



TEMA 3. Modelo de Requisitos del Sistema

Modelado del Software


Raquel Martínez España

Grado en Ingeniería Informática

Contenidos

- Modelado de casos de uso
- Conceptos de casos de uso
- Diagrama de casos de uso
- Relaciones entre casos de uso
- Organización de casos de uso
- Comportamiento de casos de uso

Contenidos

- **Modelado de casos de uso** 
- Conceptos de casos de uso
- Diagrama de casos de uso
- Relaciones entre casos de uso
- Organización de casos de uso
- Comportamiento de casos de uso

Caso de Uso (i)

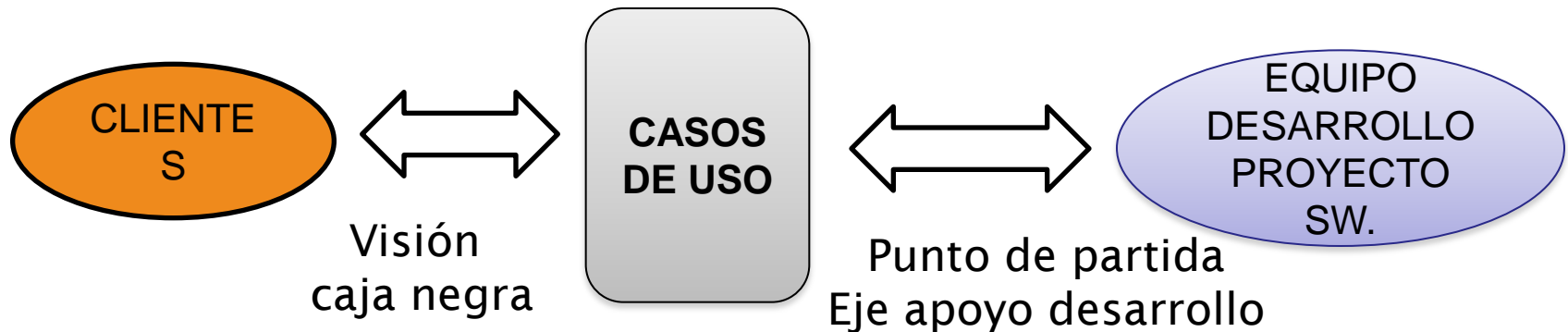
- **Definición**

- *“Unidad coherente de funcionalidad proporcionada por un clasificador (sistema, subsistema o clase)”*
 - Detalla los **servicios/funciones** que proporciona el sistema a los usuarios (especificación de requisitos).
 - Representa un **requisito funcional**
 - **El qué, no el cómo** (no detalles sobre implementación)
- Para la definición de casos de uso, tenemos que buscar los comportamientos deseados del sistema desde el punto de vista del usuario.
 - Suelen utilizarse como base para las pruebas.

Caso de Uso (ii)

- **Utilidad**

- Los casos de uso permiten **capturar los requisitos** del sistema
- Proporcionan **comunicación efectiva** entre cliente y desarrollador

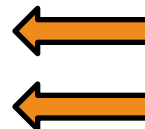


Importancia de los casos de uso


- Los casos de uso **inician y guían el proceso de desarrollo**:
 - Las **clases** recogen descripciones de los casos de uso
 - Los casos de uso son útiles para **desarrollar interfaces** de usuario.
 - Son la entrada del proceso de **pruebas**
 - Ayudar a los jefes de proyecto a **planificar, asignar y controlar las tareas** de desarrollo.
- Son un mecanismo importante para la **trazabilidad** a través de los modelos:
 - Los requisitos recogidos por un caso de uso son trazables a:
 - **Clases** implicadas en su realización
 - **Componentes** que participan en su implementación
 - **Casos de prueba** que lo verifican.

Modelo de Casos de Uso

- Un **modelo de casos de uso** especifica todas las posibles formas de usar un sistema, incluyendo:
 - **Usuarios** que interaccionan con la aplicación
 - **Casos de uso** del sistema
 - **Requisitos no funcionales** que determinan cómo proporcionará los servicios la aplicación.
- Parte del **contrato** con el cliente.
- **Primer paso** en el proceso de desarrollo.
- Formas de describir los casos de uso:
 1. Descripción textual
 2. Diagramas de Casos de Uso
 3. Especificación de los casos de uso



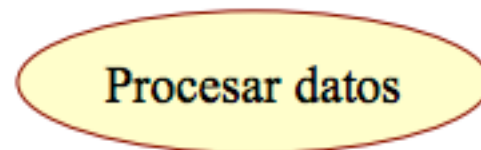
Contenidos

- Modelado de casos de uso
- **Conceptos de casos de uso** 
- Diagrama de casos de uso
- Relaciones entre casos de uso
- Organización de casos de uso
- Comportamiento de casos de uso

Conceptos de casos de uso (i)

- **Caso de Uso**

- “Descripción de un conjunto de acciones que ejecuta un sistema para producir un resultado observable, de valor para uno o varios actores”
- Notación:



Conceptos de casos de uso (ii)

