

UT6

Elaboración de
diagramas de
comportamiento

1. INTRODUCCIÓN

- ◉ Los diagramas de clases modelan la estructura estática del sistema, pero no dan información sobre el comportamiento dinámico del mismo.
- ◉ Para poder modelar esta información utilizaremos otro tipo de diagramas UML: **diagramas de comportamiento.**
- ◉ Dentro de los diagramas de comportamiento se incluyen:
 - Diagramas de casos de uso
 - Diagramas de actividad
 - Diagramas de estado
 - Diagramas de interacción
 - Diagramas de secuencia
 - Diagrama de comunicación
 -

1. INTRODUCCIÓN

DIAGRAMA	RESUMEN
D. de casos de uso	Describe el comportamiento del sistema desde el punto de vista del usuario.
D. de actividad	Muestra los pasos, puntos de decisión y bifurcaciones (similar a diagrama de flujo)
D. de estado	Muestra el conjunto de estados por los cuales pasa un objeto y cómo se producen las transiciones de un estado a otro.
D. de secuencia	Muestra cómo interactúan unos objetos con otros.
D. de comunicación	Muestra las interacciones entre los elementos en tiempo de ejecución
D. de tiempo	Definen el comportamiento de diferentes objetos dentro de una escala de tiempo

2. DIAGRAMAS DE CASOS DE USO

- ◉ Los casos de uso modelan el sistema desde el punto de vista del usuario.
- ◉ Facilita la obtención de los requisitos software durante la fase de análisis de un proyecto.
- ◉ Objetivos:
 - Definir los requisitos funcionales y operativos del sistema
 - Diseñar escenarios que describen cómo se usará el sistema.
 - Describir claramente cómo interactúa el usuario con el sistema y viceversa.
 - Proporcionar una base para la validación de las pruebas.
- ◉ Se utiliza un lenguaje sencillo para que pueda ser entendido fácilmente (similar a diagrama ER de una BD)

2.1 ELEMENTOS DEL DIAGRAMA DE CASOS DE USO

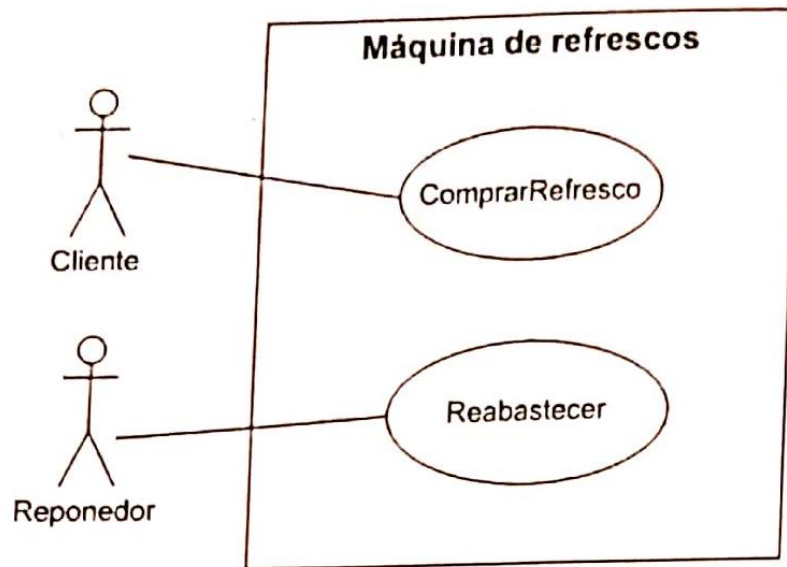
◉ Elementos de un diagrama de casos de uso:

- **Actores:** cualquier cosa que interactúa con el sistema y es externo a él. No es necesariamente una persona (dispositivo, otro sistema, etc.) Se representa mediante un monigote con un nombre debajo.
- **Casos de uso:** acción que se realiza tras la orden de un actor u otro caso de uso. Se representa mediante un óvalo o **elipse** y dentro se escribe una descripción textual.
- **Relaciones:** existen varios tipos de relaciones. La más típica es la **asociación** entre actores y casos de uso que se representan mediante una **línea continua**, representa la participación del actor en el caso de uso.

2.1 ELEMENTOS DEL DIAGRAMA DE CASOS DE USO

◉ Ejemplo:

- Actores: Cliente y reponedor
- Casos de uso: *ComprarRefresco* y *Reabastecer*
- Relaciones: asociaciones entre actores y casos de uso

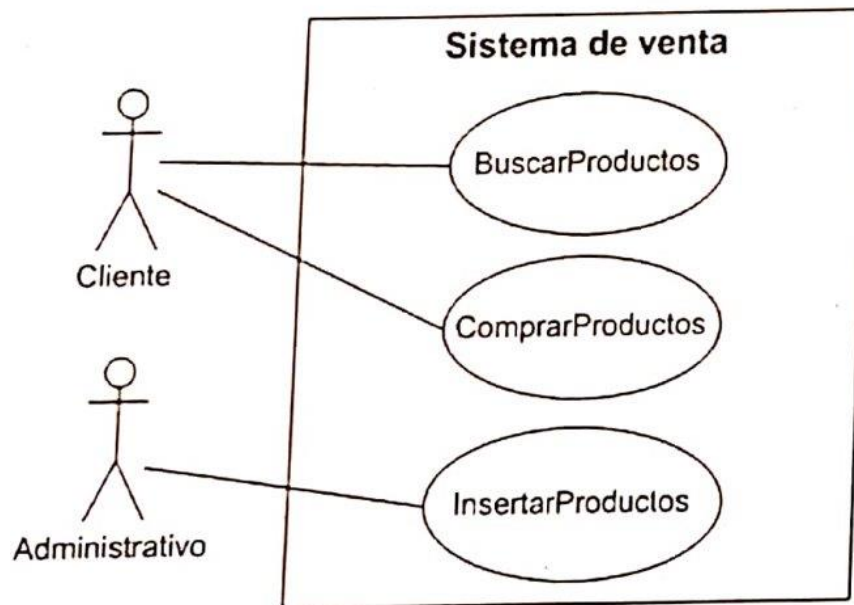


- ◉ NOTA: los actores son externos al sistema

2.1 ELEMENTOS DEL DIAGRAMA DE CASOS DE USO

○ Otro ejemplo:

- Actores: Cliente y Administrativo
- Casos de uso: *BuscarProductos*, *ComprarProductos* e *InsertarProductos*



2.1 ELEMENTOS DEL DIAGRAMA DE CASOS DE USO

- ◉ Los casos de uso lo inician los actores para solicitar o modificar información del sistema o para informar al sistema de un cambio en su entorno que le incumbe.
- ◉ El nombre del caso de uso debe coincidir con el objetivo del actor principal, que es normalmente el que comienza el caso de uso.

2.2 IDENTIFICAR ACTORES

- ⦿ Para identificar los actores se necesita saber **qué y quién interactúa con el sistema** y qué roles desempeñan cuando interactúan con él.
- ⦿ A la hora de definir actores hay que tener en cuenta:
 - Los actores son **siempre** externos al sistema
 - **Interactúan directamente** con el sistema.
 - Necesitan un nombre que describa el papel que desempeñan.
 - La misma persona o elemento puede interpretar varios roles como actores distintos.

2.2 IDENTIFICAR ACTORES

◉ Ejemplo: actores de una tienda online

- **Usuario normal:** visita la tienda sin identificarse.
- **Usuario registrado:** compra productos
- **Administrativo:** comprueba el estado del stock de los productos, realiza pedidos a proveedores y da de alta nuevos productos
- **Empleado del almacén:** encargado de tramitar los pedidos de los clientes, etc.

2.3 IDENTIFICAR CASOS DE USO

- ⦿ Para identificar los casos de uso es necesario entender lo que el sistema debe hacer
- ⦿ La mejor forma de empezar es identificar los actores que usarán el sistema y cómo lo usarán.
- ⦿ Nos debemos plantear una serie de preguntas:
 - ¿Qué tareas realizan los actores involucrados en el sistema?
 - ¿Qué información crea, almacena, modifica o lee el actor?
 - ¿Debe el actor notificar al sistema cambios externos?
 - ¿Debe el sistema informar a un actor de cambios internos?
 - ¿Interactúa el sistema con algún sistema externo?

