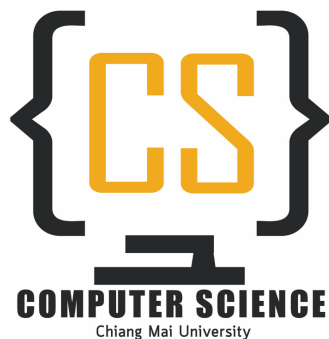


CS204362 – Object-Oriented Design

L9: Use Case Workshop

Kamonphop Srisopha



Faculty of Science, Chiang Mai University

คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่

Use Cases

Use Cases

- บ่งบอกถึง ฟังก์ชันการทำงานของระบบในมุมมองของผู้ใช้ หรือ บอกว่าผู้จะใช้งานกับระบบอย่างไร
- Use Case มีลำดับการดำเนินการของกิจกรรม (sequence of actions) ที่ผู้ใช้งานต้องทำให้บรรลุเป้าหมายอย่างหนึ่ง
- ฉะนั้นเป้าหมายของการทำ use case analysis คือการโมเดลระบบ
 - ในมุมมองว่าผู้ใช้งานมีปฏิสัมพันธ์กับระบบอย่างไร เพื่อให้บรรลุเป้าหมาย (top-level functionalities ที่ใช้งานโดยผู้ใช้งานโดยตรง)
- Use case models ประกอบด้วย
 1. A set of use cases
 2. A use case specification (คำอธิบาย use case ในข้อ 1 โดยละเอียด)

UML Notations

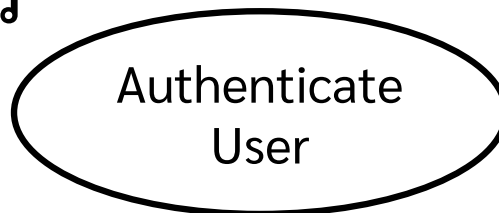
Use case

เป้าหมายหรือจุดประสงค์ที่ต้องการให้บรรลุ หรือ ลำดับกระบวนการดำเนินการเพื่อให้บรรลุเป้าหมาย

โดยส่วนใหญ่ จะเริ่มต้นด้วย**คำกริยาแล้วตามด้วยคำนาม** เช่น

- Process Sale
- Add customer

ใช้สัญลักษณ์รูปวงรี

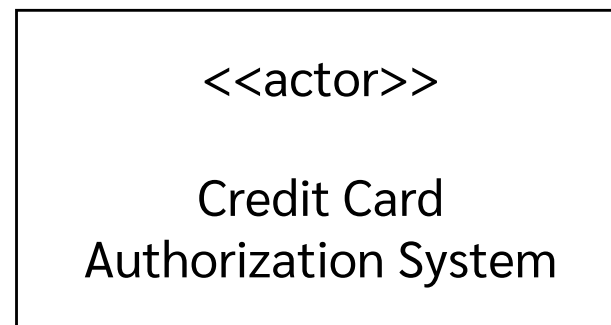


Notations

Actor

ผู้ที่มีปฏิสัมพันธ์กับระบบและกับ use case หนึ่งๆที่เป็นมนุษย์ จะใช้สัญลักษณ์รูปคน (stick figure) และที่ไม่ใช่มนุษย์ (เช่น ระบบที่อยู่ **นอกเหนือจากระบบ**ที่เราสนใจ) จะใช้สัญลักษณ์รูปสี่เหลี่ยมแล้วเขียนคำว่า “<<actor>>” มีชื่อของ actor กำกับด้วย

