

# Modelando el Ciclo de Vida de Desarrollo del Software

## Módulo 7: Fase 2 - Elaboración



# Tópicos

## Fase 2: Elaboración

- Introducción al UML
- Diagrama de Actividades
- Diagrama de Casos de Uso

## Fase 2: Elaboración

- El propósito de la fase de elaboración es analizar el dominio del problema, establecer los cimientos de la arquitectura, desarrollar el plan del proyecto y eliminar los mayores riesgos.
- En esta fase se construye un prototipo de la arquitectura, que debe evolucionar en iteraciones sucesivas hasta convertirse en el sistema final. Este prototipo debe contener los Casos de Uso críticos identificados en la fase de inicio. También debe demostrarse que se han evitado los riesgos más graves.

# Fase de Elaboración

- **Propósito**

- Analizar el dominio del problema.
- Establecer una buena arquitectura.
- Lidiar con los elementos de riesgo más altos del proyecto.
- Desarrollar un plan comprensivo mostrando como el proyecto será completado.

# Resultados de la Fase 2: Elaboración

- **Un modelo de Casos de Uso completo al menos hasta el 80%**

Todos los casos y actores identificados, la mayoría de los casos desarrollados.

- **Requisitos adicionales**, que capturan los requisitos no funcionales y cualquier requisito no asociado con un Caso de Uso específico.
- **Descripción de la arquitectura software**
- **Un prototipo ejecutable de la arquitectura.**
- **Lista de riesgos y caso de negocio revisados.**
- **Plan de desarrollo para el proyecto.**
- **Un caso de desarrollo actualizado** que especifica el proceso a seguir.
- **Un manual de usuario preliminar** (opcional).

Si el proyecto no pasa estos criterios hay que plantearse abandonarlo o repensarlo profundamente.

# Introducción al UML

- UML = Unified Modeling Language.
- Lenguaje de propósito general para el modelado orientado a objetos.
- Impulsado por el Object Management Group (OMG, [www.omg.org](http://www.omg.org))
- El UML nos permite especificar, visualizar y documentar los componentes de un sistema.
- Lenguaje, no método de desarrollo.

# Ventajas

- El UML garantiza:
  - ✓ Organización en el proceso de diseño de tal manera que analistas, clientes, desarrolladores y personas involucradas en el desarrollo del sistema sean capaces de comprenderlo.
  - ✓ Independencia entre el proceso de desarrollo y los lenguajes de programación.
  - ✓ Una base formal para entender el lenguaje de modelado.



← **Ultima Versión 2.1.1**

# Diagramas de Comportamiento

- Los diagramas de comportamiento describen las características de comportamiento de un sistema o proceso de negocios.
- Incluye:
  - Diagrama de Actividades.
  - Diagrama de Casos de Uso.



# Diagrama de Actividades

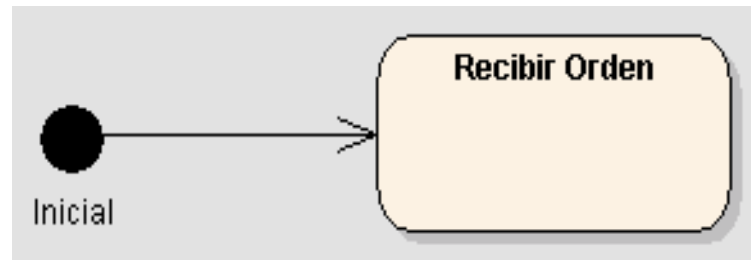
- Los diagramas de actividad se usan para modelar el comportamiento de un sistema, y la manera en que estos comportamientos se están relacionado con un flujo global del sistema.
- Se usan las rutas lógicas que sigue un proceso basado en varias condiciones, acceso de datos, interrupciones y otras distinciones de rutas lógicas para construir un proceso, sistema o procedimiento.

# Características

- Muestra el flujo de actividades de un sistema.
- Parecido al Diagramas de Flujos.
- Admite semántica de concurrencia y sincronización.
- Permite modelar decisiones.
- Se puede utilizar para modelar negocios.

