

Introducción a Requerimientos y Modelo de Negocios

Primer Semestre Año 2017

Profesora: Jeannette Padilla Burgos

Diagramas Gráficos para el Modelamiento

► UML

Que es UML?

Lenguaje Unificado de Modelado: Unified Modeling Language

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

UML no puede compararse con la programación estructurada, pues UML significa Lenguaje Unificado de Modelado, no es programación, solo se diagrama la realidad de una utilización en un requerimiento.

Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, la programación orientada a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos.

Que es UML?

UML es uno de los lenguajes de modelado, más emocionantes en el mundo actual del desarrollo de sistemas.

Esto se debe a que permite a los creadores de sistemas, generar diseños que capturen sus ideas en una forma convencional y fácil de comprender para comunicarlas a otras personas.

La comunicación de la idea es de suma importancia.

Sin embargo al ser el desarrollo de sistemas informáticos una actividad humana, se pueden cometer errores en cualquier etapa del proceso, por ejemplo:

- ▶ **El analista pudo haber mal entendido al cliente**
- ▶ **El analista pudo haber mal entendido una problemática y por ende dio una solución inadecuada.**
- ▶ **De esta forma el analista puede producir documentos que el cliente no puede entender.**
- ▶ **El ciclo continúa porque tal vez los programadores, tampoco comprendan el documento, por ende desarrollaron un programa difícil de comprender y que no soluciona ninguna problemática del cliente.**

Porqué es necesario UML?

En los inicios de la computación, los programadores no realizaban análisis muy profundos sobre las problemáticas a resolver.

Hoy en día se necesita contar con un plan muy bien analizado, el cliente debe comprender lo que hará el equipo informático de desarrollo, además debe ser capaz de señalar cambios, si no se han captado claramente sus necesidades.

Por otra parte el desarrollo de un sistema informático es un esfuerzo orientado a equipos, por lo que:

- ▶ Cada uno de los miembros del equipo debe saber que lugar debe ocupar.
- ▶ Cual es el trabajo asignado y debe asumir la responsabilidad de ello.
- ▶ Cada miembro del equipo sabe que lugar toma su trabajo en el resultado final del proyecto.
- ▶ Cual es la solución general que se está dando al cliente.

Porqué es necesario UML?

Conforme aumenta la complejidad del mundo, aumenta también la complejidad de los sistemas informáticos. En ellos se encuentran diversas piezas de hardware y software que deben comunicarse a grandes distancias mediante una red. Misma red que esta vinculada a bases de datos que a su vez contienen enormes cantidades de información. **¿Cómo se puede manejar entonces tanta complejidad?**

La clave está en organizar el proceso de desarrollo del tal forma que los analistas, clientes, desarrolladores y otras personas involucradas en el desarrollo del sistema lo comprenden y para ello UML es una solución, procurando esta organización.

Un arquitecto no podrá crear una compleja estructura, sin antes diseñar un anteproyecto detallado; del mismo modo un informático, no podrá generar un complejo sistema informático, sin el diseño de una anteproyecto detallado.

Así como el arquitecto debe mostrar un anteproyecto a la persona que lo contrató para ajustar necesidades, del mismo modo nosotros como informáticos necesitamos mostrar a nuestros clientes el diseño detallado de la solución que queremos implementar para él.

Este diseño detallado no es otra cosa que el resultado del cuidadoso análisis de las necesidades del cliente.

Porqué es necesario UML?

Otro punto importante radica hoy en día en la **reducción del tiempo de desarrollo de un proyecto de software**.

Cuando los plazos se encuentran muy acotados, un diseño sólido permitirá un mayor control y estimación adecuada de los tiempos.

Las empresas actualmente, se fusionan, se separan, crean nuevas empresas, sucursales, etc. Por lo mismo el desarrollo de software para las empresas debe **contar con un sólido y documentado diseño que permita que los software se puedan modificar o adecuar a estos cambios**. Cuando el diseño es sólido, un cambio puede implementarse sin problemas.

La creación de diseños sólidos a traído consigo la creación de una notación de diseño, que los analistas, clientes y desarrolladores acepten como pauta, que sea entendido por todos, es así como nace UML.

Diagramas de UML

UML está compuesto por distintos elementos gráficos, que se combinan para formar diagramas. Debido a que UML es un Lenguaje, cuenta con reglas para combinar tales elementos.

La finalidad de lo diagramas es presentar diversas perspectivas de un sistema. **Es importante destacar que UML describe lo que supuestamente hará el sistema, pero dice cómo implementar dicho sistema.**

Los Diagramas más comunes de UML son:

- ▶ Diagrama de Clases
- ▶ Diagrama de Objetos
- ▶ Diagrama de Casos de Uso
- ▶ Diagrama de Estado
- ▶ Diagrama de Secuencias
- ▶ Diagrama de Actividades
- ▶ Diagrama de Colaboraciones
- ▶ Diagrama de Componentes
- ▶ Diagramas de Distribución

Diagrama de Clases

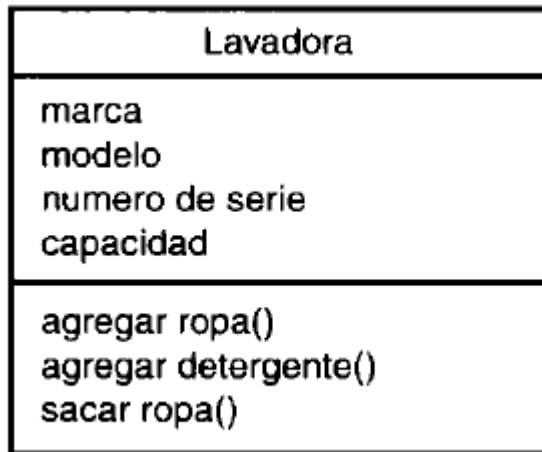
Muchas de las cosas que nos rodean, tienen **atributos** y además realizan algunas acciones, considerando que cada **acción** corresponde a un **conjunto de tareas**.

