

APUNTES DIAGRAMAS FIS

Diagrama de CU → Tema 2.3


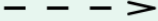
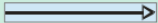

Representación → Diagrama de CU de UML y plantillas

Pasos:

1. Identificar actores(principales y secundarios)
2. Identificar principales CU(pag 53)
 - a. Tareas principales¿?
 - b. Información sistema adquirir,modificar,..¿?
 - c. Informar actor sobre cambios exterior¿?
 - d. Información adquirir actor del sistema¿?
 - e. Actor desea informarse de cambios que se producen en sistema¿?
3. Identificar nuevos CU a partir de los existentes
 - a. Variaciones significativas de los CU existentes
 - b. Acciones opuestas
 - c. Acciones que se realizan antes o después de los Cu exitentes
4. Hacer diagramas CU y de paquete
5. Hacer descripción básica de cada CU → plantilla
6. Definir prioridades y CU primarios
7. Descripción extendida CU → plantilla extendida
8. Realizar diagrama actividad
9. Desarrollar prototipos interfaz usuario

**** Digo yo que lo de crear otro tipos de diagramas serán en otros ejercicios aunque lo pongan las diapositivas ****

Tipos relaciones CU

Tipo de Relación	Definición	Notación
Asociación	Relación entre un actor y el caso de uso en el que participa.	
Extensión	Relación entre casos de uso. Representa la inserción de fragmentos de comportamiento adicional sin que el caso de uso base sepa de los casos de uso de extensión.	<code><<extend>></code> 
Generalización	Relación entre un caso de uso general y otro más específico, que hereda y añade características al caso de uso general.	
Inclusión	Relación entre casos de uso. Representa la inserción de comportamiento adicional dentro del caso de uso base que explícitamente hace referencia al caso de uso de inclusión	<code><<include>></code> 

Como saber si relación inclusión o extensión

Extensión o Inclusión

Heurística:

- **Usar** relaciones de **extensión** para comportamientos excepcionales, opcionales o que rara vez suceden.
- **Usar** relaciones de **inclusión** para comportamientos que se comparten entre dos o más casos de uso, o bien para separar un caso de uso en subunidades.

Diagrama de Paquetes → 2.3 pag 56 y **seminario**

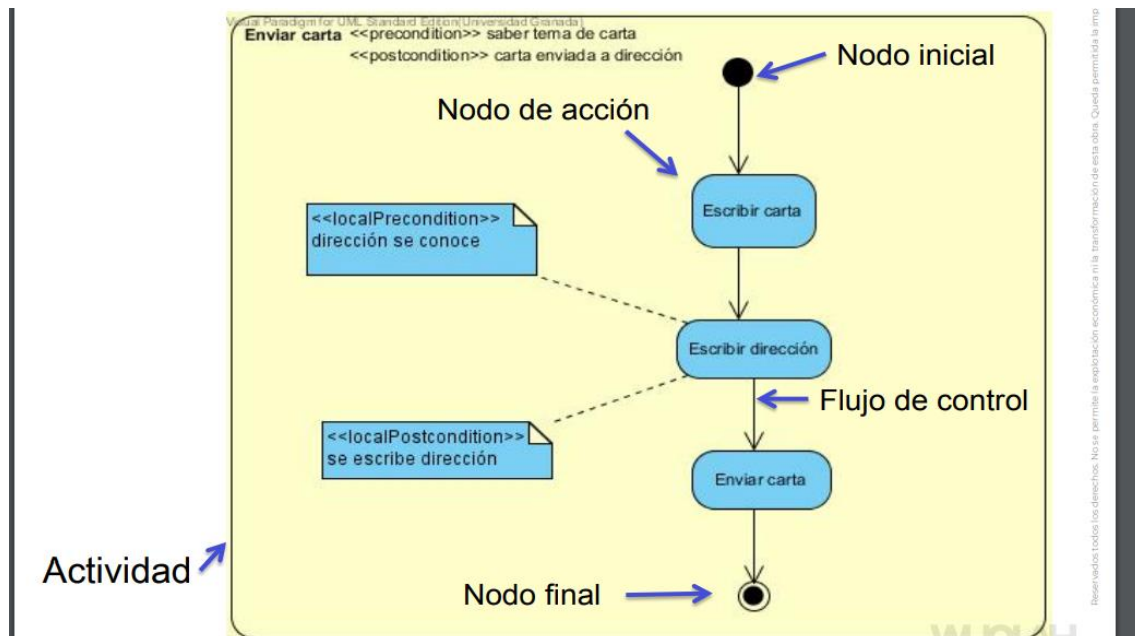
Representación → Diagrama paquetes UML

Reglas para la elaboración de paquetes: Un paquete bien estructurado debe:

- Ser lo más cohesivo posible.
- Estar poco acoplado con otros paquetes.
- Contener un grupo de clases relacionadas estrechamente.
- Poseer un conjunto equilibrado de elementos.
- Evitar dependencias cíclicas con otros paquetes

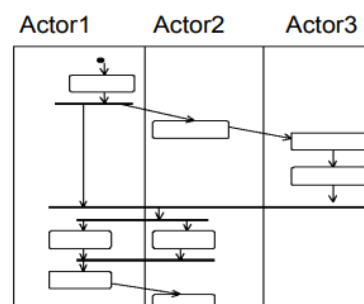
Diagrama de actividad → Seminario

Representación → Diagrama de actividad UML

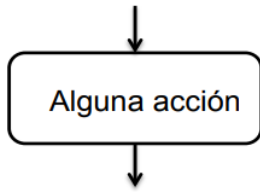


- **Particiones de actividad (Calles):**

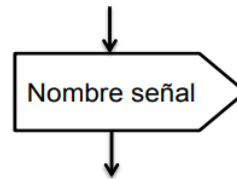
- Se usan para representar que las actividades del diagrama se realizan por distintos objetos, actores o grupos de actores.
- Cada partición de actividad representa una agrupación de alto nivel de acciones relacionadas.
- Se puede dividir las actividades en particiones utilizando líneas verticales, horizontales o curvas.
- Las particiones pueden estar jerárquicamente anidadas.



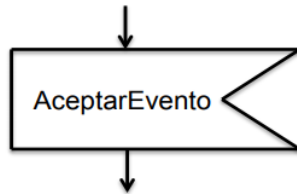
- **Tipo de nodos de acción:**



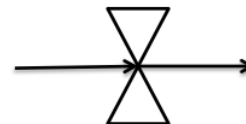
Nodo de acción de llamada



Enviar señal



Nodo de acción de aceptar evento



Expresión de tiempo

Nodo de acción de aceptar evento de tiempo

- **Nodos de control:** Los nodos de control gestionan el flujo de control dentro de una actividad.



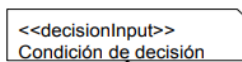
Nodo inicial



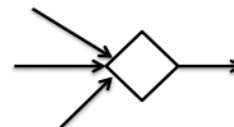
Nodo final de actividad



Nodo final de flujo



Nodo de decisión



Nodo de fusión



Nodo fork



Nodo join (sincronización)

Diagrama Conceptual → Tema 2.4

Representación → Diagrama de clases UML

Pasos a seguir:

1. Identificar e incorporar **conceptos(sustantivos)** → solo los relevantes → clases en diagrama
2. Identificar e incorporar **asociaciones** entre estos
Identificarlas, representarlas y seleccionar las válidas
Asignar nombre
Asignar multiplicidades
3. Identificar e incorporar **generalizaciones**(Herencia para que nos entendamos)
Identificarlas → clases elementos comunes y definir relaciones superclase
Validarlas → Regla 100%(conformidad con la definición de la superclase) y regla "es un "(conformidad pertenencia al conjunto que define la superclase)
4. Identificar e incorporar **atributos**
5. Estructurar / **empaquetar modelo**

Diagrama de secuencia del sistema(DSS)→ 2.4

Representación→Diagrama de secuencia UML

Pasos:

- I. Identificar actores
- II. Asignar nombre al sistema
- III. Identificar/nombrar operaciones principales de los CU
- IV. Ver parámetros de las operaciones
- V. Representarlas
- VI. Incluir operaciones en la clase que identifica a todo el sistema del diagrama conceptual(esto último no se ,a mi no me suena)