- **Actor**

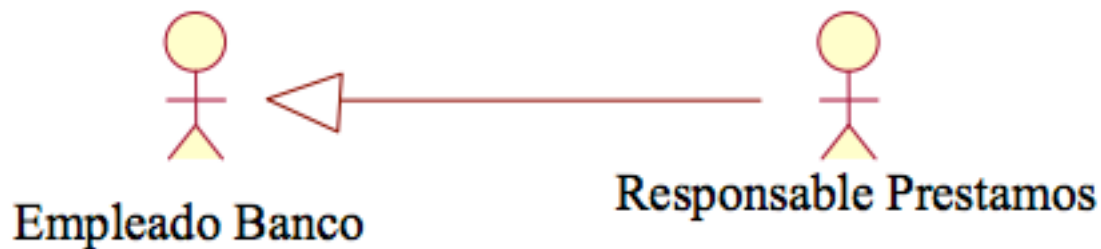
- **Rol** que los usuarios juegan al interactuar con la aplicación
- Los actores son siempre **externos** al sistema
- Un rol puede ser desempeñado por:
 - **Personas, dispositivos u otros sistemas.**
- Una misma persona, dispositivo o sistema **puede desempeñar varios roles.**



Conceptos de casos de uso (iii)

- **Actor**

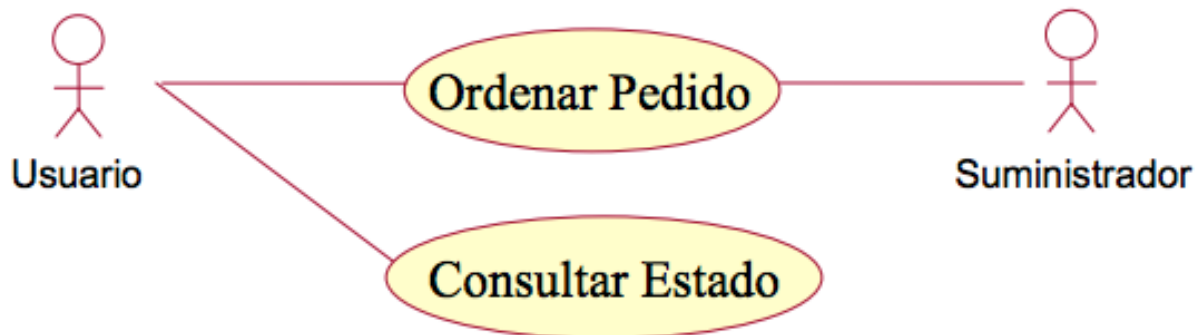
- Tipos de actores:
 - **Principales:** Utilizan el sistema directamente.
 - **Secundarios:** Supervisan y mantienen el sistema. Existan para que los principales puedan utilizar el sistema.
- Se pueden establecer categorías de actores a través de relaciones de **generalización**.



Conceptos de casos de uso (iv)

- **Actor**

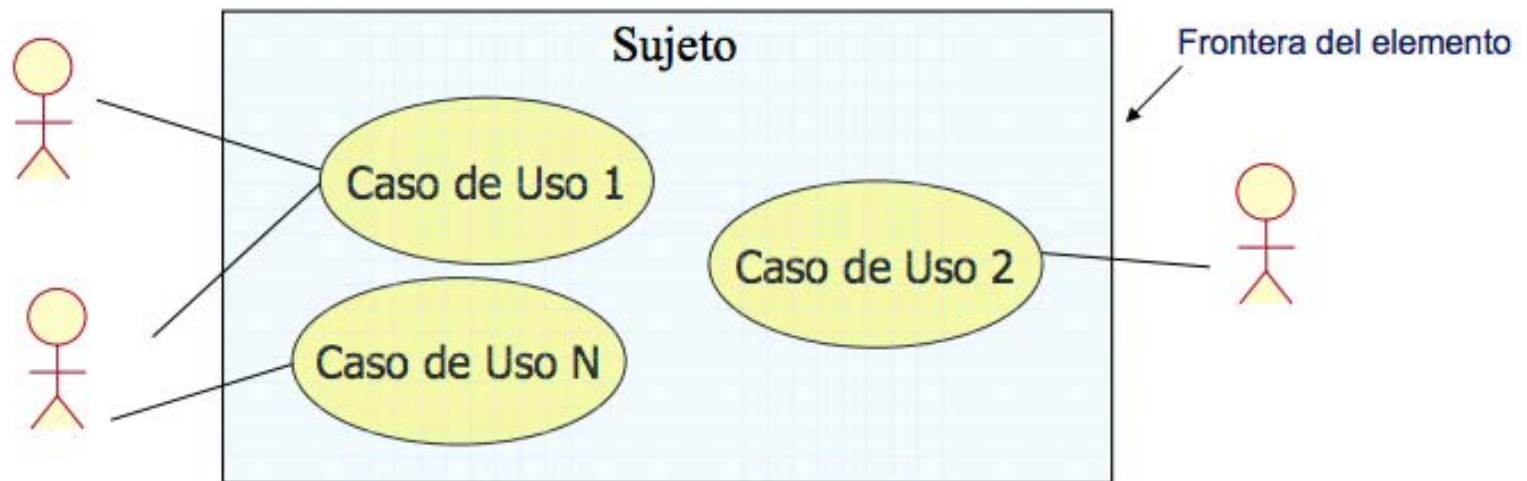
- Los actores se conectan con los casos de uso a través de relaciones tipo **asociación**:
 - Representa que el actor y el caso de uso se **comunican entre sí**, y cada uno puede **enviar y recibir mensajes o datos** del otro.
- Puede haber más de un actor asociado a un caso de uso.



