

Análisis de Sistemas

Materia:
Ingeniería de
Requerimientos

Docente contenidista: PEPE, Jonathan Leonel

Revisión: Coordinación

Contenido

Modelo de comportamiento	4
Casos de uso	5
Historia	5
Concepto	5
Utilidad	5
Componentes	6
Actores	6
Identificación de actores.....	7
Buenos y malos actores	8
Casos de uso propiamente dicho	10
Identificación de casos de uso	11
Buenos y malos casos de uso	11
Relaciones	14
Relaciones de asociación	14
Relaciones de dependencia (inclusión)	15
Relaciones de dependencia (extensión).....	16
Relaciones de generalización/especialización.....	17
Diagramas de casos de uso.....	18
Actores temporales.....	19
Bibliografía	21

Clase 7



iTe damos la bienvenida a la materia
Ingeniería de Requerimientos!

En esta clase vamos a ver los siguientes temas:

- Componentes principales de los casos de uso.
- Actores principales, secundarios y temporales.
- Tipos de relaciones.
- Diagrama de casos de uso.

¿Listos para continuar?

En la clase anterior conocimos cómo es la estructura de un documento SRS y su importancia en la Ingeniería de requerimientos.

Hoy estudiaremos uno de los diagramas que formará parte del SRS, estamos hablando del diagrama de casos de uso.

¡Seguimos adelante!

Modelo de comportamiento

El **modelo de comportamiento** muestra, bajo la forma de acciones y reacciones, el comportamiento de un sistema desde el punto de vista del usuario.

Es decir, se muestra lo que sucede o acontece cuando un sistema responde a un estímulo de su entorno.

El objetivo de este modelo es describir la **funcionalidad** del sistema a desarrollar independientemente de la implementación.

Sus componentes son:

1. **Diagrama de casos de uso.**
2. Historias de usuario.

En esta clase estudiaremos el primero de ellos.

Casos de uso

Historia



La existencia de los casos de uso es anterior a la creación de **UML** (*Unified Modeling Language*, lenguaje de modelado de sistemas más conocido y utilizado en la industria de software).

Los casos de uso fueron creados por Jacobson en 1987. Luego, los incorporó Booch y Rumbaugh para la creación de UML. Y actualmente forman parte de los elementos con los que trabaja UML.

Concepto

Los casos de uso son descripciones de un conjunto de secuencia de acciones que ejecuta el sistema para obtener un resultado.

*Especifican un comportamiento deseado,
NO cómo se ejecuta.*

Es decir, describen lo que hace el usuario y lo que hace el sistema, poniendo énfasis en la interacción desde el punto de vista del usuario.

Utilidad

Respecto a la utilidad de los casos de uso, podemos mencionar:

- Documentar las funciones del sistema (*qué debería hacer*) y los roles de los actores intervinientes (*quiénes interactúan con él*).
- Permitir que los desarrolladores y los clientes **lleguen a un acuerdo sobre los requisitos del sistema**.
- Generar la documentación de usuario y las pruebas funcionales del sistema, en paralelo con el desarrollo.

Componentes

Se compone de los siguientes elementos:

- **Actores:** son quienes interactúan con el sistema.
- **Casos de uso:** son las funciones requeridas al sistema.
- **Diagramas de casos de uso:** son las relaciones entre los casos de uso y los actores.

A continuación, veremos cada uno de estos componentes en detalle.

Actores

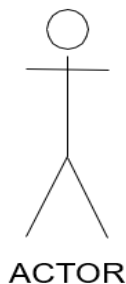


Los **actores** representan a algo o alguien (*una persona, un departamento u otro sistema*) que interactúa con el sistema **sin ser parte de él**. Es decir, los actores interactúan con el sistema, pero son externos al mismo.

Puede realizar lo siguiente:

- Ingresar información al sistema.
- Recibir información del sistema.
- Ingresar y recibir información al/del sistema.

A los actores se los representa de la siguiente manera:



Identificación de actores

Para una correcta identificación de actores se aconseja pensar en las respuestas a las siguientes preguntas:

- ¿Quién está interesado en cierto requerimiento?
- ¿En qué lugar de la organización se usa el sistema?
- ¿Quién se beneficiará con el uso del sistema?
- ¿Quién intercambiará información con el sistema?
- ¿Quién mantendrá el sistema (ABM)?
- ¿Juega una persona diferentes roles?
- ¿Juegan varias personas el mismo rol?
- ¿Interacciona el Sistema con otros sistemas?

