

Entornos de desarrollo

4

Casos de uso

IES Nervión
Miguel A. Casado Alías

Introducción

- Técnica propuesta por Ivar Jacobson
- Los modelos de casos de uso se enmarcan dentro del “Análisis de sistemas”
- Se usa principalmente en proyectos de desarrollo orientados a objetos; sin embargo, es perfectamente posible utilizar esta técnica para identificar requisitos en un proyecto “estructurado”

Definición

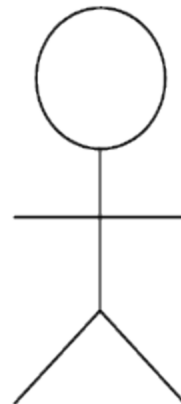
- Un caso de uso es una secuencia de interacciones entre un sistema y alguien o algo que usa alguno de sus servicios.
- Los casos de uso se emplean para capturar el comportamiento deseado del sistema en desarrollo, sin tener que especificar cómo se implementa ese desarrollo.

Objetivos

- Identificar los requisitos funcionales del sistema y expresarlos desde el punto de vista del usuario.
- Guiar todo el proceso de desarrollo del sistema de información, ayudando a validar la arquitectura y a verificar el sistema mientras evoluciona a lo largo del desarrollo.

Actores

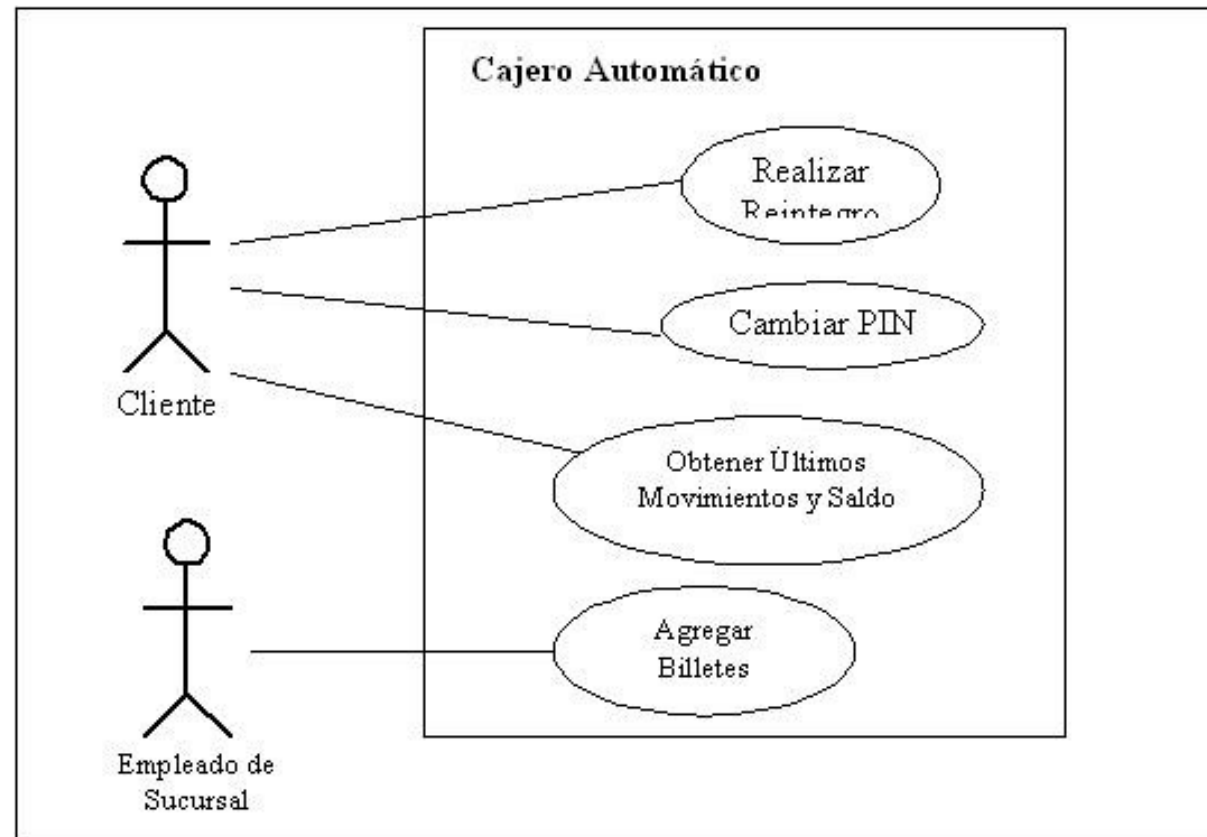
- Representan cualquier cosa que intercambia información con el sistema.
- Pueden ser una persona o no.
- Son externos al Sistema.
- Los actores representan un rol. Un mismo usuario puede interactuar con el sistema asumiendo distintos roles, por lo tanto puede estar identificado por varios actores.



