



# INGENIERÍA DEL SOFTWARE I

## Práctica 3, Sesión 1

### *Modelado de Requisitos*

*Univ. Cantabria – Fac. de Ciencias*

*María Sierra*

Ref: Aurora Vizcaino, Felix Óscar García, Ismael Caballero



## Introducción

- Contexto General. ¿Por qué Casos de Uso?
  - Ningún sistema suele estar aislado

“La perspectiva que proporcionan los **Casos de Uso** refuerza el objetivo último de la **Ingeniería del Software**: la creación de **productos** que permitan a los **clientes** realizar un **trabajo útil**” (Wieger, 1997)



## Introducción

- Los Casos de Uso son las **funciones** que proporciona un **sistema** para añadir **valor** a sus **usuarios**
- Especifican el **comportamiento** deseado del **sistema**
  - **No se especifica cómo se implementan**
- Proporcionan un medio para que los desarrolladores, los usuarios finales y los expertos del dominio lleguen a una comprensión común del sistema.
- Ayudan a validar la arquitectura y a verificar el sistema
  - Conforme se desarrolla el sistema, los casos de uso son realizados por **colaboraciones**, cuyos elementos cooperan para realizar los casos de uso



## Introducción

- **Capturar los Requisitos** que aportan **valor añadido**
  - Importancia a la perspectiva del **USUARIO**
    - ¿A quién ayudan?, ¿Qué necesidades satisfacen?, ¿Cuánto valor añaden al negocio?
    - La captura de los casos de uso implica a:
      - **Usuarios/Clientes** → Son los expertos en los **requisitos**
      - **Desarrolladores** → Deben ayudar a los usuarios y clientes a comunicar sus necesidades
    - El Modelo de Casos de Uso es la especificación con todas las formas posibles de usar un Sistema desde la perspectiva de sus usuarios



# Introducción

- Los casos de uso **inician** el proceso de desarrollo y lo **enlazan**
  - REQ-ANÁLISIS-DISEÑO-IMPLEMENT-PRUEBAS
    - Clases → Descripciones Casos Uso
    - Interfaces de Usuario
    - Son la entrada del proceso de pruebas
    - Ayudan a los jefes de proyecto a planificar, asignar y controlar las tareas de desarrollo
    - Mecanismo importante para la **trazabilidad de requisitos**



# Conceptos Básicos

- **Caso de Uso**

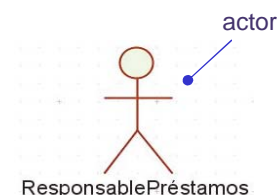


- *“Una descripción de un conjunto de secuencias de acciones, incluyendo variantes, que ejecuta un sistema para producir un resultado observable de valor para un actor”*

- Cada secuencia representa la interacción de los elementos externos al sistema (actores) con el sistema

- Variantes:

- Generalización, Inclusión, Extensión





# Conceptos Básicos

- Formas de Describir Casos de Uso:
  - Una simple descripción textual que recoja un requisito funcional
  - Diagramas de Casos de Uso
  - Especificación de Casos de Uso

