

Introducción

Los casos de uso son documentos narrativos que describen las secuencias de eventos de distintos actores (agentes externos) que utilizan un sistema para completar un proceso. Son casos de utilización de un sistema y describen posibles escenarios de interacción de los agente externos con el sistema. Cada escenario es una secuencia posible. Un caso de uso es un conjunto de escenarios.

Se desarrollan durante la etapa de análisis y sirven para mejorar la comprensión de los requerimientos y facilitar la comunicación entre los distintos analistas, así como la transición del análisis al diseño. Son independientes del método de diseño y lenguaje que se utilice para la implementación. También se utilizan como herramienta para el relevamiento de los requisitos.

En este documento se muestra los elementos principales de un caso de uso y las distintas relaciones que pueden existir entre casos de usos. Además se presenta algunos consejos prácticos y consideraciones a tener en cuenta al momento de desarrollar los casos de uso. Adicionalmente se muestra el desarrollo completo de un caso de uso, que sirve para ejemplificar los distintos conceptos que se manejan a través del documento.

Elementos principales

Actores

El actor es un agente externo al sistema que participa en la historia de un caso de uso y está representado por el papel que cumple dentro del caso de uso. Los papeles son generalmente desempeñados por seres humanos, usuarios directos del sistema, pero este no es siempre el caso, un papel puede ser desempeñado por otro sistema.

Objetivo

Un caso de uso describe una forma de uso del sistema de principio a fin y debe especificarse claramente cual es su objetivo, las funcionalidades que el caso de uso quiere describir.

Precondiciones

Describe condiciones que debe cumplir el sistema para que se pueda iniciar el caso de uso. Impone restricciones sobre el estado previo del sistema. Las precondiciones no se verifican durante el caso de uso y se asume que se cumplen durante todo el caso de uso.

Descripción

Es una descripción breve, a alto nivel, del caso de uso y del flujo de acciones del mismo. Su propósito es dar una idea de las acciones que se van a realizar en el caso de uso. No se debe hacer mención a decisiones de diseño ni a casos particulares de la ejecución.

Caso de Uso expandido

El caso de uso expandido detalla la funcionalidad más a fondo, desglosándola en una secuencia determinada de pasos que describen el flujo de acciones entre cada uno de los actores y el sistema.

El flujo principal del caso de uso muestra la secuencia de pasos más común que se lleva a cabo para la ejecución exitosa del caso de uso. Se debe especificar un único flujo principal por caso

de uso. La secuencia de pasos debe estar numerada para que pueda ser hecha referencia a un paso en particular dentro del flujo.

Además del flujo principal, un caso de uso puede tener varios flujos alternativos. Un flujo alternativo o secundario describe un escenario alternativo al principal. Permiten mostrar casos particulares y cómo el sistema maneja los distintos errores.

Para ejemplificar suponemos un sistema que le permite al cliente de un banco realizar tareas bancarias desde la casa. El sistema corre localmente en la máquina del usuario y se comunica con el servidor del banco para realizar consultas y efectuar operaciones.

Consideremos el caso de uso "Realizar Transferencia", que permite al usuario realizar una transferencia entre 2 cuentas. En este caso de uso participan 2 actores, agentes externos al sistema: el usuario de la aplicación y el banco. El flujo principal es el siguiente:

1. El usuario ingresa al sitio.
2. El sistema solicita que se registre.
3. El usuario ingresa sus datos.
4. El sistema envía sus datos al banco.
5. El banco valida los datos.
6. El sistema le permite al usuario ingresar y le ofrece un menú de servicios.
7. Usuario: indica al sistema que desea realizar una transferencia
8. Sistema: lista las cuentas del usuario
9. Sistema: pide al usuario que seleccione la cuenta de origen de la transferencia de la lista o que la ingrese manualmente.
10. Usuario: selecciona cuenta de origen de la transferencia de la lista.
11. Sistema: pide al usuario que seleccione la cuenta de destino de la transferencia de la lista o que la ingrese manualmente.
12. Usuario: selecciona cuenta de destino de la transferencia de la lista.
13. Sistema: pide al usuario que ingrese el monto de la transferencia.
14. Usuario: ingresa el monto de la transferencia.
15. Sistema: se comunica con el servidor del banco y le indica que desea realizar una transferencia entre las cuentas especificadas y con el monto indicado.
16. Banco: realiza la transferencia especificada e indica que la transferencia se realizó con éxito.
17. Sistema: actualiza el monto de las cuentas involucradas.
18. Sistema: notifica al usuario que la transferencia se efectuó correctamente.

