

Diagramas de Casos de uso

- Un caso de uso representa una interacción típica entre un usuario y un sistema informático
- Utilizaremos los casos de uso para:
 - Capturar los requisitos funcionales del sistema
- Un caso de uso es un grafo con dos tipos de nodos:
 - Actor - *que representa cualquier elemento que intercambia información con el sistema, por lo que está fuera de él*
 - Caso de uso - *Es una secuencia de intercambios en diálogo con el sistema que se encuentran relacionadas por su comportamiento*
 - Los arcos entre los actores y los casos de uso se denominan arcos de comunicación
- El actor es un agente externo. Un actor representa un cierto papel que un usuario puede jugar.
- Una máquina también puede ser un actor.
- Cada caso de uso tiene una descripción informal en lenguaje natural o en un lenguaje estructurado

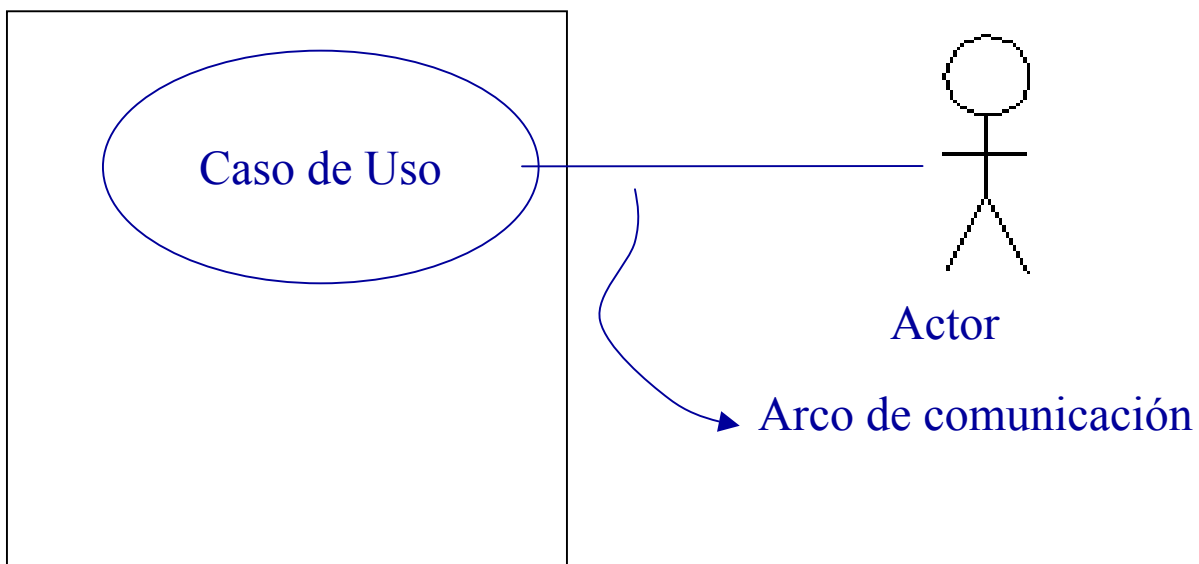
Diagramas de Casos de uso

Un diagrama de casos de uso es un grafo que incluye:

- los actores
- un conjunto de casos de uso encerrados en un recinto,
- la comunicación entre los actores y los casos de uso
- las generalizaciones sobre los casos de uso.

Notación de los casos de uso en UML

- Los casos de uso se representan por una elipse conteniendo el nombre, que opcionalmente podría ir debajo de la elipse.
- Los actores se representan con un monigote y el nombre del actor al pie de la figura. Los nombres de los actores suelen empezar por mayúscula.



Descripción de los Casos de uso

- Un caso de uso describe una funcionalidad más una interacción entre un actor y un sistema en forma de secuencia de acciones
- La descripción se centra en lo que debe hacerse, no en la manera de hacerlo
- Deben evitarse expresiones imprecisas. Se busca sencillez y claridad
- Puede utilizarse un lenguaje estructurado para representar secuencia, repeticiones y situaciones opcionales
- La descripción debe contener:
 - Inicio del caso de uso
 - Fin del caso de uso
 - Interacción entre el caso de uso y los actores
 - Intercambios de datos
 - Cronología y origen de los datos

Descripción de los Casos de uso

<Identificador>	<nombre descriptivo>	
Descripción □	El sistema deberá permitir a [lista actores] en [instante en el que se puede realizar el caso de uso] [funcionalidad que define el caso de uso] □ según se describe en el siguiente caso de uso: □	
Secuencia Normal	Paso □	Acción
	1 □	{<acción a realizar>, realizar el caso de uso [caso de uso]} □
	2	<Situación que produce una alternativa> □
	□	2a Si [Situación que produce una alternativa] el sistema deberá {<acción a realizar>, realizar el caso de uso [caso de uso]}
		2b Si [Situación que produce una alternativa] el sistema deberá {<acción a realizar>, realizar el caso de uso [caso de uso]}

