

การเปรียบเทียบจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชันจากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้  
กับรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว



นายภัทร เวชชีรานันต์วัฒน์

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

บทคัดย่อและแฟ้มข้อมูลฉบับเต็มของวิทยานิพนธ์ตั้งแต่ปีการศึกษา 2554 ที่ให้บริการในคลังปัญญาจุฬาฯ (CUIR)  
เป็นแฟ้มข้อมูลของนิสิตเจ้าของวิทยานิพนธ์ ที่ส่งผ่านทางบัณฑิตวิทยาลัย

The abstract and full text of theses from the academic year 2011 in Chulalongkorn University Intellectual Repository (CUIR)  
are the thesis authors' files submitted through the University Graduate School.

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาการพัฒนาซอฟต์แวร์ด้านธุรกิจ ภาควิชาสถิติ

คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2559

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย



A COMPARISON OF THE AMOUNT OF FUNCTION POINTS BY FUNCTION TYPES BETWEEN  
SOFTWARE DESIGN SPECIFICATIONS AND PRODUCT SPECIFICATIONS

– Mr. Pattara Wachiranantawat



A Thesis Submitted in Partial Fulfillment of the Requirements  
for the Degree of Master of Science Program in Business Software Development

Department of Statistics

Faculty of Commerce and Accountancy

Chulalongkorn University

Academic Year 2016

Copyright of Chulalongkorn University



หัวข้อวิทยานิพนธ์	การเปรียบเทียบจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชันจากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้กับรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว
โดย	นายภัทร วัชรานันต์วัฒน์
สาขาวิชา	การพัฒนาซอฟต์แวร์ด้านธุรกิจ
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร.สมจारी ปรียานนท์

---

คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้หัวข้อวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

.....คณบดีคณะพาณิชยศาสตร์และการ  
บัญชี  
(รองศาสตราจารย์ ดร.พสุ เดชะรินทร์)

คณะกรรมการสอบวิทยานิพนธ์  
.....ประธานกรรมการ  
(รองศาสตราจารย์ ดร.ถาวร อานุกาฬไตรรงค์)  
.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก  
(ผู้ช่วยศาสตราจารย์ ดร.สมจारी ปรียานนท์)  
.....กรรมการ  
(ผู้ช่วยศาสตราจารย์ ดร.วัฒนา วิริยสิทธิ์วัฒน์)  
.....กรรมการภายนอกมหาวิทยาลัย  
(ผู้ช่วยศาสตราจารย์ ดร.วรลักษณ์ วงศ์โดยหวัง ศิริเจริญ)

ภัทระ วชิรานันต์วัฒน์ : การเปรียบเทียบจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชันจากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้กับรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว (A COMPARISON OF THE AMOUNT OF FUNCTION POINTS BY FUNCTION TYPES BETWEEN SOFTWARE DESIGN SPECIFICATIONS AND PRODUCT SPECIFICATIONS) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. ดร.สมจारी ปริยานนท์, หน้า.

ในการประมาณขนาดซอฟต์แวร์ เพื่อใช้ในการกำหนดราคาซอฟต์แวร์และเตรียมทรัพยากรในด้านแรงงานและระยะเวลา แต่มักพบว่าการพัฒนาซอฟต์แวร์ส่วนใหญ่มีการประมาณการเกินกว่าที่ประมาณการไว้ งานวิจัยนี้จึงอยากทราบว่า การประมาณค่าความพยายามในขั้นตอนการออกแบบกับเมื่อพัฒนาซอฟต์แวร์เรียบร้อยแล้วมีค่ามากกว่ากัน เพื่อนำไปใช้ในการกำหนดราคาและบริหารจัดการทรัพยากรให้มีความแม่นยำมากขึ้น โดยใช้วิธีการวิเคราะห์ฟังก์ชันพอยต์จากแผนภาพกระแสดำเนินการของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้กับรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว โดยเก็บรวบรวมข้อมูลจากแผนภาพกระแสดำเนินการ แผนภาพแสดงความสัมพันธ์ระหว่างกลุ่มข้อมูล และพจนานุกรมข้อมูล จำนวน 30 ชุดข้อมูล และเพื่อให้การนับจำนวนฟังก์ชันพอยต์มีมาตรฐานเดียวกัน ผู้วิจัยได้พัฒนาเครื่องมือสำหรับการนับค่าฟังก์ชันพอยต์จากแผนภาพกระแสดำเนินการที่วาดบนเครื่องมือไมโครซอฟต์วิซิโอ โดยใช้ภาษาจาวาโดยพัฒนาเครื่องมือตามโครงสร้างไฟล์รูปแบบเอ็กเซลของแผนภาพกระแสดำเนินการที่ใช้ข้อมูลเข้าของเครื่องมือ

จากงานวิจัยพบว่าทุกประเภทฟังก์ชันของรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วมีจำนวนเพิ่มมากขึ้นกว่าเดิม โดยเฉพาะในส่วนของการเพิ่มข้อมูลภายในและข้อมูลนำเข้าภายนอก มีการเปลี่ยนแปลงเพิ่มขึ้นและมีผลโดยตรงต่อกัน โดยมีค่าผลรวมเพิ่มขึ้นเท่ากับ 13.33 เปอร์เซ็นต์ อีกทั้งยังพบว่าค่าฟังก์ชันพอยต์ที่คำนวณจากทุกระดับแผนภาพกับค่าฟังก์ชันที่คำนวณเฉพาะแผนภาพระดับที่หนึ่ง มีค่ามากกว่าเท่ากับ 7.42 เปอร์เซ็นต์ ซึ่งสามารถนำไปใช้ในการประมาณขนาดซอฟต์แวร์ที่แม่นยำมากขึ้น

ภาควิชา สถิติ ลายมือชื่อนิสิต .....

สาขาวิชา การพัฒนาซอฟต์แวร์ด้านธุรกิจ ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

ปีการศึกษา 2559

# # 5681576426 : MAJOR BUSINESS SOFTWARE DEVELOPMENT

KEYWORDS: FUNCTION POINT ESTIMATION / DATA FLOW DIAGRAMS / SOFTWARE DESIGN SPECIFICATIONS / PRODUCT SPECIFICATIONS

PATTARA WACHIRANANTAWAT: A COMPARISON OF THE AMOUNT OF FUNCTION POINTS BY FUNCTION TYPES BETWEEN SOFTWARE DESIGN SPECIFICATIONS AND PRODUCT SPECIFICATIONS. ADVISOR: ASST. PROF.SOMJAREE PREEYANONT, Ph.D., pp.

The estimation of software size is generally conducted with the means to set the price of software and prepare resources in term of labor and time. However, it is often found that the majority of software development are overestimated. Accordingly, this research had the objectives to compare the estimation values in the design process and after the software had been developed with a relatively higher accuracy. The research was conducted by analyzing the function points from data flow diagrams of software design specification and product specification. The data were collected from data flow diagrams, entity relationship diagrams, and data dictionary. Used in this research comprised of 30 data sets, in order to ensure that the calculation of function points was standardized. The researcher had developed a tool for calculating function points from the data flow diagrams, using Microsoft Visio. The tool was developed according to the structure of the .xml file of data flow diagrams.

According to the obtained results, it was evident that all types of function of the product specification had increased in number, particularly for External Input and Internal Logical files –Both have higher number of functions and have direct effects towards one another. In addition, the total value of function point was increased by 13.33%. Likewise, the function points that were calculated from all levels and from level-1 were found to have higher values at 7.42%. Hence, such data can be used to estimation of software size which enhances accuracy.

Department: Statistics Student's Signature .....

Field of Study: Business Software Advisor's Signature .....

Development

Academic Year: 2016

## กิตติกรรมประกาศ

ผู้วิจัยขอขอบพระคุณผู้ช่วยศาสตราจารย์ ดร. สมจारी ปรียานนท์ อาจารย์ที่ปรึกษาวิทยานิพนธ์เป็นอย่างสูงที่กรุณาสละเวลาในการให้คำปรึกษา ชี้แนะแนวทาง ตรวจสอบแก้ไขข้อบกพร่องต่างๆ รวมทั้งให้กำลังใจแก่ผู้วิจัยเป็นอย่างดีตลอดระยะเวลาการทำวิจัย และขอขอบพระคุณผู้ช่วยศาสตราจารย์ ดร. ถาวร อานุกาพไตรรงค์ ประธานกรรมการวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร. วัฒนา วิริยสิทธิวัฒน์ กรรมการวิทยานิพนธ์ และผู้ช่วยศาสตราจารย์ ดร. วรลักษณ์ วงศ์โดยหวัง ศิริเจริญ กรรมการภายนอกมหาวิทยาลัย ที่กรุณาสละเวลาอันมีค่า ให้คำแนะนำชี้แนะสิ่งต่างๆ ตลอดจนอาจารย์ทุกท่านที่ได้มอบความรู้แก่ผู้วิจัย

ขอขอบคุณ อลิสา แสงภู่วงษ์เจริญ และ ภัสชกฤษฎ์ เอียดเกลี้ยงที่ให้คำปรึกษาและช่วยเหลือเกี่ยวกับแนวทางกระบวนการทำวิจัย และขอขอบคุณแหล่งข้อมูลโครงการปริญญาโท วิทยาศาสตร์มหาบัณฑิต สาขาการพัฒนาระบบซอฟต์แวร์ด้านธุรกิจ ที่ให้ความช่วยเหลือด้านการเก็บรวบรวมข้อมูล อีกทั้งขอขอบคุณครอบครัวที่คอยเป็นกำลังใจสนับสนุนในการศึกษาตลอดมา และรวมถึงเพื่อนๆ พี่ๆ น้องๆ ทุกคน ที่คอยเป็นกำลังใจและช่วยเหลือในทุกด้านตลอดมา จนวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วง



## สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ฉ
สารบัญ.....	ช
สารบัญรูป.....	1
สารบัญตาราง.....	1
บทที่ 1 .....	4
บทนำ .....	4
1.1 ที่มาและความสำคัญของปัญหา.....	4
1.2 วัตถุประสงค์ของงานวิจัย .....	8
1.3 ขอบเขตของงานวิจัย .....	8
1.4 ขั้นตอนและวิธีการดำเนินงานวิจัย .....	9
1.5 ประโยชน์ที่คาดว่าจะได้รับ .....	10
1.6 นิยามศัพท์.....	11
บทที่ 2 .....	13
ทฤษฎีและวรรณกรรมที่เกี่ยวข้อง .....	13
2.1 การวิเคราะห์ฟังก์ชันพอยต์ (Function Point Analysis).....	13
2.2 แผนภาพกระแสข้อมูล (Data Flow Diagram หรือ DFD).....	23
2.3 แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล (Entity Relationship Diagrams หรือ ER-Diagram).....	29
2.4 การนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล .....	31

บทที่ 3 .....	35
ระเบียบวิธีวิจัย .....	35
3.1 แนวทางการวิจัย.....	35
3.2 ประชากรและหน่วยตัวอย่าง .....	44
3.3 เครื่องมือที่ใช้ในการศึกษา.....	46
3.4 สมมติฐานของงานวิจัย .....	46
บทที่ 4 .....	50
การออกแบบและพัฒนาเครื่องมือสำหรับการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล .....	50
4.1 การวิเคราะห์การนับการประเมินความซับซ้อนและการแปลงฟังก์ชันพอยต์จาก แผนภาพกระแสข้อมูลเป็นข้อมูลเข้า .....	50
4.2 การออกแบบและการพัฒนาเครื่องมือเครื่องมือสำหรับการนับฟังก์ชันพอยต์จาก แผนภาพกระแสข้อมูล.....	55
4.3 การทดสอบเครื่องมือสำหรับการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล.....	59
4.4 การใช้งานเครื่องมือสำหรับการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล .....	60
บทที่ 5 .....	61
การประมวลผลข้อมูลและการวิเคราะห์ผลลัพธ์การวิจัย.....	61
5.1 หน่วยตัวอย่าง.....	61
5.2 การได้มาซึ่งข้อมูล .....	61
5.3 การวิเคราะห์ผลข้อมูลการวิจัย.....	70
5.4 อภิปรายผลการวิจัย .....	87
บทที่ 6 .....	89
ผลการวิจัย.....	89
6.1 ผลการวิจัย .....	89
6.2 การนำงานวิจัยไปใช้ .....	90

6.3 ข้อจำกัดงานวิจัย .....	90
รายการอ้างอิง .....	91
ประวัติผู้เขียนวิทยานิพนธ์ .....	226



## สารบัญรูป

รูปที่ 1 แสดงขั้นตอนการวิเคราะห์ฟังก์ชันพอยต์.....	13
รูปที่ 2 การระบุขอบเขตของแอปพลิเคชันที่ระบุจำนวนฟังก์ชันพอยต์.....	15
รูปที่ 3 แสดงรูปตารางการให้ค่าน้ำหนักของแต่ละประเภทฟังก์ชันทั้ง 5 ประเภท.....	20
รูปที่ 4 สัญลักษณ์ที่ใช้ในแผนภาพกระแสข้อมูลตามมาตรฐานของ Gane และ Sarson (Gane & Sarson, 1979).....	24
รูปที่ 5 สัญลักษณ์ของกระบวนการของทั้ง 2 มาตรฐาน.....	24
รูปที่ 6 สัญลักษณ์ของกระแสข้อมูลของทั้ง 2 มาตรฐาน.....	25
รูปที่ 7 สัญลักษณ์ของแฟ้มข้อมูลของทั้ง 2 มาตรฐาน.....	25
รูปที่ 8 สัญลักษณ์ของเอนทิตีภายนอกของทั้ง 2 มาตรฐาน.....	25
รูปที่ 9 แผนภาพบริบทของซอฟต์แวร์เพื่อการออกแบบจัดสวนสาธารณะภายใต้งบประมาณที่มี ...	26
รูปที่ 10 แสดงแผนภาพระดับหนึ่งของซอฟต์แวร์เพื่อการออกแบบจัดสวนสาธารณะภายใต้ งบประมาณที่มี.....	27
รูปที่ 11 แผนภาพระดับลูก Level-2 Diagram ของระบบออกแบบสวน.....	28
รูปที่ 12 แผนภาพระดับลูก Level-3 Diagram ของระบบคำนวณปริมาณต้นไม้.....	29
รูปที่ 13 สัญลักษณ์ของ ER Diagram ตามรูปแบบของ Dr. Peter Chen.....	29
รูปที่ 14 สัญลักษณ์ของ ER Diagram ตามรูปแบบแบบจำลองวิศวกรรมสารสนเทศ.....	29
รูปที่ 15 ส่วนที่เป็นข้อมูลนำเข้าจากนอกระบบในแผนภาพกระแสข้อมูล.....	32
รูปที่ 16 ส่วนที่เป็นข้อมูลส่งออกภายนอกจากแผนภาพกระแสข้อมูล.....	33
รูปที่ 17 ส่วนที่เป็นการสอบถามจากภายนอก.....	33
รูปที่ 18 ส่วนที่เป็นแฟ้มข้อมูลภายในและสัญลักษณ์ของการรับส่งข้อมูลเข้าสู่แฟ้มข้อมูล.....	34
รูปที่ 19 แสดงแผนกระบวนการดำเนินงานวิจัย.....	36
รูปที่ 20 แสดง Tag XML ของสัญลักษณ์ต่างๆ ที่ใช้ในการวาดแผนภาพบนเครื่องมือ Visio.....	53
รูปที่ 21 แสดง Tag XML ของรายละเอียดสัญลักษณ์ที่ใช้ในการวาดแผนภาพโดย Visio.....	53

รูปที่ 22	แสดง Tag XML ของ Tag <BegTrigger> และ <EndTrigger> และ Tag <Text>.....	54
รูปที่ 23	แสดงชั้นแผนภาพแสดงขั้นตอนการทำงานของเครื่องมือ.....	56
รูปที่ 24	แสดงTag <Shape>ที่ Attribute เป็น NameU='Data store'.....	57
รูปที่ 25	แสดงTag <Shape>ที่ Attribute ที่เป็น NameU ไม่ปรากฏใน Tag .....	57
รูปที่ 26	แสดงข้อมูลเข้าภายนอก(EIs) ในรูปแบบไฟล์ XML .....	57
รูปที่ 27	แสดงข้อมูลส่งออกภายนอก(EOs) ในรูปแบบไฟล์ XML.....	58
รูปที่ 28	แสดงส่วนต่อประสานกับผู้ใช้งานของเครื่องมือการนับฟังก์ชันพอยต์ .....	60



## สารบัญตาราง

ตารางที่ 1	แสดงตารางความซับซ้อนของแฟ้มข้อมูลภายในและแฟ้มข้อมูลต่อประสานภายนอก....	17
ตารางที่ 2	แสดงตารางความซับซ้อนของข้อมูลนำเข้าภายนอก.....	19
ตารางที่ 3	แสดงตารางความซับซ้อนของข้อมูลส่งออกภายนอกและการสอบถามภายนอก.....	19
ตารางที่ 4	แสดงตารางเกณฑ์ระดับคะแนนของตัวแปรปัจจัยลักษณะของระบบ .....	21
ตารางที่ 5	แสดงตารางคำนวณผลรวมระดับคะแนนของตัวแปรปัจจัยลักษณะของระบบ.....	21
ตารางที่ 6	แสดงรูปแบบของจำนวนสมาชิกในความสัมพันธ์ .....	31
ตารางที่ 7	ตารางค่าการให้น้ำหนักความซับซ้อนของแฟ้มข้อมูลภายใน.....	37
ตารางที่ 8	ตารางค่าการให้น้ำหนักความซับซ้อนของข้อมูลนำเข้าจากภายนอก .....	38
ตารางที่ 9	ตารางค่าการให้น้ำหนักความซับซ้อนของข้อมูลส่งออกภายนอก.....	38
ตารางที่ 10	ตารางค่าการให้น้ำหนักความซับซ้อนของข้อมูลการสอบถามจากภายนอก .....	39
ตารางที่ 11	ตารางค่าการให้น้ำหนักความซับซ้อนของข้อมูลภายนอกระบบ .....	40
ตารางที่ 12	การคำนวณค่าฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า.....	41
ตารางที่ 13	แสดงตารางผลของการคำนวณค่าฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า.....	41
ตารางที่ 14	ตารางบันทึกผลของการคำนวณค่าฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า.....	43
ตารางที่ 15	สัญลักษณ์มาตรฐานและที่ใช้ในวาดแผนภาพกระแสดูข้อมูลบน Visio.....	52
ตารางที่ 16	แสดงการเปรียบเทียบผลการนับจำนวนฟังก์ชันพอยต์ของแผนภาพกระแสดูข้อมูล (1) ของเครื่องมือที่พัฒนา (2)ผู้เชี่ยวชาญทางด้านฟังก์ชันพอยต์ (3)นิสิตที่ศึกษาวิธีการอ่านค่า ฟังก์ชันพอยต์ .....	59
ตารางที่ 17	แสดงชุดข้อมูลของโครงการพัฒนาซอฟต์แวร์ .....	61
ตารางที่ 18	แสดงชื่อชุดข้อมูลจำนวน 30 ชุดข้อมูล ที่นำมาใช้เป็นข้อมูลของงานวิจัย.....	62
ตารางที่ 19	แสดงผลการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก.....	63
ตารางที่ 20	แสดงผลการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก.....	65
ตารางที่ 21	แสดงผลการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลการสอบถามภายนอก .....	66

ตารางที่ 22	แสดงผลการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลเพิ่มข้อมูลภายใน .....	67
ตารางที่ 23	แสดงผลการคำนวณค่าฟังก์ชันพอยต์ของเพิ่มข้อมูลต่อประสานภายนอก .....	68
ตารางที่ 24	แสดงผลการคำนวณค่าฟังก์ชันพอยต์ผลรวมทั้งหมด .....	69
ตารางที่ 25	แสดงผลการทดสอบสมมติฐานทางสถิติของข้อมูลนำเข้าภายนอก .....	71
ตารางที่ 26	แสดงผลการทดสอบสมมติฐานทางสถิติของข้อมูลส่งออกภายนอก .....	72
ตารางที่ 27	แสดงผลการทดสอบสมมติฐานทางสถิติของการสอบถามภายนอกที่ประเมินค่าความ ซับซ้อน .....	73
ตารางที่ 28	แสดงผลการทดสอบสมมติฐานทางสถิติของเพิ่มข้อมูลภายในที่ประเมินค่าความ ซับซ้อน .....	74
ตารางที่ 29	แสดงผลการทดสอบสมมติฐานทางสถิติของเพิ่มข้อมูลต่อประสานภายนอกที่ ประเมินค่าความซับซ้อน .....	75
ตารางที่ 30	แสดงผลการทดสอบสมมติฐานทางสถิติของผลรวมทั้งหมดของฟังก์ชันพอยต์ที่ ประเมินค่าความซับซ้อน .....	76
ตารางที่ 31	แสดงผลการทดสอบสมมติฐานทั้ง 6 สมมติฐาน .....	77
ตารางที่ 32	แสดงสัดส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลนำเข้า ภายนอก .....	78
ตารางที่ 33	แสดงสัดส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลนำเข้า ภายนอก .....	79
ตารางที่ 34	แสดงสัดส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลการ สอบถามภายนอก .....	81
ตารางที่ 35	แสดงสัดส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์ของเพิ่มข้อมูลภายใน ..	82
ตารางที่ 36	แสดงอัตราส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์ของเพิ่มข้อมูลต่อ ประสานภายนอก .....	83
ตารางที่ 37	แสดงสัดส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์ผลรวมทั้งหมด .....	84
ตารางที่ 38	แสดงเปอร์เซ็นต์ค่าสูงสุด ค่าน้อยสุด และค่าเฉลี่ยของค่าสัดส่วนผลต่างของการ คำนวณฟังก์ชันพอยต์ .....	85

ตารางที่ 39 แสดงผลต่างผลรวมฟังก์ชันพอยต์เฉพาะแผนภาพระดับที่หนึ่งกับผลรวมจำนวน	
ฟังก์ชันพอยต์ของทุกระดับแผนภาพ .....	86





## บทที่ 1

### บทนำ

#### 1.1 ที่มาและความสำคัญของปัญหา

การบริหารโครงการพัฒนาซอฟต์แวร์ (Software Project Management) เป็นการวางแผนการดำเนินงาน การจัดการและบริหารทรัพยากร โดยคำนึงถึงปัจจัยต่างๆ ที่เกี่ยวข้องกับโครงการ เช่น ขนาดซอฟต์แวร์ (Software Size) ค่าความพยายาม (Effort) ต้นทุนในการพัฒนาซอฟต์แวร์ (Cost) และรวมไปถึงระยะเวลา (Time) ในการดำเนินงาน เพื่อต้องการให้โครงการพัฒนาซอฟต์แวร์สำเร็จลุล่วงตามเป้าหมายของแผนการดำเนินงานที่วางไว้ และเกิดประสิทธิภาพสูงสุดในการบริหารโครงการพัฒนาซอฟต์แวร์ โดยปัญหาที่พบส่วนในการวางแผนดำเนินการคือการประมาณขนาดซอฟต์แวร์ที่ต่ำกว่าความเป็นจริง จึงส่งผลให้เกิดปัญหาระหว่างการพัฒนาซอฟต์แวร์เรื่อง ระยะเวลาที่ใช้ในการพัฒนาซอฟต์แวร์ หรือปัญหาเรื่องแรงงานที่ใช้ในการพัฒนาซอฟต์แวร์ ดังนั้นจึงกล่าวได้ว่าการคาดคะเนหรือการประมาณค่าปัจจัยต่างๆ ได้อย่างมีความแม่นยำตั้งแต่ขั้นตอนการวิเคราะห์และออกแบบซอฟต์แวร์จะทำให้การวางแผนการบริหารโครงการมีความถูกต้องและน่าเชื่อถือมากยิ่งขึ้น

จากปัญหาการประมาณขนาดซอฟต์แวร์ที่ต่ำกว่าความเป็นจริง ปัญหาหนึ่งก็คือการใช้แรงงานในการพัฒนาซอฟต์แวร์มากกว่าที่วางแผนไว้ (Humphrey & S., 2005) กล่าวไว้ว่ามากกว่าครึ่งหนึ่งของโครงการพัฒนาซอฟต์แวร์มีความล่าช้ากว่าแผนที่วางไว้ และปัญหาการใช้งบประมาณเกินกว่าแผนที่วางไว้ ซึ่งในปัจจุบันวิธีการประมาณขนาดซอฟต์แวร์ที่มีการใช้อย่างแพร่หลายมีด้วยกัน 2 วิธี คือ (1)การประมาณขนาดซอฟต์แวร์ด้วยจำนวนบรรทัดของการเขียนโปรแกรมซอฟต์แวร์ (Source Line of Code หรือ SLOC) และ(2)การวิเคราะห์ฟังก์ชันพอยต์ (Function Points หรือ FP) ซึ่งการวัดขนาดซอฟต์แวร์ด้วยวิธีการนับจำนวนบรรทัดของซอฟต์แวร์นั้นมีข้อจำกัดคือ ไม่สามารถวัดก่อนการสร้างซอฟต์แวร์ได้จำเป็นต้องใช้ผู้เชี่ยวชาญ (Expert Judgment) ในการประมาณขนาดซอฟต์แวร์แทน รวมไปถึงภาษาที่ใช้ในการพัฒนาซอฟต์แวร์และเทคโนโลยีที่ใช้ในการพัฒนาซอฟต์แวร์ และความเชี่ยวชาญของผู้พัฒนาซอฟต์แวร์ที่ส่งผลให้ได้จำนวนบรรทัดของซอฟต์แวร์ที่ต่างกัน (Luigi Buglione, Cigdem Gencel, & Efe, 2006) ซึ่งการวิเคราะห์ฟังก์ชันพอยต์มีข้อได้เปรียบในจุดต่างๆ เหล่านี้ เพราะการวิเคราะห์ฟังก์ชันพอยต์นั้นไม่ยึดติดกับเทคโนโลยีและภาษาที่ใช้ในการพัฒนาซอฟต์แวร์ และสามารถประมาณขนาดของซอฟต์แวร์ได้อย่างสมเหตุสมผลตั้งแต่ก่อนเริ่มสร้างซอฟต์แวร์ กล่าวคือสามารถประมาณขนาดได้ตั้งแต่กระบวนการวิเคราะห์และออกแบบซอฟต์แวร์ อีกทั้งเทคโนโลยี

และภาษาที่ใช้ในการพัฒนาซอฟต์แวร์นั้นไม่ส่งผลต่อการการประมาณขนาดซอฟต์แวร์ด้วยวิธีวิเคราะห์ฟังก์ชันพอยต์ (Longstreet, 2004)

การวิเคราะห์ฟังก์ชันพอยต์ (Function Points Analysis หรือ FP) พัฒนาขึ้นโดย Allan J. Albrecht (Albrecht, 1979) ในช่วงกลางทศวรรษปี 1970 โดยต่อมาในปี 1979 แนวคิดนี้ถูกนำเสนอเผยแพร่ โดยหลังจากการนำเสนอแนวคิดดังกล่าว ก็ได้รับความสนใจและนำไปพัฒนาต่อจากแนวคิดนี้ และต่อมาในปี 1986 กลุ่มของผู้ใช้การวิเคราะห์ฟังก์ชันพอยต์ระดับสากล (International Function point User Group หรือ IFPUG) ได้สร้างมาตรฐานของการใช้การวิเคราะห์ฟังก์ชันพอยต์ให้เป็นสากลและมีมาตรฐานเดียวกัน ซึ่งแนวคิดการวิเคราะห์ฟังก์ชันพอยต์ เป็นวิธีการประมาณขนาดซอฟต์แวร์ โดยวิธีการนับจำนวนฟังก์ชันพอยต์ (Function Point Counting) โดยพิจารณาการทำงานของฟังก์ชันที่ประกอบอยู่ในซอฟต์แวร์ การวิเคราะห์ฟังก์ชันพอยต์ประกอบด้วย 5 ประเภทฟังก์ชัน (Function Type) การทำงาน โดยแบ่งฟังก์ชันของซอฟต์แวร์ตามลักษณะการทำงานของฟังก์ชันที่ใช้ในการวัดฟังก์ชันพอยต์ออกเป็น 2 ลักษณะใหญ่ๆ คือ ดาต้าฟังก์ชัน (Data Function) เป็นฟังก์ชันเชิงข้อมูลจะประกอบไปด้วย (1) เพิ่มข้อมูลภายในระบบ (Internal Logical files หรือ ILF's) (2) เพิ่มข้อมูลต่อประสานภายนอก (External Interface file หรือ EIF's) และทรานแซคชันฟังก์ชัน (Transaction Functions) เป็นฟังก์ชันเชิงรายการเปลี่ยนแปลงจะประกอบไปด้วย (3) ข้อมูลนำเข้าภายนอก (External Input files หรือ EIs) (4) ข้อมูลส่งออกภายนอก (External Output files หรือ EOs) และ (5) การสอบถามภายนอก (External Inquiry files หรือ EQs) โดยในแต่ละประเภทฟังก์ชันจะมีการให้ค่าน้ำหนักตามความซับซ้อนที่แตกต่างกันในแต่ละประเภทฟังก์ชันของการวิเคราะห์ฟังก์ชันพอยต์ ซึ่งมี 3 ระดับคือ ต่ำ (Low) กลาง (Average) และระดับสูง (High) เพื่อนำมาใช้ในหาค่าฟังก์ชันพอยต์ และนำไปใช้ในการประมาณขนาดของซอฟต์แวร์

Raimo Rask (Rask, 1991) ได้ศึกษาหลักการนับจำนวนฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล (Data Flow Diagram) และแผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล (Entity Relationship Diagrams) โดยใช้การวิเคราะห์จากโครงสร้างของแผนภาพมาใช้ในการนับจำนวนฟังก์ชันพอยต์ เพื่อคำนวณหาขนาดของซอฟต์แวร์ ต่อมา Lamma และคณะ (E. Lamma, P. Mello, & Fabrizio, 2003) ได้ศึกษาหลักการนับฟังก์ชันพอยต์เพื่อใช้ในการคำนวณขนาดของซอฟต์แวร์ใช้แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูลมาใช้ในการนับจำนวนฟังก์ชันพอยต์ หากการคำนวณขนาดของซอฟต์แวร์ถูกนำมาใช้ในกระบวนการขั้นตอนการวิเคราะห์และออกแบบซอฟต์แวร์ จะสามารถช่วยในการประมาณเวลา การประเมินราคาของซอฟต์แวร์ และการประมาณการจัดสรรจำนวนแรงงานที่ต้องใช้ในการผลิตซอฟต์แวร์หนึ่งโครงการ โดยการประมาณค่าต่างๆ ที่ใช้ในการวางแผนการดำเนินงานพัฒนาซอฟต์แวร์นั้นจะต้องใช้ข้อมูลขอบเขตของซอฟต์แวร์หรือ

รายละเอียดซอฟต์แวร์ (Software Specification) มาใช้ในการคำนวณขนาดซอฟต์แวร์ (Pressman, 2001) รายละเอียดซอฟต์แวร์แสดงถึงขอบเขตและรายละเอียดขั้นตอนวิธีการทำงานของซอฟต์แวร์ เช่น การประมวลผลผลลัพธ์ การนำข้อมูลไปใช้ เป็นต้น เพื่อให้ซอฟต์แวร์ใช้งานได้ตามความต้องการของผู้ใช้งานซอฟต์แวร์ รายละเอียดซอฟต์แวร์ที่ออกแบบไว้นั้นจะถูกนำมาใช้โดยผู้พัฒนาซอฟต์แวร์เพื่อใช้เป็นแบบในการสร้างซอฟต์แวร์จริงขึ้นมา (Sommerville, 2001) โดยที่รายละเอียดซอฟต์แวร์จะมีการนำแบบจำลองต่างๆ เพื่อเข้ามาช่วยในการอธิบายขอบเขตและขั้นตอนกระบวนการทำงานของซอฟต์แวร์ ประเภทของแบบจำลองที่นำมาใช้ในการอธิบายขอบเขตและกระบวนการทำงานของซอฟต์แวร์นั้นมีด้วยกัน 3 ประเภท คือ (1)แบบจำลองการประมวลผลข้อมูล (Process Model) เพื่อช่วยให้สามารถเข้าใจขั้นตอนกระบวนการทำงานของซอฟต์แวร์ที่พัฒนา เช่น แผนภาพกระแสข้อมูล (2)แบบจำลองข้อมูล (Data Model) เพื่อช่วยอธิบายลักษณะของกลุ่มข้อมูล และความสัมพันธ์ระหว่างกลุ่มข้อมูล (Entity Relation Diagram) (3)แบบจำลองเชิงวัตถุ (Object Oriented Model) เพื่ออธิบายลักษณะของวัตถุ และความสัมพันธ์ระหว่างวัตถุในคลาสนั้นๆ เช่น แผนภาพแสดงคลาส (Class Diagram) ดังนั้นรายละเอียดซอฟต์แวร์จึงเป็นเสมือนต้นแบบของซอฟต์แวร์ที่จะใช้ในการผลิตซอฟต์แวร์

แผนภาพกระแสข้อมูล(Data Flow Diagram หรือ DFD) เป็นแบบจำลองประเภทแบบจำลองการประมวลผลข้อมูล (Process Model) โดยแสดงแบบจำลองขั้นตอนการทำงานของระบบ เพื่ออธิบายขั้นตอนการทำงานของระบบ โดยจะแสดงทิศทางการไหลของข้อมูลและอธิบายความสัมพันธ์ในการดำเนินงานของระบบ ซึ่งจะทำให้ทราบได้ว่าข้อมูลมาจากที่ไหน ไปที่ไหน เกิดกิจกรรมใดกับข้อมูลบ้าง หรือมีการส่งข้อมูลไปจัดเก็บไว้ที่ไหน โดยประกอบด้วยชุดสัญลักษณ์ดังนี้ ขั้นตอนการทำงาน (Process) เส้นทางการไหลของข้อมูล (Data Flow) ตัวแทนข้อมูล (External) และแฟ้มข้อมูล (Data Store) ถูกพัฒนาโดย Gane และ Sarson (Gane & Sarson, 1979) และพัฒนาโดย DeMarco และ Yourdon (DeMarco, 1979) (Yourdon & Constantine, 1979) โดยชุดสัญลักษณ์ของทั้งสองแบบนี้นิยมนำมาใช้ในปัจจุบันสำหรับการออกแบบแผนภาพกระแสข้อมูลในการสร้างแบบจำลองการพัฒนาซอฟต์แวร์

แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล(Entity Relationship Diagrams หรือ ER-Diagram) เป็นแบบจำลองประเภทแบบจำลองข้อมูล (Data Model) เพื่อช่วยในการอธิบายลักษณะของกลุ่มข้อมูล และความสัมพันธ์ระหว่างกลุ่มข้อมูล ซึ่งแผนแสดงความสัมพันธ์กลุ่มข้อมูลจะถูกใช้ในการอธิบายลักษณะความสัมพันธ์ของกลุ่มข้อมูลที่ออกแบบ โดยมีการใช้สัญลักษณ์ในการแทนความหมายของส่วนประกอบต่างๆ ของแผนภาพ (สมศักดิ์ โชคชัยชุตติกุล, 2553) โดยประกอบด้วยชุดสัญลักษณ์ดังนี้ เอนทิตี(Entity) เป็นตัวแทนชุดข้อมูล แอททริบิวต์

(Attribute) เป็นตัวแทนของคุณสมบัติเฉพาะตัวต่างๆ ของเอนทิตี และความสัมพันธ์(Relation) เป็นตัวเชื่อมระหว่างเอนทิตี ซึ่งแผนภาพแสดงความสัมพันธ์กลุ่มข้อมูลถูกพัฒนาขึ้นในปี ค.ศ. 1976 โดย Peter Chen

จากงานวิจัยของ Standish Group International พบว่า 52.7 เปอร์เซ็นต์ของโครงการพัฒนาซอฟต์แวร์ทั้งหมดใช้งบประมาณมากกว่างบประมาณเดิมที่วางไว้ (Linberg, 1999) รวมทั้งงานวิจัยของ Verner และคณะ (J. M. Verner, W. M. Evanco, & Cerpa, 2007) ได้ศึกษาเกี่ยวกับการประมาณเวลาในโครงการพัฒนาซอฟต์แวร์ โดยมีการเก็บข้อมูลจากโครงการพัฒนาซอฟต์แวร์ 153 โครงการ พบว่ามีโครงการพัฒนาซอฟต์แวร์ 65 เปอร์เซ็นต์ ที่มีการประมาณเวลาต่ำกว่าความเป็นจริง (Underestimated) และโครงการพัฒนาซอฟต์แวร์อีก 68 เปอร์เซ็นต์ มีการเปลี่ยนแปลงตารางการดำเนินงานระหว่างขั้นตอนการผลิตซอฟต์แวร์ และมีโครงการพัฒนาซอฟต์แวร์อีก 33 เปอร์เซ็นต์ ที่จำเป็นต้องมีการเพิ่มแรงงานในช่วงท้ายของโครงการพัฒนาซอฟต์แวร์ เพื่อให้งานเสร็จตามแผนการดำเนินงานเดิมที่วางไว้ จะเห็นว่าการพัฒนาซอฟต์แวร์ส่วนใหญ่ประสบปัญหาในประมาณแรงงานและระยะเวลาที่ใช้ในการพัฒนาซอฟต์แวร์มากกว่าที่คาดการณ์หรือมากกว่าที่ประมาณการไว้ต้องมีการเพิ่มแรงงานเข้ามาเพื่อช่วยในการพัฒนาซอฟต์แวร์จากที่คาดการณ์ไว้ และการเปลี่ยนแปลงแผนการดำเนินงานระหว่างกระบวนการพัฒนาซอฟต์แวร์ ทำให้ส่งผลกระทบต่อโครงการพัฒนาซอฟต์แวร์ในด้านงบประมาณทำให้มีค่าใช้จ่ายเพิ่มมากขึ้น เช่น เงินค่าจ้างแรงงานที่ต้องจ่ายเพิ่มมากขึ้นเพื่อเพิ่มจำนวนแรงงานเพื่อให้การพัฒนาซอฟต์แวร์เสร็จทันตามกำหนดส่งมอบ หรือบางโครงการพัฒนาซอฟต์แวร์หากไม่สามารถส่งมอบซอฟต์แวร์ได้ทันตามกำหนดการส่งซอฟต์แวร์จะต้องจ่ายค่าปรับตามข้อตกลงก่อนที่จะพัฒนาซอฟต์แวร์

จะพบว่าในโครงการพัฒนาซอฟต์แวร์ส่วนมากประสบปัญหาใช้เวลาและแรงงานในการพัฒนาซอฟต์แวร์มากกว่าที่คาดการณ์ไว้ โดยปัญหาที่เกิดขึ้นล้วนเกิดมาจากการประมาณค่าที่ผิดพลาด หรือการประมาณค่าที่ต่ำกว่าความเป็นจริง จากการศึกษาเบื้องต้นยังไม่พบงานวิจัยใดที่ศึกษาว่าขนาดของซอฟต์แวร์ที่ปรากฏในรายละเอียดซอฟต์แวร์ที่ออกแบบไว้นั้นน้อยกว่าขนาดของซอฟต์แวร์ที่ปรากฏในรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ซึ่งหากขนาดของซอฟต์แวร์ทั้งสองไม่เท่ากันจริง ย่อมจะส่งผลให้การประมาณเรื่องระยะเวลา แรงงาน และงบประมาณ ที่ต้องนำค่าของขนาดของซอฟต์แวร์เข้ามาใช้ในการคำนวณด้วยนั้นไม่เท่ากันด้วย ดังนั้นผู้วิจัยเลือกการประมาณขนาดซอฟต์แวร์ ด้วยวิธีการวิเคราะห์ฟังก์ชันพอยต์เพื่อใช้ในการหาค่าขนาดของซอฟต์แวร์ โดยใช้การเปรียบเทียบจำนวนฟังก์ชันพอยต์แยกตามแต่ละประเภทโดยใช้ค่าฟังก์ชันพอยต์ที่ยังไม่ได้ปรับค่า(Unadjusted Function Point) ในการเปรียบเทียบ อีกทั้งวิธีการวิเคราะห์ฟังก์ชันพอยต์ไม่มีข้อจำกัดเรื่องภาษาและเทคโนโลยีที่ใช้ในการพัฒนาซอฟต์แวร์ และยัง

สามารถใช้วัดขนาดของซอฟต์แวร์ได้ตั้งแต่การวิเคราะห์และออกแบบระบบ โดยใช้แผนภาพ กระแสข้อมูล แผนภาพแสดงความสัมพันธ์ของข้อมูล และพจนานุกรมข้อมูล ซึ่งแผนภาพเหล่านี้ จะถูกสร้างขึ้นมาในกระบวนการขั้นตอนการออกแบบและวิเคราะห์ข้อมูลของการพัฒนา ซอฟต์แวร์ และหลังจากการสร้างซอฟต์แวร์เสร็จแล้ว ซึ่งแผนภาพเหล่านี้จะปรากฏอยู่ใน รายละเอียดซอฟต์แวร์ที่ออกแบบไว้ (Software Design Specifications) และรายละเอียด ซอฟต์แวร์ที่สร้างเสร็จแล้ว (Product Specifications) โดยใช้การวิเคราะห์ฟังก์ชันพอยต์ตาม หลักการของ Raimo Rask (Rask, 1991) และ Lamma และคณะ (E. Lamma et al., 2003) และประเมินค่าความซับซ้อนของแต่ละประเภทฟังก์ชันตามหลักของ IFPUG (International Function Points User Group, 1999) โดยใช้ค่าฟังก์ชันพอยต์ที่ยังไม่ปรับค่า (Unadjusted Function Point) ในการเปรียบเทียบความแตกต่างของจำนวนฟังก์ชันพอยต์แยกตามประเภท ฟังก์ชันของซอฟต์แวร์ก่อนสร้าง และหลังสร้างซอฟต์แวร์เสร็จแล้ว

ประโยชน์จากงานวิจัยนี้จะทำให้เราทราบว่าขนาดของซอฟต์แวร์ที่ปรากฏในรายละเอียด ซอฟต์แวร์ที่ออกแบบไว้กับขนาดของซอฟต์แวร์ที่ปรากฏในรายละเอียดของซอฟต์แวร์ที่สร้าง เสร็จแล้วมีความแตกต่างกันเท่าใด และประเภทฟังก์ชันพอยต์ของซอฟต์แวร์ประเภทใดที่มีผลทำให้ มีความแตกต่างกัน ซึ่งจะช่วยให้สามารถประมาณเผื่อไว้ในการวางแผนการพัฒนาซอฟต์แวร์ จะทำให้ได้ค่าการคาดคะเนของขนาดซอฟต์แวร์ในการออกแบบได้ใกล้เคียงกับขนาดของ ซอฟต์แวร์ที่สร้างเสร็จจริงมากขึ้น เพื่อเป็นการช่วยลดผิดพลาดของการประมาณขนาดซอฟต์แวร์ ที่มีผลต่อการประมาณระยะเวลา แรงงาน และต้นทุนในการในผลิตซอฟต์แวร์ของโครงการพัฒนา ซอฟต์แวร์ อีกทั้งเพื่อลดความเสี่ยงด้านต่างๆ ในโครงการพัฒนาซอฟต์แวร์ได้มากยิ่งขึ้น

## 1.2 วัตถุประสงค์ของงานวิจัย

1. เพื่อศึกษาว่าขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้นั้น น้อยกว่าขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดซอฟต์แวร์ที่สร้างเสร็จหรือไม่
2. เพื่อศึกษาว่าประเภทฟังก์ชันพอยต์ประเภทใดของรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้วที่ส่งผลให้รายละเอียดของซอฟต์แวร์ทั้งสองมี ความแตกต่างกัน

## 1.3 ขอบเขตของงานวิจัย

1. การหาจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชันที่ปรากฏอยู่ในรายละเอียดของ ซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว โดยใช้แผนภาพ กระแสข้อมูล (Data Flow Diagrams) แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล (ER- Diagrams) และพจนานุกรมข้อมูล (Data Dictionary) มาใช้ในการนับจำนวนฟังก์ชันพอยต์ ของซอฟต์แวร์

2. เปรียบเทียบประเภทฟังก์ชันพอยต์ (EIs, EOs, EQs, ILF's, EIF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ และรายละเอียดซอฟต์แวร์ที่พัฒนาเสร็จแล้ว

#### 1.4 ขั้นตอนและวิธีการดำเนินงานวิจัย

- 1.4.1 ขั้นตอนที่ 1 เก็บรวบรวมข้อมูลของโครงการพัฒนาซอฟต์แวร์ที่ใช้แผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูลและพจนานุกรมข้อมูล

เก็บข้อมูลโครงการพัฒนาซอฟต์แวร์ที่พัฒนาขึ้นใหม่ ซึ่งจะเลือกเฉพาะโครงการที่ใช้

(1)แผนภาพกระแสข้อมูล (2)แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล และ(3)พจนานุกรมข้อมูล(Data Dictionary)ในการพัฒนาซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ และในรายละเอียดของซอฟต์แวร์ที่พัฒนาเสร็จแล้ว ข้อมูลที่ได้จะถูกนำมาใช้ในการหาขนาดของซอฟต์แวร์ด้วยวิธีการวิเคราะห์ฟังก์ชันพอยต์เพื่อหาจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชันโดยจะได้ค่าฟังก์ชันพอยต์ที่ยังไม่ปรับค่า (Unadjusted Function Point) ที่จะนำมาใช้ในการวิเคราะห์เปรียบเทียบความแตกต่างของจำนวนฟังก์ชันพอยต์แยกตามแต่ละประเภทที่ปรากฏอยู่ในรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ และในรายละเอียดของซอฟต์แวร์ที่พัฒนาเสร็จแล้ว

- 1.4.2 ขั้นตอนที่ 2 คำนวณจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชัน จากข้อมูลแผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล และพจนานุกรมข้อมูล ข้อมูลที่ได้จากการคำนวณจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชัน จากแผนภาพ

กระแสข้อมูล แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล และพจนานุกรมข้อมูล โดยจะได้ค่าฟังก์ชันพอยต์ที่ยังไม่ปรับค่า (Unadjusted Function Point) มาใช้เปรียบเทียบในขั้นตอนถัดไป เพื่อใช้ในการเปรียบเทียบขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ และขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว โดยงานวิจัยชิ้นนี้จะวัดขนาดของซอฟต์แวร์ด้วยหลักการวิเคราะห์แบบฟังก์ชันพอยต์ โดยผู้วิจัยจะสร้างเครื่องมือสำหรับการศึกษา เพื่อช่วยในการคำนวณข้อมูลจากแผนภาพกระแสข้อมูล ซึ่งจะได้จำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชันพอยต์ และแผนภาพแสดงความสัมพันธ์กลุ่มข้อมูลและพจนานุกรมข้อมูล เพื่อใช้สำหรับการประเมินค่าความซับซ้อนของแต่ละประเภทฟังก์ชันพอยต์ตามหลักของIFPUG (International Function Points User Group, 1999) เพื่อนำมาคำนวณหาค่าฟังก์ชันพอยต์ที่ยังไม่ปรับค่าใช้ในการเปรียบเทียบขนาดของซอฟต์แวร์

สำหรับค่าฟังก์ชันพอยต์ที่ยังไม่ปรับค่า(Unadjusted Function Point) นำไปใช้ในการเปรียบเทียบขนาดซอฟต์แวร์ โดยได้จากการนับจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชันจากแผนภาพกระแสข้อมูล โดยนับจำนวนฟังก์ชันพอยต์จาก 5 ประเภทฟังก์ชันซึ่งได้แก่ (1)จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (Internal Logical Files : ILF's) (2)จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (External Interface Files : EIF's) (3)จำนวนฟังก์ชันพอยต์ของ

ข้อมูลนำเข้าภายนอก (External Input Files : EIs) (4)จำนวนฟังก์ชันพอยต์ของข้อมูลส่งออก ภายนอก (External Outputs : EOs) (5)จำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (External Inquiries : EQs) และนับค่าความซับซ้อนของแต่ละประเภทฟังก์ชันพอยต์ตามหลักของIFPUG (1999) จากแผนภาพแสดงความสัมพันธ์กลุ่มข้อมูลและพจนานุกรมข้อมูล ซึ่งได้แก่ (1) Data Element Type(DET) หรือเขตข้อมูลเป็นฟิลด์(Filed) ข้อมูลที่สนใจในแต่ละแฟ้มข้อมูล(Data Store) (2) Record Element Type (RET) หรือตารางข้อมูล, ระเบียบหรือเรคคอร์ด หรือกลุ่มของย่อยของเขตข้อมูล (3) File Type Reference (FTR) หรือแฟ้มข้อมูลที่ถูกอ้างอิงสำหรับรายการเปลี่ยนแปลง โดยแฟ้มข้อมูลอ้างอิง (FTRs) จะต้องเป็นแฟ้มข้อมูลภายใน (ILF's) หรือแฟ้มข้อมูลต่อประสาน ภายนอก (EIF's)

โดยจะได้ข้อมูลค่าฟังก์ชันพอยต์ที่ยังไม่ปรับค่าของ 2 ชุดข้อมูลต่อหนึ่งโครงการพัฒนาซอฟต์แวร์ กล่าวคือ (1) ค่าฟังก์ชันพอยต์ที่ยังไม่ปรับค่าของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ และ(2)ค่าฟังก์ชันพอยต์ที่ยังไม่ปรับค่าของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว

#### 1.4.3 ขั้นตอนที่ 3 เปรียบเทียบจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชันของการประมาณซอฟต์แวร์

เปรียบเทียบข้อมูลค่าฟังก์ชันพอยต์ที่ยังไม่ปรับค่าของ 2 ชุดข้อมูลต่อหนึ่งโครงการพัฒนาซอฟต์แวร์จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว ซึ่งใช้หลักการวิเคราะห์แบบฟังก์ชันพอยต์จากขั้นตอนที่ 2 มาเปรียบเทียบกัน ผลที่ได้จากขั้นตอนนี้จะ ได้ค่าของความแตกต่างระหว่างค่าฟังก์ชันพอยต์ที่ยังไม่ถูกปรับค่าที่ได้จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และค่าฟังก์ชันพอยต์ที่ยังไม่ถูกปรับค่าของซอฟต์แวร์ที่ได้จากรายละเอียดของซอฟต์แวร์ที่พัฒนาสร้างเสร็จแล้ว ตามแต่ละประเภทฟังก์ชัน

#### 1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ช่วยให้สามารถประมาณขนาดซอฟต์แวร์ในขั้นตอนการวิเคราะห์และออกแบบซอฟต์แวร์ ได้ใกล้เคียงขนาดของซอฟต์แวร์ที่สร้างเสร็จแล้วมากขึ้นไม่ต่ำกว่าที่วางแผนการดำเนินงาน และเพื่อเป็นการลดความเสี่ยงด้านต่างๆ ของการบริหารโครงการพัฒนาซอฟต์แวร์
2. ทำให้ทราบว่าประเภทฟังก์ชันพอยต์ประเภทใดที่มีผลให้ขนาดของซอฟต์แวร์เกิดความแตกต่างกัน จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้กับรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว ซึ่งจะช่วยให้ผู้ออกแบบซอฟต์แวร์ทราบถึงข้อควรระวังเกี่ยวกับประเภทฟังก์ชันในการวิเคราะห์และออกแบบซอฟต์แวร์มากขึ้น

## 1.6 นิยามศัพท์

### ฟังก์ชันพอยต์ (Function Point)

ค่าประมาณที่ได้มาจากผลรวมของการนับจำนวนฟังก์ชันการทำงานของซอฟต์แวร์ที่จำแนกตามประเภทฟังก์ชัน เพื่อประมาณค่าขนาดของซอฟต์แวร์ตามวิธีการวิเคราะห์แบบฟังก์ชันพอยต์

### ฟังก์ชันพอยต์ยังไม่ปรับค่า (Unadjusted Function Point)

เป็นค่าประมาณฟังก์ชันพอยต์ที่คำนวณได้ แต่ยังไม่ได้นำสิ่งต่าง ๆ มาปรับค่าหรือปัจจัยสภาพแวดล้อมอื่นๆ ที่อาจจะส่งผลกระทบต่อประมาณขนาดซอฟต์แวร์

### ฟังก์ชันเชิงข้อมูล (Data Function)

การจัดกลุ่มฟังก์ชันประเภทข้อมูลประกอบด้วย(1)แฟ้มข้อมูลภายใน (Internal Logical Files : ILF's) และ(2) แฟ้มข้อมูลต่อประสานภายนอก (External Interface Files : EIF's)

### ฟังก์ชันเชิงรายการเปลี่ยนแปลง (Transaction Function)

การจัดกลุ่มฟังก์ชันประเภทรายการเปลี่ยนแปลงประกอบด้วย(1)ข้อมูลนำเข้าภายนอก (External Input Files : EIs) (2) ข้อมูลส่งออกภายนอก (External Outputs : EOs) และ (3) การสอบถามภายนอก (External Inquiries : EQs)

### ประเภทฟังก์ชัน (Function Type)

ประเภทรายการของการนับฟังก์ชันของการวิเคราะห์แบบฟังก์ชันพอยต์ สามารถแบ่งออกได้เป็น 5 ประเภท EIs, EOs, EQs, ILFs และ EIFs

### แฟ้มข้อมูลภายใน (Internal Logical Files : ILF's)

กลุ่มข้อมูลที่อยู่ในระบบตลอดช่วงอายุของระบบ และสามารถจัดการปรับปรุงเปลี่ยนแปลงข้อมูลภายในระบบ

### แฟ้มข้อมูลต่อประสานภายนอก (External Interface Files : EIF's)

กลุ่มข้อมูลที่อยู่ภายนอกระบบ หรือกลุ่มข้อมูลที่อยู่ในระบบนอกขอบเขตที่กำหนดการนับฟังก์ชัน



### ข้อมูลนำเข้าภายนอก (External Input Files : EIs)

ข้อมูลที่ได้รับเข้ามาภายในระบบ โดยรับข้อมูลจากภายนอกเพื่อปรับปรุงข้อมูลในแฟ้มข้อมูลภายใน (ILF's)

### ข้อมูลส่งออกภายนอก (External Outputs : EOs)

ข้อมูลที่เป็นผลลัพธ์จากการประมวลผลข้อมูลที่ได้รับจากภายในระบบเพื่อส่งข้อมูลไปยังภายนอก

### การสอบถามภายนอก (External Inquiries : EOs)

กระบวนการค้นหาภายในข้อมูลตามความต้องการผู้ใช้ แล้วระบบจะประมวลผลข้อมูลการค้นหาเพื่อแสดงผลต่อผู้ใช้ข้อมูล โดยข้อมูลจะไม่มี การปรับปรุงเปลี่ยนแปลงแฟ้มข้อมูลภายใน (ILF's) หรือแฟ้มข้อมูลต่อประสานภายนอก (EIS's)

### รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้

รายละเอียดของการวิเคราะห์และออกแบบซอฟต์แวร์ ที่ทำก่อนมีการพัฒนาระบบขึ้นจริง

### รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว

รายละเอียดของระบบที่มีการพัฒนาเสร็จสิ้นแล้วตามรายละเอียดซอฟต์แวร์ที่ออกแบบไว้

### แผนภาพกระแสข้อมูล (Data Flow Diagram)

แบบจำลองขั้นตอนการทำงานของระบบ เพื่ออธิบายขั้นตอนการทำงานของระบบ โดยจะแสดงทิศทางการไหลของข้อมูลและอธิบายความสัมพันธ์ในการดำเนินงานของระบบ

### แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล (ER-Diagram)

แบบจำลองข้อมูล เพื่ออธิบายลักษณะความสัมพันธ์ของกลุ่มข้อมูลที่ออกแบบ โดยมีการใช้สัญลักษณ์ในการแทนความหมายของส่วนประกอบต่างๆ ของแผนภาพ โดยประกอบด้วย เอนทิตี แอททริบิวต์ และความสัมพันธ์

### พจนานุกรมข้อมูล (Data Dictionary)

เครื่องมือที่ช่วยในการจัดเก็บรายละเอียดต่าง ๆ เกี่ยวกับข้อมูลให้เป็นหมวดหมู่ ทำให้สามารถค้นหารายละเอียดที่ต้องการได้โดยสะดวก

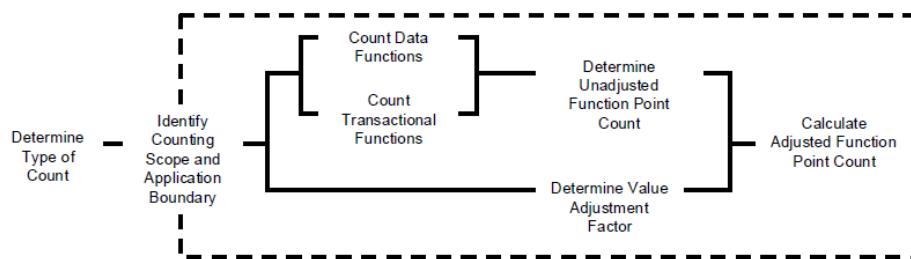
## บทที่ 2

### ทฤษฎีและวรรณกรรมที่เกี่ยวข้อง

ในบทนี้จะแสดงถึงผลจากการศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องกับงานวิจัยฉบับนี้ ประกอบด้วยเรื่องของการวิเคราะห์ฟังก์ชันพอยต์ (Function Point Analysis) แผนภาพกระแสข้อมูล(Data Flow Diagram : DFD) แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล(Entity Relationship Diagrams หรือ ER-Diagram) และการวิเคราะห์ค่านวนค่าฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล และจากแผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล

#### 2.1 การวิเคราะห์ฟังก์ชันพอยต์ (Function Point Analysis)

การประมาณขนาดซอฟต์แวร์ด้วยวิธีการวิเคราะห์ฟังก์ชันพอยต์ถูกพัฒนาขึ้นโดย Allan J. Albrecht ในขณะที่ทำงานอยู่บริษัท IBM ในช่วงกลางปีทศวรรษ 1970 ซึ่งเป็นแนวคิดการประมาณขนาดซอฟต์แวร์ด้วยวิธีการวิเคราะห์กระบวนการทำงานซอฟต์แวร์ในมุมมองของผู้ใช้(Albrecht, 1979) ซึ่งแนวคิดดังกล่าวต้องการให้การวัดขนาดซอฟต์แวร์(1)ไม่ยึดติดกับเทคโนโลยีหรือภาษาที่ใช้ในการพัฒนา (2)สามารถใช้งานได้ง่าย (3)สามารถประมาณค่าขนาดซอฟต์แวร์ได้อย่างสมเหตุสมผล ตั้งแต่ขั้นตอนการเก็บข้อมูลความต้องการ และ(4)ผู้ใช้งานสามารถเข้าใจวิธีการประมาณค่าซอฟต์แวร์ได้ง่าย (ต่อพงศ์ พรหมภูเบศ, 2555) (ชมพูนุท เผ่าประพัทธ์, 2555) การวิเคราะห์ฟังก์ชันพอยต์ได้เผยแพร่ครั้งแรกในปี 1979 โดยการนำเสนอแนวความคิดนี้ในงานประชุมนานาชาติที่แคลิฟอร์เนีย ซึ่งแนวคิดนี้ได้รับความสนใจและถูกนำไปพัฒนาให้เกิดวิธีการประมาณขนาดซอฟต์แวร์ในรูปแบบใหม่ๆจนกระทั่งในปี 1986 กลุ่มผู้ใช้งานฟังก์ชันพอยต์ระดับสากล (International Function point User Group หรือ IFPUG) ได้ก่อตั้งขึ้นเพื่อสร้างมาตรฐานการใช้งานการวิเคราะห์ฟังก์ชันพอยต์ เพื่อสร้างความเข้าใจในการใช้งานฟังก์ชันพอยต์ให้มีความถูกต้อง และมีมาตรฐานเดียวกัน สำหรับขั้นตอนการวิเคราะห์ฟังก์ชันพอยต์ดังรูปที่ 1 แสดงขั้นตอนการวิเคราะห์ฟังก์ชันพอยต์



รูปที่ 1 แสดงขั้นตอนการวิเคราะห์ฟังก์ชันพอยต์(International Function Points User Group, 1999)

### 2.1.1 ขั้นตอนที่ 1: ระบุประเภทของซอฟต์แวร์ที่จะนับ (Determine Type of Count)

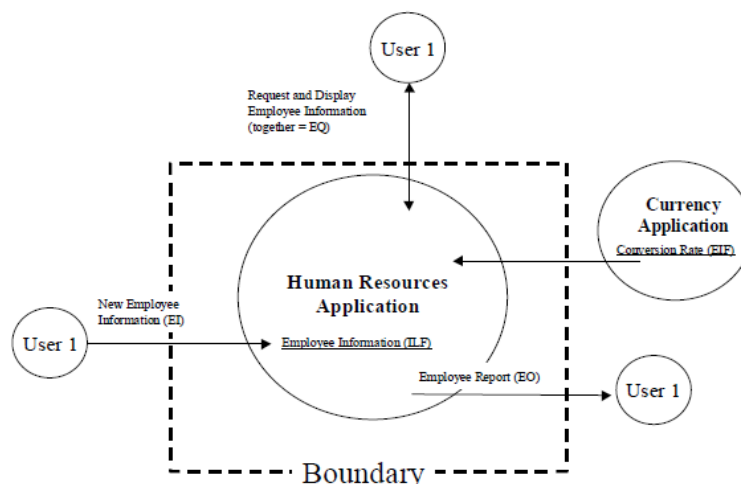
การวิเคราะห์ฟังก์ชันพอยต์สามารถใช้กับโครงการพัฒนาซอฟต์แวร์หรือการพัฒนาแอปพลิเคชัน โดยรูปแบบของโครงการพัฒนาซอฟต์แวร์หรือแอปพลิเคชันนั้นสามารถแบ่งตามประเภทโครงการได้เป็น 3 ประเภทโครงการ คือ

1. การนับฟังก์ชันพอยต์ในช่วงการพัฒนาโครงการซอฟต์แวร์ (Development Project Function Point Count) เป็นประเภทการนับสำหรับโครงการที่พัฒนาขึ้นใหม่ โดยสามารถวัดได้ทุกขั้นตอนของการพัฒนาซอฟต์แวร์ โดยสามารถใช้ฟังก์ชันพอยต์วัดได้ตั้งแต่ขั้นตอนการเก็บข้อมูลความต้องการ จนกระทั่งก่อนถึงการผลิตซอฟต์แวร์ การนับฟังก์ชันพอยต์ประเภทนี้เรียกว่า “A Baseline Function Point Count”
2. การนับฟังก์ชันพอยต์ในส่วนที่ต้องการเพิ่มเติมส่วนที่สร้างเสร็จแล้ว (Enhancement Project Function Point Count) เป็นประเภทของการนับสำหรับโครงการที่มีการปรับปรุงเพิ่มเติมจากระบบเดิมที่ต้องการปรับปรุงคุณภาพ หรือประสิทธิภาพของซอฟต์แวร์เพิ่มเติม โดยเป็นปกติของซอฟต์แวร์ที่จะได้รับการพัฒนาเพิ่มเติม โดยการนับสำหรับโครงการประเภทนี้จะเป็นการนับในส่วนที่เพิ่มเติม หรือการปรับปรุงจากระบบเก่า กล่าวคือจะนับส่วนที่เพิ่มเปลี่ยนแปลง หรือลบทิ้งจากฟังก์ชันการทำงานของซอฟต์แวร์เก่า
3. การนับฟังก์ชันพอยต์ในซอฟต์แวร์ที่พัฒนาเสร็จสมบูรณ์แล้ว (Application Function Point Count) เป็นประเภทของการนับสำหรับโครงการที่พัฒนาซอฟต์แวร์นั้นเสร็จเรียบร้อยแล้ว โดยการนับประเภทนี้เป็นการนับเพื่อประมาณขนาดซอฟต์แวร์ว่าปัจจุบันขนาดของซอฟต์แวร์เป็นอย่างไร ซึ่งจะนับทุกครั้งที่ซอฟต์แวร์สร้างเสร็จสมบูรณ์หรือนับทุกครั้งที่มีการปรับปรุงเพิ่มเติมแก้ไข ซึ่งการนับฟังก์ชันพอยต์แบบนี้จะนำไปใช้ในการติดตาม หรือตรวจสอบการบำรุงรักษาซอฟต์แวร์ เช่น การวัดจำนวนชั่วโมงต่อฟังก์ชันพอยต์

### 2.1.2 ขั้นตอนที่ 2 : ระบุขอบเขตระบบของซอฟต์แวร์ที่จะนับ (Identify Counting Scope and Application Boundary)

ขั้นตอนนี้เป็นการระบุขอบเขต (Boundary) เพื่อระบุคุณสมบัติของโครงการซอฟต์แวร์หรือแอปพลิเคชันที่เราจะประมาณขนาดซอฟต์แวร์ด้วยหลักการวิเคราะห์ฟังก์ชันพอยต์ โดยหลักการนับฟังก์ชันพอยต์จะจำแนกตามความต้องการของผู้ใช้เป็น 2 ประเภท คือ (1) ฟังก์ชันเชิงข้อมูล (Data Function) (2) ฟังก์ชันเชิงรายการเปลี่ยนแปลง (Transaction Function) โดยมีองค์ประกอบหลักในการทำงานของฟังก์ชันของซอฟต์แวร์ องค์ประกอบนี้จะเรียกชื่อว่า ประเภทฟังก์ชัน (Function Type) ซึ่งมีทั้งหมด 5 ประเภทฟังก์ชัน ได้แก่ (1) เพิ่มข้อมูลภายใน (Internal Logical Files : ILF's) (2) เพิ่มข้อมูลต่อประสานภายนอก (External Interface Files : EIF's) (3) ข้อมูลนำเข้าภายนอก (External Input Files : EIs) (4) ข้อมูลส่งออกภายนอก (External Outputs : EOs) และ (5) การ

สอบถามภายนอก (External Inquiries : EQs) ดังรูปที่ 2 แสดงการระบุดูขอบเขตของแอปพลิเคชันที่จะนับจำนวนฟังก์ชันพอยต์



รูปที่ 2 การระบุดูขอบเขตของแอปพลิเคชันที่จะนับจำนวนฟังก์ชันพอยต์

(International Function Point Users Group, 2004)

2.1.3 ขั้นตอนที่ 3 : นับจำนวนฟังก์ชันเชิงข้อมูล (Data Function)(International Function Points User Group, 1999)

ฟังก์ชันเชิงข้อมูล (Data Function) เป็นฟังก์ชันแสดงความต้องการข้อมูลทั้งภายในและภายนอกระบบ โดยฟังก์ชันที่อยู่ในการทำงานลักษณะนี้ได้แก่ (1)เพิ่มข้อมูลภายใน (Internal Logical Files : ILF's) และ (2)เพิ่มข้อมูลต่อประสานภายนอก (External Interface Files : EIF's)

- เพิ่มข้อมูลภายใน (Internal Logical Files : ILF's) คือ กลุ่มข้อมูลที่อยู่ในระบบ ตลอดช่วงอายุของระบบ และสามารถจัดการปรับปรุงเปลี่ยนแปลงข้อมูลภายในระบบ
- เพิ่มข้อมูลต่อประสานภายนอก (External Interface Files : EIF's) คือ กลุ่มข้อมูลที่อยู่ภายนอกระบบ หรือกลุ่มข้อมูลที่อยู่ในระบบนอกขอบเขตที่กำหนดการนับฟังก์ชัน กลุ่มข้อมูลจะใช้สำหรับการอ้างอิงเท่านั้น ซึ่งไม่สามารถปรับปรุงเปลี่ยนแปลงโดยระบบที่กำหนดให้นับฟังก์ชัน

การวิเคราะห์ความซับซ้อนของฟังก์ชันเชิงข้อมูล (Data Function)(International Function Points User Group, 1999)

ฟังก์ชันเชิงข้อมูล (Data Function) จะถูกประเมินค่าความซับซ้อน จากการนับองค์ประกอบของจำนวนเขตข้อมูล (Data Element Types : DETs) คือ ฟิลด์(Filed) ข้อมูลที่สนใจในแต่ละแฟ้มข้อมูล และจำนวนตารางข้อมูล (Record Element Types : RETs) ระเบียบหรือเรคคอร์ด หรือกลุ่มของย่อยของเขตข้อมูล ดังรายละเอียดดังต่อไปนี้

เกณฑ์นับจำนวนเขตข้อมูล (Data Element Types : DETs) และตารางข้อมูล (Record Element Types : RETs) ของแฟ้มข้อมูลภายใน (Internal Logical Files : ILF's) และแฟ้มข้อมูลต่อประสานภายนอก (External Interface Files : EIF's) ดังนี้

เกณฑ์การนับจำนวนเขตข้อมูล (Data Element Types : DETs)

- สำหรับแต่ละ 1 เขตข้อมูล (Filed) จะนับเป็น 1 DET's ตามการอ้างอิงหรือปรับปรุงข้อมูลเท่านั้น
- นับ 1 เขตข้อมูลสำหรับข้อมูลที่เป็นตัวเชื่อมความสัมพันธ์กับแฟ้มข้อมูลอื่น

เกณฑ์การนับจำนวนตารางข้อมูล (Record Element Types : RETs)

- นับ 1 ตารางข้อมูลสำหรับแต่ละกลุ่มข้อมูลที่เป็นกลุ่มข้อมูลย่อย
- กรณีแฟ้มข้อมูลไม่มีกลุ่มย่อยข้อมูลย่อยให้ถือว่า มี 1 ตารางข้อมูลเสมอ

ในการพิจารณาความซับซ้อนของแฟ้มข้อมูลภายใน (Internal Logical Files : ILF's) และแฟ้มข้อมูลต่อประสานภายนอก (External Interface Files : EIF's) นั้นจะนำค่าจำนวนเขตข้อมูล (DET's) และจำนวนตารางข้อมูล (RETs) ที่นับได้มาเทียบกับตารางความซับซ้อนที่กำหนด ดังตารางที่ 1 แสดงตารางความซับซ้อนของแฟ้มข้อมูลภายในและแฟ้มข้อมูลต่อประสานภายนอก ดังนี้

ตารางที่ 1 แสดงตารางความซับซ้อนของแฟ้มข้อมูลภายในและแฟ้มข้อมูลต่อประสานภายนอก

จำนวนตารางข้อมูล RET's	จำนวนเขตข้อมูล DET's		
	1 - 19	20 - 50	มากกว่า 50
1	ต่ำ	ต่ำ	ปานกลาง
2 - 5	ต่ำ	ปาน กลาง	สูง
มากกว่า 5	ปาน กลาง	สูง	สูง

2.1.4 ขั้นตอนที่ 4 : นับจำนวนฟังก์ชันเชิงรายการเปลี่ยนแปลง (Transaction Function)(International Function Points User Group, 1999)

ฟังก์ชันเชิงรายการเปลี่ยนแปลง(Transaction Function) เป็นฟังก์ชันแสดงรายการเปลี่ยนแปลงที่เกิดขึ้นภายในซอฟต์แวร์ ฟังก์ชันที่อยู่ในลักษณะการทำงานนี้ได้แก่ (1)ข้อมูลนำเข้าภายนอก (External Input Files : EIs) (2)ข้อมูลส่งออกภายนอก (External Outputs : EOs) และ(3)การสอบถามภายนอก (External Inquiries : EQs)

- ข้อมูลนำเข้าภายนอก (External Input Files : EIs) คือ ข้อมูลที่รับเข้ามาภายในระบบ โดยรับข้อมูลจากภายนอกเพื่อปรับปรุงข้อมูลในแฟ้มข้อมูลภายใน (ILF's)
- ข้อมูลส่งออกภายนอก (External Outputs : EOs) คือ ข้อมูลที่เป็นผลลัพธ์จากการประมวลผลข้อมูลที่ได้รับจากภายในระบบเพื่อส่งข้อมูลไปยังภายนอก
- การสอบถามภายนอก (External Inquiries : EQs) คือ กระบวนการค้นหาภายในข้อมูลตามความต้องการผู้ใช้ แล้วระบบจะประมวลผลข้อมูลการค้นหาเพื่อแสดงผลต่อผู้ใช้ ข้อมูล โดยข้อมูลจะไม่มีปรับปรุงเปลี่ยนแปลงแฟ้มข้อมูลภายใน (ILF's)หรือแฟ้มข้อมูลต่อประสานภายนอก (EIS's)

การวิเคราะห์ความซับซ้อนของฟังก์ชันเชิงรายการเปลี่ยนแปลง (Transaction Function)

