

# Elementos de Ingeniería de Software

## Técnicas de modelado: Casos de uso

Prof. Mg. Marisa Daniela Panizzi

*“La parte más difícil de construir un sistema es precisamente saber que construir.  
Ninguna otra parte del trabajo conceptual es tan difícil como establecer los  
requerimientos...”*

*...la tarea más importante que el ingeniero de software hace para el cliente es la  
extracción iterativa y el refinamiento de los requerimientos del producto”*

Frederick P. Brooks, 1987

## ¿QUÉ SON LOS CASOS DE USO?

- Técnica para especificar el comportamiento de un sistema.
- Secuencia de interacciones entre un sistema y alguien o algo (otro sistema o una máquina) que usa alguno de sus servicios.
- Son una excelente forma de especificar el comportamiento externo del sistema.
- Su notación fue incorporada al lenguaje estándar de modelado UML (Lenguaje Unificado de Modelado)

### Como ejemplo:

En un sistema de ventas se debe ofrecer la posibilidad de ingresar un nuevo pedido de un cliente. Cuando un usuario accede a ese servicio se está ejecutando **el caso de uso “Ingresando pedido”**

# ¿QUÉ SON LOS CASOS DE USO?

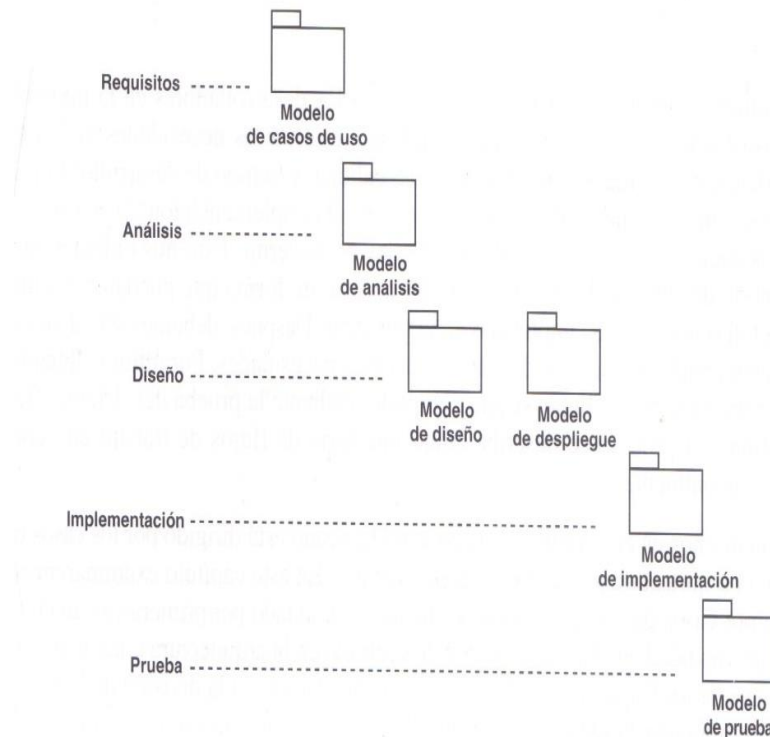
Los casos de uso, a pesar de ser considerados una técnica de Análisis Orientado a Objetos, son independientes del método de diseño que se utilice, y por lo tanto del método de programación.

Es probable esperar que en el futuro los métodos de análisis y diseño que prevalezcan adopten las principales ventajas de todos los métodos disponibles actualmente

# ¿QUÉ SON LOS CASOS DE USO?

Son mucho más que una herramienta para capturar requerimientos. Dirigen el proceso de desarrollo en su totalidad.

Para cada iteración planteada en la construcción de software, los casos de uso nos guían desde la toma de requisitos, pasando por análisis, diseño e implementación hasta la etapa de prueba.



