	โครงการเผชิญกับปัญหาซึ่งทีมไม่มีความคุ้นเคย และไม่มีการใช้รูปแบบในการ	โครงการเผชิญกับปัญหาซึ่งทีมไม่มีความคุ้นเคย และไม่มีการใช้รูปแบบในการ



ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
เป้าหมายของการพัฒนาซอฟต์แวร์	พัฒนาเพื่อแทนที่ระบบเดิมที่มีอยู่ (Replacement of existing system)	พัฒนาใหม่ (new system)	พัฒนาเพื่อแทนที่ระบบเดิมที่มีอยู่ (Replacement of existing system)	พัฒนาใหม่ (new system)	พัฒนาใหม่ (new system)	พัฒนาใหม่ (new system)	พัฒนาปรับปรุงระบบเดิมที่มีอยู่ (enhancement to existing system)	พัฒนาใหม่ (new system)
รูปแบบการใช้งานโปรแกรมที่เชื่อมต่อกับผู้ใช้	Application Software	Application Software	Application Software	Web base Technology เครื่อง client และ PDA	Web base Technology	Web base Technology	Web base Technology	Web base Technology
Team size	7-12 คน	13-18 คน	13-18 คน	13-18 คน	7-12 คน	7-12 คน	13-18 คน	13-18 คน

ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
จำนวนสมาชิกในทีมของ โครงการพัฒนาซอฟต์แวร์	7-12 คน	13-18 คน	7-12 คน	13-18 คน	7-12 คน	7-12 คน	7-12 คน	13-18 คน
ประสบการณ์การทำงานโดย เฉลี่ยของทีมที่เกี่ยวข้องกับ ขอบเขตทางธุรกิจ (Business Domain) ในรูปแบบเดียวกับ โครงการพัฒนาซอฟต์แวร์	1-2 ปี	≥ 2 เดือน	1-2 ปี	6 เดือน	> 5 ปีขึ้นไป	> 5 ปีขึ้นไป	3-4 ปี	6 เดือน
ประสบการณ์การทำงานโดย เฉลี่ยของทีมที่เกี่ยวข้องกับการ พัฒนาซอฟต์แวร์ (Software development)	7-9 ปี	4-6 ปี	7-9 ปี	7-9 ปี	> 12 ปีขึ้นไป	> 12 ปีขึ้นไป	7-9 ปี	4-6 ปี
ประสบการณ์โดยเฉลี่ยของทีม หรือความคุ้นเคยกับภาษา	6 เดือน	1-2 ปี	6 เดือน	1-2 ปี	> 5 ปีขึ้นไป	> 5 ปีขึ้นไป	6 เดือน	1-2 ปี

ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
ประสบการณ์โดยเฉลี่ยของทีม ในการทำงานกับ Platform	1-2 ปี	3-4 ปี	1-2 ปี	3-4 ปี	> 5 ปีขึ้นไป	> 5 ปีขึ้นไป	≥ 2 เดือน	1-2 ปี
ประสบการณ์โดยเฉลี่ยของทีม ในการทำงานกับระเบียบวิธีการ พัฒนาซอฟต์แวร์	≥ 2 เดือน	> 5 ปีขึ้นไป	6 เดือน	1-2 ปี	> 5 ปีขึ้นไป	> 5 ปีขึ้นไป	6 เดือน	1-2 ปี
Development style	Iterative and Incremental Development, Time Boxing	Distributed Development Development, Time Boxing	Iterative and Incremental Development, Time Boxing	Iterative and Incremental Development	Iterative and Incremental Development	Distributed Development	Iterative and Incremental Development	Iterative and Incremental Development, Distributed Development

ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
Code style	สร้างมาตรฐาน การเขียนโค้ด (Code) และ สร้างควมมี ส่วนร่วมในการ เขียนโปรแกรม (Collective ownership of code)	-	สร้างมาตรฐาน การเขียนโค้ด (Code) และ สร้างควมมี ส่วนร่วมในการ เขียนโปรแกรม (Collective ownership of code)	-	Theta	Iota	สร้างมาตรฐานการ เขียนโค้ด (Code) และ สร้างควมมี ส่วนร่วมในการ เขียนโปรแกรม (Collective ownership of code)	สร้างมาตรฐาน การเขียนโค้ด (Code) และ แบ่งระดับการ เข้าใช้เพื่อถ่าย ต่อการบริหาร จัดการ (Individual class ownership)

ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
รูปแบบการติดต่อสื่อสาร	การติดต่อ	การติดต่อ	การติดต่อ	การติดต่อ	การติดต่อ	การติดต่อ	การติดต่อ	มีความสมดุล
ภายในทีมพัฒนาซอฟต์แวร์	สื่อสารมีรูปแบบ	สื่อสารทั้งหมด	สื่อสารมีรูปแบบ	สื่อสารเอกสาร	สื่อสารเป็นแบบ	สื่อสารทั้งหมด	สื่อสารเป็นแบบ	ระหว่างการ
	ไม่เป็นการ	โดยเอกสาร	ไม่เป็นการ	การประชุม	ไม่เป็นการ	โดยเอกสาร	ไม่เป็นการ	ติดต่อสื่อสาร
	เป็นส่วนใหญ่	ประชุม หรือ	เป็นส่วนใหญ่	เอกสาร	แบบ face-to-	ประชุม หรือ	แบบ face-to-	แบบตาม
	โดยบางครั้งมี	เอกสาร	โดยบางครั้งมี	อิเล็กทรอนิกส์มี	face เน้นการ	เอกสาร	face เน้นการ	กฎเกณฑ์ และ
	รูปแบบตาม	อิเล็กทรอนิกส์ที่	รูปแบบตาม	รูปแบบตาม	ทำงานร่วมกัน	อิเล็กทรอนิกส์ที่	ทำงานร่วมกัน	ไม่เป็นการ
	กฎเกณฑ์	มีรูปแบบตาม	กฎเกณฑ์	กฎเกณฑ์ส่วน	เปิดเผยและ	มีรูปแบบตาม	เปิดเผยและ	
		กฎเกณฑ์	ใหญ่ บางครั้งมี	ใหญ่ บางครั้งมี	ตรงไปตรงมา	กฎเกณฑ์	ตรงไปตรงมา	
			รูปแบบไม่เป็น	รูปแบบไม่เป็น				
			ทางการ	ทางการ				

ตาราง 112 (ต่อ)

Criteria	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
รูปแบบการติดต่อสื่อสาร	การติดต่อ	การติดต่อ	การติดต่อ	การติดต่อ	การติดต่อ	การติดต่อ	การติดต่อ	มีความสมดุล
ระหว่างทีม และผู้ที่เกี่ยวข้อง	สื่อสารรูปแบบ	สื่อสารโดย	สื่อสารรูปแบบ	สื่อสารโดย	สื่อสารทั้งหมด	สื่อสารเป็นแบบ	สื่อสารเป็นแบบ	ระหว่างการ
กับโครงการพัฒนาซอฟต์แวร์	ไม่เป็นทางการ	เอกสาร การ	ไม่เป็นทางการ	เอกสาร การ	โดยเอกสาร การ	ไม่เป็นทางการ	ไม่เป็นทางการ	ติดต่อสื่อสาร
	เป็นส่วนใหญ่	ประชุม หรือ	เป็นส่วนใหญ่	ประชุม หรือ	ประชุม หรือ	แบบ face-to-	แบบ face-to-	แบบตาม
	โดยบางครั้งมี	เอกสาร	โดยบางครั้งมี	เอกสาร	เอกสาร	face	face	กฎเกณฑ์ และ
	รูปแบบตาม	อิเล็กทรอนิกส์มี	รูปแบบตาม	อิเล็กทรอนิกส์	อิเล็กทรอนิกส์ที่	ทำงานร่วมกัน	ทำงานร่วมกัน	ไม่เป็นทางการ
	กฎเกณฑ์	รูปแบบตาม	กฎเกณฑ์	รูปแบบตาม	มีรูปแบบตาม	เปิดเผยและ	เปิดเผยและ	
		กฎเกณฑ์ส่วน	กฎเกณฑ์ส่วน	กฎเกณฑ์ส่วน	กฎเกณฑ์	ตรงไปตรงมา	ตรงไปตรงมา	
		ใหญ่ บางครั้งมี	ใหญ่ บางครั้งมี	ใหญ่ บางครั้งมี				
		รูปแบบไม่เป็น	รูปแบบไม่เป็น	รูปแบบไม่เป็น				
		ทางการ	ทางการ	ทางการ				
Abstraction mechanism	Object-oriented	Object-oriented	Object-oriented	Object-oriented/Component-oriented	Object-oriented/Component-oriented	Object-oriented/Component-oriented	Object-oriented/Component-oriented	Object-oriented

