

1. Use Case Diagram

ความหมาย

- แสดงความสัมพันธ์ระหว่าง **ผู้ใช้งาน (Actor)** กับ **ระบบ (System)** ในแง่ของหน้าที่หรือการทำงาน (Use Case) ที่ระบบสามารถทำได้

องค์ประกอบหลัก

1. **Actor:** ผู้ใช้งานระบบ เช่น ผู้ใช้, ระบบภายนอก
 - **Primary Actor:** ผู้ใช้งานที่เริ่มต้นการทำงานของระบบ
 - **Secondary Actor:** ระบบหรือผู้ใช้งานที่ช่วยสนับสนุน
2. **Use Case:** ฟังก์ชันหรือหน้าที่ที่ระบบต้องทำ เช่น การสั่งซื้อสินค้า, การเข้าสู่ระบบ
3. **Association:** เส้นเชื่อมระหว่าง Actor กับ Use Case
4. **Relationships:**
 - **Include:** การใช้ฟังก์ชันย่อยร่วมกัน (บังคับใช้)
 - **Extend:** การเพิ่มฟังก์ชันในสถานการณ์เฉพาะ
 - **Generalization:** ความสัมพันธ์แบบคลาสแม่-ลูกของ Use Case หรือ Actor

ตัวอย่าง

ระบบร้านค้าออนไลน์:

- **Actor:** ลูกค้า, ผู้ดูแลระบบ
- **Use Case:** ลงทะเบียน, เข้าสู่ระบบ, เลือกสินค้า, ชำระเงิน
- **ความสัมพันธ์:**
 - ลูกค้าต้อง **Include** ฟังก์ชันเข้าสู่ระบบก่อนชำระเงิน

2. Use Case Description

ความหมาย

เป็นเอกสารที่อธิบายรายละเอียดของ Use Case แต่ละอัน เพื่อเข้าใจการทำงานของระบบเชิงลึก

โครงสร้างทั่วไป

1. **Use Case Name:** ชื่อของ Use Case
2. **Description:** คำอธิบายของ Use Case
3. **Actor:** ระบุ Actor ที่เกี่ยวข้อง
4. **Pre-condition:** เงื่อนไขก่อนเริ่มทำงาน
5. **Post-condition:** สถานะหลังการทำงานเสร็จ
6. **Main Flow:** กระบวนการหลักของ Use Case
7. **Alternative Flow:** กรณีทางเลือก (ถ้าเกิดข้อผิดพลาดหรือทางเลือกอื่น)
8. **Exception:** กรณีที่ระบบไม่สามารถทำงานได้
9. **Trigger:** เหตุการณ์ที่เริ่มต้น Use Case

ตัวอย่าง

Use Case: ชำระเงิน

- **Actor:** ลูกค้า
- **Pre-condition:** ลูกค้าต้องลงชื่อเข้าใช้
- **Main Flow:**
 1. ลูกค้าเลือกวิธีชำระเงิน

2. ลูกค้ากรอกข้อมูลบัตรเครดิต
 3. ระบบตรวจสอบความถูกต้อง
 4. ระบบยืนยันการชำระเงิน
- **Alternative Flow:** กรอกข้อมูลผิด → แจ้งเตือน
-

3. Sequence Diagram

ความหมาย

แสดงลำดับการส่งข้อความระหว่าง **Object** หรือ **Actor** ในระบบ และวิธีที่พวกมันมีปฏิสัมพันธ์กันเมื่อเวลาผ่านไป

องค์ประกอบหลัก

1. **Actor:** ผู้ใช้งาน
2. **Object:** หน่วยของระบบที่ทำงานร่วมกัน
3. **Lifeline:** เส้นแนวตั้งแสดงช่วงชีวิตของ Object
4. **Message:** เส้นลูกศรแสดงการสื่อสารระหว่าง Object เช่น การเรียก Method

ตัวอย่าง

การสั่งซื้อสินค้า:

1. ลูกค้าส่งคำขอสั่งซื้อสินค้า
2. ระบบตรวจสอบข้อมูลสินค้า
3. ระบบยืนยันคำสั่งซื้อ
4. ระบบแจ้งผลสำเร็จให้ลูกค้า

4. Activity Diagram

ความหมาย

แสดงลำดับขั้นตอนการทำงาน (Workflow) หรือกิจกรรมในระบบ

องค์ประกอบหลัก

1. Initial Node: จุดเริ่มต้น
2. Activity: ขั้นตอนของกระบวนการ
3. Decision Node: จุดตัดสินใจ (ทางเลือก)
4. Merge Node: รวมเส้นทางการทำงาน
5. Final Node: จุดสิ้นสุด

ตัวอย่าง

การสมัครสมาชิก

1. ผู้ใช้กรอกข้อมูล
2. ตรวจสอบข้อมูล
 - หากถูกต้อง → บันทึกข้อมูล
 - หากผิด → แจ้งข้อผิดพลาด
3. ยืนยันการสมัคร

5. State Diagram

ความหมาย

แสดงสถานะต่างๆ ของ Object ในระบบ และการเปลี่ยนสถานะ (State Transition)

องค์ประกอบหลัก

1. **State:** สถานะของ Object
2. **Transition:** การเปลี่ยนสถานะ
3. **Event:** เหตุการณ์ที่กระตุ้นให้เปลี่ยนสถานะ
4. **Action:** การกระทำเมื่อเกิด Transition

ตัวอย่าง

ระบบคำสั่งซื้อ:

- **State:** สั่งซื้อ → รอดำเนินการ → ชำระเงิน → จัดส่ง → เสร็จสิ้น
- **Transition:**
 - การชำระเงิน → ยืนยัน → เปลี่ยนสถานะเป็น "ชำระเงินแล้ว"
 - หากเกิดข้อผิดพลาด → กลับสู่สถานะ "รอดำเนินการ"