

**การสร้างแบบจำลองระบบ
(System Modeling)
และการวิเคราะห์ความต้องการ
(Requirements Analysis)**

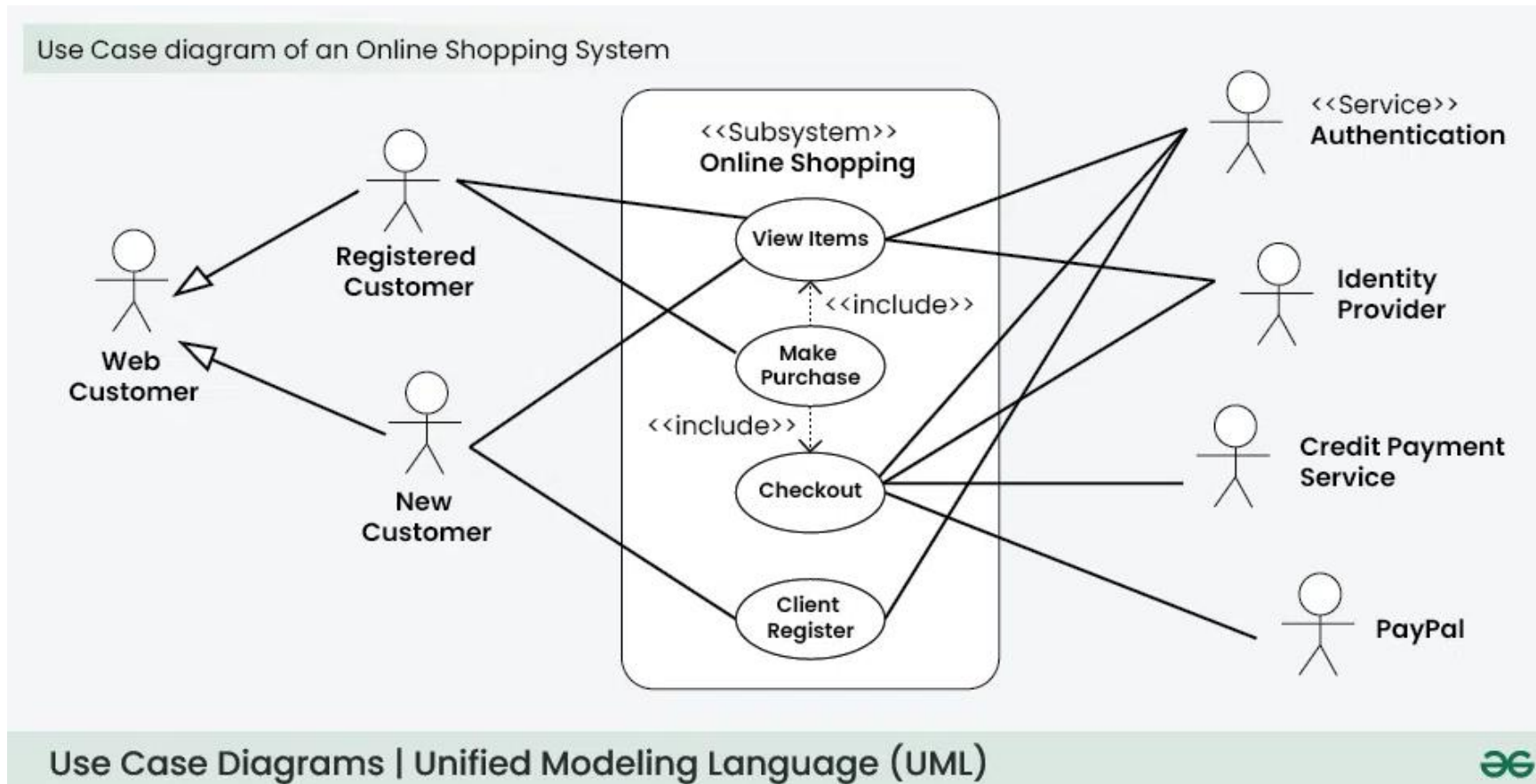
วัตถุประสงค์การเรียนรู้

- **ความเข้าใจแผนภาพ UML (UML Diagrams):** เข้าใจและวาดแผนภาพ Use Case, Activity และ Sequence Diagrams
- **การสร้างแบบจำลองความสัมพันธ์ระหว่างเอนทิตี (Entity-Relationship Modeling):** ออกแบบโครงสร้างฐานข้อมูล (Database Schema) ด้วยแผนภาพ ER (ER Diagram) ที่ถูกต้อง
- **การสร้างเอกสารข้อกำหนดซอฟต์แวร์ (SRS Document Creation):** เขียนข้อกำหนดรายละเอียดซอฟต์แวร์ (Software Requirements Specification) ที่สมบูรณ์และชัดเจน
- **การออกแบบแผนภาพ (Diagram Design):** สร้างแผนภาพที่สื่อสารความต้องการได้อย่างมีประสิทธิภาพ
- **การติดตามความต้องการ (Requirements Traceability):** เชื่อมโยงเรื่องราวผู้ใช้ (User Stories) กับแผนภาพและโครงสร้างฐานข้อมูล

ส่วนที่ 1: แผนภาพ Use Case (Use Case Diagram)

Use Case คืออะไร

- Use Case คือการอธิบายการปฏิสัมพันธ์ระหว่างตัวแสดง (Actor) (ผู้ใช้/ระบบภายนอก) กับระบบ (System) เพื่อบรรลุเป้าหมาย

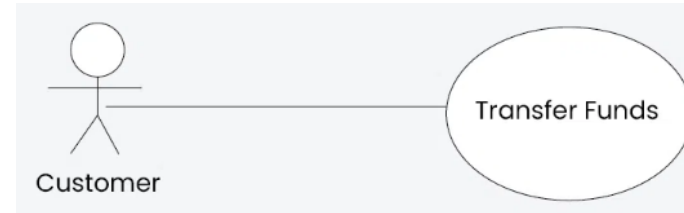


องค์ประกอบหลัก

- **Actor** - บุคคล/ระบบ/บริการภายนอกที่ปฏิสัมพันธ์กับระบบ