Aquí se muestra el escenario típico de éxito del caso de uso y por eso se identificó a este flujo como el principal. Adicionalmente, hay otros escenarios posibles y errores que pueden surgir.

Un escenario alternativo podría ser que el usuario no desee seleccionar la cuenta de origen de las cuentas que tiene guardadas en el sistema, sino que desea ingresar manualmente un número de cuenta. Para eso se genera el siguiente flujo alternativo en el punto 4 del flujo principal:

10.A. El usuario desea ingresar manualmente el número de cuenta origen.

- 10.A.1. Usuario: selecciona ingresar manualmente el número de cuenta.
- 10.A.2. Sistema: pide al usuario que ingrese número de cuenta y código del banco.
- 10.A.3. Usuario: ingresa número de cuenta y código del banco.
- 10.A.4. Vuelve al punto 5.

El formato de un flujo alternativo es el siguiente:

- En la primera línea se indica el identificador del flujo, que surge de la referencia al paso del flujo principal del cual se deriva este flujo alternativo y una letra que permite identificar a este flujo dentro de los distintos flujos alternativos que pueden surgir de un mismo punto. En este caso como es el primer flujo alternativo que deriva del paso 4, se lo identifica como 4.A.
- Adicionalmente, en la primera línea se indica la condición que hace que se tome el flujo alternativo.
- Se numera los pasos del flujo alternativo de la misma manera que en el flujo principal (1, 2, 3, ...), o bien antecediendo los números por el identificador del flujo (4.A.1, 4.A.2,...).
- Se indica al final, en caso de que corresponda, el punto de retorno del flujo, es decir, qué pasos siguen a la ejecución del flujo alternativo. En caso que no se desee continuar con ningún otro paso, se debe indicar que se da como terminado el caso de uso (fin del caso de uso).

También se utilizan flujos alternativos para especificar cómo se manejan los posibles errores que pueden surgir. En este caso consideraremos el error que puede surgir en la comunicación con el banco.

16.A. Existe un error en la comunicación con el banco.

- 1. Sistema: no se puede comunicar con el banco
- 2. Sistema: informa al usuario que no se puede comunicar con el banco en este momento.
- 3. Fin CU

Vemos que, al especificarse en el último paso Fin CU, el caso de uso se termina al terminar el flujo alternativo.

Sobre el final de la guía se encuentra el desarrollo completo del caso de uso anterior para que se utilice como ejemplo.

Poscondiciones

Las poscondiciones describen el estado del sistema luego de la ejecución exitosa del caso de uso. Indican lo que se tiene que cumplir luego de que el caso de uso termina exitosamente.

Relaciones entre casos de uso

Se pueden identificar distintas relaciones entre los casos de uso.

Inclusión

La relación de inclusión es la más común entre casos de uso. Permite definir casos de uso que «ejecuten» otros casos de uso. Al caso de uso que incluye a otro se le llama caso base y al otro se le denomina caso incluido. Se puede incluir otros casos de uso tanto en el flujo principal como en algún flujo alternativo. Cuando se llega al punto dentro del caso de uso base en el cual se incluye a otro, se ejecuta en su totalidad el caso de uso incluido y luego su finalizada la ejecución, se continúa en el punto siguiente del caso base. Sólo se continúa la ejecución del caso base si el caso incluido se ejecutó con éxito, en caso de que falle, falla el caso base también. Cabe notar que el caso de uso incluido no tiene por qué tener conocimiento de los casos de uso que lo incluyen.

