

Seminario 3. Casos de uso

Ingeniería de Requisitos – Curso 2023 / 2024

Departamento de Informática

Universidad de Oviedo

¿Qué es un caso de uso?

Un caso de uso es la **descripción** de una **funcionalidad** del sistema, y cómo interactúa dicho sistema con actores externos o actores.

¿Qué es un escenario?

Cada **secuencia de interacciones** posible de un caso de uso es un escenario. Un caso de uso, por tanto, contiene todos los escenarios posibles para que el actor primario consiga su objetivo.

¿Qué es un actor?

Una persona, grupo de personas o incluso otro sistema. Cada caso de uso está asociado con el objetivo de un actor, que es el actor primario.

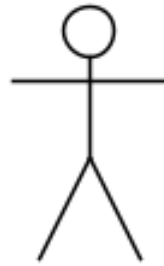
¿Stakeholders vs actores?

Todos los actores son stakeholders, pero no todos los stakeholders interactúan con el sistema y no son, por tanto, actores.

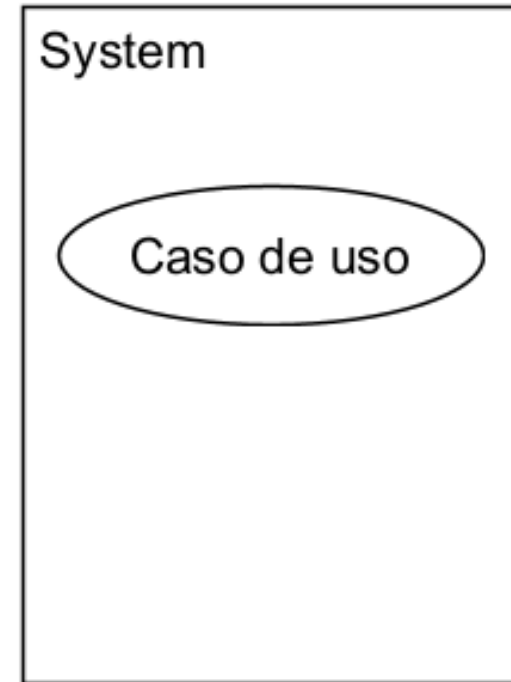
¿Cómo escribir un caso de uso?

Nombre	
Descripción breve	
Precondiciones	
Postcondiciones	
Situaciones de error	
Actores	
Disparador	
Proceso estándar	
Procesos alternativos	

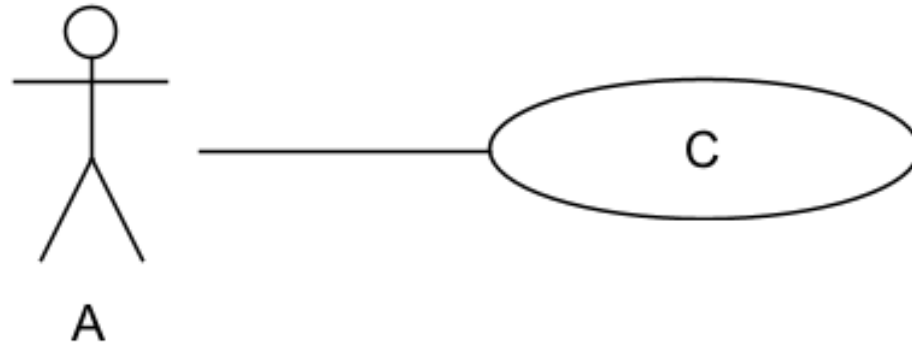
Elementos de un diagrama de casos de uso



