Indice

[**Tipos de Diagramas (UML)** 2](#_Toc7804503)

[1. Diagramas UML de comportamiento 2](#_Toc7804504)

[Diagramas de Caso de Uso 2](#_Toc7804505)

[Elementos Gráficos: 2](#_Toc7804506)

[Diagramas de Actividades 5](#_Toc7804507)

[Contenido del diagrama de actividades 5](#_Toc7804508)

[Diagramas de máquina de estados UML 11](#_Toc7804509)

[Símbolos y componentes para diagramas de estados 12](#_Toc7804510)

[2. Diagramas UML de Interacción 15](#_Toc7804511)

[Diagrama de Secuencia 15](#_Toc7804512)

[Elementos de un Diagrama de Secuencias 16](#_Toc7804513)

[Diagramas de Comunicación 20](#_Toc7804514)

[Contenidos de los diagramas de comunicación 20](#_Toc7804515)

[Diagrama de tiempos 23](#_Toc7804516)

[**BIBLIOGRAFIA** 25](#_Toc7804517)

# **Tipos de Diagramas (UML)**

# Diagramas UML de comportamiento

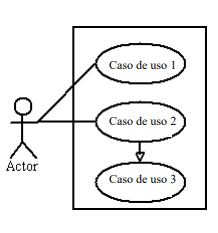
Un diagrama de comportamiento es un diagrama dinámico que tiene en cuenta elementos como la creación y destrucción de objetos, el paso de mensajes entre ellos y el orden en que deben hacerse, qué funcionalidad espera un usuario poder realizar, o como influyen elementos externos en nuestro sistema.

## Diagramas de Caso de Uso

En el [Lenguaje de Modelado Unificado](https://es.wikipedia.org/wiki/UML), un diagrama de casos de uso es una forma de diagrama de comportamiento UML mejorado. El [Lenguaje de Modelado Unificado](https://es.wikipedia.org/wiki/UML) (UML), define una [notación gráfica](https://es.wikipedia.org/w/index.php?title=Unling&action=edit&redlink=1) para representar casos de uso llamada modelo de casos de uso. UML no define estándares para que el formato escrito describa los [casos de uso](https://es.wikipedia.org/wiki/Caso_de_uso), y así mucha gente no entiende que esta notación gráfica define la naturaleza de un caso de uso; sin embargo una notación gráfica puede solo dar una vista general simple de un caso de uso o un conjunto de casos de uso. Los diagramas de casos de uso son a menudo confundidos con los casos de uso. Mientras los dos conceptos están relacionados, los casos de uso son mucho más detallados que los diagramas de casos de uso. En los conceptos se debe detallar más de un caso de uso para poder identificar qué es lo que hace un caso de uso.

Un caso de uso es una descripción de las acciones de un sistema desde el punto de vista del usuario. Es una herramienta valiosa dado que es una técnica de aciertos y errores para obtener los requerimientos del sistema, justamente desde el punto de vista del usuario. Los diagramas de caso de uso modelan la funcionalidad del sistema usando actores y casos de uso. Los casos de uso son servicios o funciones provistas por el sistema para sus usuarios.

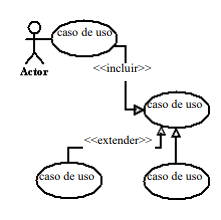
### Elementos Gráficos:

Sistema: El rectángulo represente los límites del sistema que contiene los casos de uso los actores se ubican fuera del rectángulo.

**Caso de Uso**: La elipse representa a un caso de uso, la etiqueta dentro de la elipse indica la función del sistema.

Función

**Actor**: una figura agregada representa al Actor, se utiliza de dos formas ala izquierda del caso de uso sería el que inicia y ala derecha sería el que recibe.

**Relaciones**: Las relaciones entre un actor y un caso de uso, se dibujan con una línea simple. Para relaciones entre casos de uso, se utilizan flechas etiquetadas "incluir" o "extender." Una relación "incluir" indica que un caso de uso es necesitado por otro para poder cumplir una tarea. Una relación "extender" indica opciones alternativas para un cierto caso de uso.

**Extensión** (extend)

Es otra forma de interacción, un caso de uso dado (la extensión) puede extender a otro. Esta relación indica que el comportamiento del caso de la extensión se utiliza en casos de uso, un caso de uso a otro caso siempre debe tener extensión o inclusión. El caso de uso extensión puede ser insertado en el caso de uso extendido bajo ciertas condiciones. La notación, es una flecha de punta abierta con línea discontinua, desde el caso de uso extensión al caso de uso extendido, con la etiqueta «extend». Esto puede ser útil para lidiar con casos especiales, o para acomodar nuevos requisitos durante el mantenimiento del sistema y su extensión.

"La extensión, es el conjunto de objetos a los que se aplica un concepto. Los objetos de la extensión son los ejemplos o instancias de los conceptos."