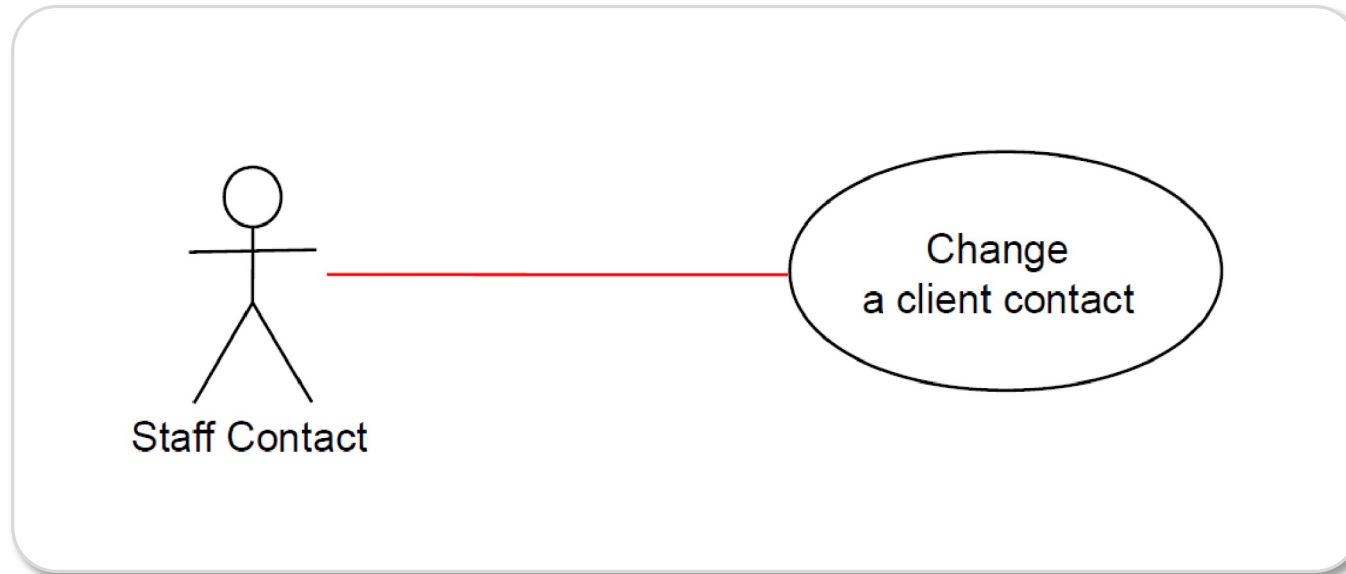
ไม่ควรเลือกชื่อของ actor ตามตำแหน่งงานเช่น ถ้า พนักงานตำแหน่ง 1, 2, และ 3 สามารถใช้งานกับระบบได้เหมือนกันก็ให้เลือกชื่อ actor ที่รวมทั้ง 3 เข้าด้วยกัน ไม่ใช่มี 3 actors.



Notations

Association (Connector)

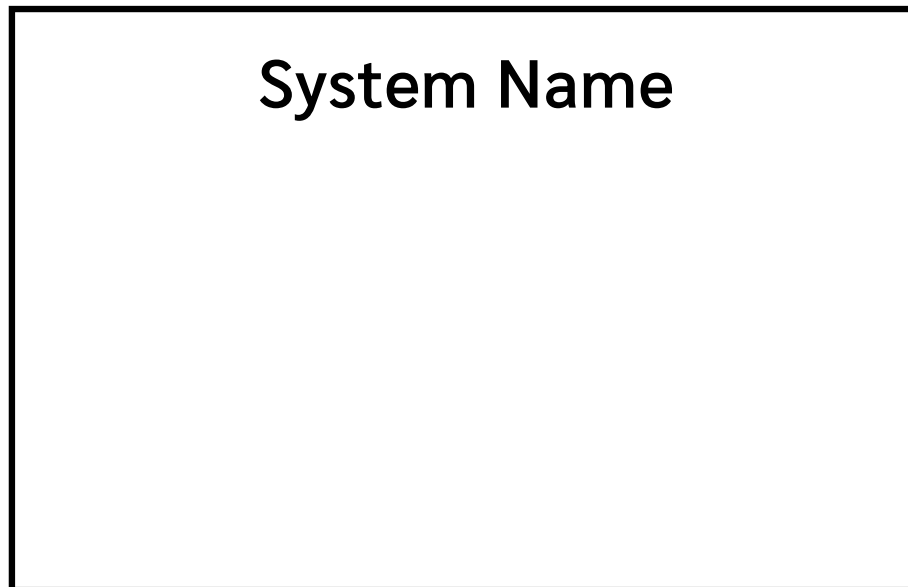
เส้นแสดงความสัมพันธ์ระหว่าง actor และ use case
เป็นเส้นตรงไม่มีหัวลูกศร