Nombre del actor

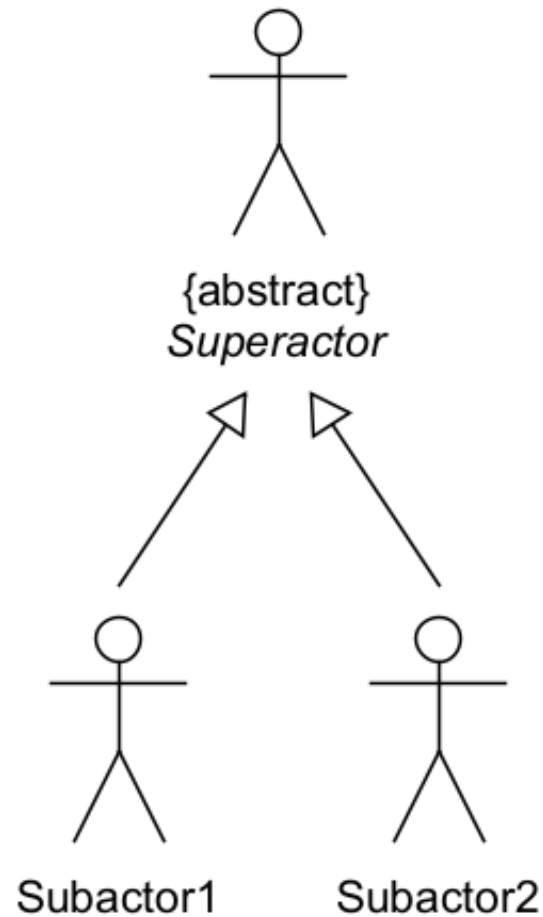


Relación entre actor y caso de uso



Relación de asociación entre el actor **A** y el caso de uso **C**. Representa que el actor interactúa con ese caso de uso.

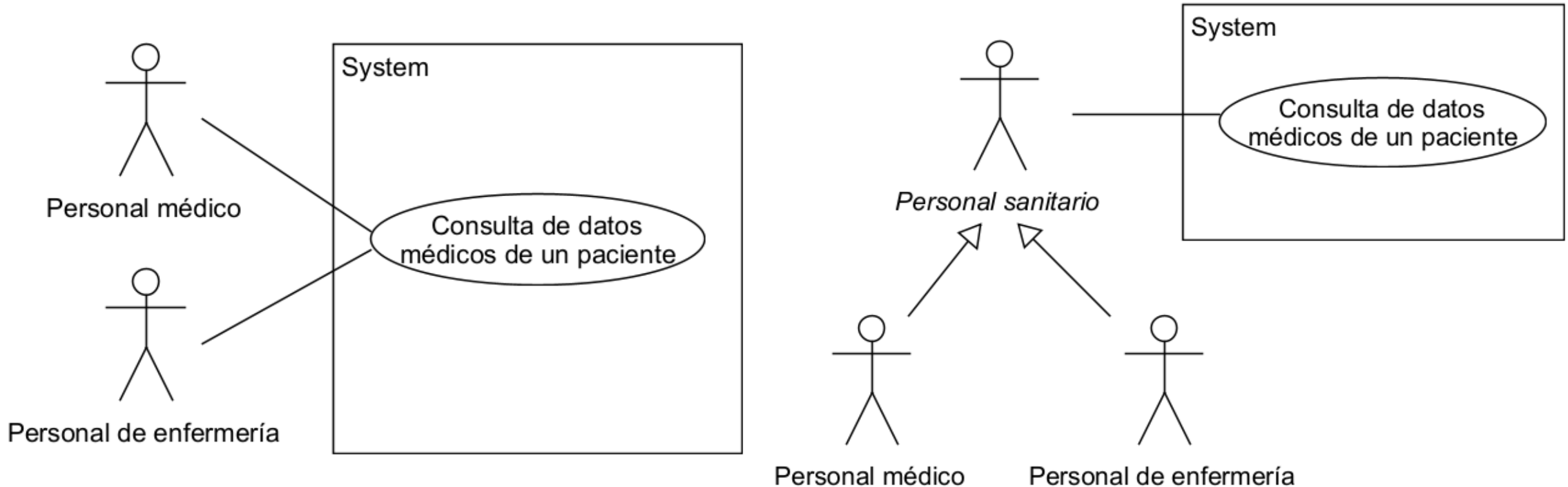
Relaciones entre actores



Es posible tener relaciones de generalización con un superactor abstracto o no abstracto.

