บทที่ 3

แนวคิดและทฤษฎี

3.1 สารสนเทศ (Information)

ข้อมูล (Data) คือข้อเท็จจริงขั้นต้นซึ่งอาจเรียกว่าเป็นวัตถุดิบของสารสนเทศ (Information) เมื่อข้อมูลถูกนำมาประมวลผล (เรียงลำดับ แยกประเภท เชื่อมโยง คำนวณ หรือ สรุปผล) และจัดให้อยู่ในรูปแบบที่นำไปใช้ประโยชน์ได้ เราจึงเรียกว่าเป็นสารสนเทศ ลักษณะของสารสนเทศที่ดีอาจจำแนกได้เป็น 5 ลักษณะ ดังนี้

3.1.1 **เป็นปัจจุบัน** (Current) ข้อมูลอาจมีการปรับเปลี่ยนไปได้เรื่อยๆ ตามกาลเวลา ข้อมูลที่ตรงตามความเป็นจริงในปัจจุบัน จะมีค่ามากกว่าข้อมูลที่เป็นอดีตไปแล้ว ระบบสารสนเทศที่ดีจะต้องสามารถยืดหยุ่น ให้มีการปรับเปลี่ยนค่าให้เป็นปัจจุบัน และ/หรือ คงค่าเก่าเก็บไว้เพื่อประโยชน์การใช้งานต่างๆ กัน

3.1.2 **ทันเวลา** (Timely) สารสนเทศมีคุณค่าทางเวลาเข้ามาเกี่ยวข้อง ถ้าไม่ได้สารสนเทศในเวลาที่ต้องการ อาจเกิดการสูญเสียโอกาสและเวลาของการดำเนินงาน ระบบสารสนเทศที่มีประสิทธิภาพคือ ระบบที่จะต้องจัดสรรให้ได้สารสนเทศเมื่อผู้ใช้ต้องการ

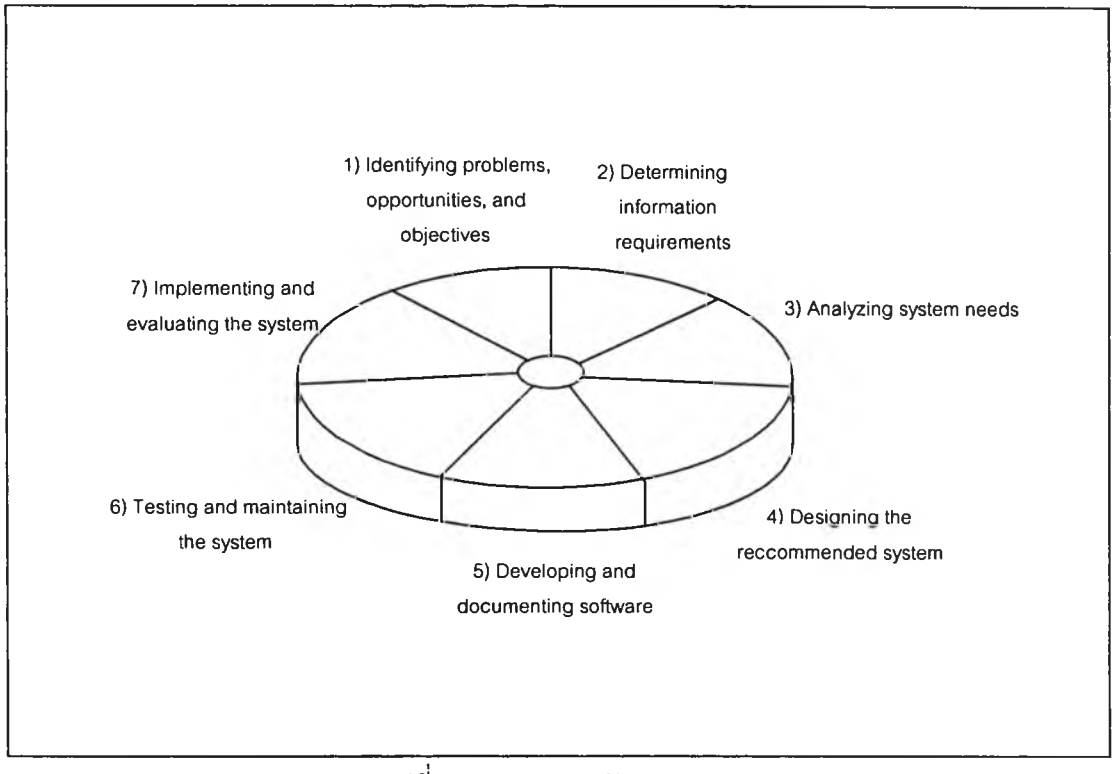
3.1.3 **มีค่าเที่ยงตรง** (Relevant) ผู้ใช้ต้องการสารสนเทศที่ตรงกับงานที่รับผิดชอบ ถ้าผู้ใช้ได้ข้อมูลที่ไม่สมบูรณ์ หรือมีรายละเอียดปลีกย่อยมากเกินไป ผู้ใช้ก็จะทำงานในส่วนของตนได้ไม่เต็มที่ การที่ผู้ใช้แต่ละคนได้รับสารสนเทศที่ตรงกับความต้องการของตนเองมากเท่าใดระบบสารสนเทศนั้นก็จะถูกจัดว่าเป็นระบบที่มีประสิทธิภาพยิ่งขึ้น

3.1.4 **มีความคงที่** (Consistent) ในหลายๆ กรณี สารสนเทศเองก่อให้เกิดความขัดแย้งข้อมูลที่จัดเก็บในหลายๆ ที่อาจไม่ตรงกัน วิธีการประมวลผลที่ต่างกันอาจทำให้เกิดความคลาดเคลื่อนขึ้นในผลลัพธ์ที่ได้ จุดมุ่งหมายหลักของระบบสารสนเทศข้อหนึ่งคือ พยายามทำให้เกิดข้อขัดแย้งน้อยที่สุด ข้อมูลมีความคงที่มากที่สุดเท่าที่จะทำได้

