

**TECNOLÓGICO DE ESTUDIOS SUPERIORES DE VALLE DE  
BRAVO**

**INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**MATERIA:**

COMPLEMENTARIAS

**DIAGRAMAS UML**

**DOCENTE:**

AMBAR GONZALEZ GUADARRAMA

**ALUMNO:**

RICARDO LOPEZ ALBITER

**GRUPO: 401**

24 DE ABRIL DE 2017



## TEMARIO

Introducción a las metodologías

Definición de metodología

Metodología ~~root~~ rop

Características de la metodología rop

Diagrama de la metodología rop

Metodología de programación extrema

Metodología agile

Metodología V-model

Herramientas case

Diagrama UML

Que es un diagrama UML

Diagrama de caso de uso

Definición de caso de uso

Generalización de caso de uso

Inclusiones y extensiones

Narrativas

Diagrama de secuencia


Definición

Elementos

Marcos de iteración

Activación

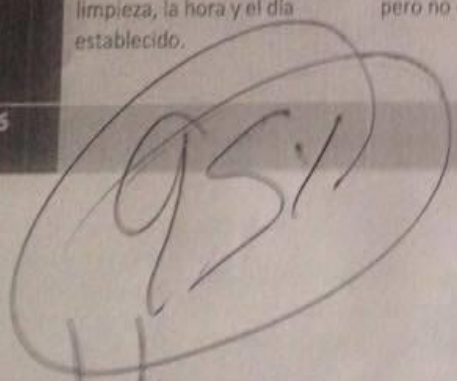
Procesos sincronos y asincronos

  06-03-12

## RUBRICA DE EVALUACIÓN DE COMPLEMENTARIAS

RUBRICA DE EVALUACIÓN DE COMPLEMENTARIAS

*Ricardo Lopez Albiter*

CRITERIOS DE EVALUACIÓN	EXCELENTE 100%	SATISFACTORIO 80%	NECESITA MEJORAR N.A
<b>PUNTUALIDAD Y LIMPIEZA</b>	Asiste puntualmente no tiene ninguna Falta y su higiene personal es buena.	Asiste puntualmente pero tiene 2 Faltas y su higiene personal es buena.	Falta mucho y no tiene cuidado con su higiene personal.
<b>PARTICIPACIÓN Y CONDUCTA</b>	Participa activamente en clases, acata órdenes y escucha atentamente.	Participa en las actividades; pero le cuesta trabajo seguir instrucciones.	No participa en las actividades y hay que estar constantemente llamándole la atención.
<b>TRABAJO COLABORATIVO</b>	Realiza las actividades perfectamente con todos sus compañeros ayudando a cumplir el objetivo de la actividad y realizando un buen trabajo en equipo.	Realiza las actividades perfectamente con todos sus compañeros pero le cuesta cumplir el objetivo de la actividad.	Le cuesta trabajo realizar las actividades en equipo por lo tanto no logra cumplir con el objetivo de la actividad.
<b>ENTREGA DE TRABAJOS</b>	Entrega los trabajos con limpieza, la hora y el día establecido.	Entrega los trabajos con limpieza pero no el día establecido.	Sus trabajos están maltratados y no los entrega ni el día ni la hora establecida.
<b>TOTAL DE CRÉDITOS LIBERADOS</b>			

*RSI*

Nota: para la evaluación de cada uno de los criterios anteriormente mencionados se evaluarán cada uno de ellos con diferentes rubricas personalizadas como son:

- 1.- Rubrica de Lista de Cotejo de Asistencia y participación 20%
- 2.- Rubrica de Exposición 20%
- 3.- Rubrica de Cuadro Comparativo 20%

4.- Rubrica de Investigación 20%

5.- Rubrica de Documento Final 20%

### RUBRICA DE DOCUMENTACION FINAL

NOMBRE DE LA MATERIA: RUBRICA DOCUMENTACIÓN FINAL  
GRUPO: 401 UNIDAD:  
NOMBRE DEL DOCENTE: ING. AMBAR GONZÁLEZ GUADARRAMA  
NOMBRE DE LOS ALUMNOS: Ricardo López Albiter

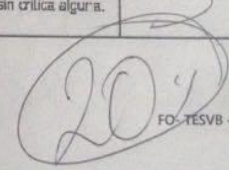
INDICADORES	DESEMPEÑO			
	Deficiente (5 o menos)	Satisfactorio (5-9%)	Bueno (10-14%)	Excelente (15-20%)
El reporte incluye todas las partes o elementos del tema o tarea asignada en forma directa y apropiada				/
Las ideas y argumentos presentados están bien fundamentados y referenciados en los recursos consultados				/
Incluye un apartado que resume y/o concluye en forma clara y precisa sobre los puntos principales abordados en la investigación				/
Presenta la bibliografía y recursos consultados				/
Se identifican el propósito e ideas principales en el escrito				/
El escrito es claro, enfocado e interesante				/
Las ideas se presentan de manera organizada, coherente y pueden seguirse con facilidad				/
Se utiliza un lenguaje apropiado con corrección sintáctica y gramatical, párrafos bien contruidos que facilitan su lectura y comprensión				/
Escrito bien editado y presentado				/
Total				100

*[Handwritten signature and circular stamp]*




# RUBRICA DE EXPOSICION

RUBRICA DE EXPOSICIÓN			
NOMBRE DE LA MATERIA:		GRUPO: 403	UNIDAD:
NOMBRE DEL DOCENTE: ING. AMBAR GONZÁLEZ GUADARRAMA			
NOMBRE DE LOS ALUMNOS: Ricardo Lopez Alcala			
EMA:			
	20%	15%	10%
Materia	El uso de la tecnología para mejorar la presentación es evidente, se muestra con buena calidad y estructura bastante legible.	El material consiste en lo mínimo necesario, sin demasiados adornos, solamente lo básico para una presentación adecuada.	El material muestra poca calidad de presentación, aparece incompleto o inconcluso, sin énfasis o consideraciones para hacerlo legible.
Domina	Se puede observar un buen dominio del tema por la seguridad con la que se habla al respecto de los distintos aspectos, tanto básicos como especializados.	Se expresa de manera adecuada respecto a los contenidos, pero se salta ciertos conceptos o ideas que no interpretan adecuadamente.	Duda al explicar los conceptos básicos y trata de completar información o la información, hablando redundante.
Expresión	Se tiene buen control de los movimientos y se usa buen volumen de voz, evitando repetir palabras y manteniendo información fluida.	Se ataca de vez en cuando, pero hay control en nerviosismo, manifestando algunas pausas durante la presentación.	Se muestra inseguridad en lo que se dice y se hacen pausas largas en las que se pretende leer lo que hay en las diapositivas.
Resultados	Se distingue claramente la fuente de la información recabada y el análisis de los datos obtenidos es conciso y comprensible.	Se basan en publicaciones previas pero hay intentos de explicación personal al respecto y adaptación de la información a lo investigado.	Los resultados aparecen como ya integrados en investigaciones previas y sin análisis personalizado de la información.
Conclusiones	Se muestra cierta originalidad en la escritura de los enunciados, basados principalmente en la información recabada e ideas originales.	Se pretende explicar a grandes rasgos algunas ideas basadas en lo leído, pero con intentos de adaptarlas a las ideas propias.	Se repite directamente lo que se leyó y simplemente se menciona lo que otras personas piensan, pero sin crítica alguna.



FO-TESVB-54-B v.1 31  
Reti

## EXPOSICION PROGRAMACION EXTREMA



Tecnológico de Estudios Superiores de Valle de Bravo

Ingeniería en Sistemas Computacionales


Equipo 5: Metodología de Programación Extrema.

Realizado por:  
Espinosa Guadarrama Jimena.  
López Albiter Ricardo.  
Medina León Jesús.  
Orozco Salmerón Alexander Ruvicel.  
Sánchez Rodríguez Valeria.  
Victoria Reyes Christian Briant

Docente: ISC Ambar González Guadarrama.

Semestre: 4°      Grupo: 401

Fecha: 13 Marzo de 2017



## Metodología de Programación Extrema

- Es programación extrema, o Extreme Programming (XP).
- Su autor principal es Kent Beck, eligió algunas características de otras metodologías y las relacionó de forma que cada una complementara a la otra.

## Definición:

XP es un conjunto de pasos de diversas metodologías, acopladas de manera que sean pasos flexibles a seguir utilizadas con el uso común, para realizar un desarrollo más agradable y sencillo.