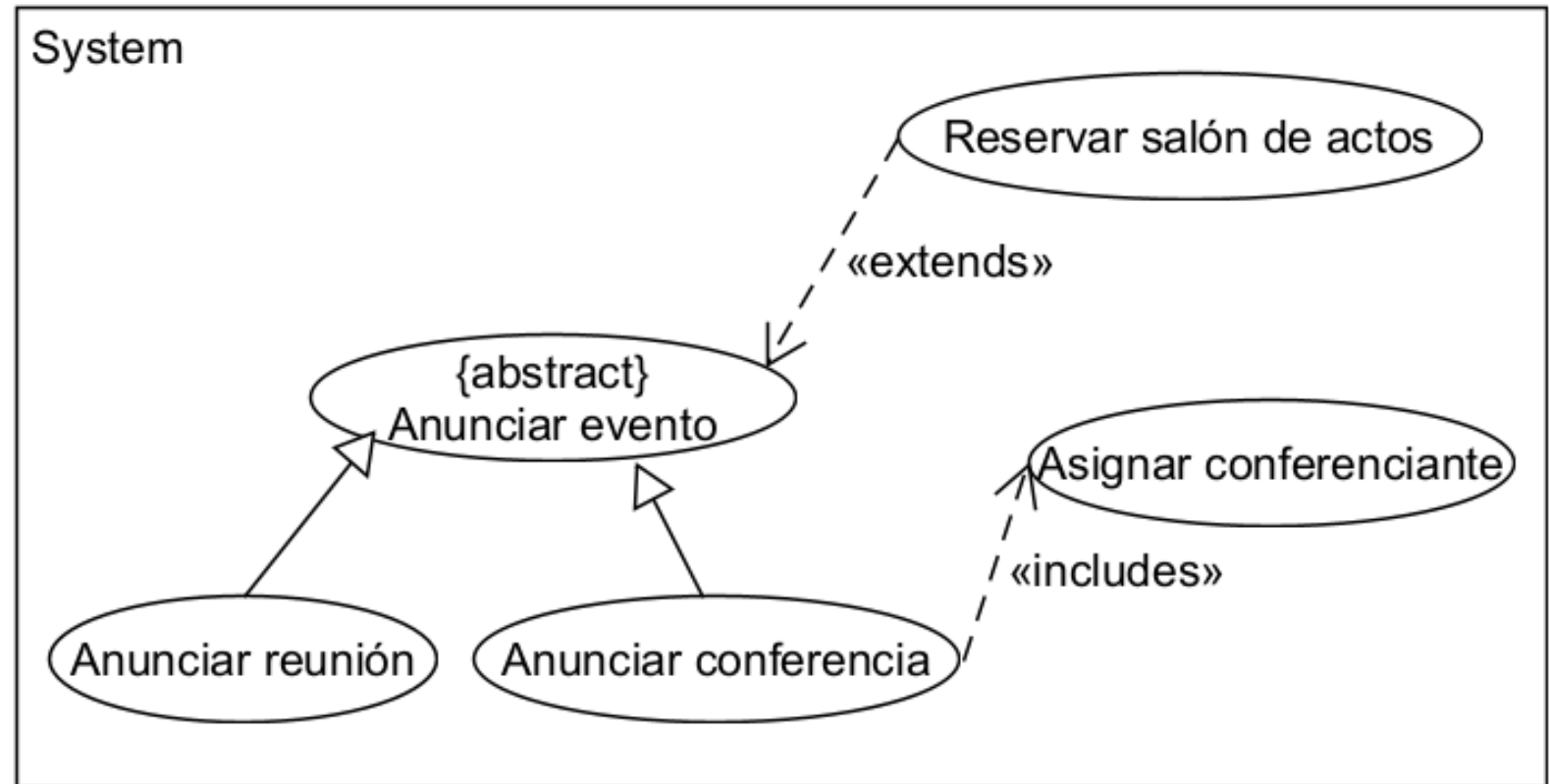
Si no existe una instancia del superactor será abstracto y se escribe en cursiva o con la palabra {abstract}.

¿Cuál es la diferencia entre estos diagramas?

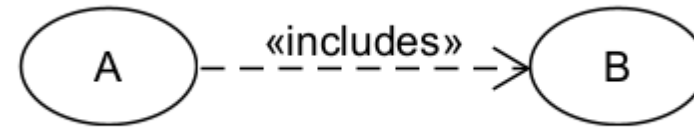


Solución: En el primero los dos actores deben participar en el caso de uso a la vez. En el segundo diagrama participan en el caso de uso de forma individual.