En el caso del sistema mencionado en la sección anterior, se requiere que el usuario indique su nombre de usuario y contraseña para autenticarse frente al banco. Entonces se podría crear un caso de uso “Autenticar” y modificar el caso de uso anterior para incluir al caso de uso de autenticarse:

1. El usuario ingresa al sitio.
2. <<incluye>> “Autenticar”
3. El sistema le ofrece un menú de servicios.
4. Usuario: indica al sistema que desea realizar una transferencia
5. Sistema: lista las cuentas del usuario
6. ...

Una vez que se llega al paso 2 se ejecuta el caso de uso autenticar y sólo si este se ejecuta con éxito, o sea que el usuario logra efectivamente autenticarse, se continúa con el caso de uso de realizar transferencias.

Extensión

La relación de extensión entre casos de uso se refiere a un fragmento de un caso de uso que extiende, es decir, agrega comportamiento, a otro caso de uso. Se utilizan para describir escenarios alternativos complejos, que sería demasiado complicado de explicar mediante flujos alternativos o para destacar flujos alternativos. Solo se ejecuta en caso de que se cumpla una condición particular en algún punto específico del caso de uso a extender. A ese punto se le llama punto de extensión. En esta relación, el caso de uso extendido conoce al caso de uso que lo extiende y viceversa.

Cuando se llega al punto de extensión se ejecuta el caso de uso que extiende y, en caso de una ejecución exitosa, se continúa la ejecución del caso base en el punto siguiente al punto de extensión.

Generalización

La relación de generalización nos permite definir casos en los cuales tenemos más de un escenario principal para un caso de uso, que comparten cosas en común. Lo que se hace en este caso es crear un caso de uso abstracto y crear distintos casos de uso (uno por cada uno de

esos escenarios que mencionamos anteriormente) que hereden del caso de uso abstracto. Los casos de usos hijos heredan del padre los escenarios, puntos de extensión y relaciones a la vez que pueden definir o enriquecer con nuevos flujos y acciones al caso de uso padre.

Otras consideraciones y casos particulares

Se tiene que tener en cuenta las siguientes consideraciones a la hora de desarrollar los casos de uso en el marco de un proceso de desarrollo iterativo e incremental:

- Los casos de uso no tienen que desarrollarse por completo todos juntos. Primero se identifican los casos de uso y se generan las descripciones (Caso de Uso en Alto Nivel). Luego se expanden aquellos que se consideren más relevantes o se desee implementar primero (en un proceso guiado por casos de uso, serán los casos de uso relevantes para la arquitectura) Expandir un caso de uso consiste en agregar el flujo principal y los alternativos, precondiciones y poscondiciones (Caso de Uso Expandido). A medida que sea necesario, se va desarrollando el resto, cuando se está más cerca de diseñar la funcionalidad.
- No se debería incluir en el flujo del caso de uso detalles de la interfaz gráfica. Aunque sí pueden incluirse borradores de las pantallas.
- Hay errores que no tienen por qué ser considerados a nivel de casos de uso, como por ejemplo lo que tiene que ver con errores en el formato de la entrada (fechas inválidas, pasar strings en lugar de números,...)
- Se puede tener un flujo alternativo dentro de un flujo alternativo. Se nota de manera similar a los flujos alternativos comunes. Por ejemplo, si hay un flujo alternativo que surge a partir del punto 4 del flujo alternativo 12.B, se anota ese flujo de la siguiente manera:

12.B.4.A. Condición que lleva a este flujo alternativo

12.B.4.A.1. Paso 1

12.B.4.A.2. Paso 2

...

o bien

12.B.4.A. Condición que lleva a este flujo alternativo

1. Paso 1

2. Paso 2

...

- Dentro de un flujo alternativo se puede hacer referencia a un paso del flujo base al momento de volver a él. Por ejemplo, es válido terminar un flujo alternativo indicando "Volver al punto 6.A.2" (opcionalmente se puede indicar a qué flujo corresponde el paso, por ejemplo "Volver al punto 6 del flujo principal / del flujo alternativo 6.A). Esto puede resultar útil al momento de manejar errores, debido a que distintos errores pueden requerir manejarse de la misma manera y el manejo de errores se realiza en flujos alternativos.
- No se debe hacer referencia al lenguaje de implementación en los casos de uso ya que estos son independientes del lenguaje y entorno de desarrollo utilizados.