Notations

System Boundary

เส้นแบ่งขอบเขตของระบบ ใช้แสดงให้เห็นขอบเขต
ระหว่างระบบกับผู้กระทำต่อระบบ (ระหว่าง actor กับ use case)
ใช้สัญลักษณ์รูปสี่เหลี่ยม พร้อมเขียนชื่อระบบไว้ด้านใน



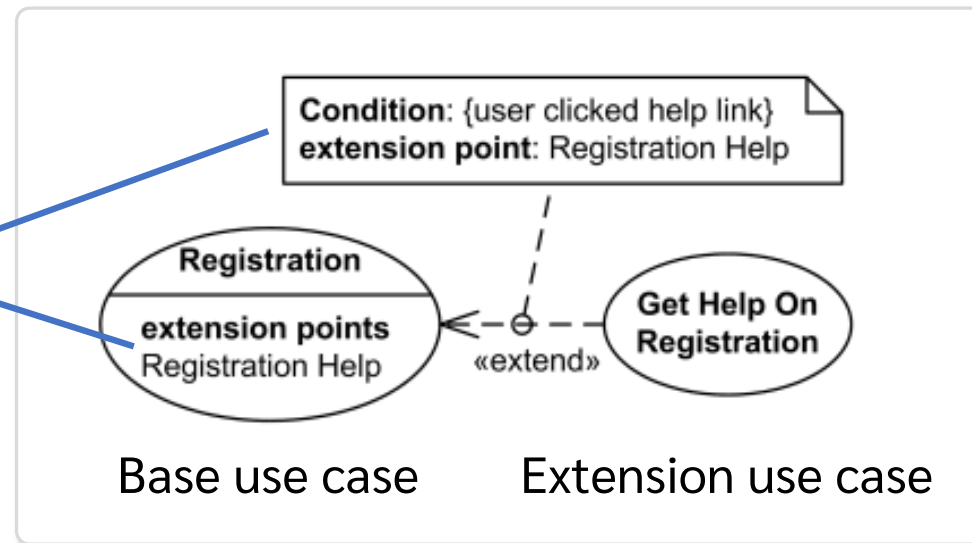
Notations

Extend Relationship (*different scenarios*)

ความสัมพันธ์แบบขยายหรือเพิ่มระหว่าง use cases เช่น กรณีที่ use case หนึ่งๆ ต้องทำบางอย่างที่ต่างออกไปจากสิ่งที่ผิปรกติหรือเพื่อตอบสนองต่อเงื่อนไขบางอย่าง เราจะสามารถเขียน เงื่อนไขหรือสิ่งที่ทำต่างเป็นอีก use case หนึ่งได้ โดยให้มันมีความสัมพันธ์ที่เรียกว่า extend relationship กับ use case หลัก

เรียก use case หลักว่า “base use case” และเรียก use case ที่เป็นส่วนเพิ่มว่า “extension use case”