2.3 IDENTIFICAR CASOS DE USO

- ◉ Plantilla para la especificación de casos de uso:
 - Nombre del caso de uso
 - ID del caso de uso
 - Actores implicados en el caso de uso: existen dos tipos de actores, principales (activan el caso de uso) y secundarios (interactúan con el caso de uso después de haberse activado).
 - Precondiciones: condiciones que se deben cumplir antes de que el caso de uso pueda empezar.
 - Curso normal: pasos que llevan a la finalización con éxito.
 - Postcondiciones: condiciones que se deben cumplir al final del caso de uso.
 - Alternativas: durante la ejecución de un caso de uso, suelen aparecer errores o excepciones (ejemplo, un usuario accede a un sistema con datos erróneos). El sistema debe informar de esta situación al usuario. Esas desviaciones del curso normal se llaman alternativas.

EJEMPLO

⦿ Especificación del caso de uso BuscarProductos:

Nombre: BuscarProductos ID: CU-1
Descripción: El cliente solicita consultar productos que cumplan una serie de criterios de búsqueda. El sistema muestra los datos de los productos solicitados.
Actores: Cliente.
Precondiciones: Se requiere que el cliente esté registrado en el sistema.
Curso normal del caso de uso: <ol style="list-style-type: none">1. El cliente selecciona buscar productos.2. El sistema solicita los criterios de búsqueda.3. El cliente introduce los criterios solicitados.4. El sistema busca productos que cumplan los criterios introducidos por el cliente.5. El sistema encuentra productos y se los muestra al cliente.
Postcondiciones: El sistema muestra los productos encontrados.
Alternativa 1: <ol style="list-style-type: none">5. El sistema no encuentra productos.6. El sistema informa al cliente de que no hay productos que cumplan el criterio de búsqueda.7. Fin del caso de uso.

EJEMPLO

- Otra forma de realizar la especificación de este caso de uso se muestra a continuación:

Curso normal del caso de uso:

1. El cliente selecciona buscar productos.
2. El sistema solicita los criterios de búsqueda.
3. El cliente introduce los criterios solicitados.
4. El sistema busca productos que cumplan los criterios introducidos por el cliente.
5. *SI* el sistema encuentra algún producto
 - 5.1 *Para* cada producto encontrado
 - 5.1.1. El sistema muestra sus datos.
6. *SI NO*
 - 6.1 El sistema comunica al cliente que no ha encontrado productos.
7. El cliente repite los pasos 2 a 6 hasta finalizar las búsquedas.

Postcondiciones:

Ninguna.

Alternativas:

Ninguna

EJEMPLO


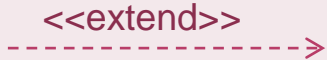

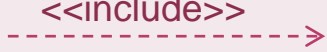
○ Especificación caso de uso InsertarProductos:

Nombre: InsertarProductos
ID: CU-2
Descripción: El administrativo solicita insertar un producto en el sistema. El sistema inserta los datos del producto introducido por el administrativo.
Actores: Administrativo.
Precondiciones: Se requiere que el administrativo esté registrado en el sistema.
Curso normal del caso de uso: <ol style="list-style-type: none">1. El administrativo solicita insertar un producto.2. El sistema solicita los datos del producto.3. El administrativo introduce los datos.4. El sistema valida los datos introducidos por el administrativo.5. SI los datos son correctos<ol style="list-style-type: none">5.1 El sistema inserta el producto.6. SI NO<ol style="list-style-type: none">6.1 El sistema comunica al cliente que ha introducido datos erróneos.7. El administrativo repite los pasos 2 a 6 hasta finalizar la inserción de productos.
Postcondiciones: Ninguna.
Alternativas: Ninguna

ACTIVIDAD 1

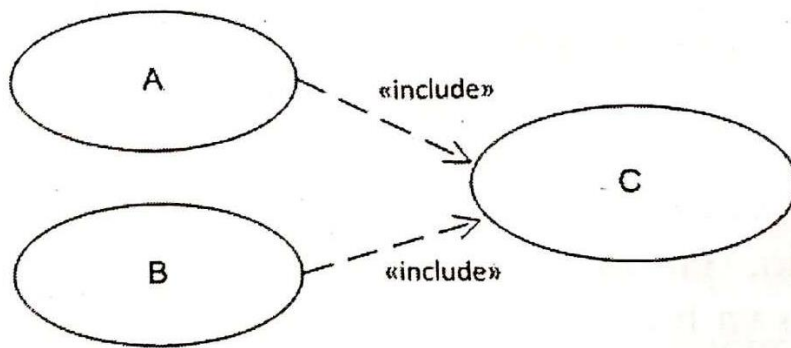
- ⦿ Realiza la especificación del caso de uso **ComprarProductos**.

2.4 RELACIONES EN UN DIAGRAMA DE CASOS DE USO

RELACIÓN	FUNCIÓN	NOTACIÓN
Asociación	Es la línea de comunicación entre un actor y un caso de uso en el que participa	
Extensión <<extend>>	Permite que un caso de uso extienda su comportamiento con uno o más fragmentos de comportamiento de otro (apunta al caso a extender)	
Generalización de casos de uso	Es como la generalización entre clases. El caso de uso hijo hereda el comportamiento y el significado del caso de uso padre.	
Inclusión <<include>> o <<uses>>	Permite que un caso de uso base incluya el comportamiento de otro caso de uso (apunta al caso a incluir)	

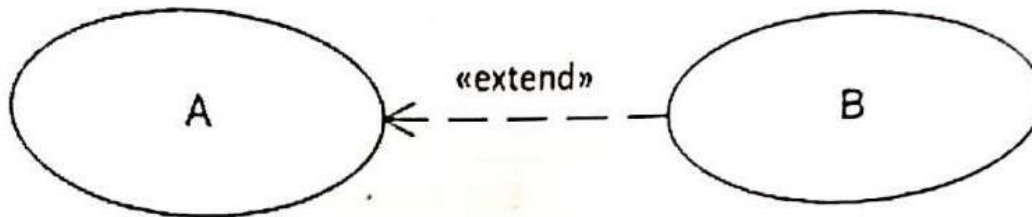
2.4.1 DIFERENCIAS <<EXTEND>> E <<INCLUDE>>

- ◉ La relación <<include>> ocurre cuando se tiene una porción de comportamiento que es similar en más de un caso de uso.
- ◉ Si por ejemplo tenemos dos casos de uso A y B que tienen una serie de pasos en común, se ponen esos pasos en un tercer caso de uso C y A y B lo incluyen para usarlo.
- ◉ Es necesario que ocurra el caso incluido para satisfacer el objetivo del caso de uso base.
- ◉ Ejemplo



2.4.1 DIFERENCIAS <<EXTEND>> E <<INCLUDE>>

- Se usa la relación <<extend>> cuando se tiene un caso de uso que extiende o amplía la funcionalidad de otro caso de uso (caso de uso base).
- Por ejemplo, el caso de uso B extiende la funcionalidad del caso de uso A añadiendo algunos pasos.

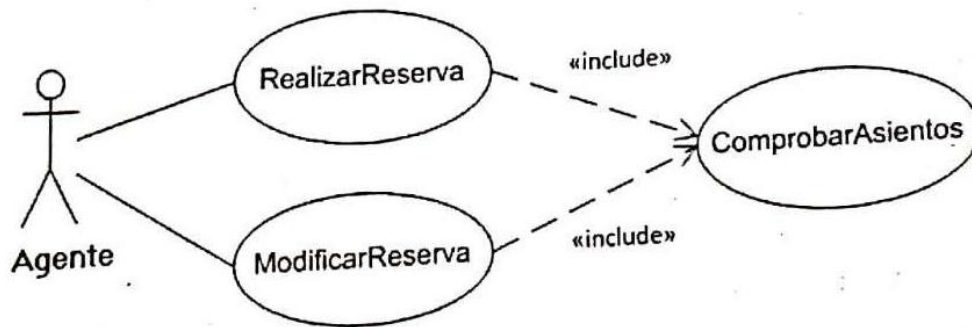


- Una diferencia con respecto al anterior es que el caso de uso que es extendido no sabe nada del caso de uso que lo extiende.

