

บทที่ 2

ทฤษฎีและโครงการที่เกี่ยวข้อง

2.1. ธุรกิจโรงงานสร้างชิ้นส่วนเฟอร์นิเจอร์ กรณีศึกษา ร้านมงคลสยาม เฟอร์นิเจอร์

ดำเนินกิจการสร้างชิ้นส่วนเฟอร์นิเจอร์ตามคำสั่งซื้อจากลูกค้า และขึ้นแบบตามที่ลูกค้าต้องการ โดยธุรกิจปัจจุบันมีพนักงานขาย เป็นผู้เข้าไปเสนอสินค้า ตามโรงงานเฟอร์นิเจอร์ต่างๆ ในการสั่งซื้อ ทำได้ผ่าน ทางโทรศัพท์ และทางแฟกซ์ ซึ่งยังไม่ได้มีการทำบันทึกการสั่งซื้อไว้ มีหลักฐานคือใบเสร็จรับเงินเท่านั้น หลังจากการส่งสินค้าแล้ว มีการตรวจสอบวัตถุดิบการผลิตโดยพนักงานเท่านั้น เมื่อขายสินค้าได้ จะมีการจัดทำบัญชีด้วยสมุดจดบันทึกเพียงอย่างเดียว ในบางครั้งอาจทำให้เกิดการผิดพลาดในการบันทึกการขาย และเกิดการผิดพลาดในการสรุปยอดขายรายเดือนได้

ในปัจจุบันอุตสาหกรรมเฟอร์นิเจอร์ได้มีการพัฒนาและมีอัตราการเติบโตอย่างรวดเร็ว ซึ่งเริ่มมีบทบาทสำคัญต่อเศรษฐกิจโลกมากยิ่งขึ้น ดังนั้นการศึกษาถึงแนวโน้มการเปลี่ยนแปลงของอุตสาหกรรมเฟอร์นิเจอร์จึงเป็นสิ่งสำคัญ เพื่อหาแนวทางและกำหนดทิศทางของอุตสาหกรรมเฟอร์นิเจอร์เพื่อรองรับกระแสการเปลี่ยนแปลงในอนาคต เช่น การเปลี่ยนแปลงกระแสโลกาภิวัตน์ กลไกการผลิตและการบริโภคซึ่งจะส่งผลโดยตรงต่ออุตสาหกรรมเฟอร์นิเจอร์ทั้งด้านการผลิต และรูปแบบสินค้าเฟอร์นิเจอร์ที่เปลี่ยนแปลงอย่างสร้างสรรค์ โดยสามารถแบ่งรูปแบบสินค้าเฟอร์นิเจอร์ในอนาคตได้ 5 ประเภท คือ