Esta metodología tiene como base la simplicidad y como objetivo principal la satisfacción del cliente; para lograrlo se deben tomar en cuenta cuatro valores fundamentales:

## 1. Comunicación:

Debe haber comunicación constante con el cliente y dentro de todo el equipo de trabajo de esto dependerá que el desarrollo se lleve a cabo de una manera sencilla, entendible y que se entregue al cliente lo que necesita.

## 2. Simplicidad :

El diseño debe ser sencillo y amigable al usuario, el código debe ser simple y entendible, programando sólo lo necesario y lo que se utilizará.

## 3. Retroalimentación :

Es la comunicación constante entre el desarrollador y el usuario.

## 4. Coraje :

Es la valentía que se debe tener al modificar o eliminar el código que se realizó con tanto esfuerzo; el desarrollador debe saber cuando el código que desarrolló no es útil en el sistema y, por lo mismo, debe ser eliminado.

## Principios :

1. El principio de pruebas.
2. Proceso de planificación.
3. El cliente en el lugar.
4. Programación en parejas.
5. Integración continua.
6. Refactorización.
7. Entregas pequeñas.
8. Diseño simple.
9. Metáfora.
10. Propiedad colectiva del código.
11. Estándar de codificación.
12. La semana de 40 horas.

## Ventajas :

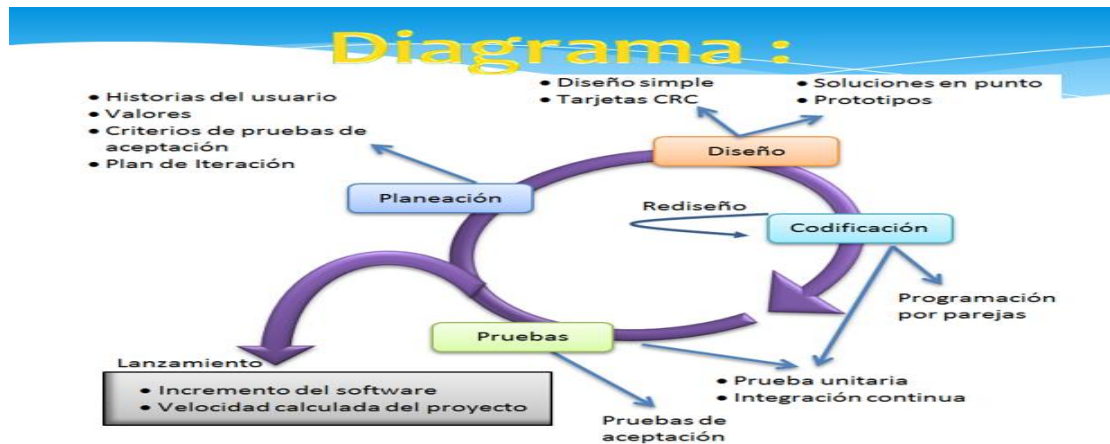
- Se adapta al desarrollo de sistemas pequeños y grandes.
- Optimiza el tiempo de desarrollo.
- Permite realizar el desarrollo del sistema en parejas para complementar los conocimientos; el código es sencillo y entendible.

## Desventajas :

No se tiene la definición del costo y el tiempo de desarrollo; el sistema va creciendo después de cada entrega al cliente y nadie puede decir que el cliente no querrá una función más; se necesita de la presencia constante del usuario, lo cual en la realidad es muy difícil de lograr.

Otra desventaja es la programación en parejas, algunos desarrolladores son celosos del código que escriben y no les es grato que alguien más modifique las funciones que realizó o que su código sea desechado por no cubrir el estándar.





## Bibliografía :

- Beck, K. A. (2004). Extreme Programming Explained. Addison Wesley.
- Martin, R., & Newkirk, J. (2002). La programación extrema en la práctica. Pearson Addison-Wesley.
- [http://www.uv.mx/universo/486/infgral/infgral\\_15.html](http://www.uv.mx/universo/486/infgral/infgral_15.html)



## PREGUNTAS DE EXPOSICION

### PROGRAMACIÓN EXTREMA

- 1- ¿Qué es la programación extrema?  
Metodología de desarrollo ágil.
- 2- ¿Cuántos principios básicos tiene?  
12 principios básicos
- 3- ¿Cuál es su principal objetivo?  
Aumentar la productividad del desarrollo software
- 4- ¿Cuál es el tiempo que tiene como máximo a semana?  
40 horas
- 5- En qué está basada esta metodología?  
En prueba y error.

Alex O.S.  
Valeria S.R.  
Christian V.R.  
Jimena E.G.  
Jesus Medina  
Ricardo



RUBRICA DE CUADRO COMPARATIVO

Rubricado Lopez Alcala

RUBRICA DE CUADRO COMPARATIVO

Aspecto	Excelente 20%	Muy bien 15%	Suficiente 10%	Deficiente 5%
Identifica las semejanzas y diferencias	Identifica de manera clara y precisa las semejanzas y diferencias entre los elementos comparados.	Identifica la mayor parte de las semejanzas y diferencias entre los elementos comparados.	Identifica varias de las semejanzas y diferencias entre los elementos comparados.	No identifica las semejanzas y diferencias de los elementos comparados.
Representación esquemática de la información	El organizador gráfico presenta los elementos centrales y sus relaciones en forma clara y precisa.	El organizador gráfico que construye representa los elementos con cierta claridad y precisión.	El organizador gráfico elaborado representa los elementos solicitados, aunque no es del todo claro y preciso.	El organizador gráfico no representa esquemáticamente los elementos a los que hace alusión el tema.
Ortografía, gramática y presentación.	Sin errores ortográficos o gramaticales.	Existen errores ortográficos y gramaticales mínimos (menos de 3).	Varios errores ortográficos y gramaticales (más de 3 pero menos de 5).	Error ortográfico y gramaticales múltiples (más de 5).

151

Metodología	Beneficios	Desventajas
	proyectos con complejidad mucho mayor tiene característica de que se adapta a los cambios que suceden.	Solamente funcionan para grandes proyectos los costos son altos
caso a caso	Es más simple de ser implementado son los mínimos la documentación se realiza en cada etapa	No se puede volver atrás si el diseño se realiza mal se puede complicar en la fase ejecución
Método	Utiliza diferentes métodos y técnicas cubre todas las bases o fases que se realicen en el proyecto.	Las alternativas que se presentan para evaluar un proyecto y solucionar un problema muchas veces no es lo más factible

RUBRICA DE INVESTIGACION

RUBRICA DE INVESTIGACIÓN

NOMBRE DE LA MATERIA:

GRUPO: 401

UNIDAD: 11

NOMBRE DEL DOCENTE: ING. AMBAR GONZÁLEZ GUADARRAMA

NOMBRE DE LOS ALUMNOS: Ricardo Lopez Albiter

INDICADORES	DESEMPEÑO			
	Deficiente (5 o menos)	Satisfactorio (5-9%)	Bueno (10-14%)	Excelente (15-20%)
El reporte incluye todas las partes o elementos del tema o tarea asignada en forma directa y apropiada				///
Las ideas y argumentos presentados están bien fundamentados y relacionados en los recursos consultados				///
Incluye un apartado que resume y/o concluye en forma clara y precisa sobre los puntos principales abordados en la investigación				///
Presenta la bibliografía y recursos consultados				///
Se identifican el propósito e ideas principales en el escrito				///
El escrito es claro, enfocado e interesante				///
Las ideas se presentan de manera organizada, coherente y pueden seguirse con facilidad				///
Se utiliza un lenguaje apropiado con corrección sintáctica y gramatical, párrafos bien contruidos que facilitan su lectura y comprensión				///
Escrito bien editado y presentado				///
Total				

20%

INVESTIGACION DIAGRAMAS UML

DEFINICION

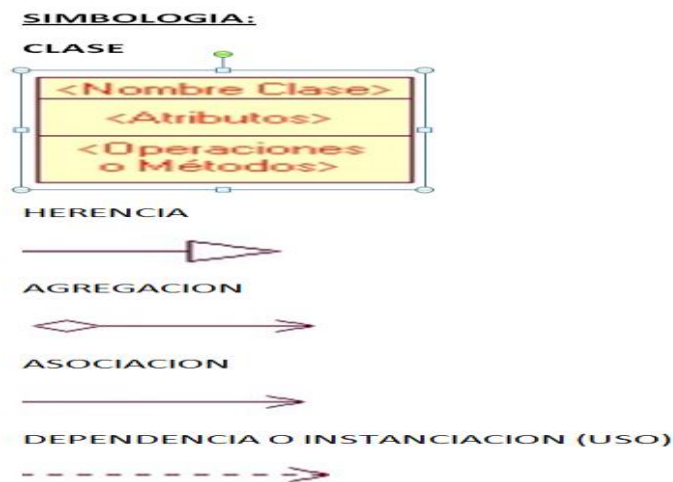
El UML está compuesto por diversos elementos gráficos que se combinan para conformar diagramas. Debido a que el UML es un lenguaje, cuenta con reglas para combinar tales elementos. La finalidad de los diagramas es presentar diversas perspectivas de un sistema, a las cuales se les conoce como modelo. Recordemos que un modelo es una representación simplificada de la realidad; el modelo UML

describe lo que supuestamente hará un sistema, pero no dice cómo implementar dicho sistema.