	... □	...
n	... □	
Excepciones □	Paso □	Acción
	p	En el caso de que [situación que provoca la excepción] el sistema deberá {<acción a realizar>, realizar el caso de uso [caso de uso]}

	□q	...
Rendimiento	El sistema deberá realizar la/s acción /es descrita/s en {los pasos [primer paso] al [último paso], el paso [número de paso]} en un máximo de [cota de tiempo]	
Frecuencia	Este caso de uso se espera que se lleve a cabo una media de [número de veces] al [unidad temporal]	
Importancia	{vital, importante, quedaría bien}	
Urgencia	{inmediatamente, hay presión, puede esperar}	
Comentarios	<otras consideraciones en formato libre>	

Construcción de Casos de uso

- Es un proceso iterativo. Se van descubriendo los escenarios desde el punto de vista del usuario, es decir los ACTORES.
- Para detectar los casos de uso es conveniente hacer las siguientes preguntas:
 - ¿Cuáles son las principales tareas de cada actor?
 - ¿Escribe/lee/modifica el actor alguna información del sistema?
 - ¿Informa el actor al sistema de los cambios externos?
 - ¿Desea el actor ser informado de cambios no esperados?
- Es un proceso iterativo, en el que pueden utilizarse distintas técnicas de observación o de entrevista estructurada (para describir los escenarios potenciales desde el punto de vista del usuario).
- Los casos de uso no pueden ser demasiado pequeños, ya que deben aportar algún valor al actor.
- En el momento de identificar los actores es conveniente distinguir entre
 - actores principales (que son los que emplean directamente el sistema llevando a cabo las tareas más importantes)
 - actores secundarios (existen para que los principales puedan utilizar el sistema).
- La estructura del sistema debe decidirse teniendo en cuenta a los actores principales.

Construcción de Casos de uso

- Proceso de elaboración de los casos de uso
- Identificar a grandes trazos los casos de uso
 - Las principales etapas de cada caso de uso se describen en un par de frases
 - Se distingue un caso principal y se identifican los casos alternativos y excepciones
- Se establece un proceso iterativo en el cual los casos de uso se amplían, profundizándose en su descripción, buscándose etapas comunes y alternativas que representen en otros casos de uso relacionados por las relaciones incluye, generaliza y extiende.
- Se debe cuidar que:
 - Exista una descripción breve que represente una verdadera imagen del caso de uso
 - Las condiciones de arranque y parada del caso de uso estén bien definidas
 - Los usuarios estén satisfechos de la secuencia de interacciones entre el actor y el caso de uso

Construcción de Casos de uso

- El problema fundamental es encontrar el nivel de abstracción adecuado. En general si un caso de uso se hace demasiado grande, a medida que se va detallando es conveniente dividirlo en varios.
- Se pueden hacer preguntas como:
 - ¿es posible ejecutar un paso de forma independiente a los otros o siempre va encadenado con ellos? Dos pasos que siempre se encadenan forman parte habitualmente del mismo caso de uso
 - ¿es lógico agrupar varios pasos para documentarlos, probarlos o modificarlos en conjunto? Si es así, deben formar parte del mismo caso de uso.

•**Escenarios**

- Un caso de uso tiene como instancias los escenarios: situaciones concretas que deben recorrer total o parcialmente el caso de uso.
- Se deben considerar en lo posible todos los escenarios de modo que se pueda validar el caso de uso.
- La última comprobación consiste por tanto en asegurar que el caso de uso represente todos los escenarios.
- A veces se confunde el caso de uso con alguno de sus escenarios: Si aparecen muchos caso de uso puede que sea un síntoma de una mala descripción del sistema

Casos de uso: Ventajas y Peligros

•Ventajas

- *Ayudan a asegurar que se desarrolla el sistema correcto.*
- *Documentan las respuestas funcionales de caja negra.*
- *Excelente forma de comunicación con los clientes y los usuarios.*
- *Ayudan a gestionar la complejidad de los proyectos grandes.*
- *Proporcionan el fundamento de los mensajes.*
- *Ofrecen una buena base para la verificación y validación.*
- *Modo objetivo para el seguimiento del proyecto.*
- *Pueden servir como base para especificar respuestas a aplicaciones de tiempo real.*

Casos de uso: Peligros

•Llevan a una descomposición funcional del sistema.