3.1.5 **นำเสนอรูปแบบที่มีประโยชน์** (Presented in Usable Form) ถึงแม้ว่าระบบจะมีลักษณะทั้ง 4 ประการข้างต้น แต่ถ้านำเสนอในรูปแบบที่ผู้ใช้นำไปใช้ประโยชน์ไม่ได้ ระบบดังกล่าวก็จะมีค่าน้อยเต็มที ระบบสารสนเทศที่มีประสิทธิภาพ คือระบบที่มีความยืดหยุ่นในการนำเสนอสารสนเทศให้กับผู้ที่ต้องการใช้สารสนเทศนั้นๆ

3.2 วงจรการพัฒนาระบบ (System Development Life Cycle : SDLC)

วงจรการพัฒนาระบบ (System Development Life Cycle : SDLC) คือกระบวนการในการพัฒนาระบบซึ่งประกอบด้วยขั้นตอนต่างๆ ในการดำเนินการวิเคราะห์และออกแบบระบบ ขั้นตอนต่างๆ ของวงจรการพัฒนาระบบแบ่งออกเป็น 7 ขั้นตอน ดังรูป 3.1



รูปที่ 3.1 วงจรการพัฒนาระบบ

3.2.1 **กำหนดปัญหา โอกาสและวัตถุประสงค์ของการพัฒนาระบบ** (Identifying Problems, Opportunities, and Objectives)

เป็นขั้นตอนในการกำหนดปัญหา วัตถุประสงค์ และโอกาสของการพัฒนาระบบ ซึ่งขั้นตอนนี้เป็นขั้นตอนที่มีความสำคัญต่อความสำเร็จของส่วนที่เหลือของโครงการ เนื่องจากการกำหนดปัญหาที่ผิดพลาดจะทำให้เสียเวลา ในขั้นตอนนี้จะเริ่มด้วยการมองภาพความเป็นไปของระบบอย่างซื่อสัตย์ จากนั้นทำการกำหนดปัญหาให้ชัดเจน

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

โอกาสของการพัฒนาระบบ (Opportunities) หมายถึงสถานการณ์ที่นักวิเคราะห์ระบบเชื่อว่าสามารถปรับปรุงให้ดีขึ้นได้โดยการใช้ระบบสารสนเทศ การใช้โอกาสนี้เพื่อพัฒนาระบบขึ้นอาจทำให้ระบบการดำเนินงานขององค์กรมีประสิทธิภาพมากขึ้น

การกำหนดวัตถุประสงค์ก็เป็นส่วนสำคัญในขั้นตอนนี้ โดยนักวิเคราะห์ระบบจะต้องค้นหางานที่ระบบต้องการทำ ซึ่งจะทำให้เห็นว่าระบบสารสนเทศสามารถช่วยให้การดำเนินงานขององค์กรบรรลุวัตถุประสงค์ได้อย่างไร

3.2.2 **กำหนดความต้องการของผู้ใช้ระบบ** (Determining Information Requirements)

ในขั้นตอนนี้นักวิเคราะห์ระบบจะกำหนดความต้องการของผู้ใช้แต่ละคนที่เกี่ยวข้อง ซึ่งสามารถใช้เครื่องมือต่างๆ ในการพิจารณา รวมไปถึงการสุ่มตัวอย่างและสำรวจข้อมูล, การสัมภาษณ์และการออกแบบสอบถาม, การสำรวจลักษณะการตัดสินใจและสภาพแวดล้อมของสถานที่ทำงาน และการทำต้นแบบ

3.2.3 **วิเคราะห์ความต้องการของระบบ** (Analyzing System Needs)

ในขั้นตอนนี้นักวิเคราะห์ระบบจะทำการวิเคราะห์ความต้องการของระบบ ซึ่งมีเครื่องมือและเทคนิคต่างๆ ที่ช่วยในการกำหนดความต้องการของระบบ เช่น แผนภาพกระแสข้อมูล (Data Flow Diagram) เป็นต้น นอกจากนี้จะต้องทำการวิเคราะห์โครงสร้างของการตัดสินใจ เพื่อกำหนดเงื่อนไข ทางเลือกและการกระทำของการตัดสินใจ ซึ่งมีวิธีวิเคราะห์ 3 วิธีคือ Structured English, Decision Table และ Decision Tree

3.2.4 **ออกแบบระบบงาน** (Designing the Recommended System)

ในขั้นตอนนี้นักวิเคราะห์ระบบจะใช้ข้อมูลที่ได้เก็บรวบรวมมาเพื่อออกแบบวิธีการและรายละเอียดต่างๆ ของระบบสารสนเทศให้สอดคล้องกับความต้องการของผู้ใช้โดยจะไม่คำนึงถึงเครื่องคอมพิวเตอร์และรูปแบบการประมวลผลข้อมูล โดยจะออกแบบกระบวนการนำเข้าข้อมูลและข้อมูลนำเข้าของระบบสารสนเทศด้วยเทคนิคของการออกแบบรูปแบบและหน้าจอรับข้อมูล

ส่วนหนึ่งของการออกแบบในขั้นตอนนี้คือการออกแบบตัวประสานผู้ใช้ ซึ่งทำหน้าที่ติดต่อระหว่างผู้ใช้กับระบบ ตัวอย่างเช่น การใช้แป้นพิมพ์เพื่อพิมพ์คำถามและคำตอบ, การใช้เมนูบนหน้าจอเพื่อเลือกทำคำสั่ง, การใช้เมาส์ และอื่นๆ การออกแบบในขั้นตอนนี้ยังรวมถึงการออกแบบแฟ้มข้อมูลหรือฐานข้อมูลเพื่อเก็บที่จำเป็นของระบบ และการออกแบบการนำข้อมูลออก (ทางหน้าจอ หรือ การพิมพ์) ให้เหมาะสมกับความต้องการของผู้ใช้

