Casos de uso

***Índice***

[**Diagrama de Casos de Uso**](#_heading=h.63mgdrw8fulo) **1**

[Elementos](#_heading=h.krimghmah4qc) 2

[Extensión de casos de uso (“extends”)](#_heading=h.o10srr2pskxy) 3

[Inclusión de casos de uso (“includes”)](#_heading=h.dlx65xi3wpk4) 3

[Tareas vs Casos de Uso](#_heading=h.uog5kecqejy5) 5

[Resumen](#_heading=h.9yvopt7gb7v3) 6

[**Especificación de un caso de uso**](#_heading=h.yk7d00w002i) **6**

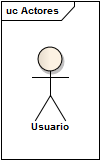
[Ejemplo Salón de Videojuegos](#_heading=h.mtki1713kpk3) 7

# Diagrama de Casos de Uso

Este diagrama nos permite definir qué es lo que el sistema debe hacer. Surge del relevamiento inicial con el usuario y se va refinando sucesivamente para llegar al desarrollo de cada funcionalidad.

## 

## Elementos



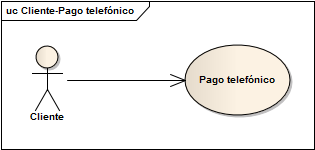
**Actores**

Representan roles o papeles que cumplen usuarios o sistemas externos. Son los que solicitan que el sistema haga determinadas cosas. Una misma persona física puede cumplir distintos roles, en distintos momentos.

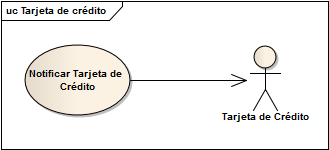


**Casos de uso**

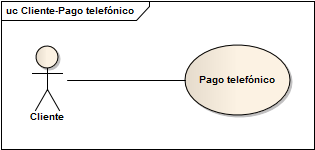
Representan requerimientos funcionales, descritos desde el punto de vista del usuario. El nombre se recomienda sea un verbo y un sustantivo; dicho nombre debe dar a entender qué es lo que el caso de uso resuelve.



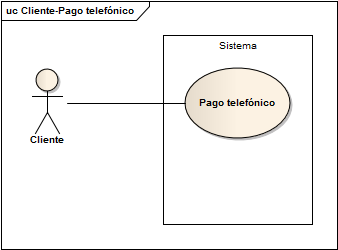
**Asociaciones**

La asociación entre un actor y un caso de uso representa la participación del actor en ese caso de uso. Si la flecha de navegación va desde el actor hacia el caso de uso indica que quien inicia el caso de uso es dicho actor.

Cuando la flecha tiene la dirección contraria lo que se indica es que el actor toma el resultado del caso de uso ejecutado.



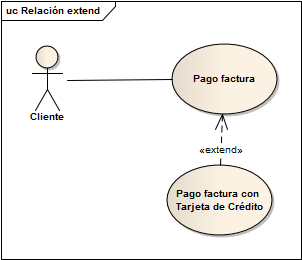
Las flechas son opcionales, en ese caso no estamos especificando quién inicia la interacción o recibe el resultado



**Límite del sistema (opcional):**

Los casos de uso se los suele ubicar dentro de un recuadro que delimita el conjunto de requerimientos que forman parte del alcance del sistema

## Extensión de casos de uso (“extends”)

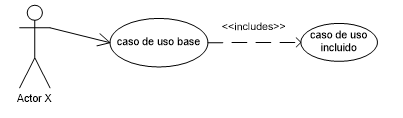


Un caso de uso puede extender el comportamiento de otro más general:

El “Pago Tarjeta de Crédito” (caso de uso extendido) representa una condición específica del Pago factura, cuando el cliente no quiera pagar en efectivo.

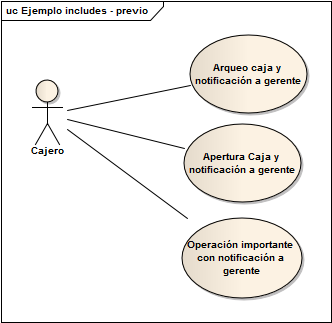
La extensión de casos de uso nos sirve cuando una parte de un caso de uso es opcional, dependiendo de una cierta condición.

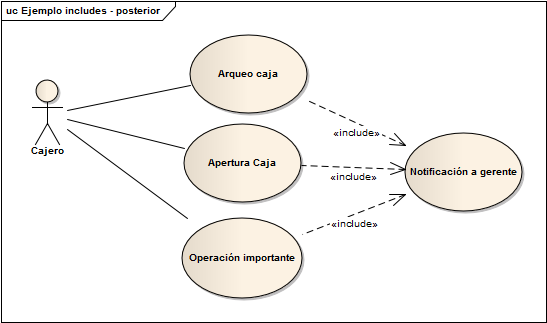
## Inclusión de casos de uso (“includes”)



Se suele utilizar la relación de inclusión entre casos de uso cuando hay comportamiento común en varios casos de uso base.

Ejemplo: Los tres casos de uso base incorporan la funcionalidad de “Notificar al Gerente de la Sucursal”.