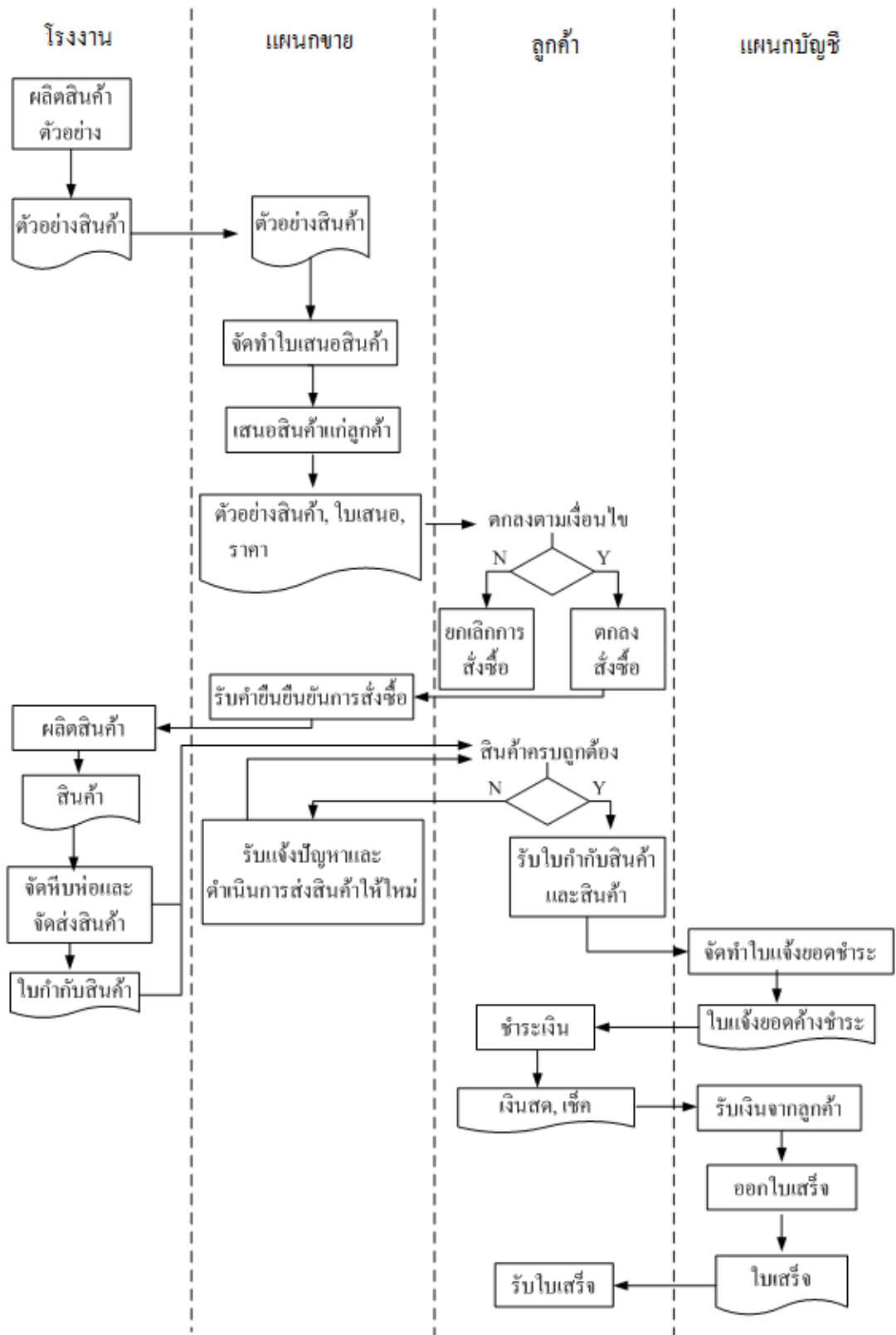
1. **เฟอร์นิเจอร์เพื่อสิ่งแวดล้อม (Eco - Furniture)** – สิ่งแวดล้อมเป็นปัจจัยสำคัญที่ทั่วโลกให้ความสนใจ เนื่องจากปัจจุบัน โลกเริ่มประสบปัญหาสภาวะโลกร้อน จึงได้มีการออกมาตรการด้านสิ่งแวดล้อมที่เข้มงวดทั้งในการผลิตและการนำเข้าสินค้า รวมถึงการนำขยะหรือวัสดุเหลือใช้มาผลิตเป็นของใช้เพิ่มขึ้นเพื่อลดปัญหาขยะล้นโลก การวิจัยและพัฒนาวัตถุดิบการผลิต และกระบวนการผลิตที่เป็นมิตรกับสิ่งแวดล้อม และการผลิตเฟอร์นิเจอร์วัสดุรีไซเคิลซึ่งขั้นตอนจะไม่ยุ่งยากซับซ้อน แต่ต้องอาศัยความคิดสร้างสรรค์ในการออกแบบเพื่อให้แปลกใหม่มีความน่าสนใจ และสามารถใช้งานได้ดี
2. **เฟอร์นิเจอร์อเนกประสงค์ (Multifunction Furniture)** แนวโน้มการเพิ่มขึ้นของประชากรในเมือง ส่งผลให้วิถีชีวิตของสังคมเปลี่ยนไป มีการพักอาศัยในพื้นที่ค่อนข้างจำกัด เช่น คอนโดมีเนียม บ้านพักอาศัยที่มีพื้นที่จำกัด เป็นต้น เฟอร์นิเจอร์จะเน้นด้านประโยชน์ใช้สอยสามารถปรับเปลี่ยนฟังก์ชันได้หลากหลายสำหรับพื้นที่จำกัด เช่น โซฟา

ที่สามารถปรับเป็นเตียงได้ (Day Bed) เฟอร์นิเจอร์ที่สามารถปรับเป็นได้ทั้งเตียง โต๊ะทำงาน และที่เก็บของเป็นต้น

3. **เฟอร์นิเจอร์ผสมผสานวัฒนธรรม (Cultural-mixed Furniture)** การออกแบบเฟอร์นิเจอร์ที่ผสมผสานทางด้านวัฒนธรรม เช่น ผสมผสานระหว่างวัฒนธรรมตะวันตกและตะวันออก เพื่อให้เป็นรูปแบบเฟอร์นิเจอร์แบบใหม่ ที่สามารถตอบสนองความต้องการผู้บริโภคทั้งตะวันตกและตะวันออก เฟอร์นิเจอร์ผสมผสานวัฒนธรรมจะมีรูปร่างที่แปลกใหม่ เช่น โต๊ะที่มีความสูงไม่มากนักเมื่อเปรียบเทียบกับโต๊ะทั่วไปการใช้เก้าอี้ที่ไม่มีพนักพิง
4. **เฟอร์นิเจอร์เน้นประโยชน์ใช้สอย (Functionalism Furniture)** การออกแบบที่มุ่งเน้นเฉพาะ “สิ่งที่จำเป็น” คือ สิ่งที่มีประโยชน์จริงในชีวิตประจำวันของคนจริงๆ สามารถตอบสนองความต้องการพื้นฐานในการใช้ชีวิตของคนในแต่ละกลุ่มได้ การออกแบบเฟอร์นิเจอร์ประเภทนี้จะต้องมีการศึกษาการตลาดอย่างจริงจังถึงความต้องการของผู้บริโภคแต่ละกลุ่มเพื่อให้สามารถนำมาใช้ให้เกิดประโยชน์อย่างแท้จริง เช่น ปัจจุบันคนเริ่มเข้าสู่สังคมเมืองมากขึ้น พื้นที่อยู่อาศัยค่อนข้างเล็ก จึงมีการออกแบบเฟอร์นิเจอร์ขนาดเล็กเหมาะสมกับพื้นที่ใช้สอยในห้องพัก
5. **เฟอร์นิเจอร์เพื่อสุขภาพ (Furniture for health)** จากแนวโน้มพฤติกรรมผู้บริโภคที่เริ่มตระหนักให้ความสำคัญด้านสุขภาพเพิ่มสูงขึ้น รวมถึงการเข้าสู่สังคมผู้สูงอายุที่มีมากยิ่งขึ้น ส่งผลให้การออกแบบเฟอร์นิเจอร์ต้องคำนึงถึงการใช้งานที่ส่งผลดีสุขภาพเช่นกัน เฟอร์นิเจอร์จะปรับให้เข้ากับรูปร่างของผู้ใช้งาน และออกแบบเพื่อช่วยลดอาการเจ็บป่วย การใช้นวัตกรรมใหม่ๆ เข้ามาผลิตเฟอร์นิเจอร์เพื่อสุขภาพ เช่น โซฟา หมอน ที่นอน ป้องกันไรฝุ่น เป็นต้น

