

บทที่ 2

ทฤษฎีและเอกสารที่เกี่ยวข้อง

การพัฒนาโปรแกรมการฝากเงินได้ทำการออกแบบโดยในบทนี้จะกล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้องที่ใช้ในการพัฒนาโปรแกรมการฝากเงิน

2.1 การวิเคราะห์และออกแบบระบบ (System Analysis and Design)

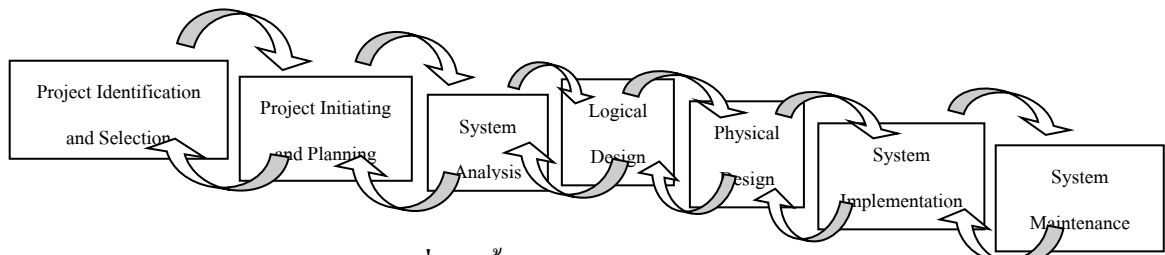
การวิเคราะห์และออกแบบระบบคือ วิธีการที่ใช้ในการสร้างระบบสารสนเทศขึ้นมาใหม่ในธุรกิจใดธุรกิจหนึ่ง หรือระบบย่อยของธุรกิจ นอกจากการสร้างระบบสารสนเทศเดิมที่มีอยู่แล้วให้ดีขึ้นด้วยก็ได้ ดังนั้นการวิเคราะห์ระบบคือ การหาความต้องการ (Requirements) ของระบบสารสนเทศว่าคืออะไร หรือต้องการเพิ่มเติมอะไรเข้ามาในระบบและการออกแบบก็คือ การนำเอาความต้องการของระบบมาเป็นแบบแผนหรือเรียกว่าพิมพ์เขียวในการสร้างระบบสารสนเทศนั้นให้ในงานได้จริง ผู้ที่ทำหน้าที่นี้ก็คือ นักวิเคราะห์และออกแบบระบบ

2.1.1 การไหลของข้อมูล (Data Flow Diagram : DFD)

Data Flow Diagram เป็นเครื่องมือของนักวิเคราะห์ระบบที่ช่วยให้สามารถเข้าใจกระบวนการทำงานของแต่ละหน่วยงาน ซึ่งทราบถึงการรับ ส่งข้อมูล การประสานงานระหว่างกิจกรรมต่าง ๆ ในระบบนั้นนอกจากนี้ยังช่วยให้รู้ถึงความต้องการข้อมูลและข้อบกพร่อง (ปัญหา) ในระบบงานเดิม เพื่อใช้ในการออกแบบการปฏิบัติงานในระบบใหม่

2.1.2 ทฤษฎีวงจรชีวิตการพัฒนาระบบ (System Development Life Cycle)

ระบบสารสนเทศทั้งหลายมีวงจรชีวิตที่เหมือนกันตั้งแต่เกิดจนตาย วงจรนี้จะเป็นขั้นตอนที่เป็นลำดับตั้งแต่ต้นจนเสร็จเรียบร้อย เป็นระบบที่ใช้งานได้ ซึ่งนักวิเคราะห์ระบบต้องทำความเข้าใจให้ได้ว่าในแต่ละขั้นตอนจะต้องทำอะไร และทำอย่างไร ขั้นตอนการพัฒนาระบบมีอยู่ด้วยกัน 7 ขั้นตอนด้วยกันคือ



รูปที่ 2.1 ขั้นตอนการพัฒนาระบบ

2.1.2.1 การค้นหาและเลือกสรรโครงการ (Project Identification and selection) เป็นขั้นตอนในการค้นหาโครงการระบบที่เหมาะสมกับสถานการณ์ปัจจุบัน สามารถแก้ปัญหาที่เกิดขึ้นได้ และให้ผลประโยชน์มากที่สุด โดยใช้ตารางเมตริกซ์ (Matrix Table) เป็นเครื่องมือประกอบการพิจารณา ซึ่งเป็นขั้นตอนที่ได้ดำเนินการผ่านไปแล้วในเบื้องต้น สามารถสรุปกิจกรรมได้ดังนี้

- 1.) ค้นหาโครงการพัฒนาระบบที่เห็นสมควรต่อการได้รับการพัฒนา
- 2.) จำแนกและจัดกลุ่มโครงการ
- 3.) เลือกโครงการที่เหมาะสมที่สุดในการพัฒนา

2.1.2.2 การเริ่มต้นและวางแผนโครงการ (Project Initiation and Planning) เป็นขั้นตอนในการเริ่มต้นจัดทำโครงการด้วยการจัดตั้งทีมงาน กำหนดตำแหน่งหน้าที่เพื่อร่วมกันสร้างแนวทางเลือกในการนำระบบใหม่มาใช้งาน และเลือกทางเลือกที่ดีที่สุด จากนั้นร่วมกันวางแผนจัดทำโครงการ กำหนดระยะเวลาในการดำเนินโครงการ ศึกษาความเป็นไปได้ของโครงการและประมาณการต้นทุนและกำไรที่จะได้รับการลงทุนในการพัฒนาระบบ เพื่อนำเสนอและพิจารณาอนุมัติดำเนินการในขั้นต่อไป สรุปกิจกรรมขั้นที่ 2 ดังนี้