 - **Primary Actor:** ผู้เริ่มต้น use case (เช่น ลูกค้า Customer)
 - **Secondary Actor:** ระบบภายนอก/ผู้เสริม (เช่น ระบบชำระเงิน Payment Gateway)
- **Use Case** - การกระทำที่ระบบทำได้ (วงรี)
 - ตั้งชื่อด้วย**กริยา**: "เข้าสู่ระบบ (Login)", "ดูสินค้า (View Products)", "ประมวลผลการชำระเงิน (Process Payment)"
- **ความสัมพันธ์ (Relationships):**
 - **การเชื่อมโยง (Association):** Actor ใช้ Use Case (เส้นตรง)
 - **รวมเข้าด้วยกัน (Include):** Use Case นี้ต้องเรียก Use Case อื่น (เส้นประ <<include>>)
 - **ขยายความ (Extend):** Use Case นี้ขยายความ Use Case อื่น (เส้นประ <<extend>>)
 - **การสรุปทั่วไป (Generalization):** Actor หรือ Use Case ทั่วไป (หัวลูกศร)

ประเภทความสัมพันธ์ (Relationship)

- การเชื่อมโยง (Association)



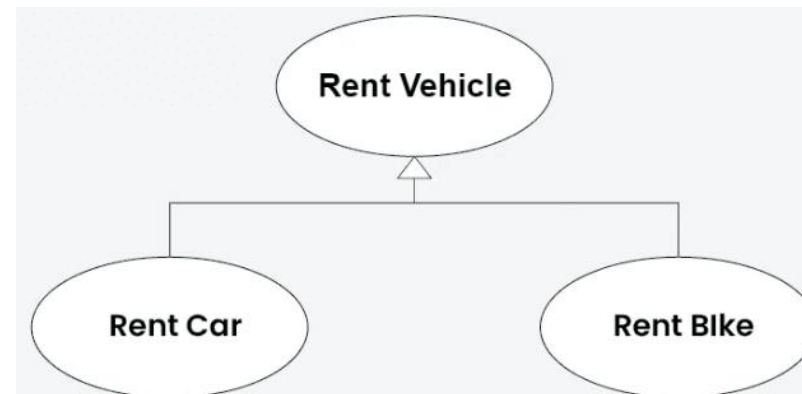
- ความสัมพันธ์แบบรวมเข้าด้วยกัน (Include Relationship)



- ความสัมพันธ์แบบขยายความ (Extend Relationship)



- ความสัมพันธ์แบบทั่วไป (Generalization)



ขั้นตอนการวาด Use Case Diagram

- **Step 1: ระบุ Actors**
- **Step 2: ระบุ Use Cases**
- **Step 3: ระบุ Relationships**
- **Step 4: วาด Diagram**