## Diagrama de clases

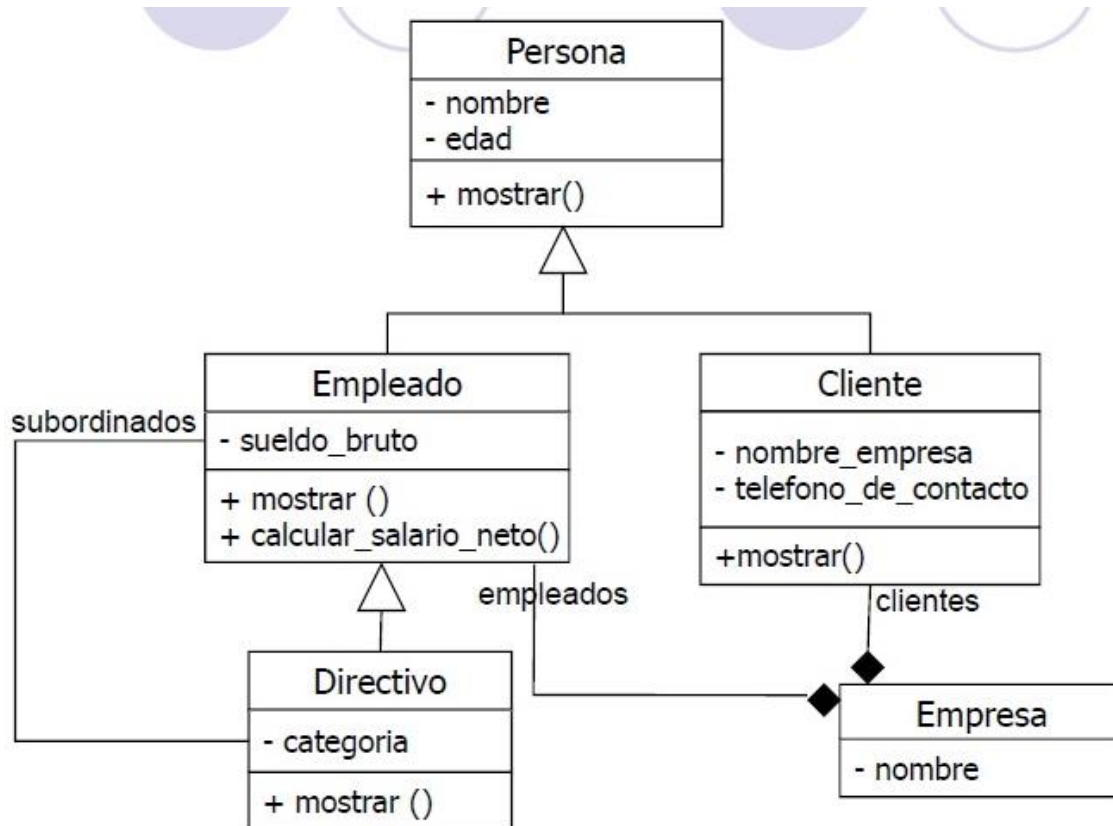
Los diagramas de clases describen la estructura estática de un sistema. Las cosas que existen y que nos rodean se agrupan naturalmente en categorías. Una clase es una categoría o grupo de cosas que tienen atributos y acciones similares.

Un rectángulo es el símbolo que representa a la clase, y se divide en tres áreas. Un diagrama de clases está formado por varios rectángulos de este tipo conectados por líneas que representan las asociaciones o maneras en que las clases se relacionan entre si.



## EJEMPLO

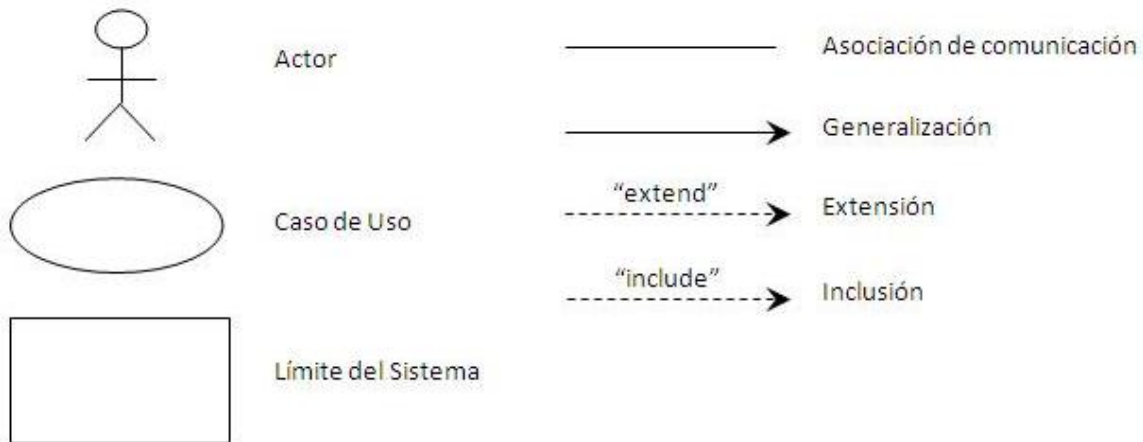




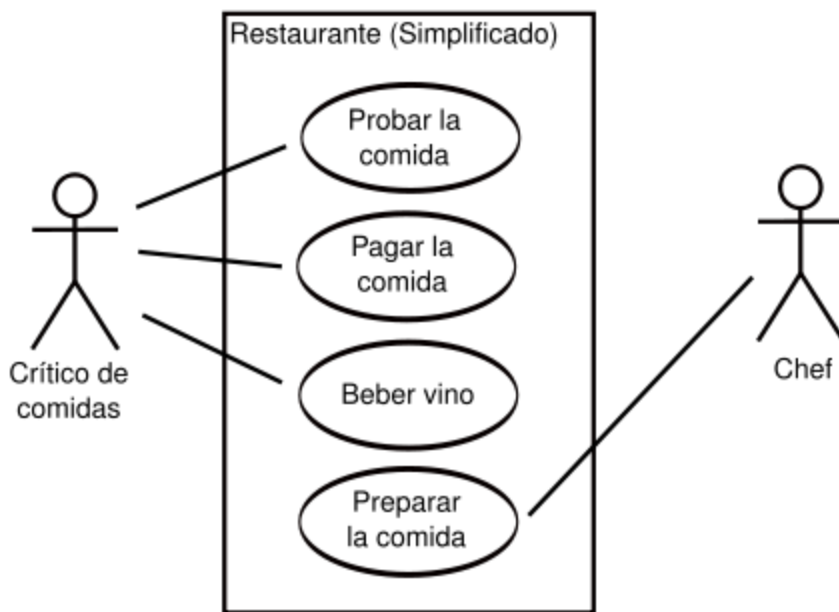
## DIAGRAMA DE CASOS DE USO

Un caso de uso es una descripción de las acciones de un sistema desde el punto de vista del usuario. Es una herramienta valiosa dado que es una técnica de aciertos y errores para obtener los requerimientos del sistema, justamente desde el punto de vista del usuario. Los diagramas de caso de uso modelan la funcionalidad del sistema usando actores y casos de uso. Los casos de uso son servicios o funciones provistas por el sistema para sus usuarios.

