



Diagramas de Casos de Uso

Índice

- ❑ 1. Introducción
- ❑ 2. Descripción de casos de uso
- ❑ 3. Relaciones
- ❑ 4. Recomendaciones
- ❑ 5. Ejercicios

1. Introducción

- ❑ *Casos de Uso* es una técnica para capturar información de cómo un sistema o negocio trabaja actualmente, o de cómo se desea que trabaje
- ❑ No pertenece realmente al enfoque orientado a objeto, más bien es una técnica para el modelado de escenarios en los cuales el sistema debe operar
- ❑ Ejemplo: Análisis de cómo una familia utilizará su casa
 - ❖ Casos de uso: “preparar comidas”, “limpiar”, “dormir”, ...

Introducción

- ❑ Un caso de uso especifica el comportamiento de un sistema o de una parte del mismo
 - ❖ Sin especificar cómo se implementa dicho comportamiento.
 - ❖ Ejemplos: “Extraer dinero” (de un cajero automático); “Procesar préstamo” (en un banco)
- ❑ Especifican el comportamiento que se desea pero tal y como se verá desde fuera
- ❑ No imponen cómo se lleva a cabo ese comportamiento
- ❑ Proporcionan un medio para que desarrolladores, usuarios finales del sistema y expertos del dominio lleguen a una comprensión común del sistema sin quedarse atascados en los detalles
- ❑ Se centran en las cuestiones más importantes para el usuario final

Casos de Uso

□ Definición:

- ❖ Descripción de una secuencia de acciones, incluyendo variantes, que ejecuta un sistema para producir un resultado observable de valor para un actor (cálculo de un resultado, creación de un objeto, cambio de estado de un objeto)

□ Un caso de uso abstrae un conjunto de escenarios

- ❖ Escenario: una situación de uso del sistema interactuando con actores
- ❖ Escenario = instancia de un caso de uso

□ Un Caso de Uso tiene un nombre (normalmente una expresión con un verbo en infinitivo)

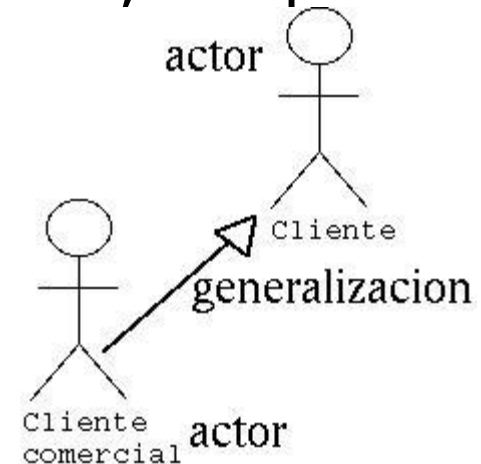
□ Se representa gráficamente en un diagrama de casos de uso con una elipse

Casos de Uso

- Normalmente, un caso de uso se corresponde con un requisito funcional del sistema, pero ...
 - ❖ Varios requisitos funcionales pueden estar representados por un mismo caso de uso
 - ❖ La descripción del caso de uso puede incluir requisitos no funcionales
 - ❖ Un caso de uso también puede representar un requisito no funcional. Ejemplo: *ValidarUsuario* (requisito de seguridad)
- Además, cada Caso de Uso se describe mediante:
 - ❖ texto
 - condiciones pre-post para que el escenario comience o termine
 - secuencia de pasos ejecutados dentro del escenario
 - (combinaciones de los elementos anteriores)

Actores

- ❑ Un *actor* es un agente, alguien o algo que solicita un servicio al sistema o actúa como catalizador para que ocurra algo.
- ❑ Un actor representa un rol que es jugado por una persona, un dispositivo *hardware*, incluso otro sistema
- ❑ Los actores no forman parte del sistema, aunque aparecen en los modelos
- ❑ Se pueden definir categorías generales de actores (ej: Cliente) y especializarlos (ej: Cliente comercial) con la relación de generalización.



2. Descripción de casos de uso

- ❑ El comportamiento del caso de uso se puede especificar textualmente con un flujo de eventos (secuencia de pasos)
- ❑ El flujo de eventos lo debería entender alguien ajeno al sistema
- ❑ Descripción de un flujo de eventos:
 - ❖ *Cómo y cuándo* empieza y acaba el caso de uso
 - ❖ Cuándo interactúa con los actores y qué objetos intercambia con ellos
- ❑ Cada caso de uso tiene un flujo de eventos básico (principal) y, opcionalmente, uno o varios flujos alternativos

Ejemplo: Caso de uso “*Validar Usuario*” en un Cajero Automático

❑ Flujo básico:

1. El caso de uso comienza cuando el sistema pide al cliente el PIN.
2. El cliente introduce el PIN y acepta la entrada pulsando el botón *Enter*.
3. El sistema comprueba si el PIN es válido. Si es válido, el sistema acepta la entrada y acaba el caso de uso.

