

Orientación a Objetos II

1er Recuperatorio - Miércoles 15 de Julio de 2015

Ejercicio de Patrones

Una empresa de supermercados está renovando sus sistemas y le ha solicitado a usted que diseñe el nuevo sistema de las cajas registradoras. Lo primero que realiza una operadora de Caja al iniciar sus actividades es desbloquear la Caja registradora para luego procesar todos los productos que el cliente quiere comprar. A medida que se van procesando los productos se va indicando en un LED el total de productos procesados y una vez que todos los productos son procesados se muestra en el LED el monto a abonar. Cada producto tiene un nombre y precio. La caja deberá respetar el protocolo compuesto por los mensajes: #desbloquear, #registrar:unProducto, #mensajeAMostrar, y #finalizar para mantener la compatibilidad con el hardware actual que no se renovará.

Una caja registradora tiene un estado inicial de *Bloqueada* donde solo es capaz de encenderse ante el mensaje "#desbloquear" que pasa la caja al estado *Lista*. Al estar *Bloqueada*, responde al mensaje #mensajeAMostrar retornando "" (string vacío) y no posee ningún comportamiento ante #finalizar y #registrar:. Una vez que pasa al estado *Lista* la caja se comporta de la siguiente forma: ante el mensaje #registrar: agrega el producto a la lista de productos procesados; ante #mensajeAMostrar retorna la cantidad total de productos procesados por la caja; y ante el mensaje #finalizar pasa al estado *Pagando*.

En el estado *Lista*, la caja puede recibir varias veces los mensajes #registrar. En el estado *Pagando*, la registradora responde al mensaje #mensajeAMostrar que muestra el monto total de la compra (no el total de productos) y al mensaje #finalizar (nuevamente) que pasa la caja al estado *Lista* para otra compra, limpiando la lista de productos de la compra de la caja registradora. En este estado, al intentar registrar un producto no se exhibe comportamiento.

En cualquier momento se puede recibir el mensaje #desbloquear que lleva a la caja registradora al estado *Bloqueada* limpiando la lista de productos procesados y cualquier dato de la memoria de la caja.

En resumen, el comportamiento de la caja según sus estados son:

	Bloqueada	Lista	Pagando
#desbloquear	Pasa a estado Lista	Pasa a Bloqueada	Pasa a bloqueada
#registrar:	Nada	Guarda en memoria el producto	Nada
#mensajeAMostrar	String vacío	total de productos en memoria	Suma los precios de productos en memoria
#finalizar	Nada	Pasa a Pagando	Pasa a Lista

Tareas:

1. Diseñe la solución utilizando diagramas de Clase UML
2. Implemente la solución en el lenguaje Smalltalk
3. Escriba un workspace que permita configurar la compra de dos productos utilizando la solución del punto anterior.

Ejercicio de Frameworks

Considere el siguiente diagrama de clases que corresponde a un solución para la construcción de interfaces de usuario del cual no se conoce su funcionamiento interno y no se permite la subclasificación de componentes. En el diagrama de la figura 1 se modela el concepto de Ventana (Window) y un conjunto de elementos que pueden ser agregados con el mensaje #addElement. Los elementos son:

- Button (botón). Permite definir una etiqueta (label) y un comportamiento (definido como un bloque) al ser presionado.
- Label (etiqueta). Permite presentar un texto que es requerido al momento de ser instanciado.
- TextField (campo). Permite al usuario introducir texto y consultar su contenido con el mensaje #value.

Finalmente, una ventana se abre en pantalla utilizando el mensaje #show.