

Entornos de Desarrollo

Ejercicio para entregar UD12

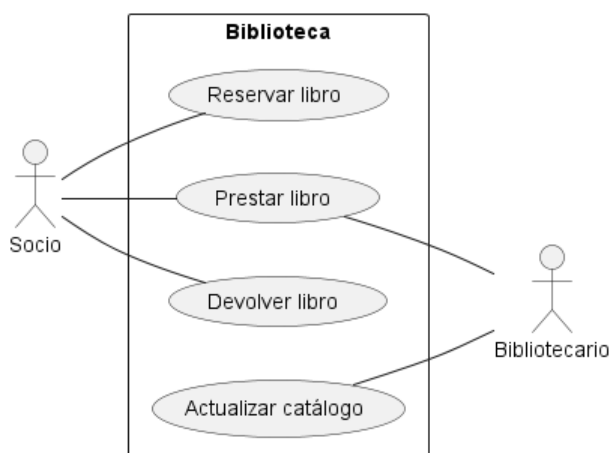
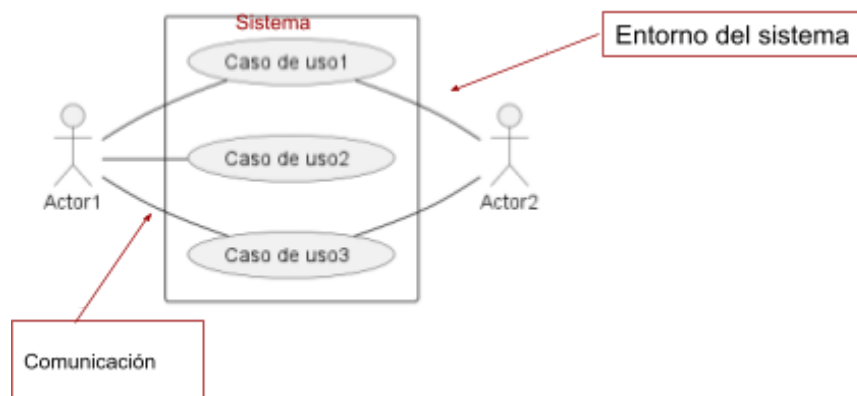
Modelado conceptual Diagrama de Casos de Uso

Objetivos:

- Modelar el comportamiento de un sistema identificando los principales requisitos funcionales.
- Aprender un subconjunto de UML, como notación de modelado OO
 - Notación del Diagrama de Casos de Usos.

Los *casos de uso* capturan los requisitos funcionales del sistema a desarrollar. Los elementos (o notación) que presentan son:

- Casos de uso. Conjunto de requisitos funcionales.
- Actor. Interactúan con el sistema.
- Comunicación. Interacción de los actores con los casos de uso.
- Entorno del sistema.
- Entorno. Software que vamos a utilizar.



En el ejemplo tenemos dos actores, el socio y el bibliotecario. En el caso del socio, puede acceder a la funcionalidad que proporciona la biblioteca de reservar un libro, prestarlo o devolverlo. El bibliotecario por su parte puede acceder a la funcionalidad de prestar un libro y actualizar el catálogo.

Cada *caso de uso* se puede describir en una plantilla de lenguaje natural como la siguiente:

<u>Caso de Uso</u>	
Actores	
Resumen	
Precondiciones	
Postcondiciones	
Incluye	
Extiende	
Hereda de	
<i>Flujo de Eventos</i>	
Actor	Sistema

En el *caso de uso* de Reservar Libro, la plantilla quedaría de la siguiente forma:

<u>Caso de Uso</u>	Reservar Libro
Actores	Socio
Resumen	El socio puede solicitar la reserva de un libro para su posterior préstamo, a partir de una fecha determinada.
Precondiciones	El socio no tiene ninguna reserva.
Postcondiciones	El socio tiene una reserva y el libro tiene una nueva reserva a partir de una fecha.
Incluye	--
Extiende	--
Hereda de	--
<i>Flujo de Eventos</i>	
Actor	Sistema
1. El socio solicita la reserva (código,	

libro, fecha).	2. El sistema comprueba que el socio no tiene reserva. 3. El sistema comprueba que el libro está libre para la fecha solicitada. 4. El sistema solicita confirmación de la reserva.
5.El socio confirma la reserva.	6. El sistema realiza la reserva.

Las relaciones entre los *casos de uso* son:

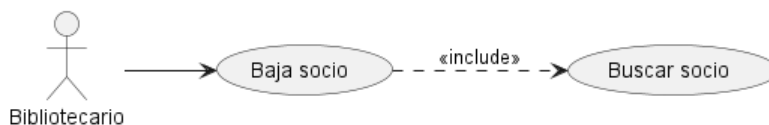
- Inclusión.
- Extensión.
- Herencia.

