



Universidad Tecnológica del Centro de Veracruz

Tecnologías de la información y la comunicación.

Saber Hacer Producto

Hora 8 - Diagramas de estados.

Hora 9 - Diagramas de secuencias.

Hora 10 - Diagramas de colaboraciones.

Presenta:

González Blanco María de los Ángeles

Matrícula:

7619

Cuitláhuac, Ver a 1 de Agosto del 2015

Contenido

Simbología	4
Adición de detalles al icono de estado.....	4
Sucesos y acciones.....	5
Objetos.....	9
Mensaje	10
Tiempo	11
La secuencia	12
El diagrama de secuencias	12
El caso de uso.....	13
Instancias y genéricos	14
Diagrama de secuencias de instancias	14
Un diagrama de secuencia genérico	15
Creación de un objeto en la secuencia.....	16
¿Cómo se representaría dicho objeto?.....	16
Como representar la recursividad.....	17

¿Qué es un diagrama de colaboración?	18
Simbología de un diagrama de colaboración:.....	18
La GUI.....	19
Cambios de estados.....	19
La máquina de gaseosas	20
Creación de un objeto	22
Varios objetos receptores en una clase.....	23
Representación de los resultados	23
Objetivos activos	24
Sincronización	25

Hora 8 Diagramas de estados

El diagrama de estados UML captura los cambios, presenta los estado en los que puede encontrarse un objeto junto con las transiciones entre los estados, y muestra los puntos inicial y final se una secuencia de cambios de estado. Un diagrama de estados muestra las condiciones de un solo objeto.

Simbología

Un rectángulo de vértices redondeados representa a un estado, junto con una línea continua y una punta de flecha, mismas que representan a una transición. La punta de la flecha apunta hacia un estado donde se hará la transición. La figura también muestra un círculo relleno que transición. La figura también muestra un círculo relleno que simboliza un punto inicial y la diana que representa a un punto final.

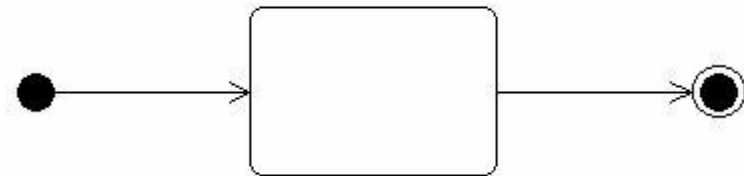


Figura 8.1

Adición de detalles al icono de estado

El UML le da la opción de agregar detalles a la simbología. Así cómo es posible dividir un símbolo de clase en tres áreas (nombre, atributos y operaciones), puede dividir el icono de estado de igual forma. El área superior contendrá el nombre del estado (que tiene que establecer ya sea que haya la subdivisión o no), el área central contendrá las variables de estado, y el área inferior las actividades. La siguiente figura le muestra estos detalles.

Las variables de estado como cronómetros o contadores son, en ocasiones, de ayuda.

Las actividades constan de sucesos y acciones: tres de las más utilizadas son entrada (qué sucede cuando el sistema entra al estado), salida (qué sucede cuando el sistema sale del estado), y hacer (qué sucede cuando el sistema está en el estado). Puede agregar otros conforme sea necesario.

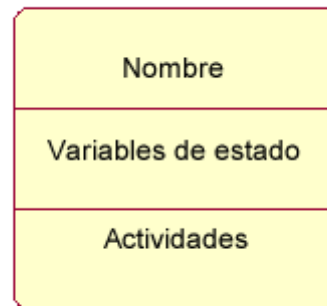


Figura 8.1

Sucesos y acciones

Los sucesos y acciones son los detalles que podemos colocar en las líneas de transición entre estados dentro de nuestro UML, donde el suceso es lo que desencadena o provoca una transición y la acción es la actividad que se ejecuta, es decir, el cómputo que realiza el sistema y que hace que se modifique el estado. No siempre aplican esas condiciones, ya que puede haber estados que no desencadenen acciones o transiciones que se generan debido a que otro estado ha finalizado.

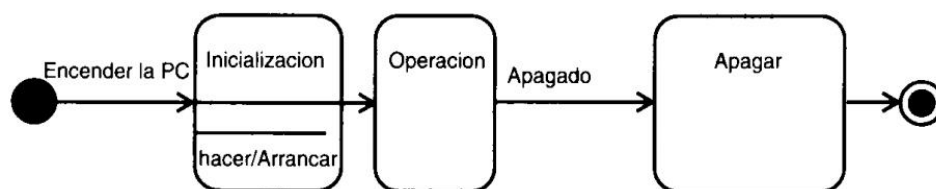


Figura 8.2

Condiciones de seguridad

Ejemplo: si se deja solo el equipo, podría aparecer un protector de pantallas que evitara el desgaste de la pantalla. El intervalo se especifica en el panel de control del sistema operativo.