3.2.5 **พัฒนาซอฟต์แวร์และจัดทำเอกสาร** (Developing and Documenting Software)

ในขั้นตอนนี้นักวิเคราะห์ระบบจะทำการออกแบบและพัฒนาซอฟต์แวร์ที่จำเป็นในระบบ ซึ่งมีการใช้เทคนิคในการออกแบบซอฟต์แวร์และการจัดทำเอกสาร เช่นวิธี HIPO, Flowchart เป็นต้น

3.2.6 **ทดสอบระบบงาน** (Testing and Maintaining the System)

ในขั้นตอนนี้จะทำการทดสอบระบบก่อนนำไปใช้จริง เริ่มทำการบำรุงรักษาและจัดทำเอกสารของระบบ

3.2.7 **ติดตั้งระบบงาน** (Implementing and Evaluating the System)

ในขั้นตอนสุดท้ายจะทำการติดตั้งระบบ การฝึกอบรมการใช้ระบบให้กับผู้ใช้งานเพื่อให้ผู้ใช้ระบบสามารถใช้งานได้อย่างมีประสิทธิภาพ

3.3 แผนภาพการเคลื่อนไหวของข้อมูล (Data Flow Diagram : DFD)

ในการวิจัยครั้งนี้ผู้วิจัยได้เลือกใช้แผนภาพการเคลื่อนไหวของข้อมูลเป็นเครื่องมือในหารวิเคราะห์และออกแบบระบบงานแบบโครงสร้าง โดยใช้สัญลักษณ์ทั้ง 4 แบบ ดังแสดงในตารางที่ 3.1 มาประกอบเข้าด้วยกันอย่างมีกฎเกณฑ์ เพื่อแสดงการเคลื่อนไหวของข้อมูล แผนภาพการเคลื่อนไหวของข้อมูลมีข้อดีหลายประการดังนี้

3.3.1 ให้ความอิสระต่อนักวิเคราะห์ระบบที่จะทำการออกแบบระบบงาน โดยไม่ต้องผูกติดกับวิธีการติดตั้งระบบงานและข้อจำกัดต่าง ๆ ของระบบมากเกินไป

3.3.2 ช่วยให้ผู้วิเคราะห์เข้าใจในระบบงานและความสัมพันธ์ระหว่างระบบงานย่อย ๆ ทั้งยังสามารถแบ่งแยกขอบเขตของระบบงานออกจากสภาวะแวดล้อมได้อย่างชัดเจน

3.3.3 ใช้เป็นเครื่องมือในการสื่อสารระหว่างผู้วิเคราะห์กับผู้ใช้ระบบงาน ทำให้ผู้ใช้มองเห็นภาพการทำงานของระบบได้อย่างชัดเจน จึงสามารถพบข้อผิดพลาดที่อาจเกิดขึ้นหรือให้ข้อคิดแนะนำต่าง ๆ เพื่อให้ผู้วิเคราะห์ระบบสามารถปรับปรุงแก้ไขได้ดียิ่งขึ้น

ตารางที่ 3.1 สัญลักษณ์ในแผนภาพการเคลื่อนไหวของข้อมูลและความหมาย

สัญลักษณ์	ความหมาย
	แหล่งรับ-ส่งข้อมูล (External Entity)
	ทิศทางการเคลื่อนไหวของข้อมูล (Flow of Data)
	กระบวนการ (Process)
	ที่เก็บข้อมูล (Data Store)

3.3.4 ใช้เป็นเอกสารประกอบระบบงาน

ในการจัดทำแผนภาพการเคลื่อนไหวของข้อมูลมีขั้นตอนและวิธีการดังนี้

- เขียนแผนภาพโดยใช้หลักการเขียนจากบนลงล่าง (Top-Down Approach) หรือจากใหญ่ไปเล็ก โดยผู้วิเคราะห์จะต้องรู้ว่าหน่วยงานภายนอกที่เกี่ยวข้องกับระบบทั้งหมดมีอะไรบ้าง ข้อมูลจะเคลื่อนไปในระบบและเคลื่อนออกจากระบบอย่างไร จะต้องมีแฟ้มข้อมูลอะไรบ้าง ซึ่งสิ่งเหล่านี้จะทำให้ผู้วิเคราะห์ระบบสามารถกำหนดขอบเขตของระบบงานได้

- เขียนแผนภาพการเคลื่อนไหวของข้อมูลระดับ 0 (Context Diagram) และทบทวนว่าได้ครอบคลุมระบบงานงานทั้งหมดหรือไม่

