

บทที่ 2

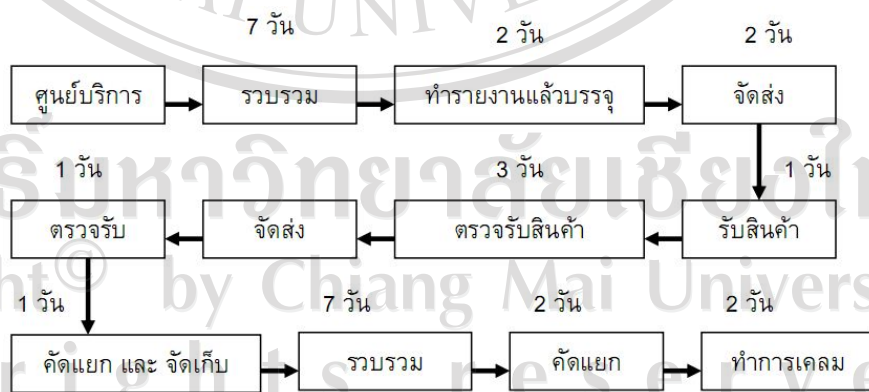
สรุปสาระสำคัญจากเอกสารที่เกี่ยวข้อง

ในการพัฒนาระบบตรวจสอบสินค้าส่งซ่อม ของร้านจำหน่ายอุปกรณ์คอมพิวเตอร์ได้ ทำการศึกษาทฤษฎีและหลักการต่างๆที่สามารถนำมาประยุกต์เข้ากับงานได้ โดยแบ่งออกเป็นหัวข้อต่างๆ ได้ดังต่อไปนี้

- 2.1 งานที่เกี่ยวข้องกับการพัฒนาระบบ
- 2.2 ทฤษฎีการบรรจุซื้อได้เปรียบทางการแข่งขัน
- 2.3 การวางแผนและจัดตารางงานโครงการ (Project Planning and Scheduling)
- 2.4 กระบวนการพัฒนาซอฟต์แวร์แบบจำลองน้ำตก (Waterfall Model)
- 2.5 มาตรฐานไอเอสโอ 12207(ISO12207)

2.1 งานที่เกี่ยวข้องกับการพัฒนาระบบ

ระบบเคลมสินค้าที่มีผู้พัฒนาหน้า ที่มีเนื้อหาและข้อความเกี่ยวข้องกับการพัฒนาระบบในครั้งนี้ ประกอบไปด้วย ชุดคอมพิวเตอร์ ที่มีการประกอบเครื่อง, รหัสสินค้า สินค้าขายปลีก ที่มีรหัสสินค้า มีการรับประกัน เช่น เครื่องเล่น MP3, การ์ดหน่วยความจำ, รายละเอียดสินค้าขายปลีก ที่ไม่มีการรับประกัน เช่น แผ่น CD/DVD, หมึกพิมพ์, วัสดุสิ้นเปลือง และมีการทำงาน



รูปที่ 2.1 แสดงถึงการตรวจสอบสินค้าส่งซ่อมเดิม

2.2 ทฤษฎีการบรรลุข้อได้เปรียบทางการแข่งขัน

เกียรติกศักดิ์ (2549) การสร้างข้อได้เปรียบเทียบทางการแข่งขันโดยการทำให้เกิดลักษณะแตกต่างที่เหนือกว่าคู่แข่งและความแตกต่างนั้นสามารถตอบสนองความพึงพอใจของลูกค้า ซึ่งมีกลยุทธ์การสร้างข้อได้เปรียบทางการแข่งขันอันได้แก่ การสร้างความแตกต่าง การเป็นผู้นำด้านการลงทุน และการตอบสนองที่รวดเร็ว โดยในการสร้างคุณค่าให้ลูกค้าในด้านคุณภาพและวิธีการที่สามารถทำต่อไปได้

การจัดการคุณภาพโดยรวม (Total Quality Management)

