

วิศวกรรมซอฟต์แวร์ (Software Engineering)

อ. พิชัย จอดพิมาย

แผนการสอน

- ❑ สัปดาห์ที่ 1 : ความรู้เบื้องต้นเกี่ยวกับวิศวกรรมซอฟต์แวร์
- ❑ สัปดาห์ที่ 2 : กระบวนการพัฒนาซอฟต์แวร์
- ❑ สัปดาห์ที่ 3-4 : การบริหารโครงการซอฟต์แวร์
- ❑ สัปดาห์ที่ 5 : วิศวกรรมความต้องการและการวิเคราะห์ระบบ
- ❑ สัปดาห์ที่ 6 : แบบจำลองระบบ
- ❑ สัปดาห์ที่ 7 : การออกแบบสถาปัตยกรรมระบบ
- ❑ สัปดาห์ที่ 8 : สรุปและทบทวนเนื้อหาก่อนสอบกลางภาค
- ❑ สัปดาห์ที่ 9 : **สอบกลางภาค**

แผนการสอน (ต่อ)

- ❑ สัปดาห์ที่ 10 : การออกแบบอย่างละเอียด
- ❑ สัปดาห์ที่ 11 : การออกแบบส่วนต่อประสานผู้ใช้
- ❑ สัปดาห์ที่ 12-13 : การพัฒนาซอฟต์แวร์
- ❑ สัปดาห์ที่ 14 : การทดสอบซอฟต์แวร์
- ❑ สัปดาห์ที่ 15 : นำเสนอผลงานการออกแบบระบบ
- ❑ สัปดาห์ที่ 16 : สรุปและทบทวนเนื้อหาก่อนสอบปลายภาค
- ❑ สัปดาห์ที่ 17 : **สอบปลายภาค**

เนื้อหาแบบจำลองระบบ (System Model)

- 1 พื้นฐานของแบบจำลองระบบ
- 2 แบบจำลองตามแนวทางเชิงโครงสร้าง
- 3 แบบจำลองตามแนวทางเชิงวัตถุ
- 4 แบบฝึกหัด (Exercise)

พื้นฐานของแบบจำลอง (System Model Foundation)

- ❑ เป็นสัญลักษณ์ที่ใช้จำลองข้อเท็จจริงต่าง ๆ ที่จะเกิดขึ้นในระบบ
- ❑ มีหลายชนิดเพื่อให้สอดคล้องกับมุมมองที่หลากหลายของผู้ใช้งาน
- ❑ เป็นเครื่องมือในการสื่อสารเพื่อให้เข้าใจตรงกัน
- ❑ นำไปใช้ทำ Project Plan (ส่วนของการประเมิน Size/Effort/Cost) ในขั้นการวางแผน
- ❑ นำไปใช้ทำเอกสารข้อกำหนดความต้องการระบบ SRS ในขั้นการวิเคราะห์ความต้องการ
- ❑ นำไปใช้ทำเอกสารออกแบบระบบ SDD ในขั้นตอนการออกแบบระบบ

วิธีการสร้างแบบจำลองระบบ

**เชิงโครงสร้าง
(Structured
Model)**

- DFD
- ERD

**เชิงวัตถุ
(Object)**

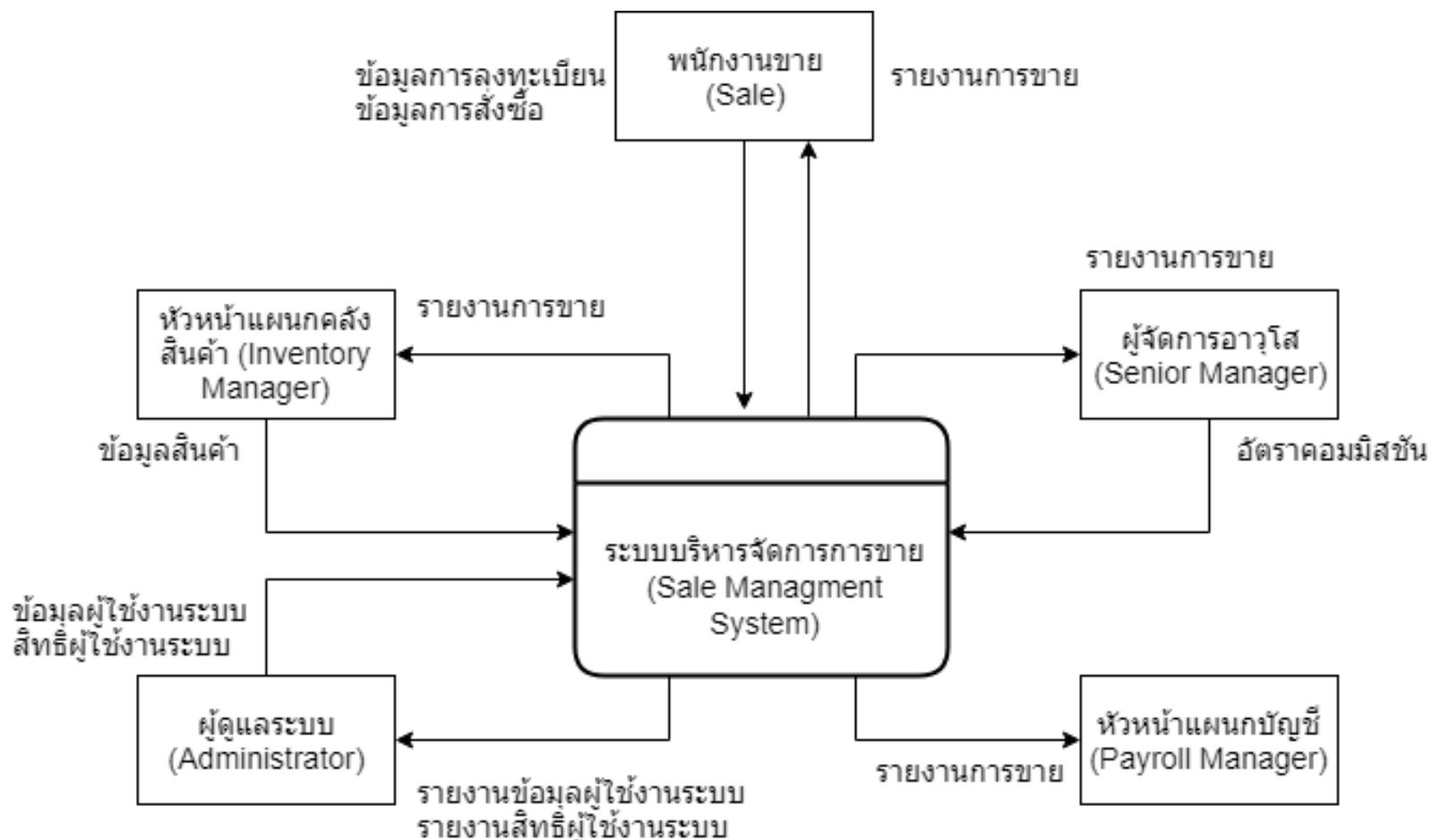
- Use-Case Diagram
- Activity Diagram
- Class Diagram
- Sequence Diagram
- Deploy Diagram

แบบจำลองตามแนวทางเชิงโครงสร้าง

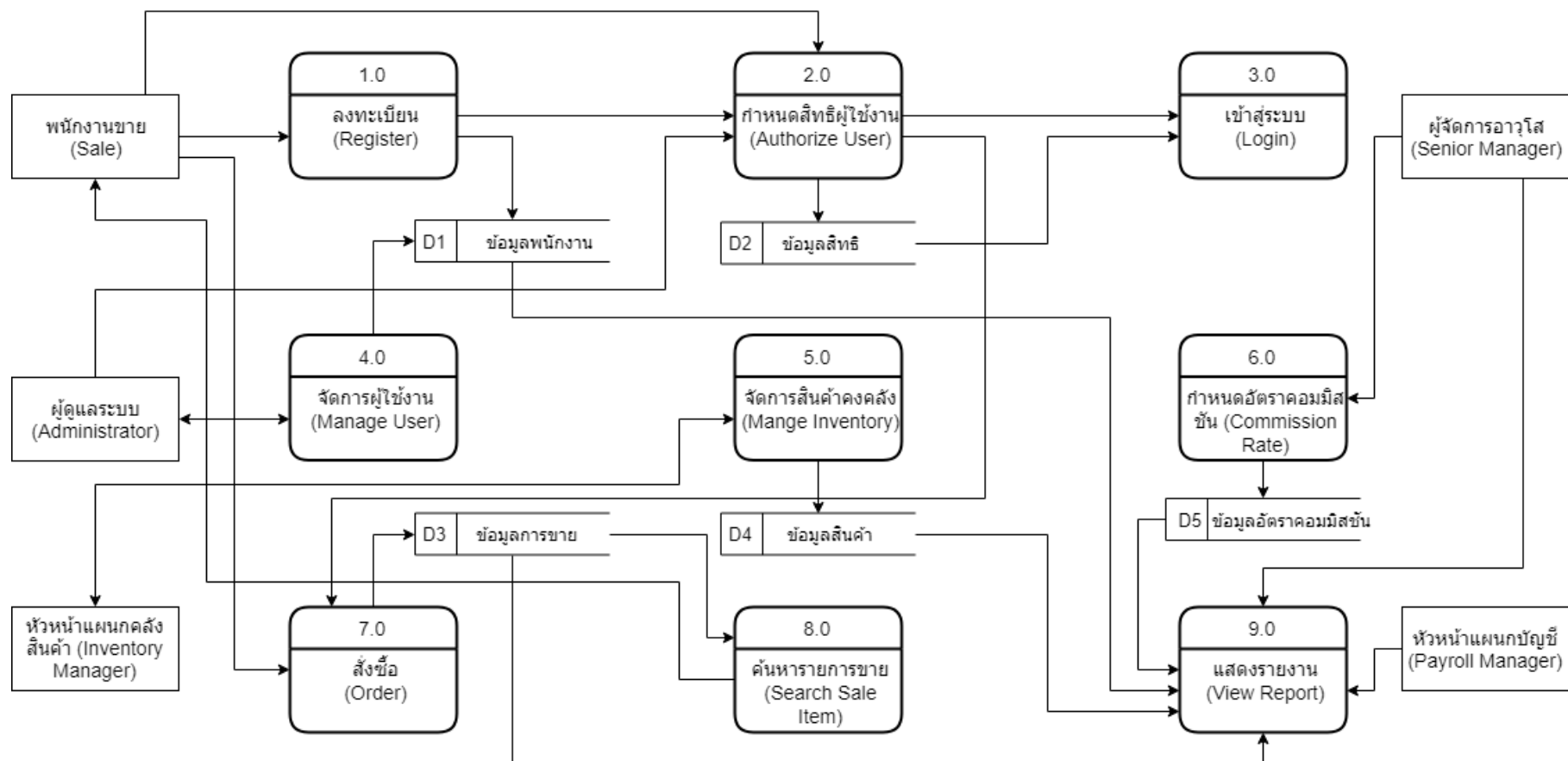
- ❑ ใช้สำหรับการวิเคราะห์และออกแบบระบบเชิงโครงสร้าง (Structured System Analysis and Design)
- ❑ เป็นแบบจำลองดั้งเดิม
- ❑ เน้น Process/Data
- ❑ ตัวอย่างแบบจำลอง
 - ❑ Process Model (Context Diagram and DFD)
 - ❑ Data Model (ERD)
- ❑ มักใช้กับการพัฒนาด้วยภาษาโครงสร้าง (Structure) เช่น
 - ❑ C
 - ❑ Pascal
 - ❑ Basic
 - ❑ COBOL

แผนภาพกระแสข้อมูล (Data Flow Diagram: DFD)

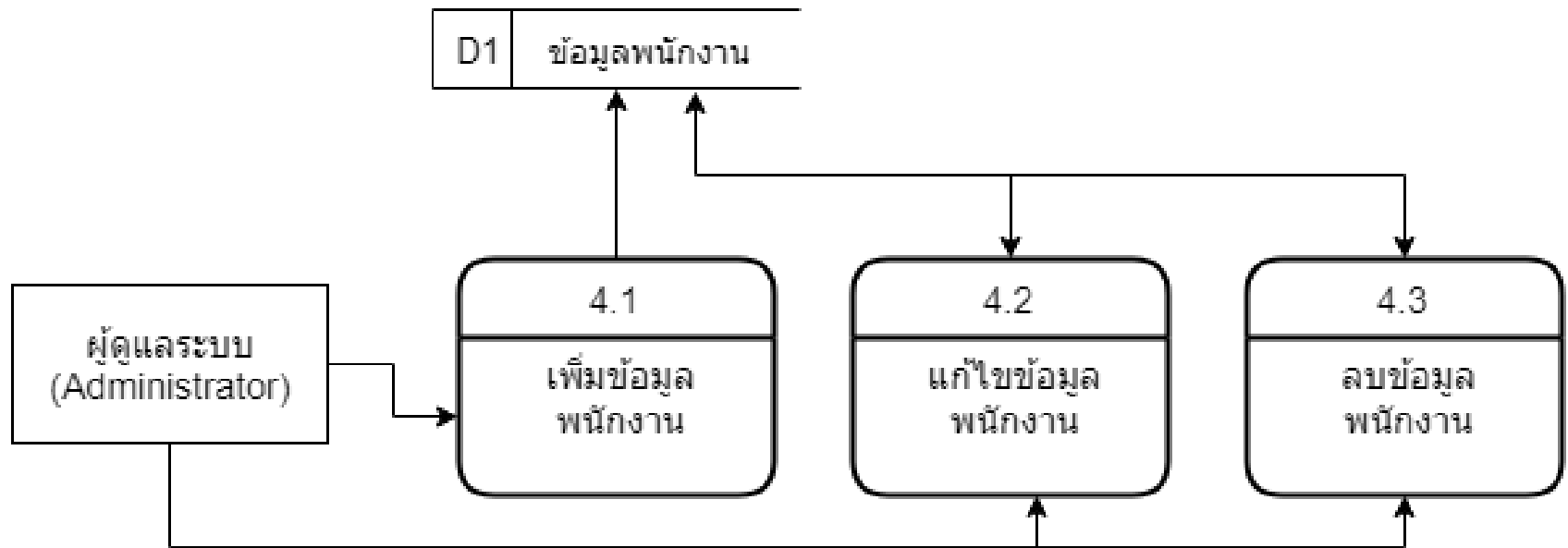
- ❑ แสดงทิศทางการไหลของข้อมูลในระบบจากกระบวนการหนึ่งไปยังกระบวนการหนึ่ง จากกระบวนการไปยังแหล่งจัดเก็บข้อมูล หรือจากกระบวนการไปยังผู้ที่เกี่ยวข้อง
- ❑ เนื่องจาก DFD ให้ทั้ง Process และ Data Store จึงสามารถนำไปใช้สำหรับการคำนวณ Function Point เพื่อหาขนาดซอฟต์แวร์ ค่าความพยายามและการประมาณต้นทุนในขั้นตอนการวางแผนโครงการ
- ❑ ในขั้นตอนการวางแผนและการวิเคราะห์อาจใช้เพียง Context Diagram และ DFD ระดับ 0 โดยที่ระดับที่สูงกว่ามักเพิ่มเข้าไปในขั้นตอนการออกแบบ



รูปแสดง ตัวอย่าง Context Diagram ของระบบบริหารจัดการการขาย



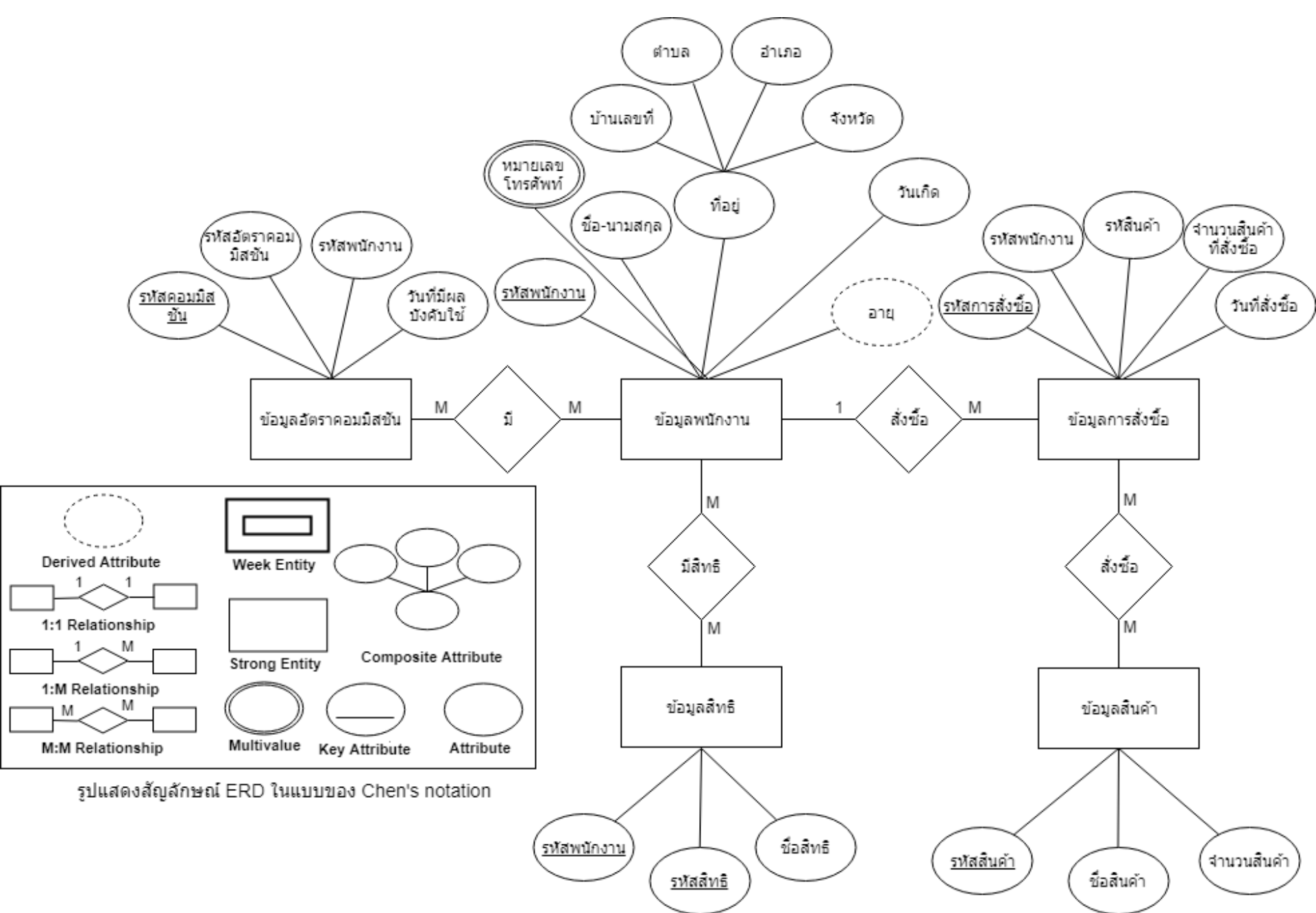
รูปแสดง ตัวอย่าง DFD ระดับ 0 ของระบบบริหารจัดการการขาย