- Los casos de uso son funcionales por naturaleza (esto es, localizan la información entorno a las funciones).
- No es un problema, pero debe tenerse cuidado cuando se utilizan dentro de un desarrollo orientado a objetos.
- Los problemas pueden surgir cuando en un desarrollo software se utilizan diferentes estrategias para localizar la información.
- Con los casos de uso pueden mezclarse diferentes paradigmas.

•Violación de la ocultación de la información.

- Cuando se describen casos de uso, se debe conocer no sólo el elemento para el que se desarrolla el caso de uso, sino también la interfaz pública definida de cada elemento.
- Los autores de casos de uso deben evitar la tentación de ir más allá de la interfaz pública, e intentar describir la estructura interna del elemento.

•Falta de formalidad.

- La informalidad de los casos de uso lleva a la gente a un falso sentido de seguridad. La gente se olvida de las normas para los nombres y otras convenciones de estilo.
 - Aumenta la posibilidad de cometer errores
 - Disminuye la probabilidad de reutilizar el caso de uso.

Casos de uso: Peligros

No saber cuando parar.

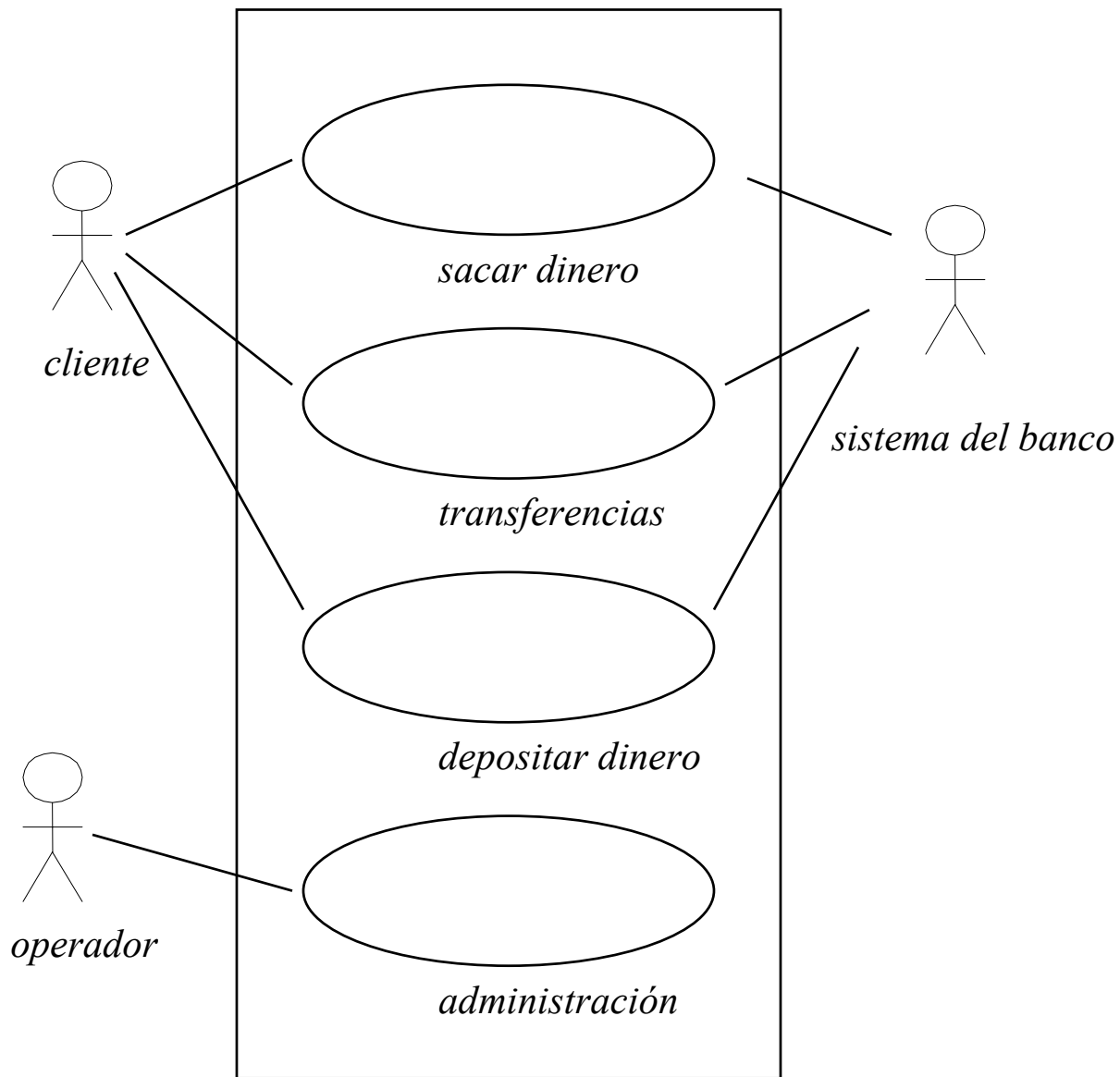
- Existe una gran confusión entre la adquisición de los requisitos y los casos de uso y entre el diseño y los casos de uso. Por ejemplo,
 - ¿Es un juego completo de casos de uso lo mismo que los requisitos de un producto?
 - ¿Existen otros requisitos (del producto o del proyecto) que no estén capturados en los casos de uso?
 - ¿Hay algún aspecto del diseño/arquitectura del sistema que no se ha capturado con los casos de uso?
 - Hay que tener claras las guías para establecer las relaciones entre los casos de uso, el análisis y el diseño.

