

# UML

TEORÍA DE DIAGRAMA DE SECUENCIA

DISEÑO DE APLICACIONES | 2021



## Contenido

Diagrama de Secuencia	2
Componentes del Diagrama de Secuencia	2
Línea de vida:	2
Mensaje	3
Creación y destrucción de líneas de vida	6
Parámetros	5
Pasos para la construcción de un diagrama de secuencia	6
Lineamientos generales	7

## Diagrama de Secuencia

Un diagrama de secuencia es un tipo de diagrama de comportamiento de UML que se utiliza para modelar la interacción entre objetos a través del tiempo.

Se modela para cada caso de uso. Informalmente decimos que un diagrama de secuencia es "el diagrama de clases en movimiento".

Mientras que el diagrama de casos de uso, como vimos anteriormente, permite tener una vista del modelo de negocio de un escenario particular, el diagrama de secuencia contiene detalles de implementación de dicho escenario, incluyendo las clases que se usan para implementarlo y los mensajes intercambiados entre los distintos objetos.

En resumen, muestra un conjunto de líneas de vida y los mensajes enviados y recibidos por estas, enfatizando el orden de los mensajes en función del tiempo.

Se modela cómo colaboran e interactúan las líneas de vida mediante el paso de mensajes a lo largo del tiempo para conseguir un resultado.

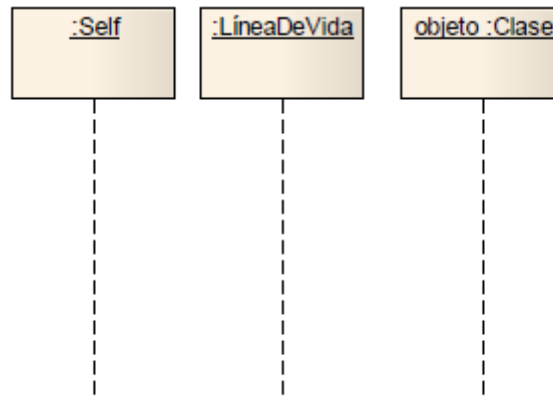
Al ser un diagrama de interacción, éste demuestra cómo los objetos cooperan para realizar los diferentes comportamientos definidos en un caso de uso. La secuencia de los mensajes queda representada por el orden de estos.

## Componentes del Diagrama de Secuencia

### Línea de vida:

Representa a un participante en una interacción. Es una instancia de un clasificador específico (clases, interfaces, componentes, actores) que participa en una interacción. Un ejemplo de línea de vida es un objeto, que constituye una instancia del clasificador clase.

Las líneas de vida se representan mediante un icono representativo del tipo, su nombre, dos puntos y el nombre del clasificador (subrayado), y una línea vertical punteada.



*Líneas de vida de los objetos en un Diagrama de Secuencia*

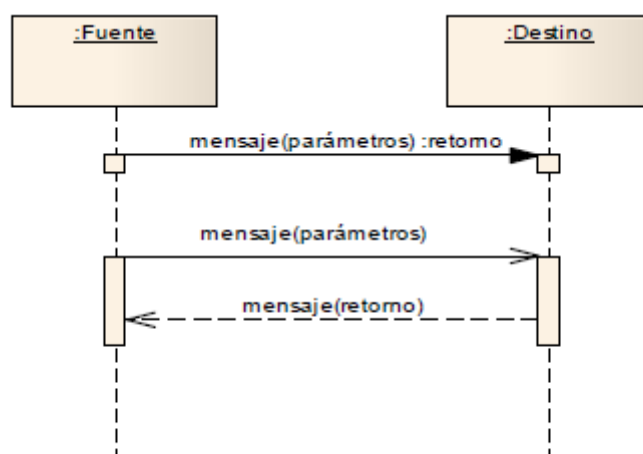
### Mensaje

Es un tipo específico de comunicación entre dos líneas de vida en una interacción; puede representar:

- la llamada a una operación, invocando un mensaje específico.
- la creación/destrucción de una instancia, invocando un mensaje de creación o destrucción.
- el envío de una señal.