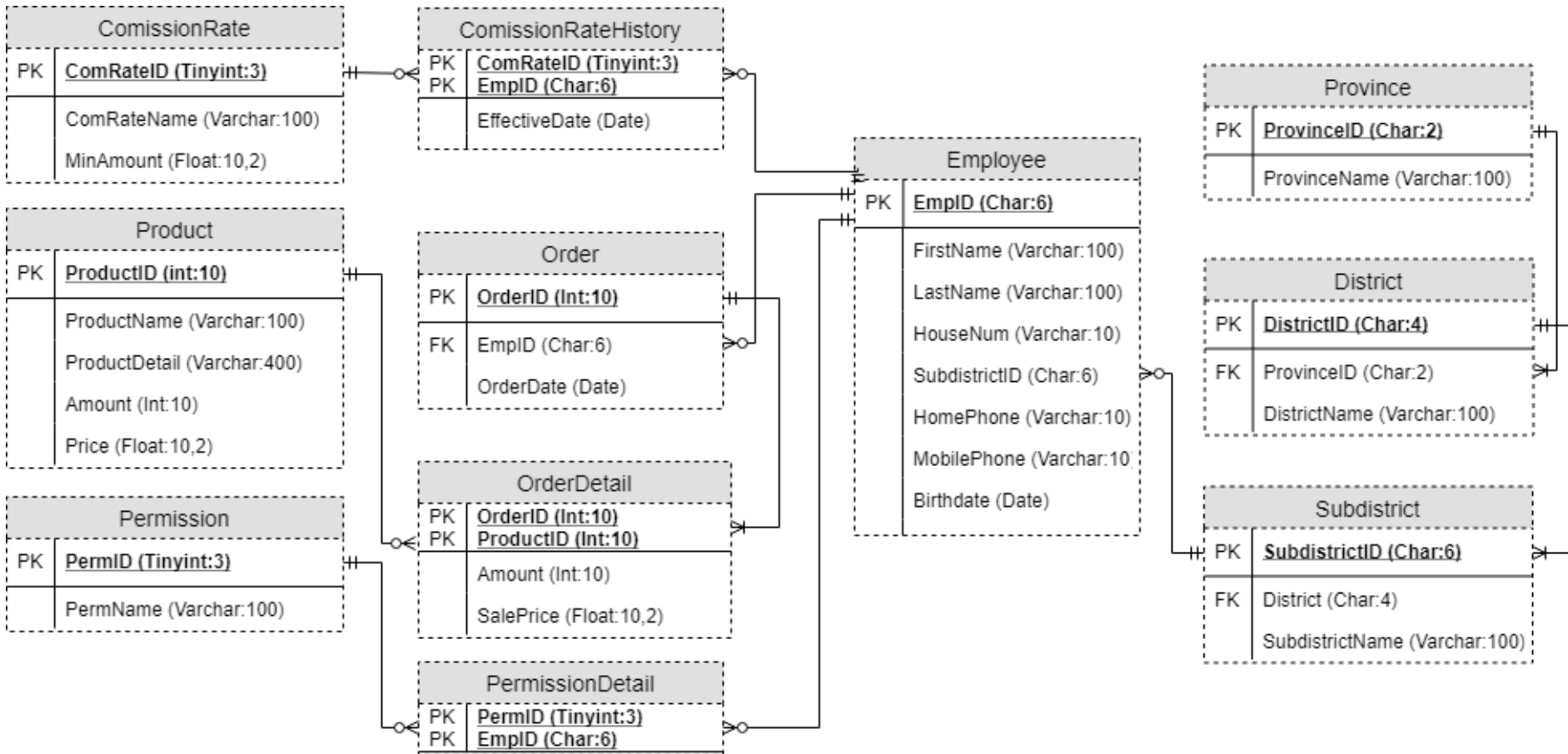
รูปแสดง ตัวอย่าง DFD ระดับ 1 ของกระบวนการ 8. จัดการพนักงาน ในระบบบริหารจัดการการขาย

Entity Relation Diagram (ERD)

- ❑ แสดงความสัมพันธ์ระหว่าง Entity หรือกลุ่มข้อมูล
- ❑ แต่ละ Entity จะประกอบไปด้วย Attribute หรือฟิลด์ข้อมูล
- ❑ สามารถนำไปคำนวณ Function Point ในส่วนของ ILF และ EIF เพื่อหาขนาดซอฟต์แวร์ ค่าความพยายามและการประมาณต้นทุนในขั้นตอนการวางแผนโครงการ
- ❑ ในขั้นตอนการวางแผนและการวิเคราะห์อาจใช้เพียง ERD ระดับ Logical โดยที่ระดับที่สูงกว่ามักเพิ่มเข้าไปในขั้นตอนการออกแบบ



รูปแสดง ตัวอย่าง ERD ระดับ Logical ของระบบบริหารจัดการการขาย



════
 1:1 Relationship
 จำเป็นต้องมี : จำเป็นต้องมี

════<══
 1:M Relationship
 จำเป็นต้องมี : จำเป็นต้องมี

>═══<══
 M:M Relationship
 จำเป็นต้องมี : จำเป็นต้องมี

════○
 1:1 Relationship
 จำเป็นต้องมี : ไม่จำเป็นต้องมี

════○<══
 1:M Relationship
 จำเป็นต้องมี : ไม่จำเป็นต้องมี

>═══○<══
 M:M Relationship
 จำเป็นต้องมี : ไม่จำเป็นต้องมี

Data Name	
Key	<u>attribute (type)</u>
	attribute (type)

รูปแสดงสัญลักษณ์ ERD ในแบบของ Crow's foot notation

แบบจำลองตามแนวทางเชิงวัตถุ

- ❑ ใช้สำหรับการวิเคราะห์และออกแบบระบบเชิงวัตถุ (Object Oriented System Analysis and Design)
- ❑ เน้น Object
- ❑ มักใช้กับการพัฒนาด้วยภาษา Object Oriented Programming (OOP) เช่น
 - ❑ C++
 - ❑ C#
 - ❑ Java

❑ Unified Modeling Language (UML) เป็นภาษามาตรฐานในการสร้างแบบจำลอง ดูแลโดย Object Management Group (OMG) ซึ่งแบ่ง UML แบ่งออกเป็น 2 กลุ่ม

❑ ไดอะแกรมแบบโครงสร้าง (Structure Diagram)

❑ Class Diagram

❑ Object Diagram

❑ Component Diagram

❑ Deployment Diagram

❑ ไดอะแกรมแบบพฤติกรรม (Behavioral Diagram)

❑ Use Case Diagram

❑ Activity Diagram

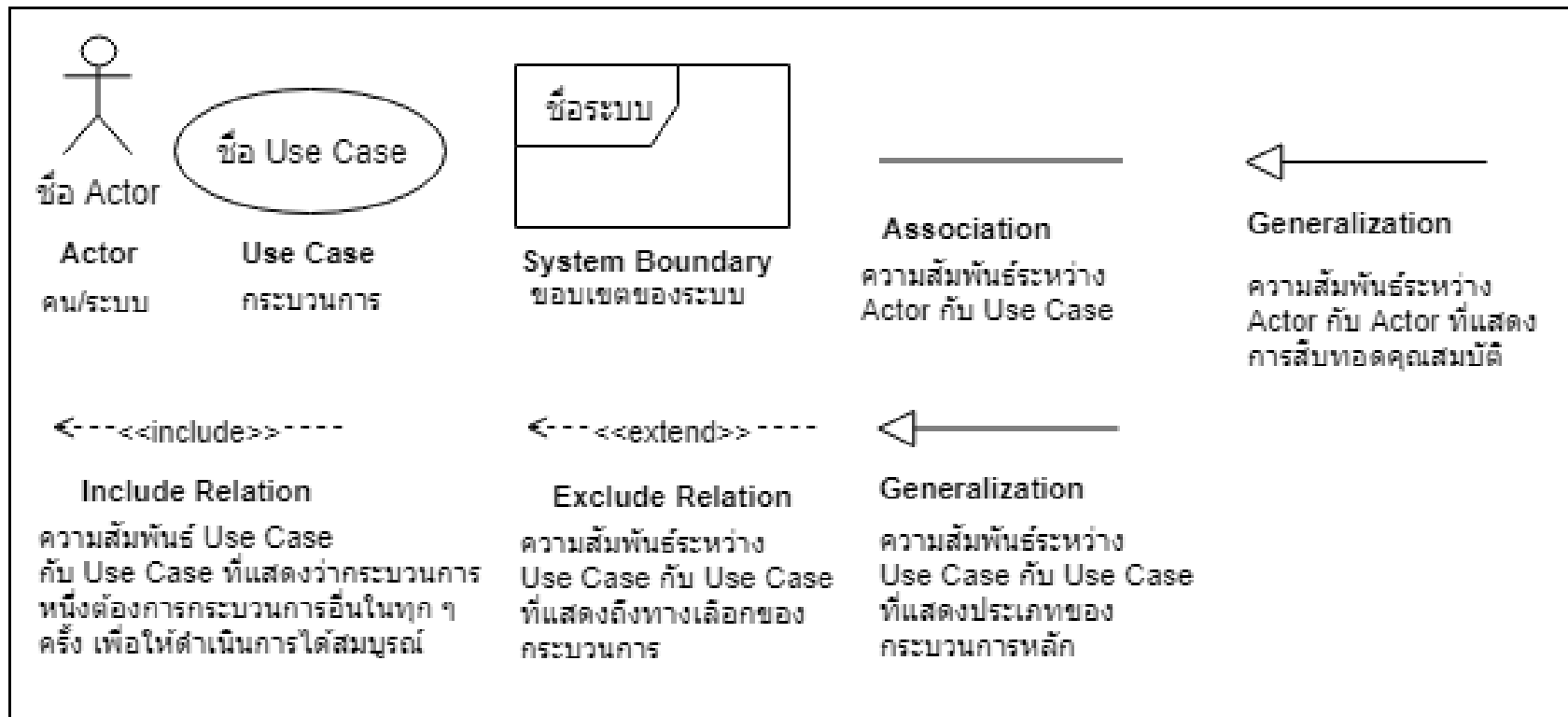
❑ Sequence Diagram

❑ Collaboration Diagram

❑ State Diagram

Use Case Diagram

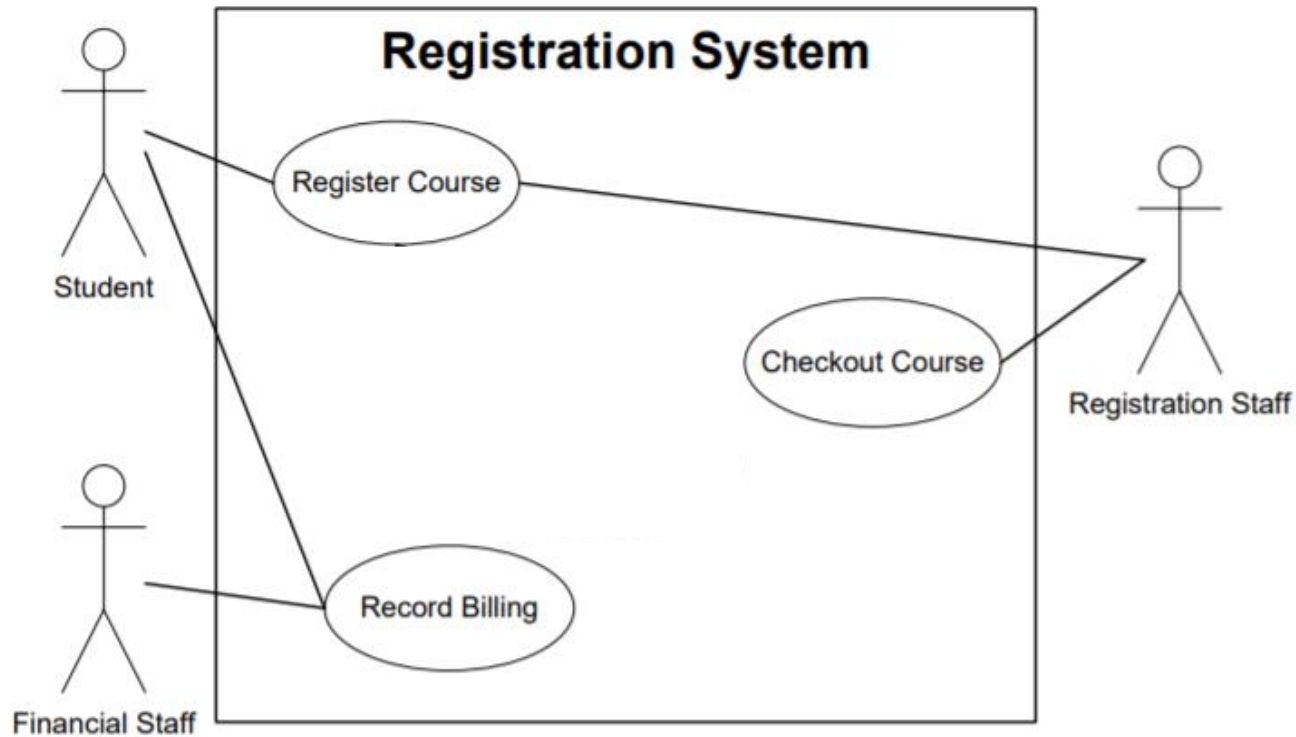
- ❑ แสดงความสัมพันธ์ระหว่าง Actor (ผู้ใช้งานระบบหรือระบบอื่นที่มีการแลกเปลี่ยนข้อมูล) กับ Use Case (กระบวนการหรือฟังก์ชันการทำงานของระบบ)
- ❑ สามารถนำไปใช้ในการคำนวณ Use Case Point เพื่อหาขนาดซอฟต์แวร์ ค่าความพยายามและการประมาณต้นทุนในขั้นตอนการวางแผนโครงการ
- ❑ องค์ประกอบของ Use Case Diagram
 - ❑ Actor แทนคนหรือระบบ
 - ❑ Use Case แทนกระบวนการ/กิจกรรม
 - ❑ System Boundary แทนขอบเขตของระบบ
 - ❑ ความสัมพันธ์



รูปแสดงสัญลักษณ์ของ Use Case Diagram

❑ ประเภทความสัมพันธ์

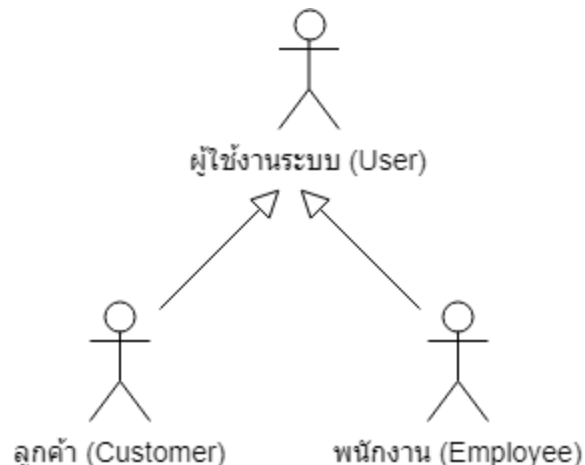
- ❑ ความเกี่ยวข้องกัน (Association) เป็นความสัมพันธ์ในระดับเดียวกัน ไม่มีการระบุการสืบทอดกัน การเป็นส่วนหนึ่งส่วนใดของกัน หรือมีความสำคัญที่มากกว่ากัน



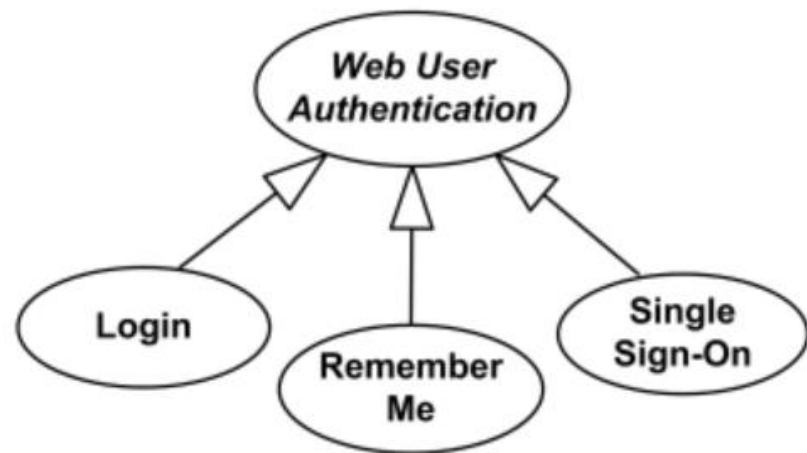
รูปแสดง ตัวอย่างความสัมพันธ์แบบ Association ระหว่าง Actor กับ Use Case

❑ การสืบทอดคุณสมบัติ (Generalization/Inheritance) แบ่งออกเป็น

- ❑ ความสัมพันธ์แบบสืบทอดคุณสมบัติ ที่ Actor ที่สืบทอดมีคุณสมบัติมากกว่า Actor หลัก
- ❑ ความสัมพันธ์แบบบอกระเภท ที่ Use Case หนึ่งประกอบไปด้วยอีกหลาย Use Case

