

UML y UP: Análisis y Diseño Orientado a Objetos

Módulo 2

Diagrama de Actividades

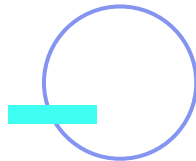
Diagrama de Actividades

Definición

Se utiliza principalmente para **modelar flujo de trabajo o workflow**, y así, visualizar las acciones de manera ordenada. Permite modelar procesos y comprender la ejecución general de un sistema. Se focaliza en la secuencia de acciones y condiciones necesarias para su ejecución.

Objetivo

Describir las acciones que ocurren en un proceso.

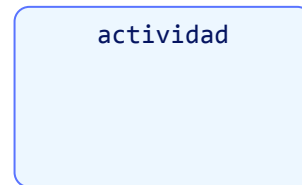


Elementos

Actividad (Activity)

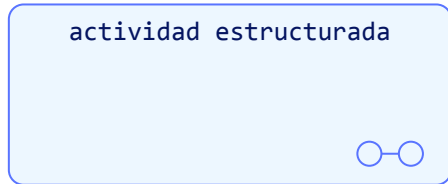
Una actividad refleja el control de flujo y de datos de un proceso, y puede incluir la participación de sub-actividades y/o acciones.

Generalmente, representa a una unidad funcional dentro de una actividad mayor o de un proceso.



Actividad Estructurada (*Structured Activity*)

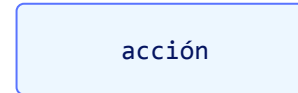
La actividad estructurada es una actividad que muestra de manera explícita que está compuesta por un conjunto de actividades o acciones.



Acción (*Action*)

La acción es la unidad funcional básica dentro de un diagrama de actividades, y representa una transformación que ocurre dentro del sistema.

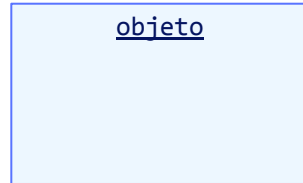
La diferencia principal con las actividades radica en que la acción no puede descomponerse, mientras que una actividad puede descomponerse en sub-actividades y/o acciones.



Objeto (*Object*)

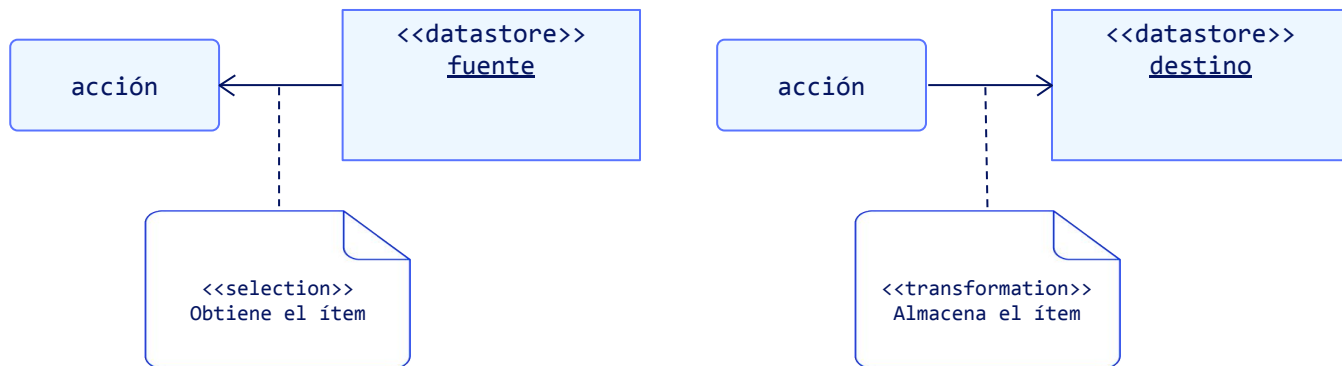
El objeto tiene el mismo significado que en los diagramas anteriores.

En este caso, adicionalmente, representa que un objeto es construido, modificado o consultado luego de una actividad.