## SIMBOLOGIA



## EJEMPLO



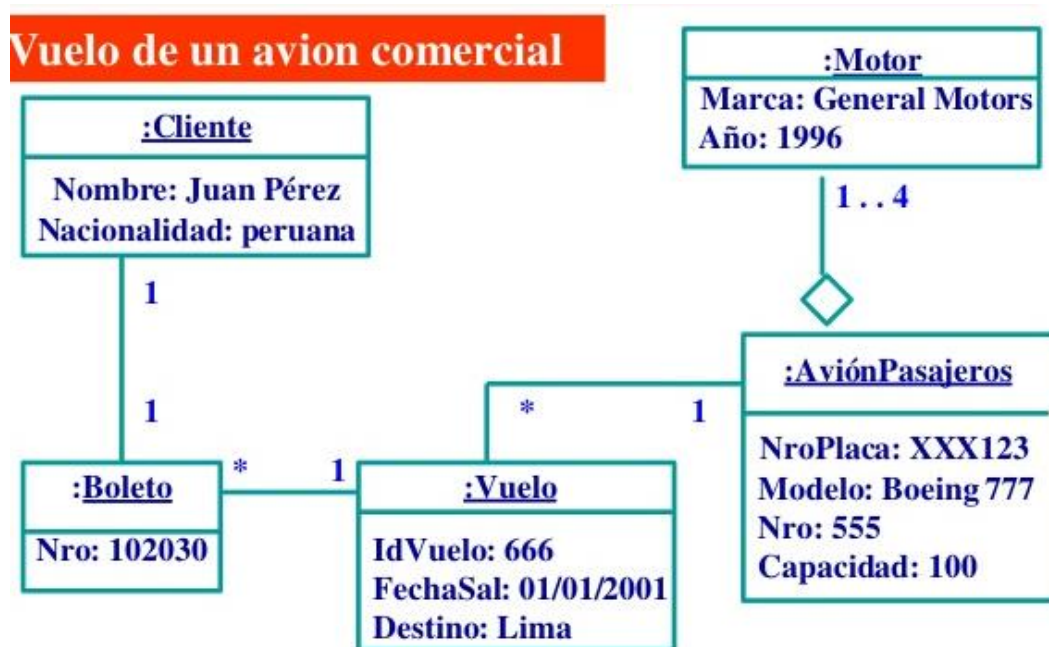
## DIAGRAMA DE OBJETOS

Pueden ayudar a explicar las clases y su herencia.

Objeto es una entidad discreta con límites bien definidos y con identidad, es una unidad atómica que encapsula estado y comportamiento. La encapsulación en un objeto permite una alta cohesión y un bajo acoplamiento. el Objeto es reconocido también como una instancia de la clase a la cual pertenece.

Simbología

Ejemplo



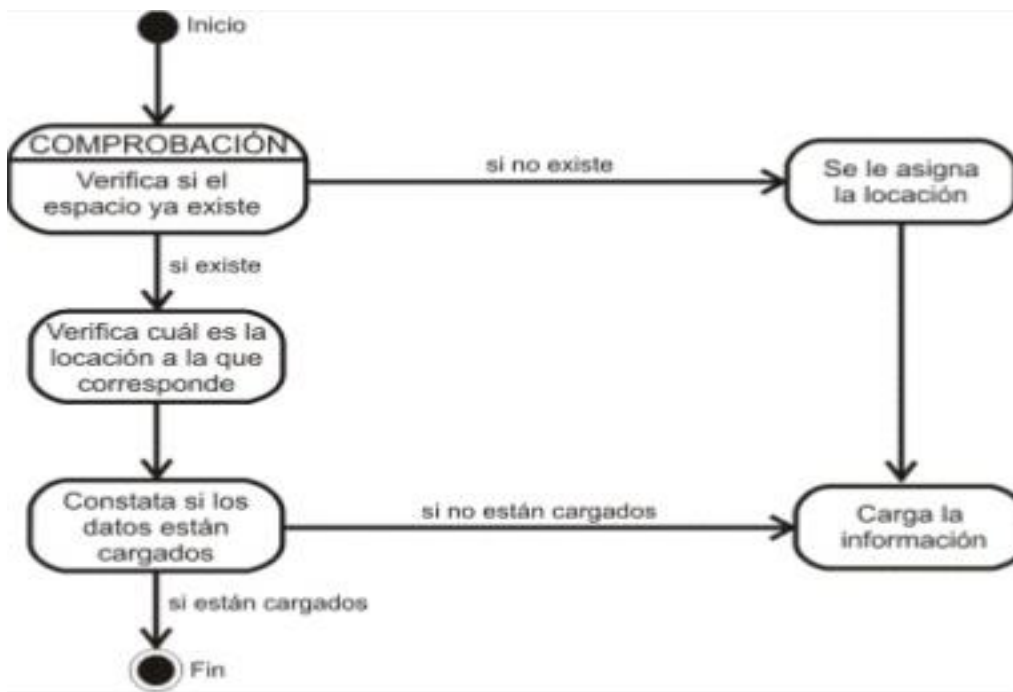
## DIAGRAMA DE ESTADOS

En cualquier momento, un objeto se encuentra en un estado particular, la luz está encendida o apagada, el auto en movimiento o detenido, la persona leyendo o cantando. El diagrama de estados UML captura esa pequeña realidad.

## SIMBOLOGIA



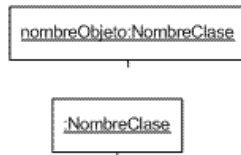
## EJEMPLO



## DIAGRAMA 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 secuencias UML muestra la mecánica de la interacción con base en tiempos.

## SIMBOLOGIA



Rectángulo de  
encabezado



Dstrucción de un  
objeto



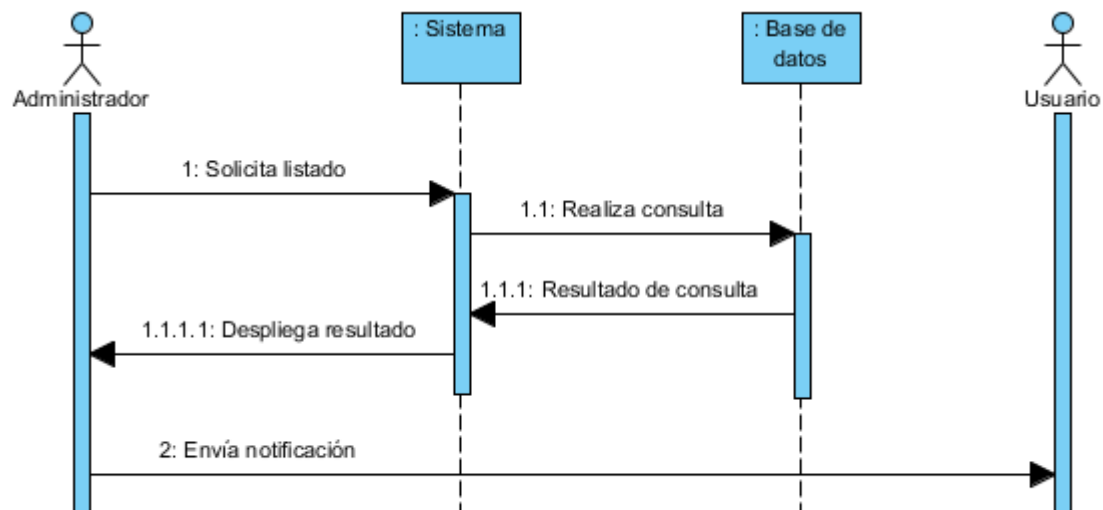
Tiempo de  
ejecución  
(Activación)



Línea de vida de  
un objeto

Flecha	Tipo de mensaje
	Simple
	Sincrónico
	Asincrónico
	Rechazado
	Time out

Ejemplo


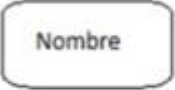









## DIAGRAMA DE ACTIVIDADES

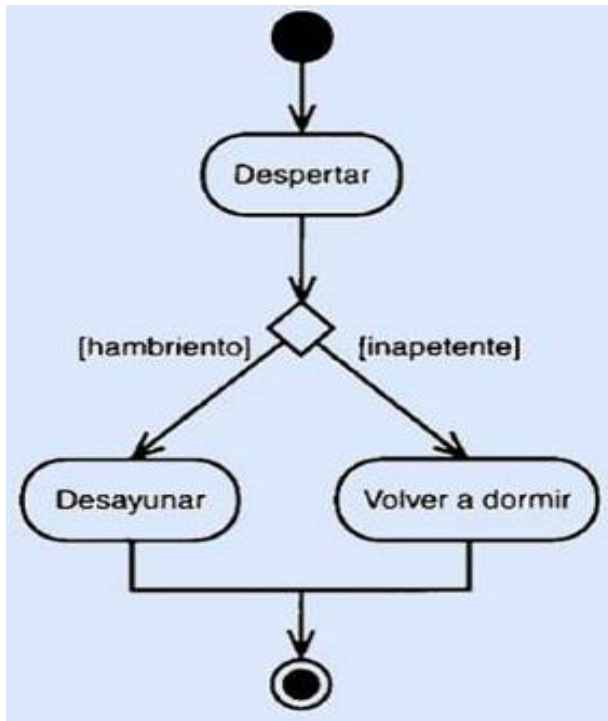
Un diagrama de actividades ilustra la naturaleza dinámica de un sistema mediante el modelado del flujo ocurrente de actividad en actividad. Una actividad representa una operación en alguna clase del sistema y que resulta en un cambio en el estado del sistema. Típicamente, los diagramas de actividad son utilizados para modelar el flujo de trabajo interno de una operación.