การดำเนินงานของระบบโรงงานผลิตชิ้นส่วนเฟอร์นิเจอร์แบบปัจจุบันสามารถอธิบายกระบวนการทำงานในแต่ละส่วนได้ดังนี้

2.1.1. รายละเอียดขั้นตอนการทำงาน (Workflow) ของระบบงานปัจจุบัน



2.2 ภาษา SQL

2.2.1 ความหมายของภาษา SQL

SQL (Structure Query Language) หมายถึง เป็นภาษาที่ใช้ในการเขียนโปรแกรม เพื่อจัดการกับฐานข้อมูล โดยเฉพาะ เราสามารถแบ่งการทำงานได้เป็น 4 ประเภท ดังนี้

1. Select query ใช้ในการดึงข้อมูลในฐานข้อมูล จะมีการค้นหารายการจากตารางในฐานข้อมูล ตั้งแต่หนึ่งตารางขึ้นไป ตามเงื่อนไขที่สั่ง ผลลัพธ์ที่ได้จะเป็นเซตของข้อมูลที่สามารถสร้างเป็นตารางใหม่ หรือใช้แสดงออกมาทางจอภาพเท่านั้น โดยมีรูปแบบดังนี้

2. Update query ใช้สำหรับแก้ไขข้อมูล โดยแก้ไขในคอลัมน์ที่มีตรงตามเงื่อนไข

3. Insert query ใช้สำหรับการเพิ่มข้อมูลใหม่ๆเข้าไปในฐานข้อมูล

4. Delete query ใช้สำหรับลบข้อมูลออกไป

2.2.2 ประเภทของคำสั่ง SQL

คำสั่งในภาษา SQL ประกอบด้วยชุดคำสั่งหลัก ๆ 3 ประเภทด้วยกัน ได้แก่

1. DDL (Data Definition Language) เป็นชุดคำสั่งที่ใช้ในการนิยาม กำหนด หรือสร้างข้อมูล เช่น table, index, view ได้แก่ คำสั่ง create table, create view เป็นต้น

2. DML (Data Manipulation Language) เป็นชุดคำสั่งที่ใช้ในการประมวลผล หรือจัดการกับข้อมูลในฐานข้อมูล ได้แก่ คำสั่ง Select, insert, update เป็นต้น

3. DCL (Data Control Language) เป็นคำสั่งที่ใช้ในการควบคุมสิทธิ์ของผู้ใช้ในการใช้ข้อมูล รวมทั้งส่วนที่ใช้ควบคุม การใช้งานฐานข้อมูลจากผู้ใช้งานหลายคนพร้อมกัน ได้แก่ คำสั่ง grant, revoke เป็นต้น

2.3 ภาษา PHP

2.3.1 ความหมายของภาษา PHP

พีเอชพี (PHP) หมายถึง ภาษาคอมพิวเตอร์ในลักษณะเซิร์ฟเวอร์-ไซด์ สคริปต์ โดยลักษณะที่อยู่ในลักษณะโอเพนซอร์ส ภาษาพีเอชพีใช้สำหรับจัดทำเว็บไซต์และแสดงผลออกมาในรูปแบบ HTML โดยมีรากฐานโครงสร้างคำสั่งมาจากภาษา ภาษาซี ภาษาจาวา และ ภาษาเพิร์ล ซึ่ง ภาษาพีเอชพี นั้นง่ายต่อการเรียนรู้ ซึ่งเป้าหมายหลักของภาษานี้ คือให้นักพัฒนาเว็บไซต์สามารถเขียนเว็บเพจ ที่มีความตอบโต้ได้อย่างรวดเร็ว ลักษณะการเขียนสคริปต์จะเขียนแทรกไว้ภายในไฟล์ HTML โดยเปิดด้วยแท็ก <?php หรือ <? หรือ <script language="php"> และปิดด้วย ?> หรือ </script> ดังนี้

```
<html>

<head>

    <title>PHP First run</title>

</head>

<body>

<? php

    Print "My first PHP script.";

?>

<br><br><font color=blue><h2>

Please enjoy the script.

</h2></font>

</body>
```