- 1) เริ่มต้นโครงการ
- 2) เสนอแนวทางเลือกในการนำระบบใหม่มาใช้
- 3) วางแผนโครงการ

2.1.2.3 การวิเคราะห์ระบบ (System Analysis) เป็นขั้นตอนในการศึกษาและวิเคราะห์ถึงขั้นตอนการดำเนินงานของระบบเดิม ซึ่งการที่จะสามารถดำเนินงานในขั้นตอนนี้ได้ จะต้องผ่านกระบวนวิธีในขั้นตอนที่ 2 ในการนำเสนอโครงการ หลังจากนั้นจะรวบรวมความต้องการในระบบใหม่จากผู้ใช้ระบบ และนำมาศึกษาและวิเคราะห์ความต้องการเหล่านั้นด้วยการใช้เครื่องมือชนิดต่าง ๆ ได้แก่ แบบจำลอง ขั้นตอนการทำงานของระบบ (Process Modelling) โดยใช้แผนภาพกระแสข้อมูล (Data Flow Diagram) และแบบจำลองข้อมูล (Data Modelling) โดยใช้แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล (Entity Relationship Diagram E-R Diagram) สรุปกิจกรรมในขั้นตอนที่ 3 ดังนี้

- 1) ศึกษาขั้นตอนการทำงานของระบบเดิม
- 2) รวบรวมความต้องการในระบบใหม่จากผู้ใช้
- 3) จำลองแบบความต้องการที่รวบรวมได้

2.1.2.4 การออกแบบเชิงตรรกะ (Logical Design) เป็นขั้นตอนในการออกแบบลักษณะการทำงานของระบบตามทางเลือกที่ได้ทำการเลือกไว้จากขั้นตอนการวิเคราะห์ระบบ โดยการออกแบบในเชิงตรรกะนี้ยังไม่ได้มีการระบุถึงคุณลักษณะของอุปกรณ์ที่จะนำมาใช้ เพียงแต่กำหนดถึงลักษณะของรูปแบบรายงานที่เกิดจากการทำงานของระบบ ลักษณะของการนำเข้าข้อมูลสู่ระบบแพลตฟอร์มที่ได้จากระบบ ซึ่งจะเลือกใช้การนำเสนอรูปแบบของรายงาน และลักษณะของจอภาพของระบบ PIS ด้วยต้นแบบ (Prototype) จะทำให้เข้าใจการทำงานของระบบได้ชัดเจนยิ่งขึ้น สรุปกิจกรรมในขั้นตอนที่ 4 ดังนี้

- 1) ออกแบบแบบฟอร์มและรายงาน (Form/Report Design)
- 2) ออกแบบส่วนติดต่อกับผู้ใช้ (User Interfaces Design)
- 3) ออกแบบฐานข้อมูลในระดับ Logical

2.1.2.5 การออกแบบเชิงกายภาพ (Physical Design) เป็นขั้นตอนที่ระบุถึงลักษณะการทำงานของระบบทางกายภาพหรือทางเทคนิคโดยระบุถึงลักษณะทางกายภาพหรือทางเทคนิคโดยระบุถึงลักษณะของอุปกรณ์ที่จะนำมาใช้เทคโนโลยีโปรแกรมภาษาที่จะนำมาเขียนโปรแกรมฐานข้อมูล ระบบปฏิบัติการ และระบบออกแบบ เพื่อส่งมอบให้กับโปรแกรมเมอร์เพื่อใช้เขียนโปรแกรมตามลักษณะการทำงานของระบบที่ได้ออกแบบและกำหนดไว้

2.1.2.6 การพัฒนาและติดตั้งระบบ (System Implementation) เป็นขั้นตอนในการนำข้อมูลเฉพาะของการออกแบบมาทำการเขียนโปรแกรม เพื่อให้เป็นไปตามคุณลักษณะและรูปแบบต่าง ๆ ที่ได้กำหนดไว้ หลังจากเขียนโปรแกรมที่พัฒนาขึ้นมา และสุดท้ายคือการ ติดตั้งระบบโดยทำการติดตั้งตัวโปรแกรม ติดตั้งอุปกรณ์ พร้อมทั้งจัดทำ คู่มือและจัดเตรียมหลักสูตรฝึกอบรมผู้ใช้งานที่เกี่ยวข้อง เพื่อให้ระบบใหม่สามารถใช้งานได้ สรุปกิจกรรมในขั้นตอนที่ 6 ดังนี้

- 1) เขียนโปรแกรม (Coding)
- 2) ทดสอบโปรแกรม (Testing)
- 3) ติดตั้งโปรแกรม (Installation)
- 4) จัดทำเอกสาร (Documentation)
- 5) จัดหลักสูตรฝึกอบรม (Training)
- 6) การบริการให้ความช่วยเหลือหลังการติดตั้งโปรแกรม (Support)