### SIMBOLOGIA

Símbolo	Nombre	Descripción
	Nodo inicial	Muestra el punto de partida del flujo de acciones.
	Acción	Representa una actividad o acción. El nombre generalmente comienza con un verbo.
	Flujo o transición	Muestra el orden de ejecución de las actividades.
	Nodo final	El final de todos los flujos de acciones en el diagrama.

SÍMBOLO	DESCRIPCION
	Indica la existencia de una acción o el estado de un objeto
	La flecha representa el movimiento de datos de un punto hacia otro, donde la punta señala el destino de los datos.
	Bifurcación de información, salida o flujo de datos combinada en varias direcciones.

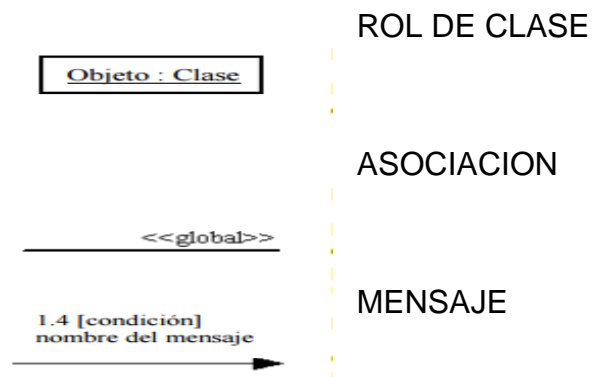
## EJEMPLO



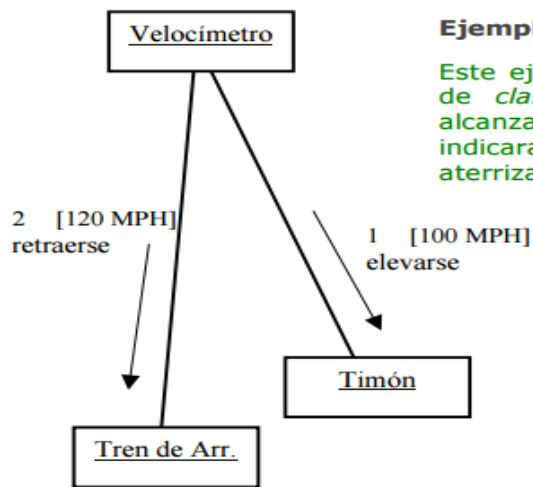
## DIAGRAMA DE COLABORACIONES

El diagrama de colaboraciones describe las interacciones entre los objetos en términos de mensajes secuenciados. Los diagramas de colaboración representan una combinación de información tomada de los diagramas de clases, de secuencias y de casos de uso, describiendo el comportamiento, tanto de la estructura estática, como de la estructura dinámica de un sistema.

## SIMBOLOGIA



## EJEMPLO



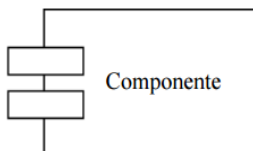
### Ejemplo de Diagrama de Colaboración

Este ejemplo agrega un velocímetro al conjunto de *clases* que constituyen a un "Avión". Al alcanzar una cierta velocidad el velocímetro indicará al timón que debe elevarse y al tren de aterrizaje que debe retraerse.

## DIAGRAMA DE COMPONENTES

Un diagrama de componentes describe la organización de los componentes físicos de un sistema.

## SIMBOLOGIA



COMPONENTE BLOQUE DE CONSTRUCCION FISICA



INTERFAZ DESCRIBE A UN GRUPO DETERMINADO DE

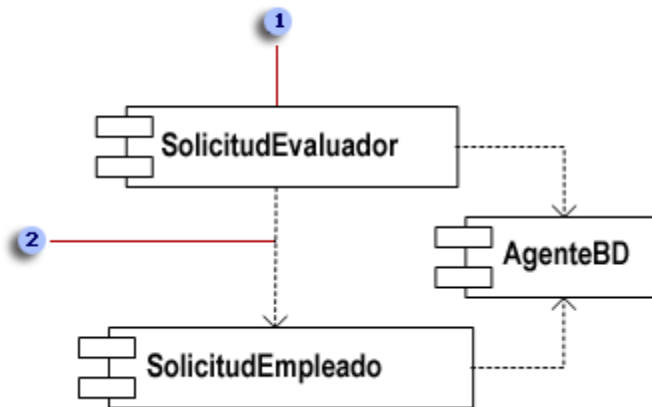
OPERACIONES



DEPENDENCIA

## EJEMPLO

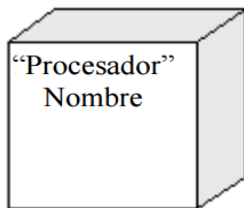
### Sistema de certificación de evaluadores