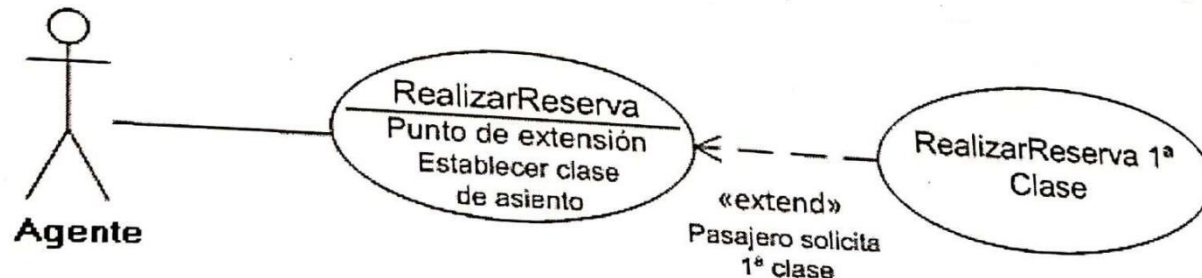
2.4.1 DIFERENCIAS <<EXTEND>> E <<INCLUDE>>

- ◉ En la relación <<extend>>, el caso de uso de extensión (el B) no es indispensable que ocurra y cuando lo hace ofrece un valor extra al objetivo original del caso de uso base (el A).
- ◉ El caso de uso base se describe como un caso de uso normal, no contiene ninguna referencia al caso de uso extendido.
- ◉ Ejemplo: en el modelo de un sistema de viajes aéreos se tienen dos casos de uso que tiene unos pasos comunes que se incluyen en otro caso de uso. Los casos de uso **RealizarReserva** y **ModificarReserva** incluyen el caso de uso **ComprobarAsientosDisponibles**.

2.4.1 DIFERENCIAS <<EXTEND>> E <<INCLUDE>>



- Ejemplo: se amplía el caso de uso **RealizarReserva** para incluir la situación de que se puedan reservar asientos en primera clase. La circunstancia que causa que se active la extensión es la condición de que el pasajero solicite 1ª clase.

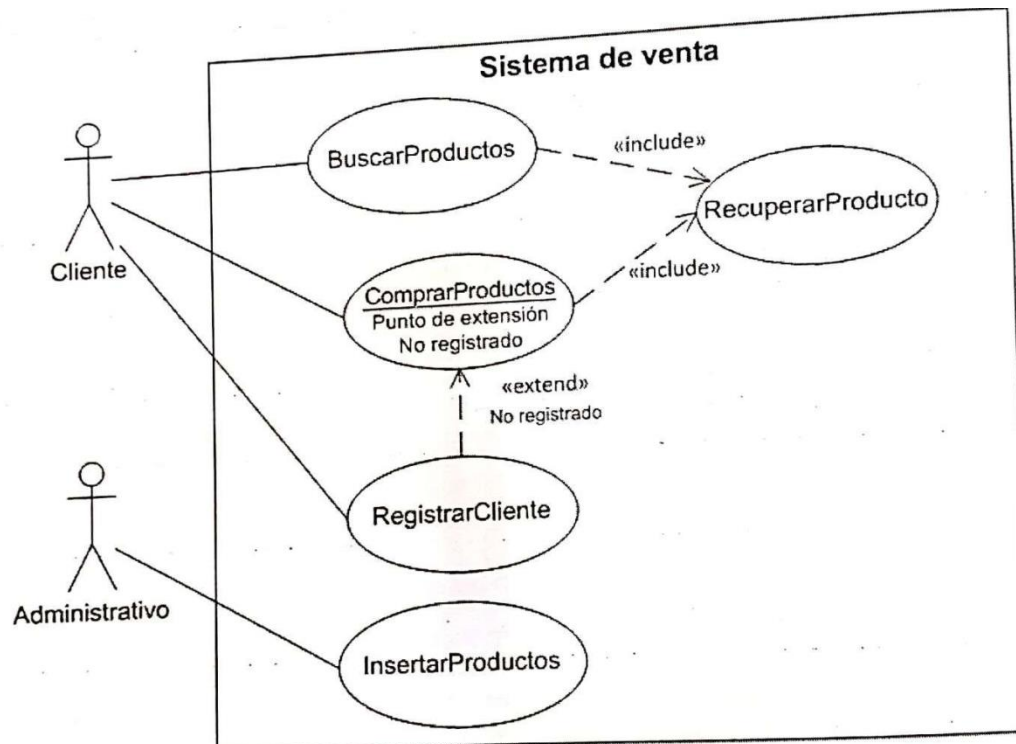


EJEMPLO

- ◉ Siguiendo con el ejemplo anterior de un sistema de ventas donde se requería que el cliente estuviese registrado en el sistema, supongamos que cuando un cliente quiere realizar una compra, si no está registrado se le ofrece la posibilidad de registrarse en ese momento.
- ◉ Para ello, se añade un caso de uso para que el cliente se registre, que extienda el caso de uso de ComprarProductos.
- ◉ Además, cuando se van comprando productos, se necesita recuperar su información para obtener el precio. También cuando se realiza la búsqueda de productos se necesita recuperar dicha información para mostrarla al cliente.

EJEMPLO

- Por tanto, podemos añadir un nuevo caso de uso en el que se recupere la información de un producto.
- Este caso de uso se incluirá mediante una relación <<include>> en los otros dos casos de uso.



EJEMPLO

- La nueva descripción de los casos de uso sería:

Nombre: RecuperarProducto
ID: CU-4
Descripción: El cliente recupera los datos de los productos solicitados.

Actores: Cliente.
Precondiciones: Se requiere que el cliente esté registrado en el sistema.
Curso normal del caso de uso: <ol style="list-style-type: none">1. El sistema solicita los criterios de búsqueda.2. El cliente introduce los criterios solicitados.3. El sistema encuentra el producto o productos solicitados según los criterios especificados.
Postcondiciones: El sistema encuentra los datos del producto o productos solicitados
Alternativas: <ol style="list-style-type: none">3. El sistema no encuentra los productos solicitados.4. Fin del caso de uso.

EJEMPLO

- La nueva descripción de los casos de uso sería:

Nombre: BuscarProductos ID: CU-1
Descripción: El cliente solicita consultar productos que cumplan una serie de criterios de búsqueda. El sistema muestra los datos de los productos solicitados.
Actores: Cliente.
Precondiciones: Se requiere que el cliente esté registrado en el sistema.
Curso normal del caso de uso: <ol style="list-style-type: none">1. El cliente selecciona buscar productos.2. Include (RecuperarProducto)3. Para cada producto encontrado.<ol style="list-style-type: none">3.1 El sistema muestra sus datos.4. El cliente repite los Pasos 2 a 3 hasta finalizar las búsquedas.
Postcondiciones: El sistema muestra al cliente los productos solicitados
Alternativas: <ol style="list-style-type: none">3. El sistema no encuentra los productos solicitados<ol style="list-style-type: none">3.1 El sistema informa al cliente.4. Continuar en el Paso 4.

ACTIVIDAD

- ◉ A partir de la siguiente descripción vamos a desarrollar un sistema de gestión de notas donde los profesores puedan registrar y actualizar las notas de los alumnos y los alumnos puedan consultarlas.
- ◉ La lista de requerimientos para este sistema es:
 - El profesor puede introducir y modificar las notas de los alumnos. Para ello, seleccionar el curso, la asignatura y el alumno y dará el valor a la nota. Para que un profesor pueda introducir o modificar notas debe estar autenticado en el sistema.
 - Los alumnos pueden consultar la nota. Las notas se pueden consultar por asignatura o se pueden consultar todas las asignaturas del alumno con su nota. Para que un alumno pueda consultar las notas debe estar autenticado en el sistema.