Por otro lado las cosas que nos rodean se albergan también en **categorías**, por ejemplo, automóviles, lavadoras, secadoras, computadores, etc. Tales categorías se denominan **clases**, una clase es una categoría o grupo de cosas que tienen atributos y acciones similares.

Por ejemplo: Cualquier cosa dentro de la **clase Lavadoras** tiene **atributos como Marca, Modelo, N°de Serie y Capacidad**, y dentro de las **acciones** que realizan estas cosas, **son agregar ropa, agregar detergente, agregar suavizante, activarse**, etc.

Un ejemplo de la notación UML que captura los atributos y acciones de una clase es la siguiente:

Diagrama de Clases



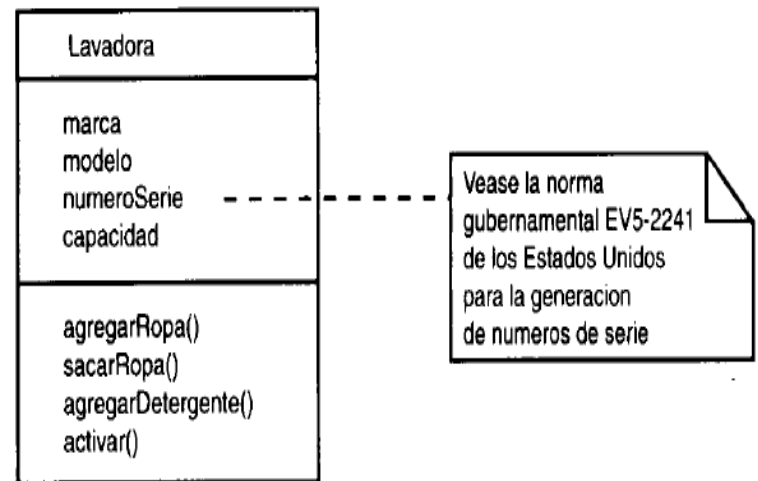
Por encima y debajo de los atributos, operaciones, responsabilidades y restricciones, puede agregar mayor información a una clase en la figura de NOTAS ADJUNTAS.

En el ejemplo se observa una nota que se refiere a una norma gubernamental que indica donde encontrar la manera de generar los números de serie para los objetos de la clase Lavadoras.

Una NOTA puede contener una imagen o un texto.

Un rectángulo es un símbolo que utiliza UML para representar una CLASE, que se divide en tres áreas:

- **Área superior = Nombre de la Clase**
- **Área Central = Atributos de la Clase**
- **Área Inferior = Acciones de la clase**



Cómo encontrar CLASES

Las clases son la terminología de un área del conocimiento, conforme converse con su cliente, analice su área del conocimiento y desarrolle sistemas de información que resuelvan los problemas de esa área, comprenderá la terminología y modelará los términos como clases de UML..

En sus conversaciones con sus clientes ponga especial atención a los **SUSTANTIVOS** que utilizan para describir a las **ENTIDADES**, estos sustantivos se convertirán en las clases del modelo.

También preste atención a los **VERBOS** que escuche, dado que constituirán las operaciones de las clases, los **ATRIBUTOS** surtirán como sustantivos relacionados con los nombres de la clase.

Una vez que tenga una lista básica de las clases pregunte a su cliente que es lo que cada **CLASE** hace dentro de su negocio, las respuestas le indicarán las responsabilidades de la Clase.

Las cosas se albergan en categorías y estas categorías se denominan clases, por ejemplo:

- **Automóviles**
- **Estudiantes**
- **Clientes**
- **Vendedores**
- **Entre Otros.**

Diagramas de Objetos

Un objeto es una instancia de una clase, una entidad que tiene valores específicos de los atributos y acciones.

Por ejemplo Una lavadora, puede ser:

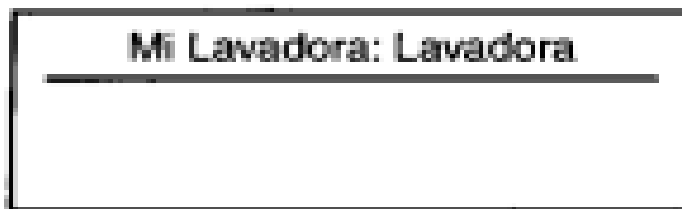
Marca: Mabe

Capacidad: 7,5 Kilos

Modelo: Mxfx

Nº de Serie: UF123678

UML representa de la siguiente forma el Objeto.