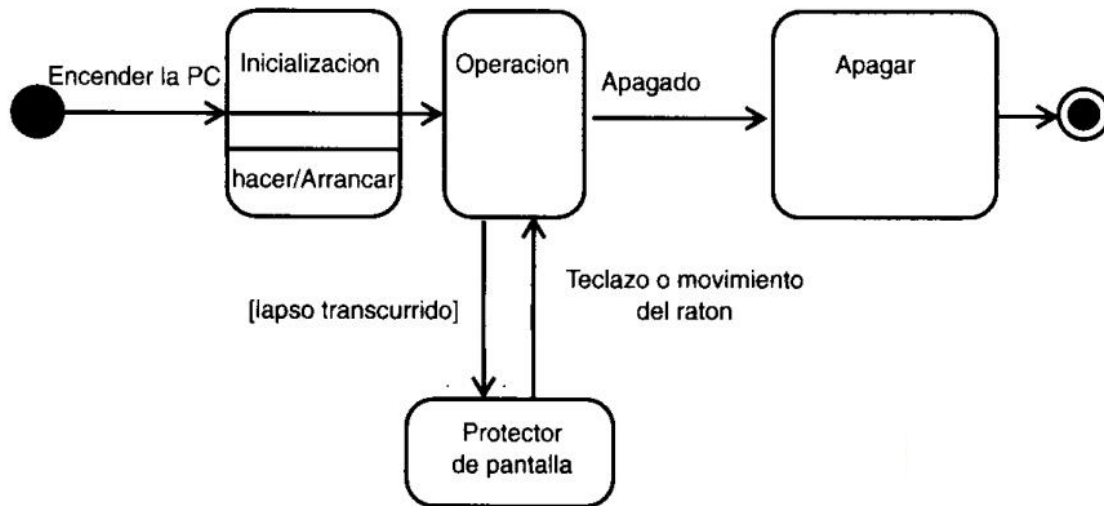


Figura 8.3 Diagrama de estados la GUI, con el estado protector de pantalla y la condición de seguridad.

Subestados

Un subestado, es un estado que se encuentra dentro de otro. Un estado que tiene subestados, se denomina estado compuesto.

Un estado compuesto puede contener subestados secuenciales (disjuntos) o bien subestados concurrentes (ortogonales).

Subestados secuenciales

Ocurren uno detrás de otro, la acción del usuario va desencadenando la transición de un estado a otro.



Figura 8.4 Subestados secuenciales dentro del estado de operación de la GUI

Subestados concurrentes

No permiten especificar dos o más subestados que se ejecutan al mismo tiempo, cuando dos subestados concurrentes llegan a su fin, se juntan de nuevo en un único flujo.

Se representan con una línea punteada. Aunque cada secuencia representa un conjunto de subestados secuenciales ambas son concurrentes entre sí.

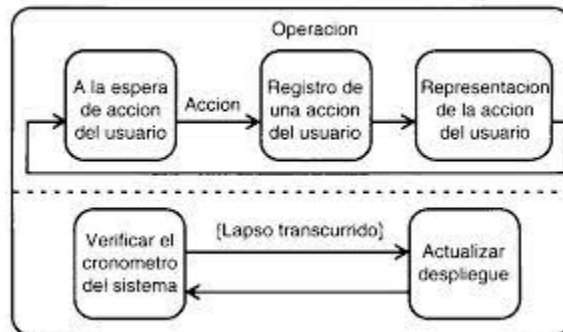


Figura 8.5 Los subestados concurrentes suceden al mismo tiempo. Una línea discontinua los separa.

Estados históricos

Muestra que un estado compuesto recuerda su subestado activo cuando el objeto trasciende fuera de tal estado compuesto.

El UML proporciona un símbolo, es la letra "H" encerrada en un círculo que se conecta por una línea continua al subestado por recordar, con una punta de flecha que apunta a tal subestado.

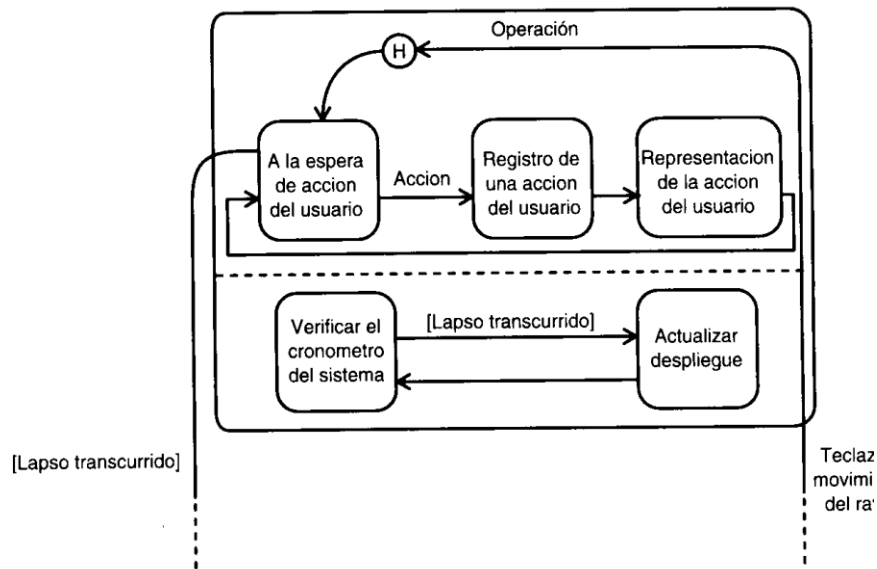


Figura 8.7 El estado histórico simbolizado con la “H” dentro del círculo, le muestra que un estado compuesto recuerda su subestado activo cuando el objeto transcurre fuera de tal estado compuesto.

