

# Pre y pos condiciones y estados de un proyecto

Sitio: Agencia de Habilidades para el Futuro

Curso: Modelado y Diseño de Software 1º D

Libro: Pre y pos condiciones y estados de un proyecto

Imprimido por: Eduardo Moreno

Día: lunes, 19 de mayo de 2025, 01:28

# Tabla de contenidos

- 1. Introducción
- 2. Pre y pos condiciones
- 3. Estado
- 4. Relación entre diagramas UML
  - 4.1. Análisis de la plantilla



## Introducción

En las semanas anteriores fuiste viendo los distintos diagramas que te permiten modelar un desarrollo software.

A continuación, retomamos las pre y pos condiciones de los casos de uso.



## ¿Qué son las precondiciones y poscondiciones?

Para trabajar en esta semana es necesario recuperar estos conceptos:

La **precondición** está formada por el conjunto de condiciones que se tienen que cumplir para que se pueda iniciar un caso de uso. En muchos casos supone la ejecución de casos de uso previos.



Y, ¿las pos condiciones?

La **pos condición** refleja el estado en que se queda el sistema una vez ejecutado el caso de uso.

En **ambos casos** lo que se representa es un estado y no la ejecución de una serie de acciones (previas o posteriores al caso de uso).

¿Siempre debo colocar en la descripción del caso de uso las pre y pos condiciones?

Ya que no se debe tratar de una decisión de carácter formal sino de una decisión basada en criterios prácticos, así que lo mejor es que para cada caso de uso te hagas la siguiente pregunta ¿aporta una mayor claridad para el usuario y para el equipo de desarrollo que para un caso de uso concreto ponga precondición y/o pos condición o estas se sobreentienden?

En general, es conveniente colocar ambas para aportar mayor claridad.

Pero, ¿qué es un estado?



## Estado

Un estado muestra el comportamiento del sistema, dependiente del tiempo.

Todo sistema pasa por diferentes estados. Hasta hace un tiempo, los modelos del comportamiento dependiente del tiempo del sistema importaban solo para una categoría especial de **sistemas, conocidos como sistemas de tiempo real**; por ejemplo, sistemas de conmutación telefónica.

Para sistemas enfocados a los negocios no se consideraba demasiado importante mostrar el estado; sin embargo, en sistemas grandes y complejos enfocados a negocios que sí tienen aspectos de comportamiento de tiempo real, la presentación del estado resultaba determinante.

### Por ejemplo:

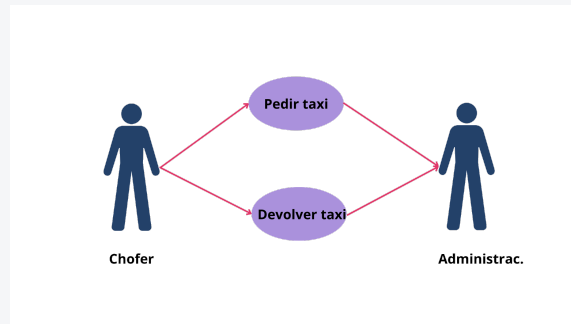
Si el sistema maneja entradas de miles de terminales y entradas de alta velocidad de otros sistemas, pueden entonces surgir aspectos de comportamiento dependientes del tiempo, del tipo que surgen en un sistema típico de tiempo real. Por esto, aunque no se apliquen en todos los sistemas, es conveniente estar familiarizado con herramientas de modelado para el comportamiento dependiente del tiempo.

Este tema lo retomaremos la semana próxima.



## Relación entre diagramas UML

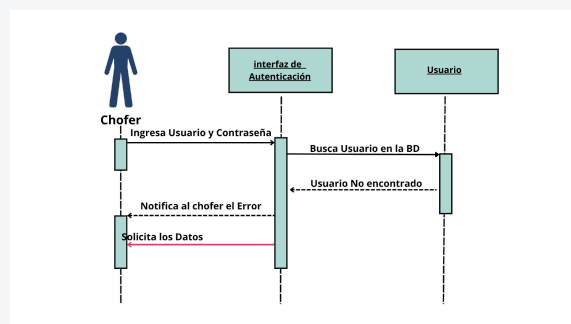
Retomemos el diagrama de caso de uso de la empresa de taxis que viste en semanas anteriores.



En este diagrama de casos de uso, no se ha diagramado ningún *include* ni *extend*.

Analicemos si es necesario colocar alguno de ellos.

Observando el diagrama de secuencia siguiente se ve que, para solicitar un taxi, se debe autenticar al solicitante (algo que no fue tenido en cuenta en el caso de uso).



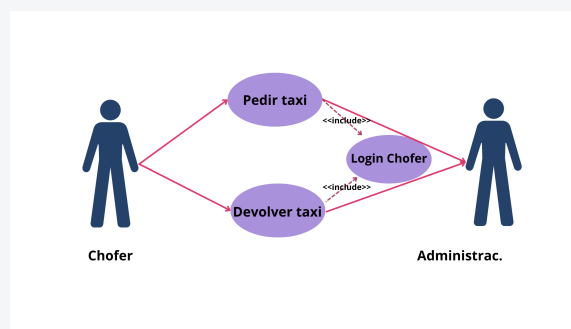
Recordando el concepto de *include* y *extend* podemos concluir que es un *include*, ya que estamos diciendo que el primero (el caso de uso base) incluye al segundo (el caso de uso incluido). Es decir, el segundo es parte esencial del primero, que es lo que sucede con el login.

El caso de uso de pedir taxi, incluye al caso de uso de login, por lo que debemos modificar el diagrama para que nos quede con el caso incluido.

¿Sucede lo mismo para la devolución del taxi?

Podemos concluir que **sí es necesario** que el chofer se autentique.