ACTIVIDAD

- El sistema debe generar un listado de notas por asignatura. Para ello, el profesor seleccionará el curso y la asignatura y el sistema generará un PDF con dicha información.
 - Cuando los usuarios son autenticados, el sistema muestra la funcionalidad disponible dependiendo del tipo de perfil del usuario.
-
- Realiza el diagrama de casos de uso
 - Realiza la especificación del caso de uso **Autenticar y ConsultarNotaAsig.**

2.5. CASOS DE USO EN ECLIPSE

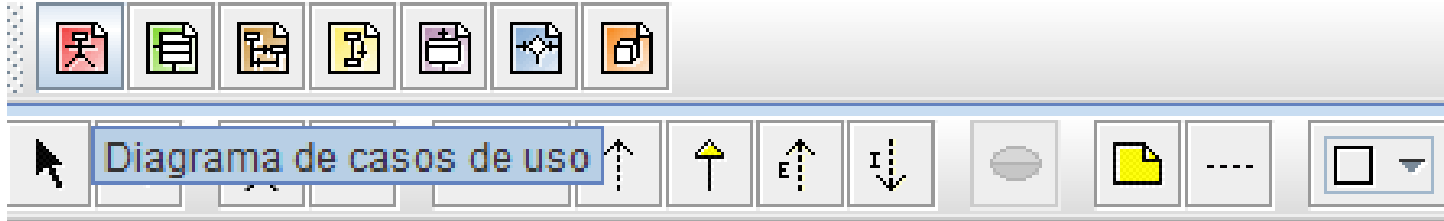
- ◉ Para crear diagramas de casos de uso sencillos en Eclipse podemos instalar el plugin **UMLet**
- ◉ Creamos un nuevo proyecto **Umlet diagram**
- ◉ Elegimos la ubicación (container) del diagrama y su nombre
 - Ejemplo: Podemos guardar los diagramas en una carpeta *model*
- ◉ Uso del plugin:
 - UMLet es muy sencillo y se utiliza arrastrando los elementos del panel derecho al diagrama
 - En nuestro caso, elegimos la opción **UML Use Case** del panel
 - Al seleccionar un elemento, aparecen debajo sus propiedades
 - Los diagramas se guardan con extensión **.uxf** dentro de un proyecto (por ejemplo, en una carpeta *model*)

ACTIVIDAD

- ◉ Realiza el diagrama de casos de uso del sistema de gestión de notas utilizando el plugin de Eclipse.

2.6 CASOS DE USO CON ARGOUML

- Con ArgoUML también podemos crear diagramas de casos de uso:



- Para crear el límite del sistema y escribir el nombre usamos la herramienta .

ACTIVIDAD

- ◉ Utilizando ArgoUML, realiza el diagrama de casos de uso del sistema de gestión de venta.

ACTIVIDAD

- Realiza los ejercicios del 1 al 4 de la Hoja de Ejercicios.

3. DIAGRAMAS DE INTERACCIÓN

- ◉ El diagrama de interacción describe el intercambio de secuencias de mensajes entre las partes de un sistema.
- ◉ Proporciona una visión integral del comportamiento de un sistema, es decir, muestra el flujo de control a través de varios objetos.
- ◉ Dentro de la categoría de diagramas de interacción tenemos dos diagramas que se centran en aspectos distintos:
 - **Diagrama de secuencia:** centrado en la ordenación temporal de los mensajes.
 - **Diagrama de comunicación o colaboración:** centrado en la organización estructural de los objetos que envían y reciben mensajes.

3.1 DIAGRAMA DE SECUENCIA

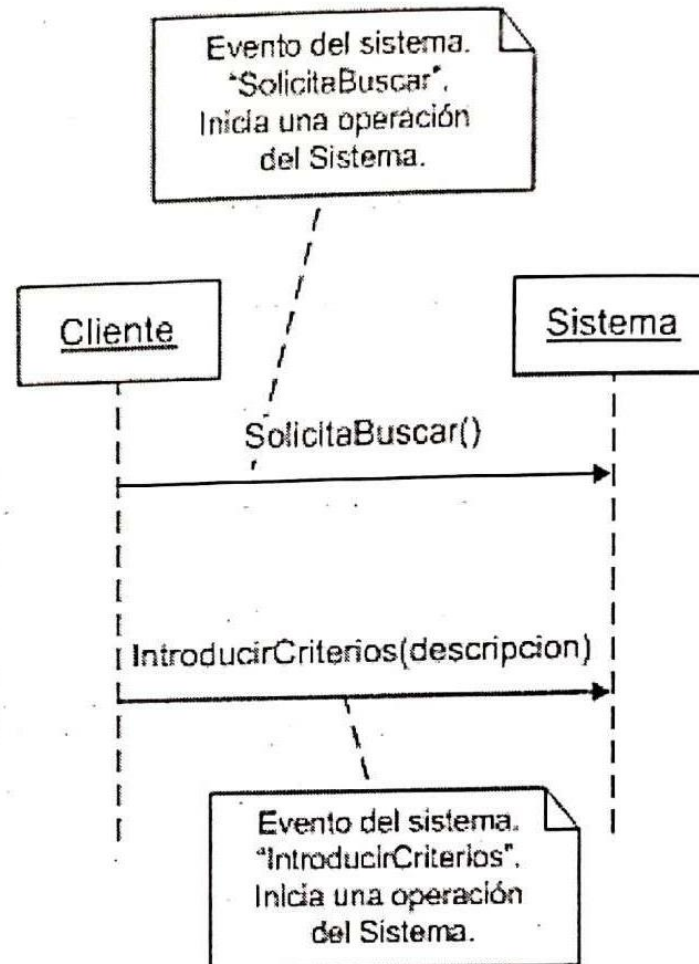
- ◉ El **diagrama de secuencia de sistema** muestra gráficamente los eventos que fluyen de los actores al sistema.
- ◉ Para su elaboración se parte de los casos de uso elaborados durante la etapa de análisis.
- ◉ Un diagrama de secuencia tiene dos dimensiones:
 - **Dimensión vertical:** representa el tiempo.
 - **Dimensión horizontal:** representa los roles que participan en la interacción.
- ◉ A continuación, se muestra el diagrama de secuencia de sistema para el caso de uso **BuscarProductos**.

3.1 DIAGRAMA DE SECUENCIA

CASO DE USO: BuscarProductos.

Curso normal del caso de uso:

1. El cliente selecciona buscar productos.
2. El sistema solicita los criterios de búsqueda.
3. El cliente introduce los criterios solicitados.
4. El sistema busca productos que cumplan los criterios introducidos por el cliente.
5. El sistema encuentra productos y se los muestra al cliente.



3.1 DIAGRAMA DE SECUENCIA

- ⦿ Cada rol, ya sea actor u objeto, se representa mediante un rectángulo distribuido horizontalmente en la zona superior del diagrama.
- ⦿ A cada uno se le asocia una línea vertical llamada **línea de vida**, donde se describe la interacción a lo largo del tiempo.
- ⦿ De cada línea vertical salen diferentes mensajes representados mediante flechas que muestran la interacción.
- ⦿ Encima de la flecha se muestra el mensaje.
- ⦿ Es importante la secuencia de mensajes en esta línea ya que indican el orden en que ocurren los eventos.

3.1 DIAGRAMA DE SECUENCIA

- ◉ En el ejemplo anterior, se muestran horizontalmente los roles que participan en la interacción: el cliente y el sistema con sus líneas de vida.
- ◉ Se muestra dos mensajes, que parten del cliente.
- ◉ El primer mensaje o evento **SolicitarBuscar()** inicia la operación de búsqueda de productos en el sistema.
- ◉ El segundo, **IntroducirCriterios(descripcion)** inicia la operación de búsqueda de productos según los criterios especificados.