Mensajes y señales

Los objetos se comunican mediante el envío de mensajes entre sí. Un mensaje que desencadena una transición en el diagrama de estados del objeto receptor se conoce como señal.

Por qué son importantes los diagramas de estado

El diagrama de estados proporciona una gran cantidad de símbolos y abarca varias ideas. Los desarrolladores, deben saber la forma en que los objetos se supone se comportarán, ya que son ellos quienes tendrán que establecer tales comportamientos en el software.

Los diagramas de estado se aseguran que no tendrán que adivinar lo que se supone que harán los objetos, con una clara representación de un objeto aumenta la probabilidad de que el equipo de desarrollo produzca un sistema que cumpla con los requerimientos.

Adiciones al panorama

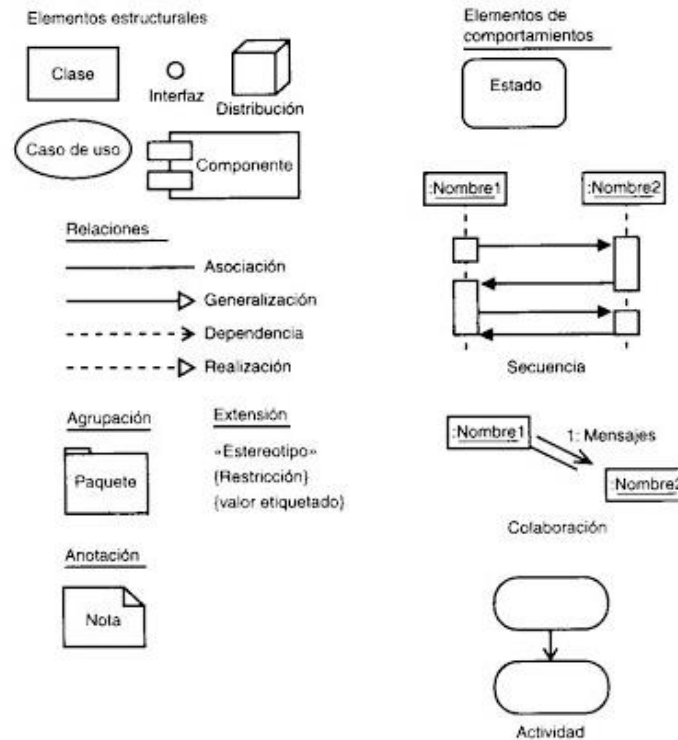


Figura 8.8

Hora 9 Diagramas de secuencia

Consta de objetos que se representan del modo usual: rectángulos con nombre (subrayado), mensajes representados por líneas continuas con una punta de flecha y el tiempo representado como una progresión vertical.

Objetos

Estos se colocan cerca de la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La sentencia que está debajo de

cada objeto es la línea de vida, y un pequeño rectángulo que su longitud representa la duración de una operación.



Figura 9.1

Mensaje

Un mensaje que va de un objeto a otro pasa de la línea de vida de un objeto:

- El objeto puede mandarse un mensaje a si mismo.
- Un mensaje puede ser simple.
- Puede ser síncrono.
- Puede ser asíncrono.

El mensaje es la transferencia de control de un objeto a otro. En un mensaje síncrono enviado no espera respuesta antes de continuar. En el diagrama de secuencia el símbolo del mensaje varía.

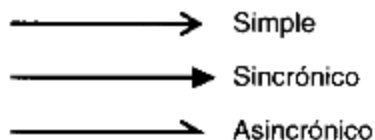


Figura 9.2

Tiempo

El tiempo se representa en dirección vertical, se inicia en la parte superior en dirección a la parte inferior (el mensaje que este más arriba ocurrirá primero).

De esta manera el diagrama de secuencia tiene dos dimensiones:

- Vertical: muestra el paso del tiempo.
- Horizontal: es la disposición de los objetos.

En el diagrama de secuencia suele haber un actor que inicia la secuencia aunque este no es un símbolo del diagrama de secuencia.

En un diagrama de secuencia:

- Los objetos se colocan de izquierda a derecha en la parte superior.
- Cada línea de vida de un objeto es una línea discontinua que se desplaza hacia abajo del objeto.
- Una línea continua con una flecha conecta a una línea de vida con otra, y representa un mensaje de un objeto a otro.

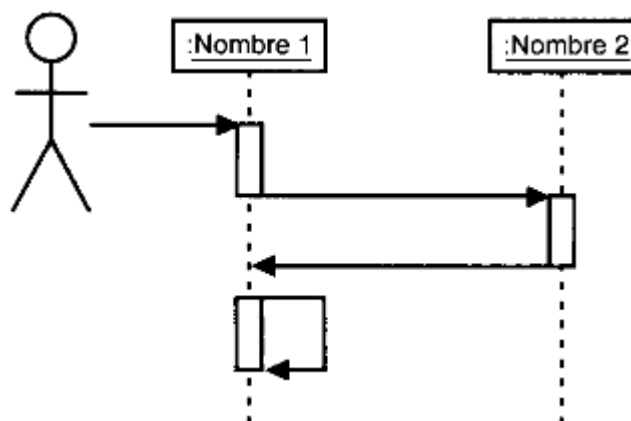


Figura 9.3

La secuencia

Suponga que el usuario de una GUI presiona una tecla alfanumérica; si asumimos que utiliza una aplicación como un procesador de textos, el carácter correspondiente deberá aparecer de inmediato en la pantalla.