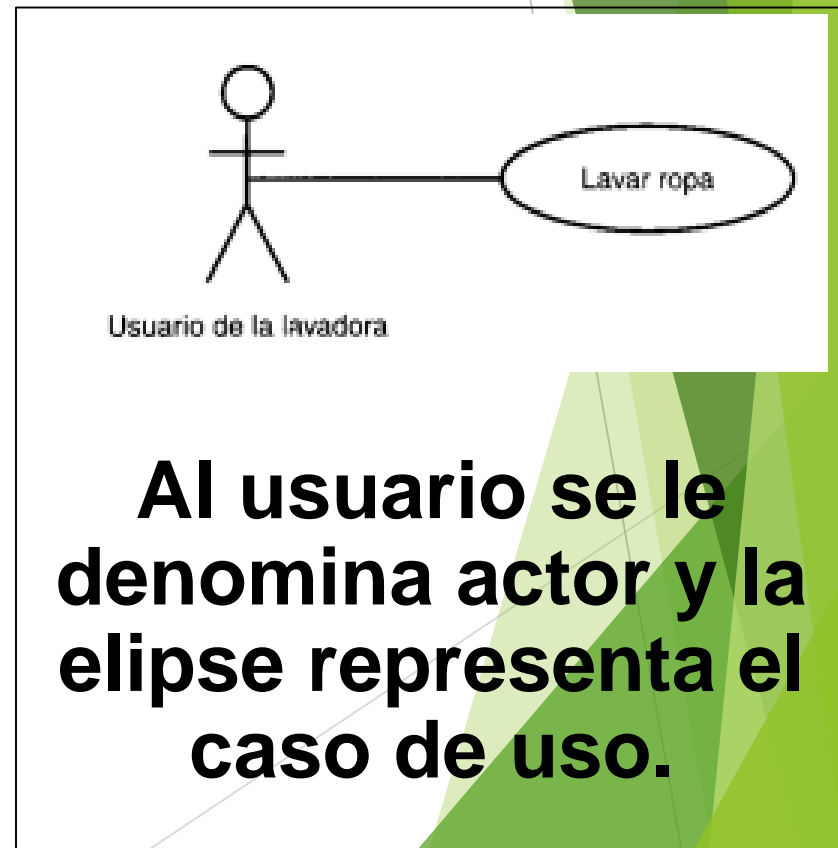
Un objeto se representa en un rectángulo, como en una clase, pero el nombre aparece subrayado, el nombre de la instancia específica se encuentra después de los dos puntos y el nombre de la clase a la derecha.

Diagramas de Casos de Uso

Un diagrama de casos de uso es la descripción de las acciones de un sistema desde el punto de vista del usuario. Para los desarrolladores del sistema esta es una herramienta valiosa, ya que es una técnica de aciertos y errores para obtener los requerimientos del sistema desde el punto de vista del usuario.

Esto es importante si la finalidad es crear un sistema que pueda ser usado por las personas en general y no sólo por los expertos informáticos.

Por ejemplo: se describe a continuación una acción de la lavadora, respecto de lo que usted desea que haga



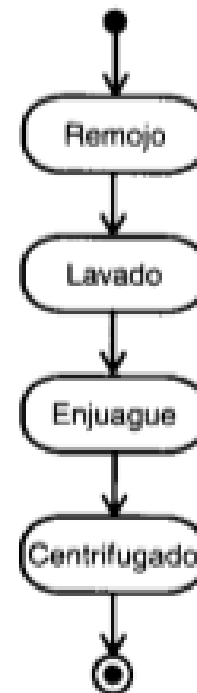
Diagramas de Estados

En cualquier momento un objeto se encuentra en un estado en particular, por ejemplo una persona puede ser; recién nacida, infante, adolescente, joven o adulta. Un ascensor se moverá hacia arriba, estará en reposo o se moverá hacia abajo.

Una lavadora podrá estar en la fase de remojo, lavado, enjuague, centrifugado, apagada.

El diagrama de estados muestra las transiciones de la lavadora de un estado a otro.

El símbolo que está en la parte superior muestra el estado inicial de la lavadora y el de la parte inferior el estado final.



Diagramas de Secuencia

Los diagramas de clases y los de objetos, representan información estática, no obstante en un sistema funcional los objetos interactúan entre sí y tales interacciones suceden con el tiempo. El diagrama de secuencia muestra la mecánica de la interacción con base en tiempos. el diagrama de secuencias muestra la forma en la que los objetos se comunican entre sí al pasar el tiempo

Continuando con el ejemplo de la lavadora, entre sus componentes se encuentran:

- Una manguera de agua
- Un tambor (donde se lava la ropa)
- Un drenaje.

Por supuesto, estos también son objetos, por que un objeto puede estar conformado por otros objetos.

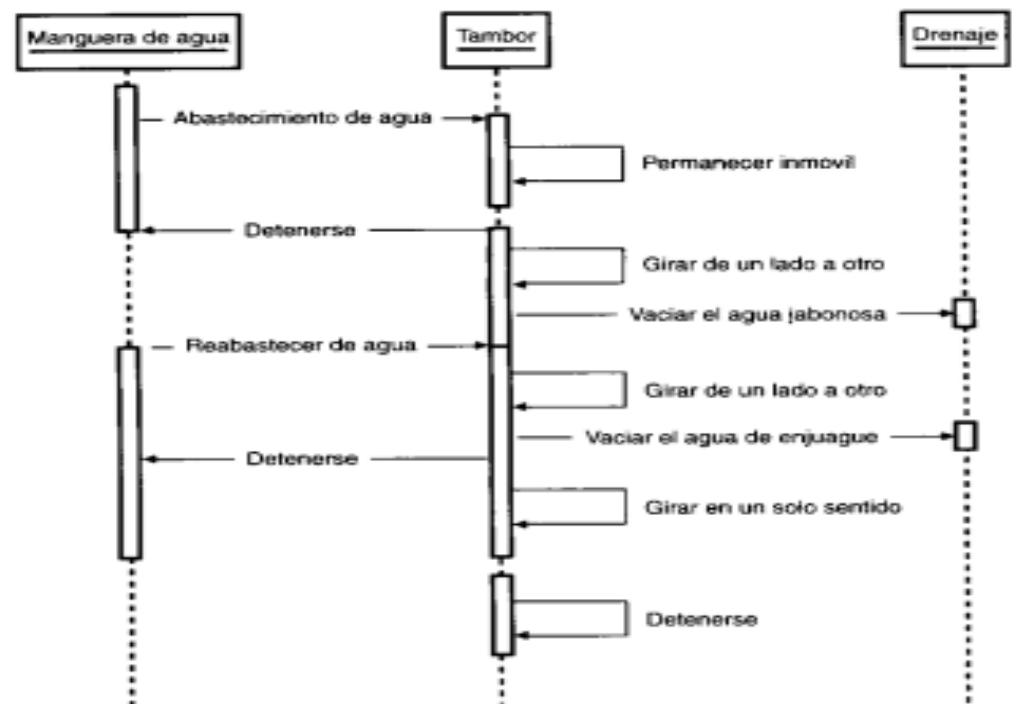
Que sucederá cuando invoque al caso de uso LAVAR ROPA, si damos por hecho que completo las operaciones “agregar ropa”, “agregar detergente”, “activar”, la secuencia sería más o menos así:

1. El agua empezará a llenar el tambor mediante una manguera.
2. El tambor permanecerá inactivo durante cinco minutos.
3. La manguera dejará de abastecer agua.
4. El tambor girará de un lado a otro durante quince minutos.
5. El agua jabonosa saldrá por el drenaje.
6. Comenzará nuevamente el abastecimiento de agua.
7. El tambor continuará girando.
8. El abastecimiento de agua se detendrá.
9. El agua del enjuague saldrá por el drenaje.
10. El tambor girará en una sola dirección y se incrementará su velocidad por cinco minutos.
11. El tambor dejará de girar y el proceso de lavado habrá finalizado.

Diagramas de Secuencia

La figura muestra el diagrama de secuencia, que captura las interacciones que se realizan a través del tiempo , entre el abastecimiento de agua, el tambor y el drenaje, .

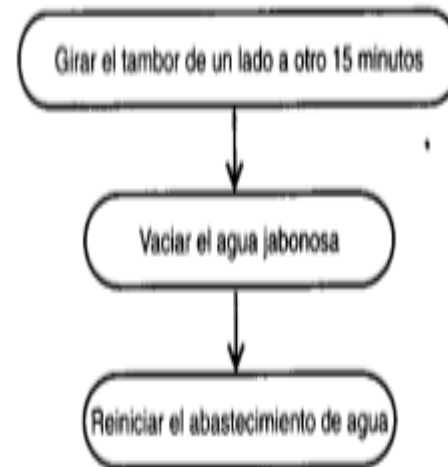
Podríamos caracterizar los pasos 1 y 2 como **estado de remojo**, 3 y 4 como **estados de lavado**, 5 y 7 como **estados de enjuague** y el 8 al 10 como el **estado de centrifugado**.



Diagramas de Actividades

Las actividades que ocurren dentro de un caso de uso, o dentro del comportamiento de un objeto, se dan normalmente en secuencia, como en los 11 pasos indicados en la sección anterior.

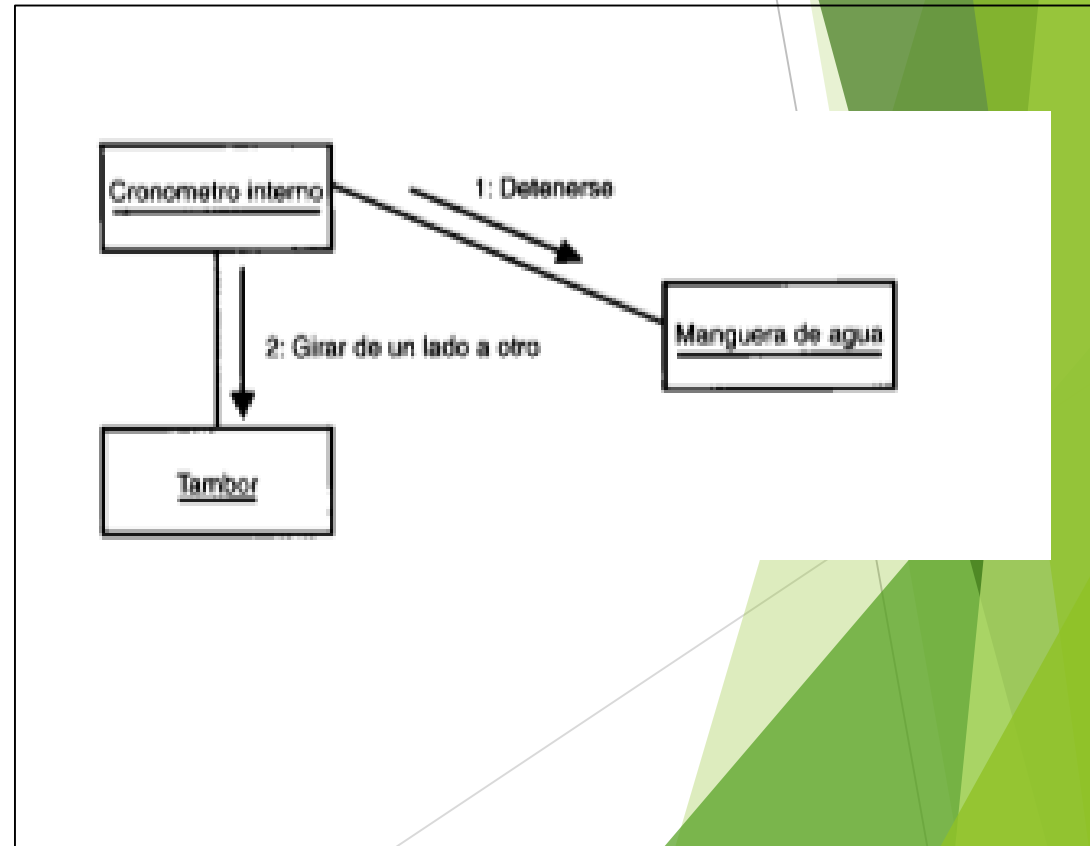
Ejemplo de Diagrama de Actividades que muestra los pasos 4 al 6 de la figura anterior.