Use Case Details1 - Dar de Alta de Socio

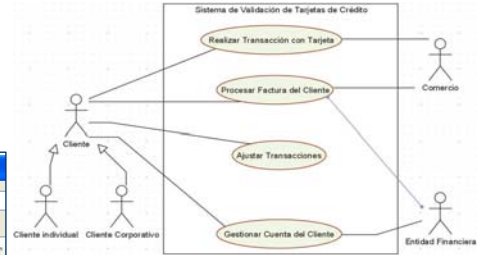
Name: Dar de Alta de Socio

Info Description Diagrams

Agency F8

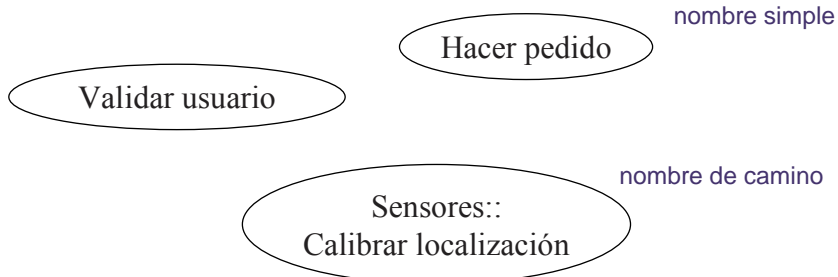
Flujo de Eventos Principal

<b>Super Use Case</b>	
<b>Author</b>	Félix
<b>Date</b>	31-mar-2007 13:07:58
<b>Brief Description</b>	En este caso de uso representa la funcionalidad de registro de un nuevo socio en la biblioteca
<b>Preconditions</b>	
<b>Post-conditions</b>	El nuevo socio es registrado en la BBDD
	<b>Actor Input</b>
	<b>System Response</b>
<b>Flow of Events</b>	1 El bibliotecario indica los datos del nuevo socio (DNI, nombre, apellidos, dirección, teléfono)
	2 El sistema comprueba si el socio ya existe en la biblioteca
	3 El sistema devuelve un mensaje en el que indica al bibliotecario que el nuevo socio ha sido registrado y le muestra su número de socio asignado



# Conceptos Básicos

- **Caso de Uso:**
  - **Nombre:**
    - Lo distingue de otros.
    - Puede incluir el nombre del paquete en que se encuentra.
    - Son expresiones verbales que describen algún comportamiento del sistema

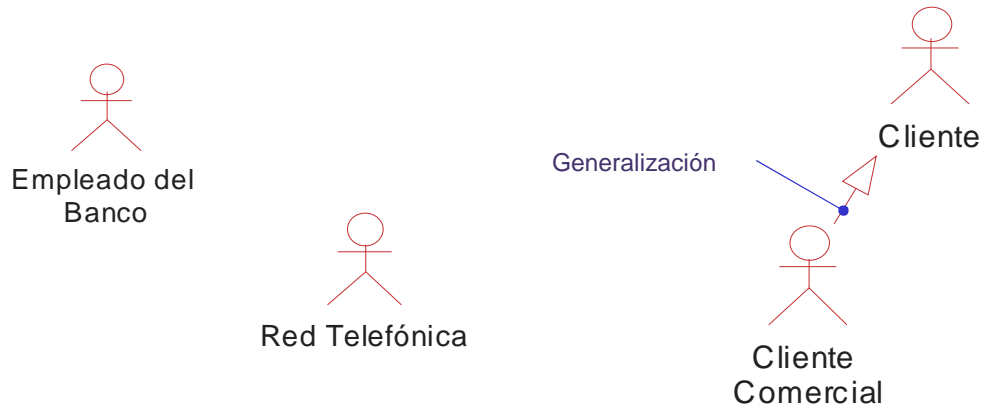




## Conceptos Básicos

- **Actor:**

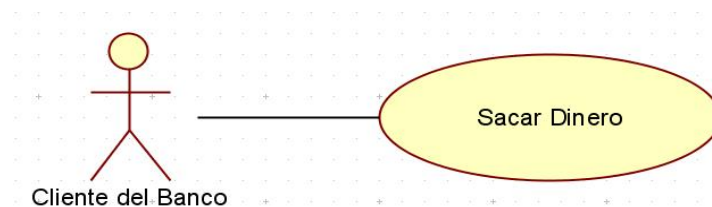
- Conjunto de roles que los usuarios de los casos de uso (persona, dispositivo HW u otro Sistema) juegan al interactuar con éstos.



## Conceptos Básicos

- **Actor:**

- Se pueden conectar a los casos de uso a través de **asociaciones**.
  - El actor y el caso de uso se comunican entre sí, y cada uno puede enviar y recibir mensajes.



**Actor ≠ Entidad externa (del análisis estructurado)**



## Conceptos Básicos

- **Actor:**
  - Tipos:
    - **Actores Primarios:** Utilizan el sistema directamente. Realizan las tareas principales.
    - **Actores Secundarios:** Supervisan y mantienen el sistema. Existen para que los primarios puedan utilizar el sistema.
  - Construcción de casos de uso desde los actores:
    - ¿Cuáles son las principales tareas de cada actor?
    - ¿Tendrá el actor que leer/escribir/cambiar información del sistema?
    - ¿Tendrá el actor que notificar al sistema de los cambios externos que se produzcan?
    - ¿Desea el actor ser informado de cambios no esperados?



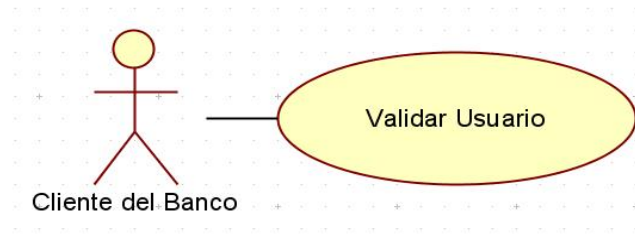
## Conceptos Básicos

- El comportamiento de un caso de uso se puede especificar describiendo un **Flujo de Eventos**.
- Cuando se describe un flujo de eventos se debe incluir:
  - Cómo y cuándo empieza y acaba el caso de uso.
  - Cuándo interactúa con los actores y qué objetos se intercambian.
  - El flujo básico y los flujos alternativos del comportamiento.