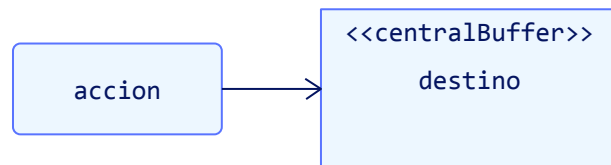
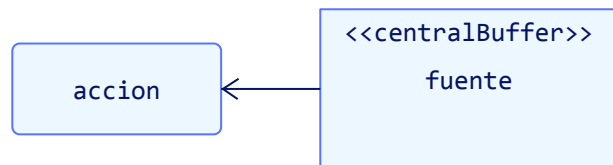
DataStore Object

El DataStore es un objeto que se utiliza para modelar la persistencia de información permanente. Se puede utilizar tanto para almacenar información como para consultarla.



CentralBuffer Node

El buffer tiene el mismo significado que el DataStore, pero se utiliza para almacenar información en forma temporal (información no persistente o transitoria).



Pseudo-Estado Inicial (*InitialState*)

El pseudo-estado inicial representa el comienzo de los flujos entre las acciones.



Pseudo-Estado Final (*FinalState*)

El pseudo-estado final representa el final de los flujos entre las acciones.



Señal de Envío (*SendSignal*)

La señal de envío se utiliza para representar que se está enviando un objeto en forma de señal hacia algún destino no necesariamente representado en el diagrama.

En el siguiente ejemplo, la señal de envío se utiliza para notificar al proveedor acerca de la creación de un pedido, donde es enviado el objeto “pedido” al proveedor.



Señal de Recepción (*ReceiveSignal*)

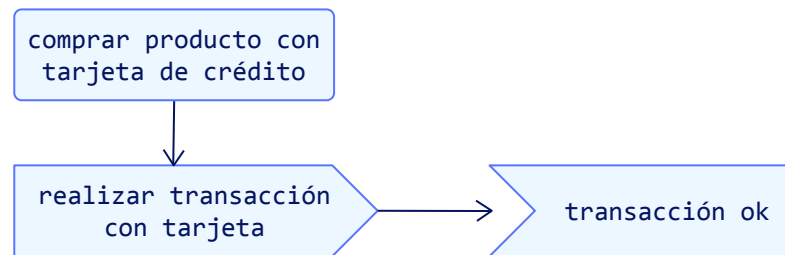
La señal de recepción se utiliza para representar la recepción y/o aceptación de un pedido.

Existen dos tipos de señales de recepción, veamos la siguiente slide.



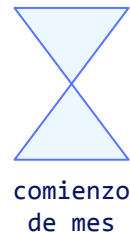
AcceptEventAction

Esta señal se utiliza para determinar la recepción de un evento por parte de terceros. Si se utiliza en conjunto con la señal de envío, establece que el flujo continuará normalmente, una vez que se produzca la señal de respuesta.



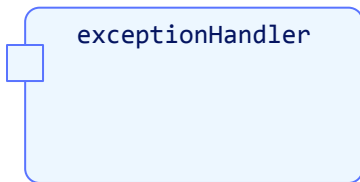
Accept Time EventAction

Esta señal se utiliza para representar el pasaje del tiempo. Es una acción que espera la ocurrencia de un evento del tipo tiempo, por ejemplo: comienzo de un mes, finalización del mes, comienzo del día, etc.



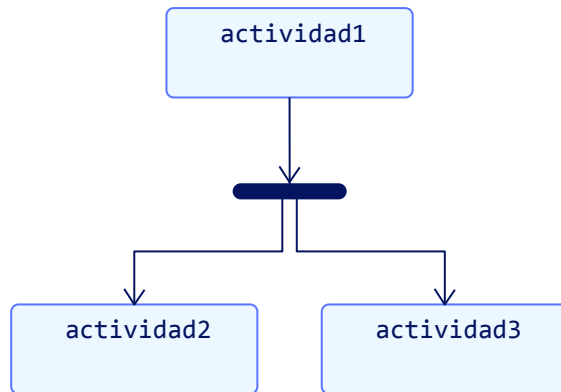
Manejador de Excepciones (*ExceptionHandler*)

El manejador de excepciones está compuesto por una o más acciones y se encarga de tomar medidas correctivas en caso de que se haya lanzado alguna excepción (o se haya producido algún error).