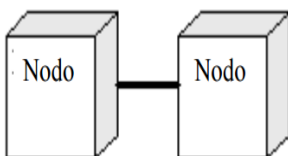
## DIAGRAMA DE DISTRIBUCION

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

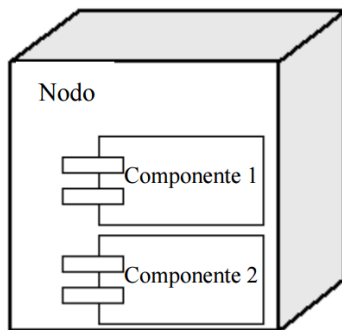
## SIMBOLOGIA



NODO RECURSO FISICO CAPAZ DE EJECUTAR COMPONENTES DE CODIGO

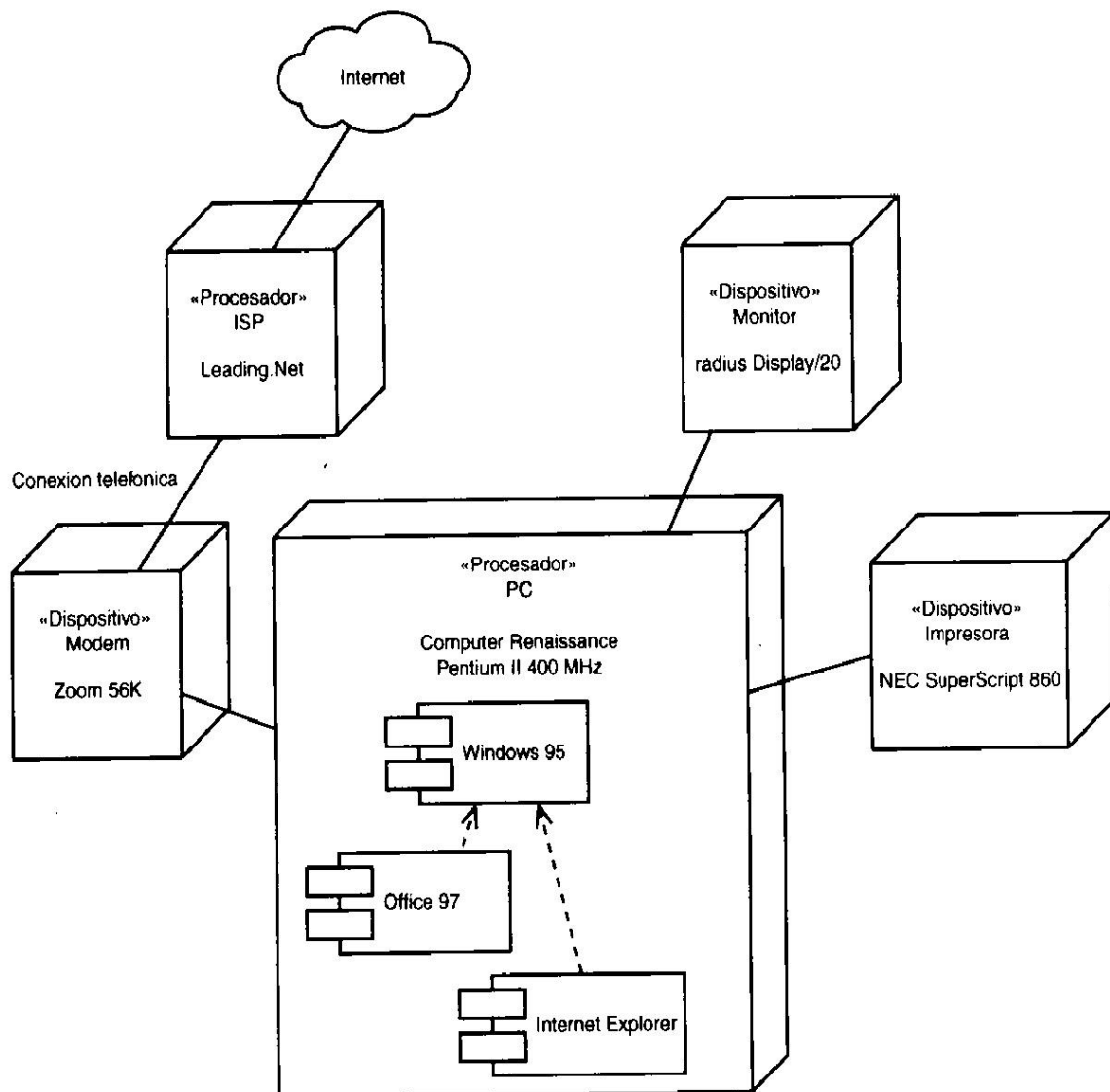


ASOCIACION SE REFIERE A LA CONEXIÓN FISICA ENTRE DOS NODOS



## Componentes y Nodos

### EJEMPLO



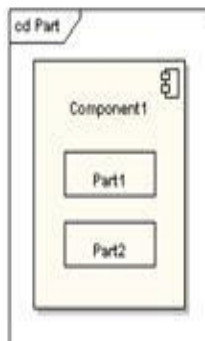


## DIAGRAMA DE ESTRUCTURA COMPUESTA

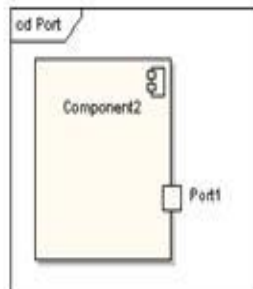
Que muestra la estructura interna de una clase y las *colaboraciones* que esta estructura hace posibles. Esto puede incluir *partes* internas, *puertas* mediante las cuales, las partes interactúan con cada una de las otras o mediante las cuales, instancias de la clase interactúan con las partes y con el mundo exterior, y *conectores* entre partes o puertas.

### SIMBOLOGIA

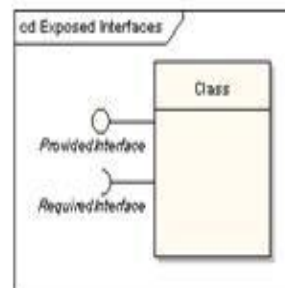
#### PARTE



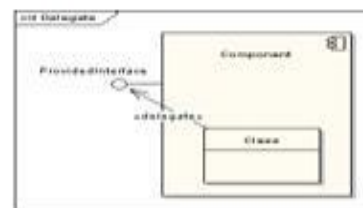
#### PUERTO



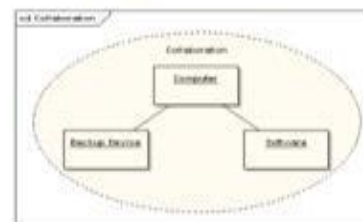
### INTERFACES



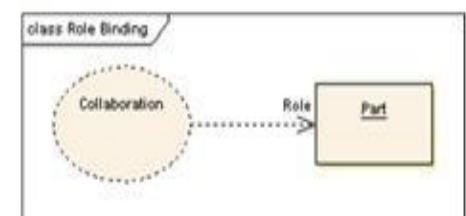
### DELEGAR



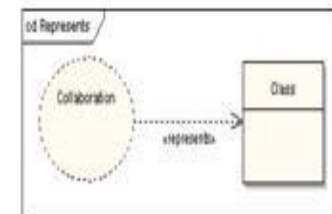
### COLABORACION



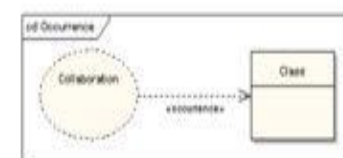
### ENLACE DE ROLES



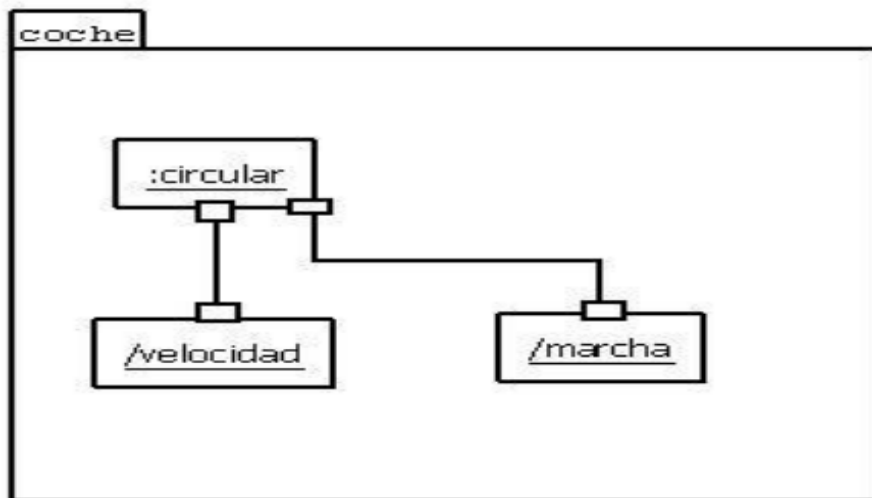
### REPRESENTA



### OCURRENCIA



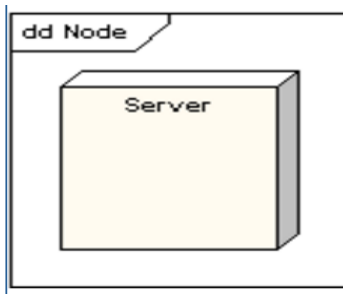
## EJEMPLO



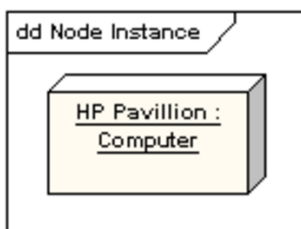
## DIAGRAMA DE DESPLIEGUE

Modela la arquitectura en tiempo de ejecución de un sistema. Esto muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se trazan en esos nodos.