2.1.2.7 การซ่อมบำรุง (System Maintenance) เป็นขั้นตอนสุดท้ายของวงจรพัฒนาระบบ หลังจากระบบใหม่ได้เริ่มดำเนินการ ผู้ใช้ระบบอาจจะพบปัญหาที่เกิดขึ้นเนื่องจากความไม่คุ้นเคยกับระบบใหม่ และค้นพบวิธีการแก้ไขปัญหานั้นเพื่อให้ตรงกับความต้องการของผู้ใช้เอง ดังนั้นจะต้องมีทีมงานเพื่อคอยแก้ไขและเปลี่ยนแปลงระบบที่ทำการพัฒนาขึ้นจนกว่าจะเป็นที่พอใจของผู้ใช้งานมากที่สุด ปัญหาที่ผู้ใช้งานค้นพบระหว่างการดำเนินงานนั้นเป็นผลดีในการทำให้ระบบใหม่มีประสิทธิภาพและมีประสิทธิผลมากขึ้น เนื่องจากผู้ใช้งานเป็นผู้ที่เข้าใจในการทำงานเป็นอย่างดีโดยใน ขั้นตอนนี้ขึ้นอยู่กับว่าจะมีการจัดตั้งทีมงานอย่างไร เพื่อเป็นการประหยัดต้นทุนในส่วนนี้ให้ได้มากที่สุด สรุปกิจกรรมในขั้นตอนที่ 7 ดังนี้


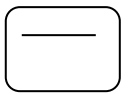
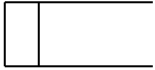

- 1) เก็บรวบรวมคำร้องขอให้ปรับปรุงระบบ
- 2) วิเคราะห์ข้อมูลร้องขอให้ปรับปรุงระบบ
- 3) ออกแบบการทำงานที่ต้องการปรับปรุง
- 4) ปรับปรุงระบบ

2.1.3 ส่วนประกอบของ DFD

เพื่อให้การเขียนแผนการไหลของข้อมูลเป็นมาตรฐานและมีแบบแผนที่ถูกต้องในที่นี้จะใช้สัญลักษณ์ในการเขียน DFD ตามทฤษฎีของ SSADM (Structure Systems Analysis and Design Method) ซึ่งมีองค์ประกอบ 4 สัญลักษณ์ ดังต่อไปนี้

2.1.4 เครื่องหมายที่ใช้ในการสร้าง DFD

ตารางที่ 2.1 เครื่องหมายที่ใช้ในการสร้าง DFD

Data Flow Diagram (DFD)	
ความหมาย	สัญลักษณ์
1. External Agent แสดง Entity ที่อยู่นอกระบบอาจหมายถึงตัวบุคคล องค์กร หรือระบบอื่นก็ได้	
2. Process แทนการประมวลผล	
3. Data Store แทนการเก็บข้อมูล หรือเพิ่มความหมายของข้อมูล	
4. Data Flow แสดงการไหลของข้อมูล	

2.1.5 ขั้นตอนการศึกษาการทำ DFD (Data Flow Diagram: DFD)

2.1.5.1 เขียน Context Diagram (Drawing a Context Diagram)

2.1.5.2 เขียน Process Hierarchy Chart (Developing Process Hierarchy Chart)

2.1.5.3 เขียน First – Level Data Flow Diagram (Developing First – Level Diagram)

2.1.5.4 เขียน DFD แสดงรายละเอียดของ Process ต่าง ๆ (Exploding Process into Greater Detail)

2.1.6 วิธีการสร้าง DFD

2.1.6.1 กำหนดสิ่งที่อยู่นอกระบบทั้งหมด และหาว่าข้อมูลอะไรบ้างที่เข้าสู่ระบบหรือออกจาก ขั้นตอนนี้จะทำให้ทราบว่าขอบเขตของระบบนั้นมีอะไรบ้าง

2.1.6.2 ใช้ข้อมูลที่ได้จากขั้นตอนที่ 1 นำมาสร้าง DFD ต่างระดับ

2.1.6.3 ขั้นตอนต่อมาจะมีอีก 4 ขั้นตอน โดยให้ทำทั้ง 4 ขั้นตอนนี้ซ้ำ ๆ จนกระทั่งได้ DFD ระดับต่ำสุด

- 1.) เขียน DFD ฉบับแรก กำหนดโพรเซสและข้อมูลที่ไหลเข้าออกจากโพรเซส
- 2.) เขียน DFD อื่น ๆ ที่เป็นไปได้จนกระทั่ง DFD ที่ถูกต้องที่สุด
- 3.) พยายามหาว่ามีข้อผิดพลาดอะไรหรือไม่
- 4.) เขียนแผนภาพแต่ละภาพอย่างดี

2.1.6.4 นำแผนภาพทั้งหมดที่เขียนมาแล้วเรียงลำดับ ทำสำเนา แล้วพร้อมที่จะนำไปตรวจสอบ ข้อผิดพลาดกับผู้ร่วมทีมงาน ถ้ามีแผนภาพใดที่มีจุดอ่อนให้กลับไปเริ่มต้นใหม่ที่ขั้นตอนที่ 2.1.6.3

2.1.6.5 นำ DFD ที่ได้ไปตรวจสอบข้อผิดพลาดกับผู้ใช้ระบบเพื่อหาว่ามีแผนภาพใดที่ไม่ถูกต้อง