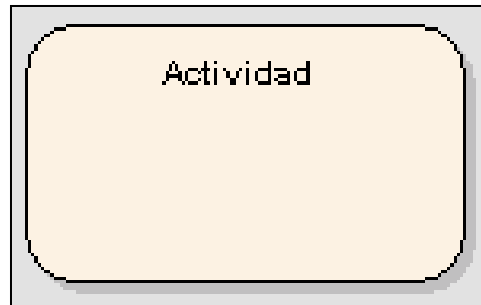
# Inicial

- Punto en donde se inician las actividades.



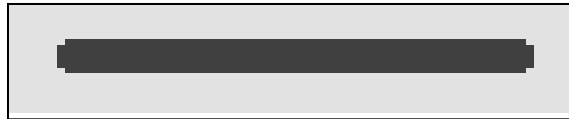
# Actividad

- Una actividad organiza y especifica la participación de comportamientos subordinados, como subactividades o acciones, para reflejar el control y el flujo de datos de un proceso.



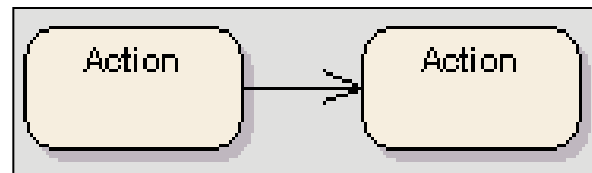
# Barra de Sincronización

- Indica el comienzo (fork) o sincronización (join) de actividades concurrentes.