Primary Actors (ผู้เริ่มต้น):

- Customer (ซื้อสินค้า)
- Admin (จัดการระบบ)

Secondary Actors (ระบบภายนอก):

- Payment Gateway (ประมวลผลการชำระเงิน)
- Email Service (ส่งอีเมล)
- Inventory System (ระบบสต็อก)

Customer Use Cases:

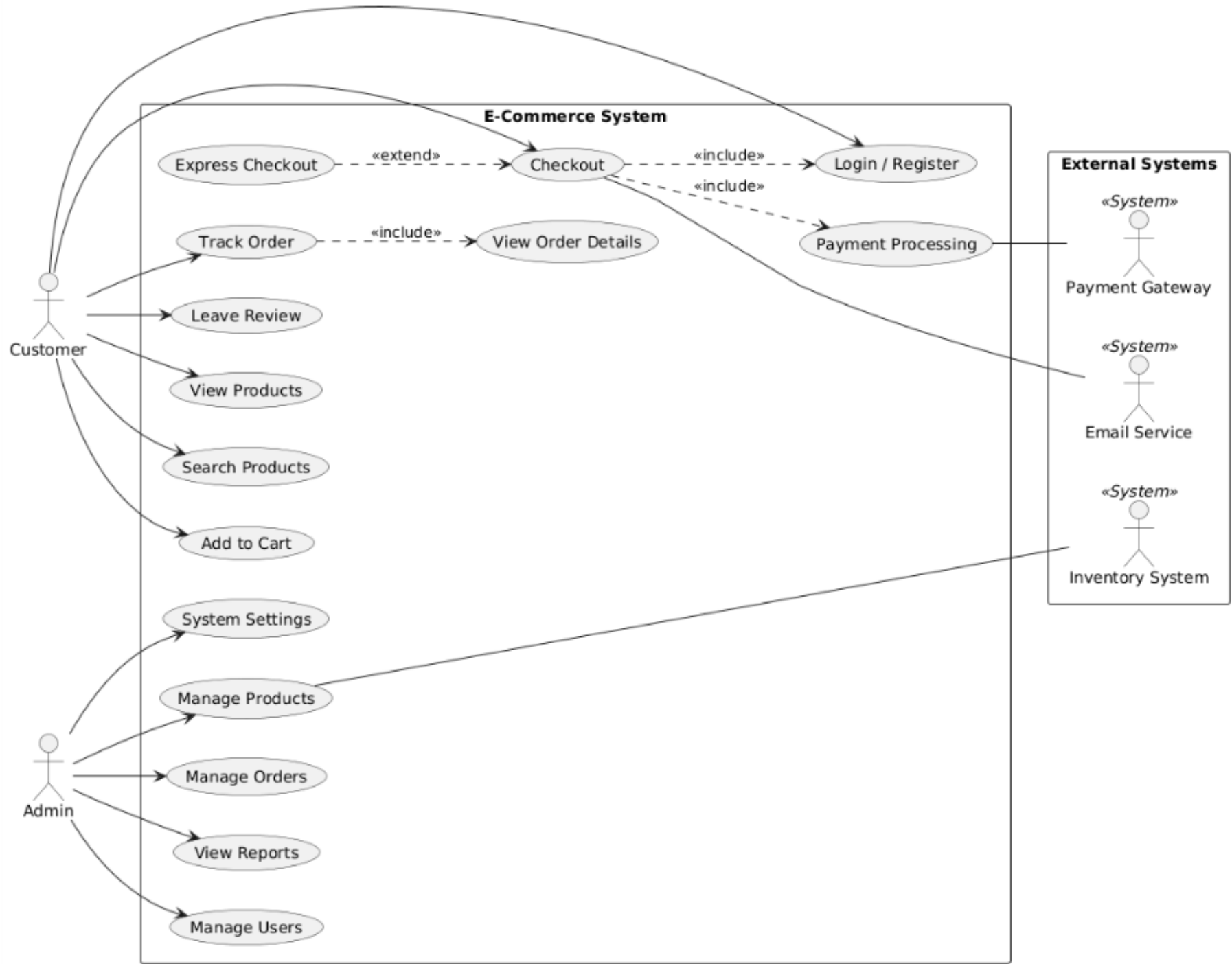
- Login / Register
- View Products
- Search Products
- Add to Cart
- Checkout
- Track Order
- Leave Review

Admin Use Cases:

- Manage Products
- Manage Orders
- View Reports
- Manage Users
- System Settings