## SIMBOLOGIA

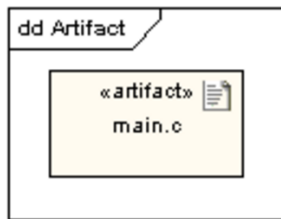


NODO



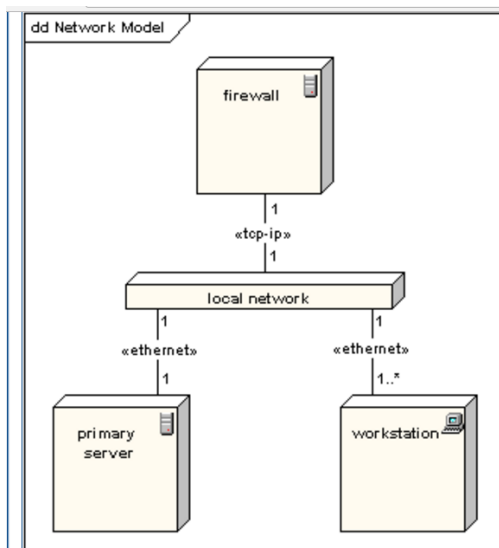
INSTANCIA DE NODO

## ESTEREOTIPOS DE NODO



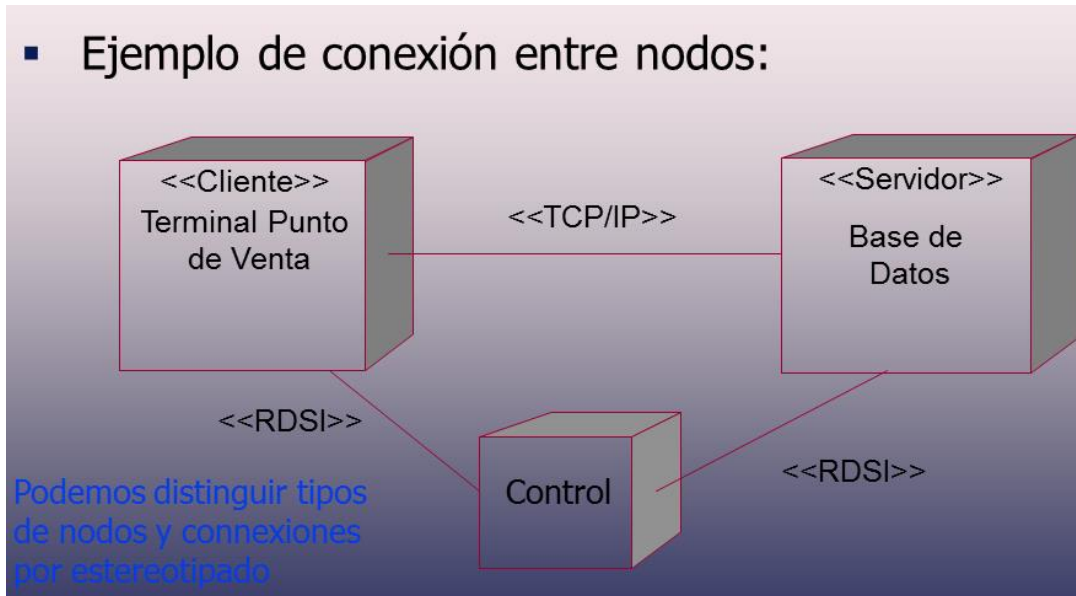
ARTEFACTO producto de desarrollo de software que puede incluir modelos del proceso

Asociación representa una ruta de comunicación entre nodos



## EJEMPLO

- Ejemplo de conexión entre nodos:



## DIAGRAMA DE PAQUETES

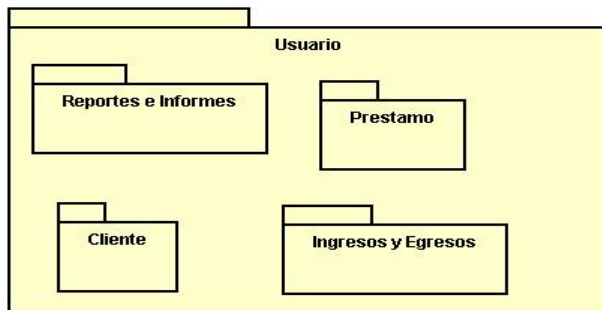
Representa las dependencias entre los paquetes que componen un modelo. Es decir, muestra cómo un sistema está dividido en agrupaciones lógicas y las dependencias entre esas agrupaciones.

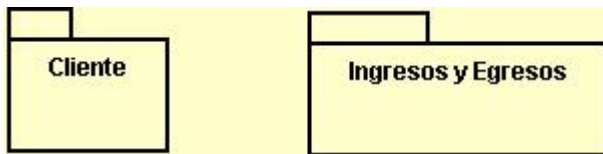
Dado que normalmente un paquete está pensado como un directorio, los diagramas de paquetes suministran una descomposición de la jerarquía lógica de un sistema.

Los paquetes están normalmente organizados para maximizar la coherencia interna dentro de cada paquete y minimizar el acoplamiento externo entre los paquetes. Con estas líneas maestras sobre la mesa, los paquetes son buenos elementos de gestión. Cada paquete puede asignarse a un individuo o a un equipo, y las dependencias entre ellos pueden indicar el orden de desarrollo requerido.

## SIMBOLOGIA

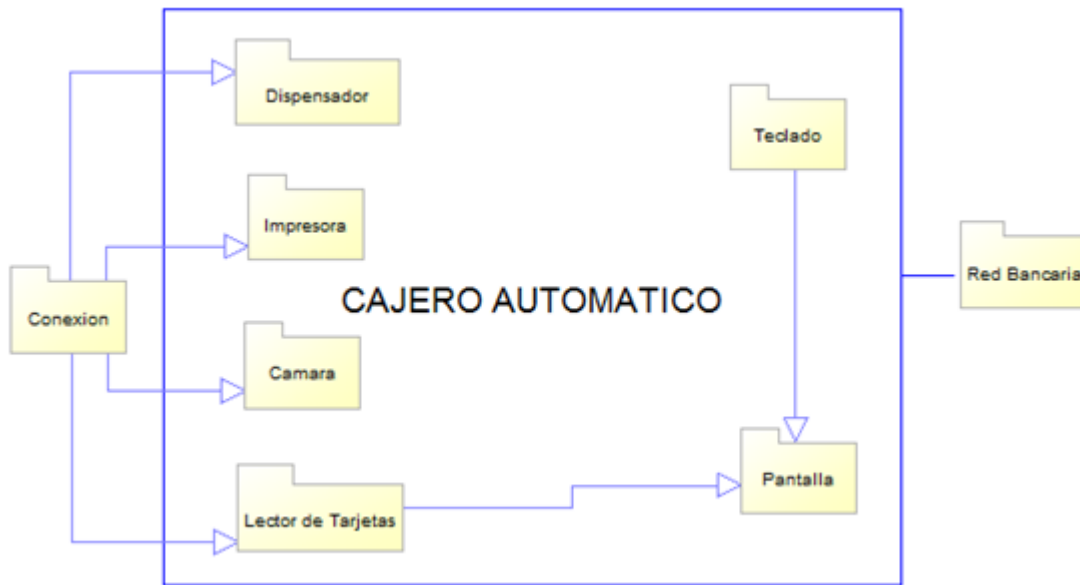
### PAQUETE





CLASES DEL PAQUETE

## EJEMPLO



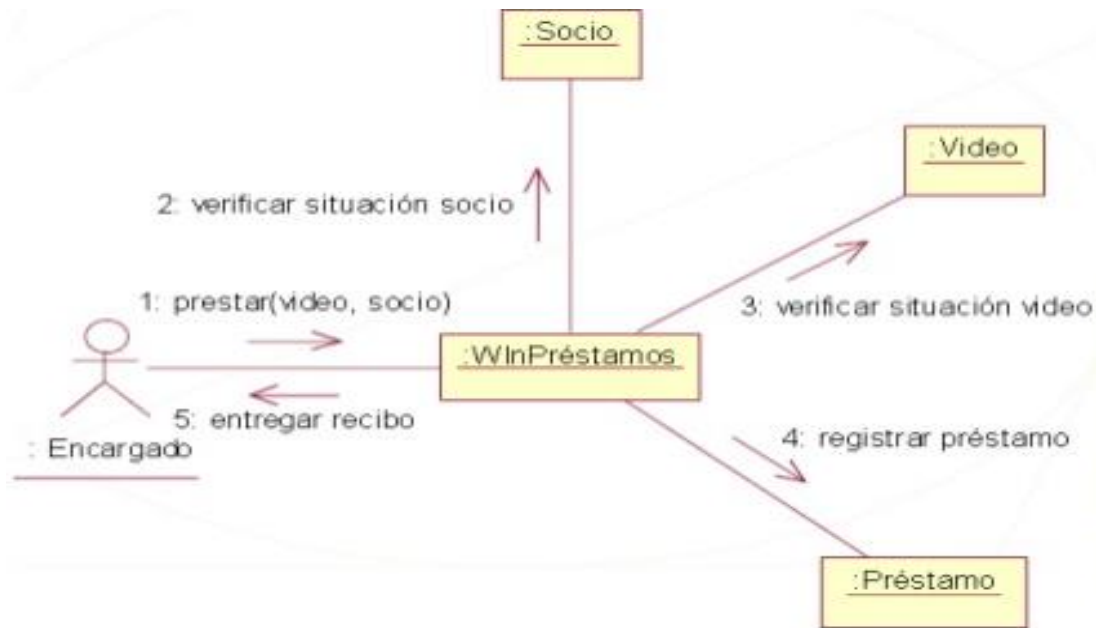
## DIAGRAMA DE COMUNICACIÓN

Un diagrama de comunicación modela las interacciones entre objetos o partes en términos de mensajes en secuencia. Los diagramas de comunicación representan una combinación de información tomada desde el diagrama de clases, secuencia, y diagrama de casos de uso describiendo tanto la estructura estática como el comportamiento dinámico de un sistema.

Los diagramas de comunicación y de secuencia describen información similar, y con ciertas transformaciones, pueden ser transformados unos en otros sin dificultad.