2.1.6.6 ผลิตแผนภาพฉบับสุดท้ายทั้งหมด

2.2 ระบบฐานข้อมูล (Database System)

ฐานข้อมูล หมายถึง ชุดของข้อมูลที่มีความสัมพันธ์กัน ที่ถูกนำมาจัดเก็บไว้ด้วยกันเพื่อให้สามารถใช้ข้อมูลเหล่านั้นร่วมกันได้อย่างรวดเร็วและมีประสิทธิภาพ การจัดเก็บข้อมูลจะมีประสิทธิภาพได้ก็ต่อเมื่อมีวิธีการจัดการข้อมูลที่ดี กล่าวคือ วิธีการจัดเก็บและค้นคืนข้อมูล ต้องเป็นไปอย่างมีประสิทธิภาพและรวดเร็ว

2.2.1 ระบบจัดการฐานข้อมูล (Database Management Systems : DBMS)

ระบบจัดการฐานข้อมูล หมายถึง ซอฟต์แวร์ของระบบที่ใช้ในการจัดการฐานข้อมูลโดยมีวัตถุประสงค์หลักคือ การสร้างสภาวะแวดล้อมที่สะดวกและมีประสิทธิภาพในการเข้าถึงและจัดเก็บข้อมูลของฐานข้อมูล ระบบจัดการฐานข้อมูลจะทำหน้าที่ในการแปลความต้องการของผู้ใช้ให้อยู่ในรูปแบบที่สามารถทำงานได้รับฐานข้อมูลตามวัตถุประสงค์ของผู้ใช้โดยระบบจัดการฐานข้อมูล (Database Management Systems: DBMS) จะทำหน้าที่ช่วยในการจัดเก็บ, การเข้าถึงข้อมูล และการควบคุมต่าง ๆ ทำให้ง่ายต่อการกำจัดปัญหาความไม่สอดคล้องกันของข้อมูล, ความผิดปกติของข้อมูล และการพึ่งพิงข้อมูล

2.2.2 หน้าที่ของระบบจัดการฐานข้อมูล

2.2.2.1 การจัดการพจนานุกรมข้อมูล ระบบจัดการฐานข้อมูลจะทำการจัดเก็บนิยามของข้อมูล และความสัมพันธ์ระหว่างข้อมูลไว้ในพจนานุกรมข้อมูล เป็นสารสนเทศที่บอกเกี่ยวกับโครงสร้างของฐานข้อมูล โปรแกรมประยุกต์ทั้งหมดที่ต้องการเข้าถึงข้อมูลในฐานข้อมูลจะต้องทำงานผ่านระบบจัดการฐานข้อมูล โดยที่ระบบจัดการฐานข้อมูลจะใช้พจนานุกรมข้อมูลเพื่อค้นหาโครงสร้างตลอดจน ส่วนประกอบของข้อมูลและความสัมพันธ์ที่ต้องการ นอกจากนั้นแล้ว การเปลี่ยนแปลงใด ๆ ที่มีต่อโครงสร้างฐานข้อมูลจะถูกบันทึกไว้โดยอัตโนมัติในพจนานุกรมข้อมูล ทำให้เราไม่ต้องไปเปลี่ยนแปลงแก้ไขโปรแกรมเมื่อ โครงสร้างข้อมูลมีการเปลี่ยนแปลง

2.2.2.3 การแปลงและนำเสนอข้อมูล ระบบจัดการฐานข้อมูลจะทำหน้าที่ในการแปลงข้อมูลที่ได้รับเข้ามา เพื่อให้สอดคล้องกับ โครงสร้างในการจัดเก็บข้อมูล ทำให้เราไม่ต้องไปยุ่งเกี่ยวกับความแตกต่างระหว่างรูปแบบของข้อมูลทางตรรกะและทางกายภาพ กว้างคือทำให้มีความเป็นอิสระของข้อมูล ระบบจัดการฐานข้อมูลจะแปลงความต้องการเชิงตรรกะของผู้ใช้ให้เป็นคำสั่งที่สามารถดึงข้อมูลทางกายภาพที่ต้องการ

2.2.2.4 การจัดการระบบความมั่นคง ระบบจัดการฐานข้อมูลจะสร้างระบบรักษาความมั่นคง โดยการกำหนดรายชื่อผู้มีสิทธิ์เข้าใช้ระบบ และความสามารถในการใช้ระบบ เช่น การ อ่าน เพิ่ม ลบ หรือแก้ไขเปลี่ยนแปลงข้อมูลการจัดการระบบความมั่นคงมีความสำคัญมากในระบบฐานข้อมูลแบบที่มีผู้ใช้หลายคน

2.2.2.5 การควบคุมการเข้าถึงข้อมูลของผู้ใช้หลายคน ระบบจัดการฐานข้อมูลจะใช้อัลกอริทึมที่เหมาะสม เพื่อให้แน่ใจว่าผู้ใช้หลายคนสามารถเข้าใช้ฐานข้อมูลในภาวะพร้อมกันและยังคงความถูกต้องของข้อมูลในฐานข้อมูลไว้ได้

2.2.2.6 การเก็บสำรองและกู้คืนข้อมูล ระบบจัดการฐานข้อมูลจะมีโปรแกรมเพื่อสนับสนุนการสำรองและกู้คืนข้อมูล เพื่อให้แน่ใจในความปลอดภัยและความมั่นคงของข้อมูลในระบบ ระบบจัดการฐานข้อมูลจะทำการกู้คืนข้อมูลในฐานข้อมูลคืนมาหลังจากระบบเกิดความล้มเหลว (FAILURE)