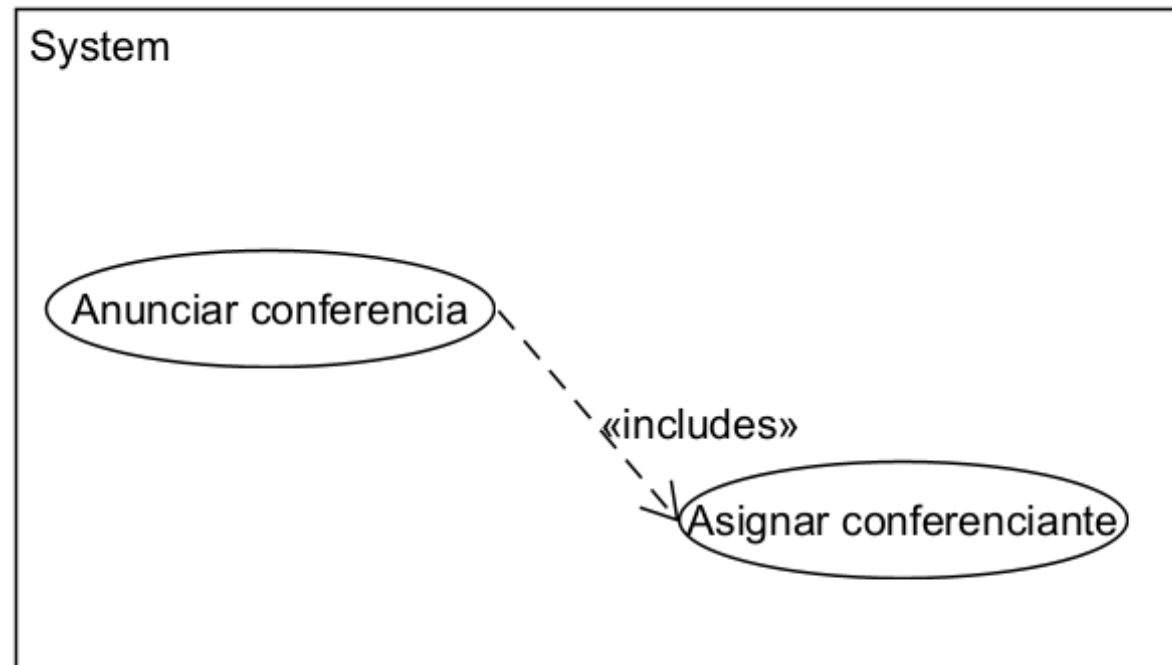
Relaciones entre casos de uso



Includes



¡Ojo con el sentido de la flecha!

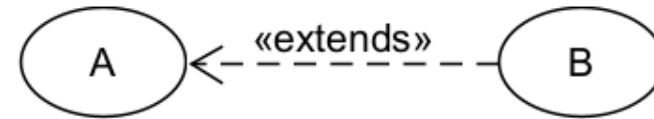


A es el caso **base** y B es el caso **incluido**

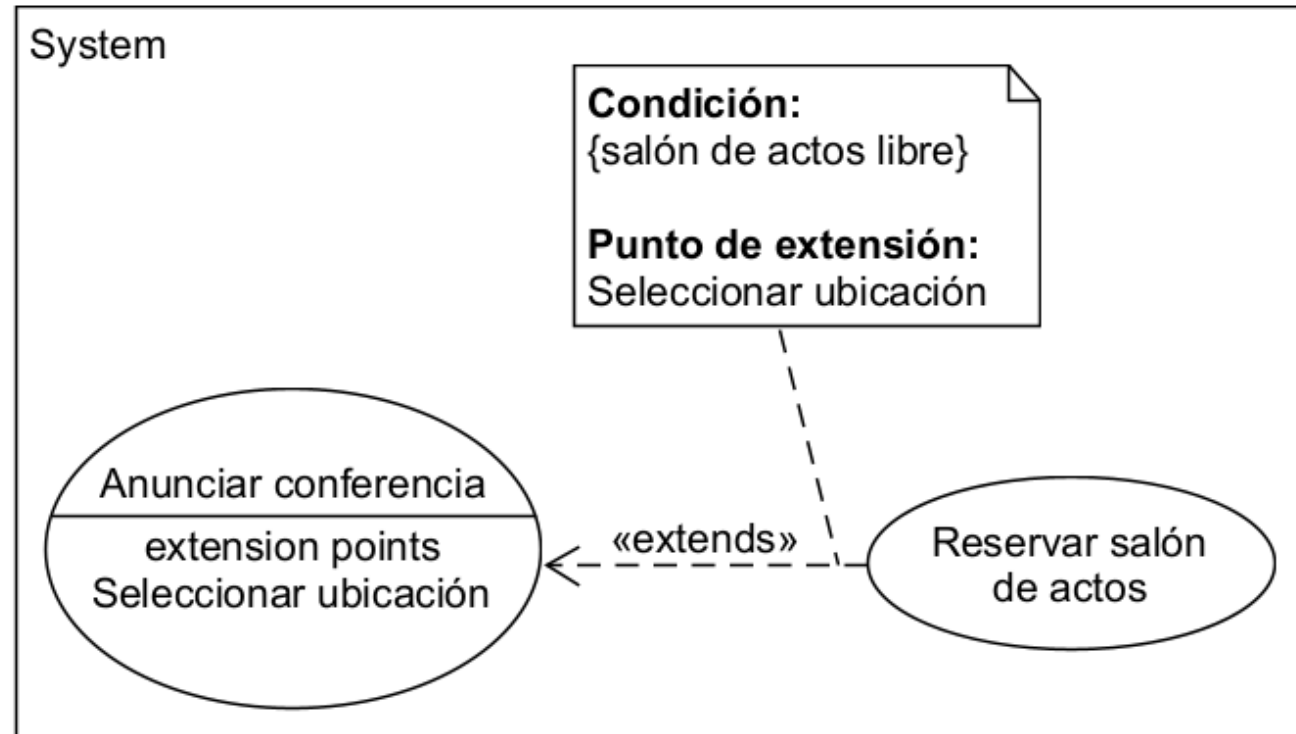
El caso base siempre necesita el incluido para ofrecer su funcionalidad.