❑ Flujo alternativo:

- ❖ En el paso 2 del flujo básico el cliente puede borrar un PIN en cualquier momento antes de introducirlo, y volver a teclear un nuevo PIN.

❑ Flujo alternativo:

- ❖ En el paso 3 del flujo básico si el cliente introduce un PIN inválido, el caso de uso vuelve a empezar. Si ocurre tres veces en una sesión, se cancela la transacción completa, lo que impide que el cliente utilice el cajero durante 60 segundos.

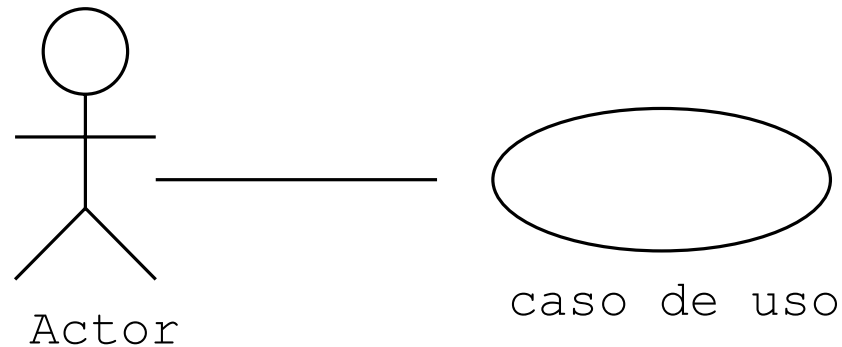
Especificación de flujos de eventos mediante diagramas de secuencia

- ❑ A partir de la descripción textual del flujo de eventos, el caso de uso se podría especificar más formalmente (gráficamente) con diagramas de secuencia (**más adelante se presenta técnica**)
- ❑ Un diagrama para el flujo principal y variaciones para los alternativos
- ❑ Conviene separarlos porque un caso de uso describe un conjunto de secuencias, no una única secuencia
 - ❖ Sería imposible expresar todos los detalles de un caso de uso en una única secuencia.
- ❑ Ejemplo: Contratar empleado puede tener muchas variantes (trasladar a una persona, contratar a un extranjero, ...)

3. Relaciones

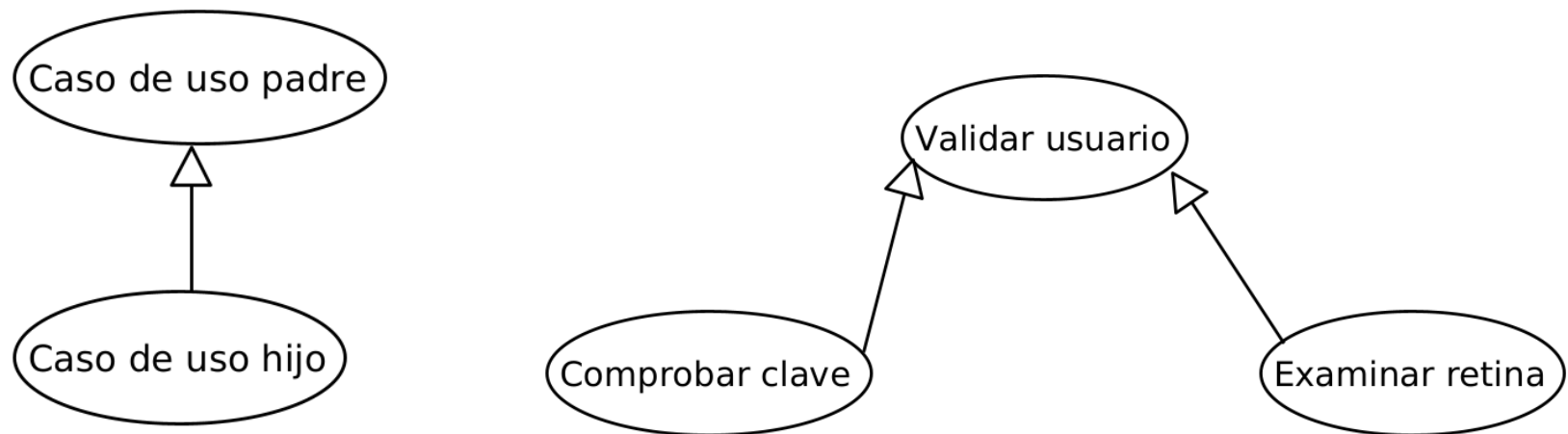
□ UML define cuatro tipos de relación en los Casos de Uso:

❖ De Comunicación:



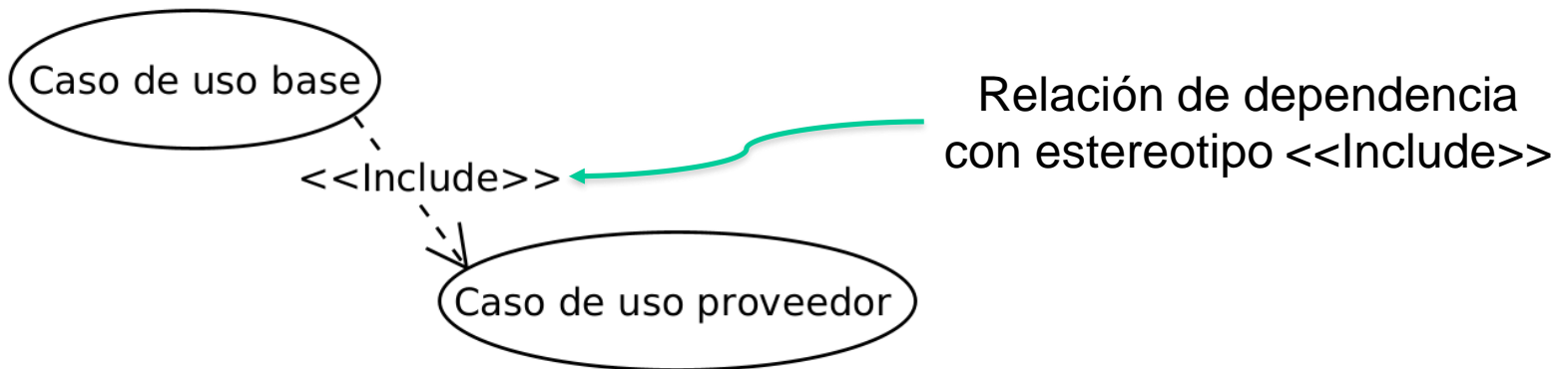
Casos de Uso: Relaciones

- ❖ De **Generalización**: El caso de uso hijo hereda la especificación del caso de uso padre.
 - El hijo puede añadir o redefinir el comportamiento del padre.
 - El hijo puede ser colocado en cualquier lugar donde aparezca el padre.



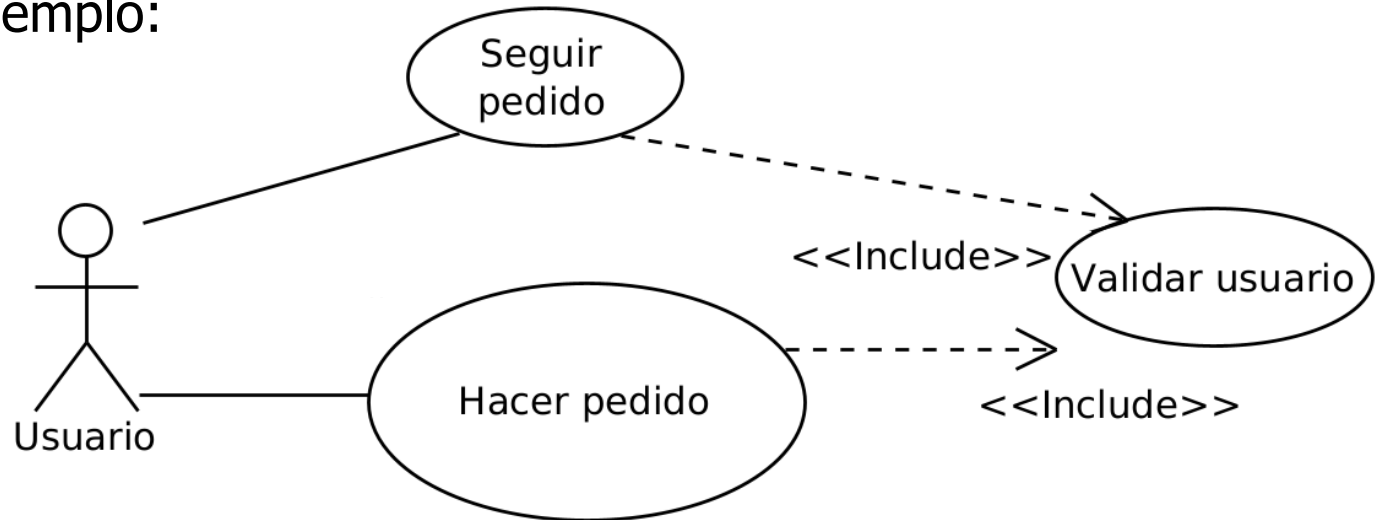
Casos de Uso: Relaciones

- ❖ De **Inclusión**: Un caso de uso base incorpora explícitamente el comportamiento de otro caso de uso (proveedor) en el lugar especificado en el caso base.
 - Se usa para evitar describir el mismo flujo de eventos repetidas veces. Este comportamiento se pone en un caso de uso aparte (que será incluido por un caso de uso base).