2.2.2.7 การควบคุมความถูกต้องของข้อมูล ระบบจัดการฐานข้อมูลจะสนับสนุนและควบคุมความถูกต้องของข้อมูล ตั้งแต่การลดความ ซ้ำซ้อนข้อมูล ไปจนถึงความไม่สอดคล้องกันของข้อมูล ความสัมพันธ์ของข้อมูลที่เก็บในพจนานุกรมข้อมูลจะถูกนำมาใช้ในการควบคุมความถูกต้องของข้อมูลด้วย

ข้อดีและข้อเสียในการใช้ระบบฐานข้อมูล

1.) ข้อดี

- 1) ทำให้ข้อมูลสามารถใช้ร่วมกันได้
- 2) สามารถควบคุมเรื่องความปลอดภัยของข้อมูลได้ดี
- 3) การพัฒนาและบำรุงรักษา สามารถกำหนดให้เป็นไปตามมาตรฐาน
- 4) ลดความซ้ำซ้อนของข้อมูล
- 5) สามารถหลีกเลี่ยงความไม่สอดคล้องกันของข้อมูล
- 6) สามารถรักษาความถูกต้องของข้อมูลในระบบ
- 7) มีความเป็นอิสระของข้อมูล
- 8) มีทฤษฎีที่สนับสนุนการทำงานที่ชัดเจน โดยเฉพาะอย่างยิ่งในระบบฐานข้อมูลเชิงสัมพันธ์

1.) ข้อเสีย

- 1) ระบบมีความซับซ้อน
- 2) มีค่าใช้จ่ายสูงขึ้น
- 3) ต้องใช้ซอฟต์แวร์และฮาร์ดแวร์เพิ่มเติม
- 4) หากระบบเกิดความล้มเหลว จะทำให้มีผลกระทบกับการทำงานขององค์กรอย่างกว้างขวาง
- 5) การกู้คืนทำได้ยาก ต้องใช้เครื่องมือต่าง ๆ มากขึ้น

2.2.3 การออกแบบฐานข้อมูลด้วย E-R โมเดล

จุดประสงค์ของการออกแบบฐานข้อมูล คือ การสร้างฐานข้อมูลที่มีประสิทธิภาพในการใช้งานสำหรับผู้ใช้ การออกแบบฐานข้อมูลจึงต้องทำอย่างมีขั้นตอน ขั้นตอนการออกแบบฐานข้อมูล เริ่มจากการศึกษา วิเคราะห์ระบบงานเดิมและปัญหาที่เกิดขึ้นในระบบนั้น รวมทั้งรวบรวมความต้องการของผู้ใช้และกฎเกณฑ์ข้อบังคับต่าง ๆ สูตรต่าง ๆ นำมาประมวลเข้าด้วยกัน เพื่อดำเนินการออกแบบฐานข้อมูลในระดับแนวคิด ซึ่งการออกแบบในระดับแนวคิดสามารถทำเป็นขั้นตอนตามลำดับ ได้แก่

2.2.3.1 สร้างเอนทิตี

2.2.3.2 ปรับริเลชันต่าง ๆ ให้อยู่ในรูปแบบบรรทัดฐาน

2.2.3.3 ระบุคีย์ที่ต้องใช้ในแต่ละริเลชัน

2.2.3.4 ระบุกฎเกณฑ์ข้อจำกัด ที่ต้องการคำนึงถึง

2.2.3.5 นำผลที่ได้จากข้อ 4 ขั้นตอนแรกมารวบรวมเข้าด้วยกัน

2.2.4 ศัพท์ที่ใช้ในระบบฐานข้อมูล


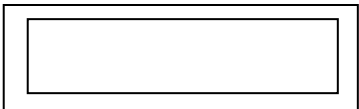
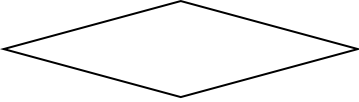
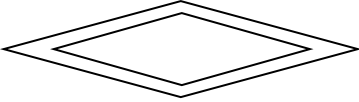


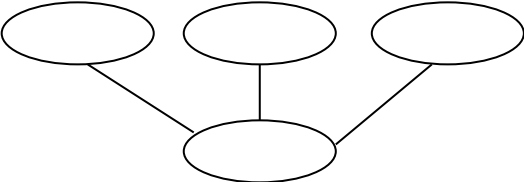
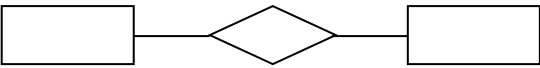
2.2.4.1 เอนทิตี (Entity) หมายถึง ชื่อของสิ่งหนึ่งสิ่งใดเปรียบเสมือนคำนาม ได้แก่ บุคคลสถานที่สิ่งของ เช่น นักศึกษา อาจารย์ ภาควิชา ฯลฯ

2.2.4.2 แอททริบิวต์ (Attribute) หมายถึง รายละเอียดของข้อมูลใน Entity หนึ่ง ซึ่งเป็นข้อมูลที่แสดงลักษณะและคุณสมบัติของ Entity เช่น Entity ของนักศึกษา ได้แก่ ชื่อ อายุ เพศ ฯลฯ

