

# Análisis y diseño de aplicaciones I

UT3 Diseño y UML



# Agenda

- Diagrama de secuencia
- Diagrama de actividades
- Diagrama de deploy

# Diagrama de Secuencia



- Los diagramas de secuencia describen cómo **colaboran grupos de objetos** en algún comportamiento. El UML define varias formas de diagramas de interacción, de las cuales la **más común es el diagrama de secuencia**.
- Normalmente, un diagrama de secuencia captura el **comportamiento de un solo escenario**. El diagrama muestra varios objetos de ejemplo y los mensajes que se pasan entre estos objetos dentro del caso de uso.
- Los diagramas de secuencia **muestran la interacción** mostrando a **cada participante con una línea de vida** que se extiende **verticalmente** por la página y el orden de los mensajes leyendo hacia abajo por la página.

# Diagrama de Secuencia



- Tiene dos dimensiones
  - **Horizontal:** Representa los objetos (**componentes, subsistemas, sistemas**) que participan en la secuencia.
  - **Vertical:** Representa **la línea de tiempo sobre la que los elementos actúan**. Va de arriba (menor tiempo) hacia abajo (mayor tiempo)

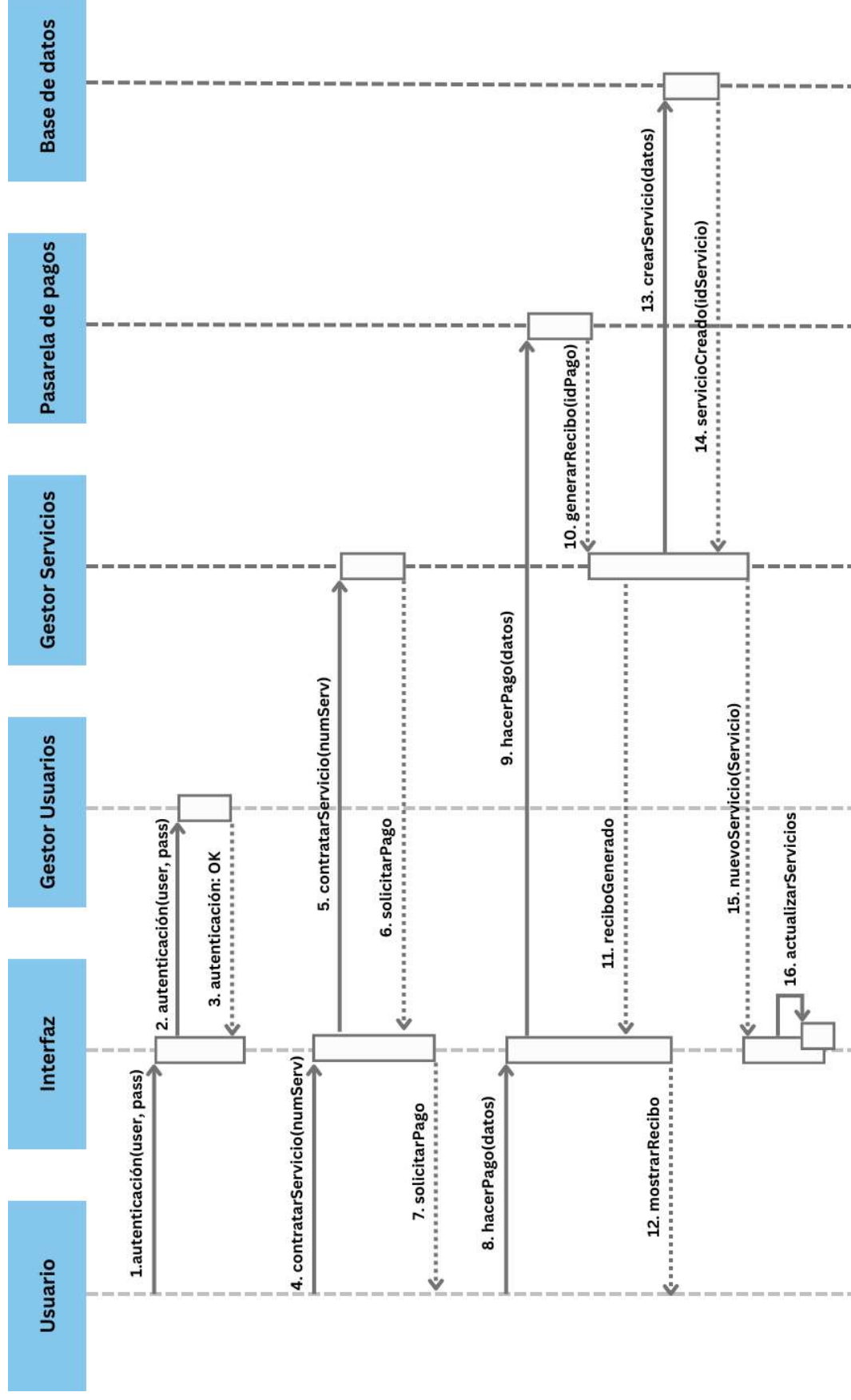
No es común «reglar» **esta dimensión mediante una escala para poner tiempos específicos**, a excepción de sistemas de tiempo real donde la velocidad para llevar a cabo la funcionalidad sí es relevante.

