



INGENIERÍA DEL SOFTWARE I

Práctica 2

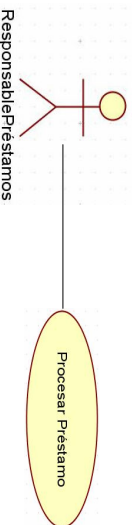
Especificación de Requisitos

Univ. Cantabria – Fac. de Ciencias
María Sierra y Patricia López



Nociones de UML para Requisitos: Casos de Uso

- **Caso de Uso**
 - "Una descripción de un conjunto de secuencias de acciones, incluyendo variantes, que ejecuta un sistema para producir un resultado observable de valor para un actor"
 - Cada caso de uso describe un comportamiento esperado de un sistema



Nociones de UML para Requisitos: Casos de Uso

- Capturan los Requisitos del sistema que aportan un valor añadido a algún usuario
 - Importancia de la perspectiva del **USUARIO**
 - ¿A quién ayudan?, ¿Qué necesidades satisfacen?, ¿Cuánto valor añaden al negocio?
 - La captura de los casos de uso implica a:
 - Usuarios/Clientes → Son los expertos en los requisitos
 - Desarrolladores → Deben ayudar a los usuarios y clientes a comunicar sus necesidades
- El Modelo de Casos de Uso es la especificación de todas las formas posibles de usar un Sistema desde la perspectiva de sus usuarios



Nociones de UML para Requisitos: Clases

- **Modelado:** Identificación de las "cosas" importantes desde una visión particular.
- En UML estas "cosas" se modelan como clases
 - Constituyen el vocabulario del sistema que se modela
 - Cada una de ellas tiene ciertas propiedades y un comportamiento.
- Utilizando OO se simplifica el mapeado del mundo real al mundo de la solución tecnológica (Dominio del problema frente a dominio de la solución).
- Para modelar el vocabulario de un sistema:
 - Hay que identificar aquellas cosas que utilizan los usuarios/programadores para describir el problema o la solución.
 - Identificar las responsabilidades.
 - Definir atributos y métodos necesarios para cumplir con las responsabilidades.
- Organización en Paquetes

Nociones de UML para Requisitos: Clases

- **Nombre:**
 - Las distingue de las demás.
 - Ha de ser simple.
 - Pueden mostrar el "camino"
- **Atributo:**
 - Propiedad que es compartida por todos los objetos de la clase.
- **Método:**
 - Es la implementación de un servicio que pueda ser requerido.
 - Es una abstracción de algo que puede hacer el objeto.
- **Responsabilidad:**
 - Contrato u obligación de una clase, debe ser tenido en cuenta al modelar clases.

Ventana
situación
minimizar()
cerrar()
maximizar()
redimensionar()

Pared
situa: float
anchura: float
grosor: float
esMaestra: Boolean = false

Pertinencias
Responsabilidades
-Determinar el riesgo de un siniestro de un cliente
-Mejorar criterios de fraude
-especificos del cliente

María Sierra, Patricia Lopez - ISI

P2.5

Análisis Textual

- **Análisis Textual:**
 - Técnica para la captura y análisis de requisitos
 - En Visual Paradigm permite identificar los siguientes elementos de modelado candidatos:
 - Actores
 - Casos de Uso
 - Clases
- **Tutorial interactivo en**
<http://www.visual-paradigm.com/product/vpuml/demos/requirements/textualanalisys.jsp>