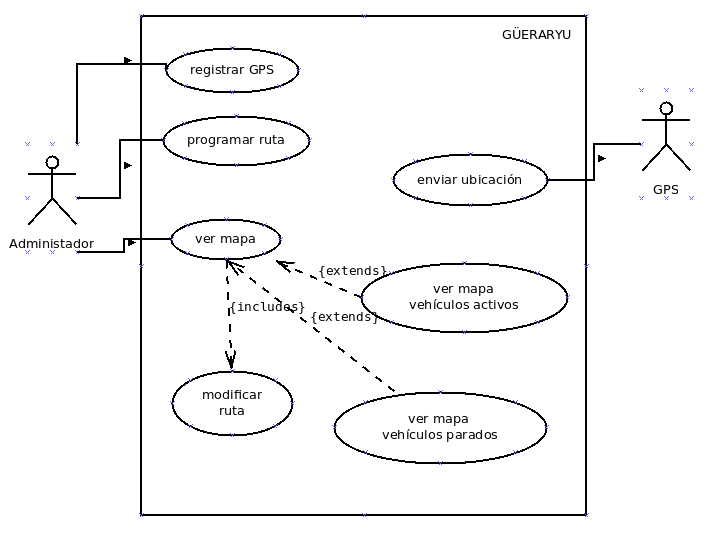
documentan el comportamiento de un sistema desde el punto de vista de un usuario

En otras palabras, será utilizado cuando un caso de uso sea similar a otro, pero con ciertas variaciones, un ejemplo claro es que se necesite comprar azúcar y podemos seleccionar de entre azúcar rubia, blanca o su unidad de medida bolsa, kilo, etc.

**Generalización**

"Entonces la Generalización es la actividad de identificar elementos en común entre conceptos y definir las relaciones de una superclase (concepto general) y subclase (concepto especializado). Es una manera de construir clasificaciones taxonómicas entre conceptos que entonces se representan en jerarquías de clases. Las subclases conceptuales son conformes con las superclases conceptuales en cuanto a la intención y extensión."

En la tercera forma de relaciones entre casos de uso, existe una relación generalización/especialización. Un caso de uso dado puede estar en una forma especializada de un caso de uso existente. La notación es una línea sólida terminada en un triángulo dibujado desde el caso de uso especializado al caso de uso general. Esto se asemeja al concepto orientado a objetos de sub-clases, en la práctica puede ser útil factorizar comportamientos comunes, restricciones al caso de uso general, describirlos una vez, y enfrentarse a los detalles excepcionales en los casos de uso especializados. UML



## Diagramas de Actividades

Un diagrama de actividades muestra el flujo de actividades, siendo una actividad una ejecución general entre los objetos que se está ejecutando en un momento dado dentro de una máquina de estados, el resultado de una actividad es una acción que producen un cambio en el estado del sistema o la devolución de un valor. Las acciones incluyen llamadas a otras operaciones, envío de señales, creación o destrucción de objetos o simples cálculos. Gráficamente un diagrama de actividades será un conjunto de arcos y nodos. Desde un punto de vista conceptual, el diagrama de actividades muestra cómo fluye el control de unas clases a otras con la finalidad de culminar con un flujo de control total que se corresponde con la consecución de un proceso más complejo. Por este motivo, en un diagrama de actividades aparecerán acciones y actividades correspondientes a distintas clases. Colaborando todas ellas para conseguir un mismo fin.

### Contenido del diagrama de actividades

Básicamente un diagrama de actividades contiene:

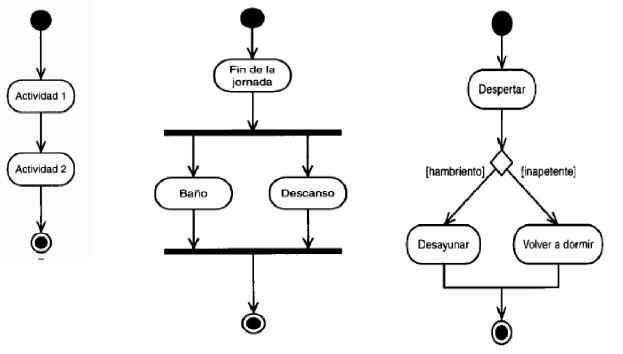
Estados de actividad

Estados de acción

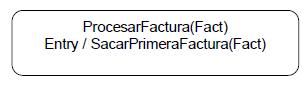
Transiciones

Objetos

**Estados de actividad y estados de acción**

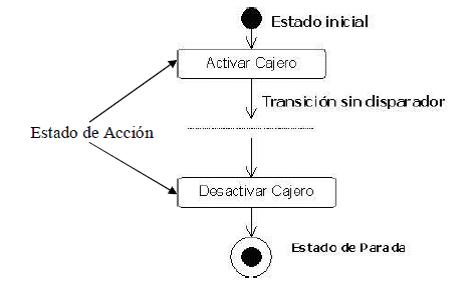
La representación de ambos es un rectángulo con las puntas redondeadas, en cuyo interior se representa bien una actividad o bien una acción. La forma de expresar tanto una actividad como una acción, no queda impuesta por UML, se podría utilizar lenguaje natural, una especificación formal de expresiones, un metalenguaje, etc. La idea central es la siguiente: “Un estado que represente una acción es atómico, lo que significa que su ejecución se puede considerar instantánea y no puede ser interrumpida”, Al igual que en el diagrama de estados, el de actividad cuenta con un punto inicial (representado por un círculo) y uno final (representado como dos círculos concéntricos). En la siguiente figura, podemos ver ejemplos de estados de acción.