El caso incluido puede ejecutarse por separado del caso base.

Extends



¡Ojo con el sentido de la flecha!

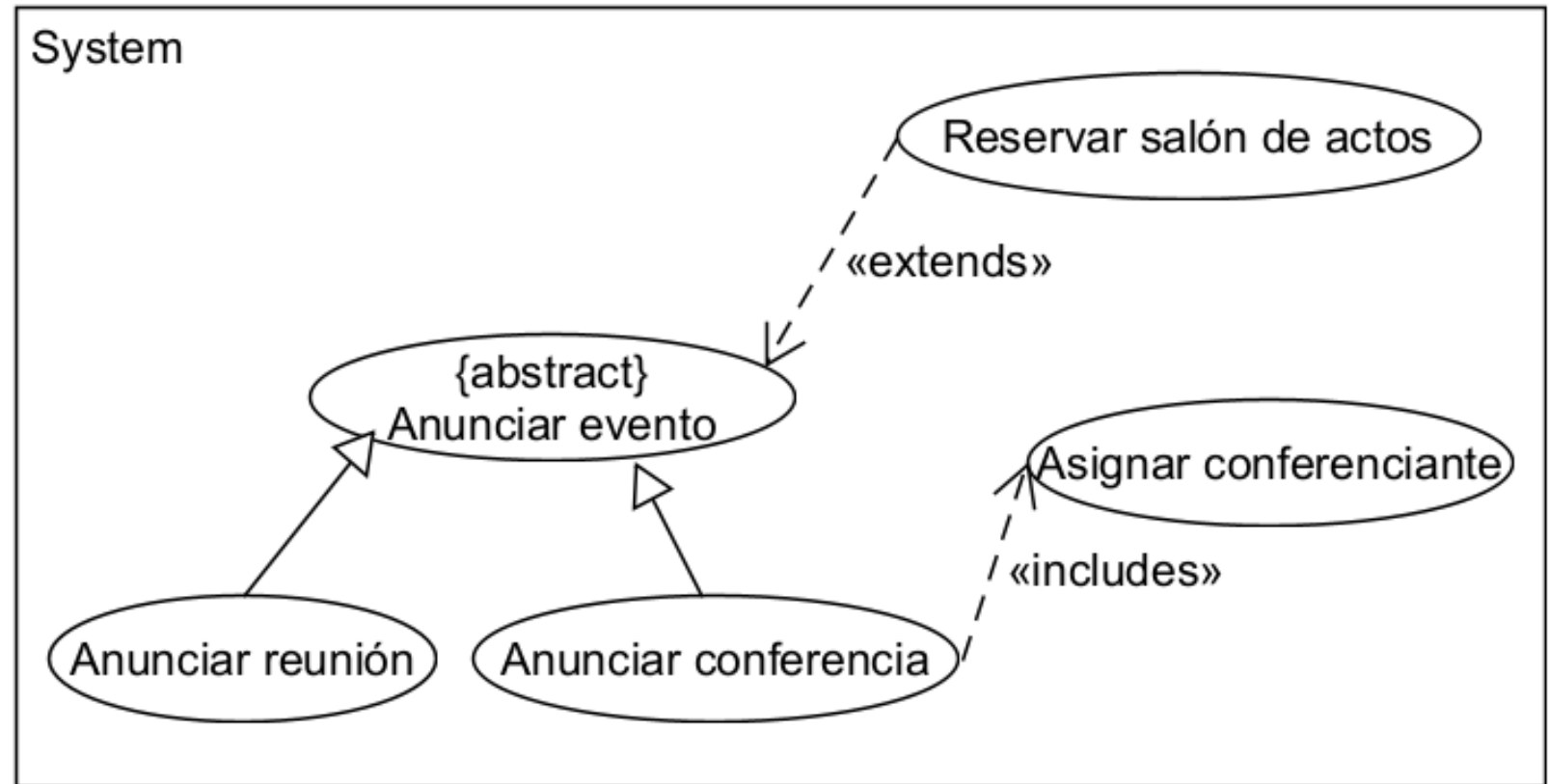
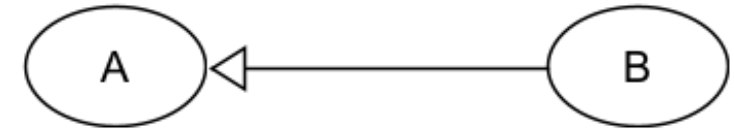