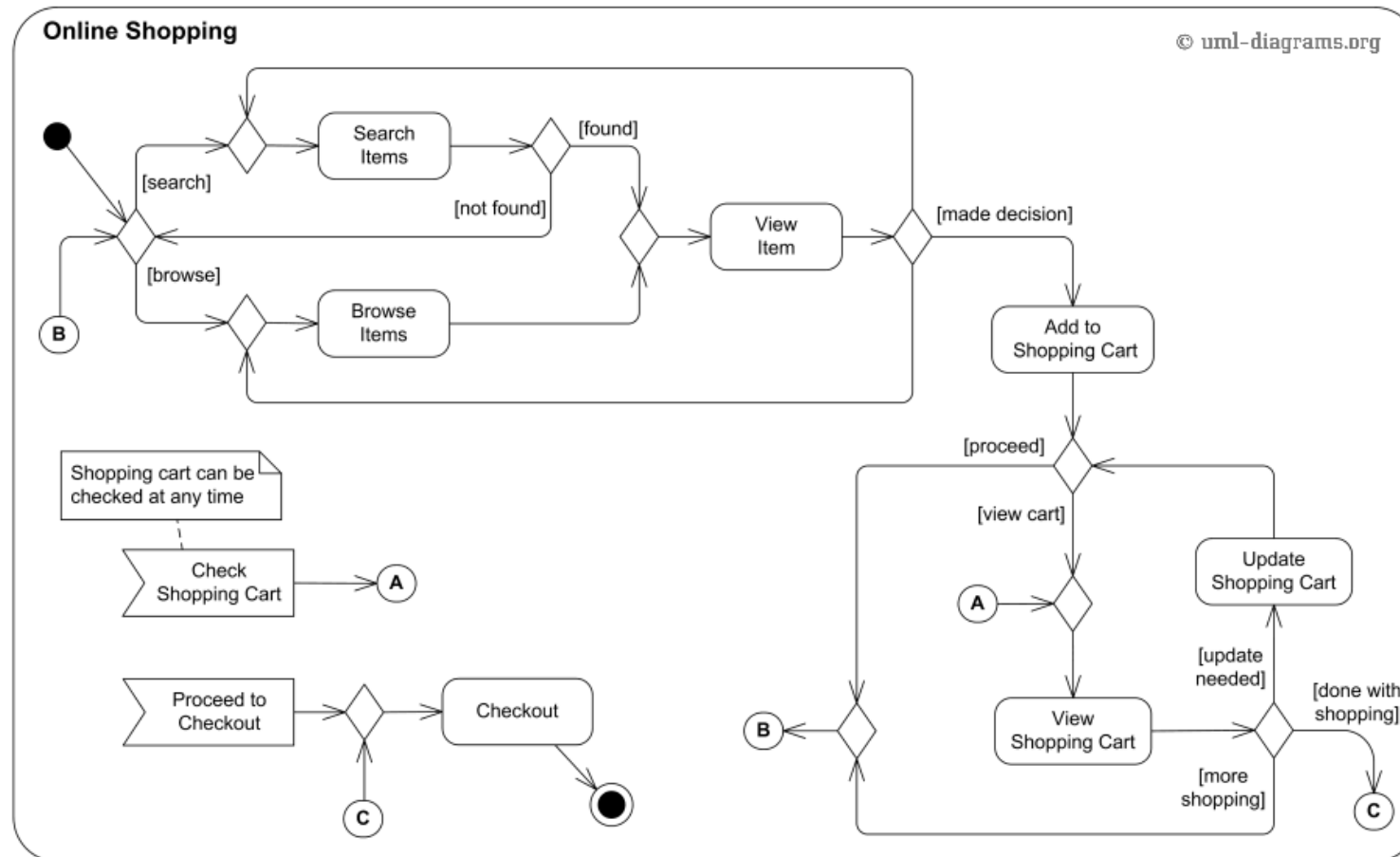
Relationships:

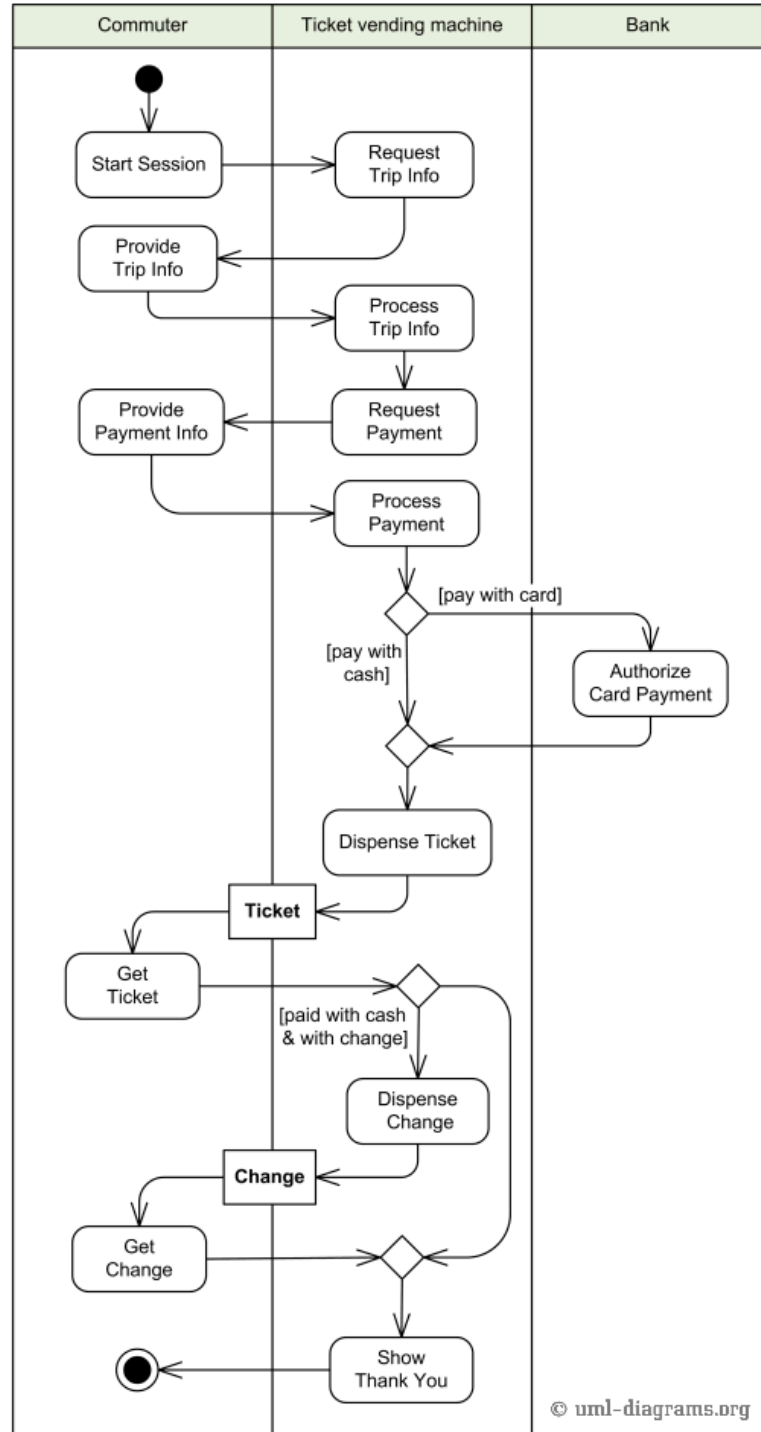
- "Checkout" —include—▶ "Login" (ต้อง login ก่อน)
- "Checkout" —include—▶ "Payment Processing"
- "Track Order" —include—▶ "View Order Details"
- "Express Checkout" —extend—▶ "Checkout"