ทรานแซคชันฟังก์ชัน (Transaction Function) จะถูกประเมินค่าความซับซ้อน จากการนับองค์ประกอบของจำนวนเขตข้อมูล (Data Element Types : DETs) คือ ไฟล์ด(Filed) ข้อมูลที่สนใจในแต่ละแฟ้มข้อมูล และจำนวนแฟ้มข้อมูลอ้างอิง (Files Type Referenced : FTRs) คือ แฟ้มข้อมูล

ที่ถูกอ้างอิงสำหรับรายการเปลี่ยนแปลง โดยเพิ่มข้อมูลอ้างอิง (FTRs) จะต้องเป็นเพิ่มข้อมูลภายใน (ILF's) หรือเพิ่มข้อมูลต่อประสานภายนอก (EIF's) ดังรายละเอียดดังต่อไปนี้

เกณฑ์นับจำนวนเขตข้อมูล (Data Element Types : DETs) และจำนวนเพิ่มข้อมูลอ้างอิง (Files Type Referenced : FTRs) ของข้อมูลนำเข้าภายนอก (External Input Files : EIs) ข้อมูลส่งออกภายนอก (External Outputs : EOs) และการสอบถามภายนอก (External Inquiries : EQs) ดังนี้

เกณฑ์การนับจำนวนเขตข้อมูล (Data Element Types : DETs)

ข้อมูลนำเข้าภายนอก (External Input Files : EIs)

- นับ 1 เขตข้อมูล (Filed) สำหรับข้อมูลที่ไม่ซ้ำกัน และปรับปรุงเพิ่มข้อมูลภายใน (ILF's)
- นับ 1 เขตข้อมูลสำหรับกลุ่มข้อมูลหลักประกอบด้วยข้อมูลรายละเอียดย่อย เช่น ข้อมูลที่อยู่ ประกอบด้วย บ้านเลขที่ ถนน ตำบล เป็นต้น ให้นับเป็น 1 เขตข้อมูลโดยการนับดังกล่าวขึ้นอยู่กับการใช้งานข้อมูลของผู้ใช้

ข้อมูลส่งออกภายนอก (External Outputs : EOs) และการสอบถามภายนอก (External Inquiries : EQs)

- นับ 1 เขตข้อมูลสำหรับข้อมูลที่ส่งไปยังภายนอกระบบ
- นับ 1 เขตข้อมูลสำหรับข้อมูลที่ส่งเข้ามาในระบบเพื่อใช้ค้นหาข้อมูลที่ต้องการ
- นับ 1 เขตข้อมูลสำหรับข้อมูลไปยังภายนอก เพื่อแจ้งเตือนสถานะหรือข้อผิดพลาด

เกณฑ์การนับจำนวนเพิ่มข้อมูลอ้างอิง (Files Type Referenced : FTRs)

ข้อมูลนำเข้าภายนอก (External Input Files : EIs)

- นับ 1 เพิ่มข้อมูลอ้างอิงสำหรับแต่ละเพิ่มข้อมูลภายใน (ILF's) ที่ถูกปรับปรุง
- นับ 1 เพิ่มข้อมูลอ้างอิงสำหรับการอ่านข้อมูลเพิ่มข้อมูลภายใน (ILF's) และเพิ่มข้อมูลต่อประสานภายนอก(EIF's)

ข้อมูลส่งออกภายนอก (External Outputs : EOs)

- นับ 1 เพิ่มข้อมูลอ้างอิงสำหรับแต่ละเพิ่มข้อมูลภายใน (ILF's) และเพิ่มข้อมูลต่อประสานภายนอก(EIF's) ที่ถูกอ่านระหว่างการประมวลผลข้อมูล
- นับ 1 เพิ่มข้อมูลอ้างอิงสำหรับการปรับปรุงเพิ่มข้อมูลภายใน (ILF's)

การสอบถามภายนอก (External Inquiries : EQs)

- นับ 1 เพิ่มข้อมูลอ้างอิงสำหรับแต่ละเพิ่มข้อมูลภายในหรือเพิ่มข้อมูลต่อประสานภายนอกที่ถูกอ่านระหว่างการประมวลผลข้อมูล

ในการพิจารณาความซับซ้อนของข้อมูลนำเข้าภายนอก (External Input Files : EIs) นั้นจะนำค่าจำนวนเขตข้อมูล (DETs) และจำนวนแฟ้มข้อมูลอ้างอิง (Files Type Referenced : FTRs) ที่นับได้มาเทียบกับตารางความซับซ้อนที่กำหนด ดังตารางที่ 2 แสดงตารางความซับซ้อนของข้อมูลนำเข้าภายนอก ดังนี้

ตารางที่ 2 แสดงตารางความซับซ้อนของข้อมูลนำเข้าภายนอก

จำนวนตารางข้อมูล FTR's	จำนวนเขตข้อมูล DET's		
	1 - 4	5 - 15	มากกว่า 15
น้อยกว่า 2	ต่ำ	ต่ำ	ปานกลาง
2	ต่ำ	ปาน กลาง	สูง
มากกว่า 2	ปาน กลาง	สูง	สูง

ในการพิจารณาความซับซ้อนของข้อมูลส่งออกภายนอก (External Outputs : EOs) และการสอบถามภายนอก (External Inquiries : EQs) นั้นจะนำค่าจำนวนเขตข้อมูล (DETs) และจำนวนแฟ้มข้อมูลอ้างอิง (Files Type Referenced : FTRs) ที่นับได้มาเทียบกับตารางความซับซ้อนที่กำหนด ดังตารางที่ 3 แสดงตารางความซับซ้อนของข้อมูลส่งออกภายนอกและการสอบถามภายนอก ดังนี้

ตารางที่ 3 แสดงตารางความซับซ้อนของข้อมูลส่งออกภายนอกและการสอบถามภายนอก

จำนวนตารางข้อมูล FTR's	จำนวนเขตข้อมูล DET's		
	1 - 5	6 - 19	มากกว่า 19
น้อยกว่า 2	ต่ำ	ต่ำ	ปานกลาง
2 - 3	ต่ำ	ปาน กลาง	สูง
มากกว่า 3	ปาน กลาง	สูง	สูง



2.1.5 ขั้นตอนที่ 5 : คำนวณค่าฟังก์ชันพอยต์ที่ยังไม่ได้ปรับค่า (Unadjusted Function Point : UAF)

ในขั้นตอนนี้จะเป็นการคำนวณค่าฟังก์ชันพอยต์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point : UAF) หลังจากที่ได้พิจารณาความซับซ้อนสำหรับแต่ละประเภทฟังก์ชันทั้ง 5 ประเภท (Eos, EQs, Eis, ILFs และ EIFs) ซึ่งเป็นคะแนนที่สะท้อนถึงจำนวนความสามารถและความซับซ้อนของฟังก์ชันที่ผู้ใช้ต้องการให้พัฒนาซอฟต์แวร์ ดังรูปที่ 3 แสดงรูปการให้ค่าน้ำหนักของแต่ละประเภทฟังก์ชันทั้ง 5 ประเภท

Type of Component	Complexity of Components			Total
	Low	Average	High	
External Inputs	x 3 =	x 4 =	x 6 =	
External Outputs	x 4 =	x 5 =	x 7 =	
External Inquiries	x 3 =	x 4 =	x 6 =	
Internal Logical Files	x 7 =	x 10 =	x 15 =	
External Interface Files	x 5 =	x 7 =	x 10 =	
Total Number of Unadjusted Function Points				

รูปที่ 3 แสดงรูปตารางการให้ค่าน้ำหนักของแต่ละประเภทฟังก์ชันทั้ง 5 ประเภท(International Function Points User Group, 1999)

2.1.6 ขั้นตอนที่ 6 : กำหนดตัวแปรปรับค่า (Determine Value Adjustment Factor)

ปัจจัยในการปรับค่าฟังก์ชันพอยต์ (Value Adjustment Factors หรือ VAF) หลังจากที่ได้ค่าฟังก์ชันพอยต์ที่ยังไม่ได้ปรับค่า (Unadjusted Function Point : UAF) นั้นยังไม่ได้มีการพิจารณาถึงปัจจัยสภาพแวดล้อมของระบบโดยทั่วไป (General System Characteristic : GSC) ที่อาจส่งผลกระทบต่อการประมาณขนาดซอฟต์แวร์ด้วยวิธีฟังก์ชันพอยต์ จึงต้องทำการปรับค่าด้วยปัจจัยสำหรับการปรับค่า (Value Adjustment Factor : VAF) หรือค่าการปรับความซับซ้อนทางเทคนิค (Technical Complexity Adjustment : TCA) ซึ่งปัจจุบันกลุ่มผู้ใช้งานฟังก์ชันพอยต์ระหว่างประเทศ (International Function Point Users Group : IFPUG)(International Function Points User Group, 1999) ได้กำหนดค่าปัจจัยไว้ทั้งหมด 14 ปัจจัย โดยปัจจัยทั้ง 14 ปัจจัยประกอบด้วย

1. ปัจจัยการสื่อสารข้อมูล (Data Communication)
2. ปัจจัยการประมวลผลข้อมูลแบบกระจาย (Distributed Data Processing)
3. ปัจจัยสมรรถนะ (Performance)
4. ปัจจัยความต้องการใช้งานหนักด้านฮาร์ดแวร์ (Heavily Used Configuration)
5. ปัจจัยอัตราการรายการเปลี่ยนแปลง (Transaction Rate)
6. ปัจจัยการบันทึกข้อมูลออนไลน์ (On-Line Data Entry)
7. ปัจจัยประสิทธิภาพเพื่อผู้ใช้ชั้นปลาย (End-User Efficiency)
8. ปัจจัยการปรับให้เป็นปัจจุบันแบบออนไลน์ (On-Line Update)
9. ปัจจัยการประมวลผลที่ซับซ้อน (Complex Processing)
10. ปัจจัยความสามารถนำกลับมาใช้ใหม่ (Reusability)

11. ปัจจัยความสะดวกในการติดตั้ง (Installation Ease)
  12. ปัจจัยความสะดวกในการปฏิบัติงาน (Operation Ease)
  13. ปัจจัยการใช้งานในหลายสถานที่ (Multiple Sites)
  14. ปัจจัยอำนวยความสะดวกในการเปลี่ยนแปลง (Facilitate Change)
- โดยให้ผู้ประมาณค่ากำหนดคะแนนอิทธิพลของปัจจัยในแต่ละด้าน ตั้งแต่ 0 – 5 คะแนนขึ้นกับว่าปัจจัยในแต่ละด้านมีผลกระทบกับระบบที่จะพัฒนามากน้อยเพียงใด ดังตารางที่ 4 แสดงตารางเกณฑ์ระดับคะแนนของตัวแปรปัจจัยลักษณะของระบบ ดังนี้

ตารางที่ 4 แสดงตารางเกณฑ์ระดับคะแนนของตัวแปรปัจจัยลักษณะของระบบ

ระดับคะแนน	เกณฑ์ระดับคะแนนตัวแปรปัจจัยลักษณะของระบบ
0	ไม่มีผลกระทบกับระบบ (Not Present, or no influence)
1	มีผลกระทบกับระบบเล็กน้อย (Insignificant influence)
2	มีผลกระทบกับระบบพอสมควร (Moderate influence)
3	มีผลกระทบกับระบบปานกลาง (Average influence)
4	มีผลกระทบกับระบบมาก (Significant influence)
5	มีผลกระทบกับระบบมากที่สุด (Strong influence Change)

โดยผู้ประมาณค่าจะใส่คะแนนลงตารางคะแนนปัจจัยลักษณะของระบบโดยทั่วไป ดังตารางที่ 5 แสดงตารางคำนวณผลรวมระดับคะแนนของตัวแปรปัจจัยลักษณะของระบบ ดังนี้

ตารางที่ 5 แสดงตารางคำนวณผลรวมระดับคะแนนของตัวแปรปัจจัยลักษณะของระบบ

ลำดับ	คุณลักษณะของระบบ	ระดับผลกระทบ (DI)
1	ปัจจัยการสื่อสารข้อมูล (Data Communication)	
2	ปัจจัยการประมวลผลข้อมูลแบบกระจาย (Distributed Data Processing)	
3	ปัจจัยสมรรถนะ (Performance)	
4	ปัจจัยความต้องการใช้งานหนักด้านฮาร์ดแวร์ (Heavily Used)	

ลำดับ	คุณลักษณะของระบบ	ระดับผลกระทบ (DI)
	Configuration)	
5	ปัจจัยอัตราการรายการเปลี่ยนแปลง (Transaction Rate)	
6	ปัจจัยการบันทึกข้อมูลออนไลน์ (On-Line Data Entry)	
7	ปัจจัยประสิทธิภาพเพื่อผู้ใช้ชั้นปลาย (End-User Efficiency)	
8	ปัจจัยการปรับให้เป็นปัจจุบันแบบออนไลน์ (On-Line Update)	
9	ปัจจัยการประมวลผลที่ซับซ้อน (Complex Processing)	
10	ปัจจัยความสามารถนำกลับมาใช้ใหม่ (Reusability)	
11	ปัจจัยความสะดวกในการติดตั้ง (Installation Ease)	
12	ปัจจัยความสะดวกในการปฏิบัติงาน (Operation Ease)	
13	ปัจจัยการใช้งานในหลายสถานที่ (Multiple Sites)	
14	ปัจจัยอำนวยความสะดวกในการเปลี่ยนแปลง (Facilitate Change)	
รวมผลกระทบ (TDI)		

เมื่อได้ผลรวมคะแนนอิทธิพลจากปัจจัยความซับซ้อนทางเทคนิคจากตารางคะแนนปัจจัยลักษณะของระบบโดยทั่วไปแล้วนำค่าผลรวมที่ได้มาคำนวณค่าตัวแปรปรับค่า (Value Adjustment Factor : VAF) จากสมการคำนวณค่าตัวคูณปรับค่าดังนี้

$$VAF = (TDI * 0.01) + 0.65$$

2.1.7 ขั้นตอนที่ 7 : คำนวณค่ารวมฟังก์ชันพอยต์ที่ได้ปรับค่าแล้ว(Calculate Adjusted Function Point Count)

หลังจากที่ได้ค่าของฟังก์ชันพอยต์ที่ยังไม่ถูกปรับค่า(UAF)และค่าของปัจจัยในการปรับค่าฟังก์ชันพอยต์(VAF) แล้ว จะสามารถหาค่าของฟังก์ชันพอยต์ที่ได้รับการปรับค่าแล้ว (Adjusted Function Points)ได้ โดยคำนวณตามสูตรการคำนวณดังนี้

$$FP = VAF \times UAF$$

## 2.2 แผนภาพกระแสข้อมูล (Data Flow Diagram หรือ DFD)

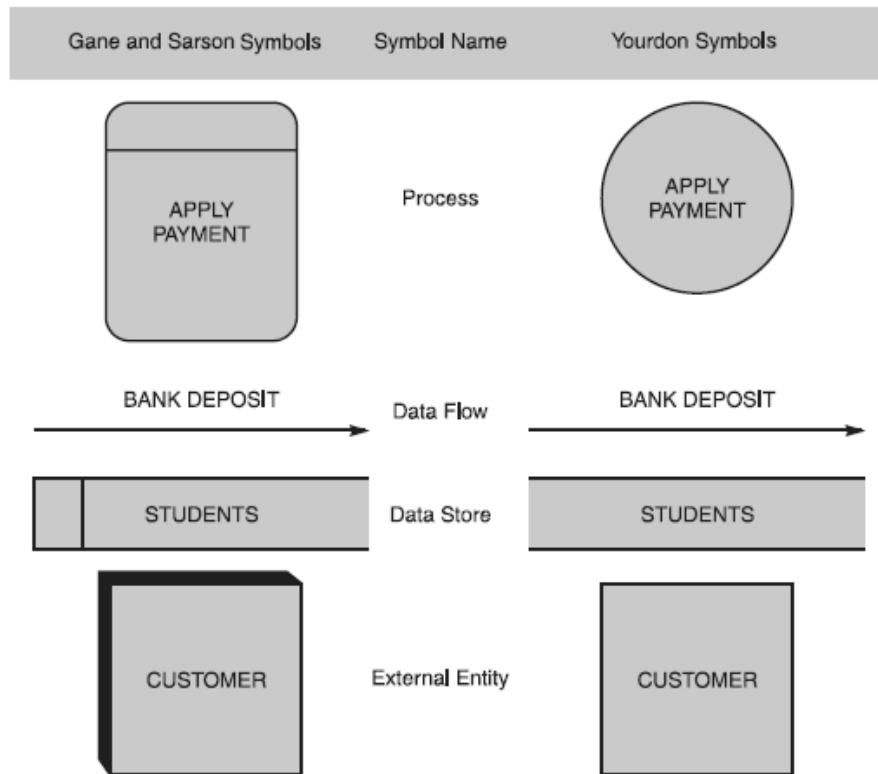
แผนภาพกระแสข้อมูล หมายถึงแผนภาพที่แสดงให้เห็นทิศทางการไหลของข้อมูลที่มีอยู่ในซอฟต์แวร์ และขั้นตอนการทำงานของซอฟต์แวร์ แผนภาพกระแสข้อมูลมีความสำคัญต่อการวิเคราะห์และออกแบบระบบสารสนเทศเพราะแผนภาพกระแสข้อมูลจะแสดงความสัมพันธ์ระหว่างกระบวนการ(Process) กับข้อมูล(Data) และสิ่งที่เกี่ยวข้องในเชิงตรรกะ (Logical) ซึ่งจะทำให้ทราบถึงการไหลเวียนของข้อมูลว่าข้อมูลมาจากไหน ข้อมูลจะไปที่ไหน จัดเก็บข้อมูลที่ไหน และเกิดอะไรกับข้อมูลบ้างในแต่ละขั้นตอนของซอฟต์แวร์ (ณัฐพันธ์ เขจรนันท์, 2551)

โดยวัตถุประสงค์ของการเขียนแผนภาพกระแสข้อมูล มีดังนี้ (ณัฐพันธ์ เขจรนันท์, 2551)

- แสดงความสัมพันธ์ของกระบวนการ และการไหลของข้อมูลในซอฟต์แวร์
- สรุปรูปแบบการวิเคราะห์ในรูปแบบที่เป็นแผนภาพ มีโครงสร้างที่ชัดเจน
- เป็นสิ่งที่ช่วยทำความเข้าใจและเป็นข้อตกลงการพัฒนาร่วมกันระหว่างผู้ที่เกี่ยวข้องกับซอฟต์แวร์ โดยเฉพาะนักวิเคราะห์ระบบ โปรแกรมเมอร์ และผู้ใช้งานซอฟต์แวร์
- เป็นเอกสารและข้อมูลที่สำคัญในการพัฒนาระบบสารสนเทศ โดยเฉพาะการออกแบบระบบใหม่

### 2.2.1 สัญลักษณ์ที่ใช้ในแผนภาพกระแสข้อมูล

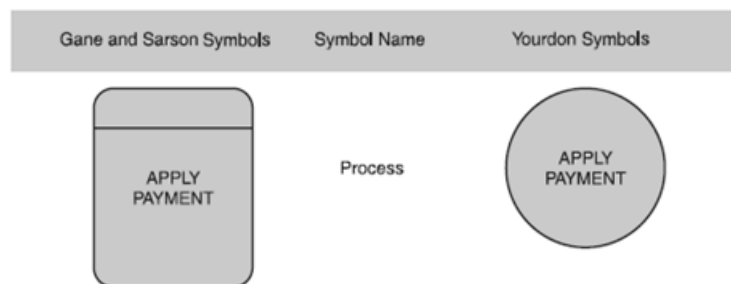
สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูลนั้นแต่เดิมมีมาตรฐานอยู่ 2 มาตรฐาน คือ มาตรฐานของ Yourdon และมาตรฐานของ Gane และ Sarson (ณัฐพันธ์ เขจรนันท์, 2551; รุจิจันทร์ พิริยะสงวนพงศ์, 2549) โดยสัญลักษณ์ที่ใช้ในการสร้างแผนภาพกระแสข้อมูล จะประกอบด้วย 4 สัญลักษณ์ ได้แก่สัญลักษณ์ที่ใช้แทนกระบวนการ(Process) กระแสข้อมูล(Data Flow) แฟ้มข้อมูล(Data Store) และหน่วยงานหรือเอนทิตีภายนอก(External Entity) ดังตัวอย่างในรูปที่ 4 โดยสัญลักษณ์ที่จะใช้ประกอบเป็นภาพอธิบายในบทนี้ จะใช้สัญลักษณ์ตามมาตรฐานของYourdon



รูปที่ 4 สัญลักษณ์ที่ใช้ในแผนภาพกระแสข้อมูลตามมาตรฐานของ Gane และ Sarson (Gane & Sarson, 1979)

หลักการเขียนสัญลักษณ์ต่างๆมีดังนี้ (ณัฐพันธ์ เขจรนนท์, 2551)

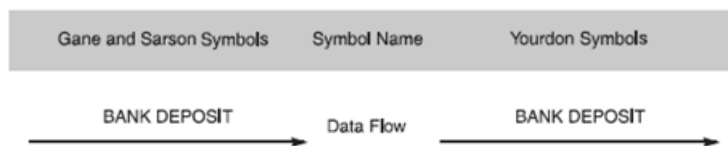
- กระบวนการ(Process)คือการทำงานหรือการกระทำที่ใช้ข้อมูลเพื่อทำให้ข้อมูลมีการจัดเก็บหรือเปลี่ยนรูปแบบของข้อมูล หรือเผยแพร่ไปยังที่ต่างๆ โดยใช้สัญลักษณ์ดังรูปที่ 5 โดยชื่อของการประมวลผลจะเป็นตัวบอกว่าการประมวลผลนั้นทำหน้าที่อะไร ควรใช้คำที่มีความหมายแน่นอน และควรใช้คำกริยา เช่น บันทึก ตรวจสอบ แก้ไข ปรับปรุง เป็นต้น



รูปที่ 5 สัญลักษณ์ของกระบวนการของทั้ง 2 มาตรฐาน

- กระแสข้อมูล(Data Flow) กระแสข้อมูลนั้นจะแทนด้วยลูกศร โดยมีชื่อข้อมูลกำกับลูกศรนั้น ดังรูปที่ 6 ข้อมูลจะไหลระหว่างการประมวลผลต่างๆ หรือมาจากเอนทิตีภายนอก ซึ่งข้อมูลที่

เคลื่อนที่อาจเป็นข้อมูลเดี่ยวๆ เช่น รหัสลูกค้า ชื่อลูกค้า หรืออาจเป็นกลุ่มข้อมูล เช่น ข้อมูลพนักงาน ข้อมูลลูกค้า เป็นต้น



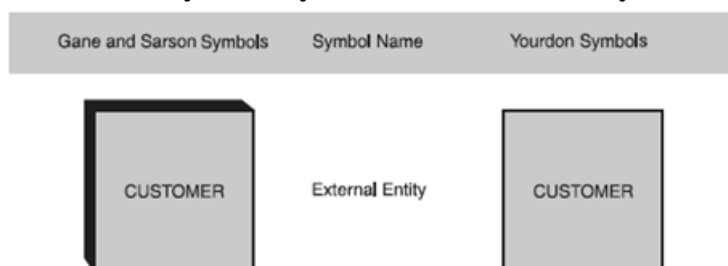
รูปที่ 6 สัญลักษณ์ของกระแสข้อมูลของทั้ง 2 มาตรฐาน

3. แฟ้มข้อมูล(Data Store)ใช้สำหรับเก็บข้อมูล แทนด้วยสัญลักษณ์ ดังรูปที่ 7 การตั้งชื่อแฟ้มข้อมูลควรเป็นคำนาม ซึ่งข้อมูลที่ถูกเก็บในแฟ้มข้อมูลจะถูกเรียกใช้เมื่อต้องการ ถ้าหัวลูกศรวิ่งเข้าสู่แฟ้มข้อมูลแสดงว่ามีการเขียนข้อมูลหรือแก้ไขข้อมูลในแฟ้มข้อมูล ถ้าลูกศรวิ่งออกจากแฟ้มข้อมูลแสดงว่ามีการอ่านข้อมูล



รูปที่ 7 สัญลักษณ์ของแฟ้มข้อมูลของทั้ง 2 มาตรฐาน

4. หน่วยงานหรือเอนทิตีภายนอก(External Entity)เป็นต้นกำเนิด หรือจุดสิ้นสุดของข้อมูล เป็นระบบที่อยู่ภายนอกขอบเขตของซอฟต์แวร์ แทนสัญลักษณ์ดังรูปที่ 8 ระบบภายนอกส่วนใหญ่นั้นจะเป็นตัวบุคคลหรือองค์กรต่างๆที่อยู่นอกระบบ ซึ่งอาจเป็นแหล่งที่ส่งข้อมูลเข้าสู่ระบบหรือแหล่งที่รับข้อมูลจากระบบ โดยเราจะไม่สนใจการทำงานภายในของสิ่งที่อยู่นอกระบบ เราจะสนใจเฉพาะข้อมูลที่วิ่งเข้าสู่ระบบหรือออกจากระบบไปสู่ภายนอกเท่านั้น

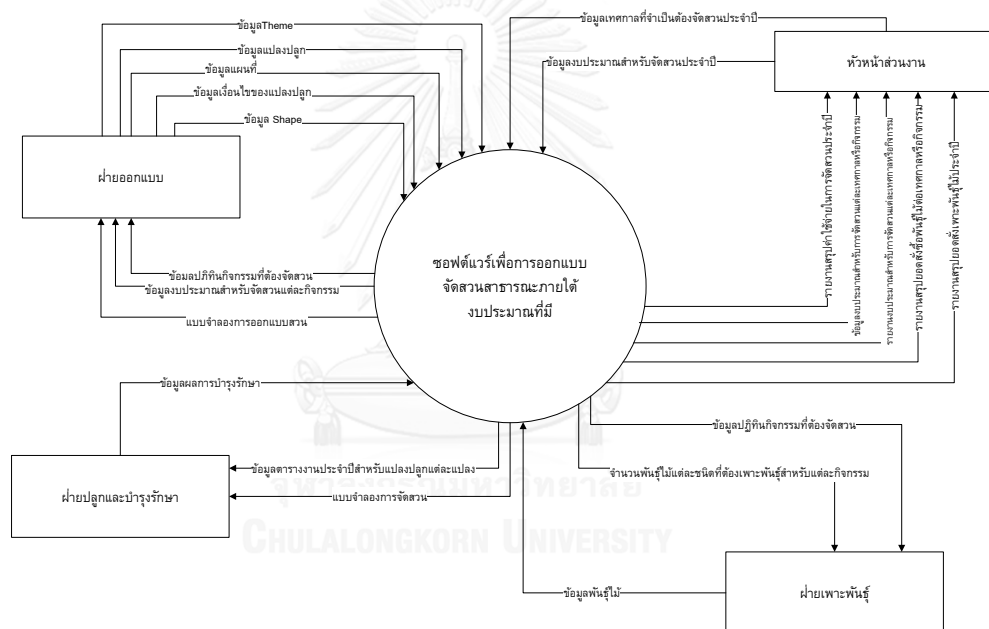


รูปที่ 8 สัญลักษณ์ของเอนทิตีภายนอกของทั้ง 2 มาตรฐาน

#### 2.2.2. ระดับของแผนภาพกระแสข้อมูล (รุจิจันทร์ พิริยะสงวนพงศ์, 2549)

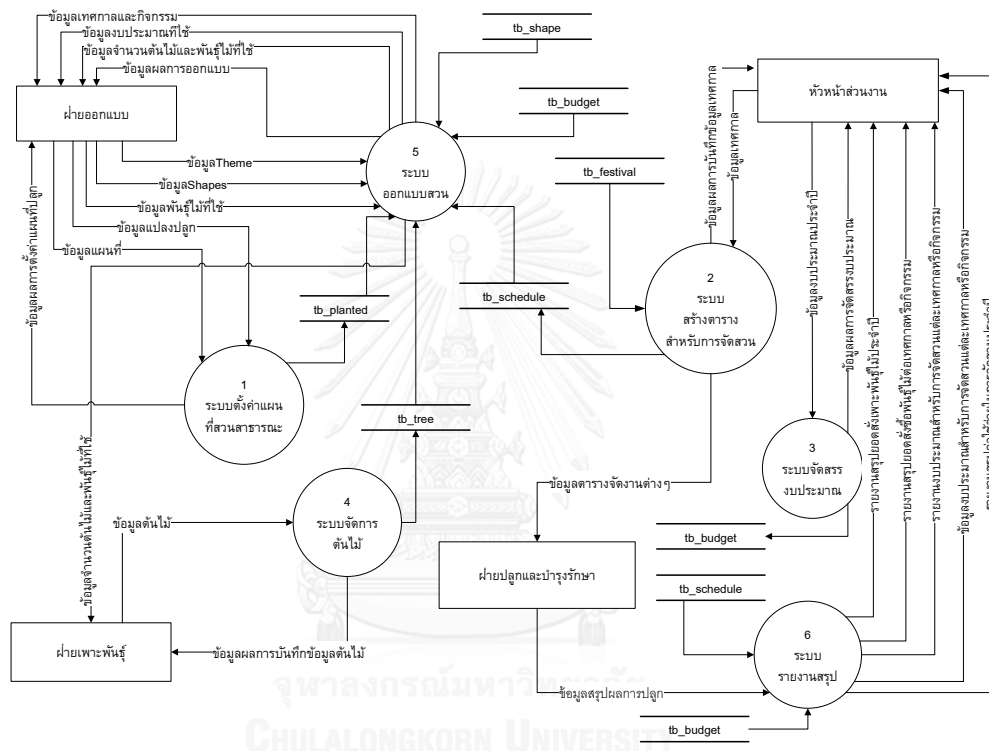
ในการแบ่งระดับของการสร้างแผนภาพกระแสข้อมูลนั้นจะไม่มีกฎเกณฑ์ที่แน่นอนตายตัวว่าควรมีกี่ระดับ โดยทั่วไปจะประกอบไปด้วย

- แผนภาพบริบท (Context Diagram) คือโครงสร้างเริ่มแรกซึ่งแสดงให้เห็นภาพรวมและขอบเขตของซอฟต์แวร์ ในการเขียนแผนภาพจะเริ่มต้นจากการวาดสัญลักษณ์ของกระบวนการที่ใช้แทนระบบงานทั้งระบบไว้กลางหน้ากระดาษ แล้ววาดสัญลักษณ์ของเอนทิตีภายนอกที่เกี่ยวข้องทั้งหมดของระบบไว้รอบๆกระบวนการ และเชื่อมกระบวนการกับเอนทิตีภายนอกด้วยสัญลักษณ์กระแสข้อมูลรับเข้าและส่งออกจากกระบวนการ การเขียนแผนภาพบริบทจะต้องเขียนให้อยู่ภายใน 1 หน้ากระดาษเท่านั้น ชื่อของกระบวนการควรเป็นชื่อของระบบงาน ตัวอย่างเช่นรูปที่ 9 แสดงแผนภาพบริบทของซอฟต์แวร์เพื่อการออกแบบจัดสวนสาธารณะภายใต้งบประมาณที่มี



รูปที่ 9 แผนภาพบริบทของซอฟต์แวร์เพื่อการออกแบบจัดสวนสาธารณะภายใต้งบประมาณที่มี

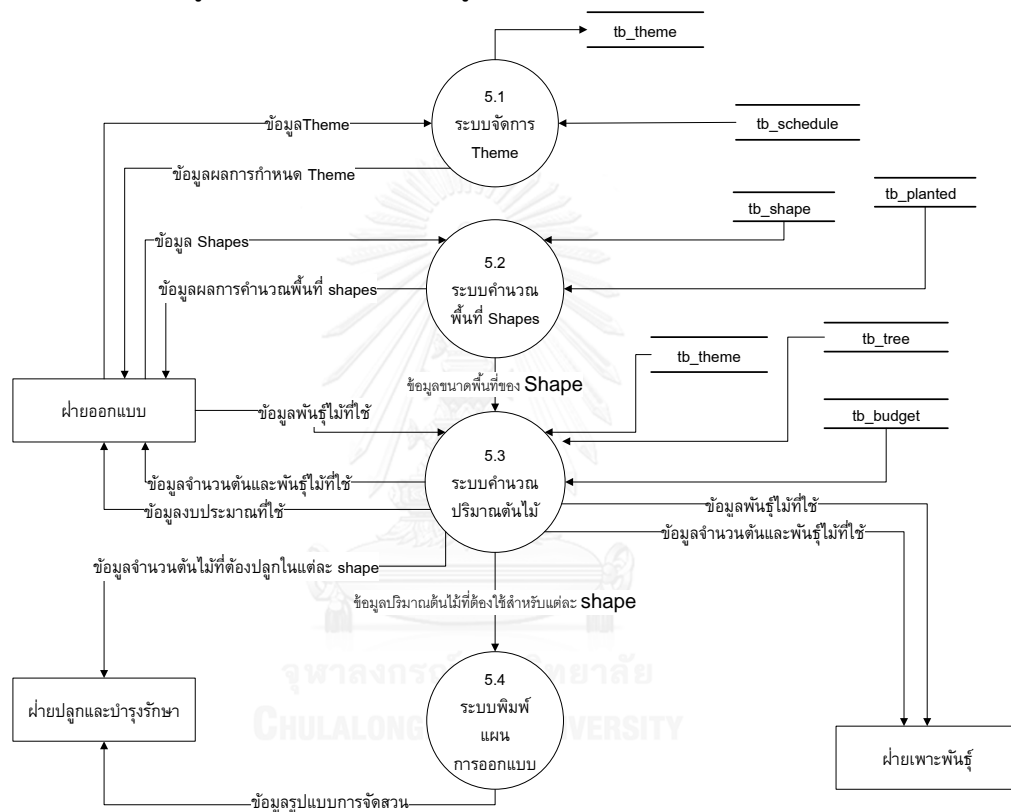
- แผนภาพระดับหนึ่ง (Level-1 Diagram หรือ / Parent Diagram)คือโครงสร้างที่ใช้อธิบายรายละเอียดของแผนภาพบริบทซึ่งเปรียบเสมือนกล่องดำที่ต้องการคำอธิบายเพิ่มเติม เป็นการนำกระบวนการหลักมาแตกเป็นรายละเอียดกระบวนการย่อยที่อยู่ภายในระบบให้ชัดเจนยิ่งขึ้น โดยมีการแสดงกระแสข้อมูลและเอนทิตีภายนอกคงเดิมแต่มีการเพิ่มเติมรายละเอียดของกระแสข้อมูลและหน่วยเก็บข้อมูลเพิ่มเติมขึ้น (รุจิจันทร์ พิริยะสงวนพงศ์, 2549) ตัวอย่างเช่นรูปที่ 10 แสดงแผนภาพระดับหนึ่งของซอฟต์แวร์เพื่อการออกแบบจัดสวนสาธารณะภายใต้งบประมาณที่มี



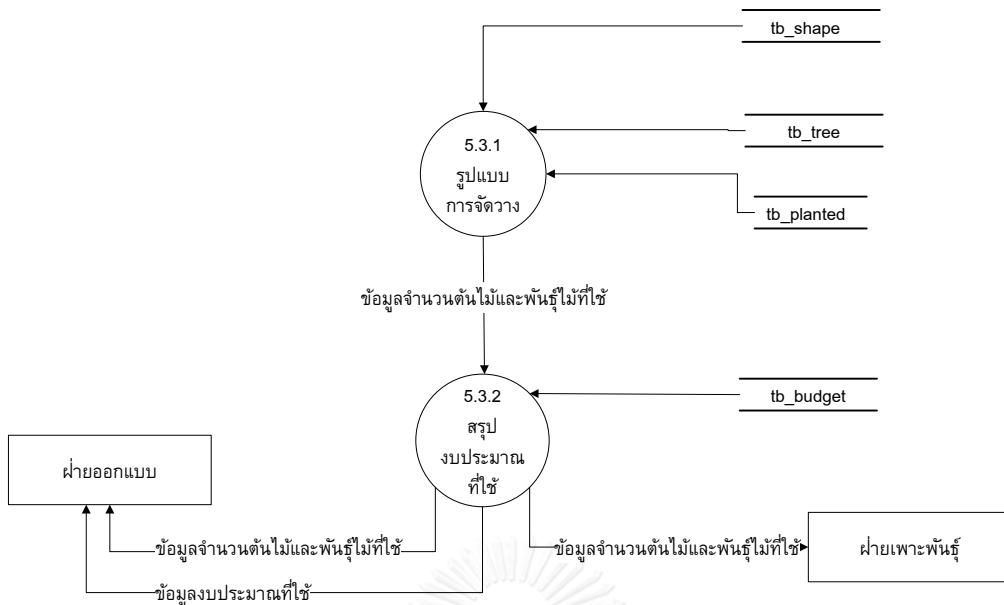
รูปที่ 10 แสดงแผนภาพระดับหนึ่งของซอฟต์แวร์เพื่อการออกแบบจัดสวนสาธารณะภายใต้งบประมาณที่มี



- แผนภาพระดับลูก (Child Diagram) คือโครงสร้างที่ใช้อธิบายในรายละเอียดของแผนภาพระดับหนึ่ง โดยมีการนำขบวนการทั้งหมดของแผนภาพระดับหนึ่งมาขยายรายละเอียดให้ชัดเจนยิ่งขึ้นในแต่ละกระบวนการ ซึ่งหากมีรายละเอียดการทำงานปลีกย่อยจากระดับนี้อีก ก็แตกรายละเอียดลงไปจนกระทั่งสิ้นสุดกระบวนการ ส่วนชื่อของระดับก็จะเปลี่ยนไปเป็น Level-2 Diagram หรือ Level-3 Diagram ไปเรื่อยๆ จนกระทั่งจบกระบวนการทั้งหมด ตัวอย่างเช่นรูปที่ 11 แผนภาพระดับลูก Level-2 Diagram ของระบบออกแบบสวน และตัวอย่างเช่นรูปที่ 12 แผนภาพระดับลูก Level-3 Diagram ของระบบคำนวณปริมาณต้นไม้



รูปที่ 11 แผนภาพระดับลูก Level-2 Diagram ของระบบออกแบบสวน



รูปที่ 12 แผนภาพระดับลูก Level-3 Diagram ของระบบคำนวณปริมาณต้นไม้

2.3 แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล (Entity Relationship Diagrams หรือ ER-Diagram)  
 แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล(Entity Relationship Diagrams หรือ ER-Diagram) เป็นแบบจำลองข้อมูลที่ถูกพัฒนาขึ้นในปี ค.ศ. 1976 โดย Dr. Peter Chen โดยแผนภาพนี้จะใช้ในการอธิบายลักษณะความสัมพันธ์ของกลุ่มข้อมูลที่ออกแบบ โดยมีการใช้สัญลักษณ์ในการแทนความหมายของส่วนประกอบต่างๆของแผนภาพทำให้เข้าใจ (สมศักดิ์ โชคชัยชุตติกุล, 2553) ดังรูปที่ 13 คือตัวอย่างสัญลักษณ์แผนภาพ ER Diagram ของ Dr. Peter Chen และรูปที่ 14 ตัวอย่างสัญลักษณ์แผนภาพ ER Diagram ตามรูปแบบแบบจำลองวิศวกรรมสารสนเทศ (Information Engineering models หรือ IE models)



รูปที่ 13 สัญลักษณ์ของ ER Diagram ตามรูปแบบของ Dr. Peter Chen



รูปที่ 14 สัญลักษณ์ของ ER Diagram ตามรูปแบบแบบจำลองวิศวกรรมสารสนเทศ

แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล(Entity Relationship Diagrams หรือ ER-Diagram)ประกอบด้วยองค์ประกอบพื้นฐานดังนี้ (สมศักดิ์ โชคชัยชุตติกุล, 2553)

องค์ประกอบที่ 1 เอนทิตี (Entity) เป็นตัวแทนของชุดข้อมูล หรือสิ่งของที่เราสนใจในระบบงานนั้น ๆ ซึ่งอาจเป็นรูปธรรม เช่น คน สัตว์สิ่งของ สถานที่ หรือเป็นนามธรรม เช่น การสั่งซื้อ การขาย การฝึกอบรม เป็นต้น ลักษณะของเอนทิตีสามารถแบ่งได้ออกเป็น 3 ประเภท คือ

- Strong Entity เอนทิตีชนิดนี้จะไม่ขึ้นกับเอนทิตีอื่นๆ โดยจะมีแอททริบิวต์(Attribute)อย่างน้อย 1 แอททริบิวต์ที่ทำหน้าที่เป็นตัวระบุ(Identifier) โดยตัวระบุนี้จะใช้แยกความแตกต่างของแต่ละรายการ(Record)ได้
- Weak Entity เป็นเอนทิตีซึ่งความมีอยู่ของข้อมูลนั้นจะขึ้นอยู่กับ Strong Entity โดยที่ลำพังแอททริบิวต์ของ Weak Entity เองนั้นจะไม่สามารถใช้เป็นตัวระบุได้ ต้องอาศัยตัวระบุจาก Strong Entity ซึ่งเป็นเอนทิตีหลักมาใช้ร่วมกับบางแอททริบิวต์ที่มีอยู่เป็นตัวระบุ ตัวอย่างเช่น เอนทิตีสินค้าคงเหลือ จะเก็บข้อมูลสินค้าคงเหลือที่อยู่ในแต่ละสาขา
- Associative Entity เป็นเอนทิตีที่เกิดจากความสัมพันธ์ระหว่างเอนทิตี 2 เอนทิตีขึ้นไป และมีความต้องการเก็บข้อมูลบางอย่างไว้ในความสัมพันธ์นั้น

องค์ประกอบที่ 2 แอททริบิวต์ (Attribute)เป็นตัวแทนของคุณสมบัติเฉพาะตัวต่างๆของเอนทิตีข้อมูล โดยเอนทิตีหนึ่งมักมีหลายแอททริบิวต์ โดยแอททริบิวต์สามารถจำแนกได้หลายลักษณะ เช่น จำแนกตามจำนวนข้อมูลในแอททริบิวต์ สามารถจำแนกได้เป็นSingle valued และ Multi valued attribute

- Single valued คือ ค่าของเอนทิตีที่สามารถมีได้แค่ค่าเดียว เช่น วันเกิด สำหรับพนักงาน แล้วสามารถมีได้เพียงค่าเดียว
- Multi valued คือ ค่าที่เป็นไปได้มากกว่า 1 ค่า เช่น ทำเลที่ตั้งของโรงงานสามารถมีได้มากกว่า 1 แห่ง

จำแนกตามองค์ประกอบของข้อมูลในแอททริบิวต์ สามารถจำแนกได้เป็น Simple และ Composite Attribute

- Simple Attribute คือ Attribute ที่ไม่สามารถแยกออกเป็นส่วนย่อยได้เช่น เลขที่บ้าน ถนน
- Composite Attribute คือ Attribute ที่สามารถแยกออกเป็นส่วนย่อยได้เช่น ชื่อ อาจจะถูกประกอบด้วยชื่อต้น และชื่อสกุล เป็นต้น

องค์ประกอบที่ 3 ความสัมพันธ์ (Relationship) คือ ทำหน้าที่เป็นตัวเชื่อมระหว่างเอนทิตี มักบอกถึงสถานะ อาการ หรือกิจกรรมต่างๆที่เกิดขึ้นระหว่างเอนทิตี


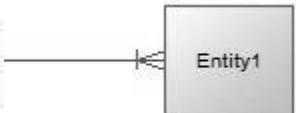


ระดับขั้นของความสัมพันธ์ (Degree of Relationship) จะบอกถึงความสัมพันธ์ระหว่างเอนทิตี มีดังนี้

- ความสัมพันธ์เอนทิตีเดียว (Unary Relationships) หมายถึง เอนทิตีหนึ่ง ๆ จะมีความสัมพันธ์กับตัวมันเอง

- ความสัมพันธ์สองเอนทิตี (Binary Relationships) หมายถึง เอนทิตีสองเอนทิตีที่มีความสัมพันธ์กัน
- ความสัมพันธ์สามเอนทิตี (Ternary Relationships) หมายถึง เอนทิตีสองเอนทิตีที่มีความสัมพันธ์กัน

จำนวนสมาชิกในความสัมพันธ์ (Relationship Cardinality) เป็นการจำแนกรูปแบบของความสัมพัทธ์ตามจำนวนของเอนทิตี ว่าแต่ละหน่วยของเอนทิตีข้อมูลหนึ่งจะสามารถมีความสัมพันธ์กับอีกเอนทิตีข้อมูลอีกเอนทิตีหนึ่งได้กี่หน่วยตัวอย่างเช่น นักศึกษากับคณะ นักศึกษาคนหนึ่งสามารถอยู่ได้เพียงคณะเดียวเท่านั้น และคณะหนึ่งคณะมีนักศึกษาได้ตั้งแต่ 1 คนขึ้นไป เป็นต้นรูปแบบของจำนวนสมาชิกในความสัมพันธ์สามารถแบ่งได้ตามตารางที่ 6 ตารางแสดงรูปแบบของจำนวนสมาชิกในความสัมพันธ์

ตารางที่ 6 แสดงรูปแบบของจำนวนสมาชิกในความสัมพันธ์

ชื่อเรียก	จำนวนต่ำสุด	จำนวนสูงสุด	สัญลักษณ์ตามรูปแบบแบบจำลองวิศวกรรมสารสนเทศ
Mandatory one	1	1	
Mandatory many	1	ไม่จำกัด	
Optional one	0	1	
Optional many	0	ไม่จำกัด	

#### 2.4 การนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล

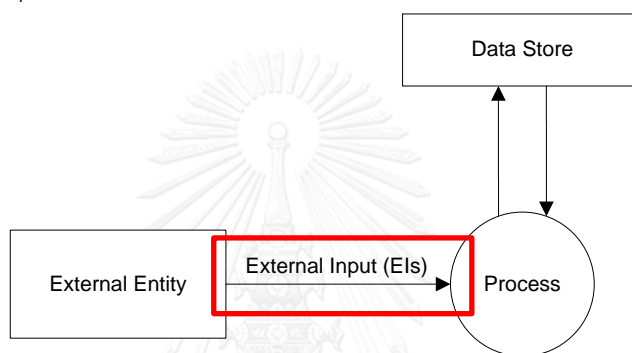
งานวิจัยนี้ศึกษาหลักในการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลของ Rask (Rask, 1991) และ Lamma และคณะ (E. Lamma et al., 2003) ซึ่งมีรายละเอียดดังนี้

##### 2.4.1 หลักการนับจำนวนฟังก์ชันพอยต์ที่ยังไม่ปรับค่าจากแผนภาพกระแสข้อมูล

Rask (Rask, 1991) ได้อธิบายวิธีการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลไว้ดังนี้

#### 2.4.1.1 การนับข้อมูลนำเข้าจากภายนอก (External Input)

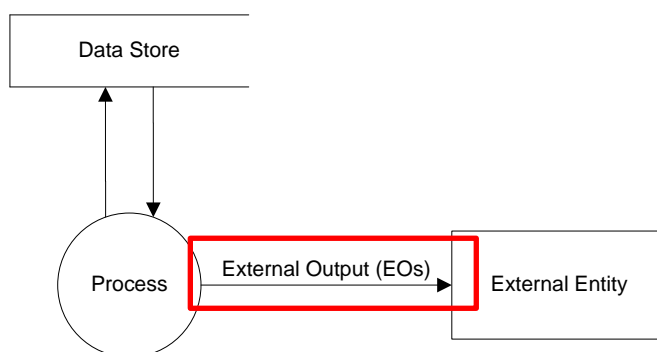
การนับข้อมูลนำเข้าจากภายนอกจากแผนภาพกระแสข้อมูลนั้นสามารถดูได้จากแผนภาพกระแสข้อมูลระดับสอง โดยพิจารณาจากข้อมูลที่ถูกส่งมาจากเอนทิตีภายนอก (External Entity) ไปยังกระบวนการ (Process) เช่นตัวอย่างในรูปที่ 15 ข้อมูลนำเข้าจากในภาพนี้ สามารถดูได้จากเส้นสัญลักษณ์กระแสข้อมูลที่มาจากเอนทิตีภายนอกเข้ามายังกระบวนการทำงาน การประเมินค่าน้ำหนักของข้อมูลนำเข้าจากภายนอกระบบนั้นสามารถดูรายละเอียดของข้อมูลนำเข้าจากภายนอกและนับจำนวนเขตข้อมูล DET (Data Element Type) ได้จากข้อมูลที่ถูกบันทึกไว้ในพจนานุกรมข้อมูล (Data Dictionary) และนับจำนวนแฟ้มข้อมูลอ้างอิง FTR (File Type Reference) ได้จากแฟ้มข้อมูลที่ถูกปรับปรุงเปลี่ยนแปลงจากกระบวนการทำงานนี้



รูปที่ 15 ส่วนที่เป็นข้อมูลนำเข้าจากภายนอกในระบบในแผนภาพกระแสข้อมูล

#### 2.4.1.2 การนับข้อมูลส่งออกภายนอก (External Output)

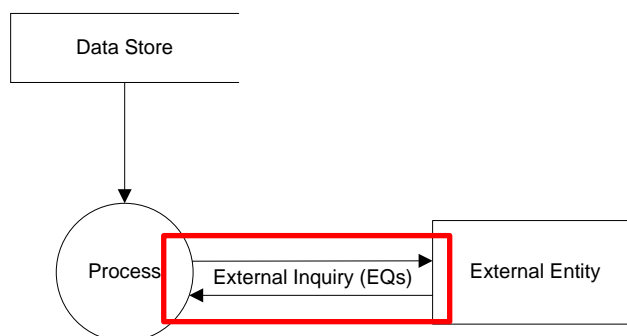
การนับข้อมูลส่งออกภายนอกจากแผนภาพกระแสข้อมูลนั้นสามารถดูได้จากแผนภาพกระแสข้อมูลระดับสอง โดยพิจารณาจากข้อมูลที่ถูกส่งจากกระบวนการ (Process) ออกไปยังเอนทิตีภายนอก (External Entity) เช่นตัวอย่างในรูปที่ 16 ข้อมูลส่งออกภายนอกจากในภาพนี้ สามารถดูได้จากเส้นสัญลักษณ์กระแสข้อมูลที่ออกจากกระบวนการทำงานไปยังเอนทิตีภายนอก การประเมินค่าน้ำหนักของข้อมูลส่งออกภายนอกนั้นสามารถดูรายละเอียดของข้อมูลนำเข้าจากภายนอกและนับจำนวนเขตข้อมูล DET (Data Element Type) ได้จากข้อมูลที่ถูกบันทึกไว้ในพจนานุกรมข้อมูล (Data Dictionary) และนับจำนวนแฟ้มข้อมูลอ้างอิง FTR (File Type Reference) ได้จากแฟ้มข้อมูลที่ถูกปรับปรุงเปลี่ยนแปลงจากกระบวนการทำงานนี้ โดยในมุมมองของ Rask (1991) นั้น ข้อมูลส่งออกภายนอกจะต้องมีลักษณะตรงกับข้อมูลนำเข้าบางข้อมูลในแผนภาพกระแสข้อมูล



รูปที่ 16 ส่วนที่เป็นข้อมูลส่งออกภายนอกจากแผนภาพกระแสข้อมูล

#### 2.4.1.3 การนับการสอบถามภายนอก (External Inquiries)

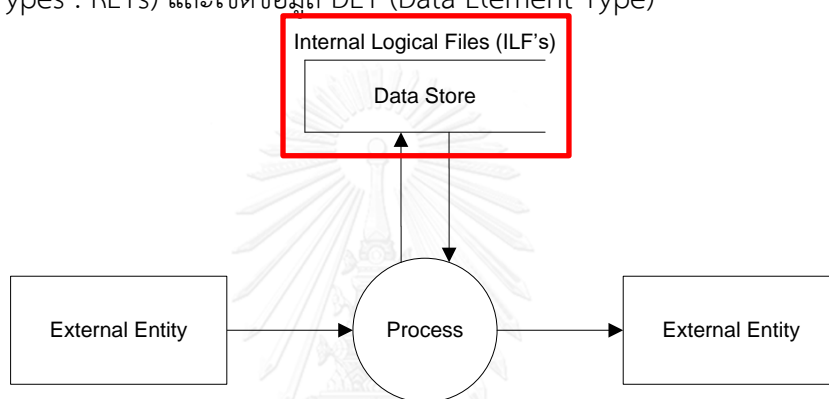
การนับการสอบถามภายนอกจะนับจากเส้นสัญลักษณ์กระแสข้อมูล สามารถดูได้จากแผนภาพกระแสข้อมูลระดับสอง โดยพิจารณาจากเส้นสัญลักษณ์กระแสข้อมูลที่ส่งเข้าและออกระหว่างเอนทิตีภายนอก (External Entity) เดียวกัน และกระบวนการ (Process) เดียวกัน และการสอบถามภายนอกนั้นยังต้องดูว่าข้อมูลนำเข้าหรือข้อมูลส่งออกนั้นไม่มีการปรับปรุงข้อมูลในแฟ้มข้อมูล ดังตัวอย่างในรูปที่ 17 จะเห็นได้ว่าการสอบถามจากภายนอกนั้นมีสัญลักษณ์ในแผนภาพกระแสข้อมูลที่ใกล้เคียงกับข้อมูลนำเข้าจากภายนอก (External Input) และข้อมูลส่งออกภายนอก (External Output) จะแตกต่างกันตรงที่การทำงานของ การสอบถามจากภายนอกนั้นจะไม่มีการบันทึกข้อมูลเข้าสู่แฟ้มข้อมูล (Data Store) กล่าวคือไม่มีเส้นสัญลักษณ์กระแสข้อมูลไปยังแฟ้มข้อมูล มีเพียงการอ้างอิงหรืออ่านข้อมูลจากแฟ้มข้อมูลเท่านั้น การประเมินค่าน้ำหนักของข้อมูลส่งออกภายนอกนั้นสามารถดูรายละเอียดของข้อมูลนำเข้าจากภายนอกและนับจำนวนเขตข้อมูล DET (Data Element Type) ได้จากข้อมูลที่ถูกบันทึกไว้ในพจนานุกรมข้อมูล (Data Dictionary) และนับจำนวนแฟ้มข้อมูลอ้างอิง FTR (File Type Reference) ได้จากแฟ้มข้อมูลที่ถูกปรับปรุงเปลี่ยนแปลงจากกระบวนการทำงานนี้



รูปที่ 17 ส่วนที่เป็นการสอบถามจากภายนอก

#### 2.4.1.4 การนับแฟ้มข้อมูลภายใน (Internal Logical Files)

แฟ้มข้อมูลภายในสามารถดูได้จากสัญลักษณ์ของหน่วยเก็บข้อมูลในแผนภาพกระแสข้อมูล โดยแฟ้มข้อมูล(Data Store) จะมีเส้นกระแสข้อมูลไหลระหว่างกระบวนการทำงาน (Process) กับแฟ้มข้อมูลอยู่ ดังตัวอย่างในรูปที่ 18 การประเมินค่าน้ำหนักของแฟ้มข้อมูลภายในนั้นจะพิจารณาจากข้อมูลที่อยู่ในเส้นกระแสข้อมูลที่ไหลระหว่างกระบวนการทำงานกับแฟ้มข้อมูล และสิ่งที่จะทำให้เข้าใจความสัมพันธ์ระหว่างหน่วยเก็บข้อมูลในแผนภาพกระแสข้อมูลมากขึ้น คือการใช้แผนภาพความสัมพันธ์ระหว่างเอนทิตีของข้อมูล(Entity Relationship Diagram) มาช่วยในการวิเคราะห์ว่าหน่วยเก็บข้อมูลมีความสัมพันธ์กันอย่างไร เพื่อมาช่วยในการนับค่าตารางข้อมูล RET (Record Element Types : RETs) และเขตข้อมูล DET (Data Element Type)



รูปที่ 18 ส่วนที่เป็นแฟ้มข้อมูลภายในและสัญลักษณ์ของการรับส่งข้อมูลเข้าสู่แฟ้มข้อมูล

#### 2.4.1.5 การนับแฟ้มข้อมูลภายนอก (External Interface File)

การใช้แผนภาพกระแสข้อมูลรูปแบบมาตรฐานมาวิเคราะห์หาแฟ้มข้อมูลต่อประสานภายนอกนั้นไม่สามารถทำได้ ผู้จัดทำแผนภาพกระแสข้อมูลอาจต้องกำหนดสัญลักษณ์พิเศษให้กับเอนทิตีภายนอกที่เป็นระบบ หรืออาจทำได้โดยการตั้งชื่อที่ทำให้ทราบว่าเอนทิตีภายนอกนั้นเป็นระบบ เช่น กำหนดให้เอนทิตีภายนอกที่เป็นระบบนั้นต้องมีคำว่า “ระบบ” ขึ้นต้นไว้ เป็นต้น การประเมินค่าน้ำหนักของแฟ้มข้อมูลภายนอกนั้นจะพิจารณาจากข้อมูลว่ามีการนำข้อมูลใดมาใช้มาจากแฟ้มข้อมูลภายนอกบ้าง

อีกจุดหนึ่งที่เราสามารถสังเกตว่าเป็นข้อมูลจากแฟ้มข้อมูลภายนอกได้ คือดูจากกระแสข้อมูลนำเข้า หรือข้อมูลส่งออก ว่ามีกระแสข้อมูลใดที่มีรูปแบบของข้อมูลหรือชื่อของข้อมูลที่แตกต่างจากข้อมูลที่ถูกเก็บไว้ภายในระบบ หรือข้อมูลใดมีการเปลี่ยนแปลงรูปแบบและถูกส่งออกไปยังนอกระบบ เราอาจอนุมานได้ว่าเป็นข้อมูลของแฟ้มข้อมูลนอกระบบเมื่อสามารถระบุประเภทของฟังก์ชันและระดับความซับซ้อนของฟังก์ชันนั้นแล้ว ก็จะนำข้อมูลเหล่านี้มาใช้ในการคำนวณค่าของฟังก์ชันพอยต์ตามหลักการวิเคราะห์ฟังก์ชันพอยต์ ตามรูปที่ 3 แสดงรูปตารางการให้ค่าน้ำหนักของแต่ละประเภทฟังก์ชันทั้ง 5 ประเภท

### บทที่ 3

#### ระเบียบวิธีวิจัย

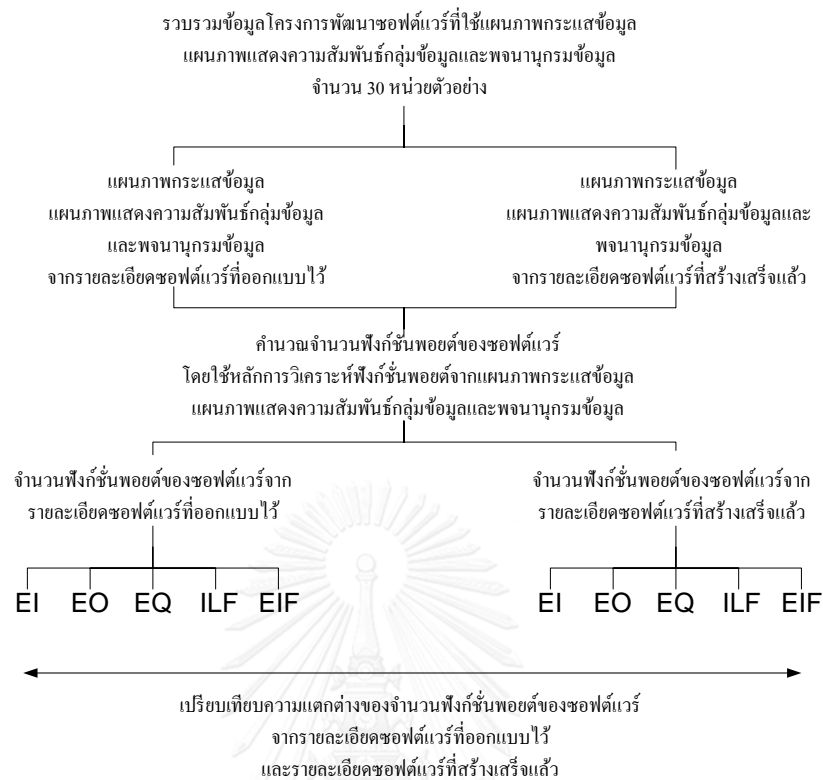
บทนี้จะแสดงถึงแนวทางในการตอบวัตถุประสงค์ของงานวิจัย โดยจะประกอบด้วยแบบแผนการดำเนินงาน ประชากรและหน่วยตัวอย่าง เครื่องมือที่ใช้ในการศึกษา การเก็บเรียบเรียงข้อมูล และการวิเคราะห์ข้อมูล

#### 3.1 แนวทางการวิจัย

งานวิจัยชิ้นนี้มีจุดประสงค์เพื่อต้องการเปรียบเทียบขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดซอฟต์แวร์ที่ออกแบบไว้เปรียบเทียบกับขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว โดยใช้วิธีการวิเคราะห์ฟังก์ชันพอยต์จากแผนภาพข้อมูลกระแส แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล และพจนานุกรมข้อมูล ที่ปรากฏในรายละเอียดซอฟต์แวร์ซึ่งมีแผนการดำเนินงานทั้งหมด 3 ขั้นตอนดังนี้

1. เก็บรวบรวมข้อมูลของโครงการพัฒนาซอฟต์แวร์ที่ใช้แผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูลและพจนานุกรมข้อมูล
2. คำนวณจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชัน จากข้อมูลแผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล และพจนานุกรมข้อมูล
3. เปรียบเทียบจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชันของการประมาณซอฟต์แวร์ ดังแผนกระบวนการดำเนินงานวิจัย ดังรูปที่ 19 แสดงแผนกระบวนการดำเนินงานวิจัย





รูปที่ 19 แสดงแผนกระบวนการดำเนินงานวิจัย

### 3.1.1 ขั้นตอนที่ 1 เก็บรวบรวมข้อมูลของโครงการพัฒนาซอฟต์แวร์ที่ใช้แผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูลและพจนานุกรมข้อมูล

ขั้นตอนนี้เป็นการรวบรวมข้อมูลโครงการพัฒนาซอฟต์แวร์ โดยเลือกโครงการพัฒนาซอฟต์แวร์ที่ใช้แผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล และพจนานุกรมข้อมูลที่ปรากฏอยู่ในรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ และอยู่ในรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว โดยข้อมูลที่ได้จะถูกนำมาใช้ในการหาขนาดของซอฟต์แวร์ ด้วยการวิเคราะห์ฟังก์ชันพอยต์จะได้ค่าจำนวนฟังก์ชันพอยต์ที่ยังไม่ปรับค่า (Unadjusted Function) เพื่อใช้ในการเปรียบเทียบขนาดความแตกต่างของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว โดยจะเก็บจากโครงการพัฒนาซอฟต์แวร์ทั้งหมด 30 หน่วยตัวอย่าง โดยแผนภาพกระแสข้อมูลที่จะนำมาใช้ในการวิเคราะห์นั้นจะอยู่ในรูปแบบนามสกุลไฟล์ XML แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูลอยู่ในรูปแบบนามสกุลไฟล์ .docx และพจนานุกรมข้อมูลอยู่ในรูปแบบนามสกุลไฟล์ .docx โดยจะเตรียมไฟล์สำหรับการหาขนาดของซอฟต์แวร์ในขั้นตอนต่อไป

### 3.1.2 ขั้นตอนที่ 2 คำนวณจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชัน จากข้อมูลแผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล และพจนานุกรมข้อมูล

ขั้นตอนนี้จะเป็นการแปลงข้อมูลที่ได้จากแผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์ กลุ่มข้อมูล และพจนานุกรมข้อมูล เป็นจำนวนฟังก์ชันพอยต์ที่ยังไม่ปรับค่าแยกตามแต่ละประเภทฟังก์ชัน เพื่อใช้ในการคำนวณขนาดของซอฟต์แวร์ โดยใช้หลักการวิเคราะห์ฟังก์ชันพอยต์ ซึ่งจะนำข้อมูลดังกล่าวที่ได้จากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้ และรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว จำนวน 2 ชุดข้อมูลจากโครงการพัฒนาซอฟต์แวร์เดียวกัน มาคำนวณโดยใช้เครื่องมือที่สร้างขึ้นมาเพื่อใช้ในการศึกษาวิจัยครั้งนี้ โดยเครื่องมือนี้จะนำหลักการวัดจำนวนฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลของ Rask (1991) มาประยุกต์ใช้ในการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์มาประยุกต์ใช้เป็นหลักในการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ โดยเกณฑ์ที่จะใช้ในการสร้างซอฟต์แวร์เพื่อคำนวณจำนวนฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล และพจนานุกรมข้อมูล มีดังนี้

1. ตรวจสอบชื่อของหน่วยเก็บข้อมูลในแผนภาพกระแสข้อมูล ชื่อตารางข้อมูลแต่ละตารางในพจนานุกรมข้อมูล (Data Dictionary) และชื่อของเอนทิตีในแผนภาพแสดงความสัมพันธ์ระหว่างเอนทิตีข้อมูล โดยบันทึกชื่อเก็บไว้ให้ตรงกัน
2. ระบุส่วนที่เป็นแฟ้มข้อมูลภายใน (ILF's) โดยตรวจสอบจากชื่อของหน่วยเก็บข้อมูลในแผนภาพกระแสข้อมูล และนำไปเปรียบเทียบกับชื่อของเอนทิตีในแผนภาพแสดงความสัมพันธ์ระหว่างเอนทิตีข้อมูล เพื่อดูความสัมพันธ์ระหว่างเอนทิตีข้อมูล และระบุส่วนที่เป็นแฟ้มข้อมูลภายใน และประเมินค่าความซับซ้อนของแฟ้มข้อมูลภายในระบบตามหลักของ IFPUG (International Function Points User Group, 1999) โดยนับจำนวน Record Element Type จากจำนวนเอนทิตีที่มีความสัมพันธ์กันในแฟ้มข้อมูลภายในระบบนั้น และนับจำนวน Data Element Type จากจำนวนของ Attribute ที่บันทึกไว้ใน Data Dictionary

ตารางที่ 7 ตารางค่าการให้น้ำหนักความซับซ้อนของแฟ้มข้อมูลภายใน

จำนวนตารางข้อมูล RET's	จำนวนเขตข้อมูล DET's		
	1 - 19	20 - 50	มากกว่า 50
1	ต่ำ (7)	ต่ำ (7)	ปานกลาง (10)
2 - 5	ต่ำ (7)	ปานกลาง (10)	สูง (15)
มากกว่า 5	ปานกลาง (10)	สูง (15)	สูง (15)

3. ระบุส่วนที่เป็นข้อมูลนำเข้าจากภายนอก (EIs) จากเส้นสัญลักษณ์ของกระแสข้อมูลที่ไหลจากเอนทิตีภายนอกเข้ามายังกระบวนการในระบบ และต้องมีสัญลักษณ์ของกระแสข้อมูลที่ไหลจากกระบวนการเข้าสู่เพิ่มข้อมูลภายในระบบด้วย และประเมินค่าความซับซ้อนของข้อมูลนำเข้าจากภายนอกตามหลักของIFPUG (International Function Points User Group, 1999) โดยการนำชื่อของกระแสข้อมูลนั้นมาเปรียบเทียบกับชื่อของหน่วยเก็บข้อมูลในแผนภาพกระแสข้อมูล และนับจำนวนData Element Type จากจำนวน Attribute ที่อยู่ใน Data Dictionary ที่มีชื่อตรงกับหน่วยเก็บข้อมูล นับจำนวน File Type Reference จากจำนวนเพิ่มข้อมูลที่มีข้อมูลใดข้อมูลหนึ่งเป็นชื่อข้อมูลเดียวกับข้อมูลบนเส้นกระแสข้อมูล

ตารางที่ 8 ตารางค่าการให้น้ำหนักความซับซ้อนของข้อมูลนำเข้าจากภายนอก

จำนวนตารางข้อมูล FTR's	จำนวนเขตข้อมูล DET's		
	1 - 4	5 - 15	มากกว่า 15
น้อยกว่า 2	ต่ำ (3)	ต่ำ (3)	ปานกลาง (4)
2	ต่ำ (3)	ปานกลาง (4)	สูง (6)
มากกว่า 2	ปานกลาง (4)	สูง (6)	สูง (6)

4. ระบุส่วนที่เป็นข้อมูลส่งออกภายนอก (EOs) จากเส้นสัญลักษณ์ของกระแสข้อมูลที่ไหลจากกระบวนการในระบบออกไปสู่เอนทิตีภายนอก และต้องมีสัญลักษณ์ของกระแสข้อมูลที่ไหลจากกระบวนการเข้าสู่เพิ่มข้อมูลภายในระบบด้วยและประเมินค่าความซับซ้อนของข้อมูลส่งออกภายนอกตามหลักของ IFPUG (International Function Points User Group, 1999) โดยการนำชื่อของกระแสข้อมูลนั้นมาเปรียบเทียบกับชื่อของหน่วยเก็บข้อมูลในแผนภาพกระแสข้อมูล และนับจำนวน Data Element Type จากจำนวน Attribute ที่อยู่ใน Data Dictionary ที่มีชื่อตรงกับหน่วยเก็บข้อมูล นับจำนวน File Type Reference จากจำนวนเพิ่มข้อมูลที่มีข้อมูลใดข้อมูลหนึ่งเป็นชื่อข้อมูลเดียวกับข้อมูลบนเส้นกระแสข้อมูล

ตารางที่ 9 ตารางค่าการให้น้ำหนักความซับซ้อนของข้อมูลส่งออกภายนอก

จำนวนตารางข้อมูล FTR's	จำนวนเขตข้อมูล DET's		
	1 - 5	6 - 19	มากกว่า 19
น้อยกว่า 2	ต่ำ (4)	ต่ำ (4)	ปานกลาง (5)
2 - 3	ต่ำ (4)	ปานกลาง (5)	สูง (7)
มากกว่า 3	ปานกลาง (5)	สูง (7)	สูง (7)

5. ระบุส่วนที่เป็นข้อมูลการสอบถามจากภายนอก (EQs) คือเส้นสัญลักษณ์กระแสข้อมูลที่มีทิศทางไหลระหว่างเอนทิตีภายนอกกับกระบวนการ(ไหลเข้า หรือไหลออกก็ได้) แต่ต้องไม่มีเส้นกระแสข้อมูลที่ไหลจากกระบวนการเข้าสู่หน่วยเก็บข้อมูล และประเมินค่าความซับซ้อนของข้อมูลการสอบถามจากภายนอกหรือการแสดงผลตามหลักของIFPUG (International Function Points User Group, 1999) โดยการนำชื่อของกระแสข้อมูลนั้นมาเปรียบเทียบกับชื่อของหน่วยเก็บข้อมูลในแผนภาพกระแสข้อมูล และนับจำนวน Data Element Type จากจำนวน Attribute ที่อยู่ใน Data Dictionary ที่มีชื่อตรงกับหน่วยเก็บข้อมูล นับจำนวน File Type Reference จากจำนวนแฟ้มข้อมูลที่มีข้อมูลใดข้อมูลหนึ่งเป็นชื่อข้อมูลเดียวกับข้อมูลบนเส้นกระแสข้อมูล

ตารางที่ 10 ตารางค่าการให้น้ำหนักความซับซ้อนของข้อมูลการสอบถามจากภายนอก

จำนวนตารางข้อมูล FTR's	จำนวนเขตข้อมูล DET's		
	1 - 5	6 - 19	มากกว่า 19
น้อยกว่า 2	ต่ำ (3)	ต่ำ (3)	ปานกลาง (4)
2 - 3	ต่ำ (3)	ปานกลาง (4)	สูง (6)
มากกว่า 3	ปานกลาง (4)	สูง (6)	สูง (6)

6. ระบุส่วนที่เป็นข้อมูลภายนอกระบบ (External Interface file : EIF) คือเส้นสัญลักษณ์ กระแสข้อมูลที่มีทิศทางไหลระหว่างเอนทิตีภายนอกกับกระบวนการ(ไหลเข้า หรือ ไหลออกก็ได้) แต่ต้องไม่มีเส้นกระแสข้อมูลที่ไหลจากกระบวนการเข้าสู่หน่วยเก็บข้อมูล และประเมินค่าความซับซ้อนของข้อมูลการสอบถามจากภายนอกหรือการแสดงผลตามหลักของIFPUG (International Function Points User Group, 1999) โดยการนำชื่อของกระแสข้อมูลนั้นมาเปรียบเทียบกับชื่อของหน่วยเก็บข้อมูลในแผนภาพกระแสข้อมูล โดยนับจำนวน Record Element Type จากจำนวนเอนทิตีที่มีความสัมพันธ์กันในแฟ้มข้อมูลภายในระบบนั้น และนับจำนวน Data Element Type จากจำนวนของแอททริบิวต์ที่บันทึกไว้ในพจนานุกรมข้อมูล