2.2.4.3 ความสัมพันธ์ (Relationship) หมายถึง คำกริยาที่แสดงความสัมพันธ์ระหว่าง Entity เช่น ความสัมพันธ์ระหว่างหลักสูตรวิชาและนักศึกษา ก็เป็นในลักษณะหลักสูตรวิชาที่นักศึกษานั้น ๆ เรียนอยู่ ส่วนความสัมพันธ์ระหว่างนักศึกษากับหลักสูตรวิชา ก็เป็นในลักษณะที่ว่า นักศึกษาเรียนในหลักสูตรวิชานั้น ๆ

2.2.4.4 สัญลักษณ์สำหรับการสร้างโมเดลแบบ E – R

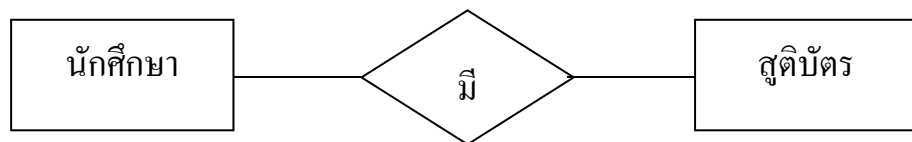
ตารางที่ 2.2 สัญลักษณ์สำหรับการสร้างโมเดลแบบ E – R

เครื่องหมาย	ความหมาย
	เอนทิตี
	เอนทิตีชนิดอ่อน
	ประเภทของความสัมพันธ์ระหว่างเอนทิตี
	ประเภทของความสัมพันธ์ระหว่างเอนทิตีหนึ่งเอนทิตีชนิดอ่อนแอ (WEAK ENTITY)
	แอททริบิวต์
	แอททริบิวต์ที่เป็นคีย์หลัก
	แอททริบิวต์ผสม
	การแสดงความสัมพันธ์ระหว่างเอนทิตี

2.2.6 ประเภทของความสัมพันธ์ระหว่างเอนทิตี

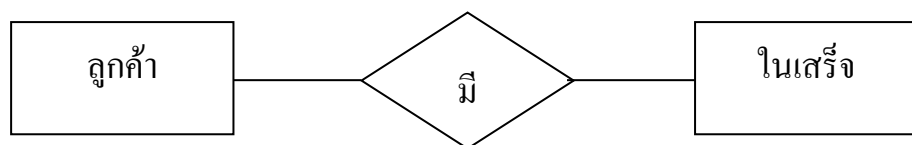
ความสัมพันธ์ระหว่างเอนทิตี เป็นความสัมพันธ์ที่สมาชิกของเอนทิตีหนึ่งสัมพันธ์กับสมาชิกของอีกเอนทิตีหนึ่ง ซึ่งสามารถแบ่งประเภทของความสัมพันธ์ออกเป็น 3 ประเภท อันได้แก่ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One to One) แบบหนึ่งต่อกลุ่ม (One to Many) แบบกลุ่มต่อกลุ่ม (Many to Many)

2.2.6.1 ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One to One) ความสัมพันธ์แบบ (๑ ถึง ๑) ตัวอย่าง เช่น ความสัมพันธ์ระหว่างเอนทิตีนักศึกษา กับ สูติบัตร จะมีความสัมพันธ์เป็นดังนี้ นักศึกษาหนึ่งคนจะมีสูติบัตรได้เพียงใบเดียวเท่านั้น และสูติบัตรหนึ่งใบก็เป็นของนักศึกษาได้เพียงคนเดียวเท่านั้นเช่นกันความสัมพันธ์แบบ วัน-ทู-วัน จะแทนด้วยสัญลักษณ์ “1:1”



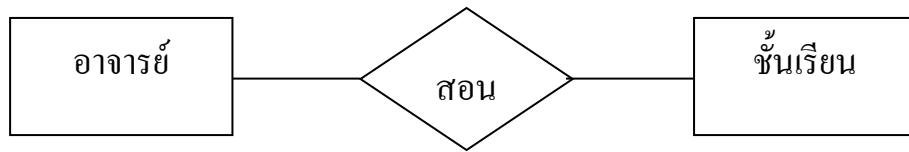
รูปที่ 2.2 ความสัมพันธ์แบบหนึ่งต่อหนึ่ง

2.2.6.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One to Many) ความสัมพันธ์แบบ (One to Many) ตัวอย่างเช่น ความสัมพันธ์ระหว่างเอนทิตีลูกค้ากับใบเสร็จ จะมีความสัมพันธ์เป็นดังนี้ คือ ลูกค้าคนหนึ่งอาจมีใบเสร็จได้หลายใบ เนื่องจากลูกค้าหนึ่งคนอาจมาซื้อสินค้าหลายครั้ง แต่ใบเสร็จหนึ่งใบต้องเป็นของลูกค้าเพียงคนเดียวเท่านั้น ความสัมพันธ์แบบวัน-ทู-แมนนี่ จะเขียนแทนด้วยสัญลักษณ์ “1:M”