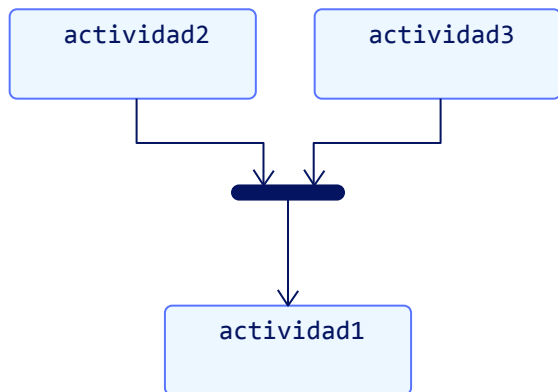
Fork

El fork es un pseudo estado que se utiliza para separar un flujo de entrada en dos flujos concurrentes que tienen como destino diferentes actividades y/o acciones.



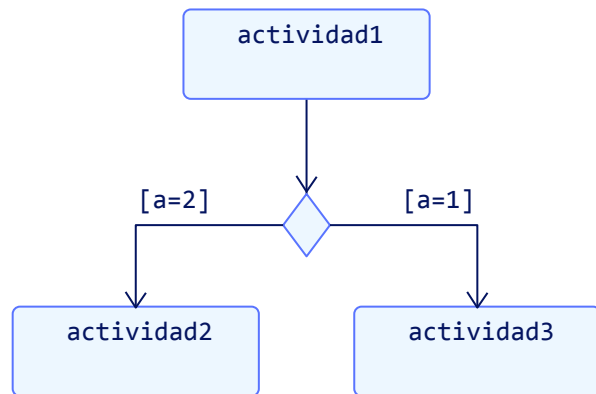
Join

Es el proceso inverso al fork. Es un pseudo estado que se utiliza para unir dos flujos de entrada concurrentes en un único flujo de salida.



Decisión (Choice)

La decisión indica un condicional en el progreso: si la condición es verdadera toma un camino, y si es falsa toma el camino alternativo.

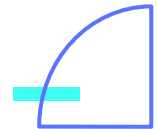
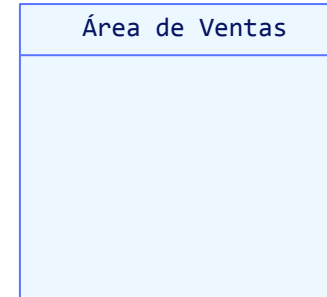


Partición (*Partition*)

Las **particiones** se utilizan para establecer responsabilidades en las acciones o actividades. Determinan qué o quién realiza cada actividad.

Son de uso opcional y permiten estructurar la vista o las partes de la actividad realizando una separación lógica.

También son denominadas **swim lanes**.



Relaciones

Flujo de control (*Control Flow*)

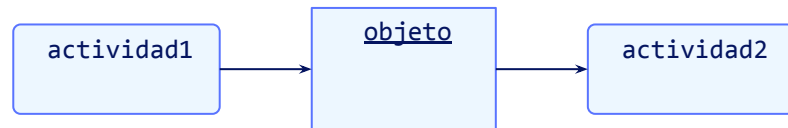
El flujo de control es un conector que une dos actividades o acciones, una inicial y una final. El conector determina la dirección del flujo. Una vez que la acción inicial es completada, el control pasa a la acción siguiente.

Es posible definir una guarda que representa una condición que debe cumplir la acción inicial para pasar a la próxima acción.



Flujo de objeto (*ObjectFlow*)

El flujo de objeto conecta una actividad con un objeto y representa un flujo de información de la actividad al objeto.



Flujo de objeto con Pines (*PinnedObjectFlow*)

El flujo de objeto con distinción es semánticamente igual al flujo de objeto tradicional. La diferencia radica en la forma de graficarlo ya que, de esta manera, resulta más compacto.



Flujo de Interrupción (*InterruptFlow*)

El flujo de interrupción representa una interrupción debido a una ocurrencia inesperada en alguna acción o actividad.

Aplicación

Desarrollo de aplicaciones procedurales

Uno de los posibles usos que tiene el Diagrama de Actividades es la posibilidad de realizar el flujo de aplicaciones de tipo procedural, al estilo diagramas de Jackson.

Se pueden representar las sentencias (acciones) junto con las estructuras de control de flujo utilizando las decisiones, *forks*, *joins* y actividades estructuradas para mostrar llamadas a “cajas negras” o procedimientos.