- La GUI notifica al sistema que se oprimió una tecla.
- El sistema operativo le notifica a la CPU.
- La CPU notifica a la tarjeta de video.
- La tarjeta de video envía un mensaje al monitor.
- El monitor presenta el carácter alfanumérico en la pantalla.

El diagrama de secuencias

Cuando teclea en un procesador de textos, en ocasiones no se ve aparecer en la pantalla el carácter correspondiente a la tecla que haya oprimido sino hasta después de haber oprimido algunas más.

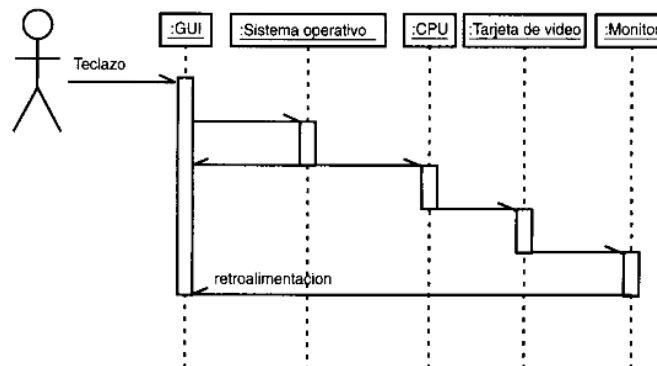


Figura 9.4

En ocasiones, es muy instructivo mostrar los estados de uno o varios de los objetos en el diagrama de secuencias. Dado que ya ha analizado los estados de la GUI esto es fácil de hacer.

Un diagrama de secuencia puede mostrar los estados de un objeto:

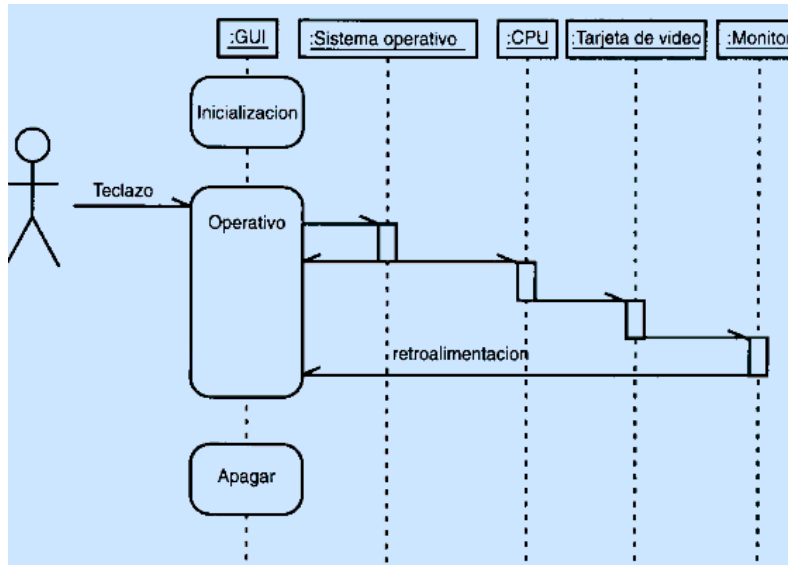


Figura 9.4

El caso de uso

Un diagrama de secuencia muestra exactamente las interacciones de objetos que se realizan durante un escenario sencillo. En el problema del teclado, al representar gráficamente las interacciones del sistema en el caso de uso, el diagrama de secuencia habrá, en efecto, “delineando” el caso de uso dentro del sistema.

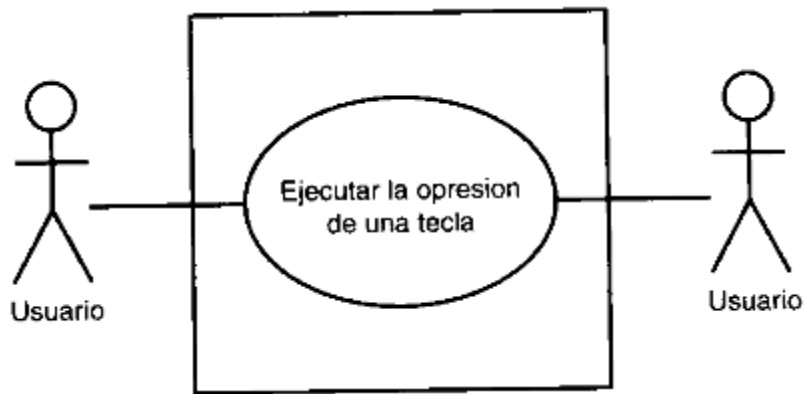


Figura 9.5

Instancias y genéricos

Diagrama de secuencias de instancias

En un caso como comprar gaseosa en una máquina expendedora:

- El cliente inicia el escenario mediante la inserción de la moneda en la maquina
- Hace una selección de gaseosa.
- La máquina entrega la gaseosa.