*La clave para identificar buenos actores es pensar en el **ROL** con el cual el actor interacciona con el sistema.*

Buenos y malos actores

Supongamos el siguiente caso:

Caso I:

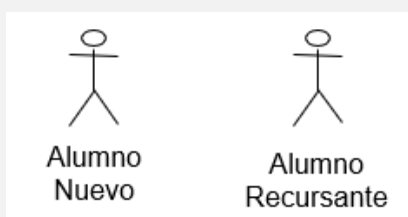
"Los cursos podrán tener alumnos nuevos y también alumnos recursantes".

Entonces...

¿Es conveniente este modelo?



¿O este modelo?



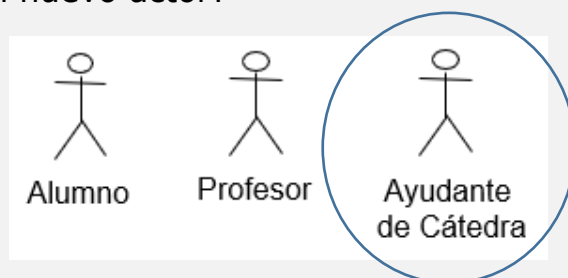
Supongamos ahora otro caso:

Caso II:

"Los alumnos podrán ser también ayudantes de cátedra, en cuyo caso deberán inscribirse en el sistema como alumnos y también como profesores".

Entonces...

¿Necesitamos un nuevo actor?





Luego de pensarlo, veamos la respuesta:

Caso I:

El alumno nuevo interacciona con el sistema (juega el mismo ROL) que el alumno recurrente, entonces **no es necesario desglosar en dos actores.**

Caso II:

El ayudante de cátedra cuando se inscribe como alumno interacciona como cualquier alumno y cuando se inscribe como profesor interacciona como cualquier profesor, entonces **no es necesario un nuevo actor.**

Por lo tanto, en para un sistema de inscripción de alumnos y profesores en una facultad, los actores posibles son:



Alumno



Profesor



Dto.Alumnos



Sist.Facturación

Casos de uso propiamente dicho



Los **casos de uso** describen un conjunto de secuencias de acciones, incluidas variaciones, que ejecuta un sistema para producir un resultado de valor para un actor. Es decir, especifican el comportamiento de una parte del sistema, representando así un **requisito funcional** del mismo.

A los casos de uso se los respetan de la siguiente manera:

Nombre del
caso de uso

La sintaxis para nombrar un caso de uso es **verbo + sustantivo**.

Ejemplo de caso de uso para el sistema de inscripción de alumnos y profesores:

Consultar Curso

Este caso de uso representa la funcionalidad que provee el sistema al actor alumno. El sistema provee al actor alumno **la funcionalidad** de consultar el catálogo de cursos.

Identificación de casos de uso

Para una correcta identificación de casos de uso se aconseja pensar en las respuestas a las siguientes preguntas:

- ¿Cuáles son las tareas de cada actor?
- ¿Qué casos de uso crearán, almacenarán, modificarán, consultarán o eliminarán información?
- ¿Algún actor necesita informar al sistema sobre cambios externos?
- ¿Necesita algún actor ser informado de ciertos cambios en el sistema?
- ¿Qué casos de uso mantendrán actualizado el sistema?
- **¿Todos los requerimientos funcionales están representados en los casos de uso?**

Buenos y malos casos de uso

Existe dos reglas básicas que se deben respetar:

- **Complejidad temporal:** un caso de uso debe ser completo de principio a fin.
- **Complejidad funcional:** cuando varios casos de uso tratan con la misma entidad y son iniciados por el mismo actor podemos resumirlos en un único caso de uso.



Supongamos el siguiente caso:

Caso I:

"cuando un alumno se inscribe primero selecciona el curso, luego se registra y al terminar el sistema envía los datos al sistema de facturación externo".



¿Por qué es incorrecto el modelo de la izquierda?
¿Qué regla se está aplicando?

Caso II:

"el departamento de alumnos es el responsable de agregar, modificar o eliminar cursos del catálogo".



¿Por qué es incorrecto el modelo de la izquierda?
¿Qué regla se está aplicando?