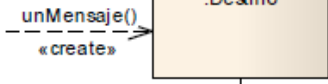
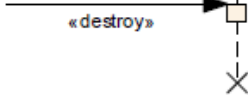
Cuando una línea de vida recibe un mensaje, éste constituye la invocación de una determinada operación que tiene la misma firma (signatura) que el mensaje. Por lo tanto, deberá existir una correspondencia entre la firma (signature) del mensaje y de la operación del objeto. Al recibir el mensaje se inicia una actividad en el objeto receptor que se llama ocurrencia de ejecución. Una ocurrencia de ejecución muestra la línea de vida de los objetos que la ejecutan, en un momento de tiempo. Un objeto activado, o bien está ejecutando su propio código o está esperando el retorno de otro objeto al cual se le ha enviado un mensaje. Esta ocurrencia de ejecución es opcional en el diagrama.

Un mensaje se grafica como una flecha horizontal entre dos líneas de vida.



*Tipos de mensajes que pueden utilizarse en un Diagrama de Secuencia*

El formato de la flecha variará según el tipo de mensaje:

Sintaxis	Nombre	Descripción
 unMensaje(unParámetro)	<b>Mensaje Síncrono</b>	El emisor espera que el receptor finalice la ejecución del mensaje.
 unMensaje(unParámetro)	<b>Mensaje Asíncrono</b>	El emisor envía el mensaje y continúa ejecutando –éste no espera que el retorno del receptor.
	<b>Mensaje de Retorno</b>	El receptor de un mensaje enviado retorna el foco de control al emisor.
	<b>Creación de Objeto</b>	El emisor crea una instancia del clasificador especificado por el receptor. Se puede graficar sin necesidad de indicar el mensaje creador – sólo con el estereotipo « <b>create</b> ».
	<b>Destrucción de Objeto</b>	El emisor destruye la instancia del receptor. Este mensaje no retorna valor.

¿Cómo se construye?:

Un diagrama de secuencia sencillo se crea a través de una serie de pasos y luego se podrá refinar utilizando adornos de modelado más avanzados que proporcionan mayor detalle y compresión al lector:

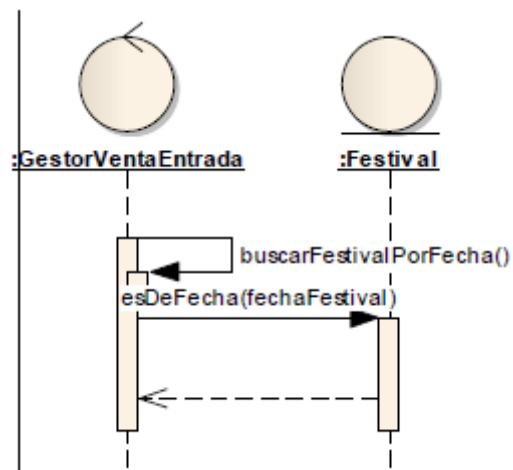
1. En primer lugar, cada línea de vida que participa en la interacción será colocada a lo largo del eje X a la altura temporal representativa de su creación. Normalmente se ubica a la izquierda el objeto que inicia la interacción y los objetos subordinados a la derecha.  
Entonces si este diagrama fuera una realización de caso de uso de análisis, estaríamos ubicando:
  - a. Un objeto “Actor” que representa al usuario que inicia el flujo de eventos.
  - b. Un objeto de Interfaz que representa la pantalla con la cual interactúa el usuario.
  - c. Un objeto de Control que se ocupa de organizar y controlar la lógica requerida para alcanzar el objetivo del caso de uso.
  - d. Un objeto de Entidad para cada una de las entidades de dominio requeridas para realizar el comportamiento definido en el caso de uso.

Cada línea de vida se graficará con una cabecera representativa del tipo y una línea discontinua vertical (indicadora del tiempo durante el cual la instancia existe e interviene en las interacciones).

2. Luego se colocarían los mensajes que esos objetos envían y reciben a lo largo del eje Y, ordenándolos cronológicamente de arriba hacia abajo.  
Cada mensaje se graficará como una flecha cuyos extremos variarán de forma según el tipo de mensaje (ver tabla anterior). La interpretación del transcurso del tiempo debe realizarse desde arriba hacia abajo.

### Parámetros

Estos proveen valores específicos para el uso de la interacción. Permiten parametrizar la interacción en cada ocurrencia. Para escribirlos se utiliza la notación de parámetros establecida por UML. En la figura se muestra una interacción que retorna un valor que luego es asignado a una variable.