2.3.2 คุณสมบัติของภาษา PHP

PHP เป็นภาษาที่เขียนง่าย สามารถนำมาใช้ทำเว็บเพจที่จำเป็นต้องมีการตอบสนองกับผู้ใช้โดยเฉพาะอย่างยิ่ง PHP มีความสามารถในการนำข้อมูล (Database) ประเภทต่างๆมาแสดงบนเว็บเพจ จึงเหมาะแก่การนำมาใช้ทำเว็บบอร์ด , เว็บเมล, ไดนามิกเว็บเพจ เพื่อประโยชน์ในทางพาณิชย์อิเล็กทรอนิกส์ (e-commerce) ตลอดจนการสร้างเว็บแอปพลิเคชันเพื่อใช้งานภายในองค์กรที่ต้องการคุณสมบัติในการเรียกใช้งานได้จากทุกที่ โดยไม่ต้องมีการติดตั้งในเครื่องผู้ใช้ เช่น การเรียกใช้แอปพลิเคชันจากสาขาต่างๆ เป็นต้น

PHP สามารถทำงานร่วมกับฐานข้อมูลได้หลายชนิด ซึ่งฐานข้อมูลส่วนหนึ่งที่รองรับได้แก่ MySQL ซึ่งจะนำฐานข้อมูลนี้มาพัฒนาระบบสนับสนุนการตัดสินใจการขนส่งสินค้า

2.4 ระบบฐานข้อมูล (Database System)

2.4.1 ความหมายของระบบฐานข้อมูล

ฐานข้อมูล(Database) หมายถึง ชุดของข้อมูลที่มีความสัมพันธ์กันที่ถูกรวบรวม จัดเก็บไว้ด้วยกัน เพื่อให้สามารถใช้ข้อมูลเหล่านั้นร่วมกันได้อย่างรวดเร็วและมีประสิทธิภาพ การจัดเก็บข้อมูลจะมีประสิทธิภาพได้ก็ต่อเมื่อมีวิธีการจัดการข้อมูลที่ดี กล่าวคือ วิธีการจัดเก็บและค้นหาข้อมูลต้องเป็นไปอย่างมีประสิทธิภาพและรวดเร็ว โดยทั่วไปเมื่อข้อมูลมีขนาดใหญ่ขึ้น การสร้างฐานข้อมูลมักจะกระทำโดยใช้เครื่องหมายคอมพิวเตอร์มาช่วย เพื่อให้สามารถจัดเก็บและใช้ข้อมูลเหล่านั้นร่วมกัน ตลอดจนสามารถค้นหาได้อย่างรวดเร็ว

2.4.2 หน้าที่ของระบบจัดการฐานข้อมูล

หน้าที่ของระบบจัดการฐานข้อมูลสามารถแบ่งออกเป็นข้อๆได้ดังต่อไปนี้

1. ช่วยกำหนดและเก็บโครงสร้างฐานข้อมูล (Define and Store Database Structure)
2. การเรียกใช้ข้อมูลจากฐานข้อมูล (Load Database) เมื่อมีการประมวลผลที่เกิดจากการทำงานของโปรแกรมประยุกต์ ระบบฐานข้อมูลจะทำการรับและเก็บข้อมูลที่ป้อนเข้ามาเอาไว้ในฐานข้อมูลเพื่อใช้ในการประมวลผลต่อไป
3. เก็บและดูแลข้อมูล (Store and Maintain Data) ข้อมูลในระบบฐานข้อมูลจะถูกเก็บรวบรวมไว้ด้วยกัน โดยมีระบบจัดการฐานข้อมูลเป็นผู้ดูแลรักษาข้อมูลนั้น
4. ประสานงานกับระบบปฏิบัติการ (Operating System) ดังที่ได้ทราบกันอยู่แล้วว่าระบบปฏิบัติการเป็นโปรแกรมที่คอยควบคุมการทำงานของอุปกรณ์คอมพิวเตอร์หรือโปรแกรมต่างๆในเครื่องคอมพิวเตอร์ระบบการจัดการฐานข้อมูลก็จะทำหน้าที่ประสานงานกับระบบปฏิบัติการเพื่อให้การทำงานเป็นไปอย่างถูกต้องตามที่ผู้ใช้ต้องการ ไม่ว่าจะเป็นการเรียกใช้ข้อมูลการแก้ไขข้อมูลหรือการออกรายงาน
5. ช่วยควบคุมความปลอดภัย (Security Control) ในระบบการจัดการฐานข้อมูล จะมีวิธีควบคุมเพื่อเป็นการป้องกันความเสียหายที่อาจจะเกิดขึ้นได้กับฐานข้อมูล ไม่ว่าจะเป็นการเรียกใช้หรือแก้ไขเปลี่ยนแปลงข้อมูลของผู้ใช้ในระบบ ผู้ใช้สามารถเรียกข้อมูลขึ้นมาทำการแก้ไขได้
6. การจัดทำข้อมูลสำรองและการกู้ (Backup and Recovery) ในระบบจัดการฐานข้อมูลจะจัดทำข้อมูลสำรองของฐานข้อมูลเอาไว้และเมื่อมีปัญหาเกี่ยวกับระบบฐานข้อมูล เช่นแฟ้มข้อมูลหาย ซึ่งอาจเกิดขึ้นเนื่องจากดิสก์เสีย ลบผิดแฟ้มข้อมูล หรือไฟไหม้ ฯลฯ ระบบจัดการฐานข้อมูลจะใช้ระบบข้อมูลสำรองนี้ในการฟื้นฟูสภาพการทำงานของระบบให้สู่ภาวะปกติได้
7. ควบคุมการใช้ข้อมูลพร้อมกัน (Concurrency Control) ในระบบคอมพิวเตอร์ที่ใช้อยู่ปัจจุบัน โปรแกรมการทำงานมักจะเป็นแบบผู้ใช้หลายคน (Multi User) จึงทำให้ผู้ใช้แต่ละคนสามารถเรียกใช้ข้อมูลได้พร้อมกัน ระบบจัดการฐานข้อมูลที่มีคุณสมบัติควบคุมการใช้ข้อมูลพร้อมกันนี้ จะทำการควบคุมการใช้ข้อมูลพร้อมกันของผู้ใช้หลายคนในเวลาเดียวกันได้ โดยมีระบบการ

