

ระบบบันทึกข้อมูลการซ่อมบำรุงรถยนต์



ศุภฤกษ์ วุฒิการณ

สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ วิทยาลัยนวัตกรรมด้านเทคโนโลยี และวิศวกรรมศาสตร์
มหาวิทยาลัยธุรกิจบัณฑิตย์

พ.ศ. 2563

Auto Service Cars

Supharoek Wuthikarn



**Thematic Paper Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Information Technology
College of Innovative Technology and Engineering
Dhurakij Pundit University**

2020

หัวข้อสารนิพนธ์	ระบบบันทึกข้อมูลซ่อมบำรุงรถยนต์
ชื่อผู้เขียน	ศุภฤกษ์ วุฒิการณ์
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. วรภัทร ไพรีเกรง
สาขาวิชา	เทคโนโลยีสารสนเทศ
ปีการศึกษา	2562

บทคัดย่อ

สารนิพนธ์ฉบับนี้ มีวัตถุประสงค์เพื่อ 1) ศึกษาและประยุกต์ใช้เทคโนโลยีการให้บริการการบันทึกการซ่อมบำรุงรถยนต์ผ่านเว็บแอปพลิเคชัน โดยใช้เอกสารตัวอย่างที่อยู่ในความรับผิดชอบของ ศูนย์บริการซ่อมบำรุงรถยนต์ สำหรับใช้ในการเก็บบันทึกข้อมูลสำหรับพนักงาน และลูกค้าที่รับบริการบริการ สามารถเข้าใช้งานระบบเว็บแอปพลิเคชันเพื่อความสะดวกรวดเร็ว

ในการประเมินผลการใช้งานระบบบันทึกการซ่อมบำรุงรถยนต์ ศูนย์บริการซ่อมบำรุงใช้แบบประเมินความพึงพอใจของผู้ใช้งานระบบ ประกอบด้วยผู้ใช้งานทั่วไปในศูนย์บริการซ่อมบำรุงรถยนต์ ประกอบไปด้วย พนักงานที่เกี่ยวข้อง 5 คน ทดลองใช้ระบบแล้วทำการประเมินผลความพึงพอใจด้านต่างๆ ผลสำรวจความพึงพอใจที่มีต่อระบบการบันทึกการซ่อมบำรุงรถยนต์โดยรวมอยู่ในระดับมาก

Thematic Paper Title	Auto Service System
Author	Supharoek Wuthikarn
Thematic Paper Advisor	Asst.Prof. Worapat Paireekreng; Ph.D.
Academic Program	Information Technology
Academic Year	2019

ABSTRACT

The purpose of this research is to: study and implement technology to car service maintenance via web application. The data has been collected from documents of Auto Service Centre which are recorded the information related to customer and services. User can access the information through Web Application conveniently.

To evaluate Auto Service Application, the questionnaire has been used to collect user's satisfaction. There are 5 administrators using this application. The results of survey showed that user's satisfaction was at high level.

ໄຊເສຍເອົາໜ້າຮຽນ
ຊຸມຊົນ ທ່ານຊື່

ຊຸມຊົນ ທ່ານຊື່



บทที่ 1

บทนำ

1.1 ความเป็นมา

เทคโนโลยีปัจจุบันมีความก้าวหน้ารวดเร็วมาก ซึ่งทำให้เกิดเทคโนโลยีใหม่ๆ ตอบสนองความสะดวกรวดเร็วในชีวิตประจำวัน สังคมในปัจจุบันที่มีการแข่งขันสูงขึ้นและทำงานแข่งกับเวลา ในมุมการตลาดมีการนำเทคโนโลยีเข้ามาเกี่ยวข้อง ในปัจจุบัน ผู้ใช้งานต้องการเข้าถึงข้อมูลได้อย่างรวดเร็วและสะดวกต่อการใช้งาน เทคโนโลยีที่ใช้กันอย่างแพร่หลายนั้นจะอยู่ในรูปแบบ Web Application ซึ่งสามารถเข้าถึงได้จากหลายช่องทางและอุปกรณ์ เช่น Web Browser , Smart Phone , Tablet , Notebook , PC ซึ่งสามารถเข้าถึงข้อมูลได้อย่างรวดเร็ว ไม่ว่าจะเป็นที่ไหน หรือ เวลาใด ปัจจุบันธุรกิจศูนย์ซ่อมบำรุงรถยนต์ ก็เป็นธุรกิจหนึ่งที่ต้องการทำงานที่มีความรวดเร็ว และมีความแม่นยำในด้านบันทึกข้อมูล ปัจจุบันศูนย์ซ่อมรถยนต์ทั่วไป มักจะเริ่มจากองค์กรมักจะใช้โปรแกรมสำเร็จรูปที่ซื้อมาจากผู้จำหน่าย Software ที่เป็นลักษณะใช้งานเพียงเครื่องเดียว (Stand alone) ทำให้เกิดปัญหาเครื่องคอมพิวเตอร์ที่ใช้งานบันทึกมีเพียงเครื่องเดียว ไม่สามารถลงเครื่องต่อไปและทำการเชื่อมต่อฐานข้อมูลได้ หากมีเครื่องคอมพิวเตอร์เครื่องอื่นที่ต้องการใช้โปรแกรมการบันทึกการซ่อมบำรุงรถยนต์ จะมีผลเสียคือจะมีฐานข้อมูลที่กระจัดกระจายออกไป ไม่เป็นระเบียบ ใช้งานยาก สืบค้นข้อมูลยาก เป็นอุปสรรคต่อคนทำงาน

ดังนั้นผู้พัฒนาระบบจึงเห็นความสำคัญที่จะทำเทคโนโลยีเข้ามาใช้เพื่อประสิทธิภาพในการทำงาน และประสิทธิภาพในการจัดเก็บข้อมูล โดยจะพัฒนาในรูปแบบของ Web Application เพื่อตอบสนองในการทำงานและความพึงพอใจแก่ลูกค้า

1.2 วัตถุประสงค์ของการพัฒนาระบบ

- เพื่อพัฒนาระบบการบันทึกข้อมูลการซ่อมบำรุงรถยนต์ ที่ผู้ปฏิบัติงานสามารถเข้าถึงด้วยอุปกรณ์ที่หลากหลาย
- เพื่อให้ผู้ปฏิบัติงานสามารถเข้าถึงข้อมูลได้อย่างรวดเร็ว
- เพื่อให้ผู้ปฏิบัติงานเก็บข้อมูล ใช้ฐานข้อมูลเดียวกัน เพื่อความ Integrity ของข้อมูล เพิ่มประสิทธิภาพในการติดต่อสื่อสารกับลูกค้า

1.3 ขอบเขตการดำเนินงาน

กลุ่มผู้ใช้งานระบบ

พนักงานตอบรับลูกค้า

- สามารถเพิ่มข้อมูลลูกค้าและบันทึกการซ่อมรถยนต์ได้
- สามารถดูสถานการณ์ซ่อมบำรุงรถยนต์ลูกค้าได้
- สามารถเรียกดูประวัติการซ่อมบำรุงรถยนต์ย้อนหลังได้
- สามารถเพิ่มงานซ่อมบำรุงรถยนต์ได้
- สามารถสรุปราคาซ่อมบำรุงรถยนต์ได้

1.4 วิธีการดำเนินงาน

1. ขั้นวางแผน

- ทำการรวบรวมข้อมูลที่เกี่ยวข้องในบริเวณพื้นที่ตัวอย่าง

2. ขั้นดำเนินงาน

- วิเคราะห์ ออกแบบระบบ จัดทำฐานข้อมูล พร้อมทั้งทำการออกแบบระบบ
- ดำเนินการพัฒนาระบบ เขียนคำสั่ง โปรแกรมประยุกต์ และทดสอบการใช้งาน
- จัดทำรายงานและคู่มือการพัฒนาระบบ

1.5 ระยะเวลาในการดำเนินงาน

กุมภาพันธ์ 2558 – มกราคม 2560

ตารางที่ 1.1 แผนการดำเนินงาน Grant Chart

ขั้นตอนการดำเนินงาน	กุมภาพันธ์ 2558 - มกราคม 2560											
	ก.พ. 58 - ก.ค. 58			ส.ค. 58 - ม.ค. 59			ก.พ. 59 - ม.ค. 59			ส.ค. 59 - ม.ค. 60		
1. ศึกษา สํารวจ และรวบรวมข้อมูลที่เกี่ยวข้อง	←→											
2. ศึกษาค้นคว้าความรู้การให้บริการยืม-คืนแฟ้มเอกสาร QR-Code และ โปรแกรมที่จำเป็นในการพัฒนาระบบ				←→								
3. วิเคราะห์ ออกแบบและจัดทำฐานข้อมูล							←→					
4. ออกแบบระบบ โครงสร้างและส่วนประกอบต่าง ๆ										←→		
5. ดำเนินการพัฒนาระบบ และทดสอบการใช้งาน										←→		
6. จัดทำรายงานและคู่มือการพัฒนาระบบ												←→

1.6 ประโยชน์ที่คาดว่าจะได้รับ

- เพื่อพัฒนาระบบการบันทึกข้อมูลการซ่อมบำรุงรถยนต์ที่ผู้ปฏิบัติงานสามารถเข้าถึงด้วยอุปกรณ์ที่หลากหลาย
- เพื่อให้ผู้ปฏิบัติงานสามารถเข้าถึงด้วยอุปกรณ์ที่หลากหลาย
- เพื่อให้ผู้ปฏิบัติงานเก็บข้อมูล ใช้ฐานข้อมูลเดียวกัน เพื่อความ Integrity ของข้อมูล เพิ่มประสิทธิภาพในการติดต่อสื่อสารกับลูกค้า

บทที่ 2

ทบทวนวรรณกรรม

การพัฒนาระบบการบันทึกข้อมูลการซ่อมบำรุงรถยนต์ จำเป็นต้องอาศัยความรู้ ความเข้าใจในเรื่องต่าง ๆ ซึ่งผู้พัฒนาได้ทำการรวบรวมทฤษฎี และวรรณกรรมที่เกี่ยวข้อง ดังนี้

- ระบบฐานข้อมูล (Database System)
- การวิเคราะห์ระบบ (System Analysis)
- ภาษา PHP (Personal Home Page)
- โมเดลเชิงความสัมพันธ์ของข้อมูล (E-R Model)
- ผังแสดงหน้าที่ (Use Case Diagram)

2.1 ระบบฐานข้อมูล (Database System)

ความหมายของระบบฐานข้อมูล

ฐานข้อมูล (database) หมายถึง กลุ่มของข้อมูลที่ถูกเก็บรวบรวมไว้ โดยมีความสัมพันธ์ซึ่งกันและกัน โดยไม่ได้บังคับว่าข้อมูลทั้งหมดนี้จะต้องเก็บไว้ในแฟ้มข้อมูลเดียวกันหรือแยกเก็บหลาย ๆ แฟ้มข้อมูล นั่นก็คือการเก็บข้อมูลในฐานข้อมูลนั้นเราอาจจะเก็บทั้งฐานข้อมูล โดยใช้แฟ้มข้อมูลเพียงแฟ้มข้อมูลเดียวกันได้ หรือจะเก็บไว้ในหลาย ๆ แฟ้มข้อมูล ที่สำคัญคือจะต้องสร้างความสัมพันธ์ระหว่างระเบียบและเรียกใช้ความสัมพันธ์นั้นได้ มีการกำจัดความซ้ำซ้อนของข้อมูลออกและเก็บแฟ้มข้อมูลเหล่านี้ไว้ที่ศูนย์กลาง เพื่อที่จะนำข้อมูลเหล่านี้มาใช้ร่วมกัน ควบคุมดูแลรักษาเมื่อผู้ต้องการใช้งานและผู้มีสิทธิ์จะใช้ข้อมูลนั้นสามารถดึงข้อมูลที่ต้องการออกไปใช้ได้ ข้อมูลบางส่วนอาจใช้ร่วมกับผู้อื่นได้ แต่บางส่วนผู้มีสิทธิ์เท่านั้นจึงจะสามารถใช้ได้ โดยทั่วไปองค์กรต่าง ๆ จะสร้างฐานข้อมูลไว้เพื่อเก็บข้อมูลต่าง ๆ ของตัวองค์กร โดยเฉพาะอย่างยิ่งข้อมูลในเชิงธุรกิจ เช่น ข้อมูลของลูกค้า ข้อมูลของสินค้า ข้อมูลของลูกจ้าง และการจ้างงาน เป็นต้น การควบคุมดูแลการใช้ฐานข้อมูลนั้น เป็นเรื่องที่ยากกว่าการใช้แฟ้มข้อมูลมาก เพราะเราจะต้องตัดสินใจว่าโครงสร้างในการจัดเก็บข้อมูลควรจะเป็นเช่นไร การเขียนโปรแกรมเพื่อสร้างและ

เรียกใช้ข้อมูลจากโครงสร้างเหล่านี้ ถ้าโปรแกรมเหล่านี้เกิดทำงานผิดพลาดขึ้นมา ก็จะเกิดความเสียหายต่อโครงสร้างของข้อมูลทั้งหมดได้ เพื่อเป็นการลดภาวะการทำงานของผู้ใช้ จึงได้มีส่วนของฮาร์ดแวร์และโปรแกรมต่าง ๆ ที่สามารถเข้าถึงและจัดการข้อมูลในฐานข้อมูลนั้น เรียกว่า ระบบจัดการฐานข้อมูล หรือ DBMS (data base management system) ระบบจัดการฐานข้อมูล คือซอฟต์แวร์ที่เปรียบเสมือนสื่อกลางระหว่างผู้ใช้และโปรแกรมต่าง ๆ ที่เกี่ยวข้องกับการใช้ฐานข้อมูล ซึ่งมีหน้าที่ช่วยให้ผู้ใช้เข้าถึงข้อมูลได้ง่ายสะดวกและมีประสิทธิภาพ การเข้าถึงข้อมูลของผู้ใช้จะเป็นการสร้างฐานข้อมูล การแก้ไขฐานข้อมูล หรือการตั้งคำถามเพื่อให้ข้อมูลมา โดยผู้ใช้ไม่จำเป็นต้องรับรู้เกี่ยวกับรายละเอียดภายในโครงสร้างของฐานข้อมูล เปรียบเสมือนเป็นสื่อกลางระหว่างผู้ใช้และโปรแกรมต่าง ๆ ที่เกี่ยวข้องกับการใช้ฐานข้อมูล

2.1.1 คุณสมบัติของฐานข้อมูลที่ดี

วัตถุประสงค์ของการออกแบบฐานข้อมูลก็เพื่อต้องการให้ได้ข้อมูลที่ดีซึ่งต้องมีคุณสมบัติดังนี้

1. ไม่ซับซ้อน (Simplicity) เข้าใจได้ง่ายสำหรับผู้ที่ เป็นบุคลากรด้านคอมพิวเตอร์และผู้ใช้ทั่วไป
2. อธิบายได้ชัดเจน (Express ability) สามารถอธิบายโครงสร้างของข้อมูลไม่ว่าจะเป็นความหมายของข้อมูล ความสัมพันธ์และข้อกำหนด (Constraint) ต่าง ๆ ได้อย่างชัดเจน
3. ลดความซ้ำซ้อน (Non Redundancy) ข้อมูลหนึ่ง ๆ จะมีเพียงแห่งเดียว จะไม่ปรากฏอยู่หลายแห่งในฐานข้อมูล
4. น่าเชื่อถือหรือมีความบูรณาภาพ (Integrity) ข้อมูลที่ออกแบบมีความถูกต้องของข้อมูลตามข้อกำหนดขององค์กร
5. ข้อมูลที่มีอยู่คือข้อมูลที่ต้องการ (Minimalist) ฐานข้อมูลที่ดีต้องประกอบด้วยข้อมูลที่องค์กรต้องการใช้งานอย่างครบถ้วนและต้องไม่รวมข้อมูลที่ไม่ต้องการใช้งานอยู่ในฐานข้อมูล
6. ขยายขอบเขตได้ (Extensibility) สามารถปรับขยายและรองรับการขยายตัวของการใช้งานข้อมูลได้

2.1.2 รูปแบบของฐานข้อมูล

1. ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) เป็นการเก็บข้อมูลในรูปแบบตาราง (Table) ในแต่ละตารางแบ่งออกเป็นแถวๆ และในแต่ละส่วนจะแบ่งคอลัมน์ (Column) ซึ่งในการเชื่อมโยงกันระหว่างข้อมูลในตารางต่างๆ จะเชื่อมโยงโดยใช้การอ้างอิงจากข้อมูลในคอลัมน์ที่กำหนดไว้ Relational Database เป็นฐานข้อมูลที่ใช้โมเดลเชิงสัมพันธ์ (Relational Database Model) เนื่องจากแนวคิดแบบจำลองมีลักษณะที่คนใช้กันทั่วๆไปคือ มีการเก็บเป็นตาราง ทำให้ง่ายต่อการ

เข้าใจและการประยุกต์ใช้งาน ด้วยเหตุนี้ ระบบฐานข้อมูลในรูปแบบตาราง และ attribute ก็เปรียบเหมือนข้อมูลส่วนความสัมพันธ์คือความสัมพันธ์ระหว่าง entity

2. ฐานข้อมูลแบบลำดับชั้น (Hierarchical Relational) ข้อมูลจะถูกเก็บไว้เป็นโครงสร้างแบบล่าง ลักษณะคล้ายๆ โครงสร้างต้นไม้ (Tree Structure) มีการสืบทอดเป็นลำดับชั้น โหนดสูงสุดจะเรียกราก (Root) โหนดระดับล่างลงมาจะเรียกว่า (Leaves) เรียกฐานข้อมูลในระดับนี้อีกชื่อหนึ่งว่า (Segment) เปรียบได้กับเรคคอร์ดในระบบแฟ้มข้อมูล เซกเมนต์ที่อยู่ระดับล่างลงไปก็คือเซกเมนต์ก่อนหน้า ความสัมพันธ์ 1:M กล่าวคือ โหนดบนจะแตกโหนดลูกได้หลายๆ โหนด ในขณะที่โหนดลูกจะมีโหนดบนเพียงโหนดเดียวกัน

3. ฐานข้อมูลแบบเครือข่าย (Network Relational) ลักษณะโครงสร้างระบบฐานข้อมูลแบบเครือข่ายจะมีโครงสร้างของข้อมูลแต่ละแฟ้มข้อมูลมีความสัมพันธ์คล้ายร่างแห โดยมีลักษณะโครงสร้างคล้ายกับโครงสร้างแบบลำดับชั้น มีข้อแตกต่างที่ว่าโครงสร้างแบบเครือข่ายสามารถยินยอมให้ระดับชั้นที่อยู่เหนือกว่าจะมีได้หลายแฟ้มข้อมูลถึงแม้ว่าระดับชั้นถัดลงมาจะมีเพียงแฟ้มข้อมูลเดียว เปรียบเสมือนมีความสัมพันธ์แบบลูกจ้างกับงานที่ทำ โดยงานชิ้นหนึ่งอาจทำโดยลูกจ้างหลายคน (m ต่อ n)

4. ฐานข้อมูลเชิงวัตถุ (The Object-Oriented Database Model) ปัจจุบันเป็นยุคโลกข้อมูลสาร การสื่อสารข้อมูลข่าวสารมีความซับซ้อนและเพิ่มปริมาณการสื่อสารมากขึ้น อีกทั้งรูปแบบของข้อมูลมีความซับซ้อนและมีขนาดใหญ่ขึ้น ตัวอย่างเช่น รูปภาพ (Image) หรือภาพกราฟิกส์ (Graphics) ข้อมูลการทดลองวิทยาศาสตร์ที่ต้องเก็บตัวเลขทศนิยมเป็นจำนวนมาก ข้อมูลทางภูมิศาสตร์ หรือสื่อมัลติมีเดีย เป็นต้น จึงมีผู้คิดค้นแบบจำลองข้อมูลขึ้นมาใหม่ โดยมองระบบการจัดเก็บข้อมูลเหล่านั้นใหม่โดยมองว่าข้อมูลเหล่านั้นเป็น วัตถุ หรือ ออบเจกต์ โดยภายในวัตถุเก็บทั้งข้อมูลและความสัมพันธ์ระหว่างวัตถุรวมไว้ในวัตถุนั้นเอง โดยมีการกำหนด โครงสร้างข้อมูลต้นแบบของวัตถุเรียกว่า คลาส และเรียกฐานข้อมูลนี้ว่า ฐานข้อมูลเชิงวัตถุ (Object Oriented Database) และโปรแกรมที่ทำหน้าที่บริหารจัดการจัดเก็บวัตถุต่าง เรียกว่า ระบบ จัดการฐานข้อมูลเชิงวัตถุ Object-Oriented Database Management System (OODBMS)

2.2 การวิเคราะห์ระบบสารสนเทศ (System Analysis)

ความรู้ในการวิเคราะห์และออกแบบระบบมีความสำคัญ เพราะเป็นปัจจัยในการสร้างและพัฒนาระบบสารสนเทศ การวิเคราะห์ระบบเป็นเรื่องที่น่าสนใจ เพราะนักวิเคราะห์ระบบต้องติดต่อกับคนหลายคน ได้รู้ถึงการจัดการและการทำงานในองค์กร ทำให้เรามีความรู้เกี่ยวกับระบบคอมพิวเตอร์หลายแบบมากขึ้น ผู้ที่สามารถวิเคราะห์ระบบได้ดี ควรมีประสบการณ์ในการเขียนโปรแกรม มีความรู้ทางด้านธุรกิจ ความรู้เกี่ยวกับระบบเครือข่ายและฐานข้อมูล ซึ่งใช้เป็นความรู้ในการออกแบบระบบที่มีความแตกต่างกันออกไปตาม สภาพงาน ดังนั้น หน้าที่ของนักวิเคราะห์ ก็คือการศึกษาระบบ แล้วให้คำแนะนำในการปรับปรุงและพัฒนาระบบนั้นจนเสร็จสมบูรณ์ ซึ่งการทำงานทั้งหมดต้องมีลำดับขั้นตอนและการศึกษาวิธีการวิเคราะห์และการออกแบบระบบในแต่ละขั้นตอน ทำให้เข้าใจการวิเคราะห์ระบบนั้นๆ ดียิ่ง และสามารถออกแบบระบบใหม่โดยไม่ยากเย็นนัก โดยสามารถตัดสินใจว่า ระบบใหม่ควรใช้คอมพิวเตอร์ประเภทไหน ใช้โปรแกรมอะไร ออกแบบInput/Output อย่างไร เป็นต้น

ระบบคือกลุ่มขององค์ประกอบต่างๆ ที่ทำงานร่วมกันเพื่อจุดประสงค์อันเดียวกัน ระบบอาจจะประกอบด้วย บุคคลากร เครื่องมือ เครื่องใช้ พัสตุ วิธีการ ซึ่งทั้งหมดนี้จะต้องมีระบบจัดการอันหนึ่ง เพื่อให้บรรลุจุดประสงค์อันเดียวกัน เช่น ระบบการเรียนการสอน มีจุดประสงค์เพื่อให้ นักเรียนได้รับความรู้ในเนื้อหาวิชาที่สอน

การวิเคราะห์ระบบและการออกแบบ (System Analysis and Design) การวิเคราะห์และออกแบบระบบคือ วิธีการที่ใช้ในการสร้างระบบสารสนเทศขึ้นมาใหม่ในธุรกิจใดธุรกิจหนึ่งหรือระบบย่อยของธุรกิจ นอกจากการสร้างระบบสารสนเทศใหม่แล้ว การวิเคราะห์ระบบ ช่วยในการแก้ไขระบบสารสนเทศเดิมที่มีอยู่แล้วให้ดีขึ้นด้วยก็ได้ การวิเคราะห์ระบบก็คือ การหาความต้องการ (Requirements) ของระบบสารสนเทศว่าคืออะไร หรือต้องการเพิ่มเติมอะไรเข้ามาในระบบ และการออกแบบก็คือ การนำเอาความต้องการของระบบมาเป็นแบบแผน หรือเรียกว่าพิมพ์เขียวในการสร้างระบบสารสนเทศนั้นให้ใช้งานได้จริง ตัวอย่างระบบสารสนเทศ เช่น ระบบการขาย ความต้องการของระบบก็คือ สามารถติดตามยอดขายได้เป็นระยะ เพื่อฝ่ายบริหารสามารถปรับปรุงการขายได้ทันทั่วถึง ตัวอย่างรายงานการขายที่กล่าวมาแล้วจะชี้ให้เห็นว่าเราสามารถติดตามการขายได้อย่างไร

2.3 วงจรการพัฒนากระบวนสารสนเทศ (System Development Life Cycle SDLC)