En cambio, un estado de actividad, sí puede descomponerse en más sub- actividades representadas a través de otros diagramas de actividades. Además, estos estados sí pueden ser interrumpidos y tardan un cierto tiempo en completarse. En los estados de actividad podemos encontrar otros elementos adicionales como son: acciones de entrada (entry) y de salida (exit) del estado en cuestión, así como definición de submáquinas, como podemos ver en la figura siguiente:



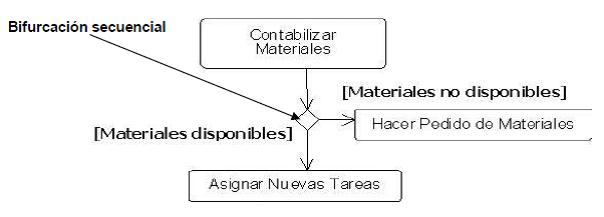
**Transiciones**

Las transiciones reflejan el paso de un estado a otro, bien sea de actividad o de acción. Esta transición se produce como resultado de la finalización del estado del que parte el arco dirigido que marca la transición. Como todo flujo de control debe empezar y terminar en algún momento, podemos indicar esto utilizando dos disparadores de inicio y fin tal y como queda reflejado en el ejemplo siguiente.

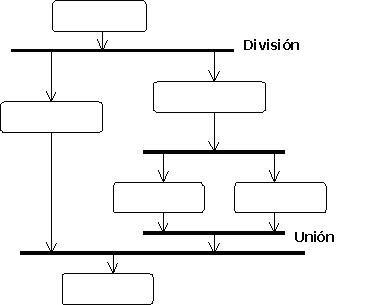


**Bifurcaciones**

Un flujo de control no tiene porqué ser siempre secuencial, puede presentar caminos alternativos. Para poder representar dichos caminos alternativos o bifurcación se utilizará como símbolo el rombo. Dicha bifurcación tendrá una transición de entrada y dos o más de salida. En cada transición de salida se colocará una expresión booleana que será evaluada una vez al llegar a la bifurcación, las guardas de la bifurcación han de ser excluyentes y contemplar todos los casos ya que de otro modo la ejecución del flujo de control quedaría interrumpida. Para poder cubrir todas las posibilidades se puede utilizar la palabra ELSE, para indicar una transición obligada a un determinado estado cuando el resto de guardas han fallado. Veamos un ejemplo de bifurcación.

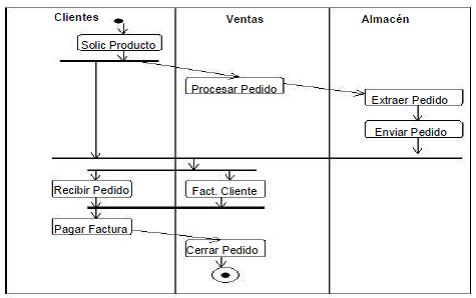


**División y unión**

No sólo existe el flujo secuencial y la bifurcación, también hay algunos casos en los que se requieren tareas concurrentes. UML representa gráficamente el proceso de división, que representa la concurrencia, y el momento de la unión de nuevo al flujo de control secuencial, por una línea horizontal ancha. Podemos ver cómo se representa gráficamente.

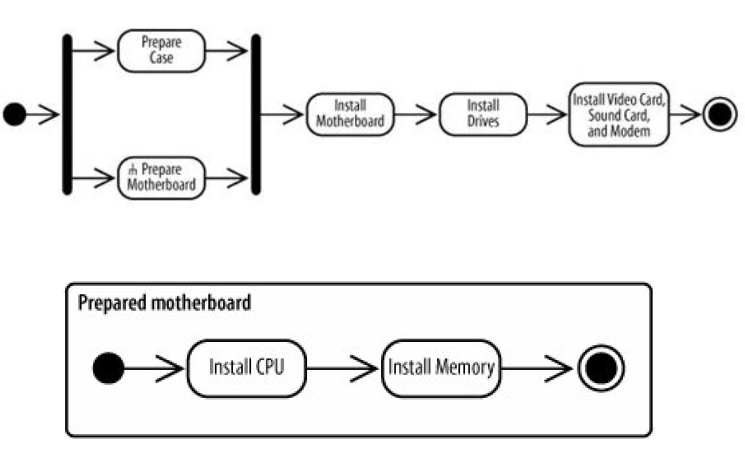
**Calles**

Cuando se modelan flujos de trabajo de organizaciones, es especialmente útil dividir los estados de actividades en grupos, cada grupo tiene un nombre concreto y se denominan calles. Cada calle representa a la parte de la organización responsable de las actividades que aparecen en esa calle. Gráficamente quedaría como se muestra a continuación.



**Llamando a otras actividades**

A medida que se agregan detalles a un diagrama de actividades, el diagrama puede llegar a ser demasiado grande, o la misma secuencia de acciones se puede producir más de una vez

Cuando esto sucede, se puede mejorar la legibilidad de los detalles de una acción en un diagrama por separado, lo que permite el diagrama de nivel superior, permanecer menos desordenado

**Nodos Inicio y Terminación**

El nodo Inicial, es el punto de partida para el flujo de una actividad

Puede haber un nodo por diagrama

El nodo final, hace que el flujo definido termine, es decir, todas las acciones y los flujos dentro de esa actividad se cerrarán

Puede haber muchos nodos finales

**Conectores**

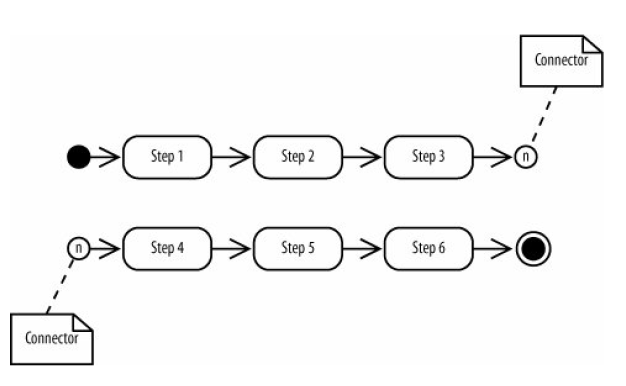
Un conector se dibuja como un círculo con su nombre escrito en el interior.