Conceptos de casos de uso (v)

- **Sujeto**

- Elemento descrito por un conjunto de casos de uso.
- **Representa el sistema que se está modelando.**



Contenidos


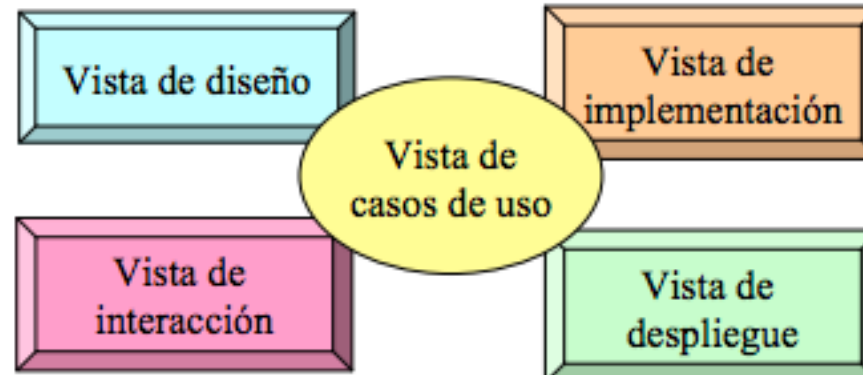
- Modelado de casos de uso
- Conceptos de casos de uso
- **Diagrama de casos de uso** 
- Relaciones entre casos de uso
- Organización de casos de uso
- Comportamiento de casos de uso

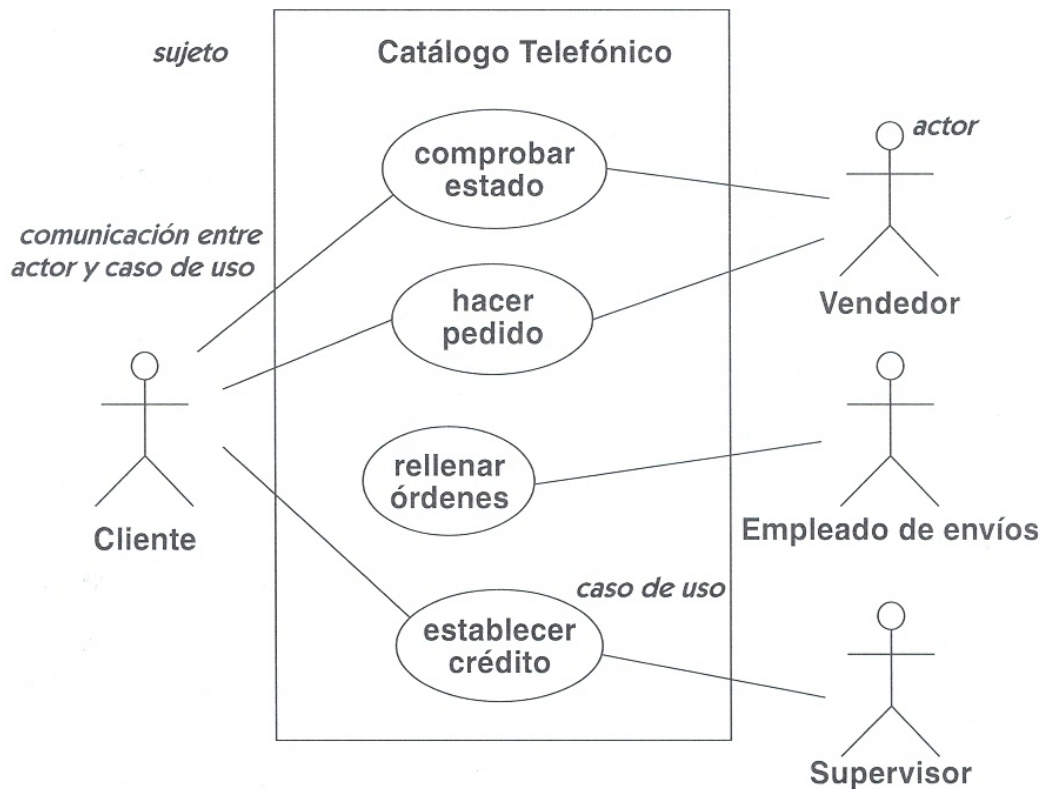
Diagrama de Casos de Uso

- Se utilizan para el modelado de aspectos estáticos de la Vista de Casos de Uso.