La cobertura es el mayor problema de quien usa los casos de uso:

Una cosa es decir que “el conjunto de todos los casos de uso especifican la totalidad de la funcionalidad del sistema” ...

y otra cosa es demostrar que se ha capturado por completo la funcionalidad del sistema en un conjunto de casos de uso.

Casos de uso - Ejemplos



Ejemplo de un Cajero automático

CU-003	Sacar dinero
Descripción	El sistema deberá permitir al cliente del banco, en cualquier momento, sacar dinero según se describe en el siguiente caso de uso:
Secuencia Normal	<p>1+ El usuario inserta la tarjeta en el cajero</p> <p>2 + El cajero lee el código de la banda magnética de la tarjeta y verifica si es aceptable y pide el código del usuario</p> <p>3+ El usuario introduce el código</p> <p>4 + Si el código es correcto, el cajero pide al usuario que seleccione el tipo de transacción deseada</p> <p>5+ El usuario selecciona la función sacar dinero,</p> <p>6 + El cajero le pide al usuario que teclee la cantidad deseada</p> <p>7 + El usuario teclea la cantidad que quiere sacar,</p> <p>8 + El cajero envía la petición al sistema del banco</p> <p>9 a Si conecta el sistema deberá comprobar si hay dinero en la cuenta</p> <p>9 b Si no conecta el sistema deberá comprobar si el dinero es menos que el límite</p> <p>10 En cualquiera de los dos casos el sistema:</p> <ul style="list-style-type: none"> + expulsa la tarjeta + imprime el recibo + entrega el dinero
Excepciones	<p>2' La tarjeta no es aceptada</p> <ul style="list-style-type: none"> + Se expulsa emitiendo un sonido <p>4' Código incorrecto (1,2)</p> <ul style="list-style-type: none"> + Se emite un mensaje dando al usuario la oportunidad de volver a introducir el código (paso 3) <p>4" Código incorrecto (3)</p> <ul style="list-style-type: none"> + Se emite un mensaje y se retiene la tarjeta <p>9' No autorizado para sacar dinero</p> <ul style="list-style-type: none"> + El sistema de banco no autoriza a sacar dinero. Se emite un mensaje de información y se expulsa la tarjeta <p>9 a ', 9 b' No hay dinero suficiente</p> <ul style="list-style-type: none"> + El cajero no dispone de la cantidad pedida. Emite un mensaje y vuelve al paso 7 <p>1..10' Cancelar</p> <ul style="list-style-type: none"> + En cualquier momento el usuario puede cancelar la transacción, con lo que se expulsa la tarjeta

