

Ejercicio 1

Modele un objeto que represente un semáforo. El protocolo de dicho objeto debe ser:

- `#siguiente`. Cambia a la siguiente luz (de rojo a amarillo, de amarillo a verde, etc).
- `#sePuedePasar`. Retorna true sólo si el semáforo se encuentra con la luz verde encendida.

Usted debe:

- Realizar un diagrama de clases que documente su solución.
- Implementar los métodos asociados a los mensajes definidos en dicho diagrama.
- Mostrar en un diagrama de instancias los objetos de su modelo para representar a un semáforo con la luz verde encendida.
- Mostrar en un diagrama de interacción cómo resuelve el mensaje `#sePuedePasar` cuando el semáforo se encuentra en luz verde.
- Mostrar en un diagrama de interacción cómo resuelve el mensaje `#cambio` cuando el semáforo se encuentra en luz verde.

Ejercicio 2

Usted debe modelar un objeto para el control de un media center. Para manejar la reproducción de una película usted cuenta con una clase de bajo nivel llamada `DVDReader` que entiende los siguientes mensajes:

- `#start: numberOfSeconds`. Reproduce la película a partir del número indicado.
- `#currentPosition`. Retorna el número de segundo en el que se encuentra la película mientras se está reproduciendo. En el caso que no esté reproduciendo retorna cero.
- `#stop`. Deja de reproducir la película.

Se pide que implemente una clase llamada `MoviePlayer` que entienda los siguientes mensajes:

- `#play`. Comienza a reproducir la película desde el momento cero. Si estaba pausada retoma desde el punto en que se la pausó.
- `#stop`. Deja de reproducir la película, poniendo la posición actual en cero.
- `#pause`. Deja de reproducir la película, manteniendo la posición actual.
- `#displayTime`. Retorna la posición de la película en formato Hora:Minuto:Segundo

Usted debe:

- Realizar un diagrama de clases que documente su solución.
- Implementar los métodos asociados a los mensajes definidos en dicho diagrama.
- Mostrar en un diagrama de instancias los objetos de su modelo para representar a una instancia de `MoviePlayer` pausado en el minuto 3:25 de la película.
- Mostrar en un diagrama de interacción cómo resuelve el mensaje `#play` a partir del estado descrito en el punto anterior.

Ejercicio 3

Usted debe modelar en objetos el manejo de una conexión TCP/IP. Toda conexión TCP/IP se realiza a una determinada IP y en un puerto, pudiendo esta encontrarse en cuatro estados (cerrada, abierta, enviando datos y recibiendo datos).

Para el manejo de bytes a bajo nivel contamos con una clase provista por el sistema operativo. Dicha clase se denomina `ByteSender` y entiende los siguientes mensajes:

- `send: byteCollection to: ip port: aPort`. Envía una colección de bytes a determinado ip + puerto. Es una llamada bloqueante y no retorna hasta que se hayan enviado todos los bytes.
- `receiveFrom: ip port: aPort`. Espera por una colección de bytes en determinado ip + puerto. Es una llamada bloqueante y no retorna hasta que se hayan recibido todos los bytes.

Como se indico anteriormente, una conexión puede estar en cuatro posibles estados:

- Cerrada. Al crearse una conexión se encuentra en este estado. No se puede realizar ninguna acción, excepto abrirla.
- Abierta. Puede recibir datos, enviar datos o cerrarse.
- Enviando datos. No se puede realizar ninguna acción, dado que depende de una llamada bloqueante del sistema operativo. Al finalizar el envío de datos pasa al estado “Abierta”.
- Recibiendo datos. No se puede realizar ninguna acción, dado que depende de una llamada bloqueante del sistema operativo. Al finalizar el envío de datos pasa al estado “Abierta”.

Usted debe:

- Realizar un diagrama de clases que documente su solución.
- Implementar los métodos asociados a los mensajes definidos en dicho diagrama.
- Mostrar en un diagrama de instancias los objetos de su modelo para representar a una conexión tcp/ip abierta.
- Mostrar en un diagrama de interacción cómo resuelve el envío de la colección de bytes #(34 66 78) a partir del estado descripto en el punto anterior.