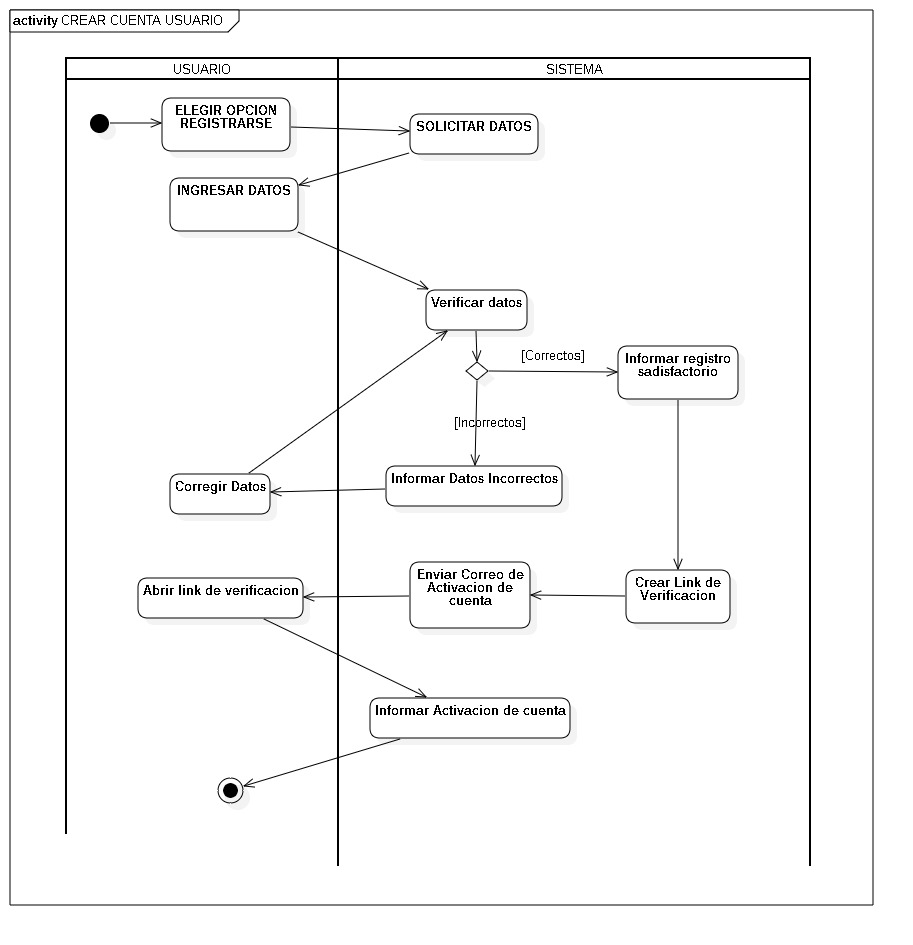
Los conectores se suele tener nombres de un solo carácter.

Conectores vienen en pares.

Uno cuenta con una flecha de entrada y el otro con una saliente.

El segundo conector comienza donde el primer conector quedó.



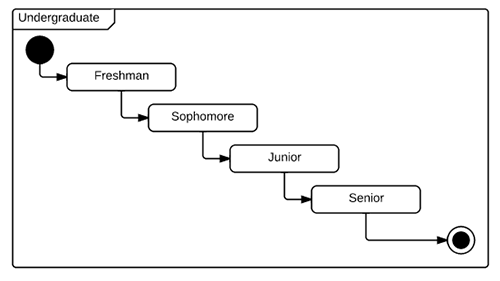


## Diagramas de máquina de estados UML

Los diagramas de máquina de estados ofrecen un método orientado a objetos de mostrar el comportamiento de un objeto y documentar cómo el objeto responde a determinados eventos, incluidos estímulos internos y externos.

Una máquina de estados es cualquier dispositivo que almacena el estado de un objeto en un momento dado y puede cambiar el estado o causar otras acciones según la entrada que reciba. Estados se refiere a las diferentes combinaciones de información que un objeto puede mantener, no la forma en que el objeto se comporta. Para comprender los diferentes estados de un objeto, podrías visualizar todos los estados posibles y mostrar cómo un objeto llega a cada estado, y puedes hacerlo con un diagrama de estados UML.

Cada diagrama de estados generalmente empieza con un círculo oscuro que indica el estado inicial y termina con un círculo de contorno blanco que denota el estado final. Sin embargo, a pesar de tener puntos de inicio y finalización definidos, los diagramas de estado no necesariamente son la mejor herramienta para plasmar un desarrollo general de eventos. En lugar de ello, ilustran tipos específicos de comportamiento —en particular, cambios de un estado a otro.