## CAPTURA DE REQUERIMIENTOS CON CASOS DE USO

Tiene principalmente dos objetivos:


- Encontrar los verdaderos requerimientos, los que “agreguen” el valor esperado por los usuarios.
- Representar los requerimientos de un modo adecuado (comprensible) por usuarios y desarrolladores.

## ¿POR QUÉ CASOS DE USO?


- Proporcionan un medio sistemático e intuitivo de capturar requerimientos funcionales, centrándose en el usuario.
- Dirigen todo el proceso de desarrollo: todas las etapas pueden plantearse en términos de casos de uso.

## CAPTURA DE REQUERIMIENTOS CON VALOR AÑADIDO

- Tomando la perspectiva de cada tipo de usuario, se pueden capturar los requerimientos que necesitan para realizar su trabajo.
- La pregunta clave: ¿Qué se quiere que haga el sistema **para cada usuario**?
- Los usuarios y clientes pueden participar del proceso junto a los desarrolladores.
- Permite conseguir un acuerdo con usuarios y clientes sobre que debería hacer el sistema



Puede utilizarse  
como parte del  
contrato con el cliente

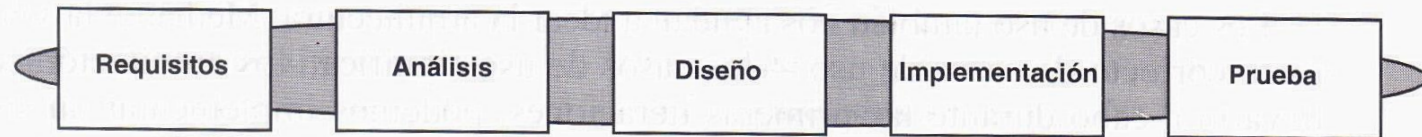


Permiten delimitar  
el sistema



## DIRIGEN EL PROCESO

Un proyecto dirigido por casos de uso progresa a través de una serie de flujos de trabajo que comienza con dichos casos (enlazan el proceso de desarrollo).



**Figura 3.2.** Los casos de uso enlazan los flujos de trabajo fundamentales. La elipse sombreada en el fondo simboliza cómo los casos de uso enlazan esos flujos de trabajo.

## DIRIGEN EL PROCESO

Permiten planificar, asignar, controlar tareas, estimar esfuerzos y tiempos, tamaño de proyecto y recursos necesarios (Jefe de Proyecto).

Dan soporte a la trazabilidad a través de todos los modelos utilizados en el desarrollo.

# DEFINICIONES BÁSICAS

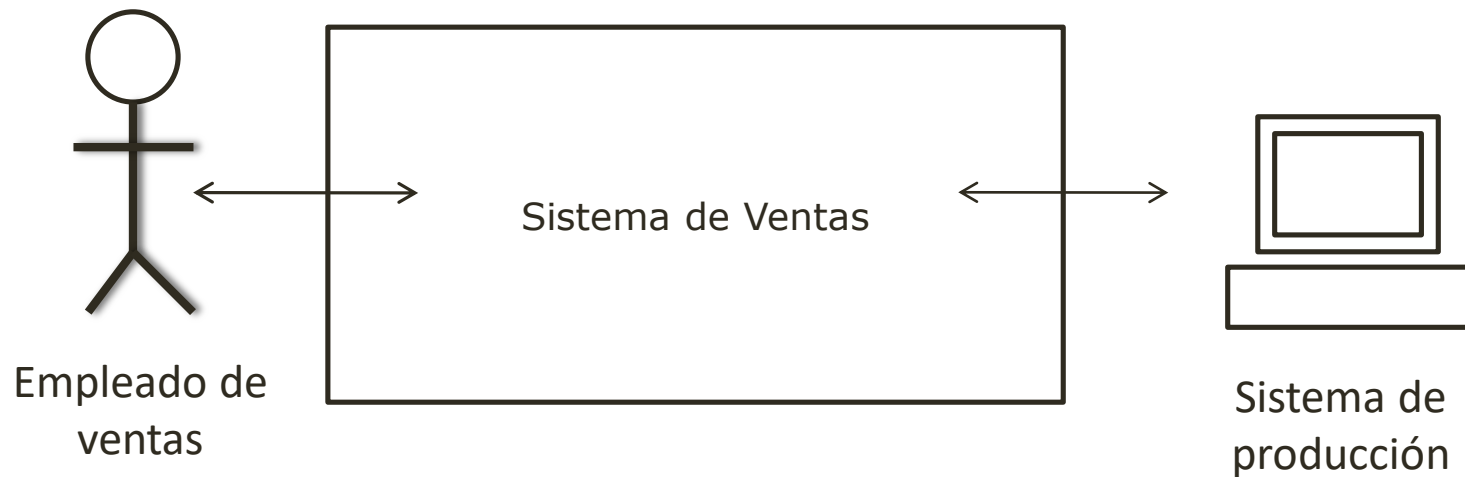
- ACTORES
- CASOS DE USO
- ALTERNATIVAS
- RELACIONES DE EXTENSIÓN
- RELACIONES DE INCLUSIÓN
- PRECONDICIÓN Y POSCONDICIÓN
- ACTORES Y CASOS DE USO ABSTRACTOS