ควบคุมที่ถูกต้องเหมาะสม เช่น ถ้าการแก้ไขข้อมูลนั้นยังไม่เรียบร้อย ผู้ใช้อื่นๆ ที่ต้องการเรียกใช้ข้อมูลนี้จะไม่สามารถเรียกข้อมูลนั้นๆ ขึ้นมาทำงานใดๆ ได้ ต้องรอจนกว่าการแก้ไขข้อมูลของผู้ที่เรียกใช้ข้อมูลนั้นก่อนจะเสร็จเรียบร้อย จึงจะสามารถเรียกข้อมูลนั้นไปใช้งานต่อได้ ทั้งนี้เพื่อป้องกันไม่ให้เกิดปัญหาการเรียกใช้ข้อมูลที่ไม่ถูกต้อง

8. ควบคุมความบูรณาการของข้อมูล (Integrity Control)ระบบจัดการฐานข้อมูลจะทำการควบคุมค่าของข้อมูลในระบบให้ถูกต้องตามที่ควรจะเป็น

9. จัดทำพจนานุกรมข้อมูล (Data Dictionary) ระบบจัดการฐานข้อมูลจะทำการสร้างพจนานุกรมข้อมูลขึ้นมาให้เมื่อมีการกำหนดโครงสร้างของกับฐานข้อมูลมา เพื่อเป็นเอกสารหรือแหล่งข้อมูล เช่น ชื่อ เพิ่มข้อมูล ชื่อเขตข้อมูล เป็นต้น

2.4.3 ความสัมพันธ์ระหว่างเอนทิตี

ความสัมพันธ์ระหว่างเอนทิตี แบ่งออกเป็น 3 ประเภท คือ

1. **ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One-to-one Relationships)** เป็นการแสดงความสัมพันธ์ของข้อมูลในเอนทิตีหนึ่งที่มีความสัมพันธ์กับข้อมูลในอีกเอนทิตีหนึ่ง ในลักษณะหนึ่งต่อหนึ่ง (1 : 1)

2. **ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One-to-many Relationships)** เป็นการแสดงความสัมพันธ์ของข้อมูลในเอนทิตีหนึ่งที่มีความสัมพันธ์กับข้อมูลหลายๆ ข้อมูลในอีกเอนทิตีหนึ่ง ในลักษณะ (1:m)

3. **ความสัมพันธ์แบบกลุ่มต่อกลุ่ม (Many-to-many Relationships)** เป็นการแสดงความสัมพันธ์ของข้อมูลสองเอนทิตีในลักษณะกลุ่มต่อกลุ่ม (m:n)

ตัวอย่าง เอนทิตีใบสั่งซื้อแต่ละใบจะสามารถสั่งสินค้าได้มากกว่าหนึ่งชนิด

ความสัมพันธ์ของข้อมูลจากเอนทิตีใบสั่งซื้อไปยังเอนทิตีสินค้า จึงเป็นแบบหนึ่งต่อกลุ่ม (1:m) ในขณะที่สินค้าแต่ละชนิดจะถูกสั่งอยู่ในใบสั่งซื้อหลายใบ ความสัมพันธ์ของข้อมูลจากเอนทิตีสินค้าไปยังเอนทิตีใบสั่งซื้อจึงเป็นแบบหนึ่งต่อกลุ่ม (1:m) ดังนั้นความสัมพันธ์ของเอนทิตีทั้งสอง จึงเป็นแบบกลุ่มต่อกลุ่ม (m:n)

2.4.4 การนอร์มัลไลซ์ (Normalization)

หลังจากที่ผู้ออกแบบได้ขอบเขตข้อมูลทั้งหมดที่ต้องการเก็บแล้ว ซึ่งโดยส่วนมากเกิดจากรูปแบบรายงานบ้าง รูปแบบใบเสร็จรับเงินบ้าง รูปแบบใบส่งสินค้าบ้าง โดยมากมักจะเหมารวมเอวานั้นคือ รูปแบบของตารางที่ต้องการเก็บข้อมูล ซึ่งสิ่งเหล่านั้นนำมาซึ่งความซ้ำซ้อนของข้อมูล และทำให้มีขนาดใหญ่เกินหน่วยความจำ ปัญหาของรีเลชันที่เกิดขึ้นเหล่านี้สามารถขจัดได้ด้วย “ขบวนการ Normalization” ซึ่งแนวคิดนี้ถูกคิดค้นโดย E.F.Codd ซึ่งเป็นกระบวนการที่นำเค้าร่างของ Relation มาทำให้อยู่ในรูปแบบที่เป็นบรรทัดฐาน (Normal Form) เพื่อให้แน่ใจว่าการออกแบบเค้าร่างของ Relation เป็นการออกแบบที่เหมาะสม

ให้ทำการตรวจสอบเอนทิตีต่าง ๆ ให้อยู่ในกฎนอร์มัลไลเซชันซึ่งประกอบด้วย 1NF, 2NF, 3NF, BCNF, 4NF, 5NF ซึ่งจะได้กล่าวในหัวข้อต่อไป โดยมีประโยชน์ดังนี้

1. ลดที่ว่างที่ต้องใช้ในการเก็บข้อมูล
2. ลดความผิดพลาดความไม่ตรงกันของข้อมูลในฐานข้อมูล
3. ลดการเกิดอะนอร์มัลไลของการลบและแก้ไขข้อมูล
4. เพิ่มความคงทนแก่โครงสร้างฐานข้อมูล

ในทางปฏิบัติการทำ นอร์มัลไลซ์จะเริ่มจาก E-R Model ก่อน แล้วจึงทำการ Map จาก E-R Model เป็น Relation แบบ 1NF ก่อน โดยให้ Attributes ที่เกี่ยวข้องกันจะอยู่ในตารางเดียวกัน สำหรับ Application ใหญ่ ๆ มี Attributes ประมาณ 500 Attributes ใช้ E-R Model จะได้ 1NF ประมาณ 80 ตาราง เมื่อทำ ถึง 5NF จะได้ไม่เกิน 100 ตาราง ในกรณีได้ตารางเป็นนอร์มัลไลเซชันที่สมบูรณ์แล้วสิ่งที่ต้องระวังคือไม่แตกตารางนั้นย่อยลงไปอีก

2.4.5 ระดับนอร์มัลไลเซชัน

นอร์มัลไลเซชันเป็นกระบวนการเพื่อพัฒนาการเชื่อมต่อของข้อมูลเพื่อแก้ปัญหาของรีเลชันที่ว่าการออกแบบฐานข้อมูลทั้งทางตรรกะและทางกายภาพที่ได้ออกมาใช้ได้หรือยัง การนอร์มัลไลเซชันแบ่งออกได้เป็นหลายระดับดังนี้

1. First Normal Form (1NF) ทุก ๆ Field ในแต่ละ Record จะเป็น Single value นั่นคือในตารางหนึ่ง ๆ จะไม่มีค่าของกลุ่มข้อมูลที่ซ้ำกัน (Repeating Group) หรือเป็นการขจัดแอตทริบิวต์หรือกลุ่มแอตทริบิวต์ที่ซ้ำกันไปอยู่ในเอนทิตีลูก เพื่อแต่ละรายการในเอนทิตีจะไม่มีค่าของแอตทริบิวต์หรือค่าของกลุ่มแอตทริบิวต์ที่ซ้ำกัน

2. Second Normal Form (2NF) ต้องเป็น First Normal Form (1NF) และต้องมี Key (บางตำราอาจจะเรียกว่า Index) ที่ทุก Non-key จะต้องขึ้นอยู่กับ Key นี้ และมีเพียง Key เดียวในหนึ่งตารางซึ่งเรียกว่า Primary Key การที่ทุกตาราง (Table) ต้องมี Key ก็เพราะเราต้องการให้แน่ใจว่าทุกข้อมูลใน Record ต่าง ๆ สามารถค้นหาได้โดยใช้ Key สรุปก็คือ นอร์มัลไลเซชันระดับที่ 2 (Second normal form : 2NF) เป็นการขจัดแอตทริบิวต์ที่ไม่ขึ้นกับทั้งส่วนของคีย์หลักออกไป เพื่อให้แอตทริบิวต์อื่นทั้งหมดขึ้นตรงกับส่วนที่เป็นคีย์หลักทั้งหมดเท่านั้น

3. Third Normal Form (3NF) นอร์มัลไลเซชันระดับที่ 3 (Third normal form : 3NF) คือ ขบวนการที่พยายามขจัดสภาพของ Transitive Dependency ออกไป โดยมีข้อกำหนดว่าต้องเป็น Second Normal Form (2NF) และ ไม่มี Transitive dependence หรือเป็นการขจัดแอตทริบิวต์ที่ไม่เป็นคีย์ที่ขึ้น (Transitive dependent) ตรงกับแอตทริบิวต์อื่นที่ไม่ใช่คีย์หลักออกไปเพื่อให้ แอตทริบิวต์ที่ไม่ใช่คีย์หลักต้องขึ้นตรงกับทั้งส่วนที่เป็นคีย์หลัก และไม่ขึ้นกับแอตทริบิวต์อื่นที่ไม่ใช่คีย์หลัก