# Diagrama de Secuencia



- Las **flechas** entre los “objetos” se llaman **mensajes**. Pueden incluir el nombre de la función y parámetros que necesita. (no tienen por qué estar todos, es un diagrama de alto nivel).
- Estas **pueden tener o no una respuesta**. En caso que sí, esta se representa con **una flecha punteada**.

# Diagrama de Secuencia

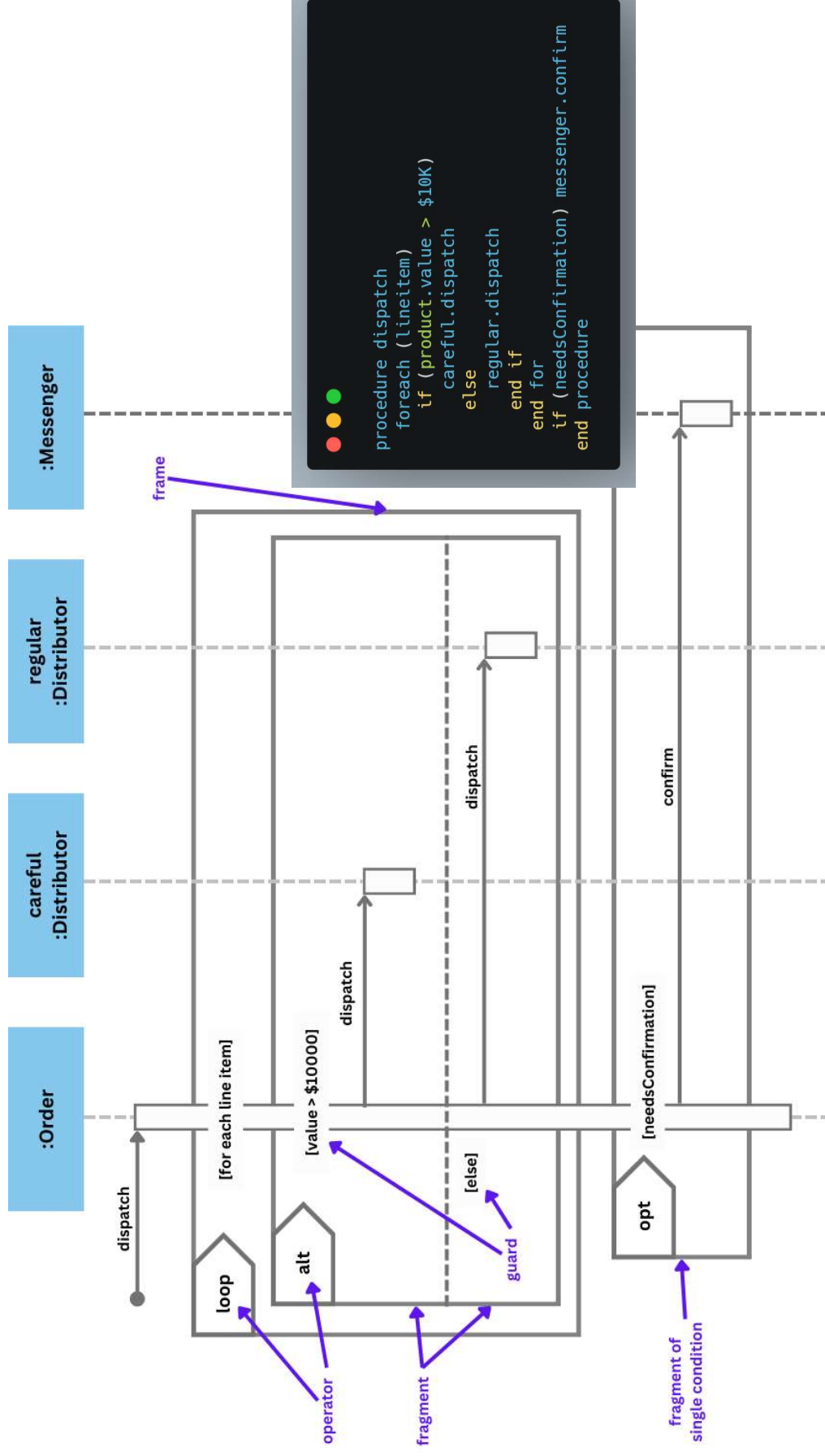


# Diagrama de Secuencia



- También se pueden representar;
  - Bucles (LOOP)
  - Flujos condicionales (ALT)
  - Flujos opcionales (OPT)
- Una recomendación sobre esto, es que si el diagrama de vuelve extremadamente complejo, conviene descomponerlo en varios diagramas (Utilizando notación UML).
- O utilizar diagramas de actividad, o directamente código.
- Es mejor tratar a los diagramas de secuencia como una manera de modelar comunicación entre componentes, en vez de modelado de lógica.

# Diagrama de Secuencia





# Tarea de Aplicación 6

## Diagrama secuencia



# Diagrama de Actividad



- Se utiliza para modelar **el flujo de trabajo** o el comportamiento de un sistema o proceso de negocio.