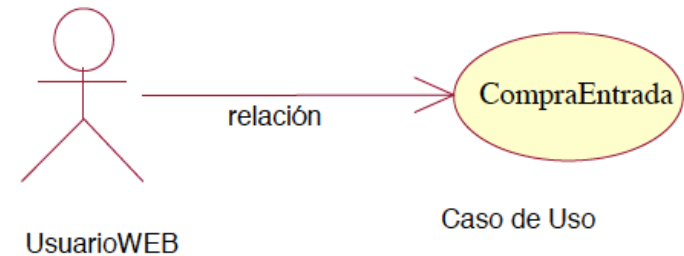
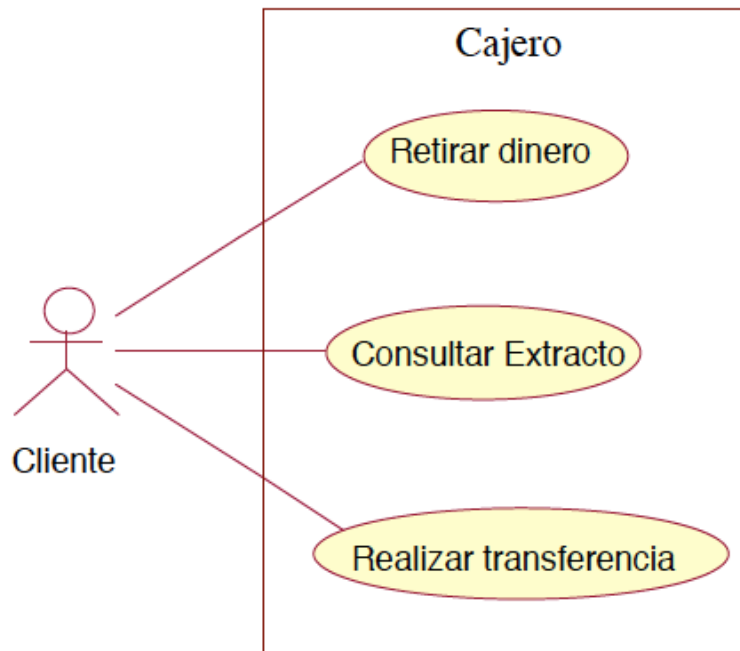


- Un diagrama de casos de uso contiene:
 - Elementos: **Actores**, **Casos de Uso**, **Sujetos**
 - **Relaciones** entre casos de uso
 - También puede contener:
 - **Paquetes** o **notas**.


Ejemplo de diagrama de casos de uso



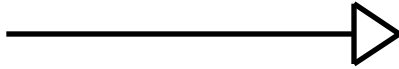
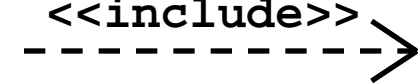
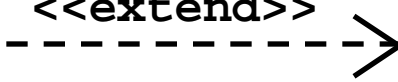
Ejemplo de diagrama de casos de uso



Contenidos

- Modelado de casos de uso
- Conceptos de casos de uso
- Diagrama de casos de uso
- **Relaciones entre casos de uso** 
- Organización de casos de uso
- Comportamiento de casos de uso

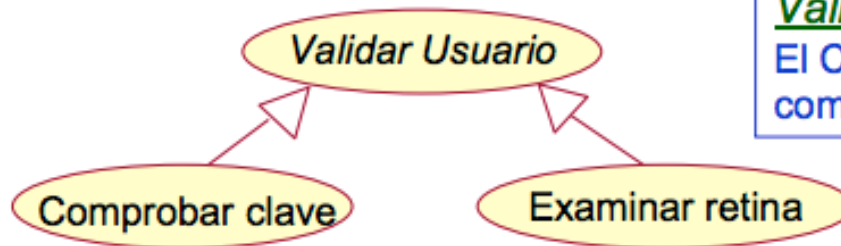
Relaciones entre Casos de Uso (i)

- **Generalización** (Herencia) 
 - El cdu hijo hereda el comportamiento y significado
 - El hijo puede añadir o redefinir el comportamiento del padre.
- **Inclusión** (Dependencia) `<<include>>` 
 - Un cdu agrega (“usa”) el conjunto de responsabilidades del otro.
- **Extensión** `<<extend>>` 
 - Un cdu base puede incorporar de forma opcional (en función de alguna condición) el comportamiento de otro caso en el lugar especificado en el caso base.
 - La funcionalidad del caso base se extiende con la del caso opcional.

Relaciones entre Casos de Uso (ii)

Generalización

- El caso de uso hijo hereda el comportamiento del padre.
- El caso de uso hijo puede:
 - Ser colocado en cualquier lugar donde aparezca el padre
 - Añadir o redefinir el comportamiento del padre.



Validar Usuario.