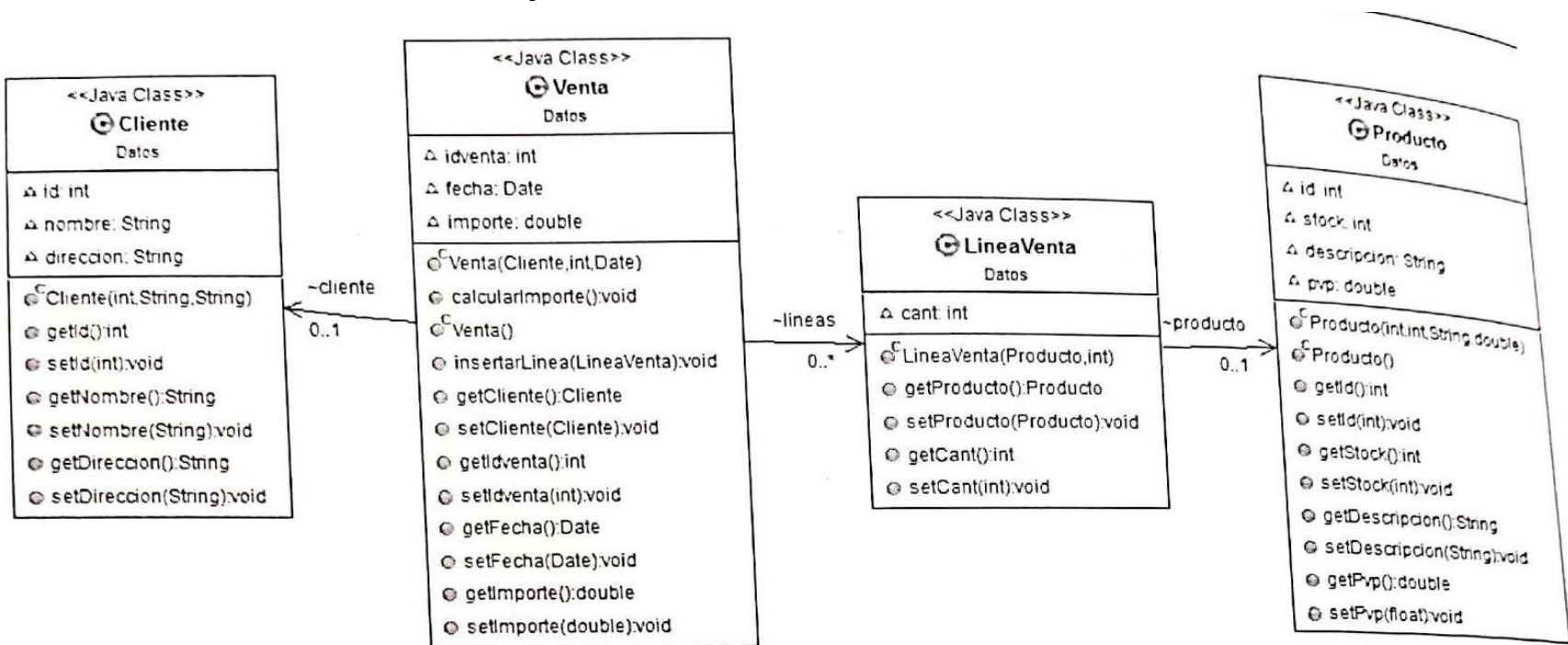
3.1 DIAGRAMA DE SECUENCIA

- En un sistema orientado a objetos, donde tenemos varias clases y varios objetos que se comunican unos con otros, las clases y los objetos se representan en la dimensión horizontal.
- Existe una nomenclatura a la hora de nombrar los rectángulos que representan los objetos: **objeto:clase**.
- El nombre del objeto es opcional**

Una clase <i>Empleado</i>	<u>Empleado</u>
<i>emp</i> es un objeto de la clase <i>Empleado</i>	<u>emp:Empleado</u>
Un objeto cualquiera de la clase <i>Empleado</i>	<u>:Empleado</u>

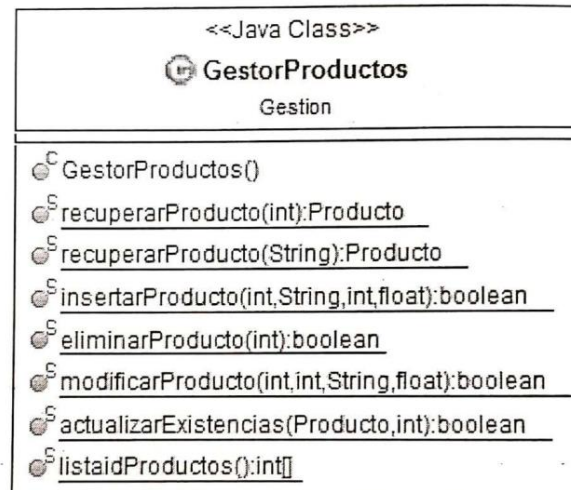
3.1 DIAGRAMA DE SECUENCIA

- **Ejemplo:** diagrama de clases de un sistema de ventas donde un cliente puede tener muchas ventas, una venta puede tener muchas líneas de venta y cada línea de venta tiene un producto.



3.1 DIAGRAMA DE SECUENCIA

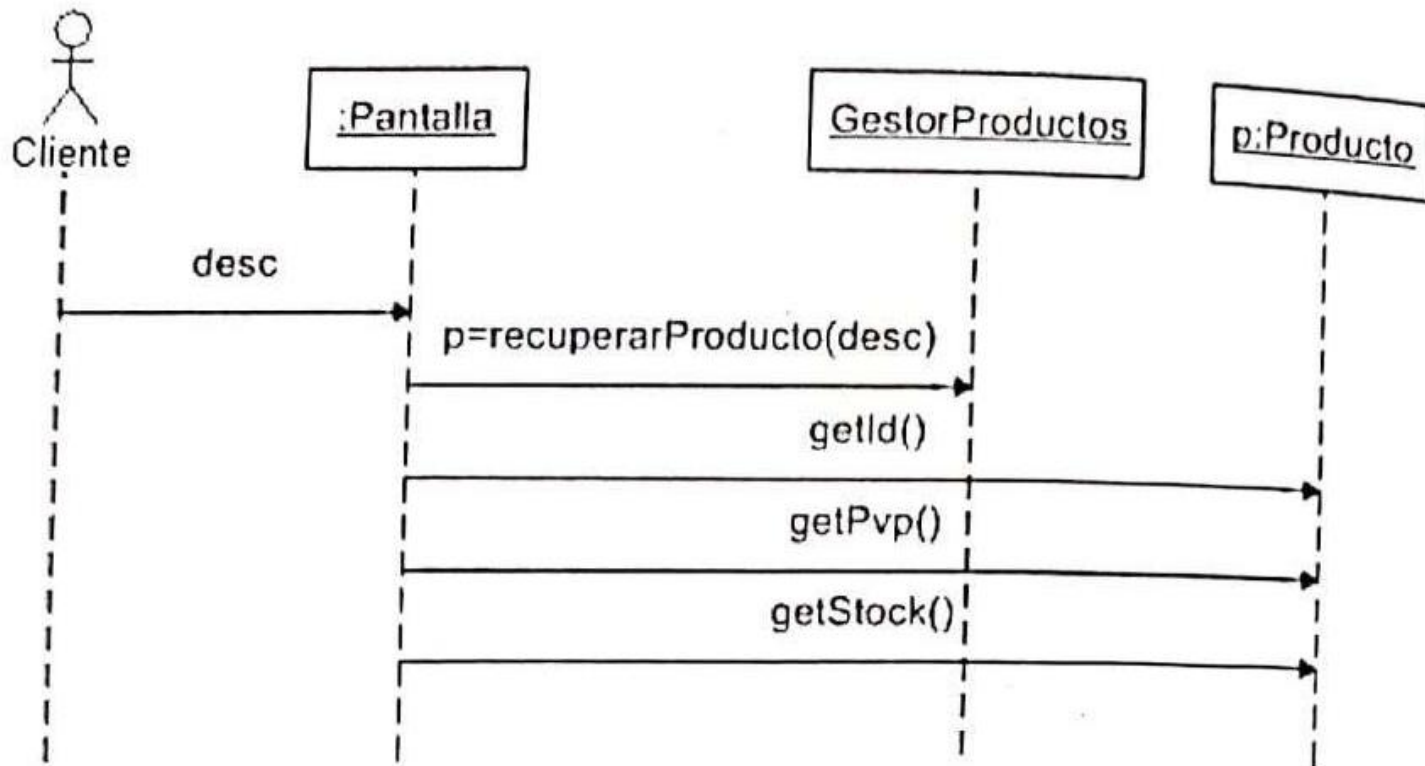
- ◉ Todas las clases definen métodos get y set para acceder a los datos.
- ◉ Se ha diseñado otra clase llamada **GestorProductos** con las operaciones a realizar sobre los productos:



- ◉ Se define también una clase **Pantalla** - interfaz gráfica desde la que el cliente se comunica con el sistema.