Modelado de procesos de negocio - Workflow

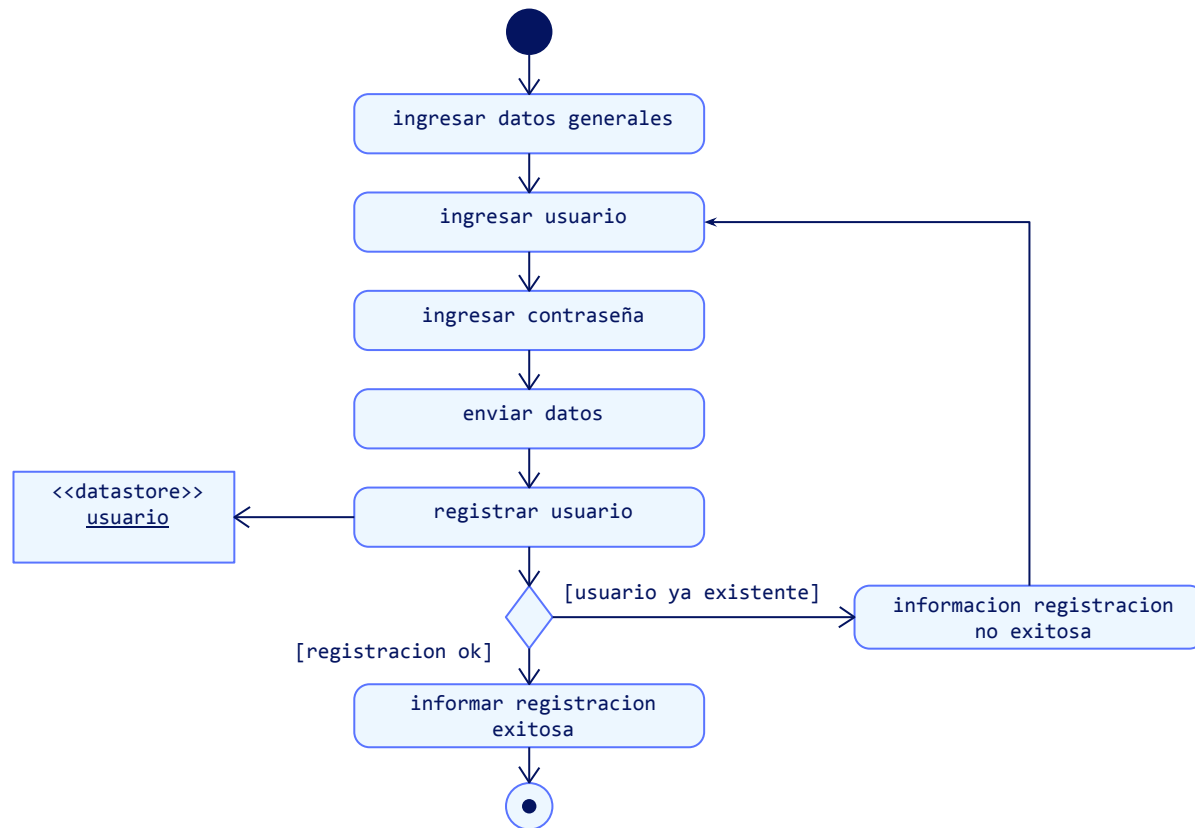
El uso más común del Diagrama de Actividades es la representación de flujos de trabajo o procesos de negocio.

Es posible modelar como arranca un proceso, a partir de quien o en qué área se produce, y visualizar la transformación de información que va ocurriendo durante todo el proceso.

En este contexto los eventos generalmente se originan dentro del sistema - al finalizar las diferentes acciones-, o también fuera del sistema - a través de señales de recepción (por ejemplo: el proveedor entregó la mercadería) y a través de eventos correspondientes al tiempo (por ejemplo: pasó un mes)-.