## Conceptos Básicos

- Ejemplo: Caso de Uso **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



## Conceptos Básicos

- Ejemplo: Caso de Uso **Validar Usuario** en un Cajero Automático:

### 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 introducirlo (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



## Conceptos Básicos

- Normalmente primero se describe el flujo de eventos mediante texto, pero a medida que se avanza en el proceso se utilizan diagramas de interacción
- En el contexto de estos diagramas, cada flujo de eventos (principal o alternativo) representa un **Escenario**:
  - Secuencia específica de acciones que ilustra un comportamiento.
  - Un caso de uso describe un conjunto de secuencias.
  - Cada secuencia representa un posible flujo a través de todas las variantes.
- Un **escenario** es una instancia de un caso de uso.
- Para cada caso de uso puede haber:
  - Escenarios principales (describen secuencias esenciales).
  - Escenarios secundarios (describen secuencias alternativas).



## Conceptos Básicos

- Especificación de Casos de Uso
  - Ejemplo: Plantilla de la herramienta Visual Paradigm

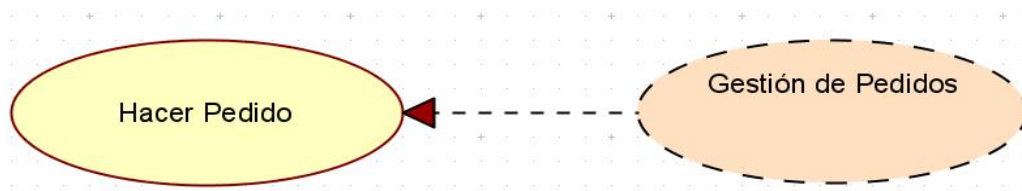
The screenshot shows the 'Use Case Details' window for 'Dar de Alta de Socio'. It includes a toolbar with various icons and a table with the following content:

Super Use Case	
Author	Félix
Date	31-mar-2007 13:07:58
Brief Description	En este caso de uso representa la funcionalidad de registro de un nuevo socio en la biblioteca
Preconditions	
Post-conditions	El nuevo socio es registrado en la BBDD
	<b>Actor Input</b>
	<b>System Response</b>
Flow of Events	1 El bibliotecario indica los datos del nuevo socio (DNI, nombre, apellidos, dirección, teléfono)
	2 El sistema comprueba si el socio ya existe en la biblioteca
	3 El sistema devuelve un mensaje en el que indica al bibliotecario que el nuevo socio ha sido registrado y le muestra su número de socio asignado



## Conceptos Básicos

- Un caso de uso captura el comportamiento deseado de un sistema (el qué) sin especificar cómo se implementa
- El Caso de Uso de debe implementar en las actividades posteriores del proceso de desarrollo:
  - REALIZACIÓN que se expresa explícitamente como una Colaboración:
    - Sociedad de Elementos incluyendo tanto su estructura estática como dinámica.