- Es útil para representar **procesos, procedimientos y algoritmos** complejos en una forma **visual** fácil de entender.
- La finalidad de este diagrama es modelar el **workflow** de una actividad a otra, pero sin tener en cuenta el paso de mensajes entre ellas. Para ello, estas actividades pueden dividirse en sistemas por lo que una finalidad (la más común) de este diagrama puede ser capturar estos sistemas y describir como se relacionan entre sí.
- También es utilizado para modelar las actividades, que podemos asemejar a requisitos funcionales de negocio, por lo que este diagrama tendrá una influencia mayor a la hora de comprender el negocio o sus funcionalidades que en la propia implementación. Hay que tener en cuenta que este **diagrama** ofrece una visión a **alto nivel**.

# Diagrama de Actividad

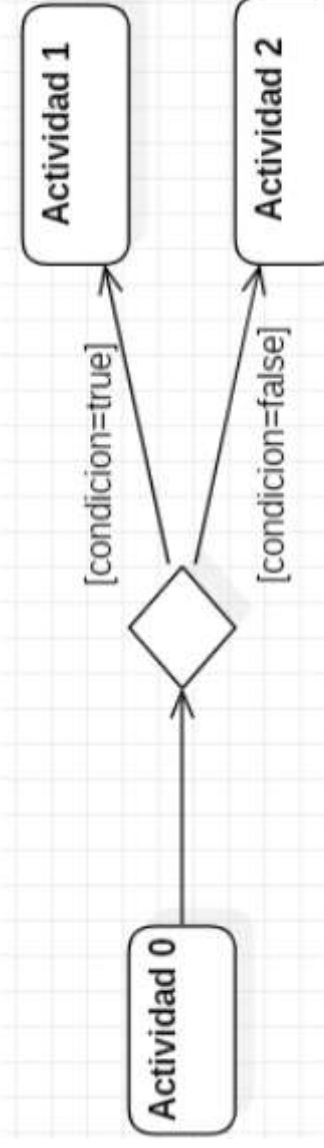


- Los flujos entre actividades se representan mediante una **flecha con la punta abierta** que simboliza el orden de ejecución de las actividades, a veces se incorpora un nombre en esta flecha que ayuda a que se entienda mejor:



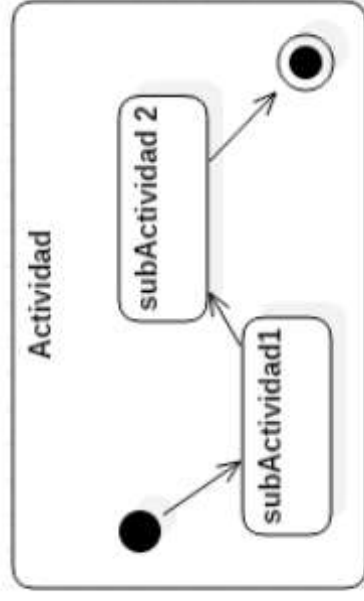
Notación de un flujo de actividad

- Estos flujos de actividades **pueden usar condiciones** para su actuación, estas condiciones se representan mediante un **rombo**, con la condición escrita entre corchetes:



# Diagrama de Actividad

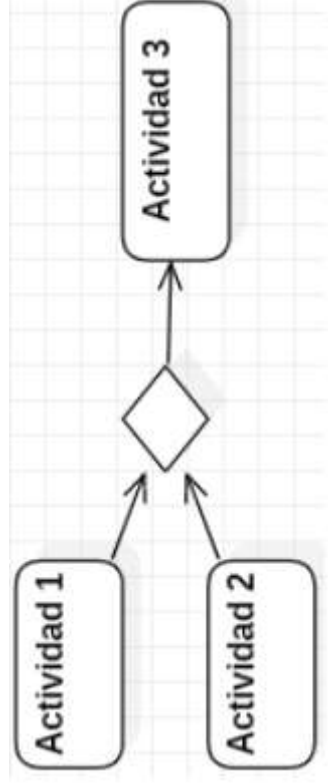
- Las flujos comienzan con un círculo cerrado y terminan con dos círculos.



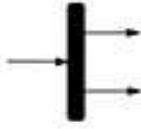
Notación de una actividad compuesta

# Diagrama de Actividad

- El caso contrario es el llamada **nodo de fusión**, que recibe 2 o más flujo y emite 1:



# Diagrama de Actividad

	Bifurcación	Representada por una barra con una entrada y varias salidas, indica el inicio de flujos de actividades en paralelo.
	Unión	Representada por una barra con varias entradas y una salida, indica el fin de flujos de actividades en paralelo. (El flujo de salida de una unión se dispara cuando se han finalizado todos los flujos de entrada en la unión)