ภาคผนวก ข ตารางสรุปแนวปฏิบัติหลักที่กรณีศึกษาเลือกใช้

ตาราง 113 แสดงรายละเอียดแนวปฏิบัติหลักของกรณีศึกษา

Practices/Techniques	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
Development								
1 Short releases	1	0	1	0	1	0	1	0
Process								
2 Metaphor	1	0	1	0	1	0	1	0
3 Simple Design	1	0	1	0	1	1	1	0
4 Refactoring	1	0	1	0	1	0	1	0
5 Pair programming	0	0	1	0	1	0	0	0
6 Collective ownership	1	0	1	1	1	0	1	0
7 Continuous integration	1	0	1	0	0	0	1	0
8 On-site customer	1	0	1	1	1	0	1	0
9 40-h week	1	0	1	0	1	0	1	0
10 Test first Development	1	0	1	0	0	0	1	0
11 Coding standards	1	0	1	0	0	0	1	0
12 Scrum teams	1	0	1	0	0	0	0	0
13 Product backlog	1	0	1	0	1	0	0	0

ตาราง 113 (ต่อ)

	Practices/Techniques	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
Development	14 Sprint	1	0	1	0	1	0	0	0
Process	15 Sprint review	1	0	1	0	1	0	0	0
	16 Domain object modeling	0	0	0	0	0	0	0	0
	17 Developing by Feature	0	0	0	0	0	0	0	0
	18 Individual class ownership	0	0	0	0	0	0	0	1
	19 Feature teams	1	0	1	0	0	0	1	1
	20 Regular Builds	0	0	0	0	0	0	0	0
	21 The project mission development	0	0	0	0	0	0	0	0
	22 Developing by components	0	1	0	1	0	1	0	1
	23 Collaborative teams	0	0	0	0	0	0	0	0
	24 Joint application development by independent agents	0	0	0	0	0	0	0	0
	25 Customer focus group reviews/ User Viewings	0	0	0	0	0	0	0	0
	26 Software inspection	0	0	0	1	0	0	0	1



ตาราง 113 (ต่อ)

	Practices/Techniques	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
Development	27 Active user involvement	0	0	0	0	0	0	0	0
Process	28 Empowered teams.	0	0	0	0	0	0	0	0
	29 Iterative and incremental development	0	0	0	1	0	0	0	1
	30 Reversible changes	0	0	0	0	0	0	0	0
	31 Integrated testing	0	0	0	0	0	0	0	0
	32 Requirements are baselined at a high level	0	0	0	0	0	1	0	0
	33 Collaboration and cooperation among stakeholders	0	0	0	0	0	0	0	0
	34 Staging	0	0	0	0	0	0	0	0
	35 Holistic Diversity and Strategy	0	0	0	0	0	0	0	0
	36 Parallelism and Flux	0	0	0	0	0	0	0	0
	37 Revision and review	0	0	0	0	0	0	0	0
Project management	1 The planning game	0	0	0	0	1	0	1	0
	2 Scrum master	1	0	1	0	0	0	0	0

ตาราง 113 (ต่อ)