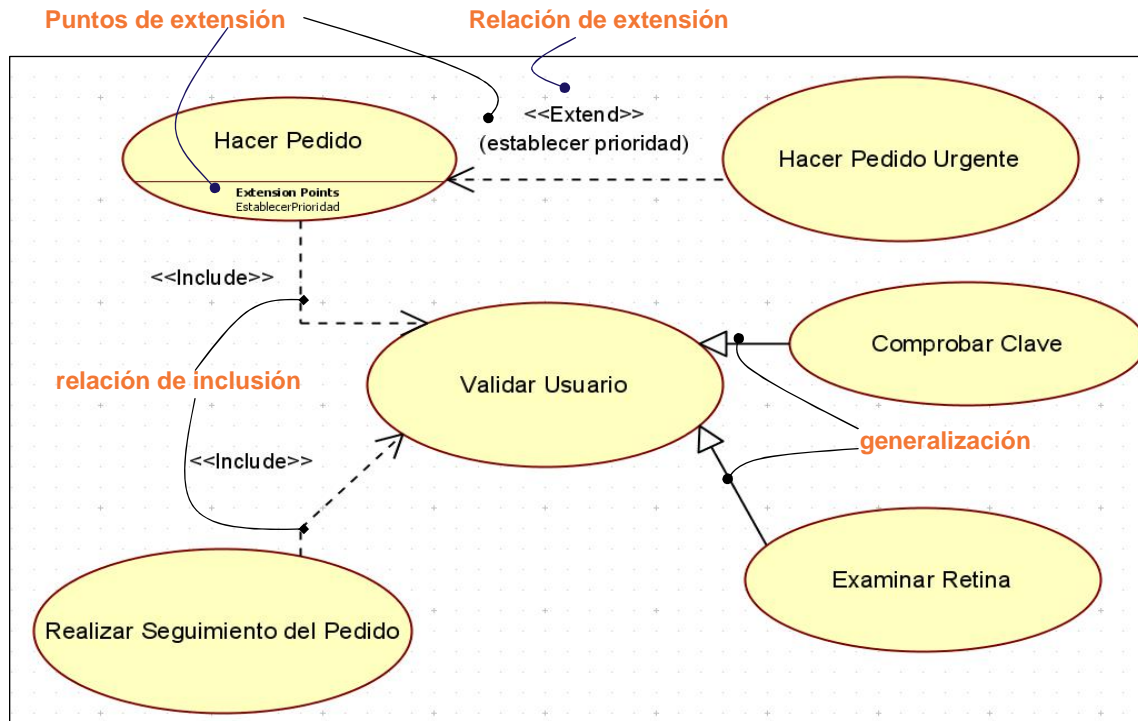


## Relaciones entre Casos de Uso

- Los casos de uso pueden organizarse agrupándolos en **paquetes**.
- Pueden **organizarse** especificando **relaciones** de:
  - **Inclusión**: un caso base incorpora explícitamente el comportamiento de otro caso en el lugar especificado en el caso base.
  - **Extensión**: significa que un caso base incorpora implícitamente el comportamiento de otro caso en el lugar especificado indirectamente por el caso que extiende al base.
  - **Generalización**: el caso hijo hereda el comportamiento del padre; el hijo puede añadir o redefinir el comportamiento del padre.



# Relaciones entre Casos de Uso



# Relaciones entre casos de Uso

## • Relación de Inclusión.

- Se usa para evitar describir el mismo flujo de eventos repetidas veces → El comportamiento común se pone en un caso de uso aparte
- Para especificarla en el flujo de eventos se debe escribir **include** seguido del nombre del caso de uso que se quiere incluir

### 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

- **Relación de Extensión.**

- Un caso extiende el comportamiento de otro caso (base)
  - Sólo es posible en ciertos puntos → Puntos de Extensión
  - Un caso de uso puede tener varios puntos de extensión
- Sirve para separar el comportamiento "obligatorio" del comportamiento "opcional" o para modelar ciertos subflujos de eventos que se ejecutan sólo bajo ciertas condiciones

Hacer Pedido. Flujo de Eventos Principal:

- Incluye (Validar Usuario)
- Obtener los productos pedidos por el Cliente
- (establecer prioridad)
- Enviar el pedido



## Relaciones entre casos de Uso

- **Relación de Generalización.**

- Relaciona un caso de uso especializado con uno más general
- El caso de uso hijo hereda los atributos, operaciones y secuencias de comportamiento del padre

Validar Usuario. Flujo de Eventos Principal:

- El cdu es abstracto por lo que su comportamiento lo proporcionan los hijos

Comprobar Clave. Flujo de Eventos Principal:

- 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. Flujo de Eventos Principal:

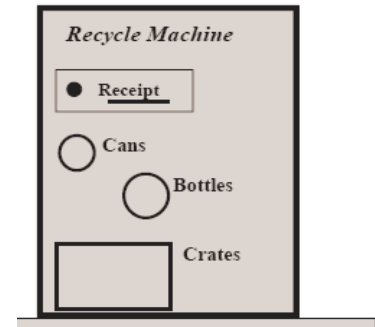
- Obtener lecturas de retinas almacenadas en la BBDD
- Escanear la retina del usuario y obtener lectura de retina
- Comprobar si la lectura de retina del usuario coincide con la de la BBDD



## Ejemplo. Máquina de Reciclaje

- **Requisitos del Sistema:**

- Se trata de construir un sistema que controle una máquina para reciclar latas, botellas y cajas de botellas. La máquina puede ser utilizada por varias personas a la vez.
- El sistema registra cuántos elementos recicla cada usuario e imprime un recibo con el valor de los elementos reciclados y el total a pagar al usuario.
- El sistema también lo utiliza un operador que desea conocer el número de elementos de cada tipo que se han introducido durante el día. También debe poder actualizar los valores de los elementos.
- El sistema activará una alarma para avisar al operador cuando se acabe el papel de los recibos o si detecta algún problema (atasco, etc..) con los elementos que se han introducido para reciclar.