Ejemplo de un Cajero automático

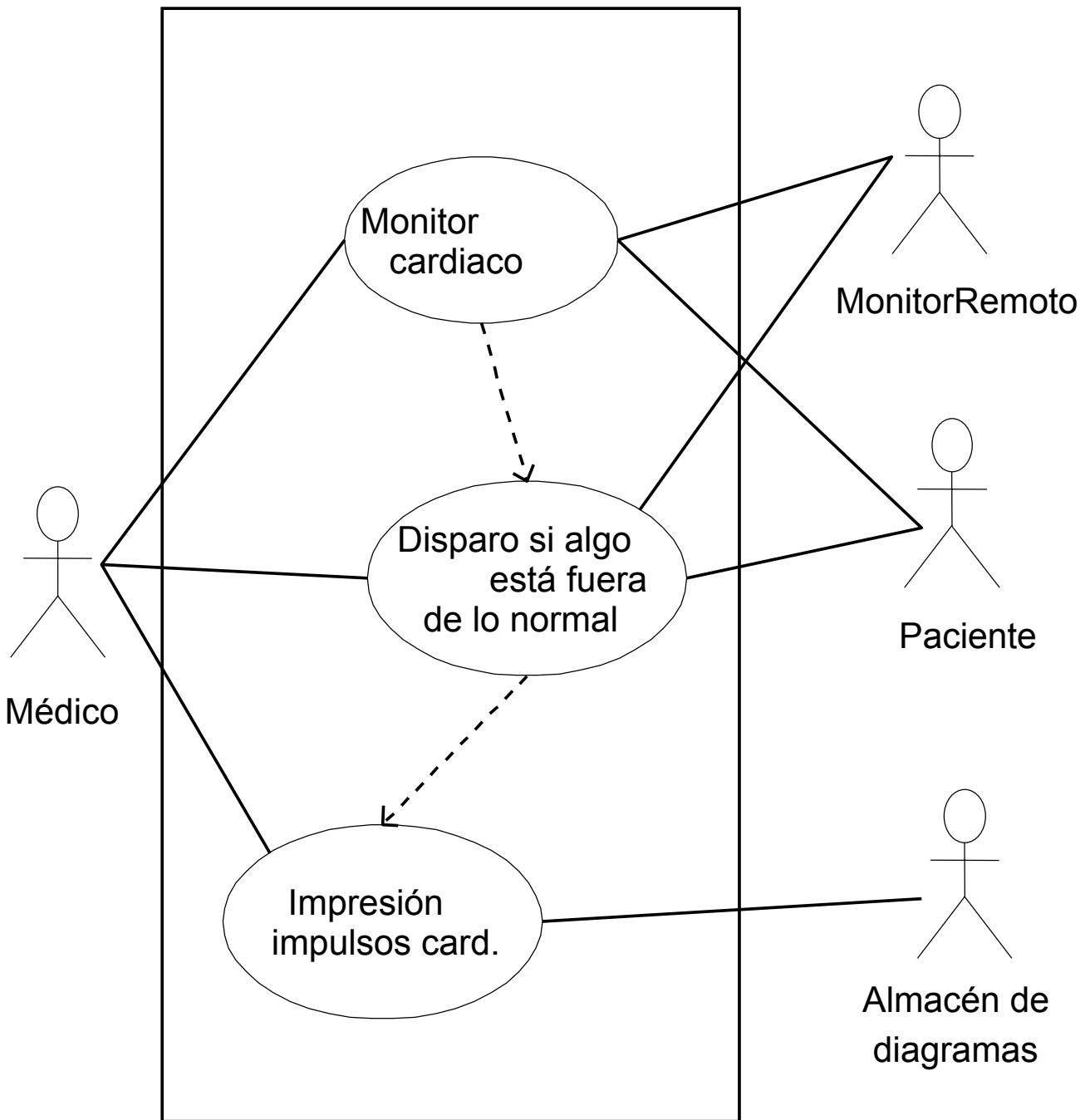
El caso de uso “sacar dinero”, puede describirse como sigue:

- o Se visualiza un mensaje de bienvenida en la pantalla:
 - + El usuario inserta la tarjeta en el cajero
 - + El cajero lee el código de la banda magnética de la tarjeta y verifica si es aceptable, en caso positivo, pide el código del usuario
- o Esperando el código:
 - + El usuario introduce el código
 - + Si el código es correcto, el cajero pide al usuario que seleccione el tipo de transacción deseada
- o Esperando el tipo de transacción:
 - + El usuario selecciona la función sacar dinero, el cajero le pide al usuario que teclee la cantidad deseada
- o Esperando la cantidad deseada:
 - + El usuario teclea la cantidad que quiere sacar, el cajero envía la petición al sistema del banco
 - + Se prepara un recibo
 - + Se expulsa la tarjeta
 - + Se imprime el recibo
 - + Se entrega el dinero

EXPCIONES

- o La tarjeta no es aceptada
 - + Se expulsa emitiendo un sonido
- o Código incorrecto
 - + Se emite un mensaje dando al usuario la oportunidad de volver a introducir el código
- o No autorizado para sacar dinero
 - + El sistema de banco no autoriza a sacar dinero. Se emite un mensaje de información y se expulsa la tarjeta
- o No hay dinero
 - + El cajero no dispone de la cantidad pedida. Emite un mensaje y expulsa la tarjeta
- o Cancelar
 - + En cualquier momento el usuario puede cancelar la transacción, con lo que se expulsa la tarjeta