Nosotros hacemos un DSS por diagrama de CU

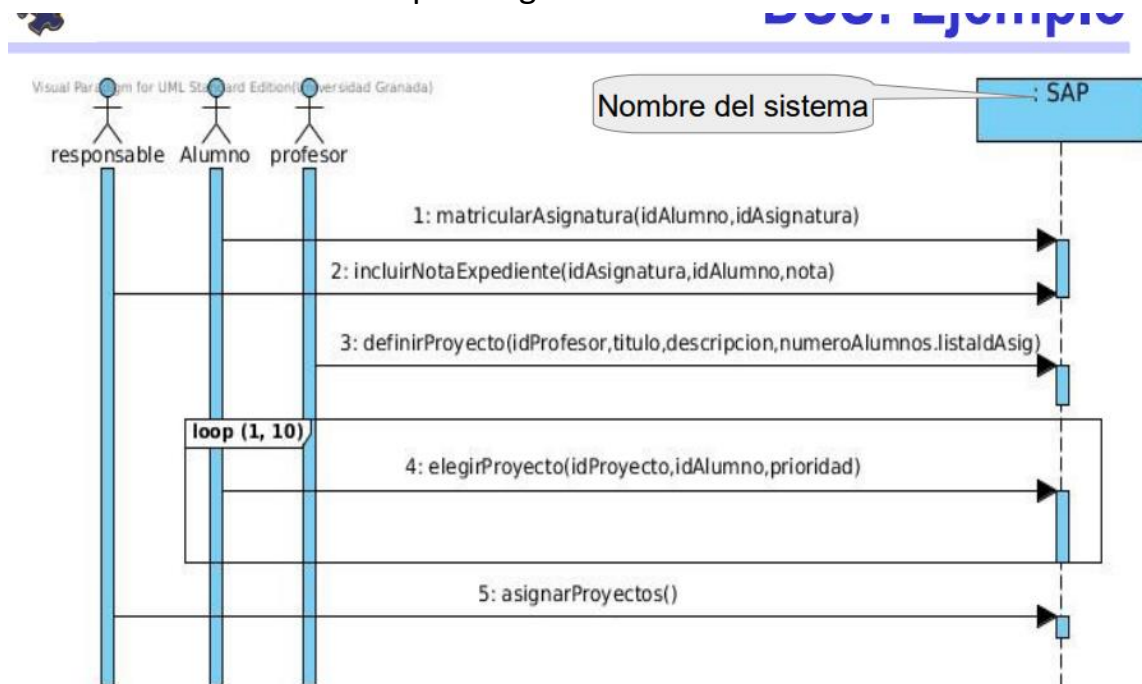


Diagrama de comunicación → **seminario**

Representación → Diagrama de comunicación UML

Muestran una colaboración concreta entre un grupo de objetos a través del envío de mensajes, para llevar a cabo una determinada operación, mostrando las vías de comunicación (enlaces) entre objetos de forma explícita .

Compuesto, esencialmente, por:

- Elementos estructurales: Objetos y Actores.
- Enlaces entre Actores/Objetos y Objetos/Objetos.
- Mensajes entre Actores/Objetos y Objetos/Objetos.
- Estereotipos y restricciones.

Estereotipos de visibilidad aplicables a los enlaces:

association: El objeto de la clase B tiene una vía de comunicación con el objeto de la clase A debido a que entre ellos existe una relación fuerte y duradera (entre A y B hay una asociación).

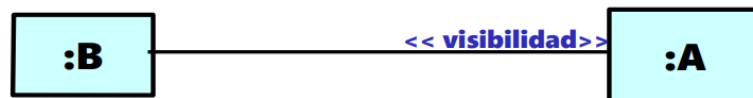
self: Un objeto siempre tiene una vía de comunicación con el mismo. (envío de mensaje a this).

global: Objeto de la clase A es visible desde el objeto B porque su alcance contiene al del B. (variables static de las clases).



local: El objeto de la clase B tiene una vía de comunicación con el objeto de la clase A debido a que tienen una relación débil y temporal (variables definidas dentro de los métodos).

parameter: El objeto de la clase B tiene una vía de comunicación con el objeto de la clase A debido a que éste es pasado como parámetro a algunas de las operaciones de B. (parámetros de las métodos).

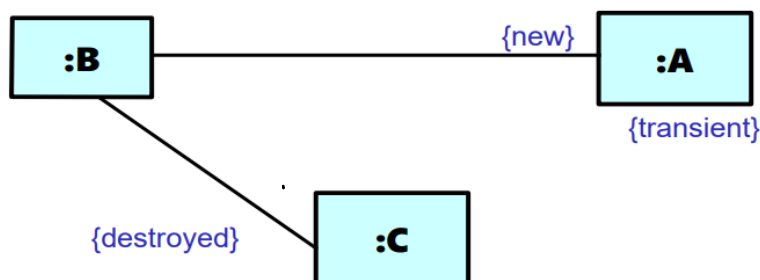


Restricciones aplicables a los objetos y enlaces:

new: Instancia o enlace que se crea durante la interacción.

destroyed: Instancia o enlace que se destruye durante la interacción.

transient: Instancia o enlace que se crea y se destruye durante la interacción.



Ejemplo de un DC: Promocionar un empleado a director.