3.1 DIAGRAMA DE SECUENCIA

- El diagrama de secuencia correspondiente al caso de uso **BuscarProductos** es el siguiente:



3.1 DIAGRAMA DE SECUENCIA

◉ Explicación del diagrama anterior:

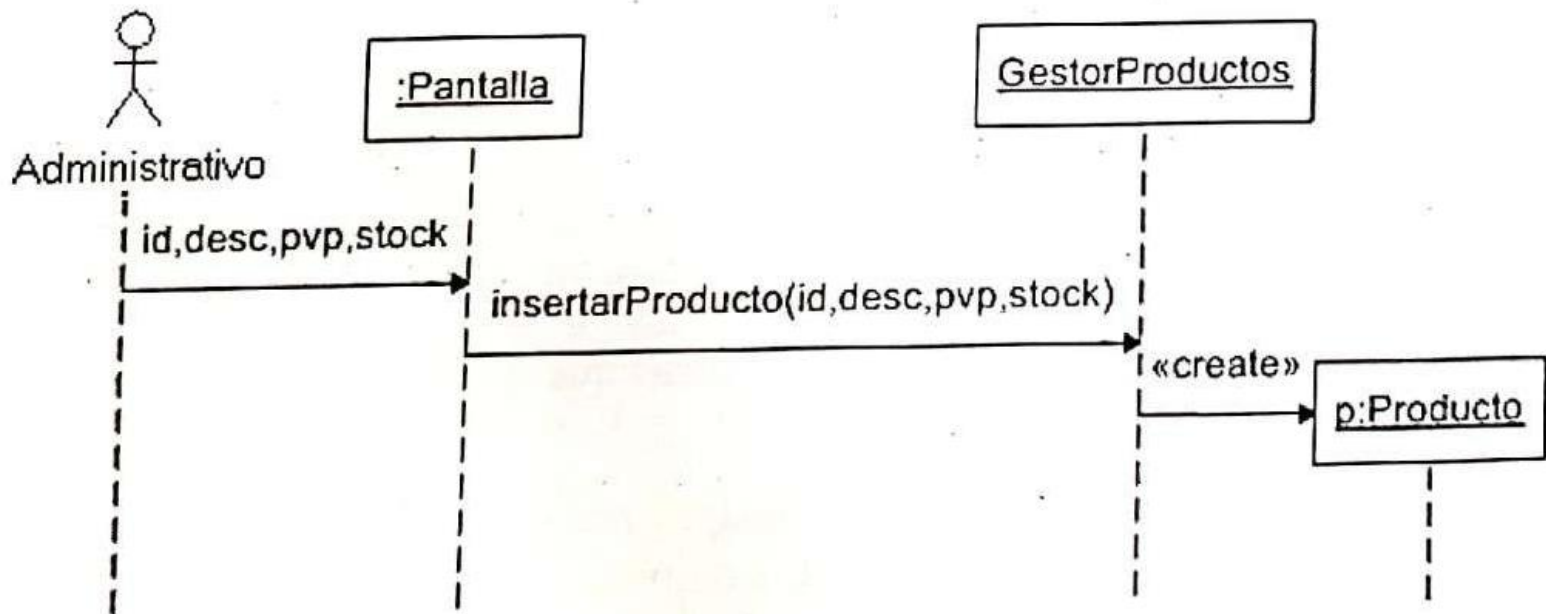
- El cliente escribe la descripción del producto que desea buscar en la pantalla gráfica (objeto de la clase Pantalla).
- El objeto pantalla ejecuta **recuperarProducto(desc)** en la clase **GestorProductos** obteniendo los datos del producto cuya descripción se envía. Esta operación devuelve un objeto de la clase **Producto**.
- El objeto pantalla envía los mensajes **getId()**, **getPvp()** y **gestStock()** al objeto **p:Producto** solicitando los datos del producto, el identificador, el pvp y el stock.

3.1 DIAGRAMA DE SECUENCIA

- ◉ En un diagrama de secuencia se puede mostrar la creación y destrucción de objetos.
- ◉ La creación se representa mediante un mensaje que termina en el objeto que va a ser creado.
- ◉ El mensaje puede llevar la identificación <<create>>.
- ◉ La destrucción del objeto da lugar a la finalización de su línea de vida, se denota mediante una X grande en su línea de vida.
- ◉ El mensaje puede llevar la identificación <<destroy>>.

3.1 DIAGRAMA DE SECUENCIA

- Ejemplo: este diagrama de secuencia corresponde al caso de uso de **InsertarProductos**.



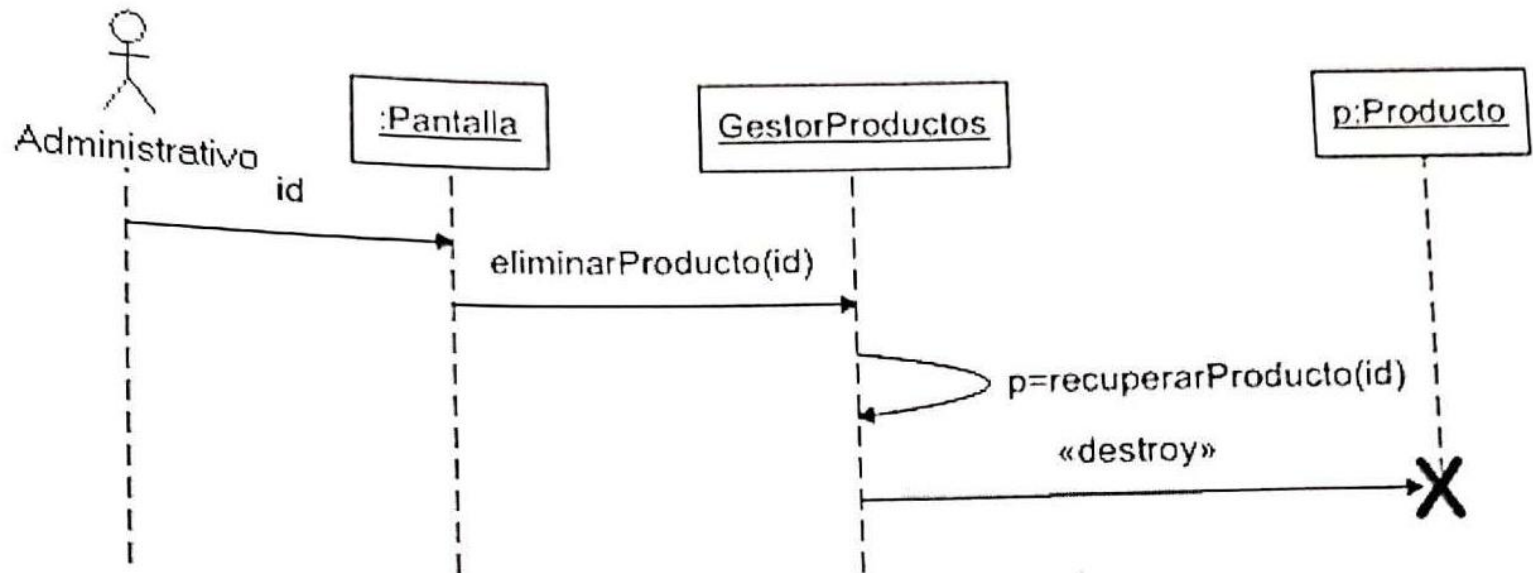
3.1 DIAGRAMA DE SECUENCIA

◉ Explicación del diagrama anterior:

- El administrativo escribe los datos del producto en la pantalla gráfica (objeto de la clase Pantalla).
- El objeto ejecuta `insertarProducto(id,desc,pvp,stock)` al gestor de productos (que se encarga de gestionar todas las operaciones para la gestión de productos) con los datos del producto a insertar.
- Desde el gestor de productos se creará el objeto **Producto** con los datos enviados por el objeto **Pantalla**.