คุณภาพ (Quality) เป็นลักษณะทั้งหมดของผลิตภัณฑ์หรือบริการ ซึ่งมีความสามารถทำให้เกิดความพอใจหรือตอบสนองความต้องการการปรับปรุงคุณภาพจะช่วยให้สามารถเพิ่มส่วนแบ่งทางการตลาดและลดต้นทุนขององค์กร ซึ่งส่งผลโดยตรงให้กำไรขององค์กรเพิ่มขึ้น โดยทั่วไปแล้วหากองค์กรสามารถตอบสนองผู้บริโภคได้อย่างรวดเร็ว พร้อมทั้งทำการผลิตสินค้าในปริมาณที่มาก จะทำให้ต้นทุนการผลิตต่อหน่วยลดลงช่วยให้องค์กรมีความสามารถในการดำเนินการ พัฒนาความสามารถในการแข่งขัน และสร้างโอกาสในการเจริญเติบโต โดยเฉพาะสถานการณ์ในปัจจุบัน

2.3 การวางแผนและจัดตารางงานโครงการ (Project Planning and Scheduling)

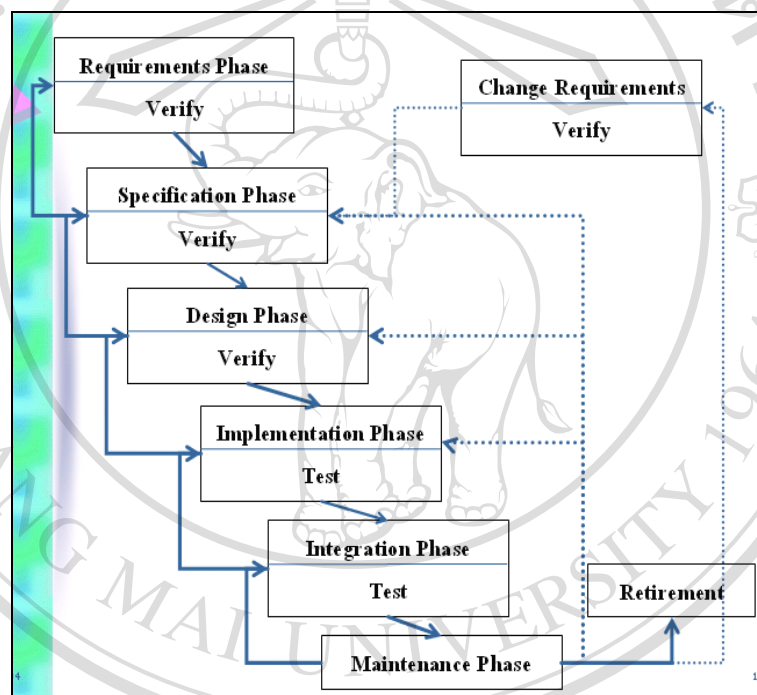
การวางแผนโครงการ เป็นการกำหนดกิจกรรมหลัก กิจกรรมย่อย เป้าหมายของแต่ละกิจกรรม (Milestone) การส่งมอบงาน และจัดตารางงาน โดยการกำหนดเวลาเริ่มต้นและส่งมอบงาน แผนงานดังกล่าวจะใช้เป็นแนวทางการดำเนินงาน โครงการให้ได้ตามเป้าหมายของโครงการ

เมสสินี นาคมณี (2545) การวางแผนโครงการถือเป็นส่วนสำคัญในการบริหารจัดการโครงการเมื่อมีการริเริ่มและคัดเลือกโครงการที่ต้องการพัฒนาเสร็จสิ้นแล้ว ขั้นตอนต่อไปคือ การวางแผนโครงการ เพื่อให้สามารถวิเคราะห์งานที่จะเกิดขึ้น และสามารถดำเนินการพัฒนา ติดตาม และสามารถทำให้โครงการดำเนินไปตามที่คาดหวังไว้ได้

