



INGENIERÍA DEL SOFTWARE I

Práctica 2, Sesión 1

Especificación de Requisitos: Análisis Textual

Univ. Cantabria – Fac. de Ciencias
María Sierra

Ref: Aurora Vizcaíno, Felix Óscar García, Ismael Caballero



Nociones de UML para Requisitos

¿Qué es UML? (Lenguaje Unificado de Modelado)

- UML es un lenguaje de modelado estándar que sirve para visualizar, especificar, construir y documentar un sistema software.
 - Lenguaje Modelado = Notación + Reglas (Sintácticas, Semánticas) para modelar (representar, comprender) un sistema (Representación conceptual y física)
 - Planos del Software
- UML es apropiado para modelar diversos sistemas
 - Desde SI en empresas hasta aplicaciones distribuidas basadas en la Web, sistemas empujados de tiempo real muy exigentes, etc.
- UML puede ser utilizado por cualquier metodología OO
- Cubre las diferentes vistas de la arquitectura de un sistema mientras evoluciona a través del ciclo de vida del desarrollo de software
 - Vistas Software (estáticas, dinámicas, etc..), son como los distintos planos de una casa (fontanería, electricidad, etc..)





Nociones de UML para Requisitos: Casos de Uso

- Capturar los Requisitos que aportan valor añadido
 - Importancia a 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 con todas las formas posibles de usar un Sistema desde la perspectiva de sus usuarios

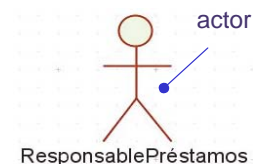


Nociones de UML para Requisitos: Casos de Uso

- **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 secuencia representa la interacción de los elementos externos al sistema (actores) con el sistema
- Variantes:
 - **Generalización**: el caso hijo hereda el comportamiento del padre; el hijo puede añadir o redefinir el comportamiento del padre.
 - **Inclusión**: un caso base incorpora explícitamente el comportamiento de otro caso en el lugar especificado en el caso base.
 - **Extensión**: significa que un caso base incorpora implícitamente el comportamiento de otro caso en el lugar especificado indirectamente por el caso que extiende al base.

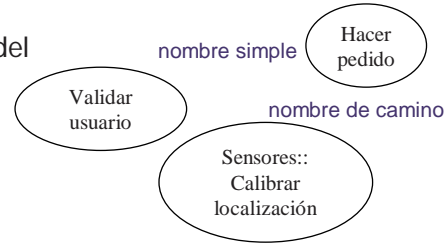




Nociones de UML para Requisitos: Casos de Uso

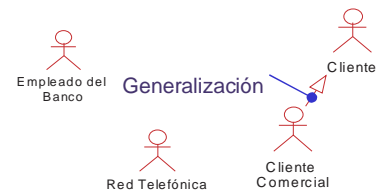
- Nombre:

- Lo distingue de otros, puede incluir el nombre del paquete en que se encuentra.
- Expresiones verbales que describen algún comportamiento del sistema



- Actor:

- Roles que los usuarios de los casos de uso (persona, dispositivo HW u otro Sistema) juegan al interactuar con éstos.
- Se pueden conectar a los casos de uso a través de asociaciones. (el actor y el caso de uso se comunican entre sí, y cada uno puede enviar y recibir mensajes).



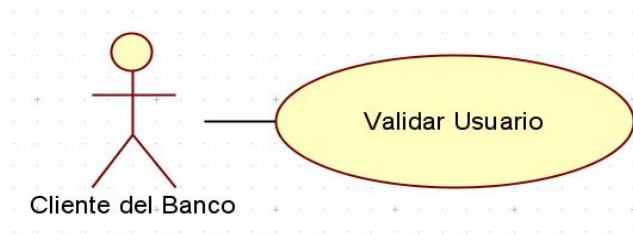
- Flujo de Eventos:

- Describe el comportamiento de los casos de uso.
 - Cómo y cuándo empieza y acaba el caso de uso.
 - Cuándo interactúa con los actores y qué objetos se intercambian.
 - El flujo básico y los flujos alternativos del comportamiento.
 - Es una abstracción de algo que puede hacer el objeto.