3.1 DIAGRAMA DE SECUENCIA

- **Ejemplo:** este diagrama de secuencia corresponde al caso de uso de **EliminarProductos**.



3.1 DIAGRAMA DE SECUENCIA

◉ Explicación del diagrama anterior:

- El administrativo escribe el identificador del producto a eliminar en la pantalla gráfica.
- El objeto pantalla envía el mensaje eliminarProducto(id) al gestor de productos con el identificador del producto a eliminar.
- Desde el gestor de productos se recupera el objeto **Producto** y se elimina incluyendo la X en su línea de vida.
- En la línea de vida de la clase GestorProductos se muestra un mensaje enviado a sí mismo, es decir, dentro de la clase GestorProductos se hace una llamada a la operación recuperarProducto(id), a este mensaje se le llama **mensaje reflexivo**.

ACTIVIDAD

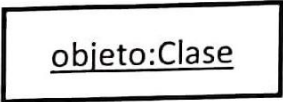

- Realiza los ejercicios 5 y 6 de la Hoja de Ejercicios.

3.2 DIAGRAMA DE COMUNICACIÓN O COLABORACIÓN

- ◉ El diagrama de colaboración muestra los objetos junto con los mensajes que se envían entre ellos.
- ◉ Representa la misma información que el diagrama de secuencia pero a diferencia de éste, el cual se centra en el **orden a través del tiempo** en que ocurren los mensajes, el diagrama de colaboración se centra en el **contexto y la organización general de los objetos** que se envían y reciben mensajes.

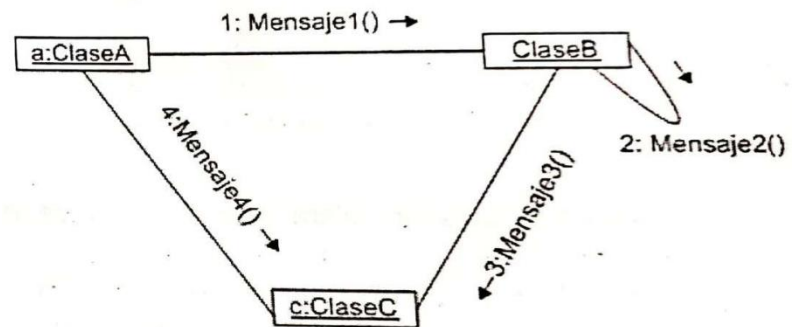
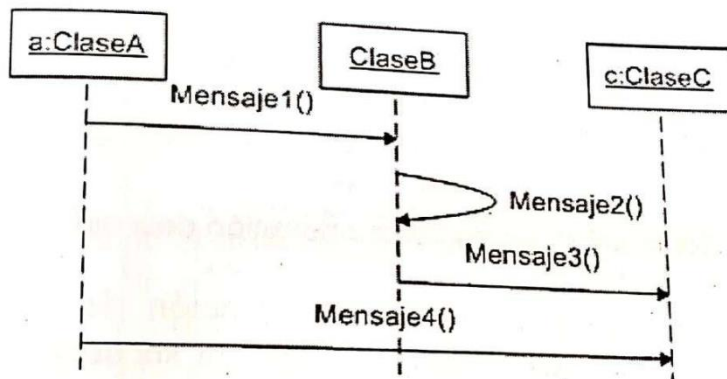
3.2 DIAGRAMA DE COMUNICACIÓN O COLABORACIÓN

- Un diagrama de colaboración es como un grafo con los siguientes elementos:

SÍMBOLO	FUNCIÓN	NOTACIÓN
Objetos o roles	Se representan mediante un rectángulo que contiene el nombre y la clase del objeto en el siguiente formato: <i>objeto:Clase</i> . Se puede omitir el nombre del objeto o la clase pero no ambos a la vez	
Enlaces	Son los arcos del grafo que conectan los objetos. En un mismo enlace se pueden mostrar muchos mensajes, pero cada uno de ellos con un número de secuencia único	
Mensajes	Se representan mediante una flecha dirigida con un nombre y un número de secuencia	1: Mensaje() →

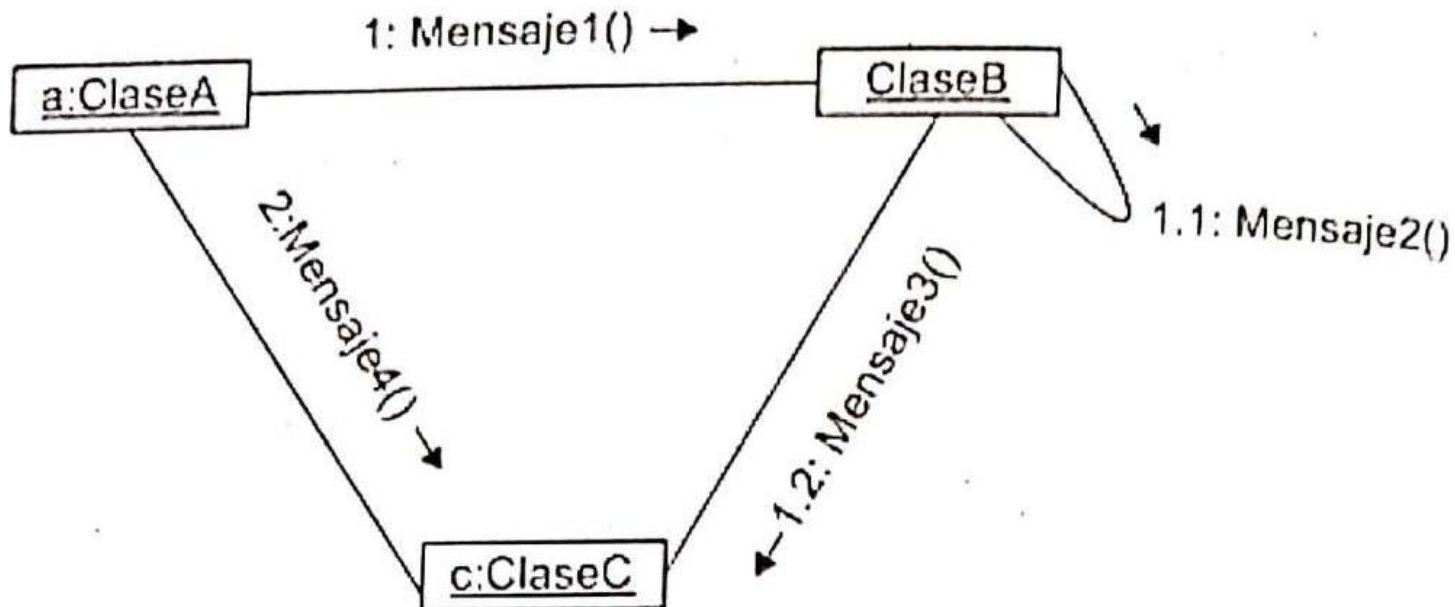
3.2 DIAGRAMA DE COMUNICACIÓN O COLABORACIÓN

- Para la numeración de los mensajes se pueden usar varios esquemas:
 - Numeración simple:** empieza en 1 y se va incrementando de 1 en 1 y no hay ningún nivel de anidamiento.