# ACTORES

- Es una agrupación uniforme de personas, sistemas o máquinas que interactúan con el sistema.
- Los actores son externos al sistema: permiten comenzar a definir límites y alcances
- **UN ACTOR ES UNA CLASE DE ROL, UN USUARIO ES UNA PERSONA QUE CUANDO USA EL SISTEMA ASUME ESE ROL**

# ACTORES

- Se representan con “stick man” (Hombres de palo)



En UML siempre se representan a los actores con “hombres de palo” pero a veces es más útil representar a otros sistemas con un dibujo más claro (ej. una computadora)

# ACTORES

- Las flechas indican el flujo de información entre el actor y el sistema (no se utilizan en UML).
- Identificar a los actores es el primer paso para utilizar casos de uso.
- Generalmente los actores coinciden con el área de la empresa o jerarquías dentro de la organización.

## ACTORES - TIPIFICACIÓN

- ▶ *Actores principales*: personas que utilizan las funciones principales del sistema
- ▶ *Actores secundarios*: personas que efectúan tareas administrativas o de mantenimiento
- ▶ *Material externo*: dispositivos materiales imprescindibles que forman parte de la aplicación y que deben ser utilizados
- ▶ *Otros sistemas*: Aquellos con los que el sistema debe interactuar

## RELACIONES ENTRE ACTORES

✓ *Relación de generalización*: Se utiliza cuando se tienen actores que representan un rol muy parecido pero con ligeras diferencias entre sí.

Se utiliza para indicar que un actor específico puede hacer lo mismo que otro más general, más otras cosas particulares de él.

Ejemplo:





# CASOS DE USO

Es una secuencia de acciones que el sistema lleva a cabo para ofrecer algún resultado de valor para el usuario

- El nombre de un caso de uso se expresa con un **verbo en gerundio, seguido por un objeto**.
- Se representan con un **óvalo** con el nombre del caso en su interior.
- Se expresan **SIEMPRE** desde el punto de vista del actor y no del sistema.
- Se documentan con **texto informal**.
- Son iniciados por un **único actor**.
- Representan una determinada funcionalidad, **claramente definida del sistema**.

# TIPOS DE CASOS DE USO

## TEMPORALES:

Su inicio es provocado por el paso del tiempo (sin actor)

Ej. “Recibiendo estadística mensual de ventas”

Para diferenciarlos pueden ser representados con un óvalo pero de bordes de línea punteada.