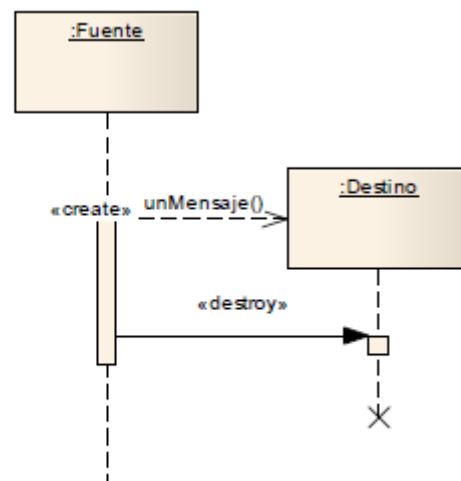
### Creación y destrucción de líneas de vida

Una línea de vida es creada y destruida durante el transcurso del tiempo, mediante el envío de mensajes de creación o de destrucción, respectivamente.

La creación de una línea de vida se grafica dibujando la cabecera a la altura del mensaje enviado por el objeto creador.

La destrucción de una línea de vida se indica por medio de un símbolo terminal, representado por una cruz.

En el siguiente diagrama se muestra la creación y destrucción de un objeto.



*Creación y destrucción de objetos en un Diagrama de Secuencia*

## Pasos para la construcción de un diagrama de secuencia

Los construiremos de la siguiente manera:

- seleccionamos un caso de uso
- examinaremos su descripción
- determinamos qué objetos son los que intervienen en el escenario.
- establecemos los mensajes que se envían entre sí a lo largo de la secuencia.

## Lineamientos generales

En el apartado siguiente, se presentan algunos lineamientos generales a considerar en el momento de modelar con diagramas de secuencia:

- **Ubicar los mensajes ordenados de izquierda a derecha:** Se debe comenzar el flujo de mensajes desde la esquina superior izquierda; un mensaje que aparezca más abajo es enviado después de uno que aparezca arriba de éste. Consistentemente con la forma en que leemos hay que tratar de organizar los clasificadores (actores, clases, objetos y casos de uso) a lo largo del borde superior del diagrama de manera tal que representa el flujo de mensajes de izquierda a derecha.
- **Nombrar los objetos sólo cuando se referencien en mensajes:** Los objetos en los diagramas de secuencia tienen etiquetas en el formato estándar de UML “nombre: NombreDeLaClase”, donde “nombre” es opcional (si no se les especificó un nombre se los denomina objetos anónimos). Un objeto necesita ser nombrado cuando es referenciado como un valor de retorno a un mensaje, en cambio una clase que no necesita ser referenciada en ningún otro lado del diagrama puede ser anónima.
- **Nombrar a los objetos cuando existen varios del mismo tipo:** Cuando el diagrama incluya varios objetos del mismo tipo, se debería especificar el nombre a todos los objetos de ese tipo a fin de crear un diagrama sin ambigüedades. Por ejemplo, en un sistema de transferencia bancaria se debería diferenciar al objeto cuenta Origen del objeto cuenta Destino, de la siguiente manera:  
origen: Cuenta      destino: Cuenta
- **Concentrarse en las interacciones críticas:** Se aconseja concentrarse en los aspectos críticos del sistema -no en los detalles superfluos- cuando se crea un modelo. Entonces logrará un modelo bien hecho, incrementando tanto la productividad como la legibilidad de los



diagramas. Por ejemplo, si se trabaja con entidades de negocio lógicas, no es necesario que se incluyan los detalles de interacción para almacenarlas en la base de datos; será suficiente con los métodos guardar() o eliminar().

- **Modelar Escenarios de Uso:** un escenario de uso es una descripción de una forma potencial en la que el sistema es utilizado. La lógica de uso de un escenario puede ser una parte de un caso de uso, tal vez un curso alternativo. También puede ser el modelado de un curso completo de acción o una porción de un curso básico de acciones más uno o más escenarios alternativos. La lógica de uso de un escenario también puede pasar por la lógica contenida en varios casos de uso. Por ejemplo, un estudiante se inscribe en una universidad y luego inmediatamente se inscribe en tres seminarios.
- **Modelar la lógica de métodos:** explorar la lógica de una operación compleja, de una función o un procedimiento. Una forma de pensar los diagramas de secuencia, particularmente los diagramas altamente detallados son como código visual de un objeto.
- Validar y profundizar la lógica y completitud de un escenario de uso, que puede ser parte de un caso de uso, por ejemplo, un curso normal o básico de acción, o una pasada por la lógica contenida en varios casos de uso.
- Mostrar la interacción temporal (paso de mensajes) entre las distintas instancias.