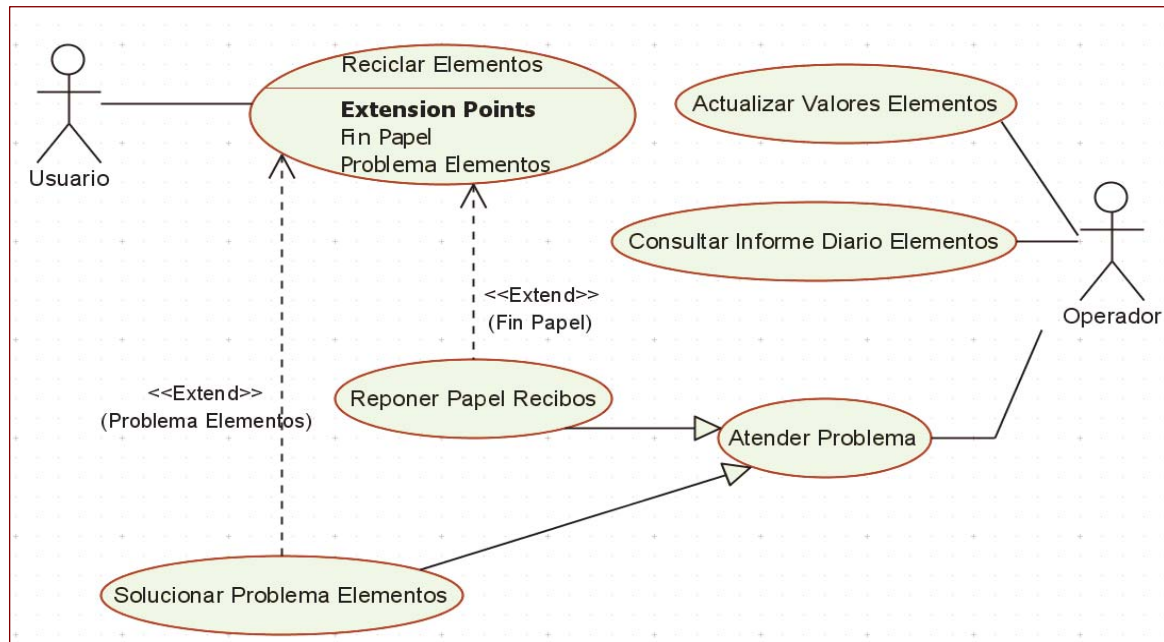
## Ejemplo. Máquina de Reciclaje

- **CDU Reciclar:**

Para el uso de la máquina los trabajadores deben pulsar el botón de inicio de reciclaje, tras lo cual la máquina muestra un mensaje indicando al usuario que puede comenzar a reciclar elementos. Cada vez que el usuario recicla un elemento la máquina debe actualizar los contadores locales (hay un contador local para cada tipo de elemento que introduce el usuario) y los contadores globales (contador diario de los elementos que de cada tipo que se han introducido durante el día), así como ir actualizando el recibo con los elementos reciclados. Cuando el usuario ha reciclado todos sus elementos acciona una palanca de fin de reciclaje y la máquina le expide un recibo con los totales de los contadores locales y la cantidad de dinero a pagar al usuario.



## Ejemplo. Máquina de Reciclaje



## Técnicas Comunes de Modelado

- Los Casos de Uso se utilizan para **modelar el comportamiento de un elemento**:
  - Identificar los actores que interactúan con el elemento.
  - Organizar los actores identificando tanto los roles más generales como los más especializados.
  - Considerar:
    - las formas más importantes que tiene cada actor de interactuar con el elemento.
    - las formas excepcionales en las que cada actor puede interactuar con el elemento.
  - Organizar estos comportamientos como casos de uso.