A es el caso **base** y B es el caso **extendido**

El caso base no necesita el extendido para ofrecer su funcionalidad.

Ambos se pueden ejecutar por separado.

Generalización



B hereda el comportamiento de A

Escribir un caso de uso

Un cliente reserva en un restaurante una mesa a través de la página web. El formulario web solicita una fecha, hora y número de personas.

Nombre	
Descripción breve	
Precondición	
Postcondición	
Situaciones de error	
Estado del sistema si ocurre algún error	
Actores	
Disparadores	
Proceso estándar	
Procesos alternativos	

Solución: reserva en un restaurante

Nombre	Reservar una mesa en un restaurante.
Descripción breve	Un cliente reserva una mesa en un restaurante.
Precondición	El cliente ha iniciado sesión en la página web del restaurante.
Postcondición	Se ha reservado una mesa en el restaurante.
Situaciones de error	No hay mesas libres.
Estado del sistema si ocurre algún error	No se ha reservado mesa en el restaurante.
Actores	Cliente.
Disparadores	El cliente solicita una mesa en un restaurante.

Proceso estándar	<p>(1) El cliente solicita una reserva para una fecha, hora y número de personas concreto.</p> <p>(2) El sistema confirma que el restaurante tiene una mesa disponible.</p> <p>(3) El cliente confirma la reserva.</p>
Procesos alternativos	<p>(2') El restaurante no tiene ninguna mesa disponible a esa hora.</p> <p>(3') El sistema propone una hora alternativa.</p> <p>(4') El cliente selecciona la hora alternativa y confirma la reserva.</p> <p>(2'') El restaurante no tiene ninguna mesa disponible en esa fecha.</p> <p>(3'') El sistema propone una fecha alternativa.</p> <p>(4'') El cliente selecciona la fecha alternativa y confirma la reserva.</p> <p>(2''') El restaurante no tiene ninguna mesa disponible para ese número de personas.</p> <p>(3''') El sistema propone una fecha alternativa.</p> <p>(4''') El cliente selecciona la fecha alternativa y confirma la reserva.</p> <p>(2''') El restaurante no tiene ninguna mesa disponible para la fecha, hora o número de personas.</p> <p>(3''') El sistema propone una fecha alternativa.</p> <p>(4''') El cliente cancela el proceso de reserva.</p>