Por lo tanto, en para un sistema de inscripción de alumnos y profesores en una facultad, los casos de uso posibles son:

- Registrar curso a tomar.
- Consultar cursos.
- Registrar curso a dictar.
- Consultar alumnos inscriptos.
- Mantener alumnos.
- Mantener profesores.
- Mantener materias.
- Mantener cursos.

Relaciones

Los tipos de relaciones posibles son:

- **Relaciones de asociación** (entre actores y casos de uso).
- **Relaciones de dependencia** (entre casos de uso).
 - **Relaciones de inclusión.**
 - **Relaciones de extensión.**
- **Relaciones de generalización/especialización** (entre actores y casos de uso).

A continuación, veremos cada una de ellas.

Relaciones de asociación

La relación entre actores y casos de uso se representa a través de una **relación de asociación**, también llamada relación de comunicación. La misma modeliza una "*comunicación*" entre el actor y el caso de uso.

Dicha comunicación implica que (**sentido de la flecha**):

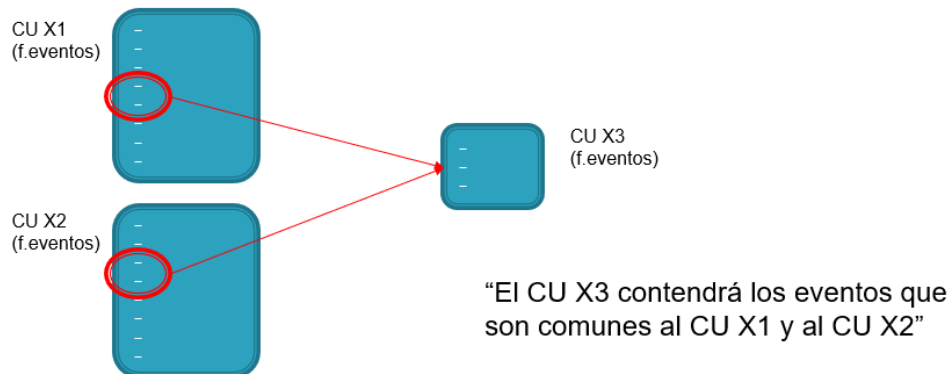
- El actor envía información al caso de uso:
- El actor recibe información del caso de uso:
- El actor envía y recibe información del caso de uso:



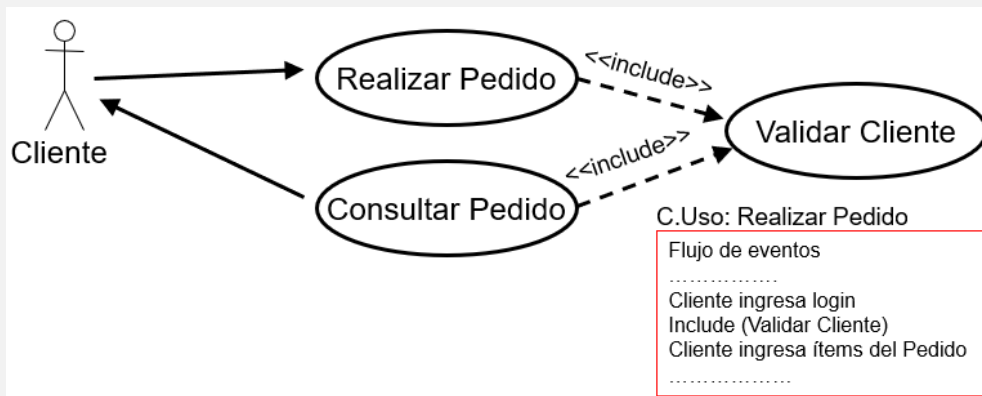
Relaciones de dependencia (inclusión)

La **inclusión** se utiliza para evitar repetir, en varios casos de uso, el mismo flujo de eventos. Para lograrlo, se “*llevar*” los casos de uso comunes a un nuevo caso de uso común a ambos.