Casos de Uso: Relaciones

➤ Ejemplo:



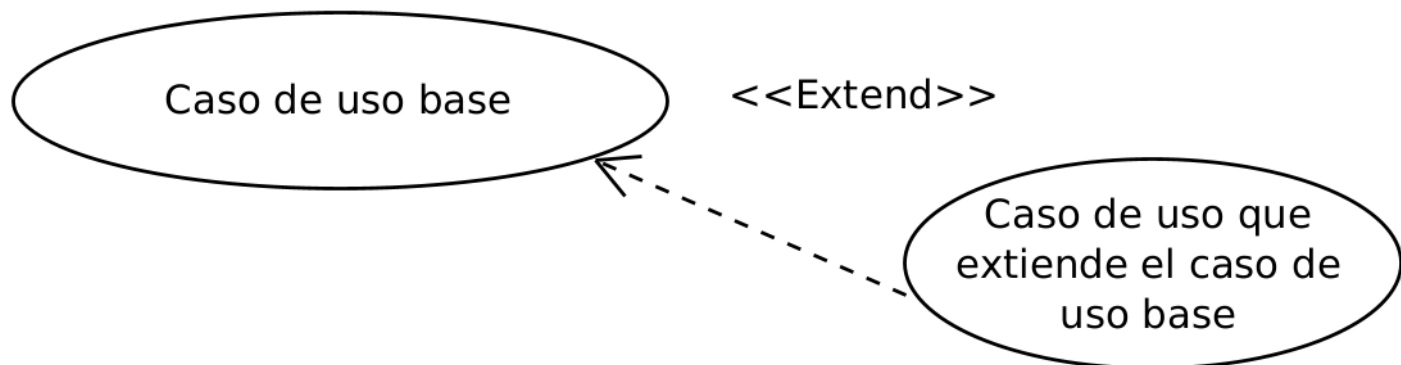
○ Descripción de Caso de Uso "*Seguir pedido*"

✓ Flujo básico:

1. Obtener y verificar el número de pedido
2. *include* (*Validar usuario*)
3. Examinar el estado de cada parte del pedido y preparar un informe para el usuario

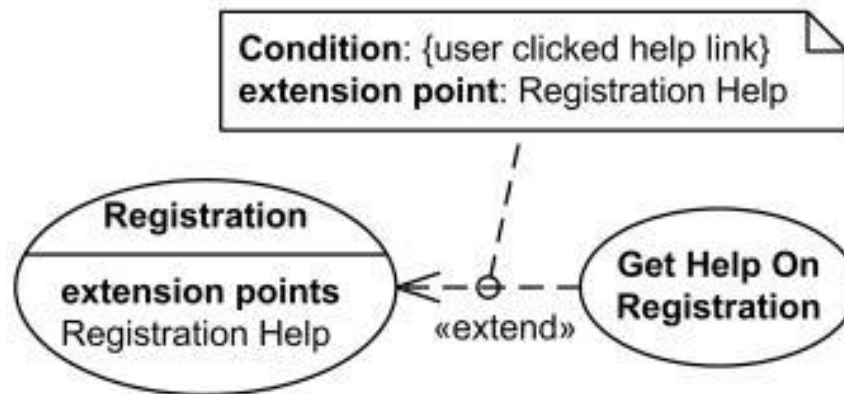
Casos de Uso: Relaciones

- ❖ De **Extensión**: El comportamiento de un caso de uso base puede ser extendido opcionalmente con un comportamiento adicional.
 - Se utiliza para modelar la parte de un caso de uso que el usuario puede ver como comportamiento opcional del sistema.
 - Separa el comportamiento opcional del obligatorio.
 - También se usa para modelar un subflujo que se ejecuta bajo ciertas condiciones.