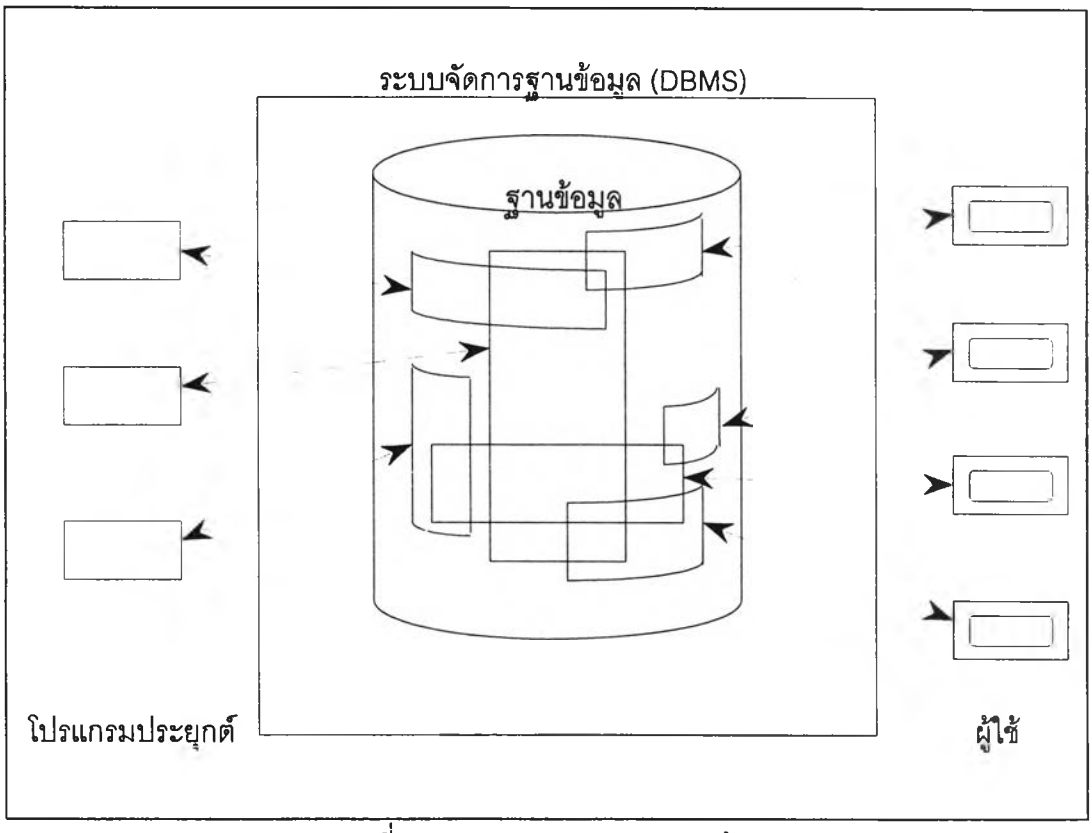
- เพิ่มเติมรายละเอียดของแต่ละกระบวนการ โดยการแบ่งออกเป็นกระบวนการย่อย ๆ รวมทั้งเพิ่มเติมข้อยกเว้นต่าง ๆ

- ปรับปรุงแผนภาพ และกำหนดชื่อหรือคำอธิบาย ให้กับสัญลักษณ์ต่าง ๆ เพื่อให้ได้ความหมายที่ชัดเจนครบถ้วนและเข้าใจได้ง่าย

3.4 ระบบฐานข้อมูล (Database System)

ฐานข้อมูล คือ การรวบรวมข้อมูลที่มีความสัมพันธ์กันและอาจอยู่ที่ต่างกัน ให้เสมือนอยู่ร่วมกัน เพื่อให้สามารถรับใช้งานที่มีวัตถุประสงค์แตกต่างกันของหน่วยงานต่างๆ โดยที่ผู้ใช้ฐานข้อมูลไม่ได้รับรู้ข้อมูลทั้งหมดในฐานข้อมูล แต่รับรู้เฉพาะส่วนที่เกี่ยวข้องกับงานของตนเท่านั้น

ระบบฐานข้อมูล คือ ระบบคอมพิวเตอร์ที่ทำหน้าที่ดูแลจัดการข้อมูลและทำให้ข้อมูลสามารถเรียกใช้ได้เมื่อต้องการ ส่วนประกอบหลักของระบบฐานข้อมูล ได้แก่ ข้อมูล, ฮาร์ดแวร์, ซอฟต์แวร์, และผู้ใช้ ดังรูปที่ 3.2



รูปที่ 3.2 ภาพรวมของระบบฐานข้อมูล

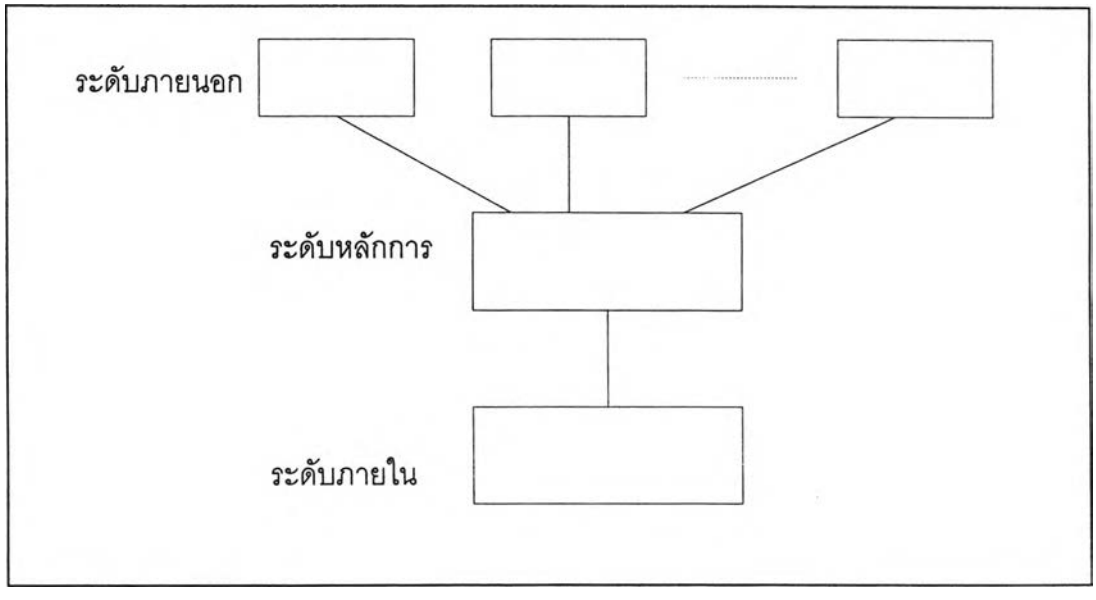
3.4.1 สถาปัตยกรรมของระบบฐานข้อมูล

ระบบฐานข้อมูลได้จัดแบ่งข้อมูลออกเป็น 3 ระดับ ดังรูปที่ 3.3 ซึ่งประกอบด้วย

3.4.1.1 ระดับภายใน (Internal Schema) คือระดับที่ต่ำที่สุด อันได้แก่ระดับของการเก็บข้อมูลทางกายภาพจริงๆ โดยพิจารณาถึงรูปแบบการบันทึกภายในสื่อของคอมพิวเตอร์