4. BCNF (Boyce/Codd Normal Form) นอร์มัลไลเซชันระดับที่ 4 ต้องเป็น 3NF และไม่มี Attribute อื่นในรีเลชันที่สามารถระบุค่าของ Attribute ที่เป็นคีย์หลัก หรือส่วนหนึ่งส่วนใดของคีย์

หลักในกรณีที่มีคีย์หลักเป็นคีย์ผสม โดยทั่วไปรูปแบบ BCNF จะอยู่ในรูปแบบ 3NF แต่ไม่จำเป็นเสมอไปที่รูปแบบ 3NF จะอยู่ในรูปแบบ BCNF ทั้งนี้เนื่องจากรูปแบบนี้เป็นการขยายขอบเขตของรูปแบบ 3NF ให้เหมาะสมยิ่งขึ้น โดยรูปแบบที่ต้องทำให้เป็น BCNF โดยจะมีคุณสมบัติ คือ เป็นรีเลชันที่มีคีย์คู่แข่งหลายคีย์ (Multiple Candidate Key) โดยที่คีย์คู่แข่งเป็นคีย์ผสม (Composite Key) และคีย์คู่แข่งนั้นมีบางส่วนซ้ำซ้อนกัน (Overlapped) หรือมี Attribute บางตัวร่วมกันอยู่

5. Forth Normal Form (4NF) ต้องอยู่ในรูปแบบ BCNF และเป็นรีเลชันที่ไม่มีความสัมพันธ์ในการระบุค่าของ Attribute แบบหลายค่าโดยที่ Attribute ที่ถูกระบุค่าเหล่านี้ไม่มีความสัมพันธ์กัน (Independently Multivalued Dependency)

6. Fifth Normal Form (5NF) หรือเรียกว่า Project-Join Normal Form (PJ/NF) โดยจะต้องอยู่ในรูปแบบ 4NF และไม่มี Symmetric Constraint กล่าวคือหากมีการแตกรีเลชันออกเป็นรีเลชันย่อย (Projection) และเมื่อทำการเชื่อมโยงรีเลชันย่อยทั้งหมด (Joint) จะไม่ก่อนให้เกิดข้อมูลใหม่ที่ไม่เหมือนรีเลชันเดิม (Spurious Tupes) ในการแตกรีเลชันออกมาจากรูปแบบ 4NF นั้น ถ้าเชื่อมโยงรีเลชันย่อยนั้นใหม่ หากไม่มีข้อมูลที่แตกต่างไปจากรีเลชันเดิมสามารถแตกรีเลชันนั้นได้ แต่ถ้าหากแตกเป็นรีเลชันย่อยแล้วเกิดข้อมูลไม่เหมือนกับรีเลชันเดิม ก็ไม่ควรแตกรีเลชัน และให้ถือว่ารีเลชันเดิมอยู่ใน 5NF แล้ว

2.5 UML (Unified Modeling Language)

ในการพัฒนาระบบด้วยหลักการ OOAD (Object Oriented Analysis and Design) นั้น สิ่งหนึ่งที่ทำเป็นอย่างยิ่งคือการสร้างแบบจำลองของอ็อบเจกต์ คลาส และองค์ประกอบอื่นๆ ของระบบ ซึ่งการถ่ายทอดแบบจำลองออกมาให้แก่ผู้มีส่วนเกี่ยวข้องกับระบบได้รับทราบนั้น ทางที่จะทำให้เข้าใจได้ตรงกันมากที่สุดคือ การแสดงในรูปแบบของสัญลักษณ์ที่มองเห็นได้ เช่น รูปภาพ แผนภาพ เป็นต้น โดยเครื่องมือที่ได้รับการยอมรับและเป็นที่ยอมรับใช้มากที่สุดคือ UML (Unified Modeling Language) ซึ่ง UML จัดได้ว่าเป็นภาษาหนึ่ง เพราะมีหน่วยของภาษาคอมพิวเตอร์ กล่าวคือ มีทั้งคำศัพท์ และไวยากรณ์ที่ชัดเจน

แผนภาพของภาษา UML นั้น ถูกแบ่งออกเป็น 2 กลุ่ม รวมทั้งหมด 9 แผนภาพ โดยกลุ่มที่หนึ่งเป็นกลุ่มแผนภาพที่แสดงให้เห็นโครงสร้างเชิงสถิตของระบบ (Structural Diagram) คือ โครงสร้างในส่วนที่ไม่มีการเปลี่ยนแปลงหรือเคลื่อนไหวแม้จะมีเหตุการณ์ใดๆเกิดขึ้น ได้แก่ Class Diagram, Object Diagram, Component Diagram และ Deployment Diagram และอีกกลุ่มหนึ่งคือ กลุ่มแผนภาพที่แสดงให้เห็นภาพเชิงกิจกรรมของระบบ (Behavioral Diagram) คือแสดงให้เห็นถึงพฤติกรรมของระบบที่มีการเปลี่ยนแปลงไปเมื่อมีเหตุการณ์ใดๆเกิดขึ้น และแสดงให้เห็นถึงความสามารถของระบบที่ดำเนินการในหน้าที่บางอย่างได้ ได้แก่ Use Case Diagram, Sequence Diagram, Collaboration Diagram, State chart Diagram และ Activity Diagram

2.5.1 Use Case Diagram

โดยทั่วไปในขั้นตอนการวิเคราะห์และออกแบบระบบตามแนวคิดเชิงวัตถุ มักจะใช้ Use Case Diagram มาเป็นเครื่องมือในการจำลองหน้าที่ของระบบที่ผู้ใช้งานต้องการ เนื่องจาก Use Case Diagram เป็นแผนภาพที่ใช้แสดงถึงขั้นตอนการทำงานที่สำคัญของระบบ หรือแสดงหน้าที่และงานที่ระบบจะต้องปฏิบัติ เพื่อตอบสนองต่อผู้กระทำต่อระบบ

2.5.2 Object Diagram

เป็นแผนภาพที่ใช้ในการแสดงกลุ่มของออบเจกต์และความสัมพันธ์ระหว่างออบเจกต์ที่เกิดขึ้นในคลาสต่างๆของ Class Diagram

2. 5.3 Class Diagram

เป็นแผนภาพที่แสดงให้เห็นถึงกลุ่มของคลาส โครงสร้างของคลาส และ Interface ตลอดจนแสดงความสัมพันธ์ระหว่างคลาส การเริ่มต้นสร้าง Class Diagram นั้นส่วนใหญ่จะต้องค้นหาออบเจกต์ใน Use Case Diagram ก่อนซึ่งเทคนิคที่ใช้ในการค้นหาจะแตกต่างกันออกไปตามประสบการณ์ของทีมงาน เช่น ค้นหาจากคำอธิบายรายละเอียดของ Use Case โดยชื่อคลาสหาได้จากคำนาม ส่วนคุณสมบัติ (Attribute) หาได้จากคำคุณศัพท์ และการดำเนินการ (Operation) สามารถหาได้จากคำกริยา เป็นต้น

2.5.4 Component Diagram

เป็นแผนภาพที่แสดงให้เห็นถึงโครงสร้างทางกายภาพของโปรแกรม ประกอบด้วยส่วนประกอบต่างๆ ที่เรียกว่า Component ซึ่งหมายถึง ส่วนประกอบย่อยของซอฟต์แวร์ที่ใช้ในระบบ ทำให้เห็นว่าประกอบไปด้วยไฟล์ใดบ้าง ส่วนใหญ่มักจะแสดงไฟล์ที่เป็น Source Code, Binary Code ได้แก่ .exe .dll เป็นต้น

2.5.5 Deployment Diagram

เป็นแผนภาพที่แสดงโครงสร้างทางด้านฮาร์ดแวร์ของระบบงาน ส่วนใหญ่จะใช้ร่วมกับ Component Diagram โดยการมองอุปกรณ์ฮาร์ดแวร์ทั้งหมดเป็นออบเจกต์หรือคลาสได้เช่นเดียวกัน

2. 5.6 Sequence Diagram

เป็นแผนภาพที่แสดงให้เห็นถึงการปฏิสัมพันธ์ (Interaction) ระหว่างออบเจกต์ โดยเฉพาะการส่ง Message ระหว่างออบเจกต์ตามลำดับเวลาที่เกิดเหตุการณ์ขึ้น โดยจะมีสัญลักษณ์ให้เห็นลำดับการส่ง Message ตามเวลาส่งอย่างชัดเจน

2.5.7 Collaboration Diagram

เป็นแผนภาพที่แสดงให้เห็นถึงปฏิสัมพันธ์ (Interaction) ระหว่างออบเจกต์เช่นเดียวกับ Sequence Diagram แต่ต่างกันตรงที่ในส่วนของ Collaboration Diagram จะไม่มีสัญลักษณ์แสดงถึง

ลำดับการส่ง Message อย่างชัดเจน แต่จะเน้นส่วนของการแสดงถึงความสัมพันธ์ระหว่างอ็อบเจกต์ตามลักษณะการทำงาน

2.5.8 State-chart Diagram

เป็นแผนภาพที่แสดงให้เห็นถึงปฏิสัมพันธ์ (Interaction) ระหว่างอ็อบเจกต์เช่นเดียวกัน แต่ Statechart Diagram จะเน้นที่การแสดงให้เห็นถึงสถานะ (State) การเปลี่ยนสถานะ (Transition) ที่มีต่อเหตุการณ์ (Event) ที่เกิดขึ้นในช่วงชีวิตของอ็อบเจกต์ 1 ช่วง

2. 5.9 Activity Diagram

เป็นแผนภาพที่แสดงให้เห็นถึงลำดับการดำเนินการกิจกรรม (Activity) จากกิจกรรมหนึ่งไปยังกิจกรรมหนึ่งซึ่งเกิดจากการทำงานของอ็อบเจกต์ภายในระบบ