Casos de Uso: Relaciones

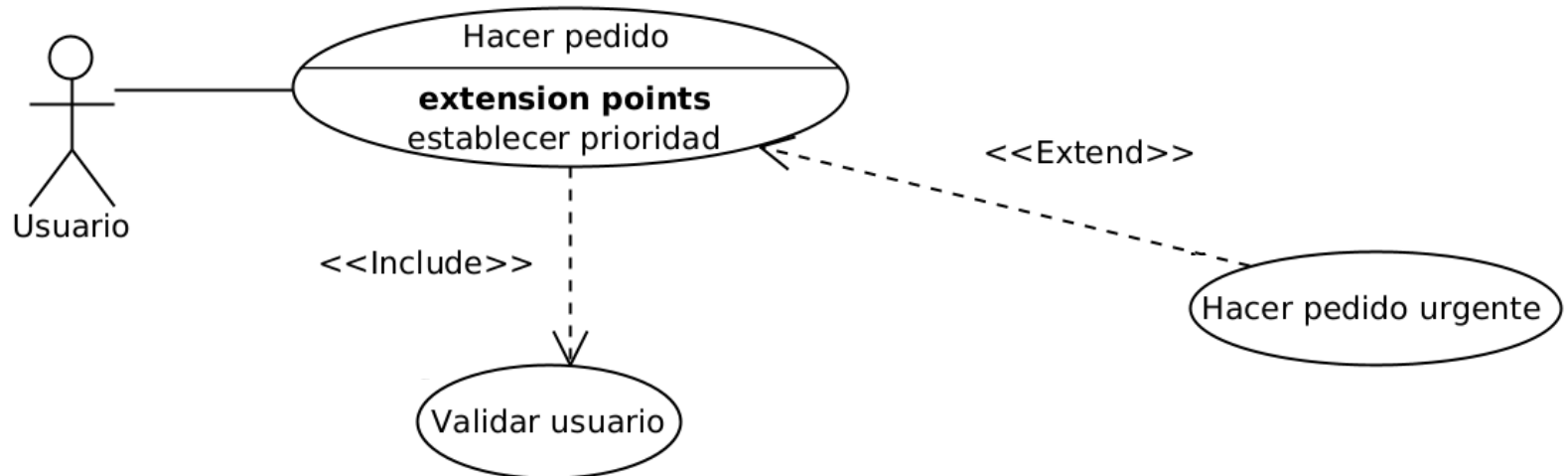
- Se puede añadir un compartimento en la elipse del caso de uso para indicar posibles puntos de extensión (puntos del caso de uso base donde se integra la extensión)
- También se puede añadir una nota para explicar las condiciones bajo las cuales un caso de uso extiende a otro en un punto de extensión



<https://www.uml-diagrams.org/use-case-extend.html>

Casos de Uso: Relaciones

➤ Ejemplo:



○ Descripción de Caso de Uso “*Hacer pedido*”

✓ Flujo básico:

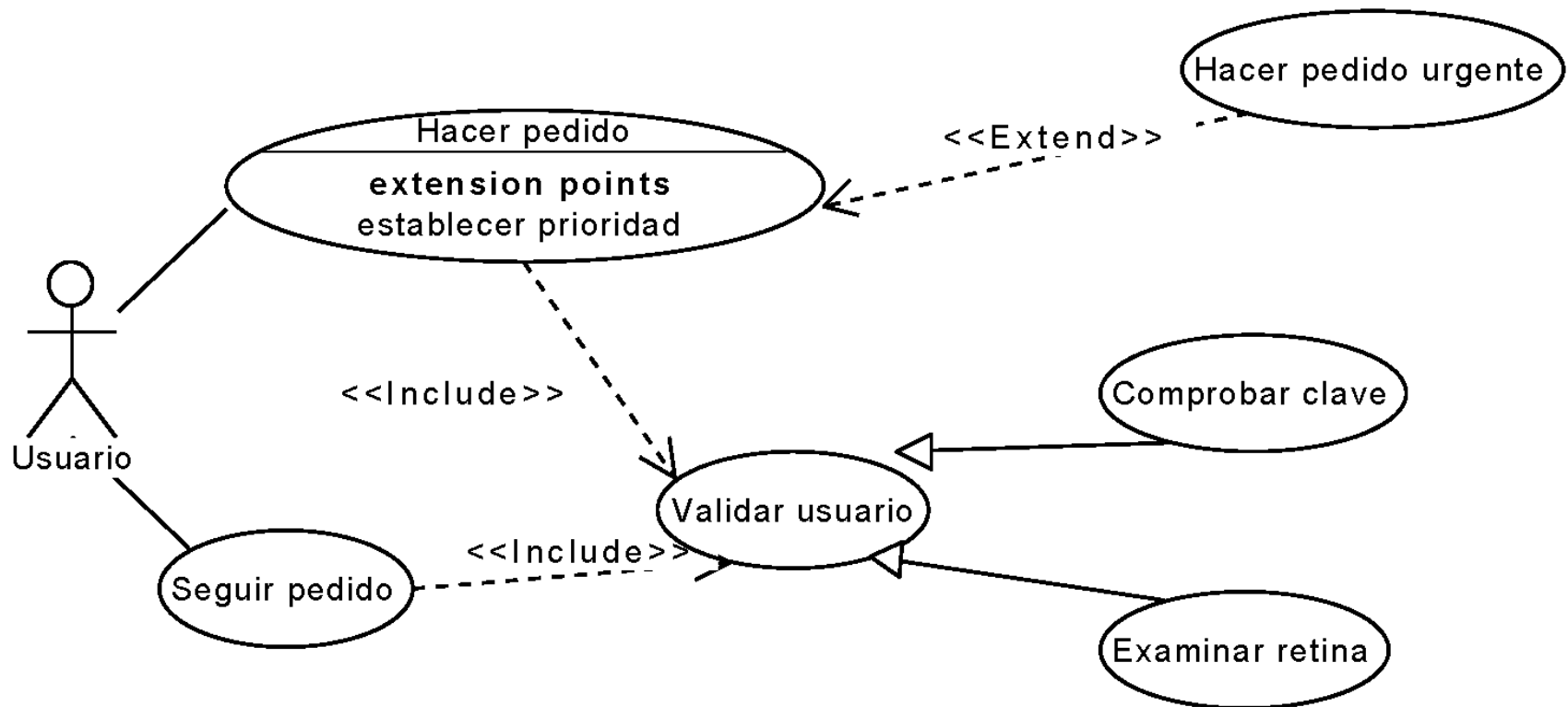
1. *include* (*Validar usuario*)
2. Recoger los items del pedido del usuario
3. *Extension points*:
establecer prioridad
4. Enviar el pedido para ser procesado