Activity Diagram

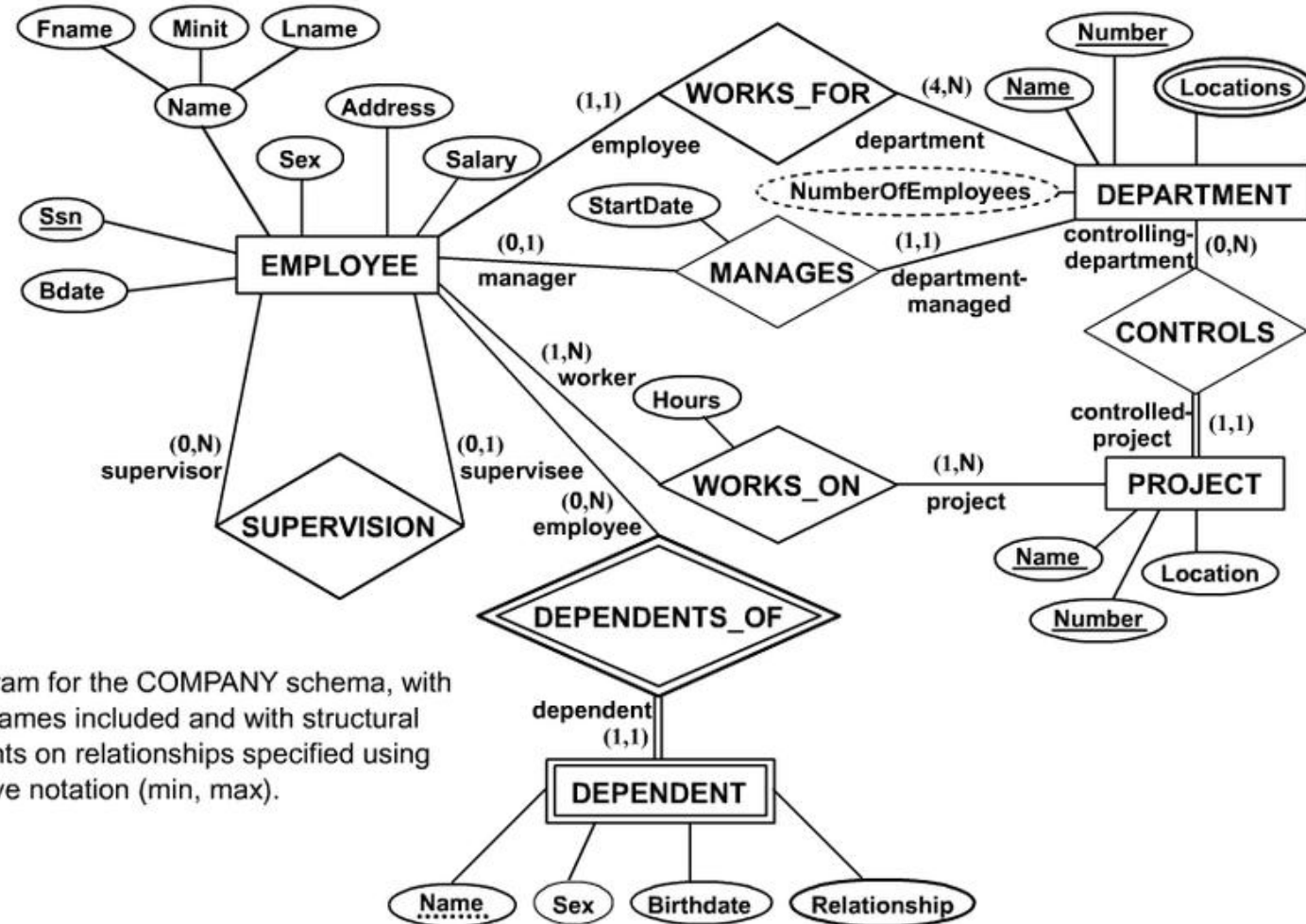
- **นิยาม:** Activity Diagram แสดงการไหลของกิจกรรม (Flow) และเงื่อนไข (Conditions) ในกระบวนการ