# Tarea de Aplicación 7

## Diagrama de actividad



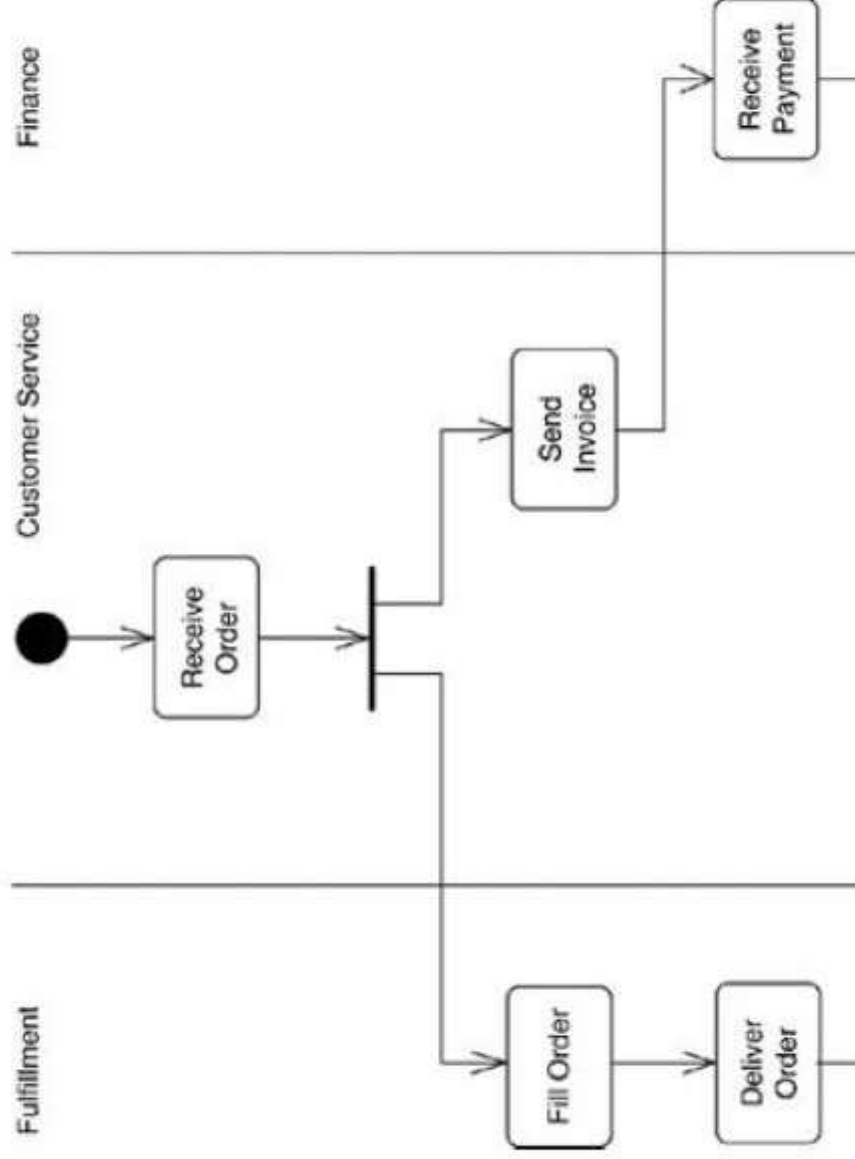
# Diagrama de Actividad



- Los diagramas de actividad te **muestran qué sucede, pero no te dicen quién hace qué.**
- Esto no necesariamente es un problema; a menudo tiene sentido centrarse en lo que se hace en lugar de en quién realiza cada parte del comportamiento.
- Si se desea mostrar quién hace qué, puedes dividir un diagrama de actividad en **particiones**.



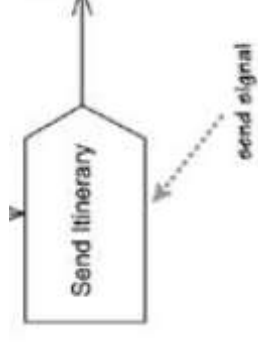
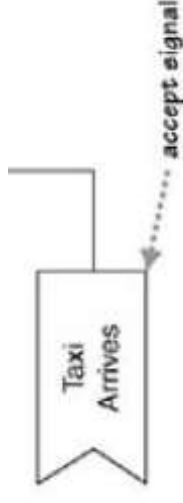
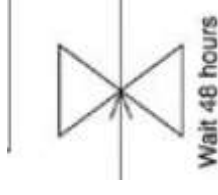
# Diagrama de Actividad