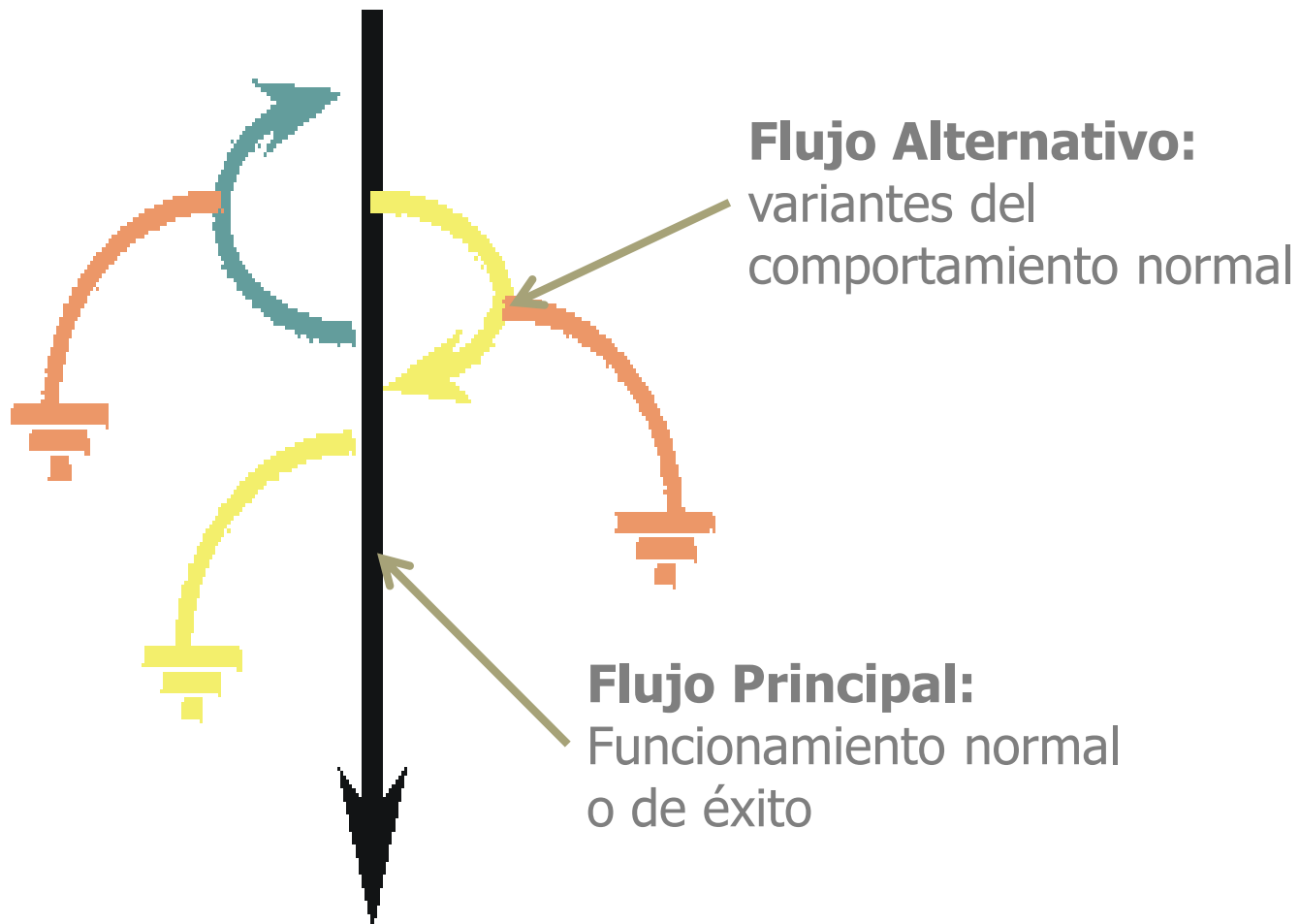
## TIPOS DE CASOS DE USO

Primarios	Secundarios
<ul style="list-style-type: none"><li>• Su ejecución es NECESARIA para que el sistema funcione</li></ul>	<ul style="list-style-type: none"><li>• No son centrales para el sistema pero son necesarios para que el sistema funcione mejor.</li><li>• No todos son identificados en la etapa de requerimientos.</li></ul>