Lo que haremos será reutilizar el comportamiento del CU (caso de uso) incluido desde varios CU base, tal como se muestra en la siguiente imagen:



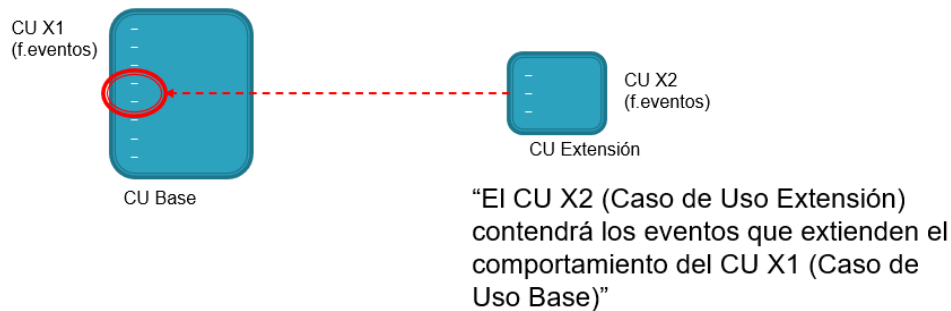
Ejemplo: el caso de uso “*Validar Cliente*” es un caso de uso cuya funcionalidad es usada por otros casos de uso, en este caso por “*realizar pedido*” y “*consultar pedido*”.



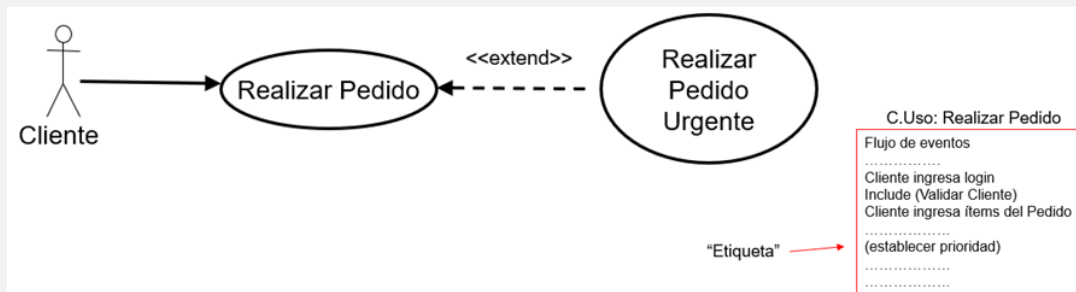
Relaciones de dependencia (extensión)

La **extensión** se utiliza cuando en un caso de uso determinados eventos se dan solamente bajo ciertas condiciones. Para simplificar el mencionado caso de uso los eventos en cuestión se pasan a otro caso de uso aparte.

En este caso, el CU base no se entera si se ejecuta el CU extendido, tal como se muestra en la siguiente imagen:



Ejemplo: “Realizar pedido” es el CU base y se sabe que, bajo determinadas condiciones, habrá que realizar eventos adicionales para que el pedido se efectivice en forma urgente. Por lo tanto, “Realizar pedido urgente” extiende el caso de uso base, con los eventos necesarios para dicha urgencia.



En conclusión

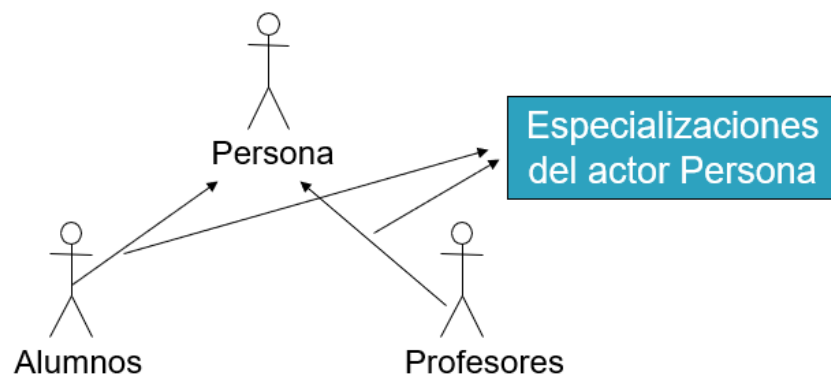
- En una inclusión, el CU incluido siempre se ejecuta luego del CU base.
- En una extensión, el CU extendido no siempre se ejecuta luego del CU base.

Relaciones de generalización/especialización

Es posible definir categorías generales de actores y especializarlos a través de **relaciones de generalización**.

Es decir, se trata de actores que pueden tener una flecha hacia otro actor para indicar qué puede iniciar los casos de uso de ese otro actor.

Esto resulta especialmente útil para representar la **herencia** en el paradigma de programación orientado a objetos.



Diagramas de casos de uso

Finalmente, los diagramas de casos de uso muestran un conjunto de casos de uso, actores y sus relaciones.