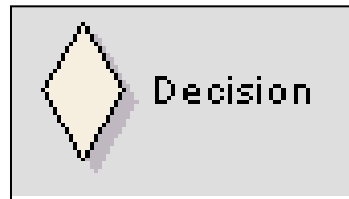
# Transición

- Denota traspaso del control desde una actividad a otra



# Decisión

- Flujos de Control alternativos



# Fin

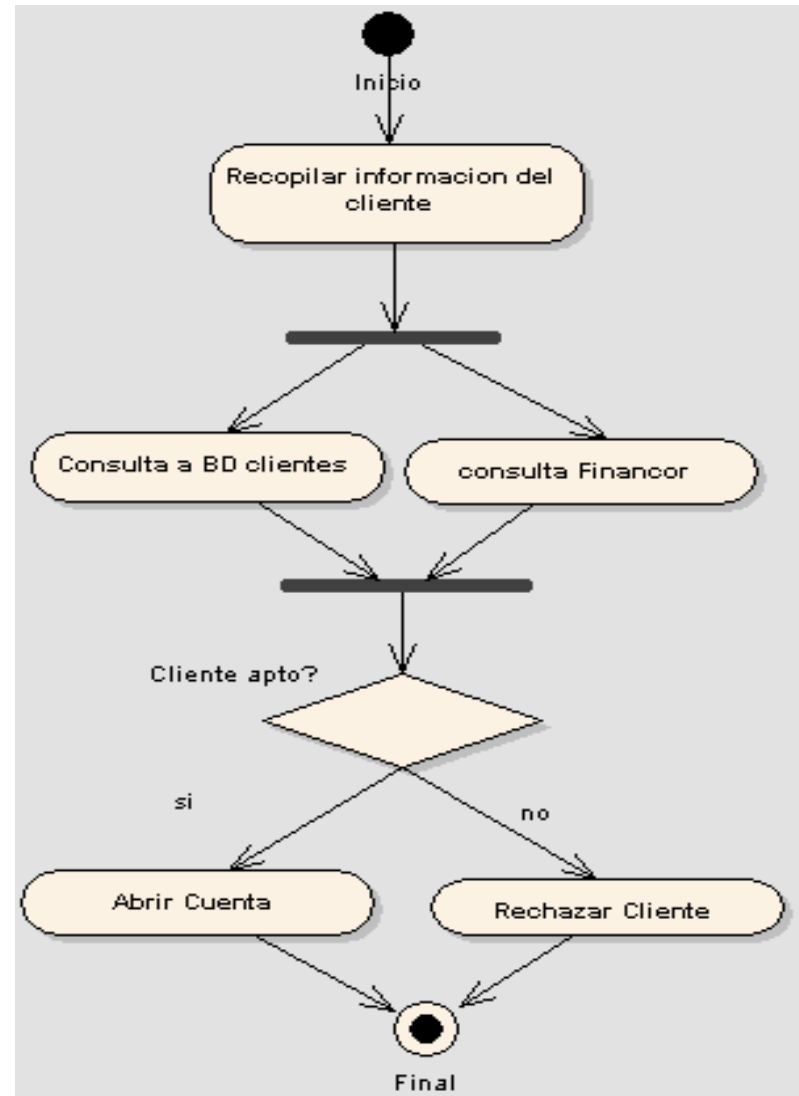
- Denota término de actividades





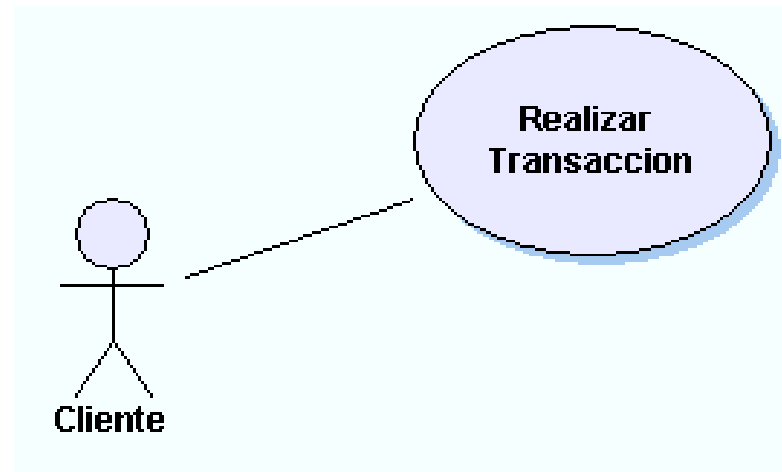
# Diagrama de Actividades

- El siguiente diagrama muestra la apertura de una cuenta bancaria.



# Diagrama de Casos de Uso

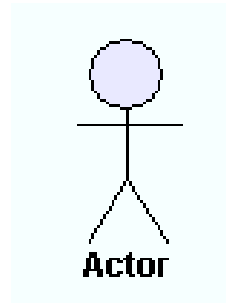
- Representa la funcionalidad que ofrece el sistema en lo que se refiere a su interacción externa.
- Muestra la relación entre los actores y los casos de uso del sistema.



# Diagrama de Casos de Uso

## Actores

- Un actor es una entidad externa al sistema que realiza algún tipo de interacción con el mismo.
- Es representado en Enterprise Architect por:



# Diagrama de Casos de Uso

## Actores

- Los actores pueden ser:
  - Principales: personas que usan el sistema.
  - Secundarios: personas que mantienen o administran el sistema.
  - Material externo: dispositivos materiales imprescindibles que forman parte del ámbito de la aplicación y deben ser utilizados.
  - Otros sistemas: sistemas con los que el sistema interactúa.
- La misma persona física puede interpretar varios papeles como actores distintos
- El nombre del actor describe el papel desempeñado.

# Diagrama de Casos de Uso

## Casos de Uso

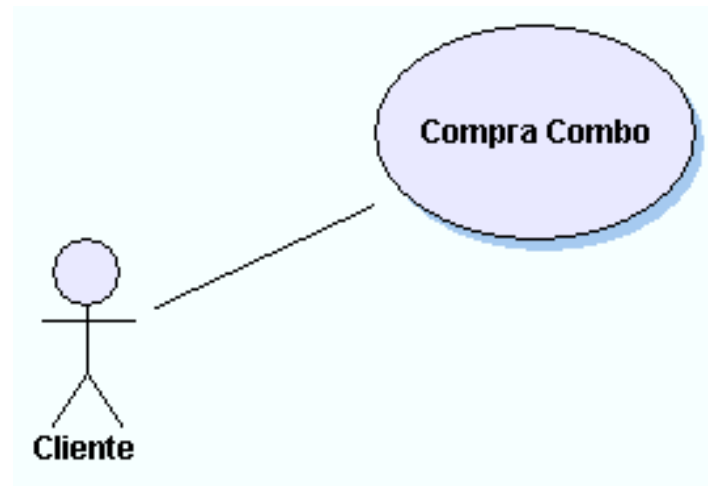
- Secuencia de acciones que produce un resultado útil y observable para un actor.
- El nombre del caso de uso debe reflejar la tarea específica que el actor desea llevar a cabo usando el sistema.
- Es representado en Enterprise Architect por:



# Diagrama de Casos de Uso

## Casos de Uso

- Un caso de uso es una descripción de la secuencia de interacciones que se producen entre un actor y el sistema, cuando el actor usa el sistema para llevar a cabo una tarea específica.