Extension points
แสดงให้เห็นว่า
extension use
case จะถูกใช้
งานในกรณีไหน



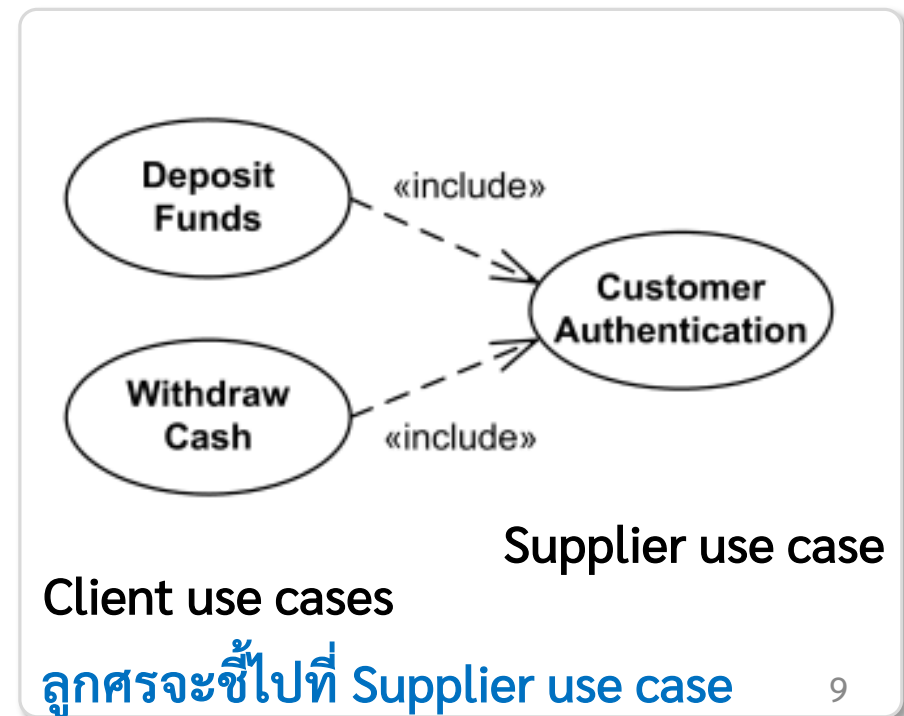
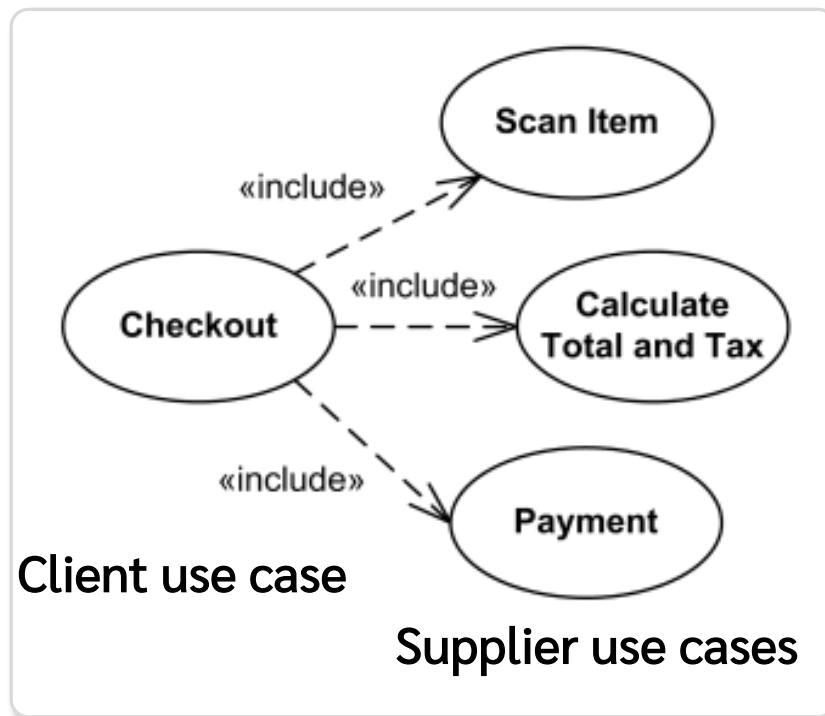
ลูกศรจะชี้ไป
ที่ base case