Diagramas de Colaboraciones

Los elementos de un sistema trabajan en conjunto para cumplir con los objetivos del sistema, y un lenguaje de modelado deberá contar con una forma de representar esto. El diagrama de colaboraciones UML, diseñado con este fin se presenta a continuación.

Este ejemplo agrega un cronómetro interno, al conjunto de clases que constituyen a una lavadora, luego de cierto tiempo el cronómetro detendrá el flujo de agua y el tambor comenzará a girar de un lado para otro.

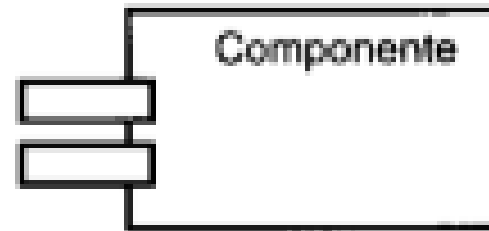


Diagramas de Componentes

Los diagramas de componentes están directamente relacionados con los sistemas informáticos, por lo anterior el modelo de desarrollo de software se realiza mediante componentes, lo que es particularmente importante en los procesos de desarrollo en equipo.

De esta forma UML representa un componente.

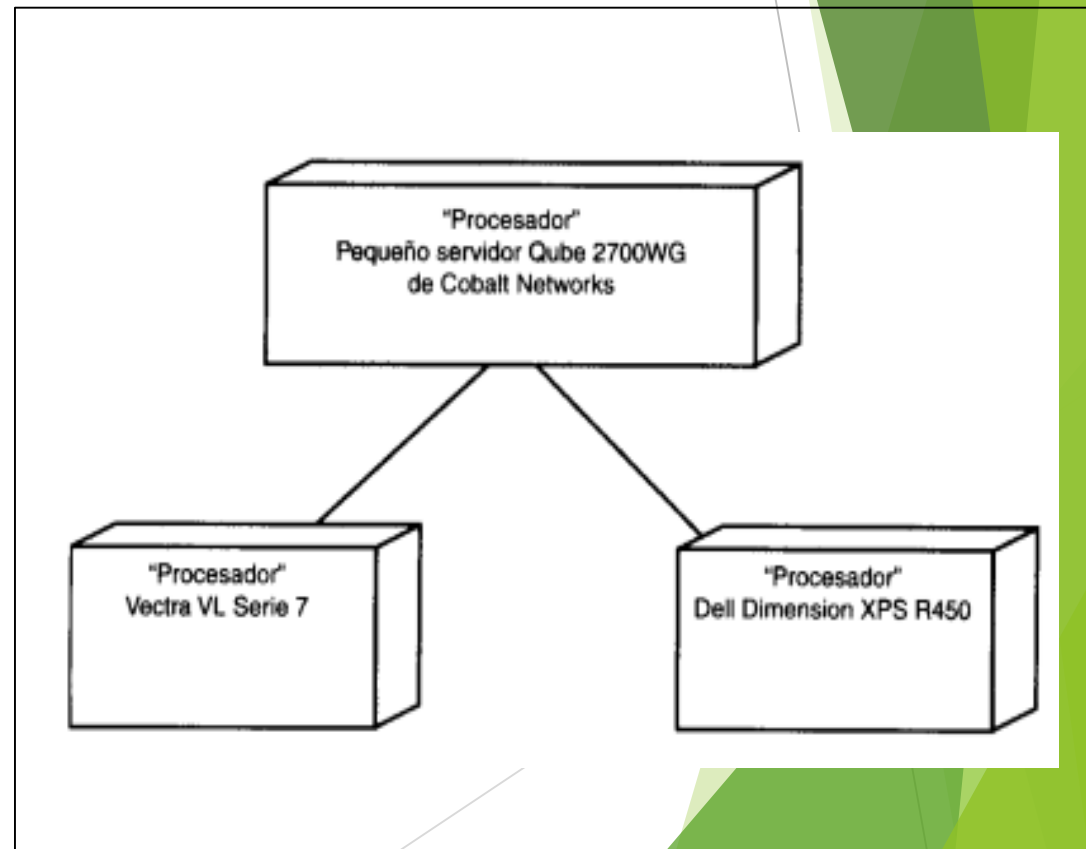
.



Diagramas de Distribución

Los diagramas de distribución muestra la arquitectura física de un sistema informático. Puede representar los equipos y dispositivos, mostrar sus interconexiones y el software que se encontrará en cada máquina.

Donde cada computador está representado por un cubo y las interacciones entre los cubos están representadas por líneas que conectan a los cubos.



Casos de Uso

Los casos de uso son una estructura que ayuda a los analistas a trabajar con los usuarios para determinar la forma en que se usará el sistema informático. Con una colección de casos de uso se puede hacer el bosquejo de un sistema informático en términos de lo que los usuarios intentan hacer con él.