# Diagrama de Casos de Uso

## Colaboración

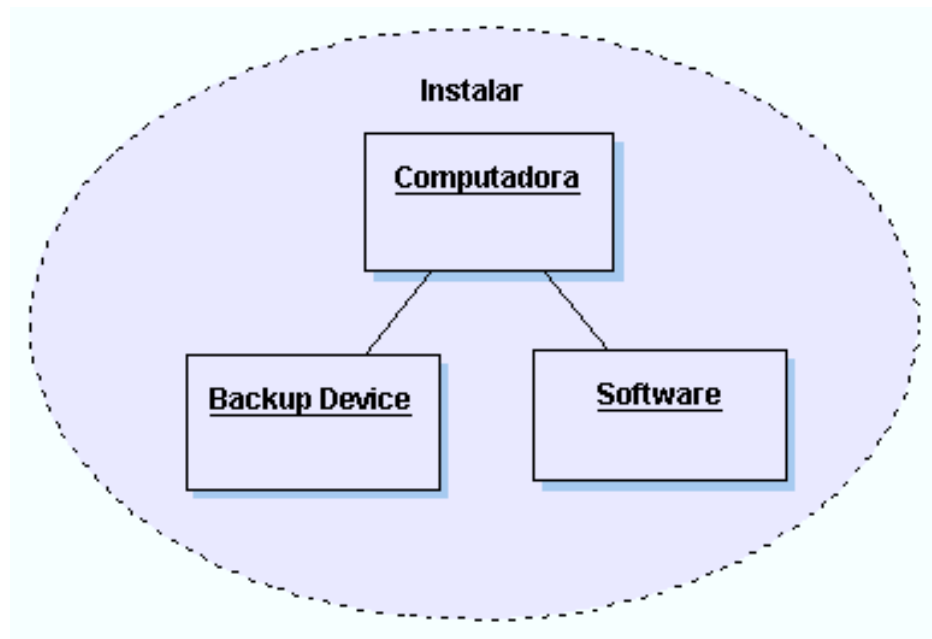
- Define un conjunto de roles cooperantes y sus conectores. Ellos se usan para ilustrar colectivamente una funcionalidad específica.
- Se representa en Enterprise Architect por:



# Diagrama de Casos de Uso

## Colaboración

- El siguiente ejemplo ilustra una colaboración "Instalar", con tres roles conectados como se muestra.

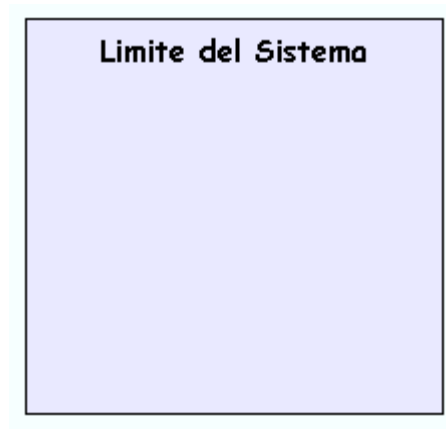




# Diagrama de Casos de Uso

## Límite del Sistema

- Barrera que define el interior y el exterior del sistema.
- Se representa en Enterprise Architect por:



# Diagrama de Casos de Uso

## Relaciones

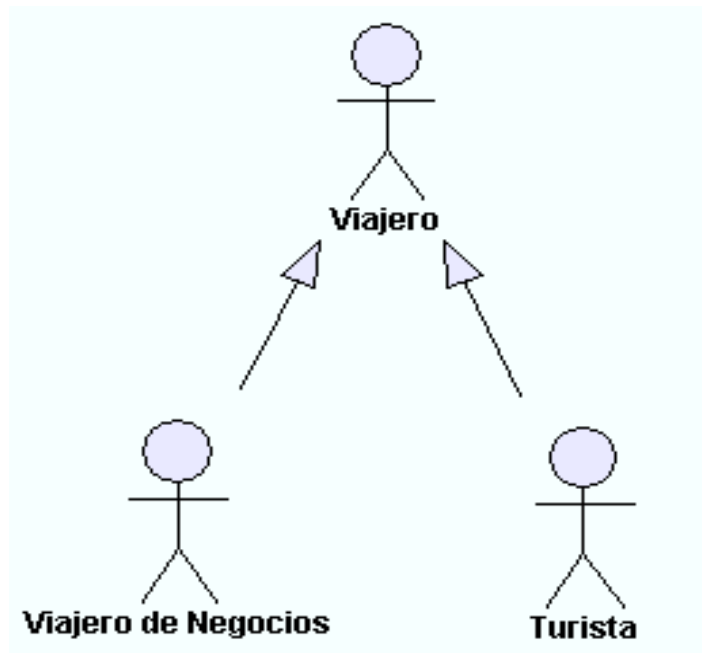
- Entre Actores:
  - Generalización
- Entre Casos de Uso y Actores:
  - Asociación
- Entre Casos de Uso:
  - Generalización
  - Inclusión
  - Extensión