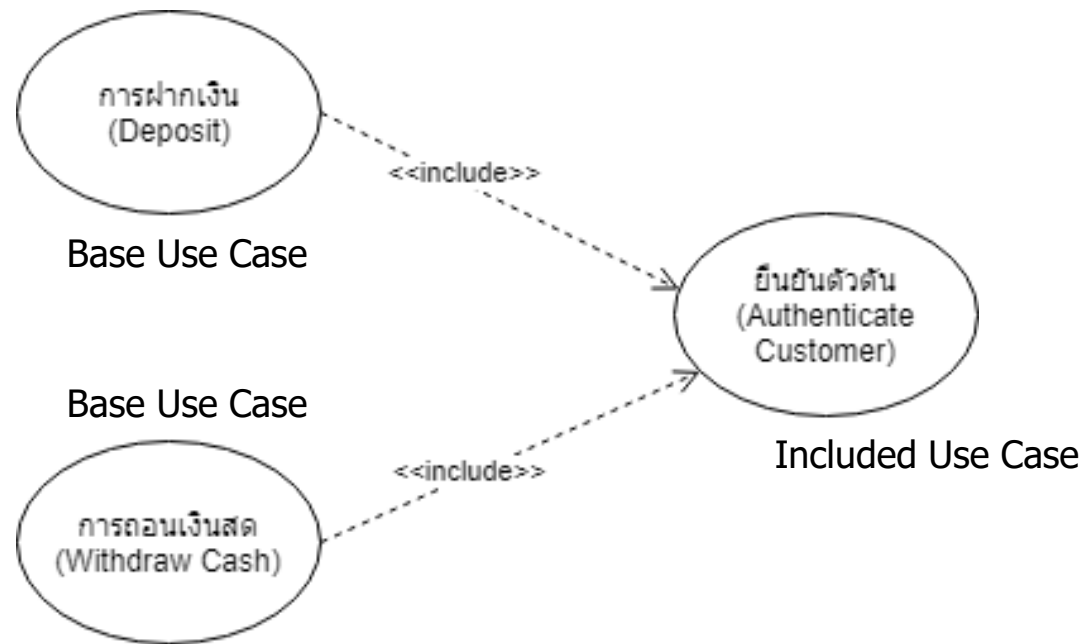


รูปแสดง ตัวอย่างความสัมพันธ์แบบ Generalization ระหว่าง Actor กับ Actor



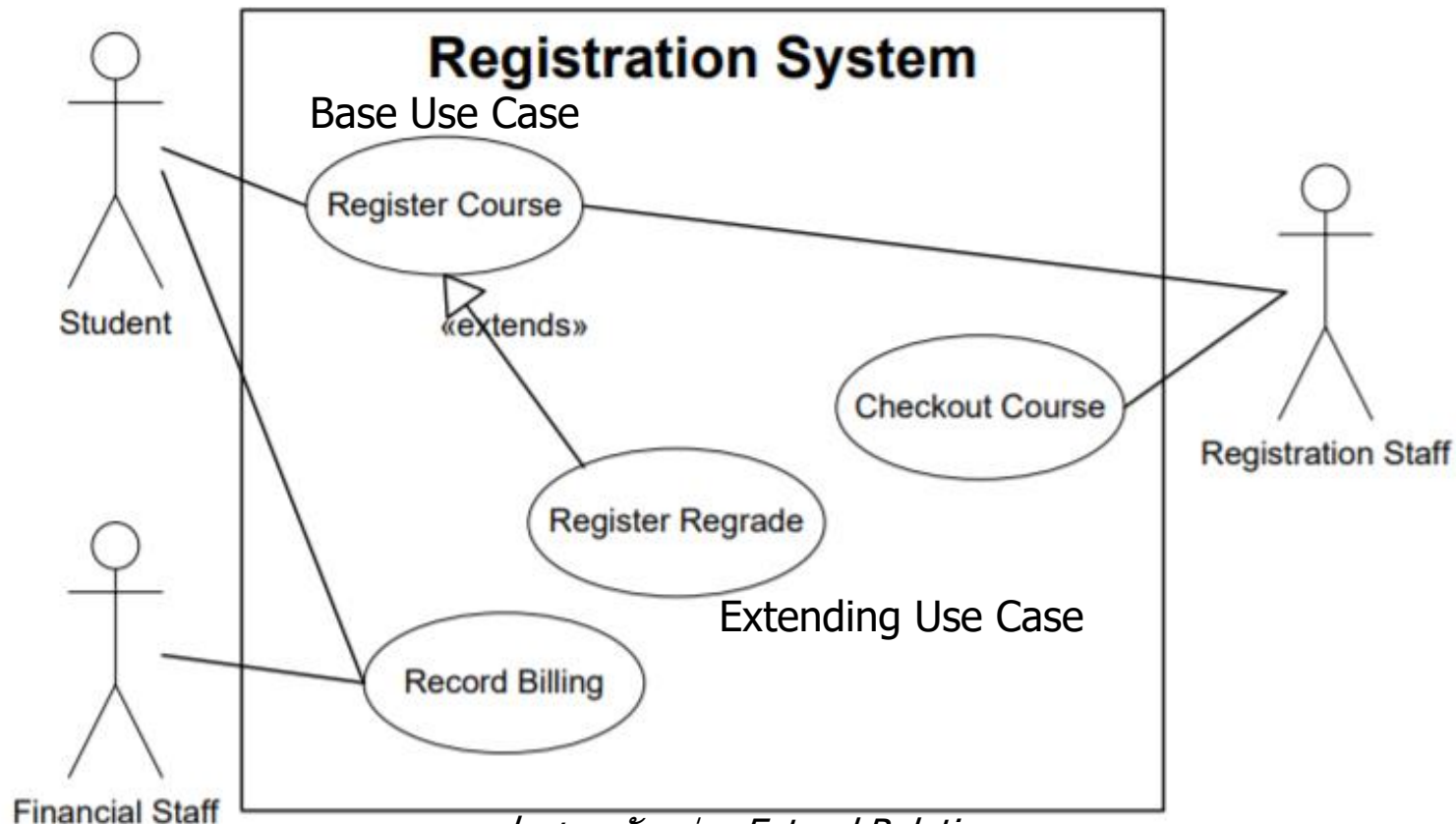
รูปแสดง ตัวอย่างความสัมพันธ์แบบ Generalization ระหว่าง Use Case กับ Use Case

- ❑ การรวมเข้า (Include Relation) แทนความสัมพันธ์ Use Case กับ Use Case ที่แสดงว่ากระบวนการหนึ่งต้องการกระบวนการอื่นในทุก ๆ ครั้ง เพื่อให้ดำเนินการได้สมบูรณ์

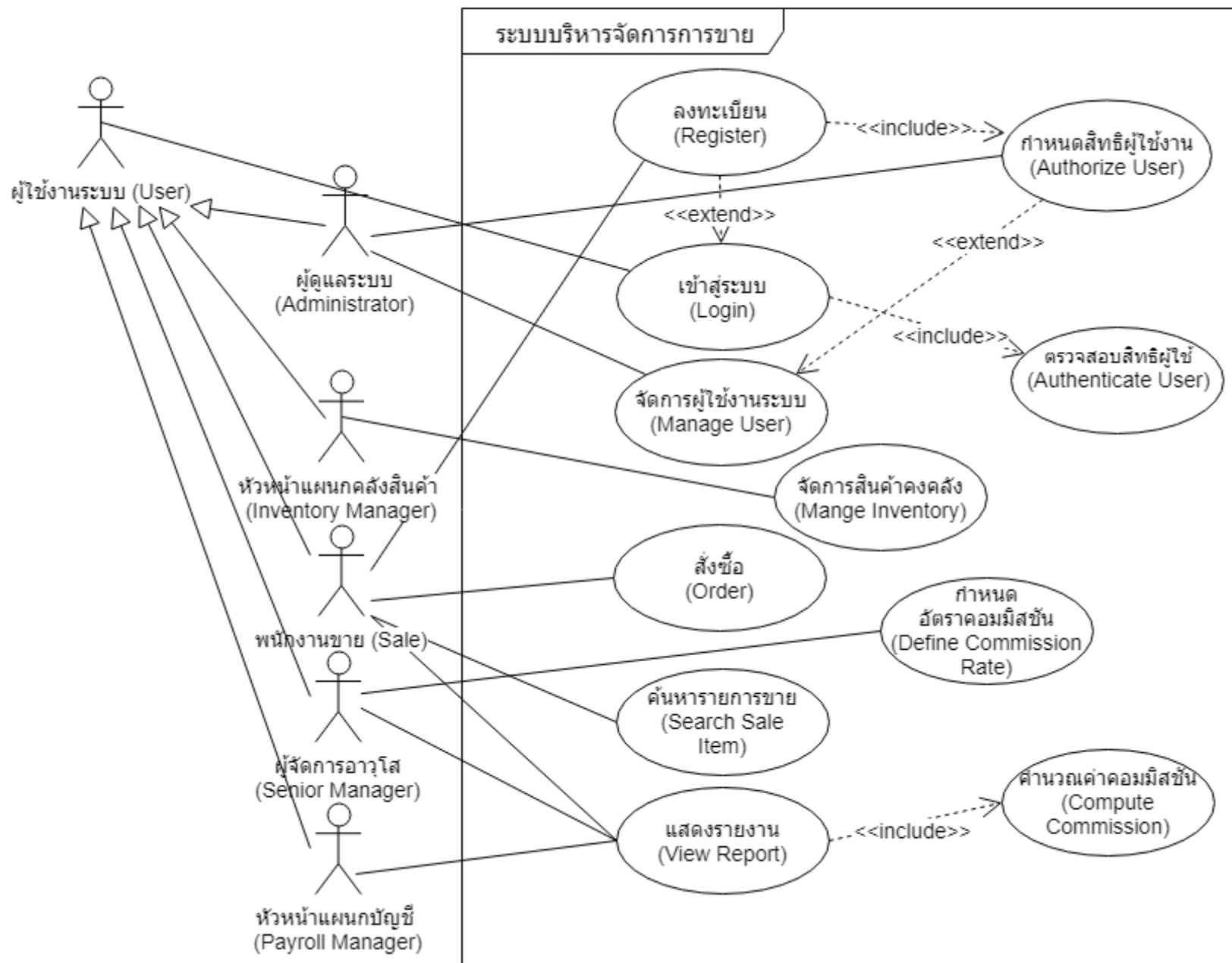


รูปแสดง ตัวอย่างของ Include Relation

- ❑ **การขยาย (Extend Relation)** แทนความสัมพันธ์ระหว่าง Use Case กับ Use Case ที่แสดงว่ากระบวนการหนึ่งต้องการกระบวนการอื่นในบางครั้ง เพื่อให้ดำเนินการได้สมบูรณ์



รูปแสดง ตัวอย่าง Extend Relation



รูปแสดง ตัวอย่าง Use Case ของระบบบริหารจัดการการขาย

- ❑ คำอธิบาย **Use Case** (Use Case Description) เป็นการอธิบายรายละเอียดของ Use Case มีองค์ประกอบดังนี้
 - ❑ Use Case Name
 - ❑ Description
 - ❑ Pre-Condition
 - ❑ Post-Condition
 - ❑ Flows of Events
 - ❑ Basic Flows of Events
 - ❑ Alternative Flows of Events

Use Case Name:	ลงทะเบียน (Register)	
Description:	ลงทะเบียนสำหรับใช้งานระบบ	
Actors:	พนักงานขาย (Sale)	
Pre-Conditions:	พนักงานขายยังไม่เป็นสมาชิกของระบบ	
Post-Conditions:	พนักงานขายพบหน้าเข้าสู่ระบบ (Login)	
Basic Flows of Events:	Actor	System
	1. ระบุชื่อผู้ใช้ รหัสผ่าน และข้อมูลพนักงาน	2. ตรวจสอบข้อมูลที่จำเป็นและรูปแบบข้อมูล 3. บันทึกข้อมูลเข้าสู่ระบบ 4. แจ้งการลงทะเบียนของพนักงานให้ผู้ดูแลระบบทราบ
Alternative Flows of Events:		2a แจ้งรายการข้อมูลที่ไม่ได้ระบุหรือความผิดพลาดของรูปแบบข้อมูล เช่น รูปแบบอีเมล

Use Case Name:	เข้าสู่ระบบ (Login)	
Description:	เข้าสู่ระบบบริหารจัดการการขาย	
Actors:	ผู้ใช้งานระบบ (User)	
Pre-Conditions:	Sale ยังไม่ทำการ Login	
Post-Conditions:	Sale พบเมนูในส่วนของตนเอง	
Basic Flows of Events:	Actor	System
	1. ระบุชื่อผู้ใช้และรหัสผ่าน	2. ตรวจสอบข้อมูลที่จำเป็นและสิทธิการเข้าถึง 3. ให้สิทธิในการเข้าถึงระบบ 4. Redirect ไปยังหน้าแรกของระบบ
Alternative Flows of Events:		2a แจ้งรายการข้อมูลที่ไม่ได้ระบุหรือสิทธิการเข้าถึงยังไม่ได้รับ

รูปแสดง ตัวอย่าง Use Case Description ของระบบบริหารจัดการการขาย

Use Case Name:	กำหนดสิทธิผู้ใช้งาน (Authorize User)	
Description:	กำหนดสิทธิให้กับผู้ใช้งานระบบ ทั้งพนักงานขาย ผู้จัดการอาวุโส หัวหน้าแผนกคลังสินค้า และ หัวหน้าแผนกบัญชี	
Actors:	ผู้ดูแลระบบ (Administrator)	
Pre-Conditions:	ผู้ดูแลระบบเข้าสู่ระบบและได้รับสิทธิในการกำหนดสิทธิผู้ใช้งาน	
Post-Conditions:	ผู้ใช้งานระบบสามารถเข้าใช้งานระบบ	
Basic Flows of Events:	Actor	System
	1. เลือกผู้ใช้ที่ต้องการอนุญาตเข้าใช้งานระบบ 3. ให้สิทธิเข้าใช้งานระบบ	2. แสดงรายละเอียดผู้ใช้งานระบบ 4. ปรับสถานะสิทธิการเข้าใช้งานระบบ 5. แจ้งผลการลงทะเบียนให้พนักงานทราบ
Alternative Flows of Events:	3a ไม่ให้สิทธิเข้าใช้งานระบบ	4a ลบข้อมูลผู้ร้องขอลงทะเบียนนออกจากระบบ

รูปแสดง ตัวอย่าง Use Case Description ของระบบบริหารจัดการการขาย (ต่อ)

Use Case Name:	จัดการผู้ใช้งานระบบ (Manage User)	
Description:	จัดการผู้ใช้งานระบบเพิ่ม แก้ไข และลบ ผู้ใช้งานระบบ ทั้งพนักงานขาย ผู้จัดการอาวุโส หัวหน้าแผนก คลังสินค้า และหัวหน้าแผนกบัญชี	
Actors:	ผู้ดูแลระบบ (Administrator)	
Pre-Conditions:	1. ผู้ดูแลระบบเข้าสู่ระบบและได้รับสิทธิในการจัดการพนักงานขาย 2. ผู้ดูแลระบบพบหน้าจอแสดงรายการผู้ใช้งานระบบ	
Post-Conditions:	-	
Basic Flows of Events:	Actor	System
	1. เพิ่มข้อมูลผู้ใช้งานระบบ 1.1. คลิกปุ่มเพิ่มข้อมูลผู้ใช้งาน 1.3. ระบุรายละเอียดผู้ใช้งาน	1.2. แสดงหน้าจอการเพิ่มข้อมูลผู้ใช้งาน 1.4. ตรวจสอบข้อมูลที่จำเป็น รูปแบบข้อมูล และข้อมูลผู้ใช้งานซ้ำ 1.5. บันทึกข้อมูลเข้าสู่ระบบ 1.6. แจ้งผลการบันทึก
Alternative Flows of: Events	2. แก้ไขข้อมูลผู้ใช้งานระบบ	
	2.1. เลือกผู้ใช้งานที่ต้องการแก้ไข 2.3. แก้ไขรายละเอียดผู้ใช้งาน	2.2. แสดงรายละเอียดผู้ใช้งาน 2.4. ตรวจสอบข้อมูลที่จำเป็นและรูปแบบข้อมูล 2.5. บันทึกข้อมูลเข้าสู่ระบบ 2.6. แจ้งผลการแก้ไข
	3. ลบข้อมูลผู้ใช้งานระบบ	
	3.1. เลือกผู้ใช้งานที่ต้องการลบ 3.3. คลิกปุ่มลบผู้ใช้งาน 3.5. ทำการยืนยันการลบ	3.2. แสดงรายละเอียดผู้ใช้งาน 3.4. แจ้งเตือนให้ทำการยืนยันการลบ 3.6. ลบข้อมูลออกจากระบบ
	4. แจ้งเตือน	
		1.4a แจ้งเตือนความผิดพลาดจากการตรวจสอบข้อมูลที่เพิ่ม 2.4a แจ้งเตือนความผิดพลาดจากการตรวจสอบข้อมูลที่แก้ไข

รูปแสดง ตัวอย่าง Use Case Description ของระบบบริหารจัดการการขาย (ต่อ)

Use Case Name:	จัดการสินค้าคงคลัง (Manage Inventory)	
Description:	จัดการสินค้าคงคลังเพิ่ม แก้ไข และลบสินค้า	
Actors:	หัวหน้าแผนกคลังสินค้า (Inventory Manager)	
Pre-Conditions:	1. ผู้จัดการคลังสินค้าเข้าสู่ระบบและได้รับสิทธิในการจัดการสินค้าคงคลัง 2. ผู้จัดการคลังสินค้าพบหน้าจอแสดงรายการสินค้า	
Post-Conditions:	-	
Basic Flows of Events:	Actor	System
	1. เพิ่มข้อมูลสินค้า 1.1. คลิกปุ่มเพิ่มข้อมูลสินค้า 1.3. ระบุรายละเอียดสินค้า	1.2. แสดงหน้าจอการเพิ่มข้อมูลสินค้า 1.4. ตรวจสอบข้อมูลที่จำเป็น รูปแบบข้อมูล และข้อมูลผู้ใช้งานซ้ำ 1.5. บันทึกข้อมูลเข้าสู่ระบบ 1.6. แจ้งผลการบันทึก
Alternative Flows of: Events	2. แก้ไขข้อมูลสินค้า	
	2.1. เลือกผู้ใช้งานที่ต้องการแก้ไข 2.3. แก้ไขรายละเอียดสินค้า	2.2. แสดงรายละเอียดสินค้า 2.4. ตรวจสอบข้อมูลที่จำเป็นและรูปแบบข้อมูล 2.5. บันทึกข้อมูลเข้าสู่ระบบ 2.6. แจ้งผลการแก้ไข
	3. ลบข้อมูลผู้ใช้งานระบบ	
	3.1. เลือกสินค้าที่ต้องการลบ 3.3. คลิกปุ่มลบสินค้า 3.5. ทำการยืนยันการลบ	3.2. แสดงรายละเอียดสินค้า 3.4. แจ้งเตือนให้ทำการยืนยันการลบ 3.6. ลบข้อมูลออกจากระบบ
	4. แจ้งเตือน	
		1.4a แจ้งเตือนความผิดพลาดจากการตรวจสอบข้อมูลที่เพิ่ม 2.4a แจ้งเตือนความผิดพลาดจากการตรวจสอบข้อมูลที่แก้ไข

รูปแสดง ตัวอย่าง Use Case Description ของระบบบริหารจัดการการขาย (ต่อ)

Use Case Name:	กำหนดอัตราคอมมิสชัน (Define Commission Rate)	
Description:	กำหนดอัตราคอมมิสชัน	
Actors:	ผู้จัดการอาวุโส (Senior Manager)	
Pre-Conditions:	1. ผู้จัดการอาวุโสเข้าสู่ระบบและได้รับสิทธิในการกำหนดอัตราคอมมิสชัน 2. ผู้จัดการอาวุโสพบหน้าจอแสดงรายการอัตราคอมมิสชัน	
Post-Conditions:	-	
Basic Flows of Events:	Actor	System
	1. เพิ่มข้อมูลอัตราคอมมิสชัน 1.1. คลิกปุ่มเพิ่มข้อมูลอัตราคอมมิสชัน 1.3. ระบุรายละเอียดอัตราคอมมิสชัน	1.2. แสดงหน้าจอการเพิ่มข้อมูลอัตราคอมมิสชัน 1.4. ตรวจสอบข้อมูลที่จำเป็น รูปแบบข้อมูล และข้อมูลผู้ใช้งานซ้ำ 1.5. บันทึกข้อมูลเข้าสู่ระบบ 1.6. แจ้งผลการบันทึก
Alternative Flows of Events	2. แก้ไขข้อมูลอัตราคอมมิสชัน	
	2.1. เลือกผู้ใช้งานที่ต้องการแก้ไข 2.3. แก้ไขรายละเอียดอัตราคอมมิสชัน	2.2. แสดงรายละเอียดอัตราคอมมิสชัน 2.4. ตรวจสอบข้อมูลที่จำเป็นและรูปแบบข้อมูล 2.5. บันทึกข้อมูลเข้าสู่ระบบ 2.6. แจ้งผลการแก้ไข
	3. ลบข้อมูลผู้ใช้งานระบบ	
	3.1. เลือกอัตราคอมมิสชันที่ต้องการลบ 3.3. คลิกปุ่มลบอัตราคอมมิสชัน 3.5. ทำการยืนยันการลบ	3.2. แสดงรายละเอียดอัตราคอมมิสชัน 3.4. แจ้งเตือนให้ทำการยืนยันการลบ 3.6. ลบข้อมูลออกจากระบบ
	4. แจ้งเตือน	
		1.4a แจ้งเตือนความผิดพลาดจากการตรวจสอบข้อมูลที่เพิ่ม 2.4a แจ้งเตือนความผิดพลาดจากการตรวจสอบข้อมูลที่แก้ไข

รูปแสดง ตัวอย่าง Use Case Description ของระบบบริหารจัดการการขาย (ต่อ)