Los diagramas de estado representan principalmente estados y transiciones. Los estados se representan con rectángulos de esquinas redondeadas que se etiquetan con el nombre del estado. Las transiciones se marcan con flechas que fluyen de un estado a otro, mostrando cómo cambian los estados. A continuación, podrás ver estos dos elementos en acción en un diagrama básico para la vida estudiantil. Nuestra [herramienta de diagramas UML](https://www.lucidchart.com/pages/examples/uml_diagram_tool) puede ayudarte a diseñar cualquier diagrama personalizado de máquina de estados.

### Símbolos y componentes para diagramas de estados

Puedes incluir muchas figuras diferentes en un diagrama de estados, particularmente si eliges combinarlo con otro diagrama. Esta lista resume las figuras más comunes que puedes encontrar.

**Estado compuesto**

Un estado que contiene sub-estados anidados. Ve el ejemplo siguiente de [diagrama de estados de universidad](https://www.lucidchart.com/pages/uml-state-machine-diagram#section-3). “Inscripción” es el estado compuesto en este ejemplo porque comprende diversos sub-estados en el proceso.

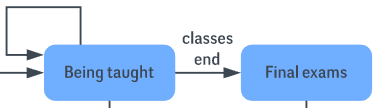
**Pseudoestado de opción**

Un símbolo de diamante que indica una condición dinámica con resultados potenciales ramificados.



**Evento**

Una instancia que activa una transición, etiquetada arriba de la flecha de transición aplicable. En este caso, "fin de clases" es el evento que activa el final del estado “Siendo instruidos” y el inicio del estado “Exámenes finales”.



**Punto de salida**

El punto en el cual un objeto escapa el estado compuesto o máquina de estados, el cual se indica por medio de un círculo cruzado con una X. El punto de salida generalmente se usa si el proceso no está completado, pero tiene que ser escapado por algún error u otro problema.

Símbolos de diagramas de estados - Punto de salida

**Primer estado**

Un marcador para el primer estado en el proceso, que se muestra mediante un círculo oscuro con una flecha de transición.



**Protección**

Una condición booleana que permite o detiene una transición. Se escribe arriba de la flecha de transición.

**Estado**

Un rectángulo de esquinas redondeadas que indica la naturaleza actual de un objeto.



**Sub-estado**

Un estado contenido dentro de la región de un estado compuesto. En el [diagrama de máquina de estados de universidad](https://www.lucidchart.com/pages/uml-state-machine-diagram#section-3) mostrado a continuación, “Abierto para inscripción” es un sub-estado en el estado compuesto más grande de “Inscripción”.

**Terminador**

Un círculo con un punto en el interior que indica que un proceso está terminado.



**Transición**

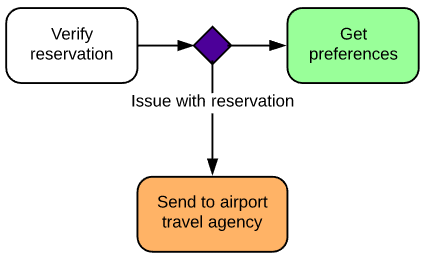
Una flecha que corre de un estado a otro, que indica un estado cambiante.

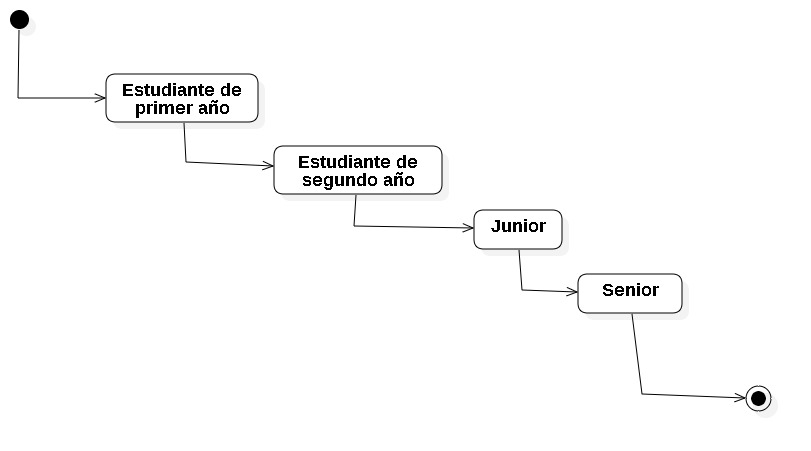


**Comportamiento transicional**

Un comportamiento que resulta cuando un estado pasa por una transición. Se escribe arriba de la flecha de transición.

**Disparador**

Un tipo de mensaje que mueve activamente un objeto de estado en estado. Se escribe arriba de la flecha de transición. En este ejemplo, “Problema con la reservación” es el disparador que enviaría a la persona a la agencia de viajes del aeropuerto en lugar de al siguiente paso en el proceso.

**Ejemplo**

# Diagramas UML de Interacción

Un diagrama de interacción describe en detalle un determinado escenario de un caso de uso. En él se muestra la interacción entre el conjunto de objetos que cooperan en la realización de dicho escenario. Suele ser conveniente especificar en la parte izquierda del diagrama el caso de uso que se está́ representando para que resulte más sencilla su validación.