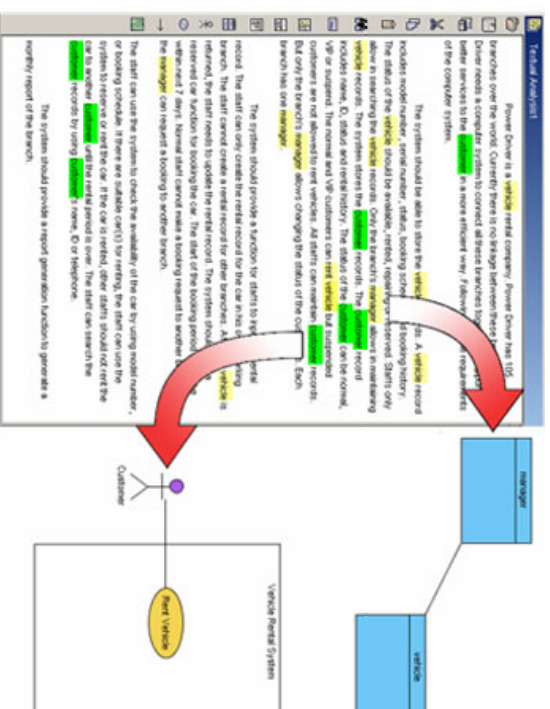
María Sierra, Patricia Lopez - ISI

P2.7

Especificación de Requisitos:

Análisis Textual

Análisis Textual en VP



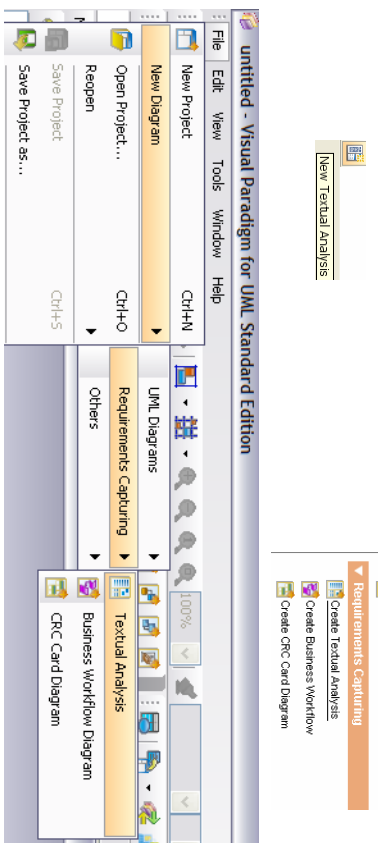
María Sierra, Patricia Lopez - ISI

P2.8



Análisis Textual en VP

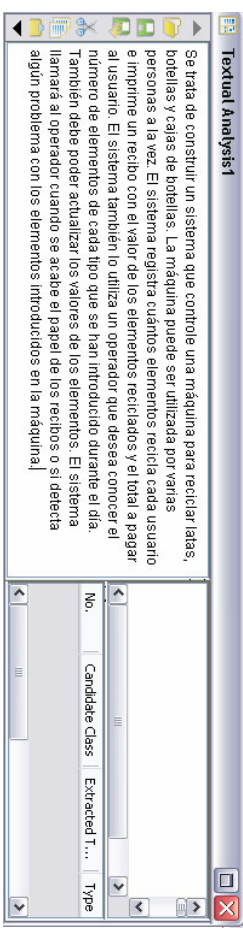
1) Crear Modelo de Análisis Textual



Análisis Textual en VP

2) Editar la Especificación Textual del Problema

- Ejemplo: Máquina de Reciclaje

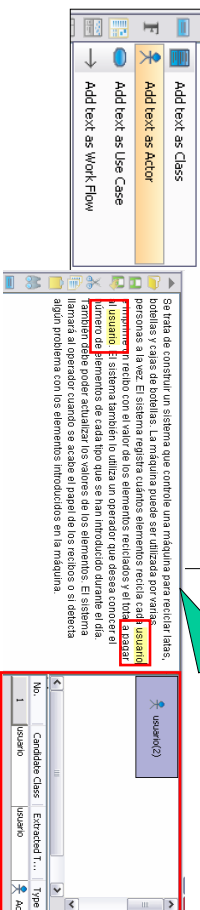


Análisis Textual en VP

3) Identificación de elementos candidatos

- Seleccionar texto – Add Candidate Class – Seleccionar el tipo

Se trata de construir un sistema que controle una máquina para reciclar latas, botellas y cajas de botellas. La máquina puede ser utilizada por varias personas a la vez. El sistema registra cuántos elementos recicla cada usuario e imprime un recibo con el valor de los elementos reciclados y el total a pagar al usuario. El sistema también lo utiliza un operador que desea conocer el número de elementos de cada tipo que se han introducido durante el día. También debe poder actualizar los valores de los elementos. El sistema llamará al operador cuando se acabe el papel de los recibos o si detecta algún problema con los elementos introducidos en la máquina.