Imagínese un caso de uso como una colección de situaciones respecto al uso de un sistema. Cada escenario describe una secuencia de eventos, cada secuencia es iniciada por una persona, por otro sistema, por hardware o por tiempo.

A las entidades que inician la secuencia se les denomina actores. El resultado de la secuencia debe ser algo utilizable ya sea por el actor que la inició o por otro actor.

Importancia de los Casos de Uso

Así como los diagramas de clase son una buena forma de estimular a que un cliente hable, respecto de un sistema desde su propio punto de vista, los casos de uso son una excelente herramienta para estimular para que los usuarios potenciales hablen.

Considerando que en la actualidad es muy importante involucrar al usuario desde las etapas iniciales del análisis y diseño de un sistema.

Esto aumenta la probabilidad de que el sistema sea de mejor provecho para las personas que lo utilizarán, en lugar de ser un manojo de expresiones informáticas incomprensibles e inmanejables para los usuarios.

Ejemplo de Casos de Uso

Escenario nº1: Comprar Bebida. Una persona quiere comprar una bebida en una máquina expendedora de bebidas.

La persona inserta la cantidad de dinero necesaria, selecciona la bebida y la máquina permite la salida de la bebida seleccionada. Esto si todo sale correctamente.

Escenario nº2: Comprar Bebida. Una persona quiere comprar una bebida en una máquina expendedora de bebidas.

La persona inserta la cantidad de dinero necesaria, selecciona la bebida y la máquina no tiene la bebida seleccionada por la persona, en este caso envía un mensaje que le indica la no existencia de la bebida, lo ideal es que la máquina permita al cliente elegir una bebida de otra marca o devolver el dinero del cliente

Escenario nº3: Comprar Bebida. Una persona quiere comprar una bebida en una máquina expendedora de bebidas.

La persona no ingresa la cantidad de dinero exacta en la máquina para comprar una bebida, el ingresa el dinero y selecciona la bebida. En este caso la máquina debe entregar la bebida y devolver el vuelto correspondiente al cliente.

Ejemplo de Casos de Uso

Consideremos ahora otros usuarios, por ejemplo para reabastecer la máquina de bebidas, para recolectar el dinero de la máquina.

Escenario nº1: Reabastecer Máquina. Después de algún intervalo, por ejemplo dos semanas, se debe reabastecer de bebidas la máquina, para ello el encargado de reabastecimiento debe abrir la máquina, quitar seguro con alguna llave o clave especial, abre la puerta y revisa cada compartimento de bebidas reabasteciendo todas aquellas que faltan, dejando la maquina hasta su máxima capacidad, luego cierra la máquina y la asegura con una llave o clave.

Escenario nº2: Recolectar dinero de la máquina. Después de algún intervalo, por ejemplo una semana, se debe recolectar el dinero de la máquina y revisar que quede dinero disponible para dar vueltos a los futuros clientes que paguen con una cantidad mayor de dinero, para ello el encargado de recolectar debe abrir la máquina, quitar seguro con alguna llave o clave especial, abre la puerta y revisa el compartimento del dinero, saca todo el dinero y luego coloca una cantidad previamente definida en dinero para los vueltos de dinero futuro, luego cierra la máquina y la asegura con una llave o clave.

Importante considerar que las acciones descritas en los casos de uso anteriores, corresponden a las acciones de los usuarios, en este caso, clientes, reponedores de bebida y recolectores de dinero.

Inclusión de los Casos de Uso

Si analiza los casos de uso reabastecer y recolectar, notará que existen algunos pasos en común.

Ambos comenzaban con abrir la máquina con alguna llave o clave y finalizaban con cerrar la máquina y asegurarla con una llave o clave.

¿Cree usted que se podría eliminar la duplicación de pasos de un caso de uso al otro?

Si podemos y a esto se le denomina Include o Inclusión

Inclusión de los Casos de Uso

La forma de hacerlo es:

Tomar cada secuencia de pasos en común y conformar un caso de uso adicional a partir de ellos.

Combinemos los pasos esenciales para abrir y cerrar la máquina:

Para abrir

- Quitar seguro
- Abrir la máquina

Este caso de uso se llamara “exhibir interior”

Para cerrar

- Cerrar la máquina
- Poner seguro

Este caso de uso se llamará “cubrir interior”

Con estos dos nuevos casos de uso, el caso de uso “**Reabastecer**”, iniciaría con el caso de uso “exhibir interior”, luego “reabastecer”, luego finalizaría con el caso de uso “cubrir interior”.

Por su parte el caso de uso “**Recolectar**”, iniciaría con “exhibir interior”, luego “Recolectar” y finalizaría con “cubrir interior”.

Extensión de los Casos de Uso

Es posible volver a utilizar un caso de uso de una forma distinta a la inclusión, en ocasiones crearemos un caso de uso agregándole algunos pasos a un caso de uso existente, por ejemplo el caso de uso Reabastecer Máquina.

Supongamos que antes de reabastecer, el reabastecedor debe revisar cuales son las bebidas que menos se venden, respecto de las que más se venden, en este caso, el reabastecedor en vez de sólo reponer las bebidas vendidas, cambia las bebidas menos vendidas por aquellas que más se venden, de esta forma además debe cambiar las etiquetas de las bebidas al frente de la máquina, indicando cuales son las bebidas disponibles en cada etiqueta.

Si se agregan estos pasos al caso de Uso “reabastecer”, tendremos un nuevo caso de uso llamado, “reabastecer de acuerdo a las ventas”.