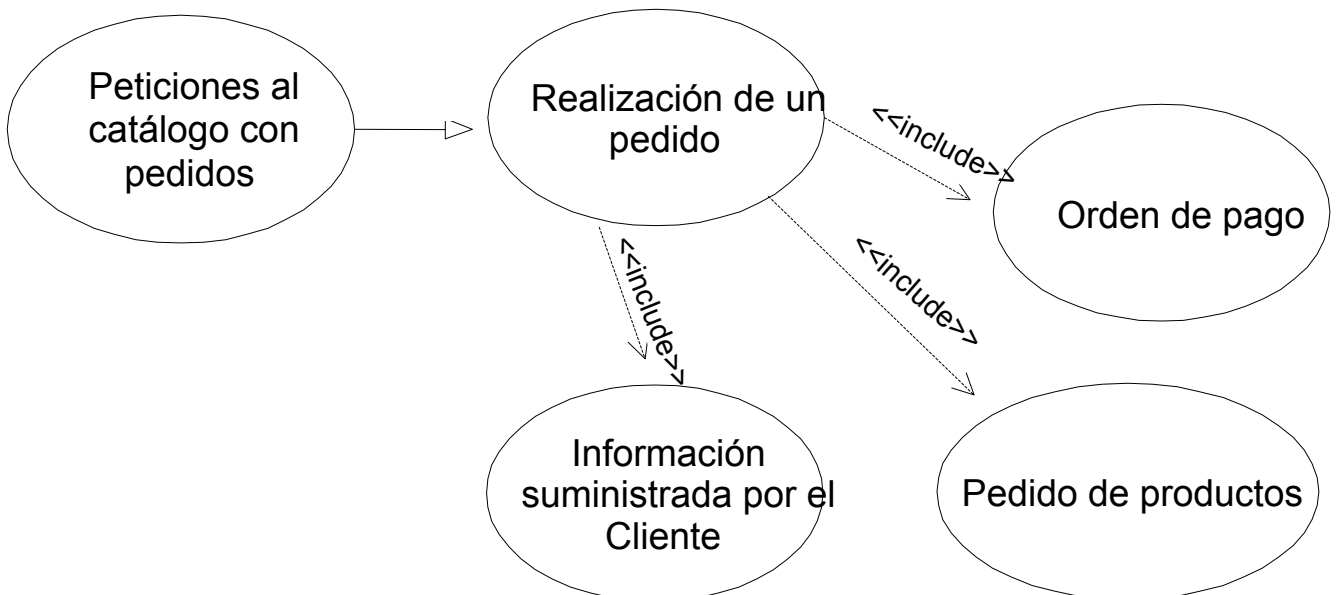
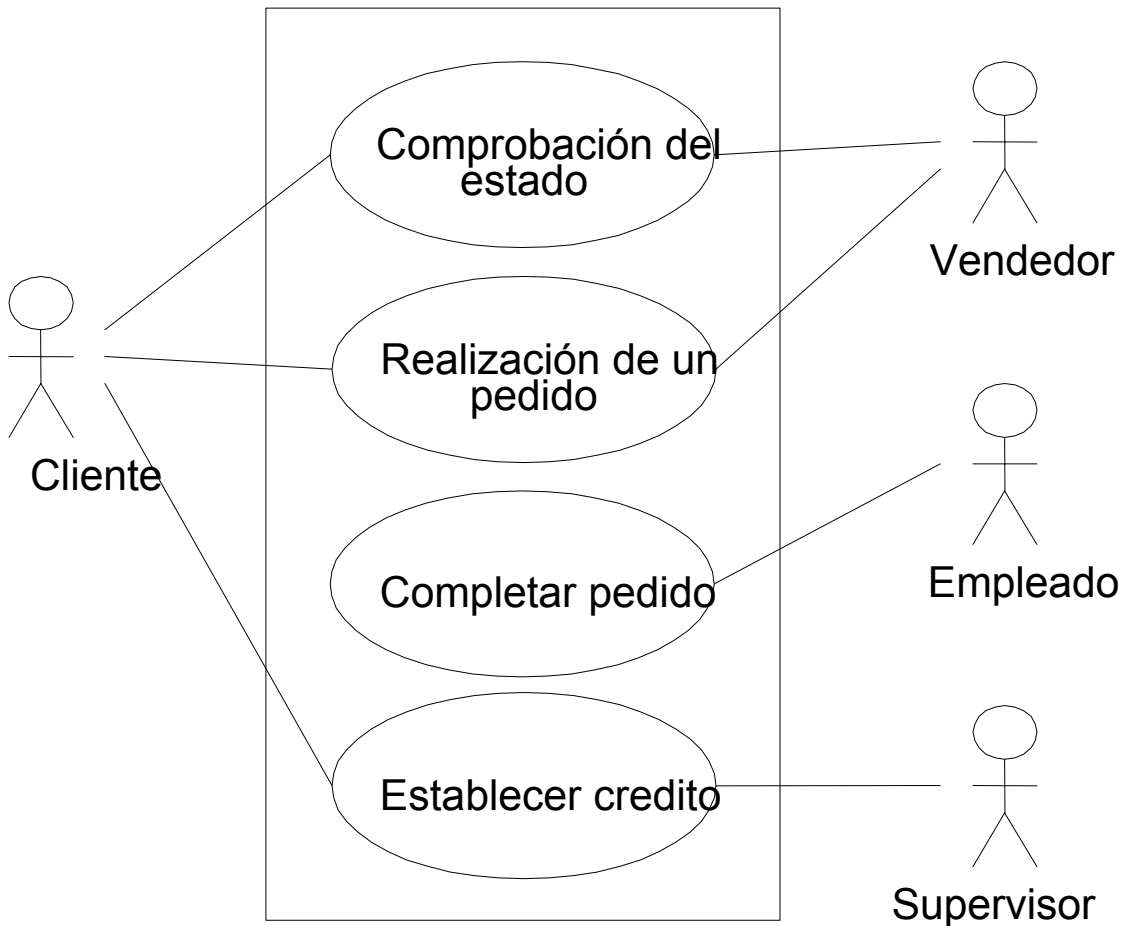
Casos de uso - Ejemplos