Casos de Uso: Relaciones

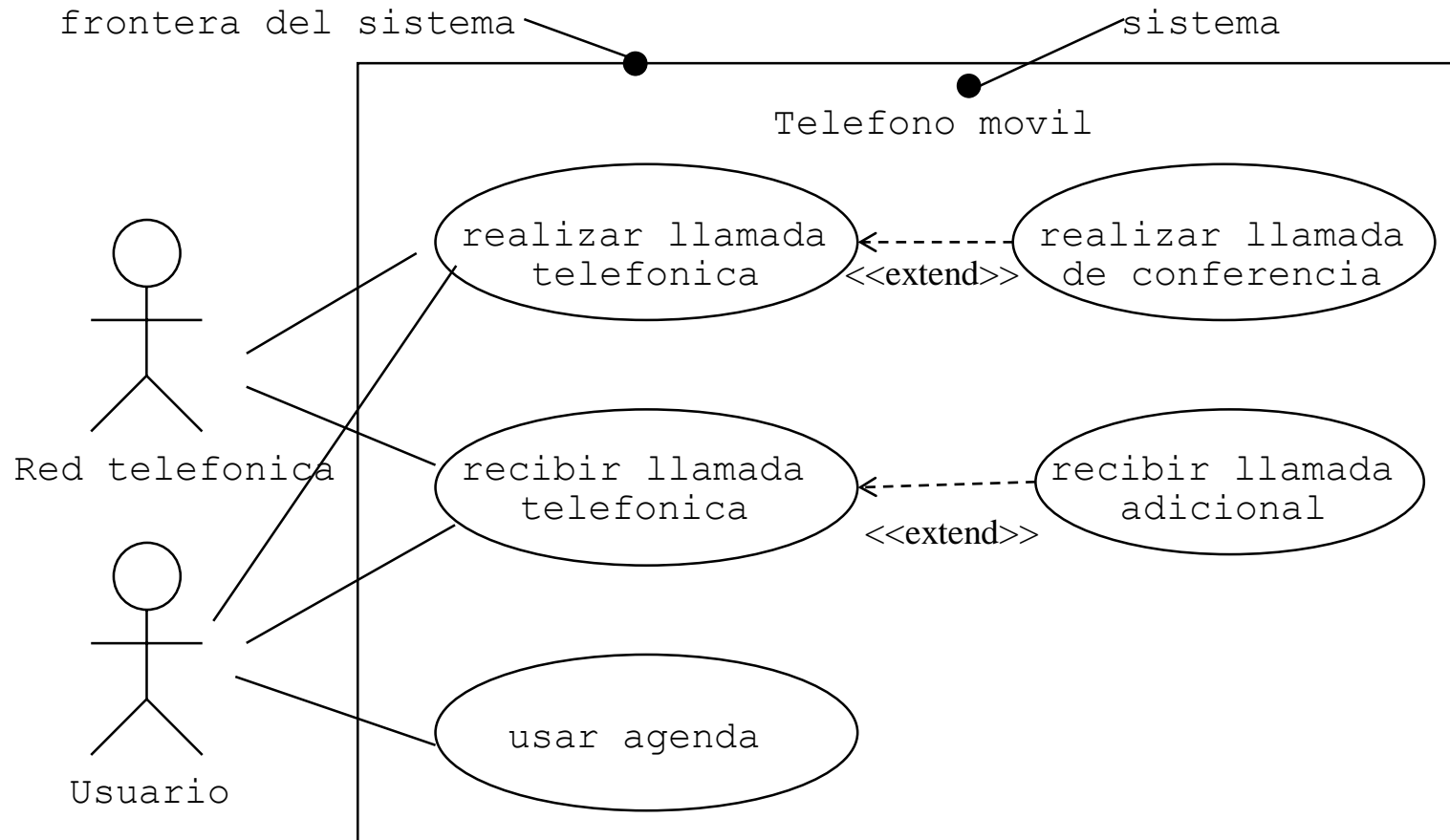
- Descripción de Caso de Uso "*Hacer pedido urgente*"
 - ✓ Este caso de uso extiende el caso de uso "*Hacer pedido*". Se inserta en el punto de extensión "establecer prioridad".
 - ✓ Flujo básico:
 - 1..2. Estos pasos son idénticos a los pasos 1..2 del caso de uso base
 3. El usuario confirma que quiere hacer un pedido urgente (y los posibles gastos adicionales)
 4. Se retoma el caso de uso base en el paso 4