Nociones de UML para Requisitos: Casos de Uso

- Ejemplo: Caso de Uso **Validar Usuario** en un Cajero Automático:



Flujo de Eventos Principal:

- El caso de uso comienza cuando el **Sistema** pide al **Cliente** un número de identificación personal (PIN)
- El **Cliente** introduce el PIN a través del teclado y acepta la entrada pulsando la tecla Enter
- El **Sistema** comprueba si el PIN es válido
- El **Sistema** acepta la entrada y así finaliza el caso de uso



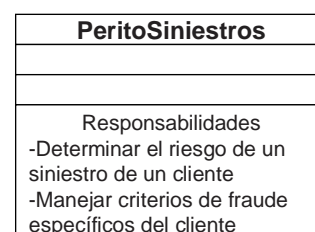
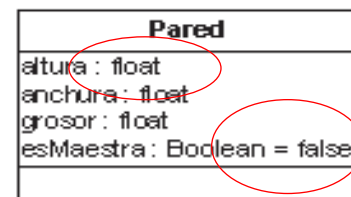
Nociones de UML para Requisitos: Clases

- **Modelado:** Identificación de las "cosas" importantes desde una visión particular.
 - Constituyen el vocabulario del sistema que se modela: generalmente el vocabulario del sistema lo componen las clases
 - Cada una de ellas tiene ciertas propiedades y un comportamiento.
- Existen **relaciones** entre esas cosas. En UML estas "cosas" se modelan como clases → Clase: Abstracción
- En el **Mundo Real** hay clases: sencillo en OO al unir el mundo real con el mundo de la solución tecnológica. 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.





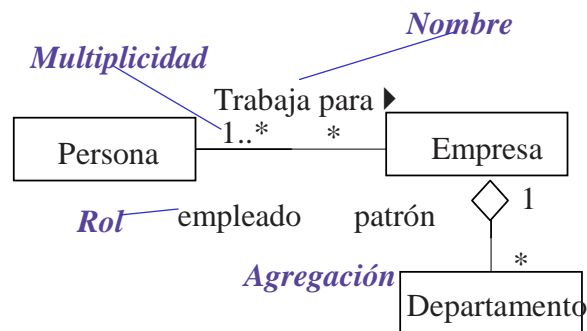
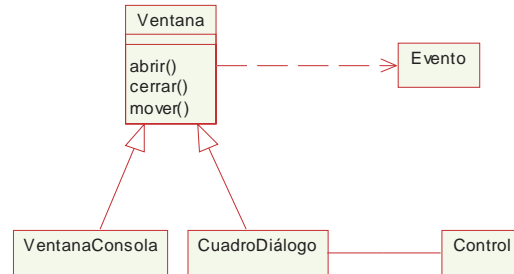
Nociones de UML para Requisitos: Clases

- **Relaciones:** Forma en que cada una de las clases del vocabulario del sistema tienen que ver unas con otras

- Dependencias
- Generalización
- Asociación

- **Adornos de las Asociaciones:**

- Nombre (con dirección, opcional)
- Rol
- Multiplicidad (varios tipos)
- Agregación



Análisis Textual en VP

- **Análisis Textual:**

- Técnica para la captura y análisis de requisitos
- En Visual Paradigm permite identificar los siguientes elementos de modelado candidatos:
 - Casos de Uso
 - Actores
 - Clases