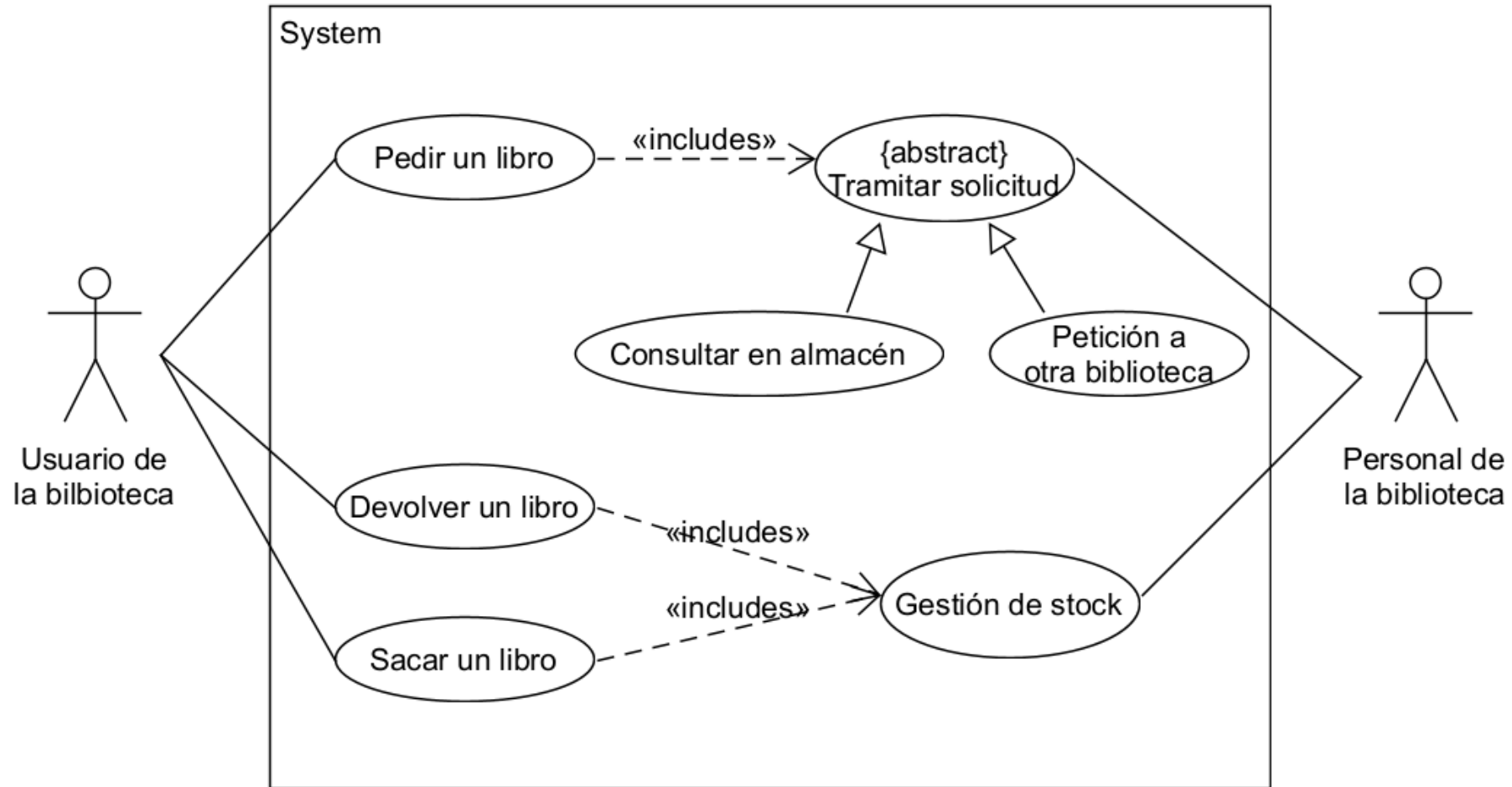
Diagrama de casos de uso: Biblioteca

Dibuja un diagrama de casos de uso que represente el funcionamiento de una biblioteca.

Un usuario de la biblioteca puede sacar un libro, pedir un libro y devolver un libro.

El personal de la biblioteca puede buscar un libro, tramitar una solicitud de un libro (porque el usuario lo pida o porque la biblioteca lo decida) y gestionar los libros que salen y entran de la biblioteca.