El diagrama representa el ejemplo de casos de uso para especificar el funcionamiento de una máquina que controla la actividad cardiaca de un paciente. El rectángulo marca los límites del sistema.

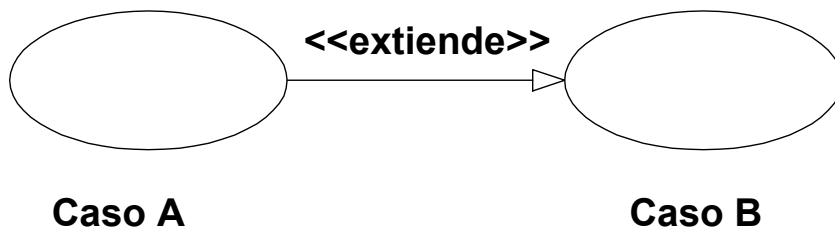
Casos de uso - Ejemplos

Venta por catalogo telefónico



Relaciones entre Casos de uso

- En UML 1.1 las relaciones *extiende* y *usa* se representan por la relación de generalización acompañadas de los textos <<extiende>> y <<utiliza>> respectivamente



El caso de uso A extiende el comportamiento del caso de uso B



El caso de uso A utiliza completamente el caso de uso B

Relaciones entre Casos de uso:

UML 1.3

- En UML 1.3 las relaciones entre casos de uso han cambiado:
 - **Incluye** (<<incluye>>) (<<include>>) - *Es un estereotipo de dependencia. Indica que un caso de uso es incluido en otro. Típicamente ocurre cuando unos casos de uso comparten unos pasos comunes. El caso de uso incluido es el factor común del comportamiento compartido. Reemplaza el uso común de la antigua relación **usa***
 - **Generalización** (sin estereotipo) - *Indica que un caso de uso es una variante de otro. El caso de uso especializado puede variar cualquier aspecto del caso de uso base*
 - **Extiende** (<<extiende>>) (<<extend>>) - *Es un estereotipo de dependencia. Ofrece una forma de extensión más controlada que la relación de generalización. El caso de uso base declara un conjunto de puntos de extensión. El caso de uso especializado sólo puede alterar el comportamiento de los puntos de extensión marcados*

Relaciones entre Casos de uso

Generalización

- Es una relación de generalización donde un caso de uso extiende otro caso de uso pudiendo añadir acciones a un caso de uso general.
- Indica que un caso de uso es una variante de otro. El caso de uso especializado puede variar cualquier aspecto del caso de uso base
- Cuando un caso de uso extiende otro, significa que el primero puede incluir parte del comportamiento del caso de uso que él extiende.
- No tiene porque incluir el comportamiento completo; pudiendo elegir que partes del comportamiento del caso más general se quieren reutilizar.
- Es una relación muy flexible.