รูปที่ 2.2 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม

2.2.6.3 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many to Many) ความสัมพันธ์แบบ (Many to Many) ตัวอย่าง เช่น ความสัมพันธ์ระหว่างเอนทิตีอาจารย์กับ ชั้นเรียน จะมีความสัมพันธ์เป็นดังนี้ คือ อาจารย์หนึ่งคนสอนได้หลายชั้นเรียน และหนึ่งชั้นเรียนถูกสอนโดยอาจารย์หลายคน คือ เป็นชั้นเรียนที่มีอาจารย์ร่วมกันสอนหลายคน ความสัมพันธ์แบบ แมนนี่-ทู-แมนนี่ จะแทนด้วยสัญลักษณ์ “M:N”



รูปที่ 2.2 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม

2.2.7 ขั้นตอนการออกแบบฐานข้อมูล

การที่จะนำฐานข้อมูลไปใช้งานได้อย่างมีประสิทธิภาพหรือไม่นั้น หัวใจสำคัญอยู่ที่การออกแบบฐานข้อมูลโดยผ่านการออกแบบจะต้องสามารถวิเคราะห์หาเอนทิตีและแอตทริบิวต์ทั้งหมดที่จะต้องนำมาใช้ในฐานข้อมูลได้อย่างครบถ้วน รวมทั้งกำหนดความสัมพันธ์ระหว่างเอนทิตีได้อย่างถูกต้องและเหมาะสม โดยขั้นตอนการออกแบบฐานข้อมูล มีดังนี้

2.2.7.1 รวบรวมข้อมูลคือการออกแบบระบบจะต้องเข้าใจระบบที่จะพัฒนาต้องเข้าใจว่าจะใช้ข้อมูลอะไรบ้าง ผลลัพธ์ที่ได้ออกมาคืออะไร โดยการรวบรวมรายละเอียดข้อมูลของระบบ

2.2.7.2 กำหนดโครงสร้างของเทเบิล คือ การวางรูปแบบระบบฐานข้อมูล ว่ามีเทเบิลอะไรบ้าง ในแต่ละเทเบิลจะมีฟิลด์ใดบ้างฟิลด์ใดเป็นคีย์หลัก ทำนอร์มอลไลเซชัน (Normalization) เพื่อลดความซ้ำซ้อนของข้อมูล

2.2.7.3 สร้างฐานข้อมูล คือ เป็นการวางรูปแบบของระบบที่เกี่ยวข้องกับ โครงสร้างการจัดเก็บข้อมูล

2.2.7.4 พจนานุกรมข้อมูล (Data Dictionary) เมื่อทำการออกแบบฐานข้อมูลแล้วนั้น จะเห็นได้ว่าจะมีข้อมูลจำนวนมากจึงเป็นการ ยากที่จะเข้าใจและนำไปใช้งานในการเขียนโปรแกรม สิ่งที่จะทำให้รู้ว่ามีข้อมูลอะไรบ้างในฐานข้อมูล คือ พจนานุกรมข้อมูล (Data Dictionary) นั่นเอง พจนานุกรมจะแสดงถึงรายละเอียดต่าง ๆ ของ ข้อมูลที่ใช้งานรวมทั้งโครงสร้างข้อมูล

2.2.8 นอร์มอลไลเซชัน (Normalization) จากขั้นตอนการออกแบบฐานข้อมูล เมื่อรวบรวมข้อมูลมาแล้วจะเห็นได้ว่าจะมีข้อมูลบางส่วนมีความซ้ำซ้อนกันไม่เหมาะกับการนำมาใช้งานในระบบฐานข้อมูล จึงจำเป็นต้องมีวิธีการลดความซ้ำซ้อนเหล่านี้ วิธีการนั้นก็คือ วิธีการนอร์มอลไลเซชัน (Normalization) เพื่อเป็นการปรับโครงสร้างข้อมูล ลดความซ้ำซ้อนของข้อมูล ให้ง่ายต่อการใช้งานและมีปัญหาน้อยที่สุด

2.2.8.1 การนอร์มอล ระดับที่ 1 (First Normal Form: 1NF) คือ เทเบิลที่จะอยู่ในรูป 1NF ได้นั้นเทเบิลนั้นจะต้องไม่มีฟิลด์ใดซ้ำกันในเรคอร์ดเดียวกัน

2.2.8.2 การนอร์มอล ระดับที่ 2 (Second Normal Form: 2NF) คือ เทเบิลที่จะอยู่ในรูป 2NF ได้นั้น จะต้องเป็น 1NF มาก่อน และทุกฟิลด์ที่ไม่ใช่คีย์หลักจะต้องขึ้นตรงกับคีย์หลักอย่างเดียวนั้น ไม่ขึ้นตรงเพียงบางส่วนของคีย์หลัก

2.2.8.3 การนอร์มอล ระดับที่ 3 (Third Normal Form: 3NF) คือ เทเบิลที่จะอยู่ในรูป 3NF ได้นั้น จะต้องเป็น 2NF มาก่อนและทุกฟิลด์ที่ไม่ใช่คีย์หลักจะต้องไม่ขึ้นตรงต่อตนเอง

2.2.8.4 การนอร์มอล ระดับที่ 4 (Third Normal Form: 4NF) คือ รีเลชันนั้นมีคุณสมบัติอยู่ในรูปแบบที่เป็นบรรทัดฐานของบอยส์และคอตต์ แล้วรีเลชันที่ไม่มีความสัมพันธ์ระหว่าง แอททริบิวต์แบบหลายค่า