Use Case Name:	สั่งซื้อ (Order)	
Description:	พนักงานขายทำรายการสั่งซื้อสินค้า (ดำเนินการแทนลูกค้า)	
Actors:	พนักงานขาย (Sale)	
Pre-Conditions:	1. พนักงานขายเข้าสู่ระบบ 2. มีรายการสินค้าในระบบ	
Post-Conditions:	-	
Basic Flows of Events:	Actor	System
	1. เลือกสินค้า 4. ระบุรายละเอียดการสั่งซื้อ 6. สั่งพิมพ์ใบสั่งซื้อ	2. ตรวจสอบสินค้าคงเหลือ 3. แสดงหน้าสรุปรายการสินค้า 5. บันทึกใบสั่งซื้อ 7. พิมพ์ใบสั่งซื้อ
Alternative Flows of Events:	4a แก้ไขจำนวนสินค้า 4b เลือกสินค้าเพิ่มเติม	3a แจ้งสินค้าคงเหลือไม่พอ

Use Case Name:	ค้นหารายการขาย (Search Sale Item)	
Description:	ค้นหาและเรียกดูรายละเอียดการขาย	
Actors:	พนักงานขาย (Sale)	
Pre-Conditions:	1. พนักงานขายเข้าสู่ระบบ 2. มีรายการขายในระบบ	
Post-Conditions:	-	
Basic Flows of Events:	Actor	System
	1. ระบุคำค้น 4. เลือกรายการขายที่ต้องการ	2. ทำการค้นหา 3. แสดงรายการขายที่เกี่ยวข้อง 5. แสดงรายละเอียดการขาย
Alternative Flows:		2a แจ้งไม่พบรายการที่ค้นหา

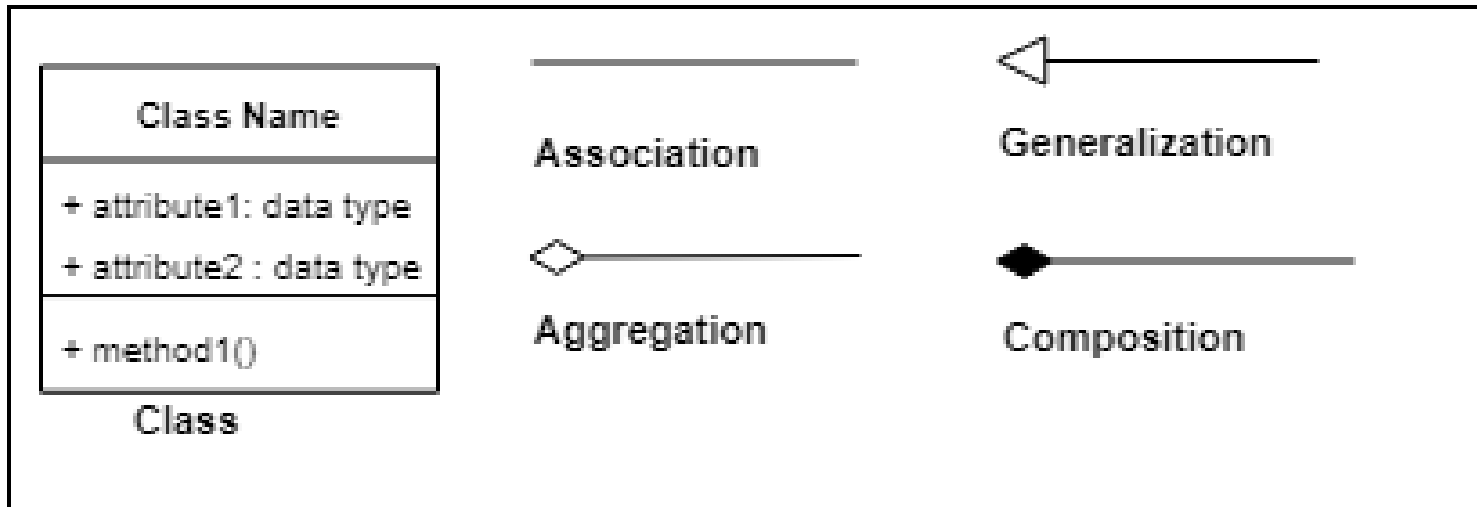
รูปแสดง ตัวอย่าง Use Case Description ของระบบบริหารจัดการการขาย (ต่อ)

Use Case Name:	สร้างรายงาน (Generate Report)	
Description:	แสดงรายงานต่าง ๆ	
Actors:	พนักงานขาย (Sale) และ ผู้จัดการอาวุโส (Senior Manager)	
Pre-Conditions:	1. พนักงานขายหรือผู้จัดการอาวุโสเข้าสู่ระบบ 2. มีรายการขายในระบบ	
Post-Conditions:	-	
Basic Flows of Events:	Actor	System
	1. เลือกรายงานการขาย 4. สั่งพิมพ์รายงานการขาย	2. สร้างรายงานการขาย 3. แสดงรายงานการขาย 5. พิมพ์รายงานการขาย
Alternative Flows of Events:		2a แจ้งเตือนความผิดพลาดของการสร้างรายงาน

รูปแสดง ตัวอย่าง Use Case Description ของระบบบริหารจัดการการขาย (ต่อ)

Class Diagram

เป็นแผนภาพที่ใช้สำหรับแสดงกลุ่มของ Class โครงสร้างของ Class และ Interface และแสดงความสัมพันธ์ระหว่าง Classes



รูปแสดงสัญลักษณ์ของ Class Diagram

- ❑ การระบุจำนวนสมาชิกในความสัมพันธ์ระหว่าง Class (Multiplicity/Cardinality)
 - ❑ 0..1 คือ ไม่จำเป็นต้องมี ถ้ามีก็มีได้เพียง 1 แฉว
 - ❑ 1..1 คือ จำเป็นต้องมี ถ้ามีก็มีได้เพียง 1 แฉว
 - ❑ 1 คือ จำเป็นต้องมี ถ้ามีก็มีได้เพียง 1 แฉว
 - ❑ 0..* คือ ไม่จำเป็นต้องมี ถ้ามีก็มีได้หลาย แฉว
 - ❑ 0 คือ ไม่จำเป็นต้องมี ถ้ามีก็มีได้หลาย แฉว
 - ❑ 1..* คือ จำเป็นต้องมี ถ้ามีก็มีได้หลาย แฉว

☐ องค์ประกอบของ Class

- ☐ Class แสดงความเป็นวัตถุ (Object) ของระบบ เช่น คน สิ่งของ สถานที่

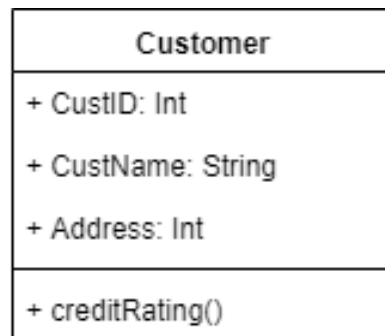
- ☐ Attribute แสดงคุณสมบัติของวัตถุ เช่น น้ำหนัก ส่วนสูง สีผิว

- ☐ Method แสดงกิจกรรมของวัตถุ เช่น วิ่ง นั่ง เดิน

☐ การตั้งชื่อ Class

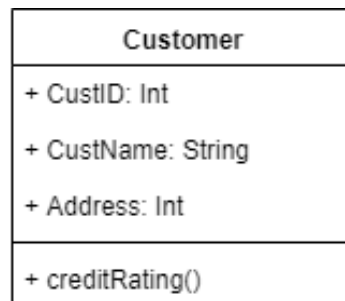
- ☐ ชื่อ Class ควรจะเป็นคำนาม (Noun) และขึ้นต้นด้วยตัวใหญ่ เช่น **Employee**

- ☐ ชื่อ Method ควรจะเป็นคำกริยา (Verb) และขึ้นต้นด้วยตัวเล็ก เช่น **addEmployee** และ **deleteEmployee**



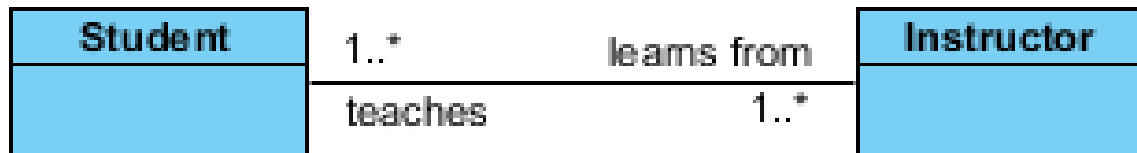
รูปแสดง ตัวอย่างของ Class

- ❑ การมองเห็น (Visibility) ของ Attribute และ Method
 - ❑ **Public (+)** เป็นการอนุญาตให้ Class อื่นสามารถมองเห็น Attribute หรือ Method ภายใน Class ได้
 - ❑ **Public (-)** เป็นการป้องกันไม่ให้ Class อื่นสามารถมองเห็น Attribute หรือ Method ภายใน Class ได้
 - ❑ **Protect (#)** เป็นการอนุญาตให้เฉพาะ Class ที่ Inherit ไปสามารถมองเห็น Attribute หรือ Method ภายใน Class ได้
- ❑ ประเภทข้อมูล (Data Type) ของ Attribute
 - ❑ **Integer** สำหรับ Attribute ที่เป็นตัวเลข
 - ❑ **String** สำหรับ Attribute ที่เป็นตัวหนังสือ
 - ❑ **Char** สำหรับ Attribute ที่เป็นตัวอักษร
 - ❑ **Boolean** สำหรับ Attribute ที่มีค่าความจริง เป็นจริงหรือเท็จ

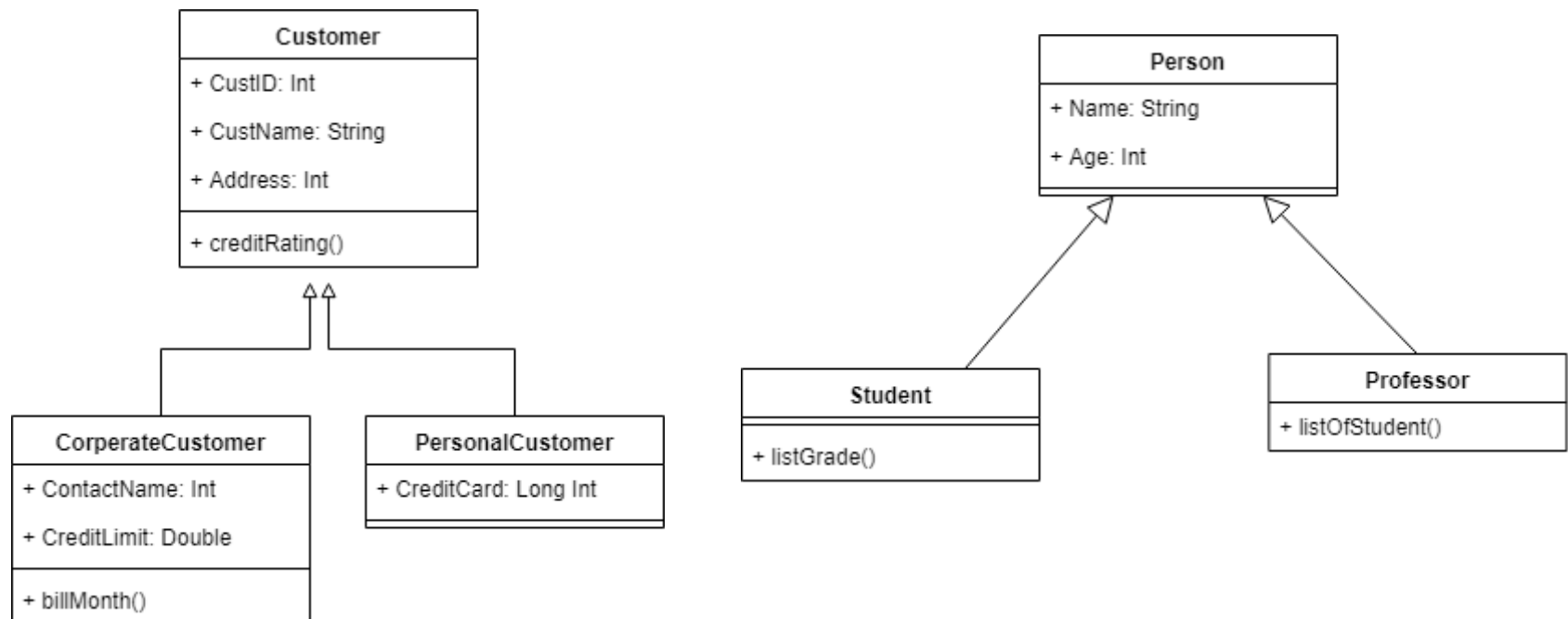


รูปแสดง ตัวอย่างของ Class

- ❑ ประเภทความสัมพันธ์ระหว่าง Class
 - ❑ ความเกี่ยวข้องกัน (Association) เป็นความสัมพันธ์ระหว่าง Class ในระดับเดียวกัน ไม่มีการระบุการสืบทอดกัน การเป็นส่วนหนึ่งของกัน หรือมีความสำคัญที่มากกว่ากัน

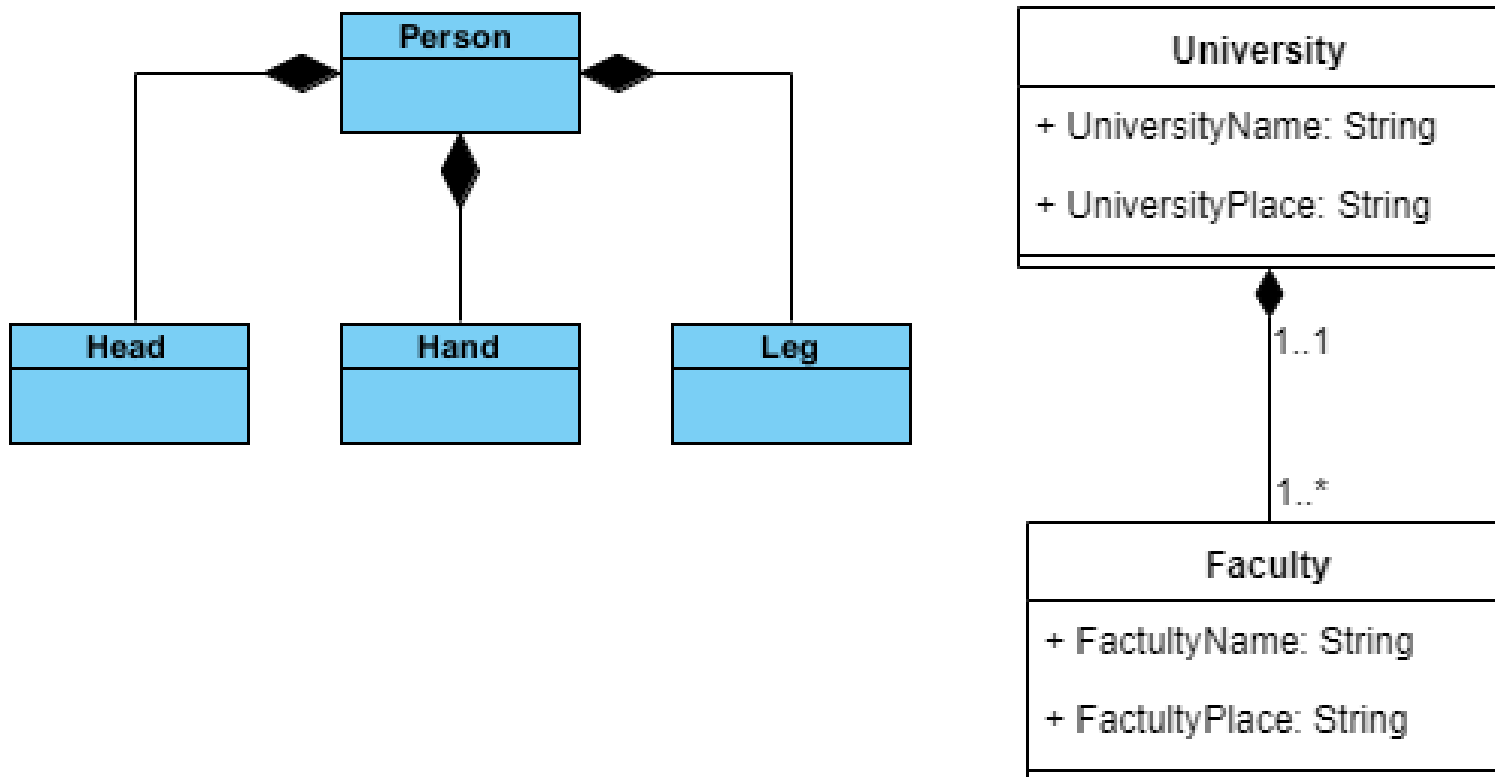


❑ การสืบทอดคุณสมบัติ (Generalization/Inheritance) เป็นความสัมพันธ์ระหว่าง Class แบบการสืบทอด (Inheritance) นั่นคือ Class ที่สืบทอด (Sub Class) มาต้องมีคุณสมบัติที่มากกว่า Class แม่ (Super Class) เช่น Class พนักงานประจำ Class พนักงาน Part-time และ Class ที่ปรึกษา สืบทอดจาก Class Employee



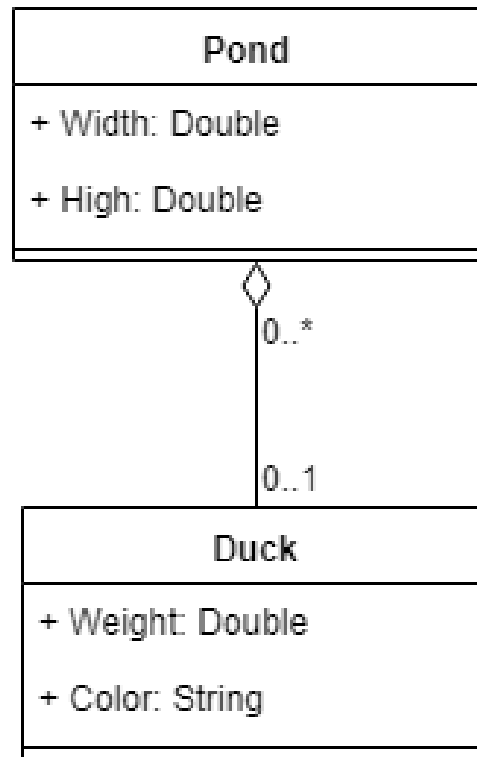
รูปแสดง ตัวอย่างความสัมพันธ์แบบ Generalization/Inheritance

- ❑ **การเป็นองค์ประกอบของ (Composition)** เป็นความสัมพันธ์ระหว่าง Class ที่ว่า Class หนึ่งเป็นองค์ประกอบของอีก Class หนึ่ง เมื่อ Class หลักถูกทำลาย Class ที่เป็นองค์ประกอบจะถูกทำลายไปด้วย หรือเมื่อ Class ที่เป็นองค์ประกอบถูกทำลายก็จะทำให้ Class หลักสูญเสียนคุณสมบัติไป