Los elementos que componen los diagramas de interacción son los objetos y los mensajes:

Un objeto es una entidad que tiene un estado, un comportamiento e identidad. La estructura y el comportamiento común de diferentes objetos se recoge en una clase. En un diagrama de interacción, los objetos serán al final instancias de una determinada clase o de un actor.

Un mensaje es una comunicación entre dos objetos. El envío de un mensaje por parte de un objeto (emisor) a otro (receptor), puede provocar que se ejecute una operación, se produzca un evento o se cree o destruya un objeto.

## Diagrama de Secuencia

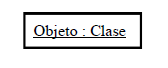
Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. A menudo es útil para complementar a un [diagrama de clases](https://es.wikipedia.org/wiki/Diagrama_de_clases), pues el diagrama de secuencia se podría describir de manera informal como "el diagrama de clases en movimiento", por lo que ambos deben estar relacionados entre sí (mismas clases, métodos, atributos...). Mientras que el [diagrama de casos de uso](https://es.wikipedia.org/wiki/Diagrama_de_casos_de_uso) permite el modelado de una vista de negocio del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario y mensajes intercambiados entre los objetos.

Típicamente se examina la descripción de un [caso de uso](https://es.wikipedia.org/wiki/Caso_de_uso) para determinar qué objetos son necesarios para la implementación del escenario. Si se dispone de la descripción de cada [caso de uso](https://es.wikipedia.org/wiki/Caso_de_uso) como una secuencia de varios pasos, entonces se puede "caminar sobre" esos pasos para descubrir qué objetos son necesarios para que se puedan seguir los pasos. Un diagrama de secuencia muestra los objetos que intervienen en el escenario con líneas discontinuas verticales, y los mensajes pasados entre los objetos como flechas horizontales.

### Elementos de un Diagrama de Secuencias

Rol de la Clase

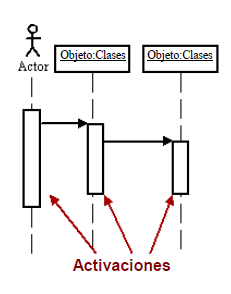
El rol de la clase describe la manera en que un objeto se va a comportar en el contexto. No se listan los atributos del objeto.

[](https://ingsotfwarekarlacevallos.files.wordpress.com/2015/07/23.png)

Objeto de una clase

**Activación**

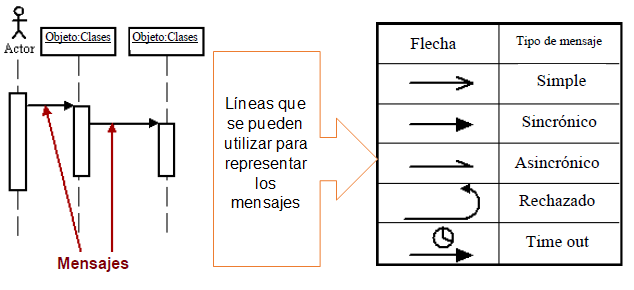
Los cuadros de activación representan el tiempo que un objeto necesita para completar una tarea.

[](https://ingsotfwarekarlacevallos.files.wordpress.com/2015/07/33.png)

Activación de una clase

**.  Mensajes**

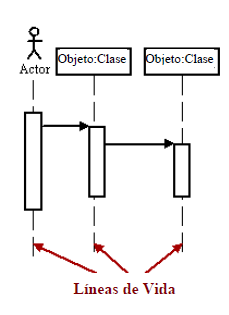
Los mensajes son flechas que representan comunicaciones entre objetos. Las medias flechas representan mensajes asincrónicos. Los mensajes asincrónicos son enviados desde un objeto que no va a esperar una respuesta del receptor para continuar con sus tareas.

[](https://ingsotfwarekarlacevallos.files.wordpress.com/2015/07/44.png)

Mensajes

**Líneas de Vida**

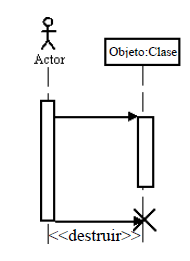
Las líneas de vida son verticales y en línea de puntos, ellas indican la presencia del objeto durante el tiempo.

[](https://ingsotfwarekarlacevallos.files.wordpress.com/2015/07/54.png)

Linea de vida

**Destrucción de Objetos**

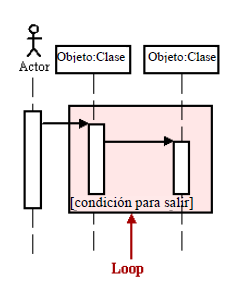
Los objetos pueden ser eliminados tempranamente usando una flecha etiquetada “<<destruir>>” que apunta a una X.

[](https://ingsotfwarekarlacevallos.files.wordpress.com/2015/07/62.png)

Destrucción de objetos

**Loops**

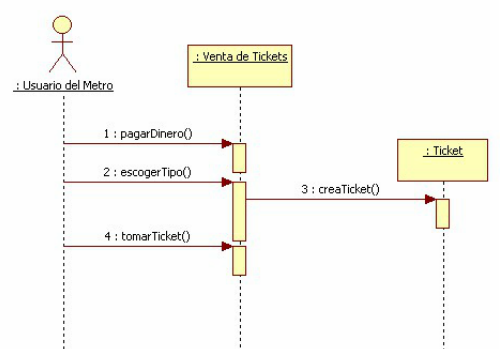
Una repetición o loop en un diagrama de secuencias, es representado como un rectángulo. La condición para abandonar el loop se coloca en la parte inferior entre corchetes [ ].