Además, la herencia actúa de relación entre actores.

Relaciones:

Inclusión:

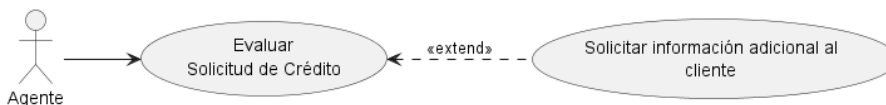
Un *caso de uso A* **incluye** a un *caso de uso B*, si una instancia de **A** puede realizar todos los eventos que aparecen descritos en **B**.



La **inclusión** se representa con una flecha discontinua y la etiqueta “<<include>>”.

Extensión:

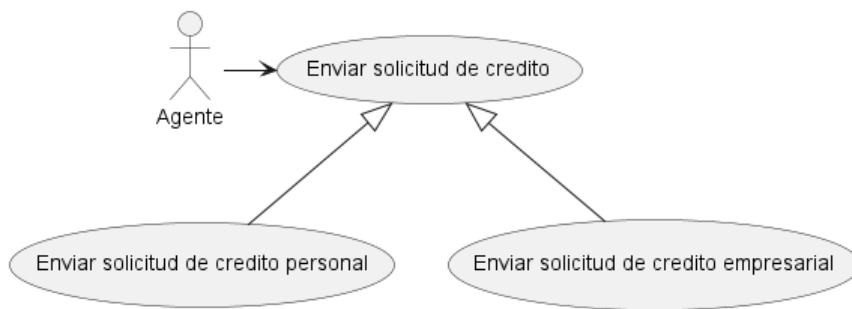
Un *caso de uso B* **extiende** a un *caso de uso A*, si en la descripción de **A** figura una condición cuyo cumplimiento origina la ejecución de todos los eventos que aparecen descritos en **B**.



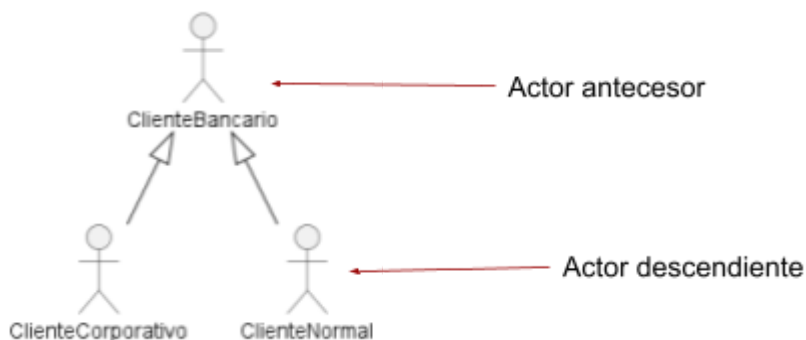
La **extensión** se representa con una flecha discontinua y la etiqueta “<<extend>>”.

Herencia:

Un *caso de uso B* **especializa** a un *caso de uso A*, si el flujo de eventos de **B** es un refinamiento del flujo de eventos de **A**.



En el caso de los actores, solo es posible establecer entre ellos una relación de **herencia**.



La **herencia** se representa con una flecha continua con la punta blanca.

Diagrama de Casos de Uso:

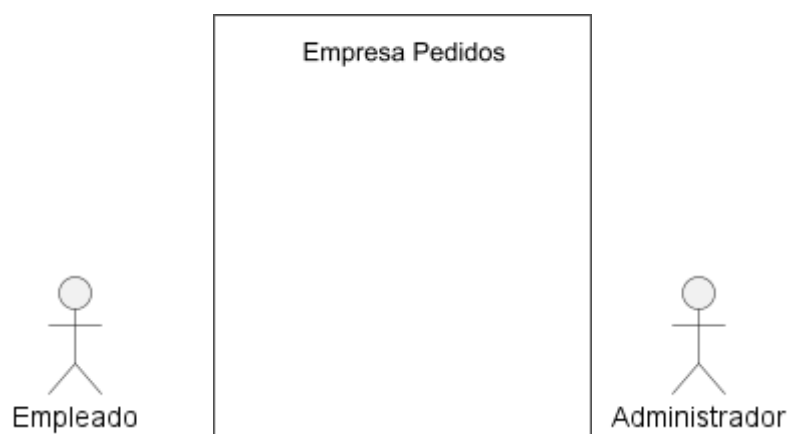
Estructura:

Se estructuran en tres niveles:

- Diagrama de contexto y Diagrama inicial.
- Plantillas de descripción.
- Diagrama estructurado o Modelo de casos de uso.