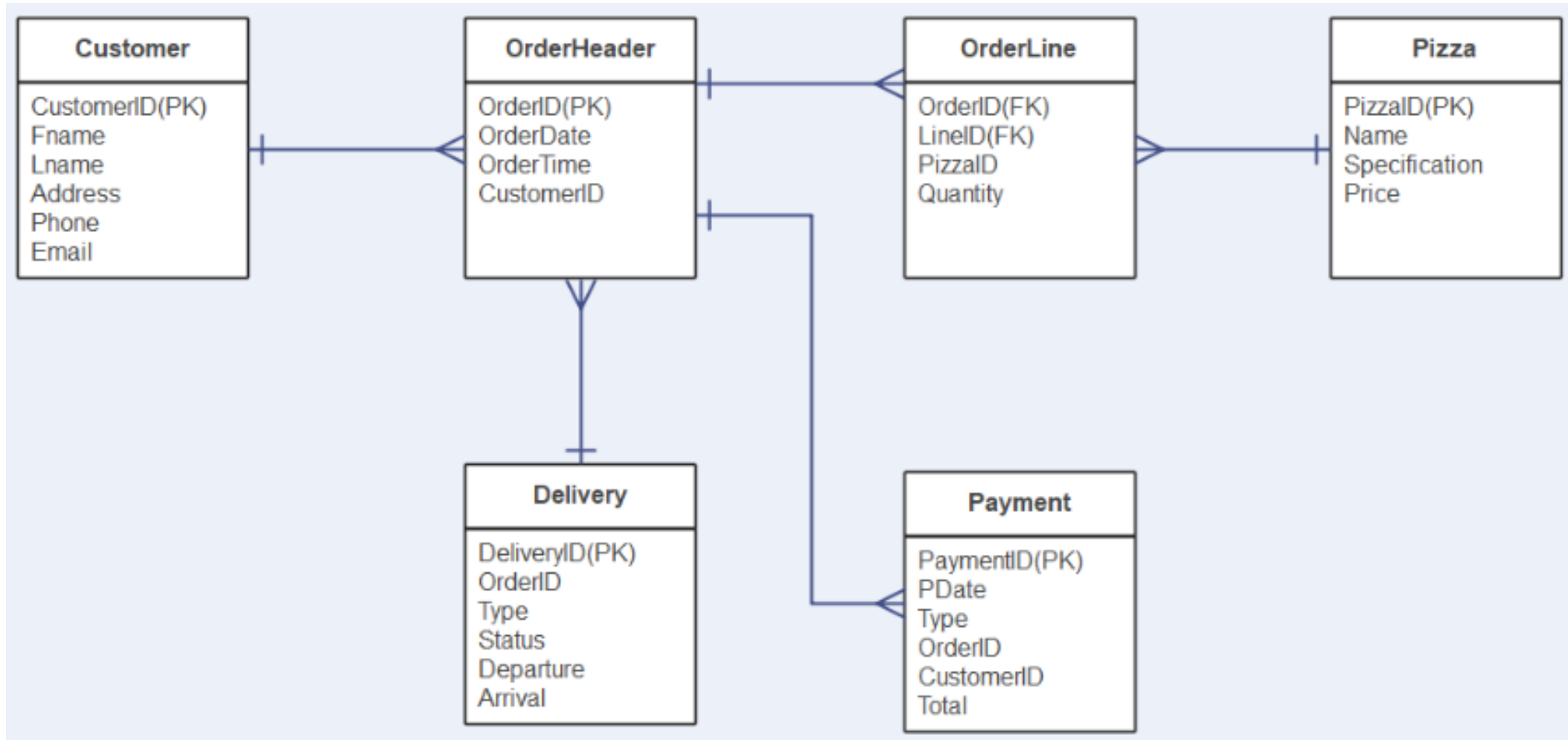
Entity-Relationship (ER) Diagram

- **นิยาม:** ER Diagram ไดอะแกรม แสดงเอนทิตี (entities) และ ความสัมพันธ์ (relationships) ของเอนทิตี



ER diagram for the COMPANY schema, with all role names included and with structural constraints on relationships specified using alternative notation (min, max).

ERD using Crow's Foot notation



Software Requirements Specification Document

1. บทนำ (Introduction)
 - 1.1 วัตถุประสงค์ (Purpose)
 - 1.2 ขอบเขต (Scope)
 - 1.3 ภาพรวมเอกสาร (Document Overview)
 - 1.4 เอกสารอ้างอิง (References)
 - 1.5 ข้อตกลงเอกสาร (Document Conventions)
2. คำอธิบายโดยรวม (Overall Description)
 - 2.1 มุมมองผลิตภัณฑ์ (Product Perspective)
 - 2.2 ฟังก์ชันผลิตภัณฑ์ (Product Functions) (ภาพรวมฟีเจอร์ feature overview)
 - 2.3 ประเภทและลักษณะผู้ใช้ (User Classes and Characteristics)
 - 2.4 สภาพแวดล้อมการทำงาน (Operating Environment)
 - 2.5 ข้อจำกัดในการออกแบบและการใช้งาน (Design and Implementation Constraints)
 - 2.6 สมมติฐานและการพึ่งพา (Assumptions and Dependencies)
3. ความต้องการเฉพาะ (Specific Requirements)
 - 3.1 ความต้องการเชิงหน้าที่ (Functional Requirements) (REQ-001 ถึง REQ-N)
 - 3.2 ความต้องการที่ไม่ใช่เชิงหน้าที่ (Non-Functional Requirements)
 - 3.2.1 ความต้องการด้านประสิทธิภาพ (Performance Requirements)
 - 3.2.2 ความต้องการด้านความปลอดภัย (Security Requirements)
 - 3.2.3 ความพร้อมใช้งาน/ความน่าเชื่อถือ (Availability/Reliability)
 - 3.2.4 ความต้องการด้านการใช้งาน (Usability Requirements)
 - 3.2.5 ความต้องการด้านความสามารถในการขยาย (Scalability Requirements)