[](https://ingsotfwarekarlacevallos.files.wordpress.com/2015/07/72.png)

Loop

**Ejemplo**

En el siguiente ejemplo se muestra la secuencia que sigue un usuario del metro para comprar un ticket:



Ejemplo de la secuencia de un usuario del metro para comprar un ticket

## Diagramas de Comunicación

Los diagramas de comunicación UML se parecen a los de secuencia, pero ofrecen una visión de conjunto de las relaciones entre los objetos, en lugar de centrarse en el orden de los mensajes, a medida que se ejecuta su software.

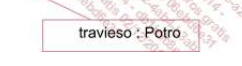
Los diagramas de comunicación tienen un diseño libre para que pueda colocar los objetos de forma que las relaciones entre ellos queden resaltadas. Si quiere, puede crear todos los objetos relacionados con la interacción y colocarlos de forma lógica antes de crear ningún mensaje. Los diagramas de comunicación suelen describir una instancia o una única ruta de acceso de ejecución del programa.

### Contenidos de los diagramas de comunicación

Puede tener objetos e instancias de actor en diagramas de comunicación, junto con enlaces y mensajes que describen cómo están relacionados y cómo interactúan. El diagrama describe lo que ocurre en los objetos participantes, respecto a cómo se comunican los objetos enviando mensajes entre sí. Puede realizar un diagrama de comunicación para cada variante del flujo de sucesos del guion de uso.

Un diagrama de comunicación que describe parte del flujo de sucesos del guión de uso Recibir elemento de depósito en el Sistema de máquina de reciclaje.

Objetos

Un objeto se representa con un símbolo de objeto que muestra el nombre del objeto y su clase subrayada, separados por dos puntos:

objectname : classname

Puede utilizar objetos en diagramas de comunicación de los modos siguientes:

La clase de un objeto puede estar sin especificar. Habitualmente, crea un diagrama de comunicación con objetos primero y especifica sus clases posteriormente.

Los objetos pueden no tener nombre, pero debería nombrarlos si desea discriminar diferentes objetos de la misma clase.

La clase de un objeto puede representarse en un diagrama de comunicación, si participa activamente en la interacción.

**Actores**

Normalmente, una instancia de actor se produce en el diagrama de comunicación, como la parte que invoca la interacción. Si dispone de varias instancias de actor en el mismo diagrama, intente mantenerlas en la periferia del diagrama.

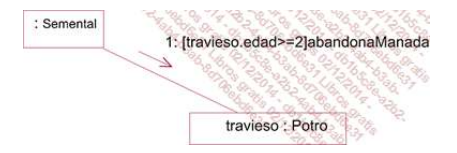
**Enlaces**

Los enlaces se definen del modo siguiente:

Un enlace es una relación entre objetos a través de la cual se pueden enviar mensajes. En diagramas de comunicación, un enlace se muestra como una línea sólida entre dos objetos.

Un objeto interactúa con, o navega hasta, otros objetos a través de sus enlaces con estos objetos.

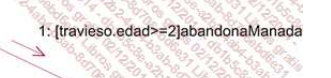
Un enlace puede ser una instancia de una asociación o bien puede ser anónimo, lo que significa que la asociación no está especificada.

Los flujos de mensaje se conectan a enlaces, consulte Mensajes.

**Mensajes**

Un mensaje es una comunicación entre objetos que transfiere información con la expectativa de que esa actividad se llevará a cabo. En diagramas de comunicación, un mensaje se muestra como una flecha etiquetada junto a un enlace. Esto significa que el enlace se utiliza para transportar, o para implementar la entrega del mensaje al objeto de destino. La flecha dirige a lo largo del enlace en la dirección del objeto de destino (el que recibe el mensaje). La flecha está etiquetada con el nombre del mensaje, y sus parámetros. La flecha también puede estar etiquetada con un número de secuencia para mostrar la secuencia del mensaje en la interacción global. Los números de secuencia suelen utilizarse en los diagramas de comunicación, porque son el único modo de describir la secuenciación relativa de los mensajes.

Un mensaje puede estar sin asignar, lo que significa que su nombre es una cadena temporal que describe el significado general del mensaje. Posteriormente, podrá asignar el mensaje especificando la operación del objeto de destino del mensaje. La operación especificada reemplazará entonces el nombre del mensaje.



**Ejemplo**



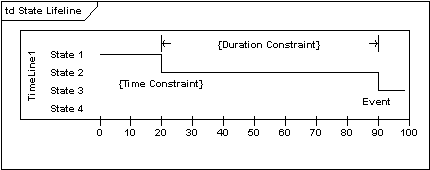
## Diagrama de tiempos

Un diagrama de tiempos o cronograma es una gráfica de formas de onda digitales que muestra la relación temporal entre varias [señales](https://es.wikipedia.org/wiki/Se%C3%B1al_digital), y cómo varía cada señal en relación a las demás.