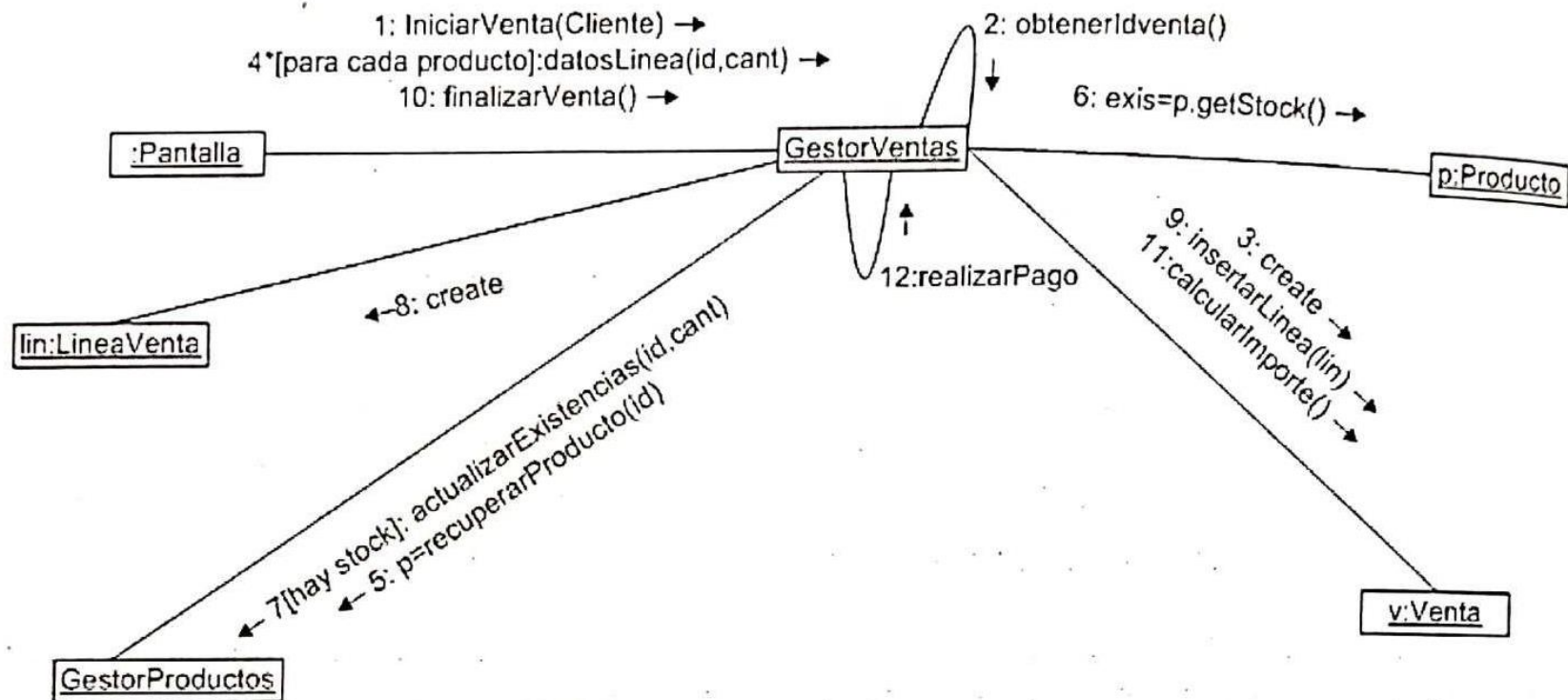
3.2 DIAGRAMA DE COMUNICACIÓN O COLABORACIÓN

- **Numeración decimal:** se muestran varios niveles de subíndices para indicar anidamiento de operaciones. Por ejemplo, 1 es el primer mensaje, 1.1 es el primer mensaje anidado en el mensaje 1, etc.



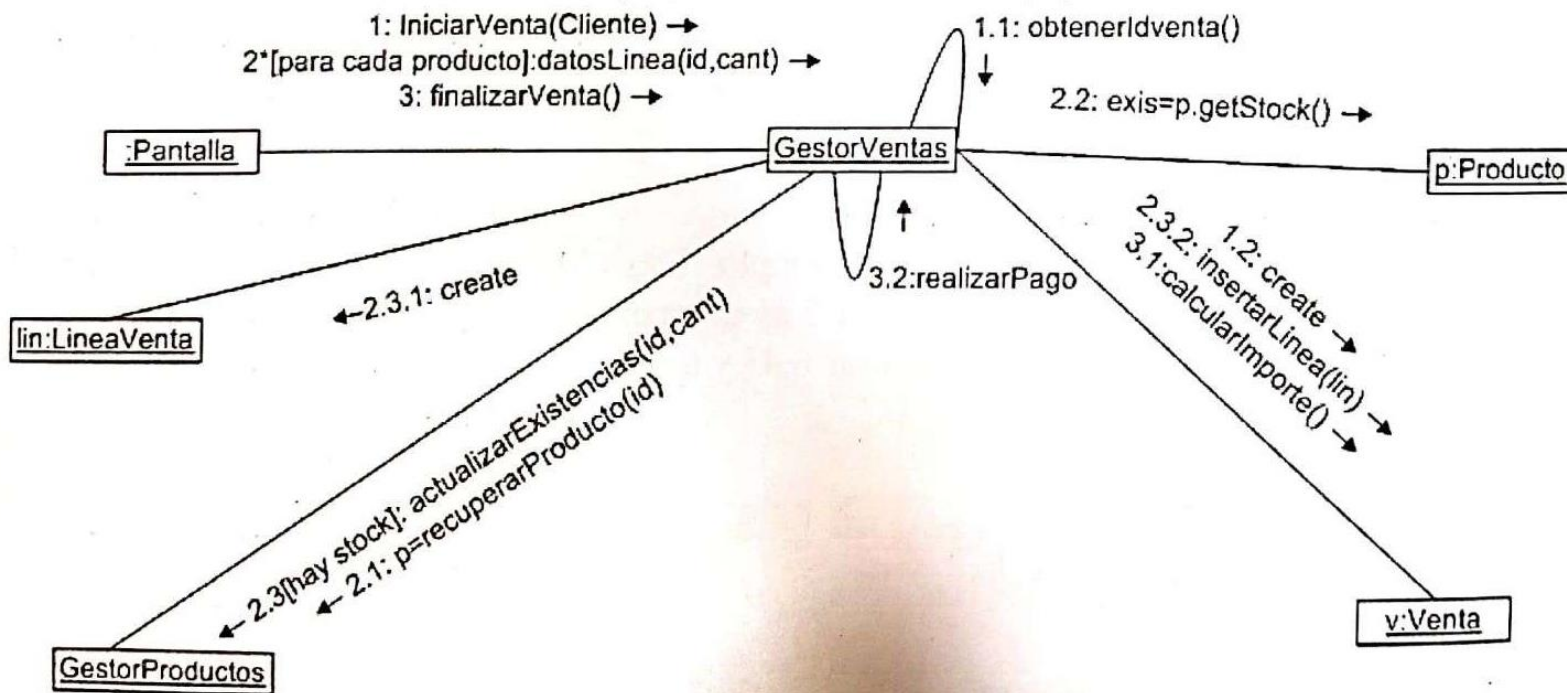
3.2 DIAGRAMA DE COMUNICACIÓN O COLABORACIÓN

- **Ejemplo:** diagrama de colaboración del caso de uso **ComprarProductos** con numeración simple



3.2 DIAGRAMA DE COMUNICACIÓN O COLABORACIÓN

- ◉ **Ejemplo:** diagrama de colaboración del caso de uso **ComprarProductos** con numeración decimal



ACTIVIDAD

- Realiza los ejercicios 7 y 8 de la Hoja de Ejercicios.