El CU es abstracto por lo que su comportamiento lo proporcionan los hijos

Comprobar Clave:

- Obtener contraseñas de la BBDD
- Pedir al usuario la contraseña
- El usuario introduce la contraseña
- Comprobar si la contraseña introducida coincide con la de la BBDD

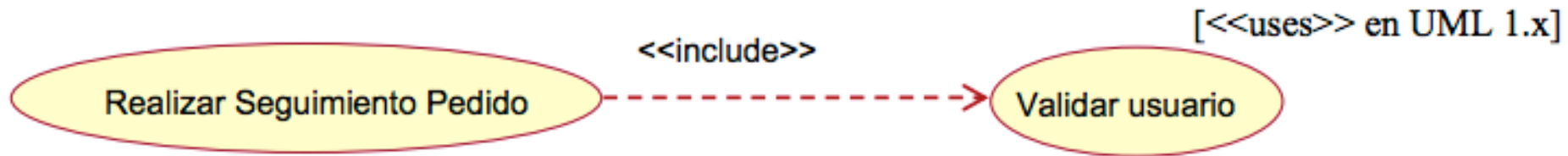
Examinar Retina:

- Obtener lecturas de retinas almacenadas en la BBDD
- Escanear la retina del usuario y obtener lectura de retina
- Comprobar si la lectura de retina del usuario coincide con la de la BBDD

Relaciones entre Casos de Uso (iii)

Inclusión

- Evitar describir el mismo flujo de eventos repetidas veces.
- El comportamiento común se pone en un caso aparte



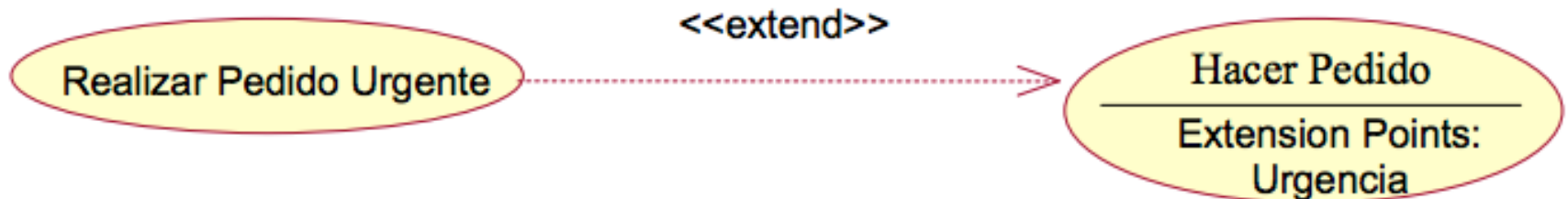
Realizar Seguimiento del Pedido. Flujo de Eventos Principal:

- Obtener y Verificar el Número de Pedido
- **Include (Validar Usuario)**
- Examinar el estado de cada parte del pedido
- Preparar un informe para el usuario

Relaciones entre Casos de Uso (iv)

Extensión

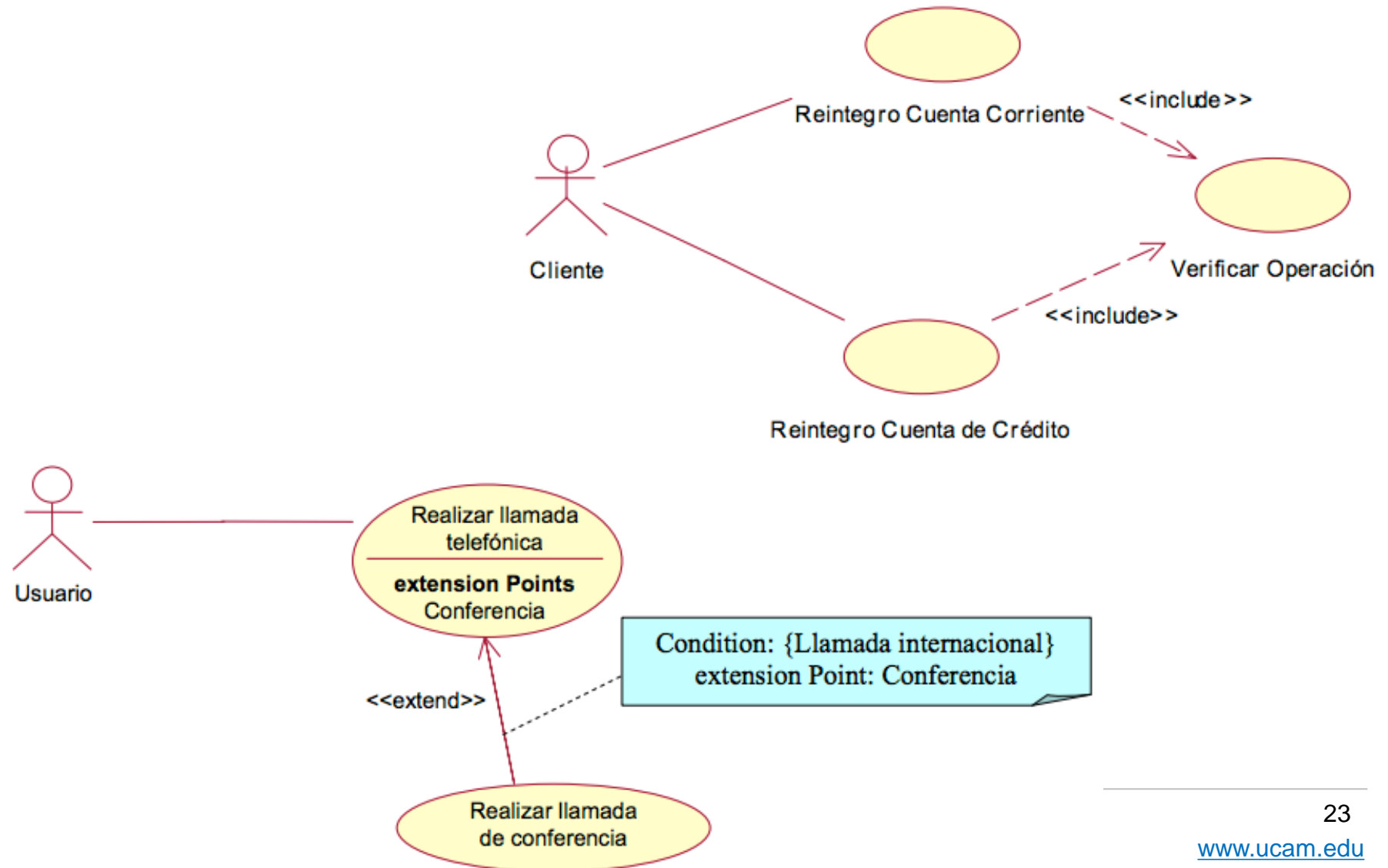
- Un caso de uso extiende el comportamiento de otro (base)
- Sólo es posible en ciertos puntos (**puntos de extensión**)
 - Un caso de uso puede tener varios puntos de extensión.
- Sirven para separar el comportamiento “obligatorio” del comportamiento “opcional” que se ejecuta bajo ciertas condiciones.