วงจรการพัฒนากระบวน (System Development Life Cycle-SDLC) กระบวนสารสนเทศทั้งหลายมีวงจรชีวิตที่เหมือนกัน ตั้งแต่เกิดจนตาย วงจรนี้จะเป็นขั้นตอนที่เป็นลำดับตั้งแต่ต้นจนเสร็จเรียบร้อยเป็นระบบที่ใช้งานได้ ซึ่งนักวิเคราะห์ระบบ ต้องทำความเข้าใจให้ได้ว่าในแต่ละขั้นตอนจะต้องทำอะไร และทำอะไร ขั้นตอนการพัฒนากระบวนมีอยู่ด้วยกัน 7 ขั้นตอนคือ

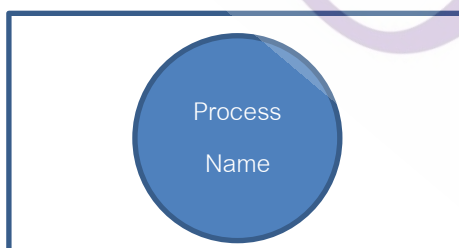
1. เข้าใจปัญหา (Problem Recognition)
2. ศึกษาความเป็นไปได้ (Feasibility Study)
3. วิเคราะห์ (Analysis)
4. ออกแบบ (Design)
5. สร้าง หรือพัฒนากระบวน (Construction)
6. การปรับเปลี่ยน (Conversion)
7. บำรุงรักษา (Maintenance)

แผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูล (DFD) เป็นเครื่องมือที่ใช้กันอย่างแพร่หลายในการเขียนแบบระบบใหม่ โดยเฉพาะกับระบบที่ "หน้าที" ของระบบมีความสำคัญและมีความสลับซับซ้อนมากกว่าข้อมูลที่ไหลเข้า

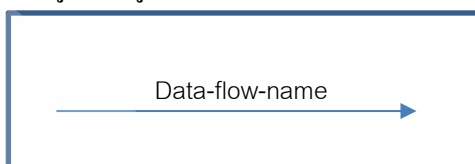
DFD มีองค์ประกอบ 4 อย่าง ซึ่งใช้สัญลักษณ์ต่าง ๆ แทนดังต่อไปนี้

1. สัญลักษณ์แทนการประมวลผล (Process) เป็นวงกลม



ภาพที่ 2.1 แทนการประมวลผล

2. สัญลักษณ์แทนกระแสข้อมูลเป็นลูกศร



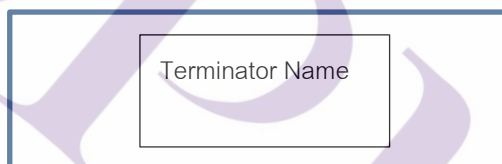
ภาพที่ 2.2 แทนกระแสข้อมูลเป็นลูกศร

3. สัญลักษณ์แทนแหล่งเก็บข้อมูลเป็นเส้นขนาน 2 เส้น โดยมีชื่อกำกับ



ภาพที่ 2.3 แทนแหล่งเก็บข้อมูลเป็นเส้นขนาน 2 เส้น

4. สัญลักษณ์พื้นผ้าเป็นสัญลักษณ์แทนสิ่งที่ยู่ในระบบ



ภาพที่ 2.4 สัญลักษณ์พื้นผ้าเป็นสัญลักษณ์แทนสิ่งที่ยู่ในระบบ

การประมวลผลโพรเซส (Process) คือ งานที่จะต้องทำแทนด้วยวงกลมและมีชื่ออยู่ภายใน วงกลมการประมวลผลจะเปลี่ยนข้อมูลขาเข้าเป็นผลลัพธ์ นั่นหมายความว่า จะต้องมีกรกระทำ บางอย่างต่อข้อมูลทำให้เกิดผลลัพธ์ขึ้นมา โดยปกติแล้วข้อมูลที่นำเข้าสู่โพรเซสจะแตกต่าง จากข้อมูลเมื่อออกจากโพรเซส

โพรเซสเป็นตัวอย่างหนึ่งของ "กล่องดำ" หมายถึงว่า เราทราบว่ามีข้อมูลเป็นอะไร ผลลัพธ์ อะไรที่เราต้องการ และหน้าที่โดยทั่วไปของโพรเซส แต่จะไม่ทราบว่าโพรเซสนั้นทำงานอย่างไร หลักการของกล่องดำมีประโยชน์ในการเขียนแผนภาพแสดงการเปลี่ยนแปลงของข้อมูล โดยที่ยังไม่ ต้องทราบในรายละเอียดว่าโพรเซสนั้นมีรายละเอียดอะไรบ้าง ซึ่งสามารถหารายละเอียดเหล่านี้ได้ ในภายหลัง

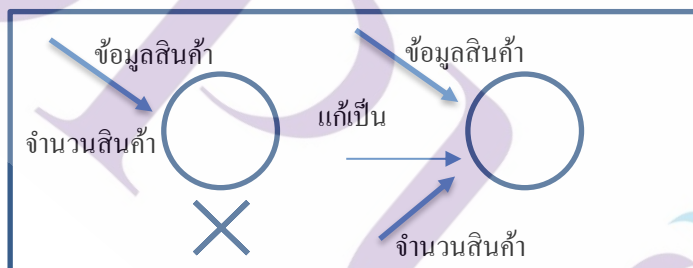
ชื่อโพรเซส เป็นตัวบอกว่าโพรเซสนั้นทำหน้าที่อะไร คำที่ใช้ควรมีความหมายที่แน่นอน ควรจะใช้คำกริยา เช่น คำนวณ แก้ไข พิมพ์ เป็นต้น ถ้าการทำงานใดที่เราไม่สามารถ หาคำแทนได้ อย่างเหมาะสม อาจจะหมายความว่า งานนั้นๆ ไม่ใช่โพรเซสก็ได้

กระแสข้อมูล (Data Flow) กระแสข้อมูลแทนด้วยลูกศร โดยที่มีชื่อข้อมูลกำกับอยู่บนลูกศร นั้น



ภาพที่ 2.5 กระแสข้อมูลแทนด้วยลูกศร

ข้อมูลที่ไหลระหว่างโปรเซสต่าง ๆ และอาจเคลื่อนที่มาจากสิ่งที่อยู่นอกระบบก็ได้ ข้อมูลที่เคลื่อนที่อาจจะเป็นเพียงข้อมูลเดี่ยวๆ เช่น เลขที่สินค้า หรือกลุ่มของข้อมูล เช่น ข้อมูล พนักงาน ข้อมูลลูกค้า เป็นต้น กลุ่มของข้อมูลควรจะเป็นเรื่องเดียวกัน หรือสัมพันธ์กัน ตัวอย่างเช่น ข้อมูลลูกค้าอาจจะมีรายละเอียดเป็นชื่อลูกค้า เลขที่ ที่อยู่ แต่ไม่ควรรวมจำนวน สินค้าในคลังอยู่ในข้อมูลเดียวกัน ถ้าต้องการอ้างอิงข้อมูลทั้งสองที่ไม่เกี่ยวข้องกันให้เขียนแยกเป็นลูกศร 2 อัน

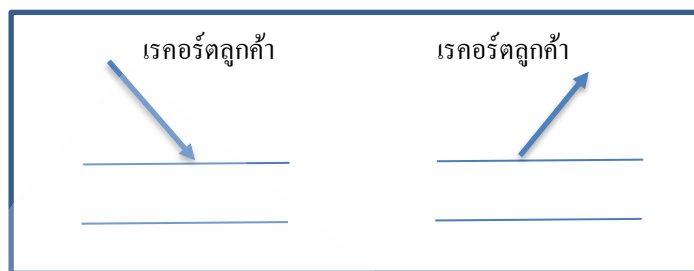


ภาพที่ 2.6 ข้อมูล 2 อันไม่เหมือนกันจะต้องแยกลูกศรออก

ข้อมูล 2 อันไม่เหมือนกันจะต้องแยกลูกศรออก

ข้อมูลแต่ละอันหรือกลุ่มข้อมูล ควรจะมีชื่อของตัวเองที่ไม่เหมือนกัน ควรหลีกเลี่ยงใช้ชื่อที่กว้างเกินไป เช่น "ข้อมูลตลาด" เพราะว่ามีในระบบหนึ่งๆ อาจจะมี "ข้อมูลตลาด" เกิดขึ้นหลายๆ แห่ง เราควรใช้ชื่อเฉพาะเจาะจงมากกว่านี้ เช่น "เลขที่ลูกค้าไม่ถูกต้อง" "ไม่มีสินค้านี้ในคลัง" หรือ "ไม่มีสินค้าในคลัง" เป็นต้น ในระบบใหญ่ๆ ต้องแยก รายละเอียดเหล่านี้ออกให้ชัดเจน

แหล่งเก็บข้อมูล (Data Store) แทนด้วยเส้นขนานสองเส้นและมีชื่อกำกับ ข้อมูลจะถูกเก็บในไฟล์และถูกเรียกใช้เมื่อต้องการ โดยปกติแล้วไฟล์อาจจะอยู่ในงานแม่เหล็กหรือเทปแม่เหล็ก ถ้าหัวลูกรังเข้าสู่ไฟล์แสดงว่า มีการเขียนข้อมูลหรือการแก้ไขข้อมูลในไฟล์ดังในรูปข้างล่างนี้ ถ้าลูกรังออกจากไฟล์แสดงว่ามีการอ่านข้อมูล การตั้งชื่อไฟล์ควรเป็นคำนาม



ภาพที่ 2.7 การแก้ไขข้อมูล

การแก้ไขข้อมูล

สิ่งที่อยู่นอกระบบ (Terminator) สิ่งที่อยู่นอกระบบแทนด้วยสี่เหลี่ยมผืนผ้า ซึ่งมีชื่อกำกับอยู่ด้วย ส่วนใหญ่จะเป็นตัวบุคคล หรือองค์กรต่างๆ สิ่งที่อยู่นอกระบบอาจจะเป็นที่ส่งข้อมูลเข้าสู่ระบบ หรืออาจจะเป็นที่รับข้อมูลจากระบบก็ได้ เราไม่สนใจการทำงานภายในของสิ่งที่อยู่นอกระบบ ถึงแม้ว่าจะมีการติดต่อผ่านทางข้อมูล เราจะสนใจเฉพาะข้อมูลที่เข้าสู่ระบบหรือออกจากระบบสู่ภายนอกเท่านั้น

เมื่อเราทราบส่วนประกอบของการเขียนแผนภาพ DFD แล้ว ลองเอาสัญลักษณ์เหล่านี้มาเขียนรวมกันเป็น DFD ของระบบทั้งระบบดังนี้

โพรเซสกับภาพรวมของ DFD

ภาพรวมของ DFD โดยทั่วไปมักจะมีโพรเซสทั้งหมดด้วยกัน 5 โพรเซส โดยมีเลขที่กำกับด้วย แต่ละโพรเซสทำงานของตัวเองแยกกัน ปัญหาของการเขียนโพรเซสคือ ทำอย่างไร จึงจะ "แบ่ง" งานออกจากกันได้ ในตัวอย่างอาจจะลบโพรเซสที่ 3 ออกแล้วรวมเอาไว้ในโพรเซสที่ 4 ก็ได้ หรือจะดึงงานบางส่วนในโพรเซสที่ 1 ไปรวมกับโพรเซสที่ 2 ก็ได้ อีก เช่นกัน การแบ่งจำนวนงานนั้นไม่มีคำตอบว่า "ถูกหรือผิด" ที่แน่นอนตายตัว แต่คำตอบหนึ่งอาจจะดีกว่าคำตอบหนึ่งก็ได้ เราอาจจะแบ่งการทำงานใหม่ซึ่งจะทำให้ระบบนั้นดีขึ้นหรือเลวลง

การแบ่งจำนวนโพรเซสใน DFD ไม่มีกฎเกณฑ์ตายตัว การแบ่งจำนวนนี้ขึ้นอยู่กับ "ความชำนาญหลังจากที่มี ประสบการณ์มาพอสมควร" ถ้าเปรียบเทียบการเขียนโปรแกรมก็เหมือนกับการแยกเขียนโปรแกรมย่อยนั่นเอง ซึ่งจะต้องอาศัยประสบการณ์ในการเขียนโปรแกรมมาช่วยมากทีเดียว ปัญหาการแบ่งงานก็คือ ขอบเขตของงานนั่นเอง

วิธีการสร้าง DFD

ทั้งหมดนี้เป็นขั้นตอนในการสร้าง DFD ที่มีระบบมากขึ้น

1. กำหนดสิ่งที่อยู่ภายนอกทั้งหมด และหาว่าข้อมูลอะไรบ้างที่เข้าสู่ระบบหรือออกจากระบบที่เราสนใจสู่ระบบที่อยู่ภายนอก ขั้นตอนนี้สำคัญมากทั้งนี้เพราะจะทำให้ทราบว่าขอบเขตของ

ระบบนั้นมีอะไรบ้าง

2. ใช้ข้อมูลที่ได้จากขั้นตอนที่ 1 นำมาสร้าง DFD ต่างระดับ
3. ขั้นตอนถัดมาอีก 4 ขั้นตอนโดยให้ทำทั้ง 4 ขั้นตอนนี้ซ้ำๆหลายๆครั้งจนกระทั่ง

ได้ DFD ระดับต่ำสุด

3.1 เขียน DFD ฉบับแรกกำหนดโพรเซสและข้อมูลที่ไหลออกจากโพรเซส

3.2 เขียน DFD อื่นๆที่เป็นไปได้จนกระทั่งได้ DFD ที่ถูกที่สุด ถ้ามีส่วนหนึ่งส่วนใด ที่รู้สึกว่าไม่ถนัดนักก็ให้พยายามเขียนใหม่อีกครั้งหนึ่งแต่ไม่ควรเสียเวลาเขียนจนกระทั่งได้ DFD ที่สมบูรณ์แบบเลือก DFD ที่เห็นว่าดีที่สุดในสายตาของเรา

3.3 พยายามหาว่ามีข้อผิดพลาดอะไรหรือไม่ซึ่งมีรายละเอียดในหัวข้อ "ข้อผิดพลาดใน DFD"

3.4 เขียนแผนภาพแต่ละภาพอย่างดีซึ่ง DFD ฉบับนี้จะใช้ต่อไปในการออกแบบและใช้ด้วยกันกับบุคคลอื่นๆที่เกี่ยวข้องในโครงการด้วย

4. นำแผนภาพทั้งหมดที่เขียนแล้วมาเรียงลำดับทำสำเนาและพร้อมที่จะนำไปตรวจสอบข้อผิดพลาดจากผู้ร่วมทีมงานถ้ามีแผนภาพใดที่มีจุดอ่อนให้กลับไปเริ่มต้นที่ขั้นตอนที่ 3 อีกครั้งหนึ่ง

5. นำ DFD ที่ได้ไปตรวจสอบข้อผิดพลาดกับผู้ใช้ระบบเพื่อหาว่ามีแผนภาพใดไม่ถูกต้องหรือไม่

6. ผลิตแผนภาพฉบับสุดท้ายทั้งหมด

ข้อผิดพลาดใน DFD

การเขียน DFD อาจเขียนได้หลายแบบ ผลลัพธ์ฉบับสุดท้ายอาจจะไม่เหมือนกันถ้าเขียนโดยนักวิเคราะห์ระบบคนละคน ถึงอย่างไรแนวทางการเขียน DFD ซึ่งจะช่วยให้ เราเขียน DFD ได้ถูกต้องมากขึ้นก็มีอยู่บ้าง ซึ่งพอสรุปได้ดังนี้

1. ถ้า DFD ซับซ้อนมาก ทุกๆ นิ้วในกระดาษถูกใช้งานทั้งหมด แสดงว่า DFD นั้นควรจะแยกย่อยไปอีกระดับหนึ่งหรือมากกว่าหนึ่ง

2. ข้อมูลที่ออกจากโพรเซส หรือผลลัพธ์มีข้อมูลขาเข้าไม่เพียงพอ เราจะต้องพิจารณาแผนภาพต่อไปอีกแต่ที่สำคัญไม่ควรใส่ข้อมูลที่ไม่เคยใช้เข้ามาในโพรเซสเป็นอันขาด

3. การตั้งชื่อโพรเซสนั้นไม่ถนัดนักอาจจะมีปัญหา 2 อย่างคือ โพรเซสนั้นควรจะแยกออกเป็น 2 ส่วน หรือเราไม่ทราบว่ามื่ออะไรเกิดขึ้นบ้างในโพรเซสนั้นๆในกรณีนี้เราต้องศึกษาระบบให้ละเอียดยิ่งขึ้น

4. จำนวนระดับในแต่ละแผนภาพแตกต่างกันมากเช่น โพรเซสที่ 1 มีลูก 2 ชั้นแต่โพรเซส

ที่ 2 มีลูก 10 ชั้น แสดงว่าการแบ่งจำนวนโพรเซสไม่คืนักจำนวนลูกของโพรเซสไม่จำเป็นต้องเท่ากัน แต่ไม่ควรจะแตกต่างกันมากนัก

5. มีการแตกแยกย่อยข้อมูล รวมตัวของข้อมูลหรือมีการตัดสินใจในโพรเซส แสดงว่าโพรเซสนั้น ไม่ถูกต้องการแยกข้อมูลหรือรวมตัวของข้อมูลเป็นหน้าที่ของพจนานุกรมข้อมูลการตัดสินใจ เป็นรายละเอียดอยู่ในคำอธิบายโพรเซส

การสร้าง DFD ที่ดีเป็นงานที่ยากที่สุดสำหรับนักวิเคราะห์ระบบมือใหม่หรือแม้แต่ผู้ที่มีประสบการณ์มาแล้วก็ตาม DFD ที่ไม่ดีจะทำให้ผลลัพธ์สุดท้ายของระบบออกมาไม่ดีเช่นเดียวกัน ทั้งนี้เนื่องจาก DFD เป็นรากฐานสำหรับการออกแบบและพัฒนาโปรแกรม

การออกแบบระดับกายภาพ

การออกแบบในระดับกายภาพแตกต่างจากระดับตรรกะในแง่ของการแสดงขั้นตอนของระบบ โดยจะให้ความสำคัญเกี่ยวกับรายละเอียดของข้อมูล ผลลัพธ์ และการประมวลผล รวมถึงชนิดของสื่อที่ใช้ในการบรรจุข้อมูลด้วย

มาดูตัวอย่างการออกแบบที่ติดกันสักตัวอย่างหนึ่ง ได้แก่โปรแกรมในการคำนวณบัญชีเงินเดือนของบริษัทแห่งหนึ่ง ซึ่งผลที่ได้จากการคำนวณนี้จะถูกนำไปปรับค่าข้อมูล ในไฟล์หลักของข้อมูลพนักงานด้วยดังแสดงไว้ด้วยผังงานระบบ(System Flowchart) การออกแบบระบบในระดับนี้จะระบุถึงข้อมูล และผลลัพธ์ที่ได้จากการทำงาน รวมถึงขั้นตอน ในการทำงานในกรณีที่เกิดข้อผิดพลาดขึ้นความสำคัญของผังงานระบบนี้เปรียบเสมือนเป็น "พิมพ์เขียว" ที่จะใช้ในการพัฒนาระบบต่อไป

การออกแบบฟอร์ม

การออกแบบรูปแบบรายงาน รูปแบบข้อมูลขาเข้า และรูปแบบหน้าจอ เป็นเรื่องสำคัญทีเดียว เพราะทั้งหมดคือ สิ่งที่ใช้เห็นได้ง่ายที่สุด และเป็นสิ่งที่ติดต่อระหว่างผู้ใช้กับ ระบบทั้งหมด และผู้ใช้จะใช้สิ่งที่เห็นเป็นตัวช่วยตัดสินใจว่าระบบดีหรือไม่ ถ้าสิ่งที่ผู้ใช้เห็น ไม่ว่าจะเป็นรายงานหรือหน้าจอคอมพิวเตอร์ ถ้าดูไม่สวยงามดูวุ่นวายทำให้ผู้ใช้ไม่พอใจ ถึงแม้ว่าในระบบจริงๆ จะทำงานได้ดีมากก็ตาม พูดอีกนัยหนึ่งก็คือ ระบบของเราควรมีลักษณะที่ว่า "สวยทั้งรูป จูบก็หอม" นั่นเอง

การแบบของกระดาษ

กระดาษที่ใช้พิมพ์รายงานมีอยู่ 2 ประเภทคือ กระดาษธรรมดา และกระดาษที่พิมพ์ข้อความไว้แล้ว(Preprinted forms) กระดาษธรรมดาก็คือ กระดาษเปล่าๆ และเป็นชนิด ต่อเนื่องซึ่งมีอยู่ด้วยกันหลายขนาด และอาจจะทำเป็นหลายก๊อปปี้ด้วยก็ได้ สำหรับกระดาษที่พิมพ์ข้อความไว้แล้วนั้นจะมีข้อความบางข้อความที่พิมพ์ไว้ก่อนแล้วซึ่งข้อความเหล่านี้ปกติจะไม่เปลี่ยนแปลง

ตัวอย่างเช่น เซ็ทซึ่งจะมีชื่อธนาการพิมพ์อยู่ในทุกๆ ใบ ซึ่งเป็นข้อความที่ไม่เปลี่ยนแปลง และจะมีข้อความบางส่วนที่จะพิมพ์เพิ่มเติมลงไป เช่น ชื่อผู้รับเงิน เป็นต้น

การออกแบบรายงาน

ปกติเวลาเราออกแบบรายงานเราจะใช้แบบฟอร์มที่มีตาราง (Spacing Chart) ซึ่งมีลักษณะเป็นช่องๆ นำมากรอกข้อความที่ต้องการจะพิมพ์ ตัวอย่างแบบฟอร์มตารางที่ใช้ในการ ออกแบบรายงาน เราจะใส่ข้อความที่ไม่มีการเปลี่ยนแปลง เช่น ชื่อรายงานในแบบฟอร์มลงในตำแหน่งที่เราต้องการให้ข้อความปรากฏในรายงาน ในกรณีที่ข้อมูลที่เปลี่ยนแปลงได้ เราจะใช้ตัวอักษร "X" แทนตัวหนังสือ และ "9" แทนตัวเลข แล้วจึงเติมจำนวนบรรทัดที่ต้องการพิมพ์ในหน้านั้นๆ นอกจากนั้นอาจจะใส่ตัวอักษรพิเศษต่างๆ เช่น \$, I, - หรือ . ด้วยก็ได้ เราจะกรอกข้อมูลเฉพาะบรรทัดแรกและบรรทัด สุดท้ายเท่านั้น ระหว่าง 2 บรรทัดนั้นจะร่างเส้นโค้งเพื่อเชื่อมโยง ซึ่งจะหมายความว่ารูปแบบข้อมูลตรงกลางจะเหมือนกัน กับบรรทัดแรกและบรรทัดสุดท้ายนั้น รูปแบบข้อมูลตรงกลางจะเหมือนกันกับบรรทัดแรกและบรรทัดสุดท้ายนั่นเอง หัวเรื่องของรายงานอาจจะประกอบด้วยชื่อรายงาน วันที่ที่พิมพ์ เลขที่หน้าของ รายงาน ข้อมูลในแต่ละแถวควรมีช่องว่างแทรกเพื่อให้ดูรายละเอียดได้สบายตามากขึ้นและดูไม่แน่นจนเกินไป

การออกแบบจอแสดงผล

จอภาพของคอมพิวเตอร์เป็นไปได้อื่นๆ รับข้อมูลและแสดงผลลัพธ์ จอภาพมีประโยชน์สำหรับแสดงผลลัพธ์ ในกรณีที่เราไม่ต้องการพิมพ์รายงานบนกระดาษ แต่ต้องการดูผลอะไรบางอย่าง เช่น ดูสถานะเครดิตลูกค้าเฉพาะราย เป็นต้น นอกจากนั้นปัจจุบันเราก็นิยมพิมพ์ข้อมูลเข้าหรืออินพุตผ่านทางหน้าจอเช่นป้อนข้อมูลการเคลื่อนไหวของสินค้า เป็นต้นปกติหน้า

จอคอมพิวเตอร์มีขนาด 80*25 (25 บรรทัด บรรทัดละ 80 ตัวอักษร) ดังนั้นเราจะมีจำนวนจำกัดในการแสดงข้อความบนจอ แบบฟอร์มจอภาพ (Screen Layout) (คล้าย ๆ Spacing Chart ของการออกแบบรายงาน) จะช่วยในการออกแบบจอภาพ โดยที่เราจะกรอกข้อความที่จะให้ปรากฏจอคอมพิวเตอร์ในแบบฟอร์มนี้