เมสสินี นาคมณี (2547) ในการวางแผนงานนั้น เมื่อได้เรียนรู้ในส่วนของกระบวนการ และประมาณการซอฟต์แวร์แล้ว สิ่งที่ต้องทำการเรียนรู้ต่อไป คือ การจะนำกรรมวิธีและขั้นตอนที่ได้กล่าวถึงนั้นมาผนวกรวมกับค่าที่ได้ประมาณการ และนำมาใช้ในการวางแผนงาน โดยแผนงานที่นำมาใช้ในการพัฒนาซอฟต์แวร์นั้น โดยหลักแล้วจะให้ความสนใจในแผนงานที่เป็นแผนโครงการ แต่ทั้งนี้ยังมีแผนงานอื่นที่เป็นแผนงานย่อยๆ เข้ามาเกี่ยวข้องอีก

2.4 กระบวนการผลิตซอฟต์แวร์แบบจำลองน้ำตก (Waterfall model)

ดวงแก้ว สวามิภักดิ์ (2548) Waterfall model กล่าวว่า เป็นโมเดลแรก เสนอเมื่อปี 1970 โดย W.W. Royce โดยพัฒนามาจากกระบวนการทางวิศวกรรมอื่นที่มีขั้นตอนที่ชัดเจน เช่น วิศวกรรมโยธา Model จะแบ่งการพัฒนาเป็นขั้นตอนที่ชัดเจนและต่อเนื่องกัน (Phases) ขึ้นงานจากขั้นตอนหนึ่งใช้เป็นอินพุตในขั้นตอนต่อไป และต้องมีการตรวจรับเมื่อเสร็จขั้นตอน (Milestones) ก่อนเริ่มขั้นตอนใหม่ ลักษณะการส่งผ่านงานจากขั้นหนึ่งไปขั้นต่อไปมีลักษณะคล้ายการไหลของน้ำตก (Waterfall)



รูปที่ 2.2 แสดงกระบวนการพัฒนาซอฟต์แวร์แบบจำลองน้ำตก

จากรูปที่ 2.2 แสดง Waterfall Model สิ่งที่ได้จาก Waterfall Model ซึ่งใช้เป็นพื้นฐานสำหรับ Process Model ต่อมาคือ ขั้นตอนหลักของการพัฒนาซอฟต์แวร์ที่มี 8 ขั้นตอน ได้แก่

- 1) **Requirement** คือค้นหา กำหนดความต้องการจากผู้ใช้ และเช็คความถูกต้องโดยผู้จ้างและกลุ่มผู้ควบคุมคุณภาพ (SQA : Software Quality Assurance)
- 2) **Specification** คือ การระบุความต้องการเป็นเอกสารข้อกำหนด (Specification) ที่ระบุสิ่งที่ซอฟต์แวร์ต้องทำได้ คุณสมบัติที่ต้องมี รวมถึงเงื่อนไขของการส่งมอบเมื่อเสร็จงานว่า จะต้องส่งมอบชิ้นงานใดบ้าง เช่นคู่มือการใช้ คู่มือเชิงเทคนิค โค้ด

- หลังจากนั้นต้องให้ผู้จ้างและ กลุ่ม SQA ตรวจสอบเอกสารข้อกำหนด หากเห็นชอบก็จะเซ็นอนุมัติ
- ก่อนจะทำการออกแบบและสร้าง จะต้องวางแผนการจัดการและการดำเนินโครงการ (Software Project Management Plan) ซึ่งจะกำหนดเวลา คนที่ใช้ในแต่ละขั้นของการผลิต
- กลุ่มผู้ควบคุมคุณภาพตรวจแผนการดำเนินโครงการ ก่อนตกลงกับผู้จ้างเรื่องเวลาและงบ

3) **Design** คือเริ่มออกแบบซอฟต์แวร์ พิจารณาว่า Design คือออกแบบว่าต้องสร้างระบบอย่างไรให้ทำงาน มีคุณสมบัติตามข้อกำหนดที่ตกลง