Notations

Include Relationship (*reuse*)

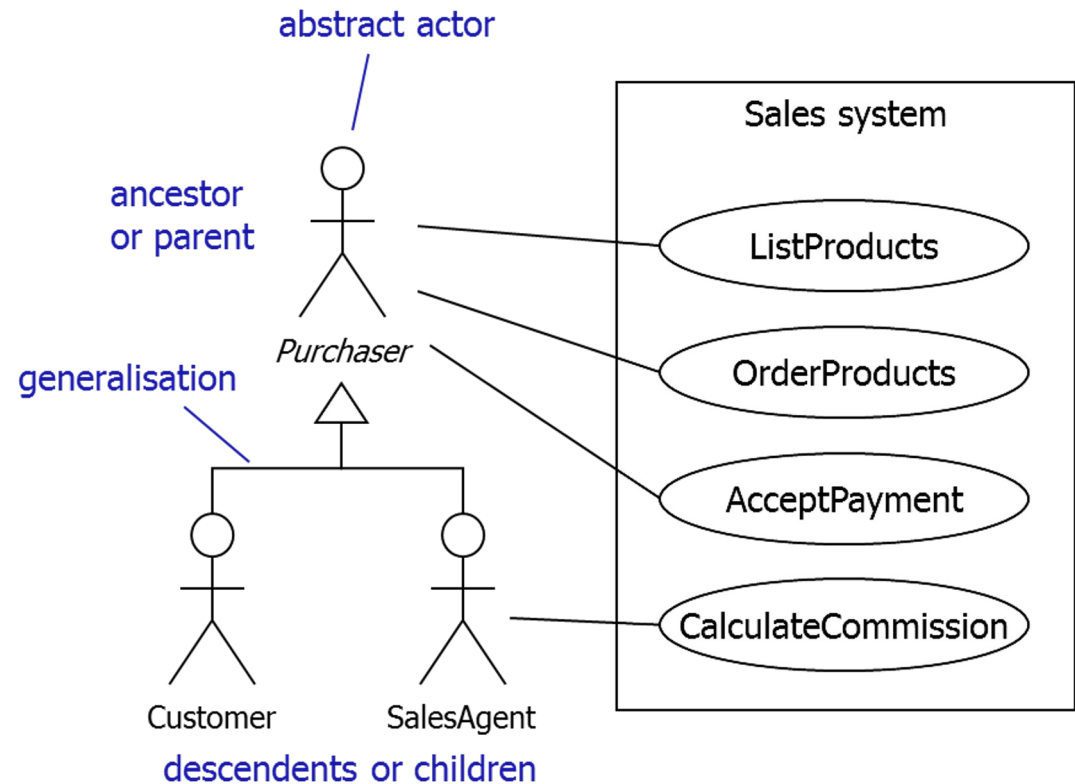
แสดงความสัมพันธ์แบบเรียกใช้ ในกรณีที่ use case หนึ่งต้องไปเรียกขั้นตอนการดำเนินงานของอีก use case หนึ่งมาใช้ **เสมอ** เพื่อให้ use case ของตัวเองบรรลุเป้าหมายได้

สนับสนุนการนำกลับมาใช้ใหม่ของ use case (reuse)





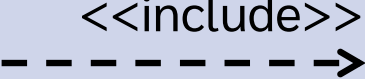

Notations: Actor Generalization

- If two actors communicate with the same set of use cases in the same way, then we can express this as a generalisation to another (possibly abstract) actor
- The **child** actors inherit the roles and relationships to use cases held by the **parent** actor
- We can substitute a child actor anywhere the parent actor is expected. This is the *substitutability principle*