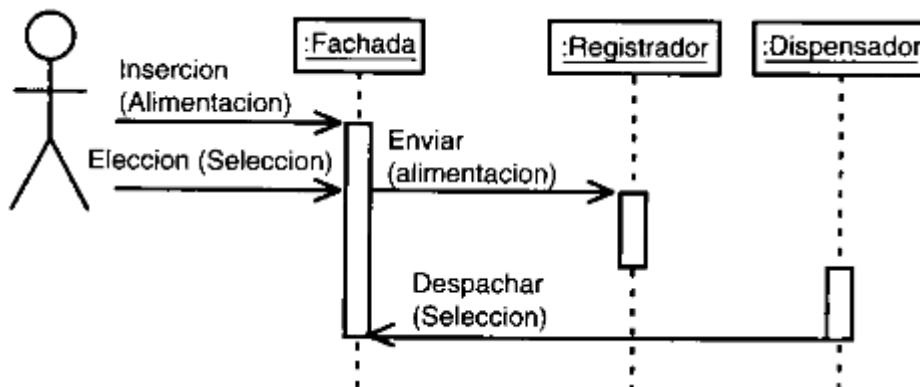


Figura 9.6

Un diagrama de secuencia genérico

Tomando en cuenta todos los escenarios de un caso de uso al momento de crear un diagrama de secuencias, se trataría de un diagrama de secuencia genérico.

Para el escenario monto incorrecto:

- Se analiza el monto insertado es adecuado al precio de la gaseosa.
- Monto mayor al precio, se calcula la diferencia y si dispone de cambio.
- Se puede devolver la diferencia, se devuelve el cambio al cliente y se entrega la gaseosa.
- Si la diferencia no se encuentra en la reserva del cambio, se devuelve el monto ingresado.
- Si la cantidad insertada es menor al precio, el registrador no hace nada y la maquina espera más dinero.

Para representar una condición en la secuencia, la condición se coloca en un “si” entre corchetes, arriba de las flechas del mensaje.

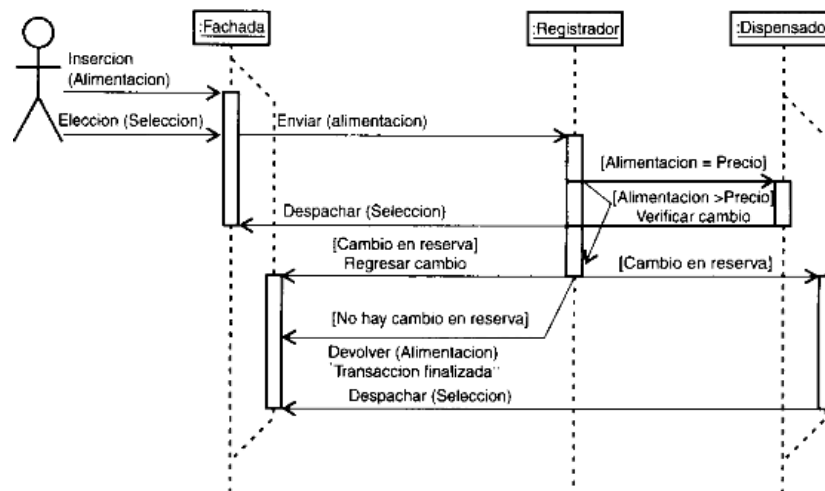


Figura 9.7

Agregando al escenario anterior uno más donde no se cuente con la gaseosa seleccionada:

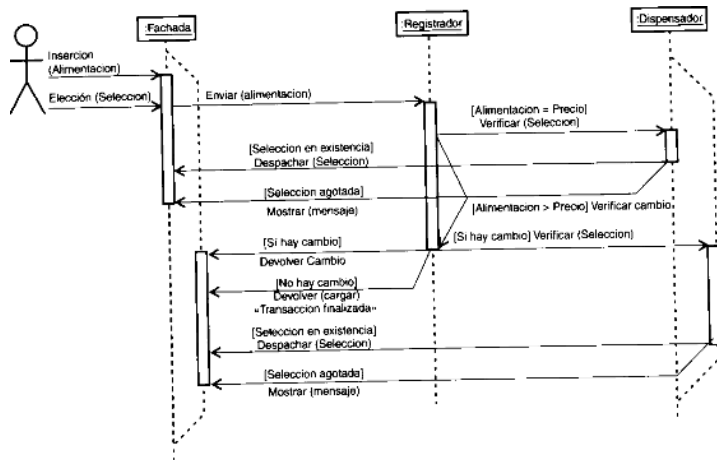


Figura 9.8

Creación de un objeto en la secuencia

Con frecuencia se da el caso de un programa orientado a objetos debe crear un objeto.

¿Cómo se representaría dicho objeto?