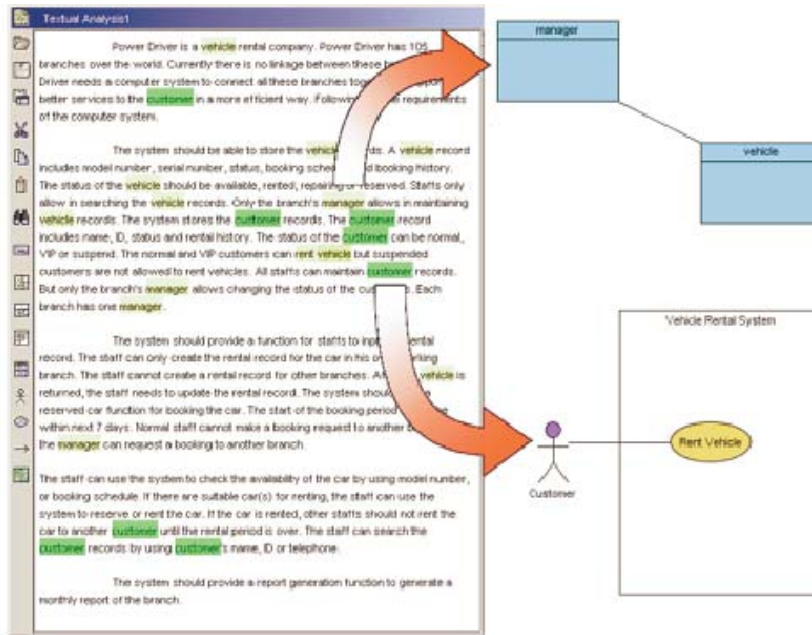
- **Tutorial interactivo en**

<http://www.visual-paradigm.com/product/vpuml/demos/requirements/textualanalisys.jsp>



Análisis Textual en VP

• Análisis Textual:

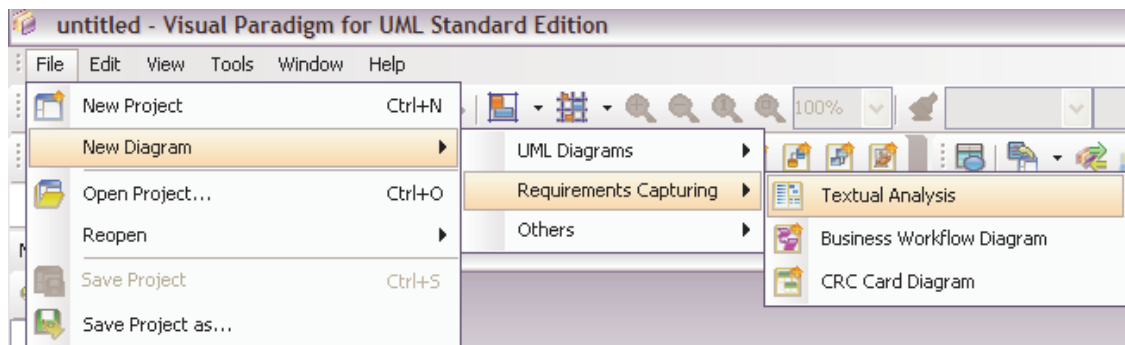


Análisis Textual en VP

1) Crear Modelo de Análisis Textual



- Requirements Capturing
 - Create Textual Analysis
 - Create Business Workflow
 - Create CRC Card Diagram