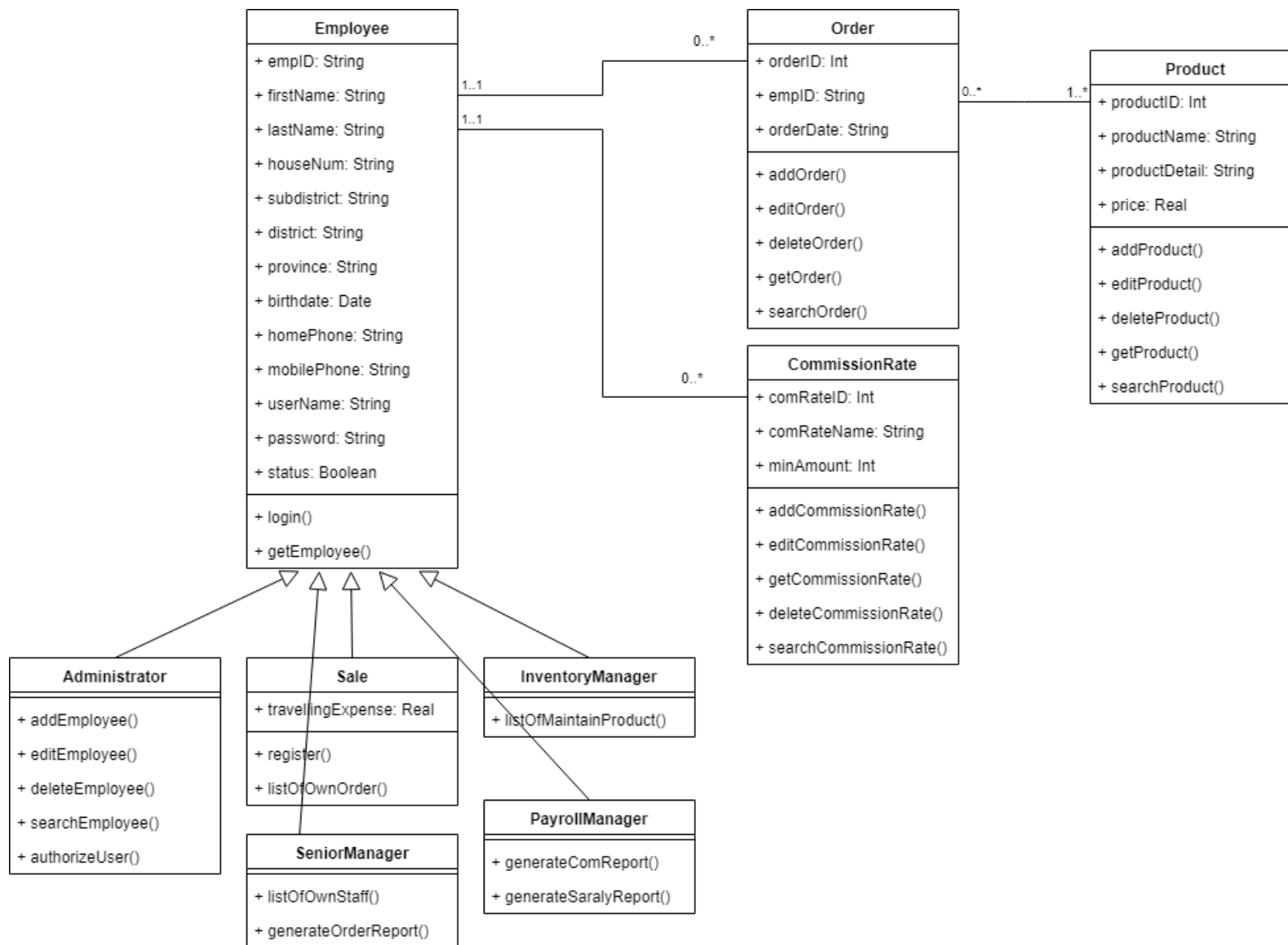


รูปแสดง ตัวอย่างความสัมพันธ์แบบ Composition

- ❑ **การเป็นส่วนหนึ่งของ (Aggregation)** เป็นความสัมพันธ์ระหว่าง Class ที่ว่า Class หนึ่งเป็นส่วนหนึ่งของอีก Class หนึ่ง เมื่อ Class หลักถูกทำลาย Class ที่เป็นส่วนหนึ่งจะไม่ถูกทำลายไปด้วย หรือเมื่อ Class ที่เป็นส่วนหนึ่งถูกทำลายก็จะไม่ทำให้ Class หลักสูญเสียคุณสมบัติไป



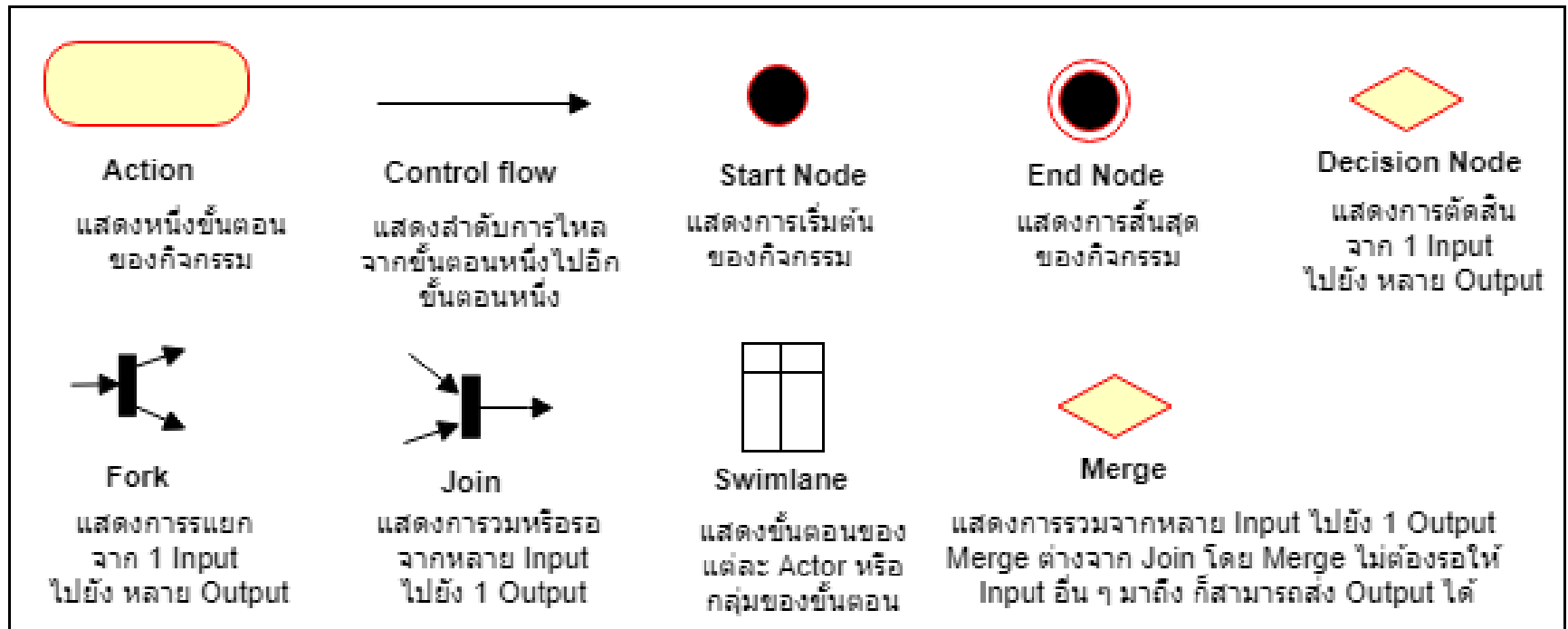
รูปแสดง ตัวอย่างความสัมพันธ์แบบ Aggregation



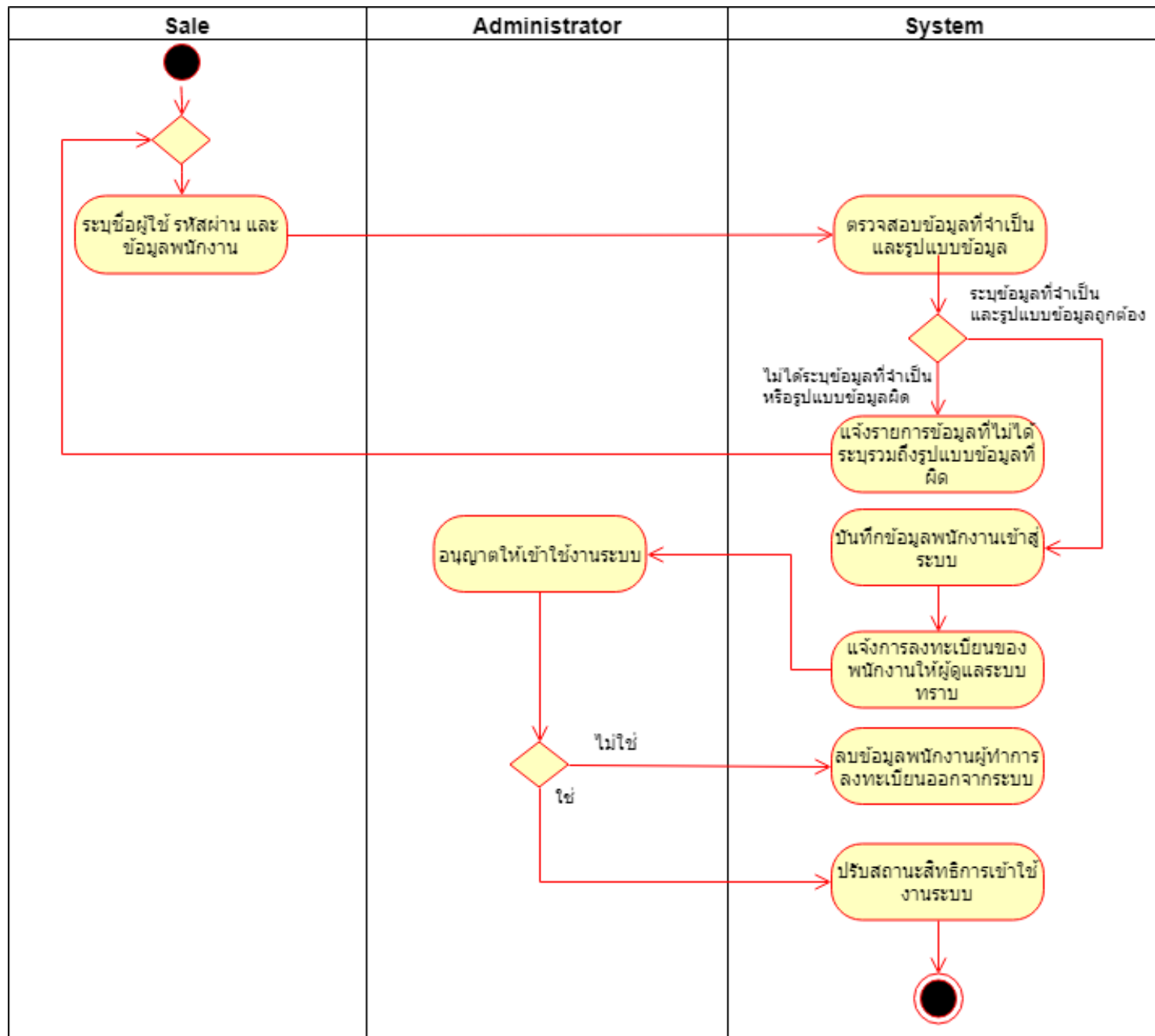
รูปแสดง ตัวอย่าง Class Diagram ของระบบบริหารจัดการการขาย

Activity Diagram

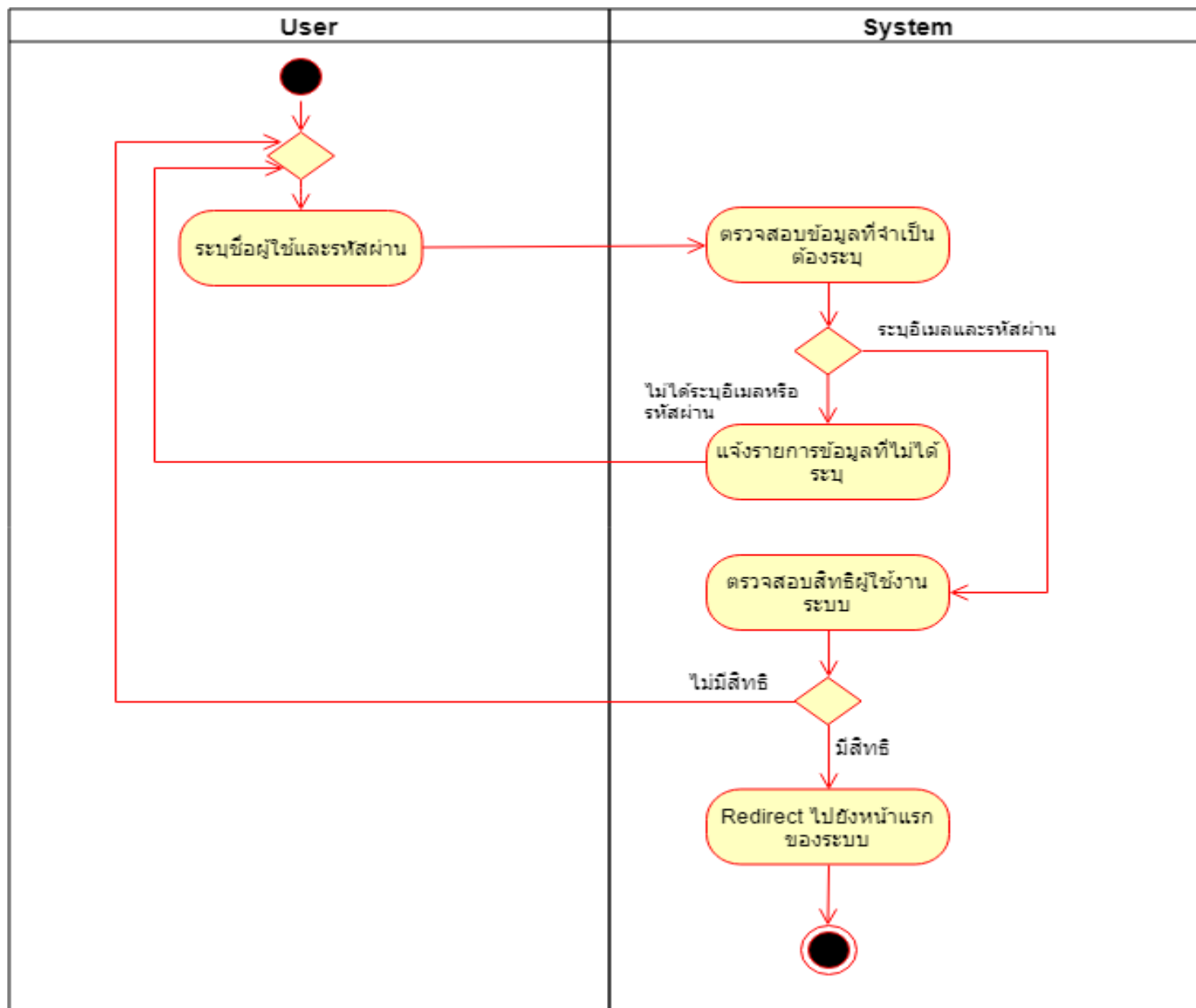
เป็นแผนภาพที่ใช้แสดงลำดับเหตุการณ์ของกิจกรรมในระบบ จากขั้นตอนหนึ่งไปยังอีกขั้นตอนหนึ่ง และแสดงให้เห็นว่าแต่ละขั้นตอนใคร (อาจเป็นคนหรือระบบ) เป็นผู้ดำเนินการ