# ALTERNATIVAS

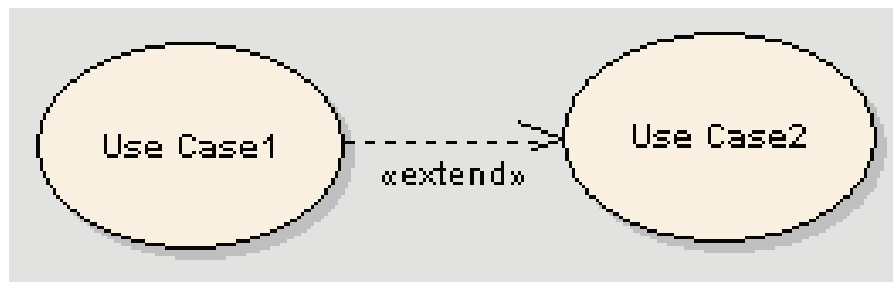
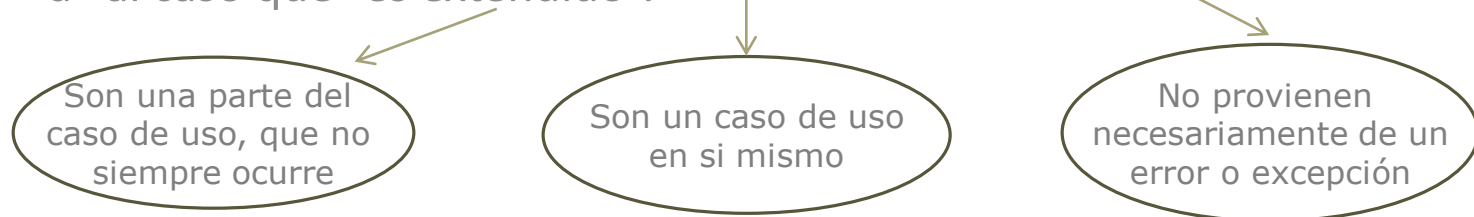
- Son errores o excepciones que “desvían” el curso normal de un caso de uso.
- No tienen sentido en si mismas.
- La bibliografía indica documentarlas al final del caso de uso, pero es de utilidad realizarlas dentro del caso de uso en dos columnas.

# ALTERNATIVAS



# RELACIONES DE EXTENSIÓN

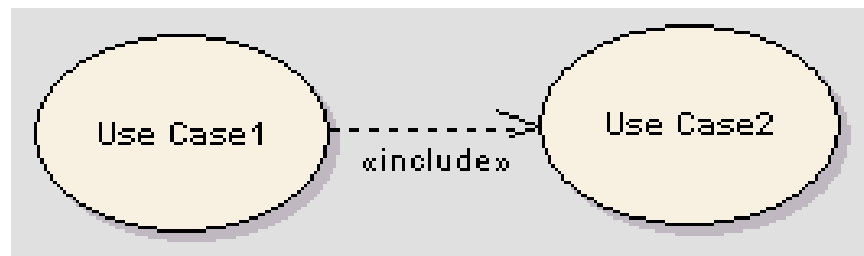
- Consisten en interrumpir un caso de uso y pasar a ejecutar otro pero solo en determinadas ocasiones.
- Se representan con una línea de trazos desde el caso que “extiende a” al caso que “es extendido”.