En un ejemplo donde:

- Un consultor querrá volver a utilizar partes de una propuesta existente y busca en un área centralizada de la red una propuesta adecuada.
- El consultor encuentra la propuesta adecuada, abre el archivo y en el proceso se abre el software integrado para la oficina relacionada.
- El consultor guarda el archivo con nuevo nombre.
- El consultor no encuentra una propuesta, abrirá la aplicación de oficina y creará un archivo propuesta.
- Al trabajar en la propuesta, utilizará las aplicaciones del software integrado para oficina.
- Cuando finaliza la propuesta, la guardará en el área de almacenamiento centralizada.

En la creación de objetos (de este caso), esta secuencia trae consigo el uso de “si” así como un ciclo “mientras”.

Cuando una secuencia da por resultado la creación de un objeto, tal objeto se representará de la forma usual: con un rectángulo con nombre. La diferencia que ira colocado junto con la dimensión vertical de modo que su ubicación corresponda la momento que se cree. El mensaje que creara al objeto se llamará “Crear ()”. Los paréntesis implican una operación: en un lenguaje orientado a objetos, una operación constructor genera un objeto.

La condición mientras se representa colocando la condición mientras entre corchetes, con un asterisco (*) antes del primer corchete.

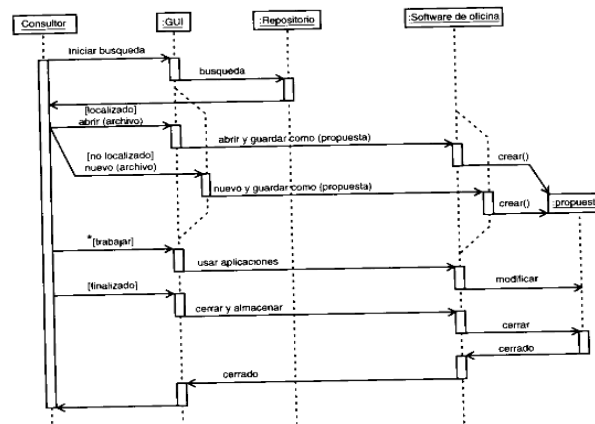


Figura 9.9

Como representar la recursividad

Cuando el objeto cuenta con una operación que se invoca a si misma se llama recursividad. Suponiendo que un objeto en un sistema sea una calculadora, y que una de sus operaciones sea el cálculo de interés. Para calcular el interés compuesto para un periodo que incluya a varios periodos, la operación del cálculo de intereses del objeto tendrá que invocarse a sí misma varias veces. Para representarlo se dibujara una flecha de mensaje fuera de la activación que signifique la operación, y un pequeño rectángulo sobrepuesto en la activación.

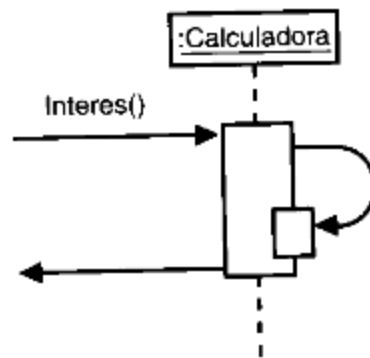


Figura 9.1.0

Hora 10 Diagramas de colaboraciones

¿Qué es un diagrama de colaboración?

Estos diagramas muestran la forma en que los objetos colaboran entre sí. Muestran los objetos junto con los mensajes que se envían entre ellos. Destacan el contexto y la organización general de los objetos que interactúan. Estos se organizan de acuerdo al espacio.

Estos son extensiones de un diagrama de objetos y en este caso se muestra los mensajes que se mandan entre sí. Por lo general evitará la multiplicidad dado que podría ser fuente de confusión.

Simbología de un diagrama de colaboración:

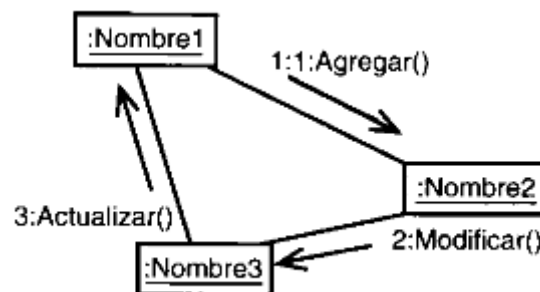


Figura 10.1

La GUI

Un actor inicia la secuencia de interacción al oprimir una tecla, con lo que los mensajes ocurrirán de manera secuencial es:

- La GUI notifica al sistema que se oprimió una tecla.
- El sistema operativo le notifica a la CPU.
- La CPU notifica a la tarjeta de video.
- La tarjeta de video envía un mensaje al monitor.
- El monitor presenta el carácter alfanumérico en la pantalla.

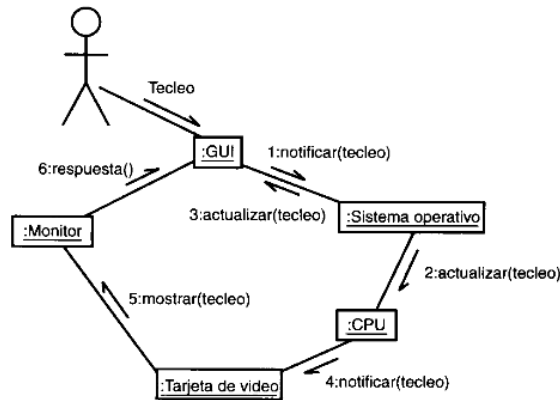


Figura 10.2

Cambios de estados

Estos se pueden representar dentro del diagrama colaboraciones los cambios de estado:

- En el rectángulo del objeto se indica el estado.
- Se agrega otro rectángulo al diagrama que haga las veces del objeto e indique el estado modificado.
- Conectar a los dos con una línea discontinua.
- Etiquetar la línea con un estereotipo << se trona>>.