## 

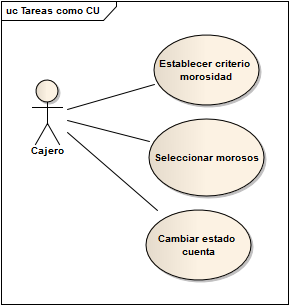
## 

## 

## 

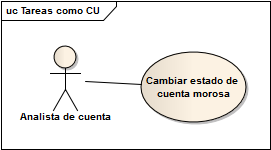
## Tareas vs Casos de Uso

Una tentación al comenzar a armar diagramas de caso de uso es asociar cada tarea como un caso de uso, de la siguiente manera:



**Ejemplo incorrecto**: Tareas documentadas como casos de uso

No obstante, el diagrama de casos de uso *no es* un diagrama de flujo. Lo que hay que mostrar son las funcionalidades que le dan valor agregado al usuario. El mismo ejemplo de arriba está especificando un solo caso de uso:



## 

## 

## Resumen

**¿Quién hace el diagrama de casos de uso?**

El analista funcional, en base a los relevamientos hechos con el usuario final.

**¿Para quién es el diagrama de casos de uso?**

Para el usuario final, ya que se capturan todos los requerimientos funcionales que él pide y se dejan registrados como una especie de *contrato* de lo que el sistema incluye y lo que no.

Para los analistas funcionales y los técnicos, el diagrama de casos de uso permite una visión global del sistema. Luego, parte de la negociación entre el usuario y el analista funcional en una metodología de trabajo iterativa es definir las fechas de entrega para cada uno de los casos de uso.

**¿Cuándo queremos hacer un diagrama de casos de uso?**

Siempre que comenzamos el desarrollo de un software y que debemos acordar con el usuario las funcionalidades de un sistema.

# Especificación de un caso de uso

Aunque no hay un estándar UML para especificar un caso de uso, la mayoría de los modelos para documentar en detalle cada caso de uso tiene un formato similar al siguiente:

* **Nombre del caso de uso:** la misma denominación que fue dada en el diagrama de casos de uso, se recomienda verbo y sustantivo.
* **Objetivo o Propósito:** representa el resultado observable y valioso para el usuario. Cada caso de uso debe tener un solo objetivo (*cohesión*).
* **Actores**: quienes intervienen en el caso de uso.
* **Trigger**: cuál es la acción que dispara el caso de uso.
* **Pre-Condiciones:** no son las causas que disparan el caso de uso (que en todo caso pueden definirse en la sección *Trigger*), sino el estado en el que las cosas deben estar para que el caso de uso se ejecute satisfactoriamente. Aquí se pueden anotar una lista de condiciones a cumplir o bien de casos de uso a ejecutar previamente.
* **Escenario:** se define una serie de pasos para completar la ejecución normal de un caso de uso: debe quedar claro qué actor es responsable de resolver cada uno de los pasos.
* paso 1: El Actor X realizará la operación Y.
* paso 2: El Actor Z realizará la operación T.

Cada paso puede derivar en flujos alternativos o excepciones.

¿Cómo documentar la relación “includes”? El caso de uso incluído es la tarea n del escenario. La tarea n-1 del caso de uso base será la precondición del caso de uso base.

¿Cómo documentar la relación “extends”? En la sección “Escenarios alternativos”…por cada caso de uso que es extensión del caso de uso base habrá un flujo alternativo.

* **Escenarios alternativos:** aquí se define una serie de pasos alternativos al caso de uso original, tanto por extensiones al caso de uso base como en el caso de las excepciones.
* **Post-Condiciones:** es el estado en el que debe quedar el sistema tras la ejecución exitosa del caso de uso. Es algo tan simple como el cumplimiento del objetivo del caso de uso.

## Ejemplo Salón de Videojuegos

Tomaremos el siguiente ejemplo para especificar el casos de uso Canjear premios

(como ejemplo también para documentar la relación “extends”)

| **Nombre** | Canjear premios |
| --- | --- |
| **Objetivo** | Canjear los puntos obtenidos en juego por premios |
| **Actor** | Cajero, Cliente |
| **Trigger** | Cajero pasa la tarjeta por la lectora |
| **Pre-condición** | El cliente debe tener una tarjeta en buen estado y puntos en premios acumulados |
| **Escenario** | 1. El cajero pasa la tarjeta por la lectora y le informa al cliente la cantidad de puntos acumulados. 2. El cliente selecciona el premio |
| **Flujo Alternativo** | 1. Flujo alternativo: el cliente selecciona un premio que no está en el catálogo. “Envío Premio a Domicilio” |
| **Post-condición** | El cajero entrega el premio al cliente y actualiza la información de la cantidad de puntos que el cliente tiene en su tarjeta. |

| **Nombre** | Envío Premio a Domicilio |
| --- | --- |
| **Objetivo** | Enviar a un cliente uno o varios premios que no se encuentren en una sucursal |
| **Actor** | Cajero, Cliente |
| **Trigger** | El cliente solicita premio a domicilio |
| **Pre-condición** | El cliente debe tener una tarjeta en buen estado y puntos en premios acumulados.  El premio que solicita el cliente no debe estar disponible en la sucursal. |
| **Escenario** | 1. El cajero toma los datos del domicilio al cliente. 2. Se despacha la orden de entrega del premio |
| **Post-condición** | El cajero actualiza la información de la cantidad de puntos que el cliente tiene en su tarjeta. En el transcurso de las 72 hs. se entrega el premio al cliente. |