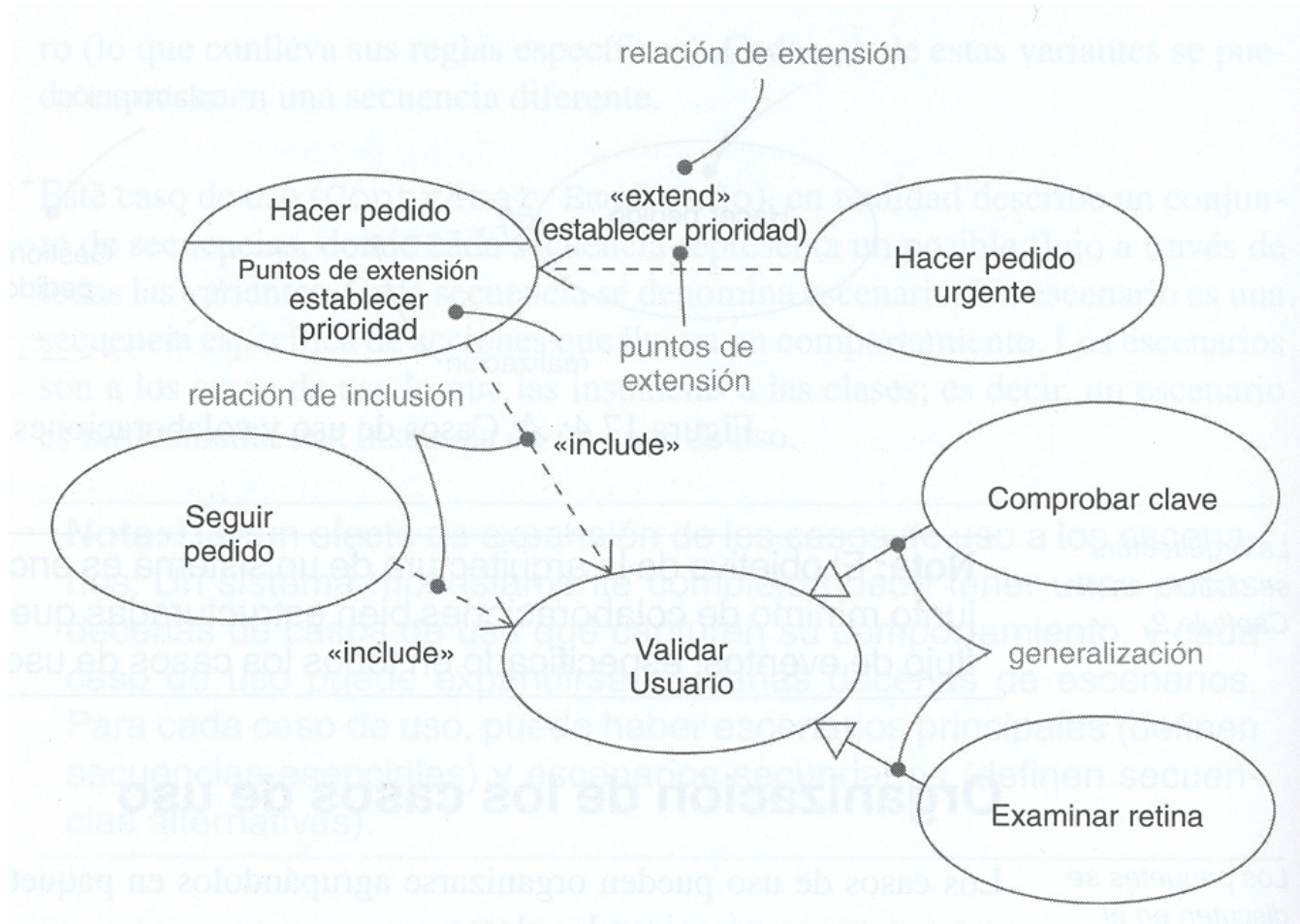
Hacer Pedido. Flujo de Eventos Principal:

- Obtener los productos pedidos por el Cliente
- Extension Point: Urgencia (Realizar Pedido Urgente)
- Enviar el pedido


Ejemplo relaciones (i)



Ejemplo relaciones (ii)

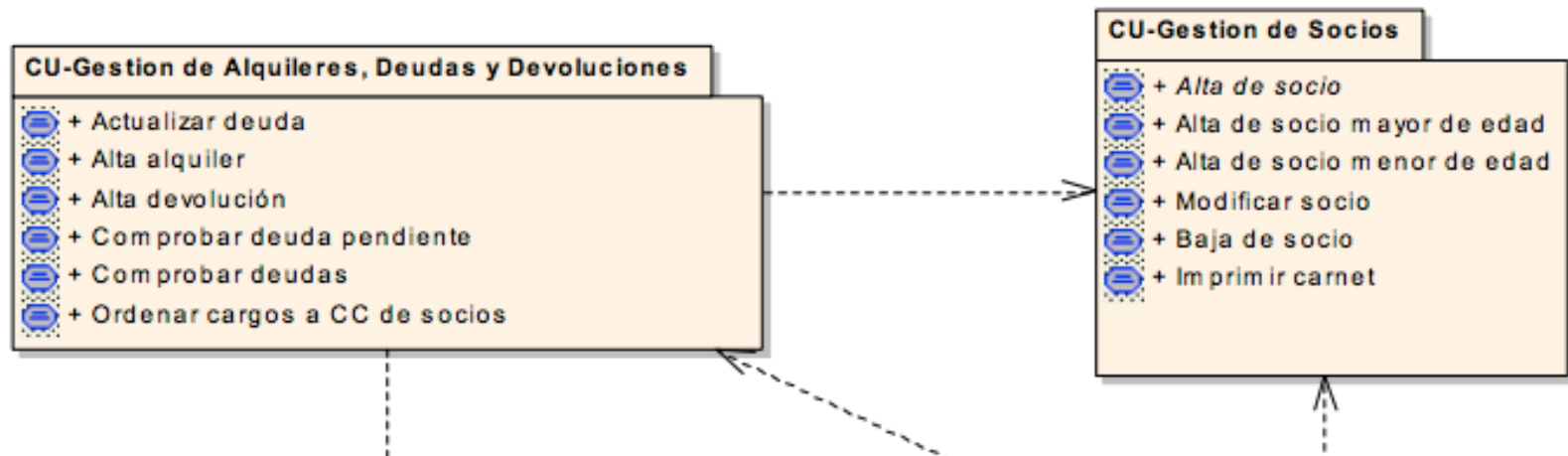


Contenidos


- Modelado de casos de uso
- Conceptos de casos de uso
- Diagrama de casos de uso
- Relaciones entre casos de uso
- **Organización de casos de uso** 
- Comportamiento de casos de uso

Organización de los casos de uso

- Los casos de uso pueden organizarse agrupándolos en **paquetes**.
- Conforme crecen los modelos, los casos de uso tienden a juntarse en grupos relacionados conceptual y semánticamente en paquetes.



Contenidos

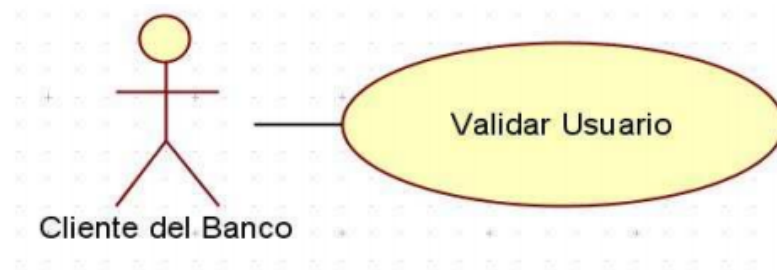
- Modelado de casos de uso
- Conceptos de casos de uso
- Diagrama de casos de uso
- Relaciones entre casos de uso
- Organización de casos de uso
- **Comportamiento de casos de uso** 