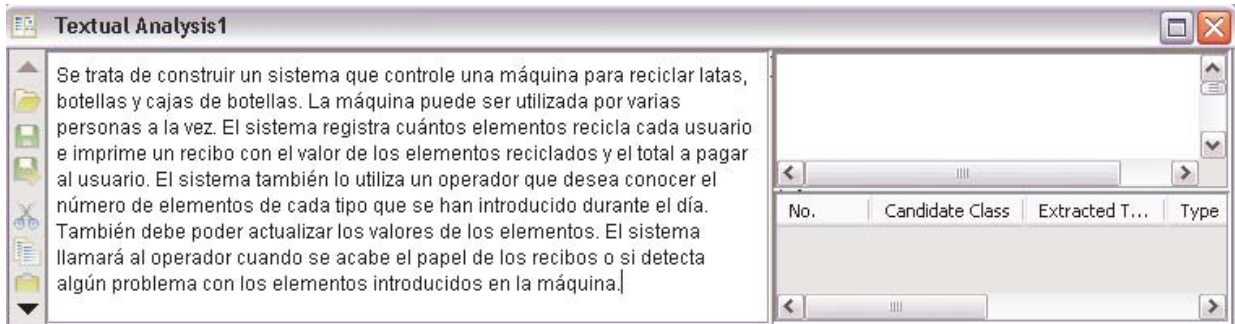




Análisis Textual en VP

2) Editar la Especificación Textual del Problema

- Ejemplo: Máquina de Reciclaje



Análisis Textual en VP

Barra de Herramientas

Icono	Notación	Icono	Notación
	Importar Archivo de Texto		Mostrar todas las Vistas
	Guardar archive texto		Ocultar el Área de Texto
	Guardar como		Mostrar solo tabla de elementos candidatos
	Cortar		Mostrar/Ocultar Clases Candidatas
	Copiar		Mostrar/Ocultar Actores Candidatos
	Pegar		Mostrar/Ocultar Casos de Uso Candidatos
	Buscar		Mostrar/Ocultar Flujos de Trabajo Candidatos
	Añadir Elemento (clase, caso uso, actor) Candidato		Mostrar/Ocultar Elementos no especificados candidatos



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.

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.

No.	Candidate Class	Extracted T...	Type
1	usuario	usuario	Act



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.

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.

No.	Candidate Class	Extracted T...	Type
1	usuario	usuario	Act
2	reciclar	reciclar	Us



Análisis Textual en VP

4) Completar la información de los elementos candidatos

The screenshot shows the 'Textual Analysis 1' interface. On the left, a text document is open, with several words highlighted in green: 'usuario', 'operador', 'reciclar', 'botellas', 'cajas', 'latas', 'conocer el número', and 'actualizar los valores'. The main workspace displays these elements as icons: 'operador(2)', 'reciclar(1)', 'botellas(2)', 'cajas(1)', 'latas(1)', 'conocer el número...', 'usuario(2)', and 'actualizar los valores...'. Below the workspace is a table with the following data:

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



Análisis Textual en VP

5) Crear modelos a partir de los elementos candidatos

The screenshot shows the 'Textual Analysis 1' interface with the same candidate elements as in the previous image. Three context menus are open over the elements 'conocer el número...', 'usuario(2)', and 'latas(1)'. Each menu has 'Create Use Case Model', 'Create Actor Model', and 'Create Class Model' options highlighted with red boxes. The table below shows the updated state of the candidate classes:

No.	Candidate Class	Extracted T...	Type	Class Description	Occurrer
3	operador	operador	Actor		
4	acabe el papel de	acabe el papel de	Use Case		
5	problema con los	problema con los	Use Case		
6	latas	latas	Class	tipo de elemento reciclable	
7	botellas	botellas	Class	tipo de elemento reciclable	
8	cajas	cajas	Class	tipo de elemento reciclable	
9	conocer el número	conocer el número	Use Case		
10	actualizar los valores	actualizar los valores	Use Case		



Análisis Textual en VP

5) Crear modelos a partir de los elementos candidatos

No.	Candidate Class	Extracted T...	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	reciclar	reciclar	Generated Model		
3	operador	operador	Generated Model	Se encarga del mantenimiento de la máquina	
4	acabe el papel de	acabe el papel de	Generated Model		
5	problema con los el	problema 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	Generated Model		
10	actualizar los valore	actualizar los val	Generated Model		



Análisis Textual en VP

6) Crear modelos a partir de los elementos candidatos

- Utilizar los elementos en los modelos correspondientes (cdu, clases, ..)
 - Vista: Model
 - Seleccionar Elementos de tipo:
 - Arrastrarlos hasta su diagrama correspondiente