Use actor generalization when it simplifies the model

Summary of Use Case Relationships

Relationship	Function	Notation
Association	The communication path between an actor and a use case it participates in	
Extend	The insertion of additional behavior into a base use case that does know about it	
Include	The insertion of additional behavior into a base case that explicitly describes the insertion	
Generalization	A relationship between a general use case and a more specific use case that inherits and adds features to it	

Common Mistakes in UML Diagrams

#	Mistakes
1	A use case describing activity outside the system
2	The direction of the <<include>> relationship
3	The direction of the <<extend>> relationship
4	<<include>> is used incorrectly among use cases
5	Using the System actor to represent the modeled system
6	Abuse the use of <<include>> where use case generalization should be used
7	Inappropriate use of inheritance between actors
8	Inheriting unwanted use case through generalization

Additional Mistakes in UML Diagrams

#	Mistakes
9*	Using User Interface actions as use cases
10*	Using system modules as use cases
11*	Showing too many details in the use case model
12*	Using the use case model to show a business process
13	Containing standalone use cases
14	Not iteratively refining use cases

Steps To Create a Use Case Diagram

1. ค้นหา actor
2. ค้นหา base use cases ที่มีปฏิสัมพันธ์กับ actor โดยตรง
3. สร้างความสัมพันธ์ระหว่าง 1 และ 2
4. เพิ่ม use case ใหม่ อาจจะเป็น extension use case, supplier use case, หรือ base use case ที่ไม่ได้มีใน 2
5. Actor ต้องมีปฏิสัมพันธ์กับอย่างน้อย 1 use case
6. ทุก Use case ต้องมีปฏิสัมพันธ์บางอย่างกับ use case อื่นๆ หรือ actor ต่างๆ
7. ถ้ามันจะทำให้ use case diagram เข้าใจง่ายขึ้น อาจมีการจัดเรียงลำดับชั้น generalization ของ use case และ actor ต่างๆ
8. เขียน description/specification (คำอธิบาย) ของแต่ละ use case

Check Your Understanding

ให้นักศึกษาลองวาด use case diagram จากข้อมูลดังต่อไปนี้

**ระบบจองตั๋วและ check-in ผู้โดยสารที่ใช้โดยพนักงานประจำ
เคาท์เตอร์ของสายการบิน CS CMU.**

พนักงานสามารถจอง, ยกเลิกการจอง, และดำเนินการ check-in ผู้โดยสาร
ได้ ในตอนทำการ check-in พนักงานจะเลือกที่นั่งให้ผู้โดยสาร โดยสายการ
บินนี้ผู้โดยสารไม่สามารถเลือกที่นั่งได้เอง เครื่องบินทุกลำของสายการบินนี้
จะมีที่นั่งสองแบบ แบบติดทางเดินและแบบติดเครื่องบิน ในการ check-in
พนักงานก็จะชั่งกระเป๋เดินทางของผู้โดยสารด้วยว่าเกินหรือไม่หากเกินกว่า
น้ำหนักกำหนดผู้โดยสารก็ต้องจ่ายค่าปรับ ส่วนในการจองและการยกเลิก
การจองระบบนั้นทำงานไม่ต่างกับระบบทั่วไปที่ทำในส่วนนี้

Workshop Assignment

- จับกลุ่ม กลุ่มละ 1 หรือ 2 คน จัดทำ Use Case Diagrams ของโปรเจค SE ของนักศึกษา
- **Constraints:**
 - ให้นักศึกษาใช้ Visual Paradigm ในการทำ Use Case Diagram
 - สามารถเขียนคำเป็นภาษาไทยหรือภาษาอังกฤษก็ได้
 - ให้เขียน use case diagram โดยมี use case ที่เป็น <<include>> อย่างน้อยหนึ่ง use case และ use case ที่เป็น <<extend>> อย่างน้อยหนึ่ง use case
- คะแนนที่จะได้รับจะขึ้นอยู่กับความถูกต้องของ diagram และรายละเอียด
- กำหนดส่งภายในวันที่ 28 ธันวาคม 2566, 23:59