- Design คือ การออกแบบว่าต้องเขียน โปรแกรมอย่างไรให้ได้ซอฟต์แวร์ที่ได้ดี
- ระหว่างออกแบบ อาจพบว่าข้อกำหนดจำเป็นต้องเปลี่ยนแปลงด้วยสาเหตุใดสาเหตุหนึ่ง เช่น ความไม่สมบูรณ์เกิดจากหลงลืมข้อกำหนดบางประการ (incomplete) หรือ ความไม่ชัดเจนคือข้อกำหนดตีความได้หลายอย่าง (ambiguous) หรือความขัดแย้งคือข้อกำหนดมีความหมายที่ขัดแย้งไม่สัมพันธ์กันเช่นบางครั้งก็ใช้เงื่อนไข บางครั้งก็ไม่ใช้
- หากเกิดกรณีเหล่านี้ ทีมต้องย้อนกลับไปทำงานที่ขั้นตอน Specification ซึ่งแสดงในรูปภาพ Waterfall Model ด้วยลูกศรจากขั้น Design ย้อนไปที่ขั้นตอน Specification และทุกครั้งที่เปลี่ยนแปลงงานที่เสร็จไปแล้ว ต้องมีการอนุมัติจากผู้จ้างและตรวจโดยทีม SQA อีกครั้งเสมอ ลักษณะนี้เป็นคุณลักษณะเฉพาะของ Waterfall Model

4) **Implementation** คือการส่งมอบงานออกแบบให้โปรแกรมเมอร์สร้างโมดูลและประกอบ(Implementation & Integration) งานทั้งสองทำงานกันไปพร้อมกันได้ และวนทำซ้ำหลายรอบจนกว่าจะได้ซอฟต์แวร์ที่ตรงตาม Specification

5) **Integration** คือการรวมโมดูลที่สร้างเสร็จและผ่านการทดสอบ (Unit Test) เข้าเป็นระบบย่อย(Subsystem) หรือ ระบบรวม (System) และทดสอบรวมอีกครั้ง(Integration Test) จริงๆแล้วจะมีการออกแบบการทำ Integration ไว้ซึ่งจะกำหนดว่าต้องสร้างโมดูลใดก่อนหลังเพื่อที่ว่าเมื่อสร้างโมดูลเสร็จทีมจะทำการ Integrate กันทีละขั้นที่วางแผนไว้เรียกว่า Integration Plan

6) **Test** การทดสอบจะมีหลายประเภทตั้งแต่ Unit test หรือ Integration test หรือ Acceptance test สิ่งที่สำคัญคือ Test ทำขึ้นเพื่อระบุข้อผิดพลาดที่มีในซอฟต์แวร์ และ Test ที่ดีต้องชี้ให้เห็นว่าข้อผิดพลาดอยู่ที่ใด ดังนั้นจึงต้องมีการออกแบบ Test (Test Design) สำหรับ Waterfall

Process นั้นหากทดสอบแล้วพบว่า ข้อผิดพลาดเกิดที่ระดับ Specification Design หรือ Implementation ก็จะต้อง

7) **Maintenance** เมื่อผ่านการทำ Test จากทีมจนพอใจแล้วไม่พบข้อผิดพลาดใดๆ ระบบทำงานได้ตาม Specification แล้วทีมจะทำการส่งมอบให้ลูกค้าโดยทีมจะนำระบบไปติดตั้ง เพื่อให้ลูกค้าทำการทดสอบก่อนรับมอบ(Acceptance Testing) หากการทดสอบผ่าน ทีมจะส่งมอบงานตามข้อกำหนด การแก้ไข ปรับปรุง หรือเพิ่มเติมใดใดกับระบบนับตั้งแต่จุดนี้ถือว่าเป็น เฟส Maintenance

8) **Verify** ถือเป็นจุดสำคัญที่สุดของ Waterfall Model ดังที่ได้อธิบายคือทุกขั้นตอนใน Waterfall Model ต้องมีการตรวจโดยทีม SQA เสมอ การผ่านการตรวจสอบในแต่ละขั้นถือเป็นการยืนยันความก้าวหน้าใน Process (Milestone)