Casos de Uso: Relaciones

□ Ejemplo completo:



Ejemplo



4. Recomendaciones: utilidad de los Casos de Uso

□ Modelar el comportamiento de un elemento (sistema, subsistema, clase):

- ❖ Centrarse en qué hace el elemento, NO en cómo lo hace
- ❖ 1º) Sirven para *intercambiar opiniones* entre los expertos del dominio, los usuarios finales y los desarrolladores
 - Los expertos del dominio especifican su vista externa para que los desarrolladores construyan su vista interna
- ❖ 2º) El creador del elemento comunica cómo se *debería usar*
 - El elemento puede ser complejo y tener muchas operaciones
- ❖ 3º) Sirven de base para probar el sistema una vez implementado

Construcción de diagramas de Casos de Uso (I)

□ Pasos a seguir:

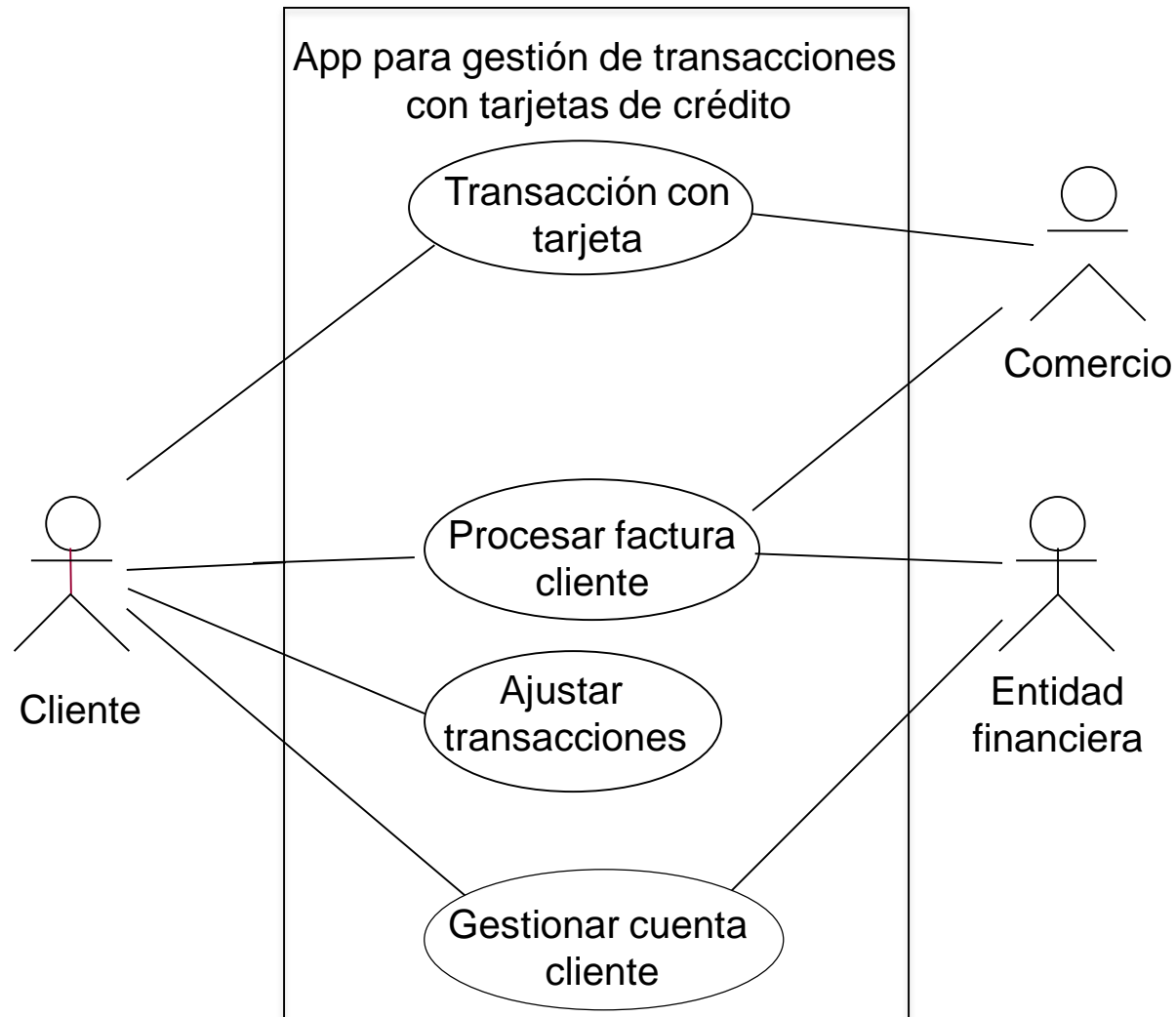
- ❖ Identificar los actores que interactúan con el elemento
- ❖ Organizar los actores (roles generales, roles especializados, ...)
- ❖ Considerar las formas más importantes que tiene cada actor de interactuar con el elemento
- ❖ Considerar las formas *excepcionales* que tiene cada actor de interactuar con el elemento
- ❖ Organizar estos comportamientos utilizando las relaciones entre casos de uso vistas
- ❖ Especificar cada caso de uso con texto y trazas de eventos

Construcción de diagramas de Casos de Uso (II)

❏ Sugerencias y consejos:

- ❖ Cada caso de uso debe representar un comportamiento distinto e identificable del sistema (razonablemente atómico)
- ❖ Factorizar el comportamiento común con *include*
- ❖ Factorizar las variantes de comportamiento con *extends*
- ❖ Describir el flujo de eventos de manera suficientemente clara para que alguien externo lo entienda