การพัฒนาโปรแกรมและการบำรุงรักษา

หลักการวิเคราะห์และออกแบบอย่างเดียวยังมิได้รับประกันความสำเร็จของระบบ เมื่อเสร็จสิ้นขั้นตอนการวิเคราะห์ออกแบบแล้ว เราต้องเริ่มพัฒนาโปรแกรมสำหรับระบบใหม่นี้ การพัฒนาโปรแกรมในขั้นนี้จะรวมถึงการเขียนโปรแกรม ทดสอบและปรับปรุง เพื่อให้ได้ระบบที่มีประสิทธิภาพสูงสุด ในขณะที่เดียวกันเราจะเริ่มอบรมผู้ใช้และเตรียมสถานที่ ให้พร้อมสำหรับคอมพิวเตอร์ (ในกรณีที่ซื้อใหม่หรือโยกย้าย) ขึ้นถัดมาเมื่อเริ่มนำโปรแกรมที่เขียนได้มาใช้งาน จะต้องถ่าย ข้อมูลเดิมเข้าสู่ระบบใหม่นี้ แล้วจึงเริ่มต้นใช้งานระบบ ใหม่ การบำรุงรักษาในขั้นตอน

การพัฒนาจะรวมถึงการบำรุงรักษาประจำวันคือ ทดสอบว่าระบบทำงานปกติ ถ้าหากพบว่ายังมีข้อบกพร่องที่จุดใด ระบบจะต้องได้รับการแก้ไข

การสร้างโปรแกรมและการประกันคุณภาพ

โปรแกรมเมอร์จะทำหน้าที่เขียนโปรแกรมสำหรับระบบใหม่ทั้งหมด หรือแก้ไขโปรแกรมสำเร็จรูปถ้าซื้อโปรแกรมมา ตัวนักวิเคราะห์ระบบจะต้องกำหนดมาตรฐานของโปรแกรม โดยเขียนเป็น "คู่มือสำหรับโปรแกรมเมอร์" ซึ่งจะกำหนด มาตรฐานของโปรแกรมและเอกสารไว้ในคู่มือนี้ มาตรฐานของโปรแกรมได้แก่ การเขียนโปรแกรมจะต้องเป็นแบบ โปรแกรมโครงสร้าง การตั้งชื่อข้อมูลก็ควรให้อยู่ในรูปแบบเดียวกัน คือโปรแกรมเมอร์ทุกคนใช้ชื่อเดียวกันทั้งหมด สำหรับชื่อโปรแกรมควรจะต้องให้มีรูปแบบเหมือนกัน เช่น ใช้ตัวอักษร 6 ตัวโดยที่สามตัวแรกเป็นตัวอักษร และสามตัวหลังเป็นตัวเลข เช่น (APY000) เป็นต้น

การประกันคุณภาพ

นักวิเคราะห์ระบบจะต้องรับประกันว่า โปรแกรมที่ได้มานั้นจะต้องมีข้อบกพร่องน้อยที่สุด ระหว่างแต่ละขั้นตอนเราจะต้องหาข้อบกพร่องที่อาจจะเกิดขึ้นได้ และกำจัดออกไปก่อนที่จะก้าวสู่ขั้นตอนถัดไป เพราะว่าข้อบกพร่องมีอยู่ในระบบมากเท่าใด ก็จะทำให้ราคาในการแก้ข้อบกพร่องมีมากเท่านั้น และจะขึ้นอยู่กับระยะเวลาของการพัฒนาระบบด้วยว่ามีข้อ บกพร่องเกิดขึ้นนานเท่าไร แล้วด้วย ซึ่งค่าใช้จ่ายในการแก้ไขระบบจะเพิ่มขึ้นด้วยอัตราแบบ "Exponential" ตัวอย่างเช่น พบว่า ลืมตรวจสอบอินพุตที่สำคัญตัวหนึ่งถ้าอยู่ในขั้นวิเคราะห์ระบบและจะแก้ไขจุดบกพร่องนี้จะเสียค่าใช้จ่ายประมาณ 10 บาท และอาจจะเพิ่มเป็น 100 บาท ถ้าพบข้อบกพร่องนี้ในการออกแบบ และเพิ่มเป็น 1,000 บาทถ้าพบในขั้นตอนเขียนโปรแกรมและทดสอบระบบและอาจจะสูงขึ้นถึง 10,000 บาท ถ้าพบหลังจากนำโปรแกรมไปใช้งานแล้ว สรุปแล้วก็คือ แก้ไขในกระดานนั้นง่าย และถูกกว่าแก้ไข ในโปรแกรมอย่างแน่นอน

ที่มา <http://information-system-analysis-and-desi.blogspot.com/>

2.4 ภาษา PHP (Personal Home Page)

ความหมาย PHP นั้นเป็นภาษาสำหรับใช้ในการเขียนโปรแกรมบนเว็บไซต์ สามารถเขียนได้หลากหลายโปรแกรมเช่นเดียวกับภาษาทั่วไป อาจมีข้อสงสัยว่า ต่างจาก HTML อย่างไร คำตอบคือ HTML นั้นเป็นภาษาที่ใช้ในการจัดรูปแบบของเว็บไซต์ จัดตำแหน่งรูป จัดรูปแบบตัวอักษร หรือใส่สีสีนให้กับ เว็บไซต์ของเรา แต่ PHP นั้นเป็นส่วนที่ใช้ในการคำนวณ ประมวลผล เก็บค่า และทำตามคำสั่งต่างๆ อย่างเช่น รับค่าจากแบบ form ที่เราทำ รับค่าจากช่องคำตอบของเว็บบอร์ดและเก็บไว้เพื่อนำมาแสดงผลต่อไป แม้แต่กระทั่งใช้ในการเขียน CMS ยอดนิยมเช่น Drupal , Joomla พูดง่ายๆคือเว็บไซต์จะโต้ตอบกับผู้ใช้ได้ ต้องมีภาษา PHP ส่วน HTML หรือ Javascript ใช้เป็นเพียงแค่ตัวควบคุมการแสดงผลเท่านั้น

การกำเนิดภาษา PHP แต่ละเวอร์ชัน

PHP เกิดในปี 1994 โดย Rasmus Lerdorf โปรแกรมเมอร์อเมริกัน ได้คิดค้นสร้างเครื่องมือที่ใช้ในการพัฒนาเว็บส่วนตัวของเขา โดยใช้ข้อดีของภาษา C และ Perl เรียกว่า Personal Home Page และได้สร้างส่วนติดต่อกับฐานข้อมูลที่ชื่อว่า Form Interpreter (FI) รวมทั้งสองส่วน เรียกว่า PHP/FI ซึ่งก็เป็นจุดเริ่มต้นของ PHP มีคนที่เข้ามาเยี่ยมชมเว็บไซต์ของเขาแล้วเกิดชอบจึงติดต่อขอเอาโค้ดไปใช้บ้าง และนำไปพัฒนาต่อ ในลักษณะของ Open Source ภายหลังจากมีความนิยมขึ้นเป็นอย่างมาก ภายใน 3 ปีมีเว็บไซต์ที่ใช้ PHP/FI ในติดต่อกับฐานข้อมูลและแสดงผลแบบ ไดนามิกและอื่นๆ มากกว่า 50000 เว็บไซต์

PHP2 (ในตอนนั้นใช้ชื่อว่า PHP/FI) ในช่วงระหว่าง 1995-1997 Rasmus Lerdorf ได้มีผู้ที่มาช่วยพัฒนาอีก 2 คนคือ Zeev Suraski และ Andi Gutmans ชาวอิสราเอล ซึ่งปรับปรุงโค้ดของ Lerdorf ใหม่โดยใช้ C++ ให้มีความสามารถจัดการเกี่ยวกับแบบฟอร์มข้อมูลที่ถูกรวบรวมมาจากภาษา HTML และสนับสนุนการติดต่อกับโปรแกรมจัดการฐานข้อมูล mSQL จึงทำให้ PHP เริ่มถูกใช้มากขึ้นอย่างรวดเร็ว และเริ่มมีผู้สนับสนุนการใช้งาน PHP มากขึ้น โดยในปลายปี 1996 PHP ถูกนำไปใช้ประมาณ 15,000 เว็บไซต์ทั่วโลก และเพิ่มจำนวนขึ้นเรื่อยๆ ต่อมาก็มีผู้เข้ามาช่วยพัฒนาอีก 3 คน คือ Stig Bakken รับผิดชอบความสามารถในการติดต่อ Oracle, Shane Caraveo รับผิดชอบดูแล PHP บน Window 9x/NT, และ Jim Winstead รับผิดชอบการตรวจความบกพร่องต่างๆ และได้เปลี่ยนชื่อเป็น Professional Home Page ในเวอร์ชันที่ 2

PHP3 ออกมาในช่วงระหว่างเดือน มิถุนายน 1997 ถึง 1999 ได้ออกสู่สายตาของนักโปรแกรมเมอร์ มีคุณสมบัติเด่นคือสนับสนุนระบบปฏิบัติการทั้ง Window 95/98/ME/NT, Linux และเว็บเซิร์ฟเวอร์ อย่าง IIS, PWS, Apache, OmniHTTPd สนับสนุน ระบบฐานข้อมูลได้หลายรูปแบบเช่น SQL Server, MySQL, mSQL, Oracle, Informix, ODBC

PHP4 ตั้งแต่ 1999 - 2007 ซึ่งได้เพิ่ม Functions การทำงานในด้านต่างๆให้มากและง่ายขึ้น โดย บริษัท Zend ซึ่งมี Zeev และ Andi Gutmans ได้ร่วมก่อตั้งขึ้น (<http://www.zend.com>) ในเวอร์ชันนี้จะเป็น compile script ซึ่งในเวอร์ชันหน้าจะเป็น embed script interpreter ในปัจจุบันมีคนได้ใช้ PHP สูงกว่า 5,100,000 ไซต์ แล้วทั่วโลก และ ผู้พัฒนาได้ตั้งชื่อของ PHP ใหม่ว่า PHP: Hypertext Preprocessor ซึ่งหมายถึงมีประสิทธิภาพระดับโปรเฟสเซอร์สำหรับไฮเปอร์เท็กซ์

PHP5 ตั้งแต่ 2007-ปัจจุบัน มี ได้เพิ่ม Functions การทำงานในด้านต่าง ๆ เช่น

- Object Oriented Model
- การกำหนดคส โคลป public/private/protected
- Exception handling
- XML และ Web Service
- MySQLi และ SQLite
- Zend Engine 2.0

รายชื่อของนักพัฒนาภาษา PHP ที่เป็นแก่นสำคัญในปัจจุบันมีดังต่อไปนี้

- Zeev Suraski, Israel
- Andi Gutmans, Israel
- Shane Caraveo, Florida USA
- Stig Bakken, Norway
- Andrey Zmievski, Nebraska USA
- Sascha Schumann, Dortmund, Germany
- Thies C. Arntzen, Hamburg, Germany
- Jim Winstead, Los Angeles, USA
- Rasmus Lerdorf, North Carolina, USA

การพัฒนาเว็บไซต์ด้วย PHP

สำหรับผู้พัฒนาเว็บไซต์ด้วย PHP นั้นปรกติจะทำการจำลองเครื่องของตัวเองให้เป็น Web Server ระหว่างการพัฒนาเพื่อดูการทำงาน ของโปรแกรมที่เขียนขึ้นมาครับ จากนั้นจึงจะอัปไฟล์ ทั้งหมดลงใน Web Server จริงครับ ในส่วนของ Web Server นั้นทาง Hellomyweb ก็มีให้บริการอยู่นะครับ ถ้ามองว่าเราจะให้เครื่องคอมพิวเตอร์ของเรานั้นทำงานได้เหมือนกับ Web server จริงได้หรือไม่ คำตอบคือได้ครับ แต่มันออกจะไม่คุ้มค่า ทางการเงินนะครับ เพราะเราต้องเสียค่าไฟ ค่าอินเทอร์เน็ต เครื่องคอมก็ต้องเปิดไว้ตลอดปิดไม่ได้ เวลาผู้ใช้งานจากภายนอกมาเรียกใช้ก็รองรับไม่ได้ไม่มาก ดังนั้นการเช่า Web Server ภายนอกจะคุ้มค่ามากกว่าครับ หากต้องการจะพัฒนาเว็บไซต์เพื่อใช้งานจริงๆ

ที่มา <http://www.showded.com/myprofile/mainblog.php?user=san&jucId=2955>

2.5 การออกแบบฐานข้อมูล (E-R Diagrams)

ปี ค.ศ. 1976 Peter Chen ได้พัฒนาแบบจำลอง E-R (Entity-Relationship Model , ERM) ขึ้นมาใช้เป็นเครื่องมือในการนำเสนอโครงสร้างของฐานข้อมูลใน ระดับ ความคิด (Conceptual Level) ออกมาในลักษณะของแผนภาพ (Diagram) ที่เข้าใจได้ง่ายในการสื่อความหมายระหว่างนักออกแบบฐานข้อมูล และผู้ใช้ข้อมูลเกี่ยวกับ ความสัมพันธ์ของเอนทิตีกับเอนทิตี และความสัมพันธ์ของเอนทิตีกับแอททริบิวท์ และ นับจากนั้นเป็นต้นมา แผนภาพ E-R ก็ได้มีการนำไปใช้งานอย่างกว้างขวาง มีการ พัฒนารูปแบบที่หลากหลาย แต่ละรูปแบบของแผนภาพ E-R ที่หลากหลายดังกล่าวล้วนอยู่บนพื้นฐานแนวคิด

แผนภาพ E-R (E-R Diagram) เป็นแบบจำลองที่ใช้อธิบายโครงสร้างของ ฐานข้อมูลซึ่งเขียนออกมาในลักษณะของรูปภาพการอธิบายโครงสร้างและความสัมพันธ์ของข้อมูลความสัมพันธ์ของเอนทิตีที่ช่วยในการออกแบบฐานข้อมูลและได้รับความนิยมเป็นอย่างมาก แผนภาพ E-R เป็นแบบ จำลองเชิงแนวคิด (Conceptual Data Model) ที่แสดงออกมาในลักษณะของแผนภาพ โดยใช้หลักการจาก โมเดลฐานข้อมูลเชิงสัมพันธ์ในการแสดงลักษณะโดยรวมของข้อมูลในระบบ ช่วยสื่อสารให้เกิดความเข้าใจร่วมกันระหว่างผู้วิเคราะห์และผู้ใช้ได้เป็นอย่างดี แผนภาพที่ นิยมใช้ในการนำเสนอโครงสร้างฐานข้อมูลที่นิยม คือแผนภาพ E-R ซึ่งประกอบไปด้วยเอนทิตี แอททริบิวท์ของแต่ละเอนทิตี ความสัมพันธ์ระหว่างเอนทิตีและดีกรีของความสัมพันธ์ (Degree Of A Relationship)

แผนภาพ E-R มีความสำคัญต่อการพัฒนาระบบงานฐานข้อมูล แอปพลิเคชัน (Applications) ต่าง ๆ ที่ต้องการการเก็บข้อมูลอย่างมีระบบ แผนภาพ E-R จึงใช้เพื่อ เป็นเอกสารในการสื่อสารระหว่างนักออกแบบระบบ และนักพัฒนาระบบ เพื่อให้ สื่อสารอย่างตรงกัน และเป็นสากล

องค์ประกอบของแผนภาพ E-R

เอนทิตี (Entity) เอนทิตีหมายถึงสิ่งที่มีอยู่จริง จับต้องได้ หรืออาจจะเป็นจินตภาพที่แสดงความเป็นหนึ่งเดียว ซึ่งเมื่อก้าวถึงแล้วทุกคนเข้าใจตรงกัน แต่โดยทั่วไปแล้วเอนทิตี มักจะ อยู่ในรูปของนาม ซึ่งอาจเป็นสิ่งที่ป็นรูปธรรมคือสามารถมองเห็นได้ด้วยตา หรืออยู่ ในรูปของนามธรรมคือไม่สามารถมองเห็นได้ด้วยตาก็ได้ เช่น บุคคล (นิสิต อาจารย์ ลูกค้า พนักงาน คนไข้) สถานที่ (โรงเรียน ห้องเก็บสินค้า คลังสินค้า ร้านค้า) วัตถุ (หนังสือ เครื่องจักรกล เครื่องยนต์ สินค้า วัตถุดิบ) เหตุการณ์ (การพิมพ์ การกิน ลงทะเบียน การฝาก-การถอนเงิน) หรือ แนวคิด (วัน วิชา บัญชี หลักสูตร ความชำนาญ) ที่ทำให้เกิดกลุ่มของข้อมูลที่ต้องการจัดเก็บรวมทั้งสามารถบ่งชี้ถึงความเป็น เอกลักษณะเฉพาะตัวได้ เอนทิตีมีคุณสมบัติได้หลายอย่าง และเอนทิตีเป็นสิ่งที่ผู้ใช้งาน ฐานข้อมูลจะต้องเกี่ยวข้องกับเมื่อมีการออกแบบระบบฐานข้อมูล สัญลักษณ์รูปสี่เหลี่ยมผืนผ้าแทนเอนทิตีหนึ่งเอนทิตีและมีชื่อเอนทิตีกำกับอยู่ ภายใน ชื่อที่ใช้จะต้องเป็น คำนาม และโดยปกติมักจะเขียนตัวพิมพ์ใหญ่

ตารางที่ 2.1 แสดงตาราง Entity ของข้อมูล

รหัสนิสิต	ชื่อ-สกุล	เพศ	คณะวิชา	วันเกิด
585150001	ภคดี งามสง่า	ชาย	บริหารธุรกิจ	20-5-38
585150002	จินตนา หวังดี	หญิง	บัญชี	5-10-38
585150003	วัชรพล สิงคี	ชาย	วิศวกรรม	15-6-38
585150004	สมคิด จินดา	ชาย	นิติศาสตร์	2-4-38

เอนทิตีสามารถแบ่งออกได้หลายประเภทด้วยกัน เช่น

เอนทิตีปกติ (Strong Entity หรือ Regular Entity) เป็นเอนทิตีที่มี คุณสมบัติเฉพาะ (Identity) ในตัวเอง สามารถเกิดขึ้นได้ด้วยตัวเอง การคงอยู่ของ เอนทิตีจะไม่ขึ้นกับเอนทิตีอื่น เช่น นิสิต หรือ อาจารย์ หรือ สินค้า สัญลักษณ์ที่ใช้คือ รูปสี่เหลี่ยมผืนผ้า

เอนทิตีอ่อนแอ (Weak Entity) เป็นเอนทิตีที่ไม่สามารถเกิดขึ้นได้ ตามล ำพัง จะขึ้นอยู่กับเอนทิตีอื่น จะมีคีย์หลักจากการสืบทอดเอนทิตีที่อ้างอิงอยู่มา ใช้เป็นคีย์หลักหรือส่วนหนึ่งของคีย์หลัก และจะถูกลบออกไปด้วยเมื่อเอนทิตีหลักถูก ลบ สัญลักษณ์ที่ใช้คือรูปสี่เหลี่ยมผืนผ้าเส้นคู่

คอมโพสิตเอนทิตี หรือเอนทิตีเชิงความสัมพันธ์ (Composite / Associate Entity) เป็นเอนทิตีที่สร้างขึ้นมาแปลงความสัมพันธ์ของเอนทิตีสองเอนทิตีที่มีความสัมพันธ์แบบ $M : N$ ให้เป็นแบบ $1 : M$ เพื่อให้เข้าใจได้ง่าย โดยการนำเอาคีย์หลักของเอนทิตีทั้งสองมารวมกันกับแอททริบิวต์อื่นๆ ที่สนใจ สัญลักษณ์ที่ใช้คือ สี่เหลี่ยม ผืนผ้าที่ภายในมีรูปสี่เหลี่ยมขนมเปียกปูน เช่น อาจารย์หนึ่งคนสอนได้หลาย วิชา และวิชาหนึ่งรายวิชามีอาจารย์สอนได้หลายคน

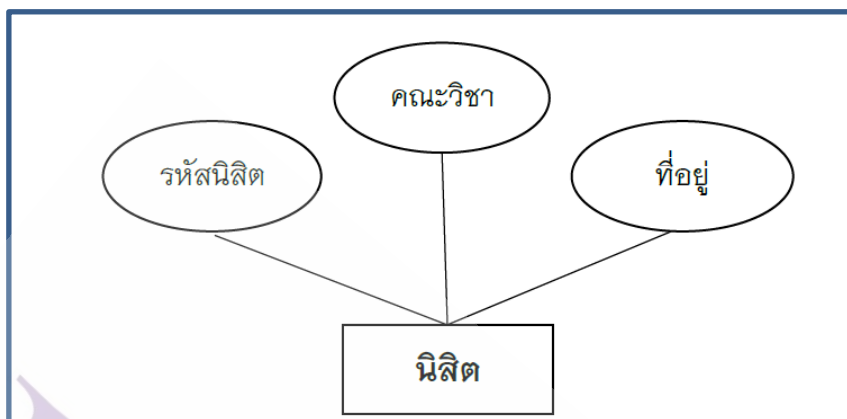
แอททริบิวต์ หรือ พรอพเพอร์ตี (Property)

เอนทิตีหนึ่งๆ จะประกอบไปด้วยคุณสมบัติ หรือ ลักษณะที่แตกต่างกัน ออกไป ข้อมูลที่แสดงถึงคุณลักษณะของเอนทิตีเรียกว่าแอททริบิวต์ ดังนั้นความหมายของแอททริบิวต์ก็คือคุณสมบัติของเอนทิตีหรือข้อมูลของเอนทิตีที่เราสนใจจะเก็บ หรืออาจจะหมายถึงกลุ่มของค่าความจริงใด ๆ ที่เป็นรายละเอียดของเอนทิตีซึ่งแสดง ลักษณะและคุณสมบัติของเอนทิตี แต่ละเอนทิตีสามารถมีคุณสมบัติต่าง ๆ ได้ มากมาย ขึ้นอยู่กับว่าต้องการจัดเก็บข้อมูลอะไรบ้าง เช่น คุณสมบัติหรือข้อมูลของ นิสิต ประกอบไปด้วย ชื่อ, ที่อยู่, หมายเลขโทรศัพท์ รหัสนิสิต, คณะวิชา, สาขาวิชา, วัน เดือนปีเกิด, เพศ, ส่วนสูง, น้ำหนัก ฯลฯ

แอททริบิวต์ สามารถแบ่งออกได้หลายประเภทเช่น

ซิมเพิลแอททริบิวต์ (Simple Attribute) เป็นแอททริบิวต์แบบ ง่าย ๆ มีเพียงองค์ประกอบเดียวหรือข้อมูลเดียว ภายในแอททริบิวต์ไม่สามารถ แบ่งย่อยได้อีก เช่น ข้อมูล เพศ, อายุ, ส่วนสูง, น้ำหนัก, รหัสนิสิต, คณะวิชา, สาขาวิชา เป็นต้น จะประกอบด้วยค่าข้อมูลเดียว สัญลักษณ์ที่ใช้แทนแอททริบิวต์ ชนิดนี้ คือวงรีที่มีเส้นเชื่อมต่อไปยังเอนทิตีของแอททริบิวต์ นั้นๆ โดยมีชื่อแอททริบิวต์ อยู่ภายในวงรี

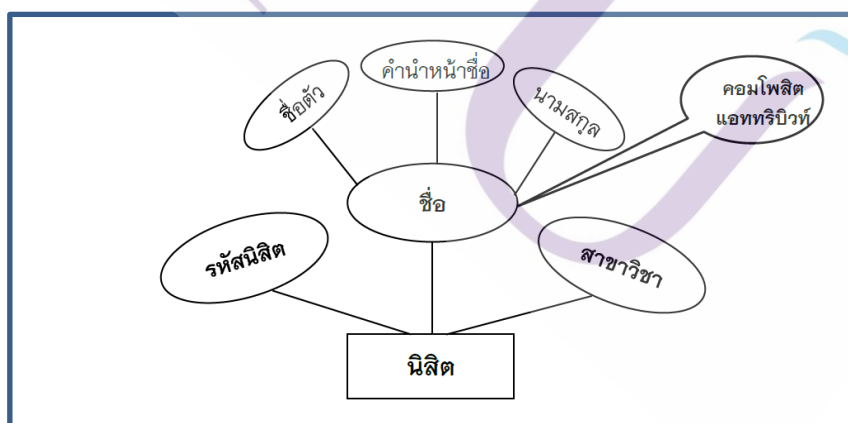
ตัวอย่าง แอททริบิวต์ “รหัสนิสิต” “เพศ” “ที่อยู่” “หมายเลขโทรศัพท์” “คณะ วิชา” ของเอนทิตี “นิสิต”



ภาพที่ 2.8 ซิมเพลแอททริบิวต์

คอมโพสิตแอททริบิวต์ (Composite Attribute) หมายถึงแอททริบิวต์ที่มีองค์ประกอบหรือข้อมูล ซึ่งจะตรงกันข้ามกับซิมเพลแอททริบิวต์ ค่าภายในแอททริบิวต์ สามารถแบ่งย่อยออกมาได้อีก แต่มีอิสระต่อกัน เช่น

- แอททริบิวต์ “ที่อยู่” จะประกอบด้วย บ้านเลขที่ หมู่ที่ ตำบล อำเภอ และจังหวัด
- แอททริบิวต์ “ชื่อ” จะประกอบด้วย คานาหน้าชื่อ ชื่อตัว และ นามสกุล