3.4.1.2 ระดับหลักการ (Conceptual Schema) คือระดับที่อยู่ที่อยู่ถัดขึ้นมาเป็นตัวเชื่อมระหว่างแบบแผนฐานข้อมูลภายนอกกับภายในและเป็นระดับที่กำหนดรูปแบบของฐานข้อมูล อันได้แก่ระดับของการมองเอนทีตี ซึ่งเป็นตัวแทนของข้อมูลทุกส่วนของฐานข้อมูล นั่นคือ ชนิด ลักษณะ และความสัมพันธ์เชิงตรรกะของข้อมูล นอกจากนี้ยังสามารถกำหนดขอบเขตของการใช้ข้อมูลของกลุ่มผู้ใช้งานได้

3.4.1.3 ระดับภายนอก (External Schema) คือระดับของข้อมูลที่แสดงผลต่อผู้ใช้แต่ละคน โดยจะอธิบายถึงวิว (View) ที่ผู้ใช้สนใจ ดังนั้นผู้ใช้งานแต่ละคนจะเห็นโครงสร้างของข้อมูลแตกต่างกัน



รูปที่ 3.3 สถาปัตยกรรมของระบบฐานข้อมูล

3.4.2 **ระบบจัดการฐานข้อมูล**(Database management System : DBMS)

ระบบจัดการฐานข้อมูล คือ ซอฟต์แวร์ที่ออกแบบขึ้นมา เพื่ออำนวยความสะดวกให้ผู้ใช้สามารถเข้าถึงข้อมูลในฐานข้อมูล ระบบจัดการฐานข้อมูล ทำหน้าที่ควบคุมดูแลการสร้างและการเรียกใช้ฐานข้อมูล ซึ่งเป็นส่วนที่มีหน้าที่อำนวยความสะดวกในการทำงานต่าง ๆ เช่น การสร้างและการแก้ไขโครงสร้างของฐานข้อมูล การจัดเก็บดูแล และการเข้าถึงฐานข้อมูล การกำหนดคุณลักษณะ ตลอดทั้งการป้องกันความเสียหาย และการรักษาความปลอดภัยของข้อมูล เป็นต้น วัตถุประสงค์หลัดของระบบจัดการฐานข้อมูลคือ การจัดหามุมมองของข้อมูลให้กับผู้ใช้ เพื่อให้สามารถใช้ข้อมูลได้อย่างมีประสิทธิภาพมากที่สุด อีกทั้งฐานข้อมูลยังมีข้อดีหลายประการเช่น

- ลดปัญหาความซ้ำซ้อนของข้อมูล (Reduce Data Redundancy)
- ลดปัญหาความขัดแย้งของข้อมูล (Inccnsistency)
- สามารถใช้ข้อมูลร่วมกัน (Sharing of Data)
- สามารถควบคุมความเป็นมาตรฐานของข้อมูล (Standard of Data)
- มีระบบความปลอดภัยที่รัดกุม (Security Restriction)
- สามารถควบคุมความคงสภาพของข้อมูลได้ (Data Integrity)
- ข้อมูลมีความเป็นอิสระ (Data Independence)

หน้าที่ของระบบจัดการฐานข้อมูลประกอบด้วย

3.4.2.1 ทำหน้าที่ติดต่อกับตัวจัดการระบบไฟล์ เนื่องจากการใช้งานส่วนใหญ่ของระบบฐานข้อมูล คือลักษณะการใช้งานกับข้อมูลขนาดใหญ่ ซึ่งเราไม่สามารถนำข้อมูลทั้งหมดนี้มาไว้ในหน่วยความจำหลักพร้อมกันได้ กล่าวคือข้อมูลทั้งหมดจะจัดเก็บอยู่ในดิสค์ และจะถูกนำมาสู่หน่วยความจำหลักหลักเฉพาะส่วนที่ต้องการใช้งาน โดยระบบจัดการฐานข้อมูลจะทำหน้าที่ประสานงานกับตัวจัดการระบบไฟล์ในการจัดเก็บ เรียกใช้ และแก้ไขข้อมูล ซึ่งทำได้โดยการออกคำสั่งด้วยภาษา DML

3.4.2.2 การควบคุมความคงสภาพ ระบบจัดการฐานข้อมูลจะต้องควบคุมค่าของข้อมูลในระบบให้อยู่ในกรอบที่ถูกต้องตามที่กำหนดไว้ในส่วนของ schema ดังนั้นการแก้ไขข้อมูลทุกครั้ง จึงเป็นหน้าที่ของระบบจัดการฐานข้อมูลที่จะต้องดูแลให้ผลลัพธ์ถูกต้องอยู่เสมอ

3.4.2.3 การควบคุมระบบความปลอดภัย ได้แก่ การป้องกันไม่ให้ผู้ที่ไม่ได้รับอนุญาตเข้ามาเห็นหรือแก้ไขข้อมูลในส่วนที่ต้องการปกป้องเอาไว้

3.4.2.4 การสร้างระบบสำรองและการฟื้นสภาพ ได้แก่ ฟังก์ชันในการจัดทำข้อมูลสำรอง และเมื่อใดก็ตามที่ปัญหาเกิดขึ้นไม่ว่าจะเป็นการขัดข้องของระบบไฟล์หรือเครื่องเกิดการเสียหาย ระบบจัดการฐานข้อมูลจะต้องใช้ระบบข้อมูลสำรองในการฟื้นสภาพ ให้ระบบข้อมูลกลับเข้าสู่สภาพที่ถูกต้องสมบูรณ์ได้