# Diagrama de Casos de Uso

## R. Generalización

- **Entre Actores**

- Se usa una **generalización** para indicar herencia.
- El origen hereda las características del destino.

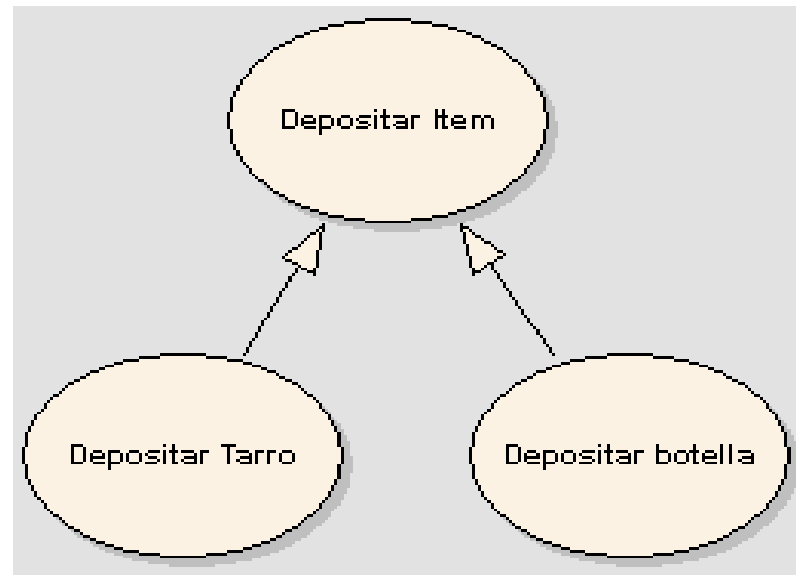


# Diagrama de Casos de Uso

## R. Generalización

- **Entre Casos de Uso**

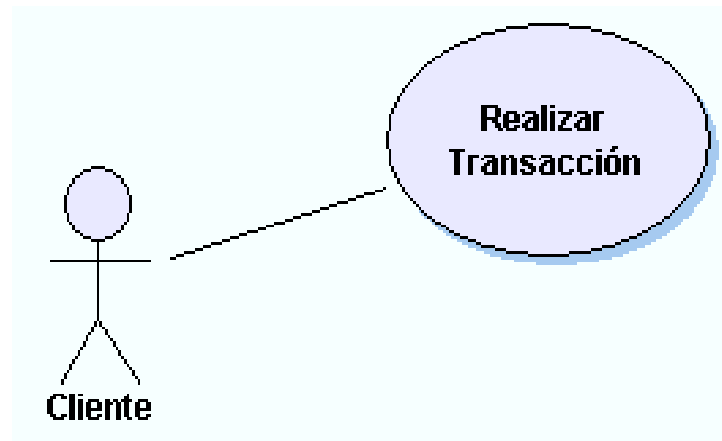
- El Caso de uso origen hereda la especificación del caso de uso destino, posiblemente la modifica y/o amplía.