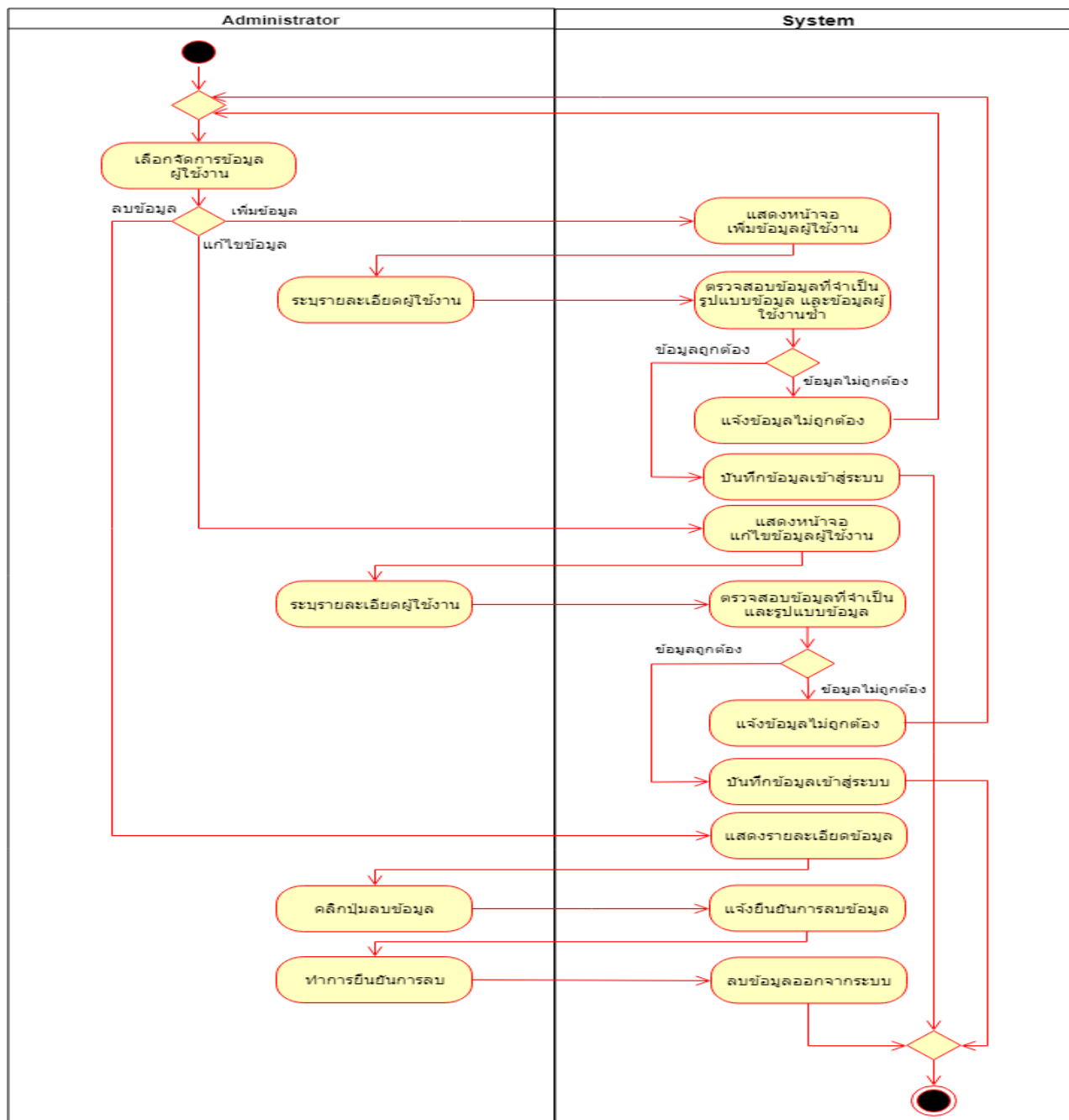
รูปแสดงสัญลักษณ์ของ Activity Diagram



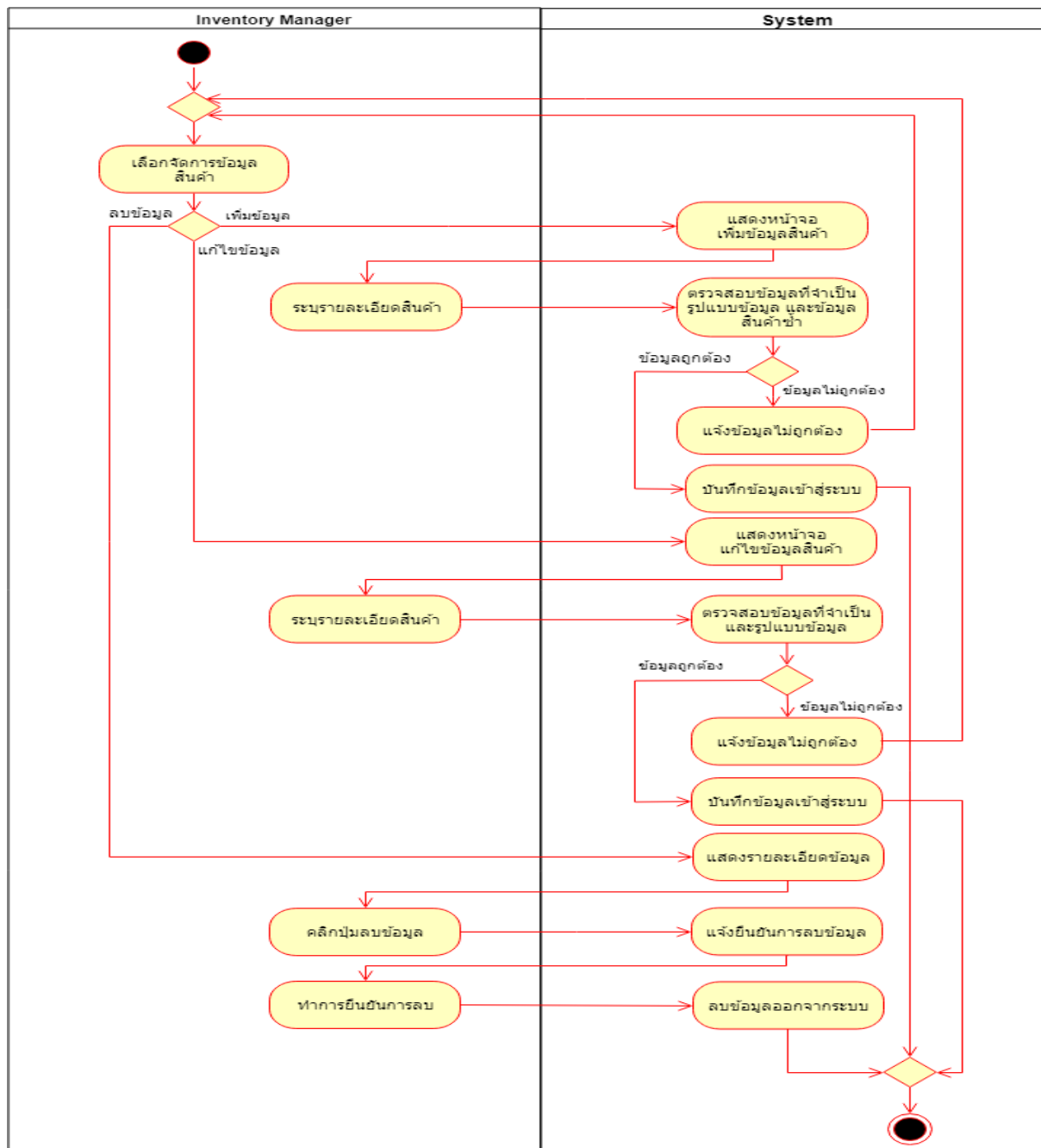
รูปแสดง ตัวอย่าง Activity Diagram ในการลงทะเบียน (Register) ของระบบบริหารจัดการการขาย



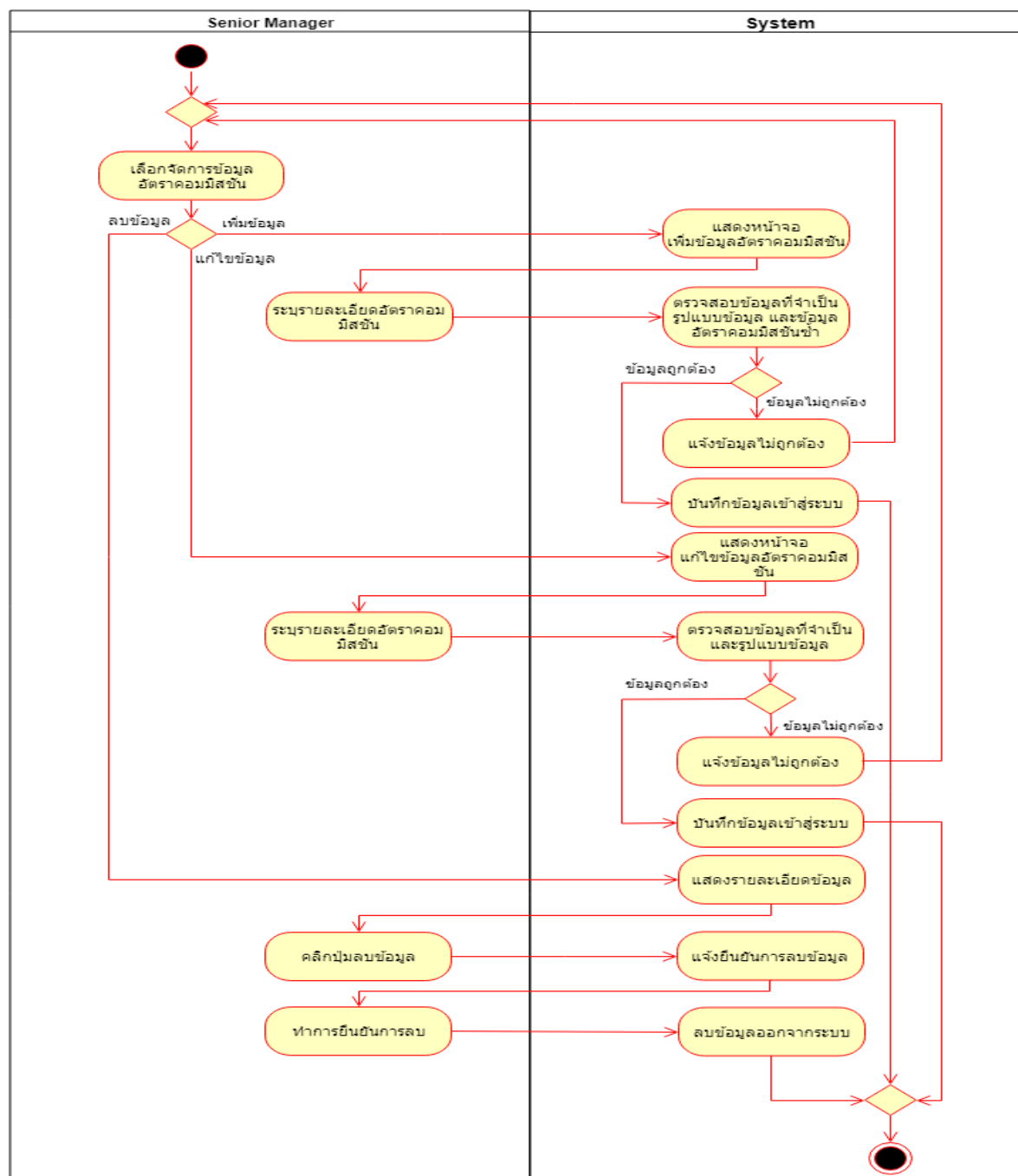
รูปแสดง ตัวอย่าง Activity Diagram ในการเข้าสู่ระบบ (Login) ของระบบบริหารจัดการการขาย



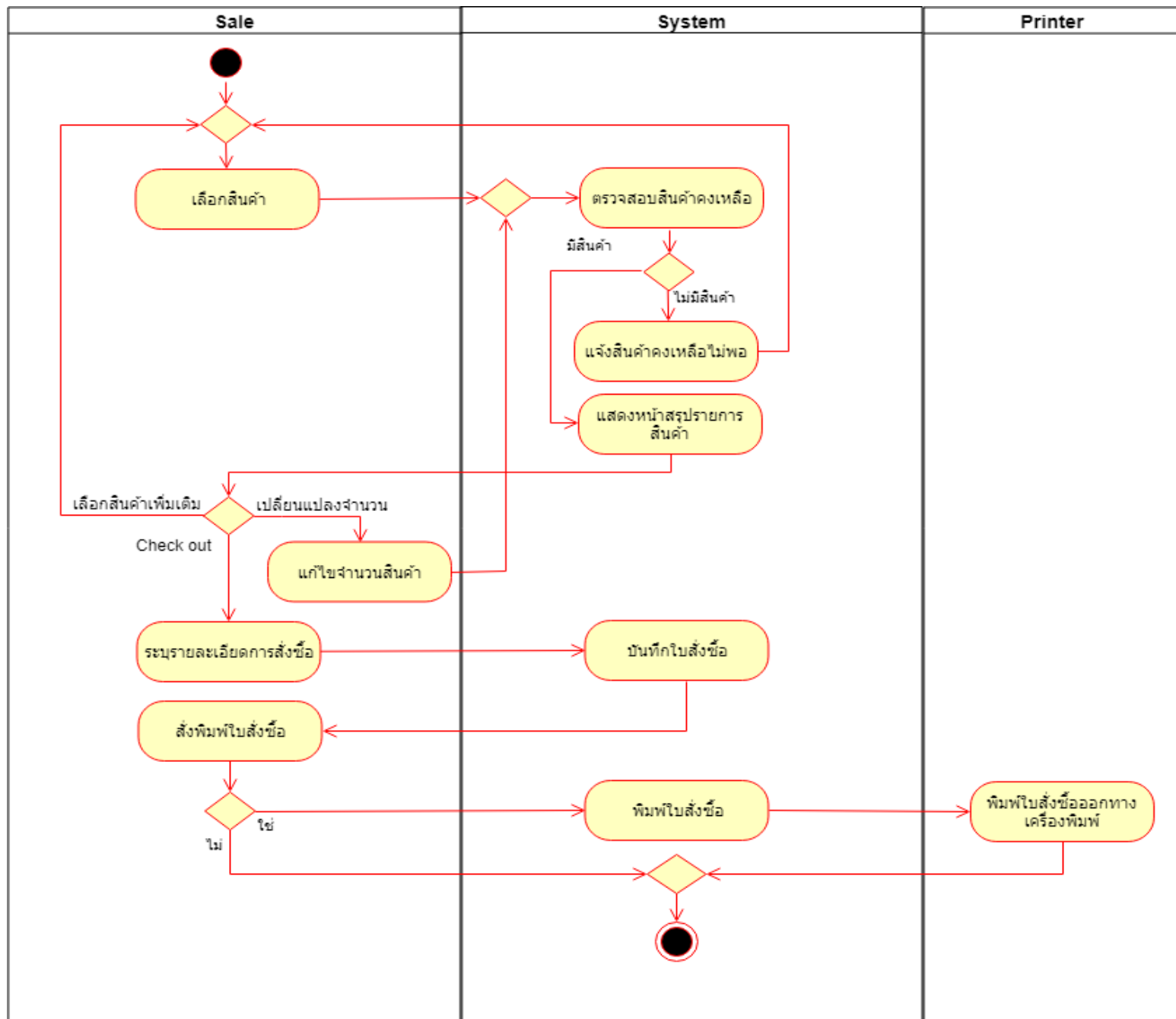
รูปแสดง ตัวอย่าง Activity Diagram ในการจัดการผู้ใช้งานระบบ (Manage User) ของระบบบริหารจัดการการขาย



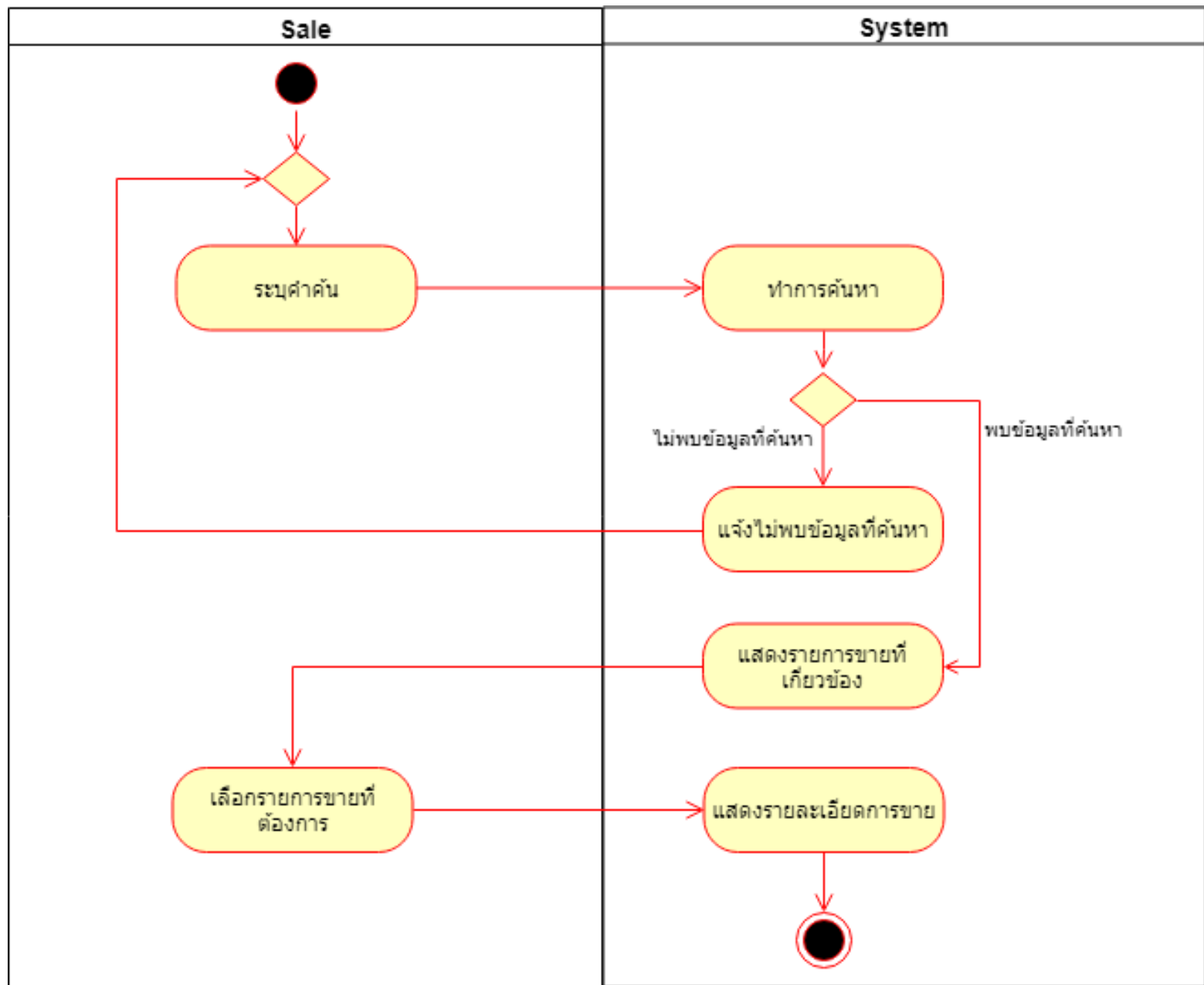
รูปแสดง ตัวอย่าง Activity Diagram ในการจัดการสินค้าคงคลัง (Manage Inventory) ของระบบบริหารจัดการการขาย



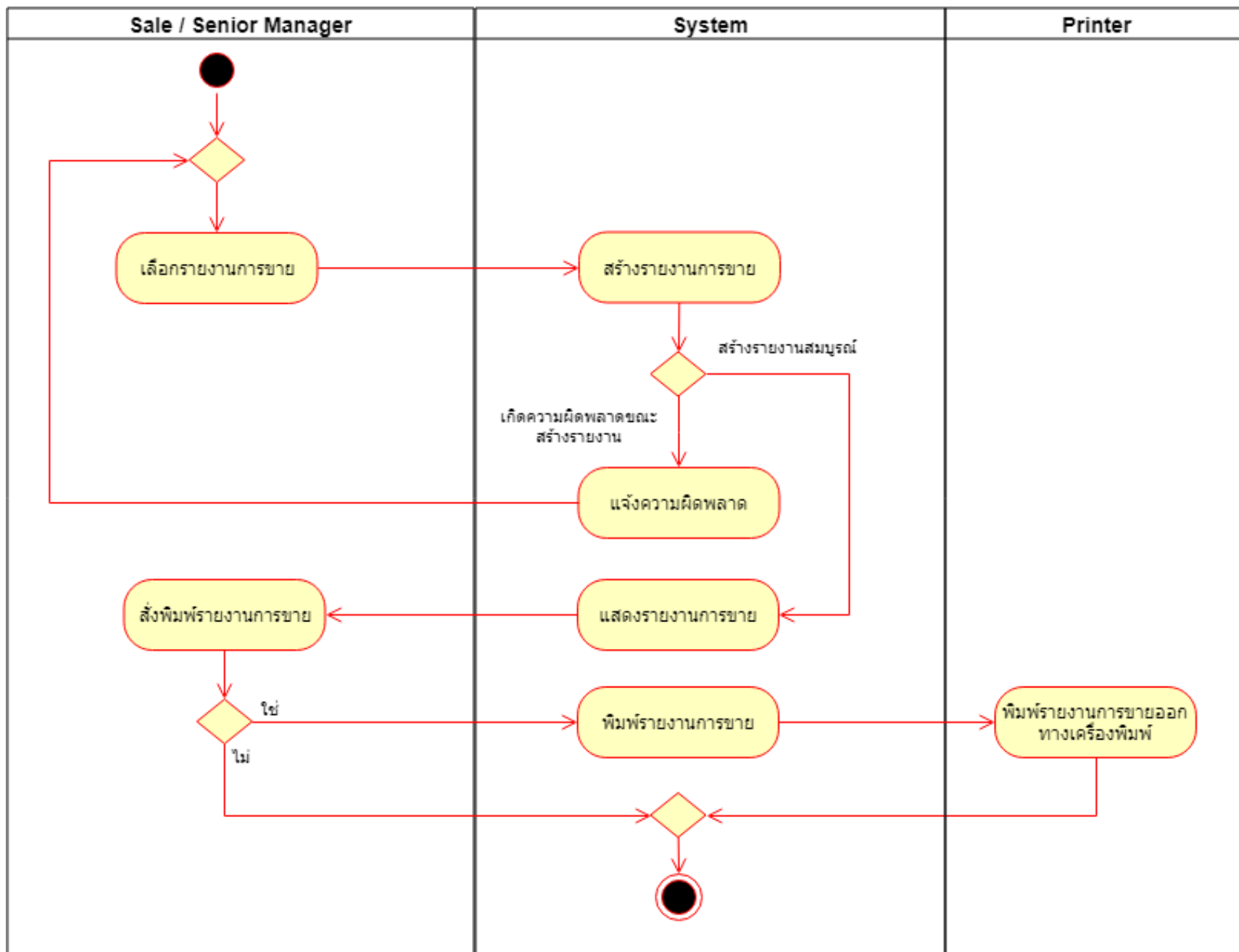
รูปแสดง ตัวอย่าง Activity Diagram ในการกำหนดอัตราคอมมิสชัน (Define Commission Rate) ของระบบบริหารจัดการการขาย



รูปแสดง ตัวอย่าง Activity Diagram ในการสั่งซื้อ (Order) ของระบบบริหารจัดการการขาย