ข้อเด่นของ Waterfall Model

- เป็นโมเดลที่มีระเบียบแบบแผน ชัดเจน มีการกำหนด Milestone ชัดเจนการเปลี่ยนแปลง การพัฒนาเอกสาร (Document-driven) และการตรวจสอบ (Verification) เป็นสำคัญ ทำให้การ Maintenance สามารถทำได้ง่าย ลดค่าใช้จ่ายการบำรุงรักษาเนื่องจากมีเอกสารประกอบ เราทราบดีว่างบประมาณการ Maintenance ประมาณ 67% ของงบประมาณตลอด Software lifecycle

- เป็นโมเดลที่นำไปใช้พัฒนาซอฟต์แวร์หลากหลายและประสบความสำเร็จมาแล้ว

ข้อด้อยของ Waterfall Model

- ลูกค้าไม่แน่ใจว่าจะได้ระบบที่ตรงตามที่ต้องการจริงหรือไม่ เนื่องจากระบวนการยี่ดเอกสารเป็นหลัก (Document-driven) สิ่งที่ลูกค้าเข้าใจจากเอกสารข้อกำหนดอาจไม่ใช่สิ่งที่ลูกค้าอยากได้ ความเข้าใจว่าซอฟต์แวร์จะมีคุณสมบัติอย่างไรเมื่อเสร็จโดยอ่านจาก Specification ที่เป็นข้อความ หรือแม้แต่ใช้แผนภาพก็ไม่เท่ากับ Product จริงโมเดลกระบวนการพัฒนาแบบการสร้างต้นแบบ (Rapid Prototyping Model) จะแก้ไขจุดด้อยนี้

- การแก้ไขเปลี่ยนแปลงใด ๆ แม้ว่าจะเล็กน้อยแค่ไหนจะสมบูรณ์ก็ต่อเมื่อเอกสารสมบูรณ์ เนื่องจากเป็นกระบวนการที่ยี่ดเอกสารเป็นหลัก (Document-driven) ทีมพัฒนาจะใช้เวลากับการทำเอกสารมาก

2.5 มาตรฐานไอเอสโอ 12207 (ISO 12207)

เนื่องจากอุตสาหกรรมซอฟต์แวร์ในปัจจุบันได้ให้ความสำคัญกับความมีมาตรฐานของหน่วยงานที่ทำการผลิตซอฟต์แวร์เป็นอย่างมากจึงทำให้หน่วยงานที่ประกอบธุรกิจพัฒนาซอฟต์แวร์รวมถึงแผนกเทคโนโลยีสารสนเทศของหน่วยงานต่างๆ ตระหนักถึงความสำคัญตรงจุดนี้ และพยายามที่จะทำให้หน่วยงานของตนมีมาตรฐานเป็นที่ยอมรับซึ่งมาตรฐานที่เป็นที่นิยมและยอมรับกันโดยทั่วไปก็คือ มาตรฐานไอเอสโอ 12207

มาตรฐานสำหรับกระบวนการผลิตและพัฒนาซอฟต์แวร์ มาตรฐานไอเอสโอ 12207 เป็นเกณฑ์คุณภาพของการผลิตซอฟต์แวร์ที่มีลักษณะเป็น โพรเซสโมเดลลิง (Process Modeling) คือเน้นในส่วนของการกำหนดขั้นตอนที่ละเอียดขั้นตอน (Process) ในการผลิตซอฟต์แวร์ที่มีคุณภาพ ตั้งแต่เริ่มต้นจนกระทั่งจบขั้นตอนของการผลิตซอฟต์แวร์ เพื่อให้การผลิตซอฟต์แวร์นั้นมีคุณภาพ โดยจะมีการกำหนดว่าจะมีผลลัพธ์จากขั้นตอนของการผลิตซอฟต์แวร์ตามที่กำหนดในมาตรฐานไอเอสโอ 12207

ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved