บทที่ 3

แนวคิดในการการพัฒนาระบบ

การพัฒนาระบบ (System Development) หมายถึง ขบวนการที่ประกอบด้วยการกำหนดความต้องการสารสนเทศ การออกแบบระบบสารสนเทศที่ตรงกับความต้องการนั้น และการนำระบบนั้นไปใช้งาน (Powers, Cheney and Crow, 1990)

โดยทั่วไป เมื่อกล่าวถึงการพัฒนาระบบ มักจะหมายถึงการวิเคราะห์และการออกแบบระบบ (System Analysis and Design) หรือการพัฒนาระบบสารสนเทศ (Information System Development) ซึ่งเป็นการนำระบบคอมพิวเตอร์ไปประยุกต์ใช้กับระบบงานที่มีอยู่เดิม เพื่อเพิ่มประสิทธิภาพการทำงานให้มากขึ้น

การวิเคราะห์และออกแบบระบบ ประกอบด้วยขั้นตอนต่าง ๆ ดังนี้ การกำหนดปัญหา การศึกษาความเหมาะสม การวิเคราะห์ระบบ การออกแบบระบบ การออกแบบระบบโดยละเอียด การติดตั้งระบบ และการบำรุงรักษาระบบ (วิชาญ เลิศวิภาตระกูล, 2537)

การกำหนดปัญหา

การกำหนดปัญหาเป็นการศึกษาถึงปัญหาที่มีอยู่ในระบบเดิม เพื่อกำหนดขอบเขตและวัตถุประสงค์ของระบบงานที่จะเข้าไปทำการวิเคราะห์และออกแบบ แล้วจัดทำรายงานเสนอผู้บริหารเพื่อให้พิจารณาดำเนินการต่อไป หรือให้ยุติการวิเคราะห์และออกแบบระบบนั้น

การศึกษาความเหมาะสม

การศึกษาความเหมาะสมเป็นการวิเคราะห์และออกแบบระบบอย่างคร่าว ๆ และมีขั้นตอนการทำงานดังนี้

1. กำหนดขอบเขตและวัตถุประสงค์ของงานให้ชัดเจนขึ้น โดยศึกษาจากข้อมูลและเอกสารต่าง ๆ เพิ่มเติมจากขั้นตอนการกำหนดปัญหา

I17054101X

2. ศึกษาระบบงานปัจจุบัน เพื่อให้ทราบว่าระบบงานปัจจุบันประกอบด้วยหน้าที่หลักอะไรบ้าง

3. นำข้อมูลที่ได้จากการศึกษาในข้อ 2 มาจัดทำเป็นแบบจำลองทางตรรกะของระบบงานเดิม (Old Logical Model) แล้ววิเคราะห์ถึงปัญหาที่มีอยู่ในระบบงานเดิมและหาแนวทางแก้ไข นำผลที่ได้จากการวิเคราะห์มาจัดทำเป็นแบบจำลองทางตรรกะของระบบงานใหม่ (New Logical Model)

4. ทบทวนงานในขั้นตอนที่ 1-3 และแก้ไขให้ถูกต้อง

5. พิจารณาส่วนที่จะใช้เครื่องคอมพิวเตอร์ทำงาน พร้อมทั้งศึกษาความเหมาะสมในด้านต่อไปนี้ (Mensching and Adams, 1991)

5.1 ศึกษาความเหมาะสมทางด้านเทคนิค (Technical Feasibility) โดยพิจารณาว่าเทคโนโลยีที่ต้องการสามารถจัดหามาได้ และมีบุคลากรที่มีความรู้พอที่จะใช้อุปกรณ์เหล่านั้นหรือไม่

5.2 ศึกษาความเหมาะสมด้านการเงิน (Financial Feasibility) เป็นการศึกษาเพื่อให้ทราบถึงค่าใช้จ่ายที่ต้องใช้ในการพัฒนาระบบ และผลประโยชน์ที่จะได้รับจากระบบใหม่ว่าคุ้มกันหรือไม่

5.3 ศึกษาความเหมาะสมด้านกฎหมาย (Legal Feasibility) โดยพิจารณาว่าเหมาะสมกับ กฎ ระเบียบ หรือมีผลกระทบต่อศีลธรรมและสังคมหรือไม่

5.4 ศึกษาความเหมาะสมด้านการปฏิบัติงาน (Operational Feasibility) โดยพิจารณาว่าเมื่อพัฒนาระบบใหม่แล้ว จะสามารถปฏิบัติงานได้จริงหรือไม่

5.5 ศึกษาความเหมาะสมด้านกำหนดการ (Schedule Feasibility) โดยพิจารณาแผนการพัฒนาระบบใหม่ว่าจะสำเร็จทันต่อการใช้งานหรือไม่

6. ตัดสินใจเลือกระบบที่เหมาะสมที่สุด

7. กำหนดแผนการพัฒนาระบบใหม่จากระบบที่เลือกไว้

8. จัดทำรายงานสรุปการศึกษาความเหมาะสม

9. นำเสนอผลสรุปต่อผู้บริหารและผู้ใช้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

การวิเคราะห์ระบบ

เป็นการวิเคราะห์ผลสรุปที่ได้จากการศึกษาความเหมาะสม เมื่อผู้บริหารอนุมัติให้ดำเนินการต่อ นักวิเคราะห์ระบบต้องวิเคราะห์ระบบที่เสนออีกครั้งหนึ่ง เพื่อให้ได้ระบบใหม่ที่สมบูรณ์ที่สุดภายใต้วัตถุประสงค์ที่ตั้งไว้

ขั้นตอนการทำงาน

1. วิเคราะห์ผลที่ได้จากขั้นตอนการศึกษาความเหมาะสม และศึกษาเพิ่มเติมในรายละเอียดของระบบ

2. วิเคราะห์ข้อมูลเข้าว่ามีอะไรบ้าง

3. วิเคราะห์ผลลัพธ์ที่ต้องการว่ามีอะไรบ้าง

4. จัดทำแบบจำลองของระบบใหม่ให้สมบูรณ์ยิ่งขึ้น

การออกแบบระบบ

การออกแบบระบบ เป็นการพิจารณาว่าจะติดตั้งระบบอย่างไร เพื่อให้ได้ข้อสนเทศตามต้องการ

ขั้นตอนการทำงาน

1. นำเอกสารที่ได้จากการวิเคราะห์ระบบ และจากการศึกษาความเหมาะสมมาพิจารณาอีกครั้งหนึ่ง

2. สรุปผลการทำงาน

การออกแบบระบบโดยละเอียด

การออกแบบระบบโดยละเอียดประกอบด้วย การออกแบบกระบวนการประยุกต์ (Application Processing Design) การออกแบบรหัสแทนข้อมูล (Code Design) การออกแบบการควบคุมความปลอดภัย (Control and Security Design) การออกแบบข้อมูลเข้า (Input Design) การออกแบบข้อมูลออก (Output Design) การออกแบบตัวเชื่อมประสานกับผู้ใช้ (User Interface Design) การออกแบบแฟ้มข้อมูลและฐานข้อมูล

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

(File and Data Base Design) และการออกแบบการประมวลผล (Processing Design) (วิชาญ เลิศวิภาตระกูล, 2537)

1. การออกแบบกระบวนการประยุกต์

การออกแบบกระบวนการประยุกต์ เป็นการออกแบบกระบวนการทำงานของระบบใหม่ ว่าประกอบไปด้วยขั้นตอนการทำงานอะไรบ้าง

2. การออกแบบรหัสแทนข้อมูล

รหัส หมายถึง ชุดของอักขระที่ใช้แทนระเบียนต่าง ๆ ของข้อมูล เพื่อให้เกิดประสิทธิภาพในการประมวลผลด้วยคอมพิวเตอร์

2.1 สาเหตุที่ต้องใช้รหัส

2.1.1 เพื่อใช้เป็นตัวแทนข้อมูล

2.1.2 เพื่อเรียกใช้ข้อมูล

2.1.3 เพื่อแบ่งประเภทของข้อมูล

2.1.4 เพื่อใช้เรียงลำดับข้อมูล

2.1.5 เพื่อใช้เปรียบเทียบ

2.1.6 เพื่อการประหยัดเวลาประมวลผลและเนื้อที่บนสื่อบันทึก

2.2 ข้อควรพิจารณาในการออกแบบรหัส

2.2.1 โครงสร้างของรหัสที่ออกแบบ ควรพอดีกับความต้องการของผู้ใช้และวิธีการประมวลผลที่ใช้

2.2.2 รหัสแต่ละตัวต้องแทนข้อมูลได้ไม่ซ้ำกัน

2.2.3 การออกแบบรหัสควรยืดหยุ่นต่อความต้องการเปลี่ยนแปลง

2.2.4 โครงสร้างของรหัสต้องง่ายต่อการทำความเข้าใจ

2.2.5 การออกแบบรหัสต้องมีมาตรฐาน เพื่อลดความสับสนในการตีความของผู้ใช้

2.2.6 ถ้าเป็นไปได้ อักขระที่ออกเสียงคล้ายกันไม่ควรนำมาใช้

2.2.7 รูปแบบของรหัสควรมีความยาวคงที่ และต้องกำหนดให้ครบทุกตัว

2.2.8 รหัสที่มีความยาวมากกว่าสี่ตัวอักษร หรือเป็นตัวเลขที่มีความยาวเกินห้าตัว ควรแบ่งออกเป็นส่วน ๆ เพื่อให้ง่ายต่อการบันทึกข้อมูล โดยอาจใช้เครื่องหมาย '-' (Hyphen) เชื่อม

2.2.9 ถ้ารหัสนั้นมีความสำคัญต่อการประมวลผลมาก ควรตรวจสอบความถูกต้องของรหัสด้วยเลขโดดตรวจสอบ (Check Digit)

3. การออกแบบการควบคุมและความปลอดภัย

3.1 เหตุผลในการออกแบบ นักออกแบบระบบจะต้องออกแบบโดยให้มีการควบคุมอย่างเพียงพอ ด้วยเหตุผล 2 ประการต่อไปนี้

3.1.1 เพื่อให้แน่ใจว่า การประมวลผลทำได้ถูกต้อง และได้ผลลัพธ์หรือรายงานที่ถูกต้อง

3.1.2 เพื่อเป็นการป้องกันบุคคลที่ไม่พึงประสงค์เข้าไปทำลายระบบ

3.2 ขั้นของการตรวจสอบ

3.2.1 การตรวจสอบด้วยมือ

3.2.1.1 การตรวจสอบขณะที่มีการจัดเตรียมเอกสารต้นฉบับ

ก) โดยการออกแบบเอกสารต้นฉบับให้สามารถตรวจสอบความถูกต้องได้ (รายละเอียดการออกแบบปรากฏในข้อ 4 การออกแบบข้อมูลเข้า)

ข) โดยการให้ความรู้และอบรมเจ้าหน้าที่ที่เกี่ยวข้อง ให้สามารถจัดเตรียมเอกสารเอกสารต้นฉบับได้อย่างถูกต้อง

3.2.1.2 การตรวจสอบขณะส่งผ่านเอกสารไปที่ต่าง ๆ

ก) โดยการใช้สำเนาส่งต่อ

ข) โดยใช้แบบฟอร์มควบคุมเอกสาร

3.2.1.3 การตรวจสอบขณะบันทึกข้อมูลบนสื่อบันทึกข้อมูล

ก) โดยการทวนสอบ (Verification) เช่น การบันทึกซ้ำ ถ้าข้อมูลที่บันทึกใหม่ไม่ตรงกับที่บันทึกครั้งแรกให้มีเสียงเตือนเป็นต้น

ข) โดยการบรรณาธิกรณ์ (Editing) เช่น อนุญาตให้สามารถไขข้อมูลที่ผิดพลาดได้

ค) โดยการตรวจสอบด้วยสายตา (Visual Check) โดยการพิมพ์ออกมาตรวจสอบหรือดูจากจอภาพ

3.2.2 การตรวจสอบด้วยการประมวลผลด้วยคอมพิวเตอร์

3.2.2.1 ขณะบันทึก มีเทคนิคการตรวจสอบ เช่น

ก) การพิสูจน์ผลรวม (Proof Total) เช่น การตรวจสอบผลรวมในเอกสารว่าตรงกับผลรวมที่เครื่องคำนวณให้หรือไม่

ข) การควบคุมผลรวม (Control Total) เช่น การนับจำนวนเอกสารเปรียบเทียบกับจำนวนระเบียนที่บันทึก

ค) การใช้เลขโดดตรวจสอบ

3.2.2.2 ขณะปรับปรุงแฟ้มข้อมูล มีเทคนิค เช่น

ก) โดยการทำระเบียนซ้ำ (Double Record Check) เช่น การบันทึกซ้ำสองครั้งแล้วนำข้อมูลไปประมวลผลสองครั้ง เปรียบเทียบผลที่ได้ว่าตรงกันหรือไม่

ข) โดยการเปรียบคู่ที่ไม่ตรงกัน (Unmatched Record Check) เช่น ให้มีการเปรียบเทียบรหัสในแฟ้มข้อมูลหลักกับแฟ้มข้อมูลรายการ หากไม่ตรงกันให้บันทึกรายการที่ไม่ตรงกันในแฟ้มใหม่เพื่อนำไปใช้แก้ไขข้อมูลต่อไป

3.2.2.3 ขณะคำนวณ มีเทคนิค เช่น

ก) โดยการตรวจสอบส่วนล้น (Overflow Check) เช่น ก่อนคำนวณจริงให้เก็บค่าในตัวแปรใหม่และลองคำนวณ แล้วตรวจสอบว่าค่าที่ได้เกินขีดจำกัดหรือไม่

ข) โดยการตรวจสอบเครื่องหมาย (Sign Check) เช่น ตรวจสอบเครื่องหมายบวก หรือ ลบ ก่อนการคำนวณ

ค) โดยการตรวจสอบการหารด้วยศูนย์ (Division-by-zero Check) เช่น ตรวจสอบว่าตัวหารเป็นศูนย์หรือไม่

3.2.3 การตรวจสอบในช่วงของการจัดเตรียมผลลัพธ์

3.2.3.1 การเข้าถึงผลลัพธ์ ควรมีการตรวจสอบให้ผู้มีสิทธิเท่านั้นที่จะจัดทำหรือเรียกดูรายงานต่างๆ ได้ เนื่องจากรายงานบางอย่างเป็นความลับ

3.2.3.2 การส่งผ่านรายงาน การส่งผ่านรายงานไปยังผู้ใช้ ต้องมีมาตรการควบคุมไม่ให้รายงานสูญหาย หรือถูกขโมย สำหรับรายงานที่เป็นความลับควรมีมาตรการควบคุมพิเศษ

3.2.4 การตรวจสอบโดยผู้ใช้ เป็นการแบ่งหน้าที่รับผิดชอบออกเป็นส่วน ๆ และให้มีผู้รับผิดชอบในแต่ละส่วน แทนที่จะให้บุคคลใดบุคคลหนึ่งหรือเจ้าหน้าที่กลุ่มใดกลุ่มหนึ่งรับผิดชอบงานตั้งแต่ต้นจนจบ จะทำให้เกิดระบบควบคุมภายในและสามารถตรวจสอบซึ่งกันและกันได้

3.3 ความปลอดภัยในองค์กร

3.3.1 สิ่งที่ควรคำนึงถึง

3.3.1.1 ความพร้อมของข้อมูลและสามารถจัดหามาได้

3.3.1.2 ความถูกต้องครบถ้วนของข้อมูล

3.3.1.3 ข้อมูลที่เป็นความลับ

3.3.2 การควบคุมการเข้าถึง (Access Control)

3.3.2.1 การควบคุมการเข้าถึงทางกายภาพ (Physical Accsess Control) เช่นการปิดห้องเครื่องคอมพิวเตอร์ การใช้ยาม หรือบัตรผ่านเข้าออกเพื่อควบคุมการเข้าถึงเครื่องคอมพิวเตอร์และอุปกรณ์อื่น ๆ จากผู้ที่ไม่มีส่วนเกี่ยวข้อง

3.3.2.2 การควบคุมการเข้าถึงทางตรรกะ (Logical Access Control) เช่น การใช้รหัสผ่าน การตรวจสอบลายนิ้วมือ

4. การออกแบบข้อมูลเข้า

4.1 สิ่งที่ควรคำนึงถึงในการออกแบบเอกสารต้นฉบับ (Kendall and Kendall, 1992)

4.1.1 ควรง่ายต่อการบันทึกข้อมูล

4.1.1.1 ควรให้ผู้ใช้บันทึกข้อมูลจากซ้ายไปขวาและจากบนลงล่าง

4.1.1.2 ควรประกอบไปด้วยส่วนต่าง ๆ เช่น

ก) ส่วนหัว (Heading Zone) ใช้แสดงชื่อของเอกสาร

ข) ส่วนควบคุม (Control Zone) ใช้ควบคุมเอกสาร เช่น หมายเลขของเอกสาร วันที่ที่บันทึกข้อมูลบนเอกสาร

ค) ส่วนแสดงคำชี้แจง (Instruction Zone) ใช้แสดงคำอธิบายประกอบการบันทึกข้อมูล เพื่อให้ผู้ใช้บันทึกข้อมูลได้ถูกต้อง

ง) ส่วนแสดงรายละเอียดต่าง ๆ ของข้อมูล (Body Zone)

จ) ส่วนแสดงผลรวม (Total Zone)

ฉ) ส่วนที่ให้ผู้มีอำนาจลงนาม (Authorization Zone)

4.1.1.3 ควรมีเนื้อที่ให้ผู้ใช้สามารถบันทึกข้อมูลได้ เช่น ให้ผู้ใช้บันทึกข้อมูลในกรอบ หรือบนจุดไข่ปลา หรือให้ทำเครื่องหมายถูก/ผิดในช่องที่ต้องการ

4.1.2 ต้องแน่ใจได้ว่าแบบพิมพ์บรรลุวัตถุประสงค์ที่กำหนดไว้ เช่น ต้องมีสำเนาสำหรับหน่วยต่าง ๆ ที่เอกสารผ่านไป

4.1.3 ต้องแน่ใจในความถูกต้องสมบูรณ์ของข้อมูล เช่น มีช่องผลรวมทั้งแนวตั้งและแนวนอน เพื่อให้สามารถตรวจสอบความถูกต้องได้

4.1.4 ควรมีรูปแบบที่น่าสนใจ โดย

4.1.4.1 ไม่ควรออกแบบให้มีข้อความแน่นมาก

4.1.4.2 การบันทึกข้อมูลควรเรียงไปตามลำดับที่ผู้ใช้คาดหมาย

4.1.4.3 ใช้รูปแบบตัวอักษรหลายรูปแบบและตีเส้นกรอบเป็นส่วน ๆ

4.2 <u>การออกแบบจอภาพ</u> แนวทางการออกแบบจอภาพ (Kendall and Kendall, 1992)

4.2.1 ออกแบบให้ดูง่าย ใช้ง่าย โดยจอภาพควรแบ่งออกเป็น 3 ส่วน คือ ส่วนหัว ใช้แสดงรายการเลือกต่าง ๆ ส่วนกลาง ใช้แสดงข้อมูลและข้อสนเทศ และส่วนท้าย ใช้แสดงข้อแนะนำการใช้งาน เช่น การใช้แป้นกำหนดหน้าที่ (Function Key) นอกจากนี้การออกแบบจอภาพเพื่อให้ใช้ง่าย อาจใช้หน้าต่าง (Window) ช่วย

4.2.2 ออกแบบให้การแสดงผลบนจอภาพมีความต้องกัน

4.2.2.1 ถ้าผู้ใช้ทำงานกับเอกสาร จอภาพควรปรากฏข้อความเช่นเดียวกัน หรือในลำดับเดียวกับที่ปรากฏบนเอกสาร

4.2.2.2 วางตำแหน่งของข้อสนเทศหลักบนจอภาพให้อยู่ในตำแหน่งเดิมทุกครั้งที่มีการเปลี่ยนจอภาพใหม่

4.2.2.3 ข้อสนเทศที่มีความสัมพันธ์กันควรจัดให้อยู่ในกลุ่มเดียวกัน

4.2.3 ออกแบบให้สะดวกต่อการเปลี่ยนจอภาพ

4.2.4 ออกแบบให้จอภาพน่าสนใจ โดย

4.2.4.1 ใช้วีดิทัศน์ผกผัน (Inverse Video) และการกระพริบ (Blinking)

4.2.4.2 ใช้แบบอักษร (Font) หลาย ๆ แบบ

4.2.4.3 ใช้สัญรูป (Icon)

4.2.4.4 ใช้สี

5. <u>การออกแบบข้อมูลออก</u>

5.1 <u>ข้อควรพิจารณาในการออกแบบ</u> (Kendall and Kendall, 1992)

5.1.1 ควรออกแบบให้ตรงตามวัตถุประสงค์

5.1.2 ควรออกแบบให้ตรงกับความต้องการของผู้ใช้

5.1.3 ควรออกแบบเพื่อให้ได้ผลลัพธ์ตามต้องการ

5.1.4 ต้องแน่ใจว่าผลลัพธ์ที่ได้อยู่ในแหล่งที่ต้องการ เช่นเก็บลงแฟ้ม

5.1.5 สามารถได้ผลลัพธ์ทันเวลาที่ต้องการใช้

5.1.6 เลือกวิธีการออกแบบผลลัพธ์ได้ถูกต้อง เช่น ให้แสดงออกทางจอภาพ หรือเครื่องพิมพ์

5.2 <u>ประเภทของรายงาน</u>

แบ่งออกเป็น 2 ประเภท คือ (Shelly, Cashman, Adamski, and Adamski, 1991)

5.2.1 รายงานที่ใช้ภายในหน่วยงาน (Internal Report) เป็นรายงานที่ถูกจัดทำขึ้นและใช้ในหน่วยงาน ผู้ออกแบบควรออกแบบให้ได้รายงานที่มีประโยชน์ แต่มีค่าใช้จ่ายต่ำ

5.2.2 รายงานที่ใช้ภายนอกหน่วยงาน (External Report) เป็นรายงานที่ถูกจัดทำขึ้น และถูกใช้ภายนอกหน่วยงาน มักมีรูปแบบพิเศษ ควรออกแบบให้น่าสนใจและถูกต้องตาม กฎ ระเบียบต่าง ๆ ที่บังคับอยู่

5.3 สื่อที่ใช้ในการออกรายงาน ที่นิยมใช้มี 2 แบบ คือ

5.3.1 เครื่องพิมพ์ โดยพิมพ์ออกทางกระดาษพิมพ์ ซึ่งสามารใช้เป็นสำเนาถาวร (Hard Copy) ได้ แต่มีข้อเสียคือถ้าปริมาณมากจะต้องเตรียมที่เก็บรายงานด้วย

5.3.2 จอภาพ โดยแสดงผลบนจอภาพ ซึ่งสะดวกและรวดเร็วกว่าการพิมพ์ออกทางเครื่องพิมพ์ แต่ไม่มีสำเนาเก็บไว้อ้างอิง

5.4 รูปแบบของรายงาน (Report Layout) แบ่งออกเป็น 3 ส่วนคือ

5.4.1 ส่วนหัว (Heading) แบ่งออกเป็น

5.4.1.1 หัวรายงาน (Report Heading) ใช้แสดงชื่อรายงาน หรือบอกว่าเป็นรายงานเกี่ยวกับอะไร มักจะพิมพ์ครั้งเดียวบนหัวกระดาษหน้าแรก

5.4.1.2 หัวกระดาษ (Page Heading) เป็นชื่อของรายงาน มักจะพิมพ์เป็นส่วนแรกของรายงานแต่ละหน้าและทุกหน้าของรายงาน แต่ถ้าเป็นหน้าแรกจะพิมพ์ต่อจากหัวชุดรายงาน

5.4.1.3 หัวรายการ (Column Heading) เป็นรายการที่พิมพ์ต่อจากหัวกระดาษ และพิมพ์ทุกหน้าเพื่อบอกชื่อข้อมูลที่ปรากฏในส่วนของตัวรายงาน

5.4.1.4 หัวควบคุม (Control Heading) ใช้แสดงชื่อข้อมูลเมื่อมีการเปลี่ยนชุดข้อมูลที่ปรากฏในรายงาน มักจะพิมพ์ในส่วนของตัวรายงาน

5.4.2 ตัวรายงาน (Body) แสดงข้อมูลและรายละเอียดต่าง ๆ

5.4.3 ส่วนท้าย (Footing)

5.4.3.1 ท้ายควบคุม (Control Footing) ใช้สรุปข้อมูลก่อนมีการเปลี่ยนชุดข้อมูล มักใช้คู่กับหัวควบคุม

5.4.3.2 ท้ายกระดาษ (Page Footing) ใช้สรุปข้อมูลในแต่ละหน้าของรายงาน

5.4.3.3 ท้ายรายงาน (Report Footing) ใช้แสดงผลสรุปของรายงานทั้งหมด

5.5 ประเภทของรายงาน

5.5.1 รายงานสอบถาม (Query Report) เป็นการสอบถามข้อมูลจากคอมพิวเตอร์ มักแสดงผลทางจอภาพ และมีลักษณะงานเป็นแบบเชิงโต้ตอบ (Interactive) คือมีการโต้ตอบระหว่างผู้ใช้กับคอมพิวเตอร์

5.5.2 รายงานละเอียด (Detailed Report) เป็นรายงานที่แสดงรายละเอียดข้อมูลทุก ๆ ระเบียน มักใช้ในระดับผู้ปฏิบัติงาน

5.5.3 รายงานสรุป (Summary Report) เป็นรายงานที่ไม่ต้องการรายละเอียดมาก แสดงเพียงผลสรุป มักใช้โดยผู้บริหาร

5.3.4 รายงานพิเศษ (Exception Report) เป็นรายงานที่แสดงข้อมูลที่แตกต่างไปจากกฎเกณฑ์ที่กำหนด หรือแสดงเฉพาะข้อมูลที่ตรงตามเงื่อนไขเฉพาะอย่าง

5.6 <u>แนวทางการออกแบบรายงาน</u> (Kendall and Kendall, 1992)

5.6.1 แบบแผนของรายงาน ประกอบด้วย

5.6.1.1 สารสนเทศคงที่ (Constant Information) เป็นข้อความที่ไม่มีการเปลี่ยนแปลงแม้ว่าข้อมูลจะเปลี่ยนไป เช่น หัวรายงาน หัวรายการ คำอธิบายคำย่อ เป็นต้น

5.6.1.2 สารสนเทศแปรได้ (Variable Information) เป็นสารสนเทศที่เปลี่ยนไปทุกรายการ ตามการประมวลผลข้อมูลนั้น

5.6.2 คุณภาพของกระดาษ ชนิด และขนาดของรายงาน ขึ้นอยู่กับความต้องการใช้งาน หากเป็นรายงานที่ใช้ในหน่วยงาน มักใช้กระดาษคุณภาพต่ำ นอกจากนี้ขนาดของรายงาน มักออกแบบให้พิมพ์ออกกระดาษขนาดมาตรฐานเพื่อการประหยัด ยกเว้นรายงานที่ต้องการความกว้างเป็นพิเศษ

5.6.3 แบบพิมพ์พิเศษต่าง ๆ เช่น ใบเสร็จรับเงิน เช็ค หรือแบบพิมพ์ล่วงหน้า (Preprinted Form) อาจจำเป็นต้องออกแบบไว้

6. <u>การออกแบบตัวเชื่อมประสานกับผู้ใช้</u>

ตัวเชื่อมประสานกับผู้ใช้ เป็นการผสมผสานการใช้ฮาร์ดแวร์และซอฟต์แวร์ และช่วยให้ผู้ใช้สามารถติดต่อและควบคุมการทำงานของระบบได้อย่างสะดวก น่าใช้ และไม่ก่อให้เกิดความเบื่อหน่ายแก่ผู้ใช้

6.1 <u>อุปกรณ์ที่จำเป็นสำหรับตัวเชื่อมประสานกับผู้ใช้</u>

6.1.1 อุปกรณ์ที่สามารถแสดงข่าวสาร โต้ตอบ และ ให้สารสนเทศ เช่น จอภาพ

6.1.2 อุปกรณ์ที่ผู้ใช้สามารถบันทึกข้อมูลและคำสั่งได้ เช่น แผงแป้นอักขระ (Keyboard) เมาส์ (Mouse)

6.1.3 ซอฟต์แวร์ที่ควบคุมการแสดงข่าวสารและกิจกรรมที่เกิดขึ้น เพื่อให้ผู้ใช้ติดต่อกับระบบงานได้ ซึ่งอาจมีลักษณะดังนี้

6.1.3.1 ภาษาควบคุม (Control Language) ใช้ควบคุมการทำงานของระบบ ส่วนใหญ่จะใช้สัญลักษณ์พิเศษนำหน้าและตามด้วยคำสั่งต่าง ๆ

6.1.3.2 ข้อความพร้อมรับ (Prompt) มักเป็นคำอธิบายและให้ผู้ใช้ป้อนข้อมูลตามต้องการ เช่น ข้อความในเครื่องรับจ่ายเงินอัตโนมัติ (Automatic Teller Machine)

6.1.3.3 บรรณาธิกรณ์ข้อมูล (Data Editing) เป็นการปรากฏรูปแบบข้อมูลแล้วให้ผู้ใช้บันทึกตามรูปแบบนั้น และข้อมูลจะถูกตรวจสอบความถูกต้องก่อนทำงานอื่น ๆ เช่น การแสดงรูปแบบวันที่เป็น ปี เดือน วัน แล้วให้ผู้ใช้บันทึกตาม

6.1.3.4 รายการเลือก (Menu) เป็นการแสดงรายการหลาย ๆ รายการและให้ผู้ใช้เลือกรายการที่ต้องการ รายการเลือกประกอบไปด้วย 3 ส่วนคือ

ก) หัวข้อของรายการเลือก

ข) รายการเลือก ซึ่งต้องมีรายการออกจากโปรแกรมได้ด้วย

ค) คำอธิบายประกอบการเลือก

6.1.3.5 ภาษาสอบถาม (Query Language) เป็นการให้ผู้ใช้ป้อนคำสั่ง เพื่อให้เครื่องคอมพิวเตอร์ทำงานต่าง ๆ ใช้มากในซอฟต์แวร์ระบบจัดการฐานข้อมูล

6.1.3.6 ภาษาธรรมชาติ (Natural Language) ให้ผู้ใช้ป้อนคำสั่งซึ่งมีลักษณะใกล้เคียงกับภาษาปกติ เพื่อให้คอมพิวเตอร์ทำงานตามต้องการ

6.2 <u>ผู้ใช้</u> ผู้ออกแบบควรพิจารณาถึง

6.2.1 เวลาที่ใช้ในการเรียนรู้ของผู้ใช้

6.2.2 ความเร็วในการทำงานของผู้ใช้

6.2.3 อัตราข้อผิดพลาดที่เกิดจากผู้ใช้

6.2.4 ความพอใจของผู้ใช้

6.2.5 เวลาที่ใช้ในการทำงาน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

6.3 <u>ระบบและกระบวนการของตัวเชื่อมประสาน</u> ผู้ออกแบบควรพิจารณาถึง

6.3.1 การสนองตอบต่อผู้ใช้ ควรออกแบบให้ง่าย

6.3.2 การออกแบบจอภาพต้องเหมาะสม เช่น แสดงด้วยตัวหนังสือหรือภาพที่ชัดเจน เข้าใจง่ายและมีลักษณะจอภาพที่เหมือนกัน

6.3.3 การโต้ตอบของผู้ใช้ ควรทำได้รวดเร็ว

6.3.4 การกู้ข้อผิดพลาด ควรระบุข้อผิดพลาด เหตุผล และแนวทางแก้ไขให้ผู้ใช้ทราบ และอาจมีเสียงเตือนเมื่อมีข้อผิดพลาดเกิดขึ้น

6.3.5 การควบคุมและความปลอดภัยของข้อมูล เช่นการใช้รหัสผ่าน การกำหนดสิทธิใช้งาน

7. <u>การออกแบบแฟ้มข้อมูลและฐานข้อมูล</u>

7.1 <u>สิ่งที่ควรคำนึงถึงในการออกแบบแฟ้มข้อมูล</u>

7.1.1 มีแฟ้มข้อมูลอะไรบ้างที่ต้องใช้ และจะบันทึกบนสื่ออะไร

7.1.2 การจัดระเบียบแฟ้ม (File Organization) ต้องออกแบบว่าเป็นประเภทใด (รายละเอียดปรากฏในข้อ 7.4)

7.2 <u>ส่วนประกอบของแฟ้มข้อมูล</u>

7.2.1 กุญแจ (Key) เป็นข้อมูลที่ใช้อ้างอิงถึงระเบียนใดระเบียนหนึ่งในแฟ้มข้อมูล และต้องไม่ซ้ำ

7.2.2 รายการอ้างอิง (Reference) เป็นข้อมูลเฉพาะของระเบียนและมีค่าคงที่ไม่เปลี่ยนแปลง

7.2.3 รายการปรับปรุง (Maintenance) เป็นข้อมูลที่ต้องได้รับการดูแลและปรับปรุงค่าอยู่เสมอ

7.2.4 ความต้องการในอนาคต (Future Requirements) เป็นข้อมูลที่ไม่จำเป็นสำหรับระบบปัจจุบัน แต่อาจถูกใช้ในอนาคต

7.3 <u>ชนิดของแฟ้มข้อมูล</u> (File Types) (ยุพิน ไทยรัตนานนท์, 2530)

7.3.1 แฟ้มหลัก (Master Files) เป็นแฟ้มข้อมูลสำคัญของระบบที่ต้องใช้อยู่เสมอ และต้องปรับปรุงให้ทันสมัยตลอดเวลาใช้งาน

7.3.2 แฟ้มตาราง (Table Files) ใช้เก็บตารางเปรียบเทียบค่าหรือใช้อ้างอิงในขณะประมวลผล

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

7.3.3 แฟ้มรายการเปลี่ยนแปลง (Transaction Files) เป็นแฟ้มข้อมูลนำเข้าของระบบและยังไม่ถูกประมวลผล แต่จะถูกนำไปปรับปรุงข้อมูลในแฟ้มหลักให้ทันสมัยอยู่เสมอ

7.3.4 แฟ้มงาน (Work Files) เป็นแฟ้มที่ถูกสร้างขึ้นชั่วคราวเพื่อความสะดวกในขณะประมวลผล

7.3.5 แฟ้มประวัติ (History Files) ใช้เก็บข้อมูลจากแฟ้มหลักที่ไม่ใช้งานแล้ว หรือถูกลบออกจากแฟ้มข้อมูลหลักแล้ว และแฟ้มประวัตินี้อาจถูกนำมาประมวลผลหาค่าทางสถิติ หรือใช้อ้างอิงในภายหลัง

7.3.6 แฟ้มรายงาน (Report Files) ใช้เก็บรายงานต่าง ๆ ที่พิมพ์ออกทางเครื่องพิมพ์

7.3.7 แฟ้มสำรอง (Backup Files) เป็นแฟ้มข้อมูลทั่วไปที่เก็บสำรองไว้ใช้ ในกรณีที่แฟ้มต้นฉบับเกิดความเสียหาย

7.4 <u>การจัดระเบียบแฟ้ม</u> จุดประสงค์ของการจัดระเบียบแฟ้มคือการเตรียมเส้นทางเข้าถึง (Access Path) ระเบียน ระหว่างปฏิบัติการเรียกใช้และปรับปรุงข้อมูล ระเบียบแฟ้มต่างชนิดกันจะให้เส้นทางเข้าถึงข้อมูลที่แตกต่างกัน (ยุพิน ไทยรัตนานนท์, 2530)

การจัดระเบียบแฟ้ม แบ่งออกเป็น 3 ประเภท

7.4.1 การจัดระเบียบแฟ้มลำดับ (Sequential File Organization) เป็นการจัดเก็บระเบียนเรียงลำดับบนสื่อบันทึกเมื่อแรกสร้างแฟ้มข้อมูล การเข้าถึงข้อมูล ต้องเข้าถึงตามลำดับของการจัดเก็บด้วย

7.4.2 การจัดระเบียบแฟ้มสุ่ม (Random File Organization) เป็นการจัดเก็บข้อมูลให้สามารถเข้าถึงระเบียนใดระเบียนหนึ่งได้โดยตรง ไม่ต้องผ่านข้อมูลตั้งแต่ต้นอย่างแฟ้มลำดับ

7.4.3 การจัดระเบียบแฟ้มลำดับเชิงดรรชนี (Indexed Sequential File Organization) เป็นการจัดเก็บระเบียนให้มีประสิทธิภาพ เพื่อให้สามารถเข้าถึงระเบียนแบบลำดับได้โดยกุญแจบางตัว ในขณะเดียวกันก็สามารถเข้าถึงระเบียนหนึ่งระเบียนใดแบบสุ่มโดยกุญแจตัวเดียวกัน ข้อกำหนดของแฟ้มชนิดนี้คือ ทุกระเบียนต้องมีกุญแจ และแฟ้มข้อมูลต้องเก็บบนอุปกรณ์หน่วยเก็บเข้าถึงโดยตรง (Direct Access Storage Device)

7.5 <u>ระบบจัดการฐานข้อมูล</u> (Database Management System)

ฐานข้อมูล คือการรวมแฟ้มข้อมูลตั้งแต่สองแฟ้มขึ้นไป ที่มีความสัมพันธ์เชิงตรรกะ เพื่อลดความซ้ำซ้อนของข้อมูล และทำให้การบำรุงรักษาทำได้ง่ายขึ้น (ยุพิน ไทยรัตนานนท์, 2530)

ระบบจัดการฐานข้อมูล เป็นซอฟต์แวร์ระบบ (System Software) ที่ออกแบบมาเพื่อบริหารข้อมูลในฐานข้อมูล อำนวยความสะดวกในการจัดการแฟ้มทางกายภาพ การเข้าถึงข้อมูล การบำรุงรักษาฐานข้อมูลให้อิสระจากโปรแกรมประยุกต์และการควบคุมความปลอดภัย การฟื้นฟูข้อมูล และการตรวจสอบความถูกต้องของข้อมูล (ยุพิน ไทยรัตนานนท์, 2530)

7.5.1 แบบจำลองข้อมูล (Data Model) แบ่งออกเป็น 3 ชนิด (Date, 1986)

7.5.1.1 แบบจำลองข้อมูลเชิงสัมพันธ์ (Relational Data Model) คือการเก็บข้อมูลแบบเป็นตารางหรือความสัมพันธ์ นั่นคือ ระเบียนที่มีความสัมพันธ์กัน ต้องมีค่าข้อมูลในเขตข้อมูลใดเขตข้อมูลหนึ่งเหมือนกัน

7.5.1.2 แบบจำลองข้อมูลเชิงเครือข่าย (Network Data Model) เป็นการรวมระเบียนต่าง ๆ และความสัมพันธ์ระหว่างระเบียน โดยแสดงความสัมพันธ์ไว้ในโครงสร้างอย่างชัดเจน และบัพลูก (Child Node) สามารถมี บัพแม่ (Parent Node) ได้มากกว่า 1 บัพ ดังนั้น การเข้าถึงข้อมูลที่อยู่ในระดับต่ำ สามารถเข้าถึงโดยผ่านบัพแม่หลาย ๆ บัพที่อยู่ในระดับสูงกว่าได้

7.5.1.3 แบบจำลองข้อมูลเชิงลำดับชั้น (Hierarchical Data Model) เป็นการรวมระเบียนที่มีความสัมพันธ์เป็นแบบแม่ถึงลูก (Parent-to-child) หรือ แต่ละบัพลูกจะมีบัพแม่ได้ 1 บัพเท่านั้น ดังนั้นการเข้าถึงข้อมูลที่อยู่ในระดับต่ำ จะสามารถเข้าถึงโดยผ่านบัพแม่ได้เท่านั้น

7.5.2 ประโยชน์ที่ได้จากระบบจัดการฐานข้อมูล (Date, 1986)

7.5.2.1 ลดความซ้ำซ้อนของข้อมูล

7.5.2.2 สามารถหลีกเลี่ยงความขัดแย้งของข้อมูลได้ในระดับหนึ่ง

7.5.2.3 สามารถใช้ข้อมูลร่วมกันได้

7.5.2.4 สามารถควบคุมความเป็นมาตรฐานได้

7.5.2.5 สามารถจัดหาระบบความปลอดภัยที่รัดกุมได้

7.5.2.6 สามารถควบคุมความคงสภาพของข้อมูลได้

7.5.2.7 สามารถสร้างสมดุลในความขัดแย้งของความต้องการได้

7.5.2.8 เกิดความเป็นอิสระของข้อมูล

7.6 การออกแบบฐานข้อมูล (ดวงแก้ว สวามิภักดิ์, 2534)

7.6.1 เปลี่ยนรูปแบบของความต้องการให้อยู่ในรูปลักษณ์ของความสัมพันธ์

7.6.2 ทำความสัมพันธ์ให้เป็นบรรทัดฐาน (Normalize Relation)

7.6.3 กำหนดเขตข้อมูลที่เป็นกุญแจ และคุณสมบัติของกุญแจแต่ละตัว

7.6.4 พิจารณาข้อจำกัดและกฎเกณฑ์อื่น ๆ

7.6.5 นำผลที่ได้จากทุกขั้นตอนมาผนวกกัน

8. การออกแบบการประมวลผล

8.1 การประมวลผลหลัก (Shelly et al., 1991)

8.1.1 การบันทึกและการตรวจสอบ ในการประมวลผลแบบกลุ่ม (Batch Processing) ข้อมูลจะถูกบันทึกบนสื่อบันทึก และพิมพ์ออกมาเพื่อตรวจสอบความถูกต้องก่อนนำไปประมวลผล ในการประมวลผลแบบเชื่อมตรง (On-line Processing) ผู้ใช้บันทึกข้อมูลเข้าเครื่องโดยตรง ควรมีซอฟต์แวร์ตรวจสอบความถูกต้องของข้อมูลที่สำคัญบางรายการก่อนนำไปประมวลผล

8.1.2 การปรับปรุงแฟ้มข้อมูล เป็นการปรับปรุงข้อมูลในแฟ้มหลัก โดยอาจเป็นการเพิ่มระเบียน ลบระเบียน หรือแก้ไขข้อมูลบางรายการในระเบียนที่มีอยู่เดิม

8.1.3 การเรียงลำดับข้อมูล หลังจากที่ข้อมูลถูกบันทึกและตรวจสอบแล้ว อาจจำเป็นต้องเรียงลำดับข้อมูลก่อนนำไปประมวลผล

8.1.4 การรายงานผล เพื่อให้ได้รายงานตามต้องการ ในการประมวลผลแบบกลุ่ม มักพิมพ์ออกมาเป็นรายงาน แต่ในการประมวลผลแบบเชื่อมตรง มักแสดงออกทางจอภาพ

8.1.5 การประมวลผลที่สนับสนุนการทำงาน

8.1.5.1 การสำรองและการกู้ (Backup and Recovery) ควรพิจารณาให้มีการสำรองแฟ้มข้อมูลไว้ ในกรณีที่แฟ้มข้อมูลถูกทำลาย สามารถ

นำแฟ้มข้อมูลสำรองมาประมวลผลเพื่อให้ระบบกลับคืนสู่สภาพเดิม

8.1.5.2 การเก็บรักษาแฟ้มข้อมูล (File Retention) ควรพิจารณาจัดเก็บแฟ้มข้อมูลทั้งหมดไว้ในที่ปลอดภัย

8.1.5.3 การเริ่มต้นใหม่ (Restart) กรณีที่ระบบล้มเหลวโดยเฉพาะทางด้านฮาร์ดแวร์ ต้องพิจารณาถึงสิ่งต่อไปนี้

ก) ก่อนการประมวลผล แฟ้มข้อมูลทั้งหมดต้องอยู่ในสภาพสมบูรณ์ โดยต้องกำหนดว่าแฟ้มข้อมูลสำรองชุดใดที่จะนำมาใช้

ข) การประมวลผลบางประเภทอาจละข้อผิดพลาดบางประการไว้ก่อนแล้วค่อยกลับไปแก้ไขในภายหลัง

8.1.5.4 การเริ่มต้นการประมวลผล (Start-Up Processing) การเริ่มต้นการประมวลใด ๆ ต้องมีการจัดเตรียมระบบให้พร้อม เช่น การสร้างแฟ้มข้อมูลใหม่

8.2 <u>การไหลของเอกสารและข้อสนเทศ</u> เพื่อให้การประมวลผลสามารถดำเนินการไปได้โดยไม่ติดขัด ต้องคำนึงถึงสิ่งต่อไปนี้

8.2.1 การรวบรวมเอกสารต้นฉบับ

8.2.1.1 กำหนดเวลาในการรวบรวมให้ชัดเจน

8.2.1.2 การจัดการกับเอกสารเมื่อถึงจุดที่ต้องการ เช่น การลงทะเบียนรับ การลงนามอนุมัติ

8.2.1.3 การจัดการกับเอกสารที่ไม่สมบูรณ์ เช่นการส่งคืน

8.2.2 การควบคุมเอกสารต้นฉบับ

8.2.2.1 การควบคุมตัวเอกสารไม่ใช้สูญหาย

8.2.2.2 การควบคุมข้อมูลบนเอกสารกรณีที่เป็นความลับ

8.2.2.3 การควบคุมเอกสารที่เกี่ยวข้อง เช่น สำเนา

8.2.3 การเตรียมข้อมูล

8.2.3.1 กำหนดเวลาส่งมอบเอกสารที่จะนำไปบันทึกบนสื่อบันทึกข้อมูล

8.2.3.2 พิจารณาเวลาที่ต้องใช้ในการเตรียมข้อมูล

8.2.3.3 การแก้ไขปัญหาเมื่อมีข้อผิดพลาดเกิดขึ้น เช่น กรณีได้รับเอกสารไม่ตรงเวลา หรือข้อมูลบนเอกสารไม่ชัดเจน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

8.2.3.4 การจัดการกับเอกสารหลังจากที่บันทึกบนสื่อบันทึกข้อมูลแล้ว

8.2.4 การควบคุมแบบพิมพ์พิเศษที่ใช้ในการทำรายงาน เช่น ใบเสร็จรับเงิน เช็คเงินเดือน ต้องมีการรักษาแบบพิมพ์และควบคุมการใช้แบบพิมพ์เหล่านั้น

8.2.5 การจัดเตรียมผลลัพธ์และการแจกจ่าย ควรออกแบบให้ผู้ใช้สามารถจัดเตรียมผลลัพธ์ และแจกจ่ายไปใช้งาน โดยต้องได้ผลลัพธ์ในรูปแบบที่ผู้ใช้ต้องการและทันต่อการใช้งาน

<u>การติดตั้งระบบ</u>

1. <u>ขั้นตอนการทำงาน</u>

1.1 พิจารณาซอฟต์แวร์ประยุกต์ที่จะนำมาใช้ ซอฟต์แวร์ประยุกต์ แบ่งออกเป็น 2 แบบคือ

1.1.1 แบบซื้อมาใช้งาน แบ่งเป็น

1.1.1.1 แบบวัตถุประสงค์เฉพาะ (Special Purpose) เป็นซอฟต์แวร์ที่ถูกพัฒนาขึ้นมาเพื่อใช้กับงานเฉพาะอย่าง

1.1.1.2 แบบวัตถุประสงค์ทั่วไป (General Purpose) เป็นซอฟต์แวร์ที่ถูกพัฒนาขึ้นมาเพื่อใช้กับงานทั่วไป

1.1.2 แบบพัฒนาขึ้นใช้เอง โดยเขียนโปรแกรมและใช้เครื่องมือต่าง ๆ ช่วย ซึ่งมีข้อควรพิจารณาในการเลือกซอฟต์แวร์ดังนี้

1.1.2.1 ความสามารถของภาษาที่จะใช้

1.1.2.2 ความรู้ของผู้เขียนโปรแกรม

1.1.2.3 ค่าใช้จ่ายของตัวแปลโปรแกรม (Compiler)

1.1.2.4 ความง่ายในการเขียน

1.1.2.5 ประสิทธิภาพของภาษาที่ใช้

1.1.2.6 ความเข้ากันได้หรือใช้แทนกันได้ของภาษา

1.2 พัฒนาข้อกำหนดของโปรแกรม ข้อกำหนดของโปรแกรมประกอบด้วย

1.2.1 คำอธิบายลักษณะระบบโดยย่อ

1.2.2 ผังงานระบบ (System Flowchart)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

1.2.3 รายชื่อโปรแกรมและคำอธิบายโดยย่อ

1.2.4 รูปแบบของข้อมูลเข้า รายงานที่ได้จากระบบ และแฟ้มข้อมูลที่ใช้

1.2.5 คำอธิบายรายละเอียดการประมวลผลของแต่ละโปรแกรม

1.3 วิเคราะห์ข้อกำหนดของโปรแกรมแล้วพัฒนาตรรกะของโปรแกรมและผังงานที่ใช้แก้ปัญหา

1.4 เขียนโปรแกรมและตรวจสอบความถูกต้องของโปรแกรม

1.5 เตรียมข้อมูลที่จะใช้ทดสอบ

1.6 ทดสอบและแก้ไขโปรแกรมให้ถูกต้อง

1.7 จัดทำเอกสารประกอบโปรแกรม

2. <u>วงจรการพัฒนาโปรแกรม</u>

2.1 <u>พิจารณาข้อกำหนดของโปรแกรม</u>

2.2 <u>ออกแบบโปรแกรม</u> โดยมีเทคนิคดังนี้

2.2.1 การเขียนโปรแกรมแบบโครงสร้าง

2.2.2 การออกแบบโครงสร้าง ใช้ในกรณีที่โปรแกรมมีขนาดใหญ่ เช่น ใช้แผนภูมิโครงสร้าง (Structure Chart) โดยมีข้อควรพิจารณาในการออกแบบดังนี้ (Eliason, 1990)

2.2.2.1 ควรแบ่งโปรแกรมออกเป็นมอดูล (Module) เล็ก ๆ แบบเป็นลำดับขั้น

2.2.2.2 แต่ละมอดูลควรขึ้นอยู่กับมอดูลอื่นๆ เพียงเล็กน้อย

2.2.2.3 แต่ละมอดูลควรทำหน้าที่ใดหน้าที่หนึ่ง

2.2.2.4 ควรตั้งชื่อมอดูลตามหน้าที่

2.2.2.5 มอดูลหนึ่ง ๆ ควรจะทำหน้าที่ประสานงานมอดูลที่อยู่ในระดับต่ำไม่มากจนเกินไป

2.2.2.6 แต่ละมอดูลควรมีขนาดพอสมควร

2.2.2.7 ควรออกแบบให้ใช้งานมอดูลร่วมกันได้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

2.3 <u>การเขียนโปรแกรม</u> ควรเขียนให้โปรแกรมมีลักษณะดังนี้ (Eliason, 1990)

2.3.1 ง่ายต่อการอ่าน

2.3.2 เขียนโปรแกรมแบบโครงสร้าง

2.3.3 มีความต้องกันกับผังโครงสร้าง

2.3.4 มีคำอธิบายโดยย่อ

2.3.5 ทดสอบได้

2.4 <u>การทดสอบโปรแกรม</u>

2.4.1 การทดสอบโปรแกรมทั่ว ๆ ไป มีวิธีการทดสอบ 2 แบบคือ

2.4.1.1 การทดสอบข้อมูล (Data Driven Testing) เป็นการเตรียมข้อมูลที่เป็นตัวแทนของข้อมูลทั้งหมดขึ้นมาเพื่อทดสอบ แล้วนำผลที่ได้ไปตรวจสอบกับผลที่ทราบอยู่แล้ว

2.4.1.2 การทดสอบตรรกะ (Logic Driven Testing) เป็นการสมมติข้อมูลให้โปรแกรมทำงาน และติดตามการทำงานทุกขั้นตอนของโปรแกรม

2.4.2 การทดสอบแบบโครงสร้าง (Structured Testing) ใช้สำหรับโปรแกรมที่แบ่งออกเป็นหน่วยย่อย ๆ

2.4.2.1 การทดสอบจากบนลงล่าง (Top-down Testing) เป็นการทดสอบจากโปรแกรมหลักไปสู่โปรแกรมย่อย โดยสมมติส่วนที่เป็นหน่วยย่อยขึ้นมา ซึ่งภายในหน่วยย่อยไม่มีคำสั่งอื่น ๆ นอกจากชื่อหน่วยย่อย และคำสั่งกลับ (Return)

2.4.2.2 การทดสอบจากล่างขึ้นบน (Bottom-up Testing) เป็นการทดสอบโปรแกรมแต่ละหน่วยอย่างอิสระ แล้วค่อยทดสอบรวมไปหาโปรแกรมหลัก

2.4.3 ขั้นตอนการทดสอบ

2.4.3.1 การแปลโปรแกรม เป็นการทดสอบวากยสัมพันธ์ (Syntax) ของโปรแกรม

2.4.3.2 การทดสอบแต่ละหน่วยของโปรแกรม โดยใช้ข้อมูลสำหรับทดสอบ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