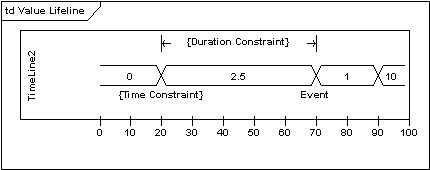
Un cronograma puede contener cualquier número de señales relacionadas entre sí. Examinando un diagrama de tiempos, se puede determinar los estados, nivel alto o nivel bajo, de cada una de las señales en cualquier instante de tiempo especificado, y el instante exacto en que cualquiera de las señales cambia de estado con respecto a las restantes.

El propósito primario del diagrama de tiempos es mostrar los cambios en el estado o la condición de una línea de vida (representando una Instancia de un Clasificador o un Rol de un clasificador) a lo largo del tiempo lineal. El uso más común es mostrar el cambio de estado de un objeto a lo largo del tiempo, en respuesta a los eventos o estímulos aceptados. Los eventos que se reciben se anotan, a medida que muestran cuándo se desea mostrar el evento que causa el cambio en la condición o en el estado.

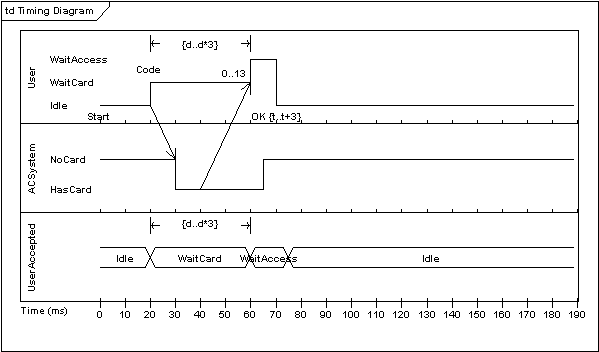
**Línea de vida del estado**  
Una línea de vida del estado muestra el cambio de estado de ítem en el tiempo. El eje-X muestra el tiempo trascurrido en cualquier unidad que se elija mientras que el eje-Y se nombra con una lista de estados proporcionados. El siguiente es un ejemplo de una línea de vida del estado.



**Línea de vida del valor**Una línea de vida del valor muestra el cambio del valor de un ítem en el tiempo. El eje-X muestra el tiempo transcurrido en cualquier unidad que se elija, lo mismo que para la línea de vida del estado. El valor se muestra entre el par de líneas horizontales que se cruzan en cada cambio del valor. El siguiente es un ejemplo de una línea de vida del valor.



**Ubicar todo junto**  
Las líneas de vida y del estado se pueden ubicar una arriba de otro en cualquier combinación. Estas deben tener el mismo eje-X. Los mensajes se pueden pasar de una línea de vida a otra. Cada transición del estado o valor puede tener un evento definido, una restricción de tiempo que indica cuándo debe ocurrir un evento, y una restricción de duración que indica cuánto tiempo debe estar en efecto un valor o estado. Una vez que estos se hayan aplicado, un diagrama de tiempo debería ser como el siguiente.



# **BIBLIOGRAFIA**

<https://moodle2.unid.edu.mx/dts_cursos_mdl/pos/TI/IS/AM/10/comunicacion_uml.pdf>

<http://www.vc.ehu.es/jiwotvim/IngenieriaSoftware/Teoria/BloqueII/UML-4.pdf>

<https://manuel.cillero.es/doc/metrica-3/tecnicas/diagrama-de-interaccion/>

[file:///C:/Users/MXCATV5/Desktop/262567348-Diagramas-de-Comportamiento.pdf](file:///C:\Users\MXCATV5\Desktop\262567348-Diagramas-de-Comportamiento.pdf)

<https://www.teatroabadia.com/es/uploads/documentos/iagramas_del_uml.pdf>

[file:///C:/Users/MXCATV5/Desktop/4.%20Diagramas%20de%20Comportamiento%20(1).pdf](file:///C:\Users\MXCATV5\Desktop\4.%20Diagramas%20de%20Comportamiento%20(1).pdf)

<https://empleo.webcindario.com/programacion/cd/modelado.html>

<https://es.slideshare.net/orjova/diagramas-de-comportamientos>

<https://jamj2000.gitbook.io/daw1-ed-apuntes/6.diagramas_comportamiento>

<https://es.wikipedia.org/wiki/Diagrama_de_casos_de_uso>

<http://stadium.unad.edu.co/ovas/10596_9839/diagramas_de_actividades.html>

<https://www.ibiblio.org/pub/Linux/docs/LuCaS/Tutoriales/doc-modelado-sistemas-UML/multiple-html/x291.html>

<https://www.lucidchart.com/pages/es/diagrama-de-maquina-de-estados>

<https://www.lucidchart.com/pages/es/diagrama-de-secuencia>

<https://es.wikipedia.org/wiki/Diagrama_de_secuencia>

<https://ingsotfwarekarlacevallos.wordpress.com/2015/07/07/uml-diagrama-de-secuencia/><https://es.wikipedia.org/wiki/Diagrama_de_comunicaci%C3%B3n>

<https://www.altova.com/es/umodel/communication-diagrams>

<https://cgrw01.cgr.go.cr/rup/RUP.es/SmallProjects/core.base_rup/guidances/guidelines/communication_diagram_FFFEA1B5.html>

<https://es.wikipedia.org/wiki/Diagrama_de_tiempos>

<http://www.sparxsystems.com.ar/resources/tutorial/uml2_timingdiagram.htm>

Manual UML Paul Kimmel

Aprendiendo UML en 24 horas Joseph Schmuller