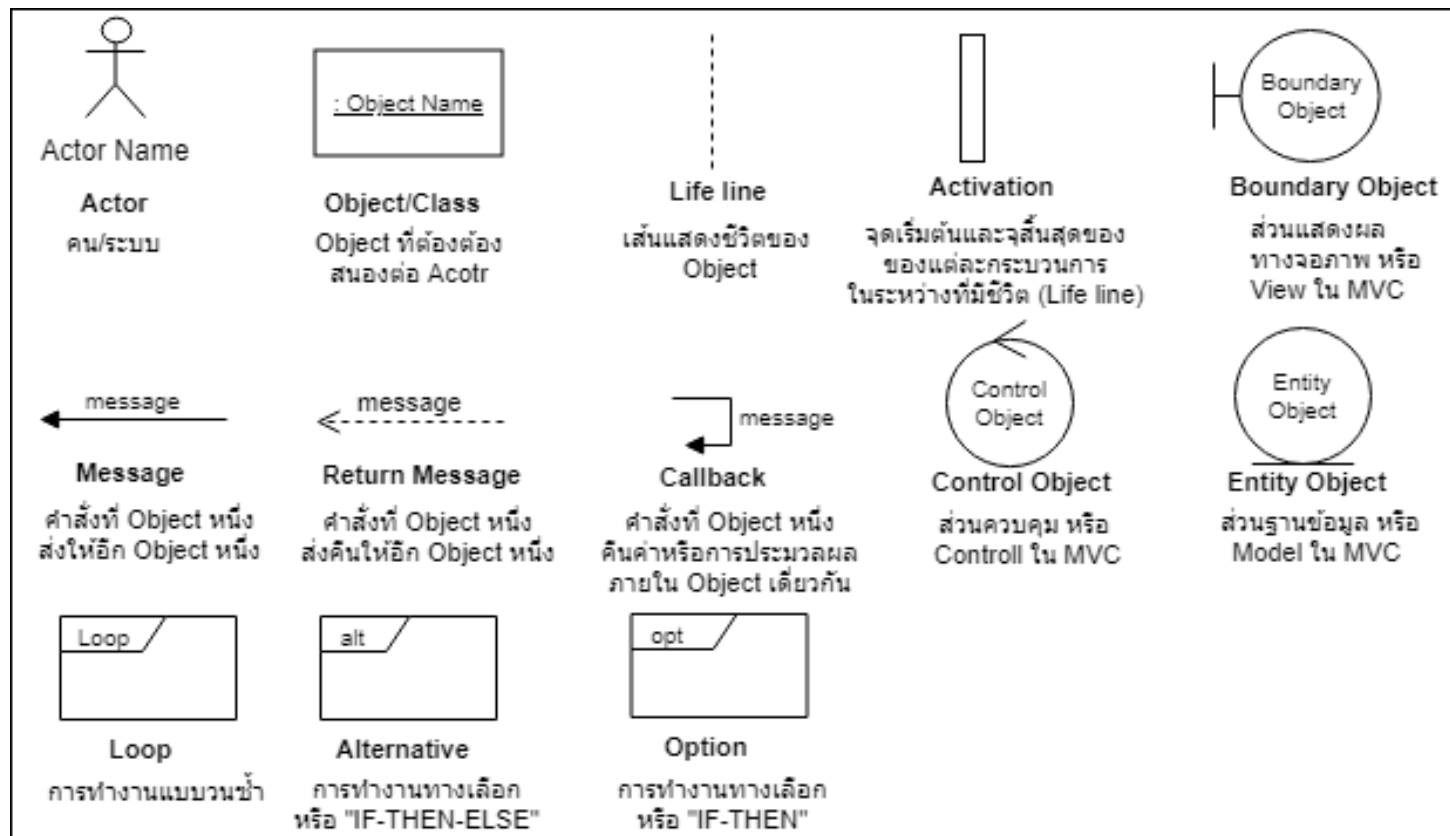
รูปแสดง ตัวอย่าง Activity Diagram ในการค้นหารายการขาย (Search Sale Item) ของระบบบริหารจัดการการขาย



รูปแสดง ตัวอย่าง Activity Diagram ในการสร้างรายงาน (Generate Report) ของระบบบริหารจัดการการขาย

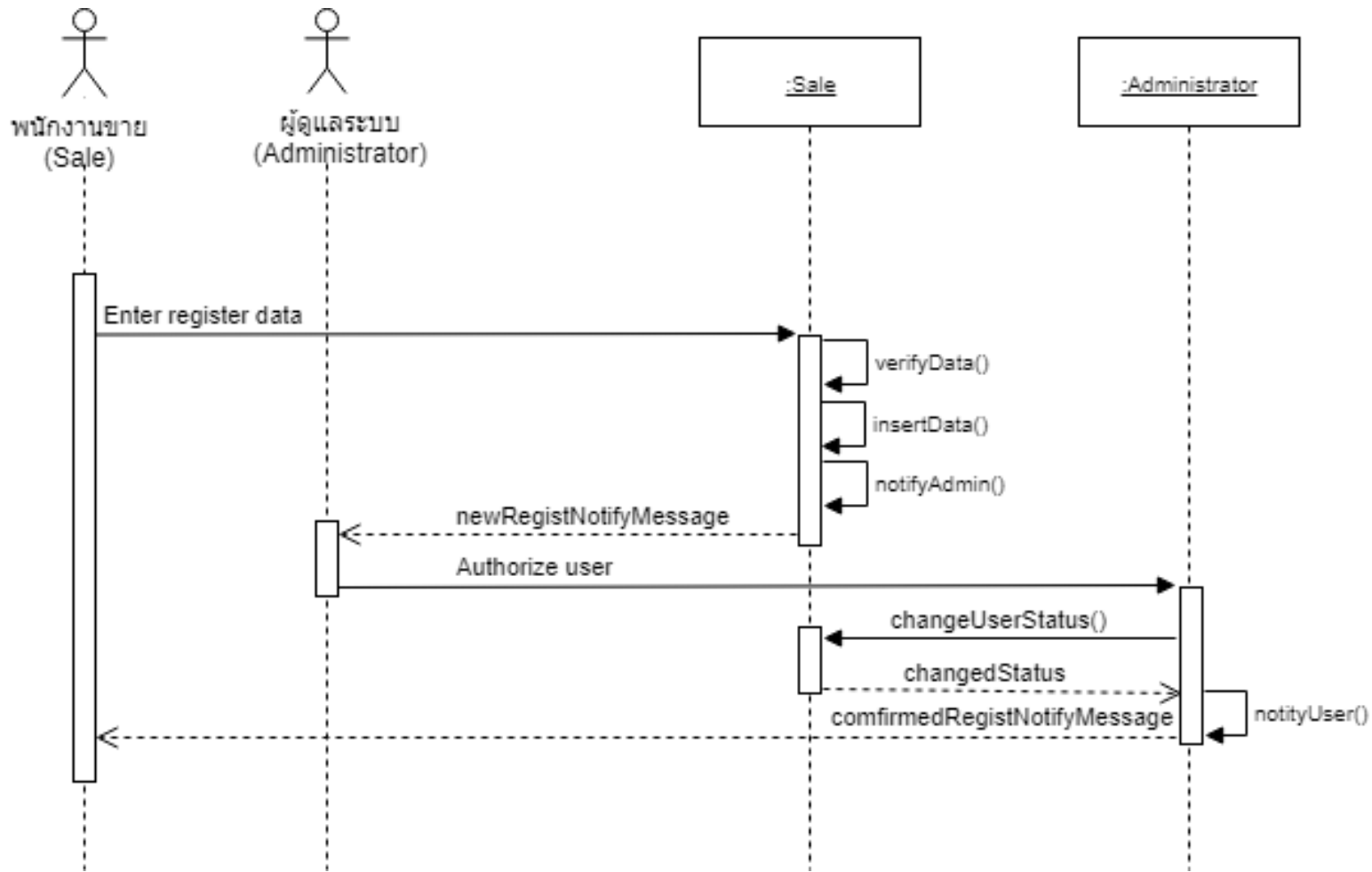
Sequence Diagram

เป็นแผนภาพแสดงการทำงานร่วมกันระหว่าง Object เพื่อให้กระบวนการสำเร็จ โดยเฉพาะการส่ง Message ระหว่าง Object ตามลำดับเวลา (Sequence)

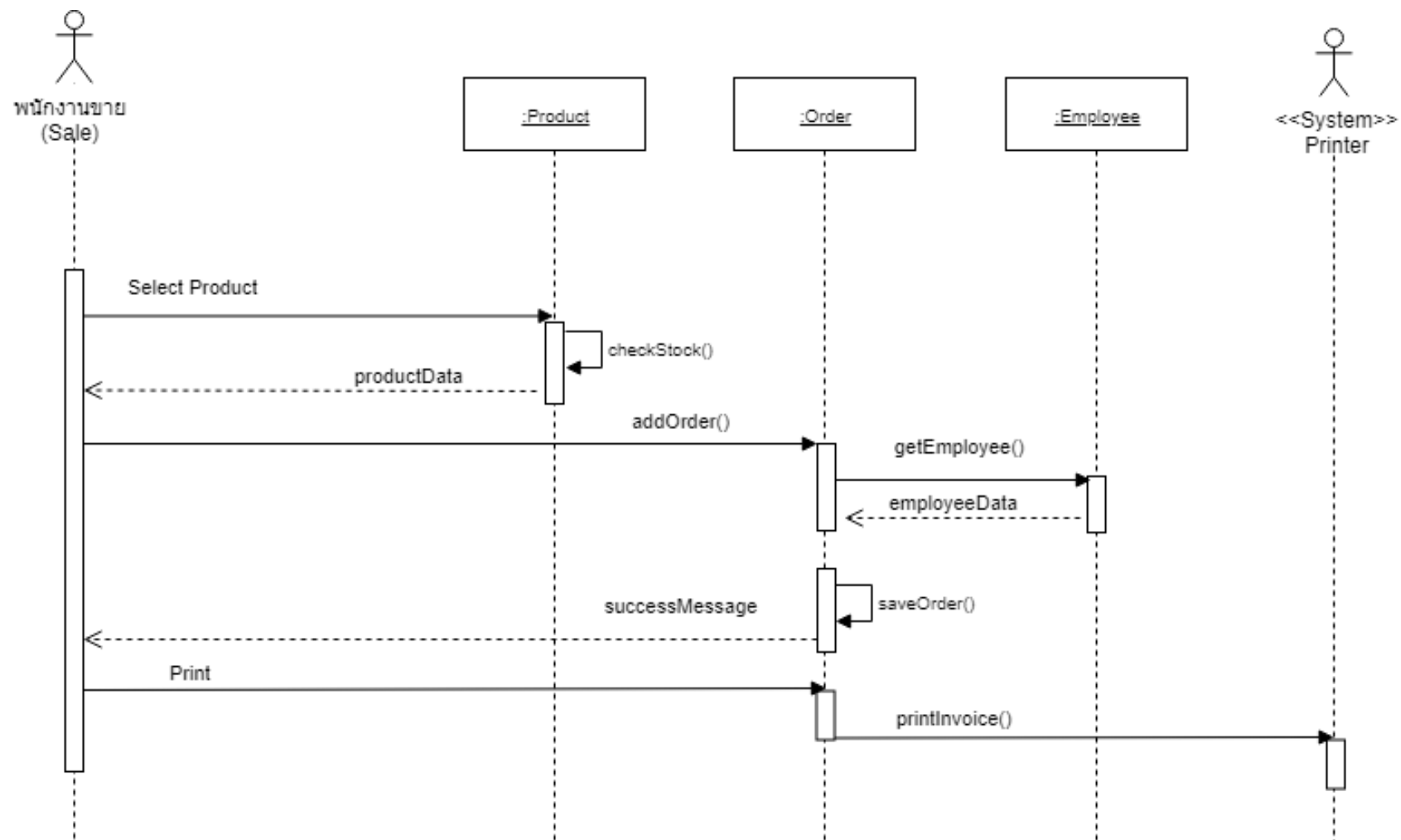


รูปแสดงสัญลักษณ์ของ Sequence Diagram

การเขียน Sequence Diagram จาก Basic Flows ใน Use Case Description

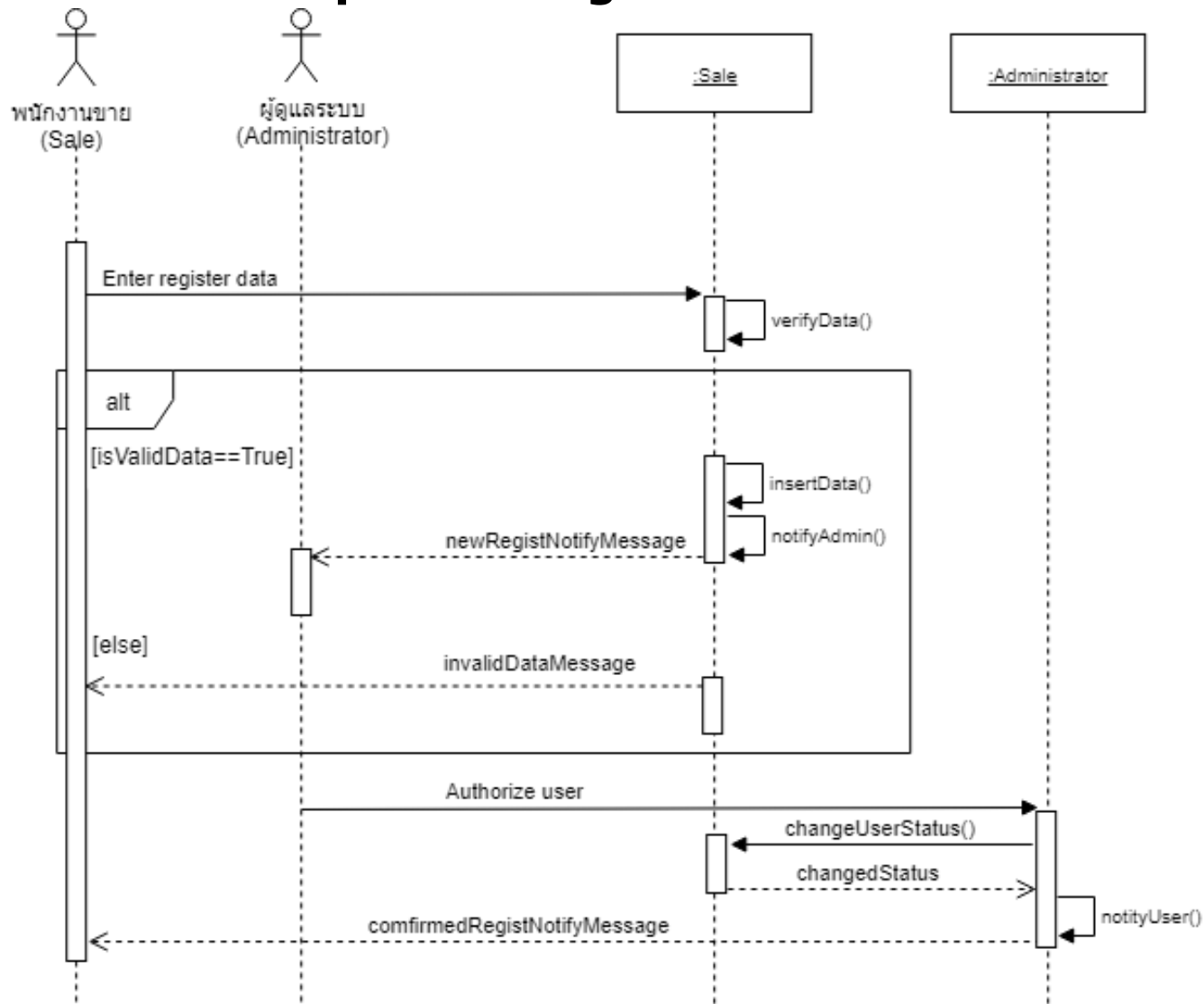


รูปแสดงตัวอย่าง Sequence Diagram ของ Use Case ลงทะเบียน (Register) และกำหนดสิทธิผู้ใช้งาน (Authorize User) ของระบบบริหารจัดการการขาย



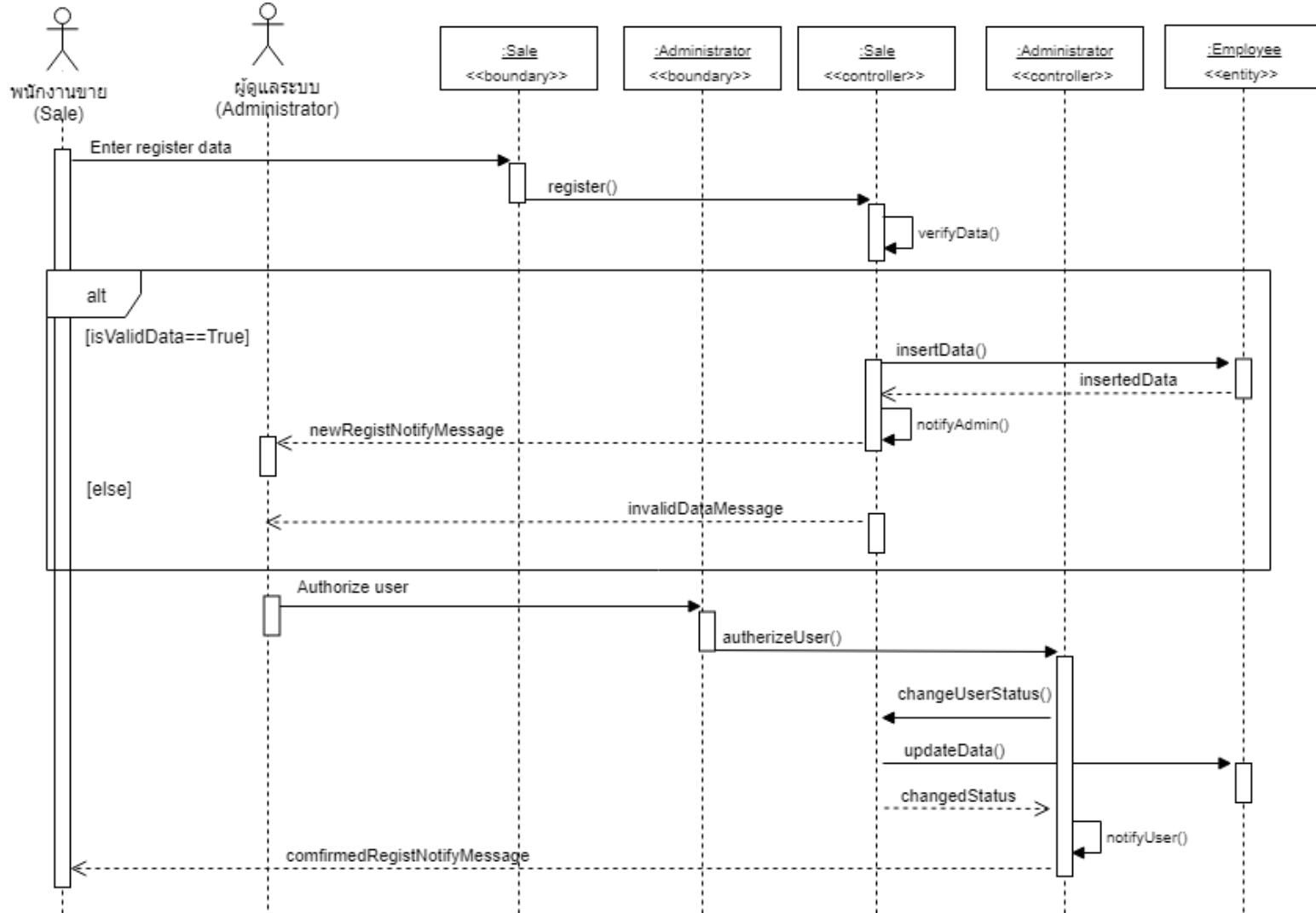
รูปแสดงตัวอย่าง Sequence Diagram จาก Use Case สั่งซื้อ ของระบบบริหารจัดการการขาย