Casos de uso

- Como dijimos antes, se trata de una secuencia de interacciones entre el sistema y algo o alguien que usa uno de sus servicios.
- Describe QUÉ hace un sistema.
- Un caso de uso es iniciado por un actor
- Se representan con una elipse
- El nombre del caso de uso se expresa desde el punto de vista del actor, no del sistema.

Casos de uso (II)



Descripción de los casos de uso

- Lo más importante del modelo es la descripción de los casos de uso, no el diagrama.
- Los casos de uso se pueden documentar con texto, lenguaje natural.
- Hay diversos modelos de plantillas que podemos usar para describir casos de uso, pero en general siempre constarán de una lista de pasos que describan la interacción del actor con el sistema.

Ejemplo de descripción

Book Flight

1. The use case begins when the agent specifies a travel itinerary for a client.
2. The system searches a set of appropriate flights and presents them to the agent.
(frequent flier a)
3. The agent chooses Select Flight.
4. The system verifies that space is available on the flight and reserves a seat on the flight.
(frequent flier b)
5. The agent finalizes the booking by supplying payment information.
(frequent flier c)
6. The system books the seats and issues the ticket.

Alternativas (descripciones)

- En la descripción se incluye también un flujo alternativo.
- Representan un error o excepción en el curso normal del caso de uso.
- No tienen sentido por sí mismas, fuera del contexto del caso de uso en el que ocurren.

Alternativas (II)

Book Flight

1. The use case begins when the agent specifies a travel itinerary for a client.
2. The system searches a set of appropriate flights and presents them to the agent.
(frequent flier a)
3. The agent chooses Select Flight.
4. The system verifies that space is available on the flight and reserves a seat on the flight.
(frequent flier b)
5. The agent finalizes the booking by supplying payment information.
(frequent flier c)
6. The system books the seats and issues the ticket.

Alternatives

4a: Seat is not available in ticket category:

- 4a1. The system informs the agent that no seats are available in the client's chosen price category.
- 4a2. The agent specifies another price preference.

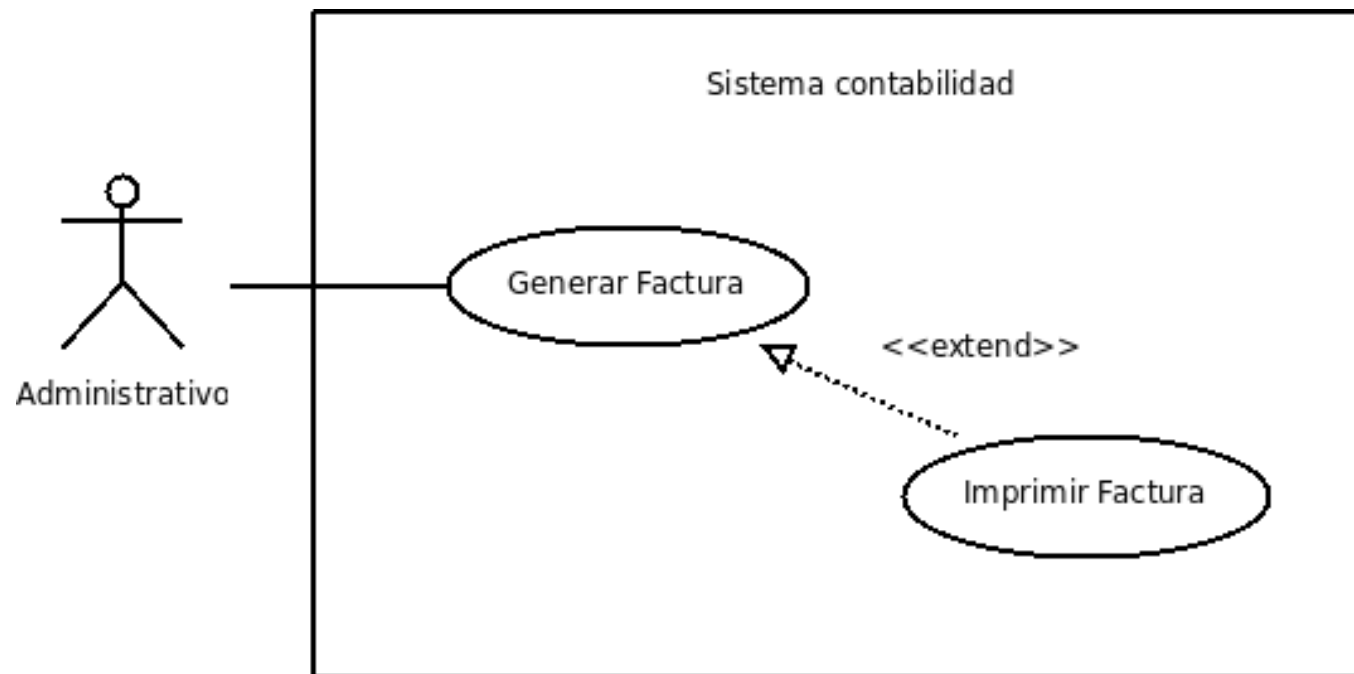
4b: Seat is not available (flight fully booked):