**SI ALGO OPCIONAL DEBE SER EXPRESADO EN MÁS DE UN PASO SEGURO ES EXCEPCIÓN Y NO ALTERNATIVA**

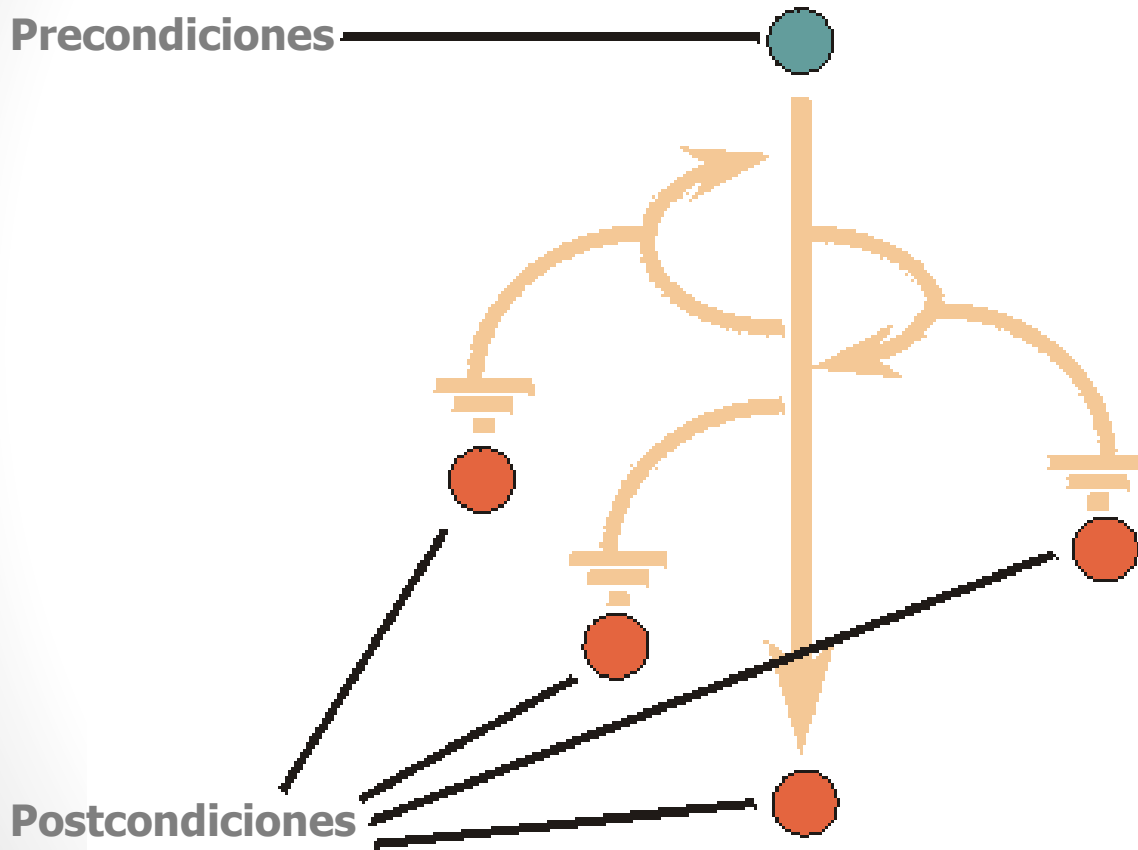
# RELACIONES DE INCLUSIÓN

- Es una funcionalidad común accedida desde varios casos de uso.
- Puede asociarse al concepto de subrutina o subprograma.
- Son casos de uso en sí mismos.
- Se utiliza cuando se detecta que en al menos dos casos de uso diferentes conceptualmente existe alguna secuencia de pasos o flujo de eventos repetida.
- Esta secuencia puede extraerse dando lugar a un nuevo caso de uso relacionado con los otros dos mediante una relación de inclusión.



## UN CASO DE USO TIENE ASOCIADAS PRECONDICIONES Y POSTCONDICIONES

**Precondiciones**



*Precondiciones:* reflejan el estado en el que debe estar el sistema y su entorno para que pueda comenzar la ejecución del caso de uso.