การเขียน Sequence Diagram โดยใช้ Alternative Fragment



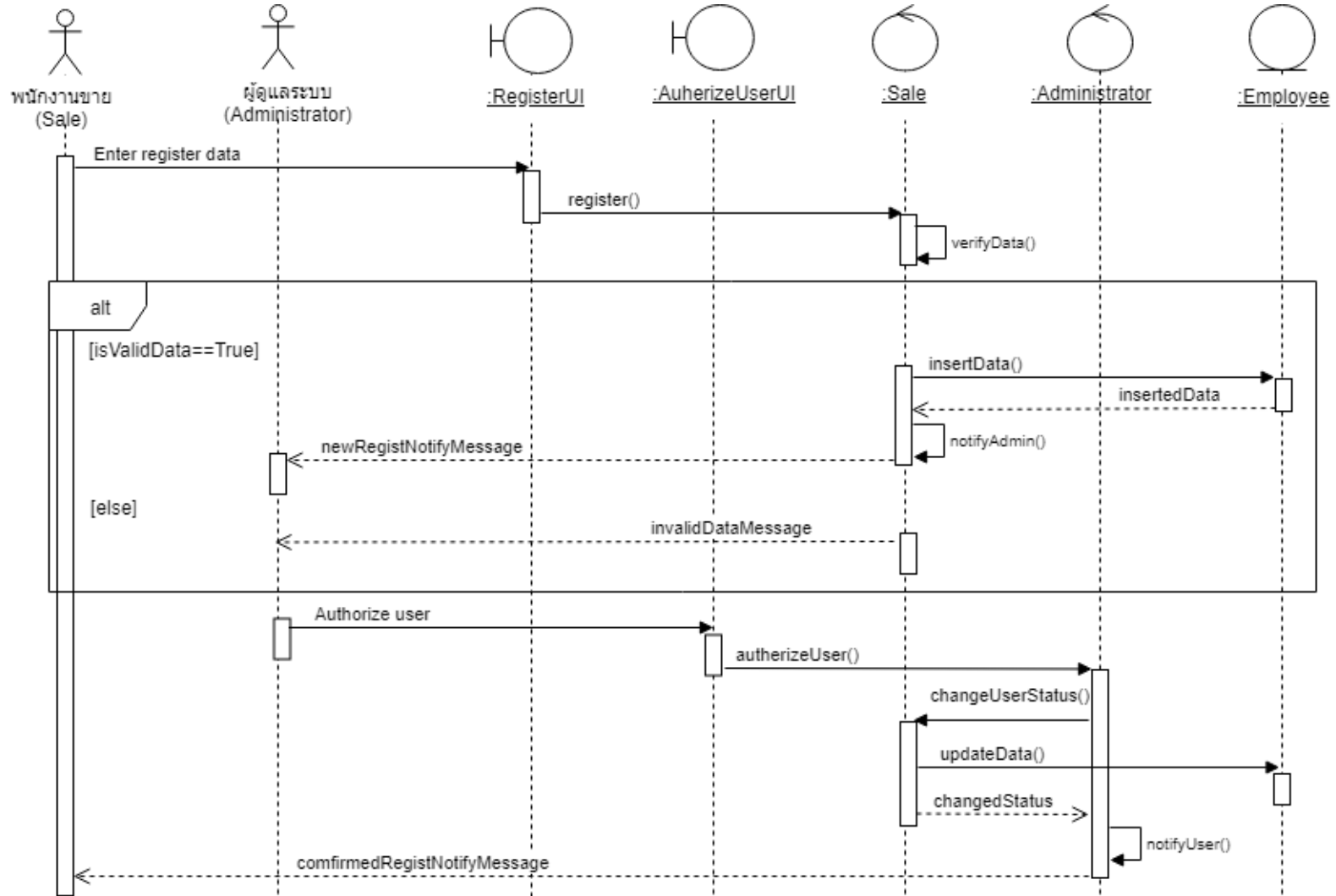
รูปแสดงตัวอย่าง Sequence Diagram จากลงทะเบียน (Register) และกำหนดสิทธิ์ผู้ใช้งาน (Authorize User) ของระบบบริหารจัดการการขาย

การเขียน Sequence Diagram ในรูปแบบ MVC



รูปแสดงตัวอย่าง Sequence Diagram จากลงทะเบียน (Register) และกำหนดสิทธิผู้ใช้งาน (Authorize User) ของระบบบริหารจัดการการขาย

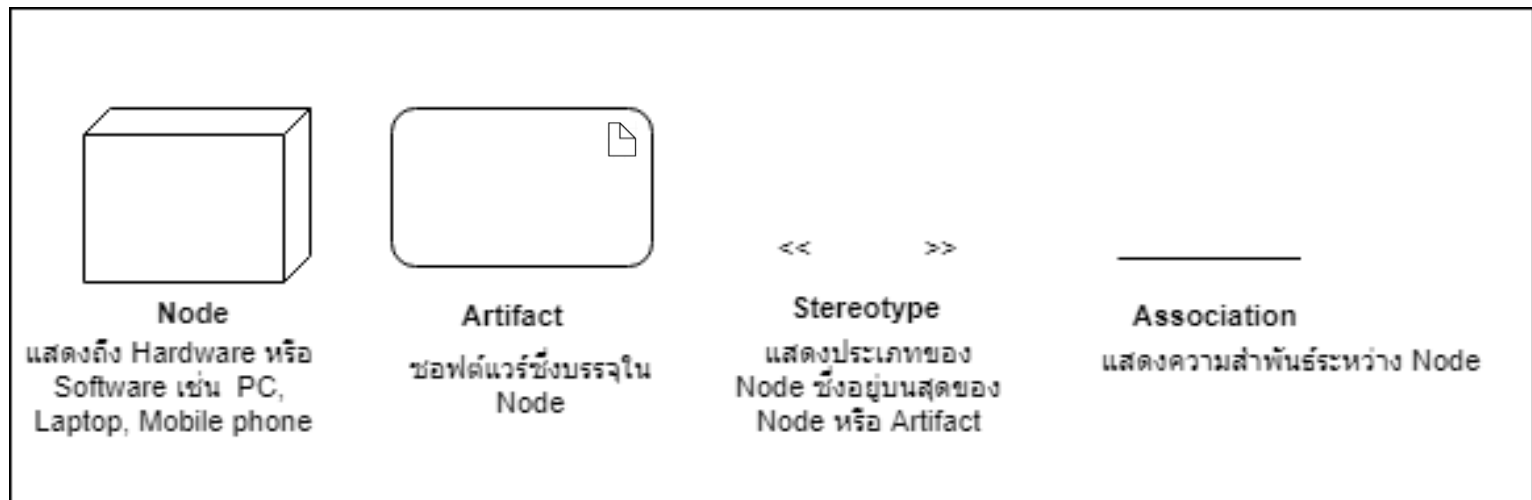
การเขียน Sequence Diagram ในรูปแบบ MVC [โดยใช้ ICON]



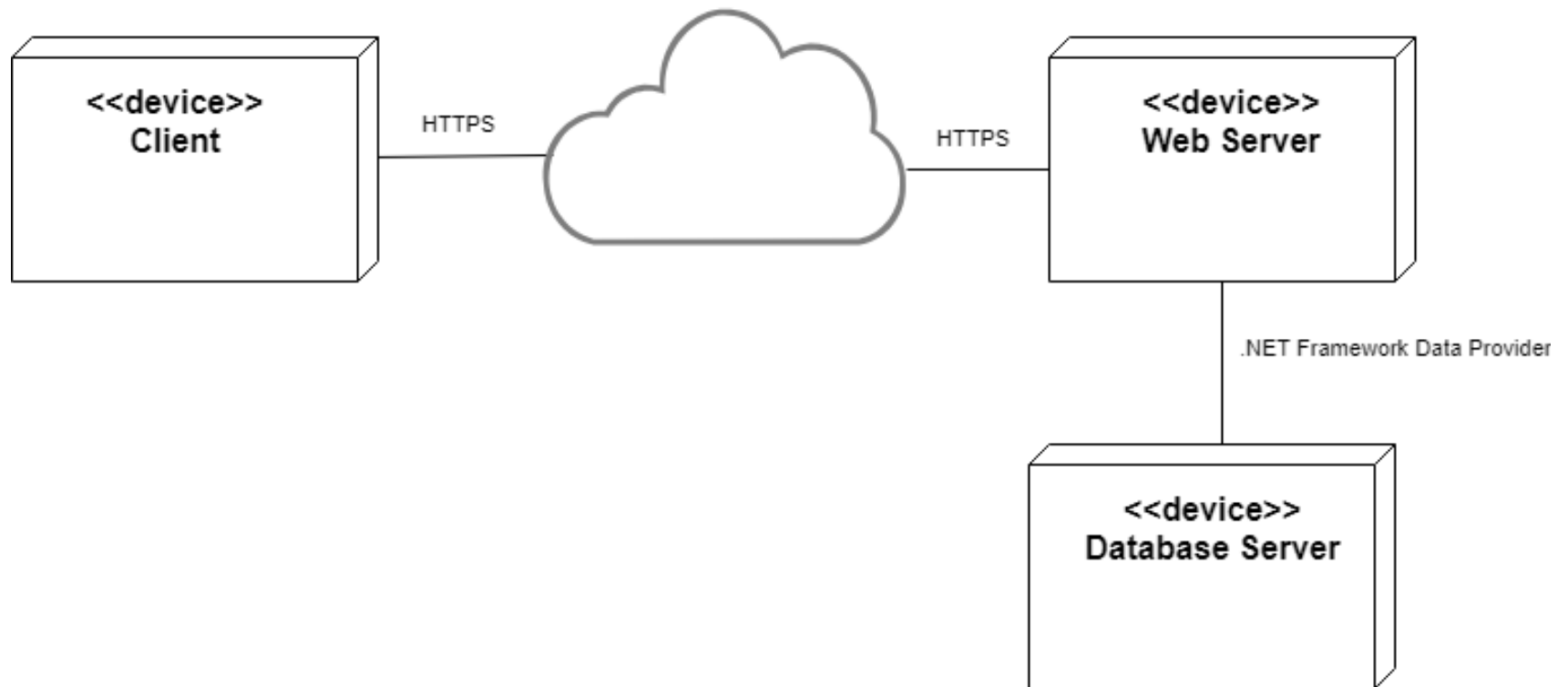
รูปแสดงตัวอย่าง Sequence Diagram จากลงทะเบียน (Register) และกำหนดสิทธิ์ผู้ใช้งาน (Authorize User) ของระบบบริหารจัดการการขาย

Deployment Diagram

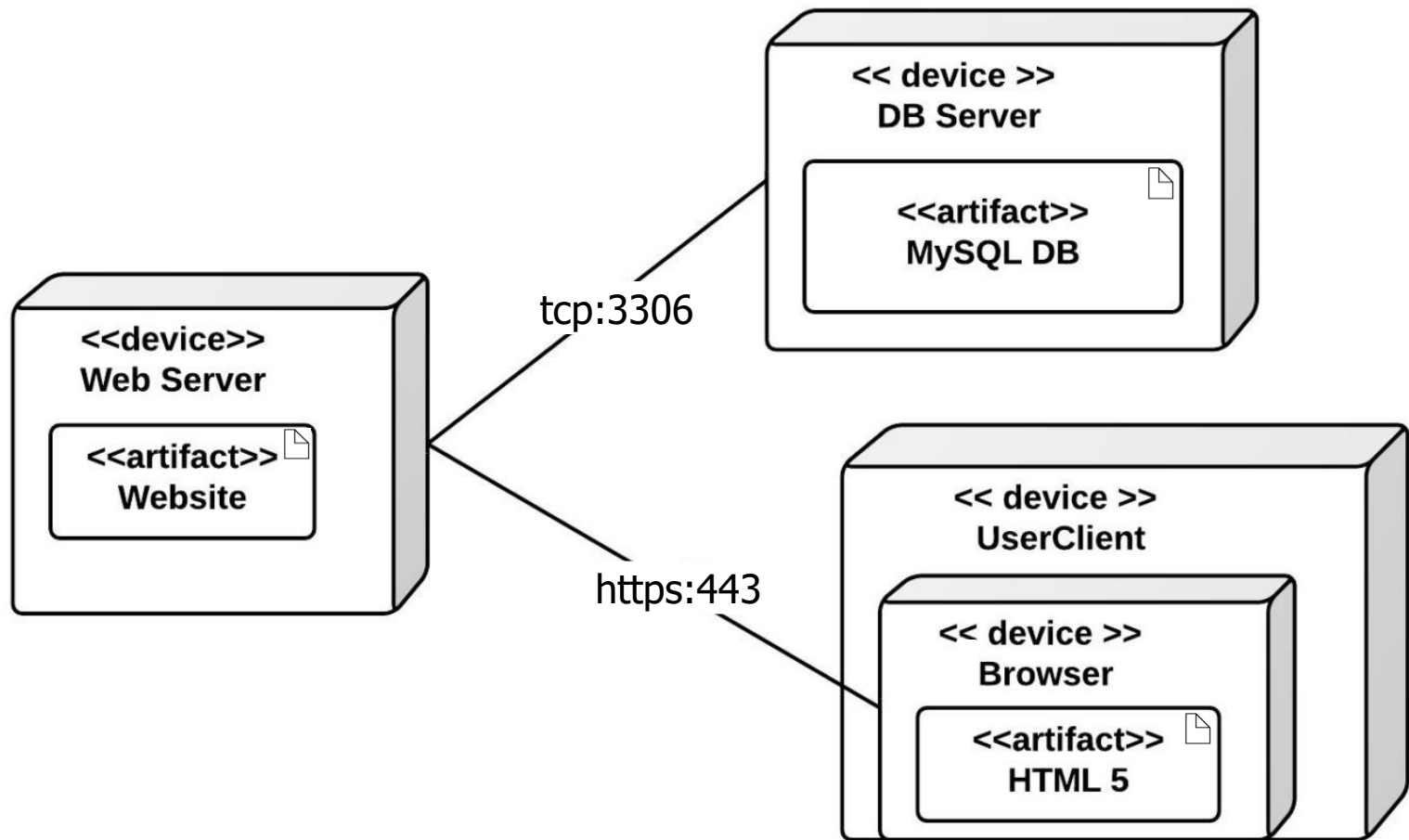
เป็นแผนภาพแสดงสถาปัตยกรรมของระบบ ประกอบด้วย ฮาร์ดแวร์ และซอฟต์แวร์ที่เกี่ยวข้องกับซอฟต์แวร์ที่พัฒนา โดยมองฮาร์ดแวร์และซอฟต์แวร์เหล่านั้นเป็น Object เช่นเดียวกันกับซอฟต์แวร์ที่พัฒนา



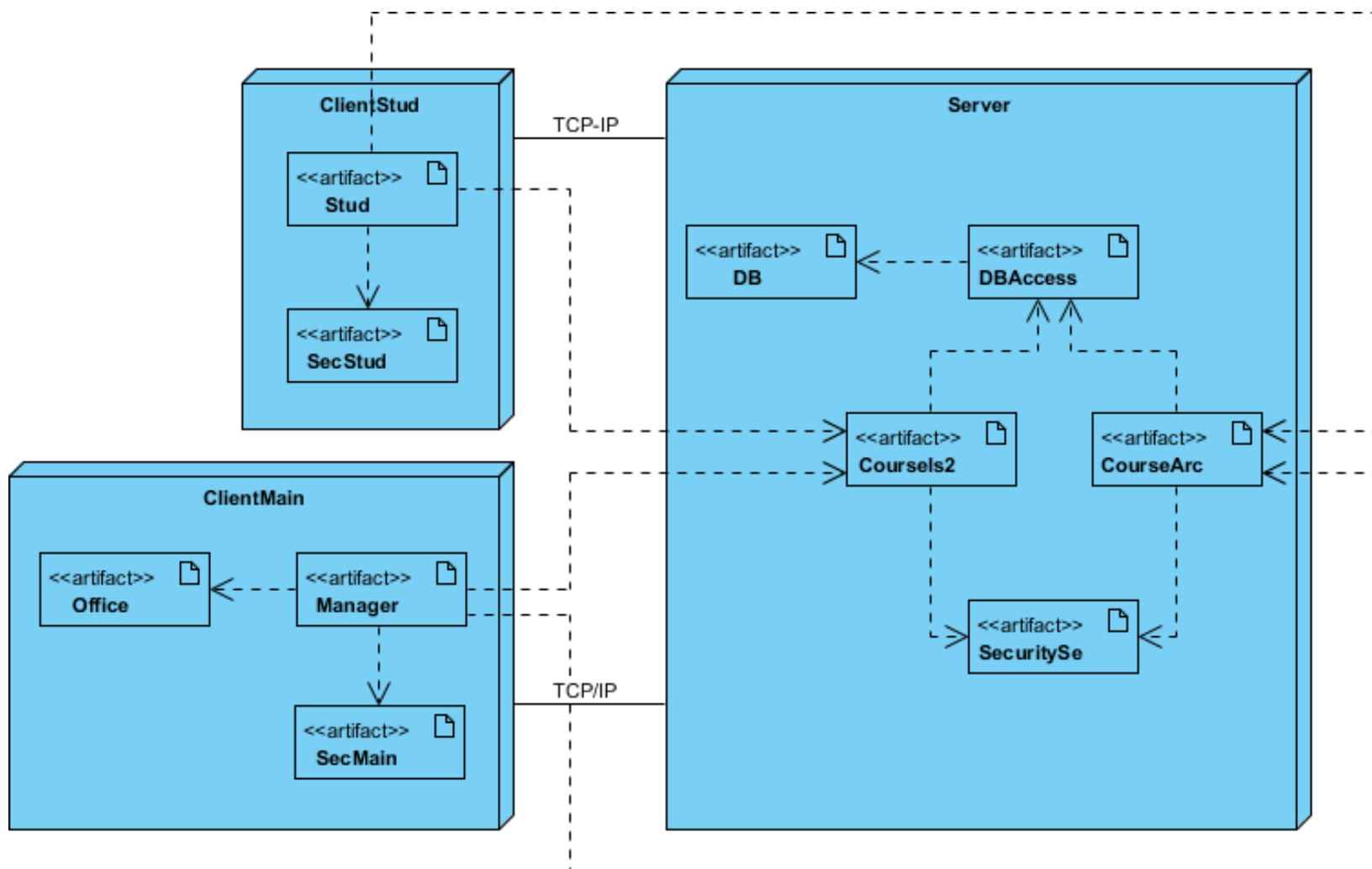
รูปแสดงสัญลักษณ์ของ Deployment Diagram



รูปแสดงตัวอย่างสถาปัตยกรรมระบบแบบ Client-Server (แบบที่ 1)



รูปแสดงตัวอย่างสถาปัตยกรรมระบบแบบ Client-Server (แบบที่ 2)



รูปแสดงตัวอย่างสถาปัตยกรรมระบบแบบ Client-Server (แบบที่ 3)