Permiten visualizar el comportamiento de un sistema, ya sea en forma **global**, mostrando todos los actores y todos los casos de uso, o bien **parcial** cuando queremos focalizar una parte del sistema con mayor nivel de detalle.

En este diagrama debemos diferenciar lo siguiente

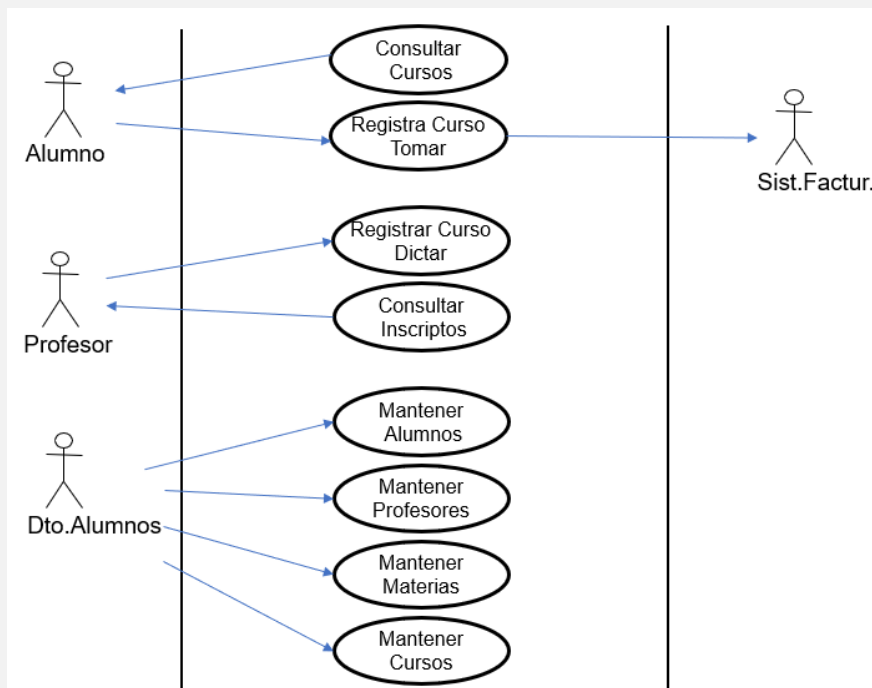
- **Actor primario:** son los actores que *disparan/inician* un caso de uso.
- **Actor secundario:** son los actores que interaccionan con el caso de uso, pero no *inician* el mismo.

Además, la convención para este diagrama indica lo siguiente:

- **Actores primarios:** se ubican a la izquierda del diagrama.
- **Casos de uso:** se ubican en el carril central del diagrama.
- **Actores secundarios:** se ubican a la derecha del diagrama.

Puede darse el caso que un actor sea principal y secundario al mismo tiempo.

Ejemplo de caso de uso para el sistema de inscripción de alumnos y profesores:



Actores temporales

Algo nos quedó en el tintero, pero primero debíamos comprender los casos de uso y las relaciones entre actores y casos de uso para entender mejor a un tipo de actor especial: **el actor temporal**.

Hay funcionalidades que no son disparadas por un actor sino por el **paso del tiempo**. Ese “disparo” se representa con un reloj debajo del cual se coloca la periodicidad (*diariamente, semanalmente, mensualmente, etc.*) y una flecha punteada hacia el caso de uso.

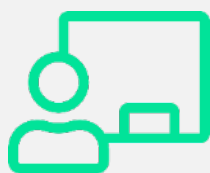
Ejemplo “*diariamente el sistema proporciona al Departamento de Alumnos las vacantes disponibles por curso*”.





Hemos llegado así al final de esta clase en la que vimos:

- Componentes principales de los casos de uso.
- Actores principales, secundarios y temporales.
- Tipos de relaciones.
- Diagrama de casos de uso.



Te esperamos en la **clase en vivo** de esta semana.
No olvides realizar el **desafío semanal**.

¡Hasta la próxima clase!

Bibliografía

Del Águila Cano, I. M. (2019). Ingeniería de requisitos.
Material didáctico. Cuaderno de teoría (Vol. 35). Universidad Almería.

Ramos, D., Noriega, R., Laínez, J. R., & Durango, A. (2017).
Curso de Ingeniería de Software: 2ª Edición. IT Campus Academy.

Vazquez, C. E., & Simoes, G. S. (2016).
Ingeniería de Requisitos: Software orientado al negocio. Brasport.