*Postcondiciones:* reflejan el estado en el que queda el sistema y su entorno luego de la ejecución del caso de uso.



# ORGANIZACIÓN DE LAS ESPECIFICACIONES

Algunas reglas útiles...

- 1- Un gráfico de casos de uso no debe mostrar más de 15 casos de uso.
- 2- Si se particiona el gráfico hacerlo por actor.
- 3- Relaciones de uso, extensiones y actores abstractos deben estar en el diagrama principal (ver punto 1).
- 4- Si las relaciones de uso no entran en el diagrama principal debo tener en cuenta que siempre muestro en un mismo diagrama todos los casos de uso que usan a otro.
- 5- Si un caso de uso es utilizado por casi todos los casos de uso debo evitar mostrarlo en el diagrama principal.
- 6- Es importante que a medida que se van encontrando los casos de uso, se identifiquen y documenten los términos que forman parte del dominio del problema en un Glosario.

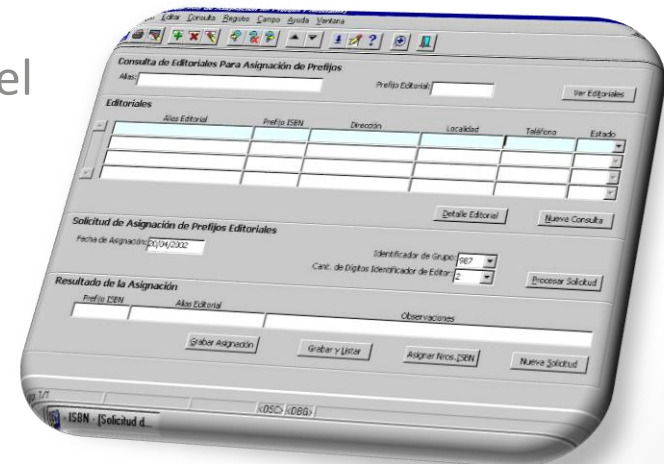
# ORGANIZACIÓN DE LAS ESPECIFICACIONES

❑ SECCIONES DE LA ESPECIFICACIÓN:  
(Orden sugerido. No es especificación completa)