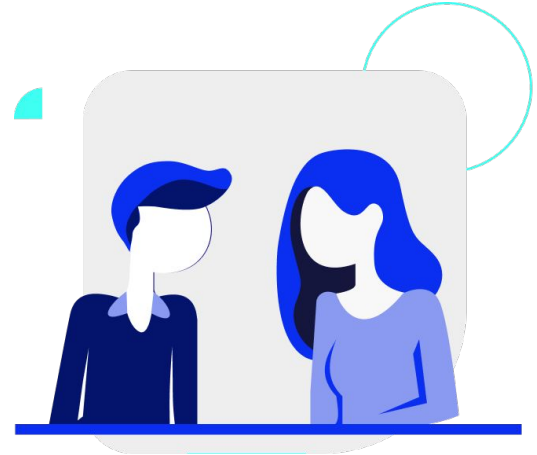


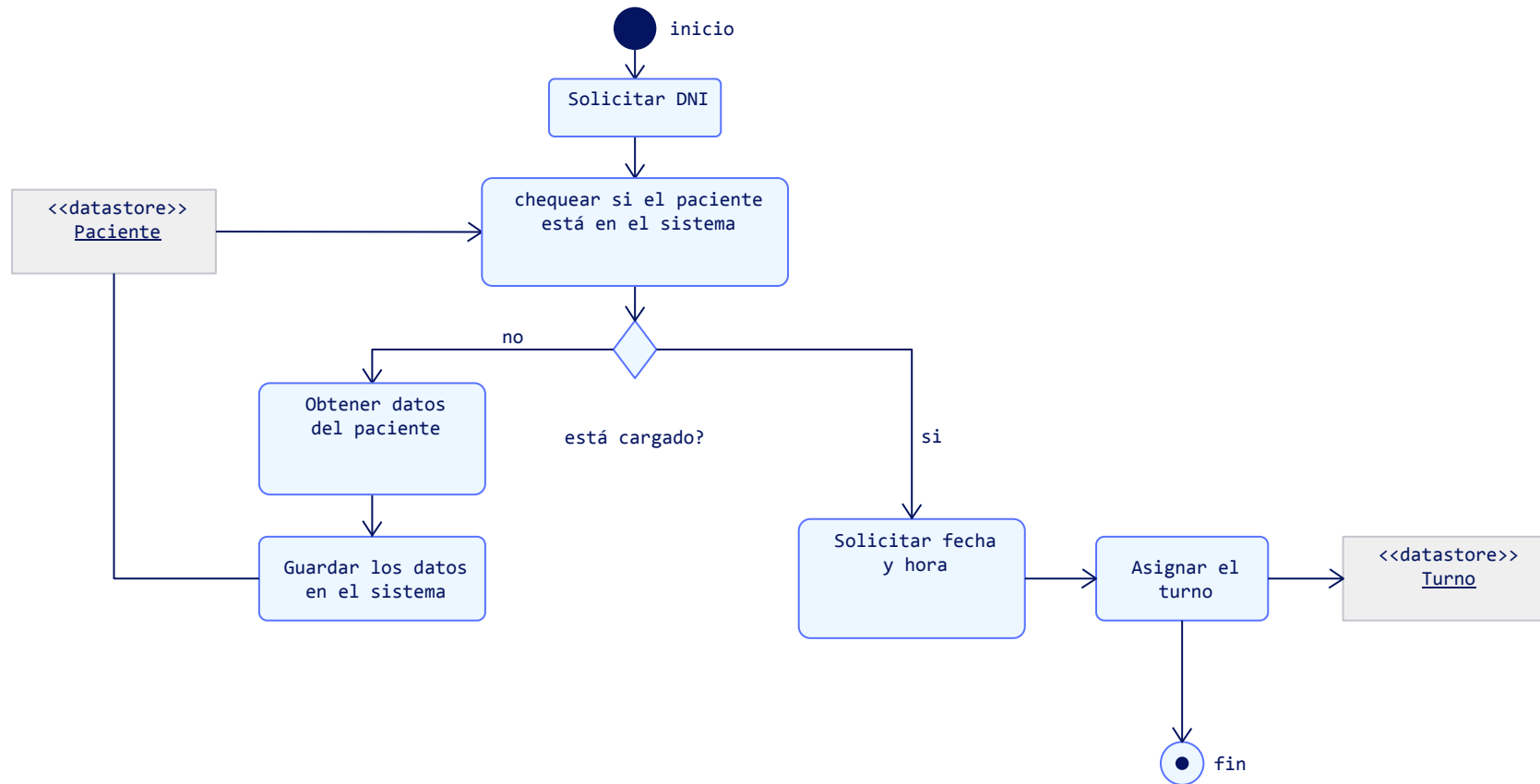
Ejemplo



Tutorial

Realizar el tutorial de la próxima slide junto con el instructor.





**¡Sigamos
trabajando!**