3.4.2.5 การควบคุมภาวะพร้อมกัน คือการควบคุมการใช้ข้อมูลในสภาวะที่มีผู้ใช้พร้อมๆ กันหลายคน อันได้แก่การควบคุมภาวะพร้อมกัน (Concurrency control) กล่าวคือระบบจัดการฐานข้อมูลจะต้องควบคุมลำดับการทำงานให้เป็นไปอย่างถูกต้อง

3.4.3 ฐานข้อมูลเชิงสัมพันธ์ (Relational Database)

ฐานข้อมูลเชิงสัมพันธ์ คือ การเก็บข้อมูลแบบเป็นตาราง และรวบรวมตารางต่างๆ ที่มีความสัมพันธ์ระหว่างกัน ซึ่งในฐานข้อมูลเชิงสัมพันธ์ได้กำหนดคำเพื่ออธิบายตารางดังนี้

รีเลชั่น (Relation)	หมายถึง	ตาราง (Table)
ทูเปิล (Tuple)	หมายถึง	แถว (Row)
แอตทริบิวต์ (Attribute)	หมายถึง	สดมภ์ (Column)
โดเมน (Domain)	หมายถึง	กรอบของค่าต่างๆ ที่เป็นไปได้

การวิจัยครั้งนี้ได้เลือกใช้ฐานข้อมูลเชิงสัมพันธ์ช่วยในการออกแบบระบบฐานข้อมูล เนื่องจากฐานข้อมูลเชิงสัมพันธ์มีข้อดีหลายประการดังนี้

- เป็นรูปแบบที่ง่ายต่อการเข้าใจ ไม่ซับซ้อนมากนัก
- สามารถที่จะค้นพบปัญหาที่เกิดขึ้นในการออกแบบฐานข้อมูลได้โดยง่าย และสะดวกในการแก้ไขข้อผิดพลาด
- เป็นรูปแบบที่สอดคล้องกับหลักการฐานข้อมูล
- นิยมใช้กันมากบนระบบคอมพิวเตอร์หลายระดับ

3.4.4 **การออกแบบฐานข้อมูลเชิงสัมพันธ์** (Relational Database Design)

เป็นขั้นตอนการแปลงจากแบบจำลองข้อมูลเชิงตรรกเข้าสู่ระบบฐานข้อมูลเชิงสัมพันธ์ อย่างมีเสถียรภาพภายใต้ระบบจัดการฐานข้อมูลที่ติดตั้งบนเครื่องคอมพิวเตอร์นั้นๆ การออกแบบฐานข้อมูลเชิงสัมพันธ์ ประกอบด้วย 2 ขั้นตอนดังนี้

3.4.4.1 การออกแบบแบบแบบจำลองข้อมูลเชิงตรรก (Logical Data Model)

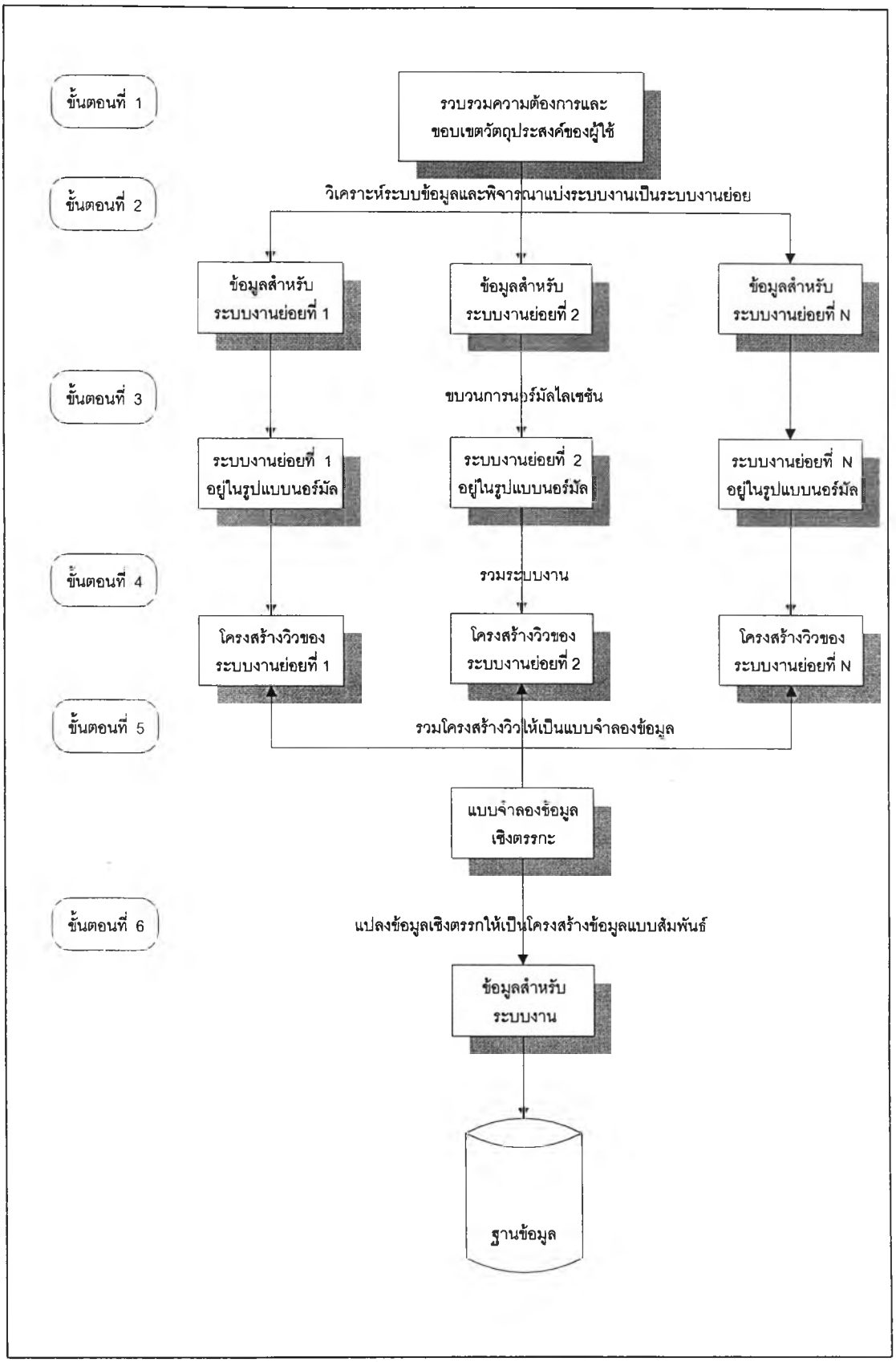
แบบจำลองข้อมูลเชิงตรรก (Logical Data Model) เป็นแบบจำลองที่สร้างขึ้นเพื่อใช้อธิบายลักษณะโครงสร้างข้อมูล และความสัมพันธ์ระหว่างข้อมูล รวมถึงการปฏิบัติการในการเรียกใช้ข้อมูลและการแก้ไขข้อมูลตามความต้องการของผู้ใช้งานในลักษณะที่ถูกต้อง ก่อนที่จะทำการแปลงให้อยู่ในรูปของฐานข้อมูลเชิงสัมพันธ์ต่อไป การออกแบบในขั้นตอนนี้จะต้องมีการกำหนดสิ่งที่เป็นพื้นฐานได้แก่