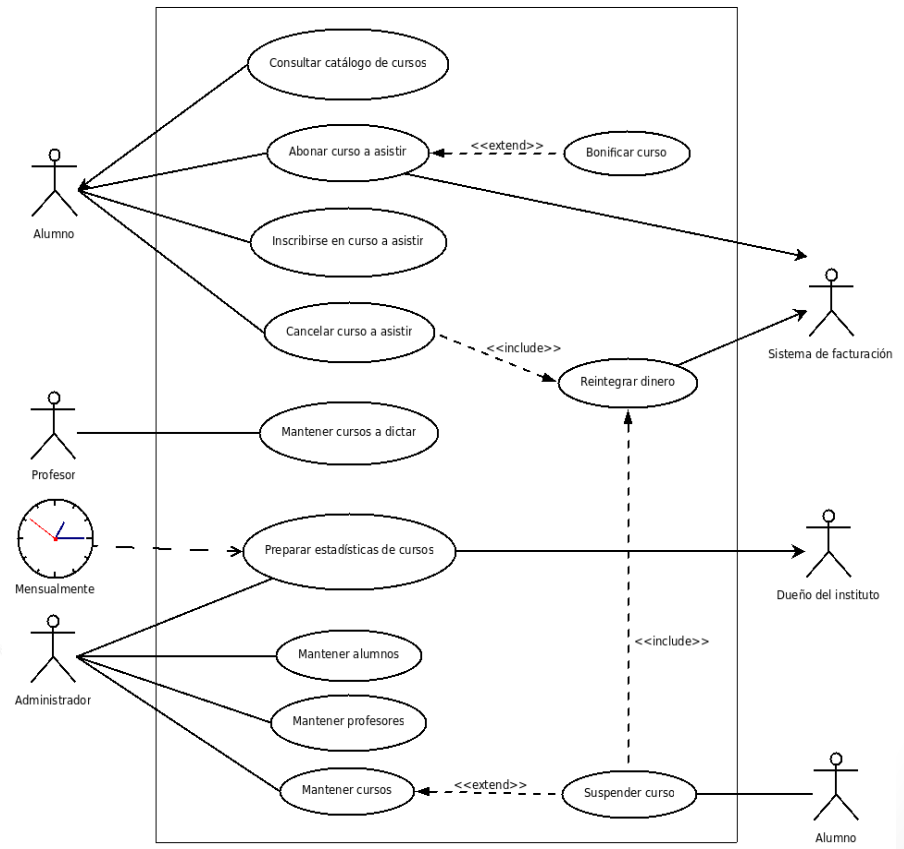
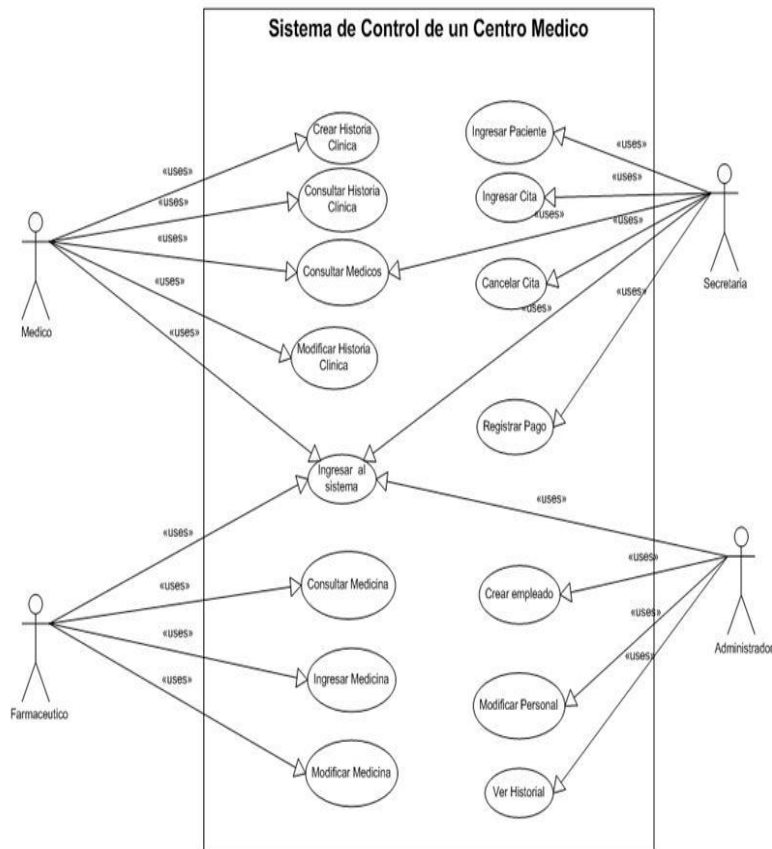
- 1- Propósito del sistema.
- 2- Gráficos de casos de uso.
- 3- Descripción de los casos de uso con sus alternativas.
- 4- Prototipos para los principales casos de uso.

# ESPECIFICACIONES - PROTOTIPOS DE INTERFAZ

- Se utilizan como ayuda para comprender y especificar las interacciones entre los actores humanos y el sistema.
- Sumamente útiles para estabilizar los requisitos. A los usuarios les resulta mucho más sencillo tomar decisiones o hacer especificaciones sobre algo visible que explicarlas en palabras.
- Documentación (no UML): dibujos en papel o en herramienta de dibujo,
- herramientas de interfaces gráficas, etc.



# EJEMPLOS...



# Manos a la Obra!!

Guía de Trabajos Prácticos  
(Casos Prácticos 11 y 12)