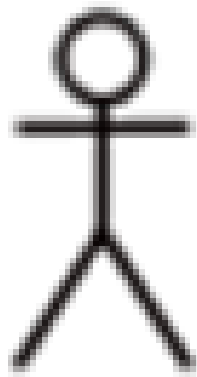
Este caso de uso es una extensión del caso de uso original.

Los Diagramas de Casos de Uso

Los casos de uso son muy poderosos, sobretodo cuando es posible representarlos a través de UML. Esta visualización permitirá mostrar los casos de uso a los usuarios para que ellos puedan aportar mayor información.

Una de las finalidades del proceso de análisis de un sistema es generar una colección de casos de uso. La idea es tener la posibilidad de catalogar y hacer referencia a esta colección, que sirve como el punto de vista de los usuarios acerca del sistema.

CASOS DE USO



Actor



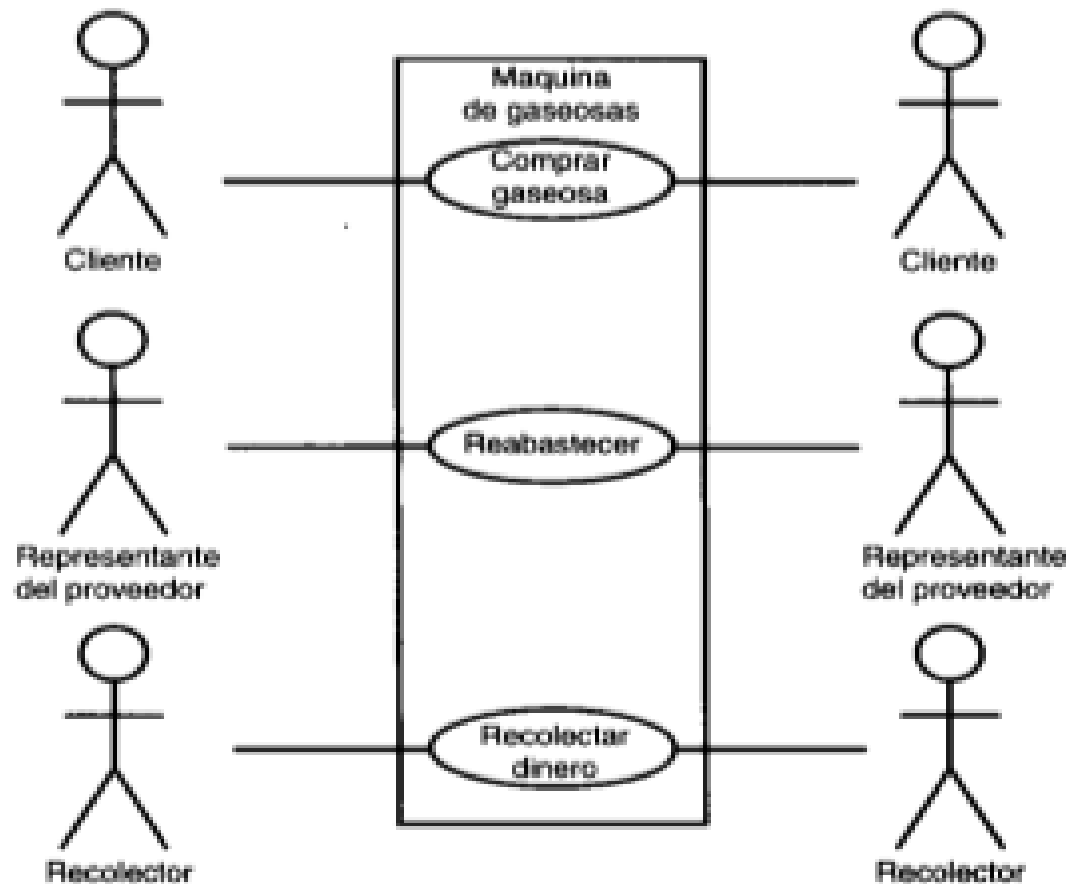
Representación de un Modelo Casos de Uso

Hay un actor que inicia un caso de uso y otro (que puede ser el mismo que inició), que recibe algo de valor de él. La representación gráfica es directa, una elipse representa un caso de uso y una figura agregada representa a un actor:



Ejemplo de Maquina expendedora de bebidas

En el ejemplo se realizaron los casos de uso, comprar bebidas, reabastecer máquina, recolectar dinero, los actores son; cliente, proveedor o reabastecedor y recolector.



Secuencia de pasos en los Escenarios

Cada caso de uso es una colección de escenarios y cada escenario es una secuencia de pasos, sin embargo tales pasos no aparecen en el diagrama.

La claridad de los diagramas es clave para que sean correctamente entendidos por el usuario.