# Diagrama de Casos de Uso

## R. Asociación

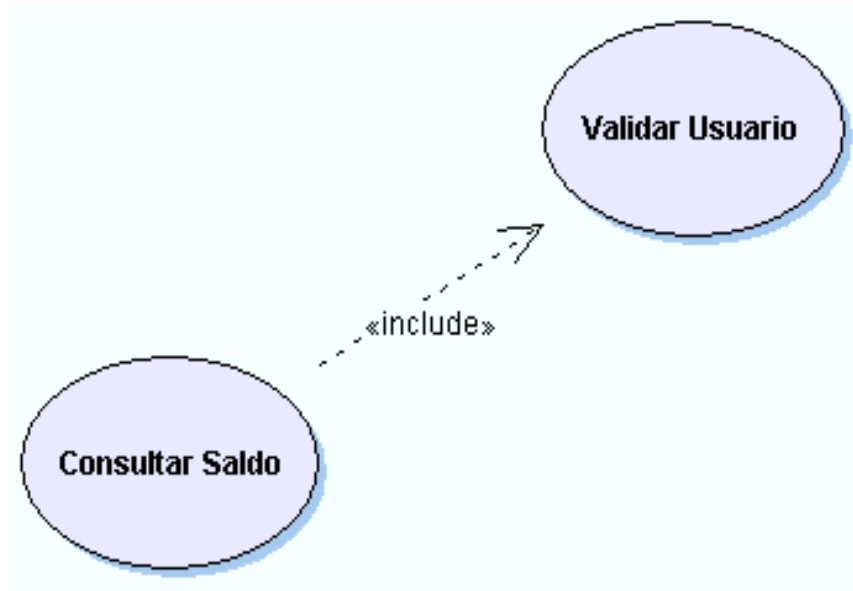
- Es el tipo de relación más básica que indica la invocación desde un actor o caso de uso a otra operación (caso de uso). Dicha relación se denota con una flecha simple.



# Diagrama de Casos de Uso

## R. Include

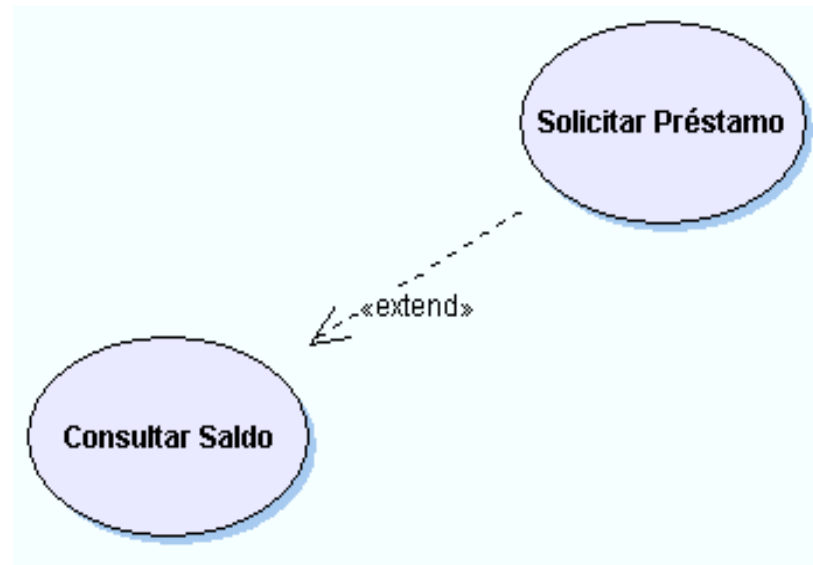
- Cuando un caso de uso utiliza a otro.



# Diagrama de Casos de Uso

## R. Extend

- Cuando un caso de uso especializa a otro extendiendo su funcionalidad.