Ejemplo de un caso de uso desarrollado

Nombre:	Realizar transferencia
Autor:	Juan Pérez
Objetivo:	Realizar una transferencia entre 2 cuentas.
Actores:	Usuario del Sistema Banco
Precondiciones:	Ninguna
Descripción:	Este caso de uso comienza cuando un usuario se loguea al sistema y le indica que desea realizar una transferencia. El usuario indica las cuentas de origen y destino de la transferencia y el monto de la misma. El sistema se comunica con el banco, el cual realiza la transferencia especificada.
Flujo principal:	<ol style="list-style-type: none">1. El usuario ingresa al sitio.2. <<incluye>> “Autenticar”3. El sistema le ofrece un menú de servicios.4. Usuario: indica al sistema que desea realizar una transferencia5. Sistema: lista las cuentas del usuario6. Sistema: pide al usuario que seleccione la cuenta de origen de la transferencia de la lista o que la ingrese manualmente.7. Usuario: selecciona cuenta de origen de la transferencia de la lista.8. Sistema: pide al usuario que seleccione la cuenta de destino de la transferencia de la lista o que la ingrese manualmente.9. Usuario: selecciona cuenta de destino de la transferencia de la lista.10. Sistema: pide al usuario que ingrese el monto de la transferencia.11. Usuario: ingresa el monto de la transferencia.12. Sistema: se comunica con el servidor del banco y le indica que desea realizar una transferencia entre las cuentas especificadas con el monto indicado.13. Banco: realiza la transferencia especificada e indica que la transferencia se realizó con éxito.14. Sistema: actualiza el monto de las cuentas involucradas.15. Sistema: notifica al usuario que la transferencia se efectuó correctamente.
Flujos alternativos:	7.A. El usuario desea ingresar manualmente el número de cuenta origen. <ol style="list-style-type: none">7.A.1. Usuario: selecciona ingresar manualmente el número de cuenta.7.A.2. Sistema: pide al usuario que ingrese número de cuenta y código del banco.7.A.3. Usuario: ingresa número de cuenta y código del banco.7.A.4. Vuelve al punto 8.

9.A. El usuario desea ingresar manualmente el número de cuenta origen.

- 9.A.1. Usuario: selecciona ingresar manualmente el número de cuenta.
- 9.A.2. Sistema: pide al usuario que ingrese número de cuenta y código del banco.
- 9.A.3. Usuario: ingresa número de cuenta y código del banco.
- 9.A.4. Vuelve al punto 10.

12.A. Existe un error en la comunicación con el banco.

- 12.A.1. Sistema: no se puede comunicar con el banco
- 12.A.2. Sistema: informa al usuario que no se puede comunicar con el banco en este momento.

13.A. El usuario no está autorizado para realizar transferencias desde la cuenta de origen especificada.

- 13.A.1. Banco: indica al sistema que el usuario no tiene permiso para realizar la transferencia
- 13.A.2. Sistema: informa al usuario que no se puede realizar la transferencia.
- 13.A.3. Vuelve al punto 3

13.B. No hay fondos suficientes en la cuenta de origen para realizar la transferencia.

- 13.B.1. Banco: indica al sistema que la cuenta no tiene fondos suficientes para realizar la transferencia.
- 13.B.2. Sistema: informa al usuario que no se puede realizar la transferencia.
- 13.B.3. Vuelve al punto 8

Poscondiciones:

Los saldos de ambas cuentas quedan actualizados.¹

Referencias Bibliográficas

- Ingeniería de Software – Teoría y práctica de Shari Lawrence Pfleeger
- UML y Patrones de Craig Larman
- Material sobre requerimientos del curso de Introducción a la Ingeniería de Software de la Facultad de Ingeniería de la Universidad de la República.

¹ Si se pone como poscondición, no sería obligatorio ponerlo como un paso, ya que no refiere a la interacción, aunque sí es recomendable. Por lo menos debe estar en alguno de los dos lados.