Qué hacemos en el caso que nos damos cuenta que no tuvimos en cuenta este include, **modificamos el diagrama**, el cual nos quedará de la siguiente manera:



Tomamos la plantilla donde se describe el caso de uso Tomar Taxi

Caso de uso (U)	PT001 à Nombre del caso de uso. Lo decide el equipo de trabajo		
Caso de uso Nombre:	Pedir Taxi (nombre de la gráfica)		
Creado por:	MCC(nombre grupo )	Última actualización por:	
Fecha Creación:	XX/XX/202x	Fecha última actualización:	
Actor:	Taxista (actor primario)		
Descripción:	El taxista ingresa sus datos y selecciona la opción Pedir Taxi. Al confirmar que el taxi está disponible, se modifica el estado del taxi y se genera un comprobante de entrega del móvil		
Precondiciones:	Existen autos disponibles. Existen datos de taxistas		
Postcondiciones:	Se genera un comprobante de entrega del taxi Se asocia el comprobante con el taxista Se asocia el comprobante con el móvil Se modifica el estado del taxi como "entregado".		
Privacidad:	Alta		
Frecuencia de uso:	A demanda (es decir cada vez que se pide un taxi)		
Flujo Normal:	1 El CU inicia cuando el taxista (T) selecciona la opción de Pedir Taxi 2 El sistema solicita el ingreso del DNI del taxista. 3 T ingresa el número de DNI 4 El sistema verifica que el T exista y el mismo existe. 5 El sistema busca el número de taxi que le corresponde al taxista y muestra la patente, y si está disponible 6 El sistema pregunta si lo quiere retirar 7 T selecciona el ok de retiro del taxi 8 El sistema verifica el horario de retiro del taxi. 9 El sistema genera el comprobante de retiro con la hora y la fecha, indicando el taxista que se lo lleva 10 El sistema asocia el comprobante de retiro al taxista. 11 El sistema cambia el estado del taxi por el estado "retrado".		
Flujo Alternativo:	El T no Existe 1. El sistema verifica que el T con su DNI existe y el mismo no existe. 2. El sistema notifica que el DNI es incorrecto. 3. El sistema solicita ingreso de nuevo número de DNI  El móvil no existe 1. El sistema busca el móvil para ese taxista y no encuentra ningún móvil asociado al taxista 2. El sistema notifica que para el T ingresado no existen taxi o móvil asociado  El móvil o taxi no está disponible 1. El sistema busca el móvil para ese taxista y encuentra el móvil asociado al taxista 2. El sistema verifica el estado del taxi y figura "entregado" 3. El sistema notifica que el taxi no está disponible		
Excepciones:	El T cancela la operatoria. 1. El T cancela la operatoria de pedido de taxi 2. Fin del Caso de Uso.		
Incluido:	NA - CU Login		
Excluido:	NA		
Requisitos/Requisitos Especiales:	NA		
Notas:	NA		







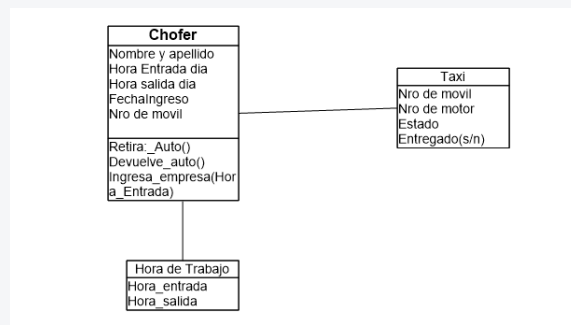
## Análisis de la plantilla

Analizando la plantilla, vemos que dentro del ítem **include no se ha colocado ningún caso de uso**, por lo que es necesario modificar la planilla y agregarlo para, posteriormente realizar la descripción del caso de uso **login**.

Colocamos en rojo (en este caso es arbitrario el color) **el nombre del caso de uso incluido**.

Observando el diagrama, éste nos aclara y modifica los otros diagramas ya realizados. **El diagrama de clases original**, ¿refleja parte de estos casos de uso?

Volvemos al diagrama de clases original (de las primeras semanas)



Podemos ver que el método login no está dentro de ninguna de las clases, y que no hemos colocado ninguna herencia (que no es obligatoria pero podría ir).

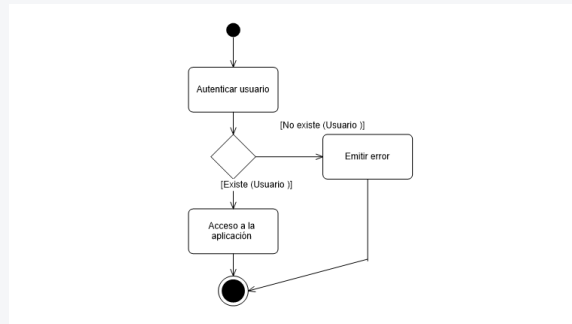
Al menos debemos agregar el **método login**, con parámetros **usuario y contraseña** que tampoco figuran como atributos. En este caso, **debemos modificar el diagrama agregando los atributos y métodos faltantes**.

Así, vamos analizando cada uno de los procesos y determinando si debemos o no modificarlos.

Ahora, analizamos el diagrama de actividades del login, para lo cual numeramos cada paso del proceso partiendo que el chofer ingresó sus datos de usuario y contraseña:

1. Autenticar usuario (consideramos que si el usuario existe, la contraseña es la correcta).
2. Si existe, se accede a la aplicación.

En caso que no existe se emite error:



No se está considerando que si no es un usuario existente, se le da de alta como nuevo usuario, en ese caso, el diagrama se modifica en la salida de la decisión de **no existe**.

Para el caso, producto del análisis, el de actividades no es necesario modificarlo.