## EJEMPLO





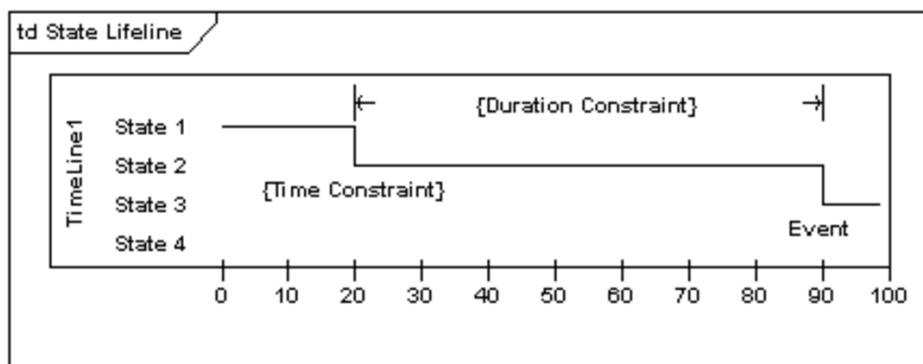
## DIAGRAMA DE TIEMPOS

Se usan para mostrar el cambio en el estado o valor de uno o más elementos en el tiempo. Este también puede mostrar la interacción entre los eventos de tiempos, las restricciones de tiempos y la duración que los gobiernan.

## SIMBOLOGIA

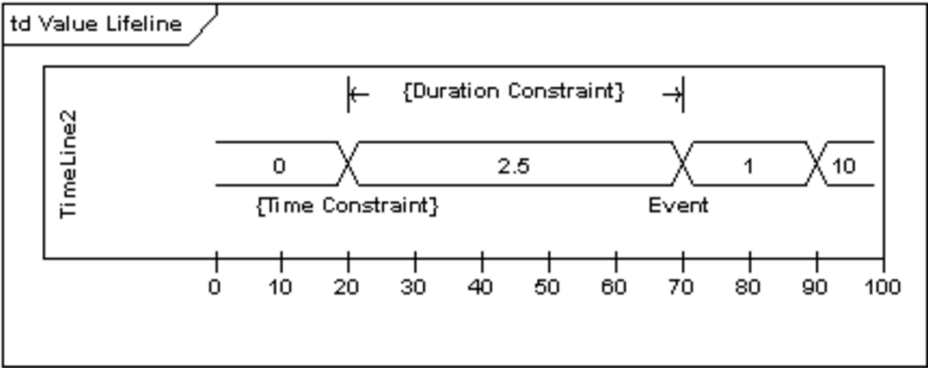
### LINEA DE VIDA DEL ESTADO

Muestra el cambio de estado en el ítem en el tiempo



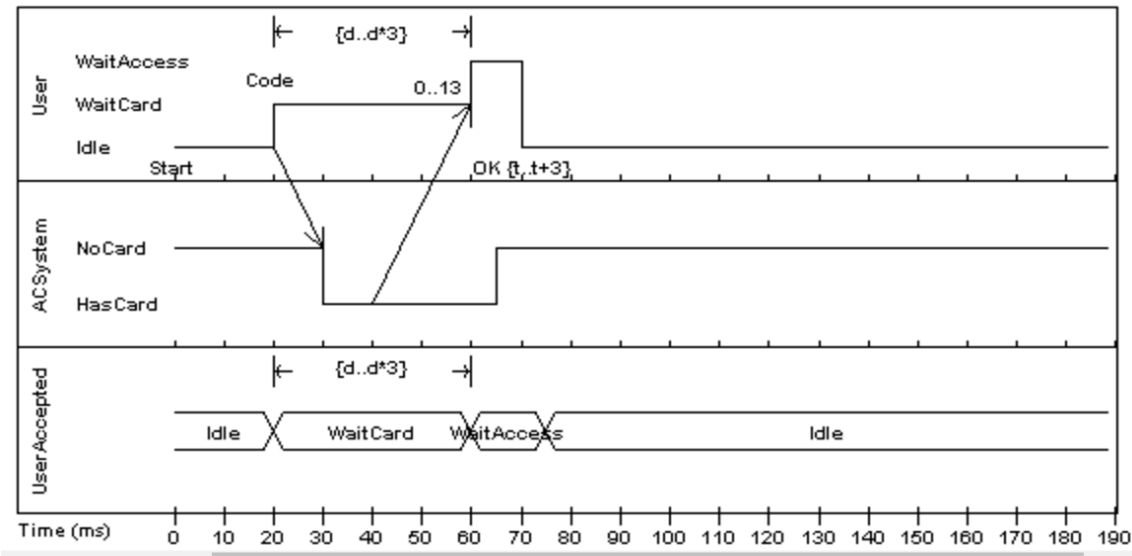
LINEA DE VIDA DEL VALOR

Muestra el cambio del valor de un ítem en el tiempo.

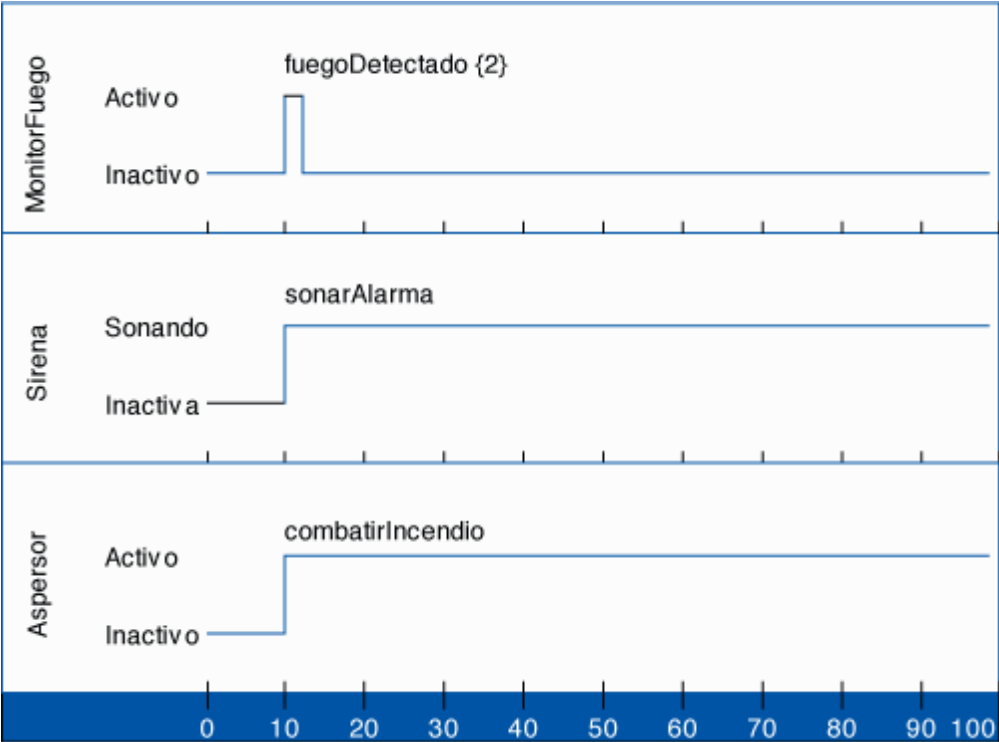


UBICAR TODO JUNTO

Las líneas de vida y del estado se pueden ubicar una arriba de otro en cualquier combinación. Estas deben tener el mismo eje-X. Los mensajes se pueden pasar de una línea de vida a otra. Cada transición del estado o valor puede tener un evento definido, una restricción de tiempo que indica cuándo debe ocurrir un evento, y una restricción de duración que indica cuánto tiempo debe estar en efecto un valor o estado.



EJEMPLO



## BIBLIOGRAFIA

Carlos Ble. (2010-2013). Modelo en cascada. 2012, de Copyright (c) Sitio web: [http://librosweb.es/libro/tdd/capitulo\\_1/modelo\\_en\\_cascada.html](http://librosweb.es/libro/tdd/capitulo_1/modelo_en_cascada.html)

María Torres . (2010-2012). Modelo de Cascada Ingeniería Software. 2012, de Copyright (c) Sitio web: <http://gestionrrhusm.blogspot.mx/2011/05/ingenieria-de-software-modelo-cascada.html>

Isabel Arsitega. (2010-2012). MODELO CASCADA Y MODELO MSF . 2011, de Copyright (c) Sitio web: <http://es.calameo.com/read/00035919610a2f1a84f70>