# Diagrama de Actividad

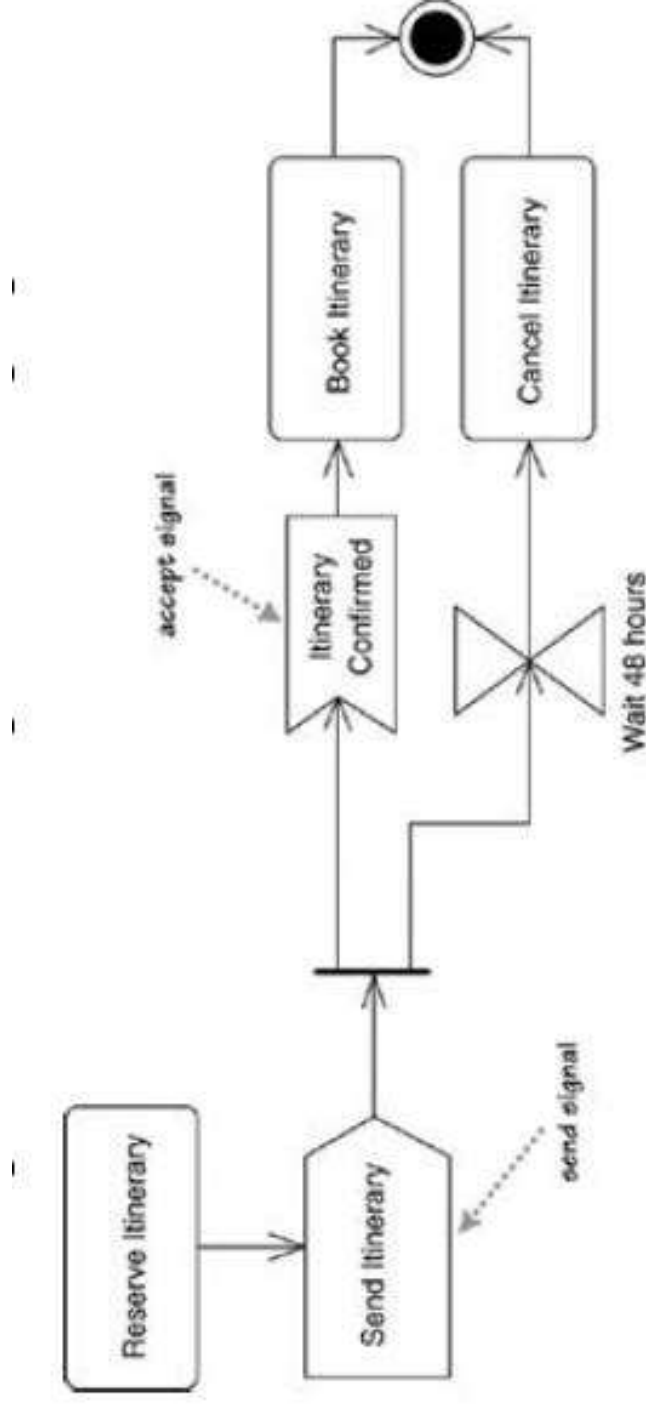


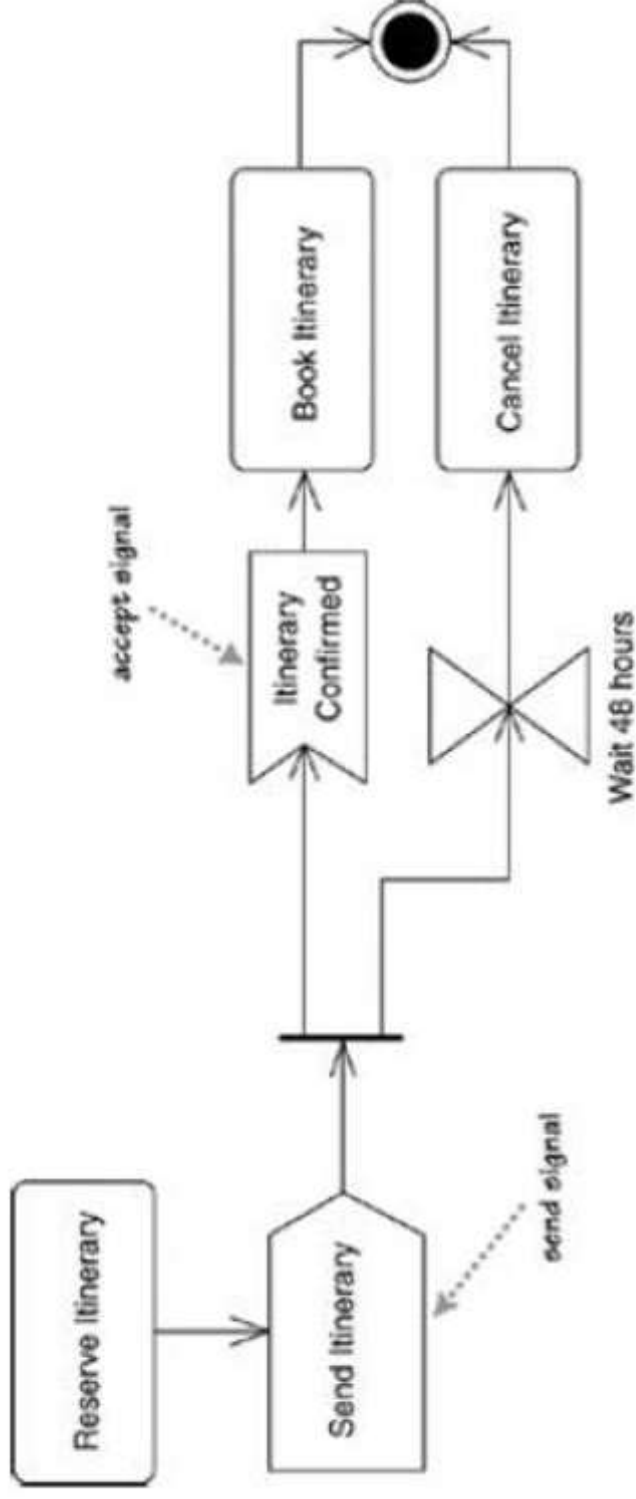
- Las acciones también pueden responder a señales:
- **Tiempo:** Ocurre debido al paso del tiempo. Estas señales pueden indicar el final de un mes en un período financiero o cada microsegundo en un controlador en tiempo real.
- **Recepción:** Una señal indica que la actividad recibe un evento de un proceso externo. Esto indica que la actividad está constantemente escuchando y el diagrama define cómo reacciona la actividad.
- **Envío:** Esto es útil cuando tenemos que enviar un mensaje y luego esperar una respuesta antes poder continuar.