- เอนทีตี (Entity) คือสิ่งต่างๆ ในระบบ อาจเป็นสิ่งที่มีอยู่จริงในทางกายภาพหรือสิ่งที่มีอยู่จริงในทางความคิด

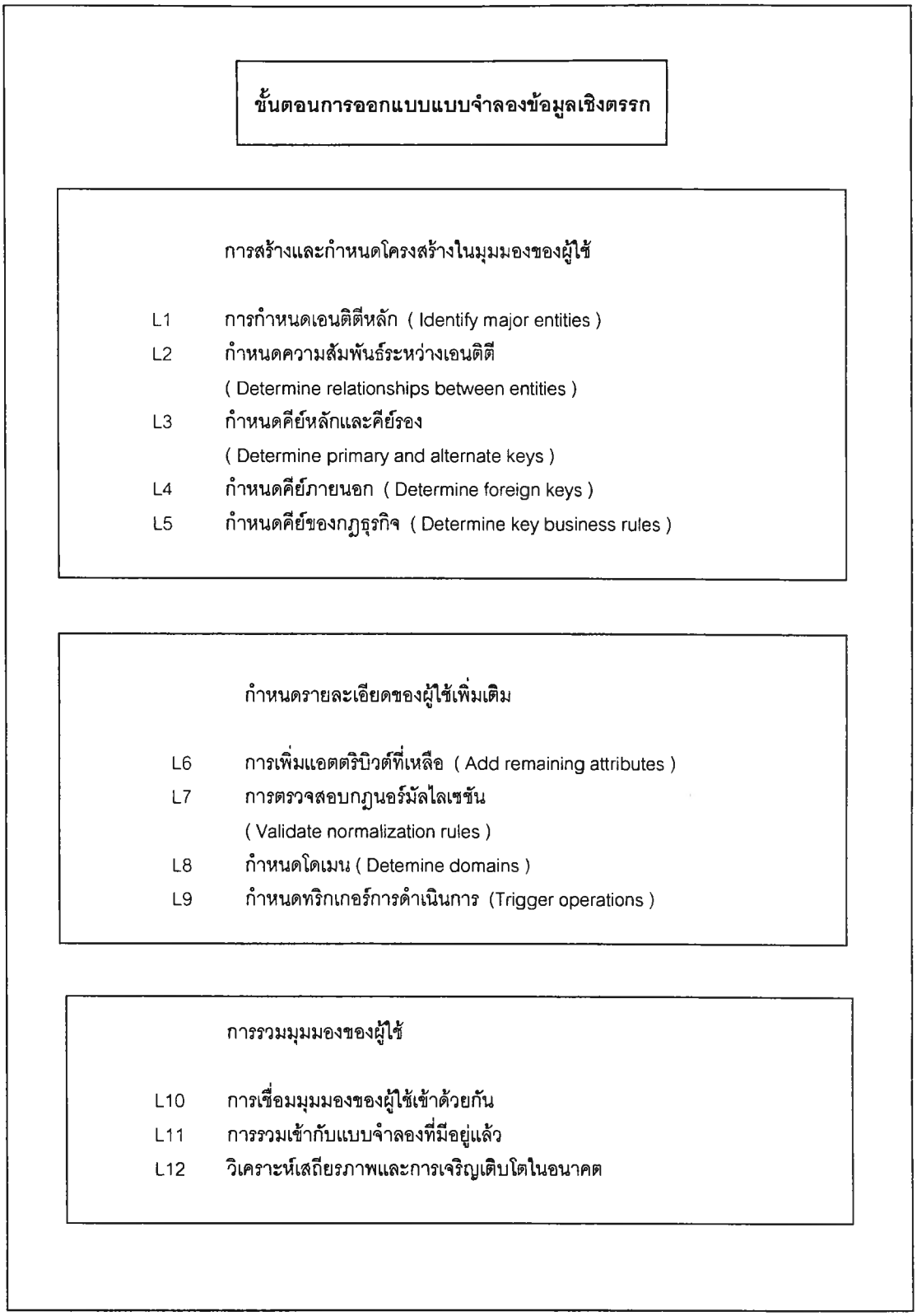
- แอตทริบิวต์ (Attribute)คือข้อมูลที่แสดงลักษณะและคุณสมบัติของเอนทิตี