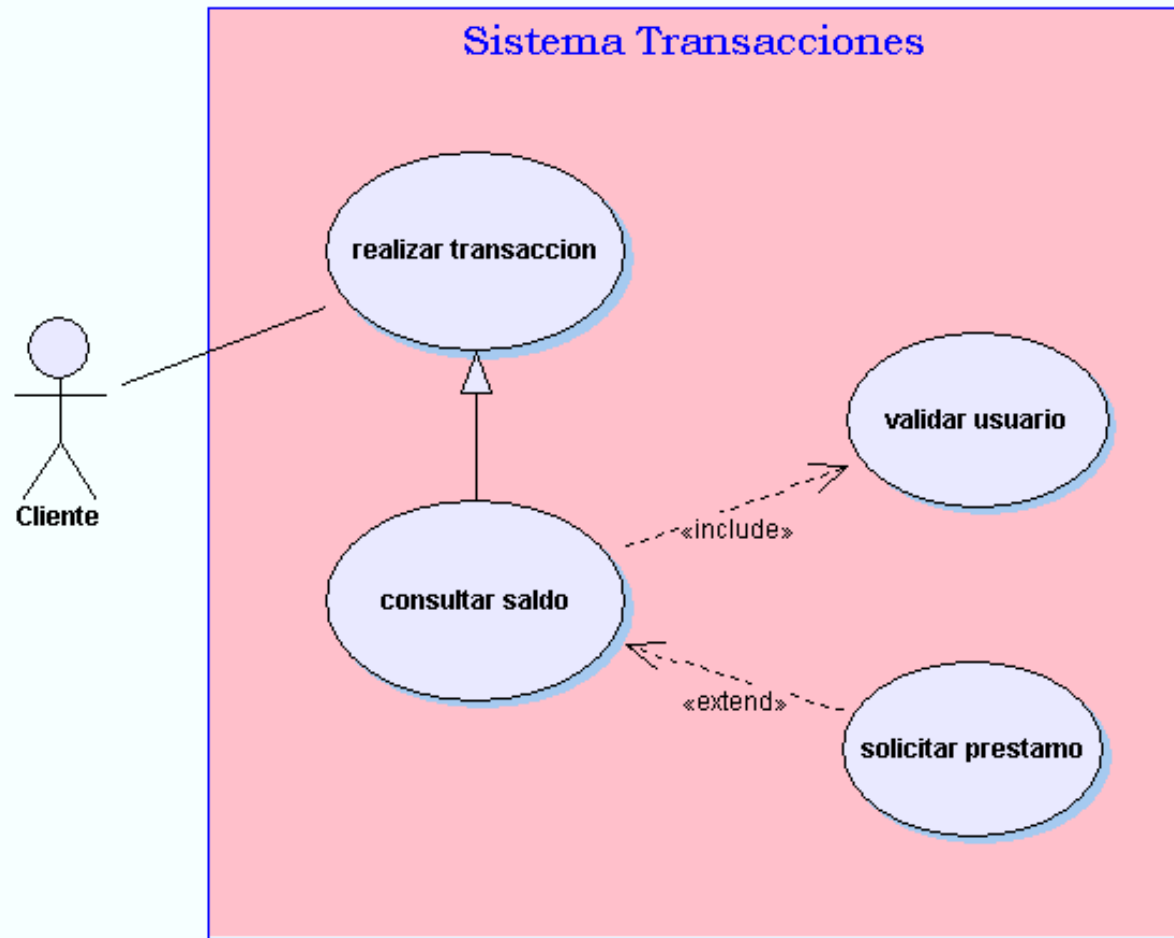


# Diagrama de Casos de Uso

- En el Diagrama de Casos de Uso se representa el sistema como una caja rectangular con el nombre en su interior.
- Los casos de uso están en el interior de la caja del sistema, y los actores fuera, y cada actor está unido a los casos de uso en los que participa mediante una línea.



# Diagrama de Casos de Uso

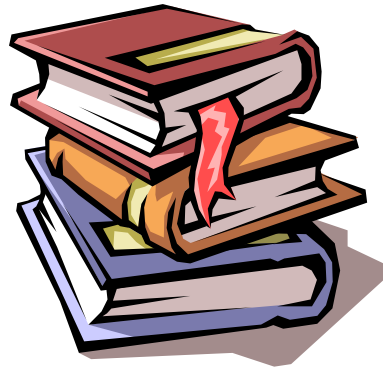


# Resumen

- Un **diagrama de Caso de Uso** describe los requisitos funcionales del sistema, la forma en la que los actores interactúan a través del límite del sistema y la respuesta del sistema.
- Elementos del **diagrama de Casos de Uso:**
  - Actores.
  - Casos de Uso.
  - Límite del Sistema.
- Relaciones del **diagrama de Casos de Uso:**
  - Entre Actores: Generalización.
  - Entre Casos de Uso y Actores: Asociación.
  - Entre Casos de Uso: Generalización, Inclusión, Extensión.

# Resumen

- El diagrama de Actividades muestra como es el comportamiento de un sistema así como esta interrelacionado por las particiones que lo conforman.



# Laboratorio

## Laboratorio 7.1

- Elaborar el Diagrama de Actividades identificando:
  - Punto de inicio
  - Actividad
  - Barra de Sincronización
  - Transición
  - Punto final

## Laboratorio 7.2

- Elaborar el Diagrama de Casos de Uso identificando:
  - Actores.
  - Casos de Uso.
  - Limites del sistema.
  - Relaciones.