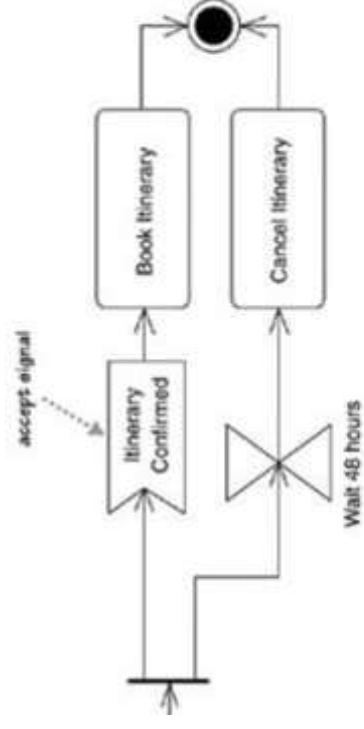
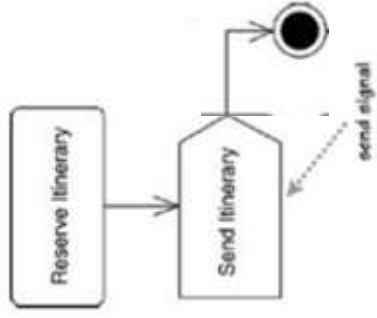


# Diagrama de Actividad

- Ejemplo con los tres tipos:







# Tarea de Aplicación 8

## Diagrama de actividad



# Diagrama de Deploy



- Los diagramas de despliegue muestran la disposición física de un sistema, revelando qué software ejecutan en qué hardware. Son bastante simples, con los elementos principales siendo nodos conectados por rutas de comunicación.
- Un **nodo** es algo que puede **alojar software** y puede ser de dos tipos:
  - un **dispositivo**, que es hardware como una computadora o una pieza más simple de hardware conectada a un sistema,
  - un **entorno de ejecución**, que es software que aloja o contiene otro software, como un sistema operativo o un proceso contenedor.
- Los nodos contienen **artefactos**, que son las manifestaciones físicas del software: **generalmente archivos**. Estos pueden ser ejecutables (como archivos WAR, binarios, DLLs, archivos JAR, ensamblajes o scripts, dockerfile), archivos de datos, archivos de configuración, documentos HTML, etc. Indicar un artefacto dentro de un nodo muestra que el artefacto se despliega en ese nodo en el sistema en ejecución.