Comportamiento de un Caso de Uso

- El comportamiento de un cdu se puede especificar mediante un **flujo de eventos**
- Inicialmente usando descripción textual
- Más adelante empleando **diagramas de interacción o actividad**.
- El comportamiento de un cdu debe incluir:
 1. Cómo y cuándo **empieza** y **acaba** el cdu
 2. Cuándo interactúan los **actores** y la información que **envían** o **reciben** del sistema.
- Distinguimos entre:
 - **Flujo normal o principal**: comportamiento normal del cdu
 - **Flujos alternativos**: comportamientos excepcionales o de error

Comportamiento de un Caso de Uso

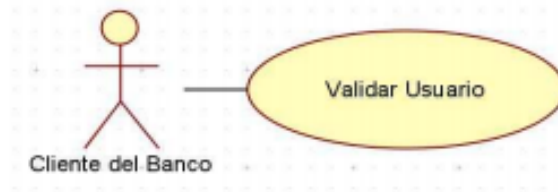
- **Ejemplo:** CdU **Validar Usuario** en un Cajero Automático



Flujo de Eventos Principal:

- El caso de uso comienza cuando el **Sistema** pide al **Cliente** un número de identificación personal (PIN).
- El **Cliente** introduce el PIN a través del teclado y acepta la entrada pulsando la tecla Enter.
- El **Sistema** comprueba si el PIN es válido.
- El **Sistema** acepta la entrada y así finaliza el caso de uso.

Comportamiento de un Caso de Uso



Flujo de Eventos Excepcional 1:

- El **Cliente** puede cancelar el proceso en cualquier momento pulsando el botón *Cancelar* reiniciando de esta forma el caso de uso.

Flujo de Eventos Excepcional 2:

- El **Cliente** puede borrar un PIN en cualquier momento antes de validarlo pulsando Enter y puede teclear un nuevo PIN.

Flujo de Eventos Excepcional 3:

- Si el **Cliente** introduce un PIN no válido, el caso de uso vuelve a empezar.
- Si esto ocurre tres veces en una sesión, el sistema se bloquea impidiendo que el **Cliente** use el cajero durante 2 minutos.

Especificación de un Caso de Uso

- **Elementos de una especificación:**
 - *Descripción general*
 - Reflejando uno o varios requisitos funcionales
 - *Pre-condiciones*
 - Condiciones que se deben cumplir para que se realice el caso de uso.
 - *Post-condiciones*
 - Condiciones que se cumplen posteriormente al caso de uso.
 - *Escenarios*
 - Descripción de todos los **flujos de eventos** posibles dentro del caso de uso (principales y alternativos).

RF- <id del requisito>	<nombre del requisito funcional>	
Versión	<numero de versión y fecha>	
Autores	<autor>	
Fuentes	<fuente de la versión actual>	
Objetivos asociados	<nombre del objetivo>	
Descripción	El sistema deberá comportarse tal como se describe en el siguiente caso de uso { concreto cuando <evento de activación> , abstracto durante la realización de los casos de uso <lista de casos de uso> }	
Precondición	<precondición del caso de uso>	
Secuencia Normal	Paso	Acción
	1	{El <actor> , El sistema} <acción realizada por el actor o sistema>, se realiza el caso de uso < caso de uso RF-x>
	2	Si <condición>, {el <actor> , el sistema} <acción realizada por el actor o sistema>>, se realiza el caso de uso < caso de uso RF-x>
	3	
	4	
	5	
	6	
	n	
Postcondición	<postcondición del caso de uso>	
Excepciones	Paso	Acción
	1	Si <condición de excepción>, {el <actor> , el sistema} }<acción realizada por el actor o sistema>>, se realiza el caso de uso < caso de uso RF-x>, a continuación este caso de uso {continua, aborta}
	2	
	3	
Rendimiento	Paso	Cota de tiempo
	1	n segundos
	2	n segundos
Frecuencia esperada	<nº de veces> veces / <unidad de tiempo>	
Importancia	{sin importancia, importante, vital}	
Urgencia	{puede esperar, hay presión, inmediatamente}	
Comentarios	<comentarios adicionales>	

Ejemplo de especificación

Realizar Venta *(en un terminal de punto de venta, TPV)*

Actor : Cajero

Descripción:

Un cliente llega al TPV con un conjunto de artículos. El Cajero registra los artículos y se genera un ticket. El cliente paga en efectivo y recoge los artículos.

Flujo:

1. El cliente llega al TPV con los artículos.
2. El cajero registra el identificador de cada artículo.
3. El sistema obtiene el precio de cada artículo y añade la información a la transacción de venta.
4. Al acabar el cajero indica la finalización de la introducción de artículos.
5. El sistema calcula el total de la compra y lo muestra.

Bibliografía básica

- Booch, Rumbaugh y Jacobson (2006): **El Lenguaje Unificado de Modelado**. Caps. 17 y 18.
- Rumbaugh, Jacobson y Booch (2007): **El Lenguaje Unificado de Modelado**. Manual de Referencia. Cap. 6.
- Booch y Rumbaugh (2000). **El Proceso Unificado de Desarrollo de Software**. Caps. 3, 6 y 7.
- Miles y Hamilton (2006): **Learning UML 2.0**. Cap. 2