Ejercicio 4

Usted debe modelar un objeto que represente a una terminal de cajeros automáticos. Un cajero se puede encontrar en los siguientes estados:

- Esperando Tarjeta. No hay ningún usuario operando. La única operación permitida es validar una tarjeta.
- Validando Tarjeta. Un usuario ingresó su tarjeta y PIN y el cajero se encuentra consultando al banco si los datos son válidos. Si la validación es satisfactoria pasa a estar “con usuario logueado”. Caso contrario devuelve la tarjeta al usuario y vuelve a “esperando tarjeta”.
- Con Usuario Logueado. El usuario está validado y se tiene la cuenta bancaria asociado al mismo. Se permiten tres operaciones sobre la cuenta: consultar saldo, depositar y extraer. Asimismo el usuario puede cerrar su sesión, llevando al cajero a “esperando tarjeta”.

Usted debe extender las clases definidas en prácticas anteriores (Banco, CajaDeAhorro y CuentaCorriente) para considerar la identidad del usuario dueño de una cuenta. Para esto considere que un usuario posee una sola cuenta en el banco y que la tarjeta tiene la información del nombre del usuario y el número de cuenta. Finalmente agregue la clase CajeroAutomático que responda a la descripción detallada anteriormente.

En este contexto usted debe:

- 1) Realizar un diagrama de clases que documente su solución.
- 2) Implementar los métodos asociados a los mensajes definidos en dicho diagrama.
- 3) Mostrar en un diagrama de instancias los objetos de su modelo que representen un cajero automático con un usuario logueado. Dicho usuario posee una caja de ahorro con número 924 y saldo de \$3.000.
- 4) Mostrar en un diagrama de interacción cómo resuelve una validación de tarjeta y PIN exitosa.
- 5) Mostrar en un diagrama de interacción cómo resuelve una extracción de \$1.000 suponiendo la situación descrita en el inciso c).
- 6) Mostrar en un diagrama de interacción cómo resuelve una extracción de \$5.000 suponiendo la situación descrita en el inciso c).

Ejercicio 5

El banco para el que usted trabaja acaba de fusionarse con un pequeño banco estadounidense y como parte de la fusión se busca integrar a ambos sistemas. Usted debe incorporar al sistema de manejo de cuentas definidas en prácticas anteriores (Banco, CajaDeAhorro y CuentaCorriente) la cuenta del banco adquirido. Dicha cuenta se encuentra modelada por una clase llamada BankAccount y su protocolo es el siguiente:

- `#balance`. Retorna el balance de la cuenta.
- `#addToBalance: anAmount`. Modifica el balance de la cuenta. Los depósitos se realizan utilizando un monto positivo y las extracciones con un monto negativo. Retorna 1 si la operación fue exitosa y 0 en caso de error.

Usted debe:

- Realizar un diagrama de clases que documente su solución.
- Implementar los métodos asociados a los mensajes definidos en dicho diagrama.
- Mostrar en un diagrama de instancias cómo se relaciona el banco con una instancia de BankAccount.
- Mostrar en un diagrama de interacción cómo se realiza una extracción exitosa de una cuenta fusionada.
- Mostrar en un diagrama de interacción cómo se realiza una extracción fallida de una cuenta fusionada.

Ejercicio 6

Extienda el ejercicio 2 para que su media center pueda interactuar con distintas fuentes de películas:

Clase MovieStream que le permite hacer *streaming* desde un archivo de película:

- `#startStreaming: numberOfSeconds`. Reproduce la película a partir del número indicado.
- `#position`. Retorna el número de segundo en el que se encuentra la película mientras se está reproduciendo. En el caso que no esté reproduciendo retorna cero.
- `#stopStreaming`. Deja de reproducir la película.

Clase FileMovieReader que le permite leer la película desde un archivo:

- `#reset`. Coloca el puntero de lectura al principio del archivo.
- `#position: milliseconds`. Coloca el puntero de lectura corrido desde el inicio en *milliseconds*.
- `#position`. Retorna el número de milisegundo en el que se encuentra el puntero de lectura del archivo.
- `#startReading`. Comienza a reproducir a partir de la posición en la que se encuentre el puntero de lectura del archivo.
- `#stopReading`. Deja de reproducir la película, manteniendo el puntero de lectura en su posición.

Usted debe:

- Realizar un diagrama de clases que documente su solución.
- Implementar los métodos asociados a los mensajes definidos en dicho diagrama.
- Mostrar en un diagrama de instancias cómo se relaciona el media center con una instancia de MovieStream.
- Mostrar en un diagrama de interacción cómo resuelve el mensaje `#play` a partir del estado inicial con una instancia de FileMovieReader.