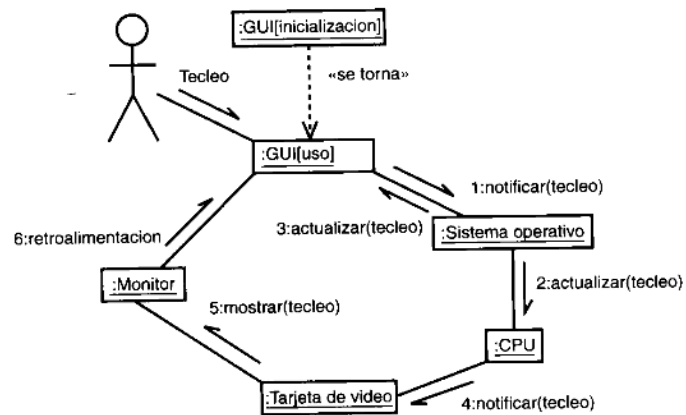


Figura 10.3

La máquina de gaseosas

En un caso como comprar gaseosa en una máquina expendedora:

- El cliente inicia el escenario mediante la inserción de la moneda en la máquina
- Hace una selección de gaseosa.
- La máquina entrega la gaseosa.

El diagrama de colaboraciones es directo:

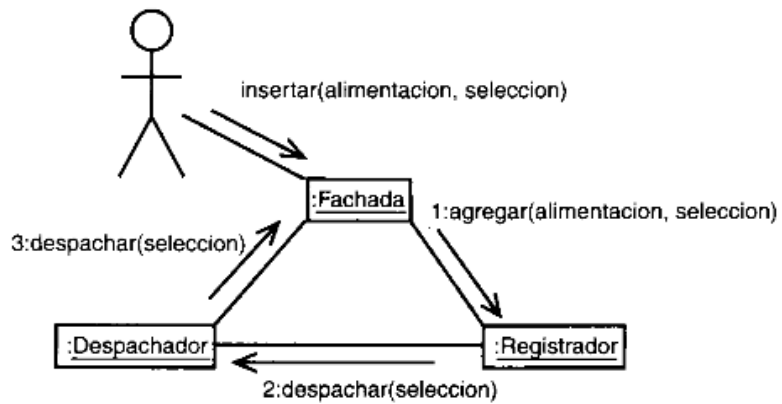


Figura 10.4

Ahora agregando el caso cantidad incorrecta de dinero:

Para indicar esto en el paso de devolver el cambio utilizará el mismo número del mensaje que verifica el cambio, y se agregará un punto decimal y un uno. A esto se le conoce como anidación.

Cuando se agregue esta condición, agregara una bifurcación e el control de flujo. Numerara esta bifurcación como un mensaje anidado.

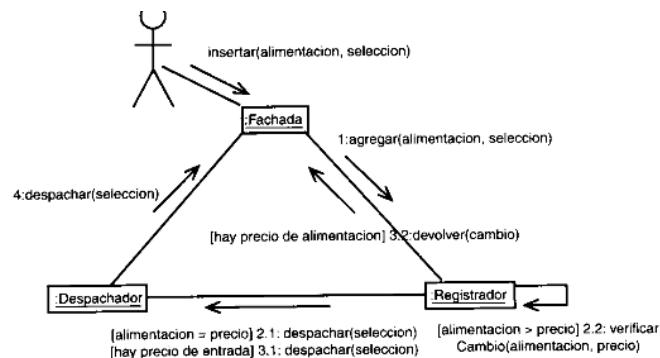


Figura 10.5

Varios objetos receptores en una clase

En ocasiones un objeto envía mensajes a varios objetos de la misma clase en este caso se agregara una condición entre corchetes precedida por un asterisco (*) para indicar que el mensaje ira a todos los objetos.

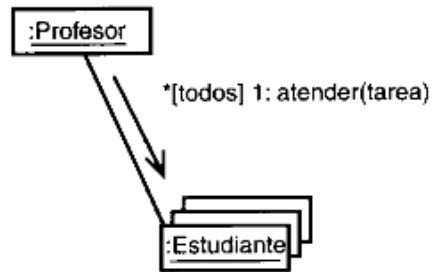


Figura 10.9

En algunos el orden de enviar mensajes es importante:

Un empleado de un banco atiende a los clientes de acuerdo al orden que van llegando y se forman en la fila.