- ❖ Mostrar sólo los importantes para comprender el comportamiento del sistema
- ❖ Mostrar sólo los actores implicados

Ejemplo



5. Ejercicios

Ejercicio 1: Aplicación bancaria

- ❑ Supón que queremos desarrollar una aplicación, que se ejecute en dispositivos móviles, para una entidad bancaria. La aplicación permitirá al cliente del banco:
 - ❖ Consultar los saldos de sus cuentas corrientes y cuentas de ahorro
 - ❖ Consultar el listado de movimientos de sus cuentas corrientes y de ahorro
 - ❖ Transferir fondos entre cuentas corriente y de ahorro
 - ❖ Pagar facturas desde las cuentas corriente y de ahorro
 - ❖ Gestionar la lista de los beneficiarios a los cuales pagan las facturas
- ❑ Los clientes accederán a la aplicación de forma segura y podrán cambiar su clave en cualquier momento. Además, los clientes deberán autenticarse por segunda vez antes de realizar una transacción que modifique el saldo de una cuenta.
- ❑ **Tareas:**
 - ❖ Identifica los requisitos funcionales y no funcionales
 - ❖ Modela el sistema con un diagrama de casos de uso

Ejercicio 2: Aplicación de gestión de notas

- ❑ Se quiere construir una aplicación para gestionar notas en un dispositivo móvil y se han identificado los siguientes requisitos funcionales:
 - ❖ RF 1. El usuario puede crear una nota.
 - ❖ RF 2. El usuario puede modificar una nota existente.
 - ❖ RF 3. El usuario puede eliminar una nota existente.
 - ❖ RF 4. El usuario puede consultar el listado de notas ya creadas.
 - ❖ RF 5. Las notas constan de título y cuerpo.

❑ Tareas:

- ❖ Modela el sistema con un diagrama de casos de uso.
- ❖ Describe cada caso de uso con los flujos principales y alternativos.