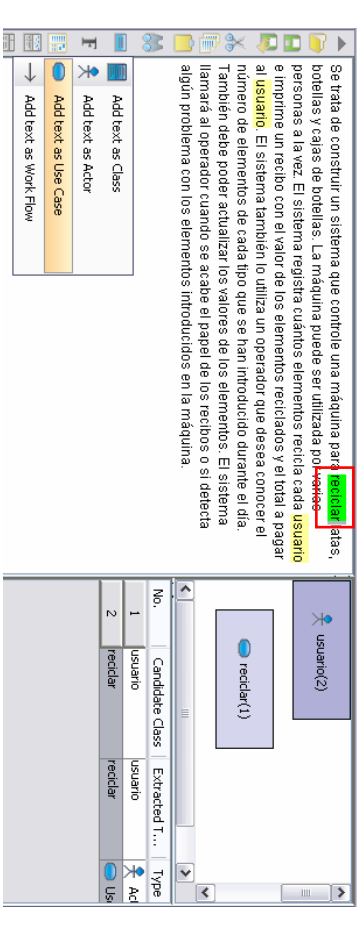


Análisis Textual en VP

3) Identificación de elementos candidatos

- Seleccionar texto – Add Candidate Class – Seleccionar el tipo

Se trata de construir un sistema que controle una máquina para reciclar latas, botellas y cajas de botellas. La máquina puede ser utilizada por varias personas a la vez. El sistema registra cuántos elementos recicla cada usuario e imprime un recibo con el valor de los elementos reciclados y el total a pagar al usuario. El sistema también lo utiliza un operador que desea conocer el número de elementos de cada tipo que se han introducido durante el día. También debe poder actualizar los valores de los elementos. El sistema llamará al operador cuando se acabe el papel de los recibos o si detecta algún problema con los elementos introducidos en la máquina.





Análisis Textual en VP

4) Completar la información de los elementos candidatos

Se trata de construir un sistema que permita controlar una máquina para reciclar latas, botellas y cajas de bodegas. La máquina puede ser utilizada por varias personas a la vez. El sistema registra cuántos elementos recicla cada operador con el valor de los elementos reciclaos y el total a pagar al operador que desea introducir durante el día. También debe poder actualizar los valores de los elementos. El sistema llamará al operador cuando se acabe el papel de los reciclo o si detecta algún problema con

No.	Candidate Class	Extracted ...	Type	Class Description	Occurrence
1	usuario	usuario	Actor	El usuario es la persona que usa la máquina para reciclar latas.	
2	operador	operador	Actor		
3	operador	operador	Actor		
4	acabar el papel de reciclar	acabar el papel de reciclar	Use Case		
5	problema con los elementos con los	problema con los elementos con los	Use Case		
6	latas	latas	Class		
7	botellas	botellas	Class		
8	cajas	cajas	Class		
9	conocer el número (conocer el número)	conocer el número (conocer el número)	Use Case		
10	actualizar los valores/actualizar los valo	actualizar los valores/actualizar los valo	Use Case		

María Sierra, Patricia López - ISI

P2.13



Análisis Textual en VP

5) Crear elementos del modelo a partir de los candidatos.

Forma 2: Creando un diagrama de casos de uso (o de clases) y arrastrando a él los candidatos que queremos crear

The image shows a transition from a list of candidates in the 'Model Explorer' to a 'Use Case Diagram' in the 'UML Diagram' window. A red arrow points from the 'usuario' candidate in the list to the 'usuario' actor in the diagram. Another red arrow points from the 'operador' candidate to the 'operador' actor in the diagram. A black arrow points from the 'usuario' actor to the 'operador' actor, indicating a relationship.

María Sierra, Patricia López - ISI

P2.15



Análisis Textual en VP

5) Crear elementos del modelo a partir de los candidatos.