ตารางที่ 11 ตารางค่าการให้น้ำหนักความซับซ้อนของข้อมูลภายนอก

จำนวนตารางข้อมูล RET's	จำนวนเขตข้อมูล DET's		
	1 - 19	20 - 50	มากกว่า 50
1	ต่ำ (5)	ต่ำ (5)	ปานกลาง (7)
2 - 5	ต่ำ (5)	ปานกลาง (7)	สูง (10)
มากกว่า 5	ปานกลาง (7)	สูง (10)	สูง (10)

เมื่อสามารถกำหนดประเภทฟังก์ชัน ประเมินค่าความซับซ้อนของข้อมูล และประเมินค่าน้ำหนักของฟังก์ชันได้แล้ว ก็สามารถนำข้อมูลทั้งหมดมาหาค่าของจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ จำนวนฟังก์ชันพอยต์ที่เป็นข้อมูลนำเข้าจากภายนอก(External Input : EI) จำนวนฟังก์ชันพอยต์ที่เป็นข้อมูลส่งออกภายนอก(External Output : EO) จำนวนฟังก์ชันพอยต์ที่เป็นข้อมูลการสอบถามจากภายนอกหรือการแสดงผล(External Inquiry : EQ) จำนวนฟังก์ชันพอยต์ที่เป็นแฟ้มข้อมูลภายในระบบ(Internal Logical files : ILF) และจำนวนฟังก์ชันพอยต์ที่เป็นแฟ้มข้อมูลภายนอก (External Interface file : EIF) ได้ โดยการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point : UAF) จะมีการคำนวณจำนวนฟังก์ชันพอยต์ดังตารางที่ 12 แสดงตารางการคำนวณค่าฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า ดังนี้

ตารางที่ 12 การคำนวณค่าฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า

ประเภทฟังก์ชัน (Function Type)	ระดับความซับซ้อน			ผลรวม
	ต่ำ	ปานกลาง	สูง	
ข้อมูลนำเข้าภายนอก (External Input Files : EIs)	..... X 3	..... X 4	..... X 6	
ข้อมูลส่งออกภายนอก (External Outputs : EOs)	..... X 4	..... X 5	..... X 7	
การสอบถามภายนอก (External Inquiries : EQs)	..... X 3	..... X 4	..... X 6	
เพิ่มข้อมูลภายใน (Internal Logical Files : ILF's)	..... X 7	..... X 10	..... X 15	
เพิ่มข้อมูลต่อประสานภายนอก (External Interface Files : EIF's)	..... X 5	..... X 7	..... X 10	
ค่าฟังก์ชันพอยต์ที่ยังไม่ได้ปรับค่า (Unadjusted Function Point)				

นำข้อมูลจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point : UAF) ที่ได้จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ กับรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว มาเปรียบเทียบกัน ซึ่งจะปรากฏข้อมูลจำนวน 2 ชุด ดังตารางผลการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า ดังตารางที่ 13 แสดงตารางผลของการคำนวณค่าฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า ดังนี้

ตารางที่ 13 แสดงตารางผลของการคำนวณค่าฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า

ผลการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point) ของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้	ผลการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point) ของรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว
1. จำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าจากภายนอก (External Input : EI)	1. จำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าจากภายนอก (External Input : EI)
2. จำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (External Output : EO)	2. จำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (External Output : EO)
3. จำนวนฟังก์ชันพอยต์ของเพิ่มข้อมูล	3. จำนวนฟังก์ชันพอยต์ของเพิ่มข้อมูล

ผลการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point) ของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้	ผลการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point) ของรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว
สอบถามภายนอกหรือการแสดงผล (External Inquiry : EQ)	สอบถามภายนอกหรือการแสดงผล (External Inquiry : EQ)
4. จำนวนฟังก์ชันพอยต์ของข้อมูลเพิ่มภายในระบบ (Internal Logical files : ILF)	4. จำนวนฟังก์ชันพอยต์ของข้อมูลเพิ่มภายในระบบ (Internal Logical files : ILF)
5. จำนวนฟังก์ชันพอยต์ของข้อมูลภายนอกในระบบ (External Interface file : EIF)	5. จำนวนฟังก์ชันพอยต์ของข้อมูลภายนอกในระบบ (External Interface file : EIF)
6. จำนวนผลรวมจำนวนฟังก์ชันพอยต์ทั้งหมดของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point : UAF)	6. จำนวนผลรวมจำนวนฟังก์ชันพอยต์ทั้งหมดของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point : UAF)

สำหรับงานวิจัยชิ้นนี้จะใช้ค่าการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point : UAF) มาเพื่อเปรียบเทียบหาขนาดความแตกต่างกันของซอฟต์แวร์ที่อยู่ในรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้เทียบกับขนาดของซอฟต์แวร์ที่อยู่ในรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว

### 3.1.3 ขั้นตอนที่ 3 เปรียบเทียบจำนวนฟังก์ชันพอยต์แยกตามประเภทฟังก์ชันของการประมาณซอฟต์แวร์

ขั้นตอนนี้จะเป็นการนำข้อมูลที่ได้จากการวิเคราะห์ฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่ามาวิเคราะห์ เพื่อตอบวัตถุประสงค์ของการศึกษาวิจัยครั้งนี้ ที่ต้องการทราบว่าขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้นั้นน้อยกว่าขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดซอฟต์แวร์ที่สร้างเสร็จหรือไม่โดยการเปรียบเทียบจำนวนฟังก์ชันพอยต์แยกตามแต่ละประเภทฟังก์ชัน และประเภทฟังก์ชันพอยต์ประเภทใดจากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว ที่มีผลทำให้เกิดความแตกต่างกันของรายละเอียดของซอฟต์แวร์ทั้งสองแบบ โดยจะนำข้อมูลผลการวิเคราะห์ฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่าทั้งหมดมาเปรียบเทียบดังต่อไปนี้

- เปรียบเทียบจำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าจากภายนอก (External Input : EI) ของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว

- เปรียบเทียบจำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (External Output : EO) ของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว
- เปรียบเทียบจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลสอบถามภายนอกหรือการแสดงผล (External Inquiry : EQ) ของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว
- เปรียบเทียบจำนวนฟังก์ชันพอยต์ของข้อมูลแฟ้มภายในระบบ (Internal Logical files : ILF) ของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว
- เปรียบเทียบจำนวนฟังก์ชันพอยต์ของข้อมูลภายนอกในระบบ (External Interface file : EIF) ของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว
- เปรียบเทียบจำนวนผลรวมจำนวนฟังก์ชันพอยต์ทั้งหมดของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า (Unadjusted Function Point : UAF) ของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว

โดยจะบันทึกข้อมูลการเปรียบเทียบลงตารางบันทึกผลกาวิเคราะห์ฟังก์ชันพอยต์ของโครงการพัฒนาซอฟต์แวร์ดังตารางที่ 14 แสดงตารางบันทึกผลของการคำนวณค่าฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า ดังนี้

ตารางที่ 14 ตารางบันทึกผลของการคำนวณค่าฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า

ชื่อโครงการการพัฒนาซอฟต์แวร์ :			
ประเภทงานของฟังก์ชันพอยต์	ผลการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่าของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้	ผลการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่าของรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว	ค่าความแตกต่างระหว่างจำนวนฟังก์ชันพอยต์
จำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าจากภายนอก (External Input : EI)			
จำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (External Output : EO)			
จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลสอบถามภายนอกหรือการแสดงผล			



(External Inquiry : EQ)			
จำนวนฟังก์ชันพอยต์ของ ข้อมูลเพิ่มภายในระบบ (Internal Logical files : ILF)			
จำนวนฟังก์ชันพอยต์ของ ข้อมูลภายนอกในระบบ (External Interface file : EIF)			
จำนวนผลรวมจำนวน ฟังก์ชันพอยต์ทั้งหมดของ ซอฟต์แวร์ที่ยังไม่ถูกปรับ ค่า (Unadjusted Function Point : UAF)			

### 3.2 ประชากรและหน่วยตัวอย่าง

#### 3.2.1 ประชากร

ประชากรของงานวิจัยนี้ คือ โครงการพัฒนาซอฟต์แวร์ที่พัฒนาขึ้นใหม่ซึ่งลักษณะของประชากรจะต้องมีคุณสมบัติดังนี้

- เป็นโครงการพัฒนาซอฟต์แวร์ที่ใช้แผนภาพกระแสข้อมูล (Data Flow Diagrams) แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล (ER-Diagrams) และพจนานุกรมข้อมูล (Data Dictionary) ในการพัฒนาซอฟต์แวร์
- เป็นโครงการพัฒนาซอฟต์แวร์ที่มีรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ (Software Design Specifications) และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว (Product Specifications)

#### 3.2.2 กลุ่มตัวอย่าง

งานวิจัยชิ้นนี้ได้ใช้ข้อมูลโครงการพัฒนาซอฟต์แวร์ โดยเก็บข้อมูลมาจาก

- เป็นซอฟต์แวร์ของนิสิตระดับบัณฑิตศึกษา ปริญญาโท สาขาการพัฒนาซอฟต์แวร์ ด้านธุรกิจ ภาควิชาสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย ตั้งแต่ปีการศึกษา 2552- 2558
- เป็นซอฟต์แวร์ที่จัดทำในวิชาโครงการพิเศษ 1 และโครงการพิเศษ 2 เป็นบุคคลเดียวกัน

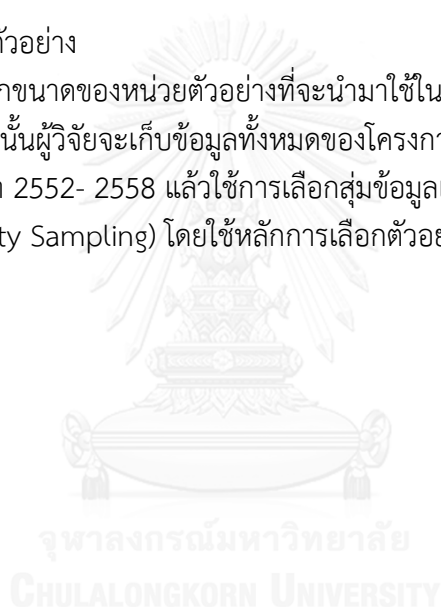
- ข้อมูลรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้จึงสามารถเก็บได้จากโครงการพิเศษ 1 และข้อมูลรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้วจะสามารถเก็บได้จากโครงการพิเศษ 2

### 3.2.3 ขนาดตัวอย่าง

โครงการพิเศษ1 และโครงการพิเศษ 2 ตั้งแต่ปีการศึกษา 2552 - 2558 โดยผู้วิจัยจะเลือกใช้หน่วยตัวอย่างเพียง 30 หน่วยตัวอย่างเพื่อทำการศึกษาวิจัยเปรียบเทียบจำนวนฟังก์ชันพอยต์ที่ปรากฏอยู่ในรายละเอียดซอฟต์แวร์ที่ออกแบบไว้ โดยใช้ข้อมูลโครงการพิเศษ 1 และจำนวนฟังก์ชันพอยต์ที่ปรากฏอยู่ในรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว โดยใช้ข้อมูลโครงการพิเศษ 2

### 3.2.4 การสุ่มตัวอย่าง

เนื่องจากขนาดของหน่วยตัวอย่างที่จะนำมาใช้ในงานวิจัยนี้มีจำนวนมากกว่า 30 หน่วยตัวอย่างดังนั้นผู้วิจัยจะเก็บข้อมูลทั้งหมดของโครงการพิเศษ 1 และโครงการพิเศษ 2 ตั้งแต่ปีการศึกษา 2552- 2558 แล้วใช้การเลือกสุ่มข้อมูลแบบไม่อาศัยความน่าจะเป็น (Non-Probability Sampling) โดยใช้หลักการเลือกตัวอย่างแบบเฉพาะเจาะจง (Purposive Sampling)



### 3.3 เครื่องมือที่ใช้ในการศึกษา

#### 1. ซอฟต์แวร์ที่ใช้ในการวิเคราะห์ฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล และพจนานุกรมข้อมูล

ซอฟต์แวร์ที่สร้างขึ้นมาเพื่อใช้เป็นเครื่องมือสำหรับการศึกษาวิจัยโดยใช้หลักการวิเคราะห์แบบฟังก์ชันพอยต์ โดยนำมาใช้ในการคำนวณเพื่อป้องกันการผิดพลาดที่เกิดขึ้นจากการนับของคน และเป็น การสร้างมาตรฐานในการคำนวณในการวิเคราะห์ข้อมูลเพื่อให้ได้ข้อมูลที่เป็นมาตรฐานเดียวกัน สำหรับการคำนวณวิเคราะห์ฟังก์ชันพอยต์แบบเดียวกัน การใช้เครื่องมือในการช่วยวิเคราะห์จะทำให้ การแปลงข้อมูลมีมาตรฐานที่ชัดเจน และแปลงข้อมูลจากทุกหน่วยตัวอย่างภายใต้กฎเกณฑ์และ มาตรฐานเดียวกันโดยการทำงานของซอฟต์แวร์นี้จะรับไฟล์ที่มีไฟล์รูปแบบ XML ของแผนภาพ กระแสข้อมูล และแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลที่ถูกสร้างขึ้นด้วยซอฟต์แวร์ที่ใช้ในการ สร้างแบบจำลอง เช่น ซอฟต์แวร์ Microsoft Visio หรือซอฟต์แวร์ Power Designer สามารถบันทึก ไฟล์ของแผนภาพกระแสข้อมูลและแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลให้เป็นไฟล์รูปแบบ XML โดยเครื่องมือนี้จะนำหลักการวัดจำนวนฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลของ Rask (Rask, 1991) และหลักการวัดจำนวนฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลและแผนภาพแสดง ความสัมพันธ์ระหว่างเอนทิตีข้อมูลของ Lamma และคณะ (E. Lamma et al., 2003) มาประยุกต์ใช้ เป็นหลักในการคำนวณจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์

#### 2. ตารางบันทึกผลการวิเคราะห์ฟังก์ชันพอยต์ของโครงการพัฒนาซอฟต์แวร์

เครื่องมือชิ้นนี้จะถูกใช้ในขั้นตอนที่ 3 ของแผนการดำเนินงานวิจัย เพื่อใช้ในการบันทึกข้อมูล ที่ได้มาจากการวิเคราะห์ฟังก์ชันพอยต์ของซอฟต์แวร์ โดยใช้แผนภาพกระแสข้อมูลมาใช้ในการ วิเคราะห์หาค่าปริมาณงานของซอฟต์แวร์ ดังตารางที่ตารางที่ 14 แสดงตารางบันทึกผลของการ คำนวณค่าฟังก์ชันพอยต์ของซอฟต์แวร์ที่ยังไม่ถูกปรับค่า

### 3.4 สมมติฐานของงานวิจัย

สำหรับงานวิจัยชิ้นนี้ต้องการศึกษาเกี่ยวกับความแตกต่างของจำนวนฟังก์ชันพอยต์แยกตาม ประเภทฟังก์ชันของรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ เปรียบเทียบกับรายละเอียดซอฟต์แวร์ที่ สร้างเสร็จแล้ว โดยใช้หลักการประมาณขนาดซอฟต์แวร์ด้วยจำนวนฟังก์ชันพอยต์ โดยใช้แผนภาพ กระแสข้อมูล (Data Flow Diagrams) แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล (ER-Diagrams) และพจนานุกรมข้อมูล (Data Dictionary) มาใช้ในการคำนวณหาจำนวนฟังก์ชันพอยต์ของซอฟต์แวร์ เพื่อใช้ในการหาขนาดของซอฟต์แวร์

จากคำกล่าวของ Sommerville (Sommerville, 2001) ที่กล่าวไว้ว่า เมื่อการสร้างซอฟต์แวร์นั้น เสร็จสิ้น แบบของซอฟต์แวร์จึงมักเปลี่ยนไปจากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ที่เขียนไว้ใน เบื้องต้นมาก เอกสารของการออกแบบนั้นมีข้อผิดพลาดและมีการอธิบายรายละเอียดของซอฟต์แวร์ที่ ไม่สมบูรณ์ และจากแนวความคิดนี้จึงเป็นไปได้ว่าขนาดของซอฟต์แวร์ที่หาได้จากรายละเอียดของ

ซอฟต์แวร์ที่ออกแบบไว้นั้นจะน้อยกว่าขนาดของซอฟต์แวร์ที่ทำได้จากรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้วซึ่งได้ตั้งสมมติฐานตามหลักสถิติ (กัลยา วานิชย์บัญชา, 2551) ไว้ดังนี้

โดยแบ่งการทดสอบสมมติฐานออกเป็น 6 สมมติฐาน ดังนี้

**สมมติฐานที่ 1 :** จำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก (EIs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้นั้นน้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก (EIs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ที่ยังไม่ประเมินค่าความซับซ้อน

$$H_0 : FP(EIs)_{\text{หลัง} - \text{ก่อน}} \leq 0$$

$$H_1 : FP(EIs)_{\text{หลัง} - \text{ก่อน}} > 0$$

$H_0$  : จำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก (EIs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้มากกว่าหรือเท่ากับจำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก (EIs) ของรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

$H_1$  : จำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก (EIs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก (EIs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

โดยที่  $FP(EIs)_{\text{ก่อน}}$  คือจำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก (EIs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และ  $FP(EIs)_{\text{หลัง}}$  คือจำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก (EIs) รายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วตามลำดับ

**สมมติฐานที่ 2 :** จำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (EOs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้นั้นน้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (EOs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ที่ยังไม่ประเมินค่าความซับซ้อน

$$H_0 : FP(EOs)_{\text{ก่อน} - \text{หลัง}} \leq 0$$

$$H_1 : FP(EOs)_{\text{ก่อน} - \text{หลัง}} > 0$$

$H_0$  : จำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (EOs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้มากกว่าหรือเท่ากับจำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (EOs) ของรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

$H_1$  : จำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (EOs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (EOs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

โดยที่  $FP(EOs)_{\text{ก่อน}}$  คือจำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (EOs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และ  $FP(EOs)_{\text{หลัง}}$  คือจำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (EOs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วตามลำดับ

สมมติฐานที่ 3 : จำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (EQs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (EQs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ที่ยังไม่ประเมินค่าความซับซ้อน

$$H_0 : FP(EQs)_{\text{ก่อน}} - \text{หลัง} \leq 0$$

$$H_1 : FP(EQs)_{\text{ก่อน}} - \text{หลัง} > 0$$

$H_0$  : จำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (EQs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้มากกว่าหรือเท่ากับจำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (EQs) ของรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

$H_1$  : จำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (EQs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (EQs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

โดยที่  $FP(EQs)_{\text{ก่อน}}$  คือจำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (EQs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และ  $FP(EQs)_{\text{หลัง}}$  คือจำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (EQs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วตามลำดับ

สมมติฐานที่ 4 : จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (ILF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (ILF's) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ที่ยังไม่ประเมินค่าความซับซ้อน

$$H_0 : FP(ILF's)_{\text{ก่อน}} - \text{หลัง} \leq 0$$

$$H_1 : FP(ILF's)_{\text{ก่อน}} - \text{หลัง} > 0$$

$H_0$  : จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (ILF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้มากกว่าหรือเท่ากับจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (ILF's) ของรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

$H_1$  : จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (ILF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (ILF's) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

โดยที่  $FP(ILF's)_{\text{ก่อน}}$  คือจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (ILF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และ  $FP(ILF's)_{\text{หลัง}}$  คือจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (ILF's) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วตามลำดับ

**สมมติฐานที่ 5 :** จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (EIF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (EIF's) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ที่ยังไม่ประเมินค่าความซับซ้อน

$$H_0 : FP(EIF's)_{\text{ก่อน} - \text{หลัง}} \leq 0$$

$$H_1 : FP(EIF's)_{\text{ก่อน} - \text{หลัง}} > 0$$

$H_0$  : จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (EIF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้มากกว่าหรือเท่ากับจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (EIF's) ของรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

$H_1$  : จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (EIF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (EIF's) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

โดยที่  $FP(EIF's)_{\text{ก่อน}}$  คือจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (EIF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และ  $FP(EIF's)_{\text{หลัง}}$  คือจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (EIF's) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วตามลำดับ

**สมมติฐานที่ 6 :** จำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์ จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ที่ยังไม่ประเมินค่าความซับซ้อน

$$H_0 : FP(Totals)_{\text{ก่อน} - \text{หลัง}} \leq 0$$

$$H_1 : FP(Totals)_{\text{ก่อน} - \text{หลัง}} > 0$$

$H_0$  : จำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้มากกว่าหรือเท่ากับจำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ที่ยังไม่ประเมินค่าความซับซ้อน

$H_1$  : จำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ที่ยังไม่ประเมินค่าความซับซ้อน

โดยที่  $FP(Totals)_{\text{ก่อน}}$  คือจำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และ  $FP(Totals)_{\text{หลัง}}$  จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วตามลำดับ

## บทที่ 4

การออกแบบและพัฒนาเครื่องมือสำหรับการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล

ในบทนี้อธิบายวิธีการออกแบบและพัฒนาเครื่องมือสำหรับการนับค่าฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล ประกอบด้วย การวิเคราะห์การนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล และการคำนวณค่าฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล การออกแบบและการพัฒนาเครื่องมือและการทดสอบเครื่องมือที่พัฒนา

4.1 การวิเคราะห์การนับการประเมินความซับซ้อนและการแปลงฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลเป็นข้อมูลเข้า

การนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล โดยแยกตามแต่ละประเภทฟังก์ชันพอยต์และการคิดค่าฟังก์ชันพอยต์ โดยการประเมินความซับซ้อนของฟังก์ชันพอยต์ ดังต่อไปนี้

4.1.1 การนับฟังก์ชันพอยต์ โดยแยกตามแต่ละประเภทฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูล จะไม่มีกฎเกณฑ์ว่าควรมีระดับแผนภาพย่อย ซึ่งในหนึ่งแผนภาพกระแสข้อมูล โดยทั่วไปจะประกอบไปด้วย 1). แผนภาพบริบท (Context Diagram) 2). แผนภาพระดับหนึ่ง (Level-1 Diagram หรือ / Parent Diagram) 3). แผนภาพระดับลูก (Child Diagram) ซึ่งในระดับแผนภาพนี้หากมีรายละเอียดการทำงานย่อยจากระดับนี้อีก ก็แตกรายละเอียดลงไปจนกระทั่งสิ้นสุดกระบวนการ ส่วนชื่อของระดับก็จะเปลี่ยนไปเป็น Level-2 Diagram หรือ Level-3 Diagram ไปเรื่อยๆ จนกระทั่งจบกระบวนการทั้งหมด

โดยการนับจำนวนฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล โดยนับจำนวนฟังก์ชันพอยต์แยกตามแต่ละประเภทฟังก์ชันพอยต์ ตั้งแต่แผนภาพระดับที่หนึ่งย่อยลงไปจนถึงสิ้นสุดแผนภาพระดับลูก

- การนับข้อมูลนำเข้าจากภายนอก (External Input)

โดยพิจารณาจากข้อมูลที่ถูกส่งมาจากเอนทิตีภายนอก (External Entity) ไปยังกระบวนการ (Process)

- การนับข้อมูลส่งออกภายนอก (External Output)

โดยพิจารณาจากข้อมูลที่ถูกส่งจากกระบวนการ (Process) ออกไปยังเอนทิตีภายนอก (External Entity)

- การนับการสอบถามภายนอก (External Inquiries)

โดยพิจารณาจากเส้นสัญลักษณ์กระแสข้อมูลที่ส่งออกระหว่างเอนทิตีภายนอก (External Entity) เดียวกัน และกระบวนการ (Process) เดียวกัน โดยส่วนใหญ่จะใช้คำว่า “ข้อมูลผลการค้นหา.....” “รายงาน...” เป็นต้น

- การนับแฟ้มข้อมูลภายใน (Internal Logical Files)

แฟ้มข้อมูลภายในสามารถดูได้จากสัญลักษณ์ของหน่วยเก็บข้อมูลในแผนภาพกระแสข้อมูล เป็นแฟ้มข้อมูล(Data Store) เนื่องจากกฎเกณฑ์ในการสร้างแฟ้มข้อมูลในแผนภาพกระแสข้อมูล ถูกกำหนดว่าจะเป็นแฟ้มข้อมูลในระดับแผนภาพแรก ก็ต่อเมื่อมีการประมวลผลมากกว่าหนึ่ง กระบวนการใช้แฟ้มข้อมูลร่วมกัน ดังนั้นในระดับแผนภาพแรกอาจจะยังไม่ปรากฏแฟ้มข้อมูลบางอันได้ เนื่องจากอาจจะแฟ้มข้อมูลที่ใช้ร่วมกันภายในแต่ละกระบวนการ โดยจะไม่นับแฟ้มข้อมูลที่ซ้ำกับแฟ้มข้อมูลในระดับแรก

- การนับแฟ้มข้อมูลภายนอกระบบ (External Interface File)

การใช้แผนภาพกระแสข้อมูลรูปแบบมาตรฐานมาวิเคราะห์หาแฟ้มข้อมูลต่อประสานภายนอกนั้นไม่สามารถทำได้ โดยกำหนดให้เอนทิตีภายนอกที่เป็นระบบนั้นต้องมีคำว่า “ระบบ” “Google...” หรือ “เว็บไซต์...” ขึ้นต้นไว้ เป็นต้น

#### 4.1.2 การคำนวณการนับค่าฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล

การคำนวณการนับค่าฟังก์ชันพอยต์ จะใช้ค่าที่ได้จากการประเมินความซับซ้อนของฟังก์ชันแต่ละประเภทตามการประยุกต์หลักการข้างต้นนำมาใช้ โดยความซับซ้อนของฟังก์ชันจะถูกแบ่งเป็น 3 ระดับ คือ 1.) ระดับความซับซ้อนต่ำ (Low) 2.) ระดับความซับซ้อนปานกลาง (Average) และ 3.) ระดับความซับซ้อนสูง(High) โดยฟังก์ชันเชิงข้อมูลสามารถประเมินระดับความซับซ้อนได้จากจำนวนเขตข้อมูล (Data Element Types : DETs) และจำนวนตารางข้อมูล (Record Element Types : RETs) สำหรับฟังก์ชันเชิงรายการเปลี่ยนแปลงสามารถประเมินระดับความซับซ้อนได้จากจำนวนเขตข้อมูล (Data Element Types : DETs) และจำนวนแฟ้มข้อมูลอ้างอิง (Files Type Referenced : FTRs) (International Function Points User Group, 1999)

จากแผนภาพกระแสข้อมูลจะไม่สามารถระบุจำนวนเขตข้อมูล (Data Element Types : DETs) และจำนวนแฟ้มข้อมูลอ้างอิง (Files Type Referenced : FTRs) ของฟังก์ชันเชิงรายการเปลี่ยนแปลงได้อย่างแน่นอนตั้งแต่แผนภาพระดับที่หนึ่ง ดังนั้นหากฟังก์ชันเชิงรายการเปลี่ยนแปลงจะกำหนดจำนวนแฟ้มข้อมูลอ้างอิงแค่ 1 แฟ้มข้อมูลอ้างอิง โดยจะดูการอ้างอิงชื่อแฟ้มข้อมูลอ้างอิงจากเส้นข้อมูลที่เป็นฟังก์ชันเชิงรายการเปลี่ยนแปลงที่คล้ายกันหรือดูได้จากระดับแผนภาพย่อยลงไปจากระดับที่หนึ่ง หากไม่ปรากฏแผนภาพระดับย่อยของการประมวลผลจากแผนภาพระดับที่หนึ่งก็จะใช้วิธีการอ้างอิงจำนวนเขตข้อมูลและจำนวนแฟ้มข้อมูลอ้างอิงดังที่วิธีที่กล่าวข้างต้นของแผนภาพระดับที่หนึ่ง ส่วนของจำนวนเขตข้อมูล (Data Element Types : DETs) และจำนวนตารางข้อมูล (Record Element Types : RETs) ของฟังก์ชันเชิงข้อมูล การประเมินระดับความซับซ้อนของแฟ้มข้อมูลภายใน (Internal Logical Files) จะสามารถระบุจำนวนเขตข้อมูลและจำนวนตารางข้อมูลได้จากแผนภาพแสดงความสัมพันธ์กลุ่มข้อมูลและพจนานุกรมข้อมูล และส่วนของแฟ้มข้อมูลภายนอกเราไม่สามารถจำนวนของเขตข้อมูลและจำนวนตารางข้อมูลได้ จึงกำหนดให้เป็นระดับความซับซ้อนต่ำ (Low)









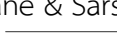





#### 4.1.3 วิธีการแปลงแผนภาพกระแสข้อมูลเป็นข้อมูลเข้าสำหรับเครื่องมือการนับฟังก์ชันพอยต์

ขั้นตอนการแปลงแผนภาพจากข้อมูลโมเดลของแผนภาพกระแสข้อมูลที่อยู่ในรูปแบบไฟล์ Microsoft Visio Drawing ที่วาดบนเครื่องมือ Microsoft Visio ซึ่งเครื่องมือสำหรับการนับฟังก์ชันพอยต์จะรับข้อมูลเข้าอยู่ในรูปแบบโครงสร้างไฟล์ XML โดยข้อมูลต่างๆ ของแผนภาพจะถูกเก็บอยู่ใน Tag โดยใช้โปรแกรม Text Editor มาใช้ในการแปลงไฟล์เป็น XML มีขั้นตอนการแปลงข้อมูลแผนภาพดังนี้

1. บันทึกการวาดข้อมูลแผนภาพบน Microsoft Visio โดยเลือกนามสกุลไฟล์เป็น .vdx (Microsoft Visio Drawing XML)
2. ใช้โปรแกรม Notepad++ ในการเปลี่ยนนามสกุลไฟล์เป็น XML โดยบันทึกไฟล์แผนภาพกระแสข้อมูลให้อยู่ในรูปแบบโครงสร้างไฟล์ XML

โดยในการวาดแผนภาพกระแสข้อมูลบน Microsoft Visio จะใช้สัญลักษณ์ต่างๆ บนเครื่องมือ Microsoft Visio (Biafore, 2007) (G. B. Shelly & Rosenblatt, 2011) เพื่อใช้ในการแทนสัญลักษณ์ที่ใช้ในการวาดแผนภาพกระแสข้อมูลดังตารางที่ 15

ตารางที่ 15 สัญลักษณ์มาตรฐานและที่ใช้ในวาดแผนภาพกระแสข้อมูลบน Visio

ชื่อสัญลักษณ์แบบมาตรฐาน	สัญลักษณ์แบบมาตรฐาน	ชื่อสัญลักษณ์บน Microsoft Visio	สัญลักษณ์บน Microsoft Visio
กระบวนการ (Process)	 Gane & Sarson  Yourdon	Data Process	
กระแสข้อมูล (Data Flow)		Dynamic Connector	
หน่วยเก็บข้อมูล (Data Store)	 Gane & Sarson  Yourdon	Data Store	 
Userหรือผู้ใช้งาน (External Entity)	 Gane & Sarson  Yourdon	External Interactor	

ข้อมูลของแผนภาพกระแสข้อมูล เมื่ออยู่ในรูปแบบโครงสร้างไฟล์ XML สัญลักษณ์ที่ใช้ในการวาดบนเครื่องมือ Microsoft Visio จะแสดงอยู่ใน Tag <Master>

โดย Tag <Masters><Master>.....</Master> </Masters> จะเก็บสัญลักษณ์ต่างๆ ที่ใช้ในการวาดแผนภาพ แสดงดังรูปที่ 20

```
<Masters>
<Master ID='0' NameU='Data process' Name='Data process' Prompt=
process. Use Center-to-center connectors between processes.' Icon
IconUpdate='0' UniqueID='{3FA793C9-0000-0000-8E40-00608CF305B2}
'(2269D92E-4080-48DA-BB5A-6BA82EBDF8F7)' PatternFlags='0' Hidden=
<Master ID='2' NameU='External interactor' Name='External interactor'
an external input, a terminator, or a state.' IconSize='1' AlignName=
UniqueID='{3FA7A92-000B-0000-8E40-00608CF305B2}' BaseID='{BCA51
PatternFlags='0' Hidden='0'>
<Master ID='3' NameU='Dynamic connector' Name='Dynamic connector'
routes between the shapes it connects.' IconSize='1' AlignName=
UniqueID='{002A9108-0000-0000-8E40-00608CF305B2}' BaseID='{F7290
PatternFlags='0' Hidden='0'>
<Master ID='4' NameU='Data store' Name='Data store' Prompt='Data
IconSize='1' AlignName='2' MatchByName='0' IconUpdate='0' UniqueID=
'{3FA7A8B8-000D-0000-8E40-00608CF305B2}' BaseID='{288D3974-4D320
Hidden='0'>
</Masters>
```

รูปที่ 20 แสดง Tag XML ของสัญลักษณ์ต่างๆ ที่ใช้ในการวาดแผนภาพบนเครื่องมือ Visio จากรูปที่ 20 แสดง Tag XML ที่เก็บข้อมูลสัญลักษณ์ที่ใช้ในการวาดแผนภาพบนเครื่องมือ Visio ดังนี้

- 1) หน่วยเก็บข้อมูล (Data Store) จะถูกระบุด้วย Tag <Master> ที่มี Attribute [ ID='0' NameU='Data process' Name='Data process' ]
- 2) User หรือผู้ใช้งาน (External Entity) จะถูกระบุด้วย Tag <Master> ที่มี Attribute [ ID='2' NameU='External interactor' Name='External interactor' ]
- 3) กระแสข้อมูล (Data Flow) จะถูกระบุด้วย Tag <Master> ที่มี Attribute [ ID='3' NameU='Dynamic connector' Name='Dynamic connector' ]
- 4) กระบวนการ (Process) จะถูกระบุด้วย Tag <Master> ที่มี Attribute [ ID='4' NameU='Data store' Name='Data store' ]

โดย Tag <Pages><Page>< Shapes> .....</Shapes></Page></Pages> จะเก็บข้อมูลรายละเอียดของข้อมูลโมเดลแผนภาพกระแสข้อมูลที่วาดบนเครื่องมือ Microsoft Visio แสดงในรูปที่

21

```
<Pages>
<Page ID='0' NameU='Level1' Name='Level_1' ViewScale=
'7.2317913385827'>
<PageSheet LineStyle='0' FillStyle='0' TextStyle=
<Shapes>
<Shape ID='1' NameU='Data process' Type='S
<Shape ID='2' NameU='External interactor'
<Shape ID='3' Type='Shape' Master='2'>
<Shape ID='4' NameU='Dynamic connector' Ty
<Shape ID='5' Type='Shape' Master='3'>
```

รูปที่ 21 แสดง Tag XML ของรายละเอียดสัญลักษณ์ที่ใช้ในการวาดแผนภาพโดย Visio

จากรูปที่ 21 ในแต่ละ Tag <Shape> จะประกอบด้วย Attribute ดังนี้ ID, NameU, TYPE และ Master โดยในแต่ละ Attribute จะระบุถึงข้อมูลรายละเอียดของข้อมูลแผนภาพกระแสข้อมูล โดยที่ในแต่ละ Tag <Shape> จะประกอบด้วย Tag <Misc> และ Tag <Text> ซึ่ง Tag <Misc>

ประกอบด้วย Tag <BegTrigger> และ <EndTrigger> ซึ่งมี Attribute F ที่มี Element แสดงจุดเริ่มต้น-สิ้นสุดของรายการข้อมูล ส่วน Tag <Text> จะเก็บ Element Text ของสัญลักษณ์ที่ใช้ในการวาดแผนภาพบน Visio แสดงในรูปแบบที่ 22

```

<Misc>
  <NoObjHandles F='Inh'>0</NoObjHandles>
  <NonPrinting F='Inh'>0</NonPrinting>
  <NoCtlHandles F='Inh'>0</NoCtlHandles>
  <NoAlignBox F='Inh'>1</NoAlignBox>
  <UpdateAlignBox F='Inh'>0</UpdateAlignBox>
  <HideText F='Inh'>0</HideText>
  <DynFeedback F='Inh'>2</DynFeedback>
  <GlueType F='Inh'>2</GlueType>
  <WalkPreference F='Inh'>0</WalkPreference>
  <BegTrigger F='_XFTRIGGER(External interactor!EventXFMod)
  <EndTrigger F='_XFTRIGGER(Data process.12!EventXFMod)'>2<
  <ObjType F='Inh'>2</ObjType>
  <Comment F='Inh'>/>
  <IsDropSource F='Inh'>0</IsDropSource>
  <NoLiveDynamics F='Inh'>1</NoLiveDynamics>
  <LocalizeMerge F='Inh'>0</LocalizeMerge>
  <Calendar F='Inh'>0</Calendar>
  <LangID F='Inh'>1033</LangID>
  <ShapeKeywords F='Inh'>/>
  <DropOnPageScale F='Inh'>1</DropOnPageScale>
</Misc>
<TextXForm>
  <Control NameU='TextPosition' ID='1'>
  <Geom IX='0'>
  <Text>PAYMENT
  </Text>

```

รูปที่ 22 แสดง Tag XML ของ Tag <BegTrigger> และ <EndTrigger> และ Tag <Text>

จากรูปที่ 22 Tag XML ของ Tag <BegTrigger> และ <EndTrigger> จะใช้ในการนับฟังก์ชันพอยต์ชนิดฟังก์ชันเชิงรายการเปลี่ยนแปลง ได้แก่ ข้อมูลเข้าภายนอก, ข้อมูลส่งออกภายนอก และการสอบถามภายนอก โดยจะวิเคราะห์จาก Element ของ Attribute F ในส่วนของ Tag <BegTrigger> และ <EndTrigger> โดยที่ในส่วนของการสอบถามภายนอก หากข้อมูลมีการใช้คำ “ค้นหาข้อมูล.....” “ความต้องการทราบข้อมูล.....” “ข้อมูลผลการค้นหา.....” “ข้อมูลผลความต้องการทราบข้อมูล.....” “รายงาน...” “การแจ้งเตือน.....” จะถูกระบุเป็นการสอบถามภายนอก ในส่วนของฟังก์ชันเชิงข้อมูลจะมีขั้นตอนการวิเคราะห์จาก Attribute ที่เป็น NameU ดังนี้

แฟ้มข้อมูลภายในระบบจะแยกได้จาก Tag <Shape> .....</Shape> โดยมี Attribute NameU ระบุเป็น “DataStore” หาก Tag <Shape> ไม่ปรากฏ Attribute ที่เป็น NameU จะพิจารณาในส่วนของ Attribute Master ที่มีการอ้างอิง ID เดียวกับ Attribute NameU ใน Tag <Master> แทน

เนื่องจากแฟ้มข้อมูลต่อประสานภายนอก การวาดแผนภาพกระแสข้อมูลบนเครื่องมือ Microsoft Visio ส่วนของผู้ใช้หรือ User ของระบบและภายนอกระบบ จะใช้สัญลักษณ์ในการวาดแบบเดียวกัน เมื่อมีการแปลงข้อมูลแผนภาพ ให้อยู่ในรูปแบบโครงสร้างไฟล์ XML จึงไม่สามารถแยกส่วนของผู้ใช้หรือ User ของระบบและภายนอกระบบออกจากกันได้ จึงจำเป็นต้องระบุส่วนดังกล่าวก่อน เพื่อใช้ในการอ้างอิงส่วนดังกล่าวว่าเป็นแฟ้มข้อมูลต่อประสานภายนอก ก่อนจะเข้าสู่ขั้นตอนการนับฟังก์ชันพอยต์แยกตามประเภทฟังก์ชัน

#### 4.2 การออกแบบและการพัฒนาเครื่องมือเครื่องมือสำหรับการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล

เครื่องมือสนับสนุนที่ใช้ในการพัฒนาเครื่องมือ

ฮาร์ดแวร์ (Hardware) และซอฟต์แวร์ (Software) ที่ใช้ในการพัฒนาเครื่องมือของงานวิจัยนี้มีรายละเอียดดังนี้

##### 1) ด้านฮาร์ดแวร์

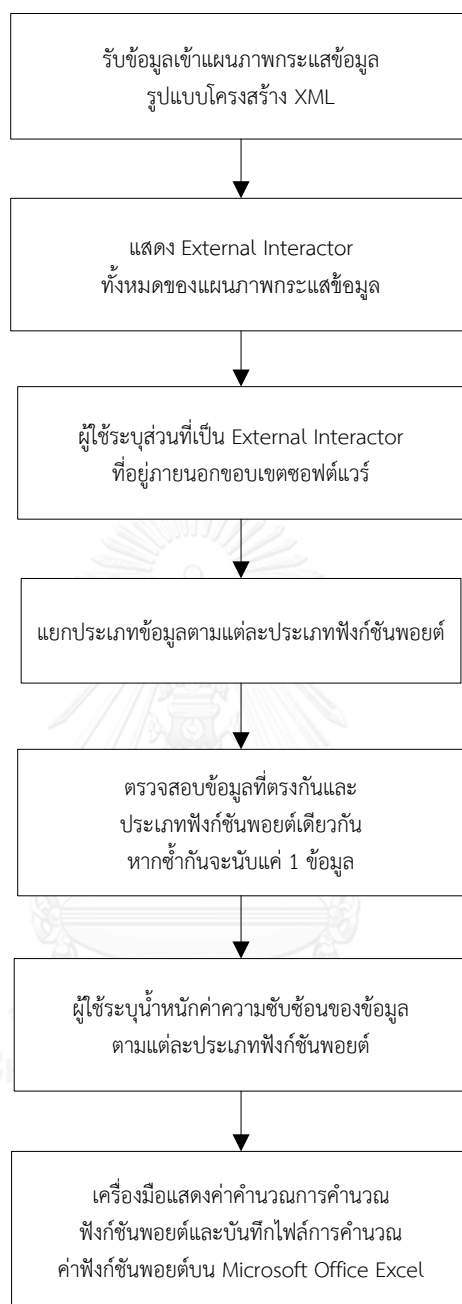
- เครื่องคอมพิวเตอร์ส่วนบุคคล (Intel core i5 2.50 GHz)
- หน่วยความจำ (Memory) 4 กิกะไบต์
- จานบันทึกแบบแข็ง (Hard disk) ความจุ 512 กิกะไบต์

##### 2) ด้านซอฟต์แวร์

- เครื่องมือพัฒนาโปรแกรมภาษาจาวาเน็ตเบินส์ (NetBeans)
- เครื่องมือวาดแผนภาพกระแสข้อมูล (Microsoft Visio)
- เครื่องมือที่ใช้ในการแปลงแผนภาพกระแสข้อมูลให้อยู่ในโครงสร้าง XML (Notepad++)
- ไมโครซอฟท์ออฟฟิศ 2010 (Microsoft office Excel)

เครื่องมือที่พัฒนามีการออกแบบตามหลักการวิเคราะห์ฟังก์ชันพอยต์ข้างต้น และตามโครงสร้างไฟล์ XML ของแผนภาพกระแสข้อมูลที่วาดโดยเครื่องมือ Microsoft Visio มาใช้ในการออกแบบเครื่องมือ เพื่อใช้สำหรับการนับฟังก์ชันพอยต์ของแผนภาพกระแสข้อมูลในทุกระดับแผนภาพกระแสข้อมูล

ขั้นตอนการทำงานของเครื่องมือสำหรับการนับค่าฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล



รูปที่ 23 แสดงชั้นแผนภาพแสดงขั้นตอนการทำงานของเครื่องมือ

1. รับข้อมูลเข้าเป็นข้อมูลโมเดลแผนภาพกระแสข้อมูลที่อยู่ในรูปแบบโครงสร้างไฟล์ XML
2. แสดงจำนวนของ External Interactor ทั้งหมดของแผนภาพกระแสข้อมูล โดยที่ผู้ใช้จะระบุส่วนนี้ จาก External Interactor ที่มีคำว่า “ระบบ” “Google...” หรือ “เว็บไซต์ ....” เป็นต้น เพื่อแยกประเภทฟังก์ชันของแฟ้มข้อมูลต่อประสานภายนอก โดยกระแสข้อมูลที่ส่งมาจากส่วนนี้จะนับเป็นแฟ้มข้อมูลต่อประสานภายนอก

3. ประมวลผลแยกประเภทเพิ่มข้อมูลภายในระบบ Tag <Shape> ที่ Attribute เป็น NameU โดยจะใช้การวิเคราะห์จาก Attribute ของ Tag <Shape> โดยที่ NameU มีการระบุ Element เป็น 'Data store' แสดงในรูปที่ 24

```
<Shape ID='10' NameU='Data store' Type='Shape' Master='4'>
  <XForm>
  <Event>
  <vx:Event xmlns:vx='http://schemas.microsoft.com/visio/
```

รูปที่ 24 แสดง Tag <Shape> ที่ Attribute เป็น NameU='Data store'

แต่จากรูปที่ 21 บาง Tag <Shape> Attribute ที่เป็น NameU= จะไม่ปรากฏทุก Tag <Shape> แต่ Attribute ที่เป็น Master= ใน Tag <Shape> จะอ้างอิง Element ในส่วน Attribute Name=U เดียวกันกับ Tag <Master> ที่มีการระบุ ID ในรูปที่ 4-4 ตัวอย่างเช่น Tag <Shape ID='21' Type='Shape' Master='4'> แสดงในรูปที่ 25

```
<Shape ID='21' Type='Shape' Master='4'>
  <XFORM>
  <Event>
  <vx:Event xmlns:vx='http://schemas.mic
  <LayerMem>
  <Misc>
```

รูปที่ 25 แสดง Tag <Shape> ที่ Attribute ที่เป็น NameU ไม่ปรากฏใน Tag

จากรูปที่ 25 Tag <Shape> ไม่ปรากฏ Attribute ที่เป็น NameU ดังนั้นจะดู Attribute ที่เป็น Master ของ Tag <Shape> แทน จากรูปที่ 25 Master='4' ซึ่งตรงกับ Tag <Master> ที่มี ID เป็น 4 ดังนั้น Name=U ของ Tag <Shape ID='21' เป็น 'Data store' และในกรณีเดียวกันหากไม่มีการระบุ Attribute ที่เป็น NameU ก็จะใช้หลักการอ้างอิงเดียวกันกับตัวอย่างในรูปที่ 25

4. ประมวลผลแยกประเภทฟังก์ชันพอยต์ของข้อมูลเข้าภายนอก, ข้อมูลส่งออกภายนอก และการสอบถามภายนอก โดยพิจารณาจาก Tag <Shape><Misc> ในส่วนของ Tag <BegTrigger> และ <EndTrigger> ดังนี้

การวิเคราะห์ข้อมูลเข้าภายนอก(EIS) โดยข้อมูลแผนภาพกระแสข้อมูลจะแสดงในรูปแบบไฟล์ XML แสดงในรูปที่ 26

```
<Misc>
  <NoObjHandles F='Inh'>0</NoObjHandles>
  <NonPrinting F='Inh'>0</NonPrinting>
  <NoCtlHandles F='Inh'>0</NoCtlHandles>
  <NoAlignBox F='Inh'>1</NoAlignBox>
  <UpdateAlignBox F='Inh'>0</UpdateAlignBox>
  <HideText F='Inh'>0</HideText>
  <DynFeedback F='Inh'>2</DynFeedback>
  <GlueType F='Inh'>2</GlueType>
  <WalkPreference F='Inh'>0</WalkPreference>
  <BegTrigger F='XFTRIGGER(External_Interaction|EventXMod)'>2</BegTrigger>
  <EndTrigger F='XFTRIGGER(Data_process.12|EventXMod)'>2</EndTrigger>
  <ObjType F='Inh'>2</ObjType>
  <Comment F='Inh'>/>
  <IsDropSource F='Inh'>0</IsDropSource>
  <NoLiveDynamics F='Inh'>1</NoLiveDynamics>
  <LocalizeMerge F='Inh'>0</LocalizeMerge>
  <Calendar F='Inh'>0</Calendar>
  <LangID F='Inh'>1033</LangID>
  <ShapeKeywords F='Inh'>/>
  <DropOnPageScale F='Inh'>1</DropOnPageScale>
</Misc>
<TextXForm>
  <Control NameU='TextPosition' ID='1'>
  <Geom Type='0'>
  <Text>PAYMENT
```

รูปที่ 26 แสดงข้อมูลเข้าภายนอก(EIS) ในรูปแบบไฟล์ XML

จากรูปที่ 26 Tag <BegTrigger> ประกอบด้วย Attribute F ที่มี Element ระบุถึง External interactor และ Tag <EndTrigger>ประกอบด้วย Attribute F ที่มี Element ระบุถึง Data process โดยจะเก็บข้อความที่แสดงใน Tag <Text> จากรูปที่ 26 ข้อความคือ PAYMENT การวิเคราะห์ข้อมูลส่งออกภายนอก(EOs) โดยข้อมูลแผนภาพกระแสข้อมูลจะแสดงในรูปแบบไฟล์ XML แสดงในรูปแบบที่ 27

```
<Misc>
  <NoObjHandles F='Inh'>0</NoObjHandles>
  <NonPrinting F='Inh'>0</NonPrinting>
  <NoCtlHandles F='Inh'>0</NoCtlHandles>
  <NoAlignBox F='Inh'>1</NoAlignBox>
  <UpdateAlignBox F='Inh'>0</UpdateAlignBox>
  <HideText F='Inh'>0</HideText>
  <DynFeedback F='Inh'>2</DynFeedback>
  <GlueType F='Inh'>2</GlueType>
  <WalkPreference F='Inh'>0</WalkPreference>
  <BegTrigger F='XFTRIGGER(Data process!EventXFMod)'>2</BegTrigger>
  <EndTrigger F='XFTRIGGER(Data process.12!EventXFMod)'>2</EndTrigger>
  <ObjType F='Inh'>2</ObjType>
  <Comment F='Inh' />
  <IsDropSource F='Inh'>0</IsDropSource>
  <NoLiveDynamics F='Inh'>1</NoLiveDynamics>
  <LocalizeMerge F='Inh'>0</LocalizeMerge>
  <Calendar F='Inh'>0</Calendar>
  <LangID F='Inh'>1033</LangID>
  <ShapeKeywords F='Inh' />
  <DropOnPageScale F='Inh'>1</DropOnPageScale>
</Misc>
<TextXForm>
  <Control NameU='TextPosition' ID='1'>
    <Geom IX='0'>
      <Text>COMMISSION EARNED
```

รูปที่ 27 แสดงข้อมูลส่งออกภายนอก(EOs) ในรูปแบบไฟล์ XML

จากรูปที่ 27 Tag <BegTrigger> ประกอบด้วย Attribute F ที่มี Element ระบุถึง Data process และ Tag <EndTrigger>ประกอบด้วย Attribute F ที่มี Element ระบุถึง External interactor โดยจะเก็บข้อความที่แสดงใน Tag <Text> จากรูปที่ 27 ข้อความคือ COMMISSION EARNED

การวิเคราะห์ข้อมูลการสอบถามภายนอก (EQs) จะวิเคราะห์จากคำ “ค้นหาข้อมูล.....” “ความต้องการทราบข้อมูล....” “ข้อมูลผลการค้นหา.....” “ข้อมูลผลความต้องการทราบข้อมูล.....” “รายงาน...” “การแจ้งเตือน.....” เพื่อให้ซอฟต์แวร์แสดงผลการสอบถามจะถูกระบุเป็นการสอบถามภายนอก

5. เครื่องมือที่พัฒนาจะไม่นับแฟ้มข้อมูลที่ซ้ำกับระดับแผนภาพแรก(Donald, 2000)
6. ในประเมินความซับซ้อนของแต่ละประเภทฟังก์ชันพอยต์จะแสดงให้เห็นผู้ใช้ระบุความซับซ้อนด้วยตนเอง โดยใช้ข้อมูลในของแผนภาพแสดงความสัมพันธ์ของข้อมูล และพจนานุกรมข้อมูลของระบบ โดยจะให้กรอกจำนวนของเขตข้อมูล (Data Element Types : DETs) และจำนวนตารางข้อมูล (Record Element Types : RETs)จำนวนแฟ้มข้อมูลอ้างอิง (Files Type Referenced : FTRs) โดยใช้หลักการประเมินระดับความซับซ้อนดังหัวข้อที่ 4.1.2 การคำนวณการนับค่าฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล

#### 4.3 การทดสอบเครื่องมือสำหรับการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล

ผู้วิจัยสร้างแผนภาพกระแสข้อมูลบนเครื่องมือ Microsoft Visio จำนวน 5 แผนภาพจากของซอฟต์แวร์ที่มีคุณสมบัติต่างกัน เช่น ใช้ฐานข้อมูลจำนวนมาก เน้นการประมวลผลในรูปการออกรายงาน หรือฐานข้อมูลจำนวนไม่มาก แต่เน้นการประมวลผลที่ลึก และพิมพ์เป็นรูปแบบกระดาษไปให้ผู้เชี่ยวชาญทางด้านฟังก์ชันพอยต์จำนวน 1 คนและนิสิตที่ศึกษาวิธีการอ่านค่าฟังก์ชันพอยต์อ่านค่าจากแผนภาพกระแสข้อมูลเพื่อนับค่าข้อมูล ผู้วิจัยได้นำข้อมูลแผนภาพทั้ง 5 จาก Microsoft Visio ไปประมวลผลด้วยเครื่องมือสำหรับการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลที่พัฒนาตามหลักการออกแบบหัวข้อที่ 4.2 ผลแสดงดังตารางที่ 16

ตารางที่ 16 แสดงการเปรียบเทียบผลการนับจำนวนฟังก์ชันพอยต์ของแผนภาพกระแสข้อมูล (1) ของเครื่องมือที่พัฒนา (2)ผู้เชี่ยวชาญทางด้านฟังก์ชันพอยต์ (3)นิสิตที่ศึกษาวิธีการอ่านค่าฟังก์ชันพอยต์

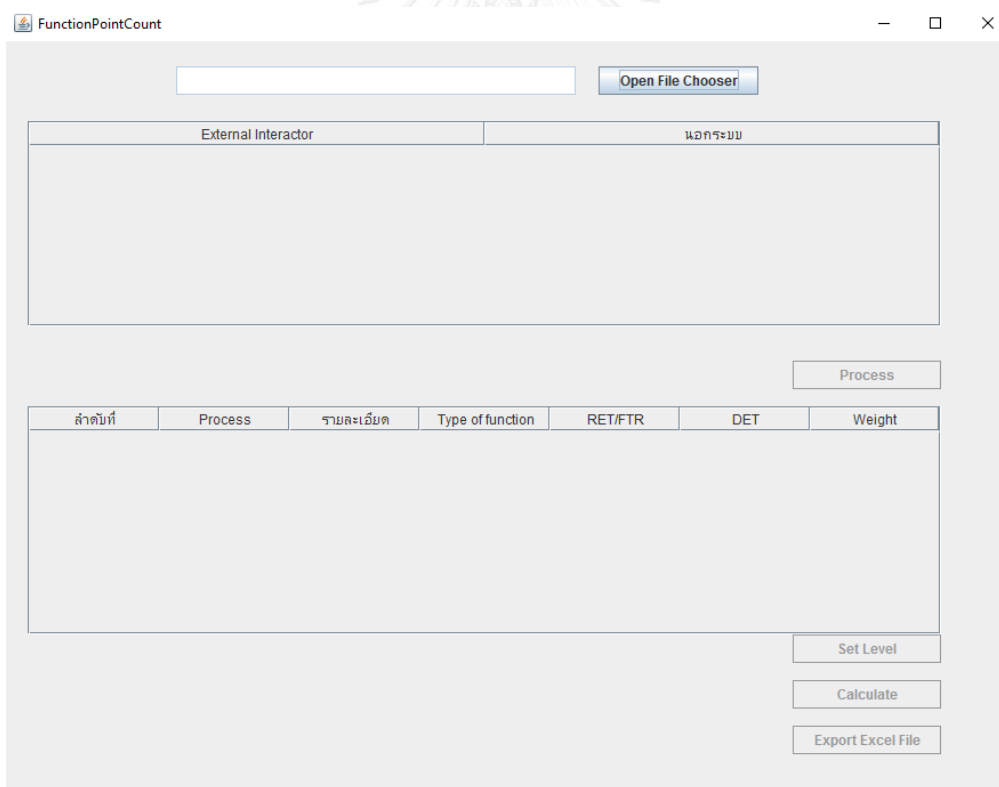
ประเภทฟังก์ชันพอยต์	ชุดที่ 1			ชุดที่ 2			ชุดที่ 3			ชุดที่ 4			ชุดที่ 5		
	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
ข้อมูลเข้าภายนอก (EIs)	38	38	38	23	23	24	34	34	32	37	37	34	28	28	25
ข้อมูลส่งออกภายนอก (EOs)	16	16	14	20	20	20	27	27	25	23	23	20	17	17	17
การสอบถามภายนอก (EQs)	8	8	8	7	7	7	5	5	5	7	7	7	6	6	6
เพิ่มข้อมูลต่อประสานภายนอก (EIFs)	2	2	2	3	3	3	2	2	2	-	-	-	-	-	-
เพิ่มข้อมูลภายในระบบ (ILFs)	27	27	25	15	15	17	24	24	24	16	16	16	20	20	18



จากตารางที่ 16 พบว่าการนับค่าฟังก์ชันพอยต์โดยผู้เชี่ยวชาญทางด้านฟังก์ชันพอยต์และโดยเครื่องมือที่สร้างขึ้นมีจำนวนค่าฟังก์ชันพอยต์เท่ากัน ส่วนค่าที่ได้จากนิตินิตมีค่าจำนวนฟังก์ชันพอยต์ผิดพลาดจากผู้เชี่ยวชาญและโปรแกรมบ้าง จึงมั่นใจว่าเครื่องมือที่พัฒนาขึ้นมีความถูกต้องแม่นยำ

#### 4.4 การใช้งานเครื่องมือสำหรับการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล

เครื่องมือสำหรับการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลเป็นเครื่องมือที่พัฒนาสำหรับงานวิจัย เพื่อใช้ในการนับจำนวนฟังก์ชันพอยต์ เปรียบเทียบจำนวนฟังก์ชันพอยต์จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วของโครงการพัฒนาซอฟต์แวร์ โดยเครื่องมือจะรับไฟล์เข้าเป็นแผนภาพกระแสข้อมูลที่อยู่ในรูปแบบโครงสร้างไฟล์ XML โดยเครื่องมือจะทำงานแบ่งเป็น 4 ส่วน คือ 1. ส่วนขั้นตอนนำเข้าไฟล์(open file chooser) 2. ส่วนที่เป็นขั้นตอนการระบุ External Interactor ที่เป็นภายนอกระบบ 3. ส่วนที่เป็นการแยกประเภทฟังก์ชันพอยต์ 4. การคำนวณค่าฟังก์ชันพอยต์และการแสดงผลการคำนวณ เพื่อใช้ในการเปรียบเทียบค่าฟังก์ชันพอยต์ ดังรูปที่ 28 แสดงส่วนต่อประสานกับผู้ใช้งานของเครื่องการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล



รูปที่ 28 แสดงส่วนต่อประสานกับผู้ใช้งานของเครื่องการนับฟังก์ชันพอยต์

## บทที่ 5

### การประมวลผลข้อมูลและการวิเคราะห์ผลลัพธ์การวิจัย

ในบทนี้อธิบายวิธีการทดสอบสมมติฐาน ประกอบด้วยหน่วยตัวอย่างของงานวิจัย การได้มาซึ่งข้อมูลของงานวิจัย และการทดสอบสมมติฐานของงานวิจัย

#### 5.1 หน่วยตัวอย่าง

เป็นข้อมูลโครงการพัฒนาซอฟต์แวร์จากโครงการพิเศษ ของนิสิตระดับบัณฑิตศึกษา ปริญญาโท สาขาการพัฒนาระบบสารสนเทศด้านธุรกิจ ที่ใช้แผนภาพกระแสข้อมูล (Data Flow Diagrams) แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล (ER-Diagrams) และพจนานุกรมข้อมูล (Data Dictionary) ในการพัฒนาซอฟต์แวร์ และประกอบด้วยชุดข้อมูลรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้จากโครงการพิเศษ 1 และรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วจากโครงการพิเศษ 2 จำนวน 30 ชุดข้อมูลโครงการพิเศษ

จากชุดข้อมูล 30 ชุดโครงการพิเศษของนิสิตระดับปริญญาโท ผู้วิจัยได้มีการปรับปรุงหน่วยตัวอย่างให้มีความถูกต้องตามหลักการเขียนแผนภาพกระแสข้อมูล ปรับปรุงโดยยึดเส้นข้อมูลที่ปรากฏในส่วนของแผนภาพบริบท(Context Diagram) และจำนวนเพิ่มข้อมูลที่ปรากฏในแผนภาพระดับที่หนึ่ง เพื่อให้ข้อมูลสามารถนำมาใช้ในการเปรียบเทียบได้อย่างถูกต้อง

#### 5.2 การได้มาซึ่งข้อมูล

ชุดข้อมูลโครงการพัฒนาซอฟต์แวร์ ที่ประกอบด้วยแผนภาพกระแสข้อมูลจากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วจำนวน 2 ชุดข้อมูลต่อ 1 โครงการพัฒนาซอฟต์แวร์

โดยแผนภาพกระแสข้อมูลที่ได้มาจะอยู่ในรูปแบบกระดาษ โดยที่ผู้วิจัยจะวาดแผนภาพกระแสข้อมูลที่ได้มาบนเครื่องมือ Microsoft Visio และบันทึกไฟล์ในรูปแบบนามสกุลไฟล์เป็น .vdx (Microsoft Visio Drawing XML) และใช้โปรแกรม Notepad++ ในการเปลี่ยนนามสกุลไฟล์เป็น XML โดยบันทึกไฟล์แผนภาพกระแสข้อมูลให้อยู่ในรูปแบบโครงสร้างไฟล์ XML

ตารางที่ 17 แสดงชุดข้อมูลของโครงการพัฒนาซอฟต์แวร์

ข้อมูลจากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้	ข้อมูลจากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว
- แผนภาพกระแสข้อมูล จัดเก็บอยู่ในรูปแบบโครงสร้างไฟล์ XML	- แผนภาพกระแสข้อมูล จัดเก็บอยู่ในรูปแบบโครงสร้างไฟล์ XML
- แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลในรูปแบบกระดาษ	- แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลในรูปแบบกระดาษ
- พจนานุกรมข้อมูลในรูปแบบกระดาษ	- พจนานุกรมข้อมูลในรูปแบบกระดาษ

เมื่อชุดข้อมูลของโครงการพัฒนาซอฟต์แวร์ที่นำมาเป็นหน่วยตัวอย่างจำนวน 30 ชุดข้อมูลแล้ว ผู้วิจัยใช้เครื่องมือสำหรับการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลนำมาใช้ในการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลที่อยู่ในรูปแบบโครงสร้างไฟล์ XML โดยใช้แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลและพจนานุกรมข้อมูล สำหรับการวิเคราะห์ฟังก์ชันพอยต์

ตารางที่ 18 แสดงชื่อชุดข้อมูลจำนวน 30 ชุดข้อมูล ที่นำมาใช้เป็นข้อมูลของงานวิจัย

ID	ชื่อโครงการพัฒนาซอฟต์แวร์
PJ1	ระบบธุรกิจโต๊ะเงินออนไลน์
PJ2	ซอฟต์แวร์เพื่อการออกแบบจัดสวนสาธารณะ
PJ3	ซอฟต์แวร์สนับสนุนกระบวนการผลิตอ้อย
PJ4	ซอฟต์แวร์สำหรับธุรกิจพิมพ์ลายสกรีนเสื้อคู่
PJ5	ซอฟต์แวร์เพื่อการแปลงบทละครโทรทัศน์
PJ6	ซอฟต์แวร์เพื่อการจัดตำแหน่งบุคคลสำหรับการถ่ายภาพหมู่คณะ
PJ7	ระบบขนส่งน้ำมัน
PJ8	ซอฟต์แวร์วางแผนการเดินทางสำหรับนักท่องเที่ยวแบบแบกเป้
PJ9	ซอฟต์แวร์ช่วยในการออกแบบรองเท้า
PJ10	ซอฟต์แวร์วางแผนเที่ยวชมภายในสถานที่ท่องเที่ยวออนไลน์
PJ11	ซอฟต์แวร์สนับสนุนการออกแบบขยายเขตไฟฟ้า
PJ12	ระบบแนะนำการถ่ายภาพก่อนแต่งงาน
PJ13	ซอฟต์แวร์ออกแบบเครื่องประดับที่เหมาะสมกับอัญมณี
PJ15	ระบบสารสนเทศสำหรับสถานีบริการน้ำมัน
PJ16	ระบบวางแผนการเดินทางด้วยขนส่งมวลชนสาธารณะ
PJ17	ระบบจัดชุดอาหารกลางวันสำหรับพนักงาน
PJ19	ระบบวางแผนสายการผลิตและควบคุมการผลิตในโรงงานเย็บผ้า
PJ21	ระบบสารสนเทศเพื่อการหาสถานที่ถ่ายทำสำหรับอุตสาหกรรมภาพยนตร์
PJ24	ระบบสารสนเทศเพื่อการบริหารการติดตั้งเคเบิลทีวี
PJ26	ซอฟต์แวร์สำหรับธุรกิจสนามไตร์กอล์ฟที่เน้นการให้บริการลูกค้า
PJ28	ระบบสารสนเทศสำหรับธุรกิจร้านเสริมสวยขนาดใหญ่
PJ29	ซอฟต์แวร์จองศาลาเว็ดสำหรับงานศพในกรุงเทพฯ
PJ30	ซอฟต์แวร์สนับสนุนการขายกิจกรรมนันทนาการที่ตรงกับวัตถุประสงค์ของการจัดกิจกรรมและเหมาะสมกับผู้เข้าร่วมกิจกรรม
PJ31	ซอฟต์แวร์สำหรับธุรกิจผู้ให้บริการโทรทางไกลต่างประเทศ
PJ32	ซอฟต์แวร์ช่วยเหลือนักแต่งเพลงในการสร้างทำนองเพลงจากเนื้อร้อง
PJ33	ซอฟต์แวร์เพื่อการขายเพลงในรูปแบบ Audio CD
PJ34	ซอฟต์แวร์ช่วยเลือกรองเท้าวิ่งระยะทางไกลตามสรีระและการใช้งานสำหรับร้านขายรองเท้าวิ่ง

ID	ชื่อโครงการพัฒนาซอฟต์แวร์
PJ35	ซอฟต์แวร์สำหรับธุรกิจขายเครื่องฟอกอากาศ
PJ36	ซอฟต์แวร์สำหรับจัดจำหน่ายผลิตภัณฑ์จากเกษตรกรรายย่อย
PJ37	ระบบสารสนเทศสำหรับการขายขนมปังกรอบตามบรรจุภัณฑ์ของลูกค้า

ข้อมูลจากโครงการพัฒนาซอฟต์แวร์จำนวน 30 ชุดข้อมูลที่ใช้แผนภาพกระแสข้อมูล แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล และพจนานุกรมข้อมูลในการพัฒนาซอฟต์แวร์ที่แสดงดัง ตารางที่ 5-2 จากแนวคิดรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้จากโครงการพิเศษ 1 และ รายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วจากโครงการพิเศษ 2 โดยรายละเอียดซอฟต์แวร์ที่สร้างเสร็จ แล้วจะมีจำนวนฟังก์ชันพอยต์เพิ่มมากขึ้น ได้รวบรวมผลการนับฟังก์ชันพอยต์ที่ใช้เครื่องมือในการ วิเคราะห์ฟังก์ชันพอยต์แบ่งเป็น 6 ส่วนดังนี้

1. ส่วนที่เป็นข้อมูลนำเข้าภายนอก (EIs)
2. ส่วนที่เป็นข้อมูลส่งออกภายนอก (EOs)
3. ส่วนที่เป็นการสอบถามภายนอก (EQs)
4. ส่วนที่เป็นแฟ้มข้อมูลภายใน (ILF's)
5. ส่วนที่เป็นแฟ้มข้อมูลต่อประสานภายนอก (EIF's)
6. ส่วนที่เป็นจำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์

ตารางที่ 19 แสดงผลการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก

ข้อมูลนำเข้า ภายนอก (EIs)	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้าง เสร็จแล้ว
PJ1	72	81
PJ2	27	27
PJ3	66	63
PJ4	51	53
PJ5	30	52
PJ6	21	30
PJ7	66	63

ข้อมูลนำเข้า ภายนอก (EIs)	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้าง เสร็จแล้ว
PJ8	44	41
PJ9	84	111
PJ10	24	24
PJ11	48	48
PJ12	58	61
PJ13	72	96
PJ15	73	223
PJ16	21	33
PJ17	57	56
PJ19	51	45
PJ21	33	33
PJ24	49	53
PJ26	63	63
PJ28	141	141
PJ29	45	58
PJ30	51	57
PJ31	39	39
PJ32	36	39
PJ33	75	80
PJ34	39	36
PJ35	84	72
PJ36	60	60
PJ37	66	69

ตารางที่ 20 แสดงผลการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก

ข้อมูลส่งออก ภายนอก (EOs)	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่ สร้างเสร็จแล้ว
PJ1	68	72
PJ2	28	36
PJ3	24	24
PJ4	48	48
PJ5	24	56
PJ6	20	40
PJ7	40	28
PJ8	8	8
PJ9	88	112
PJ10	20	20
PJ11	52	52
PJ12	20	24
PJ13	72	64
PJ15	56	25
PJ16	8	8
PJ17	24	27
PJ19	44	48
PJ21	48	48
PJ24	37	46
PJ26	112	112
PJ28	100	100
PJ29	29	61
PJ30	32	44
PJ31	24	16
PJ32	20	24
PJ33	60	64
PJ34	28	36
PJ35	12	8
PJ36	40	40
PJ37	20	24

ตารางที่ 21 แสดงผลการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลการสอบถามภายนอก

ข้อมูลการสอบถาม ภายนอก (EQs)	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่ สร้างเสร็จแล้ว
PJ1	9	9
PJ2	15	15
PJ3	3	3
PJ4	0	4
PJ5	0	0
PJ6	0	0
PJ7	9	9
PJ8	0	0
PJ9	9	6
PJ10	6	9
PJ11	6	13
PJ12	0	0
PJ13	0	0
PJ15	13	15
PJ16	3	3
PJ17	9	9
PJ19	0	0
PJ21	0	0
PJ24	6	6
PJ26	0	0
PJ28	12	21
PJ29	9	12
PJ30	0	0
PJ31	6	6
PJ32	0	0
PJ33	0	0

ข้อมูลการสอบถาม ภายนอก (EQs)	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่ สร้างเสร็จแล้ว
PJ34	9	9
PJ35	9	12
PJ36	6	6
PJ37	6	6

ตารางที่ 22 แสดงผลการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลเพิ่มข้อมูลภายใน

เพิ่มข้อมูลภายใน (ILF's)	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้าง เสร็จแล้ว
PJ1	112	182
PJ2	56	63
PJ3	105	91
PJ4	84	84
PJ5	98	77
PJ6	42	42
PJ7	168	189
PJ8	112	112
PJ9	112	196
PJ10	35	35
PJ11	133	133
PJ12	119	126
PJ13	56	63
PJ15	132	122
PJ16	49	63
PJ17	112	122
PJ19	154	154
PJ21	42	56
PJ24	117	124
PJ26	182	147
PJ28	210	210
PJ29	63	90



เพิ่มข้อมูลภายใน (ILF's)	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้าง เสร็จแล้ว
PJ30	112	133
PJ31	105	105
PJ32	98	98
PJ33	70	77
PJ34	28	42
PJ35	105	133
PJ36	63	77
PJ37	84	105

ตารางที่ 23 แสดงผลการคำนวณค่าฟังก์ชันพอยต์ของเพิ่มข้อมูลต่อประสานภายนอก

เพิ่มข้อมูลต่อประสาน ภายนอก (EIF's)	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่ สร้างเสร็จแล้ว
PJ1	0	0
PJ2	0	0
PJ3	0	0
PJ4	0	0
PJ5	0	0
PJ6	0	0
PJ7	15	15
PJ8	0	35
PJ9	0	0
PJ10	0	0
PJ11	30	30
PJ12	0	0
PJ13	5	0
PJ15	10	10
PJ16	5	5
PJ17	0	0
PJ19	0	0
PJ21	0	0
PJ24	0	0

แฟ้มข้อมูลต่อประสาน ภายนอก (EIF's)	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่ สร้างเสร็จแล้ว
PJ26	0	0
PJ28	0	0
PJ29	0	0
PJ30	0	0
PJ31	0	0
PJ32	0	0
PJ33	0	0
PJ34	0	0
PJ35	0	0
PJ36	0	0
PJ37	0	0

ตารางที่ 24 แสดงผลการคำนวณค่าฟังก์ชันพอยต์ผลรวมทั้งหมด

จำนวนผลรวมทั้งหมดของ ฟังก์ชันพอยต์	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่ สร้างเสร็จแล้ว
PJ1	261	344
PJ2	126	141
PJ3	198	181
PJ4	183	189
PJ5	152	185
PJ6	83	112
PJ7	298	304
PJ8	164	196
PJ9	293	425
PJ10	85	88
PJ11	269	276
PJ12	197	211
PJ13	205	223

จำนวนผลรวมทั้งหมดของ ฟังก์ชันพอยต์	ผลการคำนวณค่าฟังก์ชันพอยต์	
	รายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่ สร้างเสร็จแล้ว
PJ15	284	395
PJ16	86	112
PJ17	202	214
PJ19	249	247
PJ21	123	137
PJ24	209	229
PJ26	357	322
PJ28	463	472
PJ29	146	221
PJ30	195	234
PJ31	174	166
PJ32	154	161
PJ33	205	221
PJ34	104	123
PJ35	210	225
PJ36	169	183
PJ37	176	204

### 5.3 การวิเคราะห์ผลข้อมูลการวิจัย

จากการเก็บรวบรวมข้อมูลผลการคำนวณค่าฟังก์ชันพอยต์รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้จากโครงการพิเศษ 1 และรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้วจากโครงการพิเศษ 2 จำนวน 30 ชุดข้อมูลมาวิเคราะห์ โดยแบ่งออกเป็น 2 ส่วน คือ

1. การทดสอบสมมติฐานงานวิจัย
2. การวิเคราะห์เปอร์เซ็นต์การเปลี่ยนแปลงของฟังก์ชันพอยต์ของรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้กับรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว

### 5.3.1 การทดสอบสมมติฐาน

ผู้วิจัยดำเนินการนำผลการทดสอบการนับฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลจากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วของโครงการพัฒนาซอฟต์แวร์ มาวิเคราะห์เพื่อทดสอบสมมติฐานงานวิจัยด้วยสถิติด้วยวิธี Paired t-test โดยแบบการวิเคราะห์ออกเป็น 6 สมมติฐาน ดังนี้

สมมติฐานที่ 1 : จำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก (EIs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ น้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก (EIs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

$$H_0 : FP(EIs)_{\text{หลัง} - \text{ก่อน}} \leq 0$$

$$H_1 : FP(EIs)_{\text{หลัง} - \text{ก่อน}} > 0$$

จากสมมติฐาน  $df=29$  ( $N = 30$ ) ซึ่งกำหนดระดับนัยสำคัญ (95% Confidence Interval) เท่ากับ 0.05 จะได้ค่าวิกฤต  $t=2.4620$  ( $t_{\text{ตาราง}} = 2.4620$ )

จะยอมรับ  $H_0$  เมื่อค่า  $t_{\text{คำนวณ}} > t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_1$

จะยอมรับ  $H_1$  เมื่อค่า  $t_{\text{คำนวณ}} < t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_0$

ผลการวิเคราะห์เพื่อทดสอบสมมติฐานทางสถิติ โดยการเปรียบเทียบจำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ดังตารางที่ 25

ตารางที่ 25 แสดงผลการทดสอบสมมติฐานทางสถิติของข้อมูลนำเข้าภายนอก

ประเภทฟังก์ชันพอยต์	จำนวนฟังก์ชันพอยต์		t คำนวณ
	รายละเอียดของ ซอฟต์แวร์ที่ ออกแบบ	รายละเอียด ซอฟต์แวร์ที่สร้าง เสร็จแล้ว	
	Mean (SD)	Mean (SD)	
ข้อมูลนำเข้าจากภายนอก (EIs)	53.03 (25.47)	61.73 (40.32)	1.70

ผลการทดสอบสมมติฐานดังตารางที่ 25 เมื่อพิจารณาค่า  $t_{\text{คำนวณ}}$  และค่า Sig พบว่า

จากตารางพบว่าค่า  $t_{\text{คำนวณ}}$  เท่ากับ 1.70 มีค่าน้อยกว่า  $t_{\text{ตาราง}}$  ดังนั้นจะยอมรับ  $H_1$  ปฏิเสธ  $H_0$  ดังนั้นสรุปได้ว่าจำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอกจากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ น้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอกจากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

สมมติฐานที่ 2 : จำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (EOs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก (EOs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

$$H_0 : FP(EOs)_{\text{หลัง} - \text{ก่อน}} \leq 0$$

$$H_1 : FP(EOs)_{\text{หลัง} - \text{ก่อน}} > 0$$

จากสมมติฐาน  $df=29$  ( $N = 30$ ) ซึ่งกำหนดระดับนัยสำคัญ (95% Confidence Interval) เท่ากับ 0.05 จะได้ค่าวิกฤต  $t=2.4620$  ( $t_{\text{ตาราง}} = 2.4620$ )

จะยอมรับ  $H_0$  เมื่อค่า  $t_{\text{คำนวณ}} > t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_1$

จะยอมรับ  $H_1$  เมื่อค่า  $t_{\text{คำนวณ}} < t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_0$

ผลการวิเคราะห์เพื่อทดสอบสมมติฐานทางสถิติ โดยการเปรียบเทียบจำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอก จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ดังตารางที่ 26

ตารางที่ 26 แสดงผลการทดสอบสมมติฐานทางสถิติของข้อมูลส่งออกภายนอก

ประเภทฟังก์ชันพอยต์	จำนวนฟังก์ชันพอยต์		t คำนวณ
	รายละเอียดของ ซอฟต์แวร์ที่ ออกแบบ	รายละเอียด ซอฟต์แวร์ที่สร้าง เสร็จแล้ว	
	Mean (SD)	Mean (SD)	
ข้อมูลส่งออกภายนอก (EOs)	40.20 (26.38)	43.83 (27.81)	1.63

ผลการทดสอบสมมติฐานดังตารางที่ 26 เมื่อพิจารณาค่า  $t_{\text{คำนวณ}}$  และค่า Sig พบว่า

จากตารางพบว่าค่า  $t_{\text{คำนวณ}}$  เท่ากับ 1.63 มีค่าน้อยกว่า  $t_{\text{ตาราง}}$  ดังนั้นจะยอมรับ  $H_1$  ปฏิเสธ  $H_0$  ดังนั้นสรุปได้ว่าจำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอกจากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลส่งออกภายนอกจากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว แต่ไม่มีความแตกต่างอย่างเห็นได้ชัดเจน

สมมติฐานที่ 3 : จำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (EQs) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก (EQs) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

$$H_0 : FP(EQs)_{\text{หลัง} - \text{ก่อน}} \leq 0$$

$$H_1 : FP(EQs)_{\text{หลัง} - \text{ก่อน}} > 0$$

จากสมมติฐาน  $df=29$  ( $N = 30$ ) ซึ่งกำหนดระดับนัยสำคัญ (95% Confidence Interval) เท่ากับ 0.05 จะได้ค่าวิกฤต  $t=2.4620$  ( $t_{\text{ตาราง}} = 2.4620$ )

จะยอมรับ  $H_0$  เมื่อค่า  $t_{\text{คำนวณ}} > t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_1$

จะยอมรับ  $H_1$  เมื่อค่า  $t_{\text{คำนวณ}} < t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_0$

ผลการวิเคราะห์เพื่อทดสอบสมมติฐานทางสถิติ โดยการเปรียบเทียบจำนวนฟังก์ชันพอยต์ของการสอบถามภายนอก จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ดังตารางที่ 27

ตารางที่ 27 แสดงผลการทดสอบสมมติฐานทางสถิติของการสอบถามภายนอกที่ประเมินค่าความซับซ้อน

ประเภทฟังก์ชันพอยต์	จำนวนฟังก์ชันพอยต์		t
	รายละเอียดของซอฟต์แวร์ที่ออกแบบ	รายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว	
	Mean (SD)	Mean (SD)	
การสอบถามภายนอก (EQs)	4.93 (4.81)	5.87 (5.87)	2.18

ผลการทดสอบสมมติฐานดังตารางที่ 27 เมื่อพิจารณาค่า  $t_{\text{คำนวณ}}$  และค่า Sig พบว่า

จากตารางพบว่าค่า  $t_{\text{คำนวณ}}$  เท่ากับ 2.18 มีค่าน้อยกว่า  $t_{\text{ตาราง}}$  ดังนั้นจะยอมรับ  $H_1$  ปฏิเสธ  $H_0$  ดังนั้นสรุปได้ว่าจำนวนฟังก์ชันพอยต์ของข้อมูลการสอบถามภายนอกจากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลการสอบถามภายนอกจากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

สมมติฐานที่ 4 : จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (ILF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน (ILF's) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

$$H_0 : FP(ILF's)_{\text{หลัง} - \text{ก่อน}} \leq 0$$

$$H_1 : FP(ILF's)_{\text{หลัง} - \text{ก่อน}} > 0$$

จากสมมติฐาน  $df=29$  ( $N = 30$ ) ซึ่งกำหนดระดับนัยสำคัญ (95% Confidence Interval) เท่ากับ 0.05 จะได้ค่าวิกฤต  $t=2.4620$  ( $t_{\text{ตาราง}} = 2.4620$ )

จะยอมรับ  $H_0$  เมื่อค่า  $t_{\text{คำนวณ}} > t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_1$

จะยอมรับ  $H_1$  เมื่อค่า  $t_{\text{คำนวณ}} < t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_0$

ผลการวิเคราะห์เพื่อทดสอบสมมติฐานทางสถิติ โดยการเปรียบเทียบจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายใน จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ดังตารางที่ 28

ตารางที่ 28 แสดงผลการทดสอบสมมติฐานทางสถิติของแฟ้มข้อมูลภายในที่ประเมินค่าความซับซ้อน

ประเภทฟังก์ชันพอยต์	จำนวนฟังก์ชันพอยต์		t คำนวณ
	รายละเอียดของ ซอฟต์แวร์ที่ ออกแบบ	รายละเอียด ซอฟต์แวร์ที่สร้าง เสร็จแล้ว	
	Mean (SD)	Mean (SD)	
แฟ้มข้อมูลภายใน (ILF's)	98.60 (44.19)	108.37 (46.86)	2.35

ผลการทดสอบสมมติฐานดังตารางที่ 28 เมื่อพิจารณาค่า  $t_{\text{คำนวณ}}$  และค่า Sig พบว่า

จากตารางพบว่าค่า  $t_{\text{คำนวณ}}$  เท่ากับ 2.35 มีค่าน้อยกว่า  $t_{\text{ตาราง}}$  ดังนั้นจะยอมรับ  $H_1$  ปฏิเสธ  $H_0$  ดังนั้นสรุปได้ว่าจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลภายในจากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลแฟ้มข้อมูลภายในจากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

สมมติฐานที่ 5 : จำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (EIF's) จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก (EIF's) จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

$$H_0 : FP(EIF's)_{\text{หลัง} - \text{ก่อน}} \leq 0$$

$$H_1 : FP(EIF's)_{\text{หลัง} - \text{ก่อน}} > 0$$

จากสมมติฐาน  $df=29$  ( $N = 30$ ) ซึ่งกำหนดระดับนัยสำคัญ (95% Confidence Interval) เท่ากับ 0.05 จะได้ค่าวิกฤต  $t=2.4620$  ( $t_{\text{ตาราง}} = 2.4620$ )

จะยอมรับ  $H_0$  เมื่อค่า  $t_{\text{คำนวณ}} > t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_1$

จะยอมรับ  $H_1$  เมื่อค่า  $t_{\text{คำนวณ}} < t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_0$

ผลการวิเคราะห์เพื่อทดสอบสมมติฐานทางสถิติ โดยการเปรียบเทียบจำนวนฟังก์ชันพอยต์ของแฟ้มข้อมูลต่อประสานภายนอก จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ดังตารางที่ 29

ตารางที่ 29 แสดงผลการทดสอบสมมติฐานทางสถิติของแฟ้มข้อมูลต่อประสานภายนอกที่ประเมินค่าความซับซ้อน

ประเภทฟังก์ชันพอยต์	จำนวนฟังก์ชันพอยต์		t คำนวณ
	รายละเอียดของ ซอฟต์แวร์ที่ ออกแบบ	รายละเอียด ซอฟต์แวร์ที่สร้าง เสร็จแล้ว	
	Mean (SD)	Mean (SD)	
แฟ้ม ข้อมูล ต่อ ประสาน ภายนอก (EIF's)	2.17 (6.25)	3.17 (8.66)	0.84

ผลการทดสอบสมมติฐานดังตารางที่ 29 เมื่อพิจารณาค่า  $t_{\text{คำนวณ}}$  และค่า Sig พบว่า

จากตารางพบว่าค่า  $t_{\text{คำนวณ}}$  เท่ากับ 0.84 มีค่าน้อยกว่า  $t_{\text{ตาราง}}$  ดังนั้นจะยอมรับ  $H_1$  ปฏิเสธ  $H_0$  ดังนั้นสรุปได้ว่าจำนวนฟังก์ชันพอยต์ของข้อมูลแฟ้มข้อมูลต่อประสานภายนอกจากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้น้อยกว่าจำนวนฟังก์ชันพอยต์ของข้อมูลแฟ้มข้อมูลภายในจากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วจริง แต่ไม่มีความแตกต่างอย่างเห็นได้ชัดเจน



สมมติฐานที่ 6 : จำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์ จากรายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้น้อยกว่าจำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จ แล้ว ที่ประเมินค่าความซับซ้อนแล้ว

$$H_0 : FP(\text{ผลรวมทั้งหมด})_{\text{หลัง} - \text{ก่อน}} \leq 0$$

$$H_1 : FP(\text{ผลรวมทั้งหมด})_{\text{หลัง} - \text{ก่อน}} > 0$$

จากสมมติฐาน  $df=29$  ( $N = 30$ ) ซึ่งกำหนดระดับนัยสำคัญ (95% Confidence Interval) เท่ากับ 0.1 จะได้ค่าวิกฤต  $t=2.0452$  ( $t_{\text{ตาราง}} = 2.0452$ )

จะยอมรับ  $H_0$  เมื่อค่า  $t_{\text{คำนวณ}} > t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_1$

จะยอมรับ  $H_1$  เมื่อค่า  $t_{\text{คำนวณ}} < t_{\text{ตาราง}}$  และจะปฏิเสธ  $H_0$

ผลการวิเคราะห์เพื่อทดสอบสมมติฐานทางสถิติ โดยการเปรียบเทียบจำนวนฟังก์ชันพอยต์ของ จำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์ จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้และรายละเอียด ซอฟต์แวร์ที่สร้างเสร็จแล้ว ดังตารางที่ 30

ตารางที่ 30 แสดงผลการทดสอบสมมติฐานทางสถิติของผลรวมทั้งหมดของฟังก์ชันพอยต์ที่ ประเมินค่าความซับซ้อน

ประเภทฟังก์ชันพอยต์	จำนวนฟังก์ชันพอยต์		t
	รายละเอียดของ ซอฟต์แวร์ที่ ออกแบบ	รายละเอียด ซอฟต์แวร์ที่สร้าง เสร็จแล้ว	
	Mean (SD)	Mean (SD)	คำนวณ
จำนวนผลรวมทั้งหมดของ ฟังก์ชันพอยต์	198.93 (84.34)	222.97 (93.40)	3.80

ผลการทดสอบสมมติฐานดังตารางที่ 30 เมื่อพิจารณาค่า  $t_{\text{คำนวณ}}$  และค่า Sig พบว่า

จากตารางพบว่าค่า  $t_{\text{คำนวณ}}$  เท่ากับ 3.80 มีค่ามากกว่า  $t_{\text{ตาราง}}$  ดังนั้นจะยอมรับ  $H_0$  ปฏิเสธ  $H_1$  ดังนั้นจะพบว่าจำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ไม่ แตกต่างจำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

ตารางที่ 31 แสดงผลการทดสอบสมมติฐานทั้ง 6 สมมติฐาน

ประเภทฟังก์ชันพอยต์	จำนวนฟังก์ชันพอยต์		t คำนวณ
	รายละเอียดของ ซอฟต์แวร์ที่ ออกแบบ Mean (SD)	รายละเอียด ซอฟต์แวร์ที่สร้าง เสร็จแล้ว Mean (SD)	
ข้อมูลนำเข้าจากภายนอก (EIs)	53.03 (25.47)	61.73 (40.32)	1.70
ข้อมูลส่งออกภายนอก (EOs)	40.20 (26.38)	43.83 (27.81)	1.63
การสอบถามภายนอก (EQs)	4.93 (4.81)	5.87 (5.87)	2.18
เพิ่มข้อมูลภายใน (ILF's)	98.60 (44.19)	108.37 (46.86)	2.35
เพิ่มข้อมูลต่อประสานภายนอก (EIF's)	2.17 (6.25)	3.17 (8.66)	0.84
จำนวนผลรวมทั้งหมดของ ฟังก์ชันพอยต์	198.93 (84.34)	222.97 (93.40)	3.80

จากตารางที่ 31 ผลจากการทดสอบสมมติฐาน พบว่า ประเภทของฟังก์ชันพอยต์ทั้ง 5 ประเภท มีจำนวนฟังก์ชันพอยต์จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้มีค่าน้อยกว่าจำนวนฟังก์ชันพอยต์จากรายละเอียดที่สร้างเสร็จแล้วทั้ง 5 ประเภทฟังก์ชันพอยต์

โดยพบว่า ประเภทฟังก์ชันพอยต์ส่วนที่เป็นเพิ่มข้อมูลภายใน การสอบถามภายนอก ข้อมูลนำเข้าภายนอก ข้อมูลส่งออกภายนอกมีค่าฟังก์ชันพอยต์ และเพิ่มข้อมูลต่อประสานภายนอกมีค่าฟังก์ชันพอยต์จากรายละเอียดที่สร้างเสร็จแล้วมากกว่าจำนวนฟังก์ชันพอยต์ของรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้

5.3.2 การวิเคราะห์เปอร์เซ็นต์การเปลี่ยนแปลงของฟังก์ชันพอยต์ของรายละเอียดของซอฟต์แวร์ที่ ออกแบบไว้กับรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว

5.3.2.1 วิเคราะห์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์

ผู้วิจัยได้นำข้อมูลผลการคำนวณค่าฟังก์ชันพอยต์มาคำนวณค่าเปอร์เซ็นต์สัดส่วนผลต่างของ การคำนวณค่าฟังก์ชันพอยต์ของหน่วยตัวอย่างจำนวน 30 ชุดข้อมูล โดยคำนวณจากสูตร

$$X_i = \frac{X_2 - X_1}{X_1} \times 100$$

โดยที่  $X_i$  คือ สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์ของหน่วยตัวอย่าง

$X_1$  คือ ผลการคำนวณค่าฟังก์ชันพอยต์ของรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้

$X_2$  คือ ผลการคำนวณค่าฟังก์ชันพอยต์ของรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว

ผลการคำนวณสัดส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์แยกตามแต่ละประเภท ของหน่วยตัวอย่างจำนวน 30 ชุดข้อมูลดังนี้

ตารางที่ 32 แสดงสัดส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลนำเข้าภายนอก

โครงการพัฒนาซอฟต์แวร์	ข้อมูลนำเข้าภายนอก (EIs)		เปอร์เซ็นต์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์
	รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว	
PJ15	73	223	205.48
PJ5	30	52	73.33
PJ16	21	33	57.14
PJ6	21	30	42.86
PJ13	72	96	33.33
PJ9	84	111	32.14
PJ29	45	58	28.89
PJ1	72	81	12.50
PJ30	51	57	11.76
PJ32	36	39	8.33
PJ24	49	53	8.16
PJ33	75	80	6.67
PJ12	58	61	5.17
PJ37	66	69	4.55
PJ4	51	53	3.92
PJ2	27	27	0.00
PJ10	24	24	0.00

โครงการพัฒนาซอฟต์แวร์	ข้อมูลนำเข้าภายนอก (EIs)		เปอร์เซ็นต์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์
	รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว	
PJ11	48	48	0.00
PJ21	33	33	0.00
PJ26	63	63	0.00
PJ28	141	141	0.00
PJ31	39	39	0.00
PJ36	60	60	0.00
PJ17	57	56	-1.75
PJ3	66	63	-4.55
PJ7	66	63	-4.55
PJ8	44	41	-6.82
PJ34	39	36	-7.69
PJ19	51	45	-11.76
PJ35	84	72	-14.29
ค่าเฉลี่ย	54.87	63.57	16.09

จากตารางที่ 32 พบว่าจำนวนโครงการพัฒนาซอฟต์แวร์ที่มีค่าการคำนวณฟังก์ชันพอยต์ซ์ส่วนที่เป็นข้อมูลนำเข้าภายนอก (EIs) มีจำนวนเพิ่มขึ้น 15 โครงการพัฒนา ลดลง 7 โครงการพัฒนาซอฟต์แวร์ และมีจำนวนเท่ากัน 8 โครงการพัฒนาซอฟต์แวร์

ตารางที่ 33 แสดงสัดส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์ซ์ของข้อมูลนำเข้าภายนอก

โครงการพัฒนาซอฟต์แวร์	ข้อมูลนำเข้าภายนอก (EOs)		เปอร์เซ็นต์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์ซ์
	รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว	
PJ5	24	56	133.33
PJ29	29	61	110.34
PJ6	20	40	100.00
PJ30	32	44	37.50
PJ2	28	36	28.57
PJ34	28	36	28.57
PJ9	88	112	27.27

โครงการพัฒนาซอฟต์แวร์	ข้อมูลนำเข้าภายนอก (EOs)		เปอร์เซ็นต์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์
	รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว	
PJ24	37	46	24.32
PJ12	20	24	20.00
PJ32	20	24	20.00
PJ37	20	24	20.00
PJ17	24	27	12.50
PJ19	44	48	9.09
PJ33	60	64	6.67
PJ1	68	72	5.88
PJ3	24	24	0.00
PJ4	48	48	0.00
PJ8	8	8	0.00
PJ10	20	20	0.00
PJ11	52	52	0.00
PJ16	8	8	0.00
PJ21	48	48	0.00
PJ26	112	112	0.00
PJ28	100	100	0.00
PJ36	40	40	0.00
PJ13	72	64	-11.11
PJ7	40	28	-30.00
PJ31	24	16	-33.33
PJ35	12	8	-33.33
PJ15	56	25	-55.36
ค่าเฉลี่ย	40.20	43.83	14.03

จากตารางที่ 33 พบว่าจำนวนโครงการพัฒนาซอฟต์แวร์ที่มีค่าการคำนวณฟังก์ชันพอยต์มีส่วนที่เป็นข้อมูลนำเข้าภายนอก (EOs) มีจำนวนเพิ่มขึ้น 15 โครงการพัฒนา ลดลง 5 โครงการพัฒนาซอฟต์แวร์ และมีจำนวนเท่ากัน 10 โครงการพัฒนาซอฟต์แวร์

ตารางที่ 34 แสดงสัดส่วนของผลต่างของของการคำนวณค่าฟังก์ชันพอยต์ของข้อมูลการสอบถามภายนอก

โครงการพัฒนาซอฟต์แวร์	ข้อมูลการสอบถามภายนอก (EQs)		เปอร์เซ็นต์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์
	รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว	
PJ4	0	4	-
PJ11	6	13	116.67
PJ28	12	21	75.00
PJ10	6	9	50.00
PJ29	9	12	33.33
PJ35	9	12	33.33
PJ15	13	15	15.38
PJ1	9	9	0.00
PJ2	15	15	0.00
PJ3	3	3	0.00
PJ7	9	9	0.00
PJ16	3	3	0.00
PJ17	9	9	0.00
PJ24	6	6	0.00
PJ31	6	6	0.00
PJ34	9	9	0.00
PJ36	6	6	0.00
PJ37	6	6	0.00
PJ9	9	6	-33.33
ค่าเฉลี่ย	7.63	9.11	16.13

จากตารางที่ 34 พบว่าจำนวนโครงการพัฒนาซอฟต์แวร์มีข้อมูลการสอบถามภายนอก (EQs) ทั้งหมด 19 โครงการพัฒนาซอฟต์แวร์ โดยที่มีค่าการคำนวณฟังก์ชันพอยต์ข้อส่วนที่เป็นข้อมูลการสอบถามภายนอก (EQs) มีจำนวนเพิ่มขึ้น 7 โครงการพัฒนาซอฟต์แวร์ ลดลง 1 โครงการพัฒนาซอฟต์แวร์ และมีจำนวนเท่ากัน 11 โครงการพัฒนาซอฟต์แวร์ อีกทั้งพบว่ามี 1 โครงการพัฒนาซอฟต์แวร์ที่รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ไม่มีข้อมูลการสอบถามภายนอก (EQs) แต่รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้วมีส่วนที่เป็นข้อมูลการสอบถามภายนอก (EQs)

ตารางที่ 35 แสดงสัดส่วนของผลต่างของของการคำนวณค่าฟังก์ชันพอยต์ของเพิ่มข้อมูลภายใน

โครงการพัฒนาซอฟต์แวร์	ข้อมูลเพิ่มข้อมูลภายใน (ILF's)		เปอร์เซ็นต์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์
	รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว	
PJ9	112	196	75.00
PJ1	112	182	62.50
PJ34	28	42	50.00
PJ29	63	90	42.86
PJ21	42	56	33.33
PJ16	49	63	28.57
PJ35	105	133	26.67
PJ37	84	105	25.00
PJ36	63	77	22.22
PJ30	112	133	18.75
PJ2	56	63	12.50
PJ7	168	189	12.50
PJ13	56	63	12.50
PJ33	70	77	10.00
PJ17	112	122	8.93
PJ24	117	124	5.98
PJ12	119	126	5.88
PJ4	84	84	0.00
PJ6	42	42	0.00
PJ8	112	112	0.00
PJ10	35	35	0.00
PJ11	133	133	0.00
PJ19	154	154	0.00
PJ28	210	210	0.00
PJ31	105	105	0.00
PJ32	98	98	0.00
PJ15	132	122	-7.58
PJ3	105	91	-13.33
PJ26	182	147	-19.23
PJ5	98	77	-21.43

โครงการพัฒนาซอฟต์แวร์	ข้อมูลเพิ่มข้อมูลภายใน (ILF's)		เปอร์เซ็นต์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์
	รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว	
ค่าเฉลี่ย	98.60	108.37	13.05

จากตารางที่ 35 พบว่าจำนวนโครงการพัฒนาซอฟต์แวร์ที่มีค่าการคำนวณฟังก์ชันพอยต์ส่วนที่เป็นข้อมูลเพิ่มข้อมูลภายใน (ILF's) มีจำนวนเพิ่มขึ้น 17 โครงการพัฒนา ลดลง 4 โครงการพัฒนาซอฟต์แวร์ และมีจำนวนเท่ากัน 9 โครงการพัฒนาซอฟต์แวร์

ตารางที่ 36 แสดงอัตราส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์ของเพิ่มข้อมูลต่อประสานภายนอก

โครงการพัฒนาซอฟต์แวร์	เพิ่มข้อมูลต่อประสานภายนอก (EIF's)		เปอร์เซ็นต์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์
	รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว	
PJ8	0	35	-
PJ7	15	15	0
PJ11	30	30	0
PJ15	10	10	0
PJ16	5	5	0
PJ13	5	0	-100

จากตารางที่ 36 พบว่าจำนวนโครงการพัฒนาซอฟต์แวร์มีเพิ่มข้อมูลต่อประสานภายนอก (EIF's) ทั้งหมด 5 โครงการพัฒนาซอฟต์แวร์ โดยพบว่ามี 1 โครงการพัฒนาซอฟต์แวร์ที่รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ไม่มีข้อมูลการสอบถามภายนอก (EQs) และอีก 1 โครงการพัฒนาซอฟต์แวร์ที่รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้วไม่มีส่วนที่เป็นข้อมูลการสอบถามภายนอก (EQs)



ตารางที่ 37 แสดงสัดส่วนของผลต่างของการคำนวณค่าฟังก์ชันพอยต์ผลรวมทั้งหมด

โครงการ พัฒนา ซอฟต์แวร์	ค่าฟังก์ชันพอยต์ผลรวมทั้งหมด		เปอร์เซ็นต์สัดส่วน ผลต่างของการคำนวณ ค่าฟังก์ชันพอยต์
	รายละเอียดของ ซอฟต์แวร์ที่ออกแบบ ไว้	รายละเอียดของ ซอฟต์แวร์ที่สร้างเสร็จ แล้ว	
PJ29	146	221	51.37
PJ9	293	425	45.05
PJ15	284	395	39.08
PJ6	83	112	34.94
PJ1	261	344	31.80
PJ16	86	112	30.23
PJ5	152	185	21.71
PJ30	195	234	20.00
PJ8	164	196	19.51
PJ34	104	123	18.27
PJ37	176	204	15.91
PJ2	126	141	11.90
PJ21	123	137	11.38
PJ24	209	229	9.57
PJ13	205	223	8.78
PJ36	169	183	8.28
PJ33	205	221	7.80
PJ35	210	225	7.14
PJ12	197	211	7.11
PJ17	202	214	5.94
PJ32	154	161	4.55
PJ10	85	88	3.53
PJ4	183	189	3.28
PJ11	269	276	2.60
PJ7	298	304	2.01
PJ28	463	472	1.94
PJ19	249	247	-0.80
PJ31	174	166	-4.60

โครงการพัฒนาซอฟต์แวร์	ค่าฟังก์ชันพอยต์ผลรวมทั้งหมด		เปอร์เซ็นต์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์
	รายละเอียดของซอฟต์แวร์ที่ออกแบบไว้	รายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้ว	
PJ3	198	181	-8.59
PJ26	357	322	-9.80
ค่าเฉลี่ย	200.67	224.7	13.33

จากตารางที่ 37 พบว่าจำนวนโครงการพัฒนาซอฟต์แวร์ที่มีค่าการคำนวณฟังก์ชันพอยต์ส่วนที่เป็นผลรวมทั้งหมด มีจำนวนเพิ่มขึ้น 26 โครงการพัฒนาซอฟต์แวร์ ลดลง 4 โครงการพัฒนาซอฟต์แวร์

ตารางที่ 38 แสดงเปอร์เซ็นต์ค่าสูงสุด ค่าต่ำสุด และค่าเฉลี่ยของค่าสัดส่วนผลต่างของการคำนวณฟังก์ชันพอยต์

ประเภทฟังก์ชันพอยต์	เปอร์เซ็นต์สัดส่วนผลต่างสูงสุด (MAX)	เปอร์เซ็นต์สัดส่วนผลต่างน้อยสุด (Min)	เปอร์เซ็นต์เฉลี่ยสัดส่วนผลต่าง
ข้อมูลนำเข้าภายนอก (EIs)	205.48	-14.29	16.09
ข้อมูลส่งออกภายนอก (Eos)	133.33	-55.36	14.03
ข้อมูลการสอบถามภายนอก (EQs)	116.67	-33.33	16.13
แฟ้มข้อมูลภายใน (ILFs)	75	-21.43	13.05
แฟ้มข้อมูลต่อประสานภายนอก (EIFs)	0	-100	-20
ผลรวมทั้งหมด	51.37	-9.80	13.33

จากตารางที่ 38 จะเห็นได้ว่าค่าเปอร์เซ็นต์เฉลี่ยสัดส่วนผลต่างของข้อมูลส่วนที่เป็นข้อมูลนำเข้าภายนอก ข้อมูลส่งออกภายนอก ข้อมูลการสอบถามภายนอกและเพิ่มข้อมูลภายใน มีค่าเปอร์เซ็นต์เฉลี่ยสัดส่วนผลต่างของแต่ละประเภทฟังก์ชันพอยต์ มีความแตกต่างที่ระดับใกล้เคียงกัน ยกเว้นข้อมูลเพิ่มข้อมูลต่อประสานภายนอกที่พบว่าค่าเปอร์เซ็นต์เฉลี่ยสัดส่วนผลต่างที่มีค่าติดลบ

5.3.2.2 การเปรียบเทียบผลรวมจำนวนฟังก์ชันพอยต์เฉพาะแผนภาพระดับที่หนึ่งเปรียบเทียบกับผลรวมจำนวนฟังก์ชันพอยต์ของทุกระดับแผนภาพ

ผู้วิจัยได้นำข้อมูลผลการคำนวณค่าฟังก์ชันพอยต์มาคำนวณเปรียบเทียบผลรวมฟังก์ชันพอยต์เฉพาะแผนภาพระดับที่หนึ่งเปรียบเทียบกับผลรวมจำนวนฟังก์ชันพอยต์ของทุกระดับแผนภาพโดยคำนวณจากสูตร

$$X_i = \frac{X_1 - X_2}{X_1} \times 100$$

โดยที่  $X_i$  คือ เปอร์เซ็นผลต่างระหว่างผลรวมฟังก์ชันพอยต์เฉพาะแผนภาพระดับที่หนึ่งกับผลรวมจำนวนฟังก์ชันพอยต์ของทุกระดับแผนภาพ

$X_1$  คือ ผลรวมจำนวนฟังก์ชันพอยต์ของทุกระดับแผนภาพ

$X_2$  คือ ผลรวมฟังก์ชันพอยต์เฉพาะแผนภาพระดับที่หนึ่ง

ตารางที่ 39 แสดงผลต่างผลรวมฟังก์ชันพอยต์เฉพาะแผนภาพระดับที่หนึ่งกับผลรวมจำนวนฟังก์ชันพอยต์ของทุกระดับแผนภาพ

โครงการพัฒนาซอฟต์แวร์	ผลรวมจำนวนฟังก์ชันพอยต์ของทุกระดับแผนภาพ	ผลรวมฟังก์ชันพอยต์เฉพาะแผนภาพระดับที่หนึ่ง	เปอร์เซ็นต์ค่าเฉลี่ยผลต่างผลรวมฟังก์ชันพอยต์
PJ19	247	191	29.32
PJ8	196	161	21.74
PJ11	276	234	17.95
PJ37	204	176	15.91
PJ32	161	140	15.00
PJ24	229	201	13.93
PJ3	181	160	13.13
PJ26	322	287	12.20
PJ2	141	127	11.02
PJ9	425	390	8.97
PJ1	344	316	8.86
PJ5	185	171	8.19

โครงการพัฒนาซอฟต์แวร์	ผลรวมจำนวนฟังก์ชันพอยต์ของทุกระดับแผนภาพ	ผลรวมฟังก์ชันพอยต์เฉพาะแผนภาพระดับที่หนึ่ง	เปอร์เซ็นต์ค่าเฉลี่ยผลต่างผลรวมฟังก์ชันพอยต์
PJ17	214	200	7.00
PJ35	225	211	6.64
PJ34	123	116	6.03
PJ21	137	130	5.38
PJ7	304	290	4.83
PJ31	166	159	4.40
PJ36	183	176	3.98
PJ13	223	216	3.24
PJ30	234	227	3.08
PJ15	395	388	1.80
PJ4	189	189	0.00
PJ6	112	112	0.00
PJ10	88	88	0.00
PJ12	211	211	0.00
PJ16	112	112	0.00
PJ28	472	472	0.00
PJ29	221	221	0.00
PJ33	221	221	0.00
ค่าเฉลี่ย	224.7	209.77	7.42

จากตารางที่ 39 พบว่าโครงการพัฒนาซอฟต์แวร์จำนวน 22 โครงการพัฒนาซอฟต์แวร์ที่ผลรวมจำนวนฟังก์ชันพอยต์ของทุกระดับแผนภาพมากกว่าผลรวมฟังก์ชันพอยต์เฉพาะแผนภาพระดับที่หนึ่ง โดยอีก 8 โครงการพัฒนาซอฟต์แวร์ไม่มีส่วนของฟังก์ชันพอยต์เพิ่มจากระดับแผนภาพที่หนึ่ง โดยพบว่าเปอร์เซ็นต์ผลต่างผลรวมฟังก์ชันพอยต์ของโครงการพัฒนาซอฟต์แวร์ที่มีค่ามากที่สุดเท่ากับ 29.32 เปอร์เซ็นต์ โดยค่าเฉลี่ยของเปอร์เซ็นต์ผลต่างของผลรวมฟังก์ชันพอยต์ประมาณเท่ากับ 7.42 เปอร์เซ็นต์

#### 5.4 อภิปรายผลการวิจัย

จากตารางที่ 31 ผลการทดสอบสมมติฐานของการวิจัย พบว่าทุกประเภทฟังก์ชันพอยต์มีค่าฟังก์ชันพอยต์จากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้น้อยกว่ารายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วที่ระดับความเชื่อมั่น 95 เปอร์เซ็นต์ ซึ่งสอดคล้องกับการวิเคราะห์เปอร์เซ็นต์การเปลี่ยนแปลงของฟังก์ชันพอยต์ของรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้กับรายละเอียดของซอฟต์แวร์ที่สร้างเสร็จแล้วดังตารางที่ 38 การวิเคราะห์สัดส่วนผลต่างของการคำนวณค่าฟังก์ชันพอยต์ ที่พบว่าทุกประเภท

ฟังก์ชันพอยต์มีค่าสัดส่วนฟังก์ชันพอยต์จากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้มีค่าน้อยกว่า รายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว และจากตารางที่ 39 การเปรียบเทียบผลรวมจำนวนฟังก์ชันพอยต์เฉพาะแผนภาพระดับที่หนึ่งเปรียบเทียบกับผลรวมจำนวนฟังก์ชันพอยต์ของทุกระดับแผนภาพ จะพบว่าเปอร์เซ็นต์ค่าเฉลี่ยผลต่างผลรวมฟังก์ชันพอยต์เท่ากับ 7.42 เปอร์เซ็นต์

จะเห็นได้ว่าส่วนที่เป็นข้อมูลนำเข้าภายนอกและส่วนที่เป็นแฟ้มข้อมูลภายในจะเพิ่มมากขึ้น เนื่องจากด้วยส่วนที่เป็นข้อมูลนำเข้าภายนอกจะมีการถูกบันทึก หรือเก็บไว้ในแฟ้มข้อมูลภายในเสมอ ทำให้ทั้งสองส่วนนี้เมื่อมีส่วนใดส่วนหนึ่งเพิ่มขึ้นจะทำให้ค่าส่วนเพิ่มขึ้นด้วย และส่วนที่เป็นการสอบถามภายนอกที่มีค่าเพิ่มขึ้นมากขึ้นซึ่งสามารถบอกได้ว่า ความต้องการเกี่ยวกับการประมวลผล ข้อมูลที่ผ่านเงื่อนไขความต้องการของผู้ใช้งานเพื่อแสดงผล โดยความต้องการใช้งานต่อจากข้อมูลส่วนนี้จะมีค่าเพิ่มมากขึ้นระหว่างการพัฒนาซอฟต์แวร์หรือระบบ ซึ่งเปอร์เซ็นต์เฉลี่ยสัดส่วนผลรวมจำนวนฟังก์ชันพอยต์ที่เพิ่มขึ้นมีค่าเท่ากับ 13.33 เปอร์เซ็นต์

จากผลการวิเคราะห์สัดส่วนเปอร์เซ็นต์ค่าผลต่างของผลรวมฟังก์ชันพอยต์ ที่มีค่าสัดส่วนความแตกต่างที่เพิ่มขึ้นในระดับที่เท่าๆ กัน อาจจะเป็นเพราะหน่วยตัวอย่างของโครงการพัฒนาซอฟต์แวร์ที่นำมาใช้งานนั้นในส่วนของโครงการพิเศษ 1 ที่เป็นข้อมูลของรายละเอียดซอฟต์แวร์ที่ออกแบบไว้ นั้นจะมีการปรับปรุงแก้ไขฟังก์ชันการทำงานของระบบหรือซอฟต์แวร์โดยผู้เชี่ยวชาญ เพื่อให้ฟังก์ชันการทำงานของซอฟต์แวร์ให้มีความใกล้เคียงกับความเป็นจริงมากที่สุดก่อนการพัฒนาหรือซอฟต์แวร์ ซึ่งจะเห็นได้จากในแต่ละประเภทฟังก์ชันนั้นจะมีจำนวนข้อมูลจากโครงการพิเศษ 1 และโครงการพิเศษ 2 ที่มีค่าจำนวนฟังก์ชันการทำงานที่เท่ากันหรือไม่มีความแตกต่างที่เพิ่มมากขึ้นอย่างเห็นได้ชัดเจน ดังนั้นทำให้ผลการวิจัยออกมาที่มีค่าของจำนวนฟังก์ชันเพิ่มมากขึ้นจริง แต่ไม่มีความแตกต่างที่เกิดขึ้นอย่างชัดเจน

## บทที่ 6

### ผลการวิจัย

#### 6.1 ผลการวิจัย

งานวิจัยนี้เปรียบเทียบขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดซอฟต์แวร์ที่ออกแบบไว้เปรียบเทียบกับขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว โดยใช้วิธีการวิเคราะห์ฟังก์ชันพอยต์จากแผนภาพข้อมูลกระแสที่อยู่ในโครงสร้างไฟล์ XML โดยใช้แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูลและพจนานุกรมข้อมูล ที่ปรากฏในรายละเอียดซอฟต์แวร์ จำนวน 30 ชุดข้อมูล ผลการวิจัยพบว่า จำนวนฟังก์ชันพอยต์ในแต่ละประเภทฟังก์ชันพอยต์จากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้มีน้อยกว่าจำนวนฟังก์ชันพอยต์จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว หรือกล่าวได้ว่าจำนวนฟังก์ชันพอยต์ในแต่ละประเภทฟังก์ชันพอยต์มีจำนวนเพิ่มมากขึ้นจากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้ ซึ่งสอดคล้องกับคำกล่าวของ Sommerville ที่กล่าวไว้ว่า เมื่อการสร้างซอฟต์แวร์นั้นเสร็จสิ้น แบบของซอฟต์แวร์จริงมักเปลี่ยนไปจากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้ที่เขียนไว้ในเบื้องต้นมาก ซึ่งผลการวิจัยมีความสอดคล้องกัน ซึ่งพบว่ารายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้วมีค่าเปลี่ยนไปจากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้ อีกทั้งผลการวิจัยยังสอดคล้องกับงานวิจัยของ Standish Group International ที่พบโครงการพัฒนาซอฟต์แวร์ทั้งหมดใช้งบประมาณมากกว่างบประมาณเดิมที่วางไว้ และงานวิจัย Verner และคณะพบว่ามีการพัฒนาซอฟต์แวร์ 65 เปอร์เซ็นต์ ที่มีการประมาณเวลาที่ต่ำกว่าความเป็นจริง

โดยพบว่าจำนวนฟังก์ชันพอยต์ในทุกประเภทข้อมูลมีค่าเพิ่มขึ้น ส่วนที่เป็นแฟ้มข้อมูลภายใน (ILF's) การสอบถามภายนอก (EQs) ข้อมูลส่งออกภายนอก (EOs) แฟ้มข้อมูลต่อประสานภายนอก (EIF's) และส่วนที่เป็นประเภทข้อมูลนำเข้าภายนอก (EIs) มีจำนวนฟังก์ชันพอยต์จากรายละเอียดของซอฟต์แวร์ที่ออกแบบไว้มีน้อยกว่าฟังก์ชันพอยต์จากรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว โดยเปอร์เซ็นต์สัดส่วนผลต่างของจำนวนผลรวมทั้งหมดของฟังก์ชันพอยต์มีค่าเท่ากับ 13.33 เปอร์เซ็นต์ และมีค่าเฉลี่ยเปอร์เซ็นต์สัดส่วนผลต่างในแต่ละประเภทที่ระดับเท่าๆ กัน อีกทั้งยังพบว่าการคำนวณฟังก์ชันพอยต์แผนภาพระดับที่หนึ่งนั้น พบว่าเมื่อเปรียบกับการคำนวณค่าฟังก์ชันพอยต์จากทุกระดับแผนภาพจะพบว่ามีค่าเปอร์เซ็นต์ผลรวมเพิ่มขึ้น 7.42 เปอร์เซ็นต์

## 6.2 การนำงานวิจัยไปใช้

ข้อค้นพบในการศึกษานี้ สามารถนำไปต่อยอดในการประมาณขนาดซอฟต์แวร์ดังรายละเอียดต่อไปนี้

การประมาณขนาดซอฟต์แวร์ด้วยวิธีการวิเคราะห์ฟังก์ชันพอยต์จากแผนภาพ กระแสข้อมูล แผนภาพแสดงความสัมพันธ์กลุ่มข้อมูล และพจนานุกรมข้อมูล จากรายละเอียดซอฟต์แวร์ที่ออกแบบไว้เทียบกับขนาดของซอฟต์แวร์ที่ปรากฏอยู่ในรายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว ผลการทดสอบพบว่ามีความสัมพันธ์เพิ่มขึ้นในแต่ละประเภทฟังก์ชันพอยต์ ทั้งก่อนประเมินค่าความซับซ้อนและได้รับการประเมินค่าความซับซ้อนแล้ว แต่ในส่วนของข้อมูลการสอบภายนอก หากสามารถรู้ได้ว่าการแสดงผลของข้อมูลที่ต่างกันของแพลตฟอร์มอาจจะส่งผลให้เกิดงานที่เพิ่มมากขึ้นเล็กน้อยเพียงใด

เนื่องจากในปัจจุบันจะเห็นได้ว่า จะมีแพลตฟอร์มต่างๆ มากมายที่รองรับผลการแสดงข้อมูลจากการสอบถามภายนอก ไม่ว่าจะเป็นมือถือหรือข้อความ SMS เป็นต้น ซึ่งจากความต่างกันของแต่ละแพลตฟอร์มที่ใช้ในการแสดงผล ก็จะส่งผลต่อการพัฒนาซอฟต์แวร์หรือระบบที่แตกต่างกัน ทำให้ฟังก์ชันการทำงานในส่วนนี้หากมีการคำนวณค่าโดยใช้แผนภาพกระแสข้อมูลที่ไม่สามารถแสดงได้ว่าแพลตฟอร์มที่ใช้ในการแสดงผลเป็นแบบใด ก็อาจจะส่งผลต่อการคำนวณขนาดซอฟต์แวร์ที่ผิดพลาดเกิดขึ้นได้ ดังนั้นการประมาณค่าฟังก์ชันในส่วนนี้ควรมีการประมาณค่าที่จะเกิดขึ้นให้มีค่าใกล้เคียงมากที่สุด

สำหรับเครื่องมือที่พัฒนาสามารถช่วยในการนำไปใช้ในการประมาณขนาดซอฟต์แวร์ที่จะพัฒนาโดยใช้แผนภาพกระแสข้อมูลที่สร้างขึ้นมาจากเครื่องมือ Microsoft Visio ได้ เพื่อใช้ในการประมาณขนาดของซอฟต์แวร์ที่จะพัฒนา ช่วยลดความผิดพลาดในการคำนวณและระยะเวลา หรือลดความผิดพลาดที่เกิดขึ้นจากการคาดการณ์ของผู้เชี่ยวชาญ ซึ่งเครื่องมือจะช่วยลดความผิดพลาดดังกล่าวให้น้อยลง และสามารถนำแนวทางการวิเคราะห์รูปแบบโครงสร้างไฟล์ XML ของแผนภาพกระแสข้อมูลที่ได้จากเครื่องมือ Microsoft Visio ไปใช้ในการสร้างเครื่องมือหรือพัฒนาต่อยอดการคำนวณค่าฟังก์ชันพอยต์จากรูปแบบแผนภาพอื่นๆ ได้

## 6.3 ข้อจำกัดงานวิจัย

เนื่องจากงานวิจัยนี้เก็บรวบรวมข้อมูลโครงการพัฒนาซอฟต์แวร์ จากโครงการพิเศษระดับปริญญาโท สาขาการพัฒนาซอฟต์แวร์ ทำให้เห็นว่าผลการวิจัยที่ได้มีค่าความแตกต่างที่ไม่ชัดเจน แต่ก็สามารถเห็นได้ว่าในแต่ละประเภทฟังก์ชันมีส่วนการเพิ่มขึ้นเท่าไร ซึ่งในการพัฒนาระบบหรือซอฟต์แวร์ในการทำงานจริง การเก็บรวบรวมข้อมูลความต้องการในการพัฒนากับข้อมูลหลังการพัฒนาระบบจริงนั้นจะมีค่าความแตกต่างที่เห็นได้อย่างชัดเจนมากกว่านี้ อีกทั้งเครื่องมือที่พัฒนาขึ้นสามารถรองรับได้แค่แผนภาพกระแสข้อมูลที่อยู่ในรูปแบบโครงสร้าง XML ไฟล์เท่านั้น

## รายการอ้างอิง

- กัลยา วานิชย์บัญชา. (2551). หลักสถิติ.
- ชมพูนุท เผ่าประพัทธ์. (2555). การเปรียบเทียบความแม่นยำของวิธีการประมาณขนาดซอฟต์แวร์เชิงวัตถุด้วยวิธีฟังก์ชันพอยต์ ฟังก์ชันพอยต์เชิงวัตถุ และคลาสพอยต์. วิทยานิพนธ์ปริญญา มหาบัณฑิต จุฬาลงกรณ์มหาวิทยาลัย.
- ณัฐพันธ์ เขจรนันท์. (2551). การวิเคราะห์และออกแบบระบบสารสนเทศ.
- ต่อพงศ์ พรหมภูเบศ. (2555). ซอฟต์แวร์สำหรับประมาณฟังก์ชันพอยต์ ค่าความพยายาม และ ค่าใช้จ่ายในโครงการพัฒนาซอฟต์แวร์ จากแผนภาพกระแสข้อมูล. โครงการปริญญาโท วิทยาศาสตร์มหาบัณฑิต สาขาการพัฒนาระบบซอฟต์แวร์ด้านธุรกิจ.
- รุจิจันทร์ พิริยะสงวนพงศ์. (2549). สารสนเทศทางธุรกิจ.
- สมศักดิ์ โชคชัยชุกุล. (2553). คู่มือการออกแบบระบบงานฐานข้อมูล
- Albrecht, A. J. (1979). Measuring Application Development Productivity. *IBM Application Development Symposium*.
- Biafore, B. (2007). Visio 2007 Bible.
- DeMarco. (1979). Structured analysis and system specification. *Yourdon Press*, 409.
- Donald, S. L. V., Jr. (2000). Understanding Data Flow Diagram. *In Proceedings of the 47th annual conference on Society for Technical Communication*.
- E. Lamma, P. Mello, & Fabrizio, R. (2003). A System for Measuring Function Points from ER-DFD Specification. *The Computer Journal*, 47(3).
- G. B. Shelly, & Rosenblatt, H. J. (2011). Systems analysis and design. *Course Technology*.
- Gane, & Sarson. (1979). Structured systems analysis: tools and techniques. *Prentice Hall Professional Technical*.
- Group, I. F. P. U. (1999). Function Points Counting Practices Manual, Version 4.1.1. *International Function Points User Group*.



- Group, I. F. P. U. (2004). Function Point Counting Practices Manual *International Function Points User Group*(Release 4.2).
- Humphrey, & S., W. (2005). Why big software projects fail: The 12 key questions.
- J. M. Verner, W. M. Evanco, & Cerpa, N. (2007). State of the practice: An exploratory analysis of schedule estimation and software project success prediction. *Information and Software Technology Elsevier Science Inc.*
- Linberg, K. R. (1999). Software developer perceptions about software project failure: a case study. *Systems and Software* 49.
- Longstreet, D. (2004). Function Points Analysis Training Course. *SoftwareMetrics.com*.
- Luigi Buglione, Cigdem Gencel, & Efe, P. (2006). Suggestions for Improving Measurement Plans : A BMP application in Turkey. *Suggestions for Improving Measurement Plans*.
- Pressman, R. S. (2001). Software Engineering : a practitioner's approach. *McGraw-Hill*.
- Rask, R. (1991). Algorithms for Counting Unadjusted Function Points from Dataflow Diagrams. *Technical Report*.
- Sommerville, I. (2001). Software Engineering. *International Computer Science Series*.
- Yourdon, & Constantine. (1979). Structured design: Fundamentals of a discipline of computer program and systems design. *Prentice-Hall*.

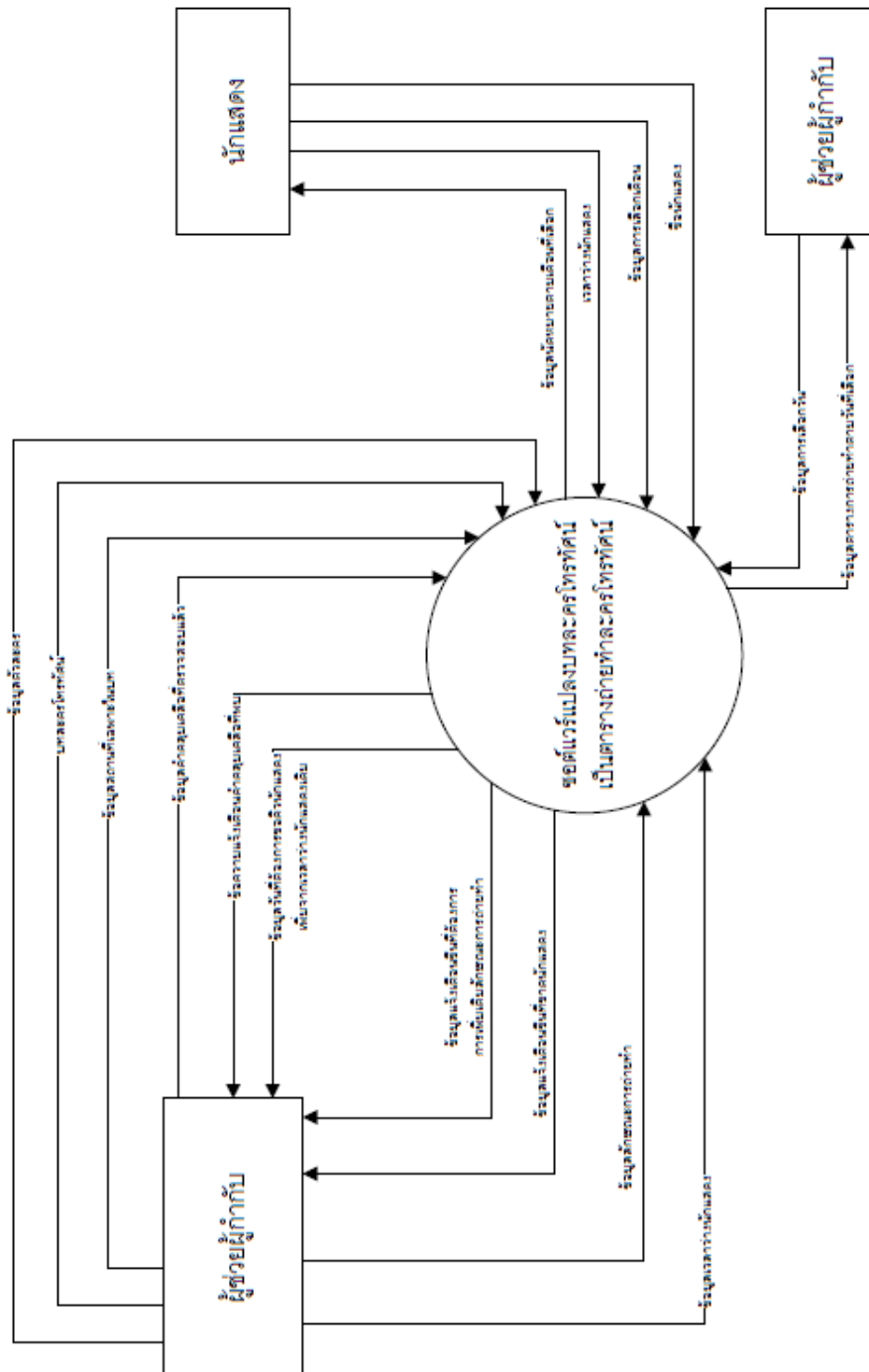


ภาคผนวก

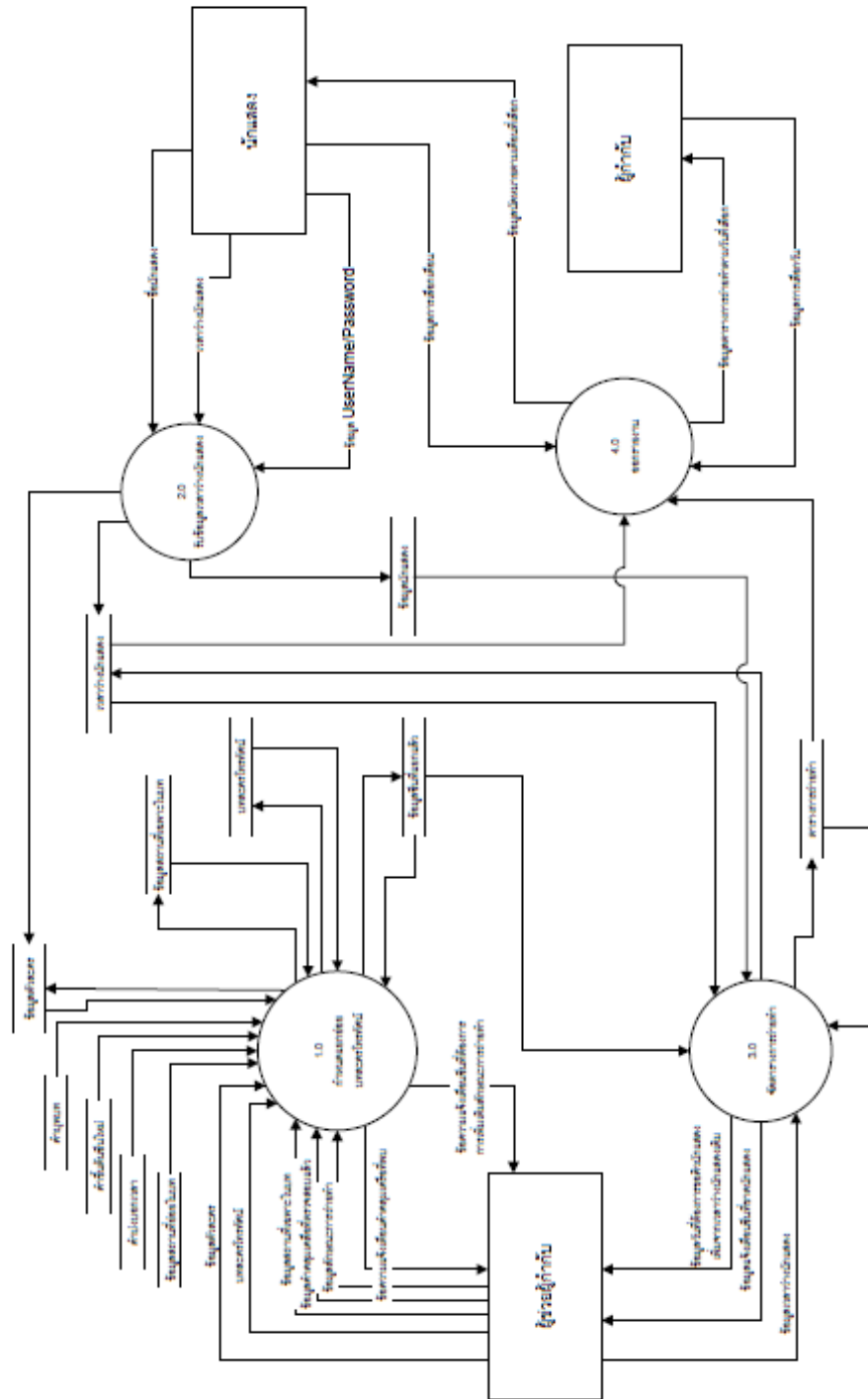
จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

ซอฟต์แวร์เพื่อการแปลงบทละครโทรทัศน์เป็นตารางการถ่ายทำละคร  
 รายละเอียดซอฟต์แวร์ที่ออกแบบไว้

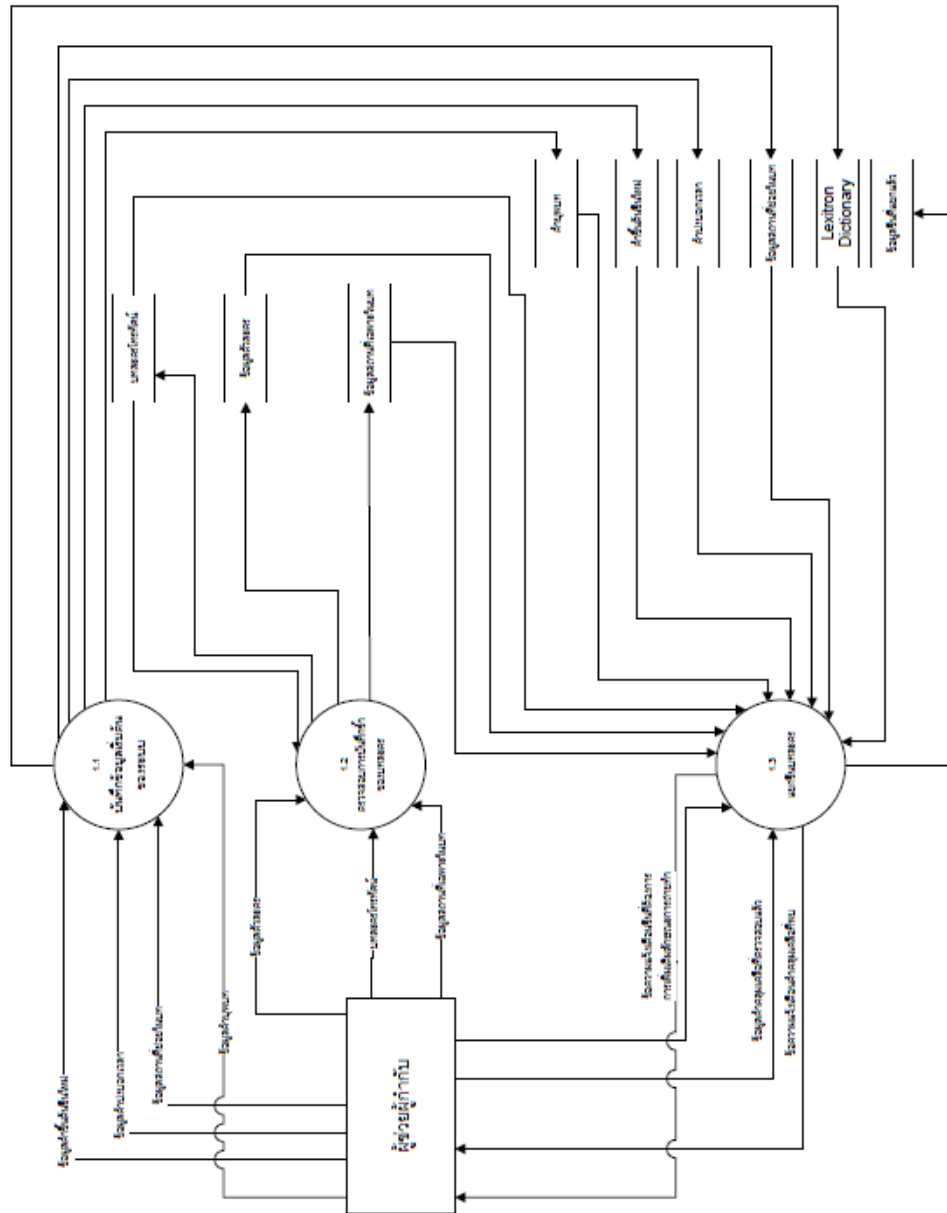
แผนภาพบริบท : ซอฟต์แวร์เพื่อการแปลงบทละครโทรทัศน์เป็นตารางการถ่ายทำละคร



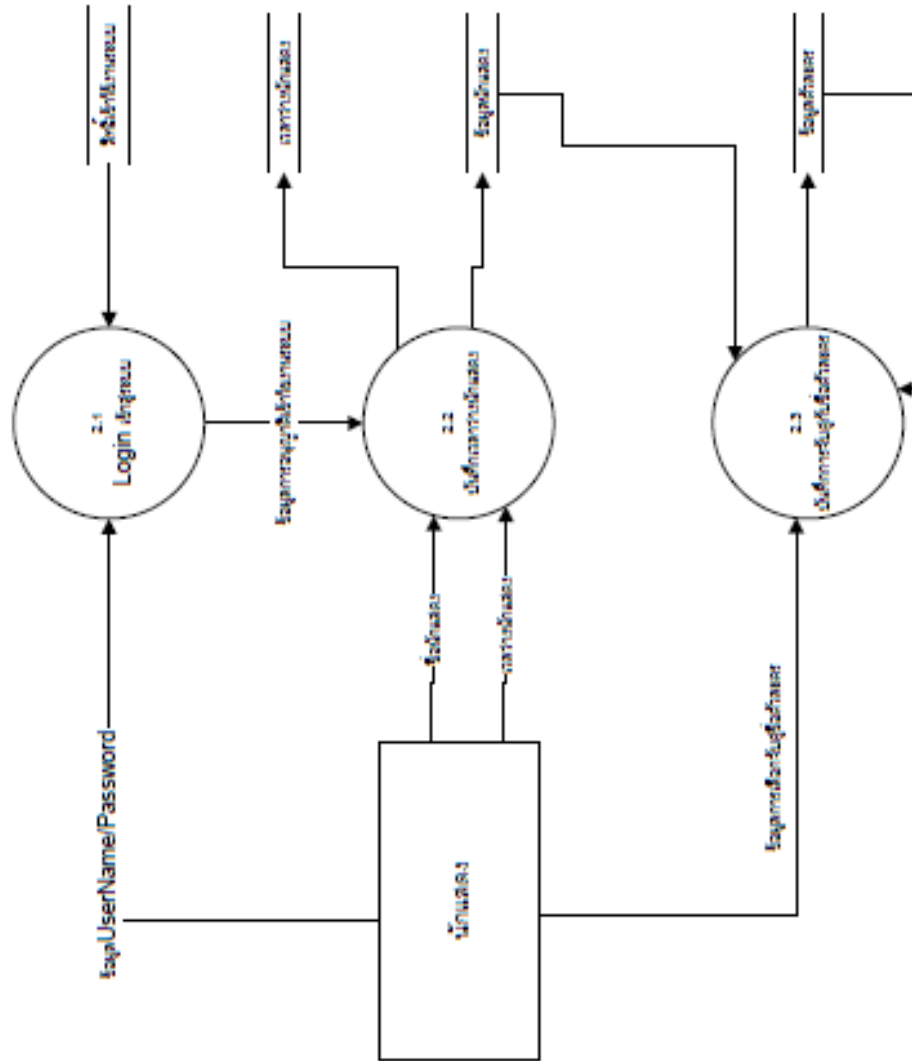
แผนภาพระดับที่ 1 : ซอฟต์แวร์เพื่อการแปลงทละครโทรทัศน์เป็นตารางการถ่ายทำละคร



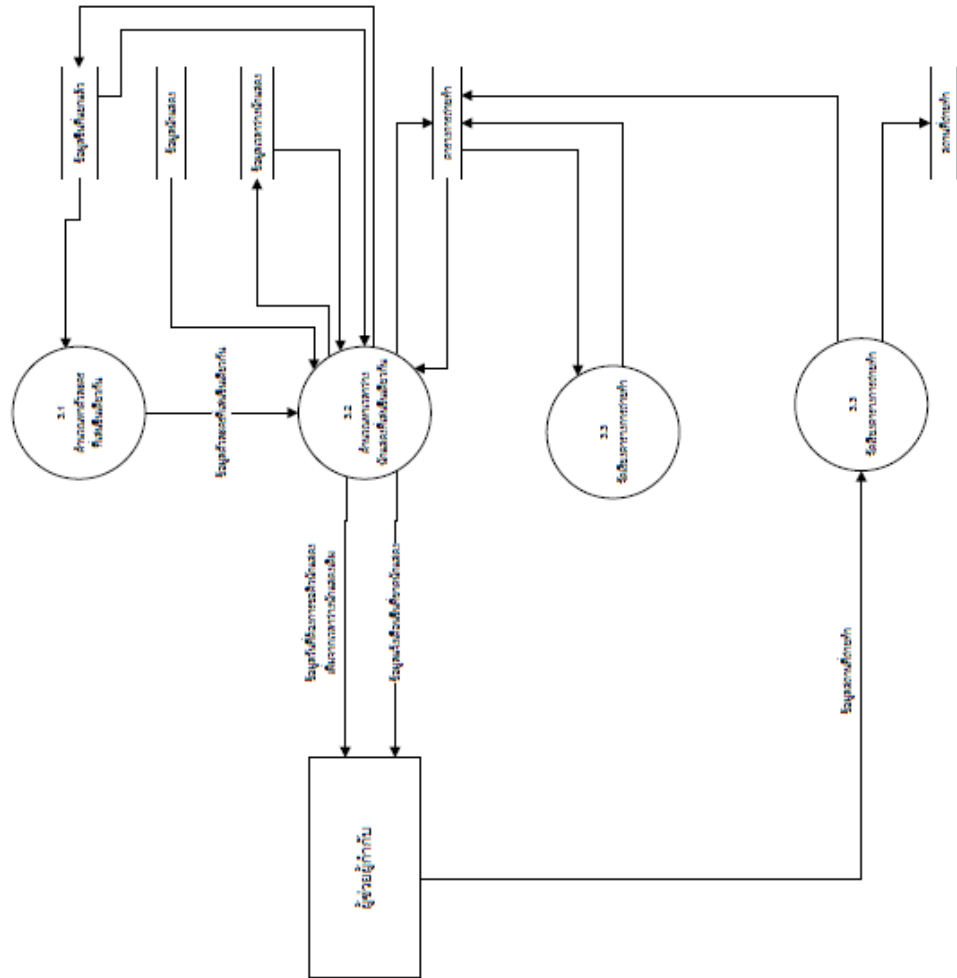
แผนภาพระดับที่ 2 : ซอฟต์แวร์เพื่อการแปลงทละครโทรทัศน์เป็นตารางถ่ายทำละคร  
แผนภาพระบบย่อยของกระบวนการที่ 1 กำหนดแยกย่อยบทละครโทรทัศน์



แผนภาพระบบย่อยของกระบวนการที่ 2 รับข้อมูลเวลาว่างนักแสดง



แผนภาพระบบย่อยของกระบวนการที่ 3 จัดตารางการถ่ายทำ



พจนานุกรมข้อมูล : ซอฟต์แวร์เพื่อการแปลงบทละครโทรทัศน์เป็นตารางการถ่ายทำละคร

43

## 4.5.2. Data Dictionary

ตารางที่ 4.1 แสดงรายละเอียดตาราง Actor

No.	Name	Data Type	Description	Key
1.	ActorID	Integer	รหัสนักแสดง	PK
2.	FirstName	String	ชื่อนักแสดง	
3.	LastName	String	นามสกุลนักแสดง	
4.	UserName	String	รหัสผู้เข้าใช้ระบบ	FK

ตารางที่ 4.2 แสดงรายละเอียดตาราง ActorAvailableSchedule

No.	Name	Data Type	Description	Key
1.	ActorID	Integer	รหัสนักแสดง	PK, FK
2.	AvailableDate	Date	วันที่ว่าง	
3.	DayOrNight	String	ว่างเวลากลางวันหรือกลางคืน	

ตารางที่ 4.3 แสดงรายละเอียดตาราง Character

No.	Name	Data Type	Description	Key
1.	CharacterID	Integer	รหัสตัวละคร	PK
2.	CharacterName	Integer	ชื่อตัวละคร	
3.	Type	Integer	เป็นตัวละครนำหรือตัวละครประกอบ	
4.	ActorID	Integer	รหัสนักแสดง	FK

ตารางที่ 4.4 แสดงรายละเอียดตาราง CharacterAppear

No.	Name	Data Type	Description	Key
1.	CharacterID	Integer	รหัสตัวละคร	PK, FK
2.	SceneNum	Integer	ลำดับที่ของซีน	PK, FK
3.	ScreenPlayID	Integer	รหัสบทละครโทรทัศน์	PK, FK

FK (3)



4.	Frequency	Integer	ความถี่ที่พบชื่อตัวละครในซีน	
----	-----------	---------	------------------------------	--

ตารางที่ 4.5 แสดงรายละเอียดตาราง GeneralPlace

No.	Name	Data Type	Description	Key
1.	GeneralPlaceName	String	ชื่อสถานที่ทั่วไป	PK
2.	Type	String	เป็นสถานที่ Indoor หรือ Outdoor	

ตารางที่ 4.6 แสดงรายละเอียดตาราง Location

No.	Name	Data Type	Description	Key
1.	LocationID	Integer	รหัสสถานที่ถ่ายทำ	PK
2.	LocationName	String	ชื่อสถานที่ถ่ายทำ	
3.	Address	String	ที่อยู่สถานที่ถ่ายทำ	
4.	SpecificPlaceName	String	ชื่อสถานที่เฉพาะในบทละคร	FK

ตารางที่ 4.7 แสดงรายละเอียดตาราง PrepositionAppearInLine

No.	Name	Data Type	Description	Key
1.	PrepositionWord	String	คำบุพบท	PK, FK
2.	SceneNum	Integer	ลำดับที่ของซีน	PK, FK
3.	ScreenPlayID	Integer	รหัสบทละครโทรทัศน์	PK, FK
4.	AppearInLine	Integer	บรรทัดของบทละครที่พบคำบุพบท	
5.	IndexInLine	Integer	ตำแหน่งของคำบุพบทที่พบในบรรทัด	
6.	GeneralPlaceName	String	ชื่อสถานที่ทั่วไป	FK
7.	SpecificPlaceName	String	ชื่อสถานที่เฉพาะ	FK

ตารางที่ 4.8 แสดงรายละเอียดตาราง PrepositionWord

No.	Name	Data Type	Description	Key
1.	PrepositionWord	String	คำบุพบท	PK

ตารางที่ 4.9 แสดงรายละเอียดตาราง Scene

No.	Name	Data Type	Description	Key
1.	ScreenplayID	Integer	รหัสบทละครโทรทัศน์	PK, FK
2.	SceneNum	Integer	ลำดับที่ของซีน	PK
3.	BeginningLine	Integer	บรรทัดเริ่มต้นซีน	
4.	EndingLine	Integer	บรรทัดจบซีน	
5.	DayOrNight	String	เป็นซีนกลางวันหรือกลางคืน	
6.	IndoorOrOutdoor	String	เป็นซีนสถานที่ภายในหรือภายนอก	
7.	ShootingScheduleID	Integer	รหัสตารางการถ่ายทำ	FK

ตารางที่ 4.10 แสดงรายละเอียดตาราง ScreenPlay

No.	Name	Data Type	Description	Key
1.	ScreenplayID	Integer	รหัสบทละครโทรทัศน์	PK
2.	Name	String	ชื่อเรื่อง	
3.	Article	String	เนื้อหาบทละคร	
4.	TotalLine	Integer	จำนวนบรรทัดทั้งหมด	
5.	CreatedDate	String	วันที่บันทึกเข้าระบบ	

ตารางที่ 4.11 แสดงรายละเอียดตาราง ShootingSchedule

No.	Name	Data Type	Description	Key
1.	ShootingScheduleID	Integer	รหัสตารางการถ่ายทำ	PK
2.	Date	Date	วันที่ถ่ายทำ	
3.	LocationAddress	String	ที่อยู่สถานที่ถ่ายทำ	

ตารางที่ 4.12 แสดงรายละเอียดตาราง SpecificPlace

No.	Name	Data Type	Description	Key
1.	SpecificPlaceName	String	ชื่อสถานที่เฉพาะ	PK
2.	Type	String	ประเภทสถานที่	

ตารางที่ 4.13 แสดงรายละเอียดตาราง TimeWord

No.	Name	Data Type	Description	Key
1.	TimeWord	String	คำบ่งบอกเวลา	PK
2.	Type	String	ประเภทคำบอกเวลากลางวันหรือกลางคืน	

ตารางที่ 4.14 แสดงรายละเอียดตาราง TimeWordAppearInLine

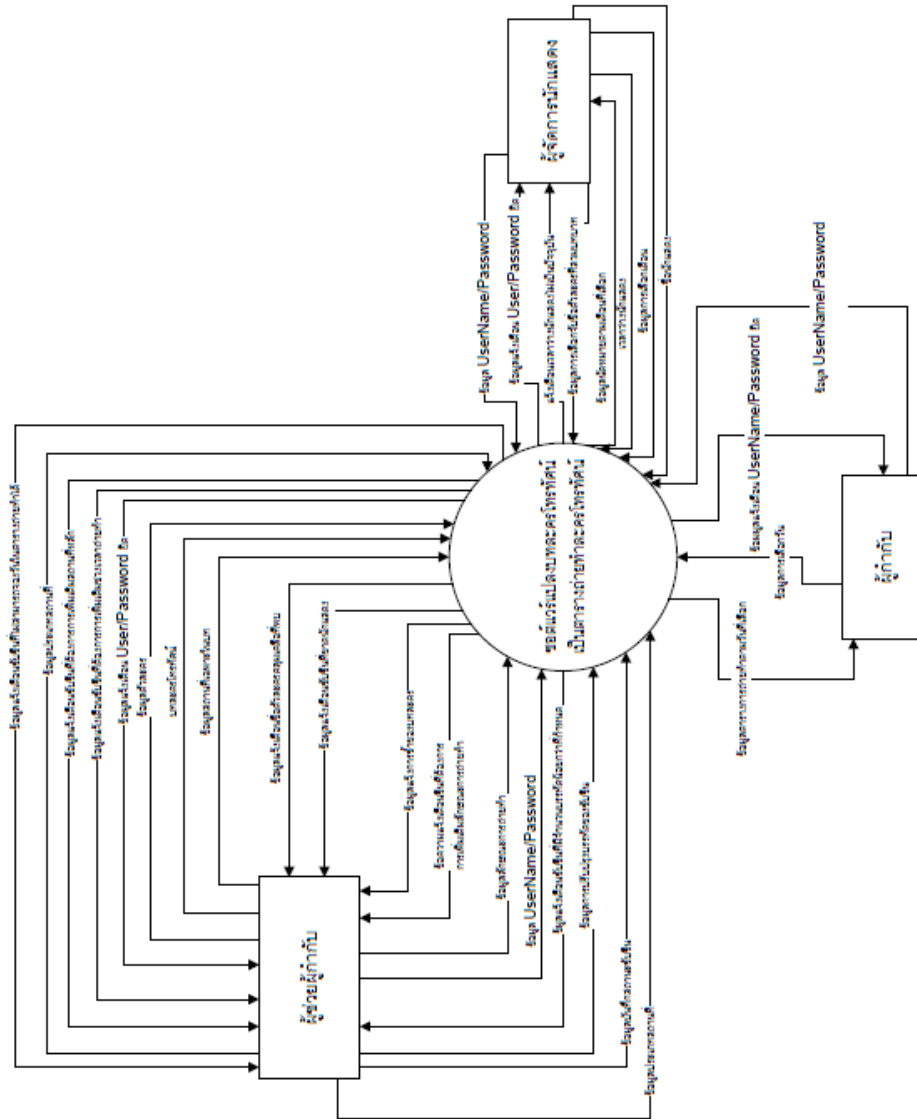
No.	Name	Data Type	Description	Key
1.	SceneNum	Integer	ลำดับที่ของซีน	PK, FK
2.	ScreenplayID	Integer	รหัสบทละครโทรทัศน์	PK, FK
3.	TimeWord	String	คำบ่งบอกเวลา	PK, FK
4.	AppearInLine	Integer	บรรทัดของบทละครที่พบคำบ่งบอกเวลา	

ตารางที่ 4.15 แสดงรายละเอียดตาราง UserLogin

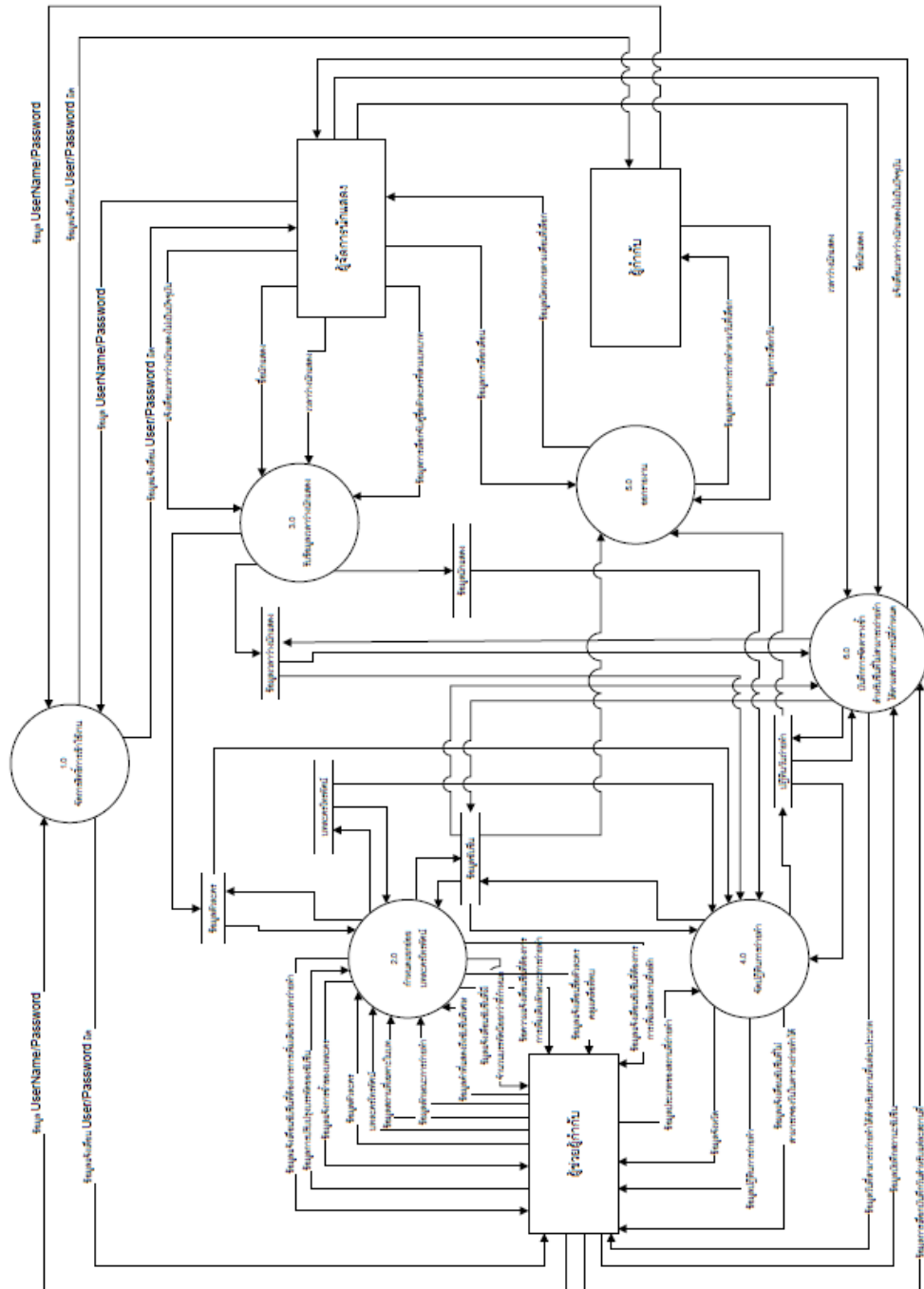
No.	Name	Data Type	Description	Key
1.	Username	String	รหัสผู้เข้าใช้ระบบ	PK
2.	Password	String	รหัสผ่าน	

ซอฟต์แวร์เพื่อการแปลงบทละครโทรทัศน์เป็นตารางการถ่ายทำละคร  
รายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

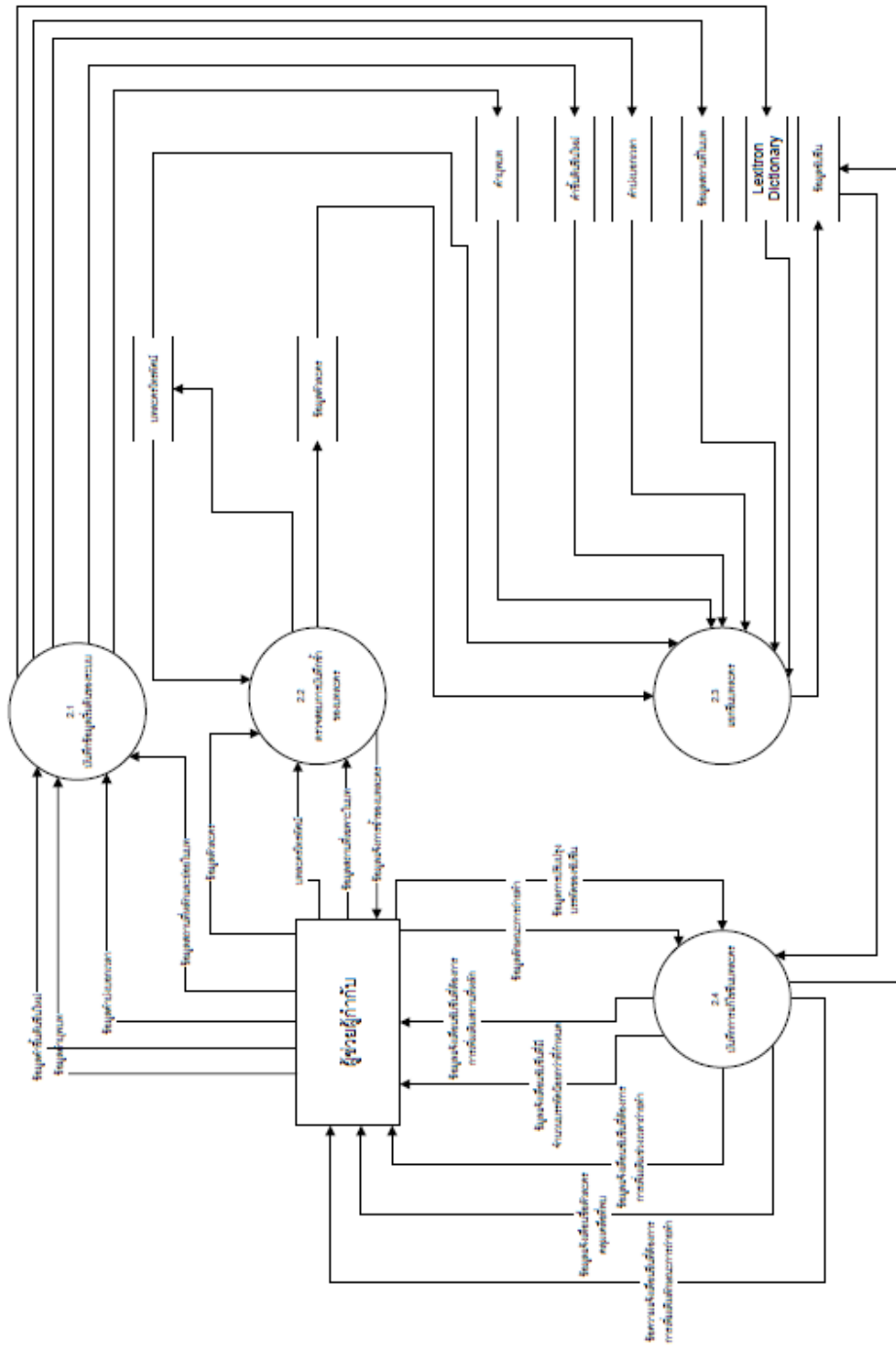
แผนภาพบริบท : ซอฟต์แวร์เพื่อการแปลงบทละครโทรทัศน์เป็นตารางการถ่ายทำละคร



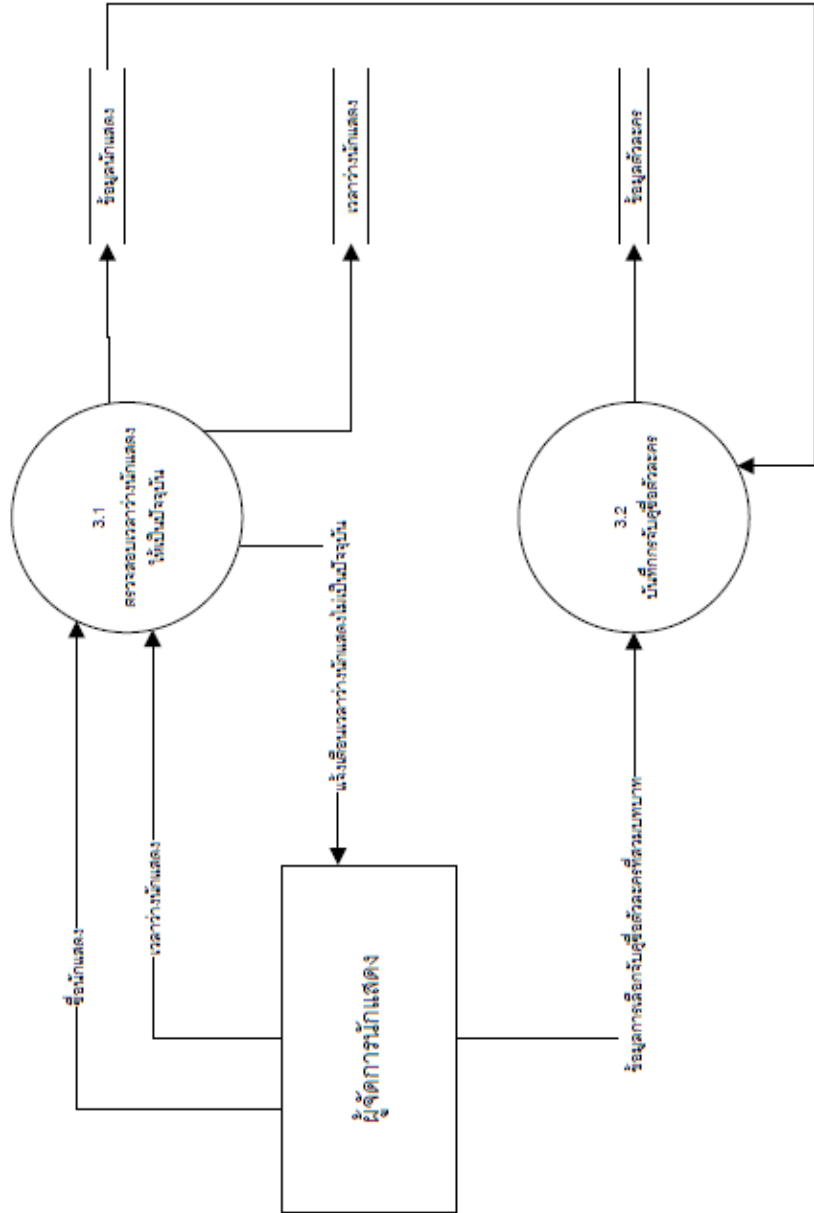
แผนภาพระดับที่ 1 : ซอฟต์แวร์เพื่อการแปลงละครโทรทัศน์เป็นตารางการถ่ายทำละคร



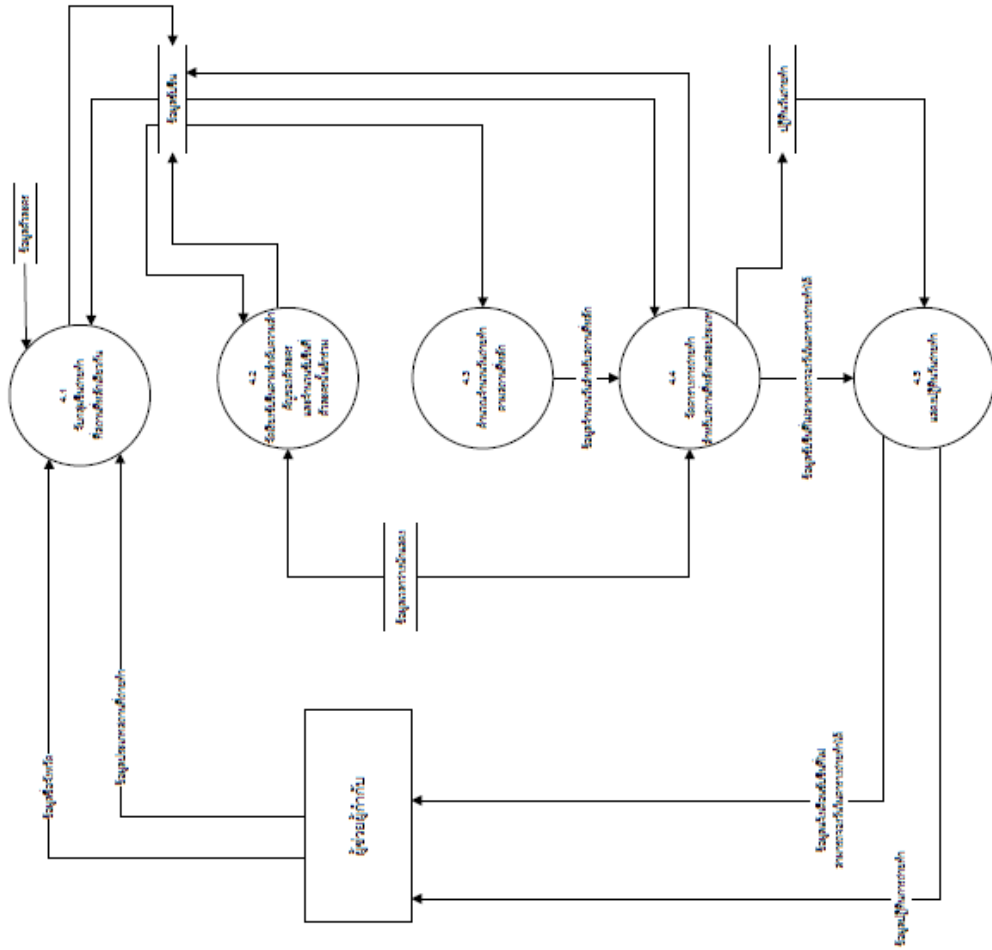
แผนภาพระดับที่ 2 : ซอฟต์แวร์เพื่อการแปลงทละครโทรทัศน์เป็นตารางการถ่ายทำละคร  
 แผนภาพระบบย่อยของกระบวนการที่ 2 กำหนดแยกย่อยทละครโทรทัศน์



แผนภาพระบบย่อยของกระบวนการที่ 3 รับข้อมูลเวลาว่างนักแสดง

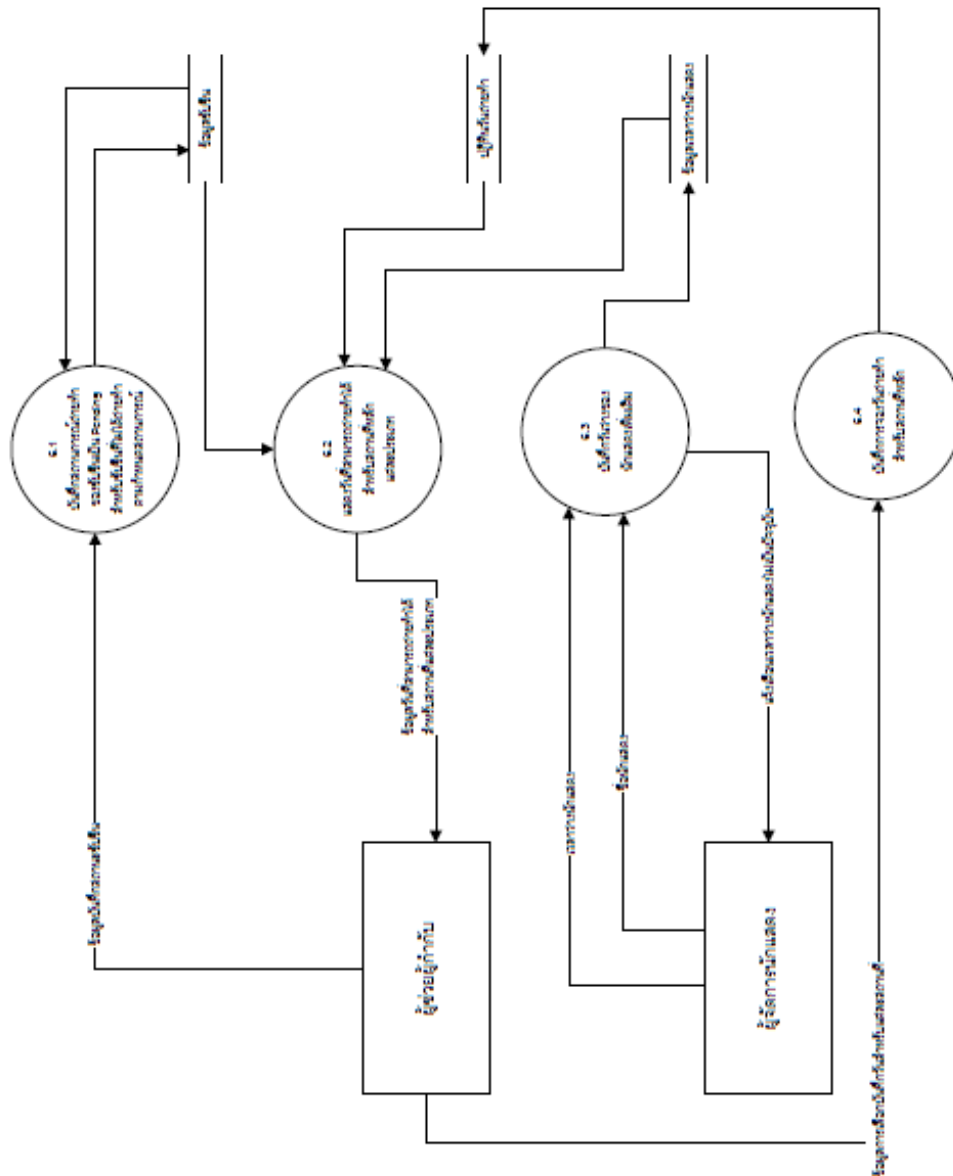


แผนภาพระบบย่อยของกระบวนการที่ 4 จัดปฏิทินการถ่ายทำ





แผนภาพระบวย่อยของกระบวนการที่ 6 บันทึกการจัดการตารางเข้าสำหรับสินค้าที่ไม่สามารถถ่ายทำได้ตามสถานการณ์ที่กำหนด



พจนานุกรมข้อมูล : ซอฟต์แวร์เพื่อการแปลงบทละครโทรทัศน์เป็นตารางการถ่ายทำละคร

123

## 4.6.2. Data Dictionary

ตารางที่ 4.24 แสดงรายละเอียดตาราง Character

No.	Name	Data Type	Description	Key
1.	CharacterId	Integer	รหัสตัวละคร	PK
2.	StoryId	Integer	รหัสบทละครโทรทัศน์	FK
3.	CharacterName	String	ชื่อตัวละคร	
4.	PriorityPoint	Integer	ความสำคัญของตัวละครในบท	
5.	IsHomograph	Boolean	สถานะความเป็นคำพ้องรูปกับคำในพจนานุกรม	
6.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
7.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
8.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
9.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
10.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.25 แสดงรายละเอียดตาราง CharacterInContent

No.	Name	Data Type	Description	Key
1.	CharacterId	Integer	รหัสตัวละคร	PK, FK
2.	ContentId	Integer	รหัสของบรรทัดเนื้อหาบทละคร	PK, FK
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.26 แสดงรายละเอียดตาราง Content

No.	Name	Data Type	Description	Key
1.	ContentId	Integer	รหัสของบรรทัดเนื้อหาบทละคร	PK
2.	StoryId	Integer	รหัสบทละครโทรทัศน์	FK
3.	LineNumber	Integer	เลขบรรทัด	
4.	ContentData	String	เนื้อหาภายในบรรทัด	
5.	ContentWthOutSph	String	เนื้อหาภายในบรรทัดที่ลบบทสนทนาออกแล้ว	
6.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
7.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
8.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
9.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
10.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.27 แสดงรายละเอียดตาราง ContentInScene

No.	Name	Data Type	Description	Key
1.	SceneId	Integer	รหัสฉากขึ้น	PK, FK
2.	ContentId	Integer	รหัสของบรรทัดเนื้อหาบทละครโทรทัศน์	PK, FK
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.28 แสดงรายละเอียดตาราง Lexitron

No.	Name	Data Type	Description	Key
1.	LexitronId	Integer	รหัสคำศัพท์	PK
2.	Tentry	String	คำศัพท์ภาษาไทย	

3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.29 แสดงรายละเอียดตาราง Location

No.	Name	Data Type	Description	Key
1.	LocationId	Integer	รหัสสถานที่ในบทละครโทรทัศน์	PK
2.	LocationTypeId	Integer	รหัสประเภทสถานที่	FK
3.	ShootingConditionTypeId	Integer	รหัสลักษณะการถ่ายทำ	FK
4.	LocationAddressId	Integer	รหัสจังหวัดที่ถ่ายทำ	FK
5.	LocationName	String	ชื่อสถานที่ในบทละคร	
6.	IsSpecificPlace	Boolean	สถานะการเป็นสถานที่เฉพาะ	
7.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
8.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
9.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
10.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
11.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.30 แสดงรายละเอียดตาราง LocationAddress

No.	Name	Data Type	Description	Key
1.	LocationAddressId	Integer	รหัสจังหวัดที่ถ่ายทำ	PK
2.	LocationDirectionId	Integer	รหัสประเภทสถานที่ที่ถ่ายทำ	FK
3.	LocationAddressName	String	ชื่อจังหวัด	
4.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
5.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
6.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
7.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
8.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.31 แสดงรายละเอียดตาราง LocationDirection

No.	Name	Data Type	Description	Key
1.	LocationDirectionId	Integer	รหัสประเภทสถานที่ถ่ายทำ	PK
2.	LocationDirectionName	String	ประเภทการถ่ายทำ	
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.32 แสดงรายละเอียดตาราง LocationInStory

No.	Name	Data Type	Description	Key
1.	LocationInStoryId	Integer	รหัสสถานที่ที่พบในบทละครโทรทัศน์	PK
2.	StoryId	Integer	รหัสบทละครโทรทัศน์	FK
3.	LocationId	Integer	รหัสสถานที่ในบทละครโทรทัศน์	FK
4.	LocationAddressId	Integer	รหัสจังหวัดที่ถ่ายทำ	FK
5.	LocationDirectionId	Integer	รหัสประเภทสถานที่ถ่ายทำ	FK
6.	Days	Integer	จำนวนวันที่ใช้ถ่ายทำของสถานที่นี้	
7.	Idx	Integer	ลำดับของการเรียงสถานที่ในกลุ่มถ่ายทำต่างจังหวัด	
8.	Group	Integer	หมายเลขกลุ่มการจัดตารางการถ่ายทำต่างจังหวัด	
9.	TotalSubScene	Integer	จำนวนซับซีนทั้งหมดที่อยู่ในสถานที่นี้	
10.	TotalCharacter	Integer	จำนวนตัวละครทั้งหมดที่อยู่ในสถานที่นี้	
11.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
12.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
13.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
14.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
15.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.33 แสดงรายละเอียดตาราง LocationInStoryReschedule

No.	Name	Data Type	Description	Key
1.	LocationInStoryRescheduleId	Integer	รหัสการจัดตารางใหม่ของสถานที่ที่พบในบทละครโทรทัศน์	PK
2.	LocationInStoryId	Integer	รหัสสถานที่ที่พบในบทละครโทรทัศน์	FK
3.	Days	Integer	จำนวนวันที่ใช้ถ่ายทำของสถานที่นี้	
4.	Idx	Integer	ลำดับของการเรียงชั้นขึ้นในกลุ่มถ่ายทำของตัวเอง	
5.	Group	Integer	หมายเลขกลุ่มการจัดตารางการถ่ายทำต่างจังหวัด	
6.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
7.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
8.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
9.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
10.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.34 แสดงรายละเอียดตาราง LocationType

No.	Name	Data Type	Description	Key
1.	LocationTypeId	Integer	รหัสประเภทสถานที่	PK
2.	Type	String	ชื่อประเภทสถานที่	
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.35 แสดงรายละเอียดตาราง Preposition

No.	Name	Data Type	Description	Key
1.	PrepositionId	Integer	รหัสคำบุพบท	PK
2.	ShootingConditionTypeId	Integer	รหัสลักษณะการถ่ายทำ	FK

3.	Word	String	คำบุพบท	
4.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
5.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
6.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
7.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
8.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.36 แสดงรายละเอียดตาราง PrepositionInContent

No.	Name	Data Type	Description	Key
1.	PrepositionId	Integer	รหัสคำบุพบท	PK, FK
2.	ContentId	Integer	รหัสของบรรทัดเนื้อหาบทละครโทรทัศน์	PK, FK
3.	Idx	Integer	ตำแหน่งที่พบคำบุพบทในบรรทัดของบทละครโทรทัศน์	PK
4.	LocationId	Integer	รหัสสถานที่ในบทละครโทรทัศน์	FK
5.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
6.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
7.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
8.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
9.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.37 แสดงรายละเอียดตาราง Scene *sub sen*

No.	Name	Data Type	Description	Key
1.	SceneId	Integer	รหัสฉาก	PK
2.	SceneControllId	Integer	รหัสตัวควบคุมฉาก	FK
3.	TimeTypeId	Integer	รหัสช่วงเวลาถ่ายทำ	FK
4.	ShootingConditionType Id	Integer	รหัสลักษณะการถ่ายทำ	FK
5.	LocationInStoryId	Integer	รหัสสถานที่ที่พบในบทละครโทรทัศน์	FK
6.	StartLine	Integer	บรรทัดเริ่มต้นของฉาก	

7.	MainLocation	String	สถานที่หลัก	
8.	SubLocation	String	สถานที่ย่อย	
9.	TimeForScene	Float	เวลาที่ใช้ถ่ายทำฉาก	
10.	LowerFlag	Boolean	สถานะฉากที่มีบรรทัดต่ำกว่าที่กำหนด	
11.	CharacterFlag	Boolean	สถานะว่ามีตัวละครอยู่ในฉากหรือไม่	
12.	MainLocationFlag	Boolean	สถานะว่ามีสถานที่หลักอยู่ในฉากหรือไม่	
13.	TimeWarningFlag	Boolean	สถานะว่ามีเวลาอยู่ในฉากหรือไม่	
14.	Idx	Integer	ลำดับของการเรียงฉากในกลุ่มถ่ายทำของตัวเอง	
15.	CharacterInScene	Integer	จำนวนตัวละครในฉาก	
16.	SceneIdx	Integer	เลขฉาก	
17.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
18.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
19.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
20.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
21.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.38 แสดงรายละเอียดตาราง SceneControl

No.	Name	Data Type	Description	Key
1.	SceneControlId	Integer	รหัสตัวควบคุมฉาก	PK
2.	StoryId	Integer	รหัสบทละครโทรทัศน์	FK
3.	CheckSubmitFlag	Boolean	สถานะการตรวจสอบการแบ่งกลุ่มตามประเภทสถานที่ถ่ายทำของฉาก	
4.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
5.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
6.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
7.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
8.	DeleteFlag	Boolean	สถานะการถูกลบออก	





ตารางที่ 4.39 แสดงรายละเอียดตาราง SceneReschedule

No.	Name	Data Type	Description	Key
1.	SceneRescheduleId	Integer	รหัสการจัดตารางการถ่ายทำซ่อม	PK
2.	ScenelId	Integer	รหัสซัปซึน	FK
3.	LocationInStoryRescheduleId	Integer	รหัสการจัดตารางใหม่ของสถานที่ที่พบในบทละครโทรทัศน์	FK
4.	SceneRescheduleStatus Id	Integer	รหัสสถานะการจัดตารางการถ่ายทำซ่อม	FK
5.	Idx	Integer	ลำดับของการเรียงซัปซึนในกลุ่มถ่ายทำของตัวเอง	
6.	RescheduleFlag	Boolean	สถานะการจัดตารางซ่อม	
7.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
8.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
9.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
10.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
11.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.40 แสดงรายละเอียดตาราง SceneRescheduleStatus

No.	Name	Data Type	Description	Key
1.	SceneRescheduleStatusId	Integer	รหัสสถานะการจัดตารางการถ่ายทำซ่อม	PK
2.	Name	String	ชื่อสถานะการจัดตารางการถ่ายทำซ่อม	
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.41 แสดงรายละเอียดตาราง ShootingConditionType

No.	Name	Data Type	Description	Key
1.	ShootingConditionTypeId	Integer	รหัสลักษณะการถ่ายทำ	PK
2.	Type	String	ลักษณะการถ่ายทำ	
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.42 แสดงรายละเอียดตาราง Story (ปฏิทินมหาชน)

No.	Name	Data Type	Description	Key
1.	StoryId	Integer	รหัสบทละครโทรทัศน์	PK
2.	StoryName	String	ชื่อบทละครโทรทัศน์	
3.	FileName	String	ชื่อไฟล์	
4.	StartPeriod	Date	วันเริ่มต้นถ่ายทำ	
5.	EndPeriod	Date	วันสิ้นสุดการถ่ายทำ	
6.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
7.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
8.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
9.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
10.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.43 แสดงรายละเอียดตาราง SubLocationInStory

No.	Name	Data Type	Description	Key
1.	SubLocationInStoryId	Integer	รหัสสถานที่ย่อยที่พบในบทละครโทรทัศน์	PK
2.	StoryId	Integer	รหัสบทละครโทรทัศน์	FK
3.	LocationId	Integer	รหัสสถานที่ในบทละครโทรทัศน์	FK
4.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	

5.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
6.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
7.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
8.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.44 แสดงรายละเอียดตาราง SubScenePrefix

No.	Name	Data Type	Description	Key
1.	SubScenePrefixId	Integer	รหัสคำขึ้นต้นซ้ำขึ้น	PK
2.	Word	String	คำขึ้นต้นซ้ำขึ้น	
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.45 แสดงรายละเอียดตาราง SubScenePrefixInContent

No.	Name	Data Type	Description	Key
1.	SubScenePrefixId	Integer	รหัสคำขึ้นต้นซ้ำขึ้นที่พบในบทละครโทรทัศน์	PK, FK
2.	ContentId	Integer	รหัสของบรรทัดเนื้อหาบทละคร	PK, FK
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.46 แสดงรายละเอียดตาราง Time

No.	Name	Data Type	Description	Key
1.	TimeId	Integer	รหัสค่าบ่งบอกเวลา	PK

2.	TimeTypeId	Integer	รหัสช่วงเวลาถ่ายทำ	FK
3.	Word	String	คำบ่งบอกเวลา	
4.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
5.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
6.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
7.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
8.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.47 แสดงรายละเอียดตาราง TimeInContent

No.	Name	Data Type	Description	Key
1.	TimeId	Integer	รหัสคำบ่งบอกเวลา	PK, FK
2.	ContentId	Integer	รหัสของบรรทัดเนื้อหาบทละคร	PK, FK
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.48 แสดงรายละเอียดตาราง TimeLine

No.	Name	Data Type	Description	Key
1.	SceneId	Integer	รหัสฉาก	PK, FK
2.	DateTime	Date-Time	วันที่จองฉากขึ้นลงตารางการถ่ายทำ	
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.49 แสดงรายละเอียดตาราง TimeType

No.	Name	Data Type	Description	Key
1.	TimeTypeId	Integer	รหัสช่วงเวลาถ่ายทำ	PK
2.	Type	String	ช่วงเวลาถ่ายทำ	
3.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
4.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
5.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.50 แสดงรายละเอียดตาราง User

No.	Name	Data Type	Description	Key
1.	UserId	Integer	รหัสผู้ใช้งานระบบ	PK
2.	Username	String	ชื่อ User	
3.	FirstName	String	ชื่อ	
4.	LastName	String	นามสกุล	
5.	Address	String	ที่อยู่	
6.	Telephone	String	หมายเลขโทรศัพท์	
7.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
8.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
9.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.51 แสดงรายละเอียดตาราง UserPlayCharacter

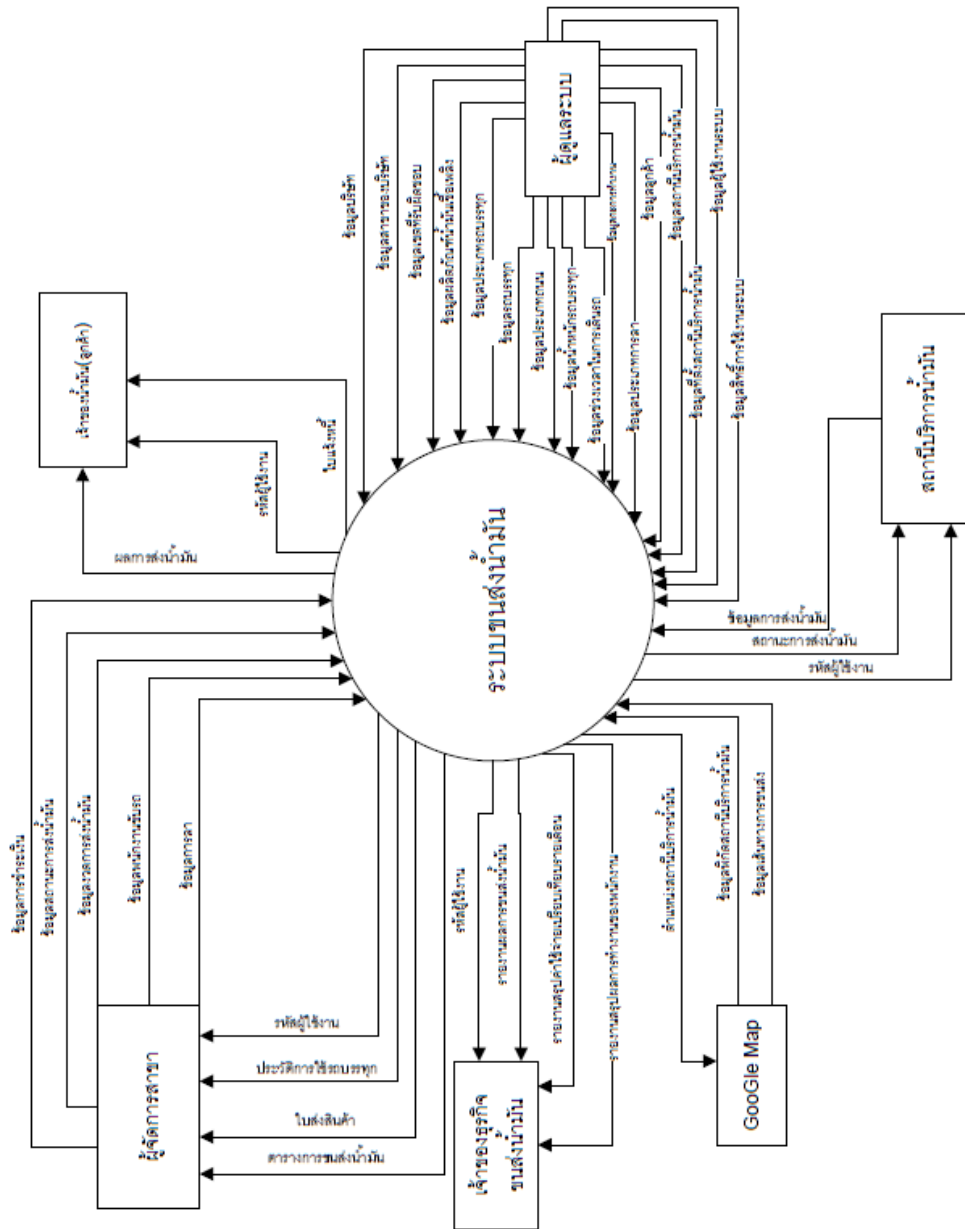
No.	Name	Data Type	Description	Key
1.	UserPlayCharacterId	Integer	รหัสการสวมบทบาทเป็นตัวละครของนักแสดง	PK
2.	UserId	Integer	รหัสผู้ใช้งานระบบ	FK
3.	CharacterId	Integer	รหัสตัวละคร	FK

4.	Priority	Integer	ระดับความสำคัญของนักแสดง	
5.	FreeDayPerWeek	Integer	วันว่างต่ออาทิตย์ของนักแสดง	
7.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
8.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
9.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
6.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
7.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ตารางที่ 4.52 แสดงรายละเอียดตาราง UserPlayCharacterFreeTime

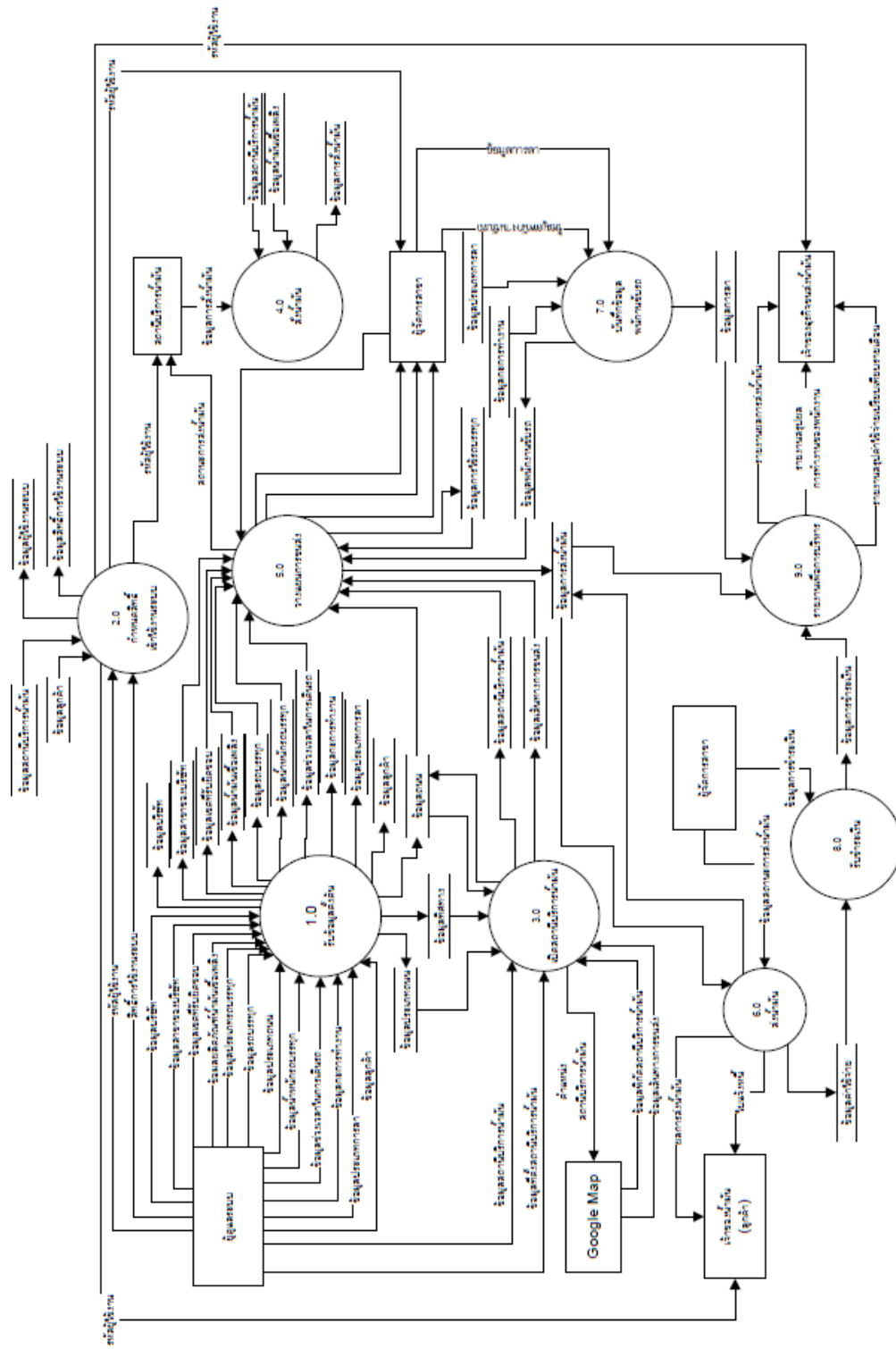
No.	Name	Data Type	Description	Key
1.	UserPlayCharacterIdFreeTime	Integer	รหัสวันว่างของนักแสดง	PK
2.	UserPlayCharacterId	Integer	รหัสผู้เข้าใช้งานระบบ	FK
3.	Day	Integer	วัน	
4.	Month	Integer	เดือน	
5.	Year	Integer	ปี	
6.	CreatedBy	String	ชื่อ User ที่สร้าง Record นี้	
7.	CreatedOn	Date-Time	วันที่สร้าง Record นี้	
8.	UpdatedBy	String	ชื่อ User ที่แก้ไข Record นี้	
9.	UpdatedOn	Date-Time	วันที่แก้ไข Record นี้	
10.	DeleteFlag	Boolean	สถานะการถูกลบออก	

ระบบขนส่งน้ำมัน  
 รายละเอียดซอฟต์แวร์ที่ออกแบบได้  
 แผนภาพบริบท : ระบบขนส่งน้ำมัน



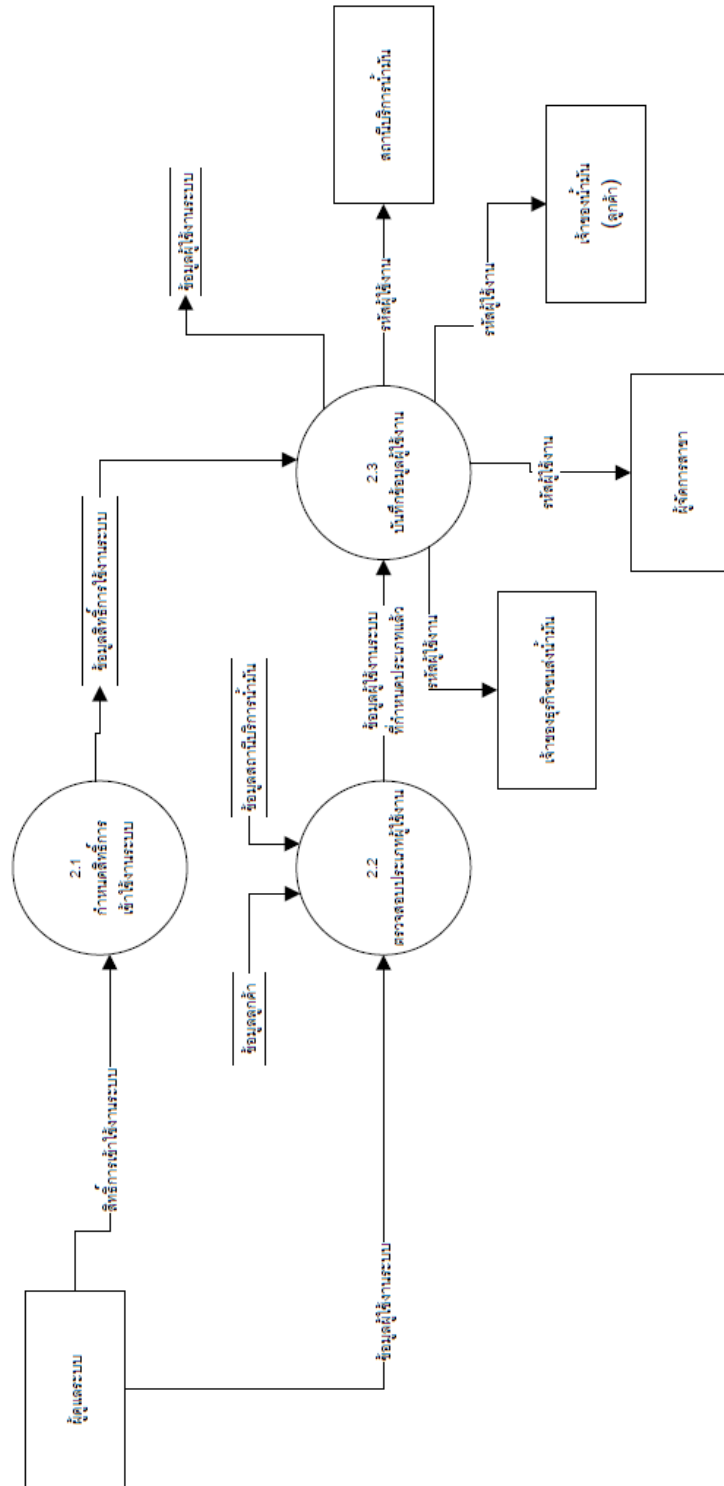


แผนภาพระดับที่ 1 : ระบบขนส่งน้ำมัน

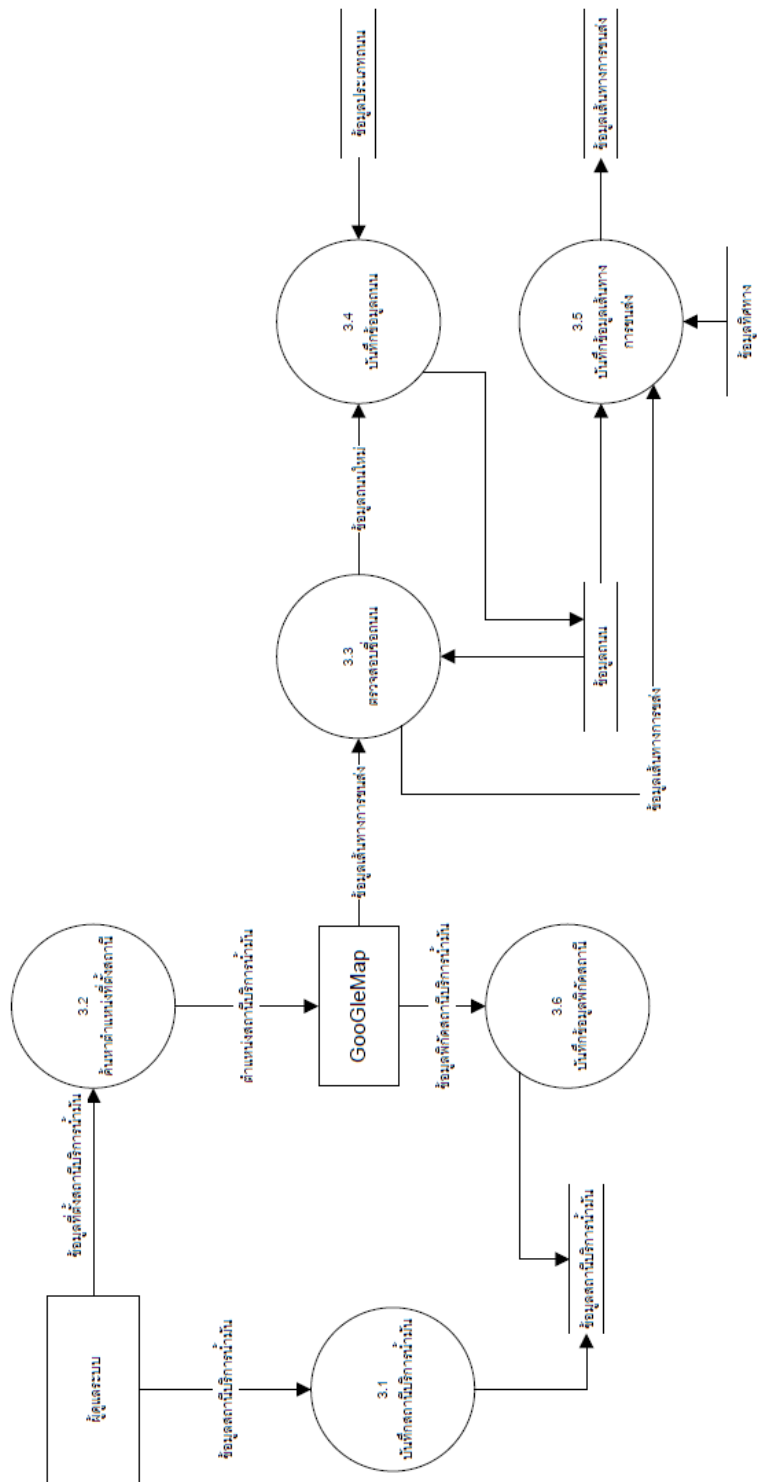


แผนภาพระดับที่ 2 : ระบบขนส่งน้ำมัน

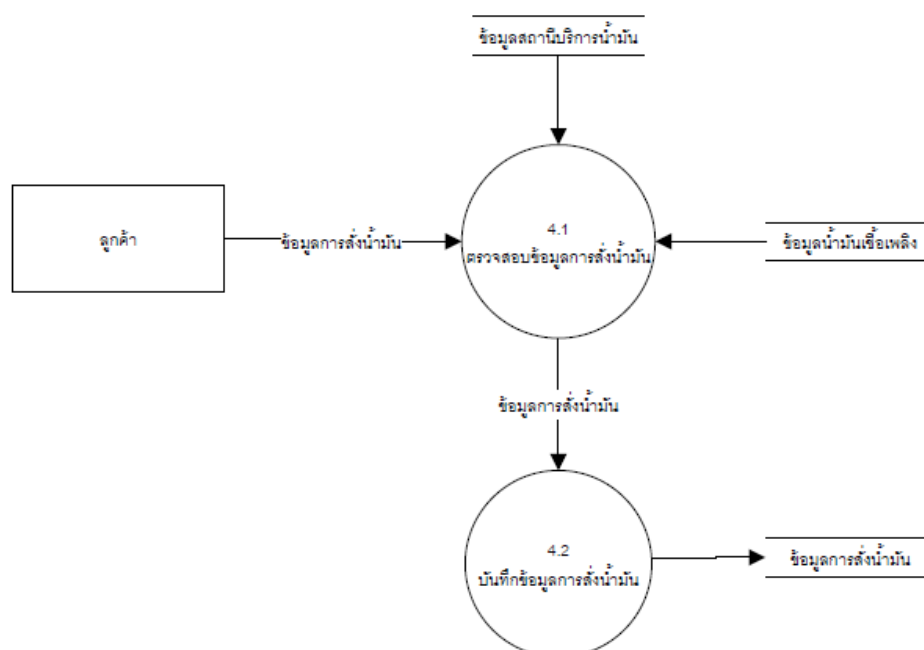
แผนภาพระบย่อยของกระบวนการที่ 2 กำหนดสิทธิการใช้งานระบบ



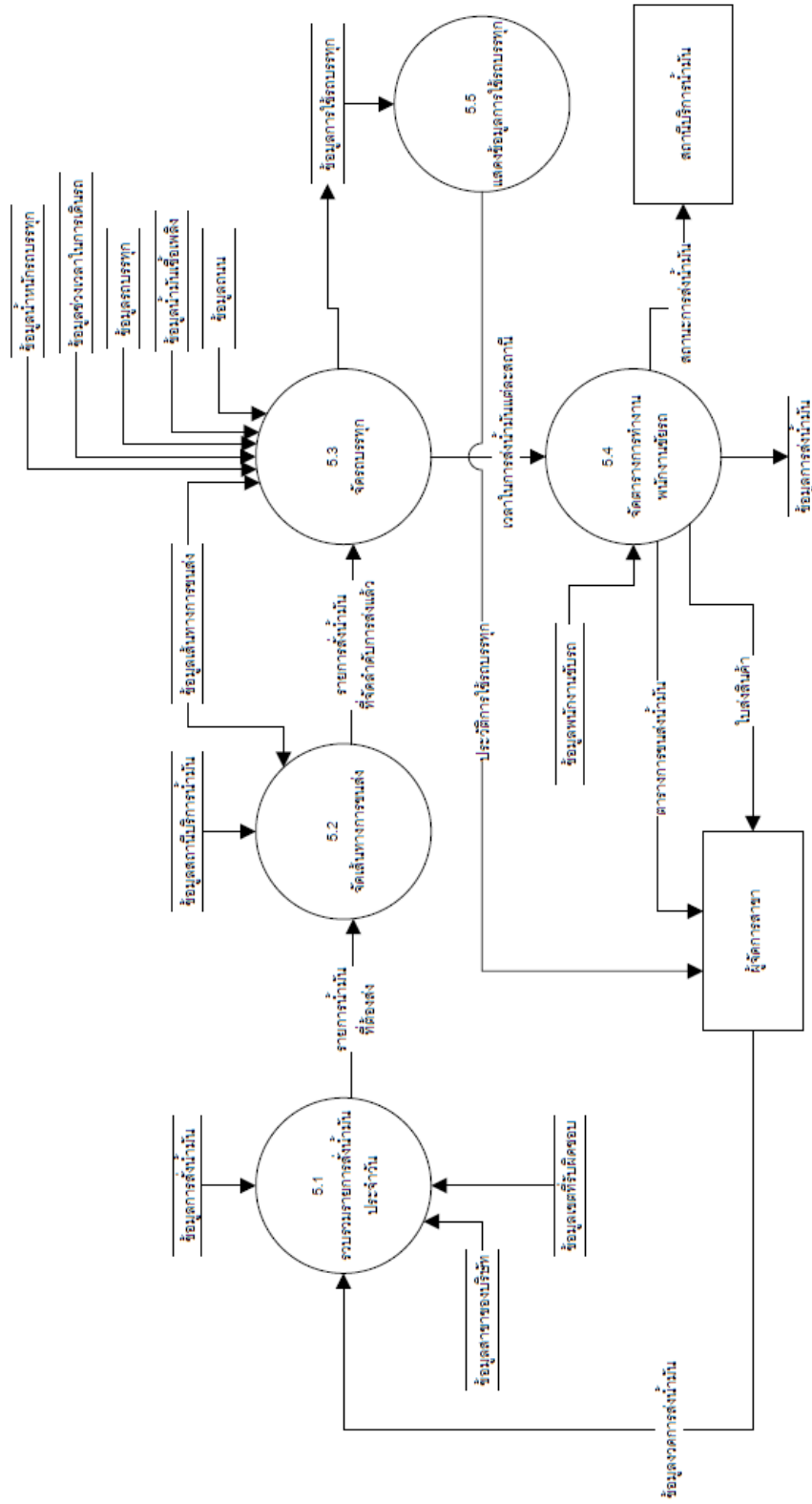
แผนภาพระบบย่อยของกระบวนการที่ 3 เปิดสถานีบริการน้ำมัน



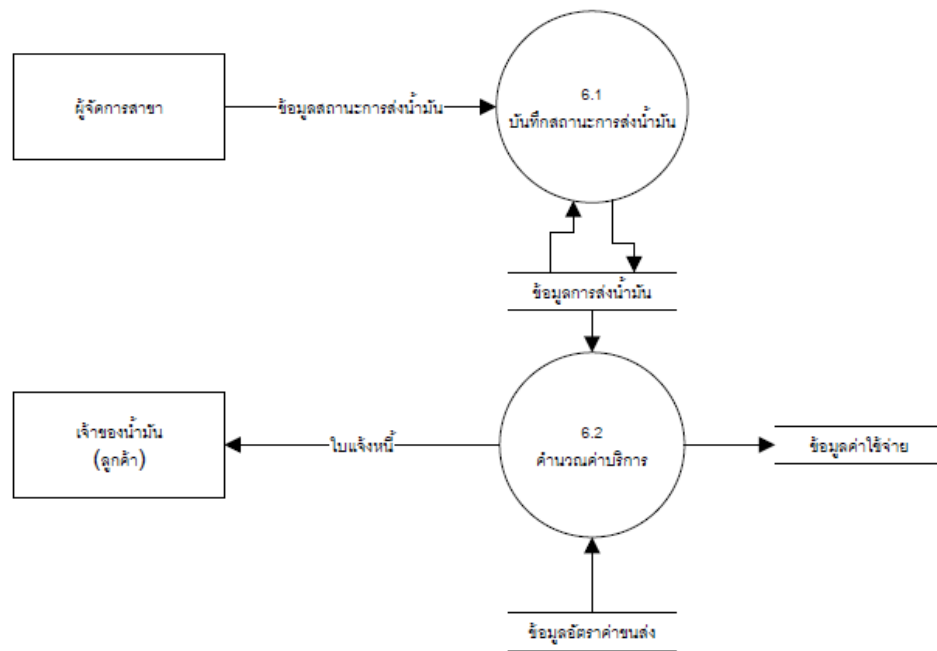
แผนภาพระบบย่อยของกระบวนการที่ 4 สั่งน้ำมัน



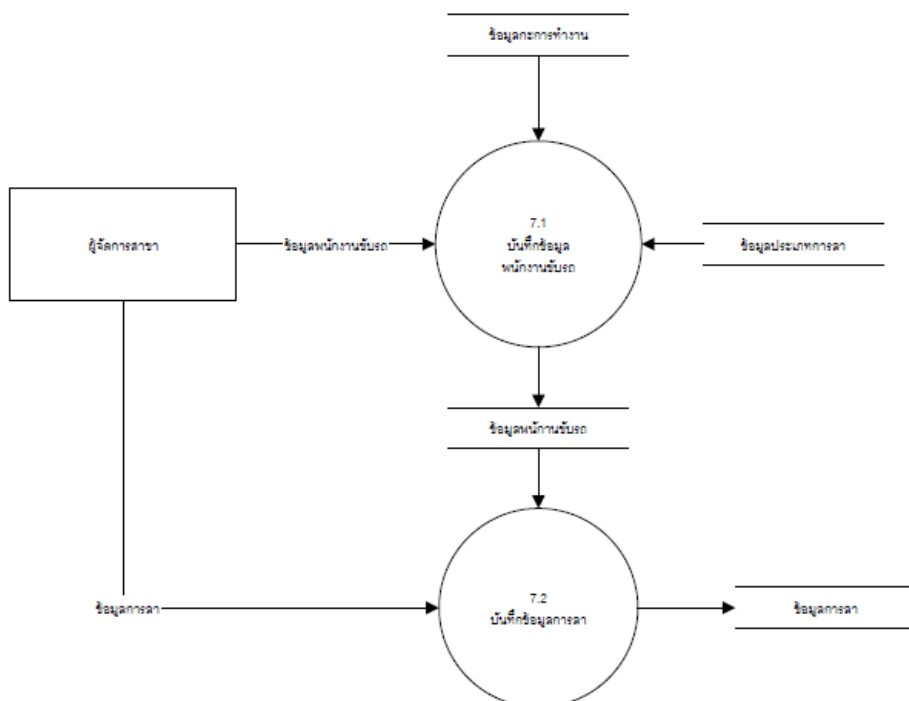
แผนภาพระบบย่อยของกระบวนการที่ 5 วางแผนการขนส่ง



แผนภาพระบบย่อยของกระบวนการที่ 6 ส่งน้ำมัน



แผนภาพระบบย่อยของกระบวนการที่ 7 บันทึกข้อมูลพนักงานขับรถ



พจนานุกรมข้อมูล : ระบบขนส่งน้ำมัน

78

4.9.2 รายละเอียดการออกแบบฐานข้อมูล (Data Dictionary)

• ข้อมูลตาราง MS\_COMPANY

ตารางที่ 4.1 แสดงรายละเอียดตาราง MS\_COMPANY

No	Column	Data Type	PK	FK	Description
1	COMPANY_ID	NUMBER	✓		รหัสบริษัท (ระบบสร้างให้อัตโนมัติ)
2	COMPANY_NAME	VARCHAR(100)			ชื่อบริษัท
3	ADDRESS	VARCHAR(100)			ที่อยู่
4	ROAD	VARCHAR(100)			ถนน
5	KHWANG	VARCHAR(50)			แขวง / ตำบล
6	KHET	VARCHAR(50)			เขต / อำเภอ
7	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
8	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
9	PHONE	VARCHAR(30)			เบอร์โทรศัพท์
10	FAX	VARCHAR(30)			แฟกซ์

• ข้อมูลตาราง MS\_BRANCH

ตารางที่ 4.2 แสดงรายละเอียดตาราง MS\_BRANCH

No	Column	Data Type	PK	FK	Description
1	BRANCH_ID	NUMBER	✓		รหัสสาขาของบริษัท(ระบบสร้างให้อัตโนมัติ)
2	BRANCH_CODE	VARCHAR(5)			ชื่อย่อสาขา
3	BRANCH_NAME	VARCHAR(100)			ชื่อสาขา
4	COMPANY_ID	NUMBER		✓	รหัสบริษัท
5	ADDRESS	VARCHAR(100)			ที่อยู่
6	ROAD	VARCHAR(100)			ถนน
7	KHWANG	VARCHAR(50)			แขวง / ตำบล
8	KHET	VARCHAR(50)			เขต / อำเภอ
9	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด

10	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
11	PHONE	VARCHAR(30)			เบอร์โทรศัพท์
12	FAX	VARCHAR(30)			แฟกซ์
13	ZONE_ID	NUMBER		✓	รหัสเขตที่ส่ง

- ข้อมูลตาราง MS\_PROVINCE

ตารางที่ 4.3 แสดงรายละเอียดตาราง MS\_PROVINCE

No	Column	Data Type	PK	FK	Description
1	PROVINCE_ID	NUMBER	✓		รหัสจังหวัด(ระบบสร้างให้อัตโนมัติ)
2	PROVINCE_NAME	VARCHAR(100)			ชื่อจังหวัด

- ข้อมูลตาราง MS\_ZONE

ตารางที่ 4.4 แสดงรายละเอียดตาราง MS\_ZONE

No	Column	Data Type	PK	FK	Description
1	ZONE_ID	NUMBER	✓		รหัสเขตที่รับผิดชอบ (ระบบสร้างให้อัตโนมัติ)
2	ZONE_NAME	VARCHAR(100)			ชื่อเขตที่รับผิดชอบ

- ข้อมูลตาราง MS\_ZONE\_DETAIL

ตารางที่ 4.5 แสดงรายละเอียดตาราง MS\_ZONE\_DETAIL

No	Column	Data Type	PK	FK	Description
1	ZONE_ID	NUMBER	✓		รหัสเขตที่รับผิดชอบ
2	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด



- ข้อมูลตาราง MS\_CUSTOMER

ตารางที่ 4.6 แสดงรายละเอียดตาราง MS\_CUSTOMER

No	Column	Data Type	PK	FK	Description
1	CUSTOMER_ID	NUMBER	✓		รหัสลูกค้า(ระบบสร้างให้อัตโนมัติ)
2	CUSTOMER_NAME	VARCHAR(100)			ชื่อลูกค้า
3	ADDRESS	VARCHAR(100)			ที่อยู่
64	ROAD	VARCHAR(100)			ถนน
5	KHWANG	VARCHAR(50)			แขวง / ตำบล
7	KHET	VARCHAR(50)			เขต / อำเภอ
8	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
9	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
10	PHONE	VARCHAR(30)			เบอร์โทรศัพท์
11	FAX	VARCHAR(30)			แฟกซ์

- ข้อมูลตาราง MS\_DEPOT

ตารางที่ 4.7 แสดงรายละเอียดตาราง MS\_DEPOT

No	Column	Data Type	PK	FK	Description
1	DEPOT_ID	NUMBER	✓		รหัสคลังน้ำมัน(ระบบสร้างให้อัตโนมัติ)
2	DEPOT_NAME	VARCHAR(100)			ชื่อคลังน้ำมัน
3	ADDRESS	VARCHAR(100)			ที่อยู่
4	ROAD_ID	VARCHAR(100)			ถนน
5	KHWANG	VARCHAR(50)			แขวง / ตำบล
6	KHET	VARCHAR(50)			เขต / อำเภอ
7	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
8	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
9	PHONE	VARCHAR(30)			เบอร์โทรศัพท์
10	FAX	VARCHAR(30)			แฟกซ์
11	ZONE_ID	NUMBER		✓	รหัสเขตที่ส่ง

- ข้อมูลตาราง MS\_PRODUCT

ตารางที่ 4.8 แสดงรายละเอียดตาราง MS\_PRODUCT

No	Column	Data Type	PK	FK	Description
1	PRODUCT_ID	NUMBER	<input type="checkbox"/>		รหัสน้ำมันเชื้อเพลิง(ระบบสร้างให้อัตโนมัติ)
2	PRODUCT_CODE	VARCHAR(10)			ชื่อย่อน้ำมันเชื้อเพลิง
3	PRODUCT_NAME	VARCHAR(100)			ชื่อน้ำมันเชื้อเพลิง
4	WEIGHT	NUMBER(10,2)			น้ำหนัก กรัมต่อลิตร

- ข้อมูลตาราง MS\_ZONE

ตารางที่ 4.10 แสดงรายละเอียดตาราง MS\_ZONE

No	Column	Data Type	PK	FK	Description
1	ZONE_ID	NUMBER	<input checked="" type="checkbox"/>		รหัสเขตที่ส่ง(ระบบสร้างให้อัตโนมัติ)
2	ZONE_NAME	VARCHAR(100)			ชื่อเขตที่ส่ง

- ข้อมูลตาราง MS\_TRUCK\_TYPE

ตารางที่ 4.11 แสดงรายละเอียดตาราง MS\_TRUCK\_TYPE

No	Column	Data Type	PK	FK	Description
1	TRUCK_TYPE_ID	NUMBER	<input checked="" type="checkbox"/>		รหัสแบบรถบรรทุก (ระบบสร้างให้อัตโนมัติ)
2	TRUCK_PATTERN	VARCHAR(100)			แบบรถบรรทุก
3	TRUCK_WEIGHT	NUMBER(10,2)			น้ำหนักรถบรรทุก
4	TRUCK_CAPACITY	NUMBER(10,2)			ความจุของรถ
5	NO_OF_TRUCK	NUMBER			จำนวนถังน้ำมัน

- ข้อมูลตาราง MS\_TRUCK\_DETAIL

ตารางที่ 4.12 แสดงรายละเอียดตาราง MS\_TRUCK\_DETAIL

No	Column	Data Type	PK	FK	Description
1	TRUCK_TYPE_ID	NUMBER	✓		รหัสแบบรถบรรทุก
2	TANK_NO	NUMBER	✓		เลขที่ถังน้ำมัน
3	TANK_CAPACITY	NUMBER(10,2)			ความจุของถังน้ำมัน

- ข้อมูลตาราง MS\_TRUCK

ตารางที่ 4.13 แสดงรายละเอียดตาราง MS\_TRUCK\_DETAIL

No	Column	Data Type	PK	FK	Description
1	TRUCK_ID	NUMBER	✓		รหัสรถบรรทุก (ระบบสร้างให้อัตโนมัติ)
2	REGIS_DATE				วันที่ลงทะเบียน
3	REGIS_NO				ทะเบียนรถ
4	TRUCK_TYPE_ID			✓	รหัสแบบรถบรรทุก

- ข้อมูลตาราง MS\_ROAD

ตารางที่ 4.14 แสดงรายละเอียดตาราง MS\_ROAD

No	Column	Data Type	PK	FK	Description
1	ROAD_ID	NUMBER	✓		รหัสถนน(ระบบสร้างให้อัตโนมัติ)
2	ROAD_NAME	NUMBER			ชื่อถนน
3	ROAD_TYPE_ID	NUMBER(10,2)		✓	ความจุของถังน้ำมัน

- ข้อมูลตาราง MS\_DIRECTION

ตารางที่ 4.15 แสดงรายละเอียดตาราง MS\_DIRECTION

No	Column	Data Type	PK	FK	Description
1	DIRECT_ID	NUMBER	✓		รหัสทิศทาง(ระบบสร้างให้อัตโนมัติ)
2	DIRECT_NAME	NUMBER			ทิศทาง

- ข้อมูลตาราง MS\_STATION

ตารางที่ 4.16 แสดงรายละเอียดตาราง MS\_STATION

No	Column	Data Type	PK	FK	Description
1	STATION_ID	NUMBER	✓		เลขที่ของสถานีบริการน้ำมัน(ระบบสร้างให้อัตโนมัติ)
2	STATION_NAME	VARCHAR(100)			ชื่อสถานีบริการน้ำมัน
3	ADDRESS	VARCHAR(100)			ที่อยู่
4	ROAD	VARCHAR(100)			ถนน
5	KHWANG	VARCHAR(50)			แขวง / ตำบล
6	KHET	VARCHAR(50)			เขต / อำเภอ
7	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
8	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
9	PHONE	VARCHAR(30)			เบอร์โทรศัพท์
11	AREA_DET_ID	NUMBER		✓	รหัสรายละเอียดพื้นที่การเดินรถ
12	CUSTOMER_ID	NUMBER		✓	รหัสลูกค้า

- ข้อมูลตารางMS\_STATION\_ROUTE

ตารางที่ 4.17 แสดงรายละเอียดตาราง MS\_STATION\_ROUTE

No	Column	Data Type	PK	FK	Description
1	STATION_ID	NUMBER	✓		รหัสสถานีบริการน้ำมัน
2	STEP_NO	NUMBER			ลำดับเส้นทาง
3	DIRECT_IID	NUMBER			ทิศทาง
4	ROAD_ID	NUMBER			รหัสถนน
2	DISTANCE	NUMBER			ระยะทาง

- ข้อมูลตาราง MS\_ROAD\_TYPE

ตารางที่ 4.18 แสดงรายละเอียดตาราง MS\_ROAD\_TYPE

No	Column	Data Type	PK	FK	Description
1	ROAD_TYPE_ID	NUMBER	✓		รหัสประเภทถนน (ระบบสร้างให้อัตโนมัติ)
2	ROAD_TYPE_NAME	VARCHAR(100)			ชื่อประเภทถนน

- ข้อมูลตาราง MS\_WEIGHT\_LIMIT

ตารางที่ 4.19 แสดงรายละเอียดตาราง MS\_WEIGHT\_LIMIT

No	Column	Data Type	PK	FK	Description
1	WH_LIMIT_ID	NUMBER	✓		รหัสน้ำหนักรถบรรทุก(ระบบสร้างให้อัตโนมัติ)
2	ROAD_TYPE_ID	NUMBER		✓	รหัสประเภทถนนที่ใช้ขนส่ง
3	EFFECTIVE_DATE	DATE			วันที่มีผล
4	TRUCK_TYPE_ID	NUMBER		✓	ประเภทรถบรรทุก
5	WEIGHT_LIMIT	NUMBER(10,2)			น้ำหนักรถบรรทุก (ตัน)

- ข้อมูลตาราง MS\_SCHEDULE

ตารางที่ 4.20 แสดงรายละเอียดตาราง MS\_SCHEDULE

No	Column	Data Type	PK	FK	Description
1	SCHEDULE_ID	NUMBER	✓		รหัสตารางเวลาการเดินทาง(ระบบสร้างให้อัตโนมัติ)
2	AREA_ID	NUMBER		✓	รหัสพื้นที่การเดินทาง
3	ROUTE_ID	NUMBER		✓	รหัสประเภทเส้นทางที่ใช้ขนส่ง
4	START_TIME	NUMBER(5,2)			เวลาเริ่มต้นการเดินทาง
5	END_TIME	NUMBER(5,2)			เวลาสิ้นสุดการเดินทาง

- ข้อมูลตาราง MS\_EXPENSE\_TYPE

ตารางที่ 4.21 แสดงรายละเอียดตาราง MS\_EXPENSE\_TYPE

No	Column	Data Type	PK	FK	Description
1	EXP_TYPE_ID	NUMBER	✓		รหัสค่าใช้จ่าย(ระบบสร้างให้อัตโนมัติ)
2	EXP_TYPE_CODE	VARCHAR(5)			ชื่อย่อ
4	EXP_NAME	VARCHAR(100)			ชื่อค่าใช้จ่าย

- ข้อมูลตาราง TS\_ORDER\_MASTER

ตารางที่ 4.22 แสดงรายละเอียดตาราง TS\_ORDER\_MASTER

No	Column	Data Type	PK	FK	Description
1	ORDER_ID	NUMBER	✓		รหัสใบสั่งซื้อ
2	ORDER_NO	VARCHAR2(9)			เลขที่ใบสั่งซื้อ
3	ORDER_DATE	DATE			วันที่สั่งซื้อ
4	STATION_ID	NUMBER		✓	รหัสสถานีบริการ
5	STATUS	VARCHAR(3)			สถานะใบสั่งซื้อ

- ข้อมูลตาราง TS\_ORDER\_DETAIL

ตารางที่ 4.23 แสดงรายละเอียดตาราง TS\_ORDER\_DETAIL

No	Column	Data Type	PK	FK	Description
1	ORDRE_DET_ID	NUMBER	✓		รหัสรายละเอียดใบสั่งซื้อ
2	ORDER_ID	NUMBER		✓	รหัสใบสั่งซื้อ
3	PRODUCT_ID	NUMBER			รหัสน้ำมันเชื้อเพลิง
4	CAPACITY	NUMBER(10,2)			ปริมาตร

- ข้อมูลตาราง OR\_BATCH\_NAME

ตารางที่ 4.24 แสดงรายละเอียดตาราง OR\_BATCH\_NAME

No	Column	Data Type	PK	FK	Description
1	OR_BATCH_ID	NUMBER	✓		รหัส Batch Order
2	BATCH_NAME	VARCHAR2(100)			ชื่อ Batch
3	BATCH_DATE	DATE			วันที่ส่งสินค้า
4	BATCH_STATUS	VARCHAR2(5)			สถานะ Batch Order

- ข้อมูลตาราง OR\_BATCH\_ALL

ตารางที่ 4.25 แสดงรายละเอียดตาราง OR\_BATCH\_ALL

No	Column	Data Type	PK	FK	Description
1	OR_BATCH_ID	NUMBER	✓		รหัสใบส่งสินค้า
2	ORDER_ID	NUMBER	✓		รหัสใบส่งสินค้า

- ข้อมูลตาราง TS\_DELIVERY\_MASTER

ตารางที่ 4.26 แสดงรายละเอียดตาราง TS\_DELIVERY\_MASTER

No	Column	Data Type	PK	FK	Description
1	DELIVERY_ID	NUMBER	✓		รหัสใบส่งสินค้า
2	DELIVERY_NO	VARCHAR2(9)			เลขที่ใบส่งสินค้า
3	START_DATE	DATE			วันที่เริ่มต้น
4	END_DATE	DATE			วันที่สิ้นสุด
5	START_TIME	NUMBER(5,2)			เวลาเริ่มต้น
6	END_TIME	NUMBER(5,2)			เวลาสิ้นสุด
7	TRUCK_ID	NUMBER		✓	รหัสรถบรรทุก
8	EMPLOYEE_CODE	VARCHAR(10)		✓	รหัสพนักงาน
9	STATUS	VARCHAR(3)			สถานะใบส่งสินค้า
10	DL_BATCH_ID	NUMBER		✓	รหัสงวดการส่งสินค้า

- ข้อมูลตาราง TS\_DELIVERY\_DETAIL1

ตารางที่ 4.27 แสดงรายละเอียดตาราง TS\_DELIVERY\_DETAIL1Y

No	Column	Data Type	PK	FK	Description
1	DELIVERY_DET_ID	NUMBER	✓		รหัสรายละเอียดใบส่งสินค้า
2	DELIVERY_ID	NUMBER		✓	รหัสใบส่งสินค้า
3	ORDER_ID	NUMBER		✓	รหัสใบส่งสินค้า
4	PLAN_START_DATE	DATE			วันที่เริ่มต้น
5	PLAN_END_DATE	DATE			วันที่สิ้นสุด
6	PLAN_START_DATE	NUMBER(5,2)			เวลาเริ่มต้น
7	PLAN_END_DATE	NUMBER(5,2)			เวลาสิ้นสุด
8	REAL_START_DATE	DATE			วันที่ส่งสินค้า
9	REAL_END_DATE	DATE			วันที่สิ้นสุดการส่งสินค้า
10	REAL_START_DATE	NUMBER(5,2)			เวลาส่งสินค้า
11	REAL_END_DATE	NUMBER(5,2)			เวลาสิ้นสุดการส่งสินค้า

- ข้อมูลตาราง TS\_DELIVERY\_DETAIL2

ตารางที่ 4.28 แสดงรายละเอียดตาราง TS\_DELIVERY\_DETAIL2

No	Column	Data Type	PK	FK	Description
1	DELIVERY_DET_ID	NUMBER	✓		รหัสรายละเอียดใบส่งสินค้า
2	DELIVERY_ID	NUMBER		✓	รหัสใบส่งสินค้า
3	TRUCK_ID	NUMBER		✓	รหัสรถบรรทุก
4	TANK_NO	NUMBER			เลขที่ถังน้ำมัน
5	PRODUCT_ID	NUMBER		✓	รหัสน้ำมันเชื้อเพลิง
6	TANK_CAPACITY	NUMBER(10,2)			ปริมาณน้ำมันเชื้อเพลิง

- ข้อมูลตาราง TS\_EXPENSE\_MASTER

ตารางที่ 4.29 แสดงรายละเอียดตาราง TS\_EXPENSE\_MASTER

No	Column	Data Type	PK	FK	Description
1	EXPENSE_ID	NUMBER	✓		รหัสข้อมูลค่าใช้จ่าย
2	DELIVERY_ID	NUMBER		✓	รหัสใบส่งสินค้า



3	TOTAL_AMOUNT	NUMBER(15,2)			ค่าใช้จ่ายรวม
---	--------------	--------------	--	--	---------------

- ข้อมูลตาราง TS\_EXPENSE\_DETAIL

ตารางที่ 4.30 แสดงรายละเอียดตาราง TS\_EXPENSE\_DETAIL

No	Column	Data Type	PK	FK	Description
1	EXPENSE_DET_ID	NUMBER	✓		รหัสข้อมูลรายละเอียดค่าใช้จ่าย
2	EXPENSE_ID	NUMBER		✓	รหัสค่าใช้จ่าย
3	EXP_TYPE_ID	NUMBER		✓	รหัสประเภทค่าใช้จ่าย
4	AMOUNT	NUMBER(15,2)			ค่าใช้จ่าย

- ข้อมูลตาราง AP\_BATCH\_NAME

ตารางที่ 4.31 แสดงรายละเอียดตาราง AP\_BATCH\_NAME

No	Column	Data Type	PK	FK	Description
1	AP_BATCH_ID	NUMBER	✓		รหัสงวดใบแจ้งหนี้
2	AP_BATCH_CODE	VARCHAR2(15)			งวดใบแจ้งหนี้
3	BATCH_NAME	VARCHAR2(100)			รายละเอียดงวดใบแจ้งหนี้
4	BATCH_DATE	DATE			วันที่งวดใบแจ้งหนี้
5	BATCH_STATUS	VARCHAR2(5)			สถานะงวดใบแจ้งหนี้

- ข้อมูลตาราง AP\_BATCH\_ALL

ตารางที่ 4.32 แสดงรายละเอียดตาราง AP\_BATCH\_ALL

No	Column	Data Type	PK	FK	Description
1	AP_BATCH_ID	NUMBER	✓		รหัสงวดใบแจ้งหนี้
2	DELIVERY_ID	NUMBER	✓		รหัสใบส่งสินค้า

- ข้อมูลตาราง TS\_INVOICE\_MASTER

ตารางที่ 4.33 แสดงรายละเอียดตาราง TS\_INVOICE\_MASTER

No	Column	Data Type	PK	FK	Description
1	INVOICE_ID	NUMBER	✓		รหัสใบแจ้งหนี้
2	INVOICE_NO	VARCHAR2(9)			เลขที่ใบแจ้งหนี้

89

3	INVOICE_DATE	DATE			วันที่ใบแจ้งหนี้
4	CUSTOMER_ID	NUMBER		✓	
5	AP_BATCH_ID	NUMBER		✓	รหัสงวดใบแจ้งหนี้
6	STATUS	VARCHAR(3)			สถานะใบแจ้งหนี้
7	TOTAL_AMOUNT	NUMBER(15,2)			จำนวนเงินรวมที่ต้องชำระ

- ข้อมูลตาราง TS\_INVOICE\_DETAIL

ตารางที่ 4.34 แสดงรายละเอียดตาราง TS\_INVOICE\_DETAIL

No	Column	Data Type	PK	FK	Description
1	INVOICE_DET_ID	NUMBER	✓		รหัสรายละเอียดใบแจ้งหนี้
2	INVOICE_ID	NUMBER		✓	รหัสใบแจ้งหนี้
3	DELIVERY_ID	NUMBER		✓	รหัสใบส่งสินค้า
4	AMOUNT	NUMBER(15,2)			จำนวนเงินที่ต้องชำระ

- ข้อมูลตาราง TS\_RECEIPT\_MASTER

ตารางที่ 4.35 แสดงรายละเอียดตาราง TS\_RECEIPT\_MASTER

No	Column	Data Type	PK	FK	Description
1	RECEIPT_ID	NUMBER	✓		รหัสใบเสร็จรับเงิน
2	RECEIPT_NO	VARCHAR2(9)			เลขที่ใบเสร็จรับเงิน
3	RECEIPT_DATE	DATE			วันที่ใบเสร็จรับเงิน
4	CUSTOMER_ID	NUMBER		✓	รหัสลูกค้า
5	INVOICE_ID	NUMBER		✓	รหัสเลขที่ใบแจ้งหนี้
6	TOTAL_AMOUNT	NUMBER(15,2)			จำนวนเงินที่ชำระ

- ข้อมูลตาราง MS\_EMPLOYEE

ตารางที่ 4.36 แสดงรายละเอียดตาราง MS\_EMPLOYEE

No	Column	Data Type	PK	FK	Description
1	EMPL_ID	NUMBER	✓		รหัสพนักงาน(ระบบสร้างให้อัตโนมัติ)
2	EMPL_CODE	VARCHAR(9)			เลขที่พนักงาน
3	FIRST_NAME	VARCHAR(150)			ชื่อพนักงาน
4	LAST_NAME				นามสกุล
5	START_DATE	DATE		✓	วันที่เริ่มงาน
6	ADDRESS	VARCHAR(100)			ที่อยู่
7	ROAD	VARCHAR(100)			ถนน
8	KHWANG	VARCHAR(50)			แขวง / ตำบล
9	KHET	VARCHAR(50)			เขต / อำเภอ
10	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
11	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
12	PHONE	VARCHAR(30)			เบอร์โทรศัพท์
13	EMPLPICT	VARCHAR(30)			ชื่อไฟล์รูปถ่าย

- ข้อมูลตาราง MS\_ASSIGNMENT

ตารางที่ 4.37 แสดงรายละเอียดตาราง MS\_ASSIGNMENT

No	Column	Data Type	PK	FK	Description
1	ASSIGN_ID	NUMBER	✓		รหัสตำแหน่ง(ระบบสร้างให้อัตโนมัติ)
2	EMPL_ID	NUMBER		✓	รหัสพนักงาน
3	POSITION	VARCHAR(40)			ตำแหน่งงาน
4	BRANCH_ID	NUMBER			รหัสสาขา
5	START_DATE	DATE			วันที่เริ่มต้น
6	END_DATE	DATE			วันที่สิ้นสุด

- ข้อมูลตาราง MS\_SHIFT

ตารางที่ 4.38 แสดงรายละเอียดตาราง MS\_SHIFT

No	Column	Data Type	PK	FK	Description
1	SHIFT_ID	VARCHAR(9)	✓		รหัสกะการทำงาน
2	SHIFT_CODE	VARCHAR(20)			ชื่อย่อกะการทำงาน
3	SHIFT_DESC	VARCHAR(150)			รายละเอียดกะการทำงาน
4	TIME_IN	NUMBER(5,2)			เวลาเข้างาน
5	TIME_OUT	NUMBER(5,2)			เวลาออก
6	BREAK_TIME	NUMBER(5,2)			จำนวนชั่วโมงพัก
7	LATE_TIME	NUMBER(5,2)			เวลานับสาย
8	ABSN_TIME	NUMBER(5,2)			เวลานับขาดงาน
9	DAY_FLAG	VARCHAR(3)			กะข้ามวัน

- ข้อมูลตาราง TS\_SHIFT\_ASSIGN

ตารางที่ 4.39 แสดงรายละเอียดตาราง TS\_SHIFT\_ASSIGN

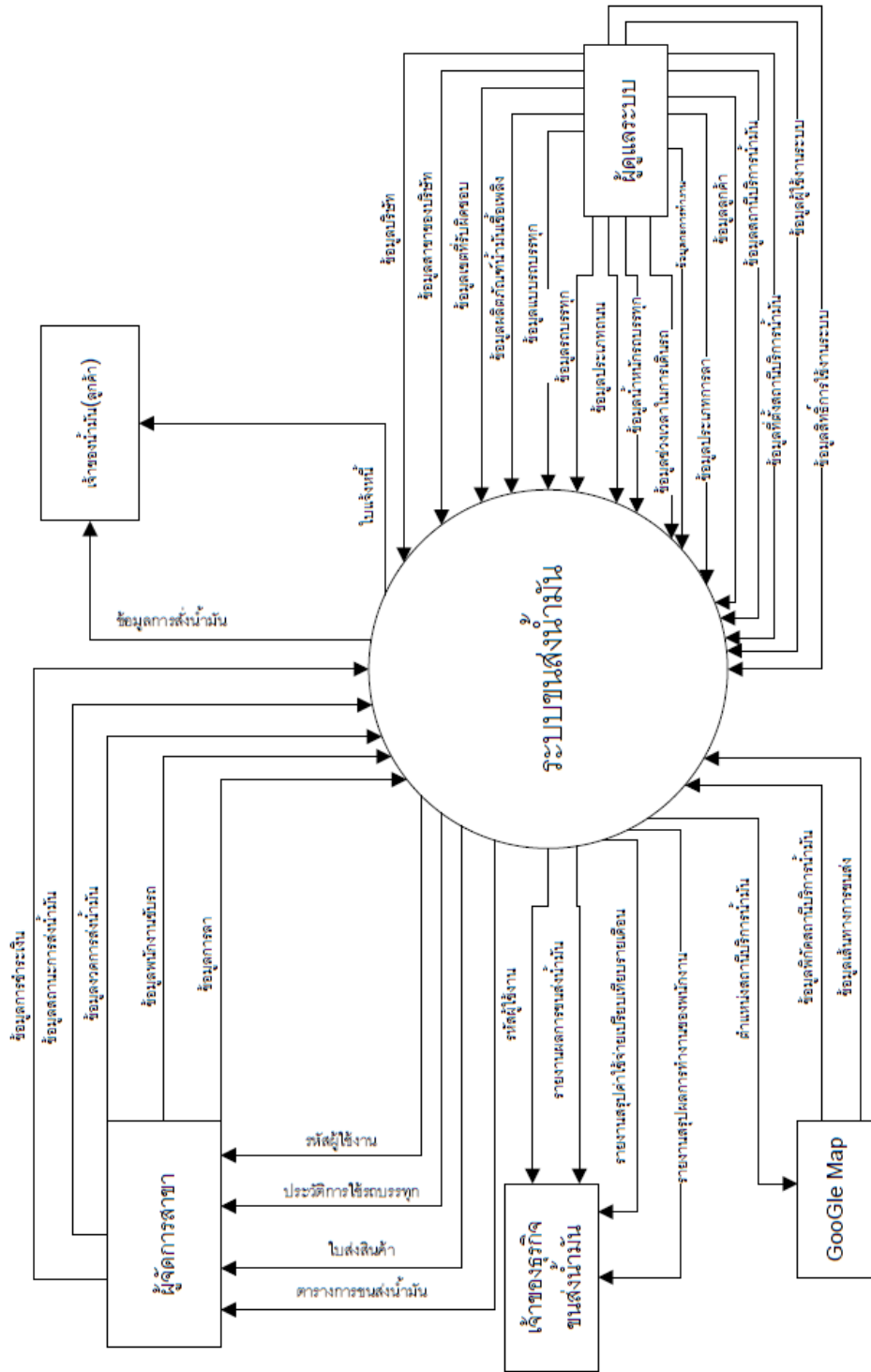
No	Column	Data Type	PK	FK	Description
1	SHIFT_ASSN_ID	VARCHAR(9)	✓		รหัสข้อมูลกะการทำงานของพนักงาน (ระบบสร้างให้อัตโนมัติ)
2	EMPL_ID	VARCHAR(9)		✓	รหัสพนักงาน
3	SHIFT_ID	NUMBER		✓	รหัสกะการทำงาน
4	SHIFT_DATE	DATE			วันที่เริ่มต้น
5	REMARK	VARCHAR(150)			หมายเหตุ

- ข้อมูลตาราง TS\_EMPL\_WORK

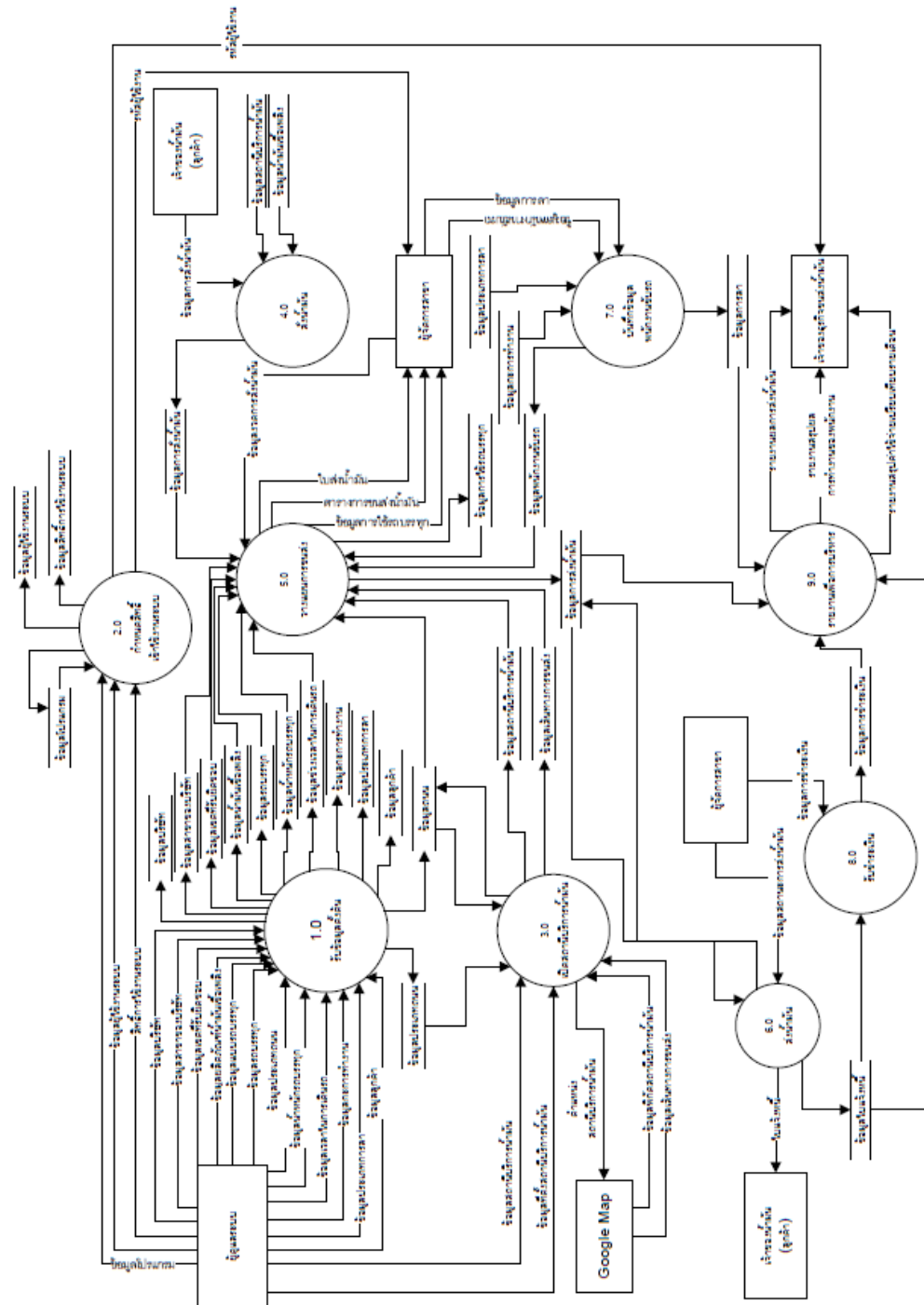
ตารางที่ 4.40 แสดงรายละเอียดตาราง TS\_EMPL\_WORK

No	Column	Data Type	PK	FK	Description
1	EMPL_ID	VARCHAR(9)	✓		รหัสพนักงาน
2	WORK_DATE	DATE	✓		รหัสกะการทำงาน
3	SHIFT_ASSN_ID	NUMBER		✓	รหัสข้อมูลกะการทำงานของพนักงาน
4	WORK_DATE	DATE			วันที่ทำงาน
5	DATE_IN	DATE			วันที่เข้างาน
6	DATE_OUT	DATE			วันที่สิ้นสุด
7	TIME_IN	NUMBER(5,2)			เวลาเข้า
8	TIME_OUT	NUMBER(5,2)			เวลาออก
9	WORK_HOUR	NUMBER(5,2)			ชั่วโมงทำงาน
10	LATE_TIME	NUMBER(5,2)			เวลานับสาย
11	ABSN_TIME	NUMBER(5,2)			เวลานับขาดงาน

ระบบขนส่งน้ำมัน  
รายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว  
แผนภาพบริบท : ระบบขนส่งน้ำมัน

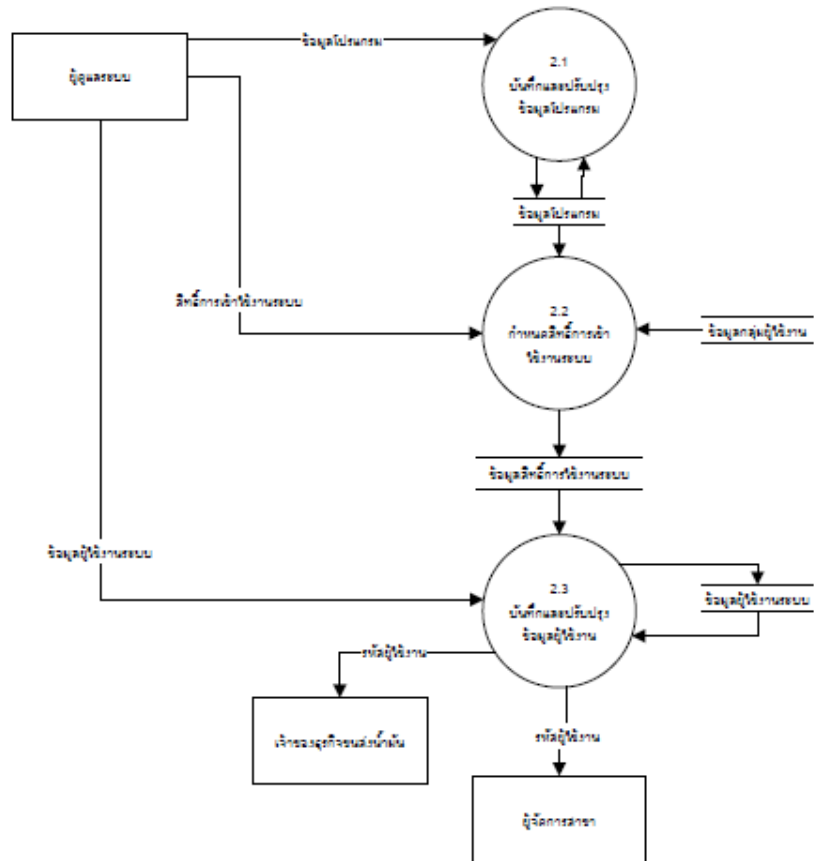


แผนภาพระดับที่ 1 : ระบบขนส่งน้ำมัน



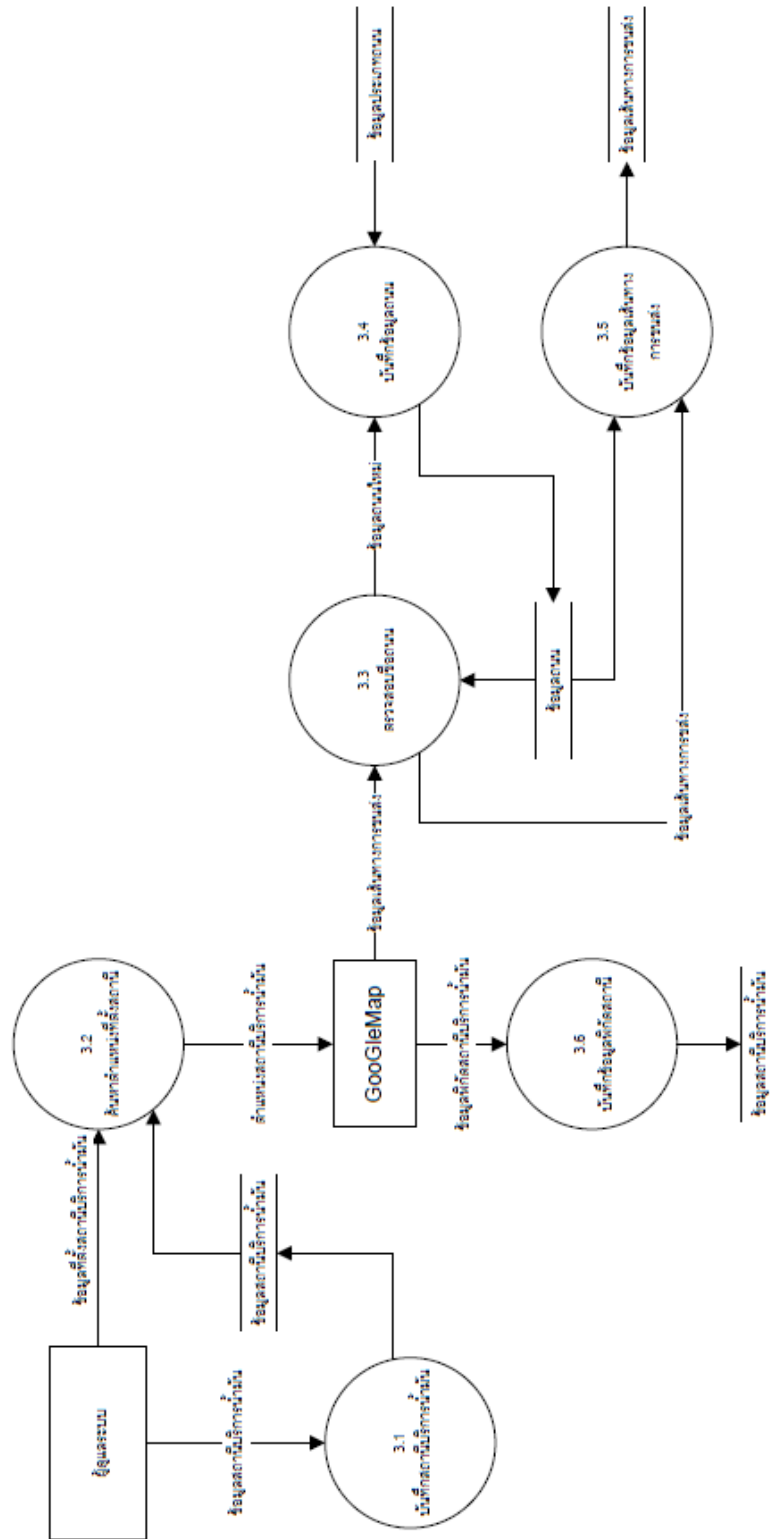
แผนภาพระดับที่ 2 : ระบบขนส่งน้ำมัน

แผนภาพระบบย่อยของกระบวนการที่ 2 กำหนดสิทธิ์การเข้าใช้งานระบบ

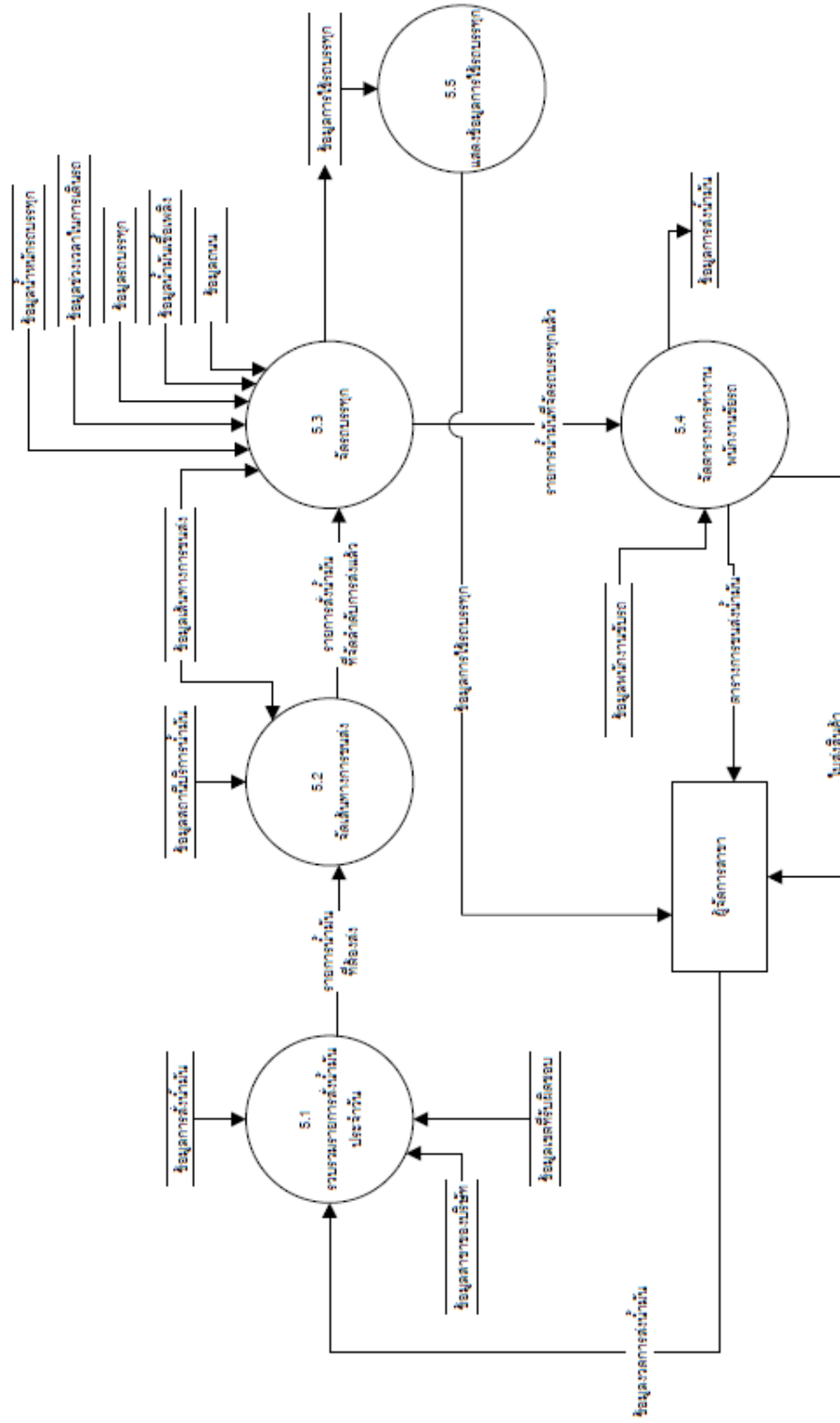




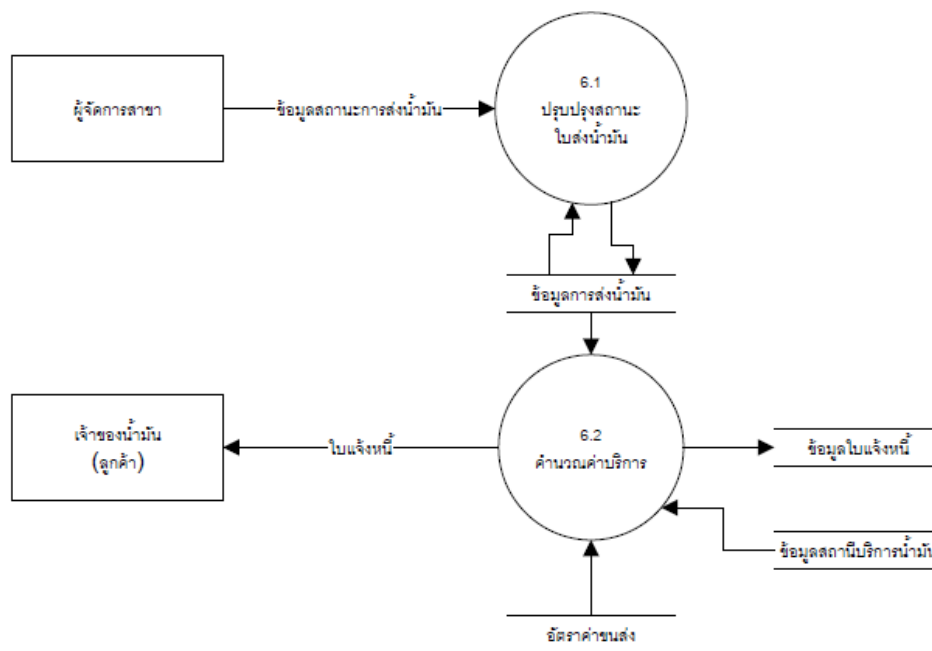
แผนภาพระบบย่อยของกระบวนการที่ 3 เปิดสถานีบริการน้ำมัน



แผนภาพระบวย่อยของกระบวนการที่ 5 วางแผนการขนส่ง



แผนภาพระบบย่อยของกระบวนการที่ 6 ส่งน้ำมัน



พจนานุกรมข้อมูล : ระบบขนส่งน้ำมัน

83

4.9.2 รายละเอียดการออกแบบฐานข้อมูล (Data Dictionary)

• ข้อมูลตาราง MS\_COMPANY

ตารางที่ 4.18 แสดงรายละเอียดตาราง MS\_COMPANY

No	Column	Data Type	PK	FK	Description
1	COMPANY_ID	NUMBER	✓		รหัสบริษัท (ระบบสร้างให้อัตโนมัติ)
2	COMPANY_NAME	VARCHAR(100)			ชื่อบริษัท
3	ADDRESS	VARCHAR(100)			ที่อยู่
4	ROAD	VARCHAR(100)			ถนน
5	KHWANG	VARCHAR(50)			แขวง / ตำบล
6	KHET	VARCHAR(50)			เขต / อำเภอ
7	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
8	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
9	PHONE	VARCHAR(30)			เบอร์โทรศัพท์

• ข้อมูลตาราง MS\_BRANCH

ตารางที่ 4.19 แสดงรายละเอียดตาราง MS\_BRANCH

No	Column	Data Type	PK	FK	Description
1	BRANCH_ID	NUMBER	✓		รหัสสาขาของบริษัท (ระบบสร้างให้อัตโนมัติ)
2	BRANCH_CODE	VARCHAR(5)			ชื่อย่อสาขา
3	BRANCH_NAME	VARCHAR(100)			ชื่อสาขา
4	COMPANY_ID	NUMBER		✓	รหัสบริษัท
5	ADDRESS	VARCHAR(100)			ที่อยู่
6	ROAD	VARCHAR(100)			ถนน
7	KHWANG	VARCHAR(50)			แขวง / ตำบล
8	KHET	VARCHAR(50)			เขต / อำเภอ
9	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
10	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
11	PHONE	VARCHAR(30)			เบอร์โทรศัพท์
13	ZONE_ID	NUMBER		✓	รหัสเขตที่ส่ง

- ข้อมูลตาราง MS\_PROVINCE

ตารางที่ 4.20 แสดงรายละเอียดตาราง MS\_PROVINCE

No	Column	Data Type	PK	FK	Description
1	PROVINCE_ID	NUMBER	✓		รหัสจังหวัด(ระบบสร้างให้อัตโนมัติ)
2	PROVINCE_NAME	VARCHAR(100)			ชื่อจังหวัด

- ข้อมูลตาราง MS\_ZONE

ตารางที่ 4.21 แสดงรายละเอียดตาราง MS\_ZONE

No	Column	Data Type	PK	FK	Description
1	ZONE_ID	NUMBER	✓		รหัสเขตที่รับผิดชอบ (ระบบสร้างให้อัตโนมัติ)
2	ZONE_NAME	VARCHAR(100)			ชื่อเขตที่รับผิดชอบ

- ข้อมูลตาราง MS\_ZONE\_DETAIL

ตารางที่ 4.22 แสดงรายละเอียดตาราง MS\_ZONE\_DETAIL

No	Column	Data Type	PK	FK	Description
1	ZONE_ID	NUMBER	✓		รหัสเขตที่รับผิดชอบ
2	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด

- ข้อมูลตาราง MS\_CUSTOMER

ตารางที่ 4.23 แสดงรายละเอียดตาราง MS\_CUSTOMER

No	Column	Data Type	PK	FK	Description
1	CUSTOMER_ID	NUMBER	✓		รหัสลูกค้า(ระบบสร้างให้อัตโนมัติ)
2	CUSTOMER_NAME	VARCHAR(100)			ชื่อลูกค้า
3	ADDRESS	VARCHAR(100)			ที่อยู่
4	ROAD	VARCHAR(100)			ถนน
5	KHWANG	VARCHAR(50)			แขวง / ตำบล
6	KHET	VARCHAR(50)			เขต / อำเภอ
7	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
8	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์

9	PHONE	VARCHAR(30)			เบอร์โทรศัพท์
10	FAX	VARCHAR(30)			แฟกซ์

- ข้อมูลตาราง MS\_DEPOT

ตารางที่ 4.24 แสดงรายละเอียดตาราง MS\_DEPOT

No	Column	Data Type	PK	FK	Description
1	DEPOT_ID	NUMBER	✓		รหัสคลังน้ำมัน (ระบบสร้างให้อัตโนมัติ)
2	DEPOT_NAME	VARCHAR(100)			ชื่อคลังน้ำมัน
3	ADDRESS	VARCHAR(100)			ที่อยู่
4	ROAD_ID	VARCHAR(100)			ถนน
5	KHWANG	VARCHAR(50)			แขวง / ตำบล
6	KHET	VARCHAR(50)			เขต / อำเภอ
7	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
8	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
9	PHONE	VARCHAR(30)			เบอร์โทรศัพท์
10	ZONE_ID	NUMBER		✓	รหัสเขตที่ส่ง
11	LATITUDE	NUMBER			พิกัดที่ตั้งคลังน้ำมัน
12	LONGITUDE	NUMBER			พิกัดที่ตั้งคลังน้ำมัน

- ข้อมูลตาราง MS\_PRODUCT

ตารางที่ 4.25 แสดงรายละเอียดตาราง MS\_PRODUCT

No	Column	Data Type	PK	FK	Description
1	PRODUCT_ID	NUMBER	✓		รหัสน้ำมันเชื้อเพลิง (ระบบสร้างให้อัตโนมัติ)
2	PRODUCT_CODE	VARCHAR(10)			ชื่อย่อน้ำมันเชื้อเพลิง
3	PRODUCT_NAME	VARCHAR(100)			ชื่อน้ำมันเชื้อเพลิง
4	WEIGHT	NUMBER(10,2)			น้ำหนัก กรัมต่อลิตร

- ข้อมูลตาราง MS\_TRUCK\_PATTERN

ตารางที่ 4.26 แสดงรายละเอียดตาราง MS\_TRUCK\_PATTERN

No	Column	Data Type	PK	FK	Description
1	TRUCK_PATTERN_ID	NUMBER	✓		รหัสแบบรถบรรทุก (ระบบสร้างให้อัตโนมัติ)
2	TRUCK_PATTERN	VARCHAR(100)			แบบรถบรรทุก
3	PATTERN_NO	NUMBER			หมายเลขแบบรถ
4	TRUCK_CAPACITY	NUMBER(10,2)			ความจุของรถ
5	TRUCK_WEIGHT	NUMBER(10,2)			น้ำหนักรถเปล่า
6	NO_OF_TANK	NUMBER			จำนวนถังน้ำมัน
7	TRUCK_LOAD	NUMBER			น้ำหนักบรรทุก

- ข้อมูลตาราง MS\_PATTERN\_DETAIL

ตารางที่ 4.27 แสดงรายละเอียดตาราง MS\_PATTERN\_DETAIL

No	Column	Data Type	PK	FK	Description
1	TRUCK_PATTERN_ID	NUMBER	✓		รหัสแบบรถบรรทุก
2	TANK_NO	NUMBER	✓		หมายเลขถังน้ำมัน
3	TANK_CAPACITY	NUMBER(10,2)			ความจุของถังน้ำมัน
4	PRODUCT_ID	NUMBER		✓	รหัสน้ำมันเชื้อเพลิง

- ข้อมูลตาราง MS\_TRUCK

ตารางที่ 4.28 แสดงรายละเอียดตาราง MS\_TRUCK

No	Column	Data Type	PK	FK	Description
1	TRUCK_ID	NUMBER	✓		รหัสรถบรรทุก (ระบบสร้างให้อัตโนมัติ)
2	TRUCK_DATE				วันที่บันทึก
3	REGIS_DATE				วันที่ลงทะเบียน
4	REGIS_NO				ทะเบียนรถ
5	TRUCK_PATTERN_ID			✓	รหัสแบบรถบรรทุก

- ข้อมูลตาราง MS\_ROAD\_TYPE

ตารางที่ 4.29 แสดงรายละเอียดตาราง MS\_ROAD\_TYPE

No	Column	Data Type	PK	FK	Description
1	ROAD_TYPE_ID	NUMBER	✓		รหัสประเภทถนน (ระบบสร้างให้อัตโนมัติ)
2	ROAD_TYPE_NAME	NUMBER			ประเภทถนน

- ข้อมูลตาราง MS\_ROAD

ตารางที่ 4.30 แสดงรายละเอียดตาราง MS\_ROAD

No	Column	Data Type	PK	FK	Description
1	ROAD_ID	NUMBER	✓		รหัสถนน (ระบบสร้างให้อัตโนมัติ)
2	ROAD_NAME	NUMBER			ชื่อถนน
3	ROAD_TYPE_NAME	NUMBER		✓	รหัสประเภทถนน

- ข้อมูลตาราง MS\_STATION

ตารางที่ 4.31 แสดงรายละเอียดตาราง MS\_STATION

No	Column	Data Type	PK	FK	Description
1	STATION_ID	NUMBER	✓		รหัสสถานีบริการน้ำมัน (ระบบสร้างให้อัตโนมัติ)
2	STATION_CODE	VARCHAR(10)			หมายเลขสถานีบริการน้ำมัน
3	STATION_NAME	VARCHAR(100)			ชื่อสถานีบริการน้ำมัน
4	ADDRESS	VARCHAR(100)			ที่อยู่
5	ROAD	VARCHAR(100)			ถนน
6	KHWANG	VARCHAR(50)			แขวง / ตำบล
7	KHET	VARCHAR(50)			เขต / อำเภอ
8	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
9	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
10	PHONE	VARCHAR(30)			เบอร์โทรศัพท์
11	BRANCH_ID	NUMBER		✓	รหัสสาขา



- ข้อมูลตาราง MS\_STATION\_ROUTE

ตารางที่ 4.32 แสดงรายละเอียดตาราง MS\_STATION\_ROUTE

No	Column	Data Type	PK	FK	Description
1	STATION_ID	NUMBER	✓		รหัสสถานีบริการน้ำมัน
2	STEP_NO	NUMBER			ลำดับเส้นทาง
3	ROAD_ID	NUMBER		✓	รหัสถนน
4	DISTANCE	NUMBER			ระยะทาง

- ข้อมูลตาราง ROUTE\_ELEMENT1

ตารางที่ 4.33 แสดงรายละเอียดตาราง ROUTE\_ELEMENT1

No	Column	Data Type	PK	FK	Description
1	NODE_ID	NUMBER	✓		รหัส JointNode (ระบบสร้างให้อัตโนมัติ)
2	ROAD_ID	NUMBER		✓	รหัสถนน
3	DIST_TOTAL	NUMBER			ระยะทางรวม
4	DIST_VALUE	NUMBER			ระยะทางระหว่าง Joint Node
5	SUB_ROUTE	VARCHAR2(1)			ประเภท Joint Node - ค่า 'Y' = Joint Node - ค่า 'N' = Station Node
6	BRANCH_ID	NUMBER			รหัสสาขา
7	WEIGHT_LIMIT	NUMBER(6,2)			น้ำหนักบรรทุกสูงสุด

- ข้อมูลตาราง ROUTE\_ELEMENT2

ตารางที่ 4.34 แสดงรายละเอียดตาราง ROUTE\_ELEMENT2

No	Column	Data Type	PK	FK	Description
1	NODE_ID	NUMBER			รหัส Joint Node
2	PARENT_ID	NUMBER			รหัส ParentNode

- ข้อมูลตาราง ROUTE\_ELEMENT3

ตารางที่ 4.35 แสดงรายละเอียดตาราง ROUTE\_ELEMENT3

No	Column	Data Type	PK	FK	Description
1	NODE_ID	NUMBER	✓		รหัส Joint Node
2	STATION_ID	NUMBER		✓	รหัสสถานีบริการน้ำมัน

- ข้อมูลตาราง MS\_WEIGHT\_LIMIT

ตารางที่ 4.36 แสดงรายละเอียดตาราง MS\_WEIGHT\_LIMIT

No	Column	Data Type	PK	FK	Description
1	WH_LIMIT_ID	NUMBER	✓		รหัสน้ำหนักรถบรรทุก (ระบบสร้างให้อัตโนมัติ)
2	ROAD_TYPE_ID	NUMBER		✓	รหัสประเภทถนนที่ใช้ขนส่ง
3	EFFECTIVE_DATE	DATE			วันที่มีผล
4	TRUCK_TYPE_ID	NUMBER		✓	ประเภทรถบรรทุก
5	WEIGHT_LIMIT	NUMBER(10,2)			น้ำหนักรถบรรทุก(ตัน)

- ข้อมูลตาราง MS\_SCHEDULE

ตารางที่ 4.37 แสดงรายละเอียดตาราง MS\_SCHEDULE

No	Column	Data Type	PK	FK	Description
1	SCHEDULE_ID	NUMBER	✓		รหัสตารางเวลาการเดินทาง (ระบบสร้างให้อัตโนมัติ)
2	ZONE_ID	NUMBER		✓	รหัสเขตที่รับผิดชอบ
3	ROUTE_ID	NUMBER		✓	รหัสประเภทเส้นทางที่ใช้ขนส่ง
4	START_TIME	NUMBER(5,2)			เวลาเริ่มต้นการเดินทาง
5	END_TIME	NUMBER(5,2)			เวลาสิ้นสุดการเดินทาง

- ข้อมูลตาราง TS\_ORDER\_MASTER

ตารางที่ 4.38 แสดงรายละเอียดตาราง TS\_ORDER\_MASTER

No	Column	Data Type	PK	FK	Description
1	ORDER_ID	NUMBER	✓		รหัสใบสั่งน้ำมัน (ระบบสร้างให้อัตโนมัติ)
2	ORDER_NO	VARCHAR2(9)			เลขที่ใบสั่งน้ำมัน
3	ORDER_DATE	DATE			วันที่สั่งซื้อ
4	STATION_ID	NUMBER		✓	รหัสสถานีบริการน้ำมัน
5	OR_BATCH_ID	NUMBER		✓	รหัสงวดการสั่งน้ำมัน

- ข้อมูลตาราง TS\_ORDER\_DETAIL

ตารางที่ 4.39 แสดงรายละเอียดตาราง TS\_ORDER\_DETAIL

No	Column	Data Type	PK	FK	Description
1	ORDRE_DET_ID	NUMBER	✓		รหัสรายละเอียดใบสั่งน้ำมัน (ระบบสร้างให้อัตโนมัติ)
2	ORDER_ID	NUMBER		✓	รหัสใบสั่งน้ำมัน
3	PRODUCT_ID	NUMBER			รหัสน้ำมันเชื้อเพลิง
4	CAPACITY	NUMBER(10,2)			ปริมาณน้ำมัน

- ข้อมูลตาราง OR\_BATCH\_NAME

ตารางที่ 4.40 แสดงรายละเอียดตาราง OR\_BATCH\_NAME

No	Column	Data Type	PK	FK	Description
1	OR_BATCH_ID	NUMBER	✓		รหัส Batch Order (ระบบสร้างให้อัตโนมัติ)
2	OR_BATCH_CODE	VARCHAR2(9)			หมายเลขงวดการสั่งน้ำมัน (Batch Code)
2	BATCH_NAME	VARCHAR2(100)			ชื่อ Batch
3	BATCH_DATE	DATE			วันที่สั่งซื้อ
4	BATCH_STATUS	VARCHAR2(5)			สถานะ Batch Order

- ข้อมูลตาราง TS\_DELIVERY\_MASTER

ตารางที่ 4.41 แสดงรายละเอียดตาราง TS\_DELIVERY\_MASTER

No	Column	Data Type	PK	FK	Description
1	DELIVERY_ID	NUMBER	✓		รหัสใบส่งน้ำมัน (ระบบสร้างให้อัตโนมัติ)
2	DELIVERY_NO	VARCHAR2(9)			เลขที่ใบส่งน้ำมัน
3	START_DATE	DATE			วันที่เริ่มต้น
4	START_TIME	NUMBER(5,2)			เวลาเริ่มต้น
5	END_TIME	NUMBER(5,2)			เวลาสิ้นสุด
6	TRUCK_ID	NUMBER		✓	รหัสรถบรรทุก
7	EMPL_ID	NUMBER		✓	รหัสพนักงาน
8	STATUS	VARCHAR(3)			สถานะใบส่งน้ำมัน
9	OR_BATCH_ID	NUMBER		✓	รหัสงวดการส่งน้ำมัน

- ข้อมูลตาราง TS\_DELIVERY\_DETAIL

ตารางที่ 4.42 แสดงรายละเอียดตาราง TS\_DELIVERY\_DETAIL

No	Column	Data Type	PK	FK	Description
1	DELIVERY_DET_ID	NUMBER	✓		รหัสรายละเอียดใบส่งน้ำมัน
2	DELIVERY_ID	NUMBER		✓	รหัสใบส่งน้ำมัน
3	STATION_ID	NUMBER		✓	รหัสสถานีบริการน้ำมัน
4	ORDER_DET_ID	NUMBER		✓	รหัสใบสั่งน้ำมัน

- ข้อมูลตาราง TS\_INVOICE\_MASTER

ตารางที่ 4.43 แสดงรายละเอียดตาราง TS\_INVOICE\_MASTER

No	Column	Data Type	PK	FK	Description
1	INVOICE_ID	NUMBER	✓		รหัสใบแจ้งหนี้
2	INVOICE_NO	VARCHAR2(9)			เลขที่ใบแจ้งหนี้
3	INVOICE_DATE	DATE			วันที่ใบแจ้งหนี้
4	CUSTOMER_ID	NUMBER		✓	
5	TOTAL_AMOUNT	NUMBER(15,2)			จำนวนเงินรวมที่ต้องชำระ

- ข้อมูลตาราง TS\_INVOICE\_DETAIL

ตารางที่ 4.44 แสดงรายละเอียดตาราง TS\_INVOICE\_DETAIL

No	Column	Data Type	PK	FK	Description
1	INVOICE_DET_ID	NUMBER	✓		รหัสรายละเอียดใบแจ้งหนี้
2	INVOICE_ID	NUMBER		<input type="checkbox"/>	รหัสใบแจ้งหนี้
3	DELIVERY_ID	NUMBER		<input type="checkbox"/>	รหัสใบส่งน้ำมัน
4	AMOUNT	NUMBER(15,2)			จำนวนเงินที่ต้องชำระ

- ข้อมูลตาราง TS\_RECEIPT\_MASTER

ตารางที่ 4.45 แสดงรายละเอียดตาราง TS\_RECEIPT\_MASTER

No	Column	Data Type	PK	FK	Description
1	RECEIPT_ID	NUMBER	✓		รหัสใบเสร็จรับเงิน
2	RECEIPT_NO	VARCHAR2(9)			เลขที่ใบเสร็จรับเงิน
3	RECEIPT_DATE	DATE			วันที่ใบเสร็จรับเงิน
4	CUSTOMER_ID	NUMBER		✓	รหัสลูกค้า
5	INVOICE_ID	NUMBER		✓	รหัสเลขที่ใบแจ้งหนี้
6	TOTAL_AMOUNT	NUMBER(15,2)			จำนวนเงินที่ชำระ

- ข้อมูลตาราง MS\_EMPLOYEE

ตารางที่ 4.46 แสดงรายละเอียดตาราง MS\_EMPLOYEE

No	Column	Data Type	PK	FK	Description
1	EMPL_ID	NUMBER	✓		รหัสพนักงาน (ระบบสร้างให้อัตโนมัติ)
2	EMPL_CODE	VARCHAR(9)			เลขที่พนักงาน
3	FIRST_NAME	VARCHAR(150)			ชื่อพนักงาน
4	LAST_NAME				นามสกุล
5	START_DATE	DATE		✓	วันที่เริ่มงาน
6	ADDRESS	VARCHAR(100)			ที่อยู่
7	ROAD	VARCHAR(100)			ถนน
8	KHWANG	VARCHAR(50)			แขวง / ตำบล

9	KHET	VARCHAR(50)			เขต / อำเภอ
10	PROVINCE_ID	NUMBER		✓	รหัสจังหวัด
11	ZIPCODE	VARCHAR(20)			รหัสไปรษณีย์
12	PHONE	VARCHAR(30)			เบอร์โทรศัพท์

- ข้อมูลตาราง MS\_SHIFT

ตารางที่ 4.47 แสดงรายละเอียดตาราง MS\_SHIFT

No	Column	Data Type	PK	FK	Description
1	SHIFT_ID	VARCHAR(9)	✓		รหัสกะการทำงาน
2	SHIFT_CODE	VARCHAR(20)			ชื่อย่อกะการทำงาน
3	SHIFT_DESC	VARCHAR(150)			รายละเอียดกะการทำงาน
4	TIME_IN	NUMBER(5,2)			เวลาเข้างาน
5	TIME_OUT	NUMBER(5,2)			เวลาออก
6	BREAK_TIME	NUMBER(5,2)			จำนวนชั่วโมงพัก
7	DAY_FLAG	VARCHAR(3)			กะข้ามวัน

- ข้อมูลตาราง TS\_SHIFT\_ASSIGN

ตารางที่ 4.48 แสดงรายละเอียดตาราง TS\_SHIFT\_ASSIGN

No	Column	Data Type	PK	FK	Description
1	SHIFT_ASSN_ID	VARCHAR(9)	✓		รหัสข้อมูลกะการทำงานของพนักงาน (ระบบสร้างให้อัตโนมัติ)
2	EMPL_ID	VARCHAR(9)		✓	รหัสพนักงาน
3	SHIFT_ID	NUMBER		✓	รหัสกะการทำงาน
4	SHIFT_DATE	DATE			วันที่เริ่มต้น
5	REMARK	VARCHAR(150)			หมายเหตุ

- ข้อมูลตาราง MS\_LEAVE\_TYPE

ตารางที่ 4.49 แสดงรายละเอียดตาราง MS\_LEAVE\_TYPE

No	Column	Data Type	PK	FK	Description
1	LEAVE_ID	VARCHAR(9)	✓		รหัสประเภทการลา (ระบบสร้างให้อัตโนมัติ)
2	LEAVE_CODE	VARCHAR(20)			ชื่อย่อประเภทการลา
3	LEAVE_NAME	VARCHAR(150)			ประเภทการลา

- ข้อมูลตาราง TS\_EMPL\_LEAVE

ตารางที่ 4.50 แสดงรายละเอียดตาราง TS\_EMPL\_LEAVE

No	Column	Data Type	PK	FK	Description
1	EMPL_ID	VARCHAR(9)	✓		รหัสพนักงาน
2	LEAVE_DATE	DATE	✓		วันที่ลา
3	LEAVE_ID	VARCHAR(9)			รหัสประเภทการลา

- ข้อมูลตาราง MS\_USER\_GROUP

ตารางที่ 4.51 แสดงรายละเอียดตาราง MS\_USER\_GROUP

No	Column	Data Type	PK	FK	Description
1	GROUP_ID	NUMBER	✓		รหัสกลุ่มผู้ใช้งาน (ระบบสร้างให้อัตโนมัติ)
2	GROUP_NAME	VARCHAR2(9)			ชื่อกลุ่มผู้ใช้งานระบบ
3	GROUP_DESC	VARCHAR2(100)			รายละเอียดกลุ่มผู้ใช้งาน

- ข้อมูลตาราง MS\_PROGRAM

ตารางที่ 4.52 แสดงรายละเอียดตาราง MS\_PROGRAM

No	Column	Data Type	PK	FK	Description
1	PROGRAM_ID	NUMBER	✓		รหัสโปรแกรม
2	PROGRAM_CODE	VARCHAR2(9)			ชื่อย่อโปรแกรม
3	PROGRAM_NAME	VARCHAR2(100)			ชื่อโปรแกรม

- ข้อมูลตาราง MS\_PERMISSION

ตารางที่ 4.53 แสดงรายละเอียดตาราง MS\_PERMISSION

No	Column	Data Type	PK	FK	Description
1	PERMISS_ID	NUMBER	✓		รหัสสิทธิ์การใช้โปรแกรม
2	GROUP_ID	NUMBER		✓	รหัสกลุ่มผู้ใช้งาน
3	PROGRAM_ID	NUMBER		✓	รหัสโปรแกรม

- ข้อมูลตาราง MS\_USER

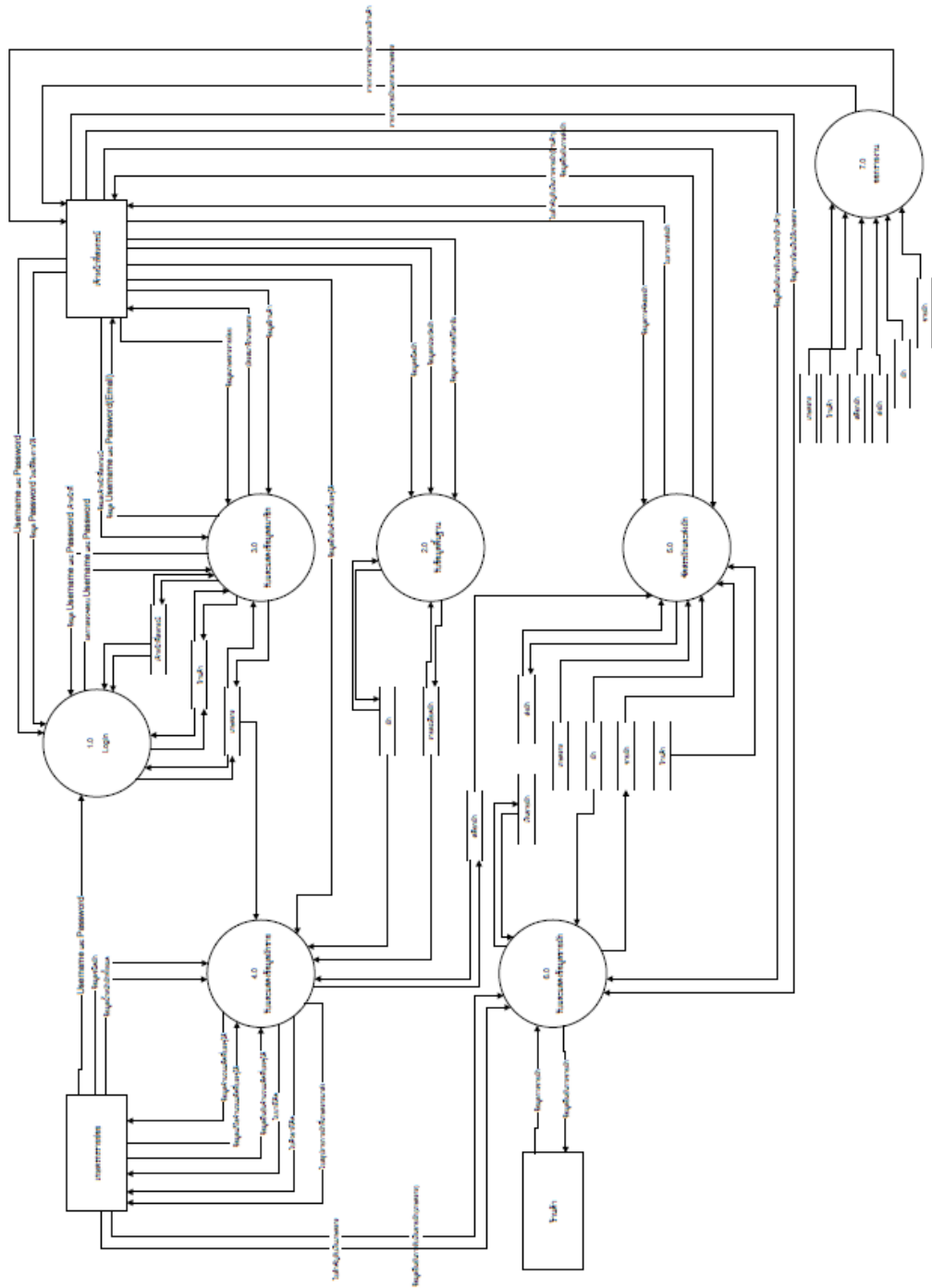
ตารางที่ 4.54 แสดงรายละเอียดตาราง MS\_USER

No	Column	Data Type	PK	FK	Description
1	USER_ID	NUMBER	✓		รหัสผู้ใช้งาน (ระบบสร้างให้อัตโนมัติ)
2	USER_CODE	VARCHAR2(9)			ชื่อผู้ใช้งานระบบ
3	USER_NAME	VARCHAR2(100)			ชื่อผู้ใช้งาน
4	USER_PASS	VARCHAR2(10)			รหัสผ่าน
5	USER_TYPE	VARCHAR2(3)			ประเภทผู้ใช้งาน
6	START_DATE	DATE			วันที่เริ่มต้นการใช้งาน
7	END_DATE	DATE			วันที่สิ้นสุดการใช้งาน
8	GROUP_ID	NUMBER		✓	รหัสกลุ่มผู้ใช้งาน





แผนภาพระดับที่ 1 : ซอฟต์แวร์สำหรับจัดจำหน่ายผลิตภัณฑ์จากเกษตรกรรายย่อย



พจนานุกรมข้อมูล : ซอฟต์แวร์สำหรับจัดจำหน่ายผลิตผลจากเกษตรกรรายย่อย

รายละเอียดการออกแบบฐานข้อมูล

Table: VEGETABLE (ผัก)

ตารางที่ 0-1 แสดงรายละเอียดตาราง VEGETABLE

No.	Column Name	Data Type	PK	FK	Description
1	VETGET_ID	INT	Y		รหัสผัก
2	VETGET_NAME	VARCHAR			ชื่อผัก

Table: MEASUREMENT (หน่วยวัดผัก)

ตารางที่ 0-2 แสดงรายละเอียดตาราง MEASUREMENT

No.	Column Name	Data Type	PK	FK	Description
1	MEASUR_ID	INT	Y		รหัสหน่วยวัด
2	SIZE	INT			ขนาด
3	weight	INT			น้ำหนัก

Table: VETGETABLE\_MEASUREMENT\_DETAIL (รายละเอียดหน่วยวัดผัก)

ตารางที่ 0-3 แสดงรายละเอียดตาราง VETGETABLE\_MEASUREMENT\_DETAIL

No.	Column Name	Data Type	PK	FK	Description
1	VET_MEASUR_ID	INT	Y		รหัส รายละเอียด หน่วยวัด
2	VETGET_ID	INT		Y	รหัสผัก
3	MEASUR_ID	INT		Y	รหัสหน่วยวัด
4	PRICE	FLOAT			ราคา

Table: STAFF (เจ้าหน้าที่สหกรณ์)

ตารางที่ 0-4 แสดงรายละเอียดตาราง STAFF

No.	Column Name	Data Type	PK	FK	Description
1	STAFF_ID	INT	Y		รหัสเจ้าหน้าที่สหกรณ์
2	STAFF_NAME	VARCHAR			ชื่อเจ้าหน้าที่สหกรณ์
3	CITIZEN_ID	INT			เลขที่บัตรประชาชน
4	STAFF_TEL	INT			เบอร์โทรศัพท์เจ้าหน้าที่สหกรณ์
5	USERNAME	VARCHAR			ชื่อที่อยู่ในระบบ
6	PASSWORD	VARCHAR			รหัสผ่าน
7	EMAIL	VARCHAR			อีเมล

Table: FARMER (เกษตรกรรายย่อย)

ตารางที่ 0-5 แสดงรายละเอียดตาราง FARMER

No.	Column Name	Data Type	PK	FK	Description
1	FARMER_ID	INT	Y		รหัสสมาชิกเกษตรกร
2	FARMER_NAME	VARCHAR			ชื่อเกษตรกร
3	FARMER_ADDRESS	VARCHAR			ที่อยู่เกษตรกร

No.	Column Name	Data Type	PK	FK	Description
4	FARM_NAME	VARCHAR			ชื่อสวน
5	CITIZEN_ID	INT			เลขที่บัตรประชาชน
6	FARMER_TEL	VARCHAR			เบอร์โทรศัพท์
7	PASSWORD	VARCHAR			รหัสผ่าน

Table: SHOP (ร้านค้า)

ตารางที่ 0-6 แสดงรายละเอียดตาราง SHOP

No.	Column Name	Data Type	PK	FK	Description
1	SHOP_ID	INT	Y		รหัสร้านค้า
2	SHOP_NAME	VARCHAR			ชื่อร้านค้า
3	COMMERCIAL_ID	INT			เลขทะเบียนพาณิชย์
4	SHOP_ADDRESS	VARCHAR			ที่อยู่ร้านค้า
5	CONTACT_NAME	VARCHAR			ชื่อผู้ติดต่อ
6	SHOP_TEL	INT			เบอร์โทรศัพท์
7	LAT_LON	TEXT			พิกัดร้านค้า
8	SHOP_EMAIL	VARCHAR			อีเมล

Table: STOCK\_VEGETABLE (สต็อกผัก)

ตารางที่ 0-7 แสดงรายละเอียดตาราง STOCK\_VEGETABLE

No.	Column Name	Data Type	PK	FK	Description
1	STOCK_ID	INT	Y		รหัสสต็อกผัก
2	VETGET_ID	INT		Y	รหัสผัก
3	FARMER_ID	INT		Y	รหัสสมาชิก

No.	Column Name	Data Type	PK	FK	Description
					เกษตรกร
4	STOCK_CODE	INT			รหัสรอบการส่ง
5	DATE_TIME	DATE			วันเดือนปี
6	QUANTITY	INT			จำนวนผัก (แพ็ค)
7	PRICE	FLOAT			ราคา

Table: TRANSPORT\_VEGETABLE (ขนส่งผัก)

ตารางที่ 0-8 แสดงรายละเอียดตาราง TRANSPORT\_VEGETABLE

No.	Column Name	Data Type	PK	FK	Description
1	TRANSPORT_ID	INT	Y		รหัสขนส่ง
2	STOCK_ID	INT		Y	รหัสสต็อกผัก
3	SHOP_ID	INT		Y	รหัสร้านค้า
4	STAFF_ID	INT		Y	รหัสเจ้าหน้าที่ สหกรณ์
5	STATUS_ID	INT		Y	รหัส สถานะการส่ง ผัก
6	DATE_TIME	DATE			วันเดือนปี ขนส่งผัก
7	AMOUNT	INT			จำนวนผักที่ส่ง

Table: SELL\_VEGETABLE (ขายผัก)

ตารางที่ 0-9 แสดงรายละเอียดตาราง SELL\_VEGETABLE

No.	Column Name	Data Type	PK	FK	Description
1	SELL_ID	INT	Y		รหัสขายผัก
2	STOCK_ID	INT		Y	รหัสสต็อก
3	SHOP_ID	INT		Y	รหัสร้านค้า
4	TRANSPORT_ID	INT		Y	รหัสขนส่ง
5	DATE_TIME	DATE			วันเดือนปีที่ขาย
6	PRICE	FLOAT			ราคาขาย
7	AMOUNT	INT			จำนวนผักขาย

Table: STATUS (สถานะ)

ตารางที่ 0-10 แสดงรายละเอียดตาราง STATUS

No.	Column Name	Data Type	PK	FK	Description
1	STATUS_ID	INT	Y		รหัสสถานะ
2	STATUS_NAME	VARCHAR			ชื่อสถานะ

Table: PAYMENT\_FARMER (เงินขายผักเกษตรกร)

ตารางที่ 0-11 แสดงรายละเอียดตาราง FABRIC

No.	Column Name	Data Type	PK	FK	Description
1	PAYMENT_ID	INT	Y		รหัสการจ่ายเงิน
2	FARMER_ID	INT		Y	รหัสเกษตรกร
3	STAFF_ID	INT		Y	รหัสเจ้าหน้าที่สหกรณ์

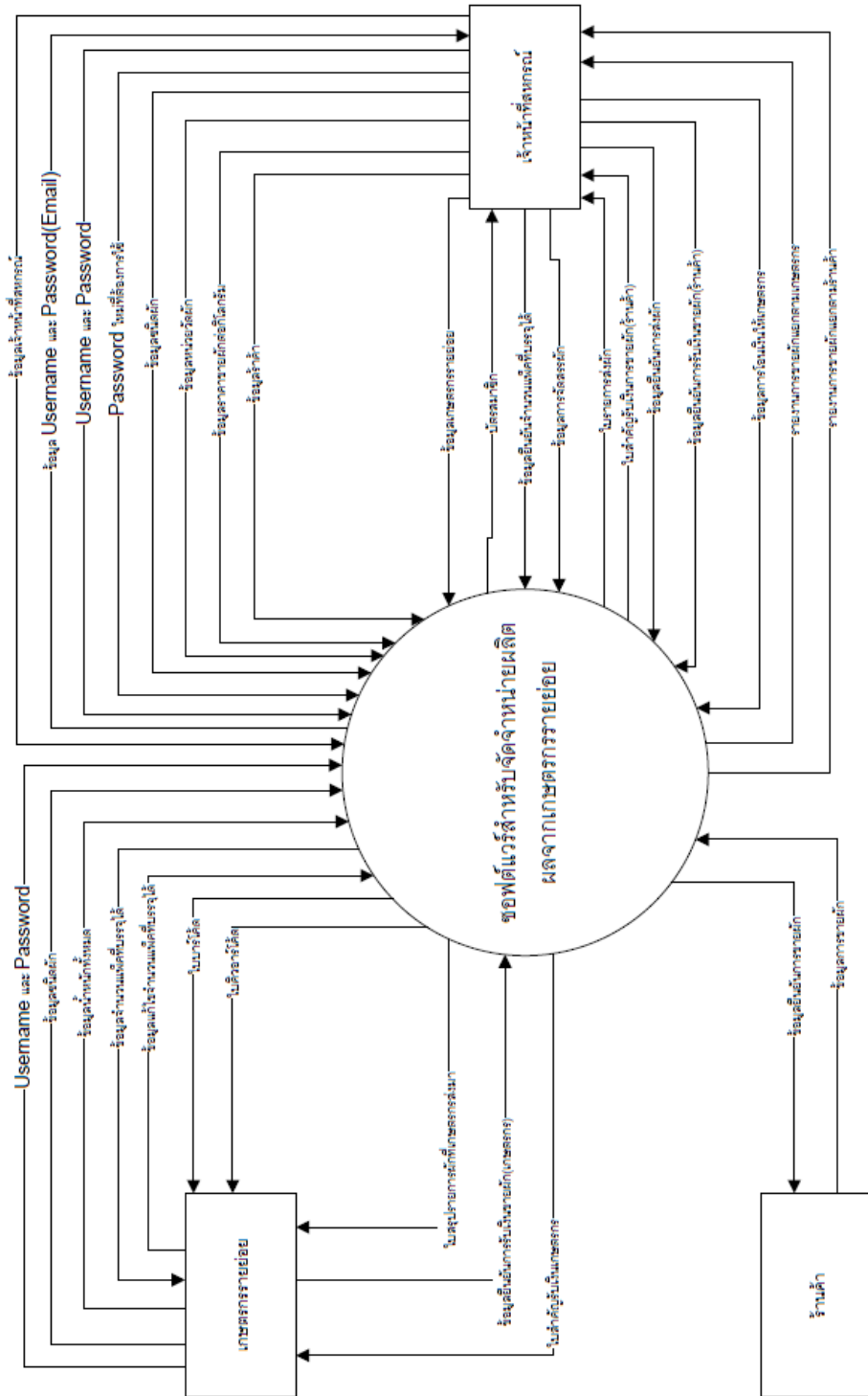
No.	Column Name	Data Type	PK	FK	Description
4	STOCK_ID	INT		Y	รหัสสต็อก
5	STATUS_ID	INT		Y	รหัสสถานะ
6	AMOUNT	INT			ยอดเงิน
7	DATE_TIME	DATE			วันเดือนปี
8	IMG_FILE	VARCHAR			รูปใบสำคัญ รับเงิน



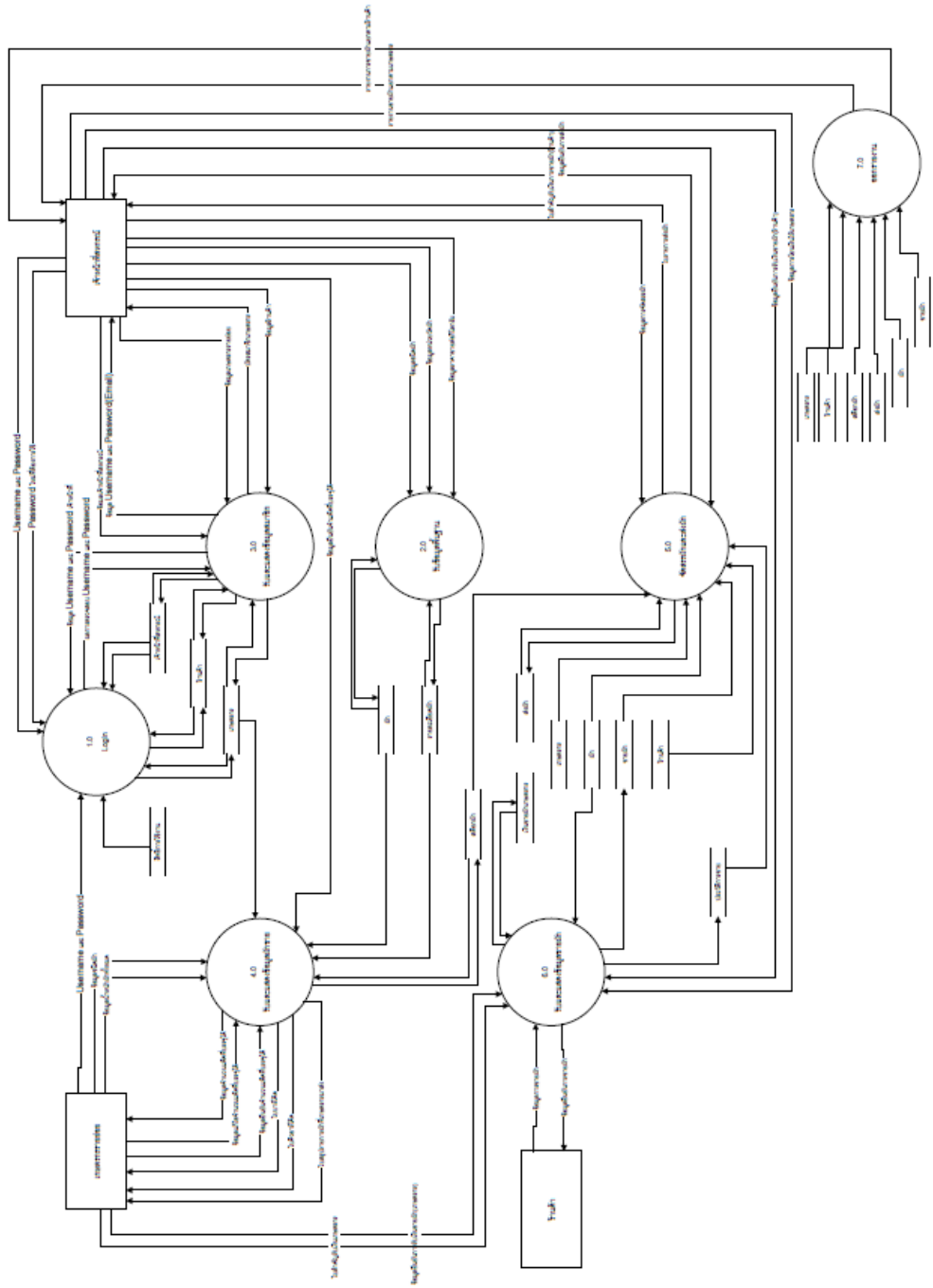


ซอฟต์แวร์สำหรับจัดจำหน่ายผลิตผลจากเกษตรกรรายย่อย  
 รายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

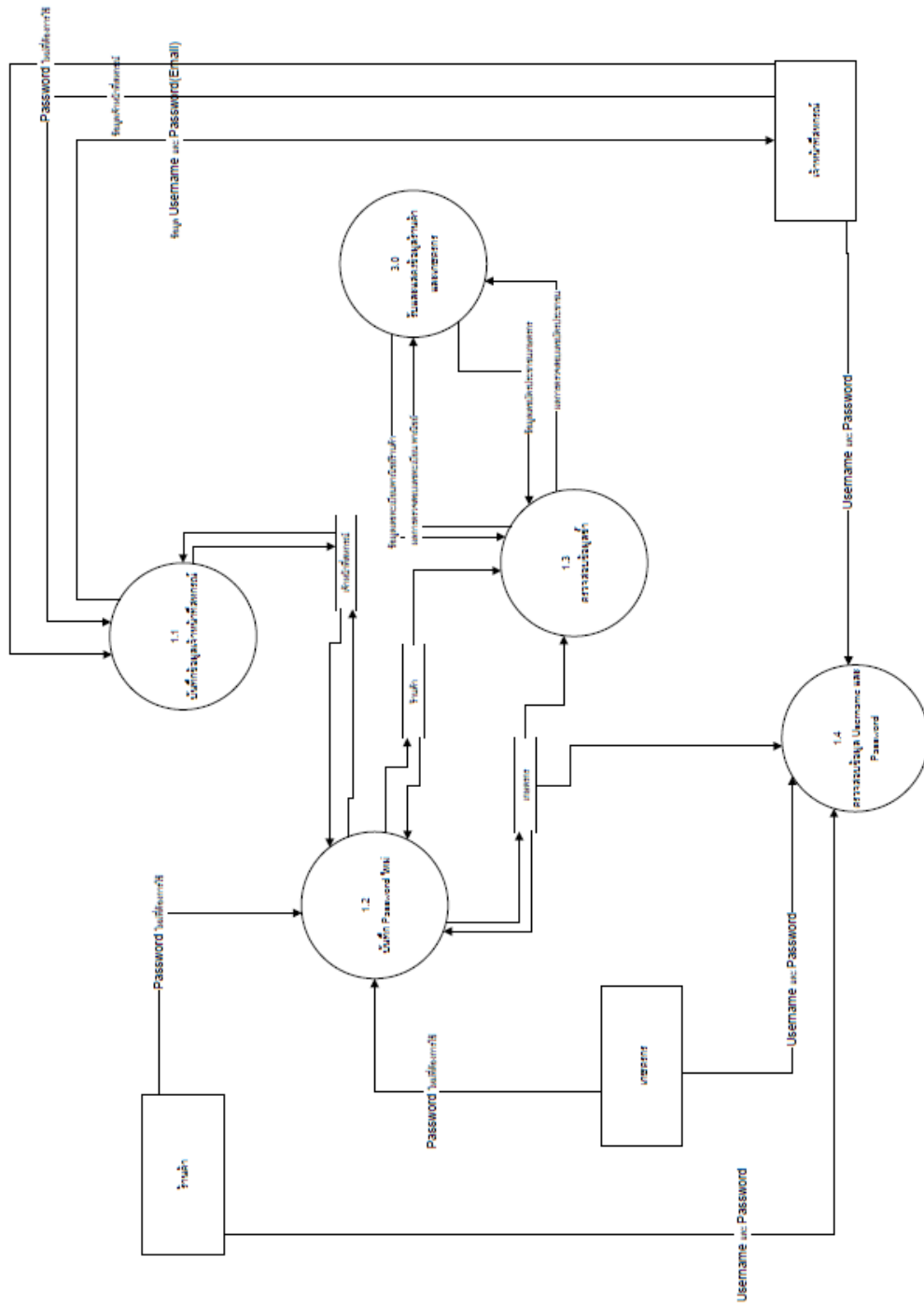
แผนภาพบริบท : ซอฟต์แวร์สำหรับจัดจำหน่ายผลิตผลจากเกษตรกรรายย่อย



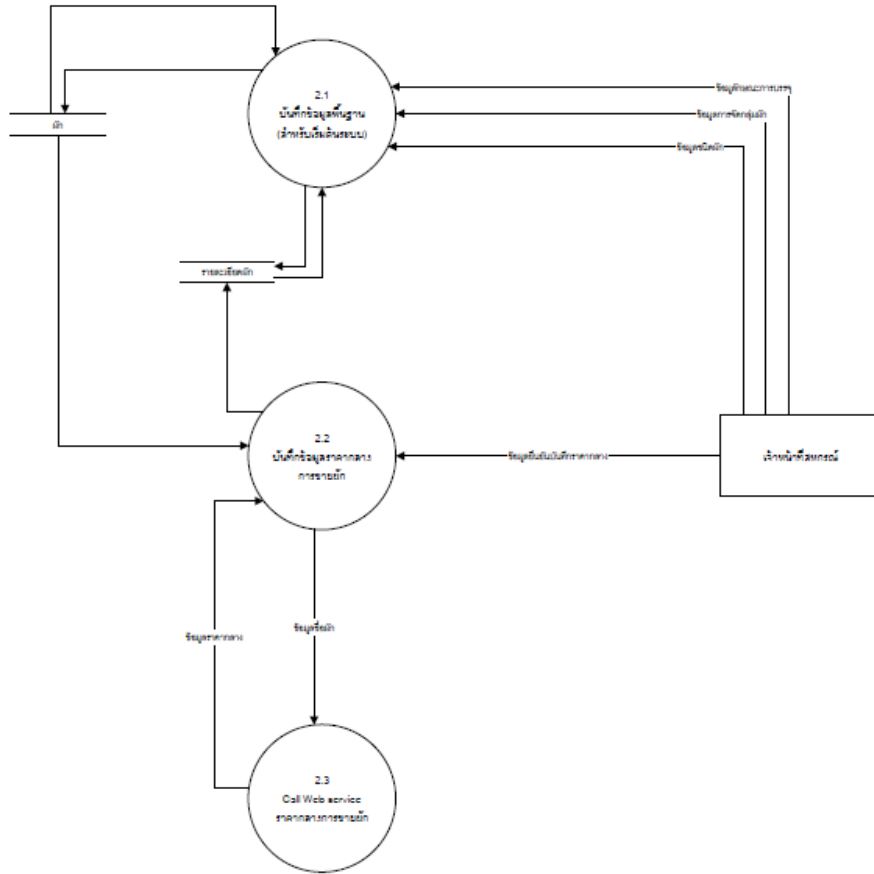
แผนภาพระดับที่ 1 : ซอฟต์แวร์สำหรับจัดจำหน่ายผลิตภัณฑ์จากเกษตรกรรายย่อย



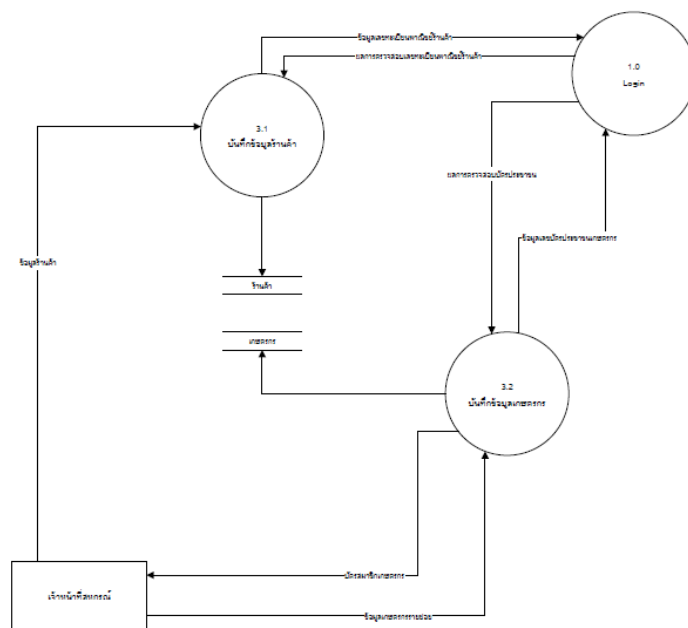
แผนภาพระดับที่ 2 : ซอฟต์แวร์สำหรับจัดจำหน่ายผลิตผลจากเกษตรกรรายย่อย  
 แผนภาพระบบย่อยของกระบวนการที่ 1 Login



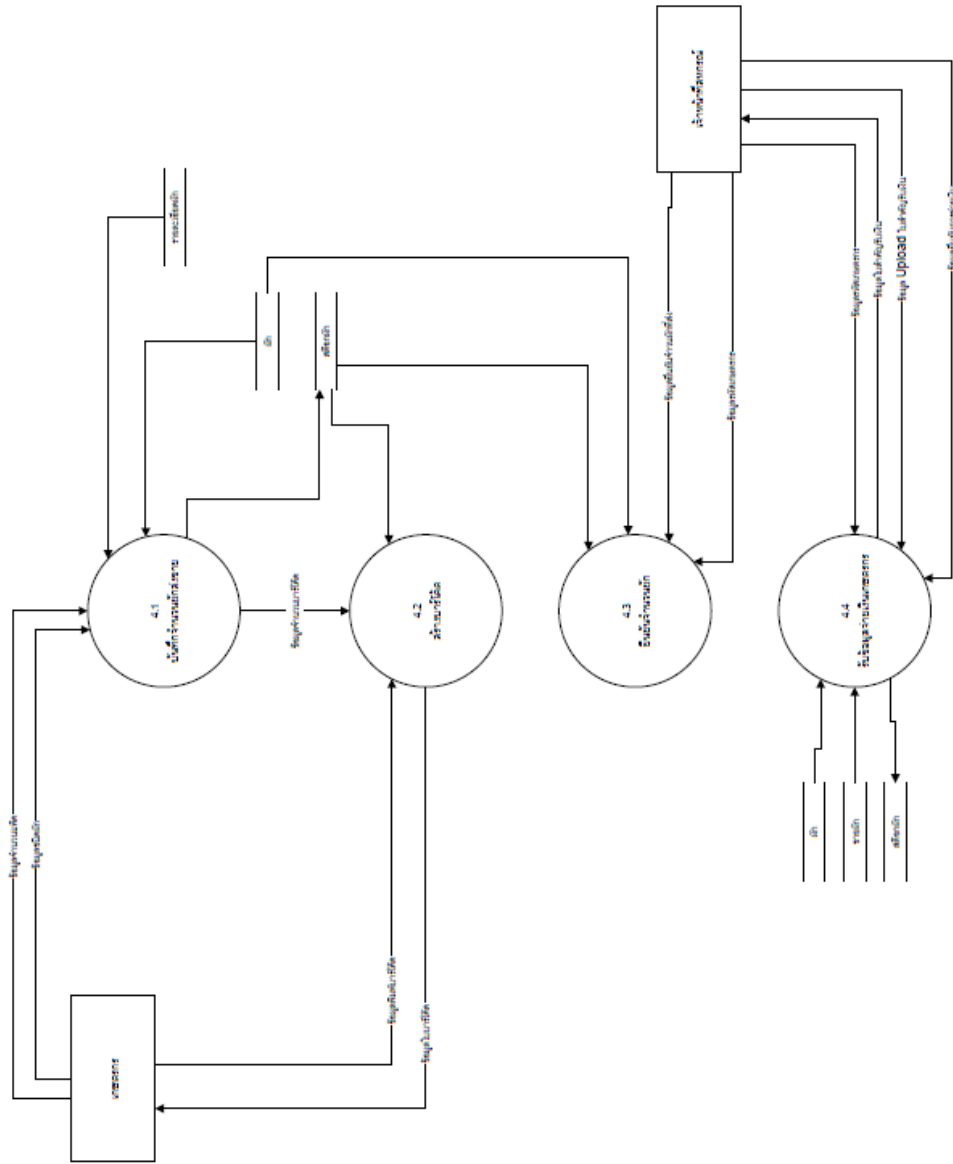
แผนภาพระบบย่อยของกระบวนการที่ 2 รับข้อมูลพื้นฐาน



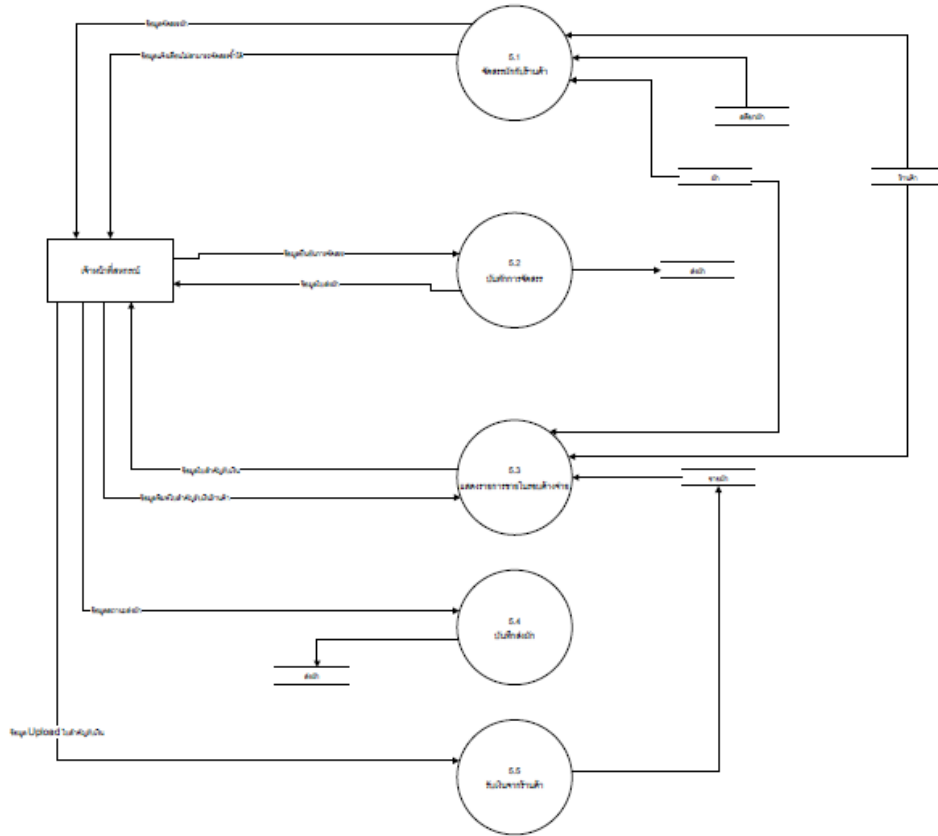
แผนภาพระบบย่อยของกระบวนการที่ 3 รับและแสดงข้อมูลสมาชิก



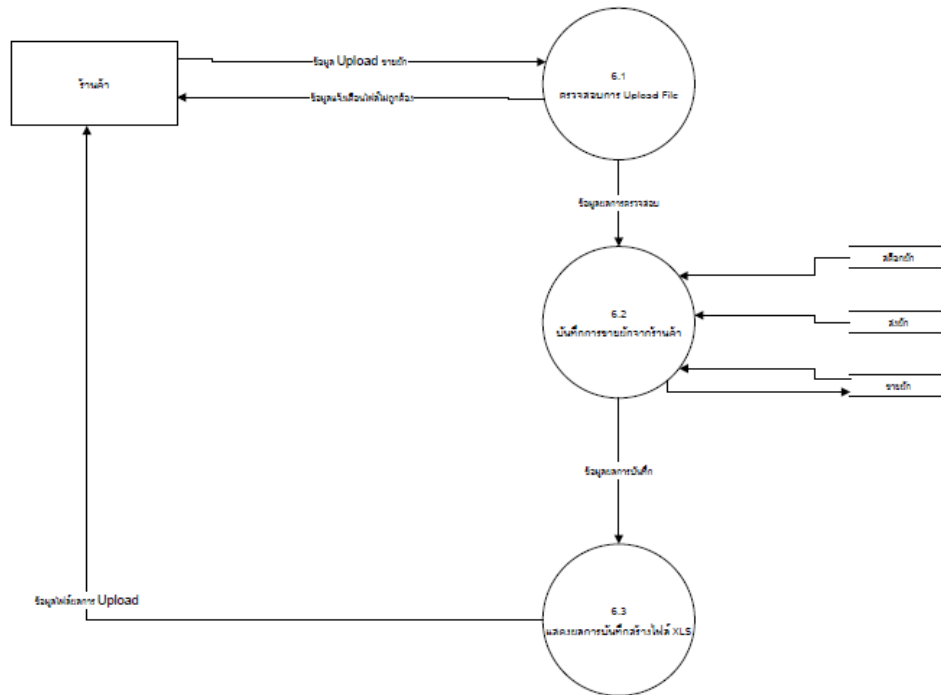
แผนภาพระบบย่อยของกระบวนการที่ 4 รับและแสดงข้อมูลฝากขาย



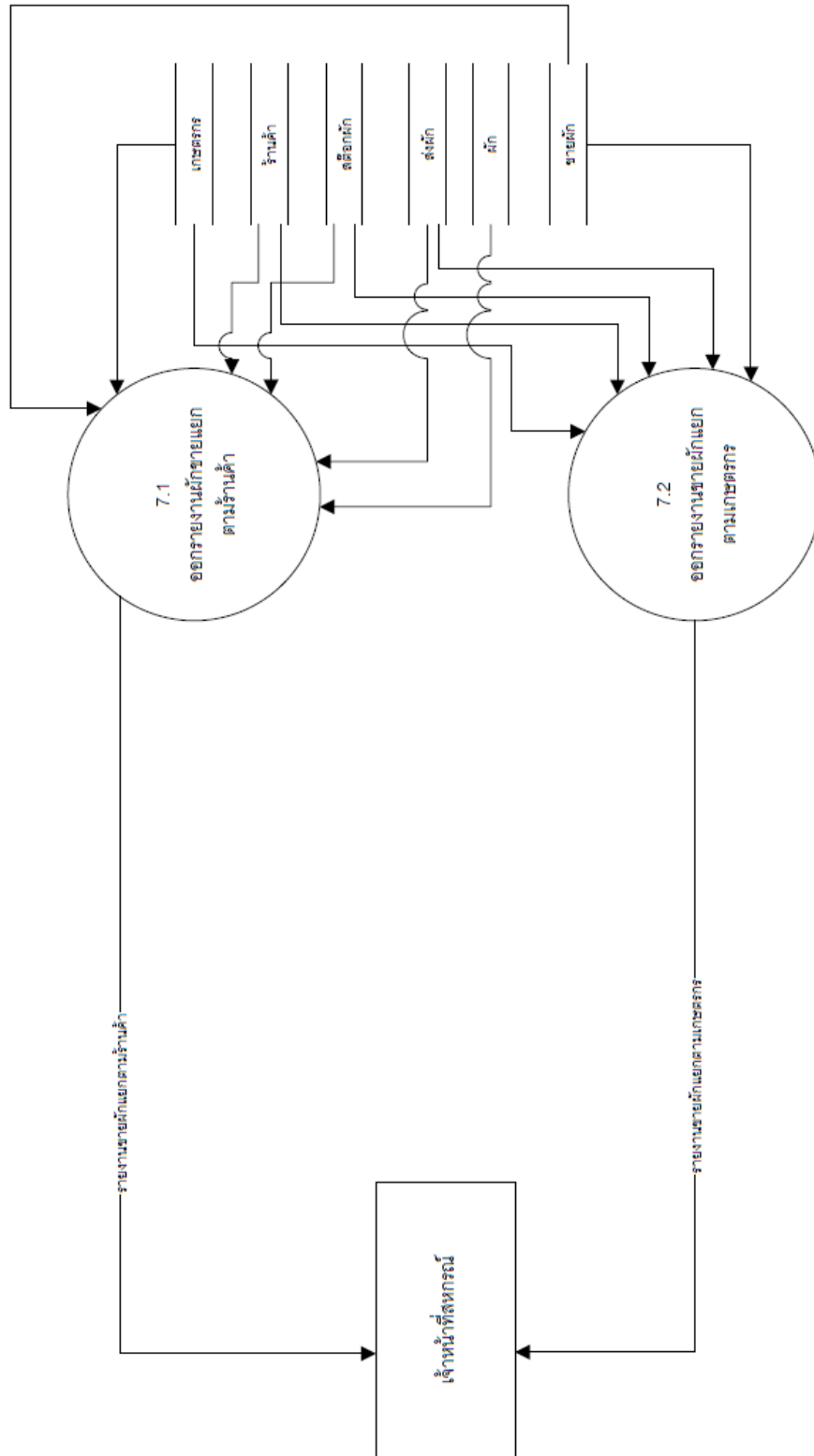
แผนภาพระบบย่อยของกระบวนการที่ 5 จัดสรรผักและส่งผัก



แผนภาพระบบย่อยของกระบวนการที่ 6 รับและแสดงข้อมูลขายผัก



แผนภาพระบบย่อยของกระบวนการที่ 7 ออกรายงาน





พจนานุกรมข้อมูล : ซอฟต์แวร์สำหรับจัดจำหน่ายผลิตผลจากเกษตรกรรายย่อย

รายละเอียดการออกแบบฐานข้อมูล

Table: VEGETABLE (ผัก)

ตารางที่ 0-12แสดงรายละเอียดตาราง VEGETABLE

No.	Column Name	Data Type	PK	FK	Description
1	VETGET_ID	INT	Y		รหัสผัก
2	VETGET_NAME	VARCHAR			ชื่อผัก

Table: MEASUREMENT (หน่วยวัดผัก)

ตารางที่ 0-13 แสดงรายละเอียดตาราง MEASUREMENT

No.	Column Name	Data Type	PK	FK	Description
1	MEASUR_ID	INT	Y		รหัสหน่วยวัด
2	SIZE	INT			ขนาด
3	weight	INT			น้ำหนัก

Table: VETGETABLE\_MEASUREMENT\_DETAIL (รายละเอียดหน่วยวัดผัก)

ตารางที่ 0-14แสดงรายละเอียดตาราง VETGETABLE\_MEASUREMENT\_DETAIL

No.	Column Name	Data Type	PK	FK	Description
1	VET_MEASUR_ID	INT	Y		รหัส รายละเอียด หน่วยวัด
2	VETGET_ID	INT		Y	รหัสผัก
3	MEASUR_ID	INT		Y	รหัสหน่วยวัด
4	PRICE	FLOAT			ราคา

Table: STAFF (เจ้าหน้าที่สหกรณ์)

ตารางที่ 0-15แสดงรายละเอียดตาราง STAFF

No.	Column Name	Data Type	PK	FK	Description
1	STAFF_ID	INT	Y		รหัสเจ้าหน้าที่สหกรณ์
2	STAFF_NAME	VARCHAR			ชื่อเจ้าหน้าที่สหกรณ์
3	CITIZEN_ID	INT			เลขที่บัตรประชาชน
4	STAFF_TEL	INT			เบอร์โทรศัพท์เจ้าหน้าที่สหกรณ์
5	USERNAME	VARCHAR			ชื่อที่อยู่ในระบบ
6	PASSWORD	VARCHAR			รหัสผ่าน
7	EMAIL	VARCHAR			อีเมล

Table: FARMER (เกษตรกรรายย่อย)

ตารางที่ 0-16แสดงรายละเอียดตาราง FARMER

No.	Column Name	Data Type	PK	FK	Description
1	FARMER_ID	INT	Y		รหัสสมาชิกเกษตรกร
2	FARMER_NAME	VARCHAR			ชื่อเกษตรกร
3	FARMER_ADDRESS	VARCHAR			ที่อยู่เกษตรกร
4	FARM_NAME	VARCHAR			ชื่อสวน

No.	Column Name	Data Type	PK	FK	Description
5	CITIZEN_ID	INT			เลขที่บัตรประชาชน
6	FARMER_TEL	VARCHAR			เบอร์โทรศัพท์
7	PASSWORD	VARCHAR			รหัสผ่าน

Table: SHOP (ร้านค้า)

ตารางที่ 0-17แสดงรายละเอียดตาราง SHOP

No.	Column Name	Data Type	PK	FK	Description
1	SHOP_ID	INT	Y		รหัสร้านค้า
2	SHOP_NAME	VARCHAR			ชื่อร้านค้า
3	COMMERCIAL_ID	INT			เลขทะเบียนพาณิชย์
4	SHOP_ADDRESS	VARCHAR			ที่อยู่ร้านค้า
5	CONTACT_NAME	VARCHAR			ชื่อผู้ติดต่อ
6	SHOP_TEL	INT			เบอร์โทรศัพท์
7	LAT_LON	TEXT			พิกัดร้านค้า
8	SHOP_EMAIL	VARCHAR			อีเมล

Table: STOCK\_VEGETABLE (สต็อกผัก)

ตารางที่ 0-18แสดงรายละเอียดตาราง STOCK\_VEGETABLE

No.	Column Name	Data Type	PK	FK	Description
1	STOCK_ID	INT	Y		รหัสสต็อกผัก
2	VETGET_ID	INT		Y	รหัสผัก
3	FARMER_ID	INT		Y	รหัสสมาชิก

No.	Column Name	Data Type	PK	FK	Description
					เกษตรกร
4	STOCK_CODE	INT			รหัสรอบการส่ง
5	DATE_TIME	DATE			วันเดือนปี
6	QUANTITY	INT			จำนวนผัก (แพ็ค)
7	PRICE	FLOAT			ราคา

Table: TRANSPORT\_VEGETABLE (ขนส่งผัก)

ตารางที่ 0-19แสดงรายละเอียดตาราง TRANSPORT\_VEGETABLE

No.	Column Name	Data Type	PK	FK	Description
1	TRANSPORT_ID	INT	Y		รหัสขนส่ง
2	STOCK_ID	INT		Y	รหัสสต็อกผัก
3	SHOP_ID	INT		Y	รหัสร้านค้า
4	STAFF_ID	INT		Y	รหัสเจ้าหน้าที่ สหกรณ์
5	STATUS_ID	INT		Y	รหัส สถานะการส่ง ผัก
6	DATE_TIME	DATE			วันเดือนปี ขนส่งผัก
7	AMOUNT	INT			จำนวนผักที่ส่ง

Table: SELL\_VEGETABLE (ขายผัก)

ตารางที่ 0-20แสดงรายละเอียดตาราง SELL\_VEGETABLE

No.	Column Name	Data Type	PK	FK	Description
1	SELL_ID	INT	Y		รหัสขายผัก

No.	Column Name	Data Type	PK	FK	Description
2	STOCK_ID	INT		Y	รหัสสต็อก
3	SHOP_ID	INT		Y	รหัสร้านค้า
4	TRANSPORT_ID	INT		Y	รหัสขนส่ง
5	DATE_TIME	DATE			วันเดือนปีที่ขาย
6	PRICE	FLOAT			ราคาขาย
7	AMOUNT	INT			จำนวนผักขาย

Table: STATUS (สถานะ)

ตารางที่ 0-21แสดงรายละเอียดตาราง STATUS

No.	Column Name	Data Type	PK	FK	Description
1	STATUS_ID	INT	Y		รหัสสถานะ
2	STATUS_NAME	VARCHAR			ชื่อสถานะ

Table: PAYMENT\_FARMER (เงินขายผักเกษตรกร)

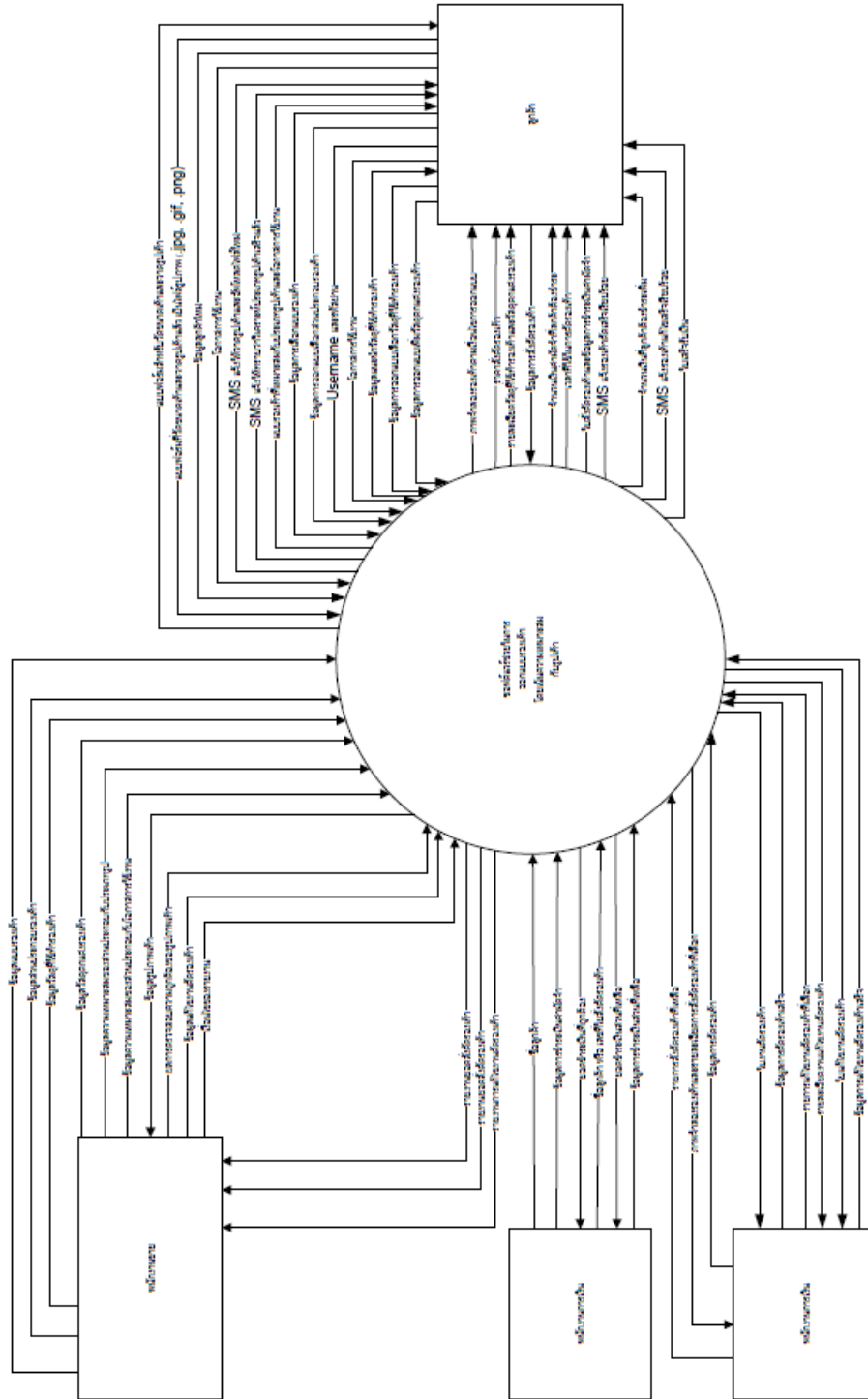
ตารางที่ 0-22แสดงรายละเอียดตาราง FABRIC

No.	Column Name	Data Type	PK	FK	Description
1	PAYMENT_ID	INT	Y		รหัสการจ่ายเงิน
2	FARMER_ID	INT		Y	รหัสเกษตรกร
3	STAFF_ID	INT		Y	รหัสเจ้าหน้าที่สหกรณ์
4	STOCK_ID	INT		Y	รหัสสต็อก
5	STATUS_ID	INT		Y	รหัสสถานะ
6	AMOUNT	INT			ยอดเงิน

No.	Column Name	Data Type	PK	FK	Description
7	DATE_TIME	DATE			วันเดือนปี
8	IMG_FILE	VARCHAR			รูปใบสำคัญ รับเงิน



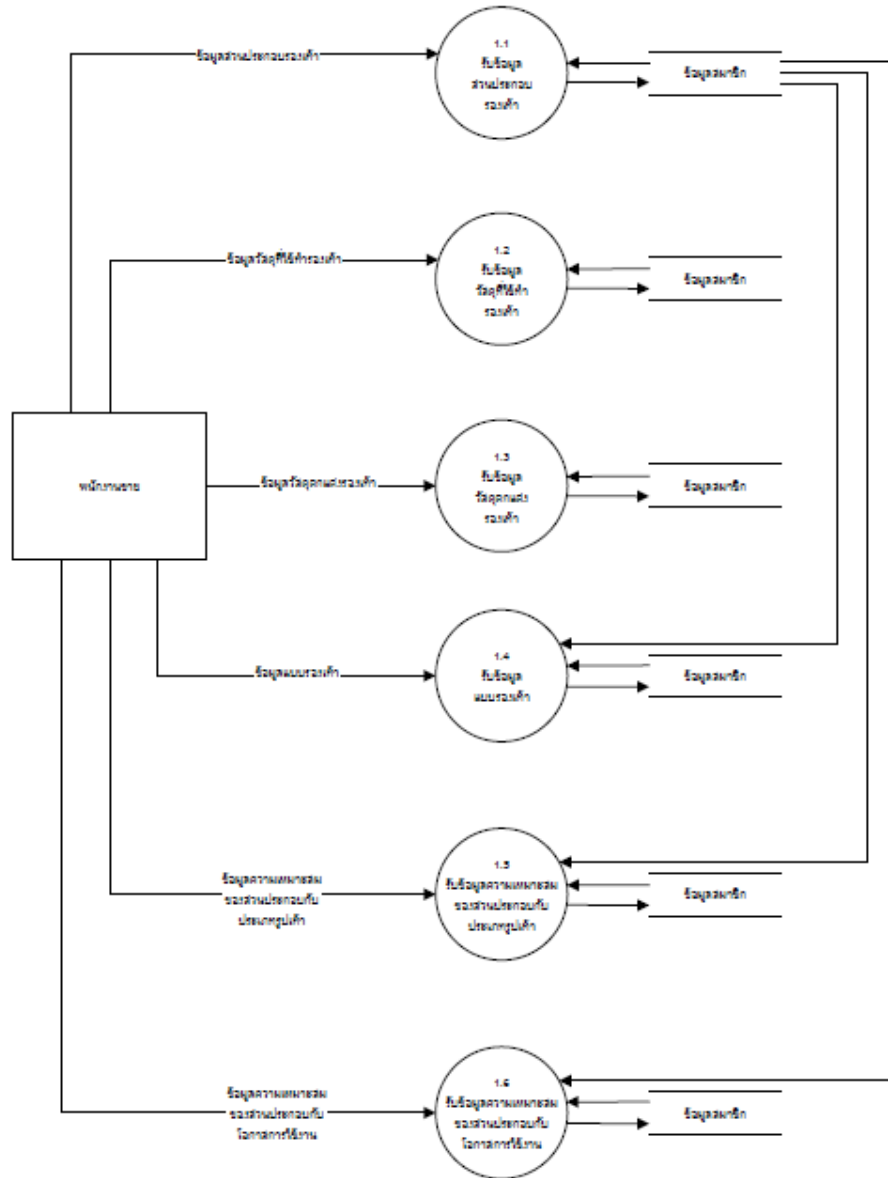
ซอฟต์แวร์ช่วยในการออกแบบรองเท้า  
รายละเอียดซอฟต์แวร์ที่ออกแบบไว้  
แผนภาพบริบท : ซอฟต์แวร์ช่วยในการออกแบบรองเท้า



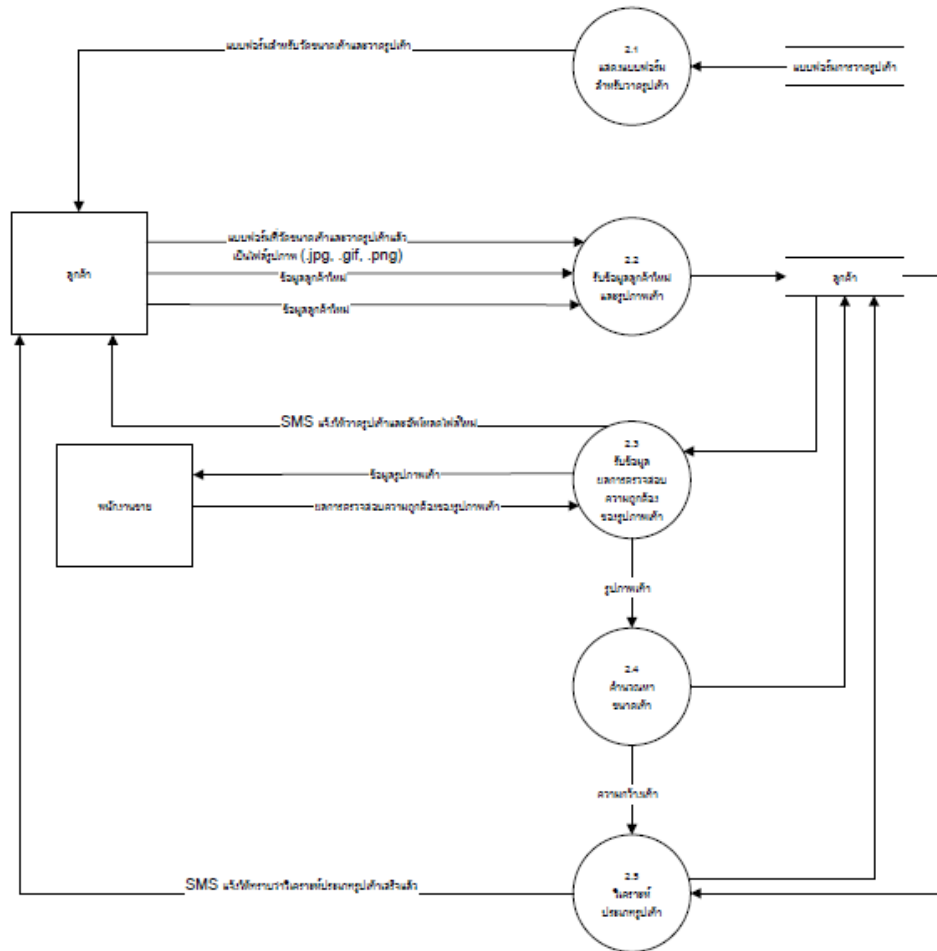




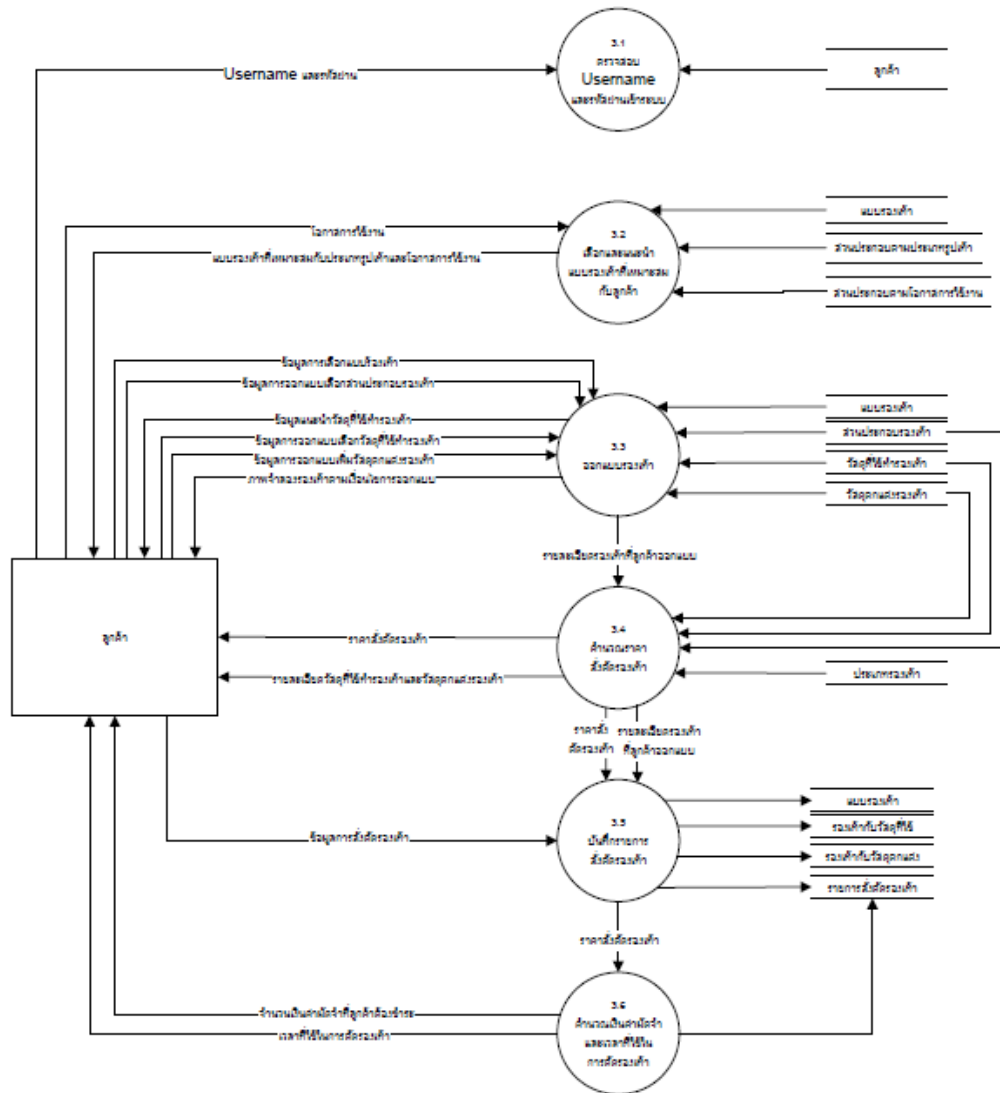
แผนภาพระดับที่ 2 : ซอฟต์แวร์ช่วยในการออกแบบรองเท้า  
 แผนภาพระบย่อยของกระบวนการที่ 1 รับข้อมูลหลัก



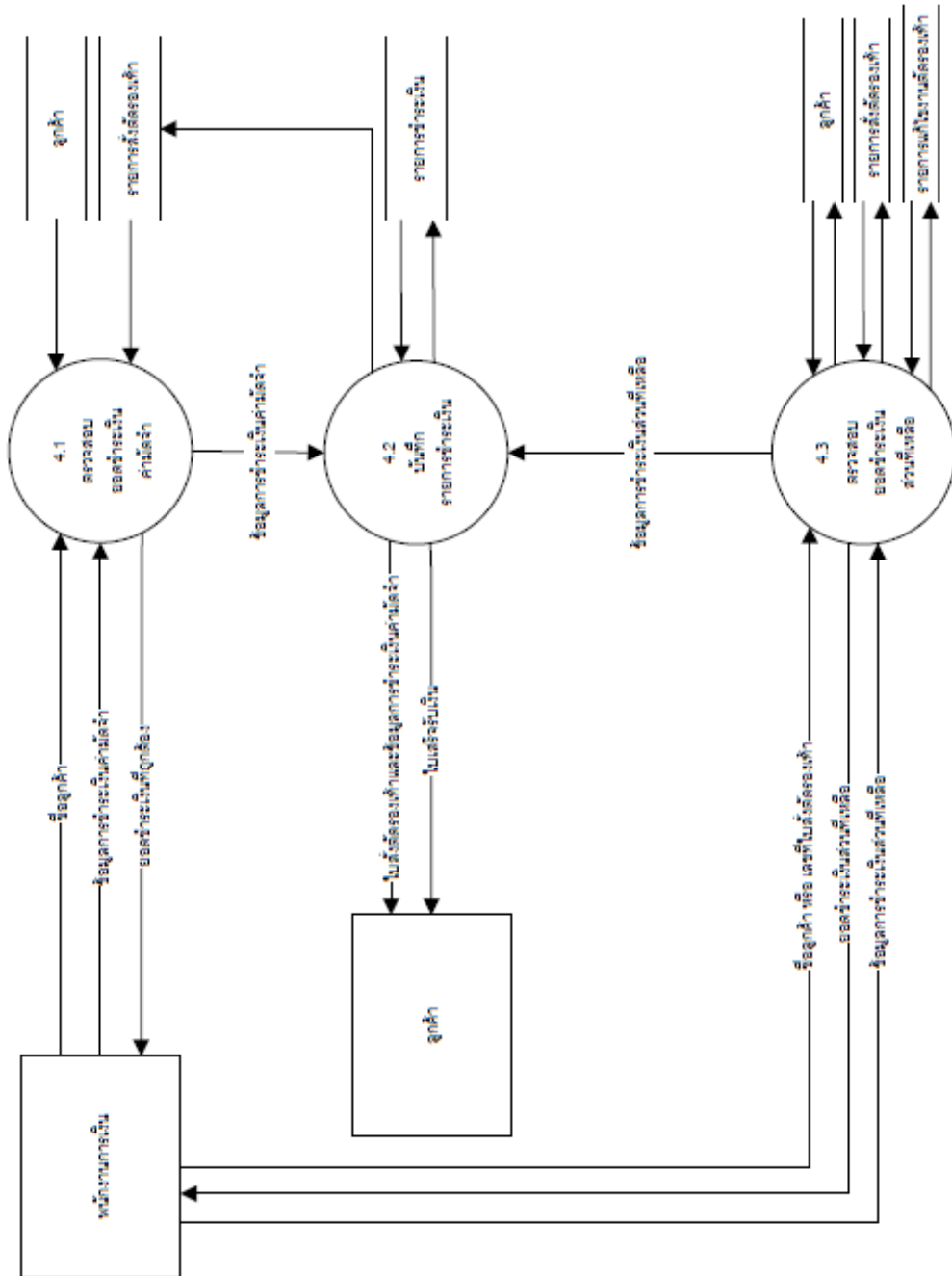
แผนภาพระบบย่อยของกระบวนการที่ 2 วิเคราะห์ประเภทรูปเท้าของลูกค้า



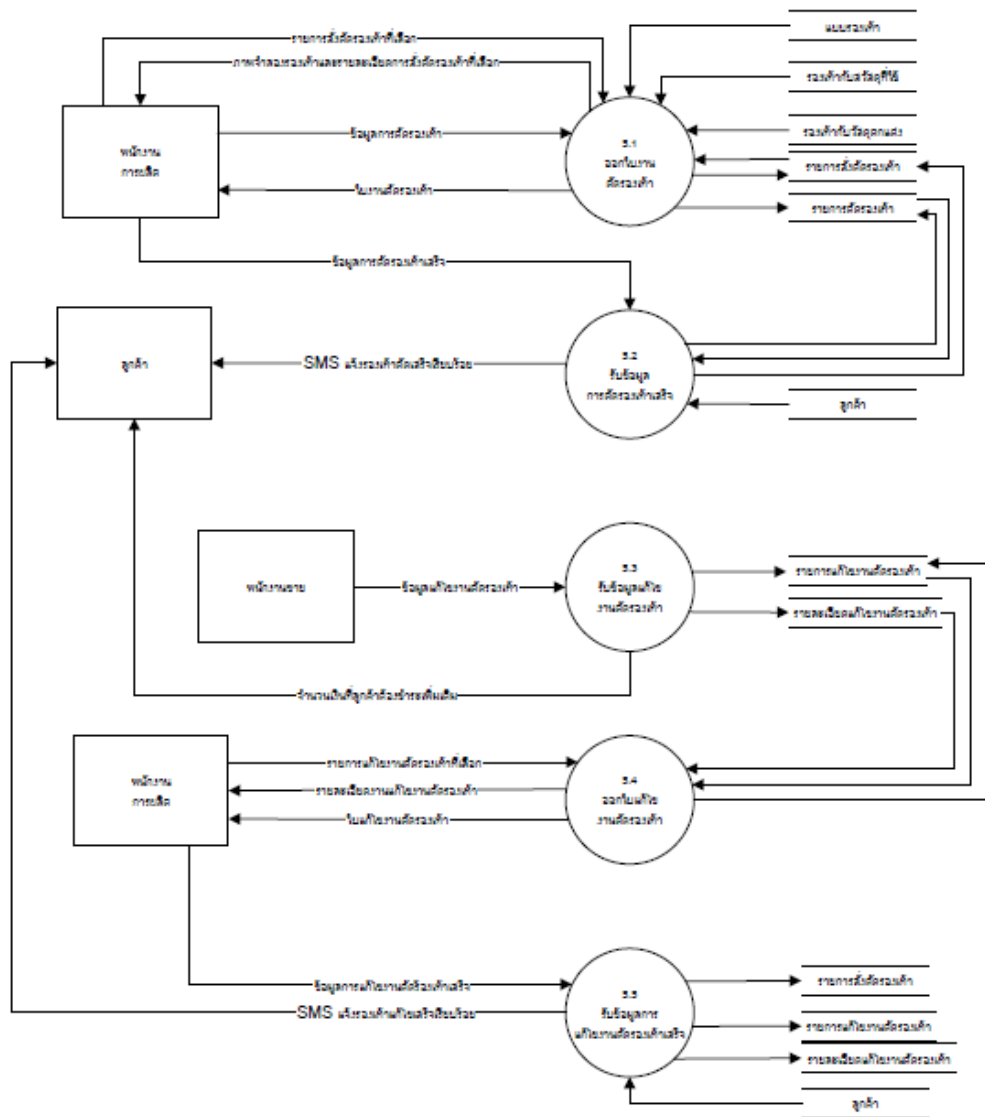
แผนภาพระบบย่อยของกระบวนการที่ 3 ออกแบบและสั่งตัดรองเท้า



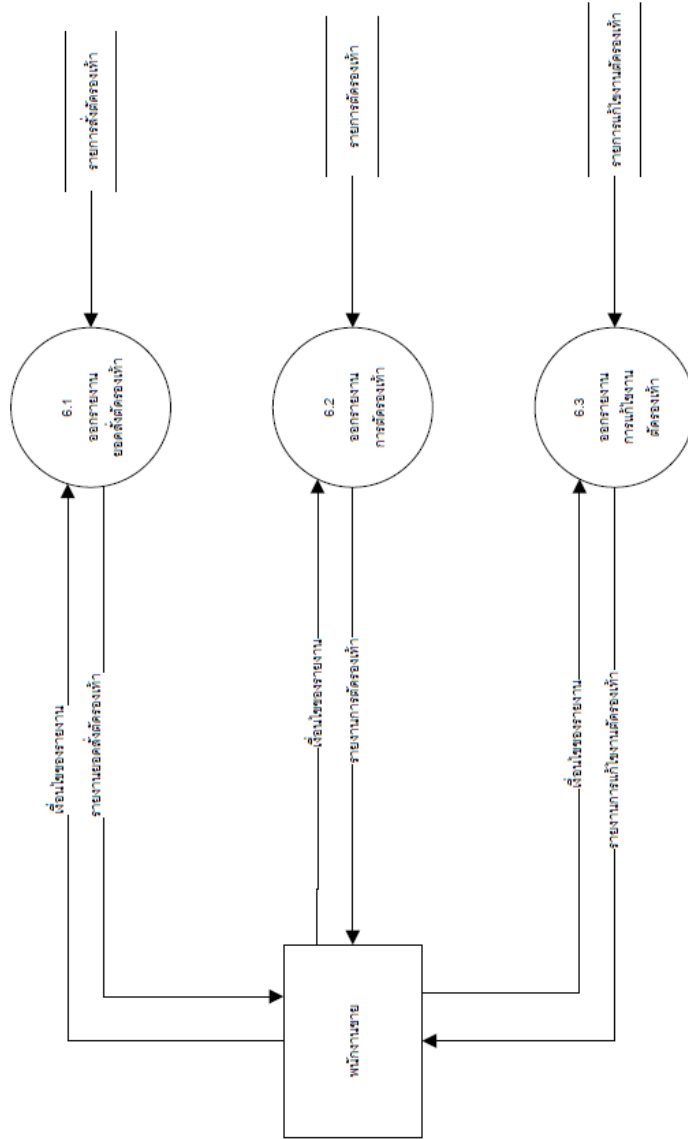
แผนภาพระบบย่อยของกระบวนการที่ 4 การรับเงิน



แผนภาพระบบย่อยของกระบวนการที่ 5 งานตัดรองเท้า



แผนภาพระบบย่อยของกระบวนการที่ 6 ออกรายงาน



พจนานุกรมข้อมูล : ซอฟต์แวร์ช่วยในการออกแบบรองเท้า

4.9.2 รายละเอียดการออกแบบฐานข้อมูล (Data Dictionary)

4.9.2.1 Table: Color (สี)

ตารางที่ 4.1 แสดงรายละเอียดตาราง Color

No.	Column Name	Data Type	PK	FK	Description
1	colored	int(11)	✓		รหัสสี
2	colorName	varchar(100)			ชื่อสี

4.9.2.2 Table: Customer (ลูกค้า)

ตารางที่ 4.2 แสดงรายละเอียดตาราง Customer

No.	Column Name	Data Type	PK	FK	Description
1	customerID	int(11)	✓		รหัสลูกค้า
2	customerName	varchar(100)			ชื่อลูกค้า
3	customerTel	varchar(10)			เบอร์โทรศัพท์ติดต่อลูกค้า
4	customerEmail	varchar(50)			อีเมลลูกค้า
5	username	varchar(50)			ชื่อผู้ใช้งานระบบ

4.9.2.3 Table: CustomerInfo (รายละเอียดลูกค้า)

ตารางที่ 4.3 แสดงรายละเอียดตาราง CustomerInfo

No.	Column Name	Data Type	PK	FK	Description
1	customerID	int(11)	✓	✓	รหัสลูกค้า
2	footSide	varchar(1)			เท้าข้างซ้าย/ข้างขวา
3	footImage	longblob			รูปภาพเท้า
4	width	float			ความกว้างเท้า (ซม.)
5	height	float			ความยาวเท้า (ซม.)
6	perimeter	float			ความยาวเส้นรอบวงรอบเท้า (ซม.)
7	toeType	varchar(10)			ลักษณะนิ้วเท้า
8	footThick	varchar(10)			ลักษณะเนื้อเท้า

4.9.2.4 Table: DecorateImage (รูปภาพวัสดุตกแต่งรองเท้า)

ตารางที่ 4.4 แสดงรายละเอียดตาราง DecorateImage

No.	Column Name	Data Type	PK	FK	Description
1	decorateID	int(11)	✓	✓	รหัสวัสดุตกแต่งรองเท้า
2	shotID	int(11)	✓		เลขที่ภาพ
3	decorateImage	longblob			รูปภาพวัสดุตกแต่งรองเท้า

4.9.2.5 Table: Decoration (วัสดุตกแต่งรองเท้า)

ตารางที่ 4.5 แสดงรายละเอียดตาราง Decoration

No.	Column Name	Data Type	PK	FK	Description
1	decorateID	int(11)	✓		รหัสวัสดุตกแต่งรองเท้า
2	decorateName	varchar(100)			ชื่อวัสดุตกแต่งรองเท้า
3	partID	int(11)		✓	รหัสส่วนประกอบรองเท้า
4	decoratePrice	float			ราคาวัสดุตกแต่งรองเท้า
5	decorateTypeID	int(11)		✓	รหัสประเภทวัสดุตกแต่งรองเท้า
6	colorID	int(11)		✓	รหัสสี

4.9.2.6 Table: DecorationType (ประเภทวัสดุตกแต่งรองเท้า)

ตารางที่ 4.6 แสดงรายละเอียดตาราง DecorationType

No.	Column Name	Data Type	PK	FK	Description
1	decorateTypeID	int(11)	✓		รหัสประเภทวัสดุตกแต่งรองเท้า
2	decorateTypeName	varchar(100)			ชื่อประเภทวัสดุตกแต่งรองเท้า

4.9.2.7 Table: FixingDetail (รายละเอียดแก้ไขงานตัดรองเท้า)

ตารางที่ 4.7 แสดงรายละเอียดตาราง FixingDetail

No.	Column Name	Data Type	PK	FK	Description
1	fixID	int(11)	✓		รหัสแก้ไขงานตัดรองเท้า
2	lineID	int(11)	✓		เลขที่รายการแก้ไข
3	partID	int(11)		✓	รหัสประเภทส่วนประกอบรองเท้า
4	fixDetail	varchar(255)			รายละเอียดการแก้ไขงานตัดรองเท้า
5	fixPrice	float			ราคาแก้ไขงานตัด
6	errorBy	varchar(50)			สาเหตุมาจาก

4.9.2.8 Table: Material (วัสดุที่ใช้ทำรองเท้า)

ตารางที่ 4.8 แสดงรายละเอียดตาราง Material

No.	Column Name	Data Type	PK	FK	Description
1	materialID	int(11)	✓		รหัสวัสดุที่ใช้ทำรองเท้า
2	materialName	varchar(100)			ชื่อวัสดุที่ใช้ทำรองเท้า
3	partID	int(11)		✓	รหัสประเภทส่วนประกอบรองเท้า
4	materialImage	longblob			รูปภาพวัสดุที่ใช้ทำรองเท้า
5	materialPrice	float			ราคาต่อหน่วยของวัสดุที่ใช้ทำรองเท้า
6	materialType	int(11)		✓	รหัสประเภทวัสดุที่ใช้ทำรองเท้า
7	colorID	int(11)		✓	รหัสสี



4.9.2.9 Table: MaterialType (ประเภทวัสดุที่ใช้ทำรองเท้า)

ตารางที่ 4.9 แสดงรายละเอียดตาราง MaterialType

No.	Column Name	Data Type	PK	FK	Description
1	materialTypeID	int(11)	✓		รหัสประเภทวัสดุที่ใช้ทำรองเท้า
2	materialTypeName	varchar(100)			ชื่อประเภทวัสดุที่ใช้ทำรองเท้า

4.9.2.10 Table: MaterialWithOccasion (วัสดุที่ใช้ทำรองเท้าตามโอกาสการใช้งาน)

ตารางที่ 4.10 แสดงรายละเอียดตาราง MaterialWithOccasion

No.	Column Name	Data Type	PK	FK	Description
1	materialID	int(11)	✓	✓	รหัสวัสดุที่ใช้ทำรองเท้า
2	occasionID	int(11)	✓	✓	รหัสโอกาสการใช้งาน
3	score	int(11)			คะแนนความเหมาะสมระหว่างวัสดุที่ใช้ทำรองเท้ากับโอกาสการใช้งาน

4.9.2.11 Table: Occasion (โอกาสการใช้งาน)

ตารางที่ 4.11 แสดงรายละเอียดตาราง Occasion

No.	Column Name	Data Type	PK	FK	Description
1	occasionID	int(11)	✓		รหัสโอกาสการใช้งาน
2	occasionName	varchar(100)			ชื่อโอกาสการใช้งาน

4.9.2.12 Table: OrderShoe (รายการสั่งซื้อรองเท้า)

ตารางที่ 4.12 แสดงรายละเอียดตาราง OrderShoe

No.	Column Name	Data Type	PK	FK	Description
1	orderID	int(11)	✓		รหัสรายการสั่งซื้อรองเท้า
2	orderDate	date			วันที่สั่งซื้อรองเท้า
3	dueDate	date			วันที่กำหนดรองเท้าเสร็จ
4	customerID	int(11)		✓	รหัสลูกค้า
5	shoeID	int(11)		✓	รหัสแบบรองเท้าที่สั่งซื้อ
6	totalPrice	float			ราคาค่าสั่งซื้อรองเท้าทั้งหมด
7	deposit	float			จำนวนเงินค้ำมัดจำ
8	balance	float			ยอดคงเหลือค้างชำระ
9	statusID	int(11)		✓	รหัสสถานะ

4.9.2.13 Table: Part (ส่วนประกอบรองเท้า)

ตารางที่ 4.13 แสดงรายละเอียดตาราง Part

No.	Column Name	Data Type	PK	FK	Description
1	partID	int(11)	✓		รหัสส่วนประกอบรองเท้า
2	partName	varchar(100)			ชื่อส่วนประกอบรองเท้า
3	partTypeID	int(11)		✓	รหัสประเภทส่วนประกอบรองเท้า
4	additionalPrice	float			ราคาส่วนประกอบรองเท้าพิเศษ
5	materialUse	float			จำนวนวัสดุที่ใช้ทำส่วนประกอบ

4.9.2.14 Table: PartImage (รูปภาพส่วนประกอบรองเท้า)

ตารางที่ 4.14 แสดงรายละเอียดตาราง PartImage

No.	Column Name	Data Type	PK	FK	Description
1	partID	int(11)	✓	✓	รหัสส่วนประกอบรองเท้า
2	shotID	int(11)	✓		เลขที่ภาพ
3	partImage	longblob			รูปภาพส่วนประกอบรองเท้า

4.9.2.15 Table: PartType (ประเภทส่วนประกอบรองเท้า)

ตารางที่ 4.15 แสดงรายละเอียดตาราง PartType

No.	Column Name	Data Type	PK	FK	Description
1	partTypeID	int(11)	✓		รหัสประเภทส่วนประกอบรองเท้า
2	partTypeName	varchar(100)			ชื่อประเภทส่วนประกอบรองเท้า

4.9.2.16 Table: PartWithFootShape (ส่วนประกอบรองเท้าตามประเภทรูปเท้า)

ตารางที่ 4.16 แสดงรายละเอียดตาราง PartWithFootShape

No.	Column Name	Data Type	PK	FK	Description
1	partID	int(11)	✓	✓	รหัสส่วนประกอบรองเท้า
2	Egyptian	tinyint(1)			เหมาะกับลักษณะนิ้วเท้าแบบ Egyptian
3	Greek	tinyint(1)			เหมาะกับลักษณะนิ้วเท้าแบบ Greek
4	Square	tinyint(1)			เหมาะกับลักษณะนิ้วเท้าแบบ Square
5	Flat	tinyint(1)			เหมาะกับลักษณะเนื้อเท้าแบนราบ
6	Thick	tinyint(1)			เหมาะกับลักษณะเนื้อเท้า อวบอูม

4.9.2.17 Table: PartWithOccasion (ส่วนประกอบรองเท้าตามโอกาสการใช้งาน)

ตารางที่ 4.17 แสดงรายละเอียดตาราง PartWithOccasion

No.	Column Name	Data Type	PK	FK	Description
1	partID	int(11)	✓	✓	รหัสส่วนประกอบรองเท้า
2	occasionID	int(11)	✓	✓	รหัสโอกาสการใช้งาน
3	score	int(11)			คะแนนความเหมาะสมระหว่างส่วนประกอบรองเท้ากับโอกาสการใช้งาน

4.9.2.18 Table: Payment (รายการชำระเงิน)

ตารางที่ 4.18 แสดงรายละเอียดตาราง Payment

No.	Column Name	Data Type	PK	FK	Description
1	paymentID	int(11)	✓		รหัสรายการชำระเงิน
2	paymentDate	date			วันที่ชำระเงิน
3	orderID	int(11)		✓	รหัสรายการสั่งซื้อรองเท้า
4	customerID	int(11)		✓	รหัสลูกค้า
5	paymentType	varchar(50)			ประเภทรายการชำระเงิน (ค่ามัดจำ/ ส่วนที่เหลือ)
6	paymentAmount	float			จำนวนเงินที่ชำระ
7	paymentDoc	longblob			หลักฐานการชำระเงิน

4.9.2.19 Table: Shoe (แบบรองเท้า)

ตารางที่ 4.19 แสดงรายละเอียดตาราง Shoe

No.	Column Name	Data Type	PK	FK	Description
1	shoeID	int(11)	✓		รหัสแบบรองเท้า
2	shoeName	varchar(100)			ชื่อแบบรองเท้า
3	shoeTypeID	int(11)		✓	รหัสประเภทรองเท้า
4	shoeImage	longblob			รูปภาพรองเท้า
5	toeBox	int(11)		✓	รหัสส่วนประกอบรองเท้า (ส่วนหัวนิ้วเท้า)
6	back	int(11)		✓	รหัสส่วนประกอบรองเท้า (ส่วนหุ้มส้นเท้า)
7	heel	int(11)		✓	รหัสส่วนประกอบรองเท้า (ส่วนส้นรองเท้า)
8	hasDecorate	tinyint(1)			มีวัสดุตกแต่ง
9	canShow	tinyint(1)			แสดงแบบรองเท้านี้ได้

4.9.2.20 Table: ShoeCutting (รายการสั่งซื้อตัดรองเท้า)

ตารางที่ 4.20 แสดงรายละเอียดตาราง ShoeCutting

No.	Column Name	Data Type	PK	FK	Description
1	cuttingID	int(11)	✓		รหัสรายการตัดรองเท้า
2	cuttingDate	date			วันที่ตัดรองเท้า
3	orderID	int(11)		✓	รหัสรายการสั่งซื้อตัดรองเท้า
4	shoeID	int(11)		✓	รหัสแบบรองเท้า
5	statusID	int(11)		✓	รหัสสถานะ
6	cuttingBy	varchar(50)			ตัดรองเท้าโดย
7	finishedDate	date			วันที่รองเท้าตัดเสร็จ

4.9.2.21 Table: ShoeFixing (รายการแก้ไขงานตัดรองเท้า)

ตารางที่ 4.21 แสดงรายละเอียดตาราง ShoeFixing

No.	Column Name	Data Type	PK	FK	Description
1	fixID	int(11)	✓		รหัสแก้ไขงานตัดรองเท้า
2	fixDate	date			วันที่แก้ไขงานตัดรองเท้า
3	orderID	int(11)		✓	รหัสรายการสั่งซื้อตัดรองเท้า
4	shoeID	int(11)		✓	รหัสแบบรองเท้า
5	totalFixPrice	float			ราคาแก้ไขงานตัดรองเท้า
6	statusID	int(11)		✓	รหัสสถานะ
7	fixBy	varchar(50)			แก้ไขงานตัดโดย
8	finishedDate	date			วันที่แก้ไขงานตัดเสร็จ

4.9.2.22 Table: ShoeType (ประเภทรองเท้า)

ตารางที่ 4.22 แสดงรายละเอียดตาราง ShoeType

No.	Column Name	Data Type	PK	FK	Description
1	shoeTypeID	int(11)	✓		รหัสประเภทรองเท้า
2	shoeTypeName	varchar(100)			ชื่อประเภทรองเท้า
3	shoeTypePrice	float			ราคาตามประเภทรองเท้า

4.9.2.23 Table: ShoeWithDecoration (รองเท้ากับวัสดุตกแต่ง)

ตารางที่ 4.23 แสดงรายละเอียดตาราง ShoeWithDecoration

No.	Column Name	Data Type	PK	FK	Description
1	shoeID	int(11)	✓	✓	รหัสแบบรองเท้า
2	partID	int(11)	✓	✓	รหัสส่วนประกอบรองเท้า
3	decorateID	int(11)		✓	รหัสวัสดุตกแต่งรองเท้า

## 4.9.2.24 Table: ShoeWithMaterial (รองเท้ากับวัสดุที่ใช้)

ตารางที่ 4.24 แสดงรายละเอียดตาราง ShoeWithMaterial

No.	Column Name	Data Type	PK	FK	Description
1	shoeID	int(11)	✓		รหัสแบบรองเท้า
2	tMaterial	int(11)		✓	รหัสวัสดุที่ใช้ทำ (ส่วนหุ้มนิ้วเท้า)
3	bMaterial	int(11)		✓	รหัสวัสดุที่ใช้ทำ (ส่วนหุ้มสันเท้า)
4	hMaterial	int(11)		✓	รหัสวัสดุที่ใช้ทำ (ส่วนสันรองเท้า)

## 4.9.2.25 Table: Status (สถานะ)

ตารางที่ 4.25 แสดงรายละเอียดตาราง Status

No.	Column Name	Data Type	PK	FK	Description
1	statusID	int(11)	✓		รหัสสถานะ
2	statusName	varchar(100)			ชื่อสถานะ

## 4.9.2.26 Table: UserLogin (ผู้ใช้ระบบ)

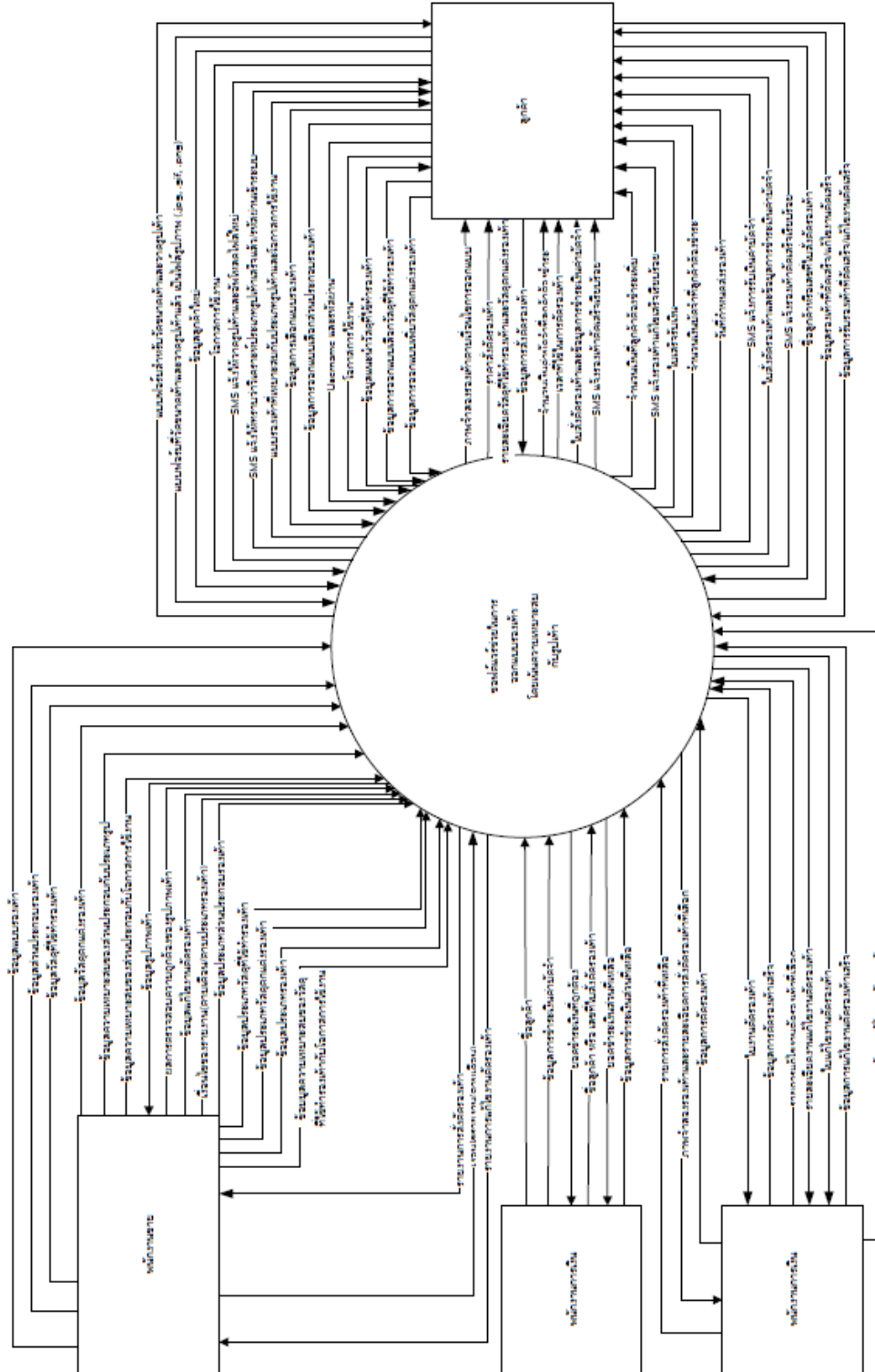
ตารางที่ 4.26 แสดงรายละเอียดตาราง UserLogin

No.	Column Name	Data Type	PK	FK	Description
1	username	varchar(10)	✓		ชื่อผู้ใช้งานระบบ
2	customerID	int(11)		✓	รหัสลูกค้า
3	password	varchar(10)			รหัสผ่านเข้าระบบ

### ซอฟต์แวร์ช่วยในการออกแบบรองเท้า

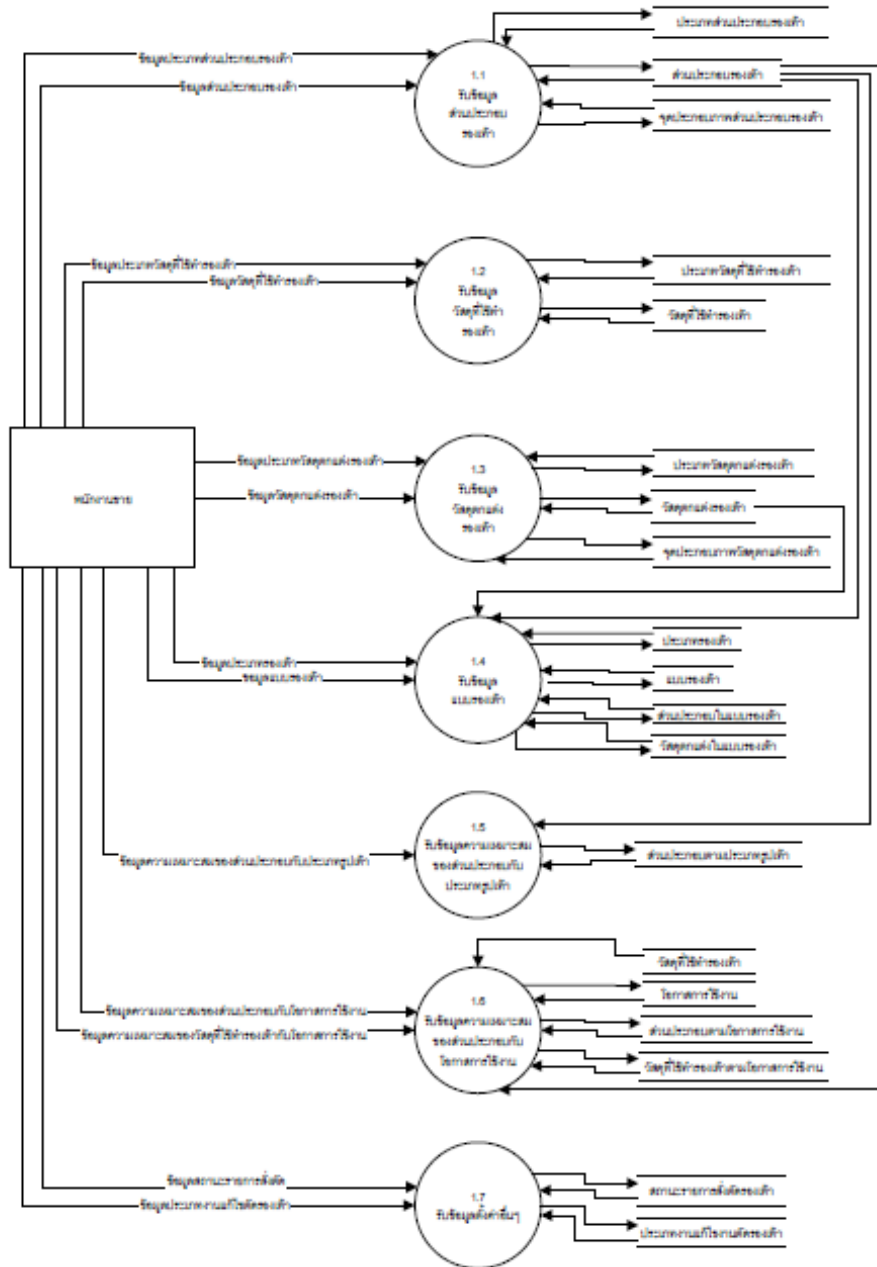
### รายละเอียดซอฟต์แวร์ที่สร้างเสร็จแล้ว

### แผนภาพบริบท : ซอฟต์แวร์ช่วยในการออกแบบรองเท้า



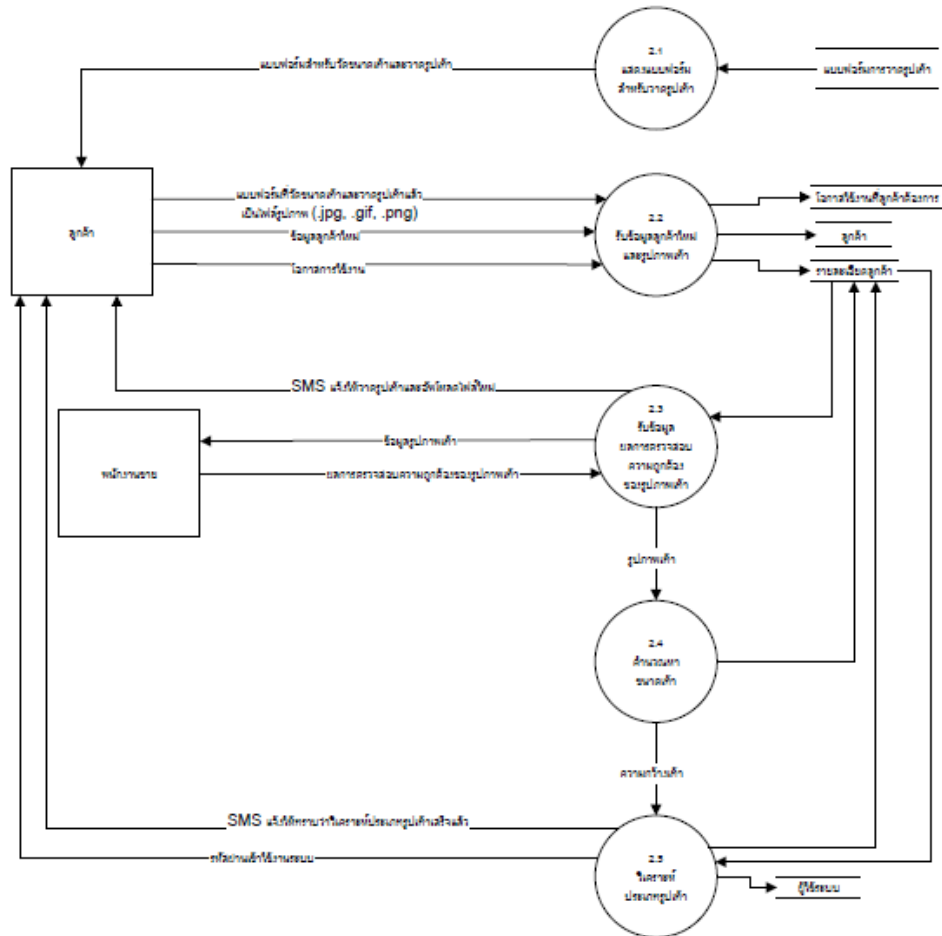


แผนภาพระดับที่ 2 : ซอฟต์แวร์ช่วยในการออกแบบรองเท้า  
 แผนภาพระบบย่อยของกระบวนการที่ 1 รับข้อมูลหลัก

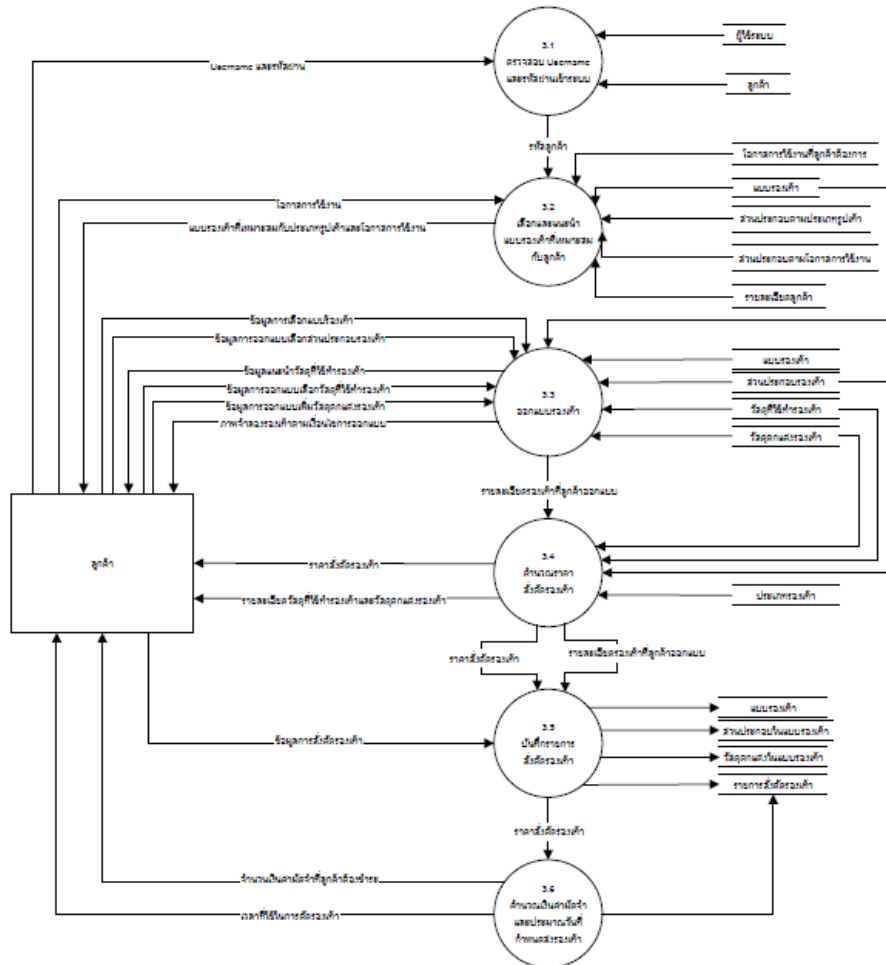




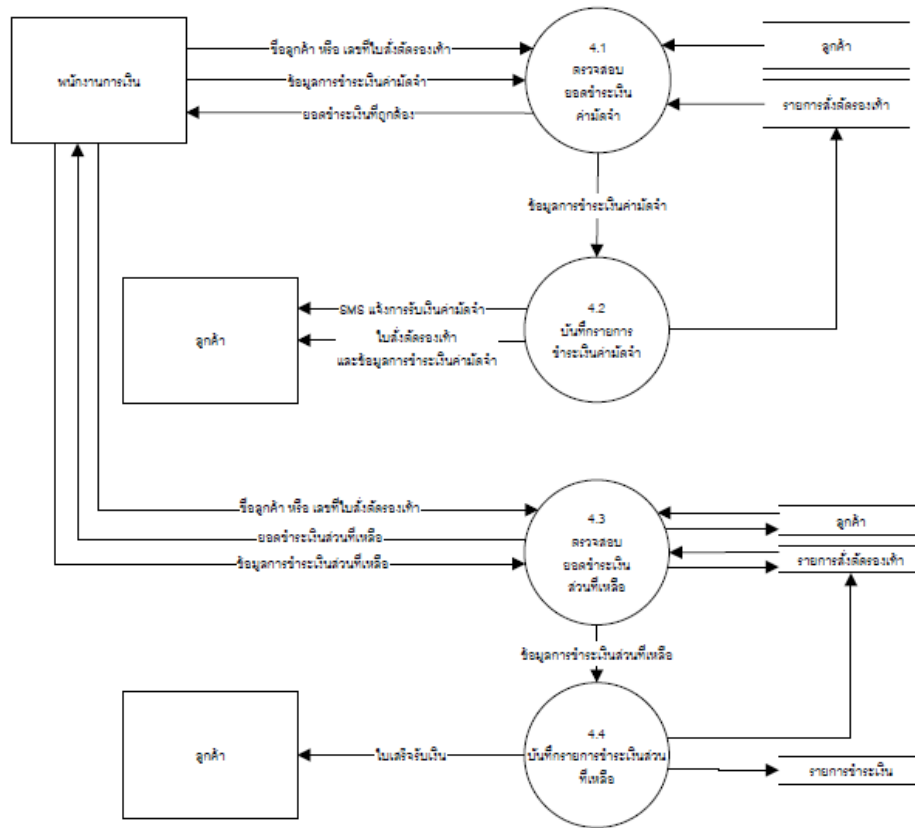
แผนภาพระบบย่อยของกระบวนการที่ 2 วิเคราะห์ประเภทรูปเท้าของลูกค้า



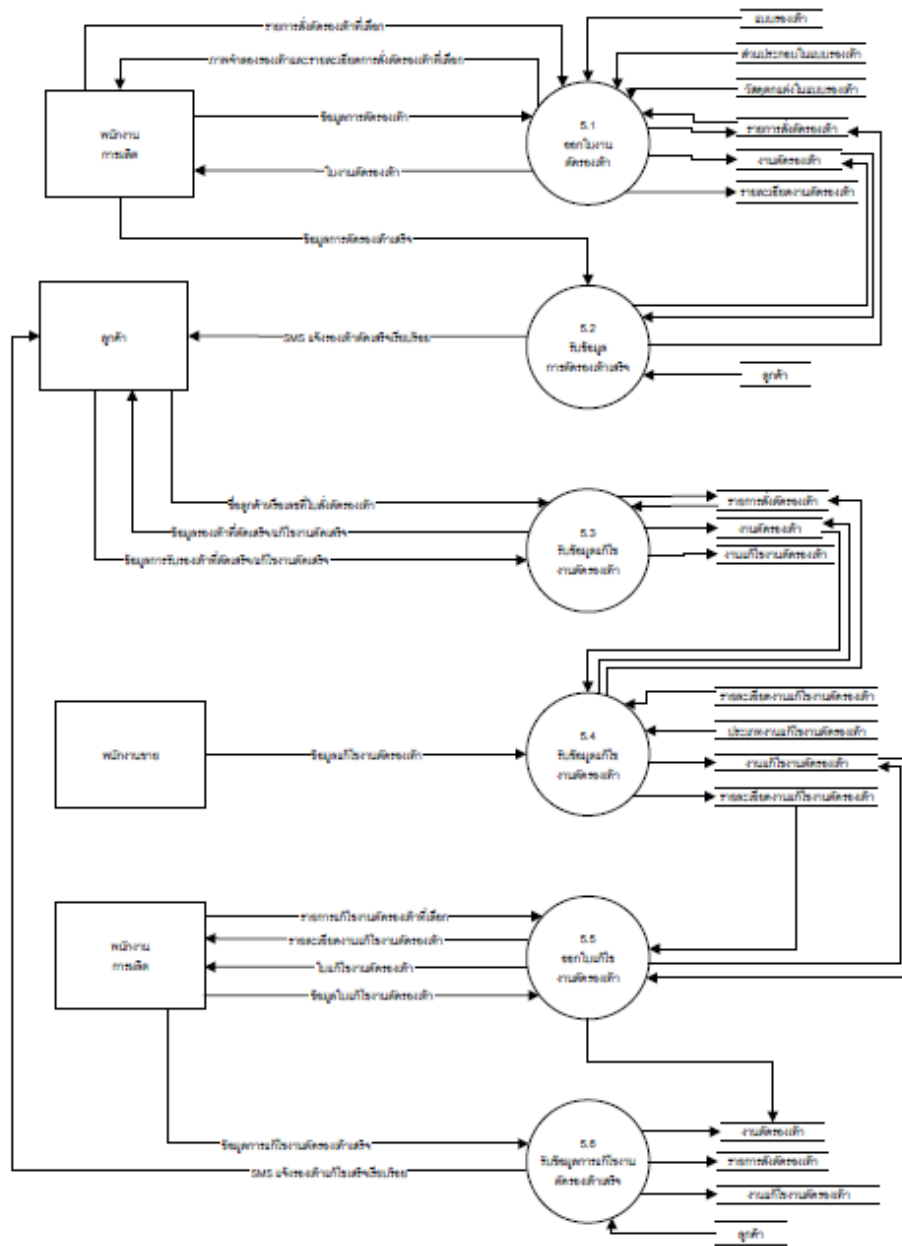
แผนภาพระบบย่อยของกระบวนการที่ 3 ออกแบบและสั่งตัดรองเท้า



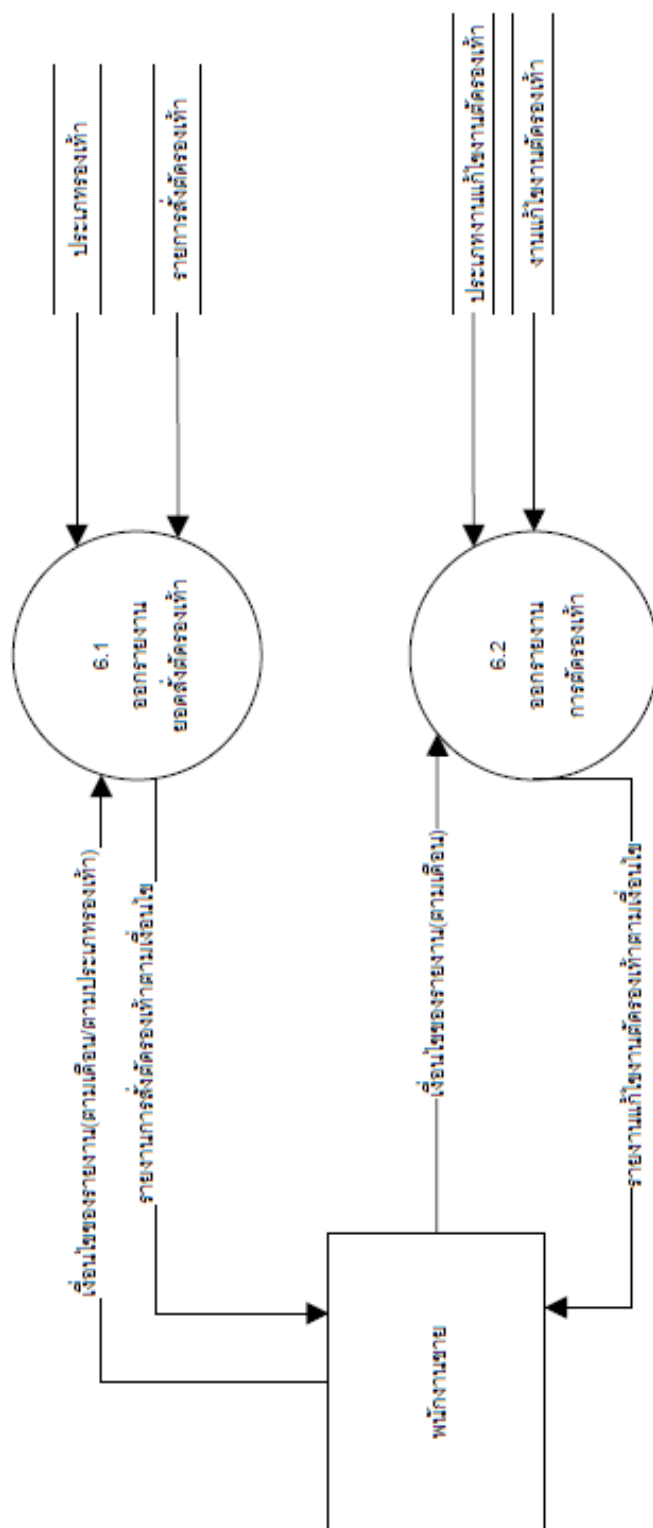
แผนภาพระบบย่อยของกระบวนการที่ 4 การรับเงิน



แผนภาพระบบย่อยของกระบวนการที่ 5 งานตัดรองเท้า



แผนภาพระบบย่อยของกระบวนการที่ 6 ออกรายงาน



พจนานุกรมข้อมูล : ซอฟต์แวร์ช่วยในการออกแบบรองเท้า

4.10.2 รายละเอียดการออกแบบฐานข้อมูล (Data Dictionary)

4.10.2.1 Table: Customer (ลูกค้า)

ตารางที่ 4.12 แสดงรายละเอียดตาราง Customer

No.	Column Name	Data Type	PK	FK	Description
1	customerID	int(11)	✓		รหัสลูกค้า
2	customerName	varchar(100)			ชื่อลูกค้า
3	customerTel	varchar(10)			เบอร์โทรศัพท์ติดต่อลูกค้า
4	customerEmail	varchar(50)			อีเมลลูกค้า
5	Username	varchar(10)		✓	ชื่อผู้ใช้งานระบบ
6	addDate	datetime			วันที่ลูกค้าข้อมูลเข้าระบบ
7	needAnalyze	tinyint(1)			ต้องการให้วิเคราะห์หรือไม่
8	analyzeDate	datetime			วันที่บันทึกผลการวิเคราะห์

4.10.2.2 Table: CustomerInfo (รายละเอียดลูกค้า)

ตารางที่ 4.13 แสดงรายละเอียดตาราง CustomerInfo

No.	Column Name	Data Type	PK	FK	Description
1	customerID	int(11)	✓		รหัสลูกค้า
2	footSide	varchar(1)	✓		เท้าข้างซ้าย/ข้างขวา
3	footImage	varchar(255)			ชื่อไฟล์รูปภาพเท้า
4	width	float			ความกว้างเท้า (ซม.)
5	height	float			ความยาวเท้า (ซม.)
6	perimeter	float			ความยาวเส้นรอบวงรอบเท้า (ซม.)
7	toeType	varchar(10)			ลักษณะนิ้วเท้า
8	footThick	varchar(10)			ลักษณะเนื้อเท้า

4.10.2.3 Table: CustomerOccasion (โอกาสการใช้งานที่ลูกค้าต้องการ)

ตารางที่ 4.14 แสดงรายละเอียดตาราง CustomerOccasion

No.	Column Name	Data Type	PK	FK	Description
1	customerID	int(11)	✓	✓	รหัสลูกค้า
2	occasionID	int(11)	✓	✓	รหัสโอกาสการใช้งาน

4.10.2.4 Table: DecorateType (ประเภทวัสดุตกแต่งรองเท้า)

ตารางที่ 4.15 แสดงรายละเอียดตาราง DecorateType

No.	Column Name	Data Type	PK	FK	Description
1	decorateTypeID	int(11)	✓		รหัสประเภทวัสดุตกแต่งรองเท้า
2	decorateTypeName	varchar(50)			ชื่อประเภทวัสดุตกแต่งรองเท้า

## 4.10.2.5 Table: Decorate (วัสดุตกแต่งรองเท้า)

ตารางที่ 4.16 แสดงรายละเอียดตาราง Decorate

No.	Column Name	Data Type	PK	FK	Description
1	decorateID	int(11)	✓		รหัสวัสดุตกแต่งรองเท้า
2	decorateName	varchar(50)			ชื่อวัสดุตกแต่งรองเท้า
3	shoeTypeID	int(11)		✓	รหัสประเภทรองเท้า
4	decorateTypeID	int(11)		✓	รหัสประเภทวัสดุตกแต่งรองเท้า
5	decoratePrice	float			ราคาวัสดุตกแต่งรองเท้า
6	decorateImage	varchar(255)			ชื่อไฟล์รูปภาพวัสดุตกแต่งรองเท้า
7	hasSubDecorate	tinyint(1)			มีวัสดุตกแต่งรองเท้าย่อย
8	isSubDecorate	tinyint(1)			เป็นวัสดุตกแต่งรองเท้าย่อย
9	materialID	int(11)		✓	รหัสวัสดุที่ใช้ทำรองเท้า

## 4.10.2.6 Table: DecorateAndPartType (วัสดุตกแต่งรองเท้ากับประเภทส่วนประกอบรองเท้า)

ตารางที่ 4.17 แสดงรายละเอียดตาราง DecorateAndPartType

No.	Column Name	Data Type	PK	FK	Description
1	deocorateID	int(11)	✓	✓	รหัสวัสดุตกแต่งรองเท้า
2	partTypeID	int(11)	✓	✓	รหัสประเภทส่วนประกอบรองเท้า

## 4.10.2.7 Table: DecorateCoordinate (จุดประกอบภาพวัสดุตกแต่งรองเท้า)

ตารางที่ 4.18 แสดงรายละเอียดตาราง DecorateCoordinate

No.	Column Name	Data Type	PK	FK	Description
1	decorateID	int(11)	✓	✓	รหัสวัสดุตกแต่งรองเท้า
2	shotID	int(11)	✓		รหัสรูปภาพ
3	lineID	int(11)	✓		รหัสเส้น
4	coordinate	varchar(255)			จุดประกอบบนเส้น
5	defaultLine	int(11)			ขนาดเส้นเริ่มต้น
6	defaultColor	varchar(100)			สีเริ่มต้น

## 4.10.2.8 Table: FixingType (ประเภทงานแก้ไขงานตัดรองเท้า)

ตารางที่ 4.19 แสดงรายละเอียดตาราง FixingType

No.	Column Name	Data Type	PK	FK	Description
1	fixTypeID	int(11)	✓		รหัสประเภทงานแก้ไขงานตัดรองเท้า
2	fixTypeName	varchar(50)			ชื่อประเภทงานแก้ไขงานตัดรองเท้า
3	fixCharge	tinyint(1)			มีค่าใช้จ่ายในการแก้ไขหรือไม่
4	errorFrom	varchar(10)			เป็นความผิดพลาดของ

## 4.10.2.9 Table: Occasion (โอกาสการใช้งาน)

ตารางที่ 4.20 แสดงรายละเอียดตาราง Occasion

No.	Column Name	Data Type	PK	FK	Description
1	occasionID	int(11)	✓		รหัสโอกาสการใช้งาน
2	occasionName	varchar(100)			ชื่อโอกาสการใช้งาน

## 4.10.2.10 Table: MaterialType (ประเภทวัสดุที่ใช้ทำรองเท้า)

ตารางที่ 4.21 แสดงรายละเอียดตาราง MaterialType

No.	Column Name	Data Type	PK	FK	Description
1	materialTypeID	int(11)	✓		รหัสประเภทวัสดุที่ใช้ทำรองเท้า
2	materialTypeName	varchar(100)			ชื่อประเภทวัสดุที่ใช้ทำรองเท้า
3	effectOccasion	tinyint(1)			มีผลกระทบบกโอกาสการใช้งานหรือไม่

## 4.10.2.11 Table: Material (วัสดุที่ใช้ทำรองเท้า)

ตารางที่ 4.22 แสดงรายละเอียดตาราง Material

No.	Column Name	Data Type	PK	FK	Description
1	materialID	int(11)	✓		รหัสวัสดุที่ใช้ทำรองเท้า
2	materialName	varchar(100)			ชื่อวัสดุที่ใช้ทำรองเท้า
3	materialTypeID	int(11)		✓	รหัสประเภทวัสดุที่ใช้ทำรองเท้า
4	materialPattern	varchar(255)			ชื่อไฟล์รูปภาพวัสดุที่ใช้ทำรองเท้า
5	priceType	smallint(1)			ประเภทราคาวัสดุที่ใช้ทำรองเท้า
6	materialPrice1	float			ราคาต่อจำนวนของวัสดุที่ใช้ทำรองเท้า
7	materialPrice2	float			ราคาเหมาะต่อ 1 ส่วนประกอบ

## 4.10.2.12 Table: MaterialTypeAndPartType (ประเภทวัสดุที่ใช้ทำรองเท้าตามประเภทส่วนประกอบรองเท้า)

ตารางที่ 4.23 แสดงรายละเอียดตาราง MaterialTypeAndPartType

No.	Column Name	Data Type	PK	FK	Description
1	materialTypeID	int(11)	✓	✓	รหัสประเภทวัสดุที่ใช้ทำรองเท้า
2	partTypeID	int(11)	✓	✓	รหัสประเภทส่วนประกอบรองเท้า

## 4.10.2.13 Table: PartWithStyle (ส่วนประกอบกับประเภทย่อยส่วนประกอบรองเท้า)

ตารางที่ 4.24 แสดงรายละเอียดตาราง PartWithStyle

No.	Column Name	Data Type	PK	FK	Description
1	partID	int(11)	✓	✓	รหัสส่วนประกอบรองเท้า
2	partStyleID	int(11)	✓	✓	รหัสประเภทย่อยส่วนประกอบรองเท้า



4.10.2.14 Table: MaterialTypeWithOccasion (ประเภทวัสดุที่ใช้ทำรองเท้าตามโอกาสการใช้งาน)  
ตารางที่ 4.25 แสดงรายละเอียดตาราง MaterialTypeWithOccasion

No.	Column Name	Data Type	PK	FK	Description
1	materialTypeID	int(11)	✓	✓	รหัสประเภทวัสดุที่ใช้ทำรองเท้า
2	occasionID	int(11)	✓	✓	รหัสโอกาสการใช้งาน
3	score	int(11)			คะแนนความเหมาะสมระหว่าง ส่วนประกอบรองเท้ากับโอกาสการใช้งาน

4.10.2.15 Table: Part (ส่วนประกอบรองเท้า)  
ตารางที่ 4.26 แสดงรายละเอียดตาราง Part

No.	Column Name	Data Type	PK	FK	Description
1	partID	int(11)	✓		รหัสส่วนประกอบรองเท้า
2	partName	varchar(100)			ชื่อส่วนประกอบรองเท้า
3	partTypeID	int(11)		✓	รหัสประเภทส่วนประกอบรองเท้า
4	shoeTypeID	int(11)		✓	รหัสประเภทรองเท้า
5	additionalPrice	float			ราคาส่วนประกอบรองเท้าพิเศษ
6	materialUse	float			จำนวนวัสดุที่ใช้ทำส่วนประกอบ
7	partImage	varchar(255)			ชื่อไฟล์รูปภาพส่วนประกอบรองเท้า
8	hasSubPart	tinyint(1)			มีส่วนประกอบย่อยหรือไม่
9	defaultPart	tinyint(1)			เป็นส่วนประกอบเริ่มต้นหรือไม่

4.10.2.16 Table: PartStyle (ประเภทย่อยส่วนประกอบรองเท้า)  
ตารางที่ 4.27 แสดงรายละเอียดตาราง PartStyle

No.	Column Name	Data Type	PK	FK	Description
1	partStyleID	int(11)	✓		รหัสประเภทย่อยส่วนประกอบรองเท้า
2	partStyleName	varchar(100)			ชื่อประเภทย่อยส่วนประกอบรองเท้า
3	effectFootShape	tinyint(1)			มีผลกระทบต่อประเภทรูปเท้าหรือไม่
4	effectOccasion	tinyint(1)			มีผลกระทบต่อโอกาสการใช้งานหรือไม่

4.10.2.17 Table: PartStyleWithOccasion (ประเภทย่อยส่วนประกอบรองเท้าตามโอกาสการใช้งาน)  
ตารางที่ 4.28 แสดงรายละเอียดตาราง PartStyleWithOccasion

No.	Column Name	Data Type	PK	FK	Description
1	partStyleID	int(11)	✓	✓	รหัสประเภทย่อยส่วนประกอบรองเท้า
2	occasionID	int(11)	✓	✓	รหัสโอกาสการใช้งาน
3	score	int(11)			คะแนนความเหมาะสมระหว่าง ส่วนประกอบรองเท้ากับโอกาสการใช้งาน

4.10.2.18 Table: PartType (ประเภทส่วนประกอบรองเท้า)

ตารางที่ 4.29 แสดงรายละเอียดตาราง PartType

No.	Column Name	Data Type	PK	FK	Description
1	partTypeID	int(11)	✓		รหัสประเภทส่วนประกอบรองเท้า
2	partTypeName	varchar(100)			ชื่อประเภทส่วนประกอบรองเท้า
3	partLayer	int(11)			ลำดับการวางภาพ
4	canDecorate	tinyint(1)			สามารถเพิ่มวัสดุตกแต่งได้หรือไม่
5	canDesign	tinyint(1)			สามารถเปลี่ยนชิ้นส่วนประกอบได้หรือไม่

4.10.2.19 Table: PartStyleWithFootShape (ประเภทย่อยส่วนประกอบรองเท้าตามประเภทรูปเท้า)

ตารางที่ 4.30 แสดงรายละเอียดตาราง PartStyleWithFootShape

No.	Column Name	Data Type	PK	FK	Description
1	partStyleID	int(11)	✓	✓	รหัสประเภทย่อยส่วนประกอบรองเท้า
2	Egyptian	tinyint(1)			เหมาะกับลักษณะนิ้วเท้าแบบ Egyptian
3	Greek	tinyint(1)			เหมาะกับลักษณะนิ้วเท้าแบบ Greek
4	Square	tinyint(1)			เหมาะกับลักษณะนิ้วเท้าแบบ Square
5	Flat	tinyint(1)			เหมาะกับลักษณะเนื้อเท้าแบนราบ
6	Thick	tinyint(1)			เหมาะกับลักษณะเนื้อเท้า อวบอูม

4.10.2.20 Table: PartStyleAndType (ประเภทย่อยส่วนประกอบรองเท้ากับประเภทส่วนประกอบรองเท้า)

ตารางที่ 4.31 แสดงรายละเอียดตาราง PartStyleAndType

No.	Column Name	Data Type	PK	FK	Description
1	partStyleID	int(11)	✓	✓	รหัสประเภทย่อยส่วนประกอบรองเท้า
2	partTypeID	int(11)	✓	✓	รหัสประเภทส่วนประกอบรองเท้า

4.10.2.21 Table: PartCoordinate (จุดประกอบภาพส่วนประกอบรองเท้า)

ตารางที่ 4.32 แสดงรายละเอียดตาราง PartCoordinate

No.	Column Name	Data Type	PK	FK	Description
1	partID	int(11)	✓	✓	รหัสส่วนประกอบรองเท้า
2	shotID	int(11)	✓		รหัสมุมมองภาพ
3	lineID	int(11)	✓		รหัสเส้น
4	coordinate	varchar(255)			จุดประกอบบนเส้น
5	defaultLine	int(11)			ขนาดเส้นเริ่มต้น
6	defaultColor	varchar(50)			สีเริ่มต้น

## 4.10.2.22 Table: Status (สถานะรายการสั่งตัดรองเท้า)

ตารางที่ 4.33 แสดงรายละเอียดตาราง Status

No.	Column Name	Data Type	PK	FK	Description
1	statusID	int(11)	✓		รหัสสถานะรายการสั่งตัดรองเท้า
2	statusName	varchar(100)			ชื่อสถานะรายการสั่งตัดรองเท้า

## 4.10.2.23 Table: Payment (รายการชำระเงิน)

ตารางที่ 4.34 แสดงรายละเอียดตาราง Payment

No.	Column Name	Data Type	PK	FK	Description
1	paymentID	varchar(8)	✓		รหัสรายการชำระเงิน
2	paymentDate	date			วันที่ชำระเงิน
3	orderId	varchar(8)		✓	รหัสรายการสั่งตัดรองเท้า
4	customerID	int(11)		✓	รหัสลูกค้า
5	paymentMethod	varchar(50)			วิธีการชำระเงิน
6	amountBeforeVat	float			ยอดเงินก่อนรวมภาษีมูลค่าเพิ่ม
7	Vat	float			ภาษีมูลค่าเพิ่ม
8	paymentAmount	float			จำนวนเงินที่ชำระ

## 4.10.2.24 Table: Shoe (แบบรองเท้า)

ตารางที่ 4.35 แสดงรายละเอียดตาราง Shoe

No.	Column Name	Data Type	PK	FK	Description
1	shoeID	int(11)	✓		รหัสแบบรองเท้า
2	shoeName	varchar(100)			ชื่อแบบรองเท้า
3	shoeTypeID	int(11)		✓	รหัสประเภทรองเท้า
4	shoeImage	varchar(255)			ชื่อไฟล์รูปภาพรองเท้า
5	shoeDrawing	varchar(255)			รูปภาพรองเท้าจำลอง
6	defaultShoe	tinyint(1)			เป็นแบบรองเท้าเริ่มต้นหรือไม่
7	canShow	tinyint(1)			แสดงแบบรองเท้านี้ได้
8	customerShoe	tinyint(1)			เป็นรองเท้าที่ลูกค้าออกแบบ

## 4.10.2.25 Table: ShoePart (ส่วนประกอบในแบบรองเท้า)

ตารางที่ 4.36 แสดงรายละเอียดตาราง ShoePart

No.	Column Name	Data Type	PK	FK	Description
1	shoeID	int(11)	✓	✓	รหัสแบบรองเท้า
2	partID	int(11)	✓	✓	รหัสส่วนประกอบรองเท้า
3	materialID	int(11)		✓	รหัสวัสดุที่ใช้ทำรองเท้า

4.10.2.26 Table: ShoeDecoration (วัสดุตกแต่งรองเท้าในแบบรองเท้า)

ตารางที่ 4.37 แสดงรายละเอียดตาราง ShoeDecoration

No.	Column Name	Data Type	PK	FK	Description
1	shoeID	int(11)	✓	✓	รหัสแบบรองเท้า
2	decoratelD	int(11)	✓	✓	รหัสวัสดุตกแต่งรองเท้า
3	partTypeID	int(11)		✓	รหัสประเภทส่วนประกอบรองเท้า

4.10.2.27 Table: ShoeCutting (งานตัดรองเท้า)

ตารางที่ 4.38 แสดงรายละเอียดตาราง ShoeCutting

No.	Column Name	Data Type	PK	FK	Description
1	cuttingID	varchar(8)	✓		รหัสงานตัดรองเท้า
2	cuttingDate	datetime			วันที่เริ่มตัดรองเท้า
3	ordered	varchar(8)		✓	รหัสรายการสั่งตัดรองเท้า
4	shoeID	int(11)		✓	รหัสแบบรองเท้า
5	finishedDate	datetime			วันที่รองเท้าตัดเสร็จ
6	statusID	int(11)		✓	รหัสสถานะ
7	Remark	varchar(255)			หมายเหตุงานตัดรองเท้า
8	addDate	datetime			วันที่-เวลาทำรายการงานตัดรองเท้า

4.10.2.28 Table: ShoeCuttingDetail (รายละเอียดงานตัดรองเท้า)

ตารางที่ 4.39 แสดงรายละเอียดตาราง ShoeCuttingDetail

No.	Column Name	Data Type	PK	FK	Description
1	cuttingID	varchar(8)	✓		รหัสงานตัดรองเท้า
2	partTypeID	int(11)	✓	✓	รหัสประเภทส่วนประกอบรองเท้า
3	objID	int(11)		✓	รหัสวัสดุ (รหัสส่วนประกอบรองเท้า หรือรหัสวัสดุตกแต่งรองเท้า)
4	objType	varchar(1)			ประเภทวัสดุ (ส่วนประกอบรองเท้า หรือวัสดุตกแต่งรองเท้า)
5	useAmount	varchar(255)			จำนวนวัสดุที่ใช้ทำรองเท้า
6	materialID	int(11)		✓	รหัสวัสดุที่ใช้ทำรองเท้า

4.10.2.29 Table: SubPart (ส่วนประกอบย่อย)

ตารางที่ 4.40 แสดงรายละเอียดตาราง SubPart

No.	Column Name	Data Type	PK	FK	Description
1	partID	int(11)	✓	✓	รหัสส่วนประกอบรองเท้า
2	subpartID	int(11)	✓	✓	รหัสส่วนประกอบรองเท้าย่อย

## 4.10.2.30 Table: SubDecorate (วัสดุตกแต่งรองเท้าย่อย)

ตารางที่ 4.41 แสดงรายละเอียดตาราง SubDecorate

No.	Column Name	Data Type	PK	FK	Description
1	decorateID	int(11)	✓	✓	รหัสวัสดุตกแต่งรองเท้า
2	subDecorateID	int(11)	✓	✓	รหัสวัสดุตกแต่งรองเท้าย่อย
3	materialID	int(11)		✓	รหัสวัสดุที่ใช้ทำรองเท้า
4	layer	int(11)			ลำดับชั้นวางภาพ

## 4.10.2.31 Table: ShoeType (ประเภทรองเท้า)

ตารางที่ 4.42 แสดงรายละเอียดตาราง ShoeType

No.	Column Name	Data Type	PK	FK	Description
1	shoeTypeID	int(11)	✓		รหัสประเภทรองเท้า
2	shoeTypeName	varchar(100)			ชื่อประเภทรองเท้า
3	shoeTypePrice	float			ราคาตามประเภทรองเท้า

## 4.10.2.32 Table: ShoeFixing (งานแก้ไขงานตัดรองเท้า)

ตารางที่ 4.43 แสดงรายละเอียดตาราง ShoeFixing

No.	Column Name	Data Type	PK	FK	Description
1	fixID	varchar(8)	✓		รหัสงานแก้ไขงานตัดรองเท้า
2	saveDate	datetime			วันที่บันทึกงานแก้ไขงานตัดรองเท้า
3	fixDate	datetime			วันที่แก้ไขงานตัดรองเท้า
4	cuttingID	varchar(8)		✓	รหัสงานตัดรองเท้า
5	orderID	varchar(8)		✓	รหัสรายการสั่งตัดรองเท้า
6	shoeID	int(11)		✓	รหัสแบบรองเท้า
7	totalFixPrice	float			ราคาแก้ไขงานตัดรองเท้า
8	finishedDate	datetime			วันที่แก้ไขงานตัดเสร็จ
9	remark	varchar(255)			หมายเหตุงานแก้ไขงานตัดรองเท้า
10	statusID	int(11)		✓	รหัสสถานะ

## 4.10.2.33 Table: UserLogin (ผู้ใช้ระบบ)

ตารางที่ 4.44 แสดงรายละเอียดตาราง UserLogin

No.	Column Name	Data Type	PK	FK	Description
1	username	varchar(10)	✓		ชื่อผู้ใช้งานระบบ
2	customerID	int(11)		✓	รหัสลูกค้า
3	password	varchar(10)			รหัสผ่านเข้าระบบ

4.10.2.34 Table: ShoeFixingDetail (รายละเอียดงานแก้ไขงานตัดรองเท้า)

ตารางที่ 4.45 แสดงรายละเอียดตาราง ShoeFixingDetail

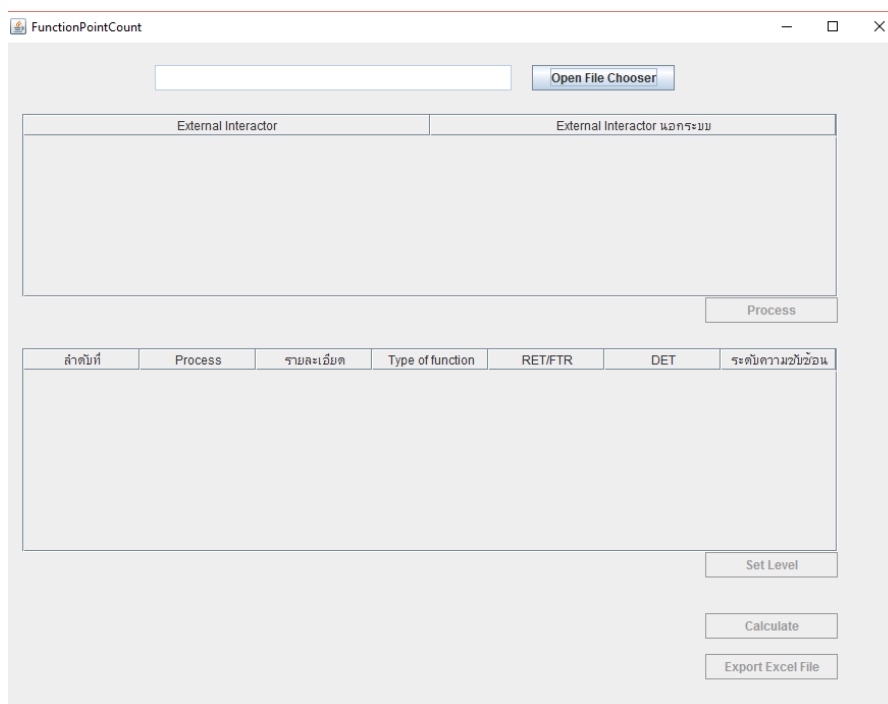
No.	Column Name	Data Type	PK	FK	Description
1	fixID	varchar(8)	✓	✓	รหัสงานแก้ไขงานตัดรองเท้า
2	lineID	int(11)	✓		เลขที่งานแก้ไขงานตัดรองเท้า
3	partTypeID	int(11)		✓	รหัสประเภทส่วนประกอบรองเท้า
4	fixTypeID	int(11)		✓	ประเภทงานแก้ไขงานตัดรองเท้า
5	fixDetail	varchar(255)			รายละเอียดการแก้ไข
6	fixPrice	Float			ค่าใช้จ่ายในการแก้ไขงานตัดรองเท้า

4.10.2.35 Table: ShoeOrder (รายการสั่งตัดรองเท้า)

ตารางที่ 4.46 แสดงรายละเอียดตาราง ShoeOrder

No.	Column Name	Data Type	PK	FK	Description
1	ordered	varchar(8)	✓		รหัสรายการสั่งตัดรองเท้า
2	orderDate	datetime			วันที่สั่งตัดรองเท้า
3	dueDate	date			วันที่กำหนดรองเท้าเสร็จ
4	customerID	int(11)		✓	รหัสลูกค้า
5	shoeID	int(11)		✓	รหัสแบบรองเท้าที่สั่งตัด
6	cuttingPrice	float			ราคาค่าสั่งตัดรองเท้าทั้งหมด
7	depositAmount	float			จำนวนเงินค้ำมัดจำ
8	fixingPrice	float			ค่าใช้จ่ายในการแก้ไข
9	totalPrice	float			ยอดเงินรวมทั้งหมด
10	statusID	int(11)		✓	รหัสสถานะ
11	depositDate	date			วันที่ชำระเงินค้ำมัดจำ
12	depositBank	varchar(100)			ชื่อนาคารที่ชำระเงิน
13	addDepositDate	datetime			วันที่-เวลาทำรายการรับเงินค้ำมัดจำ

หน้าจอและคู่มือการใช้งานเครื่องมือสำหรับการนับค่าฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล



รูปที่ 1 หน้าจอแรกของเครื่องมือการนับค่าฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูล

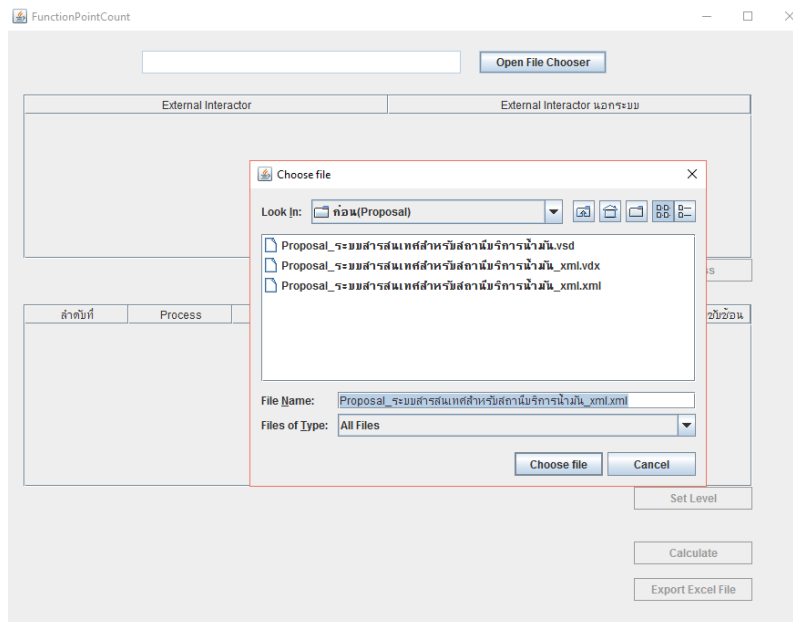
เครื่องมือการนับค่าฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลแบ่งออกเป็น 3 ส่วนการทำงาน

คือ 1.) ส่วนรับข้อมูลเข้าสำหรับเครื่องมือ

2.) ส่วนที่ผู้ใช้งานระบุส่วนที่เป็น External Interactor นอกกระบวน

3.) ส่วนที่เป็นการแยกประเภทฟังก์ชันพอยต์

## ส่วนที่ 1 : ส่วนรับข้อมูลเข้าสำหรับเครื่องมือ



รูปที่ 2 หน้าจอแสดงการเลือกไฟล์ข้อมูลเข้าสำหรับเครื่องมือ

โดยข้อมูลเข้าสำหรับเครื่องมือ จะเป็นไฟล์แผนภาพกระแสข้อมูลที่อยู่ในรูปแบบโครงสร้างไฟล์ XML โดยสร้างจากเครื่อง Microsoft Visio โดยบันทึกไฟล์เป็น .VDX และใช้เครื่องมือ Notepad++ เปลี่ยนโครงสร้างไฟล์ให้อยู่ในรูปแบบ XML



## ส่วนที่ 2 : ส่วนที่ผู้ใช้งานระบุส่วนที่เป็น External Interacter นอกกระบบ

เมื่อผู้ใช้งานเลือกไฟล์แผนภาพกระแสข้อมูลที่อยู่ในรูปแบบโครงสร้างไฟล์ XML เครื่องมือจะแสดง External Interacter ที่ปรากฏอยู่ในแผนภาพกระแสข้อมูล โดยผู้ใช้งานเครื่องมือจะเป็นผู้ระบุส่วนที่เป็น External Interacter ภายนอกกระบบ

External Interactor	External Interactor นอกกระบบ
พนักงานเก็บเงิน	<input type="checkbox"/>
เจ้าของหรือผู้บริหารสถานบริการน้ำมัน	<input type="checkbox"/>
เจ้าหน้าที่บริการสถานบริการน้ำมัน	<input type="checkbox"/>
รถยนต์ที่เข้ามาให้บริการ	<input checked="" type="checkbox"/>
ตู้จ่ายน้ำมัน	<input checked="" type="checkbox"/>
ระบบการชำระเงินด้วยบัตรเครดิต	<input checked="" type="checkbox"/>

ลำดับที่	Process	รายละเอียด	Type of function	RET/FTR	DET	ระดับความซับซ้อน

รูปที่ 3 หน้าจอแสดงการระบุส่วนที่เป็น External Interacter นอกกระบบ

เมื่อกดปุ่ม Process เครื่องมือจะแยกประเภทฟังก์ชันพอยต์จากแผนภาพกระแสข้อมูลตามหลักการแยกประเภทฟังก์ชันพอยต์ ที่ระบุไว้ในบทที่ 4 หัวข้อที่ 4.2 ดังรูปที่ 4

The screenshot shows the 'FunctionPointCount' application window. At the top, there is an 'Open File Chooser' button. Below it is a table for 'External Interactor' with columns for 'External Interactor' and 'External Interactor แอกระบบ'. The table lists various roles like 'พนักงานเก็บเงิน', 'เจ้าของหรือผู้บริหารสถานบริการน้ำมัน', etc., with checkboxes in the second column. A 'Process' button is located below this table. At the bottom, there is a table with columns: ลำดับที่, Process, รายละเอียด, Type of function, RET/FTR, DET, and ระดับความซับซ้อน. The table contains 11 rows of data. Below the table are buttons for 'Set Level', 'Calculate', and 'Export Excel File'.

ลำดับที่	Process	รายละเอียด	Type of function	RET/FTR	DET	ระดับความซับซ้อน
1	1	ข้อมูลผู้ใช้บริการ	ILF			
2	1	ข้อมูลชื่อผู้ใช้งานแ...	EI			
3	1	ข้อมูลการตรวจสอบ...	EO			
4	1	ข้อมูลชื่อร้านค้า...	EI			
5	1	ข้อมูลที่อยู่และห...	EI			
6	1	ข้อมูลบาร์โค้ด...	EI			
7	1	ข้อมูลหมายเลข...	EO			
8	1	ข้อมูลผู้ใช้งานระบบ	ILF			
9	1	ข้อมูลกลุ่มผู้ใช้งาน	ILF			
10	1	ข้อมูลสิทธิ์การช...	ILF			
11	1	ข้อมูลน้ำมันที่เติม...	ILF			

รูปที่ 4 หน้าจอแสดงเมื่อผู้ใช้งานกดปุ่ม Process

ส่วนที่ 3 : ส่วนที่เป็นการแยกประเภทฟังก์ชันพอยต์

The screenshot shows the 'FunctionPointCount' application window. At the top, there is an 'Open File Chooser' button. Below it is a table for 'External Interactor' with columns for 'External Interactor' and 'External Interactor แอกระบบ'. The table lists various roles like 'พนักงานเก็บเงิน', 'เจ้าของหรือผู้บริหารสถานบริการน้ำมัน', etc., with checkboxes in the second column. A 'Process' button is located below this table. At the bottom, there is a table with columns: ลำดับที่, Process, รายละเอียด, Type of function, RET/FTR, DET, and ระดับความซับซ้อน. The table contains 11 rows of data with numerical values in the RET/FTR, DET, and complexity columns. Below the table are buttons for 'Set Level', 'Calculate', and 'Export Excel File'.

ลำดับที่	Process	รายละเอียด	Type of function	RET/FTR	DET	ระดับความซับซ้อน
1	1	ข้อมูลผู้ใช้บริการ	ILF	2	8	-
2	1	ข้อมูลชื่อผู้ใช้งานแ...	EI	1	10	-
3	1	ข้อมูลการตรวจสอบ...	EO	1	8	-
4	1	ข้อมูลชื่อร้านค้า...	EI	1		-
5	1	ข้อมูลที่อยู่และห...	EI	1		-
6	1	ข้อมูลบาร์โค้ด...	EI	1		-
7	1	ข้อมูลหมายเลข...	EO	1		-
8	1	ข้อมูลผู้ใช้งานระบบ	ILF	3		-
9	1	ข้อมูลกลุ่มผู้ใช้งาน	ILF	4		-
10	1	ข้อมูลสิทธิ์การช...	ILF	4		-
11	1	ข้อมูลน้ำมันที่เติม...	ILF	3		-

รูปที่ 5 หน้าจอแสดงเครื่องมือแยกประเภทฟังก์ชันพอยต์

จากรูปที่ 5 จะประกอบไปด้วย

“ลำดับที่” คือลำดับของข้อมูลที่แยกประเภทฟังก์ชันออกมา

“Process” คือ ระดับของแผนภาพที่ข้อมูลปรากฏอยู่

“รายละเอียด” คือ ข้อความของข้อมูลที่เครื่องมืออ่านค่าได้

“Type of function” คือ ประเภทฟังก์ชันพอยต์ของข้อมูลที่เครื่องมืออ่านค่าได้

“RET/FTR” คือ จำนวนตารางข้อมูลและจำนวนเพิ่มข้อมูลอ้างอิง

“DET” คือ จำนวนเขตข้อมูล

“ระดับความซับซ้อน” คือ ผลการคำนวณค่าความซับซ้อนของค่าฟังก์ชันพอยต์ที่เกิดจาก RET/FTR และ DET โดยระดับความซับซ้อนแบ่งออกเป็น 3 ระดับคือ 1.) ระดับความซับซ้อนต่ำ (Low) 2.) ระดับความซับซ้อนปานกลาง (Average) และ 3.) ระดับความซับซ้อนสูง(High)

โดยส่วนนี้จะแบ่งการทำงานออกเป็น 3 ส่วนคือ 1.) ผู้ใช้งานระบุส่วนที่เป็น RET/FTR และ DET 2.) เครื่องแสดงผลคำนวณค่าฟังก์ชันพอยต์ 3.) การบันทึกผลการคำนวณค่าฟังก์ชันพอยต์

1.) ผู้ใช้งานระบุส่วนที่เป็น RET/FTR และ DET ที่ได้ข้อมูลมาจากพจนานุกรมข้อมูลที่เก็บมาจากโครงการพัฒนาซอฟต์แวร์ เมื่อผู้ใช้งานระบุส่วนดังกล่าวครบทุกข้อมูลแล้ว เมื่อกดปุ่ม Set Level เครื่องมือจะคำนวณระดับความซับซ้อนของฟังก์ชันพอยต์

The screenshot shows the 'FunctionPointCount' application window. At the top, there is an 'Open File Chooser' button. Below it is a table for selecting 'External Interactor' and 'External Interactor แอกระบบ'. The table has two columns: 'External Interactor' and 'External Interactor แอกระบบ'. The rows are:

External Interactor	External Interactor แอกระบบ
พนักงานเก็บเงิน	<input type="checkbox"/>
เจ้าของหรือผู้บริหารสถานบริการน้ำมัน	<input type="checkbox"/>
เจ้าหน้าที่ประจำสถานบริการน้ำมัน	<input type="checkbox"/>
รถยนต์ที่เข้ามาให้บริการ	<input checked="" type="checkbox"/>
ตู้จำหน่ายน้ำมัน	<input checked="" type="checkbox"/>
ระบบการชำระเงินด้วยบัตรเครดิต	<input checked="" type="checkbox"/>

Below this table is a 'Process' button. Underneath is a table with the following columns: ลำดับที่, Process, รายละเอียด, Type of function, RET/FTR, DET, and ระดับความซับซ้อน. The data rows are:

ลำดับที่	Process	รายละเอียด	Type of function	RET/FTR	DET	ระดับความซับซ้อน
19	1	รายงานการรับราคา...	EQ	1	16	Low
20	1	ข้อมูลผู้ใช้งานระบบและส...	EI	1	8	Low
21	1	ข้อมูลสถานบริการน้ำมัน	ILF	1	7	Low
22	1	ข้อมูลเวลาเปิด-ปิด	ILF	2	5	Low
23	1	ข้อมูลตู้จ่าย	ILF	3	3	Low
24	1	ข้อมูลถังเก็บ	ILF	5	5	Low
25	1	ข้อมูลชนิดน้ำมันที่กำหนด	ILF	6	6	Average
26	1	ข้อมูลบริเวณพื้นที่มีการใช้	ILF	8	7	Average
27	1	ข้อมูลจำนวนเงินที่ลูกค้าช...	EI	1	4	Low
28	1	ข้อมูลการชำระเงินด้วย...	EI	1	5	Low
29	1	ข้อมูลบริเวณพื้นที่ใช้	EI	1	5	Low

At the bottom right, there are buttons for 'Set Level', 'Calculate', and 'Export Excel File'.

รูปที่ 6 หน้าจอแสดงเมื่อกดปุ่ม Set Level

## 2.) เครื่องแสดงผลคำนวณค่าฟังก์ชันพอยต์

เมื่อกดปุ่ม Calculate เครื่องมือจะแสดงหน้าต่างการคำนวณค่าฟังก์ชันพอยต์ที่ได้

Type of Component	Low	Average	High	Total
External Inputs (EI)	26 x 3 = 78	0 x 4 = 0	0 x 6 = 0	78
External Outputs (EO)	14 x 4 = 56	0 x 5 = 0	0 x 7 = 0	56
External Inquiries (EQ)	4 x 3 = 12	0 x 4 = 0	0 x 6 = 0	12
Internal Logical Files (ILF)	16 x 7 = 112	2 x 10 = 20	0 x 15 = 0	132
External Interface Files (EIF)	0 x 5 = 0	0 x 7 = 0	0 x 10 = 0	0
<b>Total</b>	<b>258</b>	<b>20</b>	<b>0</b>	<b>278</b>

รูปที่ 7 หน้าจอแสดงผลการคำนวณค่าฟังก์ชันพอยต์

## 3.) การบันทึกผลการคำนวณค่าฟังก์ชันพอยต์

เมื่อกดปุ่ม Export Excel File เครื่องมือจะบันทึกผลการคำนวณค่าฟังก์ชันพอยต์ เข้าไปที่ Drive D: โดยชื่อไฟล์จะเป็น result\_ “ชื่อของไฟล์ที่นำเข้า”

ลำดับที่	Process	รายละเอียด	Type of function	RET/FTR	DET	ระดับความซับซ้อน
19	1	รายงานขายบริษัทนำ...	EQ	1	16	Low
20	1	ข้อมูลใช้งานระบบแอส...	EI	1	8	Low
21	1	ข้อมูลสถานะบริการนำม...	ILF	1	7	Low
22	1	ข้อมูลสถานะเช็ค-บิล...	ILF	2	5	Low
23	1	ข้อมูลค้างจ่าย...	ILF	3	3	Low
24	1	ข้อมูลตั้งถิ่น...	ILF	5	5	Low
25	1	ข้อมูลชนิดน้ำมันที่จำหน...	ILF	6	6	Average
26	1	ข้อมูลใบรวมหนี้ที่มีการ...	ILF	8	7	Average
27	1	ข้อมูลอำนาจเงินที่ลูกค...	EI	1	4	Low
28	1	ข้อมูลการชำระเงินเดบ...	EI	1	5	Low
29	1	ข้อมูลใบรวมหนี้ที่ซ...	EI	1	5	Low

รูปที่ 8 หน้าจอแสดงผลการบันทึกไฟล์การคำนวณค่าฟังก์ชันพอยต์

ผลการบันทึกไฟล์การคำนวณค่าฟังก์ชันพอยต์

	A	B	C	D	E	F	G	H	I
1	ลำดับที่	Process	รายละเอียด	Type of fu	RET/FTR	DET	ระดับความซับซ้อน		
2	1	1	ข้อมูลผู้ใช้	ILF		2	8 Low		
3	2	1	ข้อมูลชื่อผู้	EI		1	10 Low		
4	3	1	ข้อมูลการต	EO		1	8 Low		
5	4	1	ข้อมูลชื่อวี	EI		1	5 Low		
6	5	1	ข้อมูลที่อยู่	EI		1	10 Low		
7	6	1	ข้อมูลบาร์โ	EI		1	12 Low		
8	7	1	ข้อมูลหมาย	EO		1	3 Low		
9	8	1	ข้อมูลผู้ซึ่ง	ILF		3	8 Low		
10	9	1	ข้อมูลกลุ่ม	ILF		4	4 Low		
11	10	1	ข้อมูลสิทธิ์	ILF		4	4 Low		
12	11	1	ข้อมูลน้ำมัน	ILF		3	4 Low		
13	12	1	ข้อมูลการข	ILF		3	5 Low		
14	13	1	ข้อมูลการต	EO		2	10 Low		
15	14	1	ข้อมูลน้ำมัน	ILF		1	12 Low		
16	15	1	ข้อมูลราคา	ILF		4	5 Low		
17	16	1	รายงานการ	EQ		1	16 Low		
18	17	1	รายงานการ	EQ		1	11 Low		
19	18	1	รายงานบริ	EQ		1	12 Low		
20	19	1	รายงานการ	EQ		1	16 Low		
21	20	1	ข้อมูลผู้ซึ่ง	EI		1	8 Low		
22	21	1	ข้อมูลสถานี	ILF		1	7 Low		
23	22	1	ข้อมูลเวลา	ILF		2	5 Low		
24	23	1	ข้อมูลห้วง	ILF		3	3 Low		
25	24	1	ข้อมูลถึง	ILF		5	5 Low		
26	25	1	ข้อมูลชนิด	ILF		6	6 Average		
27	26	1	ข้อมูลโรง	ILF		8	7 Average		

รูปที่ 9 ผลการบันทึกไฟล์ Microsoft office excel ของการคำนวณค่าฟังก์ชันพอยต์



### ประวัติผู้เขียนวิทยานิพนธ์

นายภัทร เวชชีรนนต์วัฒน์ เกิดเมื่อวันที่ 1 กุมภาพันธ์ พ.ศ. 2531 ณ จังหวัดนครศรีธรรมราช วุฒิการศึกษาระดับปริญญาตรี หลักสูตรวิทยาศาสตร์บัณฑิต สาขาวิชาการพัฒนาซอฟต์แวร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย สำเร็จการศึกษาในปี พ.ศ. 2553 และเข้าศึกษาในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาการพัฒนซอฟต์แวร์ด้านธุรกิจ ภาควิชาการสถิติ คณะพาณิชยศาสตร์และการบัญชี จุฬาลงกรณ์มหาวิทยาลัย ในปี พ.ศ. 2556