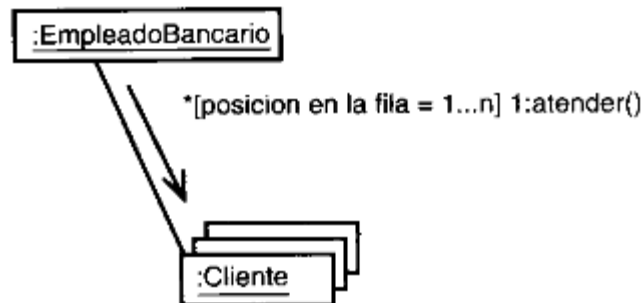


Figura 10.1.0

Representación de los resultados

Un objeto cliente podría solicitar al un objeto calculadora que calcule el precio total que sea la suma del precio de un elemento y el impuesto. En este caso un mensaje es una petición a un objeto para que realice un cálculo y devuelva un valor.

Para ser representado se escribe una expresión que tenga el nombre del valor devuelto a la izquierda, seguido de “:=”, a continuación el nombre de la operación y las cantidades con que opera para producir el resultado.

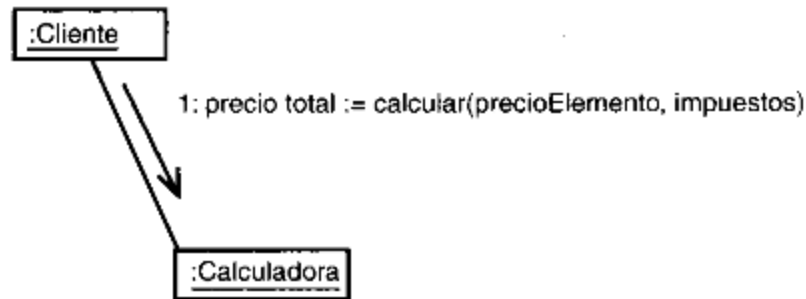


Figura 10.1.1

Objetivos activos

Cuando un objeto controla el flujo de las interacciones, este puede enviar mensajes a los objetos pasivos e interactuar con otros objetos activos. Al proceso de que dos o más objetos activos hagan sus tareas al mismo tiempo, se le conoce como concurrencia. El diagrama de colaboraciones representa a un objeto activo con un rectángulo de borde grueso y más oscuro.

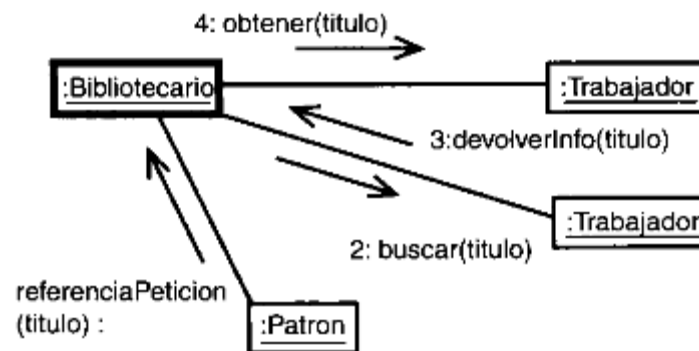


Figura 10.1.2

Sincronización

Otro caso posible es que un objeto solo puede enviar un mensaje después de que otros mensajes han sido enviados, el objeto debe “sincronizar” todos los mensajes en el orden debido. Supongamos que sus objetos son personas en un corporativo, y que están ocupados en la campaña de un nuevo producto:

- El vicepresidente de comercialización pide la de ventas que cree una campaña para un producto en particular
- El vicepresidente de ventas crea la campaña y asigna al gerente de ventas.
- El gerente de ventas instruye a un agente de ventas para que venda el producto de acuerdo con la campaña.
- El agente de ventas hace llamadas para vender el producto a los clientes en potencia.
- Luego de que el vicepresidente de ventas ha dado la comisión y el gerente de ventas ha expedido la directiva, un especialista en relaciones públicas de la corporación hace una llamada al periódico local y colocara un anuncio de la campaña.

Aquí se antecederá con un alista de mensajes que tendrán que completarse antes de que se realiza el paso cinco. La lista de elementos se separa mediante una coma, y finalizará con una diagonal.

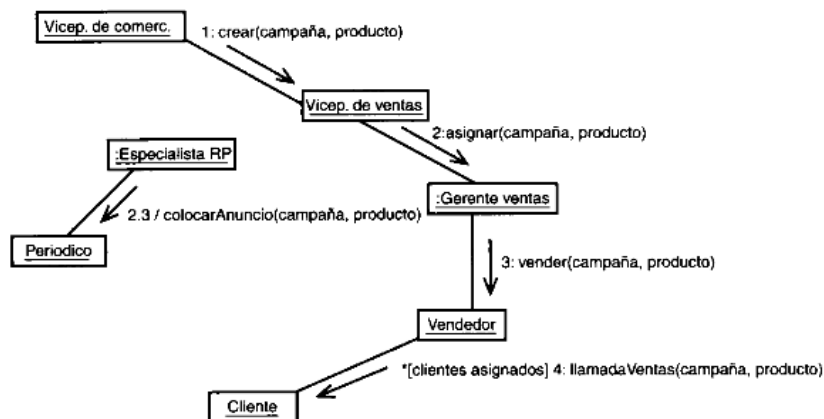


Figura 10.1.3

Bibliografía

Joseph Schmuller. (2001). Aprendiendo UML En 24 Horas. España: PRENTICE-HALL.