Relaciones entre Casos de uso

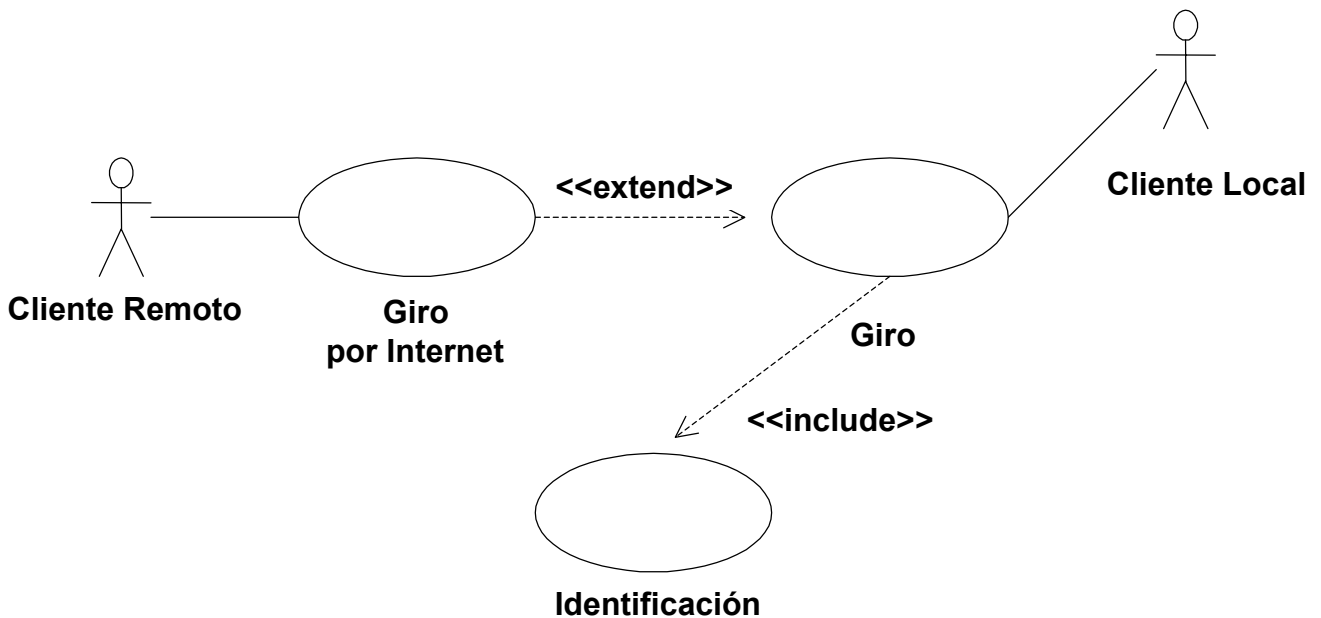
Extiende

- *Es una relación de dependencia donde un caso de uso extiende otro caso de uso añadiendo acciones a un caso de uso extendido.*
- Requiere que el caso de uso extendido tenga previstos unos puntos de extensión. Si hay más de uno, hay que identificar exactamente cual es es punto extendido.
- Un caso de uso extendido puede manejar excepciones, alternativas, etc. convirtiéndose en casos de uso específicos de casos de uso generales en los que no es fácil describir dichas situaciones.
- Es una alternativa a casos de uso complejos con múltiples alternativas, posibilidades de error, etc.

Relaciones entre Casos de uso

Incluye

- *Es una relación de dependencia donde un caso de uso utiliza otro caso de uso, indicando que es parte de un caso de uso.*
- Cuando un número de casos de uso comparten un comportamiento común, este comportamiento puede ser descrito por un caso de uso que es utilizado por otros casos de uso.
- Cuando un caso de uso incluye otro, el caso de uso completo debe ser usado. Si el caso de uso nunca se utiliza por sí mismo se denomina caso de uso **abstracto**.



Relaciones entre Casos de uso: UML 1.3

Resumen de los tipos de relaciones en UML 1.3

Relación	Función	Notación
<i>Asociación</i>	Camino de comunicación entre un actor y una caso de uso en el que participa	—————
<i>Extiende</i>	Inserción de comportamiento adicional en un caso de uso base (<i>sin que éste tenga conocimiento</i>)	- - - - - <<extiende>> →
<i>Generalización</i>	Relación entre un caso de uso general y otro más específico que hereda características y añade otras	————— → ▷
<i>Incluye</i>	Inserción de comportamiento adicional dentro de un caso de uso que explícitamente describe la inserción	- - - - - <<incluye>> →

Relación de las “viejas” relaciones con las “nuevas”:

- La mayoría utilizaba la relación <<utiliza>> de la forma que se usa ahora la relación <<incluye>>, por lo que se puede decir que la relación <<incluye>> reemplaza a la relación utiliza.
- Se utilizaba la relación <<extiende>> tanto de una forma controlada (como lo hace la relación <<extiende>> 1.3) como de una forma incontrolada (al estilo de la relación de generalización), por lo que se puede decir que la relación <<extiende >> 1.1 se ha dividido en dos.