ตย. SRS

Software Requirements Specification Document

4. ความต้องการอินเทอร์เฟซภายนอก (External Interface Requirements)
 - 4.1 อินเทอร์เฟซผู้ใช้ (User Interfaces)
 - 4.2 อินเทอร์เฟซฮาร์ดแวร์ (Hardware Interfaces)
 - 4.3 อินเทอร์เฟซซอฟต์แวร์ (Software Interfaces)
 - 4.4 อินเทอร์เฟซการสื่อสาร (Communication Interfaces)
5. ฟีเจอร์ระบบ (System Features)
 - 5.1 ฟีเจอร์ที่ 1: การยืนยันตัวตน (Authentication)
 - 5.2 ฟีเจอร์ที่ 2: การจัดการสินค้า (Product Management)
 - 5.3 ฟีเจอร์ที่ 3: การจัดการคำสั่งซื้อ (Order Management)
6. ความต้องการอื่นๆ ที่ไม่ใช่เชิงหน้าที่ (Other Nonfunctional Requirements)
 - 6.1 กฎทางธุรกิจ (Business Rules)
 - 6.2 ความต้องการด้านข้อมูล (Data Requirements)
 - 6.3 กฎหมาย/การปฏิบัติตามกฎระเบียบ (Legal/Compliance)
7. ดัชนี (Index)

ตย. SRS

แม่แบบสำหรับความต้องการเชิงหน้าที่ (Template for Functional Requirement)

REQ-001: การยืนยันตัวตนผู้ใช้ (User Authentication)

คำอธิบาย (Description):

ระบบจะต้องจัดให้มีกลไกการยืนยันตัวตนผู้ใช้ที่ปลอดภัย เพื่อตรวจสอบตัวตนของผู้ใช้ก่อนเข้าใช้งานระบบ

เหตุผล (Rationale):

เพื่อให้แน่ใจว่ามีเพียงผู้ใช้ที่ได้รับอนุญาตเท่านั้นที่สามารถเข้าใช้งานระบบ และปกป้องข้อมูลที่ละเอียดอ่อน

เกณฑ์การยอมรับ (Acceptance Criteria):

- ผู้ใช้สามารถเข้าสู่ระบบด้วยอีเมลและรหัสผ่าน
- รหัสผ่านต้องมีอย่างน้อย 8 ตัวอักษร
- เซสชันหมดอายุหลังจากไม่ใช้งาน 30 นาที
- บัญชีถูกล็อกหลังจากพยายามเข้าสู่ระบบผิดพลาด 5 ครั้ง
- การเข้ารหัสรหัสผ่านโดยใช้ bcrypt

เรื่องราวผู้ใช้ที่เกี่ยวข้อง (Related User Stories):

- US-001: ผู้ใช้สามารถเข้าสู่ระบบด้วยอีเมล/รหัสผ่าน
- US-002: ผู้ใช้ลืมรหัสผ่าน

ลำดับความสำคัญ (Priority): สูง (HIGH)

สถานะ (Status): อนุมัติแล้ว (Approved)

ผู้รับผิดชอบ (Owner): หัวหน้านักพัฒนา (Lead Developer)

กิจกรรมที่ 1: สร้างแผนภาพ Use Case Diagram

- **วัตถุประสงค์:** ทักษะวาดแผนภาพ Use Case สำหรับโครงการของตัวเอง
 - **ส่วนที่ 1: ระบุตัวแสดง (Actors)**
 - **ส่วนที่ 2: ระบุ Use Cases**
 - **ส่วนที่ 3: ระบุความสัมพันธ์ (Relationships)**
 - **ส่วนที่ 4: วาดแผนภาพ (Diagram)**

กิจกรรมที่ 2: ออกแบบแผนภาพ ER (Design ER Diagram)

- **วัตถุประสงค์:** ทึมออกแบบโครงสร้างฐานข้อมูล (Database Schema) ด้วยแผนภาพ ER
 - **ส่วนที่ 1: ระบุเอนทิตี (Entities)**
 - **ส่วนที่ 2: ระบุแอตทริบิวต์ (Attributes)**
 - **ส่วนที่ 3: ระบุความสัมพันธ์ (Relationships)**

การส่งงาน D1

- รายการตรวจสอบ (Checklist) ต้องส่ง → [ที่นี่](#)