Forma 1: Desde el propio diagrama de análisis textual

The image shows a 'Use Case Diagram' in the 'UML Diagram' window. The diagram contains several use cases and actors. The 'usuario' actor is connected to use cases 'conocer el número...', 'actualizar los valo...', and 'problema con los...'. The 'operador' actor is connected to use cases 'acabar el papel de...', 'actualizar los valo...', and 'problema con los...'. The 'lata', 'botella', and 'caja' classes are also present. The 'usuario' actor is highlighted in green, and the 'operador' actor is highlighted in red. The 'conocer el número...' use case is highlighted in green, and the 'actualizar los valo...' use case is highlighted in red. The 'problema con los...' use case is highlighted in green, and the 'acabar el papel de...' use case is highlighted in red. The 'lata', 'botella', and 'caja' classes are highlighted in green, and the 'usuario' actor is highlighted in red.

María Sierra, Patricia López - ISI

P2.14



Análisis Textual en VP

5) Crear elementos del modelo a partir de los elementos candidatos

The image shows a table of candidates in the 'Textual Analysts 1' interface. The table has columns for 'No.', 'Candidate Class', 'Extracted ...', 'Type', 'Class Description', and 'Occurrence'. The 'usuario' candidate is highlighted in red, and the 'operador' candidate is highlighted in red. The 'conocer el número...' use case is highlighted in green, and the 'actualizar los valo...' use case is highlighted in red. The 'problema con los...' use case is highlighted in green, and the 'acabar el papel de...' use case is highlighted in red. The 'lata', 'botella', and 'caja' classes are highlighted in green, and the 'usuario' actor is highlighted in red.

No.	Candidate Class	Extracted ...	Type	Class Description	Occurrence
1	usuario	usuario	Generated Model	El usuario es la persona que usa la máquina para reciclar latas, botellas o cajas.	
2	operador	operador	Generated Model	Se encarga del mantenimiento de la máquina.	
3	operador	operador	Generated Model		
4	acabar el papel de reciclar el papel de	acabar el papel de reciclar el papel de	Generated Model		
5	problema con los elementos con los	problema con los elementos con los	Generated Model		
6	latas	latas	Generated Model	tipo de elemento reciclable	
7	botellas	botellas	Generated Model	tipo de elemento reciclable	
8	cajas	cajas	Generated Model	tipo de elemento reciclable	
9	conocer el número (conocer el número)	conocer el número (conocer el número)	Generated Model		
10	actualizar los valores/actualizar los valo	actualizar los valores/actualizar los valo	Generated Model		

María Sierra, Patricia López - ISI

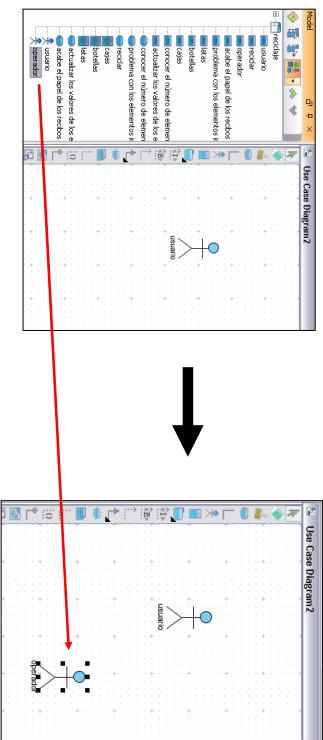
P2.16



Análisis Textual en VP

6) Crear diagramas con los elementos identificados

- Utilizar los elementos en los modelos correspondientes (cdu, clases, ..)
- Vista: Model
- Seleccionar Elementos candidatos o elementos del modelo
- Arrastrarlos hasta su diagrama correspondiente.



María Sierra, Patricia López - ISI

P2.17



Diagramas de Requisitos

Diagrama de Requisitos:

- Técnica que permite la especificación y captura de requisitos de forma efectiva y práctica. Permite mostrar de forma gráfica la taxonomía de los requisitos de usuario a un nivel de abstracción alto.