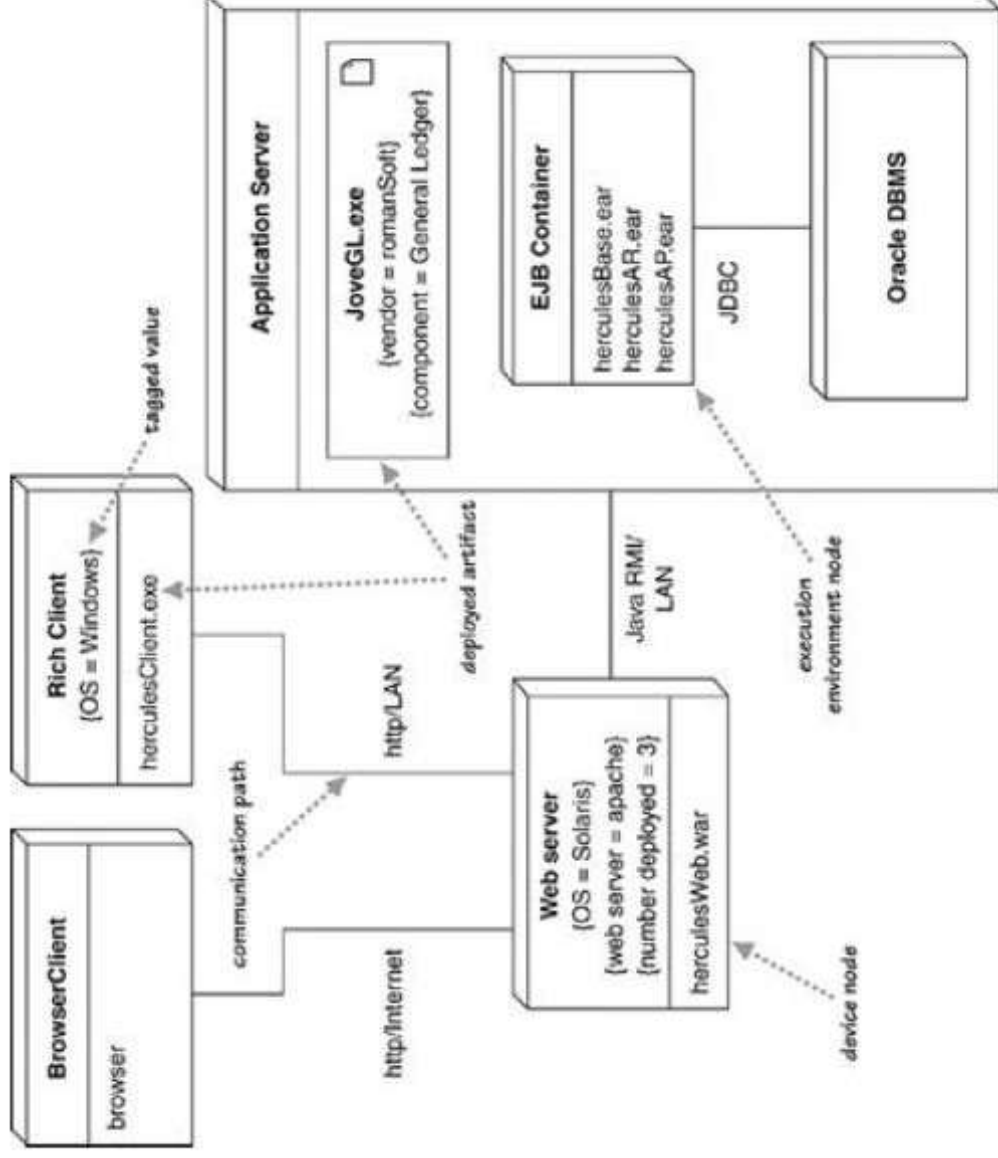
# Diagrama de Deploy



- Los artefactos se pueden mostrar como cajas de clase o listando su nombre dentro de un nodo. Si se muestran como cajas de clase, se puede agregar un icono de documento o la palabra clave «artefacto». Los nodos o artefactos pueden etiquetarse con valores etiquetados para indicar información relevante sobre el nodo, como el proveedor, el sistema operativo, la ubicación, entre otros.
- A menudo, varios nodos físicos llevan a cabo la misma tarea lógica. Se puede mostrar esto con varias cajas de nodo o indicar el número como un valor etiquetado.
- Las rutas de comunicación entre los nodos indican cómo se comunican las cosas. Estas rutas pueden llevar etiquetas con información sobre los protocolos de comunicación que se utilizan.



# Diagrama de Deploy



# Tarea de Aplicación 9

## Diagrama de deploy



# Bibliografía

- UML Distilled - Martin Fowler
- <https://diagramasuml.com/>