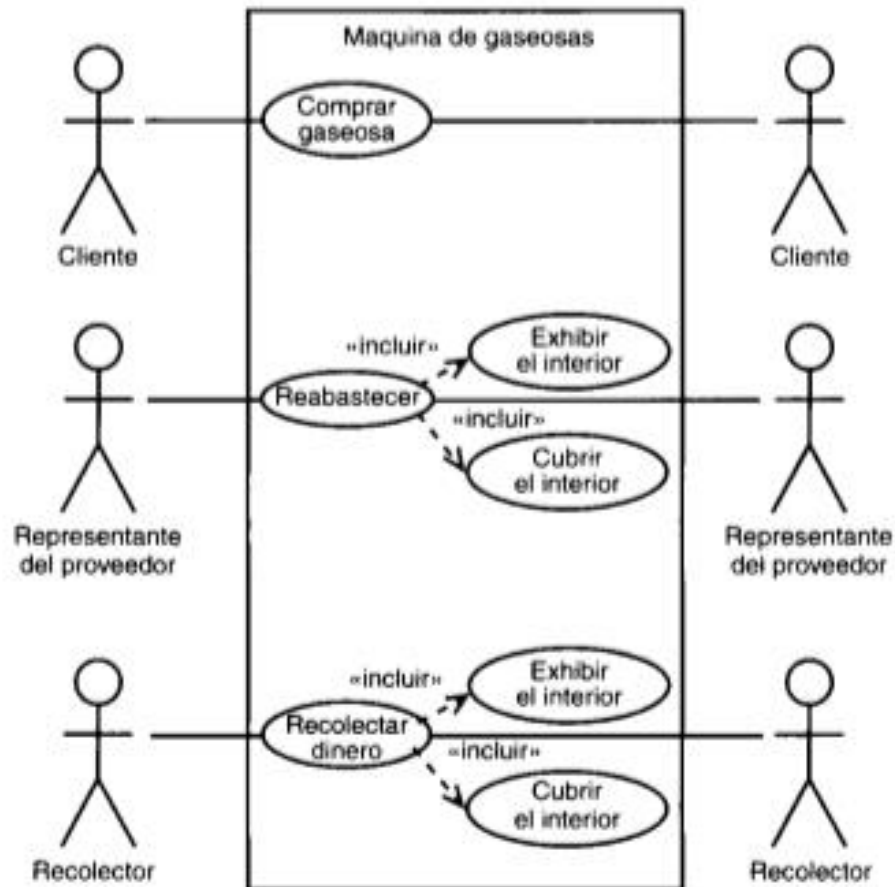
Por otra parte la documentación de los casos de uso describirá en forma detallada la secuencia de pasos, donde cada diagrama y cada escenario deben ser correctamente detallados.

La información mínima que debe contener es:

- **El actor que inicia el caso de uso**
- **Condiciones previas para el caso de uso**
- **Pasos en el escenario**
- **Condiciones posteriores cuando se finaliza el escenario**
- **El actor que se beneficia del caso de uso**

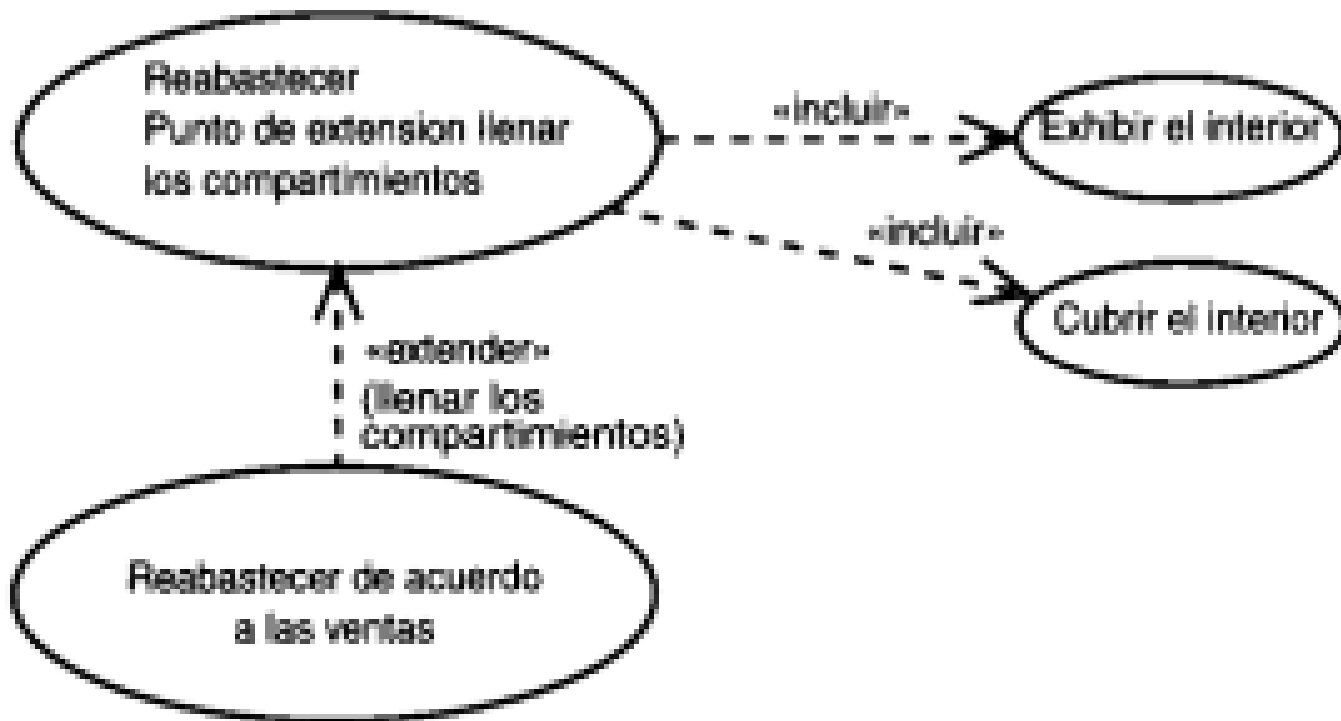
Representando la Inclusión

Recordando los casos de uso “Reabastecer” y Recolectar” del ejemplo anterior.



Representando la Extensión

Recordando los casos de uso “Reabastecer” y “Reabastecer de acuerdo a ventas” del ejemplo anterior.



Ejemplo:

Acciones, profesor y estudiantes, registro de notas.