- 4b1. The system informs the agent that no seats are available at all.
- 4b2. The agent specifies another set of departure time preferences.
(frequent flier d)

Extensiones

- Representan una parte de la funcionalidad del caso que no siempre ocurre
- Son un caso de uso en sí mismas
- No necesariamente provienen de un error o excepción.
 - P. Ej: Si “cenar en un restaurante” fuera un caso de uso, “tomar café después de cenar” podría ser una extensión de aquel.

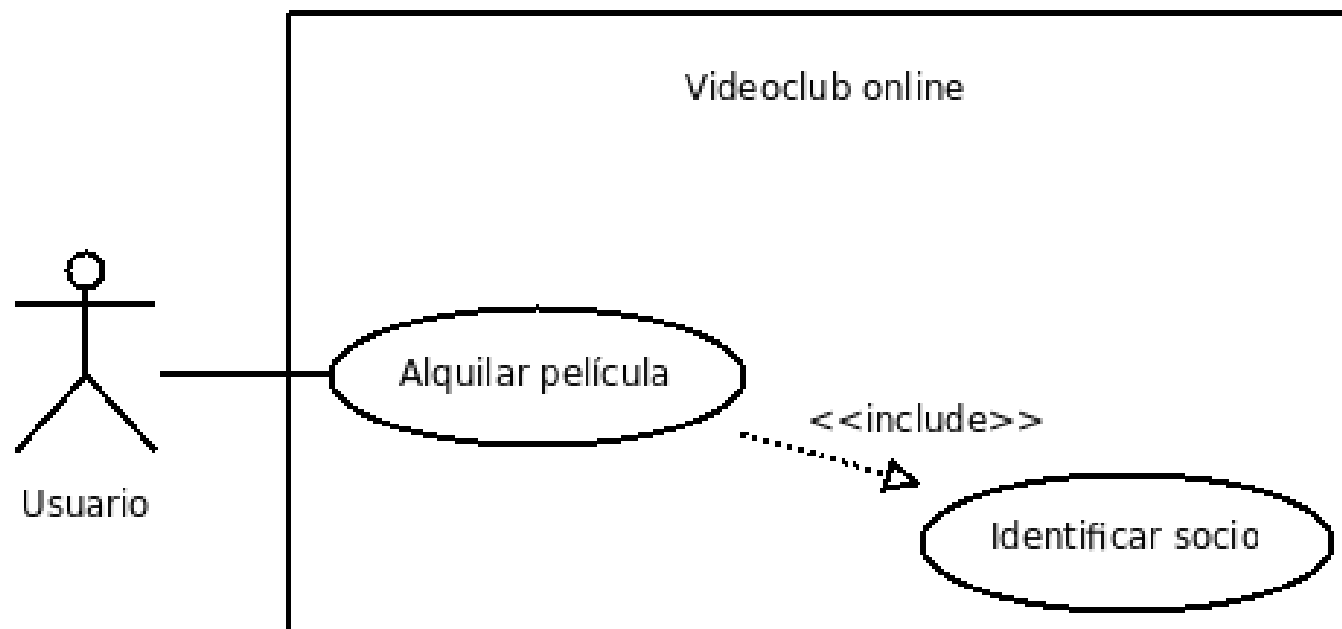
Extensiones (II)



Inclusiones

- Aparecen como funcionalidad común a varios casos de uso (se asemejan al concepto de subprograma)
- Los casos usados son casos de uso en sí mismos.
- El caso es incluido SIEMPRE que el caso que lo incluye es ejecutado, a diferencia de las extensiones, las cuales son opcionales.

Inclusiones (II)



Generalización (herencia)

- Es más común que se use entre actores que entre casos de uso.
- En el ejemplo, toda la funcionalidad habilitada para “Empleado de Ventas” también lo estará para el supervisor.