Diagrama de contexto:

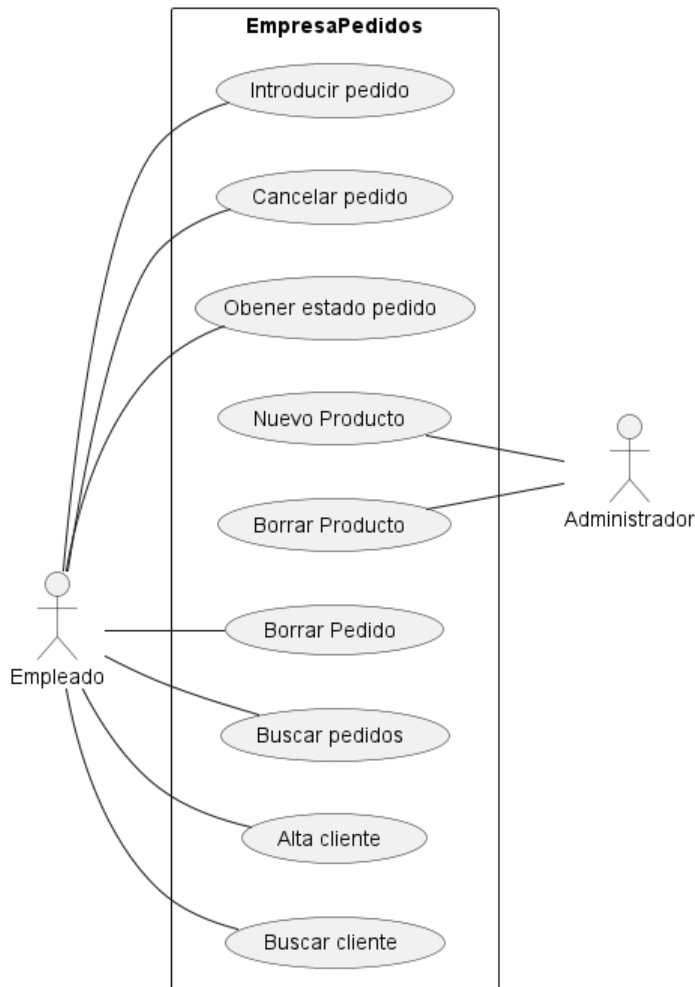
Sirve para identificar los límites del sistema. Una vez identificado el sistema software que vamos a utilizar, podemos identificar los actores que van a interactuar con el sistema.



En este caso, el sistema es una empresa de pedidos y los actores son un empleado y un administrador.

Diagrama inicial:

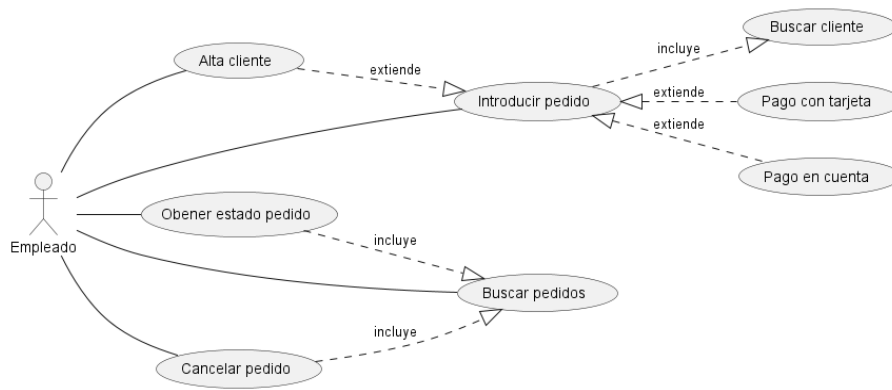
Refina el diagrama de contexto e indica los principales *casos de uso* y funcionalidades del sistema.



En el diagrama inicial definimos los *casos de uso* con los que interactúa el empleado y el administrador.

Diagrama estructurado o modelo de casos de uso:

Refina aún más el diagrama inicial y establece las relaciones de inclusión y extensión entre los *casos de uso*.



Conclusiones:

- El diagrama de casos de uso permite especificar las principales funcionalidades que el sistema ofrece a los actores.
- UML proporciona una notación para representar dicho diagrama.
 - Casos de uso y actores.
 - Relaciones entre casos de uso:
 - Inclusión.
 - Extensión.
 - Herencia.
 - Relaciones entre actores:
 - Herencia.
- Cada caso de uso se especifica en una plantilla de descripción textual.