2.2.8.5 การนอร์มอล ระดับที่ 5 (Third Normal Form: 5NF) คือ รีเลชันนั้น ๆ มีความสัมพันธ์ระหว่างแอททริบิวต์แบบ Join และรีเลชันย่อย ๆ ที่จำแนกออกมาต้องมีคีย์คู่แข่งของรีเลชันเดิมอยู่ด้วยเสมอ หรือ รีเลชันนั้น ๆ ไม่มีความสัมพันธ์ระหว่างแอททริบิวต์แบบ Join

2.2.9 ข้อควรคำนึงในการทำให้เป็นรูปแบบที่เป็นบรรทัดฐาน

การทำให้เป็นรูปแบบที่เป็นบรรทัดฐานเป็นกระบวนการเพื่อลดความซ้ำซ้อนของข้อมูล และแก้ไขปัญหาก็เกี่ยวกับการปรับปรุง การเพิ่ม หรือการลบข้อมูล ซึ่งกระบวนการทำให้เป็นรูปแบบที่เป็นบรรทัดฐานจะใช้เทคนิควิธีที่เรียกว่า เทคนิค Decomposition จำแนกรีเลชันเดิมออกเป็นรีเลชันย่อยโดยคำนึงถึงหลักการพื้นฐานว่ารีเลชันย่อยที่จำแนกออกมานั้นเมื่อนำมารวมเข้าด้วยกันจะต้องไม่มีข้อมูลสูญหายไปหรือเกินนารีเลชันเดิมเนื่องจากการทำให้เป็นรูปแบบที่เป็นบรรทัดฐานเป็นเทคนิคในการตรวจสอบและวิเคราะห์ความสัมพันธ์ระหว่างแอททริบิวต์ภายในรีเลชันนั้น โดยไม่อาจทำการตรวจสอบความสัมพันธ์ระหว่างแอททริบิวต์ที่อยู่ร่างรีเลชันได้ ดังนั้นสิ่งที่ควรคำนึงในการทำให้เป็นรูปแบบที่เป็นบรรทัดฐาน การจำแนกรีเลชันย่อยมากเกินไป (Over normalization) และ Demoralization

2.2.9.1 การจำแนกรีเลชันย่อยมากเกินไป การจำแนกรีเลชันย่อยมากเกินไป หมายถึง การจำแนกรีเลชันย่อยมากเกินไป หมายถึงการจำแนกรีเลชันออกเป็นรีเลชันย่อยมากเกินไปกว่า ความจำเป็น โดยทั่วไปแล้วในการออกแบบฐานข้อมูลระดับแนวคิด ผู้ออกแบบจะพยายามทำการวิเคราะห์ให้ รีเลชันมีคุณสมบัติอยู่ในรูปแบบที่เป็นบรรทัดฐานขั้นที่ 3 หรือรูปแบบที่เป็นบรรทัดฐานของบอยส์และคอตต์ ซึ่งอาจมีบ้างใน บางกรณีที่ต้องดำเนินการต่อไปจนกระทั่งรีเลชันมีคุณสมบัติอยู่ในรูปแบบที่เป็นบรรทัดฐานขั้นที่ 4 หรือขั้นที่ 5 ซึ่งในทางปฏิบัติอาจเกิดขึ้นได้น้อยกว่ามากหากทว่าในบางครั้งรีเลชันใด ๆ แม้จะสามารถทำการจำแนกออกเป็นรีเล

ชั้นย่อยได้ แต่ก็ควรคงรูปแบบรีเลชันนั้นไว้ในลักษณะเช่นเดิม เนื่องจากการจำแนกรีเลชันออกเป็นรีเลชันย่อยมากเกินไปจนอาจก่อให้เกิดผลเสียต่อประสิทธิภาพในการทำงานของฐานข้อมูลนั้น เช่น อาจทำให้ใช้เสียเวลาเพื่อการค้นหาข้อมูลมากขึ้น เป็นต้น

2.2.9.2 การ Demoralization แม้ว่าวัตถุประสงค์หลักของกระบวนการทำให้เป็นรูปแบบที่เป็นบรรทัดฐานตั้งแต่ขั้นตอนที่ 1 มาจนถึงขั้นตอนที่ 5 คือการแก้ไขปัญหาคำซ้ำซ้อนของข้อมูลที่ปรากฏในรีเลชันให้เหลือน้อยที่สุด ตลอดจนลดปัญหาในเรื่องเกี่ยวกับการปรับปรุงการเพิ่ม หรือการลบก็ตามข้อมูลก็ตามหาทว่าในบางกรณี การจำแนกรีเลชันออกเป็นรีเลชันย่อยอาจก่อให้เกิดผลกระทบต่อประสิทธิภาพในการค้นหาหรือดำเนินการกับข้อมูลได้ เพราะการเสียเวลาเพื่อเชื่อมโยงข้อมูลจากรีเลชันต่าง ๆ เข้าด้วยกัน อาจทำให้กระประมวลผลล่าช้า และเกิดความเสียหายได้ ดังนั้น ผู้ออกแบบฐานข้อมูลจึงจำเป็นต้องทำการ Demoralization โดยคำนึงความสำคัญในเรื่องประสิทธิภาพของการค้นหาข้อมูลมากกว่าปัญหาการซ้ำซ้อนของข้อมูลเมื่อทำการปรับปรุง เพิ่มเติม