ภาพที่ 2.9 คอมโพสิตแอททริบิวต์

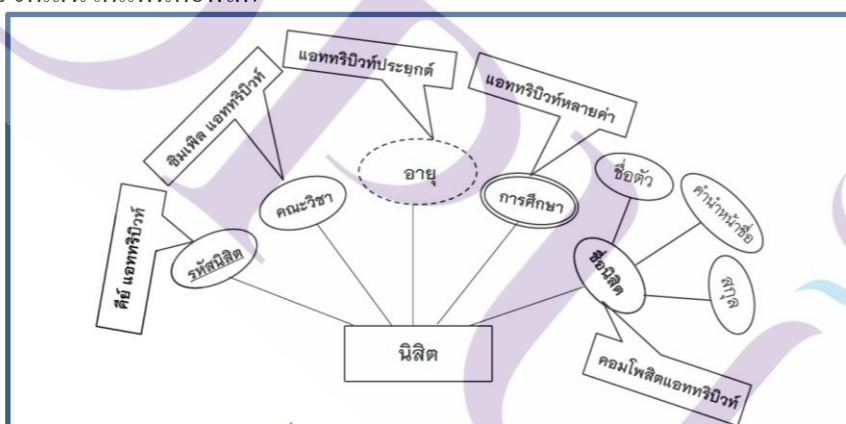
แอททริบิวต์ ค่าเดียว (Single-value Attribute) คือ แอททริบิวต์ที่มีค่าข้อมูลที่เก็บเพียงค่าเดียว เช่น รหัสนิสิต ชื่อ ภาควิชา ในเอนทิตี “นิสิต” ซึ่งนิสิตแต่ละคนก็จะมีค่าข้อมูลเหล่านี้เพียงค่าเดียวเท่านั้น หรือเอนทิตี “เพศ” ก็จะมีค่าข้อมูลที่เก็บเพียง 1 ค่า คือ หญิง หรือ ชาย เท่านั้นเช่นกัน

แอททริบิวต์ หลายค่า (Multivalued Attribute) คือ แอททริบิวต์ที่สามารถมีได้หลายค่า เช่น คนหนึ่งคนสามารถมี “วุฒิการศึกษา” ได้หลายระดับ เช่น ระดับปริญญาตรี ระดับปริญญาโท และ

ระดับปริญญาเอก หรือ คนหนึ่งคนอาจจะมี “หมายเลขโทรศัพท์” ได้หลายหมายเลข หรือข้อมูล. “โรงเรียนที่เรียนจบ” ตั้งแต่ระดับ อนุบาล ประถมศึกษา มัธยมศึกษา และอุดมศึกษา อาจจะมาจกหลายโรงเรียน เป็นต้น สัญลักษณ์ของแอททริบิวท์ หลายค่า คือรูปวงรีสองวงซ้อนกัน

แอททริบิวท์ ประยุกต์ (Derived Attribute) หมายถึง แอททริบิวท์ที่ไม่ได้เก็บอยู่ในฐานข้อมูล แต่ได้จากการประยุกต์ใช้แอททริบิวท์อื่น เช่น แอททริบิวท์ รายได้รวมของพนักงานขาย ของ ได้มาจากการรวมเงินเดือน ค่าโบนัส ค่าบาเน็จ(Commission) ของแต่ละเดือน หรือ แอททริบิวท์ อายุ ได้จากการคำนวณจากวันเดือนปีเกิด สัญลักษณ์ของแอททริบิวท์ประยุกต์ เป็นรูปวงรีแบบเส้นประ

คีย์ แอททริบิวท์ (Key Attributes) หรือ คีย์หลัก หรือ กุญแจหลัก เป็นแอททริบิวท์ที่บอกเอกลักษณ์ของเอนทิตีนั้นๆ ได้ จะมีค่าของข้อมูลที่ไม่ซ้ำกัน เช่น รหัสสนิติด จัดเป็นคีย์แอททริบิวท์ ใช้ระบุความแตกต่างของแต่ละคนในเอนทิตี สัญลักษณ์ของคีย์หลักคือรูปวงรีภายในมีชื่อแอททริบิวท์ที่มีการขีดเส้นใต้แทนคีย์หลัก



ภาพที่ 2.10 แอททริบิวท์ชนิดต่าง ๆ

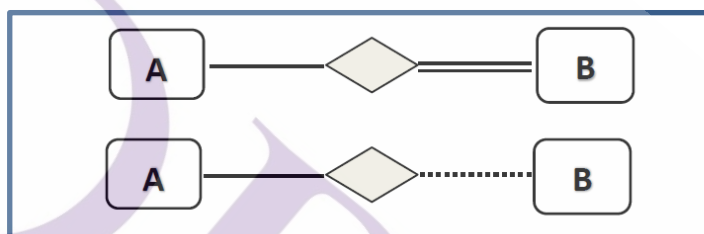
ความสัมพันธ์ระหว่างเอนทิตี (Entity Relationship)

ในปี ค.ศ. 1976 Chen ได้เสนอโมเดลข้อมูลเชิงสัมพันธ์ (Entity Relationship Model) หรือ E-R Model และได้รับการพัฒนาจนสมบูรณ์ในปี ค.ศ.1979 เพื่อใช้ในการอธิบายฐานข้อมูลในรูปแบบของแผนภาพ เรียกว่า “E-R diagram” ซึ่งจะใช้พื้นฐานของโมเดลข้อมูลเชิงตรรกะที่ Fleming ที่ได้เสนอไว้ในปี ค.ศ. 1989 เป็นการสร้างโครงร่างวิวกของผู้ใช้ จะแสดงข้อมูลในขอบเขตที่ผู้ออกแบบสนใจโดยมีสิ่งที่จะต้องกำหนดเป็นพื้นฐานได้แก่ ความสัมพันธ์ระหว่างเอนทิตีและแอททริบิวท์ ในฐานข้อมูลนั้น

ความสัมพันธ์หรือ Relationship จะหมายถึงความสัมพันธ์ระหว่างเอนทิตีหรือความสัมพันธ์ระหว่างข้อมูลเป็นลักษณะของความสัมพันธ์ระหว่างเอนทิตีหนึ่งกับตัวของมันเอง

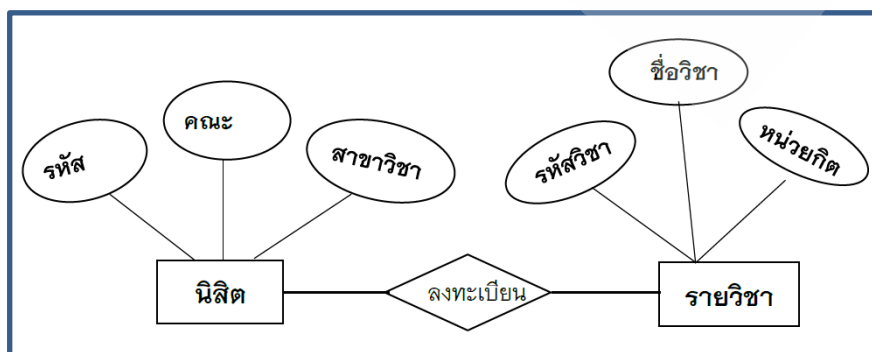
หรือกับเอนทิตีอื่น(มากกว่า 2 เอนทิตีก็ได้) ว่ามีความสัมพันธ์ของข้อมูลเป็นแบบใด ความสัมพันธ์ระหว่างเอนทิตีในแผนภาพ E-R จะแทนด้วยรูปสี่เหลี่ยมข้าวหลามตัดที่มีชื่อของความสัมพันธ์อยู่ภายในและมีเส้นเชื่อมระหว่างเอนทิตีและความสัมพันธ์ระหว่างเอนทิตี

ความสัมพันธ์ระหว่างข้อมูลที่เราสสนใจ ซึ่งพิจารณาได้ 2 แบบ คือความสัมพันธ์ แบบจำเป็นหรือแบบบังคับ (Mandatory) หมายถึงความสัมพันธ์ที่จำเป็นต้องมี หรือ จะต้องเกิดขึ้นอย่างแน่นอน นิยมแทนด้วยสัญลักษณ์เส้นตรงทึบ หรือเส้น ตรงสองเส้น และความสัมพันธ์แบบไม่จำเป็น (Optional) เป็นความสัมพันธ์ที่อาจจะมีหรือไม่มีก็ได้ นิยมแทนด้วยสัญลักษณ์เส้นตรงประหรือเส้นตรงหนึ่งเส้น



ภาพที่ 2.11 ความสัมพันธ์ระหว่างเอนทิตี หรือระหว่างข้อมูล

อธิบายได้ดังนี้ : ถ้าข้อมูล A มีความสัมพันธ์กับข้อมูล B อย่างแน่นอน หรืออย่างน้อยจะต้องมีสมาชิก (Element) ของข้อมูล A หนึ่งสมาชิกที่มีความสัมพันธ์กับสมาชิกของข้อมูล B และแทนความสัมพันธ์ด้วยเส้นตรงทึบออกจากข้อมูล A ไปยังข้อมูล B แต่ในขณะเดียวกันข้อมูล B อาจจะไม่มีความสัมพันธ์กับข้อมูล A หรือไม่มีก็ได้ เส้นตรงที่ออกจาก ข้อมูล B จึงแทนด้วยเส้นประ ถ้ากำหนดให้ A และ B เป็นข้อมูล สัญลักษณ์ที่ใช้แทนชนิดความสัมพันธ์ระหว่างข้อมูล A กับข้อมูล B คือ เส้นตรงเส้นเดียวหรือจะใช้ลูกศร 1 หัว กับ ลูกศร 2 หัวก็ได้



ภาพที่ 2.12 ความสัมพันธ์ของข้อมูล

ชนิดความสัมพันธ์ระหว่างข้อมูล

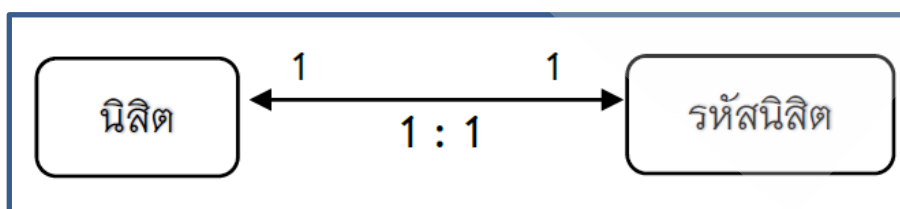
ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One to One Relationships ; 1 : 1) เป็นการแสดงถึงความสัมพันธ์ของข้อมูลของเอนทิตีหนึ่งว่ามีความสัมพันธ์กับข้อมูลในอีกเอนทิตีหนึ่งในลักษณะหนึ่งต่อหนึ่ง หมายถึงข้อมูลหนึ่งข้อมูลของเอนทิตีที่ 1 มีความสัมพันธ์กับข้อมูลในเอนทิตีที่ 2 เพียงข้อมูลเดียวเท่านั้น ซึ่งใช้ตัวเลขกำกับที่เส้นเพื่อระบุความสัมพันธ์ เช่น ข้อมูล A จะเกี่ยวข้องสัมพันธ์กับข้อมูล B ได้เพียง 1 ค่าเท่านั้น และ ข้อมูล B จะเกี่ยวข้องสัมพันธ์กับข้อมูล A ได้เพียง 1 ค่าเช่นกัน



ภาพที่ 2.13 ความสัมพันธ์หนึ่งต่อหนึ่ง หรือแบบ 1:1

ตัวอย่างที่ 1 เช่น มีเอนทิตี 2 เอนทิตี คือเอนทิตี “อาจารย์” และเอนทิตี “คณะวิชา” มีความสัมพันธ์กันชื่อ “บริหาร” แบบ 1 : 1 ซึ่งหมายถึง อาจารย์ 1 คน จะสามารถบริหาร หรือเป็นคณบดีได้เพียง 1 คณะวิชาเท่านั้น และในขณะเดียวกัน แต่ละคณะวิชา ก็จะมีคณบดี ได้เพียง 1 คน เช่นกัน หรือความสัมพันธ์ระหว่างนิสิตกับรหัสนิสิต จะเป็นแบบ 1 : 1 เพราะนิสิต 1 คน จะมีรหัสนิสิตได้เพียง 1 รหัสเท่านั้น และในขณะเดียวกันรหัสนิสิต 1 รหัส จะใช้กับนิสิตได้เพียง 1 คนเช่นกัน

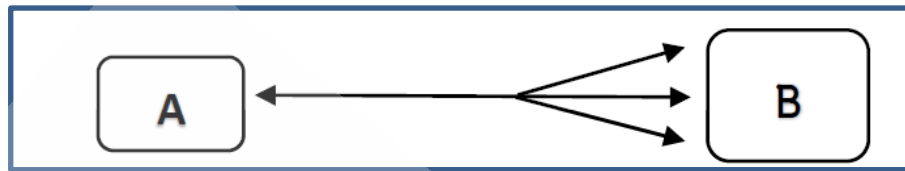
ความสัมพันธ์ระหว่างข้อมูลนิสิตกับรหัสนิสิต : นิสิต 1 คน จะมีรหัสนิสิตได้เพียง 1 รหัสเท่านั้น ในขณะเดียวกันรหัสนิสิต 1 รหัส จะใช้กับนิสิตได้เพียง 1 คนเช่นกัน



ภาพที่ 2.14 ความสัมพันธ์ระหว่างข้อมูลนิสิตกับรหัสนิสิต แบบ 1:1

ความสัมพันธ์แบบหนึ่งต่อกลุ่ม (One to Many Relationships ; 1 : N หรือ 1 : M) เป็นความสัมพันธ์ระหว่างเอนทิตีหนึ่ง ไปมีความสัมพันธ์กับอีกเอนทิตีหนึ่งได้มากกว่าหนึ่งรายการ ในลักษณะหนึ่งต่อกลุ่ม (ข้อมูลหนึ่งข้อมูลของเอนทิตีที่หนึ่งมีความสัมพันธ์กับข้อมูลในเอนทิตีที่สองหลายข้อมูล) เช่น ข้อมูล “A” มีความสัมพันธ์กับข้อมูล “B” ได้มากกว่าหนึ่งรายการ ความสัมพันธ์

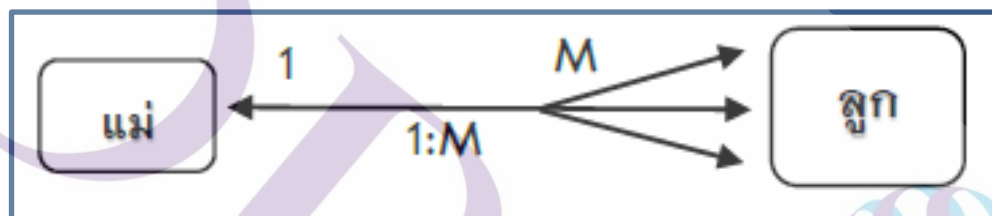
แบบนี้ข้อมูล A จะเกี่ยวข้องกับข้อมูล B ได้มากกว่า 1 ค่า แต่ข้อมูล B จะเกี่ยวข้องกับข้อมูล A ได้
อย่างมากที่สุดเพียง 1 ค่าเท่านั้น ดังตัวอย่าง



ภาพที่ 2.15 ความสัมพันธ์แบบหนึ่งต่อกลุ่ม หรือ 1 : M

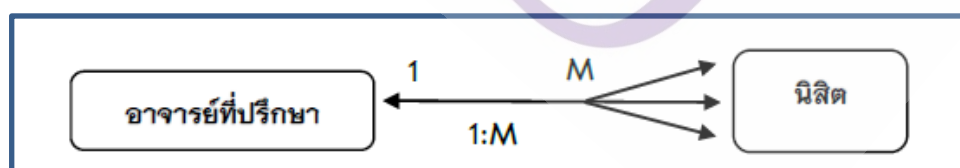
ตัวอย่างที่ 2 ความสัมพันธ์แบบหนึ่งกลุ่ม

ความสัมพันธ์ระหว่างแม่กับลูก : แม่หนึ่งคนอาจจะมีลูกได้หลายคน แต่ลูกหนึ่งคนจะมีแม่
ได้เพียงคนเดียวเท่านั้น



ภาพที่ 2.16 ความสัมพันธ์ระหว่างแม่กับลูกแบบ 1 : M

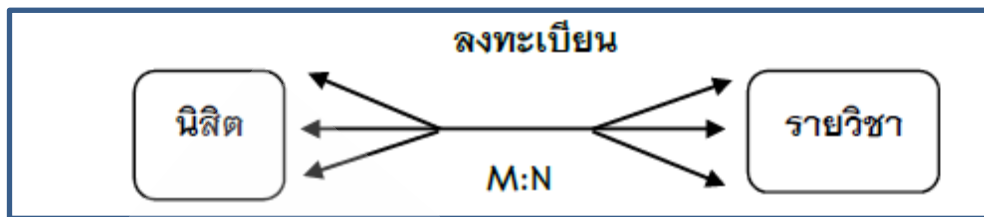
ความสัมพันธ์ระหว่างอาจารย์ที่ปรึกษากับนิสิต : อาจารย์ที่ปรึกษาหนึ่งคนจะมีนิสิตได้
หลายคน แต่นิสิตคนหนึ่ง จะมีอาจารย์ที่ปรึกษาได้เพียง 1 คน



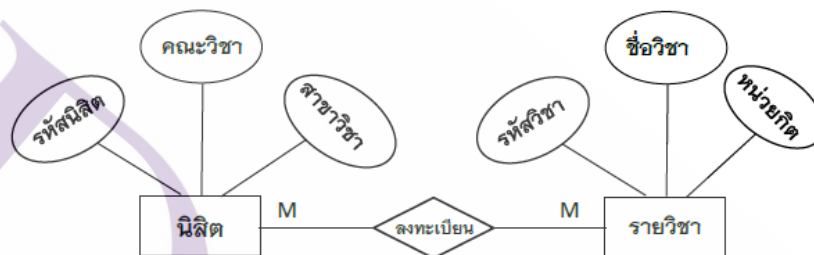
ภาพที่ 2.17 ความสัมพันธ์ระหว่างอาจารย์ที่ปรึกษากับนิสิต

ตัวอย่างที่ 3 ความสัมพันธ์แบบกลุ่ม

ความสัมพันธ์ระหว่างนิสิตกับรายวิชาที่เปิดสอน : นิสิต 1 คนลงทะเบียนเรียนได้หลายวิชา
และวิชาหนึ่ง ๆ จะมีนิสิตลงทะเบียนได้มากกว่า 1 คน



ภาพที่ 2.18 ความสัมพันธ์ระหว่างนิสิตกับรายวิชาที่เปิดสอน



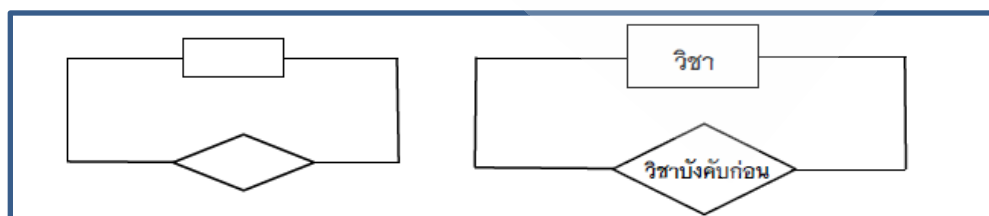
ภาพที่ 2.19 ความสัมพันธ์แบบกลุ่มต่อกลุ่ม หรือ M:N

ดีกรีของความสัมพันธ์ (Degree of Relationship)

ดีกรีความสัมพันธ์หรือระดับชั้นของความสัมพันธ์หมายถึงจำนวนเอนทิตีที่มีส่วนร่วมกับความสัมพัทธ์นั้น ๆ ดีกรีความสัมพันธ์แบ่งออกได้เป็น 3 ประเภท คือ

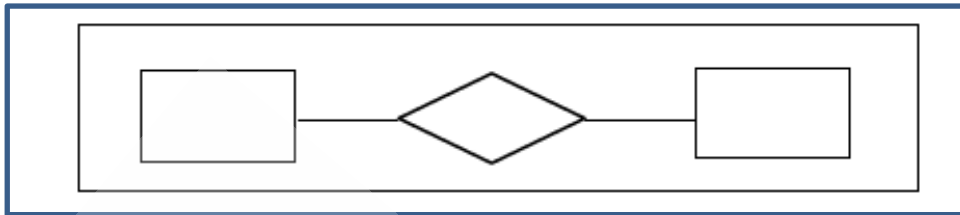
1. ความสัมพันธ์แบบยูนารี (Unary Relationship) หรือความสัมพันธ์แบบรีเคอร์ซีฟ (Recursive Relationship) หรือความสัมพันธ์เอนทิตีเดียว เป็นความสัมพันธ์ระหว่างเอนทิตีชนิดเดียวกัน หรือความสัมพันธ์ในตัวเอง เช่น

- ความสัมพันธ์ระหว่างเอนทิตี “วิชาที่บังคับเรียนก่อน” และ “วิชา” (วิชาเรียนบางวิชาบังคับให้ต้องเรียนบางวิชาก่อนจึงจะสามารถเรียนวิชาดังกล่าวได้)



ภาพที่ 2.20 ความสัมพันธ์แบบยูนารี

2. ความสัมพันธ์แบบไบนารี หรือ ความสัมพันธ์ระหว่างสองเอนทิตี (Binary Relationship) เป็นความสัมพันธ์เป็นความสัมพันธ์แบบ 2 ทางระหว่างเอนทิตี 2 เอนทิตี คือ เอนทิตีหนึ่งมีความสัมพันธ์กับเอนทิตีอื่นที่ต่างประเภทกัน เป็นความสัมพันธ์ที่พบได้บ่อยที่สุด



ภาพที่ 2.21 ความสัมพันธ์แบบไบนารี

ตัวอย่างที่ 4 ความสัมพันธ์แบบไบนารี

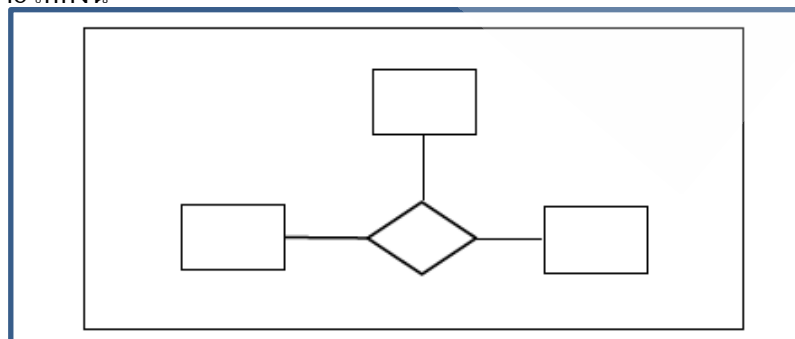
- นิสิต <สังกัด> คณะวิชา
- อาจารย์ <ที่ปรึกษา> นิสิต



ภาพที่ 2.22 ความสัมพันธ์แบบไบนารี อาจารย์กับนิสิต

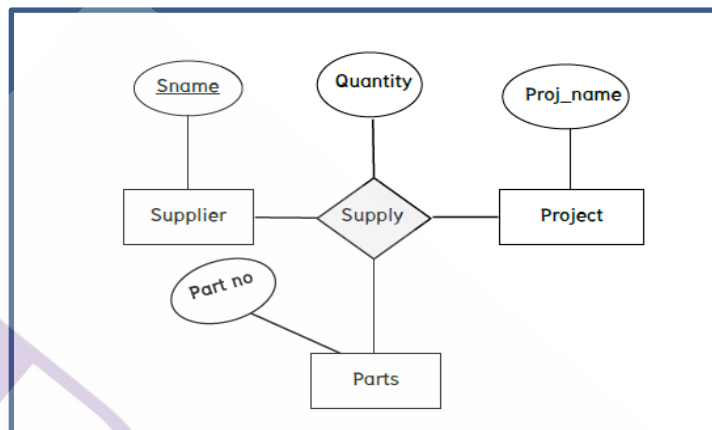
3. ความสัมพันธ์ที่มีเอนทิตีมาเกี่ยวข้องตั้งแต่สามเอนทิตีขึ้นไป (N-ary Relationship) เป็นความสัมพันธ์แบบสามทาง หรือเป็นความสัมพันธ์ระหว่างเอนทิตีต่างประเภทกันตั้งแต่ 3 เอนทิตีขึ้นไป โดยที่ N จะหมายถึงจำนวนเอนทิตีที่มาสัมพันธ์กับความสัมพันธ์หนึ่ง เช่น

ความสัมพันธ์แบบสามทางหรือ เทอนารี (Ternary Relationship) เป็น ความสัมพันธ์ระหว่างสามเอนทิตี อธิบายได้ดังนี้



ภาพที่ 2.23 ความสัมพันธ์แบบสามทาง

ตัวอย่างที่ 5 ความสัมพันธ์แบบสามทาง

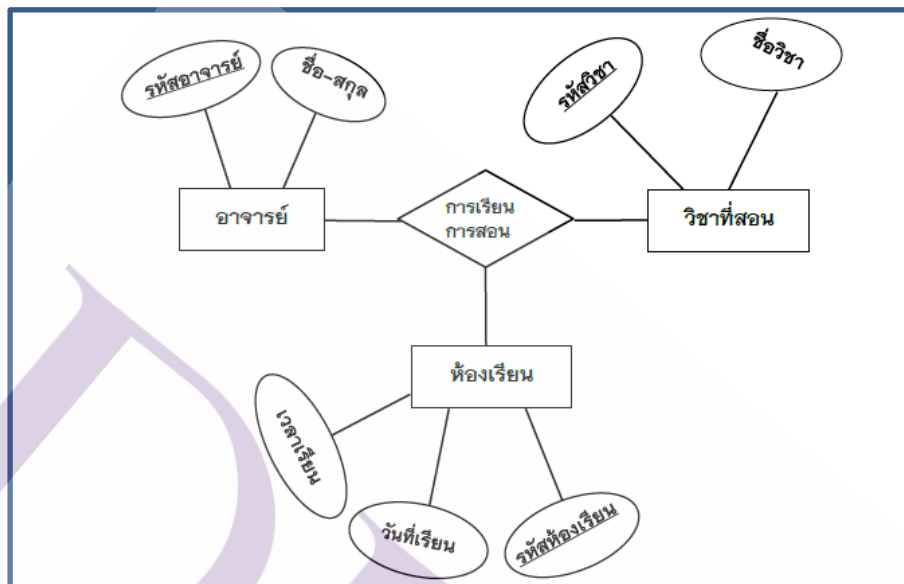


ภาพที่ 2.24 ความสัมพันธ์แบบสามทาง

เป็นความสัมพันธ์ระหว่างเอนทิตีสามเอนทิตี คือระหว่างเอนทิตี ผู้ผลิต (Supplier) เอนทิตี ชิ้นส่วน (Parts) และเอนทิตี โครงการ (Project) สามารถอธิบายความสัมพันธ์ระหว่างเอนทิตีทั้งสามเอนทิตีได้ดังนี้

- ความสัมพันธ์คู่ที่หนึ่ง คือ ความสัมพันธ์ ระหว่าง เอนทิตี ผู้ผลิตกับเอนทิตี ชิ้นส่วน ด้วยความสัมพันธ์ สามารถจัดส่ง (CAN_SUPPLY) คือ ผู้ผลิต สามารถจัดส่ง ชิ้นส่วน (SUPPLIER CAN_SUPPLY PART)
- ความสัมพันธ์คู่ที่สอง คือ ความสัมพันธ์ระหว่าง เอนทิตี ผู้ผลิต กับเอนทิตี โครงการ ด้วยความสัมพันธ์จัดส่งให้ (SUPPIES) คือ ผู้ผลิต จัดส่งให้กับ โครงการ (SUPPLIER SUPPIES PROJECT)
- ความสัมพันธ์คู่ที่สาม คือ ระหว่างเอนทิตีโครงการ กับเอนทิตี ชิ้นส่วน ด้วยความสัมพันธ์ ใช้ (USES) คือ โครงการใช้ชิ้นส่วน (PROJECT USES PART)

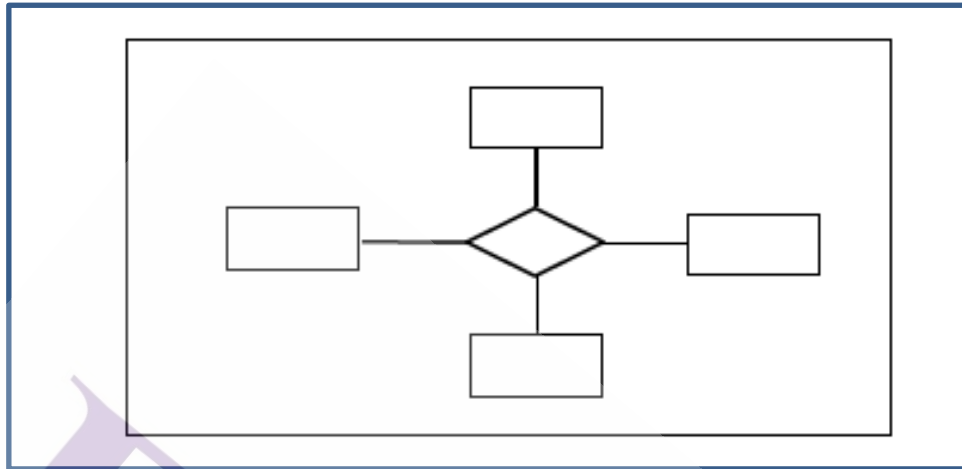
ตัวอย่างที่ 6 ความสัมพันธ์แบบสามทางหรือ เทอเนอร์ ความสัมพันธ์ตารางเรียนแสดง ความสัมพันธ์ระหว่างเอนทิตีอาจารย์ เอนทิตีวิชาที่สอน และเอนทิตีชั้นเรียน



ภาพที่ 2.25 ความสัมพันธ์แบบเทอเนอร์ ตารางเรียน

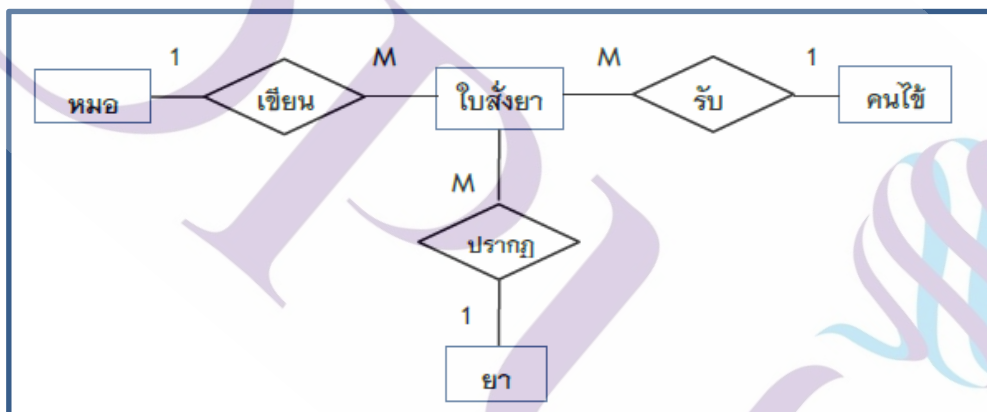
ความสัมพันธ์ระหว่างเอนทิตีทั้งสามเอนทิตี คือ

- ความสัมพันธ์คู่ที่หนึ่ง คือ ระหว่างเอนทิตีอาจารย์กับเอนทิตีวิชาที่สอนด้วยความสัมพันธ์การเรียนการสอน โดยอาจารย์มีหน้าที่รับผิดชอบการสอนในแต่ละรายวิชา
- ความสัมพันธ์คู่ที่สอง คือ ระหว่างเอนทิตีอาจารย์กับเอนทิตี ห้องเรียน ด้วยความสัมพันธ์การเรียนการสอน โดยอาจารย์จะทำการสอนรายวิชาภายในห้องเรียน
- ความสัมพันธ์คู่ที่สาม คือ ระหว่างเอนทิตีวิชาที่สอนกับเอนทิตีห้องเรียน ด้วยความสัมพันธ์การเรียนการสอน โดยแต่ละวิชาที่เปิดสอนจะต้องมีห้องเรียน
- ความสัมพันธ์แบบสี่ทางหรือควาเทอร์นารี (Quaternary Relationship) เป็นความสัมพันธ์ระหว่างสี่เอนทิตี



ภาพที่ 2.26 ความสัมพันธ์แบบควาเทอร์นารี

ตัวอย่างที่ 7 ความสัมพันธ์แบบควาเทอร์นารี

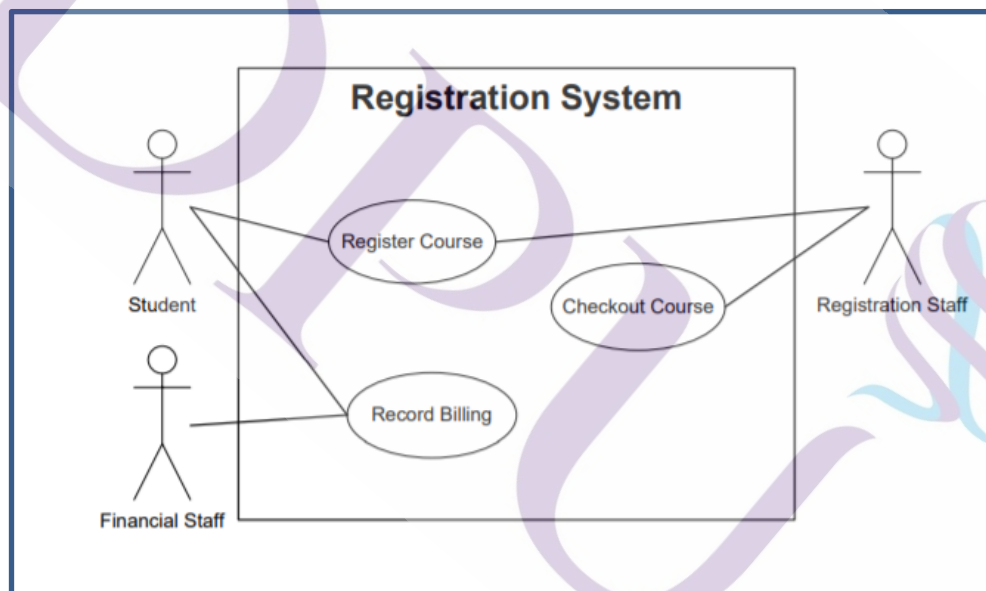


ภาพที่ 2.27 ความสัมพันธ์แบบควาเทอร์นารี ระบบการจ่ายยา

ที่มา <http://www.ict.up.ac.th/worrakits>

2.6 Use Case Diagram

Use Case Diagram เป็นแผนภาพที่ใช้แสดงให้เห็นว่าระบบทำงานหรือมีหน้าที่ใดบ้าง โดยมีสัญลักษณ์รูปวงรีแทน Use Case และสัญลักษณ์รูปคน (Stick Man Icon) แทน Actor สำหรับชื่อ Use Case นั้น ให้ใช้ค ำกริยาหรือกริยาวิเศษณ์ (คำกริยามีกรรมมารองรับ) เช่น ลงทะเบียนเรียน, ตรวจสอบรายวิชา, บันทึกการช าระเงิน, Generate Report, Enter Sales Data, Compute Commission เป็นต้น ส่วนการมีปฏิสัมพันธ์ระหว่าง Use Case และ Actor จะใช้เส้นตรงลาก เชื่อมต่อกัน หรือจะใช้เส้นตรงมีหัวลูกศรก็ได้ ในที่นี้เลือกใช้เส้นตรงไม่มีหัวลูกศร ส่วนเส้นแบ่งขอบเขต ระหว่าง Actor กับ Use Case จะใช้เส้นกรอบสี่เหลี่ยม เรียกว่า “System Boundary” และสิ่ง สำคัญส่วนสุดท้ายก็คือ “ชื่อของระบบ (System Name)” ให้แสดงไว้ด้านบนสุดของแผนภาพ



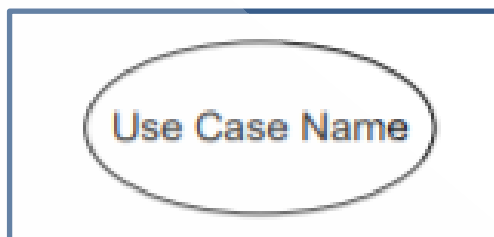
ภาพที่ 2.28 Use Case Diagram ของระบบลงทะเบียน

นอกจากนี้ หากกล่าวถึง Use Case Diagram ในด้านการพัฒนาระบบ นอกเหนือจากการนำมาใช้เก็บรวบรวมความต้องการต่างๆ แล้ว Use Case Diagram ยังถูกนำไปใช้เป็นพื้นฐานเพื่อการสร้างแผนภาพ (Diagram) ชนิดอื่นในขั้นตอนต่อไป และทีมงานยังสามารถใช้ Use Case Diagram เพื่อติดตามผลการดำเนินงานได้อีกด้วย

สัญลักษณ์และความสัมพันธ์ใน Use Case Diagram

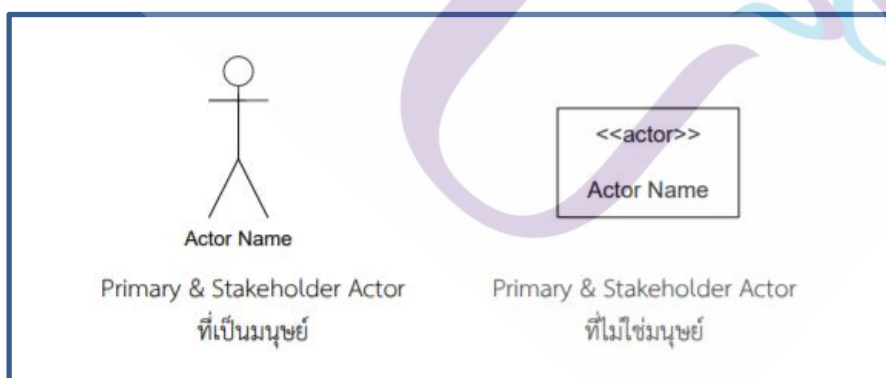
สัญลักษณ์ที่สำคัญของ Use Case Diagram มีดังต่อไปนี้

- Use Case คือ หน้าหน้าที่ระบบต้องกระทำ ใช้สัญลักษณ์รูปวงรี พร้อมทั้งเขียนชื่อ Use Case ซึ่งต้อง ใช้คำกริยาหรือกริยาวลีก็ได้



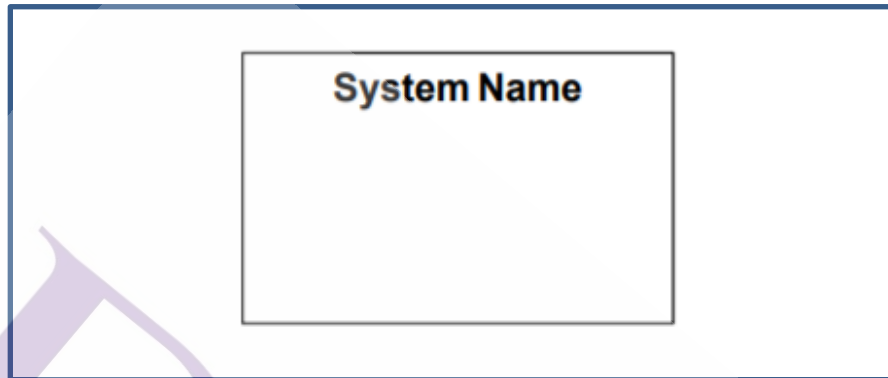
ภาพที่ 2.29 Use Case

- Actor คือ ผู้เกี่ยวข้องกับระบบ ซึ่งรวมทั้ง Primary Actor และ Stakeholder Actor ที่เป็นมนุษย์ ในที่นี้จะใช้สัญลักษณ์รูปคน (Stick Man Icon) เหมือนกัน พร้อมทั้งเขียนชื่อ Actor ไว้ด้านล่างของสัญลักษณ์ด้วย แต่หากเป็น Actor ที่ไม่ใช่มนุษย์ เช่น ระบบงาน อื่นที่อยู่ นอกเหนือระบบที่เราสนใจ จะใช้รูปสี่เหลี่ยมแล้วเขียนค ว่าเป็น Actor ว่า “<<actor>>” ไว้ด้านบน



ภาพที่ 2.30 Actor

- System Boundary เส้นแบ่งขอบเขตระหว่างระบบกับผู้กระทำต่อระบบ (Use Case กับ Actor) ใช้รูปสี่เหลี่ยมเป็นสัญลักษณ์พร้อมทั้งเขียนชื่อระบบไว้ด้านใน



ภาพที่ 2.31 System Boundary

- Connection คือ เส้นที่ลากเชื่อมต่อระหว่าง Actor กับ Use Case ที่มีปฏิสัมพันธ์กัน ใช้เส้นตรงไม่มี หัวลูกศรเป็นสัญลักษณ์ของ Connection ส่วน Connection ที่ใช้เชื่อมต่อระหว่าง Use Case กับ Use Case กรณีที่ Use Case นั้นมีความสัมพันธ์ซึ่งกันและกัน จะใช้สัญลักษณ์เส้นตรงมีหัวลูกศร พร้อมทั้งเขียนชื่อความสัมพันธ์ไว้ตรงกลางเส้นด้วย โดยเขียนไว้ภายในเครื่องหมาย <<...>>

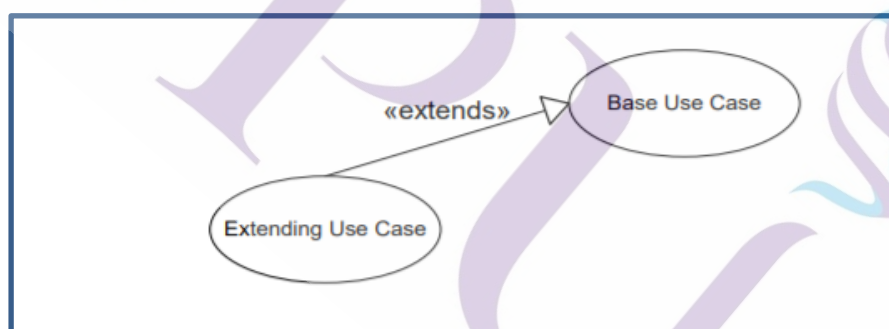


ภาพที่ 2.32 Connection

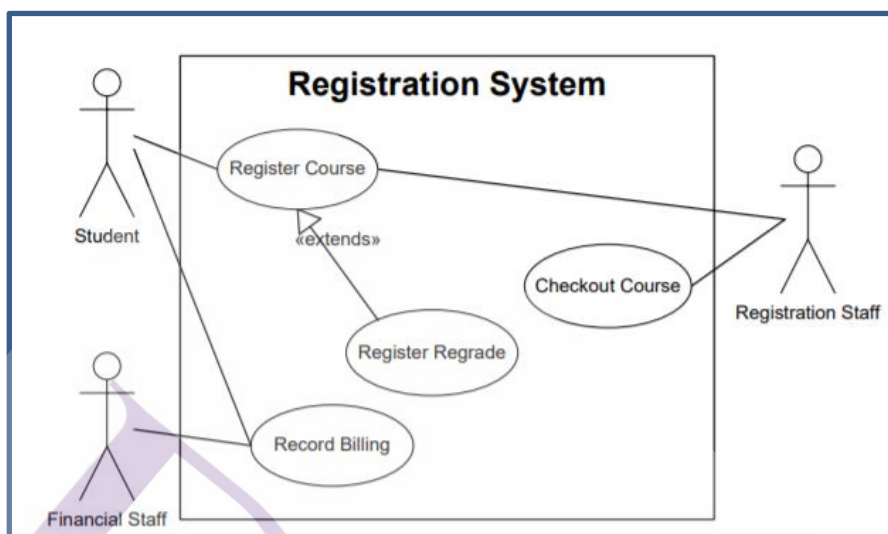
Extend Relationship

เป็นความสัมพันธ์แบบขยายหรือเพิ่ม เกิดขึ้นในกรณีที่บาง Use Case ดำเนินกิจกรรมของตนเองไปตามปกติ แต่อาจจะมีเงื่อนไขหรือสิ่งกระตุ้นบางอย่างที่ส่งผลให้กิจกรรมตามปกติของ Use Case นั้นถูกรบกวนจนเบี่ยงเบนไป ซึ่งเราสามารถแสดงเงื่อนไขหรือสิ่งกระตุ้นเหล่านั้นได้ในรูปของ “Use Case” และเรียกความสัมพันธ์ระหว่าง Use Case ในลักษณะนี้ว่า “Extend Relationship” โดยเรียก Use Case ที่ถูกรบกวนหรือ Use Case ที่ดำเนินงานตามปกติว่า “Base Use Case” และเรียก Use Case ที่ทำหน้าที่รบกวนหรือกระตุ้น Base Use Case ว่า “Extending Use Case”

กล่าวโดยสรุป ก็คือ Use Case หนึ่งทำหน้าที่ตามปกติ เมื่อเกิดเหตุการณ์ผิดปกติขึ้น จะต้องทำหน้าที่พิเศษเพิ่ม โดยหน้าที่พิเศษที่เพิ่มขึ้นก็คือ “Extending Use Case” นั่นเอง ดังนั้น อาจกล่าวได้ว่า Use Case ที่เป็น Extending Use Case จะเกิดขึ้นเพียง บางครั้งเท่านั้น (ไม่ได้เกิดขึ้นทุกครั้งที่เกิด กิจกรรมตาม Base Use Case) การวาด เส้น Connection เชื่อมระหว่าง Use Case ทั้งสอง ให้เริ่มต้นลากเส้นตรงจาก Extending Use Case หันลูกศรชี้ไปที่ Base Use Case



ภาพที่ 2.33 การวาดเส้น Connection Extending Use Case กับ Base Use Case

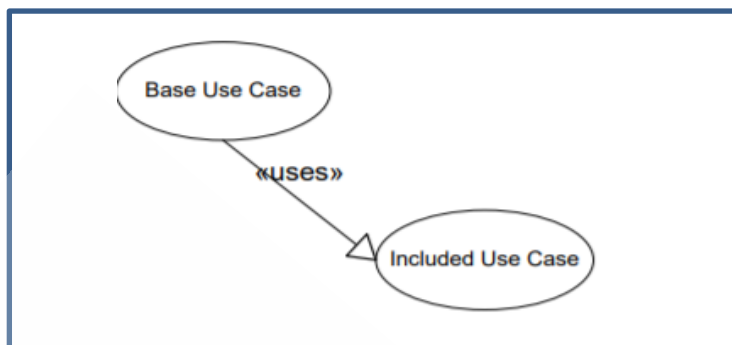


ภาพที่ 2.34 Use Case Diagram ที่มีความสัมพันธ์แบบ Extend Relationship

สังเกตที่ Use Case “Register Course” ซึ่งเป็น Base Use Case คือ ทาหน้าที่ รับลงทะเบียนตามปกติ แต่เมื่อมีเงื่อนไขหรือมีเหตุการณ์พิเศษเกิดขึ้น คือ “นักศึกษาบางคนอาจมีการ ลงทะเบียนเรียนซ้ำ เพื่อปรับเกรดด้วย (Regrade)” จึงได้เพิ่ม Extending Use Case เพื่อมารับ หน้าที่พิเศษดังกล่าว นั่นคือ “Register Regrade”

Include Relationship

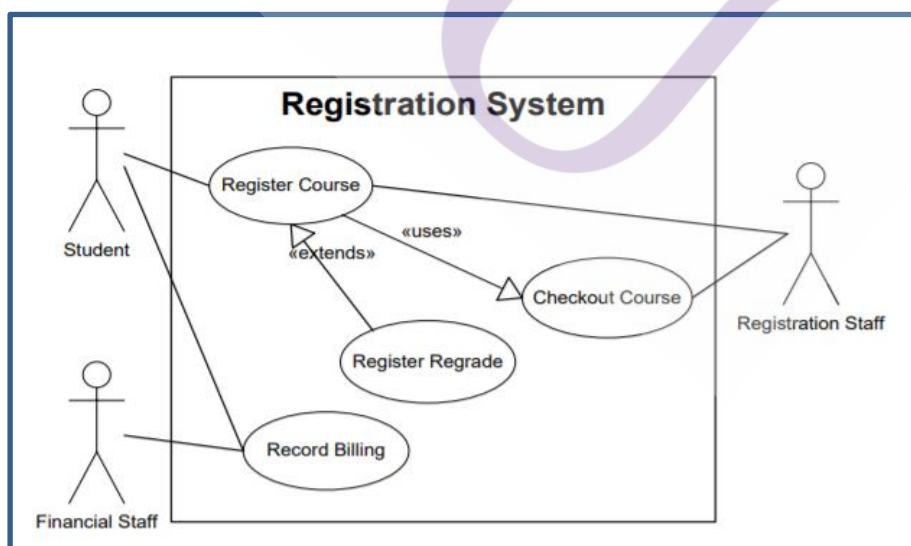
ความสัมพันธ์อีกรูปแบบหนึ่งของ Use Case Diagram ก็ความสัมพันธ์แบบเรียกใช้ เกิดขึ้นในกรณีที่ Use Case หนึ่งไปเรียกหรือดึงกิจกรรมของอีก Use Case หนึ่งมาใช้ เพื่อให้กิจกรรมนั้นเกิดขึ้นจริงใน Use Case ของตนเอง หรือกล่าวให้ง่ายกว่านั้นคือ กิจกรรมใน Use Case หนึ่ง อาจจะถูกผนวกเข้าไปรวมกับกิจกรรมของอีก Use Case หนึ่ง เราเรียกความสัมพันธ์ระหว่าง Use Case ในลักษณะนี้ว่า “Include Relationship” โดย Use Case ที่ทาหน้าที่ดึงกิจกรรมมาจาก Use Case อื่นๆ เรียกว่า “Base Use Case” ในขณะที่ Use Case ที่ถูกเรียก หรือถูกดึงกิจกรรมมาใช้ เรียกว่า “Included Use Case” สามารถเขียนเส้น Connection ได้ในทิศทางตรงกันข้ามกับ Extend Relationship โดยเริ่มต้นลากเส้นตรงจาก Base Use Case หันลูกศรชี้ไปที่ Included Use Case แล้วเขียนชื่อ Relationship “<<user>>” (บางตำราจะใช้คำว่า <<include>>) ไว้ตรงกลางเส้นด้วย



ภาพที่ 2.35 การลากเส้น Connection ระหว่าง Base Use Case กับ Included Use Case

ความสัมพันธ์ระหว่าง Use Case แบบ Include เป็นการสนับสนุนหลักการน กลับมาใช้ใหม่ ของ Use Case (Use Case Reusability) กล่าวคือ Use Case หนึ่งสามารถถูก Include ได้โดย Base Use Case หลายๆ ตัว และสามารถถูก Include ได้มากกว่าหนึ่งครั้งด้วย เช่น ในการท างาน ของระบบเอทีเอ็ม Use Case “การตรวจสอบผู้เข้าใช้ระบบ (Validate User)” สามารถเป็น Included Use Case ให้กับ Base Use Case หลายๆ ตัว ได้แก่ Base Use Case “การถอนเงิน (Withdraw Money)” และ Base Use Case “การโอนเงิน (Transfer Money)”

ดังนั้น เมื่อพิจารณาแล้ว Use Case “ตรวจสอบรายวิชา (Checkout Course)” สามารถถูกเรียกใช้จาก Use Case “ลงทะเบียนเรียน (Register Course)” ได้ ดังนั้น Use Case “Checkout Course” มีความสัมพันธ์กับ Use Case “Register Course” แบบ Include แสดง Use Case Diagram



ภาพที่ 2.36 Use Case Diagram ที่มี Included Relationship

ที่มา <https://med.mahidol.ac.th/ramapharmacy/sites/default/files/public/>

บทที่ 3

วิธีการดำเนินการและเครื่องมือ

ในการจัดทำระบบการบันทึกและการซ่อมบำรุงรถยนต์ นี้ เป็นการดำเนินการโดยใช้ Software Open Source มีกระบวนการเริ่มจากการบันทึกข้อมูลลูกค้า ข้อมูลรถยนต์ ข้อมูลการแจ้งซ่อม และบันทึกการซ่อมรถยนต์ของลูกค้า โดยสามารถใช้งานในรูปแบบ Online ผ่าน Web Application ได้ ซึ่งกระบวนการพัฒนาระบบ จะประกอบไปด้วย

- 3.1 การศึกษาและรวบรวมข้อมูล
- 3.2 การวิเคราะห์ความต้องการของระบบ
- 3.3 ยูสเคสของระบบงาน (System Use Case Diagram)
- 3.4 การออกแบบฐานข้อมูลระบบ (Database Design)

3.1 การศึกษา และรวบรวมข้อมูล

ในการศึกษาและรวบรวมข้อมูล เพื่อพัฒนาระบบการบันทึกและการซ่อมบำรุงรถยนต์ แบบ Online ผ่าน Web Application โดย ผู้พัฒนาได้ทำการศึกษา และรวบรวมข้อมูลที่ต้องการใช้ คือ ข้อมูลลูกค้า ข้อมูลรถยนต์ และข้อมูลการซ่อมบำรุงรถยนต์ รวมถึงเครื่องมือในการพัฒนาระบบ

1. การศึกษาปัญหา ได้แก่ การศึกษาข้อมูลทางระบบปัจจุบัน นั้นใช้ Software ที่มีราคาแพง ระบบไม่สามารถ Online ได้ ซึ่งส่งผลกระทบต่อให้ผู้ปฏิบัติงานเกิดความล่าช้า ฐานข้อมูล (Database) ไม่เหมือนกัน ทำให้ข้อมูลกระจัดกระจาย ยากต่อการทำงานและเกิดปัญหาในเรื่อง การตรวจสอบข้อมูล
2. การรวบรวมและสรุปข้อมูล พบว่า ระบบในปัจจุบันนั้น เป็นระบบในลักษณะโปรแกรมสำเร็จรูปที่ใช้ฐานข้อมูล (Database) ในลักษณะเครื่องคอมพิวเตอร์ 1 เครื่องทำงานเพื่อการเก็บข้อมูล (Stand alone) เมื่อมีการทำงาน เมื่อมีคอมพิวเตอร์อื่น ที่ลง Software เดียวกัน ก็จะทำให้เกิดฐานข้อมูล (Database) ที่มีจำนวนมากขึ้น ทำให้ข้อมูลเกิดการกระจายกันออกไป ยากต่อการค้นหา และสรุปข้อมูลให้ลูกค้า

3. การศึกษาเครื่องมือที่ใช้ในการพัฒนาระบบ มีดังนี้

- 3.1 การใช้ภาษา PHP และ Java script เพื่อการพัฒนาระบบ
- 3.2 ศึกษาการตั้งเครื่องแม่ข่ายเพื่อการจัดการ Web Application คือ XAMPP
- 3.3 การใช้งาน PHPmyAdmin เพื่อการสร้างฐานข้อมูล (Database) ของระบบ

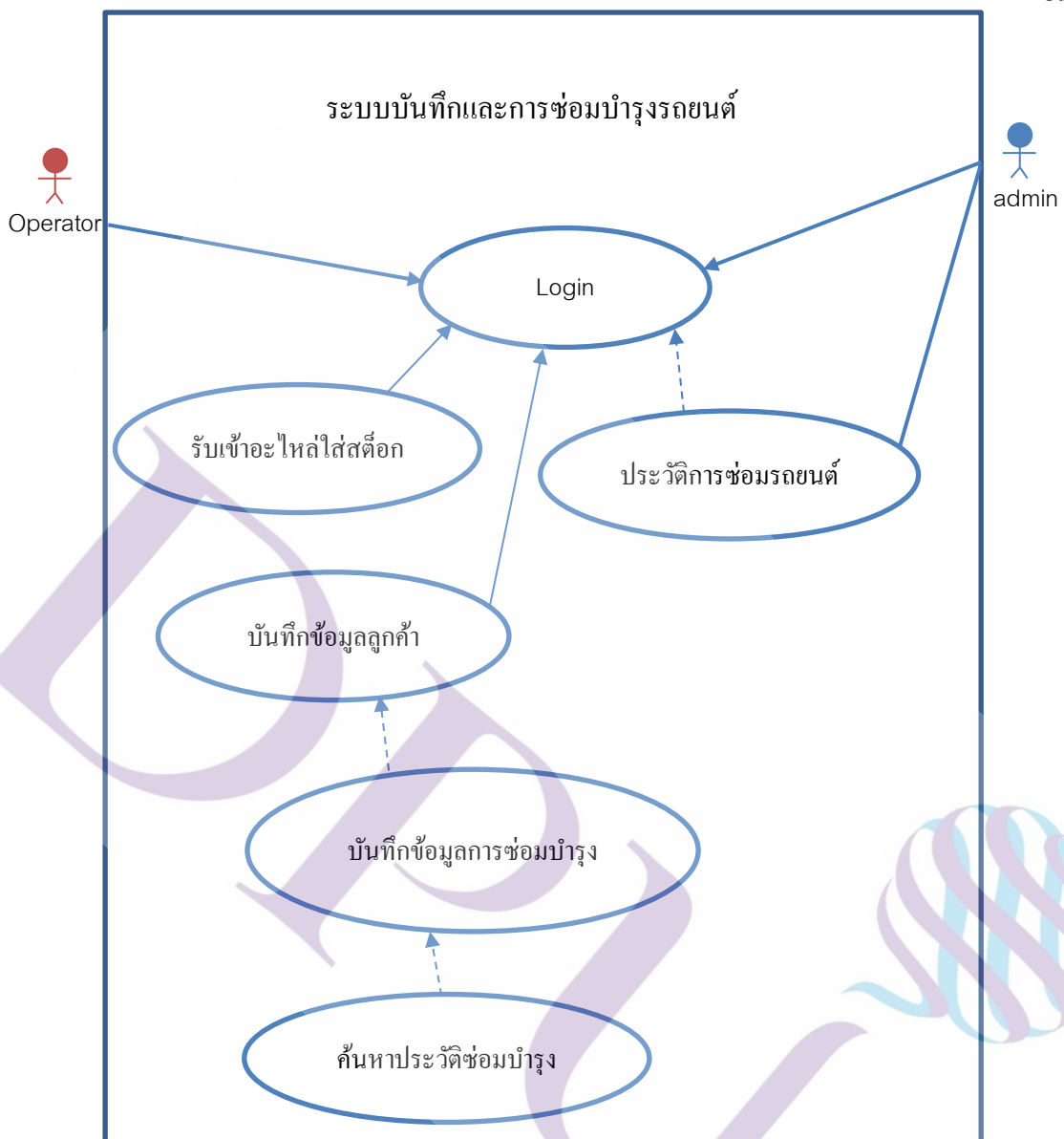
3.2 การวิเคราะห์ความต้องการของระบบ

การวิเคราะห์ความต้องการของระบบในการจัดทำระบบการบันทึกและการซ่อมบำรุงรถยนต์ แบบ Online ผ่าน Web Application ซึ่งวัตถุประสงค์ของการพัฒนาระบบ ผู้พัฒนาได้มีการรวบรวมความต้องการ โดยการสัมภาษณ์เจ้าหน้าที่ ที่ปฏิบัติงาน สามารถสรุปความต้องการได้ดังนี้

1. ความต้องการด้านรูปแบบการใช้งาน (Platform)
 - ระบบมีการทำงาน Online ผ่าน Web Application
 - ระบบสามารถรองรับการใช้งานทุกอุปกรณ์ เช่น PC, Tablet, Smart Phone
2. ความต้องการด้านความสามารถของ Application (Feature)
 - ระบบสามารถแสดงข้อมูลลูกค้า, รถยนต์, การแจ้งซ่อมรถยนต์จากลูกค้าได้
 - ระบบมีการ Login ก่อนเข้าใช้งานระบบ
 - ระบบสามารถบอกสถานะว่ารถยนต์อยู่ในสถานะอะไร
 - ระบบสามารถคำนวณค่าใช้จ่ายในการซ่อมบำรุงได้
 - ระบบสามารถสืบค้นประวัติการซ่อมบำรุงรถยนต์ได้

3.3 ยูสเคสของระบบงาน (System Use Case Diagram)

การวิเคราะห์ระบบบันทึกและการซ่อมบำรุงรถยนต์ แบบ Online ผ่าน Web Application สามารถแบ่งได้ 2 กลุ่มคือผู้ดูแลระบบ (Admin) , เจ้าหน้าที่ปฏิบัติงาน (Operation) จากการศึกษาและวิเคราะห์ข้อมูลทำให้ทราบถึงกระบวนการทำงานของระบบ ข้อมูลที่จะนำเข้าสู่ระบบ และความสัมพันธ์ของระบบกับผู้ใช้งาน จึงได้ใช้แผนภาพ Unified Modeling Language (UML) ดังภาพ



ภาพที่ 3.1 Use Case Diagram ระบบการบันทึกและการซ่อมบำรุงรถยนต์

ตารางที่ 3.1 รายชื่อ Use Case ระบบบันทึกและการซ่อมบำรุงรถยนต์

รหัส Use Case	ชื่อ Use Case	คำอธิบาย
Use Case UC1	การบันทึกข้อมูลลูกค้าและรถยนต์	เจ้าหน้าที่ปฏิบัติงานบันทึกข้อมูลลูกค้าและรถยนต์จากลูกค้าที่เข้ามาใช้บริการ
Use Case UC2	การบันทึกข้อมูลการซ่อมบำรุงรถยนต์	การบันทึกข้อมูลรถยนต์เมื่อมีการซ่อมบำรุงรถยนต์ของลูกค้า



ตารางที่ 3.2 หน้าที่ของ Use Case การบันทึกข้อมูลลูกค้าและรถยนต์

Use Case UC1	การบันทึกข้อมูลลูกค้าและรถยนต์
Primary Actor :	ผู้ปฏิบัติงาน (Operator)
Preconditions :	ผู้ปฏิบัติงาน (Operator) ต้อง Login เพื่อยืนยันผู้มีสิทธิใช้งานระบบ
Success Guarantee (Post conditions) :	ผู้ปฏิบัติงาน (Operator) สามารถกรอกข้อมูลลูกค้า ข้อมูลรถยนต์ ข้อมูลแจ้งซ่อม และข้อมูลการซ่อมบำรุงรถยนต์ และสามารถ เพิ่ม ลบ แก้ไข ข้อมูลต่างๆ ได้
Main Success Scenario (Basic Flow):	<ul style="list-style-type: none"> - ผู้ปฏิบัติงาน (Operator) ดำเนินการ Login เข้าใช้งานโดยการกรอก User และ Password ของผู้ปฏิบัติงาน - กดเข้าไปที่ “ผู้ดูแลระบบ” จากนั้นให้เลือก “จัดการลูกค้า” หรือ “จัดการข้อมูลรถ” เพื่อทำการเพิ่ม แก้ไข ลบ ข้อมูลลูกค้าและรถยนต์
Alternative Flows (Extension)	-

ตารางที่ 3.3 หน้าหนึ่งของ Use Case การบันทึกข้อมูลการซ่อมบำรุงรถยนต์

Use Case UC2	การบันทึกข้อมูลการซ่อมบำรุงรถยนต์
Primary Actor :	ผู้ปฏิบัติงาน (Operator)
Preconditions :	ผู้ปฏิบัติงาน (Operator) ต้อง Login เพื่อยืนยันผู้มีสิทธิใช้งานระบบ
Success Guarantee (Post conditions) :	ผู้ปฏิบัติงาน (Operator) บันทึก แก้ไข ลบ ข้อมูลการซ่อมบำรุงรถยนต์ลูกค้านำได้
Main Success Scenario (Basic Flow):	<ul style="list-style-type: none"> - ผู้ปฏิบัติงาน (Operator) ดำเนินการ Login เข้าใช้งานโดยการกรอก User และ Password ของผู้ปฏิบัติงาน - กดเข้าไปที่ “ผู้ดูแลระบบ” จากนั้นให้เลือก “จัดการลูกค้า” หรือ “จัดการข้อมูลรถ” เพื่อทำการเพิ่ม แก้ไข ลบ ข้อมูลลูกค้าและรถยนต์ - ระบบแสดงวันที่ และเวลาที่ทำการปิดงาน เพื่อรอให้ลูกค้ามารับรถ - ระบบแสดงข้อมูลวันที่ เวลาที่ลูกค้ามารับรถ
Alternative Flows (Extension)	-

3.4 การออกแบบฐานข้อมูลระบบ (Database Design)

ER-Diagram



ภาพที่ 3.2 ER-Diagram ระบบการบันทึกและการซ่อมบำรุงรถยนต์

ในการพัฒนาระบบการบันทึกและการซ่อมบำรุงรถยนต์ ได้ออกแบบตารางฐานข้อมูลไว้ โดยมีรายละเอียดดังต่อไปนี้

ตารางที่ 3.4 ข้อมูลรถยนต์

ชื่อตาราง		Tblcar			
คำอธิบาย		ตารางข้อมูลเกี่ยวกับรถยนต์ของลูกค้า			
No.	Fifld Name	Description	Data Type	Length	PK/FK
1	carID	Id รถยนต์	Int	11	PK
2	carCID	ID รถยนต์ของลูกค้า	int	11	
3	carRegID	หมายเลขทะเบียนรถยนต์	Varchar	50	
4	carBrand	ยี่ห้อรถยนต์	Varchar	20	
5	carModel	รุ่นรถยนต์	Varchar	30	
6	carChassis	หมายเลขตัวถังรถยนต์	Varchar	30	
7	carColor	สีรถยนต์	Varchar	15	
8	carYear	ปีรถยนต์	Int	4	
9	carFule	เชื้อเพลิงรถยนต์	Varchar	100	
10	carCateID	ประเภทรถยนต์	Int	11	FK

ตารางที่ 3.5 ประเภทรถยนต์

ชื่อตาราง		Tblcarcate			
คำอธิบาย		ตารางข้อมูลเกี่ยวกับรถยนต์ของลูกค้า			
No.	Fifld Name	Description	Data Type	Length	PK/FK
1	carCateID	Id ประเภทรถยนต์	Int	11	PK
2	carCateName	ประเภทรถยนต์	varchar	100	
3	carCateDate	วันที่รับรถยนต์	date		
4	carCateTime	เวลารับรถยนต์	Time		

ตารางที่ 3.6 ข้อมูลลูกค้า

ชื่อตาราง		Tblcustomer			
คำอธิบาย		ตารางข้อมูลเกี่ยวกับรถยนต์ของลูกค้า			
No.	Field Name	Description	Data Type	Length	PK/FK
1	cID	Id ลูกค้า	Int	11	PK
2	cRID	หมายเลขบัตรประชาชน	Varchar	13	
3	cTitle	คำนำหน้าของลูกค้า	Varchar	10	
4	cName	ชื่อลูกค้า	Varchar	100	
5	cSurName	นามสกุลลูกค้า	Varchar	100	
6	cAddress	ที่อยู่ลูกค้า	Text		
7	cTumbon	ตำบล	Varchar	100	
8	cAmphoe	อำเภอ	Varchar	100	
9	cProvince	จังหวัด	Varchar	100	
10	cPostCode	รหัสไปรษณีย์	Int	5	
11	cEmail	Email ลูกค้า	Varchar	50	
12	cUsername	Username สำหรับการ Login	Varchar	100	
13	cPassword	Password สำหรับการ Login	Varchar	100	
14	carID	ทะเบียนรถยนต์	Int	11	

ตารางที่ 3.7 คลังอะไหล่รถยนต์

ชื่อตาราง		Tblpartstock			
คำอธิบาย		ตารางข้อมูล Stock อะไหล่รถยนต์			
No.	Fifld Name	Description	Data Type	Length	PK/FK
1	PartSID	ID ของอะไหล่รถยนต์	Int	11	PK
2	partSName	ชื่ออะไหล่รถยนต์	Varchar	100	
3	partSPrice	ราคาอะไหล่รถยนต์	Decimal	10,0	
4	partSAmount	จำนวนอะไหล่รถยนต์	Int	11	
5	partSDate	วันที่รับเข้าอะไหล่รถยนต์	Date		
6	partSTime	เวลาที่รับเข้าอะไหล่รถยนต์	Time		
7	partSStaff	พนักงานที่รับเข้าอะไหล่รถยนต์	Int	11	

ตารางที่ 3.8 การบันทึกการแจ้งซ่อมและบันทึกการซ่อมบำรุงรถยนต์

ชื่อตาราง		Tblservice			
คำอธิบาย		ตารางการบันทึกการแจ้งซ่อมและบันทึกการซ่อมบำรุงรถยนต์			
No.	Fifld Name	Description	Data Type	Length	PK/FK
1	sID	หมายเลข ID Service	Int	11	PK
2	sDate	วันที่ Service	Date		
3	sTime	เวลา Service	Time		
4	mID	ID ของพนักงาน	Int	11	
5	cID	ID ลูกค้า	Int	11	
6	reMark	รับแจ้งซ่อมจากลูกค้า	Text		
7	carID	ทะเบียนรถยนต์	Int	11	
8	Service_status_id	สถานะการซ่อมบำรุงรถยนต์	Int	11	
9	Remark_engineer	บันทึกการซ่อมบำรุงของช่าง	Varchar	255	
10	Close_datetime	วันที่และเวลาปิดการซ่อม	Datetime		
11	Receipt_car_date	วันที่และเวลาลูกค้ามารับรถ	Datetime		

ตารางที่ 3.9 การคิดค่าอะไหล่รถยนต์

ชื่อตาราง		Tblservice_subpart			
คำอธิบาย		ตารางข้อมูลคำนวณค่าบริการ			
No.	Fifld Name	Description	Data Type	Length	PK/FK
1	Id_subpart	Id การคิดค่าบริการ	int	11	PK
2	partSID	หลายเลขอะไหล่รถยนต์	int	11	
3	Amount_part	จำนวนอะไหล่	int	2	
4	Price_part	ราคาอะไหล่	decimal	20,0	
5	Service_tpye_id	ประเภทการ Service	int	11	
6	Service_feecharge	ราคา Service	decimal	20,0	

ตารางที่ 3.10 แสดงผู้มีสิทธิใช้ระบบ

ชื่อตาราง		Tblmember			
คำอธิบาย		ตารางแสดงข้อมูลสมาชิกที่ใช้งานระบบ			
No.	Fifld Name	Description	Data Type	Length	PK/FK
1	mID	ID สมาชิกใช้งานระบบ	int	11	PK
2	mTitle	คำนำหน้าชื่อสมาชิก	varchar	10	
3	mName	ชื่อสมาชิก	varchar	100	
4	mSurName	นามสกุลสมาชิก	varchar	100	
5	mAddress	ที่อยู่สมาชิก	Text		
6	mTumbon	ตำบล	varchar	100	
7	mAmphoe	อำเภอ	varchar	100	
8	mPostCode	รหัสไปรษณีย์	int	5	
9	mEmail	อีเมลสมาชิก	varchar	50	
10	mUsername	Username สำหรับ login	varchar	100	
11	mPassword	Password สำหรับ login	varchar	100	
12	mPicture	รูปภาพสมาชิก	varchar	10	
13	mTID	ลำดับสมาชิก	int	11	FK

ตารางที่ 3.11 บอกลักษณะรถยนต์

ชื่อตาราง	Service_status				
คำอธิบาย	ตารางข้อมูลบอกลักษณะการซ่อมรถยนต์				
No.	Fifld Name	Description	Data Type	Length	PK/FK
1	Service_status_id	Id สถานะซ่อมรถยนต์	int	11	PK
2	Service_status_type	ประเภทสถานะซ่อมรถยนต์	varchar	25	

ตารางที่ 3.12 บอกลการคิดประเภทค่าแรง

ชื่อตาราง	Service_type				
คำอธิบาย	ตารางบอกลการคิดประเภทค่าแรง				
No.	Fifld Name	Description	Data Type	Length	PK/FK
1	Service_status_id	Id สถานะซ่อมรถยนต์	int	11	PK
2	Service_status_name	สถานะการซ่อมรถยนต์	varchar	25	

บทที่ 4

การวิเคราะห์ ออกแบบและพัฒนาระบบ

การวิเคราะห์และออกแบบระบบการบันทึกและการซ่อมบำรุงรถยนต์ นี้มีโครงสร้างของระบบเป็น Web Application ที่ทำงานบน Internet เพื่อให้ผู้ใช้งานสามารถเข้าถึงระบบได้ง่าย และมีฐานข้อมูลเดียวกัน จะมีการแบ่งความคิดเห็นของข้อมูลออกตามสิทธิ์ของผู้ใช้งานประกอบด้วย

1. ผู้ใช้งานทั่วไป
2. ผู้ปฏิบัติงานที่เกี่ยวข้อง
3. ผู้ดูแลระบบ

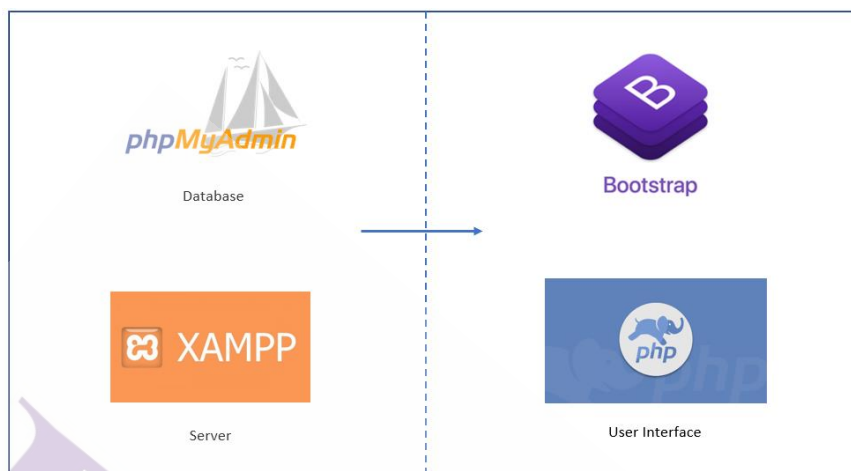
4.1 การออกแบบระบบในภาพรวม

การออกแบบระบบในภาพรวมของระบบการบันทึกและการซ่อมบำรุงรถยนต์ ตามภาพ 4.1 ประกอบด้วย

1. การออกแบบ Web Application ในรูปแบบ (Web Responsive) ให้รองรับการใช้งานทุกอุปกรณ์ มีเทคโนโลยี Bootstrap ซึ่งเป็นชุดเครื่องมือที่มีส่วนประกอบของ HTML5 และ CSS มาช่วยในการออกแบบ Web Responsive ซึ่งจะเป็นอุปกรณ์ใดก็ตาม สามารถจัดการได้ง่าย และได้ใช้ PHP มาทำหน้าที่ดึงฐานข้อมูล

2. Database มีหน้าที่ในการเก็บข้อมูลต่างๆ ในระบบ phpMyAdmin

3. เครื่องแม่ข่าย (Web Server) ใช้ XAMPP มาใช้ในการจำลอง Web Server ที่เครื่องลูกข่าย



ภาพที่ 4.1 ระบบการบันทึกและการซ่อมบำรุงรถยนต์

4.2 การออกแบบภาพแสดงผล

ส่วนติดต่อกับผู้ใช้ระบบในกลุ่มต่างๆ มีดังนี้

1. การออกแบบส่วนต่อประสานผู้ใช้งาน User Interface Design

การออกแบบส่วนติดต่อกับผู้ใช้งานระบบ คือ การออกแบบส่วนหน้าที่โต้ตอบระหว่างผู้ใช้งานกับระบบ ซึ่งส่วนประกอบการแสดงผลภาพ และ โครงสร้างเมนูในระบบ การออกแบบ Web Application ให้สวยงามด้วย Bootstrap จะทำให้สามารถใช้งานได้กับทุกอุปกรณ์ที่มีขนาดแสดงผลต่างกัน ตั้งแต่จอภาพที่มีขนาดเล็กๆ อย่าง Mobile Phone, ขนาดกลางอย่าง Tablet ไปจนถึงขนาดใหญ่ อย่าง PC Desktop ซึ่งการออกแบบให้รองรับการทำงานกับทุกอุปกรณ์ที่หลากหลายเช่นนี้จะเรียกว่า Responsive Design

Auto Service

ภาพที่ 4.2 โครงร่างจอแสดงผลของระบบ

การออกแบบโครงสร้างส่วนประกอบจอแสดงผลการทำงานของระบบ (Layout) ประกอบด้วยส่วนต่าง ดังนี้

1.1 ส่วนหัวของจอแสดงผล

แสดงภาพสัญลักษณ์ ชื่อระบบการบันทึกและการซ่อมบำรุงรถยนต์

1.2 เมนูระบบ แสดงเมนูการเข้าสู่ระบบ การบันทึกและการซ่อมบำรุงรถยนต์

2. การออกแบบส่วนต่อประสานงานผู้ใช้งาน (User Interface Design) ในมุมมองของการมองเห็นของข้อมูลออกตามสิทธิ์ของ User โดยมีรายละเอียดดังนี้

2.1 ผู้ใช้งานระบบ Operator

ลำดับที่	ยี่ห้อ	รุ่น	ทะเบียน	วันที่รับแจ้งซ่อม	ชื่อลูกค้า	อาการเสีย	ช่างวิเคราะห์	สถานะ	ควบคุม
1	Toyota	Camry	กก-1234	1/5/60	นายสมหวัง	เสียงดังจากเครื่อง	เปลี่ยนสายพาน	อยู่ระหว่างซ่อม	เพิ่ม / แก้ไข / ลบ

ภาพที่ 4.3 การออกแบบจอแสดงผลของผู้ใช้งานระบบหน้าแรก

- แสดงภาพรายการแจ้งซ่อมทั้งหมดของลูกค้า

AutoService

User

หน้าหลัก > รายการอะไหล่รถยนต์

เพิ่มอะไหล่รถยนต์

ชื่ออะไหล่	ราคา	จำนวน	วันที่ - เวลา	พนักงาน	ควบคุม
ไส้กรองน้ำมันเครื่อง	150	10	5/5/60 9:30:00	ศุภฤกษ์	เพิ่ม / แก้ไข / ลบ

รับงานซ่อม
จัดการอะไหล่รถยนต์
รายงานข้อมูลการซ่อม
จัดการข้อมูลลูกค้า
จัดการข้อมูลรถยนต์
ลงชื่อออก

ภาพที่ 4.4 การออกแบบจอแสดงผลของผู้ใช้งานของรายละเอียดอะไหล่

- แสดงภาพรายละเอียดอะไหล่ที่มีสต็อก

AutoService

User

หน้าหลัก > รายงานข้อมูลการซ่อม

กรอกข้อมูลที่ท่านต้องการค้นหา

submit Export to Excel

ลำดับ	ชื่อ - สกุล	ที่อยู่	วันที่ - เวลา
1	นาย สมหวัง รัตน์	10/10 ถนนนารา กรุงเทพฯ	จำนวนรถที่ลูกค้าส่งซ่อม

สาเหตุเสียที่ช่างวิเคราะห์: เปลี่ยนถ่ายน้ำมันเครื่อง ซ่อมพวงมาลัย

ชื่ออุปกรณ์	จำนวนอุปกรณ์	ราคา / ชิ้น	ราคารวม อุปกรณ์	serviceCharge	serviceCharge
ไส้กรองน้ำมัน เครื่อง	1	200	800	Lumpsum	500

รับงานซ่อม
จัดการอะไหล่รถยนต์
รายงานข้อมูลการซ่อม
จัดการข้อมูลลูกค้า
จัดการข้อมูลรถยนต์
ลงชื่อออก

ภาพที่ 4.5 การออกแบบจอแสดงผลภาพของผู้ใช้งานรายงานข้อมูลการซ่อม

- แสดงผลภาพรายละเอียด รายงานข้อมูลการซ่อมรถยนต์ของลูกค้า

AutoService

User

หน้าหลัก > รายชื่อลูกค้า

ลำดับที่	ชื่อ	นามสกุล	ที่อยู่	ตำบล	อำเภอ	จังหวัด	รหัสไปรษณีย์	ปุ่มควบคุม
1	สมหวัง	จิตใจดี	123/456	บางเสาธง	บางใหญ่	นนทบุรี	11140	<input type="button" value="แก้ไข"/> <input type="button" value="ลบ"/>

รับงานซ่อม
จัดการอะไหล่รถยนต์
รายงานข้อมูลการซ่อม
จัดการข้อมูลลูกค้า
จัดการข้อมูลรถยนต์
ลงชื่อออก

ภาพที่ 4.6 การออกแบบจอแสดงผลภาพของผู้ใช้งานรายชื่อลูกค้า

- แสดงภาพรายละเอียด ข้อมูลของลูกค้า

AutoService

User

หน้าหลัก > ข้อมูลรถยนต์

ชื่อลูกค้า	ทะเบียนรถยนต์	ยี่ห้อ	รุ่น	สี	ปุ่มควบคุม
สมหวัง	กก-1234	Toyota	Camry	ดำ	<input type="button" value="แก้ไข"/> <input type="button" value="ลบ"/>

รับงานซ่อม
จัดการอะไหล่รถยนต์
รายงานข้อมูลการซ่อม
จัดการข้อมูลลูกค้า
จัดการข้อมูลรถยนต์
ลงชื่อออก

ภาพที่ 4.7 การออกแบบจอแสดงผลผลของผู้ใช้งานข้อมูลรถยนต์

- แสดงภาพของรายละเอียดข้อมูลรถยนต์ของลูกค้า

AutoService

User

หน้าหลัก > รายชื่อบุคลากร

รับงานซ่อม

จัดการอะไหล่รถยนต์

รายงานข้อมูลการซ่อม

จัดการข้อมูลลูกค้า

จัดการข้อมูลรถยนต์

ลงชื่อออก

เพิ่มเติมรายละเอียดการซ่อมของทะเบียน : *****

เพิ่มตาราง
ลบตาราง
บันทึกข้อมูล

รายละเอียดที่ช่างวิเคราะห์

ราคารวมอุปกรณ์	0
ราคารวม Service Charge	0
รวมราคา	0

คำนวณราคา

ชื่ออุปกรณ์	จำนวน	ราคา / ชิ้น	ราคารวมอุปกรณ์	Service Charge	ราคา Service Charge

ภาพที่ 4.8 การออกแบบจอแสดงผลของผู้ใช้งานเพิ่มรายละเอียดการซ่อม

- แสดงภาพของรายละเอียดข้อมูลการเพิ่มรายการซ่อมรถยนต์ของลูกค้า

AutoService

User

หน้าหลัก > รายละเอียดอะไหล่รถยนต์ > แบบฟอร์มเพิ่มข้อมูลอะไหล่รถยนต์

รับงานซ่อม

จัดการอะไหล่รถยนต์

รายงานข้อมูลการซ่อม

จัดการข้อมูลลูกค้า

จัดการข้อมูลรถยนต์

ลงชื่อออก

แบบฟอร์มเพิ่มข้อมูลอะไหล่รถยนต์

ชื่ออะไหล่*

ราคา*

จำนวน*

วันที่*

MM/DD/YYYY

--:--

บันทึก
ยกเลิก

ภาพที่ 4.9 การออกแบบจอแสดงผลของผู้ใช้งานเพิ่มอะไหล่

- แสดงภาพของรายละเอียดข้อมูลอะไหล่

The screenshot shows the 'AutoService' interface. On the left is a navigation menu with buttons: 'รับงานซ่อม', 'จัดการอะไหล่รถยนต์', 'รายงานข้อมูลการซ่อม', 'จัดการข้อมูลลูกค้า', 'จัดการข้อมูลรถยนต์', and 'ลงชื่อออก'. The main content area has a breadcrumb 'หน้าหลัก > เพิ่มข้อมูลลูกค้า'. The 'เพิ่มข้อมูลลูกค้า' form contains the following fields:

- หมายเลขบัตรประชาชน (National ID Number)
- ชื่อ (Name) / กรอกชื่อ (Enter name)
- ที่อยู่ (Address) / กรอกที่อยู่ (Enter address)
- อำเภอ (District) / กรอกอำเภอ (Enter district)
- รหัสไปรษณีย์ (Postal Code) / กรอกรหัสไปรษณีย์ (Enter postal code)
- ตำแหน่ง (Position) / เลือก (Select)
- นามสกุล (Surname) / กรอกนามสกุล (Enter surname)
- ตำบล (Sub-district) / กรอกตำบล (Enter sub-district)
- จังหวัด (Province) / กรอกจังหวัด (Enter province)

Buttons at the bottom right are 'บันทึก' (Save) and 'ยกเลิก' (Cancel).

ภาพที่ 4.10 การออกแบบจอแสดงผลของผู้ใช้งานเพิ่มข้อมูลลูกค้า

- แสดงภาพของรายละเอียดการเพิ่มข้อมูลลูกค้า

The screenshot shows the 'AutoService' interface. On the left is a navigation menu with buttons: 'รับงานซ่อม', 'จัดการอะไหล่รถยนต์', 'รายงานข้อมูลการซ่อม', 'จัดการข้อมูลลูกค้า', 'จัดการข้อมูลรถยนต์', and 'ลงชื่อออก'. The main content area has a breadcrumb 'หน้าหลัก > เพิ่มข้อมูลรถยนต์'. The 'เพิ่มข้อมูลรถยนต์' form contains the following fields:

- เลือกข้อมูล (Select information) / เลือกข้อมูล (Select information)
- ทะเบียนรถยนต์ (Vehicle License Plate) / กรอกทะเบียนรถยนต์ (Enter vehicle license plate)
- รุ่น (Model) / กรอกรุ่น (Enter model)
- สี (Color) / กรอกสี (Enter color)
- ชื่อเพลิง (Fuel Name) / กรอกชื่อเพลิง (Enter fuel name)
- ประเภท (Type) / เลือกประเภท (Select type)
- ยี่ห้อ (Brand) / กรอกยี่ห้อ (Enter brand)
- หมายเลขตัวถัง (Chassis Number) / กรอกเลขตัวถัง (Enter chassis number)
- ปี (Year) / กรอกปี (Enter year)

Buttons at the bottom right are 'บันทึก' (Save) and 'ยกเลิก' (Cancel).

ภาพที่ 4.11 การออกแบบจอแสดงผลของผู้ใช้งานเพิ่มข้อมูลรถยนต์

- แสดงภาพของรายละเอียดการเพิ่มข้อมูลรถยนต์

4.3 การพัฒนาระบบ

การพัฒนาระบบ ผู้พัฒนาต้องมีความรู้เกี่ยวกับกระบวนการจัดทำข้อมูล กระบวนการจัดตั้งเครื่องคอมพิวเตอร์แม่ข่ายที่ให้บริการการบันทึกและการซ่อมบำรุงรถยนต์ รวมถึงการพัฒนา Web Application ซึ่งในแต่ละกระบวนการจะมีการใช้โปรแกรมต่างๆ ที่เกี่ยวข้องและภาษาที่ใช้พัฒนาระบบ นั้นประกอบไปด้วย

1. กระบวนการจัดทำข้อมูล รวบรวมข้อมูลต่างๆเกี่ยวกับบริการซ่อมบำรุงรถยนต์ประกอบไปด้วย

- ข้อมูลลูกค้า
- ข้อมูลรถยนต์
- ข้อมูลการแจ้งซ่อมบำรุงของลูกค้า
- ข้อมูลการบันทึกซ่อมบำรุงรถยนต์
- ข้อมูลอะไหล่

2. กระบวนการจัดตั้งเครื่องแม่ข่าย

การติดตั้งและปรับตั้งค่าของเครื่องแม่ข่ายเพื่อให้สามารถใช้งาน Web Application ซึ่งจะใช้โปรแกรม XAMPP

3. กระบวนการพัฒนา Web Application ที่แสดงระบบการบันทึกและการซ่อมบำรุงรถยนต์ผ่าน คอมพิวเตอร์, Notebook, Smart Phone หรือ Tablet ซึ่งประกอบด้วยส่วนของผู้ใช้งาน ส่วนของการพัฒนาระบบประกอบไปด้วย

- PHP ใช้งานร่วมกับ JavaScript สำหรับ Web Application ที่พัฒนา จะใช้ภาษา PHP
- Bootstrap ใช้งานร่วมกับ HTML, CSS และ JavaScript เข้าด้วยกันสำหรับพัฒนา Web Application

4.4 การทดสอบและประเมินผลระบบ

- การทดสอบโดยผู้พัฒนา Web Application ใช้วิธีการทดสอบแบบ Black Box Testing) เป็นการทดสอบการทำงานของระบบโดยรวมทั้งหมดว่ามีกระบวนการทำงานถูกต้องตามวัตถุประสงค์หรือไม่
- การประเมินผลระบบ ดำเนินการ โดยทำการประเมินความพึงพอใจของผู้ใช้งานระบบด้วย ผู้ใช้งานระบบจำนวน 5 คน พร้อมด้วยนำผลที่ได้มาหาสถิติ

บทที่ 5

ผลการดำเนินงาน

ระบบการบันทึกข้อมูลการซ่อมบำรุงรถยนต์ จัดทำเพื่อให้มีระบบสารสนเทศของระบบการบันทึกข้อมูลการซ่อมบำรุงรถยนต์ แบบ Online ผ่าน Web Application โดยการประยุกต์ใช้เทคโนโลยี Web base ช่วยให้การบันทึกข้อมูลการซ่อมบำรุงรถยนต์ผ่านระบบ Internet พนักงานผู้ปฏิบัติงานสามารถนำข้อมูลที่ได้ไป สามารถใช้งานร่วมกันได้ สะดวกรวดเร็ว และใช้ฐานข้อมูลเดียวกันทั้งองค์กร ผู้พัฒนาระบบจึงได้สรุปผลการทำดำเนินงานดังนี้

5.1 ผลการออกแบบและพัฒนาระบบ

- การเข้าใช้งานระบบ
- การบันทึกข้อมูลลูกค้า
- การบันทึกข้อมูลรถยนต์
- การบันทึกข้อมูลอะไหล่
- การบันทึกการรับแจ้งการซ่อมบำรุงรถยนต์
- การบันทึกข้อมูลการซ่อมบำรุงรถยนต์
- รายงานการซ่อมบำรุงรถยนต์

5.2 การสำรวจความพึงพอใจของผู้ใช้ระบบ

5.1 ผลการออกแบบและพัฒนาระบบ

การเข้าใช้งานระบบ

ผู้ใช้งานระบบสามารถใช้งานได้อย่างรวดเร็ว เนื่องจากดำเนินการผ่าน Internet ซึ่งผู้ใช้งานระบบต้องดำเนินการ Login ผ่านระบบตามภาพ 5.1 แสดงผลภาพจอการ Login เมื่อ Login ผ่านเรียบร้อยแล้ว ผู้ใช้งานระบบสามารถใช้งานระบบการบันทึกข้อมูลการซ่อมบำรุงรถยนต์ได้

ภาพที่ 5.1 จอแสดงผล Login

5.1.1 การบันทึกข้อมูลลูกค้า

เมื่อมีลูกค้าเข้ามาใช้บริการซ่อมบำรุงรถยนต์ ผู้ปฏิบัติงานจำเป็นที่จะต้องบันทึกข้อมูลลูกค้าต่างๆ ดังภาพที่ 5.2 การบันทึกข้อมูลลูกค้า เพื่อทางผู้ปฏิบัติงานจะได้มีข้อมูลลูกค้าที่เข้ามาใช้บริการ ซึ่งข้อมูลนั้นจะประกอบไปด้วย

- หมายเลขบัตรประชาชน
- คำนำหน้า
- ชื่อ
- นามสกุล
- ที่อยู่
- ตำบล
- อำเภอ
- จังหวัด
- รหัสไปรษณีย์
- อีเมลล์

The screenshot displays the 'เพิ่มข้อมูลลูกค้า' (Add Customer Information) page in the AutoService system. The page has a sidebar on the left with navigation links: 'รายการซ่อม', 'จัดการอะไหล่รถยนต์', 'รายงาน', 'ผู้ดูแลระบบ', and 'ลงชื่อออก'. The main content area contains a form with the following fields:

- หมายเลขบัตรประชาชน (ID Card Number)
- ชื่อ (Name) - split into 'ชื่อ' (First Name) and 'ที่อยู่' (Last Name)
- ที่อยู่ (Address) - split into 'ที่อยู่' (Street) and 'อำเภอ' (District)
- รหัสไปรษณีย์ (Postal Code)
- รถสีไปรษณีย์ (Vehicle Color)
- รถสีไปรษณีย์ (Vehicle Color) - split into 'สี' (Color) and 'แบบ' (Type)
- จำนวนล้อ (Number of Wheels)
- จำนวนล้อ (Number of Wheels) - split into 'จำนวน' (Number) and 'ล้อ' (Wheels)
- จังหวัด (Province)
- จังหวัด (Province) - split into 'จังหวัด' (Province) and 'เมือง' (City)
- เมือง (City)
- เมือง (City) - split into 'เมือง' (City) and 'เขต' (District)

At the bottom of the form, there are buttons for 'บันทึก' (Save) and 'ยกเลิก' (Cancel). The footer of the page reads: '© 2019 AutoService. All Rights Reserved | Design by Supharoek.w'.

ภาพที่ 5.2 การบันทึกข้อมูลลูกค้า

5.1.2 การบันทึกข้อมูลรถยนต์

เมื่อลูกค้าเข้ามาใช้บริการ การซ่อมบำรุงรถยนต์ หลังจากบันทึกข้อมูลของลูกค้าแล้ว ผู้ปฏิบัติงานจำเป็นต้องบันทึกข้อมูลรถยนต์ เพื่อให้ผู้ปฏิบัติงานจะต้องทราบรายละเอียดต่างๆ ของรถยนต์ ตามภาพที่ 5.3 การบันทึกข้อมูลรถยนต์ลูกค้า โดยข้อมูลรถยนต์ของลูกค้านั้นประกอบไปด้วย

- ชื่อลูกค้า
- ประเภทรถยนต์
- ทะเบียนรถยนต์
- ยี่ห้อ
- รุ่น
- หมายเลขตัวถัง
- สี
- ปี
- เชื้อเพลิง

เมื่อมีการบันทึกข้อมูลในส่วนข้อมูลรถยนต์แล้ว ทางผู้ปฏิบัติงาน จะทราบว่า รถยนต์เป็นของลูกค้าชื่ออะไร และรายละเอียดของรถยนต์นั้นมีข้อมูลอะไรบ้าง เนื่องจากการซ่อมบำรุงรถยนต์นั้นจะมีลักษณะการซ่อมบำรุงและการใช้อะไหล่ที่ต่างกันออกไป

ภาพที่ 5.3 การบันทึกข้อมูลรถยนต์ลูกค้า

5.1.3 การบันทึกข้อมูลอะไหล่

ในการซ่อมบำรุงรถยนต์ ผู้ซ่อมบำรุงจำเป็นต้องมีอะไหล่รถยนต์ในสต็อกสินค้า เนื่องจากรถยนต์แต่ละคันใช้อะไหล่แตกต่างกันออกไป เช่น ใส้กรองน้ำมันเครื่อง, ใส้กรองอากาศ, สายพานเครื่องยนต์ ฯลฯ ดังนั้นการมีบันทึกอะไหล่ในสต็อกสินค้า จึงเป็นระบบหนึ่งที่จะต้อง มี ซึ่งจะนำไปตามภาพที่ 5.4 การบันทึกอะไหล่รถยนต์ ซึ่งข้อมูลนั้นจะประกอบไปด้วย

- ชื่ออะไหล่
- ราคา
- จำนวน
- วันที่ เวลา

ภาพที่ 5.4 การบันทึกอะไหล่รถยนต์

5.1.4 การบันทึกการรับแจ้งการซ่อมบำรุงรถยนต์

การบันทึกการรับแจ้งการซ่อมบำรุงรถยนต์นั้น เป็นการบันทึกจากการแจ้งจากลูกค้า ถึงอาการผิดปกติของรถยนต์ เพื่อให้เป็นข้อมูลสำหรับการวิเคราะห์หาสาเหตุของช่าง เพื่อให้ช่างซ่อมบำรุงสามารถวางแผนการซ่อมและการจัดหาอะไหล่ที่ใช้งานกับรถยนต์ได้ ซึ่งจะเป็นไปตามภาพที่ 5.5 การบันทึกการรับแจ้งการซ่อมบำรุงรถยนต์ ข้อมูลการรับแจ้งจะประกอบไปด้วย

- ชื่อ – นามสกุล ลูกค้า
- ทะเบียนรถยนต์
- สาเหตุอาการเสียเบื้องต้น

The screenshot shows a web application interface for 'AutoService'. The main content area is titled 'กรอกแบบฟอร์มแจ้งซ่อม' (Fill in the repair form). It contains several input fields: 'ชื่อ-นามสกุลลูกค้า' (Customer name and surname), 'ทะเบียนรถยนต์' (Vehicle license plate), and 'สาเหตุอาการเสียเบื้องต้น' (Initial symptoms). There are also 'บันทึก' (Save) and 'ยกเลิก' (Cancel) buttons at the bottom. The page has a sidebar with navigation options like 'รายการซ่อม' (Repair list) and 'จัดการอะไหล่รถยนต์' (Manage car parts).

ภาพที่ 5.5 การบันทึกการรับแจ้งการซ่อมบำรุงรถยนต์

5.1.5 การบันทึกข้อมูลการซ่อมบำรุงรถยนต์

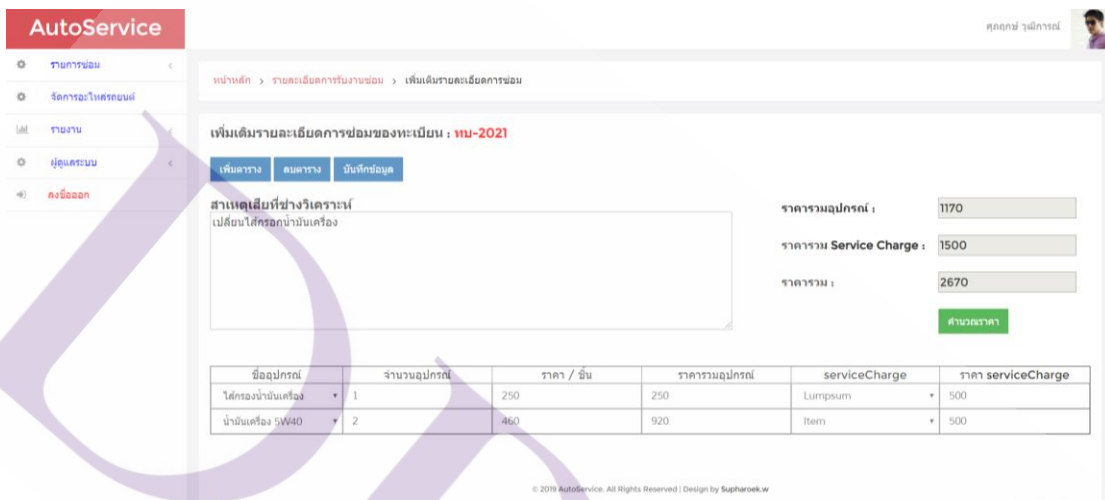
การบันทึกข้อมูลการซ่อมบำรุงรถยนต์ เป็นการบันทึกจากผู้ปฏิบัติงานว่า รถยนต์คันที่ใช้บริการการซ่อมบำรุงนั้น มีอาการอย่างไร ซ่อมบำรุงอย่างไร เปลี่ยนอะไหล่อะไรบ้าง คิดค่าบริการเท่าไรตามภาพที่ 5.6 การบันทึกการซ่อมบำรุงรถยนต์ ซึ่งข้อมูลในการบันทึกการซ่อมบำรุงรถยนต์นั้นจะประกอบไปด้วย

- “สาเหตุเสียที่ช่างวิเคราะห์” เป็นช่องสำหรับช่างบันทึกว่าได้ทำการซ่อมบำรุงอย่างไรบ้าง
- “ชื่ออุปกรณ์” เป็นช่องสำหรับแสดงอะไหล่ที่เปลี่ยนเพื่อการซ่อมบำรุงรถยนต์ลูกค้า
- “จำนวนอะไหล่ที่เปลี่ยน”
- “ราคา/ชิ้น” รวมราคาอุปกรณ์

เป็นการนำจำนวนอะไหล่ที่เปลี่ยน คูณกับราคา/ชิ้น จะกลายเป็นราคารวมของอะไหล่รถยนต์ที่เปลี่ยน

ServiceCharge

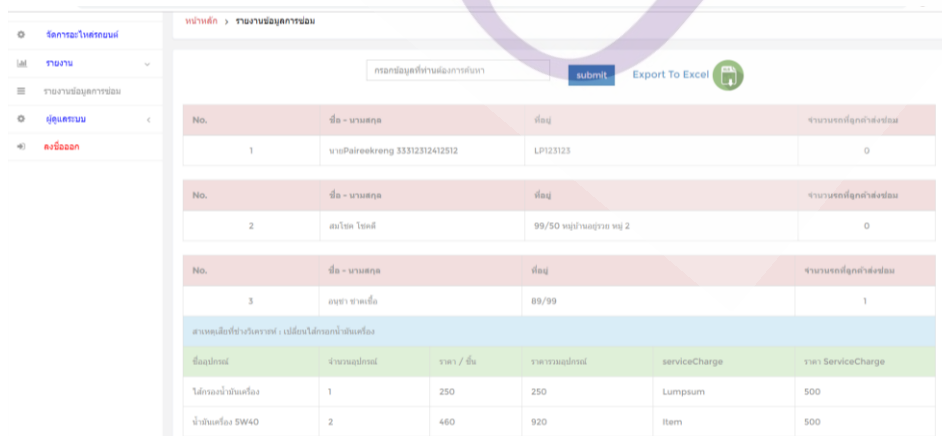
- Lumpsum เป็นการคิดค่าบริการแบบเหมาค่าแรง
- Item เป็นการคิดค่าบริการแบบเฉพาะชิ้นส่วน



ภาพที่ 5.6 การบันทึกการซ่อมบำรุงรถยนต์

5.1.6 รายงานการซ่อมบำรุงรถยนต์

เป็นการแสดงรายงานการซ่อมบำรุงรถยนต์ของลูกค้าแต่ละคัน ซึ่งจะแสดงข้อมูลหลังจากมีการบันทึกการซ่อมแล้ว เพื่อให้เป็นประวัติการซ่อมเพื่อให้ผู้ปฏิบัติงานหน้าที่ได้รับทราบประวัติการซ่อมรถยนต์ต่างๆของลูกค้าที่เคยใช้บริการ ซึ่งจะเป็นไปตามภาพที่ 5.7 แสดงประวัติการซ่อมบำรุง



ภาพที่ 5.7 ประวัติการซ่อมบำรุง

5.2 การสำรวจความพึงพอใจของผู้ใช้ระบบ

ในการสำรวจความพึงพอใจของผู้ใช้ระบบการบันทึกข้อมูลการซ่อมบำรุงรถยนต์ ผู้พัฒนาได้สร้างแบบสอบถามเพื่อประเมินระดับความพึงพอใจของระบบที่พัฒนาขึ้น และให้กลุ่มผู้ใช้งานที่กำหนดเข้าตอบ ซึ่งมีดังนี้

5.2.1 กลุ่มผู้ใช้งาน

1. กลุ่มตัวอย่างของเจ้าหน้าที่ที่เกี่ยวข้องภายในบริษัท ธนบุรีพานิช จำกัด

5.2.2 วิธีการสำรวจความพึงพอใจ

ใช้แบบสอบถามในการสำรวจความพึงพอใจของผู้ใช้ระบบ โดยใช้แบบสอบถามแบบเอกสาร และมีเกณฑ์ในการประเมินความพึงพอใจดังนี้

- 5 หมายถึง มีความพึงพอใจในระดับ พอใจมาก
- 4 หมายถึง มีความพึงพอใจในระดับ พอใจ
- 3 หมายถึง มีความพึงพอใจในระดับ ปานกลาง
- 2 หมายถึง มีความพึงพอใจในระดับ น้อย
- 1 หมายถึง มีความพึงพอใจในระดับ น้อยมาก

5.2.3 สรุปผลความพึงพอใจ

การสำรวจความพึงพอใจของระบบจากกลุ่มผู้ใช้งานที่ทดลองใช้งานระบบ โดยใช้แบบสอบถามเพื่อการประเมินหาระดับความพึงพอใจของระบบที่ผู้พัฒนา ได้ทำการพัฒนาขึ้นมา สรุปผลความพึงพอใจเป็นคำถาม โดยผู้ใช้งานที่ทดลองใช้งานระบบและร่วมตอบแบบสอบถาม จำนวนทั้งสิ้น 5 ราย ผลการสำรวจความพึงพอใจเป็นดังนี้

ตารางที่ 5.1 ผลการสำรวจความพึงพอใจแยกเป็นรายประเด็นคำถาม

คำถามการสอบถาม	ระดับความพึงพอใจ (คน)				
	พอใจมาก	พอใจ	ปานกลาง	พอใจน้อย	พอใจน้อยมาก
การเข้าถึงระบบที่ง่าย	5	0	0	0	0
การบันทึกข้อมูลลูกค้าและรถยนต์	0	4	1	0	0
ความสอดคล้องข้อมูลระหว่างลูกค้าและรถยนต์	0	3	2	0	0
การบันทึกข้อมูลการซ่อมบำรุงรถยนต์	4	1	0	0	0
ความสะดวกต่อการใช้งานกับอุปกรณ์ต่างๆ	4	1	0	0	0
ข้อมูลมีความถูกต้องชัดเจน	0	4	1	0	0
การสืบค้นประวัติการซ่อมบำรุงที่มีประโยชน์ต่อการตรวจสอบ	0	3	2	0	0
ความสะดวกต่อการใช้งานระบบที่รองรับทุกอุปกรณ์	5	0	0	0	0

คำถามที่ 1 การเข้าถึงระบบที่ง่าย พบว่าระดับความพึงพอใจของผู้ใช้งาน ส่วนใหญ่อยู่ในระดับมากที่สุด คิดเป็นร้อยละ 100 (จำนวนผู้ตอบ 5 ราย)

คำถามที่ 2 การบันทึกข้อมูลลูกค้าและรถยนต์ พบว่าระดับความพึงพอใจของผู้ใช้งาน ส่วนใหญ่อยู่ในระดับพอใจ คิดเป็นร้อยละ 80 (จำนวนผู้ตอบ 4 ราย) รองลงมามีความพึงพอใจในระดับปานกลาง คิดเป็นร้อยละ 20 (จำนวนผู้ตอบ 1 คน)

คำถามที่ 3 ความสอดคล้องข้อมูลระหว่างลูกค้าและรถยนต์ พบว่าระดับความพึงพอใจของผู้ใช้งาน ส่วนใหญ่อยู่ในระดับพอใจ คิดเป็นร้อยละ 60 (จำนวนผู้ตอบ 3 ราย) รองลงมามีความพึงพอใจในระดับปานกลาง คิดเป็นร้อยละ 40 (จำนวนผู้ตอบ 2 คน)

คำถามที่ 4 การบันทึกข้อมูลการซ่อมบำรุงรถยนต์ พบว่าระดับความพึงพอใจของผู้ใช้งาน ส่วนใหญ่อยู่ในระดับพอใจมาก คิดเป็นร้อยละ 80 (จำนวนผู้ตอบ 4 คน) รองลงมามีความพึงพอใจของผู้ใช้งานในระดับพอใจ คิดเป็นร้อยละ 20 (จำนวนผู้ตอบ 1 คน)

คำถามที่ 5 ความสะดวกต่อการใช้งานกับอุปกรณ์ต่างๆ พบว่าระดับความพึงพอใจของผู้ใช้งาน ส่วนใหญ่อยู่ในระดับพอใจ คิดเป็นร้อยละ 80 (จำนวนผู้ตอบ 4 คน) รองลงมาคือความพึงพอใจของผู้ใช้งานในระดับพอใจ คิดเป็นร้อยละ 20 (จำนวนผู้ตอบ 1 คน)

คำถามที่ 6 ข้อมูลมีความถูกต้องชัดเจน พบว่าระดับความพึงพอใจของผู้ใช้งาน ส่วนใหญ่อยู่ในระดับพอใจ คิดเป็นร้อยละ 80 (จำนวนผู้ตอบ 4 คน) รองลงมาคือความพึงพอใจของผู้ใช้งานในระดับปานกลาง คิดเป็นร้อยละ 20 (จำนวนผู้ตอบ 1 คน)

คำถามที่ 7 การสืบค้นประวัติการซ่อมบำรุงที่มีประโยชน์ต่อการตรวจสอบ พบว่าระดับความพึงพอใจของผู้ใช้งาน ส่วนใหญ่อยู่ในระดับพอใจ คิดเป็นร้อยละ 60 (จำนวนผู้ตอบ 3 คน) รองลงมาคือความพึงพอใจของผู้ใช้งานในระดับปานกลาง คิดเป็นร้อยละ 40 (จำนวนผู้ตอบ 2 คน)

คำถามที่ 8 ความสะดวกต่อการใช้งานระบบที่รองรับทุกอุปกรณ์ พบว่าระดับความพึงพอใจของผู้ใช้งาน ส่วนใหญ่อยู่ในระดับมากที่สุด คิดเป็นร้อยละ 100 (จำนวนผู้ตอบ 5 ราย)

ข้อเสนอแนะจากผู้ใช้งาน

1. อยากให้ระบบสามารถ แสกน QR Code เพื่อแสดงประวัติการซ่อมบำรุงรถยนต์ได้
2. อยากให้ระบบสามารถแสกน Bar Code เพื่อทำการบันทึกจำนวนอะไหล่ที่ส่งรายการซ่อมบำรุงรถยนต์ได้



บรรณานุกรม

- ชนิดา แก้วเพชร. (2540). *ภาษา PHP (Personal Home Page)*. สืบค้นเมื่อวันที่ 11 พฤศจิกายน 2560, จาก <http://www.showded.com/myprofile/mainblog.php?user=san&jucId=2955>
- วรกฤต แสนโกชน์.(2553) *การออกแบบฐานข้อมูลด้วยภาพ E-R Diagrams*. สืบค้นเมื่อวันที่ 12 พฤศจิกายน 2560, จาก <http://www.ict.up.ac.th/worrakits>
- วรกฤต แสนโกชน์.(2553) *ผังแสดงหน้าที่ (Use Case Diagram)*. สืบค้นเมื่อวันที่ 12 พฤศจิกายน 2560, จาก <https://med.mahidol.ac.th/ramapharmacy/sites/default/files/public/>
- พรเทพ รัตนบุรี.(2541) *ระบบฐานข้อมูล Database System*. สืบค้นเมื่อวันที่ 9 พฤศจิกายน 2560, จาก <http://information-system-analysis-and-desi.blogspot.com/>
- พรเทพ รัตนบุรี.(2541) *การวิเคราะห์ระบบ (System Analysis)*. สืบค้นเมื่อวันที่ 9 พฤศจิกายน 2560, จาก <http://information-system-analysis-and-desi.blogspot.com/>



ภาคผนวก

ภาคผนวก ก
การติดตั้งระบบ



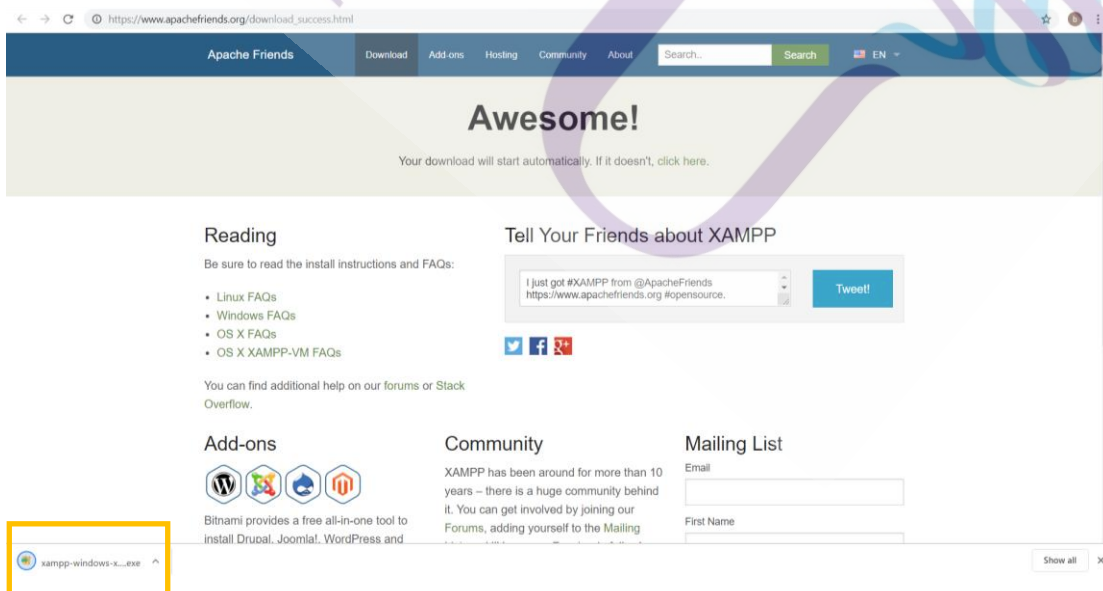
1. การติดตั้ง XAMPP

1. ดาวน์โหลดโปรแกรม XAMPP ที่เว็บไซต์ <https://www.apachefriends.org/index.html>

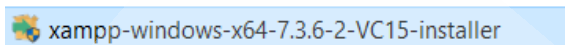
1. คลิก Download XAMPP For Windows



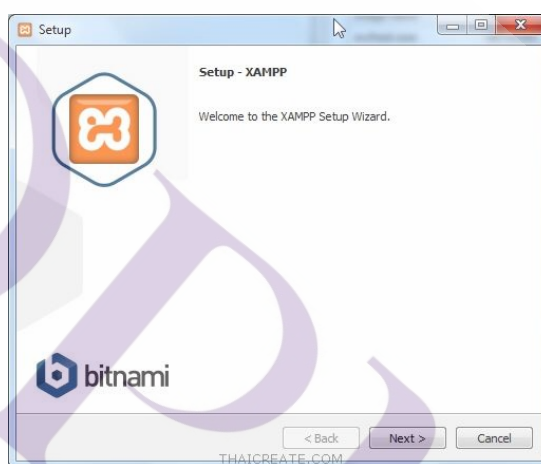
2. เมื่อคลิก Download แล้ว จะแสดงหน้าจอถัดไปขึ้นมา เสร็จแล้วจะปรากฏแถบเมนู ดาวน์โหลดขึ้นมาทางด้านล่าง



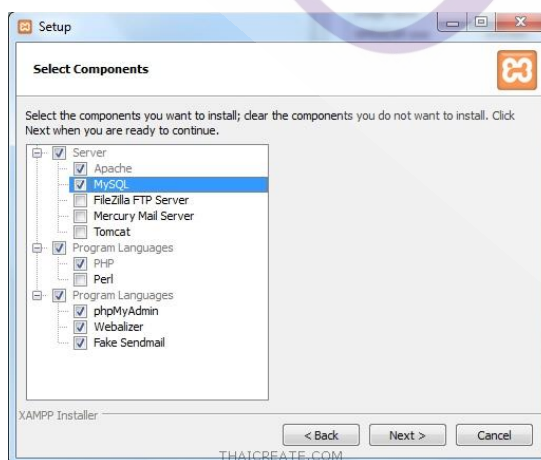
2. หลังจากดาวน์โหลดเสร็จเรียบร้อยแล้ว ให้ดำเนินการติดตั้ง โดย Double Click ที่ Icon xampp-windows



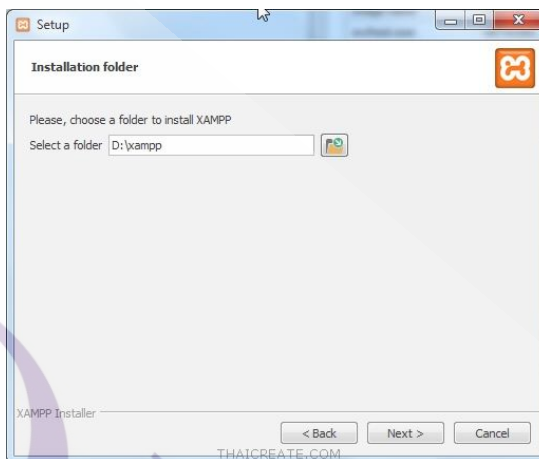
3. จะปรากฏหน้าจอ Install Wizard จากนั้นให้คลิกปุ่ม Next



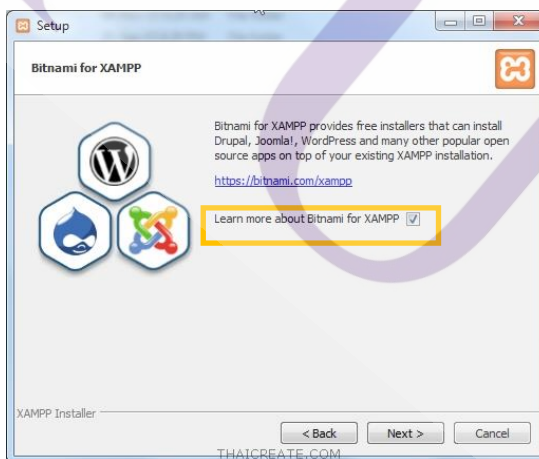
4. กำหนดการตั้งค่าโปรแกรมที่ต้องการใช้งานโดยทำการเลือก Apache , MySQL , PHP , และ PHPMyAdmin



5. สถานะแสดงขั้นตอนการติดตั้งโดยค่ามาตรฐาน (Default) จะถูกติดตั้งไว้ที่ไดร์ C:\Program File\xampp จากนั้นคลิกที่ปุ่ม Next



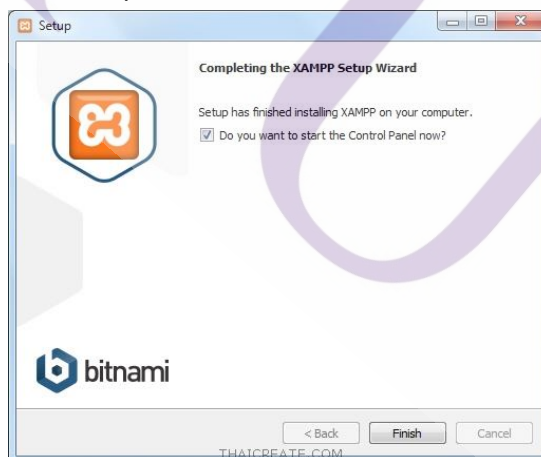
6. สถานะแสดงขั้นตอนการติดตั้งตัวโปรแกรม BitNami หากต้องการเรียนรู้เพิ่มเติม ให้คลิกที่ข้อความ Learn more about Bitnami for XAMMP หากไม่ต้องการให้นำเครื่องหมายถูก ออก



7. หน้าจอแสดงสถานะการติดตั้งโปรแกรม XAMPP ลงบนเครื่องคอมพิวเตอร์จนจบการติดตั้งโปรแกรม



8. เมื่อจบการติดตั้งโปรแกรมแล้ว จะปรากฏข้อความแจ้งให้ทราบ และถามว่าต้องการให้เริ่มต้นการทำงานหรือไม่ ถ้าต้องการให้ทำเครื่องหมายถูกที่ปุ่ม Do you want to start the Control Panel now จากนั้นให้คลิกปุ่ม Finish



ภาคผนวก ข
ตัวอย่างแบบสอบถาม



ตัวอย่างแบบสอบถาม

แบบสอบถามที่ใช้ในการสำรวจความพึงพอใจของผู้ใช้ระบบการบันทึกข้อมูลการซ่อมบำรุงรถยนต์มีการกำหนดลักษณะของคำถามเป็นคำถามปลายปิด ใช้มาตราส่วนประมาณค่า (Rating Scale) และคำถามปลายเปิด (Open-Ended Question) เพื่อให้ผู้ใช้แสดงความคิดเห็นและข้อเสนอแนะ

ตารางที่ 1 ตัวอย่างแบบสอบถามความพึงพอใจของผู้ใช้การบันทึกข้อมูลการซ่อมบำรุงรถยนต์
แบบสอบถามความพึงพอใจ / ไม่พึงพอใจระบบการบันทึกข้อมูลการซ่อมบำรุงรถยนต์

เดือน.....พ.ศ.2562

ข้อชี้แจง กรุณาทำเครื่องหมาย **V** ในข้อที่ตรงกับความเป็นจริงและในช่องที่ตรงกับความคิดเห็นของท่านมากที่สุด

ตอนที่ 1 ข้อมูลทั่วไปของผู้ตอบแบบสอบถาม

1. เพศ

- 1) ชาย 2) หญิง

2. อายุ

- 1) ต่ำกว่า 20 ปี 2) 21 - 40 ปี 4) 41 - 60 ปี 6) 60 ปีขึ้นไป

3. ระดับการศึกษาสูงสุด

- 1) ประถมศึกษา 2) มัธยมศึกษาตอนต้น/ตอนปลาย/หรือเทียบเท่า
 3) ปริญญาตรี 4) สูงกว่าปริญญาตรี

4. สถานภาพของผู้มารับบริการ

- 1) บุคคลผู้ใช้งานระบบ 2) บุคคลภายนอก 3) อื่นๆ.....

ตอนที่ 2 ความพึงพอใจ / ไม่พึงพอใจต่อการให้บริการ

ประเด็น/ด้าน	ระดับความพึงพอใจระดับความพึงพอใจ				
	พอใจมาก	พอใจ	ปานกลาง	น้อย	น้อยมาก
การเข้าถึงระบบที่ง่าย					
การบันทึกข้อมูลลูกค้าและรถยนต์					
ความสอดคล้องข้อมูลระหว่างลูกค้าและรถยนต์					
การบันทึกข้อมูลการซ่อมบำรุงรถยนต์					
ความสะดวกต่อการใช้งานกับอุปกรณ์ต่างๆ					
ข้อมูลมีความถูกต้องชัดเจน					
การสืบค้นประวัติการซ่อมบำรุงที่มีประโยชน์ต่อการตรวจสอบ					
ความสะดวกต่อการใช้งานระบบที่รองรับทุกอุปกรณ์					

ตอนที่ 3 ปัญหา / ข้อเสนอแนะ

- ปัญหา
1.
 2.
- ข้อเสนอแนะ
1.
 2.

ขอขอบคุณในความร่วมมือที่ท่านได้เสียสละเวลาให้ข้อมูลที่เป็นประโยชน์แก่ทางสำนักงานในครั้งนี้

ประวัติผู้เขียน

ชื่อ-นามสกุล	นายศุภฤกษ์ วุฒิการณ์
วันเดือนปีเกิด	21 ธันวาคม 2528
สถานที่เกิด	259/106 ต.ปากคลอง อ.ภาษีเจริญ กรุงเทพมหานคร 10120
ที่อยู่ปัจจุบัน	50/7 หมู่ 8 หมู่บ้านมิตรประชาวิมล ต.เสาธงหิน อ.บางใหญ่ นนทบุรี 11140
ประวัติการศึกษา	พ.ศ. 2551 บริหารธุรกิจบัณฑิต สาขา เทคโนโลยีสารสนเทศ สาขาเทคโนโลยี สารสนเทศธุรกิจ มหาวิทยาลัยธุรกิจบัณฑิต
ตำแหน่งและสถานที่ทำงานปัจจุบัน	Engineer บริษัท จีไอเอส กรุ๊ป จำกัด