Solución 1



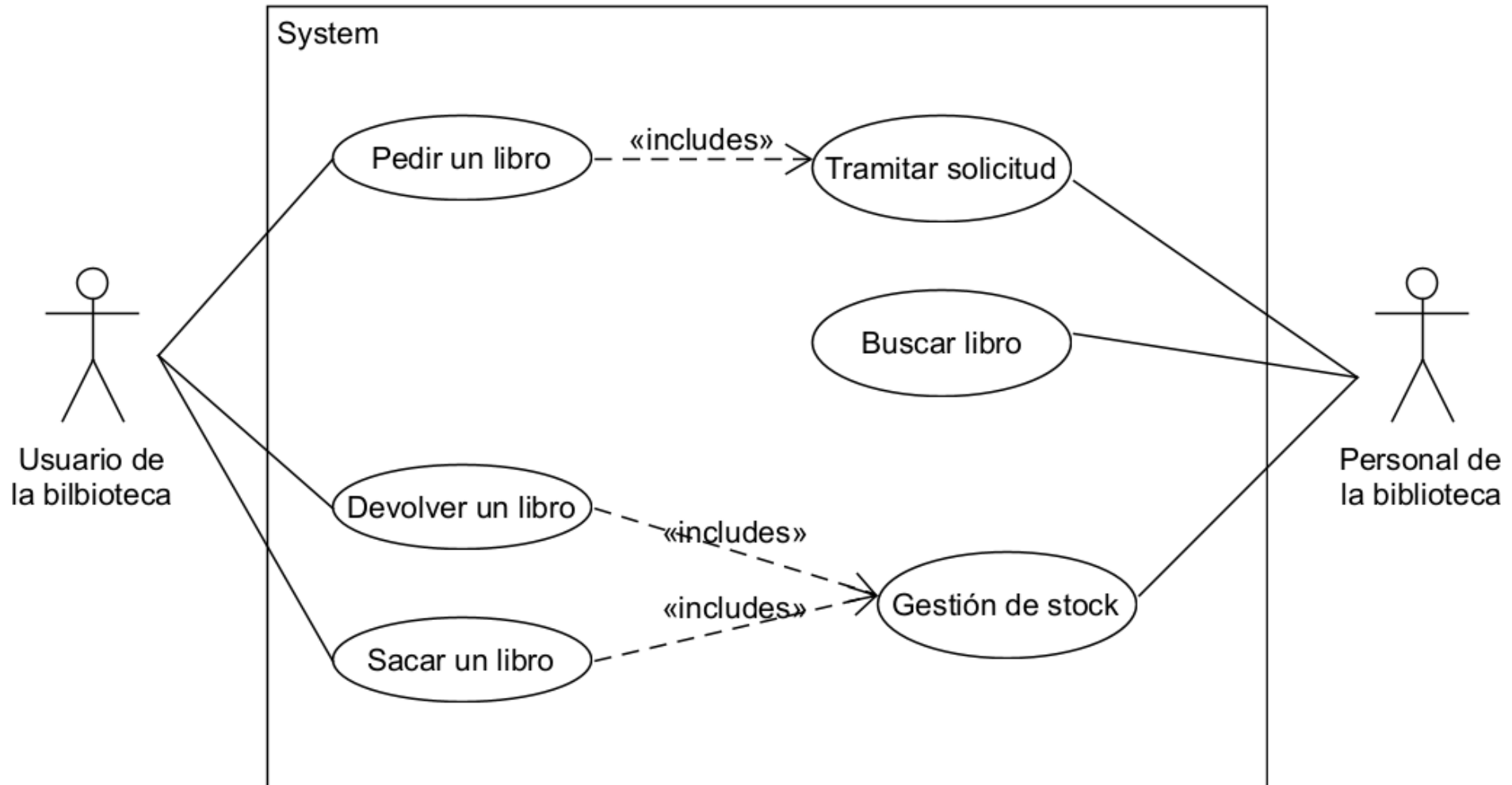
Notas sobre la solución 1

La relación entre sacar un libro y devolver un libro y la gestión de stock es de tipo **includes** ya que el personal de la biblioteca siempre va a gestionar el stock cuando se devuelvan o saquen libros, pero también se puede realizar esta funcionalidad por separado, por ejemplo, cuando llega stock extra de algún libro. Esta relación no podría ser de tipo **extends** porque **siempre** se debe gestionar el stock al hacer estas operaciones.

La relación entre pedir un libro y tramitar solicitud es de tipo **includes** porque para completar el caso de uso de pedir un libro es necesario que el personal de la biblioteca tramite una solicitud. A su vez el personal de la biblioteca puede tramitar solicitudes sin que un usuario lo pida.

Se ha añadido caso de uso tramitar solicitud como **abstracto** para que se pueda o solicitar un libro a otra biblioteca o consultar en el almacén (buscar libro) y que siempre que se pida un libro se tenga que hacer una de las dos opciones.

Solución 2



Notas sobre la solución 2

En esta solución se entiende el caso de uso buscar libro como una funcionalidad que realiza el personal de la biblioteca y que no está unida a otro caso de uso. Dado que el enunciado deja esto abierto, podría ser otra posible solución para este diagrama.

Bibliografía

Alistair Cockburn. 2000. Writing Effective Use Cases (1st. ed.). Addison-Wesley Longman Publishing Co., Inc., USA.

Seidl, Martina & Scholz, Marion & Huemer, Christian.
(2015). UML @ Classroom. <https://doi.org/10.1007/978-3-319-12742-2>