- รีเลชันชิป (Relationship) คือความสัมพันธ์ระหว่างเอนทิตี

ขั้นตอนการออกแบบฐานข้อมูลเชิงตรรกได้แสดงในรูปที่ 3.4 และรายละเอียดของการออกแบบแบบจำลองข้อมูลเชิงตรรกได้แสดงในรูปที่ 3.5



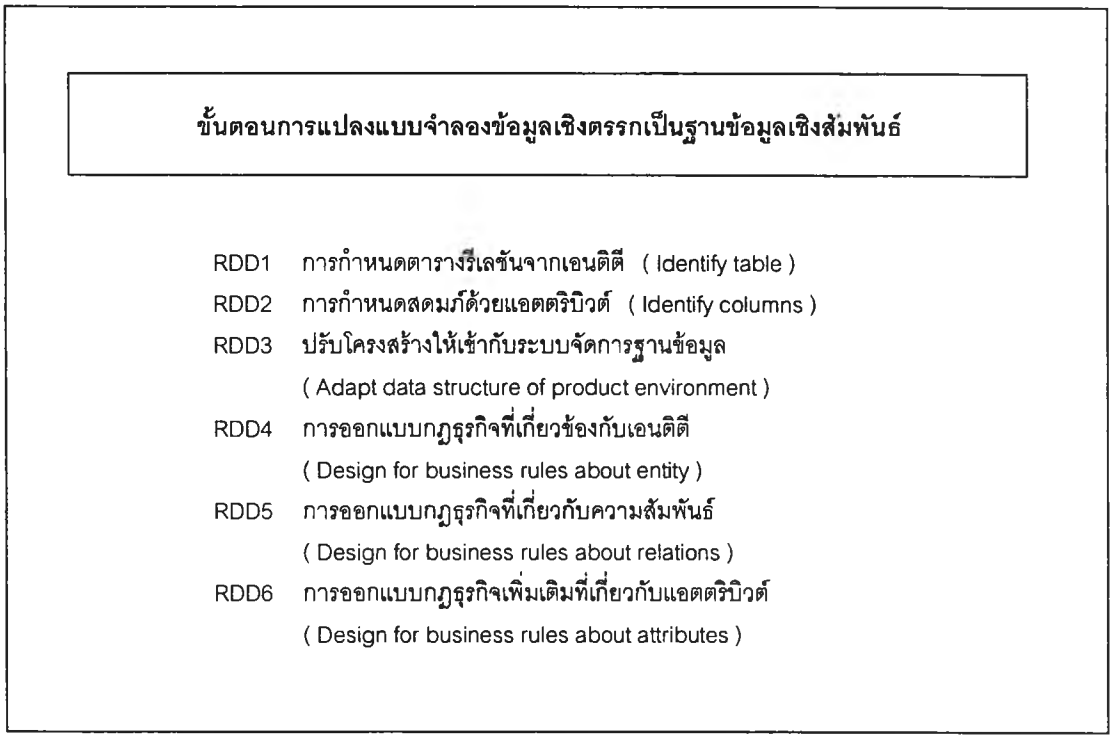
รูปที่ 3.4 ขั้นตอนการออกแบบระบบฐานข้อมูล



รูปที่ 3.5 ขั้นตอนการออกแบบแบบจำลองข้อมูลเชิงตรรก

3.4.4.2 การออกแบบจำลองข้อมูลเชิงสัมพันธ์ (Relational Data Model)

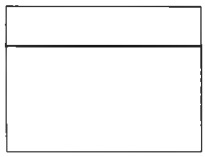
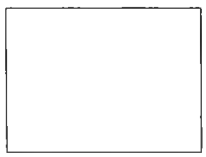
แบบจำลองข้อมูลเชิงสัมพันธ์ (Relational Data Model) เป็นแบบจำลองข้อมูลที่เปลี่ยนรูปแบบมาจากแบบจำลองข้อมูลข้อมูลเชิงตรรก เพื่อเป็นแนวคิดที่แสดงให้เห็นถึงข้อมูลที่ถูกมองเห็นโดยผู้ใช้งาน โดยจะแสดงรายละเอียดของข้อมูลที่ปรากฏแก่ผู้ใช้ การดำเนินการกับข้อมูลและกฎเกณฑ์ต่างๆ โดยใช้หลักการซึ่งเรียกว่า Relational Database Design (RDD) ซึ่งประกอบด้วย 6 ขั้นตอนดังแสดงในรูปที่ 3.6



รูปที่ 3.6 ขั้นตอนการแปลงแบบจำลองข้อมูลเชิงตรรกเป็นฐานข้อมูลเชิงสัมพันธ์

สัญลักษณ์ต่าง ๆ และความหมาย ที่ใช้ในแผนภาพแบบจำลองข้อมูลมีรายละเอียดดังแสดงในตารางที่ 3.2

ตารางที่ 3.2 สัญลักษณ์ต่าง ๆ ในแผนภาพแบบจำลองข้อมูล

สัญลักษณ์	ความหมาย
	เอนติตีของระบบ
	เอนติตีที่แสดงมาก่อนแล้วในแบบจำลองที่ระบุชื่อ
	ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (1:1)
	ความสัมพันธ์แบบหนึ่งต่อหลาย (1:M)
	ความสัมพันธ์แบบหลายต่อหลาย (M:N)
ชื่อแอตตริบิวต์ (pk)	แอตตริบิวต์ที่เป็นคีย์หลัก (Primary Key)
ชื่อแอตตริบิวต์ (ak)	แอตตริบิวต์ที่เป็นคีย์รอง (Secondary Key)
ชื่อแอตตริบิวต์ (fk)	แอตตริบิวต์ที่เป็นคีย์ภายนอก (Foreign Key)
ชื่อแอตตริบิวต์ (d)	แอตตริบิวต์ที่เป็นดิไรฟ์แอตตริบิวต์ (Derived Attribute)