Practices/Techniques	Alpha	Beta	Delta	Zeta	Theta	Iota	Rho	Tau
Project management process	1	0	1	0	1	0	0	0
3 Sprint Planning meeting								
4 Daily scrum meeting	1	0	1	0	0	0	0	0
5 Reporting/ visibility of results	0	0	0	0	0	0	0	0
6 Adaptive cycle planning	0	0	0	0	0	0	0	0
7 Adaptive management model	0	0	0	0	0	0	0	0
8 Monitoring of a progress	0	0	0	0	0	0	0	0
9 Prototype planning	0	0	0	0	0	0	0	0
support process	0	0	0	0	0	0	0	1
1 Configuration management								
2 Risk management	0	0	0	0	0	0	0	0
Process management process	0	0	0	0	0	0	0	0
1 Project postmortem								
2 Reflection workshops Methodology tuning	0	0	0	0	0	0	0	0
<b>Total</b>	<b>18</b>	<b>1</b>	<b>19</b>	<b>5</b>	<b>13</b>	<b>3</b>	<b>12</b>	<b>6</b>
<b>% of Agile</b>	<b>36%</b>	<b>2%</b>	<b>38%</b>	<b>10%</b>	<b>26%</b>	<b>6%</b>	<b>24%</b>	<b>12%</b>



## อภิธานศัพท์

- การพัฒนาซอฟต์แวร์แบบ Agile : การพัฒนาซอฟต์แวร์แบบใดที่มีแนวทางปฏิบัติหลักตามที่ได้ประกาศไว้ในแถลงการณ์การของ อาไจล์ (Agile) ครบทุกข้อจะถือว่าหลักการดังกล่าวเป็นการพัฒนาซอฟต์แวร์แบบ Agile
- การพัฒนาซอฟต์แวร์แบบ Non-Agile : การพัฒนาซอฟต์แวร์แบบใดที่ไม่มีแนวทางปฏิบัติหลักตามที่ได้ประกาศไว้ในแถลงการณ์การของ อาไจล์ (Agile) ครบทุกข้อจะถือว่าหลักการดังกล่าวเป็นการพัฒนาซอฟต์แวร์แบบไม่ใช่หลักการตามแบบอาไจล์ (Non-Agile)
- ประสิทธิภาพการพัฒนาซอฟต์แวร์ : ประสิทธิภาพขั้นสุดท้ายของโครงการพัฒนาซอฟต์แวร์ตามหลักการบริหารโครงการพัฒนาซอฟต์แวร์แบ่งเป็น 4 มิติ ดังนี้ 1) มิติด้านระยะเวลาการพัฒนา (cycle time) 2) มิติด้านประสิทธิผลในด้านคุณภาพของกระบวนการพัฒนาซอฟต์แวร์ (Quality of process) 3) มิติด้านต้นทุน (Cost Effective) และ 4) มิติด้านความพึงพอใจของลูกค้า และประสิทธิภาพการทำงานร่วมกันของผู้ที่เกี่ยวข้องกับโครงการพัฒนา (Inter-supplier performance and Customer satisfactions)
- บริบทของโครงการพัฒนาซอฟต์แวร์ : สภาพแวดล้อม และเงื่อนไขต่างๆ ที่เกี่ยวข้องกับโครงการพัฒนาซอฟต์แวร์
- ปัจจัยนำเข้า : การระบุรายละเอียดในแง่ของทรัพยากรที่จำเป็นต้องใช้ในการดำเนินกิจกรรมการพัฒนาซอฟต์แวร์ รวมทั้งขอบเขตของโครงการพัฒนาซอฟต์แวร์
- ผลลัพธ์ของโครงการพัฒนาซอฟต์แวร์ (Outcomes) : ผลประโยชน์ที่ได้จากผลผลิตและผลกระทบที่มีต่อโครงการพัฒนาซอฟต์แวร์ และสิ่งแวดล้อมของโครงการพัฒนาซอฟต์แวร์
- กิจกรรมของโครงการ : การระบุรายละเอียดกิจกรรมต่างๆ ที่เกิดขึ้นในโครงการพัฒนาซอฟต์แวร์