2.4.3.3 การตรวจสอบการทำงานของโปรแกรม ทำได้ 2 วิธี คือ

ก) ตรวจสอบโดยผู้เขียนโปรแกรม

ข) ตรวจสอบโดยคณะกรรมการ

2.4.3.4 ใช้โปรแกรมตรวจสอบจุดบกพร่อง (Debugger)

2.4.3.5 ใช้โปรแกรมวิเคราะห์ (Analyzer)

2.5 <u>การจัดทำเอกสาร</u> (Shelly et al., 1991)

2.5.1 เอกสารประกอบโปรแกรม กล่าวถึงวัตถุประสงค์ของโปรแกรม รูปแบบข้อมูลที่ใช้ การบันทึกข้อมูล แฟ้มข้อมูลที่ใช้ ผังงานระบบ รายละเอียดการประมวลผลของแต่ละโปรแกรม ผังโครงสร้างโปรแกรม รายชื่อโปรแกรม และผลที่ได้

2.5.2 เอกสารประกอบระบบ กล่าวถึงความสามารถของระบบและขบวนการพัฒนาระบบ

2.5.3 เอกสารประกอบการปฏิบัติงาน จะจัดทำขึ้นเฉพาะการพัฒนาระบบเครื่องคอมพิวเตอร์ระดับเล็ก (Minicomputer) หรือเครื่องคอมพิวเตอร์ระดับใหญ่ (Mainframe) กล่าวถึงการปฏิบัติงานของเจ้าหน้าที่ปฏิบัติการในห้องเครื่อง

2.5.4 เอกสารสำหรับผู้ใช้ เป็นเอกสารที่ให้ผู้ใช้เรียนรู้และเป็นแนวทางในการใช้โปรแกรม

3. <u>การทดสอบระบบ</u> (Shelly et al., 1991)

การทดสอบระบบ เป็นการทดสอบระบบทั้งหมดโดยใช้ข้อมูลที่ถูกจัดเตรียมมาจากผู้ใช้ วัตถุประสงค์ของการทดสอบคือ

3.1 เพื่อทำการทดสอบโปรแกรมกับข้อกำหนดที่ออกแบบไว้ว่าตรงกันหรือไม่

3.2 เพื่อให้มั่นใจว่าผู้ใช้มีเอกสารแนะนำการใช้อย่างเพียงพอ

3.3 เพื่อเป็นการรับประกันว่าว่าผู้ใช้สามารถโต้ตอบกับระบบได้

3.4 เพื่อทดสอบการสำรองข้อมูลและการเริ่มระบบ

3.5 เพื่อตรวจสอบส่วนประกอบของระบบทั้งหมดว่ามีความถูกต้อง ข้อมูลและข้อสนเทศสามารถเคลื่อนไปในแต่ละหน่วยงานได้อย่างราบรื่นภายในเวลาที่กำหนด

3.6 เพื่อยืนยันว่าระบบสามารถตอบรับงานในปริมาณปกติได้ทันเวลา

4. <u>การติดตั้งและประเมินผล</u> (Shelly et al., 1991)

4.1 <u>สภาพแวดล้อมการปฏิบัติงานและการทดสอบ</u> หลังจากติดตั้งระบบควรจะมีการติดตามการปฏิบัติงานของเจ้าหน้าที่กับระบบใหม่ และหากเกิดความต้องการเปลี่ยนแปลงใด ๆ ต้องทำการทดสอบโปรแกรมส่วนที่มีการเปลี่ยนแปลงนั้น ก่อนที่จะนำไปติดตั้งเพื่อใช้งานต่อไป

4.2 <u>การอบรมผู้ใช้</u> เป็นการอบรมเพื่อให้ผู้ใช้สามารถปฏิบัติงานในระบบใหม่ได้ ในบางกรณีอาจมีการอบรมด้านฮาร์ดแวร์หรือซอฟต์แวร์โดยบริษัทตัวแทนจำหน่ายด้วย

4.3 <u>การแปลงสภาพแฟ้ม</u> (File Conversion) เป็นการนำแฟ้มข้อมูลที่มีอยู่ในระบบเดิมมาแปลงให้อยู่ในรูปที่สามารถจะนำไปประมวลผลในระบบใหม่ได้

4.4 <u>การแปลงผันระบบ</u> (System Change Over) เป็นการเปลี่ยนการทำงานจากระบบงานเดิมไปใช้ระบบที่พัฒนาขึ้นใหม่ ซึ่งมีวิธีการแปลงผัน 4 วิธี คือ

4.4.1 แบบขนาน (Parallel) เป็นการติดตั้งระบบใหม่และให้ดำเนินการไปพร้อม ๆ กับระบบเก่า จนแน่ใจว่าระบบใหม่มีความสมบูรณ์จึงเลิกใช้ระบบเก่า วิธีนี้เสียค่าใช้จ่ายมาก แต่มีความเสี่ยงน้อย

4.4.2 แบบระยะ (Phase) เป็นการเปลี่ยนไปใช้ระบบใหม่ทีละขั้น ทำให้ผู้ใช้เกิดความคุ้นเคยกับระบบใหม่ แต่ข้อเสียคือ งานบางงานหาจุดแบ่งงานได้ยาก

4.4.3 แบบนำร่อง (Pilot) ใช้กับหน่วยงานที่มีหลายสาขาโดยให้สาขาหนึ่งทดลองใช้ระบบใหม่ก่อน และเมื่อเห็นว่าระบบสมบูรณ์แล้วจึงให้สาขาอื่นใช้

4.4.4 แบบตรง (Direct) เป็นการเปลี่ยนจากระบบเดิมเป็นระบบใหม่ทันที วิธีนี้เหมาะกับระบบงานขนาดเล็กและไม่ซับซ้อน ค่าใช่จ่ายต่ำ แต่ความเสี่ยงสูง

4.5 <u>การประเมินผล</u> หลังจากติดตั้งระบบแล้วควรจะมีการประเมินผลระบบที่ออกแบบ โดย

4.5.1 สัมภาษณ์เจ้าหน้าที่ในระดับบริหารและผู้ใช้

4.5.2 สังเกตผู้ใช้และการปฏิบัติงานกับระบบใหม่

4.5.3 อ่านเอกสารประกอบระบบและเอกสารประกอบการอบรม

4.5.4 พิจารณาเอกสารต้นฉบับ รายงานที่ได้และการแสดงข้อสนเทศบนจอภาพ

4.5.5 ใช้แบบสอบถามเพื่อเก็บข้อมูลและความคิดเห็นของผู้ใช้หากเป็นระบบที่ใหญ่มาก

4.6 จัดทำรายงานเสนอผู้บริหาร เป็นรายงานสรุปการติดตั้งระบบใหม่และสรุปผลการติดตั้ง

การบำรุงรักษา

1. ลักษณะของการบำรุงรักษา (Eliason, 1990)

1.1 การทำให้ถูกต้อง (Corrective Maintenance) เป็นการแก้ไขข้อผิดพลาดที่เกิดขึ้น เช่น แก้ไขโปรแกรมและเอกสารประกอบการทำงาน เพื่อให้ระบบทำงานได้ถูกต้อง

1.2 การปรับปรุง (Adaptive Maintenance) เป็นการปรับปรุงระบบให้สอดคล้องกับความต้องการที่เปลี่ยนไป

1.3 การทำให้สมบูรณ์ (Perfective Maintenance) เป็นการทำให้ระบบทำงานได้มีประสิทธิภาพมากขึ้น

2. งานที่ต้องทำ

2.1 ทำความเข้าใจกับโปรแกรม

2.2 ปรับปรุงโปรแกรม

2.3 ทดสอบความถูกต้องของโปรแกรม

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย