

# Hospital Management System: A Case Study on the Wat Huai Pla Kang Hospital Central Warehouse Management System

ระบบบริหารโรงพยาบาล : กรณีศึกษาระบบการบริหารจัดการ

คลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคม

นฤพนธ์ วงศ์ชัย 65021824

พีรพัฒน์ วงศ์ชัย 65021969

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะเทคโนโลยีสารสนเทศและการสื่อสาร

มหาวิทยาลัยพะเยา

ปีการศึกษา 2567

# Hospital Management System: A Case Study on the Wat Huai Pla Kang Hospital Central Warehouse Management System

ระบบบริหารโรงพยาบาล : กรณีศึกษาระบบการบริหารจัดการ

คลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคม

นฤพนธ์ วงศ์ชัย      65021824

พีรพัฒน์ วงศ์ชัย      65021969

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะเทคโนโลยีสารสนเทศและการสื่อสาร

มหาวิทยาลัยพะเยา

ปีการศึกษา 256

# สารบัญ

## หน้า

สารบัญ.....	I
สารบัญตาราง.....	V
สารบัญรูป.....	VI
บทที่ 1 บทนำ .....	1
1.1 ความเป็นมาและความสำคัญของปัญหา .....	1
1.2 วัตถุประสงค์ของการศึกษา.....	2
1.3 สมมติฐานการศึกษา .....	2
1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการทำโครงการ .....	3
1.5 ขอบเขตของโครงการ.....	3
1.5.1 มุมมองเชิงความสามารถของระบบ.....	3
1.5.2 มุมมองเชิงความสามารถของผู้ใช้งานระบบ.....	4
1.6 แผนการดำเนินงาน .....	7
1.7 เครื่องมือและอุปกรณ์ขั้นต่ำที่ใช้ในการสร้าง และรองรับระบบคลังโรงพยาบาล.....	8
1.7.1 เครื่องมือและอุปกรณ์ที่ใช้ในการสร้างระบบคลังโรงพยาบาล.....	8
1.7.2 เครื่องมือและอุปกรณ์ขั้นต่ำที่รองรับระบบบริหารจัดการคลังโรงพยาบาล .....	8
1.8 ประโยชน์ที่คาดว่าจะได้รับ .....	9
1.9 โครงสร้างของปริญญานิพนธ์ .....	9
บทที่ 2 ทฤษฎีพื้นฐานพื้นฐานและงานวิจัยที่เกี่ยวข้อง .....	10
2.1 หลักการบริหารจัดการคลังโรงพยาบาล .....	10
2.1.1 วัตถุประสงค์คลังโรงพยาบาล .....	10

2.1.2	หน้าที่ของคลังโรงพยาบาล .....	11
2.1.3	ประเภทและหลักการจัดเก็บของคลังโรงพยาบาล .....	12
2.1.4	ประโยชน์ของคลังโรงพยาบาล .....	16
2.1.5	การจัดการคลังโรงพยาบาล .....	17
2.1.6	กิจกรรมหลักภายในคลังโรงพยาบาล .....	17
2.1.7	การจัดการของแผนกบริการปลอดเชื้อส่วนกลาง .....	18
2.2	หลักการของบาร์โค้ดและคิวอาร์โค้ด .....	19
2.2.1	บาร์โค้ด .....	20
2.2.2	คิวอาร์โค้ด .....	24
2.2.3	เครื่องพิมพ์บาร์โค้ดและคิวอาร์โค้ด .....	28
2.2.4	เครื่องอ่านบาร์โค้ด .....	28
2.3	หลักการรับความต้องการ .....	33
2.3.1	การรับความต้องการ (Requirement Gathering) .....	33
2.4	หลักการการวิเคราะห์และออกแบบระบบคอมพิวเตอร์ .....	36
2.4.1	การวิเคราะห์ระบบ (System Analysis) .....	36
2.4.2	การออกแบบระบบ (System Design) .....	38
2.4.3	ความสำคัญของการวิเคราะห์และออกแบบระบบ .....	39
2.5	ทฤษฎีด้านการออกแบบหน้าบ้าน .....	40
2.5.1	ทฤษฎีการออกแบบแผนภาพบริบท (Context Diagram) .....	41
2.5.2	ทฤษฎีการออกแบบแผนภาพกระแสข้อมูล (Data Flow Diagram : DFD) .....	43
2.5.3	แผนภาพต้นไม้ (Tree Diagram) .....	51
2.5.4	ทฤษฎีการออกแบบส่วนติดต่อประสานงานกับผู้ใช้ (User Experience and User Interface Design หรือ UX/UI Design) .....	51

2.6 ทฤษฎีด้านการออกแบบหลังบ้าน .....	53
2.6.1 ทฤษฎีการออกแบบองค์ประกอบของแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล (Data Relationship Diagrams : E-R Diagrams).....	57
2.6.2 ทฤษฎีและความรู้พื้นฐานเกี่ยวกับฐานข้อมูล .....	58
2.6.3 ฐานข้อมูลในการพัฒนาระบบ .....	59
2.6.4 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล .....	61
2.7 ทฤษฎีการเชื่อมต่อระหว่างส่วนหน้าบ้านและหลังบ้าน.....	66
2.7.1 หลักการเชื่อมต่อระหว่างส่วนหน้าบ้านและหลังบ้าน .....	66
2.7.2 ความรู้พื้นฐานเกี่ยวกับเรสตเอพียู (Representational State Transfer Application Programming Interface : REST API).....	68
2.8 ภาษาและเฟรมเวิร์คในการพัฒนาโปรแกรม .....	71
2.8.1 ภาษาและเฟรมเวิร์คในการทำหน้าบ้าน .....	72
2.8.2 ภาษาและเฟรมเวิร์คที่เลือกใช้ในการทำหลังบ้าน .....	78
2.9 งานวิจัยที่เกี่ยวข้อง .....	80
2.10 หลักการทดสอบระบบ (System Testing).....	83
2.11 บทสรุป .....	86
 บทที่ 3 การวิเคราะห์และออกแบบระบบ .....	87
3.1 วิเคราะห์ความต้องการของระบบคลังโรงพยาบาล .....	87
3.2 ภาพรวมของระบบคลังโรงพยาบาล .....	89
3.3 แผนภาพบริบทระบบคลังโรงพยาบาล .....	89
3.4 การออกแบบแผนภาพกระแสข้อมูล .....	89
3.5 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล .....	91
3.6 การออกแบบส่วนติดต่อประสานงานกับผู้ใช้งาน .....	91

3.7 บทสรุป.....	113
เอกสารอ้างอิง .....	108

## สารบัญตาราง

ตารางที่	หน้า
1.1 ขั้นตอนและแผนการดำเนินงานระบบบริหารจัดการโรงพยาบาล : ระบบบริหารจัดการคลังหลัก ของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคม .....	7
2.1 ข้อมูลเพิ่มเติมเกี่ยวกับวิธีการพิมพ์บาร์โค้ด.....	23
2.2 สัญลักษณ์ต่างๆ ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล.....	41
2.3 สัญลักษณ์ต่างๆ ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล.....	42
2.4 สัญลักษณ์ต่าง ๆ ที่ใช้ในการเขียนแผนภาพการไหลของข้อมูล.....	44
2.5 กฎการใช้สัญลักษณ์ในการเขียนแผนภาพ .....	48
2.6 สัญลักษณ์ที่ใช้ในแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล .....	65
3.1 แสดงรายชื่อกระบวนการของระบบในแผนภาพกระแสข้อมูลระดับที่ 1 .....	97
3.2 แสดงกระบวนการที่ 1 การเข้า/ออกระบบ.....	97
3.3 แสดงกระบวนการที่ 2 ตรวจสอบยอดคลังคง .....	98
3.4 แสดงกระบวนการที่ 3 การสั่งซื้อ .....	99
3.5 แสดงกระบวนการที่ 4 การนำเข้า .....	99
3.6 แสดงกระบวนการที่ 5 การเบิก.....	100
3.7 แสดงกระบวนการที่ 6 การยืม/การคืน .....	101
3.8 แสดงกระบวนการที่ 7 การแจ้งเตือนต่าง ๆ .....	102
3.9 แสดงกระบวนการที่ 8 การออกรายงานต่าง ๆ.....	102
3.10 แสดงกระบวนการที่ 9 การตั้งค่าต่าง ๆ แก่ไขข้อมูลต่าง ๆ.....	103

## สารบัญรูป

รูปที่	หน้า
2.1 รหัสสินค้า.....	20
2.2 หมายเลขประจำตัว EAN.UPC 13.....	20
2.3 CODE 39 .....	21
2.4 CODE128 .....	21
2.5 PDF417 .....	22
2.6 Data Matrix .....	22
2.7 QR Code (Quick Response) .....	23
2.8 QR Code Model 1 .....	25
2.9 Micro QR Code.....	25
2.10 IQR Code.....	26
2.11 SQRC.....	26
2.12 Frame QR.....	27
2.13 KS1689 Barcode .....	29
2.14 KS 1689 Barcode .....	30
2.15 CCD Scanner .....	30
2.16 Laser Scanner .....	31
2.17 Omnidirectional Scanner.....	32
2.18 Imager Scanner.....	32
2.19 ตัวอย่างการใช้สัญลักษณ์การประมวลผล (Process).....	45
2.20 ตัวอย่างการใช้สัญลักษณ์กระแสข้อมูล.....	46
2.21 ตัวอย่างการใช้สัญลักษณ์แหล่งที่เก็บข้อมูล .....	47
2.22 ตัวอย่างการใช้สัญลักษณ์สิ่งที่อยู่ภายนอก.....	47
2.23 สัญลักษณ์ของโพสต์เกรสคิวแอล .....	61
2.24 ตัวอย่างการใส่ชื่อเอ็นทีดี .....	62
2.25 ภาพแสดงความสัมพันธ์แบบหนึ่งต่อหนึ่ง .....	62
2.26 ภาพแสดงความสัมพันธ์แบบหนึ่งต่อหลาย.....	63
2.27 ภาพแสดงความสัมพันธ์แบบหลายต่อหลาย.....	63
2.28 ตัวอย่างการใส่แอททริบิวต์ .....	64



## สารบัญรูป(ต่อ)

รูปที่	หน้า
2.29 สัญลักษณ์ของเอชทีเอ็มแอลไฟท์ .....	72
2.30 สัญลักษณ์ของซีเอสเอส .....	74
2.31 สัญลักษณ์ของจาวาสคริปต์.....	75
2.32 สัญลักษณ์ของโนดเจเอส.....	78
3.1 แผนภาพต้นไม้ ของระบบการจัดการคลังโรงพยาบาล.....	92
3.2 แผนภาพต้นไม้ ของระบบการจัดการคลังโรงพยาบาล(ต่อ).....	93
3.3 แผนภาพบริบทของระบบการจัดการคลังโรงพยาบาล .....	94
3.4 แผนภาพกระแสข้อมูลระดับที่ 1 ของระบบการจัดการคลังโรงพยาบาล.....	95
3.5 แผนภาพกระแสข้อมูลระดับที่ 1 ของระบบการจัดการคลังโรงพยาบาล(ต่อ).....	96
3.6 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล ของระบบการจัดการคลังโรงพยาบาล .....	105
3.7 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล ของระบบการจัดการคลังโรงพยาบาล(ต่อ).....	106
3.8 หน้าเข้าสู่ระบบ .....	107
3.9 หน้าหลักแสดงรายละเอียดต่าง ๆ .....	107
3.10 หน้าตรวจสอบยอดคงคลัง .....	108
3.11 หน้าตรวจสอบรายการเบิก ยืม .....	108
3.12 หน้าตรวจสอบรายละเอียดคำร้องขอ .....	109
3.13 หน้าจัดการแก้ไขข้อมูล .....	109
3.14 หน้าเพิ่มรายการ.....	110
3.15 หน้าแก้ไขรายการ .....	110
3.16 หน้าตรวจสอบประวัติการนำเข้า นำออก .....	111
3.17 หน้าออกรายงาน .....	111
3.18 หน้าทำรายการเบิก ยืม.....	112
3.19 หน้าทำรายการเบิก.....	112
3.20 หน้าทำรายการยืม .....	113
3.21 หน้าตรวจสอบรายการที่เลือก.....	113

## บทที่ 1

### บทนำ

#### 1.1 ความเป็นมาและความสำคัญของปัญหา

ระบบคลังโรงพยาบาลถือเป็นหัวใจสำคัญในการบริหารจัดการทรัพยากรทางการแพทย์ที่ซับซ้อน และมีต้นทุนสูง ช่วยให้สามารถจัดเก็บเวชภัณฑ์และอุปกรณ์ทางการแพทย์ได้อย่างเป็นระเบียบ มีประสิทธิภาพและพร้อมใช้งานในทุกสถานการณ์ ระบบนี้ยังช่วยลดความสูญเสียจากการจัดเก็บไม่เหมาะสม เช่น ยา เวชภัณฑ์หมดอายุ หรือการเก็บสต็อกมากเกินไปจนเกิดความจำเป็น ช่วยให้โรงพยาบาลประหยัดงบประมาณและลดค่าใช้จ่ายที่ไม่จำเป็น นอกจากนี้ยังสนับสนุนการบริหารจัดการที่โปร่งใส และสามารถติดตามได้ ลดความผิดพลาดที่อาจเกิดจากการจัดการแบบเดิม ระบบคลังในปัจจุบันยังสามารถผสมผสานเทคโนโลยีขั้นสูง เช่น การใช้ซอฟต์แวร์จัดการข้อมูล การแจ้งเตือนแบบอัตโนมัติ และการเชื่อมต่อกับระบบฐานข้อมูลแบบเรียลไทม์ ทำให้กระบวนการจัดการมีประสิทธิภาพมากยิ่งขึ้น อีกทั้งยังช่วยสนับสนุนการวางแผนเชิงกลยุทธ์ในการจัดซื้อหรือจัดเก็บทรัพยากร ช่วยลดเวลาที่ใช้ในการตรวจสอบหรือปรับปรุงข้อมูล และเพิ่มความแม่นยำในการตัดสินใจเกี่ยวกับการจัดการทรัพยากรเหล่านี้ได้อย่างเหมาะสมและยั่งยืน

โรงพยาบาลวัดห้วยปลากั้งเป็นโรงพยาบาลเอกชนขนาดเล็กในจังหวัดเชียงรายที่ดำเนินงานโดยไม่หวังผลกำไร ด้วยเป้าหมายหลักในการส่งเสริมคุณภาพชีวิตและช่วยเหลือผู้ยากไร้และผู้ด้อยโอกาส ระบบคลังโรงพยาบาลเป็นหัวใจสำคัญของการบริหารจัดการทรัพยากรทางการแพทย์ให้เกิดประโยชน์สูงสุด เพื่อสนับสนุนการรักษาและดูแลผู้ป่วยอย่างต่อเนื่องและมีประสิทธิภาพ ด้วยการใช้เทคโนโลยีที่ทันสมัย ระบบนี้ช่วยให้สามารถควบคุมทรัพยากรได้อย่างแม่นยำ สามารถเบิก ยืม คืน ตรวจสอบยอดคงคลังแบบเรียลไทม์ การแจ้งเตือนเกี่ยวกับงานคลัง และการออกรายงานที่สำคัญต่อระบบคลังสนับสนุนการวางแผนการจัดซื้ออย่างเหมาะสม ลดปัญหาของหมดอายุหรือขาดแคลน เพิ่มความคล่องตัวในการยืมคืนอุปกรณ์การดำเนินงานที่มีประสิทธิภาพของระบบคลังโรงพยาบาลไม่เพียงช่วยประหยัดงบประมาณ แต่ยังเสริมสร้างความมั่นใจในคุณภาพการให้บริการทางการแพทย์ สร้างผลกระทบเชิงบวกต่อทั้งผู้ป่วยและบุคลากรทางการแพทย์

จากปัญหาข้างต้นที่กล่าวมาทางคณะผู้จัดทำได้เล็งเห็นถึงความสำคัญในเรื่องการจัดการคลังโดยมีแนวคิดที่จะสร้างระบบสนับสนุนการบริหารจัดการคลังโรงพยาบาลในรูปแบบของเว็บแอปพลิเคชัน (Web Application) พร้อมทั้งฟังก์ชันการทำงานที่จำเป็นต่อการจัดการคลัง โดยคาดหวังว่าระบบจะสามารถอำนวยความสะดวกสบายให้กับผู้ใช้งาน ลดความซ้ำซ้อน ลดความผิดพลาดจากข้อมูลปริมาณมาก ลดเวลาในการจัดการคลัง เพิ่มความสามารถในการเข้าถึงข้อมูลต่าง ๆ ภายในระบบได้อย่างมีประสิทธิภาพ

## 1.2 วัตถุประสงค์ของการศึกษา

1.2.1 เพื่อออกแบบระบบบริหารจัดการโรงพยาบาล: กรณีศึกษาระบบการบริหารจัดการคลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคมในรูปแบบเว็บแอปพลิเคชัน

1.2.2 เพื่อพัฒนาระบบบริหารจัดการโรงพยาบาล: กรณีศึกษาระบบการบริหารจัดการคลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคมในรูปแบบเว็บแอปพลิเคชันในรูปแบบเว็บแอปพลิเคชัน

1.2.3 เพื่อทดสอบการทำงานของระบบบริหารจัดการโรงพยาบาล: กรณีศึกษาระบบการบริหารจัดการคลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคมในรูปแบบเว็บแอปพลิเคชัน

## 1.3 สมมติฐานการศึกษา

การสร้างและออกแบบระบบบริหารจัดการคลังโรงพยาบาลในรูปแบบของเว็บแอปพลิเคชัน (Web Application) จำเป็นต้องมีความรู้ ความเข้าใจทางด้านวิศวกรรมคอมพิวเตอร์มาประยุกต์ใช้ร่วมกับองค์ความรู้ในการบริหารจัดการคลังโรงพยาบาล หลักการออกแบบแผนภาพบริบท (Context Diagram) เพื่อสร้างภาพรวมของระบบ โดยแสดงความสัมพันธ์ระหว่างระบบกับสิ่งแวดล้อมภายนอก หลักการออกแบบแผนภาพการไหลข้อมูล (Data Flow Diagram : DFD ) เพื่ออธิบายขั้นตอนและกระบวนการทำงานของระบบ จะนำเสนอในรูปแบบแผนภาพส่วนประกอบและความสัมพันธ์ (ER Diagram) และหลักการออกแบบส่วนติดต่อประสานงานต่อกับผู้ใช้งาน (Graphic User Interface) สำหรับการพัฒนาในระบบในการติดกับผู้ใช้งานแบ่งออกเป็นส่วนหน้าบ้าน (Front-end) โดยใช้เอชทีเอ็มแอล เอสเอส หลักการทำงานของรีแอค เฟรมเวิร์ค (React Framework) ร่วมกับภาษาจาวาสคริปต์ (JavaScript) ส่วนเชื่อมต่อประสานงานระหว่างหน้าบ้านและหลังบ้านจะถูกพัฒนาด้วยเรสต์เอพียู (REST API) ส่วนหลังบ้าน (Back-end) จะถูกพัฒนาด้วยโพสทรีสคิวเอล (PostgreSQL) เพื่อใช้เก็บข้อมูลต่าง ๆ ในระบบ สำหรับการทดสอบโพสต์แมน โดยคาดหวังว่าระบบจะสามารถลดปัญหาและความซ้ำซ้อนของข้อมูลคลัง จากการดำเนินงานแบบแมนนวล (Manual) พร้อมทั้งเพิ่มฟังก์ชันการทำงานที่จำเป็นกับระบบคลังโรงพยาบาล ในรูปแบบของเว็บแอปพลิเคชัน

แอปพลิเคชัน การค้นหารายการของในระบบ ตรวจสอบยอดคงคลัง การเบิก การยืม การคืน การจัดซื้อจัดจ้างผ่านระบบได้ การแจ้งเตือนต่าง ๆ การตั้งค่าแก้ไขข้อมูลของระบบ การออกสถิติและรายงานของคลังโรงพยาบาลได้ ดังนั้นโครงการนี้จะทำการศึกษาและออกแบบระบบบริหารจัดการคลังโรงพยาบาล ที่มีมาตรฐานและประสิทธิภาพ สะดวกสบายแก่ผู้ใช้งาน มีเครื่องมือในระบบคลังที่สนับสนุนการทำงานของผูู้้งานมากยิ่งขึ้น

## 1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการทำโครงการ

การสร้างและออกแบบเว็บแอปพลิเคชันระบบบริหารจัดการคลังโรงพยาบาลได้อย่างถูกต้อง มีประสิทธิภาพและตรงตามมาตรฐานของโรงพยาบาล จำเป็นต้องมีความรู้ความเข้าใจทฤษฎี หลักการ และองค์ความรู้พื้นฐานเฉพาะทางที่เกี่ยวข้องต่าง ๆ ดังนี้

### 1.4.1 หลักการพื้นฐานสำหรับการจัดการคลังโรงพยาบาล

### 1.4.2 หลักการวิเคราะห์และออกแบบระบบ

- หลักการออกแบบแผนภาพบริบท
- หลักการออกแบบแผนภาพกระแสข้อมูล
- หลักการออกแบบแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล
- หลักการออกแบบส่วนติดต่อกับผู้ใช้งาน

### 1.4.3 หลักการพัฒนาระบบ

- ส่วนของหน้าบ้าน
- ส่วนของหลังบ้าน
- ส่วนเชื่อมต่อประสานงานหน้าบ้านและหลังบ้าน
- ส่วนของการทดสอบระบบ

## 1.5 ขอบเขตของโครงการ

โครงการนี้มุ่งเน้นการพัฒนาระบบบริหารจัดการคลังโรงพยาบาลที่ครอบคลุมการเบิก การยืม-คืนจัดการข้อมูลเกี่ยวกับการนำเข้านำออก ยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์รวมถึงของใช้ทั่วไปในระบบคลัง การดำเนินกระบวนการสั่งซื้อ การแจ้งเตือนที่สำคัญและการสร้างรายงาน สามารถแบ่งตามมุมมองออกเป็น 2 หัวข้อหลัก ๆ คือ มุมมองเชิงความสามารถของระบบและมุมมองเชิงความสามารถของผู้ใช้งาน โดยรายละเอียดต่าง ๆ สามารถอธิบายได้ดังนี้

### 1.5.1 มุมมองเชิงความสามารถของระบบ

- 1.5.1.1 ระบบสามารถเพิ่ม ลบ แก้ไข ตรวจสอบข้อมูลที่เกี่ยวข้องกับยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์และของใช้ทั่วไป
- 1.5.1.2 ระบบสามารถตรวจสอบยอดคงคลังแบบเรียลไทม์ พร้อมแสดงรายละเอียดที่สำคัญ เช่น วันหมดอายุ ประเภท และจำนวนคงเหลือ
- 1.5.1.3 ระบบสามารถเบิก-ยืมคืนยา เวชภัณฑ์ ครุภัณฑ์ และอุปกรณ์ทางการแพทย์ผ่านระบบออนไลน์ ช่วยให้การดำเนินงานสะดวกและมีประสิทธิภาพมากขึ้น
- 1.5.1.4 ระบบสามารถแจ้งเตือนเหตุการณ์สำคัญเกี่ยวกับคลังเวชภัณฑ์ เช่น ยอดคงคลังต่ำกว่าระดับที่กำหนด การนำเข้า การนำออก การสั่งซื้อ และสถานะคำสั่งซื้อ

- 1.5.1.5 ระบบสามารถกำหนดและจัดการสิทธิ์การเข้าถึงของผู้ใช้งานตามบทบาทที่ได้รับ รวมถึงการตั้งค่าการแจ้งเตือนตามสิทธิ์ที่กำหนด
- 1.5.1.6 ระบบสามารถสร้างและออกรายงานที่เกี่ยวข้องกับการบริหารจัดการคลังเวชภัณฑ์ เช่น รายงานยอดคงคลัง รายงานการนำเข้า และรายงานการนำออก โดยรองรับการส่งออกรายงานในหลายรูปแบบ เช่น PDF, Excel และ CSV
- 1.5.1.7 ระบบสามารถบันทึกและจัดเก็บประวัติการดำเนินงานภายในระบบคลัง เพื่อรองรับการตรวจสอบย้อนหลังและเพิ่มความปลอดภัยของข้อมูล
- 1.5.1.8 ระบบสามารถจัดการกระบวนการสั่งซื้อได้อย่างมีประสิทธิภาพ เช่น การสร้างคำสั่งซื้ออัตโนมัติเมื่อยอดคงคลังต่ำกว่าที่กำหนด การออกใบเสนอราคา และการติดตามสถานะคำสั่งซื้อ
- 1.5.1.9 ระบบสามารถจัดการการนำเข้าและการนำออกของสินค้าได้อย่างมีประสิทธิภาพ เช่น การบันทึกข้อมูลการนำเข้าและการนำออก การตรวจสอบความถูกต้องของข้อมูล และการอัปเดตยอดคงคลังโดยอัตโนมัติ

## 1.5.2 มุมมองเชิงความสามารถของผู้ใช้งานระบบ

### 1.5.2.1 ผู้ดูแลระบบ

- สามารถกำหนดบทบาทและสิทธิ์การเข้าถึงของผู้ใช้งานในระบบ
- สามารถจัดการข้อมูลผู้ใช้งาน เช่น เพิ่ม แก้ไข ปรับปรุง และลบข้อมูลบัญชีผู้ใช้
- สามารถตั้งค่าและปรับแต่งระบบ เช่น การแจ้งเตือน การสำรองข้อมูล
- สามารถตรวจสอบและบริหารจัดการประวัติการดำเนินงานภายในระบบ เพื่อรองรับการตรวจสอบย้อนหลัง
- สามารถกำกับดูแลกระบวนการทำงานภายในคลัง เช่น การอนุมัติคำขอเบิก-ยืม การอนุมัติคำสั่งซื้อ และการปรับปรุงข้อมูลสินค้าคงคลัง
- สามารถจัดทำ วิเคราะห์ และส่งออกรายงานที่เกี่ยวข้องกับการบริหารจัดการคลัง ในรูปแบบที่หลากหลาย เช่น PDF, Excel และ CSV
- สามารถควบคุมความปลอดภัยของข้อมูลภายในระบบ รวมถึงการตรวจสอบและบันทึกกิจกรรมของผู้ใช้งาน

### 1.5.2.2 เจ้าหน้าที่จัดการคลัง

- สามารถเพิ่ม แก้ไข และอัปเดตข้อมูลสินค้าคงคลัง เช่น ยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์ และของใช้ทั่วไป
- สามารถตรวจสอบยอดคงคลังแบบเรียลไทม์ พร้อมแสดงรายละเอียดสำคัญ เช่น วันหมดอายุ ประเภท และจำนวนคงเหลือ
- สามารถรับสินค้า ตรวจสอบคุณภาพ ปริมาณ และความถูกต้องของสินค้าก่อนอัปเดตเข้าสู่ระบบคลัง

- สามารถดำเนินการตรวจนับสต็อกสินค้า พร้อมจัดทำรายงานการตรวจนับสต็อกเพื่อใช้ในการวิเคราะห์และวางแผน
- สามารถจัดการกระบวนการนำเข้าและนำออกของสินค้า พร้อมตรวจสอบความถูกต้องของข้อมูลก่อนบันทึกลงระบบ
- สามารถดำเนินการเบิก-ยืม-คืนสินค้าได้อย่างเป็นระบบ รวมถึงติดตามและอนุมัติค่าของต่าง ๆ ตามเงื่อนไขที่กำหนด
- สามารถแจ้งเตือนและจัดการสถานการณ์เร่งด่วน เช่น ยอดคงคลังต่ำกว่าระดับที่กำหนด หรือสินค้าหมดอายุ
- สามารถสร้างและส่งออกรายงานสินค้าคงคลังเพื่อใช้ในการบริหารจัดการและวางแผนในอนาคต

#### 1.5.2.3 แผนกจัดซื้อ

- สามารถดำเนินการจัดซื้อสินค้าเมื่อยอดคงคลังต่ำกว่าระดับที่กำหนด โดยสามารถสร้างคำสั่งซื้ออัตโนมัติหรือตั้งค่าให้มีการตรวจสอบจากคณะกรรมการสามารถรับสินค้า ตรวจสอบคุณภาพ ปริมาณ และความถูกต้องของสินค้าก่อนอัปเดตเข้าสู่ระบบคลัง
- สามารถจัดทำรายงานการจัดซื้อ เช่น รายงานคำสั่งซื้อ รายงานสถานะการสั่งซื้อ และรายงานประวัติการซื้อสินค้า
- สามารถติดตามและตรวจสอบกระบวนการชำระเงิน รวมถึงจัดการเอกสารทางการเงินที่เกี่ยวข้องกับการสั่งซื้อสินค้าผ่านระบบ

#### 1.5.2.4 แผนกเภสัชกรรม

- สามารถตรวจสอบยอดคงคลังยาแบบเรียลไทม์และดำเนินการขอเบิกยาและเวชภัณฑ์ผ่านระบบได้อย่างเป็นระบบ พร้อมตรวจสอบความถูกต้องของข้อมูลก่อนดำเนินการ
- สามารถร้องขอการสั่งซื้อยาได้ เมื่อพบว่ายามีปริมาณต่ำกว่าที่กำหนดหรือกำลังจะหมด พร้อมส่งต่อคำขอไปยังแผนกที่เกี่ยวข้อง
- สามารถติดตามสถานะคำขอเบิก
- สามารถจัดทำรายงานที่เกี่ยวข้องกับยา เช่น รายงานปริมาณคงเหลือ รายงานการใช้ยา และรายงานยาหมดอายุ

#### 1.5.2.5 ผู้ใช้งานทั่วไป

- สามารถดำเนินการขอเบิกสินค้าผ่านระบบ โดยระบุรายละเอียดที่จำเป็น เช่น จำนวนที่ต้องการ เหตุผลในการเบิก และวันที่ต้องการใช้
- สามารถติดตามสถานะคำขอเบิกสินค้าแบบเรียลไทม์ เช่น รอดำเนินการ อนุมัติ หรือถูกปฏิเสธ

- สามารถดูประวัติคำขอเบิกสินค้าและตรวจสอบรายละเอียดการทำรายการย้อนหลัง เพื่อช่วยให้สามารถวางแผนการใช้งานได้อย่างมีประสิทธิภาพ





## 1.7 เครื่องมือและอุปกรณ์ขั้นต่ำที่ใช้ในการสร้าง และรองรับระบบคลังโรงพยาบาล

ในกระบวนการพัฒนาระบบคลังโรงพยาบาล มีความจำเป็นต้องเลือกเครื่องมือและอุปกรณ์ที่เหมาะสมเพื่อให้การพัฒนาระบบมีประสิทธิภาพ รวมถึงการรองรับการทำงานของระบบในระยะยาว เครื่องมือและอุปกรณ์เหล่านี้ต้องตอบสนองต่อความต้องการทั้งในด้านการเขียนโปรแกรม การทดสอบ และการจัดการข้อมูล ตลอดจนช่วยให้ระบบสามารถทำงานได้อย่างราบรื่นและมีเสถียรภาพ

### 1.7.1 เครื่องมือและอุปกรณ์ที่ใช้ในการสร้างระบบคลังโรงพยาบาล

- ซอฟต์แวร์ (Software)
  - Visual Studio Code: สำหรับการเขียนและแก้ไขโค้ด
  - Node.js: สำหรับการพัฒนา Backend และจัดการ REST API
  - PostgreSQL: สำหรับจัดการฐานข้อมูล
  - Postman: สำหรับการทดสอบ API และการเชื่อมต่อ
- ฮาร์ดแวร์ (Hardware)
  - CPU: Intel Core i5 หรือ AMD Ryzen 5 ขึ้นไป
  - RAM: 16GB (เพื่อรองรับการทำงานหลายโปรแกรมพร้อมกัน)
  - SSD: 512GB (สำหรับประสิทธิภาพการอ่านเขียนข้อมูลที่เร็ว)
  - GPU: ไม่จำเป็นสำหรับงานพัฒนาเว็บทั่วไป
  - จอภาพ (LED): จอภาพ Full HD ขนาด 21 นิ้ว

### 1.7.2 เครื่องมือและอุปกรณ์ขั้นต่ำที่รองรับระบบบริหารจัดการคลังโรงพยาบาล

- ซอฟต์แวร์ (Software)
  - ระบบปฏิบัติการ: Microsoft Windows 10 หรือสูงกว่า
  - เว็บเบราว์เซอร์: Google Chrome, Microsoft Edge หรือ Firefox
  - Adobe Acrobat Reader: สำหรับเปิดเอกสาร PDF
- ฮาร์ดแวร์ (Hardware)
  - CPU: Intel Core i3 หรือ AMD Ryzen 3 ขึ้นไป
  - RAM: 8GB
  - SSD: 256GB
  - จอภาพ (LED): จอภาพ Full HD ขนาด 19 นิ้ว

## 1.8 ประโยชน์ที่คาดว่าจะได้รับ

1.8.1 ผลออกแบบระบบบริหารจัดการโรงพยาบาล: กรณีศึกษาระบบการบริหารจัดการคลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคม

1.8.2 ผลการพัฒนาระบบบริหารจัดการโรงพยาบาล: กรณีศึกษาระบบการบริหารจัดการคลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคม

1.8.3 ผลการทดสอบการทำงานของระบบบริหารจัดการโรงพยาบาล: กรณีศึกษาระบบการบริหารจัดการคลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคม

## 1.9 โครงสร้างของปฏิญานิพนธ์

ปฏิญานิพนธ์เล่มนี้แบ่งออกเป็น 5 บท แต่ละบทมีองค์ประกอบด้วยเนื้อหาดังต่อไปนี้

บทที่ 1 ความเป็นมาและความสำคัญของปัญหา วัตถุประสงค์ของการศึกษา ทฤษฎีหรือแนวคิดที่ใช้ในการทำโครงงาน ขอบเขตของโครงงาน ขั้นตอนการศึกษาค้นคว้า อุปกรณ์และเครื่องมือที่ใช้ในการสร้างระบบการจัดการคลัง ทรัพยากรขั้นต่ำที่รองรับระบบการบริหารจัดการคลังโรงพยาบาล ประโยชน์ที่คาดว่าจะได้รับและโครงสร้างของปฏิญานิพนธ์

บทที่ 2 ทฤษฎีพื้นฐานและงานวิจัยที่เกี่ยวข้อง กล่าวถึงทฤษฎีและหลักการพื้นฐานที่เกี่ยวข้องกับการจัดการคลังโรงพยาบาล แนวทางการจัดเก็บพัสดุและเวชภัณฑ์ทางการแพทย์แต่ละประเภท รวมถึงองค์ความรู้พื้นฐานเฉพาะทางในสาขาวิศวกรรมคอมพิวเตอร์ อันได้แก่ การพัฒนาระบบ การออกแบบระบบ และการทดสอบระบบ

บทที่ 3 การวิเคราะห์และออกแบบฐานข้อมูล เริ่มจากการสำรวจความต้องการของระบบจากผู้ใช้งานคลังในโรงพยาบาล ภาพรวมของระบบ ผู้มีส่วนเกี่ยวข้องกับระบบ ขอบเขตของผู้ใช้งานแต่ละแผนก ออกแบบฐานข้อมูลการบริหารจัดการคลัง โดยใช้แผนภาพบริบท การเขียนแผนภาพกระแสนการไหลของระบบบริหารจัดการคลังโรงพยาบาล แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล การออกแบบส่วนติดต่อกับผู้ใช้งาน และการออกแบบประสบการณ์การใช้งานของผู้ใช้

บทที่ 4 กล่าวถึงผลของการออกแบบและสร้างส่วนติดต่อผู้ใช้งาน ตัวอย่างการทดสอบความถูกต้องของการดำเนินงาน วิเคราะห์และอภิปรายผลการทดสอบของระบบ

บทที่ 5 กล่าวถึงสรุปและวิเคราะห์ผลการดำเนินงาน ปัญหาและอุปกรณ์ และแนวทางในการพัฒนาต่อและข้อเสนอแนะ

## บทที่ 2

### ทฤษฎีพื้นฐานพื้นฐานและงานวิจัยที่เกี่ยวข้อง

ในส่วนของบทที่ 2 จะกล่าวถึงทฤษฎีบทและหลักการพื้นฐานที่เกี่ยวข้องกับระบบคลังสำหรับโรงพยาบาล การจัดเก็บและการกระจายของอย่างมีประสิทธิภาพ เพื่อให้เกิดความรู้ความเข้าใจเพื่อนำมาสร้างและพัฒนาระบบคลังโรงพยาบาล โดยสามารถแบ่งเนื้อหาที่จะนำเสนอเป็น 10 ส่วนหลัก ได้แก่ หลักการบริหารจัดการคลังโรงพยาบาล หลักการของบาร์โค้ดและคิวอาร์โค้ด หลักการรับความต้องการ หลักการวิเคราะห์และออกแบบระบบคอมพิวเตอร์ ทฤษฎีด้านการออกแบบหน้าบ้าน ทฤษฎีด้านการออกแบบหลังบ้าน ทฤษฎีด้านการเชื่อมต่อหน้าบ้านและหลังบ้าน ภาษาและเฟรมเวิร์คที่ใช้ในการเขียนโปรแกรม งานวิจัยที่เกี่ยวข้อง และบทสรุปส่งท้ายบท

#### 2.1 หลักการบริหารจัดการคลังโรงพยาบาล

การบริหารจัดการคลังกลางของโรงพยาบาลเป็นหัวใจสำคัญที่ทำให้การดำเนินงานด้านบริการสุขภาพมีประสิทธิภาพมากยิ่งขึ้น คลังกลางโรงพยาบาลมีหน้าที่ในการจัดเก็บยา เวชภัณฑ์ ครุภัณฑ์ และอุปกรณ์ทางการแพทย์ รวมถึงการนำเข้าสินค้า การเบิกจ่าย การยืม การคืน การเก็บรักษาควบคุมคุณภาพ ตรวจสอบติดตามยอดคงคลัง และการควบคุมปริมาณคลังให้เพียงพอต่อการใช้งาน เพื่อตอบสนองความต้องการของบุคลากรทางการแพทย์และผู้ป่วยที่มาใช้บริการอย่างถูกต้อง

##### 2.1.1 วัตถุประสงค์คลังโรงพยาบาล

คลังโรงพยาบาลมีบทบาทสำคัญต่อการดำเนินงานของโรงพยาบาล โดยช่วยให้มั่นใจว่าผู้ป่วยจะได้รับการรักษาที่มีคุณภาพ ปลอดภัย และมีประสิทธิภาพ ทรัพยากรของโรงพยาบาลถูกบริหารจัดการและสนับสนุนงานอื่น ๆ อย่างแม่นยำเพื่อให้เกิดประโยชน์สูงสุด ตามรายละเอียดดังนี้

- เพื่อให้มั่นใจว่ายา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์ที่จำเป็นในโรงพยาบาลมีพร้อมใช้งานและไม่ขาดแคลนเมื่อมีความต้องการ
- เพื่อลดค่าใช้จ่ายที่ไม่จำเป็นผ่านการจัดการสต็อกที่มีประสิทธิภาพ เช่น การซื้อในปริมาณที่เหมาะสมและการหมุนเวียนยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์อย่างมีประสิทธิภาพ
- การจัดการคลังยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์เป็นไปอย่างมีระเบียบเพื่อหลีกเลี่ยงการสูญเสียหรือเสื่อมสภาพของสินค้า เช่น การหมดอายุของยาและวัสดุทางการแพทย์
- การลดระยะเวลาในการจัดหา ยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์และการกระจายวัสดุเพื่อให้การดำเนินงานในโรงพยาบาลเป็นไปอย่างราบรื่นและมีประสิทธิภาพ

- การปฏิบัติตามกฎหมายและมาตรฐานที่เกี่ยวข้องในการเก็บรักษาและการจัดการยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์ เพื่อให้การดำเนินการเป็นไปตามข้อบังคับ และปลอดภัย

### 2.1.2 หน้าที่ของคลังโรงพยาบาล

หน้าที่หลักของคลังโรงพยาบาลคือ การจัดการคลังกลาง สนับสนุนการรักษาพยาบาล รักษาความปลอดภัย และพัฒนาระบบการทำงาน เพื่อให้มั่นใจว่ามีสินค้าคงคลังเพียงพอต่อความต้องการของผู้ป่วย และสามารถนำไปใช้รักษาผู้ป่วยได้อย่างมีประสิทธิภาพและปลอดภัย

#### 1. จัดการคลังกลาง

- การนำเข้าคลัง ตรวจสอบสภาพ บันทึกรายการยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์
- เก็บรักษาตามประเภท อุณหภูมิ และความชื้นที่เหมาะสม
- ควบคุมยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์ให้อยู่ในระดับที่เหมาะสม ไม่มากหรือน้อยจนเกินไป
- เบิกจ่ายยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์ให้กับแผนกต่าง ๆ ภายในโรงพยาบาล
- ตรวจสอบยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์เป็นประจำ ป้องกันการหมดอายุ หรือเสื่อมสภาพ

#### 2. สนับสนุนการรักษาพยาบาล

- จัดเตรียมยาและเวชภัณฑ์ให้พร้อมสำหรับการรักษาพยาบาล
- จัดส่งยาและเวชภัณฑ์ให้กับผู้ป่วยตามใบสั่งยาของแพทย์
- ให้ข้อมูลเกี่ยวกับยาและเวชภัณฑ์แก่ผู้ป่วยและบุคลากรทางการแพทย์
- เก็บข้อมูลสถิติการใช้ยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์

#### 3. รักษาความปลอดภัย

- ควบคุมการเข้าออกคลังยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์
- ตรวจสอบยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์ก่อนนำเข้าคลังและก่อนเบิกจ่าย
- ป้องกันการสูญหาย ขโมย หรือการนำไปใช้ผิดประเภท

#### 4. หน้าที่อื่น ๆ เพิ่มเติม

- จัดซื้อจัดหา ยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์
- จัดทำเอกสารต่าง ๆ เกี่ยวกับการจัดการคลังกลาง
- ตรวจสอบและบำรุงรักษาอุปกรณ์ต่าง ๆ ภายในคลังกลาง
- รายงานข้อมูลการจัดการคลังกลางให้กับผู้บริหาร

### 2.1.3 ประเภทและหลักการจัดเก็บของคลังโรงพยาบาล

การจัดการคลังในโรงพยาบาลถือเป็นส่วนสำคัญที่สนับสนุนการดำเนินงานในทุกด้าน ทั้งในแง่ของการรักษาพยาบาล การบริหารทรัพยากร และการเตรียมพร้อมสำหรับสถานการณ์ฉุกเฉิน คลังในโรงพยาบาลมีบทบาทสำคัญในการจัดเก็บและกระจายวัสดุ อุปกรณ์ และยาเพื่อรองรับการดำเนินงานในแต่ละวันอย่างมีประสิทธิภาพ การบริหารจัดการที่เหมาะสมไม่เพียงช่วยลดปัญหาความล่าช้าในการจัดหาวัสดุและยา แต่ยังช่วยลดความสูญเสียจากการเก็บรักษาไม่เหมาะสม รวมถึงการเพิ่มคุณภาพและความต่อเนื่องของการให้บริการผู้ป่วย

ในบทความนี้จะกล่าวถึงประเภทและหลักการจัดเก็บของคลังในโรงพยาบาลอย่างละเอียด โดยแบ่งประเภทคลังออกเป็น 5 รูปแบบหลัก ได้แก่ การแบ่งตามสิ่งที่จัดเก็บ ระดับการกระจาย และการใช้งาน การควบคุมและการจัดการ ลักษณะเฉพาะของการจัดเก็บ และหน่วยงานภายในโรงพยาบาล เพื่อให้ผู้อ่านเข้าใจถึงความสำคัญและความจำเป็นของการจัดการคลังอย่างเป็นระบบ

#### 1. การแบ่งประเภทตามสิ่งที่จัดเก็บ

##### 1.1 คลังยา (Pharmacy Store)

คลังยาทำหน้าที่จัดเก็บและบริหารจัดการยาทุกประเภทที่จำเป็นสำหรับการรักษาพยาบาล ตั้งแต่ยาสามัญ ยาเฉพาะทาง ไปจนถึงยาที่อยู่ภายใต้การควบคุม เช่น ยาเสพติดหรือยาที่ต้องการการเก็บรักษาในสภาวะแวดล้อมพิเศษ เช่น การควบคุมอุณหภูมิในระบบ Cold Chain การบริหารจัดการคลังยามุ่งเน้นที่ความถูกต้องแม่นยำ เพื่อป้องกันข้อผิดพลาดที่อาจเกิดขึ้นจากการจ่ายยาผิดประเภท ปริมาณ หรือคุณภาพ นอกจากนี้ยังต้องคำนึงถึงการตรวจสอบอายุการใช้งานของยา (Expiry Date) เพื่อป้องกันการสูญเสียทางเศรษฐกิจและความเสี่ยงต่อผู้ป่วย

##### 1.2 คลังเวชภัณฑ์ (Medical Supplies Store)

คลังเวชภัณฑ์จัดเก็บวัสดุและอุปกรณ์ที่ใช้ในการดูแลผู้ป่วย เช่น ถุงมือ ผ้าก๊อช หน้ากากอนามัย และวัสดุสิ้นเปลืองต่าง ๆ การจัดการในคลังเวชภัณฑ์ต้องมุ่งเน้นการจัดเก็บให้เพียงพอต่อการใช้งานของแต่ละแผนก ลดความล่าช้าในการจัดหา และหลีกเลี่ยงการขาดแคลนในสถานการณ์ฉุกเฉิน

##### 1.3 คลังเครื่องมือแพทย์ (Medical Equipment Store)

เครื่องมือทางการแพทย์ที่มีมูลค่าสูงและจำเป็น เช่น เครื่องช่วยหายใจ เครื่องวัดความดัน และอุปกรณ์ผ่าตัด ถูกจัดเก็บในคลังเครื่องมือแพทย์โดยมีมาตรการควบคุมคุณภาพและการบำรุงรักษาอย่างต่อเนื่อง การตรวจสอบและการบันทึกประวัติการใช้งานเป็นสิ่งสำคัญ เนื่องจากเครื่องมือเหล่านี้มักมีอายุการใช้งานและต้องการการดูแลพิเศษเพื่อรักษาสภาพการใช้งาน

#### 1.4 คลังวัสดุทั่วไป (General Supplies Store)

วัสดุที่ไม่เกี่ยวข้องโดยตรงกับงานการแพทย์ เช่น กระดาษ เครื่องเขียน หรืออุปกรณ์สำนักงานอื่น ๆ จะถูกจัดเก็บในคลังวัสดุทั่วไป คลังประเภทนี้มักมีการจัดการที่เรียบง่ายกว่าเนื่องจากไม่มีความซับซ้อนเรื่องการควบคุมคุณภาพหรือการหมดอายุของสินค้า

### 2. การแบ่งประเภทตามระดับการกระจายและการใช้งาน

#### 2.1 คลังหลัก (Central Store)

คลังหลักเป็นจุดศูนย์กลางในการจัดเก็บและกระจายวัสดุไปยังหน่วยงานหรือคลังย่อยต่าง ๆ ภายในโรงพยาบาล การจัดการในคลังหลักต้องมุ่งเน้นที่การควบคุมปริมาณวัสดุให้เพียงพอต่อความต้องการของทั้งโรงพยาบาล การติดตามและวิเคราะห์การใช้งานในแต่ละช่วงเวลาเป็นสิ่งสำคัญ เพื่อป้องกันการขาดแคลนวัสดุในสถานการณ์ที่มีความต้องการสูง

#### 2.2 คลังย่อย (Sub-store or Departmental Store)

คลังย่อยตั้งอยู่ในแผนกหรือหน่วยงานเฉพาะ เช่น แผนกผู้ป่วยใน (IPD) หรือแผนกผู้ป่วยนอก (OPD) คลังย่อยมีหน้าที่เก็บรักษาวัสดุและยาที่จำเป็นสำหรับการใช้งานประจำวันในแผนกนั้น ๆ เพื่อให้สามารถตอบสนองต่อความต้องการได้อย่างรวดเร็วและลดเวลาที่เสียไปในการขอเบิกวัสดุจากคลังหลัก

### 3. การแบ่งประเภทตามการควบคุมและการจัดการ

#### 3.1 คลังควบคุมพิเศษ (Controlled Access Store)

คลังควบคุมพิเศษจัดเก็บวัสดุหรือยาที่มีความเสี่ยงสูง เช่น ยาเสพติด วัคซีน และสารเคมีอันตราย การเข้าถึงคลังประเภทนี้มักมีข้อกำหนดและมาตรการความปลอดภัยที่เข้มงวด เช่น การติดตั้งระบบล็อกอิเล็กทรอนิกส์ การบันทึกประวัติการเบิกจ่าย และการตรวจสอบจากเจ้าหน้าที่ผู้มีอำนาจ

#### 3.2 คลังทั่วไป (General Store)

คลังทั่วไปจัดเก็บวัสดุที่ไม่มีความเสี่ยงสูงและไม่ต้องการการควบคุมพิเศษ เช่น อุปกรณ์สำนักงานและวัสดุสนับสนุนการทำงานทั่วไป การจัดการในคลังประเภทนี้เน้นที่ความสะดวกในการเข้าถึงและความชัดเจนในการจัดเก็บ

### 4. การแบ่งประเภทตามลักษณะเฉพาะของการจัดเก็บ

#### 4.1 คลังเย็น (Cold Storage)

วัสดุที่ต้องการการควบคุมอุณหภูมิ เช่น วัคซีน ยาบางประเภท และเลือด ถูกจัดเก็บในคลังเย็น ซึ่งมีระบบควบคุมอุณหภูมิที่สามารถรักษาคุณภาพของวัสดุได้อย่างเหมาะสม การตรวจสอบอุณหภูมิอย่างต่อเนื่องเป็นสิ่งสำคัญเพื่อหลีกเลี่ยงความเสียหายที่อาจเกิดขึ้น

#### 4.2 คลังแห้ง (Dry Storage)

คลังแห้งจัดเก็บวัสดุที่ไม่ต้องการความชื้น เช่น อุปกรณ์การแพทย์บางชนิด การบริหารคลังแห้งมุ่งเน้นที่การป้องกันความชื้นและฝุ่น เพื่อรักษาคุณภาพของวัสดุให้อยู่ในสภาพดี

### 4.3 คลังฉุกเฉิน (Emergency Stock Store)

คลังฉุกเฉินเก็บวัสดุและอุปกรณ์ที่ใช้ในสถานการณ์วิกฤต เช่น ชุดปฐมพยาบาล อุปกรณ์ช่วยชีวิต และเวชภัณฑ์สำหรับภัยพิบัติ คลังประเภทนี้ต้องพร้อมใช้งานตลอดเวลา และมีการจัดการสต็อกอย่างเคร่งครัด

## 5. การแบ่งประเภทตามหน่วยงานภายในโรงพยาบาล

- **คลังแผนกเภสัชกรรม:** ดูแลการจัดเก็บและกระจายยาภายในโรงพยาบาล
- **คลังแผนกศัลยกรรม:** เก็บเครื่องมือและวัสดุที่ใช้ในงานผ่าตัด
- **คลังแผนกฉุกเฉิน:** สำรองอุปกรณ์สำหรับการให้บริการในกรณีฉุกเฉิน

### รูปแบบการจัดเก็บสินค้าในคลังโรงพยาบาล

#### 1. การจัดเก็บตามประเภทสินค้า

- แยกหมวดหมู่สินค้า เช่น ยา เวชภัณฑ์ อุปกรณ์การแพทย์ และวัสดุสำนักงาน
- วัสดุที่มีความเสี่ยงสูง เช่น ยาเสพติดหรือสารเคมี ควรจัดเก็บแยกในพื้นที่ปลอดภัย เช่น ตู้ล็อกที่มีระบบควบคุมการเข้าถึง
- ใช้ป้ายกำกับที่ชัดเจนระบุหมวดหมู่และประเภทสินค้าเพื่อความสะดวกในการจัดการ

#### 2. การจัดเก็บตามอายุสินค้า

- ใช้หลัก First Expired, First Out (FEFO): สินค้าที่มีวันหมดอายุก่อนต้องถูกนำออกใช้ก่อน เพื่อลดความเสี่ยงของการเสื่อมสภาพหรือหมดอายุ
- สำหรับสินค้าที่ไม่มีวันหมดอายุ อาจใช้หลัก First In, First Out (FIFO): สินค้าที่ถูกนำเข้ามาคลังก่อนต้องถูกนำออกใช้ก่อน
- การจัดเรียงสินค้าควรระบุวันหมดอายุไว้ที่ด้านหน้าของชั้นวาง เพื่อให้มองเห็นได้ง่ายและหยิบใช้ได้สะดวก

#### 3. การจัดเก็บตามลักษณะการใช้งาน

- วัสดุที่ใช้งานบ่อย เช่น ยาพื้นฐานหรือเวชภัณฑ์ที่ใช้ประจำ ควรจัดเก็บในพื้นที่ที่หยิบใช้ได้ง่าย เช่น ชั้นวางที่เข้าถึงได้ทันที
- วัสดุเฉพาะทางหรือสินค้าที่ใช้น้อย ควรจัดเก็บในพื้นที่ห่างไกลหรือชั้นสูง เพื่อประหยัดพื้นที่สำหรับสินค้าที่ใช้งานบ่อย

#### 4. การจัดเก็บตามขนาดและน้ำหนัก

- วัสดุที่มีน้ำหนักมาก เช่น กล่องยา หรืออุปกรณ์ทางการแพทย์ขนาดใหญ่ ควรจัดเก็บในชั้นล่างเพื่อความปลอดภัยและลดความเสี่ยงของอุบัติเหตุ
- วัสดุขนาดเล็กหรือเบา เช่น ถุงมือแพทย์หรือหลอดฉีดยา สามารถจัดเก็บในชั้นสูงกว่าได้ เพื่อเพิ่มพื้นที่ใช้สอยในส่วนล่าง

## 5. การจัดเก็บตามสถานะสินค้า

- สินค้าพร้อมใช้งาน: จัดเก็บในพื้นที่ที่เข้าถึงง่ายที่สุด
- สินค้าที่รอการตรวจสอบหรืออนุมัติ: จัดเก็บในพื้นที่เฉพาะพร้อมป้ายระบุสถานะสินค้า
- สินค้าที่หมดอายุหรือชำรุด: จัดเก็บแยกไว้ในพื้นที่ที่ชัดเจนและระบุให้เห็นชัดเจน เพื่อลดความเสี่ยงของการนำมาใช้งานโดยไม่ตั้งใจ

## 6. การจัดเก็บแบบรหัสสินค้า (Coding System)

- ใช้ระบบรหัสสินค้า เช่น เลขที่ล็อต หมายเลขจัดเก็บ หรือบาร์โค้ด เพื่อติดตามและระบุสินค้าในคลัง
- ใช้บาร์โค้ดหรือ QR Code เพื่อเพิ่มความสะดวกในการสแกนค้นหาและตรวจสอบสินค้า

## 7. การจัดเก็บในพื้นที่ควบคุมอุณหภูมิ

- วัสดุที่ต้องการการควบคุมอุณหภูมิ เช่น วัคซีนหรือเลือด ควรจัดเก็บในตู้เย็นหรือห้องเย็นที่มีการตรวจสอบอุณหภูมิอย่างสม่ำเสมอ
- อุณหภูมิต้องเหมาะสม เช่น วัคซีนควรเก็บที่ 2-8°C และโลหิตควรเก็บที่ 1-6°C
- ใช้ระบบบันทึกอุณหภูมิแบบเรียลไทม์เพื่อป้องกันความผิดพลาด

## 8. การจัดเก็บแบบโซน (Zone Storage)

- แบ่งพื้นที่ในคลังออกเป็นโซนตามประเภทสินค้า เช่น โซนเวชภัณฑ์ โซนอุปกรณ์ฉุกเฉิน
- โซนที่มีความสำคัญ เช่น โซนสำหรับยา High Alert Drugs ควรจัดให้อยู่ในพื้นที่ที่เข้าถึงง่ายและปลอดภัย

## 9. การจัดเก็บแบบชั้นวาง (Shelf Storage)

- ใช้ชั้นวางที่ออกแบบให้เหมาะสมกับประเภทและขนาดของสินค้า
- ติดป้ายกำกับระบุตำแหน่ง เช่น SHELF-A1, ROW-B3 เพื่อช่วยในการค้นหา

## 10. การจัดเก็บแบบ Pallet หรือ Container

- เหมาะสำหรับสินค้าในปริมาณมาก เช่น กล่องยา หรือวัสดุอุปกรณ์ที่จัดส่งเป็นล็อต
- สินค้าถูกจัดเรียงบนพาเลทหรือในคอนเทนเนอร์เพื่อง่ายต่อการเคลื่อนย้าย

## 11. การจัดเก็บแบบอัตโนมัติ (Automated Storage)

- ใช้เทคโนโลยี เช่น ระบบจัดเก็บและเบิกจ่ายสินค้าอัตโนมัติ (Automated Storage and Retrieval System - ASRS)
- เหมาะสำหรับคลังขนาดใหญ่ที่ต้องการความแม่นยำและรวดเร็วในการจัดการ



## มาตรฐานที่เกี่ยวข้องกับการจัดเก็บในคลังโรงพยาบาล

### 1. Good Storage Practices (GSP)

- GSP เป็นมาตรฐานการจัดเก็บสินค้าที่เน้นการรักษาคุณภาพของสินค้า โดยเฉพาะยาและเวชภัณฑ์
- ระบุแนวทางในการควบคุมอุณหภูมิ ความชื้น และแสง เพื่อป้องกันการเสื่อมสภาพของสินค้า

### 2. WHO Good Distribution Practices (GDP)

- มาตรฐาน GDP ครอบคลุมการจัดเก็บ การขนส่ง และการกระจายสินค้า เพื่อรักษาคุณภาพตั้งแต่ต้นทางจนถึงปลายทาง

### 3. ISO 9001 (Quality Management System)

- ISO 9001 ระบุแนวทางการจัดการคุณภาพในทุกขั้นตอนของการบริหารคลังสินค้า เพื่อเพิ่มประสิทธิภาพและลดข้อผิดพลาด

### 4. Cold Chain Standards

- มาตรฐานสำหรับการจัดเก็บวัคซีน โลหิต และผลิตภัณฑ์ที่ต้องควบคุมอุณหภูมิ
- กำหนดให้มีการตรวจสอบและบันทึกอุณหภูมิอย่างต่อเนื่อง

#### 2.1.4 ประโยชน์ของคลังโรงพยาบาล

1. ความพร้อมในการให้บริการ คลังโรงพยาบาลช่วยให้มีทรัพยากรทางการแพทย์พร้อมใช้งานอยู่เสมอ เช่น ยา เวชภัณฑ์ ครุภัณฑ์ และอุปกรณ์การแพทย์ การจัดการยอตคงคลังอย่างมีประสิทธิภาพช่วยลดปัญหาการขาดแคลนวัสดุ ซึ่งมีความสำคัญโดยเฉพาะในสถานการณ์ฉุกเฉินหรือการระบาดของโรค

2. การควบคุมคุณภาพ คลังโรงพยาบาลมีบทบาทในการควบคุมคุณภาพของสินค้าที่จัดเก็บ โดยเฉพาะยา เวชภัณฑ์ ครุภัณฑ์ และอุปกรณ์ทางการแพทย์ การตรวจสอบคุณภาพและการจัดเก็บที่เหมาะสมช่วยให้ทรัพยากรมีคุณภาพตามมาตรฐานและปลอดภัยต่อการใช้งาน

3. การบริหารจัดการทรัพยากรอย่างมีประสิทธิภาพ การบริหารจัดการคลังช่วยให้ทรัพยากรถูกใช้ไปอย่างคุ้มค่า ลดการสูญเสียและการหมดอายุของสินค้า และการจัดการสต็อกที่มีประสิทธิภาพยังช่วยให้การเบิกจ่ายเป็นไปอย่างเป็นระบบ ลดความซ้ำซ้อน

4. การสนับสนุนการทำงานของบุคลากรทางการแพทย์ คลังโรงพยาบาลที่มีการจัดการที่ดีช่วยลดภาระงานของบุคลากรทางการแพทย์ในการค้นหาและเบิกจ่ายวัสดุ ทำให้บุคลากรสามารถทุ่มเทเวลาในการดูแลผู้ป่วยได้มากขึ้น

5. การควบคุมต้นทุน การจัดการคลังอย่างมีประสิทธิภาพช่วยลดต้นทุนในการจัดซื้อและการจัดเก็บ การสั่งซื้อในปริมาณที่เหมาะสมและการจัดเก็บที่ถูกต้องช่วยลดการสูญเสียจากการหมดอายุและการจัดเก็บที่ไม่ถูกต้อง

### 2.1.5 การจัดการคลังโรงพยาบาล

การจัดการคลังโรงพยาบาล หมายถึง การบริหารจัดการสินค้าคงคลังที่ใช้ในโรงพยาบาล เช่น ยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์ และวัสดุอื่น ๆ ที่จำเป็นในการให้บริการทางการแพทย์ เพื่อให้การดำเนินงานของโรงพยาบาลเป็นไปอย่างราบรื่น และสามารถตอบสนองความต้องการของผู้ป่วยได้อย่างมีประสิทธิภาพและปลอดภัย การจัดการคลังโรงพยาบาลเกี่ยวข้องกับหลายขั้นตอน เช่น การรับสินค้า การจัดเก็บ การเบิกจ่าย การควบคุมคุณภาพ และการตรวจสอบสินค้าคงคลัง รวมถึงการควบคุมต้นทุน การจัดการข้อมูล และการประเมินผลการทำงานเพื่อให้การบริการทางการแพทย์เป็นไปอย่างราบรื่นและปลอดภัย

#### ● วัตถุประสงค์ของการจัดการคลังโรงพยาบาล

- การบริหารจัดการคลังกลาง: ตรวจสอบและควบคุมปริมาณยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์ให้มีความเหมาะสม ลดการขาดแคลนหรือการมีโดยมีทรัพยากรเกินความจำเป็น
- การควบคุมคุณภาพ: ควบคุมคุณภาพยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์ที่จัดเก็บเพื่อป้องกันการเสื่อมสภาพและมั่นใจว่ามีมาตรฐานและปลอดภัยต่อการใช้งาน
- การบริหารจัดการพื้นที่จัดเก็บ: วางแผนและจัดสรรพื้นที่อย่างมีประสิทธิภาพ โดยใช้เทคโนโลยีในการบริหารพื้นที่เพื่อลดเวลาการค้นหายา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์
- การควบคุมความปลอดภัย: ควบคุมการจัดเก็บวัสดุอันตรายและป้องกันอุบัติเหตุโดยการติดตั้งระบบป้องกันอัคคีภัย

### 2.1.6 กิจกรรมหลักภายในคลังโรงพยาบาล

การจัดการคลังโรงพยาบาลประกอบด้วยกิจกรรมหลักหลายประการที่ต้องดำเนินการอย่างมีระบบและประสิทธิภาพ เพื่อบริการบริการทางการแพทย์ โดยสามารถแบ่งออกเป็น 8 งานหลักดังนี้

1. **งานรับสินค้า (Receiving)** การรับสินค้าเป็นขั้นตอนแรกในการจัดการคลัง โดยต้องตรวจสอบจำนวนสินค้าและความถูกต้องของรายการสินค้าตามคำสั่งซื้อ การตรวจสอบคุณภาพของสินค้า เช่น การเช็กวันหมดอายุ สภาพบรรจุภัณฑ์ และการตรวจสอบสินค้าเบื้องต้นเพื่อให้แน่ใจว่าสินค้ามีคุณภาพที่เหมาะสมก่อนนำมาเก็บในคลัง นอกจากนี้ยังต้องมีการบันทึกข้อมูลสินค้าเข้าสู่ระบบคลังอย่างถูกต้องและแม่นยำ
2. **การตรวจพิสูจน์ (Inspection)** หลังจากการรับสินค้า จะมีการตรวจสอบคุณภาพเพิ่มเติมเพื่อให้แน่ใจว่าสินค้ามีมาตรฐานที่เหมาะสม การตรวจพิสูจน์นี้จะครอบคลุมการทดสอบการทำงานของเครื่องมือแพทย์หรือการตรวจสอบยาและวัสดุที่ใช้ในการรักษา

เช่น การทดสอบการทำงานของเครื่องมือทางการแพทย์หรือการทดสอบยาเพื่อความปลอดภัยในการใช้งาน

3. **การตรวจแยกประเภท (Sorting)** การแยกประเภทสินค้ามีความสำคัญเพื่อให้การจัดเก็บและการทำงานในภายหลังเป็นไปอย่างมีประสิทธิภาพ สินค้าที่มีเงื่อนไขพิเศษ เช่น ยาหรือวัสดุที่ต้องเก็บในอุณหภูมิหรือลักษณะเฉพาะต้องแยกออกจากสินค้าทั่วไป เพื่อให้สามารถเข้าถึงได้ง่ายและลดความเสี่ยงในการสูญหายหรือการใช้ผิดประเภท
4. **งานจัดเก็บสินค้า (Storage)** การจัดเก็บสินค้าต้องมีการวางแผนพื้นที่อย่างรอบคอบ การจัดเก็บควรคำนึงถึงความสะดวกในการเข้าถึงและการจัดเรียงให้เหมาะสมตามประเภทของสินค้า เช่น การเก็บสินค้าที่ต้องการการควบคุมอุณหภูมิหรือความชื้นให้เหมาะสม ระบบการจัดเก็บจะต้องมีการตรวจสอบและบำรุงรักษาเป็นระยะเพื่อให้สภาพการจัดเก็บเป็นไปตามมาตรฐานและปลอดภัย
5. **งานดูแลรักษา (Maintenance)** การดูแลรักษาเป็นสิ่งสำคัญเพื่อให้สินค้ามีคุณภาพที่ดีตลอดระยะเวลาการจัดเก็บ โดยการตรวจสอบสภาพสินค้าทุกประเภทอย่างสม่ำเสมอ เช่น การตรวจสอบวันหมดอายุของยา หรือการทดสอบการทำงานของอุปกรณ์การแพทย์ การดูแลรักษาคุณภาพช่วยป้องกันปัญหาการใช้สินค้าหมดอายุหรือสินค้าที่เสื่อมสภาพ
6. **งานจัดส่ง (Distribution)** การจัดส่งภายในโรงพยาบาลต้องทำตามคำสั่งของ แต่ละแผนก และต้องกระจายยา เวชภัณฑ์ วัสดุภัณฑ์ และอุปกรณ์ทางการแพทย์ในเวลาที่เหมาะสมเพื่อให้การดำเนินงานของโรงพยาบาลไม่ขาดตอน การเบิกจ่ายจะต้องเป็นไปอย่างรวดเร็วและแม่นยำ การบันทึกข้อมูลการเบิกจ่ายจะช่วยให้สามารถติดตามสถานะได้อย่างมีประสิทธิภาพ
7. **การนำออกจากที่จัดเก็บ (Withdrawal)** การเบิกจ่ายจากคลังกลางต้องเป็นไปตามระบบที่กำหนด เพื่อให้มั่นใจว่าไม่มีการเบิกจ่ายผิดพลาด การนำยา เวชภัณฑ์ วัสดุภัณฑ์ และอุปกรณ์ทางการแพทย์ จากที่จัดเก็บออกต้องทำอย่างมีระเบียบ โดยบันทึกข้อมูลการเบิกจ่ายในระบบการจัดการคลังเสมอ นอกจากนี้ยังต้องคำนึงถึงการเบิกจ่ายตามความเร่งด่วนและความต้องการของแผนก
8. **การจัดส่ง (Delivery)** การจัดส่งผ่านคลังจะต้องมีการควบคุมอย่างเข้มงวดเพื่อให้มั่นใจว่าสินค้าถูกส่งไปยังหน่วยงานที่ต้องการตามคำสั่ง การจัดส่งอาจมีการส่งยาไปยังผู้ป่วยหรือส่งอุปกรณ์ทางการแพทย์ไปยังห้องผ่าตัด ซึ่งต้องมีความรวดเร็วและแม่นยำ การตรวจสอบการจัดส่งช่วยลดความผิดพลาดและเพิ่มความปลอดภัยในการใช้งาน

#### 2.1.7 การจัดการของแผนกบริการปลอดเชื้อส่วนกลาง

แผนกบริการปลอดเชื้อส่วนกลาง (Central Sterile Supply Department : CSSD) มีบทบาทสำคัญในการสนับสนุนการให้บริการทางการแพทย์ ด้วยการทำความสะอาด การทำให้ปราศจากเชื้อ และการจัดหาเครื่องมือแพทย์ปลอดเชื้อแก่หน่วยงานต่าง ๆ ภายในโรงพยาบาล การ

จัดการที่มีประสิทธิภาพช่วยลดความเสี่ยงในการติดเชื้อและเพิ่มประสิทธิภาพการใช้งานทรัพยากร โดยกระบวนการหลักประกอบด้วย

### 1. การรับเครื่องมือแพทย์ (Receiving)

การรับเครื่องมือจากหน่วยงานต่าง ๆ ต้องตรวจสอบความสมบูรณ์ของเครื่องมือ และบันทึกข้อมูลเข้าสู่ระบบเพื่อติดตามการใช้งานอย่างเป็นระบบ

### 2. การทำความสะอาด (Cleaning)

เครื่องมือแพทย์ที่รับมาจะผ่านกระบวนการทำความสะอาดด้วยเครื่องมือเฉพาะ เช่น เครื่องล้างอัลตราโซนิก หรือการล้างด้วยมือ สำหรับเครื่องมือที่ซับซ้อนหรือเปราะบาง

### 3. การตรวจสอบและบำรุงรักษา (Inspection and Maintenance)

เครื่องมือทุกชิ้นต้องได้รับการตรวจสอบสภาพและความสมบูรณ์ รวมถึงการซ่อมบำรุงเล็กน้อย เพื่อให้พร้อมใช้งานต่อไป

### 4. การทำให้ปราศจากเชื้อ (Sterilization)

เครื่องมือที่ผ่านการทำความสะอาดจะเข้าสู่กระบวนการทำให้ปราศจากเชื้อ เช่น การใช้ไอน้ำแรงดันสูง (Autoclave) การใช้ก๊าซ ETO หรือรังสี การเลือกวิธีขึ้นอยู่กับวัสดุของเครื่องมือ

### 5. การบรรจุและจัดเก็บ (Packaging and Storage)

เครื่องมือที่ผ่านการฆ่าเชื้อจะบรรจุในบรรจุภัณฑ์ที่ปลอดภัย พร้อมระบุข้อมูลสำคัญ เช่น วันที่ฆ่าเชื้อ วันหมดอายุ จากนั้นจัดเก็บในพื้นที่ควบคุมที่เหมาะสม

### 6. การแจกจ่าย (Distribution)

กระบวนการแจกจ่ายเครื่องมือแพทย์ต้องรวดเร็วและแม่นยำ พร้อมบันทึกข้อมูลการแจกจ่ายเพื่อความโปร่งใสและตรวจสอบได้

### 7. การควบคุมคุณภาพและความปลอดภัย (Quality Control and Safety)

การตรวจสอบคุณภาพในทุกขั้นตอน เพื่อให้มั่นใจว่าเครื่องมือมีมาตรฐานและปลอดภัยต่อการใช้งาน การควบคุมกระบวนการฆ่าเชื้อและปฏิบัติตามมาตรฐานความปลอดภัยเป็นสิ่งสำคัญ

## 2.2 หลักการของบาร์โค้ดและคิวอาร์โค้ด

การเก็บข้อมูลในรูปแบบรหัสที่สามารถอ่านและแปลงข้อมูลได้ด้วยเครื่องอ่านเป็นหนึ่งในเทคโนโลยีที่มีความสำคัญในยุคดิจิทัล โดยบาร์โค้ดและคิวอาร์โค้ดถูกพัฒนาขึ้นเพื่อเพิ่มประสิทธิภาพในการจัดการข้อมูลที่หลากหลาย เช่น ข้อมูลสินค้า การติดต่อ ลิงก์เว็บไซต์ หรือการชำระเงิน เทคโนโลยีนี้ช่วยให้การจัดการข้อมูลเป็นไปอย่างรวดเร็ว ลดข้อผิดพลาด และเพิ่มความสะดวกในการดำเนินงานในหลากหลายอุตสาหกรรม

### 2.2.1 บาร์โค้ด

บาร์โค้ด (Barcode) คือรหัสแท่งที่ประกอบด้วยเส้นมืดและเส้นสว่างเรียงกันในแนวนิ่ง ใช้แทนตัวเลขและตัวอักษร เพื่อให้คอมพิวเตอร์สามารถรับข้อมูลได้ง่ายและรวดเร็วผ่านเครื่องอ่านบาร์โค้ด (Barcode Scanner) โดยเครื่องอ่านจะแปลงแสงสะท้อนจากบาร์โค้ดเป็นรหัสและส่งเข้าสู่ระบบคอมพิวเตอร์ บาร์โค้ดช่วยลดข้อผิดพลาดในการคีย์ข้อมูล ประหยัดเวลา ลดค่าใช้จ่าย และลดการใช้ทรัพยากรบุคคล

#### ● บาร์โค้ด แบ่งออกเป็น 2 ประเภท

บาร์โค้ด 1 มิติ (1D) มีลักษณะเป็นเส้นขาวดำที่มีความหนาบางสลับกัน เหมาะกับการใช้งานทั่วไปที่ไม่ต้องการใช้พื้นที่ในการเก็บข้อมูลเยอะ เช่น รหัสสินค้า ชื่อผลิตภัณฑ์ เลขประจำตัว เลขสมาชิก Serial no. และ Lot no. ของสินค้า



รูปที่ 2.1 รหัสสินค้า

#### - บาร์โค้ด 1 มิติ

EAN 13 (European Article Numbering) เป็นแบบบาร์โค้ดที่ได้รับการยอมรับมากที่สุดในโลก และเป็นบาร์โค้ดมาตรฐานสากลที่เราต้องไปขึ้นทะเบียนกับ GS1 เป็นตัวเลขชุด 13 หลัก นิยมใช้กับสินค้าอุปโภคบริโภคทั่วไป เช่น อาหาร เครื่องดื่ม



รูปที่ 2.2 หมายเลขประจำตัว EAN.UPC 13

บาร์โค้ด EAN 13 จะมีลักษณะเฉพาะของชุดตัวเลขจำนวน 13 หลัก ซึ่งมีความหมายดังนี้

- หลักแรก คือ รหัสของประเทศผู้ผลิต
- หลักถัดมา คือ รหัสโรงงานที่ผลิต
- หลักถัดมา คือ รหัสของสินค้า

และตัวเลขในหลักสุดท้าย จะเป็นตัวเลขตรวจสอบความถูกต้องของบาร์โค้ด (check digit)

Code 39 และ Code 128 เป็นบาร์โค้ดที่ไม่ต้องขึ้นทะเบียนและสามารถใช้งานได้ฟรี เหมาะสำหรับงานอุตสาหกรรม หน่วยงานรัฐ และองค์กรเอกชน โดยไม่มีการกำหนดจำนวนหลักตายตัว ขึ้นอยู่กับความสามารถของเครื่องอ่านบาร์โค้ด สามารถแสดงตัวเลข ตัวอักษร และอักขระพิเศษ (ASCII) ได้ โดยไม่จำเป็นต้องมีตัวเลขตรวจสอบความถูกต้องของบาร์โค้ด



รูปที่ 2.3 CODE 39

Code 39 มีการเพิ่มเครื่องหมาย “\*” ที่หลักแรกและหลักสุดท้ายเพื่อบอกตำแหน่งเริ่มต้นและตำแหน่งสิ้นสุด จำกัดตัวอักษรเฉพาะ A-Z, 0-9, +, -, %, \$



รูปที่ 2.4 CODE128

Code 128 บาร์โค้ดชนิดนี้ถูกนำมาใช้มากกว่า Code 39 เพราะสามารถใช้ตัวอักษร และอักขระพิเศษได้ทุกตัวบนคีย์บอร์ด

### - บาร์โค้ด 2 มิติ

บาร์โค้ด 2 มิติ เป็นเทคโนโลยีที่พัฒนาจากบาร์โค้ด 1 มิติ โดยสามารถบรรจุข้อมูลได้ทั้งแนวดิ่งและแนวนอน รองรับข้อมูลได้มากถึง 4,000 ตัวอักษร หรือประมาณ 200 เท่าของบาร์โค้ด 1 มิติ ในพื้นที่เท่ากันหรือเล็กกว่า สามารถเก็บข้อมูลได้หลากหลายภาษา เช่น ภาษาไทย ญี่ปุ่น จีน และเกาหลี อีกทั้งยังสามารถถอดรหัสได้แม้บาร์โค้ดบางส่วนจะเสียหาย



รูปที่ 2.5 PDF417

PDF417 (Portable Data File) เป็นบาร์โค้ดรูปสี่เหลี่ยมผืนผ้าที่เครื่องอ่านสามารถอ่านได้ในทิศทางเดียว เช่น จากซ้ายไปขวา ขวาไปซ้าย หรือจากบนลงล่าง ล่างขึ้นบน เหมาะสำหรับงานที่ต้องการความละเอียดและความถูกต้องสูงเป็นพิเศษ



รูปที่ 2.6 Data Matrix

Data Matrix เป็นบาร์โค้ดที่มีลักษณะเป็นรูปสี่เหลี่ยมจัตุรัสหรือสี่เหลี่ยมผืนผ้า เหมาะสำหรับงานที่มีพื้นที่จำกัดและต้องการบาร์โค้ดขนาดเล็ก มักใช้ในอุตสาหกรรมอิเล็กทรอนิกส์ อาหาร อะไหล่รถยนต์ และเครื่องจักร



รูปที่ 2.7 QR Code (Quick Response)

QR Code (Quick Response) เป็นบาร์โค้ดรูปสี่เหลี่ยมที่เหมาะสมสำหรับการใช้งานหลากหลาย เช่น การเชื่อมต่อกับเว็บไซต์ การจ่ายบิล ชำระเงิน เพิ่มเพื่อนใน LINE ตัวใช้งานต่าง ๆ บอร์ดดิ้งพาส (Boarding Pass) หรือตั๋วเครื่องบิน

●หลักปฏิบัติที่ดีที่สุดในการจัดวางและพิมพ์บาร์โค้ด

วิธีการพิมพ์และการจัดวางบาร์โค้ดบนสินค้าเป็นปัจจัยสำคัญที่ส่งผลต่อความสามารถของเครื่องสแกนในการอ่านบาร์โค้ด วิธีการพิมพ์บาร์โค้ด มีดังตารางดังต่อไปนี้

ตารางที่ 2.1 ข้อมูลเพิ่มเติมเกี่ยวกับวิธีการพิมพ์บาร์โค้ด

วิธี	คำอธิบาย	ตัวอย่างสินค้า
บาร์โค้ดที่พิมพ์ลงบนสินค้าโดยตรง	วิธีการที่พบเห็นได้บ่อย โดยจะใช้เครื่องพิมพ์เลเซอร์ในการพิมพ์บาร์โค้ดลงบนสินค้าโดยตรงในกรณีที่บาร์โค้ดไม่ควรอยู่แยกจากตัวสินค้า	หนังสือ ภาชนะโลหะ ขวดพลาสติก
บาร์โค้ดที่พิมพ์ลงบนป้ายแปะที่ติดแนบกับสินค้า	ใช้โดยธุรกิจขนาดเล็กหรือธุรกิจที่ไม่มีเครื่องพิมพ์เลเซอร์สำหรับพิมพ์บาร์โค้ดลงบนสินค้าโดยตรง	เครื่องเขียน กาแฟ ผลิตภัณฑ์ไม่ได้บรรจุหีบห่อ
บาร์โค้ดที่พิมพ์ลงบนป้ายห้อยที่ติดแนบกับสินค้า	ใช้สำหรับสินค้าซึ่งไม่มีพื้นผิวที่เหมาะสมกับการแปะบาร์โค้ด ตลอดจนสำหรับสินค้าที่อาจจะได้รับความเสียหายจากกาติดป้ายแปะ	เครื่องแต่งกาย สินค้าสำหรับให้เป็นของขวัญ สินค้าสำหรับใช้บนโต๊ะ
บาร์โค้ดที่พิมพ์ลงบนบรรจุภัณฑ์สินค้า	ใช้สำหรับสินค้าที่มีกล่องหรือบรรจุภัณฑ์	รองเท้า รองเท้าสเก็ต อุปกรณ์อิเล็กทรอนิกส์ และสินค้าใด ๆ ที่บรรจุหีบห่อ



### ●แนวทางการพิมพ์บาร์โค้ด

แนวทางทั่วไปในการพิมพ์บาร์โค้ดอย่างถูกต้อง เพื่อให้แน่ใจว่าบาร์โค้ดมีขนาดและตำแหน่งเหมาะสม รวมถึงสามารถสแกนได้อย่างมีประสิทธิภาพ มีดังนี้:

- ใช้ความละเอียดไม่ต่ำกว่า 300 DPI เมื่อพิมพ์บาร์โค้ด เพื่อความคมชัดและความสามารถในการสแกน
- ทดสอบการพิมพ์บาร์โค้ด 1 แผ่นก่อนพิมพ์จำนวนมาก เพื่อตรวจสอบว่าใช้งานได้ถูกต้อง
- พิมพ์หมายเลขบาร์โค้ดไว้ใต้สัญลักษณ์ เพื่อให้สามารถระบุสินค้าได้ในกรณีที่บาร์โค้ดเสียหาย
- ขนาดขั้นต่ำของบาร์โค้ดควรกว้าง 1.5 นิ้ว x สูง 1 นิ้ว และตรวจสอบขนาดตามมาตรฐาน GS1
- ใช้หมึกสีดำหรือสีเข้ม หลีกเลี่ยงการใช้หมึกสีแดงหรือน้ำตาล
- พิมพ์บาร์โค้ดที่ไม่ซ้ำกันสำหรับสินค้าแต่ละรายการ เพื่อให้สามารถระบุและติดตามสินค้าในห่วงโซ่อุปทานได้อย่างแม่นยำ

### ●แนวทางการจัดวางบาร์โค้ด

ตรวจสอบแนวทางทั่วไปต่อไปนี้เกี่ยวกับวิธีการจัดวางบาร์โค้ดอย่างถูกต้อง

- จัดวางบาร์โค้ดให้เรียบไปบนตัวสินค้าและอย่าพับบาร์โค้ด
- อย่าให้มีเทปหรือห่อฟิล์มล้อมรอบบาร์โค้ด
- หลีกเลี่ยงการจัดวางสติ๊กเกอร์หรือองค์ประกอบของข้อความไว้ใกล้กับบาร์โค้ดของคุณมากเกินไป
- อย่าหั่นหรือตัดทอนส่วนบนหรือส่วนล่างของบาร์โค้ดเพื่อให้มีพื้นที่สำหรับป้ายกำกับของคุณมากขึ้น
- อย่าใส่บาร์โค้ดหลายชิ้นลงในสินค้าของคุณ
- อย่าใช้ลวดเย็บทะลุบาร์โค้ด
- อย่าจัดวางบาร์โค้ดในตำแหน่งที่มีพื้นที่ไม่เพียงพอบนตัวสินค้า
- อย่าจัดวางบาร์โค้ดที่ขอบสินค้า
- หากสินค้าของคุณมีรูปทรงสี่เหลี่ยมจัตุรัส สี่เหลี่ยมผืนผ้า กลม เว้า หรือนูน เราแนะนำให้ลองใช้ป้ายแปะเฉพาะจุด ป้ายห้อย หรือการ์ดสติ๊กเกอร์

#### 2.2.2 คิวอาร์โค้ด

QR Code (Quick Response Code) คือ สัญลักษณ์สี่เหลี่ยมที่ใช้แทนข้อมูลต่าง ๆ ซึ่งพัฒนามาจากบาร์โค้ด 2 มิติ โดยบริษัท Denso-Wave ประเทศญี่ปุ่นในปี 1994 ปัจจุบัน QR Code ได้รับความนิยมอย่างแพร่หลายในด้านสินค้า การชำระเงิน และการโฆษณา

เนื่องจากใช้งานง่ายและสอดคล้องกับเทคโนโลยีสมัยใหม่ เพียงนำกล้องโทรศัพท์มือถือสแกน QR Code ก็สามารถเข้าถึงข้อมูลที่ต้องการได้ทันที

- ประเภทของคิวอาร์โค้ด (QR Code)

สำหรับประเภทของ QR Code ในปัจจุบันนี้จะมีทั้งหมด 5 ประเภท ดังนี้

#### 1. QR Code Model 1

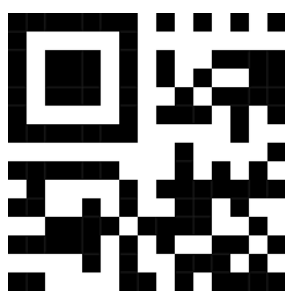
เป็น QR Code แบบดั้งเดิมที่มีขนาดใหญ่ที่สุด โดยจะมีขนาด 73\*73 โมดูล สามารถบรรจุข้อมูลได้มากถึง 1,167 ตัว และอีกหนึ่งแบบ คือ Model 2 เป็นเวอร์ชันที่ถูกพัฒนามาจาก Model 1 สามารถบรรจุข้อมูลได้มากถึง 7,089 ตัว ซึ่งในปัจจุบันนี้ Model 2 เป็นที่นิยมใช้กันอย่างมาก



รูปที่ 2.8 QR Code Model 1

#### 2. Micro QR Code

เป็น QR Code ที่มีขนาดเล็กกว่าแบบแรกมากพอสมควร เนื่องจากจะแสดงผลบางจุดตรงตำแหน่งเพียงตำแหน่งเดียว ซึ่งขนาดใหญ่ที่สุดของ Micro QR Code 17\*17 โมดูล ที่สามารถบรรจุข้อมูลได้ทั้งหมด 35 ตัว



รูปที่ 2.9 Micro QR Code

### 3.IQR Code

เป็น QR Code ที่มีขนาดเล็กและพิมพ์ออกมาแนวนอน ที่สามารถทำการเก็บข้อมูลได้มากกว่า 80% แต่หากมีการจัดเก็บข้อมูลในปริมาณที่เท่ากัน ก็จะประหยัดพื้นที่ในการแสดงผลได้มากถึง 30% ซึ่งสามารถเก็บข้อมูล 40,000 ตัวอักษร



รูปที่ 2.10 IQR Code

### 4.SQRC

สำหรับ SQRC เป็น QR Code ที่จะมีคุณลักษณะเหมือนกับ QR Code Model 1 และ QR Code Model 2 ทุกประการ แต่จะมีข้อแตกต่างกันเล็กน้อย คือ สามารถทำการเก็บข้อมูลกลับได้นั่นเอง



รูปที่ 2.11 SQRC

### 5.Frame QR

สำหรับ QR Code ประเภท Frame QR นั้น สามารถนำรูปภาพ กราฟิกมาติดบริเวณตรงกลางของ QR Code ได้ ส่วนใหญ่ก็มักจะนิยมใช้ QR Code ประเภท Frame QR ในงานประชาสัมพันธ์ Event นิทรรศการ เพื่อทำให้เกิดความสะดุดตามากที่สุดนั่นเอง



รูปที่ 2.12 Frame QR

● ประโยชน์ของระบบ QR Code

1. การเข้าถึงข้อมูลอย่างรวดเร็ว ระบบ QR Code ช่วยให้ผู้ใช้งานสามารถเข้าถึงข้อมูล เช่น เว็บไซต์ รายละเอียดสินค้า หรือบริการ ได้อย่างสะดวกและรวดเร็ว เพียงแค่ทำการสแกนโค้ดโดยใช้อุปกรณ์เคลื่อนที่
2. ลดความผิดพลาดในการป้อนข้อมูล การสแกน QR Code ช่วยลดความเสี่ยงที่อาจเกิดจากความผิดพลาดในการป้อนข้อมูลด้วยตนเอง เช่น การป้อน URL ที่ไม่ถูกต้อง
3. สนับสนุนการทำธุรกรรมดิจิทัล ระบบ QR Code มีบทบาทสำคัญในธุรกรรมทางการเงิน เช่น การชำระเงินผ่าน e-Wallet การโอนเงิน และการยืนยันการชำระเงิน ช่วยเพิ่มความสะดวกและความปลอดภัยให้แก่ผู้ใช้งาน
4. เสริมประสิทธิภาพในการจัดการข้อมูล QR Code สามารถใช้ในการติดตามและจัดเก็บข้อมูล เช่น การตรวจสอบสินค้าคงคลัง การบริหารจัดการระบบโลจิสติกส์ และการบันทึกข้อมูลในภาคส่วนการแพทย์
5. เพิ่มประสิทธิภาพในการตลาดและการประชาสัมพันธ์ QR Code เป็นเครื่องมือที่ช่วยสนับสนุนการตลาด เช่น การเชื่อมโยงไปยังเว็บไซต์โปรโมชั่นหรือข้อมูลผลิตภัณฑ์ เพื่อเพิ่มโอกาสในการเข้าถึงลูกค้า
6. ลดการใช้ทรัพยากร การใช้งาน QR Code ช่วยลดการใช้กระดาษในกระบวนการต่าง ๆ เช่น การพิมพ์เอกสาร บัตรโดยสาร หรือใบเสร็จรับเงิน
7. ความยืดหยุ่นในการประยุกต์ใช้ในอุตสาหกรรมต่าง ๆ QR Code มีการนำไปใช้งานในหลากหลายอุตสาหกรรม เช่น การขนส่ง การศึกษา การแพทย์ และการท่องเที่ยว
8. เพิ่มความปลอดภัยของข้อมูล QR Code สามารถผสมผสานการเข้ารหัสหรือมาตรการความปลอดภัยเพิ่มเติม เพื่อลดความเสี่ยงจากการปลอมแปลงข้อมูล
9. ช่วยลดเวลาในกระบวนการทำงานการสแกน QR Code ใช้เวลาน้อยและสามารถดำเนินการได้ทันที ซึ่งช่วยลดเวลาที่ใช้ในขั้นตอนต่าง ๆ

10. รองรับการใช้งานผ่านอุปกรณ์เคลื่อนที่ QR Code สามารถใช้งานร่วมกับสมาร์ทโฟนหรืออุปกรณ์เคลื่อนที่ได้อย่างสะดวก ซึ่งเหมาะสมกับผู้ใช้งานในยุคดิจิทัล

### 2.2.3 เครื่องพิมพ์บาร์โค้ดและคิวอาร์โค้ด

เครื่องพิมพ์บาร์โค้ดและคิวอาร์โค้ดถือเป็นอุปกรณ์สำคัญที่มีบทบาทอย่างมากในกระบวนการสร้างฉลากสำหรับการติดตามสินค้า การจัดการคลังสินค้า และการดำเนินธุรกิจ ในหลากหลายอุตสาหกรรม การเลือกใช้เครื่องพิมพ์ที่เหมาะสมสามารถช่วยเพิ่มประสิทธิภาพในการดำเนินงานได้อย่างมีนัยสำคัญ

#### ● ประเภทของเครื่องพิมพ์บาร์โค้ด

เครื่องพิมพ์แบบ Direct Thermal เครื่องพิมพ์ประเภทนี้ใช้ความร้อนโดยตรงในการพิมพ์บนกระดาษที่เคลือบสารเคมี เหมาะสำหรับฉลากที่มีอายุการใช้งานสั้น เช่น ฉลากพัสดุหรือฉลากสำหรับงานที่ไม่ต้องการความทนทานในระยะยาว

เครื่องพิมพ์แบบ Thermal Transfer เครื่องพิมพ์ประเภทนี้ใช้ความร้อนในการถ่ายโอนหมึกจากริบบอนลงบนวัสดุฉลาก ทำให้ฉลากมีความทนทานต่อการขีดข่วน ความชื้น และแสงแดด เหมาะสำหรับการใช้งานในระยะยาว

เครื่องพิมพ์แบบอิงค์เจ็ทและเลเซอร์ เครื่องพิมพ์ประเภทนี้เหมาะสำหรับการพิมพ์บาร์โค้ดหรือคิวอาร์โค้ดที่ต้องการความละเอียดสูง และรองรับการพิมพ์ในปริมาณมาก เช่น การพิมพ์ฉลากสินค้าที่มีกราฟิกหรือรายละเอียดที่ซับซ้อน

#### ● หลักเกณฑ์ในการเลือกเครื่องพิมพ์ที่เหมาะสม

ปริมาณการพิมพ์ : หากต้องการพิมพ์ฉลากในปริมาณมาก ควรเลือกเครื่องพิมพ์ที่มีความเร็วสูงและมีความทนทานเพียงพอ เพื่อรองรับการทำงานอย่างต่อเนื่อง

ขนาดและประเภทของฉลาก : พิจารณาว่าเครื่องพิมพ์สามารถรองรับขนาดและวัสดุของฉลากที่ต้องการใช้งานได้ เช่น กระดาษเคลือบ ริบบอน หรือวัสดุพิเศษอื่น ๆ

ความละเอียดในการพิมพ์ : สำหรับบาร์โค้ดหรือคิวอาร์โค้ดที่มีรายละเอียดสูง ควรเลือกเครื่องพิมพ์ที่มีความละเอียดในการพิมพ์สูง (DPI) เพื่อให้ได้ฉลากที่คมชัดและสามารถอ่านได้อย่างแม่นยำ

การเชื่อมต่อ : เลือกเครื่องพิมพ์ที่รองรับการเชื่อมต่อที่เหมาะสมกับระบบที่ใช้งาน เช่น การเชื่อมต่อผ่าน USB, Ethernet, หรือ Wi-Fi เพื่อความสะดวกในการติดตั้งและใช้งาน

### 2.2.4 เครื่องอ่านบาร์โค้ด

เครื่องอ่านบาร์โค้ด (Barcode Scanner) เป็นอุปกรณ์ที่มีหน้าที่นำข้อมูลเข้าสู่ระบบคอมพิวเตอร์ เวกเช่นเดียวกันกับคีย์บอร์ด แต่มีคุณสมบัติเพิ่มเติมคือความสามารถในการอ่านบาร์โค้ดแล้วนำไปประมวลผล แปลงค่าเป็นข้อมูลตัวเลขหรือตัวอักษรที่สามารถเข้าใจได้ด้วยภาษามนุษย์ก่อนส่งต่อไปให้กับระบบคอมพิวเตอร์ เครื่องอ่านบาร์โค้ดถูกคิดค้นขึ้นเพื่อเพิ่ม

ประสิทธิภาพในเรื่องของความรวดเร็วและแม่นยำในการทำงาน แทนการใช้แรงงานคนในการพิมพ์ข้อมูลเข้าสู่ระบบที่ละตัวอักษร ทีละตัวเลข ซึ่งมีโอกาสเกิดข้อผิดพลาดได้อยู่เสมอ และมีความล่าช้าอย่างมาก

ในการเลือกซื้อเครื่องอ่านบาร์โค้ด ผู้ใช้จำเป็นต้องพิจารณาถึงรูปแบบของข้อมูลบาร์โค้ดที่เราจะนำเครื่องอ่านไปใช้งานด้วย เพื่อให้เครื่องอ่านบาร์โค้ดที่ซื้อมา สามารถอ่านค่าบาร์โค้ดในรูปแบบนั้น ๆ ได้ โดยบาร์โค้ดจะมีอยู่ 2 แบบใหญ่ ๆ คือ บาร์โค้ดแท่งในแบบ 1D (1 มิติ) และ บาร์โค้ดในแบบ 2D (2 มิติ)

บาร์โค้ดในแบบ 1D จะมีลักษณะเป็นแท่งบาร์โค้ดในแนวนอนทั่ว ๆ ไปที่เราสามารถพบเห็นได้บนตัวสินค้าต่าง ๆ ที่มีการจำหน่ายอยู่ตามซูเปอร์มาร์เก็ต ห้างสรรพสินค้า เป็นสินค้าที่เราใช้สอยอยู่ในชีวิตประจำวัน โดยบาร์โค้ดแบบ 1D จะมีชนิดย่อยอยู่หลายชนิดด้วยกัน มีชื่อเรียกต่าง ๆ กัน เช่น EAN-13, Code 128, Code 39 และอื่น ๆ อีกมากมาย ตัวอย่างของบาร์โค้ดแท่งแบบ 1D ดังรูปด้านล่าง



รูปที่ 2.13 KS1689 Barcode

ซึ่งบาร์โค้ดในแบบ 1D นั้นมีข้อจำกัดในเรื่องของปริมาณการเก็บข้อมูลในแท่งของบาร์โค้ด เนื่องจากข้อมูลจะถูกบันทึกขยายออกไปในแนวนอนกว้างขึ้นเรื่อย ๆ ตามปริมาณข้อมูล จนถึงในอัตราส่วนที่เครื่องอ่านไม่สามารถอ่านได้ครอบคลุม จึงเหมาะกับการใช้ในงานทั่ว ๆ ไปที่ไม่มีความจำเป็นในการเก็บข้อมูลปริมาณมาก เช่น การเก็บตัวเลขรหัสสินค้า การเก็บชื่อผลิตภัณฑ์ หรือการเก็บหมายเลขเครื่อง Serial Number เป็นต้น

บาร์โค้ดในอีกรูปแบบคือบาร์โค้ดแบบ 2D (2 มิติ) ซึ่งถูกนำมาใช้ทดแทนข้อด้อยของบาร์โค้ดในแบบ 1D โดยบาร์โค้ดในแบบ 2D นั้นสามารถเก็บข้อมูลได้ทั้งในแนวตั้งและแนวนอนจึงสามารถเก็บข้อมูลได้ในปริมาณที่มากกว่าเพราะสามารถขยายออกไปได้ทั้งแนวตั้งและแนวนอน บาร์โค้ดแบบ 2 มิติ ยังไม่ได้มีการใช้งานกันอย่างแพร่หลายเมื่อเทียบกับบาร์โค้ดแท่ง 1D แต่ก็มีมีการนำบาร์โค้ดในรูปแบบนี้มาใช้งานกันมากขึ้นเรื่อย ๆ ดังจะเห็นได้ใน Application ต่าง ๆ ในโทรศัพท์มือถือ บาร์โค้ดที่อยู่ในลักษณะของรูปแบบนี้ เช่น QR Code Data Matrix PDF417 และอื่น ๆ อีกมากมาย ดังรูปตัวอย่างด้านล่าง



รูปที่ 2.14 KS 1689 Barcode

ดังนั้นในการเลือกซื้อเครื่องอ่านบาร์โค้ดจึงต้องคำนึงถึงคุณสมบัติของเครื่องอ่านบาร์โค้ดด้วยว่าจะสามารถยิงบาร์โค้ดที่ติดอยู่บนสินค้าของผู้ใช้ได้หรือไม่ นอกจากนี้เครื่องอ่านบาร์โค้ดยังสามารถแบ่งได้ตามชนิดของหัวอ่าน คือ CCD Laser Omnidirectional และ Imager โดยจะมีข้อแตกต่างในการใช้งาน ดังต่อไปนี้

1. CCD Scanner เป็นเทคโนโลยีการอ่านบาร์โค้ดที่เก่ากว่ารูปแบบอื่น ๆ ที่เหลือ จะเป็นเครื่องอ่านบาร์โค้ดที่มีลักษณะเป็นตัวปืน มีข้อดีในการใช้งานกลางแจ้ง บริเวณที่มีแสงสว่างมาก ๆ แต่ข้อเสียก็คือการยิงบาร์โค้ดด้วยเครื่องอ่านชนิดนี้ จำเป็นต้องใช้กับบาร์โค้ดที่มีลักษณะพื้นผิวแบนเรียบเท่านั้น จำเป็นต้องยิงในระยะที่ไม่ห่างจากตัวบาร์โค้ดมากเกินไป 1 นิ้ว และไม่สามารถอ่านบาร์โค้ดที่กว้างกว่าผิวหน้านำเข้าของตัวเครื่องได้



รูปที่ 2. 15 CCD Scanner

- 2 Laser Scanner เป็นเครื่องอ่านบาร์โค้ดที่มีทั้งแบบพกพาติดตัวและการติดตั้งอยู่กับที่ มีข้อดีที่สามารถอ่านข้อมูลบาร์โค้ดในระยะที่ห่างจากตัวบาร์โค้ดได้พอสมควร การยิงจะใช้แสงเลเซอร์ยิงผ่านกระจกและไปตกกระทบที่ตัวบาร์โค้ดเพื่ออ่านข้อมูลจากแสงสะท้อนที่ย้อนกลับมาที่ตัวรับแสง ในการยิงจะเป็นการฉายแสงเลเซอร์ออกมาเป็นเส้นตรงเส้นเดียว มีขนาดเล็ก และความถี่เดียว แสงเลเซอร์จึงไม่กระจายออกไปนอกพื้นที่ที่ต้องการอ่านข้อมูลทำให้สามารถอ่านรหัสที่มีขนาดเล็กได้ดี นอกจากนี้ในหลาย ๆ รุ่นยังสามารถตั้งให้ทำงานโดยอัตโนมัติได้เมื่อมีแถบบาร์โค้ดเคลื่อนผ่านหน้าหัวอ่าน โดยจะประยุกต์ใช้ร่วมกับขาตั้งเครื่องอ่านบาร์โค้ด



รูปที่ 2.16 Laser Scanner

- 3 Omnidirectional Scanner เป็นเครื่องอ่านแบบเลเซอร์ ลักษณะการทำงานเหมือนกัน แต่มีการฉายแสงเลเซอร์ออกมาหลายเส้นหลายทิศทาง มีลักษณะตัดกันไปมาเหมือนใยแมงมุม ซึ่งจะเหมาะกับการอ่านบาร์โค้ดบนสินค้าซึ่งไม่ได้มีการติดตั้งตำแหน่งของบาร์โค้ดในจุดเดียวกันเสมอ ๆ ซึ่งจะช่วยเพิ่มความเร็วในการทำงาน แต่จะมีราคาที่สูงกว่าเครื่อง Laser Scanner จึงมักนิยมใช้ในซูเปอร์มาร์เก็ต หรือห้างสรรพสินค้าขนาดใหญ่





รูปที่ 2.17 Omnidirectional Scanner

- 4 Imager Scanner เป็นเครื่องอ่านที่ใช้หลักการในการจับภาพของตัวบาร์โค้ด เช่นเดียวกันกับกล้องถ่ายรูป และใช้เทคนิคการประมวลผลภาพที่ทันสมัยในการถอดรหัสบาร์โค้ด สามารถอ่านบาร์โค้ดที่มีขนาดเล็กมาก ๆ ได้ และสามารถทำงานได้ในระยะห่างจากบาร์โค้ดมากขึ้น แต่จะประมวลผลข้อมูลได้ช้ากว่าเครื่องอ่านแบบเลเซอร์อยู่เล็กน้อย



รูปที่ 2.18 Imager Scanner

เครื่องอ่านบาร์โค้ดยังสามารถแบ่งได้ตามลักษณะของการเคลื่อนย้าย คือในบางรุ่นสามารถเคลื่อนย้ายเครื่องอ่านบาร์โค้ดได้แบบไร้สาย (wireless scanner) โดยมีการทำงานเหมือนโทรศัพท์ไร้สายภายในบ้าน ในบางรุ่นเป็นเครื่องอ่านแบบมีสายที่ต้องประจำอยู่กับคอมพิวเตอร์แต่น้ำหนักเบาและสามารถขยับเคลื่อนย้ายเพื่อใช้งานได้ตามความต้องการเป็นเครื่องอ่านที่มีลักษณะเป็นรูปปืนที่มีปุ่มยิง และในบางรุ่นเป็นเครื่องอ่านชนิดตั้งโต๊ะติดอยู่กับที่ ในการทำงานไม่สามารถเคลื่อนย้ายได้แต่จะทำการยิงโดยอัตโนมัติเมื่อมีบาร์โค้ดผ่านหัวอ่าน

## 2.3 หลักการรับความต้องการ

### 2.3.1 การรับความต้องการ (Requirement Gathering)

การรับความต้องการเป็นขั้นตอนแรกที่สำคัญในกระบวนการพัฒนาระบบคอมพิวเตอร์มีวัตถุประสงค์เพื่อรวบรวมและทำความเข้าใจความต้องการของผู้ใช้งานหรือองค์กรอย่างครบถ้วนและถูกต้อง ความต้องการที่ได้จะถูกนำมาใช้ในการกำหนดคุณลักษณะและฟังก์ชันการทำงานของระบบ เพื่อให้ระบบที่พัฒนาขึ้นสามารถตอบสนองต่อความต้องการและปัญหาที่แท้จริงของผู้ใช้งานได้ การวิเคราะห์และจำแนกผู้ที่มีส่วนได้ส่วนเสียอย่างครบถ้วนและครอบคลุมทุกฝ่าย ทุกแผนก ตลอดจนรับความต้องการที่มีความแม่นยำจากที่มีส่วนได้ส่วนเสียทุกภาคส่วนอย่างครบถ้วนจะช่วยลดข้อผิดพลาดทั้งในกระบวนการออกแบบและการพัฒนาระบบ ซึ่งจะช่วยเสริมสร้างความมั่นใจว่าผลลัพธ์ที่ได้จะตรงตามเป้าหมายและความสำคัญที่ผลของระบบที่ตั้งไว้อย่างมีประสิทธิภาพ โดยเนื้อหารายละเอียดสามารถอธิบายได้ดังนี้

#### ● หลักการและแนวทางในการรับความต้องการ

##### 1. การสัมภาษณ์ (Interviews)

การสัมภาษณ์เป็นกระบวนการพูดคุยกับผู้ใชหรือผู้มีส่วนได้เสียเพื่อรวบรวมความต้องการ ปัญหา และความคาดหวังเกี่ยวกับระบบที่ต้องการพัฒนา

##### ประเภทของการสัมภาษณ์

- การสัมภาษณ์แบบมีโครงสร้าง (Structured Interview) มีคำถามที่กำหนดไว้ล่วงหน้า
- การสัมภาษณ์แบบไม่มีโครงสร้าง (Unstructured Interview) เปิดโอกาสให้สนทนาอย่างอิสระ
- การสัมภาษณ์แบบกึ่งโครงสร้าง (Semi-Structured Interview) ผสมผสานทั้งสองรูปแบบ

##### ข้อดี

- ได้รับข้อมูลเชิงลึกจากมุมมองของผู้ใช้
- สามารถเปิดเผยข้อมูลที่ไม่ได้ถูกระบุในเอกสารหรือการสังเกตการณ์

##### ข้อจำกัด

- ใช้เวลามากและต้องการทักษะการสื่อสารที่ดี
- อาจมีอคติจากผู้สัมภาษณ์ที่ส่งผลต่อการตีความคำตอบ

##### 2. การประชุมเชิงปฏิบัติการ (Workshops)

การประชุมเชิงปฏิบัติการเป็นการรวมกลุ่มผู้มีส่วนได้ส่วนเสียจากหลากหลายฝ่ายเพื่อระดมความคิดเห็นและข้อเสนอแนะเกี่ยวกับระบบที่ต้องการพัฒนา

### กระบวนการจัดประชุม

- กำหนดวัตถุประสงค์ของการประชุม
- คัดเลือกผู้เข้าร่วมจากฝ่ายต่าง ๆ
- ใช้เครื่องมือ เช่น กระดานไวท์บอร์ดหรือซอฟต์แวร์ระดมความคิด
- กระตุ้นให้ทุกคนในกลุ่มมีส่วนร่วมในการแสดงความคิดเห็น

#### ข้อดี

- ส่งเสริมการสื่อสารและความเข้าใจร่วมกัน
- สร้างสภาพแวดล้อมที่เปิดโอกาสให้เกิดไอเดียใหม่ ๆ

#### ข้อจำกัด

- อาจเกิดความขัดแย้งที่ต้องจัดการ
- การระดมความคิดเห็นอาจนำไปสู่ข้อสรุปที่ไม่ตรงกับความต้องการที่แท้จริงของผู้ใช้งานทั้งหมด

### 3. การสำรวจ (Surveys)

การสำรวจใช้แบบสอบถามเพื่อรวบรวมข้อมูลจากกลุ่มเป้าหมายจำนวนมากในเวลาอันสั้น

#### องค์ประกอบของแบบสอบถาม

- คำถามปลายปิด เพื่อรวบรวมข้อมูลเชิงปริมาณ เช่น ตัวเลือก “ใช่” หรือ “ไม่”
- คำถามปลายเปิด เพื่อให้ผู้ตอบสามารถแสดงความคิดเห็นเชิงลึก
- คำถามเชิงประเมิน ใช้เก็บข้อมูลเพื่อประเมินความพึงพอใจหรือความสำคัญของประเด็นต่าง ๆ

#### ข้อดี

- ได้ข้อมูลจากกลุ่มใหญ่ในเวลาอันสั้น
- สามารถนำข้อมูลมาวิเคราะห์ทางสถิติได้

#### ข้อจำกัด

- อาจขาดความลึกซึ้งหากแบบสอบถามไม่ถูกออกแบบมาอย่างดี
- ผู้ตอบบางรายอาจไม่ให้ข้อมูลที่ตรงกับความเป็นจริง

### 4. การศึกษาระบบเดิม (Existing System Study)

การศึกษาระบบเดิมช่วยให้เข้าใจโครงสร้างและฟังก์ชันที่สามารถพัฒนาต่อยอดได้

#### กระบวนการศึกษาระบบเดิม

- การตรวจสอบเอกสารที่เกี่ยวข้องกับระบบเดิม
- การสัมภาษณ์ผู้ดูแลระบบหรือผู้ใช้งานปัจจุบัน
- การทดสอบระบบเดิมเพื่อค้นหาข้อดีและข้อจำกัด

#### ข้อดี

- ช่วยระบุจุดเด่นและจุดอ่อนของระบบเดิม

### ข้อจำกัด

- อาจไม่ครอบคลุมความต้องการใหม่ที่ยังไม่ได้ระบุ

## 5. การสังเกตการณ์ (Observation)

ศึกษาพฤติกรรมการทำงานจริงของผู้ใช้เพื่อนำมาวิเคราะห์

### รูปแบบการสังเกตการณ์

- Direct Observation ฝึ่สังเกตโดยตรงโดยไม่รบกวนการทำงาน
- Participant Observation ผู้สังเกตการณ์เข้าร่วมกิจกรรมด้วย

### ข้อดี

- ได้ข้อมูลพฤติกรรมจริงของผู้ใช้
- สามารถสังเกตพฤติกรรมที่อาจไม่สามารถได้จากการสัมภาษณ์หรือแบบสอบถาม

### ข้อจำกัด

- อาจใช้เวลานานในการศึกษาพฤติกรรม
- มีผลกระทบจากอคติของผู้สังเกตการณ์ ซึ่งอาจส่งผลต่อการตีความพฤติกรรม

## 6. การสร้าง Use Case

การสร้าง Use Case ช่วยจำลองสถานการณ์การใช้งานระบบในลักษณะเป็นกรณีศึกษาเพื่อทำความเข้าใจลำดับขั้นตอนการทำงานและการโต้ตอบกับระบบ

### องค์ประกอบของ Use Case

- ผู้ใช้งาน (Actor) ผู้ที่มีปฏิสัมพันธ์กับระบบ
- กรณีการใช้งาน (Use Case) ลำดับการทำงานที่แสดงถึงการโต้ตอบกับระบบ
- ขอบเขต (Scope) ระบุขอบเขตของการทำงานในแต่ละกรณี

### ข้อดี

- ช่วยให้เห็นภาพการใช้งานจริงของระบบ

### ข้อจำกัด

- ต้องใช้เวลาในการวิเคราะห์และเขียน Use Case อย่างละเอียด

## 7. การระบุผู้มีส่วนได้ส่วนเสีย (Stakeholder Identification)

การระบุผู้มีส่วนได้เสียมีความสำคัญเพื่อกำหนดกลุ่มผู้ใช้และผู้ที่มีผลกระทบต่อโครงการ

### ประเภทของผู้มีส่วนได้เสีย

- Primary Stakeholders ผู้ใช้หลักของระบบ
- Secondary Stakeholders ผู้ที่ได้รับผลกระทบทางอ้อมจากการใช้งานระบบ
- Key Stakeholders ผู้ที่มีอำนาจตัดสินใจในโครงการ

### ข้อดี

- ช่วยให้นั่นใจว่าความต้องการของทุกฝ่ายถูกนำมาพิจารณา
- สามารถบริหารจัดการความคาดหวังและผลกระทบจากผู้มีส่วนได้เสียได้

### ข้อจำกัด

- อาจมีความขัดแย้งระหว่างกลุ่มผู้มีส่วนได้เสียที่ต้องจัดการ
- บางครั้งกลุ่มผู้มีส่วนได้เสียอาจไม่เห็นด้วยกับการตัดสินใจที่ต้องการทำ

### ●การจัดทำเอกสารความต้องการ

ผลลัพธ์จากการรับความต้องการต้องถูกบันทึกในรูปแบบเอกสารที่ชัดเจนและครบถ้วน เช่น เอกสาร SRS (Software Requirements Specification) ซึ่งครอบคลุมถึง

- ข้อกำหนดเชิงหน้าที่ (Functional Requirements) ระบุฟังก์ชันที่ระบบต้องทำ เช่น การบันทึกข้อมูล การค้นหา หรือการรายงานผล
- ข้อกำหนดที่ไม่ใช่เชิงหน้าที่ (Non-Functional Requirements) ระบุข้อกำหนดด้านประสิทธิภาพ ความปลอดภัย และความสามารถในการปรับขยาย เช่น ความเร็วในการประมวลผลและความปลอดภัยของข้อมูล
- ข้อจำกัด (Constraints) ระบุข้อจำกัดที่เกี่ยวข้อง เช่น งบประมาณ เวลา หรือทรัพยากรที่มีอยู่

### ●ความสำคัญของการรับความต้องการที่แม่นยำ

1. ลดความเสี่ยงของความเข้าใจผิดระหว่างทีมพัฒนาและผู้ใช้งาน
2. เพิ่มโอกาสในการพัฒนาระบบที่ตอบสนองความต้องการได้อย่างแท้จริง
3. ช่วยวางแผนการพัฒนาและการจัดสรรทรัพยากรได้อย่างมีประสิทธิภาพ
4. ลดต้นทุนในการแก้ไขปัญหาที่เกิดจากข้อผิดพลาดในขั้นตอนหลัง
5. สร้างความพึงพอใจให้กับผู้ใช้งาน โดยระบบสามารถตอบสนองต่อความต้องการได้อย่างถูกต้องและครบถ้วน

## 2.4 หลักการการวิเคราะห์และออกแบบระบบคอมพิวเตอร์

การวิเคราะห์และออกแบบระบบคอมพิวเตอร์เป็นกระบวนการที่สำคัญและจำเป็นในงานพัฒนาระบบคอมพิวเตอร์ โดยมีจุดมุ่งหมายเพื่อทำความเข้าใจปัญหาหรือข้อกำหนดของระบบ รวมถึงการออกแบบโครงสร้างและฟังก์ชันต่าง ๆ ของระบบให้มีประสิทธิภาพและตอบสนองความต้องการของผู้ใช้งาน การดำเนินการในขั้นตอนนี้จะช่วยให้ระบบที่พัฒนามีความเหมาะสมและสามารถดำเนินงานได้ง่าย โดยหัวข้อที่จะนำเสนอประกอบไปด้วย

### 2.4.1 การวิเคราะห์ระบบ (System Analysis)

การวิเคราะห์ระบบเป็นกระบวนการที่มุ่งเน้นในการศึกษาลักษณะการทำงานของระบบและปัญหาที่เกิดขึ้น เริ่มจากการวิเคราะห์และจำแนกผู้ที่มีส่วนได้ส่วนเสีย จากนั้นทำการรวบรวมข้อมูลความต้องการจากกลุ่มผู้มีส่วนได้ส่วนเสียเหล่านั้นเพื่อทำความเข้าใจสภาพการทำงานปัจจุบันและปัญหาที่เกิดขึ้นกับระบบ พร้อมทั้งหาวิธีการในการปรับปรุงแก้ไข กระบวนการนี้ประกอบด้วย

กิจกรรมหลายประการที่ช่วยให้การออกแบบระบบที่มีประสิทธิภาพเพื่อตอบสนองต่อความต้องการของผู้ใช้งานได้อย่างถูกต้องและตรงตามวัตถุประสงค์

#### 2.4.1.1 การวิเคราะห์เชิงฟังก์ชัน (Functional Analysis)

การวิเคราะห์เชิงฟังก์ชันเน้นไปที่การระบุและแยกแยะหน้าที่การทำงานที่ระบบจะต้องปฏิบัติ การเข้าใจฟังก์ชันต่าง ๆ ที่ระบบต้องการจะช่วยในการออกแบบโครงสร้างและวิธีการทำงานของระบบอย่างมีระเบียบ โดยการวิเคราะห์ฟังก์ชันจะช่วยให้ทราบว่า ระบบต้องทำอะไร เช่น การคำนวณข้อมูล การเก็บข้อมูล หรือการแสดงผล

##### ●วิธีการที่ใช้ในการวิเคราะห์ฟังก์ชันมีดังนี้

- การสร้างรายการฟังก์ชัน (Function List) เป็นการระบุฟังก์ชันหลัก ๆ ที่ระบบต้องมี ซึ่งอาจรวมถึงการจัดการข้อมูล การคำนวณ การให้บริการต่าง ๆ และการแสดงผล
- การประเมินความสำคัญของฟังก์ชันแต่ละฟังก์ชันจะได้รับการประเมินความสำคัญว่าเกี่ยวข้องกับเป้าหมายของระบบอย่างไร
- การใช้แผนภาพการไหลของข้อมูล แผนภาพการไหลของข้อมูลเป็นเครื่องมือที่ช่วยในการแสดงการไหลของข้อมูลระหว่างกระบวนการต่าง ๆ ภายในระบบ

#### 2.4.1.2 การวิเคราะห์การไหลของข้อมูล (Data Flow Analysis : DFA)

แผนภาพการไหลของข้อมูลใช้ในการแสดงการไหลของข้อมูลและความสัมพันธ์ระหว่างกระบวนการต่าง ๆ ภายในระบบ การใช้แผนภาพการไหลของข้อมูลจะช่วยให้เห็นภาพรวมของการทำงานและการไหลของข้อมูลในระบบ รวมถึงความสัมพันธ์ระหว่างแหล่งข้อมูล กระบวนการที่ประมวลผลข้อมูล และแหล่งเก็บข้อมูลต่าง ๆ

##### ●องค์ประกอบของ DFD ประกอบด้วย

- กระบวนการ (Process) ตัวแทนของกิจกรรมที่ระบบต้องทำ เช่น การคำนวณ การแสดงผล
- แหล่งข้อมูล (Source/Sink) จุดที่ข้อมูลเข้ามาและออกจากระบบ
- การไหลของข้อมูล (Data Flow) เส้นทางที่ข้อมูลเคลื่อนที่ไปในกระบวนการต่าง ๆ
- ที่เก็บข้อมูล (Data Store) จุดที่ข้อมูลถูกจัดเก็บและเรียกใช้ภายในระบบ

#### 2.4.1.3 การวิเคราะห์ความสัมพันธ์ของข้อมูล

การใช้แผนภาพความสัมพันธ์ระหว่างเอนทิตีช่วยให้สามารถแสดงความสัมพันธ์ระหว่างเอนทิตี และข้อมูลต่าง ๆ ในระบบได้อย่างชัดเจน แผนภาพความสัมพันธ์ระหว่างเอนทิตีเป็นเครื่องมือที่สำคัญในการออกแบบฐานข้อมูล ซึ่งจะช่วยให้ทราบว่าแต่ละข้อมูลมีความสัมพันธ์กันอย่างไร และต้องมีการจัดการอย่างไรเพื่อให้ฐานข้อมูลทำงานได้อย่างมีประสิทธิภาพ

### ●องค์ประกอบของแผนภาพความสัมพันธ์ระหว่างเอนทิตีประกอบด้วย

- เอนทิตี (Entity) ตัวแทนของข้อมูลหรือวัตถุในระบบ เช่น ผู้ใช้งาน สินค้า
- แอตทริบิวต์ (Attribute) คุณสมบัติของเอนทิตี เช่น ชื่อผู้ใช้งาน ราคา
- ความสัมพันธ์ (Relationship) การเชื่อมโยงระหว่างเอนทิตี เช่น ผู้ใช้งานซื้อสินค้า
- ข้อจำกัด (Constraints) กฎเกณฑ์ที่ควบคุมความสัมพันธ์ระหว่างเอนทิตี เช่น ข้อจำกัดเกี่ยวกับการเชื่อมโยงข้อมูล

## 2.4.2 การออกแบบระบบ (System Design)

การออกแบบระบบเป็นกระบวนการที่นำผลจากการวิเคราะห์ระบบมาสร้างโครงสร้างและสถาปัตยกรรมของระบบ โดยมุ่งเน้นในการกำหนดวิธีการทำงานที่เหมาะสมของระบบ เพื่อให้สามารถพัฒนาและนำไปใช้งานได้จริง การออกแบบระบบที่ดีจะช่วยให้ระบบทำงานได้อย่างมีประสิทธิภาพ ลดความซับซ้อนในการพัฒนา และลดข้อผิดพลาดที่อาจเกิดขึ้น

### ●การออกแบบเชิงโครงสร้าง (Structural Design)

การออกแบบเชิงโครงสร้างเน้นในการกำหนดโครงสร้างและลำดับขั้นตอนของกระบวนการต่าง ๆ ในระบบ เพื่อให้สามารถจัดการและควบคุมการทำงานของระบบได้อย่างมีประสิทธิภาพ กระบวนการนี้จะช่วยในการจัดระเบียบระบบและแบ่งแยกงานต่าง ๆ ให้ชัดเจน องค์ประกอบที่สำคัญในการออกแบบเชิงโครงสร้างมีดังนี้

- การสร้างผังงานระบบ (System Flowchart) ใช้ในการแสดงลำดับขั้นตอนของการทำงานในระบบ
- การจัดระเบียบระบบเป็นโมดูล การแบ่งระบบออกเป็นโมดูลที่สามารถพัฒนาและทดสอบได้แยกจากกัน

### ●การออกแบบฐานข้อมูล (Database Design)

การออกแบบฐานข้อมูลมุ่งเน้นไปที่การจัดโครงสร้างข้อมูลให้มีประสิทธิภาพในการจัดเก็บและเข้าถึงข้อมูลอย่างรวดเร็ว การสร้างแบบจำลองข้อมูล (Data Modeling) และการเลือกใช้คีย์หลักและคีย์ต่าง ๆ ในฐานข้อมูลจะช่วยให้ระบบทำงานได้เร็วขึ้นและมีความแม่นยำในการเข้าถึงข้อมูล

### ●การออกแบบส่วนติดต่อผู้ใช้งาน (User Interface Design)

การออกแบบส่วนติดต่อผู้ใช้งานมีความสำคัญในการทำให้ผู้ใช้งานสามารถใช้งานระบบได้อย่างสะดวกและไม่ซับซ้อน การออกแบบ UI ต้องคำนึงถึงการจัดเรียงองค์ประกอบต่าง ๆ บนหน้าจอ การเลือกใช้สี การจัดการข้อความ และการทำให้ระบบสามารถใช้งานได้ง่ายและมีความสวยงาม

### ●การออกแบบสถาปัตยกรรมระบบ (System Architecture Design)

การออกแบบสถาปัตยกรรมระบบมุ่งเน้นในการกำหนดโครงสร้างทางเทคนิคที่รองรับการทำงานของระบบ เช่น การกำหนดประเภทของเซิร์ฟเวอร์ เครือข่าย และอุปกรณ์ที่จำเป็นสำหรับการดำเนินการ การออกแบบสถาปัตยกรรมจะช่วยให้ระบบมีความยืดหยุ่นและสามารถขยายตัว

## ●ประเภทของสถาปัตยกรรม

### 1. สถาปัตยกรรมแบบรวมศูนย์ (Centralized Architecture)

- ลักษณะ: ทุกข้อมูลและกระบวนการจะถูกรวบรวมและดำเนินการในศูนย์กลาง เช่น เซิร์ฟเวอร์หลัก หรือระบบคลาวด์ที่จัดการข้อมูลและบริการทั้งหมดจากจุดเดียว
- ข้อดี: ง่ายต่อการจัดการและดูแลรักษา เนื่องจากการควบคุมทั้งหมดจากจุดเดียว
- ข้อเสีย: หากเกิดความล้มเหลวที่ศูนย์กลางอาจทำให้ระบบทั้งหมดหยุดทำงาน

### 2. สถาปัตยกรรมแบบกระจาย (Distributed Architecture)

- ลักษณะ: ระบบประกอบด้วยหลาย ๆ หน่วยงานที่ทำงานร่วมกันเพื่อให้บริการ ซึ่งสามารถกระจายไปตามหลาย ๆ เซิร์ฟเวอร์หรือหลาย ๆ ตำแหน่ง
- ข้อดี: ช่วยเพิ่มความเสถียรและความสามารถในการขยายระบบได้ดีกว่า เนื่องจากไม่พึ่งพาศูนย์กลาง
- ข้อเสีย: ต้องการการบริหารจัดการที่ซับซ้อน เนื่องจากมีหลายส่วนต้องคอยตรวจสอบและปรับปรุง

### 3. สถาปัตยกรรมบนคลาวด์ (Cloud-Based Architecture)

- ลักษณะ: ระบบใช้บริการจากผู้ให้บริการคลาวด์ในการจัดการข้อมูลและการทำงานทั้งหมด ซึ่งช่วยลดภาระการจัดการทรัพยากรภายในองค์กร
- ข้อดี: ความยืดหยุ่นในการขยายระบบและการปรับขนาดตามความต้องการ
- ข้อเสีย: ขึ้นอยู่กับการเชื่อมต่ออินเทอร์เน็ตและอาจมีปัญหาด้านความปลอดภัยในการจัดเก็บข้อมูล

## 2.4.3 ความสำคัญของการวิเคราะห์และออกแบบระบบ

1. ลดความซับซ้อนของปัญหาที่อาจเกิดขึ้นการวิเคราะห์และออกแบบระบบจะช่วยแยกแยะปัญหาหรือข้อกำหนดต่าง ๆ ออกเป็นส่วน ๆ ทำให้สามารถจัดการและเข้าใจปัญหาต่าง ๆ ได้ง่ายขึ้น ลดความซับซ้อนที่อาจเกิดขึ้นในกระบวนการพัฒนา
2. ช่วยให้การพัฒนาระบบมีประสิทธิภาพเมื่อมีการวิเคราะห์และออกแบบที่ดี ระบบจะสามารถทำงานได้ตามที่ต้องการและมีประสิทธิภาพสูงสุด เพราะจะมีการวางแผนการทำงานที่เหมาะสมและไม่เกิดปัญหาที่ไม่คาดคิดระหว่างการพัฒนา
3. ตอบสนองความต้องการของผู้ใช้งานได้ดีขึ้นการวิเคราะห์ความต้องการของผู้ใช้งานในขั้นต้น จะช่วยให้ระบบที่พัฒนาขึ้นสามารถตอบสนองต่อความต้องการเหล่านั้นได้อย่างแม่นยำและตรงจุด ทำให้ผู้ใช้งานสามารถใช้งานระบบได้อย่างเต็มประสิทธิภาพ
4. ลดต้นทุนในการพัฒนาการออกแบบระบบที่ดีช่วยให้กระบวนการพัฒนามีความชัดเจนตั้งแต่เริ่มต้น จึงช่วยลดเวลาและทรัพยากรที่จำเป็นในการพัฒนา ลดการทำงานซ้ำซ้อนและการแก้ไขที่เกิดขึ้นในภายหลัง ซึ่งจะช่วยลดต้นทุนโดยรวม



5. เพิ่มความสามารถในการป้องกันข้อผิดพลาดการวิเคราะห์และออกแบบระบบช่วยให้สามารถมองเห็นปัญหาที่อาจเกิดขึ้นตั้งแต่ขั้นตอนแรก และทำการป้องกันไม่ให้เกิดข้อผิดพลาดในระหว่างการพัฒนา เช่น การออกแบบที่ไม่เหมาะสม หรือการจัดการข้อมูลผิดพลาด

## 2.5 ทฤษฎีด้านการออกแบบหน้าบ้าน

การออกแบบหน้าบ้านเป็นกระบวนการที่สำคัญในการพัฒนาเว็บไซต์หรือแอปพลิเคชัน ซึ่งเกี่ยวข้องกับการออกแบบส่วนที่ผู้ใช้สามารถมองเห็นและโต้ตอบได้โดยตรง เช่น รูปแบบการจัดวางข้อมูล การใช้สี การจัดเรียงเนื้อหา และองค์ประกอบกราฟิกต่าง ๆ ซึ่งมีผลต่อประสบการณ์การใช้งานของผู้ใช้ (User Experience : UX) และความสะดวกในการใช้งาน (User Interface : UI) การออกแบบหน้าบ้านที่ดีนั้นจะช่วยให้ผู้ใช้สามารถเข้าถึงข้อมูลและฟังก์ชันต่าง ๆ ของระบบได้อย่างสะดวกและรวดเร็ว

### ● หลักการของการออกแบบหน้าบ้าน

#### 1. การออกแบบประสบการณ์ผู้ใช้ (User Experience Design : UX)

การออกแบบประสบการณ์ผู้ใช้คือการมุ่งเน้นในการพัฒนาและปรับปรุงประสบการณ์ที่ผู้ใช้ได้รับจากการใช้งานระบบต่าง ๆ โดยจะพิจารณาถึงความสะดวกสบายในการใช้งาน และการจัดเรียงข้อมูลให้ผู้ใช้สามารถเข้าถึงได้ง่าย การออกแบบประสบการณ์ผู้ใช้จะช่วยให้ผู้ใช้สามารถทำกิจกรรมต่าง ๆ ได้อย่างมีประสิทธิภาพและลดความสับสนที่อาจเกิดขึ้นระหว่างการใช้งาน ระบบที่มีการออกแบบประสบการณ์ผู้ใช้ที่ดีจะช่วยเพิ่มความพึงพอใจของผู้ใช้

#### 2. การออกแบบอินเทอร์เฟซผู้ใช้ (User Interface Design : UI)

การออกแบบอินเทอร์เฟซผู้ใช้คือการออกแบบที่เกี่ยวข้องกับการนำเสนอข้อมูลในลักษณะกราฟิกที่ผู้ใช้สามารถเห็นและโต้ตอบได้ เช่น ปุ่ม เมนู และฟอร์มต่าง ๆ การออกแบบอินเทอร์เฟซผู้ใช้ จำเป็นต้องคำนึงถึงความสะดวกในการใช้งานและความสวยงามขององค์ประกอบต่าง ๆ ที่ประกอบกันในระบบ ทั้งนี้ เพื่อให้ผู้ใช้สามารถใช้งานได้ง่ายและสะดวกยิ่งขึ้น การออกแบบอินเทอร์เฟซผู้ใช้ ที่ดีควรคำนึงถึงการใช้สีที่เหมาะสม การจัดเรียงเนื้อหาที่เข้าใจง่าย และการใช้งานที่ไม่ยุ่งยาก

#### 3. การออกแบบที่ตอบสนองต่ออุปกรณ์ (Responsive Design)

การออกแบบที่ตอบสนองต่ออุปกรณ์หมายถึงการปรับเปลี่ยนรูปแบบการแสดงผลของระบบให้เหมาะสมกับอุปกรณ์ที่แตกต่างกัน เช่น คอมพิวเตอร์ แท็บเล็ต หรือสมาร์ทโฟน โดยการออกแบบลักษณะนี้จะช่วยให้ผู้ใช้สามารถเข้าถึงและใช้งานเว็บไซต์หรือแอปพลิเคชันได้อย่างมีประสิทธิภาพไม่ว่าจะใช้อุปกรณ์ใด การออกแบบที่ตอบสนองช่วยให้หน้าเว็บหรือแอปพลิเคชันสามารถปรับ

ขนาดและการจัดเรียงข้อมูลตามขนาดหน้าจอได้อย่างอัตโนมัติ ซึ่งจะช่วยทำให้ผู้ใช้ไม่รู้สึกรำคาญในการใช้งาน

#### 4. การเข้าถึงและความสามารถในการใช้งาน (Accessibility and Usability)

การออกแบบหน้าบ้านต้องคำนึงถึงการเข้าถึงที่เหมาะสมสำหรับทุกกลุ่มผู้ใช้ รวมถึงผู้ที่มีข้อจำกัดด้านการมองเห็น การได้ยิน หรือการเคลื่อนไหว ตัวอย่างเช่น การใช้สีที่มีความคมชัดและตัดกันชัดเจน การรองรับเครื่องมือช่วยเหลือการเข้าถึง และการปรับขนาดข้อความเพื่อรองรับผู้ใช้ที่มีปัญหาการมองเห็น การออกแบบที่สามารถเข้าถึงได้จะช่วยให้ผู้ใช้ทุกคนสามารถใช้งานเว็บไซต์หรือแอปพลิเคชันได้อย่างมีประสิทธิภาพ

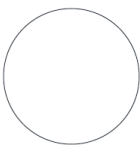


#### 5. การออกแบบที่มุ่งเน้นผลลัพธ์ (Goal-Oriented Design)

การออกแบบหน้าบ้านที่มุ่งเน้นผลลัพธ์หมายถึงการออกแบบที่มีเป้าหมายเพื่อทำให้ผู้ใช้สามารถบรรลุเป้าหมายของตนได้โดยเร็วที่สุด เช่น หากเป็นเว็บไซต์ร้านค้าออนไลน์ การออกแบบหน้าบ้านต้องช่วยให้ผู้ใช้สามารถเลือกสินค้าหรือทำการสั่งซื้อได้อย่างรวดเร็วและไม่ยุ่งยาก หรือถ้าเป็นระบบค้นหาข้อมูล ต้องช่วยให้ผู้ใช้ค้นหาข้อมูลได้อย่างแม่นยำและสะดวก การออกแบบที่มุ่งเน้นผลลัพธ์จะช่วยให้ผู้ใช้สามารถทำกิจกรรมต่าง ๆ ได้สำเร็จตามวัตถุประสงค์


##### 2.5.1 ทฤษฎีการออกแบบแผนภาพบริบท (Context Diagram)

แผนภาพบริบทหรือแผนภาพการไหลของข้อมูลระดับที่ 0 เป็นเครื่องมือที่สำคัญในการออกแบบและวิเคราะห์ระบบคอมพิวเตอร์ ซึ่งใช้ในการแสดงความสัมพันธ์ระหว่างระบบและสภาพแวดล้อมภายนอกที่มีผลต่อการทำงานของระบบ แผนภาพนี้ช่วยให้ผู้พัฒนาระบบเข้าใจและกำหนดขอบเขตของระบบ รวมถึงการระบุแหล่งข้อมูลและการเชื่อมโยงที่เกิดขึ้นระหว่างระบบและผู้ใช้หรือหน่วยงานอื่น ๆ

ตารางที่ 2.2 สัญลักษณ์ต่างๆ ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์	ความหมาย
	ระบบ (System or Application)
	หน่วยงาน / ตัวแปรภายนอก (External Entity Symbol)
	หน่วยงาน/ตัวแปรภายนอกที่ซ้ำ (Duplication External Entity Symbol)

ตารางที่ 2.2 สัญลักษณ์ต่างๆ ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

สัญลักษณ์	ความหมาย
	การไหลของข้อมูล (Data Flow Symbol)

### ● ความสำคัญของแผนภาพบริบท

#### 1. การกำหนดขอบเขตของระบบ

แผนภาพบริบทช่วยให้ทีมพัฒนาระบบสามารถกำหนดขอบเขตของระบบได้อย่างชัดเจน โดยแสดงว่าอะไรคือส่วนที่เป็นระบบและอะไรที่อยู่นอกขอบเขตของระบบ

#### 2. การเข้าใจความสัมพันธ์ระหว่างระบบและบริบทภายนอก (สภาพแวดล้อมภายนอก)

แผนภาพบริบทช่วยให้สามารถแสดงความสัมพันธ์ที่ชัดเจนระหว่างระบบและสภาพแวดล้อมภายนอกหรือผู้ใช้งาน ซึ่งเป็นข้อมูลสำคัญในการกำหนดว่าใครมีบทบาทในการใช้งานระบบ และข้อมูลอะไรที่ถูกแลกเปลี่ยน

#### 3. การสื่อสารกับผู้ที่เกี่ยวข้อง

แผนภาพบริบทช่วยให้การสื่อสารระหว่างทีมพัฒนาและผู้มีส่วนได้ส่วนเสียเป็นไปอย่างมีประสิทธิภาพ โดยช่วยให้ทุกฝ่ายเข้าใจภาพรวมของระบบและวิธีการที่ระบบจะเชื่อมโยงกับส่วนอื่น ๆ

### ● หลักการออกแบบแผนภาพบริบท

การออกแบบแผนภาพบริบทควรคำนึงถึงหลักการและแนวทางต่าง ๆ ที่ช่วยให้การสื่อสารและแสดงข้อมูลเป็นไปอย่างชัดเจนและมีประสิทธิภาพ

1. การระบุระบบหลัก แผนภาพบริบทควรเริ่มต้นด้วยการระบุระบบหลัก (หรือระบบที่กำลังพัฒนา) ซึ่งมักจะแสดงเป็นวงกลมหรือสี่เหลี่ยมที่อยู่กลางแผนภาพ ระบบนี้จะถูกเชื่อมต่อกับส่วนอื่น ๆ ภายนอก

2. การระบุส่วนภายนอกที่เชื่อมต่อ ส่วนที่อยู่ภายนอกระบบ เช่น ผู้ใช้งาน ระบบอื่น ๆ หน่วยงานที่เกี่ยวข้อง หรือเครื่องมืออุปกรณ์ ซึ่งต้องได้รับการระบุและเชื่อมโยงกับระบบหลักอย่างชัดเจน การแสดงความสัมพันธ์นี้จะช่วยให้เข้าใจถึงวิธีการแลกเปลี่ยนข้อมูลระหว่างกัน

3. การระบุข้อมูลที่ไหลระหว่างระบบและส่วนภายนอก แผนภาพบริบทควรแสดงถึงข้อมูลที่ถูกแลกเปลี่ยนระหว่างระบบและหน่วยงานภายนอก เช่น การส่งข้อมูลจากผู้ใช้ไปยังระบบ หรือการดึงข้อมูลจากระบบภายนอกเข้ามาในระบบหลัก

4. การใช้เครื่องหมายที่เข้าใจง่าย เครื่องหมายที่ใช้ในแผนภาพควรเป็นสัญลักษณ์ที่เข้าใจง่าย เช่น ลูกศรที่แสดงการไหลของข้อมูล หรือสัญลักษณ์ที่แสดงถึงระบบและหน่วยงานภายนอก การใช้สัญลักษณ์มาตรฐานจะช่วยให้แผนภาพอ่านง่ายและไม่ซับซ้อน

### ●ประโยชน์ของแผนภาพบริบท

- ช่วยให้เข้าใจภาพรวมของระบบ แผนภาพบริบทช่วยให้ทีมพัฒนาและผู้มีส่วนได้ส่วนเสียสามารถเห็นภาพรวมของระบบที่กำลังพัฒนา รวมถึงการเชื่อมโยงกับระบบและผู้ใช้ภายนอก
- ช่วยในการวิเคราะห์ข้อกำหนดของระบบ โดยการแสดงความสัมพันธ์ระหว่างระบบและส่วนต่าง ๆ ภายนอก ทำให้สามารถระบุข้อกำหนดและฟังก์ชันที่จำเป็นได้อย่างถูกต้อง
- การพัฒนาและปรับปรุงระบบ เมื่อมีแผนภาพบริบทที่ชัดเจน สามารถใช้ในการพัฒนาและปรับปรุงระบบได้ง่ายขึ้น เพราะจะเข้าใจถึงความสัมพันธ์ระหว่างระบบกับส่วนต่าง ๆ ที่ต้องการ
- การสื่อสารที่มีประสิทธิภาพ แผนภาพบริบทช่วยให้การสื่อสารระหว่างทีมพัฒนา ผู้ใช้ และผู้มีส่วนได้ส่วนเสียอื่น ๆ เป็นไปอย่างมีประสิทธิภาพ โดยสามารถแสดงความต้องการและข้อกำหนดของระบบได้ชัดเจน

### 2.5.2 ทฤษฎีการออกแบบแผนภาพกระแสข้อมูล (Data Flow Diagram : DFD)

แผนภาพกระแสข้อมูลหรือที่เรียกว่า “แผนภาพการไหลของข้อมูล” เป็นเครื่องมือสำคัญที่ใช้ในการแสดงการไหลของข้อมูลและการประมวลผลต่าง ๆ ภายในระบบ โดยแสดงความสัมพันธ์ระหว่างข้อมูลกับแหล่งเก็บข้อมูลต่าง ๆ แผนภาพนี้ช่วยให้การวิเคราะห์ระบบเป็นไปอย่างมีประสิทธิภาพและเข้าใจง่าย พร้อมทั้งช่วยสร้างความเข้าใจร่วมกันระหว่างผู้วิเคราะห์ระบบ ทีมพัฒนาระบบ และผู้ใช้งานระบบ ซึ่งส่งผลให้พัฒนาระบบมีความถูกต้องและตอบโจทย์ความต้องการมากยิ่งขึ้น

### ●ประโยชน์ที่ได้จากการใช้แผนภาพกระแสข้อมูล มีดังนี้

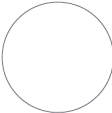







- แผนภาพกระแสข้อมูลมีความอิสระในการใช้งาน โดยไม่ต้องมีเทคนิคอื่นมาช่วย เนื่องจากสามารถใช้สัญลักษณ์ต่าง ๆ แทนสิ่งทีวิเคราะห์มา
- แผนภาพกระแสข้อมูลเป็นสื่อที่ง่ายต่อการแสดงความสัมพันธ์ระหว่างระบบใหญ่และระบบย่อย ซึ่งจะทำให้เข้าใจความสัมพันธ์ต่าง ๆ ได้ดี
- แผนภาพกระแสข้อมูลเป็นสื่อที่ช่วยในการวิเคราะห์ระบบให้เป็นไปได้ง่าย และมีความเข้าใจตรงกันระหว่างผู้วิเคราะห์ระบบเอง หรือระหว่างผู้วิเคราะห์ระบบกับทีมพัฒนาระบบหรือระหว่างผู้วิเคราะห์ระบบกับผู้ใช้งานระบบ
- แผนภาพกระแสข้อมูลช่วยในการวิเคราะห์ระบบให้สะดวกโดยสามารถเห็นข้อมูลและขั้นตอนต่าง ๆ เป็นแผนภาพ

### สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

●สัญลักษณ์ที่ใช้ในการเขียนแผนภาพกระแสข้อมูลนั้น ประกอบด้วย 4 สัญลักษณ์ที่แสดงถึงการประมวลผล การไหลของข้อมูล ส่วนที่ใช้เก็บข้อมูล และสิ่งที่ยื่นออกระบบ โดยได้มีการศึกษาคิดค้นพัฒนาวิธีการอยู่หลายแบบ แต่ที่เป็นมาตรฐานมี 2 กลุ่ม คือกลุ่มที่คิดค้นโดย Gane and

Sarson และกลุ่มของ DeMarco and Yourdon ถึงแม้สัญลักษณ์บางอย่างของสององค์กรนี้จะต่างกัน แต่องค์ประกอบของแผนภาพและหลักการเขียนแผนภาพไม่ได้แตกต่างกัน รายละเอียดดังตารางที่ 2.3

**ตารางที่ 2.3** สัญลักษณ์ต่าง ๆ ที่ใช้ในการเขียนแผนภาพการไหลของข้อมูล

ชื่อสัญลักษณ์	DeMarco & Yourdon symbols	Gane & Sarson symbols
การประมวลผล (Process)		
แหล่งเก็บข้อมูล (Data Store)		
กระแสข้อมูล (Data Flow)		
สิ่งที่อยู่ภายนอก (External Entity)		

เนื่องจากสัญลักษณ์การเขียนแผนภาพกระแสข้อมูล ประกอบด้วยสัญลักษณ์ต่าง ๆ ซึ่งมีความหมายในตัวเอง ดังนั้นการเขียนแผนภาพกระแสข้อมูลจึงต้องมีกฎเกณฑ์ต่าง ๆ เพื่อแสดงถึงความถูกต้องในการเขียนแผนภาพ ดังนี้

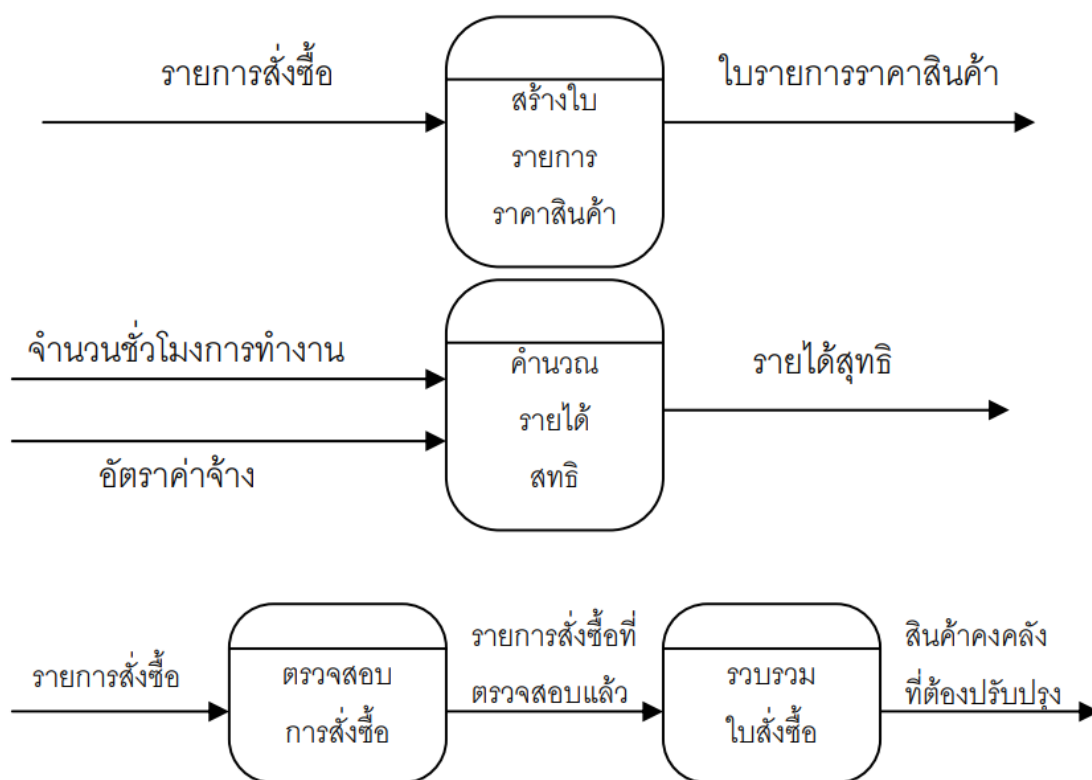
- สัญลักษณ์การประมวลผล (Process Symbol) เป็นการเปลี่ยนแปลงข้อมูลจากรูปแบบหนึ่ง (Input) ไปเป็นอีกรูปแบบหนึ่ง (Output) เช่น การคำนวณรายได้สุทธิของลูกจ้างรายวัน จะต้องประกอบด้วยข้อมูลนำเข้าที่เป็นจำนวนชั่วโมงการทำงานและอัตราค่าจ้างต่อชั่วโมง เมื่อผ่านการประมวลผลแล้วจะได้รายได้สุทธิ

#### หลักการใช้สัญลักษณ์การประมวลผล

- ต้องใช้สัญลักษณ์การประมวลผลคู่กับสัญลักษณ์กระแสข้อมูลเสมอโดยที่ ลูกศรชี้เข้าหมายถึงเป็นข้อมูลนำเข้า ถ้าลูกศรชี้ออกหมายถึงเป็นข้อมูลออกจากการประมวลผล ซึ่ง 1 กระบวนการสามารถมีข้อมูลนำเข้ามากกว่า 1 เส้น หรือข้อมูลออกมากกว่า 1 เส้นได้
- การตั้งชื่อของการประมวลผล ควรเป็นวลีเดียวที่อธิบายการทำงานทั้งหมดได้และควรอธิบายการทำงานอย่างใดอย่างหนึ่งโดยเฉพาะมากกว่าที่จะอธิบายการทำงานอย่างกว้าง ๆ เช่น หาก

แสดงถึงการประมวลผล ตรวจสอบรายการ ควรจะระบุว่าเป็น การตรวจสอบรายการถอนเงิน หรือ ตรวจสอบรายการค่าใช้จ่ายรายสัปดาห์ เป็นต้น

- แต่ละการประมวลผล จะมีแต่ข้อมูลเข้าอย่างเดียว หรือออกอย่างเดียวไม่ได้



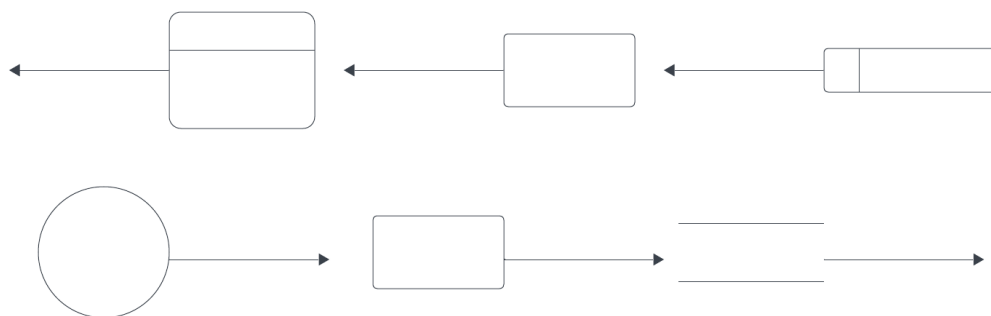
รูปที่ 2.19 ตัวอย่างการใช้สัญลักษณ์การประมวลผล (Process)

●สัญลักษณ์กระแสข้อมูล (Data Flow Symbol) ใช้แทนเส้นทางการไหลของข้อมูลระหว่างส่วนต่าง ๆ ในระบบสารสนเทศ โดยแสดงทิศทางการไหลด้วยลูกศรที่ชี้จากต้นทางไปยังปลายทาง ข้อมูลที่ปรากฏบนเส้นนี้อาจเป็นข้อความ ตัวเลข หรือรายการบันทึกที่ระบบคอมพิวเตอร์สามารถนำไปประมวลผลได้ สัญลักษณ์นี้มีบทบาทสำคัญในการอธิบายการส่งผ่านข้อมูลภายในระบบและช่วยให้การวิเคราะห์ระบบชัดเจนและครบถ้วนยิ่งขึ้น

#### หลักการใช้สัญลักษณ์กระแสข้อมูล

- กระแสข้อมูลสามารถใช้คู่กับการประมวลผลและสิ่งที่ยอยู่นอกระบบหรือแหล่งเก็บข้อมูลก็ได้ ขึ้นอยู่กับระบบงานว่าข้อมูลนั้นจะนำไปไว้ที่ไหนหรือข้อมูลนั้นจะนำออกจากส่วนใด
- การตั้งชื่อกระแสข้อมูลโดยทั่วไปจะตั้งชื่อด้วยคำเพียงคำเดียวที่มีความหมายชัดเจนและเข้าใจง่าย ควรกำกับชื่อบนเส้นด้วยคำนาม เช่น เวลาทำงาน ใบสั่งซื้อสินค้า เป็นต้น

- ควรตั้งชื่อกระแสข้อมูลตามข้อมูลที่ได้เปลี่ยนแปลงไปแล้วหลังจากออกจากการประมวลผล เนื่องจากการประมวลผลใช้แสดงถึงการเปลี่ยนข้อมูลหรือการส่งผ่านข้อมูล ดังนั้นการไหลข้อมูลที่ออกจากการประมวลผลมักจะมีการเขียนชื่อกำกับให้แตกต่างออกไปจากการไหลข้อมูลที่เข้ามาในการประมวลผลเสมอ

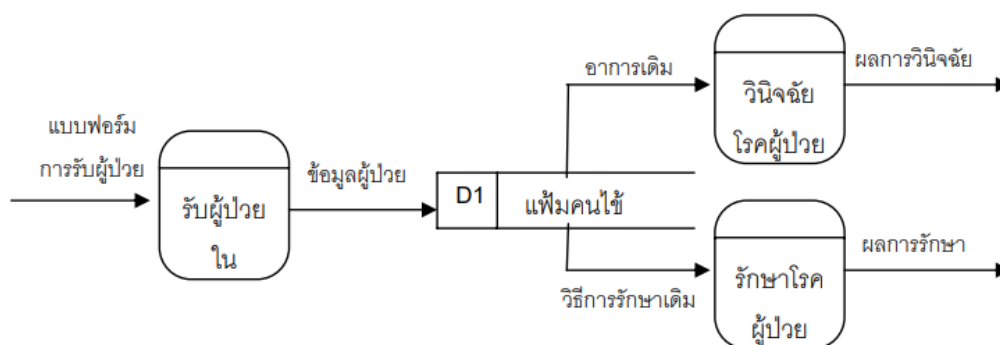


รูปที่ 2.20 ตัวอย่างการใช้สัญลักษณ์กระแสข้อมูล

- สัญลักษณ์แหล่งเก็บข้อมูล (Data Store Symbol) เป็นส่วนที่ใช้แทนชื่อแฟ้มข้อมูลที่เก็บข้อมูล เพราะมีการประมวลผลหลายแบบที่จะต้องการเก็บข้อมูลไว้เพื่อที่จะได้นำไปใช้ภายหลัง ซึ่งแหล่งเก็บข้อมูลจะต้องมีทั้งข้อมูลเข้าและข้อมูลออก โดยข้อมูลที่ออกจากแหล่งเก็บข้อมูลจะอยู่ในลักษณะที่ถูกอ่านขึ้นมา ส่วนข้อมูลที่ไหลเข้าสู่แหล่งเก็บข้อมูลจะอยู่ในรูปของการบันทึกการเพิ่มลบ และแก้ไข

#### หลักการใช้สัญลักษณ์แหล่งเก็บข้อมูล

- ต้องใช้สัญลักษณ์แหล่งเก็บข้อมูลคู่กับสัญลักษณ์กระแสข้อมูลเสมอโดยที่ ลูกศรชี้เข้าหมายถึงเป็นข้อมูลนำเข้าเก็บยังแหล่งเก็บ ลูกศรชี้ออกหมายถึงอ่านข้อมูลจากแหล่งเก็บข้อมูลไปใช้ในการประมวลผล
- แหล่งเก็บข้อมูลต้องเชื่อมต่อการประมวลผลเสมอโดยเชื่อมผ่านกระแสข้อมูล



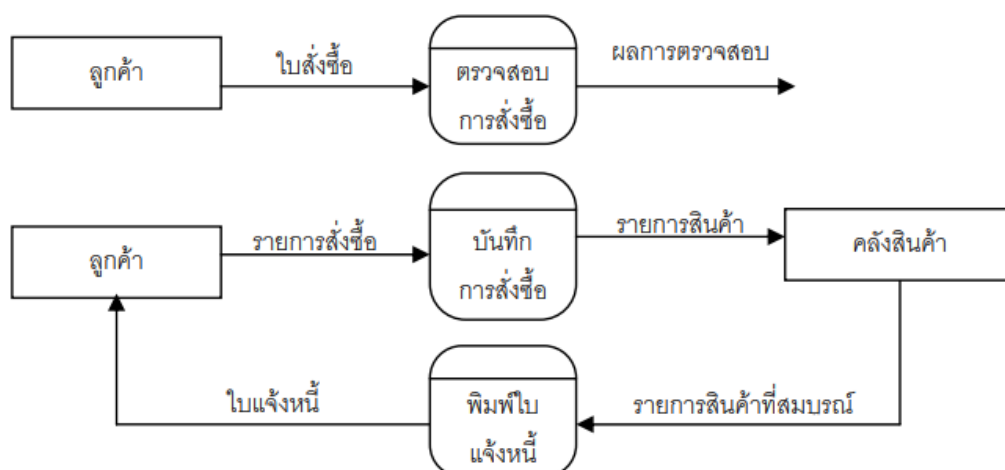
รูปที่ 2.21 ตัวอย่างการใช้สัญลักษณ์แหล่งที่เก็บข้อมูล

- เนื่องจากแหล่งเก็บข้อมูลใช้แทนสิ่งที่เก็บข้อมูลเกี่ยวกับคน สถานที่ หรือสิ่งของ ดังนั้นควรเขียนชื่อกำกับด้วยคำนาม เช่น แฟ้มข้อมูลสินค้า แฟ้มเวลาทำงานของพนักงาน เป็นต้น

● สัญลักษณ์สิ่งที่อยู่ภายนอก (External Entity Symbol) ใช้แทนแหล่งข้อมูลภายนอกในระบบ เช่น บุคคล หน่วยงานภายในหรือภายนอกองค์กร และระบบสารสนเทศอื่น ๆ ที่มีส่วนเกี่ยวข้องในการให้หรือรับข้อมูลกับระบบที่ศึกษาอยู่ สัญลักษณ์นี้ช่วยแสดงขอบเขตของระบบสารสนเทศ รวมถึงวิธีการติดต่อหรือแลกเปลี่ยนข้อมูลระหว่างระบบกับสิ่งที่อยู่นอกระบบ ซึ่งเป็นข้อมูลสำคัญในการออกแบบและพัฒนาระบบให้สอดคล้องกับการทำงานจริง

#### หลักการใช้สัญลักษณ์สิ่งที่อยู่ภายนอก

- ใช้สัญลักษณ์สิ่งที่อยู่ภายนอกคู่กับสัญลักษณ์กระแสข้อมูลเสมอ โดยที่ถ้าลูกศรชี้เข้า หมายถึงการนำข้อมูลจากหน่วยงานภายนอกเข้าสู่ระบบ ถ้าลูกศรชี้ออกหมายถึงส่งข้อมูลจากระบบไปให้หน่วยงานภายนอก



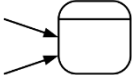
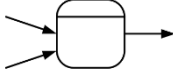

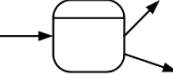


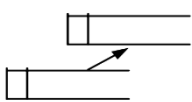
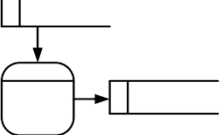
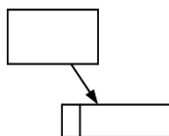
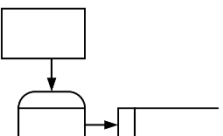
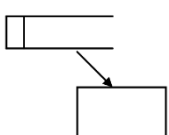
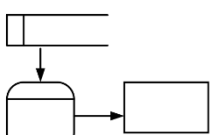
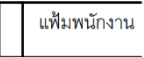
รูปที่ 2.22 ตัวอย่างการใช้สัญลักษณ์สิ่งที่อยู่ภายนอก



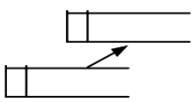
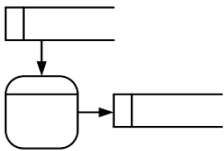
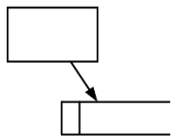
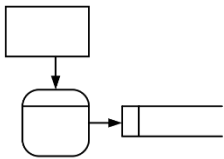
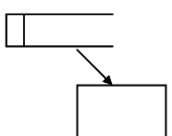
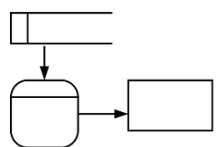
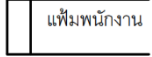
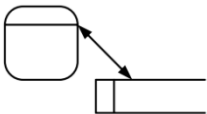
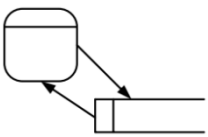
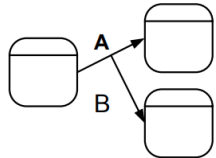
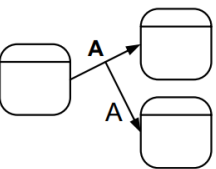
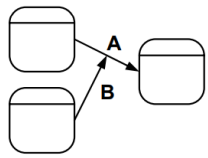
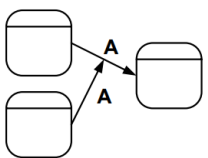
### กฎที่ใช้ในการเขียนแผนภาพกระแสข้อมูล

การเขียนแผนภาพกระแสข้อมูลต้องเป็นต้องเป็นระเบียบเรียบร้อยและมีกฎการใช้สัญลักษณ์ในการเขียนแผนภาพกระแสข้อมูลในตารางที่ 2.4

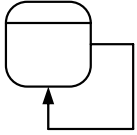
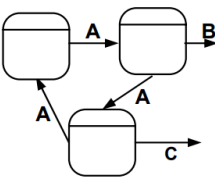
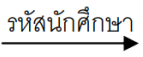
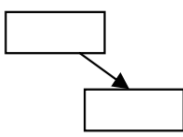
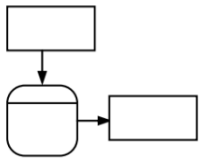
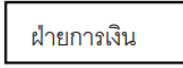
ตารางที่ 2.4 กฎการใช้สัญลักษณ์ในการเขียนแผนภาพ

ผิด	ถูก	คำอธิบาย
การประมวลผล (Process)		
		- ในการประมวลผลจะมีข้อมูลเข้าอย่างเดียวนั้นไม่ได้
		- ในการประมวลผลจะมีข้อมูลออกอย่างเดียวไม่ได้
		- ข้อความที่บรรจุอยู่ในการประมวลผลจะต้องเป็นคำกริยา หรือคำนามที่แสดงถึงกริยาเท่านั้น
แหล่งเก็บข้อมูล (Data Store)		
		- ไม่สามารถส่งข้อมูลโดยตรงจาก แหล่งเก็บข้อมูลอันหนึ่งไปยังแหล่งเก็บข้อมูลอีกอันได้โดยตรง ข้อมูลต้องผ่านการจาก ประมวลผล จึงจะไป ที่ แหล่งเก็บข้อมูลได้
		- ไม่สามารถส่งข้อมูลจากสิ่งที่อยู่ภายนอกไปยังแหล่งเก็บข้อมูลได้โดยตรง ข้อมูลต้องผ่านการจากการประมวลผลที่ได้รับข้อมูลจากแหล่งข้อมูลภายนอกแล้วจึงส่งต่อไปให้ แหล่งเก็บข้อมูล
		- ไม่สามารถส่งข้อมูลจากแหล่งเก็บข้อมูลไปยังสิ่งที่อยู่ภายนอกระบบปลายทางได้โดยตรง ต้องส่งผ่านการประมวลผลไปยังสิ่งที่อยู่ภายนอกระบบ
		- ข้อความที่บรรจุอยู่ในแหล่งเก็บข้อมูลต้องเป็นคำนาม

ตารางที่ 2.4 กฎการใช้สัญลักษณ์ในการเขียนแผนภาพ(ต่อ)

ผิด	ถูก	คำอธิบาย
แหล่งเก็บข้อมูล (Data Store)		
		- ไม่สามารถส่งข้อมูลโดยตรงจาก แหล่งเก็บข้อมูลอันหนึ่งไปยังแหล่งเก็บข้อมูลอีกอันได้โดยตรง ข้อมูลต้องผ่านการจาก ประมวลผลจึงจะไป ที่ แหล่งเก็บข้อมูลได้
		- ไม่สามารถส่งข้อมูลจากสิ่งที่อยู่ภายนอกไปยังแหล่งเก็บข้อมูลได้โดยตรง ข้อมูลต้องผ่านการจากการประมวลผลที่ได้รับข้อมูลจากแหล่งข้อมูลภายนอกแล้วจึงส่งต่อให้ แหล่งเก็บข้อมูล
		- ไม่สามารถส่งข้อมูลจากแหล่งเก็บข้อมูลไปยังสิ่งที่อยู่ภายนอกระบบปลายทางได้โดยตรง ต้องส่งผ่านการประมวลผลไปยังสิ่งที่อยู่ภายนอกระบบ
		- ข้อความที่บรรจุอยู่ในแหล่งเก็บข้อมูลต้องเป็นคำนาม
กระแสข้อมูล (Data Flow)		
		- กระแสข้อมูล จะมีทิศทางการไหลของข้อมูลเพียงทิศเดียวใน 1 เนื่องจากไม่มีการทำงานใดที่เกิดขึ้นพร้อมกัน
		- การแยกของข้อมูลออกจากเส้นกระแสข้อมูล ไปสู่การประมวลผลอื่น ๆ พร้อมกัน ต้องเป็นข้อมูลเดียวกัน
		- การเชื่อมข้อมูลจากข้อมูลมากกว่า 1 แหล่งเข้าสู่ เส้นกระแสข้อมูล เดียวกันต้องเป็นข้อมูลเดียวกัน

ตารางที่ 2.4 กฎการใช้สัญลักษณ์ในการเขียนแผนภาพ(ต่อ)

ผิด	ถูก	คำอธิบาย
กระแสข้อมูล (Data Flow)		
		- กระแสข้อมูลไม่สามารถไหลกลับเข้าสู่การประมวลผลเดิมได้โดยตรงจะต้องมีการประมวลอย่างน้อยหนึ่งอย่างมาจัดการกับข้อมูลดังกล่าวก่อนสร้างกระแสข้อมูลอื่น แล้วจึงคืนค่ากระแส ข้อมูลเดิมกลับมายังการประมวลผลเริ่มต้น
		- ข้อความที่อยู่บน Data Flow จะเป็นคำนาม
สิ่งที่อยู่ภายนอก (External Entity)		
		- ไม่สามารถส่งข้อมูลโดยตรงจากสิ่งที่อยู่ภายนอกระบบอันหนึ่งไปยังสิ่งที่อยู่ภายนอกระบบอีกอันได้โดยตรงข้อมูลต้องผ่านการจากประมวลผลภายในระบบก่อนจึงส่งไปยังสิ่งที่อยู่ภายนอกระบบภายนอกปลายทางได้
		- ข้อความที่บรรจุอยู่ในสิ่งที่อยู่ภายนอกระบบต้องเป็นคำนาม

#### ●การพัฒนาแผนภาพกระแสข้อมูล

- กำหนดรายการกิจกรรมต่าง ๆ ของธุรกิจ และแยกออกมาว่าอยู่ในรูปแบบใด
- สร้างแผนภาพระดับสูงสุด(แผนภาพบริบท)ซึ่งแสดงถึงสิ่งที่อยู่ภายนอกและข้อมูลที่ไหลเข้าและออกจากระบบหลัก โดยไม่สนใจแหล่งเก็บข้อมูล
- เขียนแผนภาพกระแสข้อมูลระดับถัดไปคือแผนภาพระดับที่ 1 (เรียกว่า “Parent Diagram”) ซึ่งแสดงถึงการประมวลผลต่าง ๆ ที่มีแต่เขียนให้อยู่ในรูปแบบทั่วไป พร้อมกันนั้นให้แสดงแหล่งเก็บข้อมูลในระดับนี้ด้วย
- สร้างแผนภาพระดับลูกของแต่ละการประมวลผลใน แผนภาพระดับที่ 2 เรียกแผนภาพในระดับนี้ว่า “แผนภาพข้อมูลระดับ 2 (Level-2 Diagram)” ถ้าหากมีรายละเอียดของการทำงานย่อยจากระดับนี้อีก ก็ให้แตกรายละเอียด ลงไปจนกระทั่งสิ้นสุด ส่วนชื่อของระดับก็จะ เป็น Level-3 Diagram, Level-4 Diagram ไปเรื่อย ๆ จนกระทั่งหมด

- ตรวจสอบหาข้อผิดพลาด และดูว่าค่ากำกับบนเส้นการไหลข้อมูลแต่ละเส้นรวมถึงการประมวลผล แต่ละกระบวนการนั้นสื่อความหมายหรือไม่
- หลังจากเขียนแผนภาพจนครบทุกการทำงานแล้วให้ทำการตรวจสอบสมมูลระหว่างข้อมูลเข้าและข้อมูลออกของแผนภาพกระแสข้อมูลกับแผนภาพบริบท
- พัฒนารูปแบบใหม่จากแผนภาพข้อมูลเชิงตรรกะให้ไปอยู่ในรูปแผนภาพข้อมูลเชิงกายภาพเพื่อแยกแยะระหว่างระบบที่ทำด้วยมือกับระบบที่ทำงานอัตโนมัติ
- แบ่งส่วนของแผนภาพกระแสข้อมูลเชิงกายภาพการแยกหรือแบ่งกลุ่มของแผนภาพออกเพื่อให้สามารถนำไปเขียนโปรแกรมหรือเพื่อการดำเนินการระบบได้

### 2.5.3 แผนภาพต้นไม้ (Tree Diagram)

แผนภาพต้นไม้เป็นเครื่องมือที่ใช้เพื่อแสดงโมดูลทั้งหมดที่อยู่ในระบบรวมถึงโมดูลย่อยที่อยู่ในโมดูลนั้น ๆ แผนภาพดังกล่าวช่วยให้ผู้พัฒนาระบบสามารถเข้าใจถึงภาพรวมของระบบอย่างชัดเจน ซึ่งแผนภาพดังกล่าวช่วยให้ผู้พัฒนาระบบสามารถมองเห็นถึงภาพรวมโมดูลของระบบได้อย่างชัดเจน

### 2.5.4 ทฤษฎีการออกแบบส่วนติดต่อประสานงานกับผู้ใช้ (User Experience and User Interface Design หรือ UX/UI Design)

ส่วนติดต่อประสานงานกับผู้ใช้และประสบการณ์ของผู้ใช้ ถือเป็นองค์ประกอบสำคัญในการพัฒนาระบบหรือแอปพลิเคชันที่มีประสิทธิภาพเป็นมิตรต่อผู้ใช้งาน การออกแบบส่วนติดต่อประสานงานกับผู้ใช้และประสบการณ์ของผู้ใช้เป็นกระบวนการที่มุ่งเน้นการสร้าง ความพึงพอใจและตอบสนองความต้องการของผู้ใช้ โดยให้ความสำคัญทั้งในด้านการใช้งาน ความสวยงาม เพื่อให้ผู้ใช้งานสามารถใช้งานระบบได้อย่างสะดวกและเข้าใจได้ง่าย

#### 2.5.3.1 ความสำคัญของการออกแบบส่วนติดต่อประสานงานกับผู้ใช้

- ทำให้ผู้ใช้งานสามารถใช้งานระบบได้โดยไม่รู้สึกลำบากหรือหลงทาง
- สร้างความประทับใจให้กับผู้ใช้ทำให้ระบบได้รับการใช้งานอย่างต่อเนื่อง
- การออกแบบที่ดีสามารถลดความสับสนหรือความยุ่งยากที่อาจทำให้ผู้ใช้หยุดใช้งานระบบ
- ผู้ใช้งานสามารถทำงานได้เร็วขึ้นและมีประสิทธิภาพมากขึ้น เนื่องจากระบบใช้งานง่ายและไม่ยุ่งยาก

#### 2.5.3.2 กระบวนการออกแบบส่วนติดต่อประสานงานกับผู้ใช้

- การเข้าใจผู้ใช้ (User-Centered Design) การออกแบบส่วนติดต่อประสานงานกับผู้ใช้และประสบการณ์ของผู้ใช้ ต้องมุ่งเน้นผู้ใช้เป็นศูนย์กลาง โดยเริ่มต้นจากการทำความเข้าใจความต้องการของผู้ใช้ผ่านการวิจัยเชิงลึก เช่น การสัมภาษณ์ การสำรวจ การวิเคราะห์พฤติกรรมผู้ใช้ (User Behavior) และการทดสอบการใช้งาน (Usability)

Testing) เพื่อหาข้อกำหนดและพฤติกรรมที่เหมาะสมจากกลุ่มเป้าหมาย การเข้าใจผู้ใช้งานจะช่วยให้สามารถออกแบบที่ตอบสนองความต้องการของผู้ใช้ได้ดียิ่งขึ้น

- ความง่ายในการใช้งาน (Usability) การใช้งานระบบหรือแอปพลิเคชันต้องมีความเรียบง่ายและตรงไปตรงมา ผู้ใช้ไม่ควรต้องใช้เวลาในการเรียนรู้วิธีการใช้งาน การออกแบบที่ดีจะช่วยให้การทำงานของผู้ใช้มีประสิทธิภาพสูงสุด โดยจะเน้นที่การลดความซับซ้อนและอำนวยความสะดวกในการทำงานให้กับผู้ใช้
- การนำทางที่ชัดเจน (Clear Navigation) การออกแบบระบบนำทางหรือเมนูให้ชัดเจนและเข้าใจได้ง่ายถือเป็นเรื่องสำคัญ เมื่อผู้ใช้สามารถหาฟังก์ชันต่าง ๆ หรือข้อมูลที่ต้องการได้อย่างรวดเร็ว ระบบจะมีความสะดวกในการใช้งานมากขึ้น โดยเฉพาะในกรณีที่แอปพลิเคชันหรือเว็บไซต์มีหลายหน้าหรือมีฟังก์ชันหลากหลาย
- การตอบสนองที่รวดเร็ว (Fast Feedback) การตอบสนองที่รวดเร็วและทันทีเป็นสิ่งสำคัญในการทำให้ผู้ใช้นั่นใจในความถูกต้องของการกระทำของตน เช่น การแสดงข้อความแจ้งเตือนเมื่อมีการดำเนินการสำเร็จหรือเกิดข้อผิดพลาด การทำให้ผู้ใช้รู้ว่าระบบรับรู้การกระทำของพวกเขาในทันทีจะช่วยเพิ่มความพึงพอใจและลดความกังวล
- ความสอดคล้อง (Consistency) การรักษาความสอดคล้องในการออกแบบเป็นสิ่งที่จะช่วยสร้างความคุ้นเคยให้กับผู้ใช้ โดยการใช้สี รูปแบบของปุ่ม หรือสัญลักษณ์ที่เหมือนกันในทุกหน้า ผู้ใช้จะไม่รู้สึกหลงทางหรือสับสนในการใช้งาน เพราะทุกอย่างที่พวกเขาเห็นจะเหมือนเดิมและเข้าใจได้ง่าย
- การเข้าถึงได้ (Accessibility) การออกแบบควรคำนึงถึงการเข้าถึงของผู้ใช้ที่มีความต้องการพิเศษ เช่น ผู้ที่มีปัญหาด้านการมองเห็น หรือผู้ที่มีข้อจำกัดในการใช้งานระบบต่าง ๆ โดยการใช้ฟอนต์ที่อ่านง่าย ขนาดปุ่มที่เหมาะสม และการรองรับคำแนะนำเสียงจึงทำให้ระบบสามารถใช้งานได้โดยไม่มีข้อจำกัด
- การออกแบบที่สวยงามและมีประสิทธิภาพ (Aesthetics and Efficiency) การออกแบบที่สวยงามไม่เพียงแต่ช่วยเพิ่มประสบการณ์ในการใช้งาน แต่ยังช่วยสร้างความรู้สึกพึงพอใจให้กับผู้ใช้ ในขณะเดียวกันการออกแบบจะต้องมีประสิทธิภาพ โดยต้องคำนึงถึงการใช้งานที่ง่ายและรวดเร็วเพื่อไม่ให้เกิดความยุ่งยาก
- ความยืดหยุ่นและการปรับเปลี่ยน (Flexibility and Adaptability) เนื่องจากการใช้งานของผู้ใช้แตกต่างกันไปตามอุปกรณ์และขนาดหน้าจอการออกแบบส่วนติดต่อประสานงานกับผู้ใช้ควรสามารถปรับตัวได้ตามอุปกรณ์ที่ใช้ เช่น การรองรับการใช้งานทั้งบนมือถือและเดสก์ท็อป ระบบควรมีความยืดหยุ่นในการแสดงผลโดยไม่ทำให้การใช้งานเสียหายหรือไม่สะดวก

### ●ประโยชน์ของการออกแบบส่วนติดต่อประสานงานกับผู้ใช้

- การออกแบบที่ดีทำให้ผู้ใช้สามารถใช้งานได้ง่าย ราบรื่น และรู้สึกพึงพอใจ
- การออกแบบที่ใช้งานง่ายสามารถลดจำนวนผู้ใช้งานที่ล้มเลิกการใช้ระบบ
- ช่วยให้ผู้ใช้สามารถทำงานได้เร็วขึ้นและมีประสิทธิภาพ
- การออกแบบที่ดีทำให้ผลิตภัณฑ์โดดเด่นในตลาดและดึงดูดผู้ใช้ใหม่ ๆ

## 2.6 ทฤษฎีด้านการออกแบบหลังบ้าน

ในการพัฒนาเว็บไซต์หรือแอปพลิเคชัน ส่วนที่ผู้ใช้เห็นและโต้ตอบได้หรือที่เรียกว่า “หน้าบ้าน (Frontend)” มักได้รับความสนใจจากผู้ใช้งานอย่างมาก อย่างไรก็ตามส่วนสำคัญที่ทำให้ระบบทำงานได้อย่างราบรื่นคือ “หลังบ้าน (Backend)” ซึ่งเป็นส่วนที่ทำงานอยู่เบื้องหลังในการจัดการข้อมูล ประมวลผลคำสั่งและเชื่อมต่อกับส่วนต่าง ๆ ของระบบการออกแบบหลังบ้านเป็นกระบวนการที่ซับซ้อนและต้องการความเข้าใจอย่างลึกซึ้ง ไม่เพียงแต่ในด้านเทคนิค (เช่น การเขียนโค้ดหรือการจัดการฐานข้อมูล) แต่ยังรวมถึงการวางแผนเชิงโครงสร้างเพื่อรองรับความต้องการที่อาจเปลี่ยนแปลงในอนาคต

### ●วัตถุประสงค์ของการออกแบบหลังบ้าน

- การจัดการข้อมูลอย่างมีประสิทธิภาพ

วัตถุประสงค์หลักของการออกแบบหลังบ้านคือการจัดการข้อมูลที่ได้รับจากผู้ใช้หรือจากระบบอื่น ๆ อย่างมีประสิทธิภาพ เช่น การเก็บข้อมูล การดึงข้อมูลจากฐานข้อมูล และการประมวลผลข้อมูลให้ได้ผลลัพธ์ที่ถูกต้องและรวดเร็ว

- การทำให้ระบบทำงานได้อย่างราบรื่นและเสถียร

การออกแบบหลังบ้านที่ดีต้องทำให้ระบบสามารถทำงานได้อย่างราบรื่นโดยไม่เกิดปัญหา เช่น ความล่าช้าในการประมวลผลหรือการหยุดทำงาน โดยการเลือกใช้เทคโนโลยีที่เหมาะสมและมีความเสถียร

- การเพิ่มความปลอดภัยในการจัดการข้อมูล

การออกแบบหลังบ้านต้องคำนึงถึงความปลอดภัยในการจัดการข้อมูล โดยการใช้วิธีการต่าง ๆ เช่น การเข้ารหัสข้อมูล การยืนยันตัวตนผู้ใช้ และการป้องกันการโจมตีจากภายนอก เพื่อปกป้องข้อมูลสำคัญจากการถูกโจรกรรมหรือการเข้าถึงโดยไม่ได้รับอนุญาต

- การรองรับการขยายตัวในอนาคต

การออกแบบหลังบ้านต้องรองรับการเติบโตของระบบในอนาคต ไม่ว่าจะเป็นจำนวนผู้ใช้งานที่เพิ่มขึ้นหรือฟังก์ชันใหม่ ๆ ที่จะถูกเพิ่มเข้ามา การออกแบบที่ยืดหยุ่นช่วยให้สามารถปรับเปลี่ยนและขยายระบบได้ง่าย

- การสนับสนุนประสบการณ์ผู้ใช้ที่ดี

วัตถุประสงค์สำคัญของหลังบ้านคือการสร้างประสบการณ์การใช้งานที่ดีและเสถียรแก่ผู้ใช้ โดยทำให้กระบวนการต่าง ๆ เช่น การสั่งซื้อสินค้า การประมวลผลคำสั่ง หรือการเข้าถึงข้อมูล เป็นไปอย่างรวดเร็วและมีประสิทธิภาพ

- การลดความซับซ้อนในการพัฒนาและบำรุงรักษา

การออกแบบหลังบ้านต้องเน้นที่ทำให้โครงสร้างของระบบมีความเข้าใจง่าย ไม่ซับซ้อน และง่ายต่อการบำรุงรักษา การใช้สถาปัตยกรรมที่เหมาะสมและการจัดการโค้ดที่ดีช่วยให้สามารถเพิ่มคุณสมบัติใหม่ ๆ หรือแก้ไขปัญหาได้อย่างรวดเร็ว

- การจัดการกับความท้าทายทางเทคนิค

การออกแบบหลังบ้านต้องรองรับการจัดการกับความท้าทายทางเทคนิค เช่น การจัดการโหลดที่สูงในช่วงเวลาที่มีผู้ใช้งานจำนวนมาก กระบวนการกระจายภาระงาน หรือการใช้แคชเพื่อเพิ่มประสิทธิภาพของระบบ

### ● องค์ประกอบที่สำคัญในการออกแบบหลังบ้าน

การออกแบบระบบหลังบ้านให้มีประสิทธิภาพและเสถียรเป็นเรื่องสำคัญที่มีผลต่อการทำงานของแอปพลิเคชันหรือเว็บไซต์ในระยะยาว องค์ประกอบหลักในการออกแบบหลังบ้านมีดังนี้

#### 1. เซิร์ฟเวอร์

เซิร์ฟเวอร์เป็นส่วนที่ทำหน้าที่ประมวลผลคำสั่งที่มาจากผู้ใช้และส่งข้อมูลระหว่างผู้ใช้กับฐานข้อมูล ในกระบวนการทำงานของระบบหลังบ้านความเสถียรและประสิทธิภาพของเซิร์ฟเวอร์เป็นสิ่งที่สำคัญมาก โดยเฉพาะในระบบที่มีผู้ใช้งานจำนวนมาก หรือในช่วงเวลาที่มีการทำงานหนัก การเลือกใช้เซิร์ฟเวอร์ที่เหมาะสม และการตั้งค่าระบบเซิร์ฟเวอร์อย่างถูกต้องสามารถช่วยให้การประมวลผลคำสั่งเป็นไปได้อย่างรวดเร็วและมีเสถียรภาพ

#### 2. ฐานข้อมูล

ฐานข้อมูลในหลังบ้านมีหน้าที่ในการจัดการและเก็บข้อมูลสำคัญของระบบ เช่น ข้อมูลผู้ใช้ ข้อมูลการสั่งซื้อ หรือข้อมูลผลิตภัณฑ์ ฐานข้อมูลต้องสามารถเก็บข้อมูลได้อย่างปลอดภัยและสามารถจัดการคำสั่งต่าง ๆ เช่น การเพิ่ม ลบ หรือแก้ไขข้อมูลได้อย่างรวดเร็ว โดยเฉพาะในระบบที่มีข้อมูลจำนวนมาก ซึ่งฐานข้อมูลสามารถแบ่งออกเป็นสองประเภทหลัก ได้แก่

- **ฐานข้อมูลเชิงสัมพันธ์ (Relational Database)** เช่น MySQL, PostgreSQL ฐานข้อมูลประเภทนี้เหมาะสำหรับระบบที่ต้องการการเชื่อมโยงข้อมูลระหว่างตารางต่าง ๆ เช่น การเก็บข้อมูลการสั่งซื้อที่เชื่อมโยงกับข้อมูลผู้ใช้และผลิตภัณฑ์
- **ฐานข้อมูล NoSQL** เช่น MongoDB ฐานข้อมูลประเภทนี้เหมาะสำหรับระบบที่ต้องการความยืดหยุ่นในการจัดเก็บข้อมูลและมีลักษณะไม่เป็นระเบียบหรือไม่ต้องการการเชื่อมโยงข้อมูลระหว่างตารางอย่างซับซ้อน

### 3.การสื่อสารผ่านส่วนประสานงานการเขียนโปรแกรมประยุกต์ (Application Programming Interface : API)

ส่วนประสานงานการเขียนโปรแกรมประยุกต์เป็นตัวกลางที่เชื่อมต่อระหว่างหลังบ้านและหน้าบ้านช่วยให้การส่งข้อมูลจากเซิร์ฟเวอร์ไปยังผู้ใช้หรือจากหน้าบ้านไปยังหลังบ้านเป็นไปอย่างมีประสิทธิภาพและราบรื่น การออกแบบส่วนประสานงานการเขียนโปรแกรมประยุกต์ควรคำนึงถึงปัจจัยหลายประการ เช่น ความยืดหยุ่นในการรองรับการใช้งานที่หลากหลาย ความปลอดภัยในการปกป้องข้อมูลที่ส่งผ่าน และความสะดวกในการใช้ส่วนประสานงานการเขียนโปรแกรมประยุกต์ซึ่งจะช่วยให้ทั้งหลังบ้านและหน้าบ้าน สามารถทำงานร่วมกันได้อย่างสมบูรณ์ โดยไม่เกิดข้อผิดพลาดหรือปัญหาในการส่งข้อมูล

#### ●แนวคิดสำคัญในการออกแบบหลังบ้าน

การออกแบบระบบหลังบ้านเป็นกระบวนการที่ซับซ้อนและต้องคำนึงถึงหลายปัจจัยเพื่อให้ระบบทำงานได้อย่างราบรื่น มีประสิทธิภาพ และสามารถรองรับการเติบโตในอนาคตได้ โดยเฉพาะในระบบที่มีผู้ใช้งานจำนวนมากหรือมีการใช้งานฟังก์ชันที่ซับซ้อน แนวคิดหลักที่ควรนำมาพิจารณาในการออกแบบหลังบ้าน มีดังนี้

##### • ความปลอดภัยเป็นหัวใจหลัก

ระบบหลังบ้านเป็นส่วนที่จัดการข้อมูลสำคัญของผู้ใช้ เช่น ข้อมูลส่วนตัว ข้อมูลธุรกรรมหรือข้อมูลที่ละเอียดอ่อน ดังนั้นการรักษาความปลอดภัยของข้อมูลเหล่านี้จึงเป็นเรื่องที่ต้องให้ความสำคัญอย่างยิ่ง การออกแบบระบบต้องคำนึงถึงวิธีการป้องกันข้อมูลจากการถูกโจมตีหรือการเข้าถึงโดยไม่ได้รับอนุญาต โดยสามารถทำได้ผ่านการใช้เทคโนโลยีการเข้ารหัสข้อมูล (Encryption) การยืนยันตัวตนผู้ใช้ (Authentication) และการป้องกันการโจมตีจากภายนอก เช่น การใช้ระบบป้องกัน DDoS หรือการตรวจสอบช่องโหว่ต่าง ๆ ที่อาจเกิดขึ้น

##### • ความสามารถในการขยายตัว

การออกแบบหลังบ้านควรคำนึงถึงความสามารถในการรองรับการเติบโตของระบบในอนาคต ทั้งในแง่ของจำนวนผู้ใช้ที่เพิ่มขึ้นหรือการเพิ่มฟังก์ชันใหม่ที่เกิดขึ้นในอนาคต การเลือกใช้สถาปัตยกรรมที่ยืดหยุ่นและสามารถขยายตัวได้ เช่น สถาปัตยกรรมแบบไมโครเซอร์วิส (Microservices) เป็นแนวทางที่ดี เพราะช่วยให้สามารถพัฒนาหรือปรับปรุงส่วนต่าง ๆ ของระบบได้โดยไม่กระทบต่อส่วนอื่น โดยการใช้บริการที่แยกออกจากกันจึงทำให้ระบบสามารถขยายตัวได้ง่ายขึ้น

##### • การทำงานที่รวดเร็วและเสถียร

ระบบหลังบ้านต้องสามารถประมวลผลคำสั่งและตอบสนองต่อผู้ใช้ได้อย่างรวดเร็ว แม้ในช่วงเวลาที่มีผู้ใช้งานจำนวนมากหรือการดำเนินการที่ซับซ้อน โดยการใช้เซิร์ฟเวอร์ที่มีประสิทธิภาพสูงและการตั้งค่าเซิร์ฟเวอร์ให้เหมาะสมกับภาระงานเป็นสิ่งจำเป็น นอกจากนี้การใช้กระบวนการกระจายภาระงานช่วยในการกระจายการทำงานไปยังเซิร์ฟเวอร์หลายเครื่อง ทำ



ให้ระบบสามารถรองรับผู้ใช้งานจำนวนมากได้ดีขึ้น และการใช้แคชเพื่อลดภาระการเข้าถึงฐานข้อมูลซ้ำ ๆ ยังช่วยเพิ่มความเร็วในการประมวลผลอีกด้วย

### ●ประโยชน์ของการออกแบบหลังบ้าน (Backend Design)

การออกแบบหลังบ้านเป็นปัจจัยสำคัญในกระบวนการพัฒนาแอปพลิเคชันหรือเว็บไซต์ โดยมีผลต่อประสิทธิภาพการรองรับการใช้งานในระยะยาวและการเพิ่มประสบการณ์ของผู้ใช้ การออกแบบที่ดีในส่วนของหลังบ้านไม่เพียงแต่ทำให้ระบบทำงานได้อย่างราบรื่น แต่ยังช่วยให้ระบบมีความยืดหยุ่นรองรับการขยายตัว และมีความปลอดภัยสูง

#### 1. การจัดการข้อมูลอย่างมีประสิทธิภาพ

การออกแบบหลังบ้านที่ดีช่วยให้ระบบสามารถจัดการข้อมูลได้อย่างรวดเร็วและมีประสิทธิภาพ สามารถเก็บ แก้ไข และดึงข้อมูลออกมาใช้งานได้ไม่มีความล่าช้า ซึ่งส่งผลให้การทำงานของแอปพลิเคชันหรือเว็บไซต์มีความราบรื่น ไม่เกิดปัญหาการรอคอยข้อมูลที่นานเกินไป นอกจากนี้การออกแบบที่ดีสามารถช่วยลดการทำงานที่ซ้ำซ้อนและเพิ่มความสามารถในการจัดการข้อมูลจำนวนมากได้อย่างมีประสิทธิภาพ

#### 2. ความปลอดภัยของข้อมูล

การออกแบบส่วนหลังบ้านที่ดีจะช่วยรักษาความปลอดภัยของข้อมูลสำคัญ เช่น ข้อมูลส่วนตัวของผู้ใช้งาน ข้อมูลธุรกรรมทางการเงิน หรือข้อมูลทางธุรกิจที่มีความสำคัญ โดยใช้วิธีการป้องกันข้อมูลที่ทันสมัย เช่น การเข้ารหัสข้อมูลและการยืนยันตัวตนที่มั่นคง การออกแบบที่ดีจะช่วยป้องกันการโจมตีจากภายนอกและการเข้าถึงข้อมูลที่ไม่ได้รับอนุญาต ทำให้ผู้ใช้งานมั่นใจในความปลอดภัยของระบบ

#### 3. การรองรับการขยายตัว

ระบบที่มีการออกแบบหลังบ้านอย่างดีย่อมสามารถรองรับการขยายตัวในอนาคตได้ เช่น การเพิ่มจำนวนผู้ใช้ หรือการเพิ่มฟังก์ชันใหม่ ๆ โดยไม่กระทบต่อประสิทธิภาพของระบบ การใช้สถาปัตยกรรมที่ยืดหยุ่น เช่น ไมโครเซอร์วิส หรือ สถาปัตยกรรมที่ไม่ใช้เซิร์ฟเวอร์ ช่วยให้การขยายระบบในอนาคตสามารถทำได้ง่ายและรวดเร็ว ไม่ต้องปรับโครงสร้างระบบทั้งหมด การออกแบบที่ดีนี้จะช่วยให้ระบบรองรับความเปลี่ยนแปลงในอนาคตได้อย่างมีประสิทธิภาพ

#### 4. ประสิทธิภาพในการประมวลผล

การออกแบบหลังบ้านที่ดีสามารถช่วยให้ระบบประมวลผลคำสั่งได้อย่างรวดเร็วและมีประสิทธิภาพ แม้ในกรณีที่มีการใช้งานที่ซับซ้อนหรือมีผู้ใช้งานจำนวนมาก การเลือกใช้เซิร์ฟเวอร์ที่เหมาะสมการตั้งค่ากระบวนการกระจายภาระงานหรือการใช้แคชจะช่วยลดเวลาในการประมวลผลและเพิ่มประสิทธิภาพรวมของระบบ ระบบที่มีการออกแบบที่ดีจะทำให้ผู้ใช้ได้รับบริการที่รวดเร็วและมีประสิทธิภาพ โดยไม่พบปัญหาความล่าช้าในการใช้งาน

## 5. การปรับปรุงประสบการณ์ผู้ใช้

การออกแบบหลังบ้านที่ดีช่วยให้ประสบการณ์ของผู้ใช้มีความเสถียรและราบรื่น การลดเวลาในการโหลดข้อมูลและการเพิ่มประสิทธิภาพของระบบช่วยให้ผู้ใช้สามารถเข้าถึงบริการต่าง ๆ ได้อย่างรวดเร็วและไม่มีปัญหาการขัดข้องของระบบ สิ่งนี้ทำให้ผู้ใช้รู้สึกพึงพอใจและมีความมั่นใจในการใช้งาน

## 6. การบำรุงรักษาระบบที่ง่ายขึ้น

การออกแบบหลังบ้านที่มีโครงสร้างที่ดีและมีความยืดหยุ่นสามารถช่วยให้ง่ายต่อการบำรุงรักษาและการปรับปรุงระบบ เมื่อเกิดปัญหาหรือมีการเปลี่ยนแปลงฟังก์ชันในอนาคต การออกแบบที่ดีจะช่วยให้ทีมพัฒนาสามารถปรับปรุงหรือแก้ไขปัญหาต่าง ๆ ได้อย่างรวดเร็วและมีประสิทธิภาพ ซึ่งจะช่วยลดเวลาและค่าใช้จ่ายในการบำรุงรักษาระบบ

## 7. ความพร้อมใช้งานสูง

การออกแบบหลังบ้านที่สามารถรองรับการทำงานที่มีความพร้อมใช้งานสูง จะช่วยลดโอกาสที่ระบบจะหยุดทำงานหรือขัดข้อง ระบบที่มีการสำรองข้อมูลและการกระจายภาระงานจะทำให้บริการสามารถทำงานได้อย่างต่อเนื่อง แม้ในกรณีที่มียอดผิดพลาดในบางส่วนของระบบ การออกแบบหลังบ้านที่ดีนี้จะช่วยทำให้ระบบมีความน่าเชื่อถือและสามารถให้บริการได้ตลอดเวลา

### 2.6.1 ทฤษฎีการออกแบบองค์ประกอบของแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล (Data Relationship Diagrams : E-R Diagrams)

ทฤษฎีการออกแบบองค์ประกอบของแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล เป็นแนวคิดพื้นฐานในการออกแบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database) โดยใช้เพื่อแสดงถึงความสัมพันธ์ระหว่างข้อมูลและโครงสร้างของฐานข้อมูล ตัวแผนภาพจะแสดงถึงองค์ประกอบหลักดังนี้

- **เอนทิตี (Entity)** เป็นสิ่งของหรือวัตถุที่สามารถระบุได้ชัดเจน เช่น ลูกค้า สินค้า หรือ คำสั่งซื้อ แต่ละเอนทิตีจะมีแอตทริบิวต์ซึ่งเป็นลักษณะหรือคุณสมบัติที่บอกข้อมูลเกี่ยวกับเอนทิตีนั้น เช่น ชื่อลูกค้า หรือรหัสสินค้า
- **แอตทริบิวต์ (Attributes)** คือคุณสมบัติหรือข้อมูลที่อธิบายลักษณะของเอนทิตีหรือความสัมพันธ์ เช่น ชื่อลูกค้า วันเกิด หรือราคาสินค้า แต่ละเอนทิตีหรือความสัมพันธ์จะมีแอตทริบิวต์เฉพาะของตนเอง
- **คีย์หลัก (Primary Key)** เป็นแอตทริบิวต์หรือชุดของแอตทริบิวต์ที่ใช้ระบุเอนทิตีในตารางได้อย่างชัดเจน โดยไม่มีซ้ำกัน เช่น รหัสลูกค้า หรือรหัสสินค้า
- **ความสัมพันธ์ (Relationships)** คือการเชื่อมโยงระหว่างเอนทิตีสองตัวขึ้นไป เช่น ความสัมพันธ์ระหว่างลูกค้าและคำสั่งซื้อ โดยจะระบุประเภทของความสัมพันธ์ เช่น หนึ่งต่อหนึ่ง (1:1) หนึ่งต่อหลาย (1:N) หรือหลายต่อหลาย (M:N)

- **คาร์ดินาลิตี (Cardinality)** คือจำนวนขั้นต่ำและสูงสุดของเอนทิตีที่สามารถเชื่อมโยงกับอีกเอนทิตีหนึ่งผ่านความสัมพันธ์เช่น ความสัมพันธ์ "ลูกค้า - คำสั่งซื้อ" อาจมีคาร์ดินาลิตี 1:N หมายถึงลูกค้าหนึ่งคนสามารถมีคำสั่งซื้อได้หลายคำสั่ง
- **เส้นโยง (Lines)** ใช้ในการเชื่อมโยงเอนทิตีกับความสัมพันธ์และอาจมีการระบุถึงข้อจำกัดหรือเงื่อนไขต่าง ๆ ในการเชื่อมโยงนั้น เช่น การบังคับให้มีข้อมูล หรือการอนุญาตให้ข้อมูลเป็นค่า Null ได้

## 2.6.2 ทฤษฎีและความรู้พื้นฐานเกี่ยวกับฐานข้อมูล

ทฤษฎีและความรู้พื้นฐานเกี่ยวกับฐานข้อมูลเป็นแนวคิดที่สำคัญในการจัดการและจัดเก็บข้อมูลอย่างมีประสิทธิภาพ ซึ่งประกอบด้วยองค์ประกอบหลักดังนี้

### 1. ฐานข้อมูล

ฐานข้อมูลคือการเก็บรวบรวมข้อมูลที่เกี่ยวข้องกันในรูปแบบที่มีโครงสร้าง และสามารถจัดการหรือค้นหาข้อมูลได้อย่างมีประสิทธิภาพ ฐานข้อมูลอาจเป็นแบบดิจิทัลหรือไม่ดิจิทัลก็ได้ แต่ในทางปฏิบัติมักหมายถึงฐานข้อมูลดิจิทัลที่เก็บในคอมพิวเตอร์

### 2. ระบบการจัดการฐานข้อมูล (Database Management System - DBMS)

ระบบการจัดการฐานข้อมูลเป็นซอฟต์แวร์ที่ใช้ในการสร้าง จัดการ และควบคุมการเข้าถึงฐานข้อมูล มันช่วยให้ผู้ใช้สามารถสร้างฐานข้อมูลใหม่ เพิ่ม ลบ หรือแก้ไขข้อมูลได้ ระบบการจัดการฐานข้อมูลยังช่วยจัดการการรักษาความปลอดภัยและความถูกต้องของข้อมูล

### 3. แบบจำลองข้อมูล (Data Models)

แบบจำลองข้อมูลเป็นวิธีการโครงสร้างข้อมูลในฐานข้อมูลที่แสดงความสัมพันธ์ระหว่างข้อมูลและกำหนดรูปแบบในการเก็บข้อมูล แบบจำลองที่ใช้บ่อยคือ

- **แบบจำลองเชิงสัมพันธ์ (Relational Model)** ใช้ตารางในการเก็บข้อมูลแต่ละตารางหรือเรียกว่าความสัมพันธ์ ประกอบด้วยแถว (เรกคอร์ด) และคอลัมน์ (ฟิลด์)
- **แบบจำลองเชิงลำดับชั้น (Hierarchical Model)** ข้อมูลถูกจัดเก็บในรูปแบบของโครงสร้างต้นไม้
- **แบบจำลองเชิงเครือข่าย (Network Model)** ข้อมูลถูกจัดเก็บในโครงสร้างแบบกราฟ

### 4. ภาษาสอบถามเชิงโครงสร้าง (Structured Query Language : SQL)

ภาษาสอบถามเชิงโครงสร้างเป็นภาษามาตรฐานที่ใช้ในการสื่อสารกับฐานข้อมูลเชิงสัมพันธ์ โดยสามารถใช้ในการสร้างโครงสร้างฐานข้อมูล (Data Definition Language : DDL) และการจัดการข้อมูล (Data Manipulation Language : DML)

## 5. ความสมบูรณ์ของข้อมูล (Data Integrity)

ความสมบูรณ์ของข้อมูลหมายถึงการรักษาความถูกต้องและความสมบูรณ์ของข้อมูลในฐานข้อมูล DBMS ใช้ข้อจำกัดและกฎที่กำหนดไว้เพื่อรักษาความสมบูรณ์นี้ เช่น ข้อจำกัดของคีย์หลัก (Primary Key Constraints) และข้อจำกัดของค่าไม่เป็นค่าว่าง (NOT NULL Constraints)

## 6. การทำให้เป็นปกติ (Normalization)

การทำให้เป็นปกติเป็นกระบวนการจัดโครงสร้างฐานข้อมูลเพื่อขจัดความซ้ำซ้อนของข้อมูลและปรับปรุงความสมบูรณ์ของข้อมูล การทำให้เป็นปกติประกอบด้วยหลายระดับ (Normal Forms) เช่น 1NF, 2NF, 3NF และ BCNF

## 7. การทำงานพร้อมกัน (Concurrency)

ฐานข้อมูลต้องรองรับการทำงานพร้อมกันจากผู้ใช้หลายคน การจัดการการทำงานพร้อมกันอย่างเหมาะสมช่วยป้องกันปัญหาการล็อกข้อมูลและปัญหาความขัดแย้งในการเข้าถึงข้อมูล

## 8. การสำรองข้อมูลและการกู้คืน (Backup and Recovery)

การสำรองข้อมูลเป็นกระบวนการคัดลอกข้อมูลจากฐานข้อมูลไปยังสถานที่ที่ปลอดภัยเพื่อป้องกันการสูญหายของข้อมูล ส่วนการกู้คืนเป็นกระบวนการนำข้อมูลกลับมาในกรณีที่เกิดข้อผิดพลาดหรือความเสียหาย

## 9. การรักษาความปลอดภัยของข้อมูล (Data Security)

ฐานข้อมูลต้องมีมาตรการรักษาความปลอดภัยเพื่อป้องกันการเข้าถึงข้อมูลโดยไม่ได้รับอนุญาต DBMS มักมีฟีเจอร์สำหรับการควบคุมการเข้าถึง การเข้ารหัสข้อมูล และการตรวจสอบการเข้าถึง

### 2.6.3 ฐานข้อมูลในการพัฒนาระบบ

การเลือกฐานข้อมูลสำหรับการพัฒนาแอปพลิเคชันการจัดการคลังสินค้าของโรงพยาบาลเป็นขั้นตอนที่สำคัญ เนื่องจากข้อมูลที่ต้องจัดการมีลักษณะที่หลากหลายและซับซ้อน ทั้งข้อมูลทางการแพทย์ เช่น ข้อมูลผู้ป่วย รายการยาและเวชภัณฑ์ รวมถึงข้อมูลเกี่ยวกับการเบิก-ยืม-คืนวัสดุ และการติดตามสินค้าคงคลัง ในการเลือกฐานข้อมูลสำหรับโครงการนี้ ปรียูณานิพนธ์นี้จึงตัดสินใจใช้ PostgreSQL ซึ่งเป็นระบบฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System : RDBMS) ที่มีความน่าเชื่อถือและมีคุณสมบัติที่เหมาะสมกับการจัดการข้อมูลในระบบโรงพยาบาล

#### 2.6.3.1 ความรู้พื้นฐานเกี่ยวกับโพสต์เกรสควแอล (PostgreSQL)

โพสต์เกรสควแอลเป็นระบบการจัดการฐานข้อมูลเชิงสัมพันธ์แบบโอเพนซอร์สที่ได้รับความนิยมและมีความสามารถสูง โดยออกแบบมาเพื่อรองรับการจัดการข้อมูลที่มี

ความซับซ้อนและมีความยืดหยุ่นสูง โพสต์เกรสคิวแอลรองรับมาตรฐาน SQL (Structured Query Language) ในการจัดการข้อมูลและมีคุณสมบัติหลายอย่างที่ทำให้มันเหมาะสมสำหรับการใช้งานในระบบที่ต้องการประสิทธิภาพสูงและความสามารถในการขยายตัวโดยสรุปคุณสมบัติหลักของโพสต์เกรสคิวแอลประกอบด้วย

#### 1.การทำงานเชิงสัมพันธ์

โพสต์เกรสคิวแอลใช้หลักการของฐานข้อมูลเชิงสัมพันธ์ในการจัดเก็บข้อมูล โดยข้อมูลจะถูกจัดเก็บในรูปแบบตาราง ที่มีแถว และสดมภ์ การออกแบบฐานข้อมูลในโพสต์เกรสคิวแอลจะประกอบด้วยตารางที่เชื่อมโยงกัน โดยใช้คีย์หลักและคีย์รองเพื่อสร้างความสัมพันธ์ระหว่างข้อมูลในตารางต่าง ๆ

#### 2.การรองรับข้อมูลที่ซับซ้อน

โพสต์เกรสคิวแอลรองรับการเก็บข้อมูลที่มีความซับซ้อนได้ดี เช่น ข้อมูลหลายมิติ ข้อมูลในรูปแบบ JSON หรือ XML รวมถึงข้อมูลเชิงพื้นที่ (Geospatial Data) ผ่านส่วนขยาย PostGIS ซึ่งช่วยให้การทำงานกับข้อมูลที่ซับซ้อนและไม่สามารถเก็บในรูปแบบตารางทั่วไปได้ง่าย

#### 3.การรองรับธุรกรรม (ACID Compliant)

โพสต์เกรสคิวแอลรองรับการทำธุรกรรมที่เป็นไปตามหลักการ ACID (Atomicity, Consistency, Isolation, Durability) ซึ่งรับประกันว่าการดำเนินการทุกครั้งจะมีความสมบูรณ์และปลอดภัย โดยธุรกรรมจะสำเร็จทั้งหมดหรือยกเลิกทั้งหมดหากเกิดข้อผิดพลาด

#### 4.การขยายฟังก์ชันการใช้งาน (Extensions)

โพสต์เกรสคิวแอลสามารถขยายฟังก์ชันการใช้งานผ่านการใช้ Extensions เช่น PostGIS สำหรับการจัดการข้อมูลภูมิศาสตร์ การค้นหาแบบข้อความเต็ม สำหรับการค้นหาข้อความ และ pg\_partman สำหรับการจัดการข้อมูลแบบแบ่งส่วน (Partitioning) ซึ่งทำให้สามารถปรับฟังก์ชันการใช้งานให้ตรงกับความต้องการของระบบ

#### 5.ประสิทธิภาพในการเข้าถึงข้อมูล

โพสต์เกรสคิวแอลใช้เทคนิคต่าง ๆ เช่น Indexing และ Query Optimization เพื่อเพิ่มประสิทธิภาพในการดึงข้อมูลจากฐานข้อมูล โดยการสร้างดัชนี (Index) บนคอลัมน์ที่ใช้ค้นหาบ่อย ๆ จะช่วยให้การค้นหาข้อมูลรวดเร็วยิ่งขึ้น อีกทั้งยังรองรับการทำ Parallel Query Processing เพื่อเพิ่มประสิทธิภาพในการประมวลผลคำสั่ง SQL บนหลายคอร์ของ CPU

#### 6.การรักษาความปลอดภัย

โพสต์เกรสคิวแอลให้ความสำคัญกับการรักษาความปลอดภัยของข้อมูล โดยรองรับการควบคุมสิทธิ์การเข้าถึงข้อมูล (Privileges) และการเข้ารหัสข้อมูลทั้งในระหว่างการส่ง

ข้อมูล (SSL/TLS) และการจัดเก็บข้อมูลภายในฐานข้อมูล (Transparent Data Encryption)

#### 7.การสำรองและกู้คืนข้อมูล

โพสต์เกรสคิวแอลมีเครื่องมือที่ช่วยในการสำรองข้อมูลและกู้คืนข้อมูล เช่น pg\_dump และ pg\_restore ซึ่งทำให้การสำรองและกู้คืนข้อมูลเป็นไปอย่างมีประสิทธิภาพ ช่วยให้สามารถกู้คืนข้อมูลที่สูญหายได้ในกรณีที่เกิดความผิดพลาด

#### 8.การสนับสนุนการทำงานหลายเซิร์ฟเวอร์ (Replication)

โพสต์เกรสคิวแอลรองรับการทำ Replication ซึ่งช่วยในการสำรองข้อมูลและการกระจายโหลดระหว่างเซิร์ฟเวอร์หลายเครื่อง เพิ่มความน่าเชื่อถือและความสามารถในการปรับขยายระบบฐานข้อมูลในกรณีที่ต้องการประสิทธิภาพสูง

#### 9.การทำงานแบบ Open-Source

โพสต์เกรสคิวแอลเป็นซอฟต์แวร์โอเพนซอร์ส (Open-Source) ที่ผู้ใช้งานสามารถดาวน์โหลดและใช้งานได้โดยไม่เสียค่าใช้จ่าย และยังสามารถปรับแต่งซอร์สโค้ดได้ ชุมชนผู้พัฒนาโพสต์เกรสคิวแอลยังช่วยให้ระบบนี้ได้รับการปรับปรุงและพัฒนาอย่างต่อเนื่อง



รูปที่ 2.23 สัญลักษณ์ของโพสต์เกรสคิวแอล

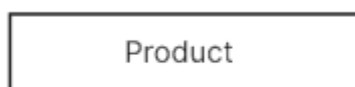
### 2.6.4 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล

แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลเป็นเครื่องมือสำคัญที่ใช้ในการออกแบบโครงสร้างฐานข้อมูล โดยช่วยแสดงความสัมพันธ์ระหว่างข้อมูลในระบบให้ชัดเจนและเข้าใจง่ายขึ้น แผนภาพนี้ถูกนำมาใช้เพื่อกำหนดเอนทิตี ความสัมพันธ์ และแอตทริบิวต์ของข้อมูลที่เกี่ยวข้องในระบบ ฐานข้อมูลที่มีการออกแบบอย่างเหมาะสมด้วยแผนภาพความสัมพันธ์ระหว่างเอนทิตีจะช่วยลดความซับซ้อนของการจัดเก็บข้อมูลและเพิ่มประสิทธิภาพในการดึงข้อมูล

#### 2.6.4.1 องค์ประกอบของแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล

เอนทิตี คือ สิ่งที่ต้องการเก็บข้อมูล ซึ่งอาจหมายถึงบุคคล วัตถุ สถานที่ หรือเหตุการณ์ เช่น ในระบบโรงพยาบาล เอนทิตีอาจเป็น ผู้ป่วย (Patient) แพทย์ (Doctor) หรือ ยา (Medication) แต่ละเอนทิตีจะประกอบด้วยคุณลักษณะ (Attribute) ที่ใช้เก็บรายละเอียดข้อมูล เช่น ชื่อ อายุ หรือ หมายเลขประจำตัว (ID) นอกจากนี้เอนทิตีสามารถแบ่งออกเป็น

เอนทิตีหลัก (Strong Entity) ซึ่งมีคีย์หลักที่ระบุความเป็นเอกลักษณ์ และเอนทิตีอ่อน (Weak Entity) ซึ่งต้องพึ่งพาเอนทิตีอื่นเพื่อระบุเอกลักษณ์

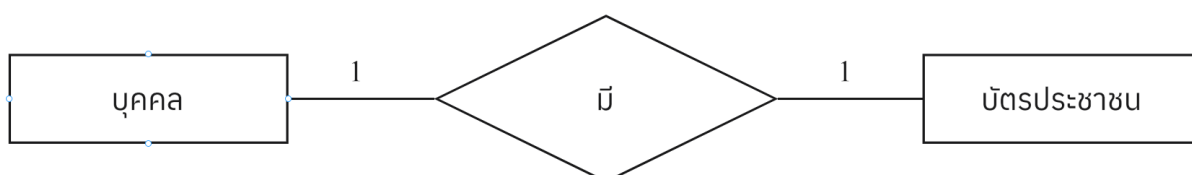


รูปที่ 2.24 ตัวอย่างการใส่ชื่อเอนทิตี

ความสัมพันธ์เป็นองค์ประกอบสำคัญในแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลที่ใช้ในการแสดงการเชื่อมโยงระหว่างเอนทิตีต่าง ๆ ภายในระบบฐานข้อมูล ความสัมพันธ์ช่วยให้เข้าใจถึงการทำงานร่วมกันระหว่างข้อมูล และเป็นพื้นฐานในการออกแบบฐานข้อมูลที่มีประสิทธิภาพ

●ความสัมพันธ์สามารถแบ่งออกได้เป็น 3 ประเภทหลัก ดังนี้

1. ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (One-to-One หรือ 1:1) ความสัมพันธ์ประเภทนี้หมายถึง เอนทิตีหนึ่งสามารถเชื่อมโยงกับเอนทิตีในอีกชุดได้เพียงหนึ่งรายการเท่านั้น เช่น ผู้ป่วยแต่ละคนจะมีเวชระเบียนเฉพาะตัว หรือพนักงานแต่ละคนอาจมีโต๊ะทำงานเป็นของตัวเอง ความสัมพันธ์ประเภทนี้มักใช้ในกรณีที่ข้อมูลมีการจับคู่แบบเฉพาะเจาะจง แสดงดังรูปที่ 2.25



รูปที่ 2.25 ภาพแสดงความสัมพันธ์แบบหนึ่งต่อหนึ่ง

2. ความสัมพันธ์แบบหนึ่งต่อหลาย (One-to-Many หรือ 1:N) ความสัมพันธ์ประเภทนี้หมายถึง เอนทิตีหนึ่งสามารถเชื่อมโยงกับหลายเอนทิตีในอีกชุดหนึ่ง แต่เอนทิตีในชุดหลังสามารถเชื่อมโยงกลับมายังเอนทิตีแรกได้เพียงหนึ่งรายการ เช่น แพทย์หนึ่งคนสามารถดูแลผู้ป่วยได้หลายคน แต่ผู้ป่วยแต่ละคนจะมีแพทย์ประจำตัวเพียงคนเดียว ความสัมพันธ์ประเภทนี้พบได้บ่อยในระบบฐานข้อมูล เช่น ระบบจัดการผู้ป่วยในโรงพยาบาล แสดงดังรูปที่ 2.26



รูปที่ 2.26 ภาพแสดงความสัมพันธ์แบบหนึ่งต่อหลาย

3. ความสัมพันธ์แบบหลายต่อหลาย (Many-to-Many หรือ M:N) ความสัมพันธ์ประเภทนี้หมายถึง เอนทิตีในชุดหนึ่งสามารถเชื่อมโยงกับเอนทิตีในอีกชุดได้หลายรายการ และในทางกลับกัน เช่น นักศึกษาหลายคนสามารถลงทะเบียนเรียนในหลายวิชา และวิชาเดียวกันอาจมีนักศึกษาหลายคนลงเรียน ในกรณีนี้ การสร้างความสัมพันธ์ในระบบฐานข้อมูลมักต้องใช้ตารางกลาง (Junction Table) เพื่อจัดเก็บข้อมูลการเชื่อมโยงระหว่างเอนทิตีทั้งสอง แสดงดังรูปที่ 2.27



รูปที่ 2.27 ภาพแสดงความสัมพันธ์แบบหลายต่อหลาย

แอตทริบิวต์คือข้อมูลหรือคุณลักษณะที่อธิบายลักษณะเฉพาะของเอนทิตีหรือความสัมพันธ์ เช่น ชื่อ ที่อยู่ หรือ ค่ารักษา ในบางกรณีแอตทริบิวต์หลักถูกกำหนดให้เป็นข้อมูลที่มีความเป็นเอกลักษณ์และใช้ระบุเอนทิตีในฐานข้อมูล

แอตทริบิวต์สามารถแบ่งออกได้หลายประเภทตามลักษณะและบทบาทของการใช้งาน ดังนี้

1. แอตทริบิวต์ธรรมดา (Simple Attribute) แอตทริบิวต์ประเภทนี้ไม่สามารถแบ่งย่อยได้อีก และมักเป็นข้อมูลที่มีความชัดเจน เช่น ชื่อ (Name) อายุ (Age) หรือเลขประจำตัวประชาชน (ID Number)
2. แอตทริบิวต์ประกอบ (Composite Attribute) เป็นแอตทริบิวต์ที่สามารถแบ่งย่อยออกเป็นส่วนย่อย ๆ ได้ เช่น ที่อยู่ (Address) ซึ่งอาจประกอบด้วยบ้านเลขที่ (House Number) ถนน (Street) และรหัสไปรษณีย์ (Postal Code) การแยกย่อยข้อมูลในลักษณะนี้ช่วยให้สามารถจัดการข้อมูลได้ง่ายขึ้นและมีความละเอียดมากขึ้น
3. แอตทริบิวต์ที่ได้มาหรือคำนวณได้ (Derived Attribute) เป็นแอตทริบิวต์ที่ไม่ได้ถูกเก็บในฐานข้อมูลโดยตรง แต่สามารถคำนวณหรืออนุมานได้จากข้อมูลอื่น เช่น อายุ (Age) ที่สามารถคำนวณได้จากวันเกิด (Date of Birth) หรือระยะเวลาทำงาน (Years of Service) ที่คำนวณจากวันที่เริ่มงาน (Start Date)



4. แอตทริบิวต์หลายค่า (Multivalued Attribute) เป็นแอตทริบิวต์ที่สามารถเก็บค่าได้มากกว่าหนึ่งค่าในเวลาเดียวกัน เช่น เบอร์โทรศัพท์ (Phone Numbers) หรืออีเมล (Email Addresses) ซึ่งแสดงถึงลักษณะที่ข้อมูลสามารถเปลี่ยนแปลงหรือมีหลายทางเลือกได้
5. แอตทริบิวต์หลัก (Primary Key Attribute) เป็นแอตทริบิวต์ที่ใช้ระบุเอนทิตีแต่ละตัวในฐานข้อมูลให้มีความเป็นเอกลักษณ์ เช่น รหัสนักศึกษา (Student ID) หรือเลขประจำตัวผู้ป่วย (Patient ID) ซึ่งค่าในแอตทริบิวต์นี้จะไม่ซ้ำกันในแต่ละแถวของฐานข้อมูล
6. แอตทริบิวต์รอง (Foreign Key Attribute) เป็นแอตทริบิวต์ที่ใช้เชื่อมโยงข้อมูลระหว่างตาราง โดยค่าของแอตทริบิวต์นี้จะอ้างอิงไปยังคีย์หลักของอีกตารางหนึ่ง เช่น รหัสแผนก (Department ID) ในตารางพนักงาน (Employee) ที่เชื่อมโยงกับตารางแผนก (Department)



รูปที่ 2.28 ตัวอย่างการใส่แอตทริบิวต์

คาร์ดินาลิตีระบุจำนวนความสัมพันธ์ที่เกิดขึ้นระหว่างเอนทิตี เช่น

- ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (1:1): เอนทิตีหนึ่งในกลุ่มสามารถเชื่อมโยงกับเอนทิตีอีกกลุ่มได้เพียงหนึ่งรายการ
- ความสัมพันธ์แบบหนึ่งต่อหลาย (1:N): เอนทิตีหนึ่งสามารถเชื่อมโยงกับเอนทิตีหลายรายการในอีกกลุ่ม
- ความสัมพันธ์แบบหลายต่อหลาย (M:N): เอนทิตีในกลุ่มหนึ่งสามารถเชื่อมโยงกับเอนทิตีหลายรายการในอีกกลุ่ม


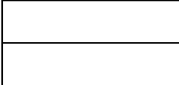


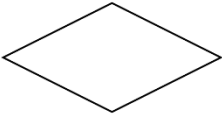

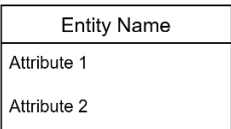

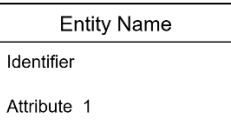
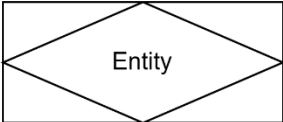

เส้นเชื่อมและสัญลักษณ์ (Lines and Symbols) เส้นเชื่อมและสัญลักษณ์ใช้แสดงความสัมพันธ์ระหว่างเอนทิตีในแผนภาพความสัมพันธ์ระหว่างเอนทิตีโดยมักมีการระบุประเภทของความสัมพันธ์และคาร์ดินาลิตีอย่างชัดเจน

### 2.6.4.2 สัญลักษณ์ที่ใช้ในแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล

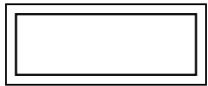
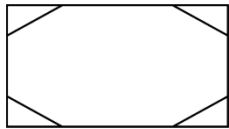
ในกระบวนการออกแบบฐานข้อมูล การสร้างแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลเป็นเครื่องมือที่สำคัญในการแสดงความสัมพันธ์ระหว่างเอนทิตี แอตทริบิวต์ และความสัมพันธ์ภายในฐานข้อมูล การใช้แผนภาพนี้ช่วยให้เข้าใจโครงสร้างข้อมูลและการเชื่อมโยงระหว่างข้อมูลต่าง ๆ ได้อย่างชัดเจน

แผนภาพความสัมพันธ์ระหว่างเอนทิตีมีหลายรูปแบบที่นิยมใช้ในการออกแบบฐานข้อมูล แต่ละรูปแบบจะมีวิธีการแสดงข้อมูลและความสัมพันธ์ที่แตกต่างกัน ซึ่งในที่นี้เราจะพูดถึงสามรูปแบบที่นิยมใช้มากที่สุด ได้แก่ Chen ERD Crow's Foot ERD และ UML Class Diagram โดยแต่ละรูปแบบจะมีสัญลักษณ์และวิธีการนำเสนอที่เป็นเอกลักษณ์ที่เหมาะสมกับลักษณะงานต่าง ๆ การเลือกใช้รูปแบบแผนภาพที่เหมาะสมจะช่วยให้การออกแบบฐานข้อมูลมีความชัดเจนและมีประสิทธิภาพมากยิ่งขึ้น

ตารางที่ 2.5 สัญลักษณ์ที่ใช้ในแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล

Chen Model	Crow's Foot Model	ความหมาย
		- แสดง Entity
		- Relationship Line เส้นเชื่อมความสัมพันธ์ระหว่าง Entity
	-	- Relationship ใช้แสดงความสัมพันธ์ระหว่าง Entity สำหรับ Crow's Foot Model ใช้ตัวอักษรเขียนแสดงความสัมพันธ์
		- Attribute ใช้แสดงคุณลักษณะของ Entity
		- ใช้แสดงคีย์หลัก (Identifier)
		- Associative Entity

ตารางที่ 2.5 สัญลักษณ์ที่ใช้ในแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล

Chen Model	Crow's Foot Model	ความหมาย
		- Weak Entity

## 2.7 ทฤษฎีการเชื่อมต่อระหว่างส่วนหน้าบ้านและหลังบ้าน

การเชื่อมต่อระหว่างส่วนหน้าบ้านและหลังบ้านเป็นกระบวนการที่สำคัญสำหรับการพัฒนาเว็บแอปพลิเคชัน โดยมีการใช้งานส่วนต่อประสานโปรแกรมประยุกต์เป็นตัวกลางในการส่งผ่านข้อมูลและคำสั่งระหว่างสองส่วนนี้อย่างมีประสิทธิภาพ

### 2.7.1 หลักการเชื่อมต่อระหว่างส่วนหน้าบ้านและหลังบ้าน

การเชื่อมต่อระหว่างส่วนหน้าบ้านและหลังบ้านใช้วิธีการรับ-ส่งข้อมูลระหว่างกัน โดยอาศัยส่วนต่อประสานโปรแกรมประยุกต์เป็นเครื่องมือหลักในการสื่อสารและแลกเปลี่ยนข้อมูล การเชื่อมต่อจะต้องคำนึงถึงความรวดเร็ว ความถูกต้อง และความปลอดภัยของข้อมูล เพื่อให้การทำงานสอดคล้องกับความต้องการของผู้ใช้งานและองค์กร

การสื่อสารระหว่างส่วนหน้าบ้านและหลังบ้านส่วนใหญ่ส่วนต่อประสานโปรแกรมประยุกต์เป็นเครื่องมือหลัก โดยทำหน้าที่เป็น "สะพานเชื่อม" ระหว่างสองส่วนนี้เพื่อให้ระบบสามารถรับ-ส่งข้อมูลและดำเนินการได้อย่างรวดเร็วและแม่นยำ การออกแบบการเชื่อมต่อจำเป็นต้องคำนึงถึง 3 ปัจจัยสำคัญ ได้แก่

#### 1. ความรวดเร็ว (Performance)

การเชื่อมต่อควรสามารถส่งข้อมูลได้อย่างรวดเร็ว เพื่อลดเวลารอคอยของผู้ใช้งาน

#### 2. ความถูกต้อง (Accuracy)

ระบบต้องรับ-ส่งข้อมูลที่ถูกต้องเพื่อป้องกันข้อผิดพลาด

#### 3. ความปลอดภัย (Security)

ข้อมูลที่ส่งระหว่างหน้าบ้านและหลังบ้านต้องได้รับการป้องกันจากการเข้าถึงโดยไม่ได้รับอนุญาต เช่น การเข้ารหัสข้อมูล

#### 2.7.1.1 กระบวนการทำงานของการเชื่อมต่อ

กระบวนการเชื่อมต่อระหว่างส่วนหน้าบ้านและหลังบ้านสามารถแบ่งออกเป็น 3 ขั้นตอนหลักดังนี้

### 1. การส่งคำขอ (Request)

ผู้ใช้งานโต้ตอบผ่านหน้าบ้าน เช่น การกรอกแบบฟอร์ม การเลือกเมนู หรือการกดปุ่มคำสั่ง หน้าบ้านจะสร้างคำขอในรูปแบบที่กำหนด (เช่น JSON หรือ XML) และส่งคำขอนี้ไปยังหลังบ้านผ่านส่วนต่อประสานโปรแกรมประยุกต์

### 2. การประมวลผลคำขอ (Processing)

หลังบ้านจะรับคำขอและทำการประมวลผล เช่น ดึงข้อมูลจากฐานข้อมูล ตรวจสอบความถูกต้อง หรือดำเนินการตามคำสั่งที่ได้รับ

### 3. การส่งคำตอบ (Response)

หลังบ้านส่งผลลัพธ์กลับไปยังหน้าบ้านในรูปแบบข้อมูล que ผู้ใช้งานสามารถเข้าใจได้ เช่น การแสดงข้อมูลในตาราง หรือข้อความแจ้งเตือน

#### 2.7.1.2 ความสำคัญของการเชื่อมต่อระหว่างหน้าบ้านและหลังบ้าน

การเชื่อมต่อที่ดีมีผลโดยตรงต่อประสบการณ์ผู้ใช้งานและประสิทธิภาพของระบบ ตัวอย่างของความสำคัญในแง่มุมต่าง ๆ มีดังนี้

##### 1. การทำงานร่วมกันที่สอดคล้อง

ส่วนหน้าบ้านและหลังบ้านต้องทำงานร่วมกันอย่างประสานกันโดยไม่มีข้อผิดพลาด เพื่อให้ข้อมูลที่แสดงต่อผู้ใช้งานถูกต้องและทันเวลา เช่น เมื่อผู้ใช้งานกรอกแบบฟอร์มคำสั่งซื้อสินค้า ส่วนหน้าบ้านจะต้องส่งข้อมูลคำขอไปยังหลังบ้านเพื่อประมวลผลคำสั่งซื้อ และหลังบ้านจะต้องตอบกลับสถานะคำสั่งซื้อที่ถูกต้องและชัดเจนให้ส่วนหน้าบ้านแสดงผล

##### 2. การลดเวลาตอบสนอง

เวลาตอบสนองของระบบ (Response Time) เป็นปัจจัยสำคัญที่ส่งผลต่อความพึงพอใจของผู้ใช้งาน หากการเชื่อมต่อระหว่างหน้าบ้านและหลังบ้านมีประสิทธิภาพ การรับส่งข้อมูลจะเกิดขึ้นอย่างรวดเร็ว ทำให้ผู้ใช้งานไม่ต้องรอนาน

##### 3. การเพิ่มความปลอดภัย

ในยุคดิจิทัลที่ข้อมูลมีค่ามาก การรักษาความปลอดภัยในการเชื่อมต่อระหว่างหน้าบ้านและหลังบ้านเป็นสิ่งจำเป็น ข้อมูลที่ถูกส่งผ่าน เช่น ข้อมูลส่วนบุคคลหรือข้อมูลธุรกรรม ต้องได้รับการป้องกันจากการโจมตี เช่น การดักจับข้อมูล (Man-in-the-Middle Attack) หรือการแฮ็กข้อมูล (Data Breach)

วิธีที่ใช้ในการเพิ่มความปลอดภัย ได้แก่

- การเข้ารหัสข้อมูลเพื่อป้องกันการเข้าถึงข้อมูลโดยไม่ได้รับอนุญาต
- การใช้ HTTPS แทน HTTP เพื่อป้องกันการโจมตีจากบุคคลที่สาม

- การใช้โทเคน (Token) และการตรวจสอบสิทธิ์ (Authentication) เพื่อควบคุมการเข้าถึงระบบ

#### 4. การรองรับการขยายระบบ

การออกแบบการเชื่อมต่อที่ยืดหยุ่นและสามารถปรับปรุงได้ช่วยให้ระบบสามารถรองรับความต้องการในอนาคตได้อย่างง่ายดาย ไม่ว่าจะเป็นการเพิ่มฟีเจอร์ใหม่ การปรับปรุงประสิทธิภาพของระบบ หรือการรองรับจำนวนผู้ใช้งานที่เพิ่มขึ้นอย่างต่อเนื่อง

### 2.7.2 ความรู้พื้นฐานเกี่ยวกับเรสตเอพีไอ (Representational State Transfer Application Programming Interface : REST API)

เรสตเอพีไอเป็นสถาปัตยกรรมที่ใช้สำหรับการสร้างและใช้งาน API ในการเชื่อมต่อและสื่อสารระหว่างระบบที่พัฒนาแอปพลิเคชัน โดยเฉพาะในยุคของเว็บแอปพลิเคชันที่มีการติดต่อระหว่างผู้ใช้กับฐานข้อมูลผ่านอินเทอร์เน็ต เรสตเอพีไอเป็นที่นิยมอย่างแพร่หลายในการพัฒนาเว็บและแอปพลิเคชันมือถือ ด้วยความยืดหยุ่นและการใช้งานที่ง่าย โดยเฉพาะการใช้ HTTP methods สำหรับการเข้าถึงและจัดการทรัพยากรต่าง ๆ ในระบบแนวคิดและหลักการของ เรสตเอพีไอ

เรสตเอพีไอเป็นการออกแบบสถาปัตยกรรมการเชื่อมต่อที่มีลักษณะการทำงานที่ชัดเจนและตรงไปตรงมา ตามแนวทางของ Roy Fielding ในปี 2000 ซึ่งเขานำเสนอแนวคิดของการเชื่อมต่อระหว่างคอมพิวเตอร์ผ่านเว็บที่ไม่ต้องการสถานะ (stateless) ซึ่งหลักการที่สำคัญของเรสตเอพีไอ ได้แก่

- 1.Stateless: การทำงานของแต่ละคำขอต้องไม่พึ่งพาข้อมูลจากคำขอก่อนหน้านี้หรือหลังทุกคำขอจากไคลเอนต์ต้องรวมข้อมูลที่จำเป็นทั้งหมดที่เซิร์ฟเวอร์ต้องใช้ในการประมวลผล
- 2.Client-server architecture: โครงสร้างนี้แยกการทำงานของเครื่องไคลเอนต์และเซิร์ฟเวอร์อย่างชัดเจน เพื่อเสริมความอิสระและความสามารถในการขยายของทั้งสองฝ่าย
- 3.Uniform Interface: การมีส่วนต่อประสานที่มาตรฐานช่วยให้การสื่อสารระหว่างไคลเอนต์และเซิร์ฟเวอร์เป็นไปอย่างไม่มีปัญหา การมีส่วนต่อประสานที่เหมือนกันทำให้การพัฒนากระบวนนั้นง่ายขึ้นและลดความซับซ้อน
- 4.Cacheable: คำตอบจากเซิร์ฟเวอร์สามารถบอกไคลเอนต์ได้ว่าข้อมูลนั้นสามารถเก็บในแคชได้หรือไม่ ซึ่งจะช่วยให้ประสิทธิภาพการทำงานของระบบ
- 5.Layered system: ระบบที่มีหลายชั้นช่วยให้สามารถจัดการระบบในแต่ละชั้นได้ง่ายขึ้น โดยไคลเอนต์ไม่จำเป็นต้องรู้ว่ามันกำลังติดต่อกับเซิร์ฟเวอร์หลักหรือตัวกลาง

#### ●วิธีการทำงานของเรสตเอพีไอ

การทำงานของเรสตเอพีไอนั้นมีขั้นตอนหลักที่สำคัญในการส่งคำขอและการตอบสนองระหว่างไคลเอนต์และเซิร์ฟเวอร์ ผ่านการใช้ HTTP Methods ที่แตกต่างกัน โดยมีรายละเอียดดังนี้:

- 1.GET ใช้เพื่อดึงข้อมูลจากเซิร์ฟเวอร์ โดยไม่ทำการเปลี่ยนแปลงหรือสร้างข้อมูลใหม่ ข้อมูลที่ส่งกลับจะเป็นการแสดงผลข้อมูลที่ร้องขอ เช่น การดึงข้อมูลผู้ใช้ ข้อมูลสินค้า
- 2.POST ใช้สำหรับการสร้างข้อมูลใหม่ในเซิร์ฟเวอร์ ข้อมูลที่ได้รับจากไคลเอนต์จะถูกนำไปสร้างข้อมูลใหม่ในฐานข้อมูล
- 3.PUT ใช้สำหรับการอัปเดตข้อมูลที่มีอยู่ในเซิร์ฟเวอร์ โดยการส่งข้อมูลทั้งหมดที่ต้องการอัปเดต หากบางส่วนของข้อมูลไม่ถูกส่งมาจะถูกลบออก
- 4.DELETE ใช้สำหรับการลบข้อมูลจากเซิร์ฟเวอร์ เช่น ลบผู้ใช้หรือสินค้าจากฐานข้อมูล
- 5.PATCH ใช้สำหรับการอัปเดตบางส่วนของข้อมูลที่มีอยู่ในเซิร์ฟเวอร์ โดยไม่ต้องส่งข้อมูลทั้งหมด

เมื่อไคลเอนต์ส่งคำขอไปยังเซิร์ฟเวอร์ คำขอเหล่านี้จะถูกส่งผ่าน URL (Uniform Resource Locator) ซึ่งจะระบุถึงทรัพยากรที่ต้องการทำงานด้วย เช่น /users สำหรับดึงข้อมูลผู้ใช้ หรือ /products/{id} สำหรับการดึงข้อมูลสินค้าตาม ID ที่ระบุ

#### ●การจัดการกับทรัพยากรในเรสตเอพีไอ

ทรัพยากรของเรสตเอพีไอจะถูกระบุโดย URI (Uniform Resource Identifier) ซึ่งเป็นการระบุแหล่งข้อมูลที่ต้องการเข้าถึง เช่นการดึงข้อมูลของผู้ใช้ที่มี ID เฉพาะ สามารถใช้ /users/{id} โดยที่ {id} คือรหัสเฉพาะของผู้ใช้ที่ต้องการเข้าถึงข้อมูล

การออกแบบ URI ควรทำให้สามารถระบุทรัพยากรได้อย่างชัดเจนและสะดวก ซึ่งช่วยในการทำงานร่วมกันระหว่างระบบต่าง ๆ ในการสร้างระบบ API ที่มีความยืดหยุ่นและสามารถขยายได้ในอนาคต

#### ●การตอบสนอง (Response) และสถานะ HTTP Codes

หลังจากที่เซิร์ฟเวอร์ได้รับคำขอจากไคลเอนต์แล้ว มันจะตอบกลับด้วย สถานะ HTTP ซึ่งเป็นรหัสสถานะที่ใช้ในการบอกว่าเกิดอะไรขึ้นกับคำขอที่ส่งมา ตัวอย่างรหัสสถานะที่สำคัญมีดังนี้

- 200 OK: หมายถึงคำขอสำเร็จและเซิร์ฟเวอร์ได้ตอบกลับข้อมูลที่ร้องขอ
- 201 Created: หมายถึงข้อมูลใหม่ได้ถูกสร้างขึ้นในเซิร์ฟเวอร์
- 400 Bad Request: หมายถึงคำขอไม่ถูกต้องหรือข้อมูลที่ส่งมาไม่สมบูรณ์
- 404 Not Found: หมายถึงทรัพยากรที่ขอไม่พบในเซิร์ฟเวอร์
- 500 Internal Server Error: หมายถึงเกิดข้อผิดพลาดภายในเซิร์ฟเวอร์

การใช้รหัสสถานะเหล่านี้ช่วยให้ไคลเอนต์สามารถเข้าใจสถานะของคำขอและทำการตอบสนองตามความเหมาะสม เช่น หากพบข้อผิดพลาดในการร้องขอ ผู้ใช้จะได้รับข้อความแสดงข้อผิดพลาดที่สามารถเข้าใจได้ง่าย

## ●ความปลอดภัยในเรสตเอพีไอ

การรักษาความปลอดภัยของข้อมูลในเรสตเอพีไอเป็นสิ่งสำคัญเพื่อป้องกันการถูกโจมตีหรือการเข้าถึงข้อมูลโดยไม่ได้รับอนุญาต วิธีการรักษาความปลอดภัยที่นิยมใช้มีดังนี้:

1. HTTPS: การใช้ HTTPS แทน HTTP จะช่วยในการเข้ารหัสข้อมูลระหว่างไคลเอนต์และเซิร์ฟเวอร์ เพื่อป้องกันการโจมตีประเภท Man-in-the-Middle (MITM) ซึ่งอาจทำให้ข้อมูลถูกดักจับหรือถูกแก้ไขระหว่างทาง
2. Authentication: การใช้เทคนิคการตรวจสอบตัวตน เช่น API Keys หรือ JSON Web Tokens (JWT) เพื่อยืนยันตัวตนของผู้ใช้ที่เรียกใช้ส่วนต่อประสานโปรแกรมประยุกต์ เพื่อให้แน่ใจว่าเป็นผู้ที่มีสิทธิ์เข้าถึงข้อมูล
3. Authorization: การตรวจสอบสิทธิ์การเข้าถึงข้อมูล เช่น การกำหนดสิทธิ์ในการเข้าถึงข้อมูลต่าง ๆ ภายในระบบ โดยเฉพาะการใช้ Role-Based Access Control (RBAC) เพื่อกำหนดบทบาทและสิทธิ์ของผู้ใช้ในระบบ

## ●ข้อดีของเรสตเอพีไอ

### 1. ความง่ายในการใช้งานและการเรียนรู้

เรสตเอพีไอใช้ HTTP Methods (GET, POST, PUT, DELETE) ซึ่งเป็นวิธีการที่คุ้นเคยและใช้กันอย่างแพร่หลาย ทำให้เป็นเครื่องมือที่ง่ายต่อการเรียนรู้และใช้งาน นักพัฒนาสามารถเข้าใจและนำไปใช้ได้อย่างรวดเร็ว โดยไม่ต้องมีความซับซ้อนในการศึกษา

### 2. รองรับการทำงานข้ามแพลตฟอร์ม

สามารถทำงานได้อย่างมีประสิทธิภาพในหลายแพลตฟอร์ม ไม่ว่าจะเป็นระบบเว็บ แพลตฟอร์มมือถือ หรือแอปพลิเคชันที่ใช้ในสภาพแวดล้อมที่แตกต่างกัน โดยไม่จำเป็นต้องมีการปรับเปลี่ยนโครงสร้างข้อมูลหรือการตั้งค่าเพิ่มเติม ซึ่งทำให้เรสตเอพีไอเป็นโซลูชันที่เหมาะสมสำหรับการพัฒนาระบบที่ต้องการการเข้าถึงข้อมูลจากหลายแพลตฟอร์ม

### 3. การขยายตัวที่ง่าย

เมื่อระบบเติบโตขึ้นหรือมีการเพิ่มฟังก์ชันการใช้งานใหม่เรสตเอพีไอสามารถปรับขยายได้โดยไม่ทำให้ระบบเดิมเกิดความยุ่งยากหรือล้มเหลว เนื่องจากการออกแบบที่มีความยืดหยุ่นและการแยกส่วนการทำงานระหว่างไคลเอนต์และเซิร์ฟเวอร์

### 4. ประสิทธิภาพสูง

ด้วยลักษณะการทำงานแบบ stateless (ไม่เก็บสถานะ) เรสตเอพีไอช่วยให้สามารถประมวลผลคำขอได้อย่างรวดเร็ว โดยไม่ต้องมีการจัดการกับข้อมูลสถานะจากคำขอก่อนหน้า ซึ่งช่วยลดการใช้ทรัพยากรและเพิ่มประสิทธิภาพในการทำงานของระบบ

### 5. สนับสนุนหลายรูปแบบข้อมูล

เรสตเอพีไอสามารถรับและส่งข้อมูลในหลากหลายรูปแบบ เช่น JSON, XML, HTML, หรือ plain text ซึ่งเพิ่มความยืดหยุ่นให้กับการออกแบบและการใช้งานส่วนต่อประสาน

โปรแกรมประยุกต์ในระบบต่าง ๆ โดยสามารถเลือกใช้รูปแบบข้อมูลที่เหมาะสมกับการประมวลผลหรือความต้องการของผู้ใช้งาน

#### ●ข้อจำกัดของเรสต์เอพีไอ

##### 1. การสื่อสารแบบทางเดียว (Stateless)

การทำงานแบบ stateless หมายความว่าแต่ละคำขอจะไม่พึ่งพาคำขอก่อนหน้านี้ ซึ่งทำให้ เรสต์เอพีไอไม่เหมาะสมกับกรณีที่ต้องการการอัปเดตข้อมูลแบบเรียลไทม์หรือข้อมูลที่มีการเปลี่ยนแปลงอยู่ตลอดเวลา เช่น ระบบที่ต้องการการติดต่อแบบต่อเนื่องระหว่างเซิร์ฟเวอร์และไคลเอนต์

##### 2. ไม่มีการเชื่อมต่อแบบสองทาง (Bidirectional Communication)

สำหรับกรณีที่ต้องการการสื่อสารสองทางหรือการสื่อสารแบบเรียลไทม์ เช่น ในระบบแชทหรือเกมออนไลน์เรสต์เอพีไออาจไม่เหมาะสม เนื่องจากไม่สามารถจัดการการเชื่อมต่อที่ต่อเนื่องระหว่างเซิร์ฟเวอร์และไคลเอนต์ได้ดีเท่ากับเทคโนโลยีอื่น ๆ เช่น WebSockets

##### 3. การจัดการกับข้อมูลขนาดใหญ่

การส่งข้อมูลขนาดใหญ่ผ่านเรสต์เอพีไอ (โดยเฉพาะในรูปแบบ JSON หรือ XML) อาจทำให้เกิดปัญหาด้านประสิทธิภาพ เช่น การล่าช้าในการรับส่งข้อมูล การส่งข้อมูลขนาดใหญ่ในแต่ละครั้งอาจทำให้เซิร์ฟเวอร์ต้องใช้ทรัพยากรมากขึ้นและทำให้เวลาตอบสนองของระบบช้าลง

##### 4. ข้อจำกัดในการรักษาสถานะ (Stateful Interactions)

เรสต์เอพีไอไม่สามารถจัดการกับสถานะในระดับที่ซับซ้อน เช่น การทำงานที่ต้องการความต่อเนื่องระหว่างคำขอหลาย ๆ ครั้ง โดยไม่ต้องมีการบันทึกสถานะจากคำขอแต่ละครั้ง การออกแบบระบบที่ต้องการจัดการสถานะที่ซับซ้อนจึงอาจไม่เหมาะกับการใช้เรสต์เอพีไอในบางกรณี

##### 5. การจัดการเวอร์ชันที่ซับซ้อน

เมื่อมีการเปลี่ยนแปลงในส่วนต่อประสานโปรแกรมประยุกต์เช่น การเพิ่มฟีเจอร์ใหม่หรือการปรับปรุงโครงสร้างข้อมูล อาจทำให้เกิดความยุ่งยากในการจัดการเวอร์ชันของส่วนต่อประสานโปรแกรมประยุกต์การปรับปรุงหรือการเปลี่ยนแปลงที่ไม่มีการจัดการเวอร์ชันที่ดีอาจทำให้เกิดปัญหาเกี่ยวกับการใช้งานในระยะยาว ซึ่งต้องมีการออกแบบกลยุทธ์ในการรองรับเวอร์ชันที่แตกต่างกันของส่วนต่อประสานโปรแกรมประยุกต์อย่างระมัดระวัง

## 2.8 ภาษาและเฟรมเวิร์คในการพัฒนาโปรแกรม

ในการพัฒนาโปรแกรมหรือระบบแอปพลิเคชันที่ทันสมัย การเลือกใช้ภาษาและเฟรมเวิร์คที่เหมาะสมเป็นสิ่งสำคัญในการสร้างระบบที่มีประสิทธิภาพและสามารถตอบสนองต่อความต้องการของ



ผู้ใช้งานได้อย่างรวดเร็วและปลอดภัย ทั้งนี้ การเลือกใช้เทคโนโลยีที่ทันสมัยจะช่วยให้ นักพัฒนาสามารถทำงานได้อย่างมีประสิทธิภาพ ลดระยะเวลาในการพัฒนา และรองรับการขยายตัวในอนาคต โดยมีการแบ่งประเภทเทคโนโลยีออกเป็นสองส่วนหลัก ได้แก่ การพัฒนาส่วนหน้าบ้านและส่วนหลังบ้าน ที่ต้องทำงานร่วมกันเพื่อให้ระบบทำงานได้อย่างครบวงจร

### 2.8.1 ภาษาและเฟรมเวิร์คในการทำหน้าบ้าน

ส่วนหน้าบ้านของระบบเป็นส่วนที่ผู้ใช้งานสามารถมองเห็นและโต้ตอบได้โดยตรง ซึ่งทำให้การเลือกใช้ภาษาและเฟรมเวิร์คที่เหมาะสมมีความสำคัญอย่างยิ่งในการสร้างประสบการณ์การใช้งานที่ดี การออกแบบและพัฒนาหน้าบ้านต้องเน้นที่ความสวยงามและการตอบสนองที่รวดเร็ว รวมถึงสามารถรองรับการปรับตัวและการพัฒนาในอนาคต เมื่อมีการเพิ่มฟีเจอร์หรือการเปลี่ยนแปลงในระบบ การใช้เครื่องมือและเทคโนโลยีที่ทันสมัยทำให้การพัฒนาหน้าบ้านมีความยืดหยุ่นสูง สามารถตอบโจทย์ความต้องการของผู้ใช้งานในหลายอุปกรณ์และหลายแพลตฟอร์มได้อย่างมีประสิทธิภาพ

#### 2.8.1.1 ความรู้พื้นฐานเกี่ยวกับเอชทีเอ็มแอลไฟท์ (Hypertext Markup Language 5 : HTML5)

เอชทีเอ็มแอลไฟท์คือเวอร์ชันล่าสุดของภาษาเอชทีเอ็มแอลที่ได้รับการพัฒนาให้รองรับฟีเจอร์ใหม่ ๆ เพื่อให้สามารถใช้งานในการพัฒนาเว็บแอปพลิเคชันที่ทันสมัยได้อย่างมีประสิทธิภาพ โดย ได้รับการเปิดตัวในปี ค.ศ. 2014 และมีการปรับปรุงให้สามารถรองรับการแสดงผลมัลติมีเดีย การโต้ตอบ และฟังก์ชันการทำงานต่าง ๆ ที่ซับซ้อนได้ดีขึ้นเช่น การรองรับการเล่นวิดีโอและเสียงโดยไม่จำเป็นต้องใช้ปลั๊กอิน (Plugins) การใช้งานแผนที่ การจัดการข้อมูลผ่าน Web Storage และการสร้างแอปพลิเคชันที่สามารถทำงานได้แม้ในสถานะออฟไลน์ (Offline Web Apps) นอกจากนี้ เอชทีเอ็มแอลไฟท์ยังช่วยปรับปรุงการทำงานของเว็บไซต์ในแง่ของการแสดงผลบนอุปกรณ์ต่าง ๆ เช่น สมาร์ท-โฟนและแท็บเล็ต รวมถึงมีการพัฒนาในเรื่องของ SEO และการเข้าถึงข้อมูล ทำให้เอชทีเอ็มแอลไฟท์เป็นเครื่องมือที่สำคัญสำหรับการพัฒนาเว็บแอปพลิเคชันในปัจจุบัน



รูปที่ 2.29 สัญลักษณ์ของเอชทีเอ็มแอลไฟท์

### ● โครงสร้างพื้นฐานของเอชทีเอ็มแอลเอฟ

โครงสร้างพื้นฐานของเอชทีเอ็มแอลประกอบไปด้วยส่วนประกอบหลัก ๆ ที่จำเป็นในการสร้างหน้าเว็บเพจ โดยจะเริ่มจากการใช้แท็กพื้นฐาน เช่น `<!DOCTYPE html>` เพื่อกำหนดชนิดของเอกสารเอชทีเอ็มแอลตามด้วยแท็ก `<html>` ซึ่งเป็นแท็กหลักของเอกสารเอชทีเอ็มแอลทั้งหมดภายในแท็ก `<html>` จะประกอบไปด้วยสองส่วนหลักคือ `<head>` และ `<body>`

1. `<head>`: ส่วนนี้ใช้ในการจัดเก็บข้อมูลเกี่ยวกับเอกสาร เช่น ชื่อของหน้าเว็บ (`<title>`) การเชื่อมต่อกับไฟล์สไตลชีตซีเอสเอส การเชื่อมต่อกับไลบรารี หรือสคริปต์ต่าง ๆ ซึ่งข้อมูลเหล่านี้จะไม่แสดงผลในหน้าเว็บโดยตรง แต่มีความสำคัญต่อการทำงานของหน้าเว็บ
2. `<body>`: ส่วนนี้เป็นส่วนที่แสดงผลข้อมูลบนหน้าเว็บ เช่น ข้อความ รูปภาพ ลิงก์ ตาราง หรือเนื้อหาที่ผู้ใช้งานสามารถมองเห็นได้ การใช้แท็กต่าง ๆ เช่น `<h1>` ถึง `<h6>` สำหรับหัวข้อ `<p>` สำหรับย่อหน้า `<a>` สำหรับลิงก์ `<img>` สำหรับรูปภาพ `<ul>` และ `<ol>` สำหรับรายการ จะช่วยในการจัดระเบียบและแสดงผลข้อมูลในรูปแบบที่เหมาะสม

### ● ข้อดีของเอชทีเอ็มแอลเอฟ

1. รองรับมัลติมีเดียโดยไม่ต้องใช้ปลั๊กอิน เอชทีเอ็มแอลเอฟรองรับการเล่นวิดีโอและเสียงในเบราว์เซอร์โดยตรงผ่านแท็ก `<video>` และ `<audio>`, ทำให้ไม่จำเป็นต้องติดตั้งปลั๊กอินเสริมเช่น Flash หรือ QuickTime อีกต่อไป ซึ่งทำให้การแสดงผลมีประสิทธิภาพและสะดวกยิ่งขึ้น
2. รองรับการทำงานออฟไลน์ (Offline) เอชทีเอ็มแอลเอฟมีคุณสมบัติสำหรับการสร้างแอปพลิเคชันที่สามารถทำงานได้โดยไม่ต้องเชื่อมต่ออินเทอร์เน็ตตลอดเวลา โดยใช้เทคโนโลยี Web Storage และ Application Cache เพื่อเก็บข้อมูลและทำงานได้เมื่อไม่มีการเชื่อมต่อ
3. รองรับการแสดงผลบนอุปกรณ์หลากหลาย เอชทีเอ็มแอลเอฟได้รับการพัฒนาให้รองรับการแสดงผลบนอุปกรณ์ที่หลากหลาย เช่น สมาร์ทโฟนและแท็บเล็ต โดยสามารถปรับการแสดงผลให้เหมาะสมกับขนาดหน้าจอและอุปกรณ์ได้อย่างอัตโนมัติ
4. เพิ่มประสิทธิภาพในการพัฒนาเว็บแอปพลิเคชัน เอชทีเอ็มแอลเอฟช่วยลดการพึ่งพาปลั๊กอินภายนอก ทำให้การพัฒนาแอปพลิเคชันทำได้ง่ายและรวดเร็วขึ้น ทั้งยังรองรับฟีเจอร์ต่าง ๆ เช่น การจัดการข้อมูลในฐานข้อมูล (Local Storage) การใช้แผนที่ (Geolocation) และการทำงานกับกราฟิก (Canvas, SVG) ที่ช่วยเพิ่มฟังก์ชันการทำงานของเว็บไซต์
5. การปรับปรุง SEO เอชทีเอ็มแอลเอฟมีฟีเจอร์ที่ช่วยให้เว็บไซต์ปรับปรุงการเข้าถึงข้อมูลและรองรับการทำงานกับเครื่องมือค้นหาทางอินเทอร์เน็ต (SEO) ได้ดียิ่งขึ้น โดย

การใช้แท็กที่มีความหมาย (Semantic Tags) เช่น <article>, <section>, <header> และ <footer> ซึ่งช่วยให้การแสดงผลและการค้นหาข้อมูลในเว็บทำได้มีประสิทธิภาพมากขึ้น

6. รองรับการทำงานร่วมกับจาวาสคริปต์ได้ดียิ่งขึ้น เอชทีเอ็มแอลไฟท์รองรับการทำงานร่วมกับจาวาสคริปต์ได้มีประสิทธิภาพมากขึ้น ซึ่งช่วยให้การสร้างแอปพลิเคชันเว็บที่มีความซับซ้อนและอินเทอร์แอคทีฟทำได้ง่ายและรวดเร็วกว่าเดิม

### 2.8.1.2 ความรู้พื้นฐานเกี่ยวกับซีเอสเอส

เอสเอส (Cascading Style Sheets : CSS) คือภาษาที่ใช้ในการกำหนดลักษณะการแสดงผลของเนื้อหาภายในเว็บเพจ ซึ่งช่วยให้การออกแบบเว็บสามารถแยกจากโครงสร้าง เอชทีเอ็มแอลได้อย่างชัดเจน โดยสามารถควบคุมรูปแบบต่าง ๆ เช่น สี ขนาดตัวอักษร รูปแบบการจัดวาง และระยะห่างขององค์ประกอบบนหน้าเว็บ การใช้ซีเอสเอสทำให้การออกแบบเว็บไซต์มีความยืดหยุ่นและสามารถบำรุงรักษาได้ง่ายขึ้น ทั้งยังช่วยเพิ่มประสิทธิภาพในการโหลดเว็บไซต์และรองรับการแสดงผลบนอุปกรณ์หลากหลายประเภท ผ่านเทคนิคต่าง ๆ เช่น Media Queries และการแยกการออกแบบออกจากโครงสร้าง เอชทีเอ็มแอลทำให้เว็บไซต์สามารถปรับตัวได้ตามขนาดหน้าจอหรืออุปกรณ์ที่ใช้งาน การเรียนรู้และใช้งานซีเอสเอสจึงเป็นสิ่งจำเป็นสำหรับนักพัฒนาเว็บเพื่อสร้างเว็บแอปพลิเคชันที่มีประสิทธิภาพและมีรูปแบบที่ตอบสนองต่อผู้ใช้ได้อย่างเหมาะสม



รูปที่ 2.30 สัญลักษณ์ของซีเอสเอส

#### ● ข้อดีของซีเอสเอส

1. แยกการออกแบบจากเนื้อหา เอสเอสช่วยแยกการออกแบบ (Style) ออกจากโครงสร้างเนื้อหา เอชทีเอ็มแอลทำให้โค้ดมีความชัดเจนและจัดการได้ง่ายขึ้น
2. เพิ่มความเร็วในการโหลด การใช้เอสเอสทำให้สามารถจัดการการตกแต่งเว็บไซต์ในไฟล์เดียวและสามารถใช้ได้กับหลายหน้าเว็บ ลดการซ้ำซ้อนของโค้ดและเพิ่มความเร็วในการโหลดเว็บไซต์

3. ปรับแต่งรูปแบบได้อย่างยืดหยุ่น เอสเอสสามารถกำหนดสไตล์ต่าง ๆ ได้ตามต้องการ เช่น สี ขนาดตัวอักษร และระยะห่าง ทำให้เว็บไซต์มีความสวยงามและตรงตามความต้องการของผู้พัฒนา
4. รองรับการแสดงผลบนหลายอุปกรณ์ เอสเอสรองรับการออกแบบที่ตอบสนองต่ออุปกรณ์ต่าง ๆ (Responsive Design) โดยใช้เทคนิค เช่น Media Queries ซึ่งทำให้เว็บไซต์สามารถปรับตัวได้กับขนาดหน้าจอที่แตกต่างกัน
5. บำรุงรักษาง่ายเมื่อใช้เอสเอสในการออกแบบเว็บไซต์ นักพัฒนาสามารถทำการแก้ไขหรือปรับปรุงรูปลักษณ์ของเว็บไซต์ได้โดยไม่กระทบกับเนื้อหาหลัก ทำให้การบำรุงรักษาเว็บไซต์ทำได้ง่ายและรวดเร็ว
6. ช่วยเพิ่ม SEO การใช้เอสเอสในการจัดการการแสดงผลสามารถทำให้โค้ดเอชทีเอ็มแอลสะอาดและมีความหมายมากขึ้น ซึ่งช่วยในด้าน SEO และการค้นหาข้อมูลในเว็บไซต์

### 2.8.1.3 ความรู้พื้นฐานเกี่ยวกับจาวาสคริปต์

จาวาสคริปต์เป็นภาษาการเขียนโปรแกรมแบบไดนามิกที่ใช้สำหรับเพิ่มความสามารถในการโต้ตอบ (interactivity) และความซับซ้อนให้กับเว็บไซต์และเว็บแอปพลิเคชัน เป็นหนึ่งในเทคโนโลยีสำคัญของการพัฒนาเว็บร่วมกับ เอชทีเอ็มแอลและเอสเอสโดยจาวาสคริปต์ทำงานในฝั่งไคลเอนต์ (Client-Side) แต่สามารถประยุกต์ใช้งานในฝั่งเซิร์ฟเวอร์ (Server-side) ได้เช่นกัน ด้วยการใช้งานร่วมกับแพลตฟอร์มเช่น Node.js จุดเด่นของ JavaScript อยู่ที่ความสามารถในการจัดการเหตุการณ์ (Event Handling) การปรับเปลี่ยน DOM แบบเรียลไทม์ การตรวจสอบข้อมูล การสร้างอนิเมชัน รวมถึงการสื่อสารกับส่วนต่อประสานโปรแกรมประยุกต์และฐานข้อมูลในรูปแบบที่ทันสมัย จาวาสคริปต์ได้กลายเป็นมาตรฐานสำคัญในโลกการพัฒนาเว็บและเป็นเครื่องมือที่นักพัฒนาจำเป็นต้องเรียนรู้เพื่อสร้างประสบการณ์การใช้งานที่ดีและตอบสนองต่อผู้ใช้งานได้อย่างเต็มที่



รูปที่ 2.31 สัญลักษณ์ของจาวาสคริปต์

### ●ข้อดีของจาวาสคริปต์

- 1.การทำงานในฝั่งไคลเอนต์ (Client-side Execution) ช่วยเพิ่มความเร็วในการทำงานของเว็บไซต์ เนื่องจากการประมวลผลเกิดขึ้นในเบราว์เซอร์ของผู้ใช้ ทำให้ไม่ต้องรอการตอบสนองจากเซิร์ฟเวอร์
- 2.ปรับเปลี่ยนหน้าเว็บแบบเรียลไทม์ สามารถปรับเปลี่ยนเนื้อหาและองค์ประกอบบนหน้าเว็บได้ทันทีโดยไม่ต้องโหลดหน้าใหม่
- 3.รองรับการโต้ตอบกับผู้ใช้ (Interactive) รองรับการจัดการเหตุการณ์ต่าง ๆ เช่น การคลิก การป้อนข้อมูล หรือการเลื่อนเมาส์ ทำให้เว็บไซต์มีความน่าสนใจและเป็นมิตรกับผู้ใช้งาน
- 4.มีไลบรารีและเฟรมเวิร์คเสริมมีเครื่องมือและเฟรมเวิร์ค เช่น React, Angular และ Vue.js ที่ช่วยเพิ่มประสิทธิภาพและลดความซับซ้อนในการพัฒนาเว็บแอปพลิเคชัน
- 5.รองรับทั้งฝั่งไคลเอนต์และเซิร์ฟเวอร์ด้วยโนดเจสทำให้จาวาสคริปต์สามารถใช้งานได้ทั้งการพัฒนาเว็บฝั่งผู้ใช้และฝั่งเซิร์ฟเวอร์
- 6.ใช้งานได้กับเบราว์เซอร์ทุกตัวจาวาสคริปต์เป็นมาตรฐานที่รองรับโดยเบราว์เซอร์หลักทุกประเภท ทำให้การใช้งานเป็นไปอย่างราบรื่น
- 7.เพิ่มประสิทธิภาพและลดภาระเซิร์ฟเวอร์ เนื่องจากสามารถประมวลผลบางส่วนในฝั่งไคลเอนต์ จึงช่วยลดภาระการทำงานของเซิร์ฟเวอร์ได้อย่างมาก

#### 2.8.1.4 ความรู้พื้นฐานเกี่ยวกับรีแอคต์เจส

รีแอคต์เจสหรือรีแอคต์คือไลบรารีโอเพนซอร์สของภาษาจาวาสคริปต์ที่ถูกพัฒนาขึ้นโดย Facebook ในปี พ.ศ. 2556 (ค.ศ. 2013) มีจุดประสงค์เพื่อช่วยนักพัฒนาสร้างส่วนหน้าบ้านของเว็บแอปพลิเคชันที่มีประสิทธิภาพและยืดหยุ่นรีแอคต์ใช้แนวคิด Component-based Architecture ที่ช่วยแบ่งส่วนประกอบของหน้าเว็บให้แยกจากกันและนำกลับมาใช้ซ้ำได้ อีกทั้งยังใช้ Virtual DOM เพื่อเพิ่มความเร็วในการอัปเดตข้อมูล พร้อมทั้งสนับสนุนการทำงานแบบ Unidirectional Data Flow ที่ทำให้การควบคุมข้อมูลง่ายขึ้น นอกจากนี้รีแอคต์ยังใช้ JSX ซึ่งเป็นการผสมผสานจาวาสคริปต์และเอชทีเอ็มแอลเพื่อให้การเขียนโครงสร้างหน้าเว็บสะดวกและชัดเจน ด้วยคุณสมบัติที่โดดเด่น เช่น การทำงานที่รวดเร็ว การพัฒนาแบบโมดูล ความยืดหยุ่นสูง และการสนับสนุนจากชุมชนขนาดใหญ่จึงได้รับความนิยมในวงการพัฒนาเว็บและถูกนำไปใช้ในระบบขนาดใหญ่ เช่น Facebook, Instagram, และ Netflix แม้จะมีข้อเสียบางประการ เช่น การเรียนรู้ที่อาจซับซ้อนสำหรับผู้เริ่มต้น แต่ React.js ยังคงเป็นเครื่องมือสำคัญที่นักพัฒนาควรศึกษาและนำไปใช้ในการพัฒนาระบบที่ต้องการการโต้ตอบสูงในยุคปัจจุบัน

### ●คุณลักษณะเด่นของรีแอคต์

1. รีแอคต์ใช้ สถาปัตยกรรมแบบคอมโพเนนต์ ซึ่งช่วยให้แอปพลิเคชันแบ่งเป็นส่วน ๆ ที่สามารถพัฒนาและทดสอบได้อย่างอิสระ ส่งผลให้การจัดการโค้ดในแอปพลิเคชันขนาดใหญ่มีประสิทธิภาพสูงขึ้น และลดความซับซ้อนในการบำรุงรักษา
2. รีแอคต์ใช้แนวคิดการเขียน ส่วนติดต่อประสานงานกับผู้ใช้ในลักษณะ Declarative ซึ่งหมายถึงการบอกรีแอคต์ว่าต้องการให้ส่วนติดต่อประสานงานกับผู้ใช้เป็นอย่างไร แทนที่จะบอกขั้นตอนการทำงานแบบ Imperative วิธีนี้ทำให้โค้ดอ่านง่ายและช่วยให้อัปเดตโดยอัตโนมัติเมื่อข้อมูลเปลี่ยนแปลง
3. รีแอคต์ใช้เทคโนโลยี Virtual DOM ซึ่งช่วยในการเปรียบเทียบระหว่าง DOM จริงและ Virtual DOM ก่อนอัปเดตส่วนที่จำเป็นในหน้าเว็บ ซึ่งทำให้การประมวลผลและการตอบสนองของส่วนติดต่อประสานงานกับผู้ใช้
4. รวดเร็วขึ้น ลดการทำงานซ้ำซ้อนของการเรนเดอร์
5. การไหลของข้อมูลในทิศทางเดียว (Unidirectional Data Flow) ช่วยให้การจัดการข้อมูลในแอปพลิเคชันมีความชัดเจนและง่ายต่อการดีบัก เพราะข้อมูลที่ส่งจากคอมโพเนนต์หนึ่งไปยังคอมโพเนนต์อื่นจะทำในทิศทางเดียว
6. รีแอคต์ใช้ JSX ซึ่งเป็นการผสมผสานระหว่างเอชทีเอ็มแอลและจาวาสคริปต์ในไฟล์เดียวกัน การใช้ JSX ช่วยให้การพัฒนาส่วนติดต่อประสานงานกับผู้ใช้สะดวกและรวดเร็ว เพราะสามารถจัดการทั้งโครงสร้างและการควบคุมลอจิกจาวาสคริปต์ได้ในไฟล์เดียว
7. React Hooks ช่วยให้สามารถใช้ State และ Lifecycle Methods ในฟังก์ชันคอมโพเนนต์ได้ โดยไม่ต้องเขียนคอมโพเนนต์แบบคลาส ทำให้โค้ดสะอาดและอ่านง่ายขึ้น
8. รีแอคต์มีระบบนิเวศที่กว้างขวางและชุมชนผู้พัฒนาขนาดใหญ่ ซึ่งทำให้ผู้พัฒนาสามารถเข้าถึงเครื่องมือและไลบรารีต่าง ๆ ได้อย่างสะดวก ช่วยให้กระบวนการพัฒนาแอปพลิเคชันเป็นไปอย่างรวดเร็วและมีประสิทธิภาพ
9. รีแอคต์ยังสามารถใช้ร่วมกับ React Native ในการพัฒนาแอปพลิเคชันมือถือ (ทั้ง iOS และ Android) โดยใช้โค้ดที่มีความคล้ายคลึงกันในการพัฒนาแอปทั้งบนเว็บและบนมือถือ ทำให้การพัฒนาแอปพลิเคชันข้ามแพลตฟอร์มสะดวกและประหยัดเวลา
10. Server-Side Rendering (SSR) รองรับการเรนเดอร์ฝั่งเซิร์ฟเวอร์ ซึ่งช่วยเพิ่มประสิทธิภาพในการโหลดหน้าเว็บและปรับปรุง SEO โดยการเรนเดอร์เอชทีเอ็มแอล ก่อนส่งไปยังเบราว์เซอร์ ทำให้เว็บไซต์โหลดเร็วและเป็นมิตรกับเครื่องมือค้นหา
11. React DevTools เป็นเครื่องมือที่ช่วยให้การตรวจสอบและดีบั๊กแอปพลิเคชันทำได้ง่ายขึ้น สามารถดูโครงสร้างของคอมโพเนนต์ ตรวจสอบ state และ props รวมถึงวิเคราะห์การทำงานของแอปในรูปแบบเรียลไทม์

12.รีแอกต์รองรับการใช้งาน TypeScript ซึ่งช่วยให้การพัฒนาแอปพลิเคชันมีความปลอดภัยมากขึ้น โดยสามารถตรวจสอบประเภทข้อมูลและลดข้อผิดพลาดที่อาจเกิดขึ้นในระหว่างการพัฒนา

13.การใช้ Redux หรือ Context API ในการจัดการสถานะ (state) ของแอปพลิเคชันช่วยให้การจัดการข้อมูลในหลายคอมโพเนนต์มีความยืดหยุ่นและง่ายขึ้น โดยเฉพาะในแอปพลิเคชันขนาดใหญ่

## 2.8.2 ภาษาและเฟรมเวิร์คที่เลือกใช้ในการทำหลังบ้าน

การพัฒนาส่วนหลังบ้านของระบบนั้นต้องคำนึงถึงประสิทธิภาพในการประมวลผลข้อมูล ความปลอดภัยในการจัดการข้อมูลผู้ใช้ รวมถึงความยืดหยุ่นในการขยายระบบในอนาคต การเลือกใช้ภาษาและเฟรมเวิร์คที่เหมาะสมจึงเป็นสิ่งสำคัญในการสร้างระบบที่มีความเสถียรและรองรับการทำงานที่มีปริมาณข้อมูลมากและการร้องขอจากผู้ใช้จำนวนมาก ภาษาที่ใช้สำหรับการพัฒนาหลังบ้านต้องสามารถจัดการกับการประมวลผลข้อมูลและเชื่อมต่อกับฐานข้อมูลได้อย่างมีประสิทธิภาพ ในขณะเดียวกันเฟรมเวิร์คที่ใช้จะช่วยให้การพัฒนาระบบเป็นไปอย่างรวดเร็วและยืดหยุ่น

### 2.8.2.1 ความรู้พื้นฐานเกี่ยวกับโนดเจเอส

โนดเจเอสเป็นเทคโนโลยีที่ได้รับความนิยมอย่างแพร่หลายในการพัฒนาระบบหลังบ้านของแอปพลิเคชันต่าง ๆ รวมถึงแอปพลิเคชันจัดการข้อมูลในโรงพยาบาล เนื่องจากความสามารถในการประมวลผลคำขอจำนวนมากพร้อมกันโดยไม่บล็อก (Non-blocking I/O) ซึ่งทำให้ สามารถรองรับการเข้าถึงระบบของผู้ป่วย พยาบาล และแพทย์ได้อย่างราบรื่นและมีประสิทธิภาพสูงโดยโนดเจเอสใช้ V8 JavaScript engine ของ Google ในการประมวลผล และรองรับการทำงานแบบอะซิงโครนัส ทำให้สามารถดำเนินการหลายคำขอพร้อมกันได้โดยไม่ทำให้การทำงานอื่น ๆ ถูกบล็อก จึงเหมาะสำหรับการพัฒนาแอปพลิเคชันที่ต้องการประสิทธิภาพและการรองรับการขยายตัวในอนาคต



รูปที่ 2.32 สัญลักษณ์ของโนดเจเอส

### ●ข้อดีของโนดเจเอส

- 1.ประสิทธิภาพสูงโนดเจเอสใช้ V8 JavaScript engine ที่ทำให้การประมวลผลรวดเร็วและรองรับคำขอจำนวนมากพร้อมกันได้ดี
- 2.การประมวลผลอะซิงโครนัส ด้วยคุณสมบัติ non-blocking I/O จึงสามารถจัดการคำขอหลายคำขอพร้อมกันได้โดยไม่ทำให้คำขออื่น ๆ ถูกบล็อก
- 3.รองรับการขยายระบบได้อย่างมีประสิทธิภาพ เหมาะสำหรับแอปพลิเคชันที่ต้องการรองรับการเติบโตในอนาคต
- 4.การพัฒนารวดเร็วใช้จาวาสคริปต์ทั้งในฝั่งเซิร์ฟเวอร์และฝั่งไคลเอนต์ ช่วยให้นักพัฒนาสามารถทำงานได้สะดวกและรวดเร็ว
- 5.เครื่องมือและไลบรารีครบครัน npm มีไลบรารีและเครื่องมือมากมายที่ช่วยให้การพัฒนาแอปพลิเคชันเป็นไปอย่างรวดเร็ว
- 6.รองรับการทำงานข้ามแพลตฟอร์มโนดเจเอสรองรับการทำงานบนหลายระบบปฏิบัติการ เช่น Windows Linux และ macOS โดยไม่ต้องปรับเปลี่ยนโค้ดมากมาย

### 2.8.2.2 ความรู้พื้นฐานเกี่ยวกับเอ็กซ์เพรสเจเอส

เอ็กซ์เพรสเจเอสเป็นเฟรมเวิร์คสำหรับพัฒนาแอปพลิเคชันเว็บและแอปพลิเคชันมือถือที่มีความเบาและยืดหยุ่นบนโนดเจเอส ด้วยชุดคุณสมบัติที่ครบถ้วนเพื่อจัดการคำขอและคำตอบในรูปแบบ HTTP โดยมุ่งเน้นการทำให้กระบวนการพัฒนาแอปพลิเคชันที่สามารถขยายได้และมีประสิทธิภาพสูงทำได้ง่ายและมีประสิทธิผล โดยเอ็กซ์เพรสเจเอสได้รับความนิยมอย่างแพร่หลายในการพัฒนาส่วนต่อประสานโปรแกรมประยุกต์แบบ RESTful และแอปพลิเคชันเว็บในรูปแบบฟูลสแต็ก (Full Stack) เนื่องจากการออกแบบที่เรียบง่าย คุณสมบัติที่หลากหลาย และความสามารถในการขยายฟังก์ชันการใช้งานได้อย่างยืดหยุ่น

### ●คุณสมบัติหลักของเอ็กซ์เพรสเจเอส

- 1.การกำหนดเส้นทาง (Routing) เอ็กซ์เพรสเจเอสให้วิธีที่สะดวกในการกำหนดเส้นทาง (Routes) ภายในแอปพลิเคชันเว็บ ซึ่งเส้นทางคือจุดที่แอปพลิเคชันตอบสนองเมื่อผู้ใช้ส่งคำขอ HTTP ไปยังเซิร์ฟเวอร์ Express รองรับ HTTP Methods มาตรฐานเช่น GET, POST, PUT, DELETE และ PATCH ซึ่งช่วยให้การออกแบบส่วนต่อประสานโปรแกรมประยุกต์และเว็บไซต์ตอบสนองตามคำขอได้อย่างมีประสิทธิภาพ
- 2.มิดเดิลแวร์ (Middleware) คือฟังก์ชันที่ทำงานระหว่างคำขอ (Request) และคำตอบ (Response) โดยจะใช้สำหรับการบันทึกข้อมูลการเข้าถึง การตรวจสอบสิทธิ์ผู้ใช้ การตรวจสอบข้อมูลในคำขอ การจัดการข้อผิดพลาด และอื่น ๆ เอ็กซ์เพรสเจเอสรองรับทั้งมิดเดิลแวร์ในตัวและที่กำหนดเองได้ตามความต้องการ



3.การจัดการคำขอและคำตอบ (Request and Response Handling) เอ็กซ์เพรสเจเอส ช่วยให้การจัดการคำขอ HTTP และการส่งคำตอบกลับง่ายขึ้น โดยสามารถดึงข้อมูลจากคำขอ เช่น พารามิเตอร์ใน URL ข้อมูลในฟอร์ม หรือข้อมูลในหัวข้อคำขอ รวมถึงส่งคำตอบในรูปแบบต่าง ๆ เช่น JSON หรือเอชทีเอ็มแอลตามความต้องการของผู้ใช้

4.เครื่องมือสร้างเทมเพลต (Template Engines) เอ็กซ์เพรสเจเอส รองรับการใช้เครื่องมือสร้างเทมเพลตเช่น EJS, Pug และ Handlebars เพื่อสร้างเนื้อหาพลลัฟ เอชทีเอ็มแอลที่สามารถปรับเปลี่ยนได้ตามข้อมูลที่ได้รับ ซึ่งช่วยในการเรนเดอร์หน้าเว็บที่สามารถปรับเปลี่ยนได้ตามข้อมูลฝั่งเซิร์ฟเวอร์

5.การจัดการไฟล์ Static (Static File Handling) เอ็กซ์เพรสเจเอส ให้บริการไฟล์ Static เช่น รูปภาพซีเอสเอสและจาวาสคริปต์โดยใช้ฟังก์ชัน `express.static()` ซึ่งมีความสำคัญในแอปพลิเคชันที่ต้องให้บริการไฟล์เหล่านี้ในส่วนของแอปพลิเคชันฝั่งผู้ใช้

## 2.9 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องในหัวข้อระบบจัดการคลังยา เวชภัณฑ์ ครุภัณฑ์และอุปกรณ์ทางการแพทย์มีความสำคัญในการพัฒนาระบบใหม่ที่สามารถแก้ไขปัญหาหรือข้อจำกัดจากระบบเดิม การศึกษางานวิจัยเหล่านี้ช่วยให้เห็นถึงแนวทางและกรอบแนวคิดที่สามารถนำไปปรับปรุงให้เหมาะสมกับบริบทของโรงพยาบาลหรือองค์กรแต่ละแห่ง โดยในบทนี้จะทำการวิเคราะห์เปรียบเทียบงานวิจัยต่าง ๆ ที่เกี่ยวข้องกับระบบการจัดการคลังพัสดุและเวชภัณฑ์ โดยเฉพาะระบบ IPISS, INVS และ BMS ซึ่งเป็นระบบที่ได้รับการพัฒนาและนำมาใช้ในโรงพยาบาลต่าง ๆ เพื่อเพิ่มประสิทธิภาพการจัดการคลังโรงพยาบาล

จากการศึกษางานวิจัยที่เกี่ยวข้องกับระบบจัดการคลังโรงพยาบาล พบว่ามีการพัฒนาระบบหลายประเภทเพื่อตอบสนองความต้องการที่หลากหลาย ดังนี้

### 1. INVS (Inventory Management and Vendor System)

ระบบ INVS เป็นโปรแกรมที่พัฒนาขึ้นโดยคณะทำงานพัฒนาระบบสารสนเทศด้านยา ระบบนี้มีเป้าหมายหลักในการเพิ่มประสิทธิภาพการจัดการคลังเวชภัณฑ์และการสั่งซื้อเวชภัณฑ์อย่างเป็นระบบเช่น งานวิจัยของกรมสนับสนุนบริการสุขภาพ ระบุว่าการนำระบบ INVS มาใช้ในโรงพยาบาลระดับภูมิภาคช่วยลดข้อผิดพลาดในการจัดเก็บข้อมูลเวชภัณฑ์ และเพิ่มความแม่นยำในการตรวจสอบสถานะสินค้าคงคลังถึง 25%

#### ฟังก์ชันเด่นของ INVS ได้แก่

- การบันทึกข้อมูลการสั่งซื้อและการรับเข้า
- การจัดการคลังสินค้าในรูปแบบเรียลไทม์

- การเชื่อมโยงฐานข้อมูลผู้ขายและสัญญาการจัดซื้อเพื่อให้กระบวนการจัดซื้อเกิดขึ้นอย่างรวดเร็ว

#### จุดเด่นของระบบ INVS ได้แก่

- เพิ่มประสิทธิภาพในการจัดการข้อมูล
- ลดข้อผิดพลาดในการคำนวณสินค้าคงคลัง
- ระบบแจ้งเตือนเวชภัณฑ์ใกล้หมดอายุเพื่อป้องกันการขาดแคลนสินค้า

#### ข้อจำกัดของระบบ INVS ได้แก่

- การฝึกอบรมผู้ใช้งานยังต้องใช้เวลาและทรัพยากรเพิ่มเติม
- ค่าใช้จ่ายสูงสำหรับการบำรุงรักษาในโรงพยาบาลขนาดเล็ก

## 2 ระบบ BMS HOSxP

ระบบ BMS HOSxP พัฒนาโดย บริษัท บางกอก เมดิคอล ซอฟต์แวร์ จำกัด (Bangkok Medical Software Co., Ltd.) ซึ่งเป็นผู้พัฒนาซอฟต์แวร์แอปพลิเคชัน HOSxP ที่ได้รับการยอมรับในด้านประสิทธิภาพและการรองรับการทำงานหลากหลายระดับ ตัวอย่างงานวิจัยของสำนักบริหารระบบสุขภาพ [x4] ชี้ให้เห็นว่าการใช้ระบบ BMS HOSxP ช่วยลดความซับซ้อนในการบริหารจัดการสินค้าคงคลังและลดเวลาในการตรวจสอบสินค้าคงเหลือได้ถึง 50%

#### ฟังก์ชันเด่นของระบบ BMS HOSxP ประกอบด้วย

- การบันทึกข้อมูลเวชภัณฑ์และพัสดุ
- การจัดการเบิก-จ่ายสินค้าไปยังคลังย่อย
- การตรวจสอบสินค้าคงคลังแบบเรียลไทม์
- การแจ้งเตือนสินค้าใกล้หมดอายุ
- การสร้างรายงานการเคลื่อนไหวของสินค้า
- การเชื่อมโยงข้อมูลกับระบบอื่น ๆ ใน HOSxP
- การวิเคราะห์การใช้สินค้าเพื่อวางแผนจัดซื้อในอนาคต
- ผลการวิจัยยังระบุว่าโรงพยาบาลที่นำระบบนี้มาใช้สามารถลดข้อผิดพลาดในการเบิกจ่ายสินค้าได้ถึง 30%

#### จุดเด่นของระบบ BMS HOSxP

- ใช้งานง่ายและเหมาะสำหรับผู้ใช้งานทุกระดับ
- ค่าใช้จ่ายต่ำ เหมาะสำหรับโรงพยาบาลที่มีงบประมาณจำกัด
- รองรับการปรับแต่งให้เหมาะสมกับความต้องการเฉพาะของโรงพยาบาล
- การเชื่อมโยงข้อมูลกับระบบอื่น ๆ ทำให้ข้อมูลมีความสอดคล้องและลดการทำงานซ้ำซ้อน

### ข้อจำกัดของระบบ BMS HOSxP

- ขาดฟังก์ชันเชิงลึกในการวิเคราะห์ข้อมูลขั้นสูง
- การบำรุงรักษาและปรับปรุงซอฟต์แวร์ต้องการผู้เชี่ยวชาญเฉพาะทาง
- ระบบอาจต้องการการปรับแต่งเพิ่มเติมเพื่อรองรับความต้องการเฉพาะของโรงพยาบาลขนาดใหญ่

### 3 ระบบ IPISS (Integrated Pharmaceutical Inventory System Software)

ระบบ IPISS ได้รับการพัฒนาขึ้นเพื่อใช้ในการจัดการคลังพัสดุและสินค้าคงคลังในโรงพยาบาลขนาดใหญ่ โดยพบว่าระบบนี้ถูกนำมาใช้ในโรงพยาบาลมหาวิทยาลัยศรีนครราชสีมาและโรงพยาบาลเชียงรายประชานุเคราะห์ ตัวอย่างเช่น งานวิจัยของโรงพยาบาลเชียงรายประชานุเคราะห์ ระบุว่า การนำระบบ IPISS มาใช้ช่วยลดข้อผิดพลาดในการบันทึกข้อมูลถึง 35% และเพิ่มความแม่นยำในการจัดเก็บสินค้าในคลัง ผลการศึกษาเพิ่มเติมยังชี้ให้เห็นว่าระบบนี้สามารถแจ้งเตือนสินค้าคงเหลือและรายการสินค้าที่ต้องสั่งซื้อใหม่ได้อย่างมีประสิทธิภาพ

#### ฟังก์ชันเด่นของระบบ IPISS ได้แก่

- การบันทึกข้อมูลแบบอัตโนมัติ
- การวิเคราะห์ข้อมูลแบบเรียลไทม์
- การสร้างรายงานเพื่อสนับสนุนการวางแผนการจัดซื้อ

#### จุดเด่นของระบบ IPISS ได้แก่

- การปรับปรุงการจัดการคลังพัสดุในโรงพยาบาลขนาดใหญ่ได้อย่างมีประสิทธิภาพ
- การวิเคราะห์ข้อมูลแบบเรียลไทม์เพื่อสนับสนุนการตัดสินใจ
- ระบบบันทึกข้อมูลแบบอัตโนมัติที่ลดข้อผิดพลาดในการจัดเก็บข้อมูล

#### ข้อจำกัดของระบบ IPISS ได้แก่

- การติดตั้งระบบมีค่าใช้จ่ายสูง
- การฝึกอบรมผู้ใช้งานต้องใช้เวลาและทรัพยากรมาก
- ไม่เหมาะสำหรับโรงพยาบาลที่มีทรัพยากรจำกัด

### 3. ระบบบริหารจัดการโรงพยาบาลของโรงพยาบาลแพร์

โรงพยาบาลแพร์ ได้ทำการศึกษาค้นคว้าและพัฒนาระบบบริหารจัดการโรงพยาบาลในรูปแบบเว็บแอปพลิเคชัน มีวัตถุประสงค์ในการแก้ไขจุดบกพร่องของระบบบริหารจัดการโรงพยาบาลที่มีอยู่เดิม ได้แก่ ระบบมีขนาดใหญ่และซับซ้อนจนเกินไป ระบบมีฟังก์ชันที่ซ้ำซ้อนและไม่ถูกใช้งาน ระบบไม่มีโมดูลเฉพาะสำหรับคลังยาและเวชภัณฑ์ และฐานข้อมูลของระบบมีขนาดใหญ่ อีกทั้งยังพบว่าตารางในฐานข้อมูลบางส่วนเป็นเพียงข้อมูลทดสอบมิได้นำมาใช้งานจริง ระบบดังกล่าวมีจุดเด่น คือ ระบบสามารถทำให้บุคลากรทางการแพทย์ทำงานได้อย่างมีประสิทธิภาพ ใช้งานง่ายไม่ซับซ้อน อีกทั้งยังมีส่วนติดต่อประสานงานกับผู้ใช้ที่สายงามทันสมัย

และในส่วนจุดด้อยของระบบ คือ ระบบไม่สามารถเชื่อมต่อข้อมูลของผู้ป่วยภายในโรงพยาบาลกับข้อมูลผู้ป่วยบนเครือข่ายภายนอกได้

## 2.10 หลักการทดสอบระบบ (System Testing)

การทดสอบระบบเป็นส่วนหนึ่งของกระบวนการพัฒนาซอฟต์แวร์และระบบคอมพิวเตอร์ ซึ่งมีความสำคัญในการตรวจสอบให้แน่ใจว่าระบบที่พัฒนาขึ้นมานั้นสามารถทำงานได้อย่างถูกต้องและตอบสนองความต้องการของผู้ใช้งาน การทดสอบระบบไม่เพียงแต่ครอบคลุมการทดสอบฟังก์ชันการทำงานของแต่ละส่วน แต่ยังรวมถึงการทำงานร่วมกันของโมดูลต่าง ๆ การทดสอบประสิทธิภาพ และการตรวจสอบความปลอดภัย ซึ่งกระบวนการเหล่านี้จะช่วยให้ผู้พัฒนาสามารถระบุและแก้ไขข้อผิดพลาดได้ก่อนที่ระบบจะถูกนำไปใช้งานจริง

### ● วัตถุประสงค์ของการทดสอบระบบ

- **ตรวจสอบความถูกต้อง (Validation)** ยืนยันว่าระบบสามารถตอบสนองความต้องการของผู้ใช้งานได้อย่างครบถ้วน
- **ตรวจสอบความสมบูรณ์ (Verification)** ยืนยันว่าระบบทำงานถูกต้องตามข้อกำหนดและการออกแบบ
- **ค้นหาข้อผิดพลาด (Defect Detection)** ระบุข้อบกพร่องหรือปัญหาที่อาจส่งผลกระทบต่อการทำงานของระบบ
- **ประเมินประสิทธิภาพ (Performance Evaluation)** ตรวจสอบความสามารถของระบบในการรองรับการทำงานภายใต้เงื่อนไขและภาระงานที่หลากหลาย

### ● ประเภทของการทดสอบระบบ

#### 1. การทดสอบหน่วย (Unit Testing)

การทดสอบหน่วยเป็นกระบวนการที่เน้นการตรวจสอบการทำงานของโมดูลหรือฟังก์ชันย่อยในระบบ เพื่อให้มั่นใจว่าฟังก์ชันเหล่านั้นสามารถทำงานได้ถูกต้องตามข้อกำหนด โดยการทดสอบนี้มักดำเนินการโดยนักพัฒนาซอฟต์แวร์ และใช้เครื่องมืออัตโนมัติ

#### 2. การทดสอบฟังก์ชันการทำงาน (Functional Testing)

การทดสอบประเภทนี้มุ่งเน้นที่การตรวจสอบว่าระบบสามารถทำงานได้ตามข้อกำหนด โดยมักใช้วิธีการทดสอบแบบกล่องดำ (Black Box Testing) ซึ่งไม่สนใจโครงสร้างภายในของระบบ แต่เน้นที่ผลลัพธ์ของการป้อนข้อมูลเข้าไปในระบบ เช่น การทดสอบว่าผู้ใช้สามารถเข้าสู่ระบบได้ด้วยชื่อผู้ใช้และรหัสผ่านที่ถูกต้อง

#### 3. การทดสอบประสิทธิภาพ (Performance Testing)

การทดสอบนี้มีจุดมุ่งหมายเพื่อวัดความเร็วและประสิทธิภาพของระบบเมื่อทำงานภายใต้ภาระงานที่หลากหลาย เช่น การทดสอบการรองรับผู้ใช้พร้อมกันจำนวนมาก (Load Testing) หรือการตรวจสอบความเสถียรของระบบภายใต้ภาระงานที่เกินปกติ (Stress Testing)

#### 4. การทดสอบความปลอดภัย (Security Testing)

การทดสอบความปลอดภัยเน้นการตรวจสอบว่าระบบสามารถป้องกันการเข้าถึงโดยไม่ได้รับอนุญาต และสามารถปกป้องข้อมูลที่สำคัญได้อย่างปลอดภัย ตัวอย่างของการทดสอบนี้ได้แก่ การตรวจสอบการเข้ารหัสข้อมูล การทดสอบการป้องกันการโจมตีแบบ SQL Injection และการทดสอบการจัดการสิทธิ์ของผู้ใช้งาน

#### 5. การทดสอบการทำงานร่วมกัน (Integration Testing)

มุ่งเน้นไปที่การตรวจสอบว่าโมดูลต่าง ๆ ภายในระบบสามารถทำงานร่วมกันได้อย่างราบรื่น เช่น การเชื่อมต่อระหว่างส่วนติดต่อผู้ใช้กับฐานข้อมูลหรือการทำงานร่วมกันระหว่างส่วนต่อประสานโปรแกรมประยุกต์ภายนอก

#### 6. การทดสอบการยอมรับ (User Acceptance Testing - UAT)

การทดสอบนี้ดำเนินการโดยผู้ใช้งานจริงหรือผู้ใช้ปลายทาง (End Users) เพื่อยืนยันว่าระบบสามารถตอบสนองต่อความต้องการและการใช้งานจริง เช่น การตรวจสอบฟังก์ชันทั้งหมดในระบบงานบริหารคลังสินค้าของโรงพยาบาล

### ● กระบวนการทดสอบระบบ

#### 1. การวางแผนการทดสอบ (Test Planning)

- กำหนดวัตถุประสงค์ของการทดสอบ เช่น การตรวจสอบความถูกต้องของฟังก์ชันหรือการประเมินประสิทธิภาพของระบบ
- ระบุขอบเขตการทดสอบ รวมถึงข้อกำหนดของระบบที่ต้องทดสอบ
- จัดทำแผนการทดสอบซึ่งประกอบด้วยรายละเอียดเกี่ยวกับกรณีทดสอบเครื่องมือที่ใช้และทรัพยากรที่จำเป็น เช่น บุคลากร เวลา และอุปกรณ์
- กำหนดตัวชี้วัดความสำเร็จของการทดสอบเพื่อใช้ประเมินผล

#### 2. การออกแบบกรณีทดสอบ (Test Case Design)

- สร้างกรณีทดสอบที่ครอบคลุมทุกสถานการณ์ที่อาจเกิดขึ้น รวมถึงกรณีปกติ กรณีผิดพลาด และกรณีที่ไม่คาดคิด
- ระบุข้อมูลอินพุตและผลลัพธ์ที่คาดหวังอย่างชัดเจน
- ใช้เทคนิคการออกแบบ เช่น การแบ่งกลุ่มข้อมูล (Equivalence Partitioning) และการวิเคราะห์ขอบเขต (Boundary Value Analysis)

#### 3. การดำเนินการทดสอบ (Test Execution)

- ดำเนินการทดสอบตามกรณีทดสอบที่ออกแบบไว้ โดยใช้เครื่องมืออัตโนมัติหรือการทดสอบด้วยมือ (Manual Testing)
- บันทึกผลลัพธ์ของการทดสอบเปรียบเทียบกับผลลัพธ์ที่คาดหวัง

- หากพบข้อผิดพลาด ให้บันทึกรายละเอียดเกี่ยวกับข้อผิดพลาด รวมถึงขั้นตอนการทำงานซ้ำ (Reproduction Steps) เพื่อให้นักพัฒนาสามารถแก้ไขได้

#### 4. การตรวจสอบและรายงานผล (Test Reporting)

- รวบรวมและวิเคราะห์ผลการทดสอบทั้งหมด เพื่อสรุปว่าแต่ละฟังก์ชันของระบบทำงานได้ตามที่คาดหวังหรือไม่
- จัดทำรายงานการทดสอบ (Test Report) ที่ระบุข้อมูล เช่น จำนวนกรณีทดสอบที่สำเร็จ จำนวนข้อผิดพลาดที่พบ และสถานะของการทดสอบ
- เสนอแนวทางการแก้ไขข้อผิดพลาด และระบุว่าจะระบบพร้อมสำหรับการใช้งานจริงหรือไม่

#### ●หลักการสำคัญในการทดสอบระบบ

1. เริ่มต้นการทดสอบตั้งแต่ระยะแรก (Early Testing)
  - การเริ่มต้นการทดสอบตั้งแต่ขั้นตอนการออกแบบและการพัฒนาช่วยลดความเสี่ยงของข้อผิดพลาดในระยะหลัง
  - การตรวจสอบข้อกำหนดและการออกแบบระบบช่วยให้สามารถระบุปัญหาได้ก่อนการเขียนโค้ด
2. ครอบคลุมทุกสถานการณ์ (Comprehensive Coverage)
  - การทดสอบควรครอบคลุมทุกฟังก์ชันของระบบ รวมถึงสถานการณ์ที่คาดการณ์ไว้และสถานการณ์ที่ไม่คาดคิด
  - ใช้กรณีทดสอบที่หลากหลาย เช่น การทดสอบด้วยข้อมูลที่ถูกต้อง ข้อมูลที่ไม่ถูกต้อง และข้อมูลขอบเขต
3. การทดสอบซ้ำ (Regression Testing)
  - การทดสอบระบบซ้ำหลังจากการแก้ไขข้อผิดพลาดหรือการเพิ่มฟีเจอร์ใหม่ เพื่อยืนยันว่าการเปลี่ยนแปลงไม่ส่งผลกระทบต่อส่วนอื่นของระบบ
4. การใช้เครื่องมืออัตโนมัติ (Automation Testing)
  - ใช้เครื่องมืออัตโนมัติเพื่อเพิ่มประสิทธิภาพและความแม่นยำในการทดสอบ โดยเฉพาะสำหรับกรณีที่ต้องทดสอบซ้ำบ่อยครั้ง
5. การรายงานข้อผิดพลาดอย่างละเอียด (Detailed Reporting)
  - ข้อผิดพลาดที่พบควรถูกบันทึกอย่างละเอียด เพื่อให้นักพัฒนาสามารถแก้ไขได้อย่างรวดเร็วและแม่นยำ
  - รายงานควรมีข้อมูล เช่น ข้อผิดพลาดที่พบ วิธีการทำซ้ำ และผลกระทบที่เกิดขึ้น
6. การประเมินผลลัพธ์ตามเป้าหมาย (Goal-Oriented Evaluation)
  - ผลการทดสอบควรถูกเปรียบเทียบกับเป้าหมายที่ตั้งไว้ในขั้นตอนการวางแผน เช่น ความถูกต้องของฟังก์ชัน ความเสถียรของระบบ และความพึงพอใจของผู้ใช้งาน

### 2.11 บทสรุป

เนื้อหาทั้งหมดที่กล่าวมาข้างต้นเป็นทฤษฎีและความรู้พื้นฐานที่จำเป็นเพื่อสร้างความเข้าใจในการพัฒนาระบบการจัดการคลังโรงพยาบาล ซึ่งการออกแบบระบบในรูปแบบของแผนภาพบริบท แผนภาพกระแสข้อมูล และการออกแบบฐานข้อมูล จะถูกนำเสนอในบทที่ 3 ต่อไป

### บทที่ 3

## การวิเคราะห์และออกแบบระบบ

ในบทนี้จะกล่าวถึงหลักการวิเคราะห์และออกแบบระบบบริหารจัดการโรงพยาบาล : กรณีศึกษาระบบการบริหารจัดการคลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคมอันประกอบด้วย 6 หัวข้อ คือ การวิเคราะห์ความต้องการของระบบ จากนั้นจะออกแบบภาพรวมของระบบด้วยแผนภาพต้นไม้ แผนภาพบริบทจะถูกออกแบบเพื่อนำเสนอความสัมพันธ์ของระบบกับสภาพแวดล้อมภายนอก แผนภาพกระแสข้อมูลจะแสดงความสัมพันธ์ของข้อมูลกับกระบวนการทำงานต่าง ๆ ในขณะที่การออกแบบฐานข้อมูลจะนำเสนอโดยใช้แผนภาพความสัมพันธ์ของข้อมูล และหัวข้อสุดท้ายเป็นการออกแบบส่วนติดต่อประสานงานกับผู้ใช้งาน รายละเอียดในหัวข้อต่าง ๆ สามารถอธิบายได้ดังต่อไปนี้

### 3.1 วิเคราะห์ความต้องการของระบบคลังโรงพยาบาล

การจัดการคลังสินค้าภายในโรงพยาบาลมีความสำคัญต่อการสนับสนุนการทำงานของบุคลากรทางการแพทย์และการรักษาผู้ป่วย ระบบคลังสินค้าที่ดีจะช่วยให้การเบิก-จ่ายสินค้า การยืม-คืน และการตรวจสอบยอดคงคลังเป็นไปอย่างมีระเบียบและแม่นยำ นอกจากนี้ยังสามารถรองรับการแจ้งเตือนเมื่อสินค้ากำลังจะหมดอายุหรือมีปริมาณต่ำกว่าระดับที่กำหนด เพื่อให้การจัดซื้อหรือการจัดการสินค้าสามารถดำเนินการได้ทันเวลา การพัฒนาระบบคลังสินค้าภายในโรงพยาบาลในรูปแบบเว็บแอปพลิเคชันจึงเป็นทางเลือกที่มีความเหมาะสม ช่วยให้การบริหารจัดการทรัพยากรทางการแพทย์มีความสะดวกและมีประสิทธิภาพมากขึ้น โดยสามารถตอบสนองความต้องการจากทั้งมุมมองของผู้ใช้งานและมุมมองของระบบอย่างครอบคลุม

#### 1. ความต้องการจากมุมมองของผู้ใช้งาน

- การเบิก-จ่ายสินค้า ผู้ใช้งานสามารถทำรายการเบิก-จ่ายสินค้าผ่านระบบได้ง่ายและรวดเร็ว โดยไม่ต้องพิมพ์เอกสารหรือการทำงานด้วยมือ ระบบต้องรองรับการตรวจสอบสถานะการเบิกสินค้าและสามารถอนุมัติหรือปฏิเสธคำขอได้
- การยืม-คืนสินค้า ผู้ใช้งานสามารถยืมสินค้าและบันทึกการคืนได้ ระบบต้องสามารถติดตามสถานะของสินค้าที่ถูกยืมไปแล้วและแจ้งเตือนเมื่อถึงเวลาคืนสินค้า ช่วยให้การจัดการเป็นไปอย่างมีระเบียบ
- การตรวจสอบยอดคงคลัง ผู้ใช้งานต้องสามารถดูข้อมูลยอดคงคลังสินค้าต่างๆ ได้แบบเรียลไทม์ ช่วยให้การตัดสินใจในการเบิกหรือยืมสินค้ามีความแม่นยำ โดยไม่ต้องรอนานหรือคาดเดา



- การจัดซื้อและติดตามคำสั่งซื้อ แผนกเภสัชกรรมและหน่วยงานที่เกี่ยวข้องต้องสามารถทำการสั่งซื้อสินค้าใหม่และติดตามสถานะคำสั่งซื้อได้ตลอดเวลา โดยระบบต้องรองรับการขออนุมัติและจัดซื้อสินค้า
- การแจ้งเตือนอัตโนมัติ ระบบจะต้องแจ้งเตือนผู้ใช้งานเมื่อพบสินค้าที่ใกล้หมดอายุหรือปริมาณต่ำกว่าระดับที่กำหนด เพื่อให้สามารถดำเนินการจัดซื้อหรือยืมสินค้าได้ทันเวลา
- การออกรายงาน ผู้ใช้งานสามารถสร้างและดาวน์โหลดรายงานต่างๆ ที่เกี่ยวข้อง เช่น รายงานปริมาณคงเหลือ รายงานการใช้สินค้า และรายงานสินค้าหมดอายุ ซึ่งจะช่วยในการวางแผนการจัดการคลังได้ดีขึ้น

## 2. ความต้องการจากมุมมองของระบบ

จากมุมมองของระบบ ต้องมีฟังก์ชันที่รองรับความต้องการของผู้ใช้งาน และทำงานได้อย่างมีประสิทธิภาพ ระบบคลังโรงพยาบาลต้องมีความสามารถดังนี้

- การจัดการข้อมูลสินค้าหลายประเภท ระบบต้องสามารถจัดการสินค้าที่มีหลายประเภท เช่น ยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์และของใช้ทั่วไป ซึ่งจะต้องมีการจัดหมวดหมู่และรหัสเฉพาะเพื่อการระบุสินค้าที่สะดวกและชัดเจน
- การบันทึกและอัปเดตข้อมูล ระบบต้องสามารถบันทึกข้อมูลการเบิก-จ่าย ยืม-คืน การนำเข้าและนำออกสินค้า และอัปเดตสต็อกอัตโนมัติในเวลาจริงเพื่อให้ข้อมูลคงความถูกต้องและทันสมัยอยู่เสมอ
- ระบบแจ้งเตือนอัตโนมัติ ระบบต้องสามารถส่งการแจ้งเตือนเกี่ยวกับสถานะของสินค้าหรือการดำเนินการที่สำคัญ เช่น การแจ้งเตือนเมื่อสินค้าหมดอายุ หรือปริมาณสินค้าต่ำกว่าระดับที่กำหนด ซึ่งจะช่วยให้บุคลากรสามารถดำเนินการได้ทันเวลา
- การจัดทำรายงาน ระบบต้องสามารถสร้างและจัดเก็บรายงานต่าง ๆ ที่เกี่ยวข้องกับการจัดการคลังสินค้า เช่น รายงานยอดคงคลัง รายงานการเบิกใช้สินค้า หรือรายงานสินค้าหมดอายุ พร้อมทั้งรองรับการส่งออกข้อมูลเป็นไฟล์ PDF หรือ Excel เพื่อให้ผู้ใช้งานสามารถนำไปใช้ในการวิเคราะห์และวางแผนการจัดการคลังได้
- ความปลอดภัยของข้อมูล ระบบต้องมีมาตรการในการรักษาความปลอดภัยของข้อมูลทั้งในด้านการเข้าถึงและการจัดเก็บ เช่น การกำหนดสิทธิ์ผู้ใช้งาน และการเข้ารหัสข้อมูล เพื่อป้องกันการเข้าถึงข้อมูลโดยไม่ได้รับอนุญาต
- รองรับการทำงานร่วมกับอุปกรณ์ต่าง ๆ ระบบจะต้องสามารถรองรับการใช้งานจากอุปกรณ์ต่าง ๆ เพื่อให้การทำงานของบุคลากรสามารถทำได้จากทุกที่ทุกเวลา

### 3.2 ภาพรวมของระบบคลังโรงพยาบาล

แผนภาพต้นไม้จะแสดงถึงโมดูลการทำงานในระบบบริหารจัดการโรงพยาบาล: กรณีสืบหากระบวนการบริหารจัดการคลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคม จากทฤษฎีในบทที่ 2 หัวข้อที่ 2.5 ทำให้สามารถออกแบบแผนภาพต้นไม้ โดยกระบวนการที่เกี่ยวข้องมีทั้งหมด 9 โมดูล ได้แก่ กระบวนการเข้าออกสู่ระบบ กระบวนการตรวจสอบยอดคงคลัง กระบวนการการสั่งซื้อ กระบวนการการนำเข้า กระบวนการการเบิก กระบวนการการยืม-การคืน กระบวนการตั้งค่าแก้ไขข้อมูล กระบวนการระบบแจ้งเตือน และกระบวนการออกรายงานทั้งนี้เพื่อความเข้าใจ ระบบจะแบ่งโมดูลออกเป็น 3 ส่วนดังแสดงในรูปที่ 3.1 – 3.3

### 3.3 แผนภาพบริบทระบบคลังโรงพยาบาล

แผนภาพบริบทแสดงถึงภาพรวมของระบบการจัดการคลังและแสดงความสัมพันธ์ระหว่างผู้ใช้งาน จากทฤษฎีของแผนภาพบริบทในบทที่ 2 หัวข้อที่ 2.5 ทำให้สามารถออกแบบแผนภาพบริบทของระบบการจัดการคลัง โดยผู้ที่เกี่ยวข้องกับระบบมีทั้งหมด 5 ประเภท ได้แก่ ผู้ดูแลระบบ เจ้าหน้าที่คลัง แผนกจัดซื้อ ผู้ใช้งานทั่วไป และแผนกเภสัชกรรม ทั้งนี้ผู้ใช้แต่ละประเภทจะมีสิทธิ์การเข้าใช้งานระบบการจัดการคลังแตกต่างกันออกตามหน้าที่ ดังแสดงในรูปที่ 3.1

### 3.4 การออกแบบแผนภาพกระแสข้อมูล

แผนภาพกระแสข้อมูลจะแสดงให้เห็นถึงทิศทางการไหลของข้อมูลที่มีอยู่ในระบบและการดำเนินงานที่เกิดขึ้นในระบบการจัดการคลัง จากทฤษฎีของแผนภาพบริบทในบทที่ 2 หัวข้อที่ 2.5 สามารถออกแบบแผนผังการไหลของข้อมูลระบบการจัดการคลัง โดยมีกระบวนการ ทำงานหลักอยู่ 9 กระบวนการ คือ เข้าใช้งานระบบและออกจากระบบ ตรวจสอบยอดคงคลัง การออกใบสั่งซื้อ การนำเข้า การเบิก การยืมและการคืน การแจ้งเตือน การออกรายงานต่าง ๆ และการตั้งค่าแก้ไขข้อมูล นอกจากนี้ผู้ใช้งานระบบจะถูกแบ่งออกเป็น 5 ประเภท คือ ผู้ดูแลระบบ เจ้าหน้าที่คลัง แผนกจัดซื้อ ผู้ใช้งานทั่วไป และแผนกเภสัชกรรม ซึ่งแต่ละประเภทจะมีสิทธิ์การเข้าใช้งานกระบวนการทำงานย่อยแตกต่างกันออกตามหน้าที่ดังรายละเอียดสิทธิ์การเข้าใช้งานระบบต่อไปนี้ และแสดงภาพประกอบการเข้าใช้งานระบบของผู้ใช้งานในรูปแบบของ แผนภาพกระแสข้อมูลดังรูปที่ 3.2 – 3.3

1. สิทธิ์ของผู้ดูแลระบบ สามารถเข้าใช้งานระบบได้ตามรายละเอียดที่แสดงภายใต้ข้อความนี้
  - สามารถกำหนดสิทธิ์การเข้าใช้งานของระบบแก่ผู้ใช้งานอื่นได้
  - สามารถบันทึก ตรวจสอบและแก้ไขข้อมูลของระบบทั้งหมดได้
  - สามารถตั้งค่าการแจ้งเตือนต่าง ๆ ได้
  - สามารถตรวจสอบยอดคงคลังทั้งหมดได้
  - สามารถสำรองข้อมูลได้
2. สิทธิ์ของเจ้าหน้าที่แผนกคลัง สามารถเข้าใช้งานระบบตามรายละเอียดที่ได้แสดงภายใต้ข้อความนี้
  - สามารถบันทึก ตรวจสอบและแก้ไขข้อมูลของระบบทั้งหมดได้

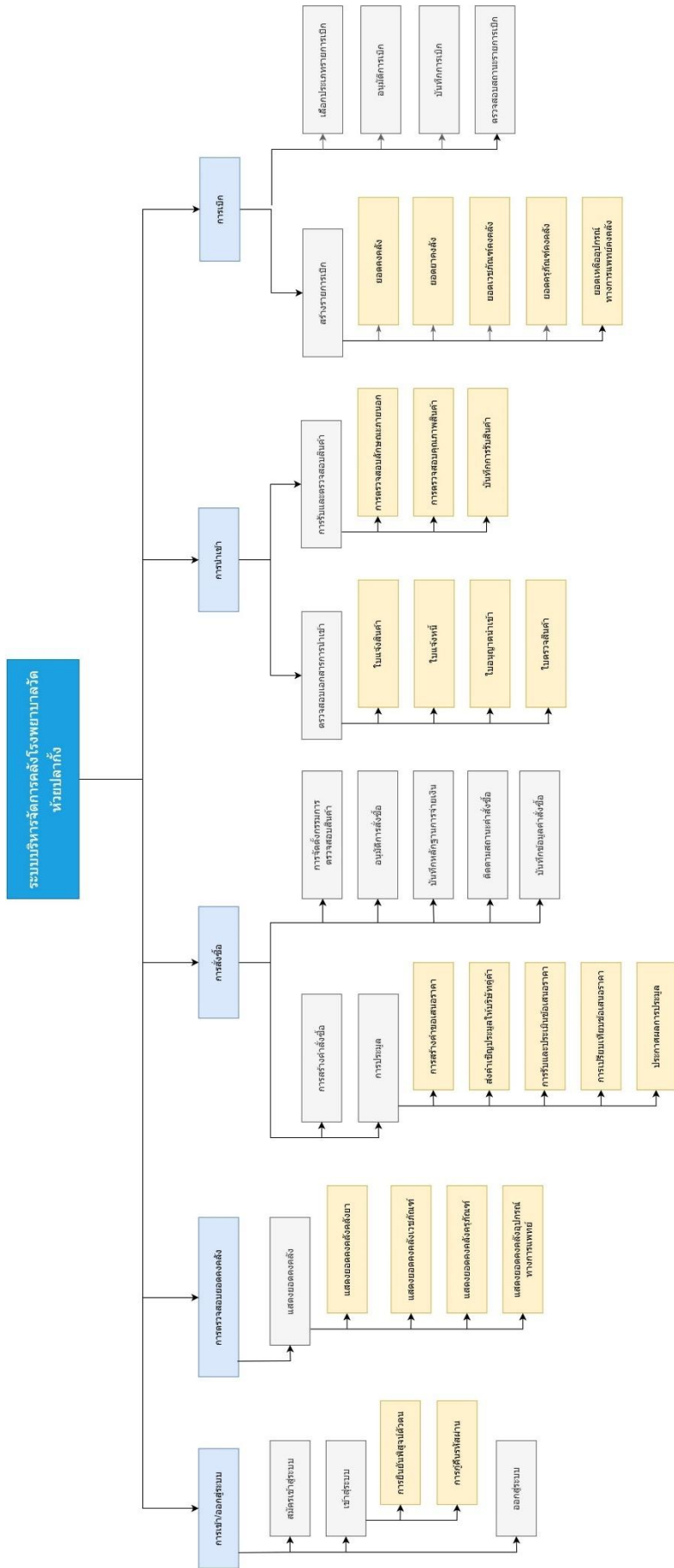
- สามารถเรียกดูคำร้องขอการเบิกได้
  - สามารถจัดการคำร้องขอการเบิกยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์และของใช้ทั่วไปได้
  - สามารถเรียกดูคำร้องขอการยืม/การคืนได้
  - สามารถจัดการคำร้องขอยืม-คืน เวชภัณฑ์ ครุภัณฑ์ และของใช้ทั่วไปได้
  - สามารถนำเข้า/นำออกยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์และของใช้ทั่วไปได้
  - สามารถตั้งค่าการแจ้งเตือนต่าง ๆ ได้
  - สามารถตรวจสอบยอดคงคลังทั้งหมดได้
  - สามารถออกรายงานได้
3. สิทธิ์ของแผนกจัดซื้อทั่วไป สามารถเข้าใช้งานระบบได้ตามรายละเอียดที่ได้แสดงภายใต้ข้อความนี้
- สามารถตรวจสอบยอดคงคลังทั้งหมดได้
  - สามารถเรียกดูคำร้องขอการสั่งซื้อได้
  - สามารถขอเอกสาร TOR ได้
  - สามารถขอจัดตั้งคณะกรรมการตรวจสอบได้
  - สามารถจัดการแก้ไขรายการสั่งซื้อได้
  - สามารถขออนอกคำสั่งซื้อได้
  - สามารถตรวจสอบสถานะการสั่งซื้อได้
  - สามารถออกรายงานการสั่งซื้อได้
4. สิทธิ์ของผู้ใช้ทั่วไป สามารถเข้าใช้งานระบบได้ตามรายละเอียดที่ได้แสดงภายใต้ข้อความนี้
- สามารถดูรายละเอียดข้อมูลคงคลังเวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์และของใช้ทั่วไป
  - สามารถขอเบิกได้
  - สามารถขอยืม-คืน เวชภัณฑ์ ครุภัณฑ์ และของใช้ทั่วไป
  - สามารถแจ้งปัญหา ครุภัณฑ์ อุปกรณ์ทางการแพทย์และของใช้ทั่วไป ที่ได้รับได้
  - สามารถแจ้งเตือนสถานะคำร้องขอการเบิกได้
5. สิทธิ์ของแผนกเภสัชกรรม สามารถเข้าใช้งานระบบได้ตามรายละเอียดที่ได้แสดงภายใต้ข้อความนี้
- สามารถดูรายละเอียดข้อมูลคงคลังยา เวชภัณฑ์ และครุภัณฑ์ได้
  - สามารถขอเบิกยา เวชภัณฑ์ และครุภัณฑ์ได้
  - สามารถยืม-คืน เวชภัณฑ์ และครุภัณฑ์ได้
  - สามารถแจ้งปัญหา ยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์และของใช้ทั่วไป ที่ได้รับ
  - สามารถขอการสั่งซื้อได้
  - สามารถขอการแจ้งเตือนคำร้องขอการเบิกได้

### 3.5 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล

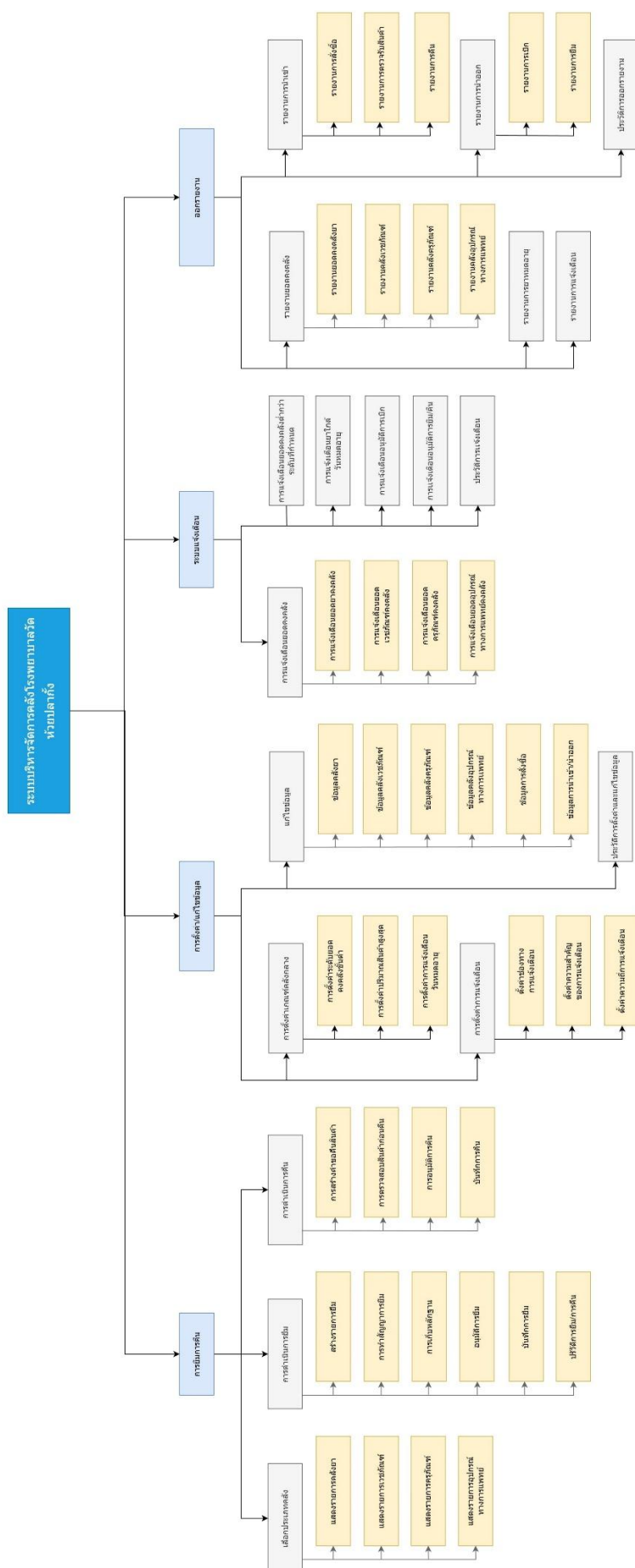
แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลแสดงให้เห็นถึงการจัดเก็บข้อมูลต่าง ๆ และความสัมพันธระหว่างข้อมูลของระบบการจัดการคลัง จากทฤษฎีของแผนภาพบริบทในบทที่ 2 หัวข้อที่ 2. ทำให้สามารถออกแบบแผนภาพแสดงความสัมพันธ์ระหว่างข้อมูลของระบบมี ทั้งหมด 23 ตาราง ได้แก่ users, general\_supplies, inventory, inventory\_report, inventory\_update, stock\_report, purchase\_order, purchase\_order\_documents, purchase\_order\_items, suppliers, receiving, withdrawal, withdrawal\_items, loan, loan\_items, loan\_return\_report, main\_asset, main\_med\_table, main\_supply, medical\_device, notification\_settings, notifications, purchase\_report โดยทางคณะผู้จัดทำนำเสนอแผนภาพความสัมพันธ์ระหว่างข้อมูลในรูปแบบของ Crow's Foot Model มีรายละเอียดข้อมูลต่าง ๆ ดังรูปที่ 3.6-3.7

### 3.6 การออกแบบส่วนติดต่อประสานงานกับผู้ใช้งาน

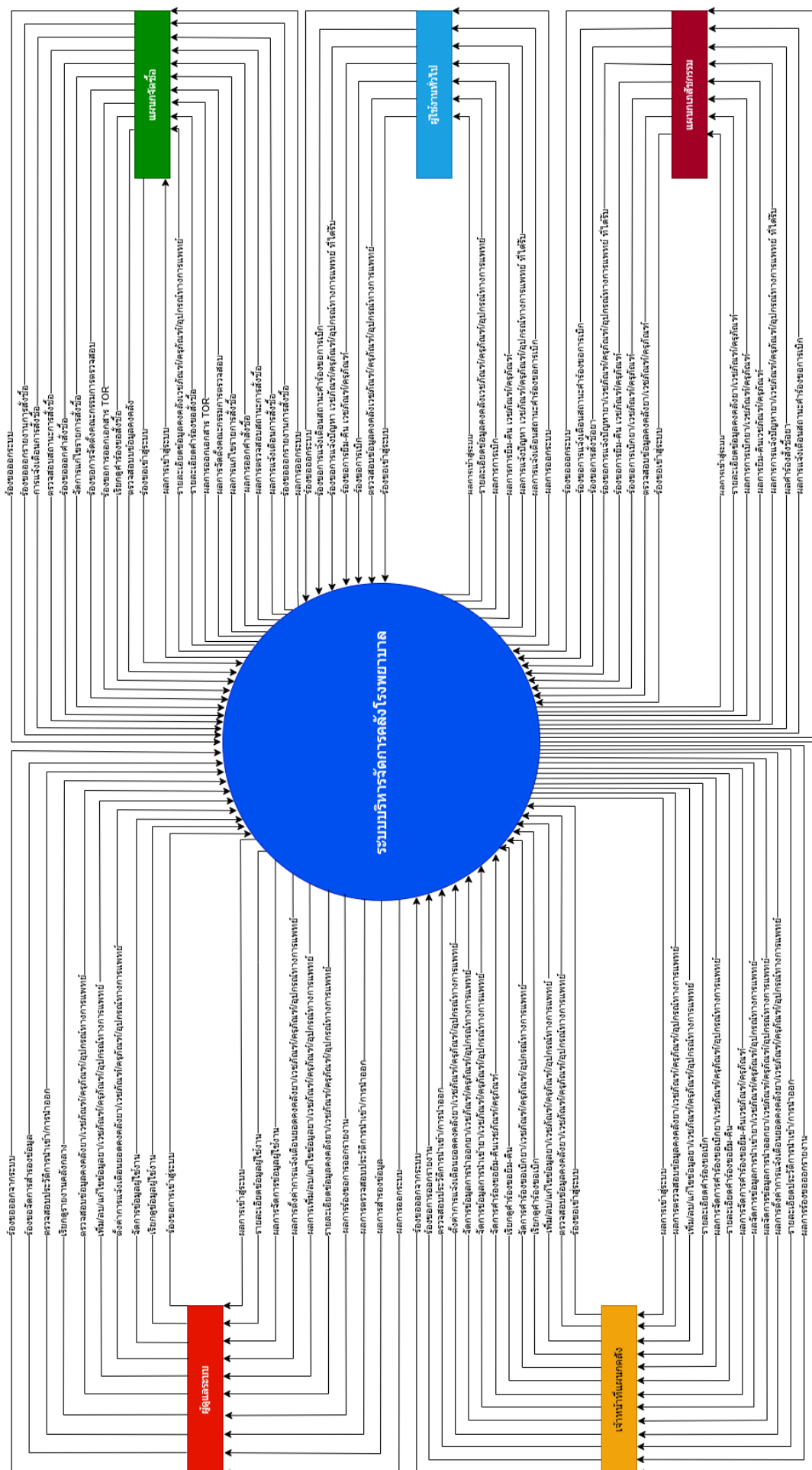
จากการลงพื้นที่สัมภาษณ์ และศึกษาหลักการออกแบบส่วนติดต่อประสานงานกับผู้ใช้งาน เพื่อนำมาออกแบบหน้าเว็บแอปพลิเคชัน ซึ่งผู้จัดทำได้นำเนิการออกแบบหน้าเว็บดังกล่าวสำหรับโมดูล 14 โมดูล ได้แก่ โมดูลเข้าสู่ระบบ โมดูลหลักแสดงรายละเอียดต่าง ๆ โมดูลตรวจสอบยอดคงคลัง โมดูลตรวจสอบรายการเบิก-ยืม โมดูลตรวจสอบรายละเอียดคำร้องขอ โมดูลจัดการแก้ไขข้อมูล โมดูล โมดูลเพิ่มรายการ โมดูลแก้ไขรายการ โมดูลตรวจสอบประวัติการนำเข้า-นำออก โมดูลออกรายงาน โมดูลทำรายการเบิก-ยืม โมดูลทำรายการเบิก โมดูลทำรายการยืม โมดูลตรวจสอบรายการที่เลือก ดังรูปที่ 3.8-3.21



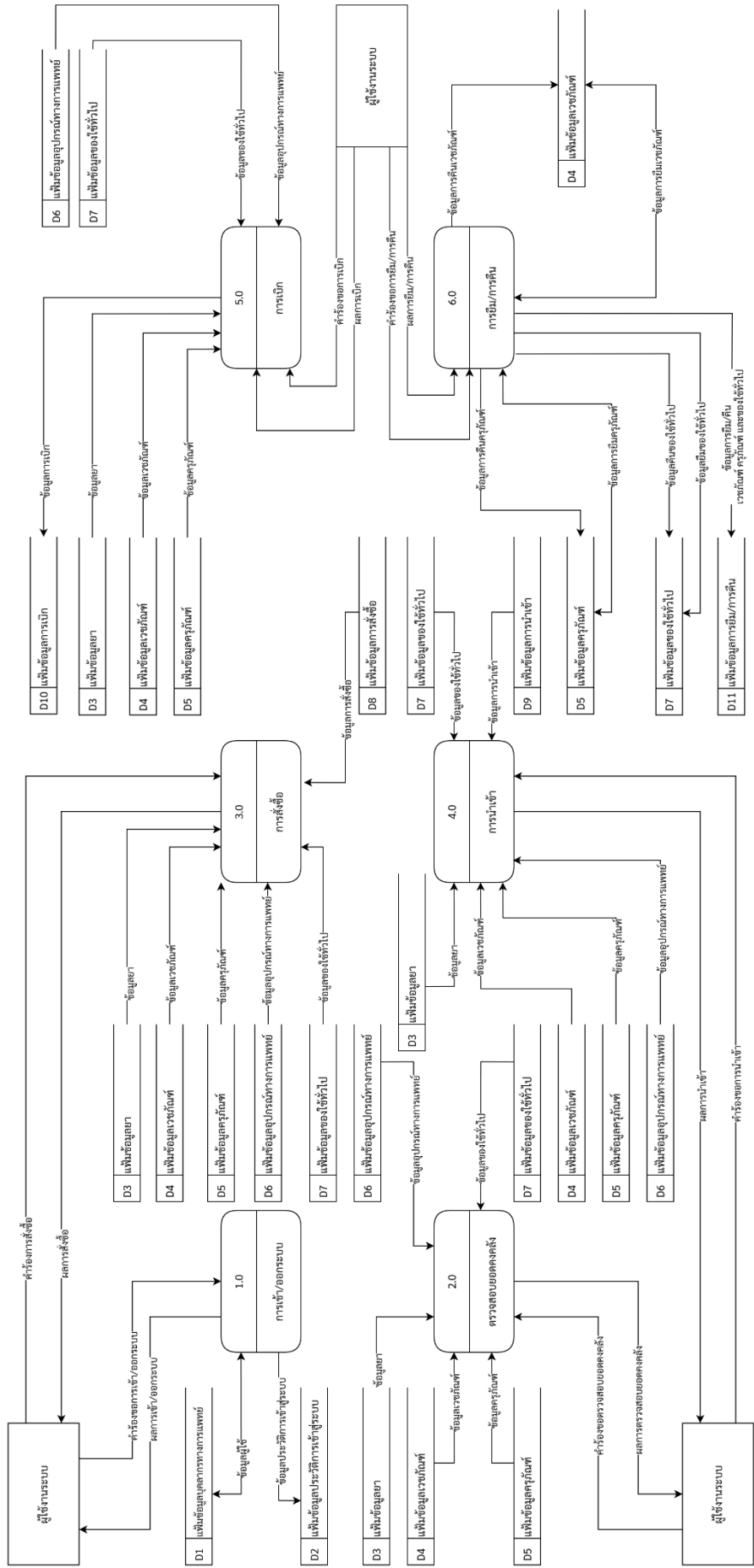
รูปที่ 3.1 แผนภาพต้นไม้ ของระบบการจัดการคลังโรงพยาบาล



รูปที่ 3.2 แผนภาพต้นไม้ ของระบบการจัดการคลังโรงพยาบาล(ต่อ)

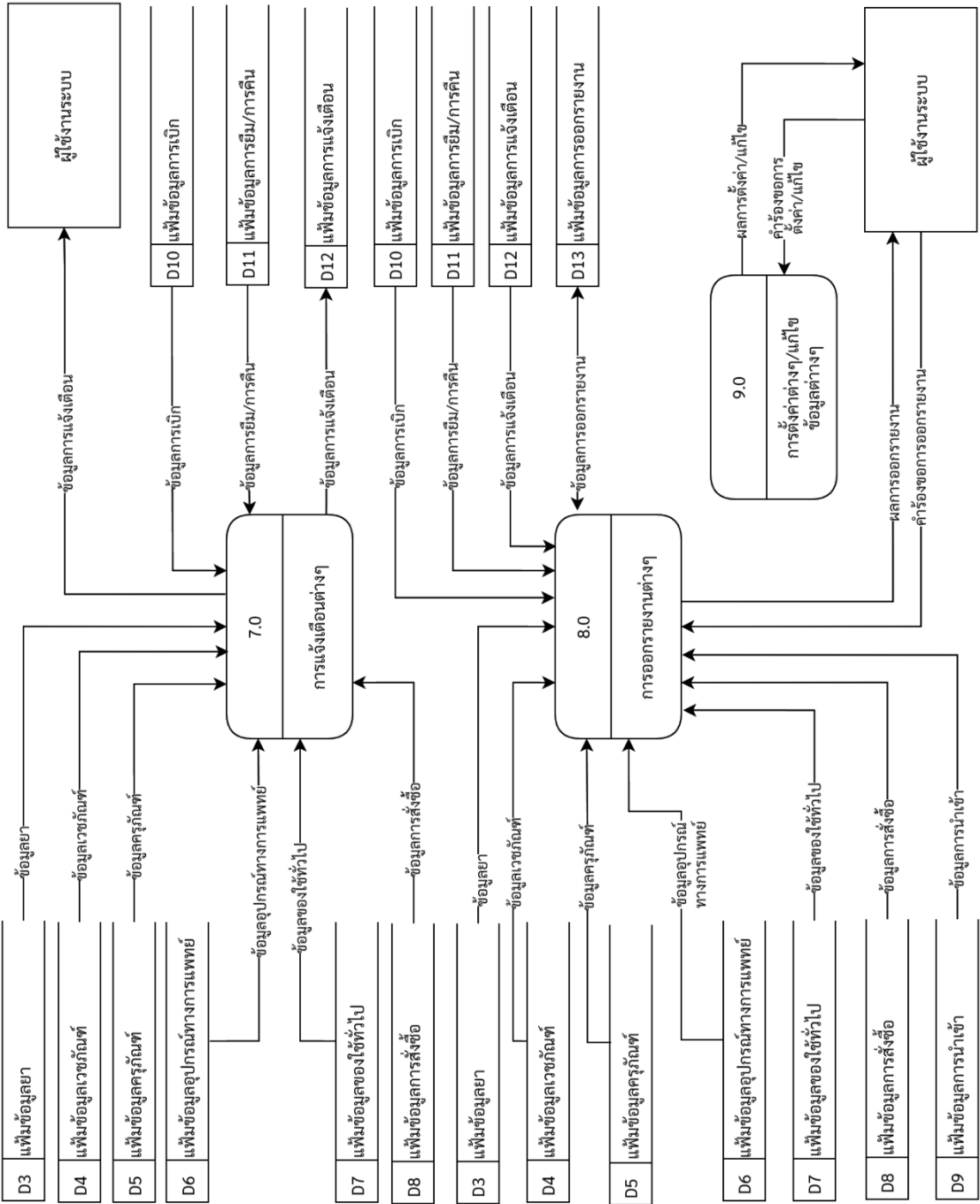


### รูปที่ 3.3 แผนภาพบริบทของระบบการจัดการคลังโรงพยาบาล



รูปที่ 3.4 แผนภาพกระแสข้อมูลระดับที่ 1 ของระบบการจัดการคลังโรงพยาบาล





รูปที่ 3.5 แผนภาพกระแสข้อมูลระดับที่ 1 ของระบบการจัดการคลังโรงพยาบาล(ต่อ)

ตารางที่ 3.1แสดงรายชื่อกระบวนการของระบบในแผนภาพกระแสข้อมูลระดับที่ 1

กระบวนการที่	รายละเอียดกระบวนการ
1	การเข้า/ออกระบบ
2	ตรวจสอบยอดคงคลัง
3	การสั่งซื้อ
4	การนำเข้า
5	การเบิก
6	การยืม/การคืน
7	การตั้งค่าต่าง ๆ /แก้ไขข้อมูลต่าง ๆ
8	การแจ้งเตือนต่าง ๆ
9	การออกรายงานต่าง ๆ

แผนภาพกระแสข้อมูลระดับที่ 1 กระบวนการที่ 1 คือการเข้า/ออกระบบ การตรวจสอบยอดคงคลังเป็น กระบวนการสำหรับเข้า/ออกระบบ โดยกระบวนการเข้า/ออกระบบได้แสดง รายละเอียดการไหลของข้อมูลไว้ดังตารางที่ 3.2

ตารางที่ 3.2 แสดงกระบวนการที่ 1 การเข้า/ออกระบบ

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
กระบวนการที่ :	1
ชื่อกระบวนการ :	การเข้า/ออกระบบ
ข้อมูลนำเข้า :	- คำร้องขอการเข้าสู่ระบบ - คำร้องขอการออกระบบ
ข้อมูลนำออก :	- ผลการเข้าสู่ระบบ - ผลการออกระบบ
แหล่งจัดเก็บข้อมูล :	- แฟ้มข้อมูลผู้ใช้งาน - แฟ้มข้อมูลประวัติการเข้าสู่ระบบ
คำอธิบายการทำงาน :	1. ผู้ใช้งานเข้าสู่ระบบและออกระบบ

ตารางที่ 3.2 แสดงกระบวนการที่ 1 การเข้า/ออกระบบ(ต่อ)

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
	3. แสดงข้อมูลของยอดคงคลังตามที่ใช้ใช้งาน ต้องการตรวจสอบ

แผนภาพกระแสข้อมูลระดับที่ 1 กระบวนการที่ 2 คือการตรวจสอบยอดคงคลังเป็นกระบวนการสำหรับตรวจสอบยอดคงคลังของยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์ และของใช้ทั่วไป ที่อยู่ในคลังว่ามีจำนวนเหลือน้อยมากน้อยเพียงใด โดยกระบวนการตรวจสอบยอดคงคลังได้แสดงรายละเอียดการไหลของข้อมูลไว้ดังตารางที่ 3.3

ตารางที่ 3.3 แสดงกระบวนการที่ 2 ตรวจสอบยอดคงคลัง

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
กระบวนการที่ :	2
ชื่อกระบวนการ :	ตรวจสอบยอดคงคลัง
ข้อมูลนำเข้า :	คำร้องขอตรวจสอบยอดคงคลัง
ข้อมูลนำออก :	ผลการตรวจสอบยอดคงคลัง
แหล่งจัดเก็บข้อมูล :	<ul style="list-style-type: none"> <li>- แฟ้มข้อมูลยา</li> <li>- แฟ้มข้อมูลเวชภัณฑ์</li> <li>- แฟ้มข้อมูลครุภัณฑ์</li> <li>- แฟ้มข้อมูลอุปกรณ์ทางการแพทย์</li> <li>- แฟ้มข้อมูลของใช้ทั่วไป</li> </ul>
คำอธิบายการทำงาน :	<ol style="list-style-type: none"> <li>1. ผู้ใช้งานทำการเลือกสิ่งที่ต้องการจะทำการตรวจสอบยอดคงคลัง</li> <li>2. ระบบทำการดึงข้อมูลจากแฟ้มข้อมูลต่าง ๆ ที่ใช้ต้องการตรวจสอบ</li> <li>3. แสดงข้อมูลของยอดคงคลังตามที่ใช้ใช้งาน ต้องการตรวจสอบ</li> </ol>

แผนภาพกระแสข้อมูลระดับที่ 1 กระบวนการที่ 3 คือการออกไปสั่งซื้อเป็นกระบวนการสำหรับออกไปสั่งซื้อยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์ และของใช้ทั่วไปที่จะนำมาจัดเก็บไว้ในคลัง โดยกระบวนการออกไปสั่งซื้อได้แสดงรายละเอียดการไหลของข้อมูลไว้ดังตารางที่ 3.4

ตารางที่ 3.4 แสดงกระบวนการที่ 3 การสั่งซื้อ

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
กระบวนการที่ :	3
ชื่อกระบวนการ :	การสั่งซื้อ
ข้อมูลนำเข้า :	คำร้องขอการสั่งซื้อ
ข้อมูลนำออก :	ผลการสั่งซื้อ
แหล่งจัดเก็บข้อมูล :	<ul style="list-style-type: none"> <li>- แฟ้มข้อมูลการสั่งซื้อ</li> <li>- แฟ้มข้อมูลยา</li> <li>- แฟ้มข้อมูลเวชภัณฑ์</li> <li>- แฟ้มข้อมูลครุภัณฑ์</li> <li>- แฟ้มข้อมูลอุปกรณ์ทางการแพทย์</li> <li>- แฟ้มข้อมูลของใช้ทั่วไป</li> </ul>
คำอธิบายการทำงาน :	<ol style="list-style-type: none"> <li>1. ผู้ใช้งานเลือกประเภทสิ่งที่ต้องการจะออกใบสั่งซื้อ</li> <li>2. ผู้ใช้งานทำการกำหนดรายละเอียดการออกใบสั่งซื้อ</li> <li>3. ระบบทำการดึงข้อมูลจากแฟ้มข้อมูลตามสิ่ง que ผู้ใช้งานได้ทำการกำหนดรายละเอียด</li> <li>4. รวบรวมการอนุมัติการออกใบสั่งซื้อ</li> <li>5. แสดงผลการออกใบสั่งซื้อ</li> </ol>

แผนภาพกระแสข้อมูลระดับที่ 1 กระบวนการที่ 4 คือการนำเข้าเป็น กระบวนการสำหรับการนำเข้ายา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์ และของใช้ทั่วไป เพื่อทำการจัดเก็บรักษา โดยกระบวนการการนำเข้าได้แสดงรายละเอียดการไหลของข้อมูลไว้ดังตารางที่ 3.5

ตารางที่ 3.5 แสดงกระบวนการที่ 4 การนำเข้า

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
กระบวนการที่ :	4
ชื่อกระบวนการ :	การนำเข้า
ข้อมูลนำเข้า :	คำร้องขอการนำเข้า
ข้อมูลนำออก :	ผลการนำเข้า
แหล่งจัดเก็บข้อมูล :	<ul style="list-style-type: none"> <li>- แฟ้มข้อมูลการนำเข้า</li> <li>- แฟ้มข้อมูลยา</li> </ul>

ตารางที่ 3.5 แสดงกระบวนการที่ 4 การนำเข้า(ต่อ)

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
แหล่งจัดเก็บข้อมูล :	<ul style="list-style-type: none"> <li>- แฟ้มข้อมูลเวชภัณฑ์</li> <li>- แฟ้มข้อมูลครุภัณฑ์</li> <li>- แฟ้มข้อมูลอุปกรณ์ทางการแพทย์</li> <li>- แฟ้มข้อมูลของใช้ทั่วไป</li> </ul>
คำอธิบายการทำงาน :	<ol style="list-style-type: none"> <li>1. ผู้ใช้งานเลือกรายการนำเข้า</li> <li>2. ผู้ใช้งานกรอกข้อมูลสิ่งที่ต้องการนำเข้า</li> <li>3. รอการอนุมัติการนำเข้า</li> <li>4. แสดงผลการนำเข้า</li> <li>5. ระบบทำการจัดเก็บและบันทึกข้อมูลสิ่งที่นำเข้า</li> </ol>

แผนภาพกระแสข้อมูลระดับที่ 1 กระบวนการที่ 5 คือการเบิกเป็น กระบวนการสำหรับการเบิกยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์ และของใช้ทั่วไป เพื่อนำออกไปใช้ โดยกระบวนการเบิกได้แสดงรายละเอียดการไหลของข้อมูลไว้ดังตารางที่ 3.6

ตารางที่ 3.6 แสดงกระบวนการที่ 5 การเบิก

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
กระบวนการที่ :	5
ชื่อกระบวนการ :	การเบิก
ข้อมูลนำเข้า :	คำร้องขอการเบิก
ข้อมูลนำออก :	ผลการเบิก
แหล่งจัดเก็บข้อมูล :	<ul style="list-style-type: none"> <li>- แฟ้มข้อมูลการเบิก</li> <li>- แฟ้มข้อมูลยา</li> <li>- แฟ้มข้อมูลเวชภัณฑ์</li> <li>- แฟ้มข้อมูลครุภัณฑ์</li> <li>- แฟ้มข้อมูลอุปกรณ์ทางการแพทย์</li> <li>- แฟ้มข้อมูลของใช้ทั่วไป</li> </ul>
คำอธิบายการทำงาน :	<ol style="list-style-type: none"> <li>1. ผู้ใช้งานเลือกรายการเบิก</li> <li>2. ผู้ใช้งานกรอกข้อมูลสิ่งที่ต้องการเบิก</li> <li>3. รอการอนุมัติการเบิก</li> <li>4. แสดงผลการเบิก</li> </ol>

ตารางที่ 3.6 แสดงกระบวนการที่ 5 การเบิก(ต่อ)

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
คำอธิบายการทำงาน :	5. ระบบทำการจัดเก็บและบันทึกข้อมูลที่เบิก

แผนภาพกระแสข้อมูลระดับที่ 1 กระบวนการที่ 6 คือการยืม/การคืนเวชภัณฑ์ ครุภัณฑ์ และของใช้ทั่วไปเป็นกระบวนการสำหรับยืมและคืนของ เพื่อนำออกไปใช้งานและนำมาจัดเก็บรักษาหลังการใช้งานโดยกระบวนการการยืม/การคืนได้แสดงรายละเอียดการไหลของข้อมูลไว้ดังตารางที่ 3.7

ตารางที่ 3.7 แสดงกระบวนการที่ 6 การยืม/การคืน

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
กระบวนการที่ :	6
ชื่อกระบวนการ :	การสั่งซื้อ
ข้อมูลนำเข้า :	- คำร้องขอการยืม - คำร้องขอการคืน
ข้อมูลนำออก :	- ผลการยืม - ผลการคืน
แหล่งจัดเก็บข้อมูล :	- แฟ้มข้อมูลการยืม/การคืน - แฟ้มข้อมูลเวชภัณฑ์ - แฟ้มข้อมูลครุภัณฑ์ - แฟ้มข้อมูลของใช้ทั่วไป
คำอธิบายการทำงาน :	1. ผู้ใช้งานเลือกว่าจะยืมหรือคืน 2. ผู้ใช้งานกรอกข้อมูลสิ่งที่ต้องการยืมหรือคืน 3. รออนุมัติการยืมและการคืน 4. แสดงผลการยืมและการคืน 5. ระบบทำการจัดเก็บและบันทึกข้อมูลการยืมและการคืน

แผนภาพกระแสข้อมูลระดับที่ 1 กระบวนการที่ 7 คือการแจ้งเตือนต่างๆ เป็นกระบวนการสำหรับการแจ้งเตือนเกี่ยวกับยา เวชภัณฑ์ ครุภัณฑ์ อุปกรณ์ทางการแพทย์ และของใช้ทั่วไป คงคลังต่ำกว่าที่กำหนด รวมถึงการหมดอายุของยา และปริมาณยาต่ำกว่าที่กำหนด เพื่อผู้ใช้งานระบบเข้าไปแก้ไขและจัดการสิ่งเหล่านั้นให้ทัน่วงทีก่อนที่จะเกิดความเสียหายภายในคลัง โดยกระบวนการแจ้งเตือนต่างๆ ได้แสดงรายละเอียดการไหลของข้อมูลไว้ดังตารางที่ 3.8

ตารางที่ 3.8 แสดงกระบวนการที่ 7 การแจ้งเตือนต่าง ๆ

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
กระบวนการที่ :	7
ชื่อกระบวนการ :	การแจ้งเตือน
ข้อมูลนำเข้า :	
ข้อมูลนำออก :	ข้อมูลการแจ้งเตือน
แหล่งจัดเก็บข้อมูล :	<ul style="list-style-type: none"> <li>- แฟ้มข้อมูลการแจ้งเตือน</li> <li>- แฟ้มข้อมูลยา</li> <li>- แฟ้มข้อมูลเวชภัณฑ์</li> <li>- แฟ้มข้อมูลครุภัณฑ์</li> <li>- แฟ้มข้อมูลอุปกรณ์ทางการแพทย์</li> <li>- แฟ้มข้อมูลของใช้ทั่วไป</li> <li>- แฟ้มข้อมูลการสั่งซื้อ</li> <li>- แฟ้มข้อมูลการเบิก</li> <li>- แฟ้มข้อมูลการยืม/การคืน</li> </ul>
คำอธิบายการทำงาน :	<ol style="list-style-type: none"> <li>1. ระบบทำการรวบรวมข้อมูลสิ่งต่าง ๆ ที่อยู่ภายในคลังตามเกณฑ์ที่ระบบได้กำหนดค่าเอาไว้</li> <li>2. ระบบทำการแยกประเภทของสิ่งจะทำการแจ้งเตือนไปยังผู้ใช้งาน</li> <li>3. ส่งข้อมูลการแจ้งเตือนไปยังผู้ใช้งาน</li> <li>4. ระบบทำการบันทึกจัดเก็บสิ่งที่แจ้งเตือน</li> </ol>

แผนภาพกระแสข้อมูลระดับที่ 1 กระบวนการที่ 8 คือการออกรายงานต่างๆ เป็นกระบวนการสำหรับออกรายงานการสั่งซื้อ นำเข้า การเบิก การยืม การคืน และการแจ้งเตือนต่างๆ เพื่อเป็นการผลสรุปผลสิ่งต่างๆ ที่เกิดขึ้นภายในคลังให้แก่ผู้ใช้งาน โดยกระบวนการการ แจ้งเตือนต่างๆ ได้แสดงรายละเอียดการไหลของข้อมูลไว้ดังตารางที่ 3.9

ตารางที่ 3.9 แสดงกระบวนการที่ 8 การออกรายงานต่าง ๆ

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
กระบวนการที่ :	8
ชื่อกระบวนการ :	การออกรายงานต่าง ๆ
ข้อมูลนำเข้า :	คำร้องขอการออกรายงาน
ข้อมูลนำออก :	ผลการออกรายงาน

ตารางที่ 3.9 แสดงกระบวนการที่ 8 การออกรายงานต่าง ๆ (ต่อ)

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
แหล่งจัดเก็บข้อมูล :	<ul style="list-style-type: none"> <li>- แฟ้มข้อมูลยา</li> <li>- แฟ้มข้อมูลเวชภัณฑ์</li> <li>- แฟ้มข้อมูลครุภัณฑ์</li> <li>- แฟ้มข้อมูลอุปกรณ์ทางการแพทย์</li> <li>- แฟ้มข้อมูลของใช้ทั่วไป</li> <li>- แฟ้มข้อมูลการสั่งซื้อ</li> <li>- แฟ้มข้อมูลการนำ</li> <li>- แฟ้มข้อมูลการเบิก</li> <li>- แฟ้มข้อมูลการยืม/การคืน</li> <li>- แฟ้มข้อมูลการแจ้งเตือน</li> <li>- แฟ้มข้อมูลการออกรายงาน</li> </ul>
คำอธิบายการทำงาน :	<ol style="list-style-type: none"> <li>1. ผู้ใช้งานทำการเลือกสิ่งที่ต้องการจะนำไปออกรายงาน</li> <li>2. กำหนดค่าต่าง ๆ ในการออกรายงานเช่นวันที่ต้องการจะออกรายงานตั้งแต่วันที่เท่าไร เป็นต้น</li> <li>3. แสดงผลการออกรายงาน</li> </ol>

แผนภาพกระแสข้อมูลระดับที่ 1 กระบวนการที่ 9 คือการตั้งค่าต่าง ๆ แก่ไขข้อมูลต่าง ๆ เป็นกระบวนการสำหรับการกำหนดค่าให้แก่ระบบและทำการแก้ไขข้อมูลต่าง ๆ ภายในคลังเช่น ข้อมูลยา ข้อมูลเวชภัณฑ์ ข้อมูลเวชภัณฑ์ ข้อมูลครุภัณฑ์ ข้อมูลอุปกรณ์ทางการแพทย์ ข้อมูลของใช้ทั่วไป และข้อมูลการสั่งซื้อ เป็นต้น อีกทั้งยังรวมถึงการแก้ไขข้อมูลส่วนของของผู้ใช้งานอีกด้วย โดยกระบวนการการตั้งค่าต่าง ๆ แก่ไขข้อมูลต่าง ๆ ได้แสดงรายละเอียดการไหลของข้อมูลไว้ดังตารางที่ 3.10

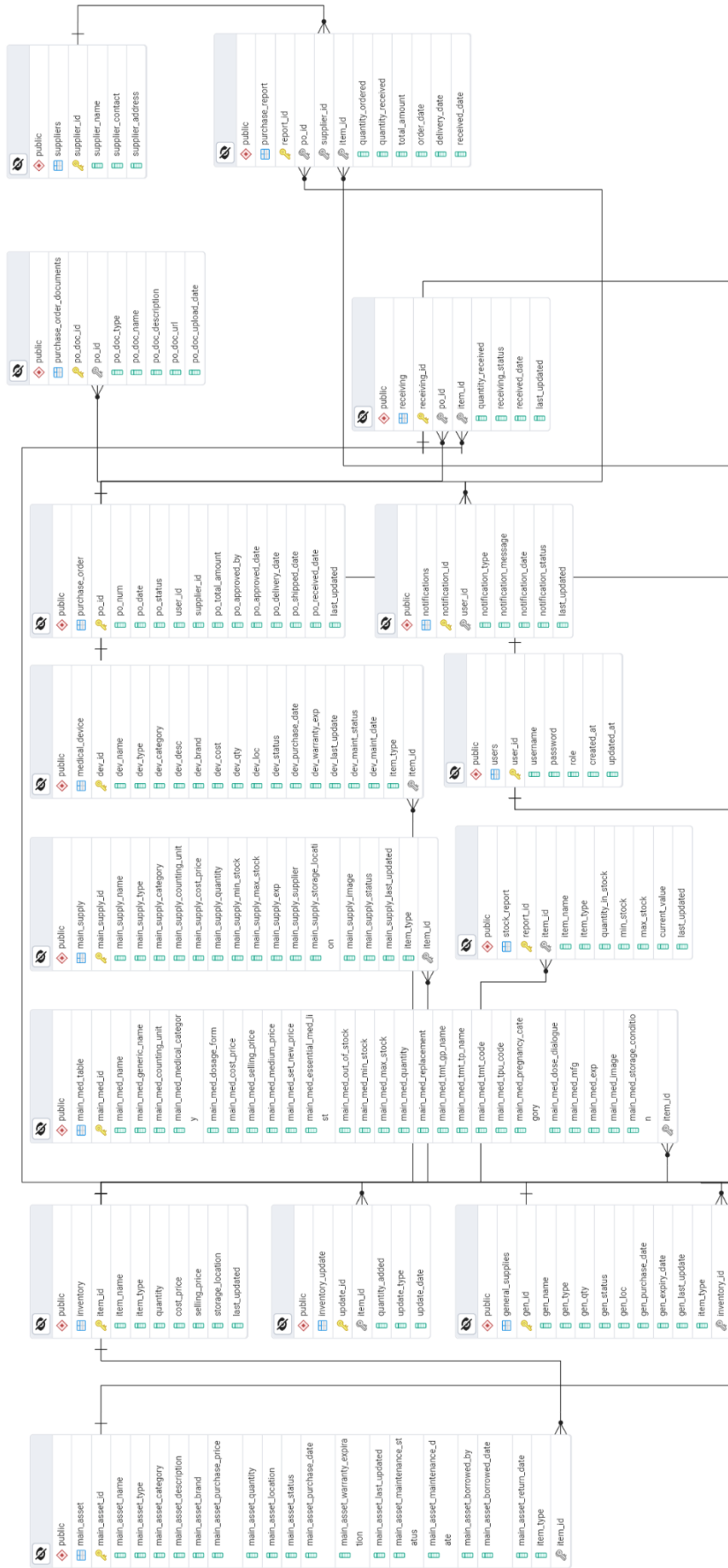
ตารางที่ 3.10 แสดงกระบวนการที่ 9 การตั้งค่าต่าง ๆ แก่ไขข้อมูลต่าง ๆ

ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
กระบวนการที่ :	9
ชื่อกระบวนการ :	การตั้งค่าต่าง ๆ /แก้ไขข้อมูลต่าง ๆ
ข้อมูลนำเข้า :	<ul style="list-style-type: none"> <li>- คำร้องขอการตั้งค่า</li> <li>- คำร้องขอการแก้ไข</li> </ul>
ข้อมูลนำออก :	- ผลการตั้งค่า

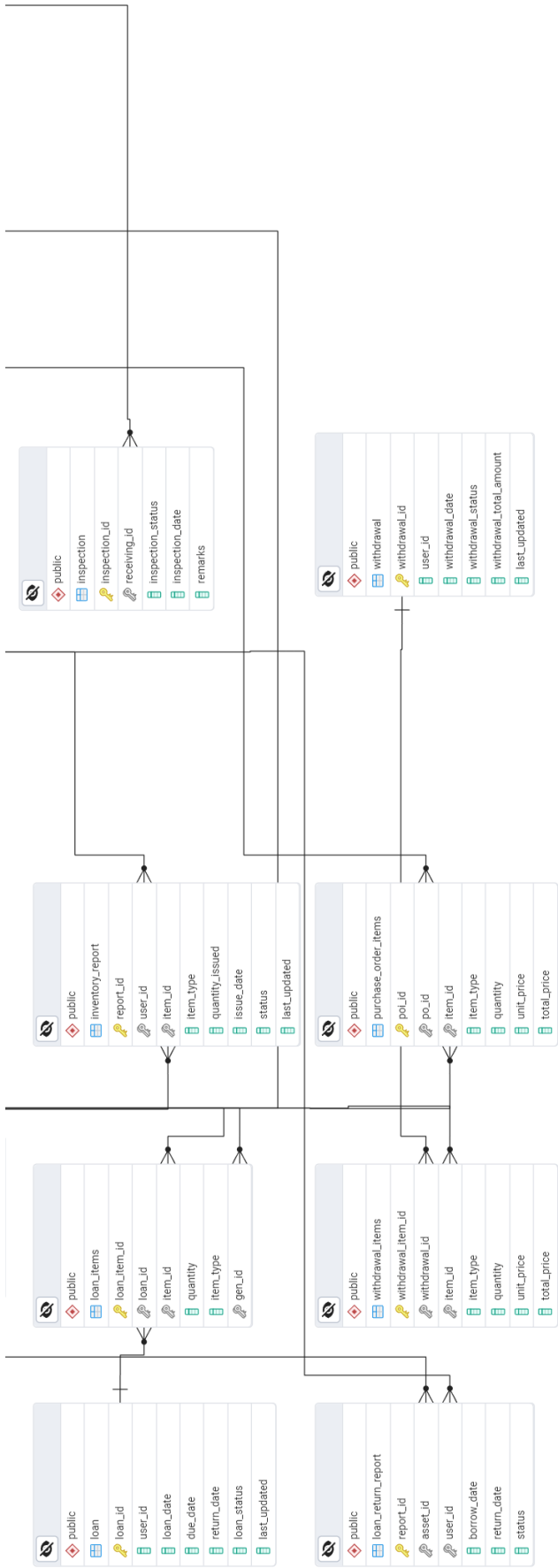


ตารางที่ 3.10 แสดงกระบวนการที่ 9 การตั้งค่าต่าง ๆ แก่ไขข้อมูลต่าง ๆ (ต่อ)

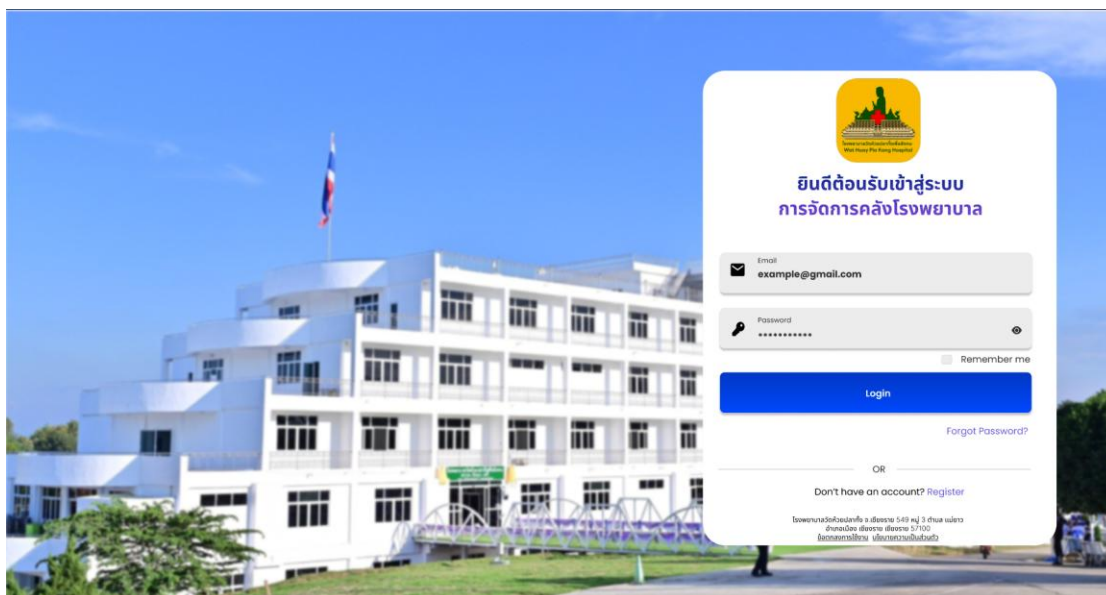
ชื่อระบบ :	ระบบการจัดการคลังโรงพยาบาล
ข้อมูลนำออก :	- ผลการแก้ไข
แหล่งจัดเก็บข้อมูล :	<ul style="list-style-type: none"> <li>- แฟ้มข้อมูลยา</li> <li>- แฟ้มข้อมูลเวชภัณฑ์</li> <li>- แฟ้มข้อมูลครุภัณฑ์</li> <li>- แฟ้มข้อมูลอุปกรณ์ทางการแพทย์</li> <li>- แฟ้มข้อมูลของใช้ทั่วไป</li> <li>- แฟ้มข้อมูลการสั่งซื้อ</li> <li>- แฟ้มข้อมูลการนำเข้า</li> <li>- แฟ้มข้อมูลการเบิก</li> <li>- แฟ้มข้อมูลการยืม/การคืน</li> <li>- แฟ้มข้อมูลการแจ้งเตือน</li> <li>- แฟ้มข้อมูลการออกรายงาน</li> </ul>



รูปที่ 3.6 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล ของระบบการจัดการคลังโรงพยาบาล



รูปที่ 3.7 แผนภาพแสดงความสัมพันธ์ระหว่างข้อมูล ของระบบการจัดการคลังโรงพยาบาล(ต่อ)



รูปที่ 3.8 หน้าเข้าสู่ระบบ



รูปที่ 3.9 หน้าหลักแสดงรายละเอียดต่าง ๆ

แผนกคลังหลักโรงพยาบาล								
ตรวจสอบยอดคงคลัง			ยา	เวชภัณฑ์	ครุภัณฑ์	อุปกรณ์ทางการแพทย์	ของใช้ทั่วไป	
หมายเลข	รูปภาพ	ชื่อ	หมวดหมู่	จำนวน	หน่วย	สถานะ	สถานที่จัดเก็บ	วันที่หมดอายุ
12345		ผ้าพันแผล	เวชภัณฑ์	100	กล่อง	พร้อมใช้งาน	คลังกลาง	2025-12-31
12034		ยาพาราเซตามอล	ยา	200	กระปุก	พร้อมใช้งาน	คลังยา	2025-03-06
12734		เข็มฉีดยา	อุปกรณ์ทางการแพทย์	300	กล่อง	ใกล้หมด	คลังอุปกรณ์แพทย์	2025-03-05
20547		หน้ากากอนามัย	เวชภัณฑ์	50	กล่อง	ใกล้หมด	คลังกลาง	2025-03-04
27070		น้ำเกลือ 0.9%	ยา	100	กล่อง	พร้อมใช้งาน	คลังกลาง	2025-03-06
25407		ถุงมือแพทย์	เวชภัณฑ์	200	กล่อง	ใกล้หมด	คลังกลาง	2025-03-06
13404		เครื่องช่วยหายใจ	อุปกรณ์ทางการแพทย์	5	เครื่อง	พร้อมใช้งาน	ห้องฉุกเฉิน	2025-03-06
10244		สำลี	เวชภัณฑ์	200	ห่อ	พร้อมใช้งาน	คลังกลาง	2025-03-06

รูปที่ 3.10 หน้าตรวจสอบยอดคงคลัง

แผนกคลังหลักโรงพยาบาล								
ตรวจสอบรายการเบิก ยืม								
หมายเลข	วันที่เบิก	เวลาเบิก	ผู้เบิก	แผนก	จำนวนรายการ	ประเภท	สถานะ	ตรวจสอบ
12345	27/02/2025	10:30 AM	นางนงนุช	ฉุกเฉิน	2 รายการ	เบิก	รอดำเนินการ	
12005	27/02/2025	11:00 AM	นพ.สมชาย	อำนวยการ	3 รายการ	เบิก	รอดำเนินการ	
00123	27/02/2025	11:45 AM	พญ.สสส	ผู้ป่วยนอก	6 รายการ	เบิก	รอดำเนินการ	

รูปที่ 3.11 หน้าตรวจสอบรายการเบิก ยืม

แดชบอร์ดหลักโรงพยาบาล

ตรวจสอบรายละเอียดคำร้องขอ

หมายเลข	รูปภาพ	ชื่อ	จำนวน	หน่วย	หมวดหมู่	สถานะ
00123		ผ้าพันแผล	5	กล่อง	เวชภัณฑ์	✓ มีของ
00456		ยาพาราเซตามอล	10	กระป๋อง	ยา	✓ มีของ
00789		เข็มฉีดยา	3	กล่อง	อุปกรณ์การแพทย์	✓ มีของ
01012		น้ำเกลือ 0.9%	30	ขวด	เวชภัณฑ์	✓ มีของ

ข้อมูลคำร้อง

หมายเลขคำขอ: 12345  
วันที่ขอเบิก: 06/03/2025  
ผู้ขอเบิก: บพ.สมชาย  
แผนก/หน่วยงาน: แผนกเภสัชกรรม  
จำนวนรายการ: 4 รายการ  
สถานะคำขอ: รอดำเนินการ  
วันที่ต้องการรับวัสดุ: 01/03/2025

สถานะการนำส่ง

- ส่งคำขอ: 2025-03-01 10:15
- รอตรวจสอบก่อน: 2025-03-01 11:00
- กำลังจัดส่ง
- อยู่ระหว่างนำส่ง
- เสร็จสิ้น

ยกเลิก    อนุมัติ

ออกจากระบบ

รูปที่ 3.12 หน้าตรวจสอบรายละเอียดคำร้องขอ

แดชบอร์ดหลักโรงพยาบาล

จัดการแก้ไขข้อมูล

หมวดหมู่:  หน่วย:  สถานะ:

+ เพิ่มรายการ

หมายเลข	รูปภาพ	ชื่อ	หมวดหมู่	จำนวน	หน่วย	สถานะ	สถานที่จัดเก็บ	วันหมดอายุ	แก้ไขล่าสุด	การจัดการ
12345		ผ้าพันแผล	เวชภัณฑ์	100	กล่อง	พร้อมใช้งาน	คลังกลาง	2025-12-31	2025-11-30	<input type="button" value="แก้ไข"/> <input type="button" value="ลบ"/>
12034		ยาพาราเซตามอล	ยา	200	กระป๋อง	พร้อมใช้งาน	คลังยา	2025-12-31	2025-03-06	<input type="button" value="แก้ไข"/> <input type="button" value="ลบ"/>
12734		เข็มฉีดยา	อุปกรณ์ทางการแพทย์	50	กล่อง	ใกล้หมด	คลังอุปกรณ์แพทย์	2025-12-31	2025-03-05	<input type="button" value="แก้ไข"/> <input type="button" value="ลบ"/>
20547		หน้ากากอนามัย	เวชภัณฑ์	100	กล่อง	ใกล้หมด	คลังกลาง	2025-12-31	2025-03-04	<input type="button" value="แก้ไข"/> <input type="button" value="ลบ"/>
27070		น้ำเกลือ 0.9%	เวชภัณฑ์	200	ขวด	พร้อมใช้งาน	คลังกลาง	2025-12-31	2025-03-06	<input type="button" value="แก้ไข"/> <input type="button" value="ลบ"/>
25407		ถุงมือแพทย์	เวชภัณฑ์	5	กล่อง	ใกล้หมด	คลังกลาง	2025-12-31	2025-03-06	<input type="button" value="แก้ไข"/> <input type="button" value="ลบ"/>
13404		เครื่องช่วยหายใจ	อุปกรณ์ทางการแพทย์	200	เครื่อง	พร้อมใช้งาน	ห้องฉุกเฉิน	-	2025-03-06	<input type="button" value="แก้ไข"/> <input type="button" value="ลบ"/>

ออกจากระบบ

รูปที่ 3.13 หน้าจัดการแก้ไขข้อมูล

แผนกคลังหลักโรงพยาบาล

### เพิ่มรายการ

ชื่อ					
หมายเลข	เลข Lot			ตำแหน่งที่จัดเก็บ	
หมวดหมู่หลัก	หมวดหมู่ย่อย	สถานะพัสดุ	วันที่หมดอายุ		
จำนวนคงเหลือ	หน่วย	จำนวนขั้นต่ำ	จำนวนสูงสุด		

ผู้จำหน่าย	เลขที่ใบสั่งซื้อ	วันที่สั่งซื้อ
ผู้รับเข้า		

ยกเลิก    บันทึก    บันทึก

รูปที่ 3.14 หน้าเพิ่มรายการ

แผนกคลังหลักโรงพยาบาล

### แก้ไขรายการ

ชื่อ					
หมายเลข	เลข Lot			ตำแหน่งที่จัดเก็บ	
หมวดหมู่หลัก	หมวดหมู่ย่อย	สถานะพัสดุ	วันที่หมดอายุ		
จำนวนคงเหลือ	หน่วย	จำนวนขั้นต่ำ	จำนวนสูงสุด		





















ผู้จำหน่าย	เลขที่ใบสั่งซื้อ	วันที่สั่งซื้อ
บริษัท A	PO123456	2024-03-01
ผู้รับเข้า		
นายสมศักดิ์		

ยกเลิก    บันทึก    บันทึก

รูปที่ 3.15 หน้าแก้ไขรายการ

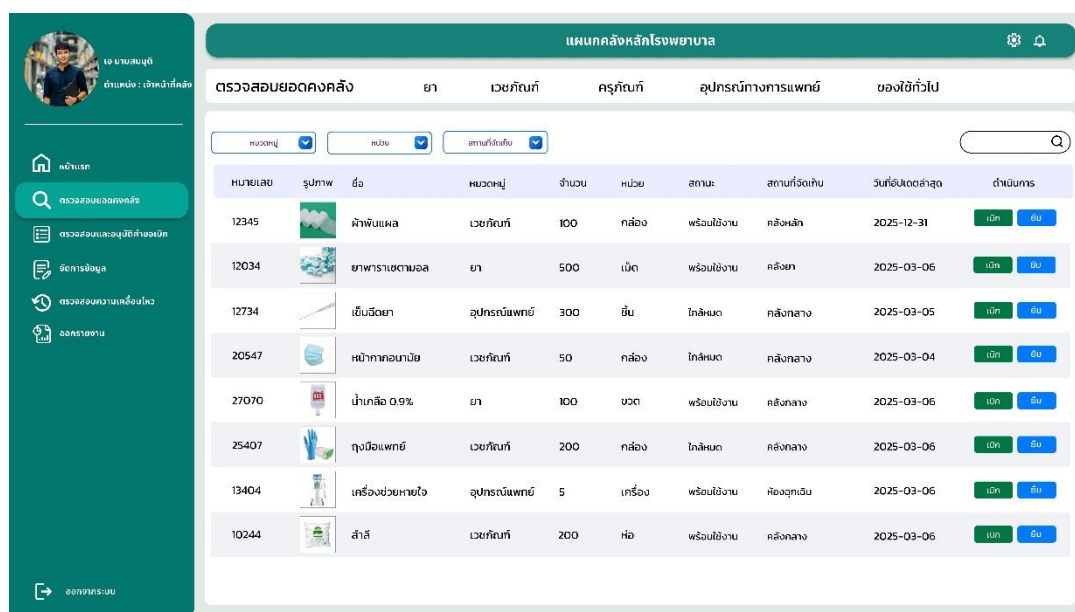
แผนกคลังหลักโรงพยาบาล						
ประวัติการนำเข้า-นำออก						
หมวดหมู่	หน่วย	สถานะดำเนินการ				
หมายเลขรายการ	ชื่อ	จำนวน	หน่วย	หมวดหมู่	สถานะการดำเนินการ	วันที่ดำเนินการ
IMT-20240301-001	ผ้าหือ	10	กล่อง	เวชภัณฑ์	สำเร็จ	2025-03-06 09:30 น.
IMT-20240301-002	น้ำยาฆ่าเชื้อ	5	ขวด	เวชภัณฑ์	สำเร็จ	2025-03-05 14:00 น.
IMT-20240301-003	ยาฆ่าเชื้อ	20	เม็ด	ยา	ยกเลิก	2025-03-04 10:15 น.
IMT-20240301-004	แอลกอฮอล์ 70%	15	ขวด	เวชภัณฑ์	สำเร็จ	2025-03-03 11:45 น.
IMT-20240301-005	ผ้าปิดปาก	50	ชิ้น	อุปกรณ์ทางการแพทย์	ยกเลิก	2025-03-02 08:30 น.
IMT-20240301-006	หมวกกาวน์	100	ชิ้น	อุปกรณ์ทางการแพทย์	สำเร็จ	2025-03-01 16:00 น.
IMT-20240301-007	ชุดตรวจ COVID-19	30	ชุด	อุปกรณ์ทางการแพทย์	สำเร็จ	2025-02-28 12:00 น.
IMT-20240301-008	น้ำเกลือ	25	ขวด	เวชภัณฑ์	ยกเลิก	2025-02-27 13:20 น.
IMT-20240301-009	พลาสเตอร์	200	ชิ้น	อุปกรณ์ทางการแพทย์	สำเร็จ	2025-02-26 15:10 น.
IMT-20240301-010	สำลี	50	แผ่น	เวชภัณฑ์	สำเร็จ	2025-02-25 09:00 น.
IMT-20240301-011	ถุงมือยาง	150	กล่อง	อุปกรณ์ทางการแพทย์	สำเร็จ	2025-02-24 14:30 น.
IMT-20240301-012	วาล์ว	20	กล่อง	เวชภัณฑ์	สำเร็จ	2025-02-23 17:50 น.
IMT-20240301-013	สเปรย์ฆ่าเชื้อ	40	ขวด	เวชภัณฑ์	สำเร็จ	2025-02-22 10:00 น.
IMT-20240301-014	ยาฉีด	30	หลอด	เวชภัณฑ์	ยกเลิก	2025-03-06 10:00 น.

รูปที่ 3.16 หน้าตรวจสอบประวัติการนำเข้า นำออก

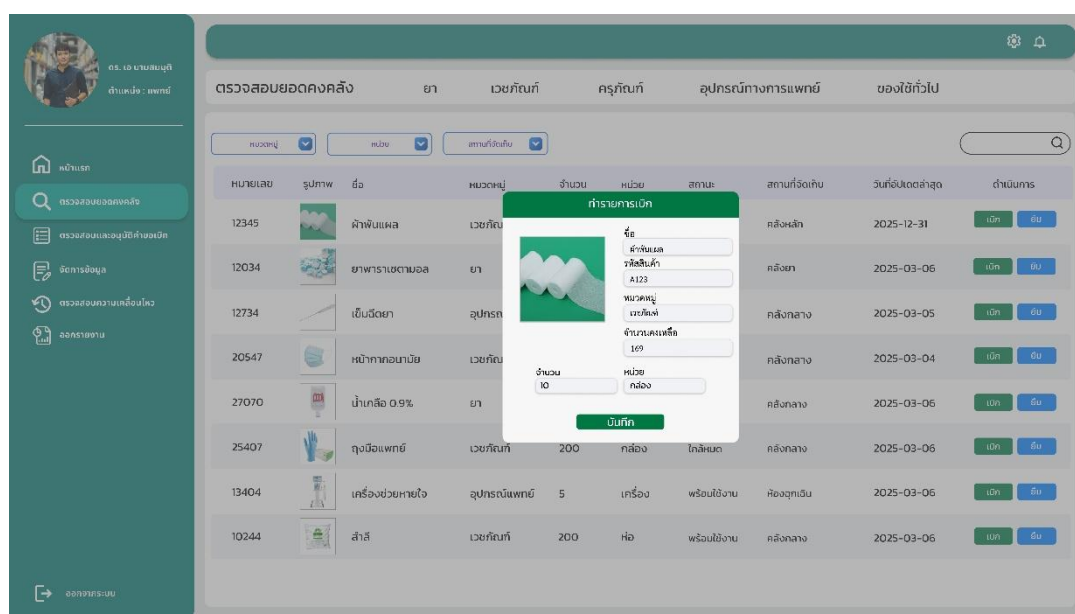
แผนกคลังหลักโรงพยาบาล		
ออกรายงาน		
ลำดับ	รายงาน	จัดการ
01	รายงานการเคลื่อนไหวของสินค้า	 
02	รายงานยอดคงคลัง	 
03	รายงานการเบิก-ยืม	 
04	รายงานสินค้าที่ใกล้หมดอายุ	 
05	รายงานสินค้าต่ำกว่าระดับที่กำหนด	 
06	รายงานการแจ้งเตือน	 
07	รายงานการใช้งานสินค้า	 
08	รายงานปริมาณงานค่าใช้จ่าย	 
09	รายงานการติดตามสถานะล่าช้า	 
10	รายงานสรุปภาพรวม	 

รูปที่ 3.17 หน้าออกรายงาน





รูปที่ 3.18 หน้าทำรายการเบิก ยืม



รูปที่ 3.19 หน้าทำรายการเบิก

The screenshot shows a web application for managing medical supplies. On the left is a sidebar with a user profile and navigation menu. The main area displays a table of supplies with columns for item number, image, name, unit, quantity, status, location, expiration date, and action buttons. A modal window titled 'ทำการแก้ไข' (Edit) is open over the first row, showing fields for name, code, unit, quantity, and status.

หมายเลข	รูปภาพ	ชื่อ	หมวดหมู่	จำนวน	หน่วย	สถานะ	สถานที่รองรับ	วันถึงหมดอายุ	ดำเนินการ
12345		ผ้าพันแผล	เวชภัณฑ์				คลังหลัก	2025-12-31	<button>แก้ไข</button> <button>ลบ</button>
12034		ยาพาราเซตามอล	ยา				คลังยา	2025-03-06	<button>แก้ไข</button> <button>ลบ</button>
12734		เข็มฉีดยา	อุปกรณ์				คลังกลาง	2025-03-05	<button>แก้ไข</button> <button>ลบ</button>
20547		หน้ากากอนามัย	เวชภัณฑ์				คลังกลาง	2025-03-04	<button>แก้ไข</button> <button>ลบ</button>
27070		น้ำเกลือ 0.9%	ยา				คลังกลาง	2025-03-06	<button>แก้ไข</button> <button>ลบ</button>
25407		ถุงมือแพทย์	เวชภัณฑ์				คลังกลาง	2025-03-06	<button>แก้ไข</button> <button>ลบ</button>
13404		เครื่องช่วยหายใจ	อุปกรณ์แพทย์	5	เครื่อง	พร้อมใช้งาน	ห้องฉุกเฉิน	2025-03-06	<button>แก้ไข</button> <button>ลบ</button>
10244		สำลี	เวชภัณฑ์	200	ห่อ	พร้อมใช้งาน	คลังกลาง	2025-03-06	<button>แก้ไข</button> <button>ลบ</button>

รูปที่ 3.20 หน้าทำการรายการยืม

The screenshot shows the 'รายการที่เลือก' (Selected Items) section of the system. It displays a table with columns for image, item number, name, quantity, unit, status, location, and action buttons. Below the table are checkboxes for 'ต้องการเร่งด่วน' (Need urgent) and 'จัดส่งตามปกติ' (Deliver normally), a date picker, and buttons for 'ยกเลิก' (Cancel), 'ฉบับร่าง' (Draft), and 'ยืนยัน' (Confirm).

รูปภาพ	หมายเลข	ชื่อ	จำนวน	หน่วย	หมวดหมู่	สถานที่รองรับ	ประเภทรายการ	
	00123	ผ้าก๊อช	20	กล่อง	เวชภัณฑ์	คลังกลาง	การเบิก	
	00123	เครื่องวัดความดัน	5	เครื่อง	อุปกรณ์ทางการแพทย์	คลังอุปกรณ์แพทย์	การคืน	
	00123	เข็มฉีดยา 5 นล.	50	กล่อง	เวชภัณฑ์	คลังกลาง	การเบิก	
	00123	เตียงผู้ป่วยไฟฟ้า	2	ชุด	ครุภัณฑ์	คลังครุภัณฑ์	การเบิก	
	00123	น้ำยาฆ่าเชื้อ	2	ลิตร	เวชภัณฑ์	คลังกลาง	การเบิก	
	00123	หน้ากากอนามัย N95	25	กล่อง	เวชภัณฑ์	คลังกลาง	การเบิก	

รูปที่ 3.21 หน้าตรวจสอบรายการที่เลือก

### 3.7 บทสรุป

จากหัวข้อทั้งหมดที่กล่าวมาข้างต้นนั้นเป็นการวิเคราะห์และออกแบบระบบบริหารจัดการโรงพยาบาล : กรณีศึกษาระบบการบริหารจัดการคลังหลักของโรงพยาบาลวัดห้วยปลากั้งเพื่อสังคม ซึ่งการวิเคราะห์ในลักษณะแผนภาพต่าง ๆ ดังกล่าวจะนำไปพัฒนาระบบในบทที่ 4 ต่อไป

## เอกสารอ้างอิง

- [1] สำนักงานสาธารณสุขจังหวัดสระแก้ว. (2566). คู่มือการปฏิบัติงานโปรแกรม IPISS ระบบคลังพัสดุ. แหล่งที่มา: <https://www.skhospital.go.th/wp-content/uploads/2023/10/13.-IPISS->. วันที่สืบค้น: 1 ธันวาคม 2567.
- [2] สำนักงานปลัดกระทรวงสาธารณสุข. (2563). คู่มือการใช้งานโปรแกรม INVS (ระบบคลังเวชภัณฑ์). แหล่งที่มา: [http://inv.dmsic.moph.go.th/inv-s/INVS\\_MANUAL.pdf](http://inv.dmsic.moph.go.th/inv-s/INVS_MANUAL.pdf). วันที่สืบค้น: 1 ธันวาคม 2567.
- [3] โรงพยาบาลศรีสะเกษ. (2566). การจัดการคลังสินค้าสำหรับ HOSXP. แหล่งที่มา: <http://www.srshos.go.th/attachments/article/369/inventory%20for%20hosxp.pdf>. วันที่สืบค้น: 1 ธันวาคม 2567.
- [4] Gupta, S., & Kant, S. (2020). Hospital Inventory Management: A Review on Current Trends and Future Prospects. *International Journal of Healthcare Management*, 13(2), 95-108.
- [5] Kwon, I. G., & Kim, S. H. (2021). Warehouse Management in Healthcare: An Overview and Future Directions. *Journal of Operations & Logistics*, 34(1), 45-63.
- [6] กระทรวงสาธารณสุข. (2566). แนวทางการบริหารจัดการเวชภัณฑ์ในโรงพยาบาล. สำนักนโยบายและยุทธศาสตร์ กระทรวงสาธารณสุข.
- [7] GS1. (2023). Introduction to Barcode and QR Code for Supply Chain Management. แหล่งที่มา: <https://www.gs1.org/>. วันที่สืบค้น: 18 พฤศจิกายน 2566.
- [8] Wang, Y., & Chang, T. (2022). Barcode-Based Inventory Control System for Medical Supplies. *Journal of Medical Informatics*, 28(4), 112-125.
- [9] ISO/IEC 15459-1:2014. Information Technology – Automatic Identification and Data Capture Techniques – Unique Identification for Supply Chain Management.
- [10] Fielding, R. T. (2000). Architectural Styles and the Design of Network-based Software Architectures. Doctoral dissertation, University of California, Irvine.
- [11] PostgreSQL Global Development Group. (2023). PostgreSQL Documentation: The World's Most Advanced Open Source Relational Database. แหล่งที่มา: <https://www.postgresql.org/docs/>. วันที่สืบค้น: 18 พฤศจิกายน 2566.
- [12] LogRocket. (2023). CRUD REST API with Node.js, Express, and PostgreSQL. แหล่งที่มา: <https://blog.logrocket.com/crud-rest-api-node-js-express-postgresql/>. วันที่สืบค้น: 1 ธันวาคม 2566

- [13] Nielsen, J. (2012). Usability Engineering. Elsevier.
- [14] Cooper, A., Reimann, R., Cronin, D., & Noessel, C. (2014). About Face: The Essentials of Interaction Design. John Wiley & Sons.
- [15] Google Material Design. (2023). Design Guidelines for UI/UX. แหล่งที่มา: <https://material.io/design>. วันที่สืบค้น: 18 พฤศจิกายน 2566.
- [16] ไชศรี เต็มเปี่ยม. (2567). รูปแบบการบริหารจัดการการเงินการคลังของโรงพยาบาล จังหวัดสระแก้ว. สำนักงานสาธารณสุขจังหวัดสระแก้ว.
- [17] สมชาย ใจดี. (2566). การพัฒนารูปแบบการบริหารจัดการการเงินการคลังของโรงพยาบาล จังหวัดนครราชสีมา. วารสารการบริหารสาธารณสุข, 14(2), 45-60.
- [18] QR Tiger. (2020). วิธีการใช้ QR Code สำหรับระบบบริหารสินค้าคงคลัง. แหล่งที่มา: <https://www.qrcode-tiger.com/th/how-to-use-qr-codes-for-inventory-management-system>. วันที่สืบค้น: 18 พฤศจิกายน 2566.
- [19] Packhai. (2023). การจัดการคลังสินค้าด้วยระบบบาร์โค้ด คืออะไร? มีประโยชน์อย่างไรบ้าง. แหล่งที่มา: <https://packhai.com/barcode-system-for-warehouse-management/>. วันที่สืบค้น: 18 พฤศจิกายน 2566.
- [20] LabelOne. (2024). การนำระบบบาร์โค้ดมาใช้ในคลังสินค้า. แหล่งที่มา: <https://labelone.co.th/blog/การนำระบบบาร์โค้ดมาใช้ในคลังสินค้า>. วันที่สืบค้น: 18 พฤศจิกายน 2566.
- [21] OfficeAceShop. (2024). บาร์โค้ด (Barcode) มีส่วนสำคัญอย่างไร ในการจัดการสินค้าคงคลัง? แหล่งที่มา: <https://officeaceshop.com/why-is-barcode-important-to-inventory-management/>. วันที่สืบค้น: 18 พฤศจิกายน 2566.
- [22] IDPRT. (2024). วิธีใช้ ระบบบาร์โค้ด คลังสินค้า? แหล่งที่มา: <https://th.idprt.com/barcode-printers/How-to-Implement-Warehouse-Barcode-System.html>. วันที่สืบค้น: 18 พฤศจิกายน 2566.
- [23] สมชาย ใจดี. (2566). การพัฒนาระบบการจัดการพัสดุคงคลัง โรงพยาบาลฝาง. วารสารการพยาบาลและการดูแลสุขภาพ, 12(1), 75-88.
- [24] มณีรัตน์ สุขสวัสดิ์. (2565). S-A-M-NG-A-M: ระบบการบริหารจัดการคลังเวชภัณฑ์และระบบบัญชีที่กำกับดูแลด้วยเทคโนโลยีเครือข่าย โรงพยาบาลสามง่าม. วารสารการพัฒนาสุขภาพชุมชน, 10(2), 123-135.
- [25] ภก.รัชตะ อุลมาน. (2562). งานประชุมหัวหน้ากลุ่มงานเภสัชกรรมโรงพยาบาล ครั้งที่ 1/2562. สำนักปลัดกระทรวงสาธารณสุข ร่วมกับชมรมเภสัชกรโรงพยาบาล
- [26] โรงพยาบาลเชียงรายประชานุเคราะห์. (2566). คู่มือการปฏิบัติงานโปรแกรม IPISS ระบบคลังพัสดุ.

- [27] สำนักงานปลัดกระทรวงสาธารณสุข. (2563). คู่มือการใช้งานโปรแกรม INVS (ระบบคลังเวชภัณฑ์).