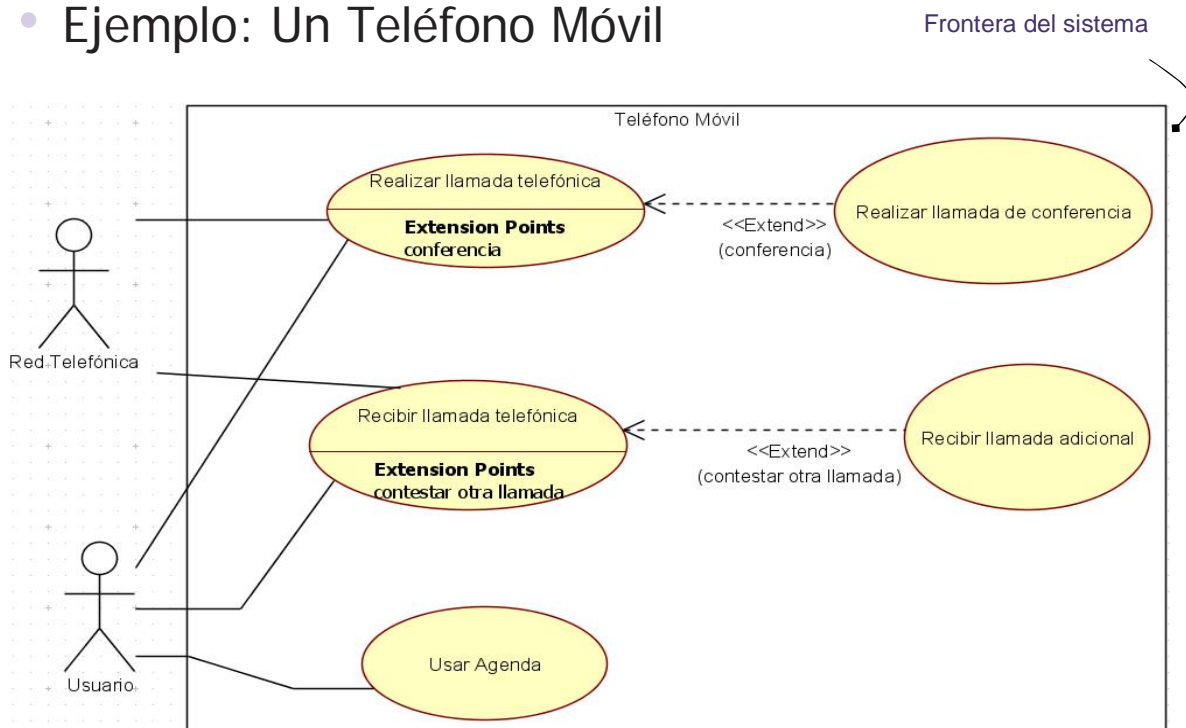
# Casos de Uso bien Estructurados

- Cuando se modelan los Casos de Uso en UML, cada caso debe representar un comportamiento distinto e identificable del sistema o de una parte del mismo.
- Un Caso de Uso bien estructurado:
  - Nombra un comportamiento simple, identificable y razonablemente atómico.
  - Factoriza el comportamiento común.
  - Factoriza las variantes.
  - Describe el flujo de eventos de forma suficientemente clara para que alguien externo al sistema lo entienda fácilmente.
  - Se describe por un conjunto mínimo de escenarios que especifican la semántica normal y de las variantes del caso de uso.



# Diagramas de Casos de Uso

- Ejemplo: Un Teléfono Móvil





## Diagramas de Casos de Uso

- Permiten visualizar, especificar y documentar el contexto del sistema, de un subsistema o clase, o el modelado de requisitos de comportamiento de esos elementos
- Contienen:
  - Casos de Uso
  - Actores
  - Relaciones de dependencia, generalización y asociación



## Ingeniería directa e inversa

- **Ingeniería directa**
  - Los casos de uso describen cómo se comporta un elemento, no cómo se implementa este comportamiento
    - No se puede aplicar ingeniería directa para generar código.
  - Si es posible aplicar ingeniería directa para generar pruebas del elemento al que se aplica.
- **Ingeniería inversa**
  - Conseguir un diagrama de casos de uso a través de ingeniería inversa es impensable con la tecnología actual, ya que se produce una pérdida importante al pasar de la especificación a la implementación.
  - Sí se puede hacer de manera manual



# Diagramas de Casos de Uso

---

- Al **dibujar** un Diagrama de Casos de Uso:
  - **Asignarle** un **nombre** que comunique su propósito.
  - **Distribuir** sus **elementos** para minimizar los cruces de líneas.
  - **Organizar** sus **elementos** espacialmente para que los comportamientos y roles semánticamente cercanos se encuentren cercanos físicamente.
  - **Utilizar** las notas y los colores como señales visuales.
  - **Intentar no mostrar** demasiados tipos de relaciones.