Tutorial interactivo de diagrama de requisitos en VP en

<http://www.visual-paradigm.com/product/vpuml/demos/requirements/requirements.jsp?type=voice>

María Sierra, Patricia López - ISI

P2.19

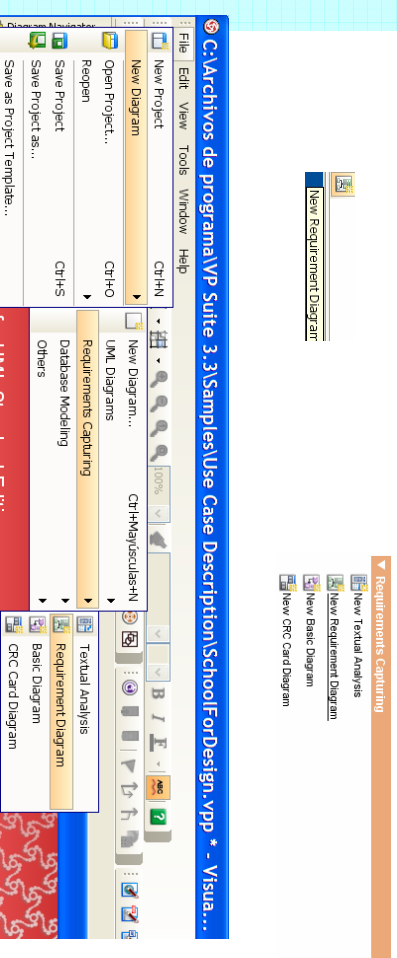


Especificación de Requisitos: Diagramas de Requisitos



Diagramas de requisitos en VP

1) Crear un Diagrama de Requisitos



María Sierra, Patricia López - ISI

P2.20



Diagramas de requisitos en VP

2) Creación de un Requisito

- Nombre
- Texto
- ID
- Tipo
- Método de Verificación
- Riesgo

Example 1: ID = "11", Text = "El coche debe satisfacer el prof...", Source = "Weakling", Kind = "Performance", Verify Method = "Test", Risk = "Medium".

Example 2: ID = "22", Text = "El coche debe satisfacer el prof...", Source = "Weakling", Kind = "Performance", Verify Method = "Test", Risk = "Low".

Example 3: ID = "22", Text = "El coche debe satisfacer el prof...", Source = "Weakling", Kind = "Performance", Verify Method = "Test", Risk = "High".

P2.21



Diagramas de requisitos en VP

3) Personalización de Requisitos

Name	Type	Default Value
Text	Text	Text
ID	Text	Text
Source	Text	Text
Kind	Enumeration	Enumeration
Verify Method	Enumeration	Enumeration
Risk	Enumeration	Enumeration

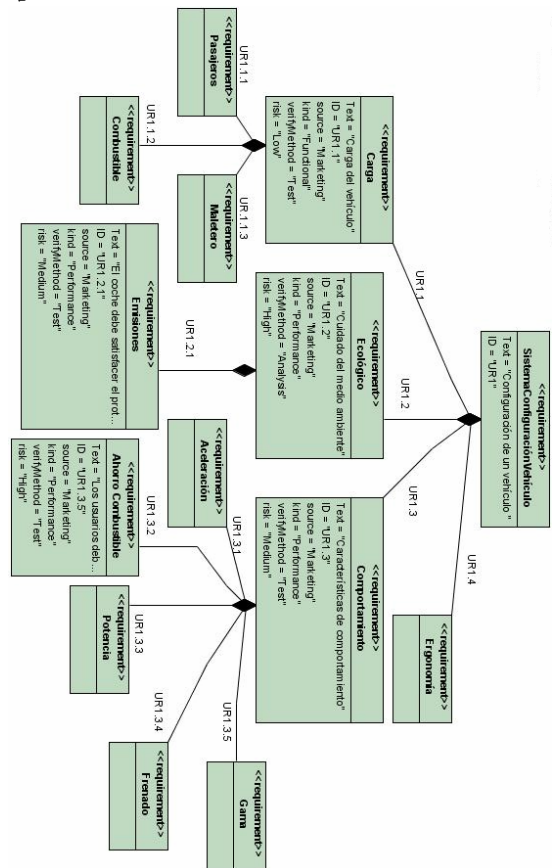
María Sierra, Patricia Lopez - ISI

P2.22



Diagramas de requisitos en VP

Ejemplo: Requisitos para un sistema de configuración de un vehículo



María



Especificación de Requisitos:

Tarjetas CRC



Tarjetas CRC

- Tarjetas CRC: Class-Responsability-Collaborator
 - Técnica de modelado orientado a objetos que permite identificar las clases que forman el vocabulario del sistema y sus responsabilidades e interacciones.
 - Utilizada en la fase inicial del desarrollo
 - *Tal clase hace tal cosa* apoyándose en *tal(es) clase(es)*
- Consiste en elaborar para cada clase identificada una **TARJETA**:
 - Nombre
 - Lista de superclases
 - Lista de subclases
 - Descripción
 - Atributos
 - Responsabilidades / Colaboraciones

CRCCard		
Super Classes:		
Sub Classes:		
Description:		
Attributes:		
Name	Description	
Responsibilities:	Name	Collaborator

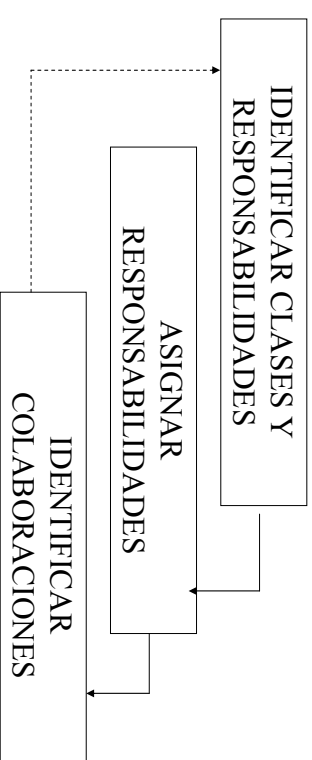
María Sierra, Patricia López - ISI

p2.25



Tarjetas de Clase

- **Construcción de las Tarjetas**



María Sierra, Patricia López - ISI

p2.26



Tarjetas CRC

- **Construcción de las tarjetas (I):**
 - Identificar las clases y validarlas “jugando” (brainstorming)
 - Identificar responsabilidades y asignárselas a las clases
 - En este proceso iterativo es usual que una responsabilidad migre de un objeto a otro y aparezcan nuevos objetos
 - Al asignar responsabilidades es conveniente:
 - Distribuir la inteligencia del sistema de forma equilibrada
 - Determinar las responsabilidades de manera tan general como sea posible

María Sierra, Patricia López - ISI

p2.27



Tarjetas CRC

- **Ejemplo:**
 - Se trata de modelar un sistema de biblioteca con los siguientes requisitos:
 - La aplicación soportará las operaciones de una biblioteca que incluye la búsqueda y préstamo de material bibliográfico: libros, proyectos y revistas.
 - Los socios de la biblioteca podrán llevar a cabo consultas sobre el material bibliográfico.
 - Cada socio puede tener prestado un número máximo de elementos.
 - Cada tipo de material tiene diferentes periodos de préstamo (por ejemplo, libros un máximo de 7 días, mientras que los proyectos no deben superar los 15 días).
 - Si se devuelve un elemento después de la fecha prevista, la biblioteca multará al socio con diferentes cantidades dependiendo también del tipo.

María Sierra, Patricia López - ISI

p2.28



Tarjetas CRC : Ejemplo

Sistema de biblioteca	
Super Classes:	Super Classes: Sistema de biblioteca
Sub Classes:	Sub Classes: Sistema de biblioteca
Descripción:	Descripción: guarda información del sistema biblioteca
Atributos:	
Responsables:	
Nombre	Descripción
Responsables:	Colaborador
Nombre	Material
Responsables:	Personal
Nombre	horquilla_material
Responsables:	Biblioteca
Nombre	visualiza_informacion
Responsables:	obtener_informacion

Material	
Super Classes:	Super Classes: Material
Sub Classes:	Sub Classes: Sistema de biblioteca
Descripción:	Descripción: guarda información del material del S. biblioteca
Atributos:	
Responsables:	
Nombre	Descripción
Responsables:	Colaborador
Nombre	Localizador
Responsables:	Código de localización
Nombre	consulta_datos
Responsables:	personal
Nombre	devuelve
Responsables:	devolver_fecha_devolucion
Nombre	cancelar_multa
Responsables:	cancelar_multa

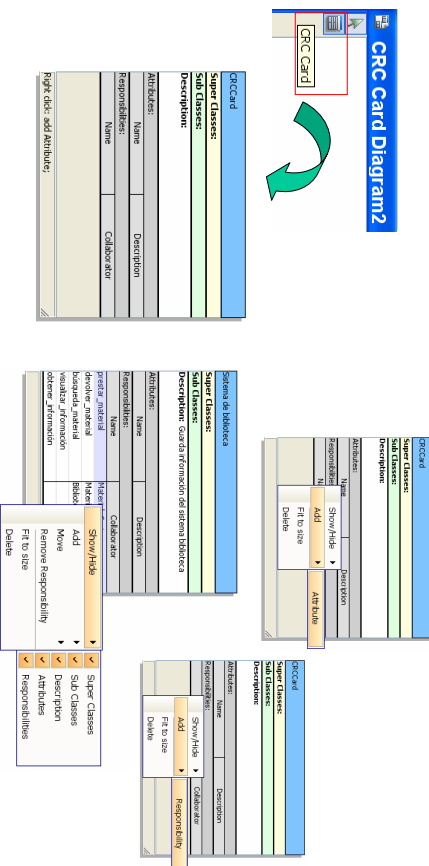
Sistema de biblioteca	
Super Classes:	Super Classes: Sistema de biblioteca
Sub Classes:	Sub Classes: Sistema de biblioteca
Descripción:	Descripción: guarda información del sistema biblioteca
Atributos:	
Responsables:	
Nombre	Descripción
Responsables:	Colaborador
Nombre	visualiza_informacion
Responsables:	obtener_informacion

María Sierra, Patricia López



Tarjetas CRC en VP

2) Creación de una Tarjeta CRC

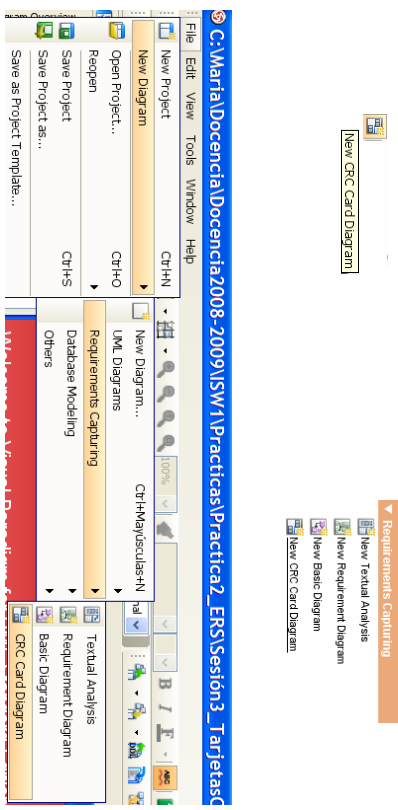


María Sierra, Patricia López - ISI



Tarjetas CRC en VP

1) Crear un Diagrama de Tarjetas CRC



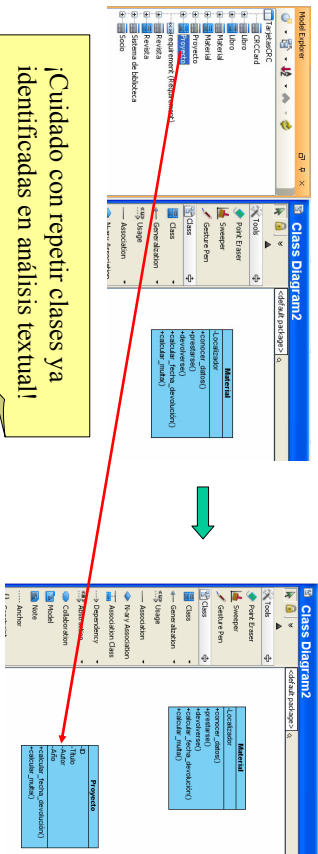
María Sierra, Patricia López - ISI



Tarjetas CRC en VP

3) Crear clases del modelo a partir de las Tarjetas CRC identificadas

- Vista: Modelo
- Seleccionar Elementos de tipo:
- Arrastrarlos hasta su diagrama correspondiente



María Sierra, Patricia López - ISI