



Programación II

Ejercicios de Patrones de diseño de comportamiento

Ejercicio 1

Se debe crear una aplicación que simule el manejo de envío de noticias a los usuarios inscritos a un sitio web de noticias. Cuando los usuarios crean su usuario del sitio ingresan su nombre, apellido, edad, correo, número de celular. El usuario podrá seleccionar la opción de inscribirse para recibir noticias por correo o por mensaje de texto, para lo cual se deberá validar que tenga el correo o celular según corresponda. Cada vez que se publica una nueva noticia en el sitio se le debe enviar una notificación a los usuarios inscritos con el encabezado de la noticia, la fecha y la hora en que se publicó. En el caso de que fuera una aplicación web el mensaje sería un link a la noticia completa de la página, y se implementaría de forma completa el envío a través de correo o mensaje de texto en el celular, para esta implementación solamente se enviará la notificación y el objeto que la reciba la mostrará en consola junto con el celular o correo.

Deberá crear una prueba que compruebe de forma completa el funcionamiento y utilización correcta del patrón. En la misma deberá inscribir dos usuarios, uno que escoge celular y otro correo y se publican 2 noticias. Luego se inscribe un usuario más que escoja ambas opciones y se publican 2 noticias más. Después el primer usuario con la opción de celular también escogerá la opción de correo y los otros dos se desinscriben de la opción de correo y se publican 2 noticias más.

Para la realización de este ejercicio debe identificar el patrón de comportamiento que solucione de la forma más adecuada el problema. Además deberá analizar qué otros patrones podrían ser aplicados en el contexto del ejercicio e implementarlos si lo considera necesario.

Ejercicio 2

Se debe crear una aplicación que simule el almacenamiento de una lista de correos electrónicos, De un correo se almacena un emisor, un destinatario, un asunto, un mensaje que representa el contenido del correo y una fecha de envío (se toma del sistema).

El sistema deberá dar la posibilidad de buscar en la lista los correos enviados, recibidos o ambos de una determinada dirección de correo, por un rango de fechas o por palabra clave(esta se busca tanto en el asunto como en el contenido del correo). Para todos los casos se devolverá un resumen de la lista de correos que cumplan el criterio de búsqueda siguiendo el siguiente formato:

- Enviados: aparecen los datos del destinatario, el asunto y la fecha. (En el caso de que el asunto está vacío aparecería “Sin asunto”, aplica en los tres casos)
- Recibidos: aparecen los datos del emisor, el asunto y la fecha.
- Los demás: aparecen los datos del emisor, del destinatario, el asunto y la fecha.

Deberá crear una prueba que compruebe de forma completa el funcionamiento y utilización correcta del patrón. En la misma deberá almacenar 5 correos e imprimir el resumen de la lista de las diferentes formas posibles.

Para la realización de este ejercicio debe identificar el patrón de comportamiento que solucione de la forma más adecuada el problema. Además deberá analizar qué otros patrones podrían ser aplicados en el contexto del ejercicio e implementarlos si lo considera necesario.

Ejercicio 3

Se debe crear un programa que simula una aplicación para el manejo de versiones de un proyecto. Un proyecto cuenta con un conjunto de archivos, los cuales se van a simular como un conjunto de Strings.



Cada desarrollador que participe en el proyecto puede solicitar a la aplicación una copia del proyecto actual para trabajar localmente (o sea sobre la copia del proyecto, no sobre el proyecto original). El desarrollador puede actualizar(agregar, modificar o eliminar) cuantas veces requiera cualquiera de los archivos(Strings) del programa. Una vez que el desarrollador ha concluido sus cambios, puede realizar COMMIT del proyecto, lo cual quiere decir que envía su copia local para que actualice el proyecto remoto(el original). En el caso de que haya ocurrido un error con alguno de los COMMIT del proyecto, cualquiera de los desarrolladores puede realizar un REVERT, el cual regresa el proyecto a la versión anterior. Se pueden hacer cuantos REVERT se necesiten pero solo a la versión anterior, no a una versión específica y una vez que se realiza un REVERT no es posible deshacerlo.

Deberá crear una prueba que compruebe de forma completa el funcionamiento y utilización correcta del patrón. En la misma deberá realizar las siguientes operaciones:

- Se crea un proyecto.
- Se crea una copia (como si la solicitara un primer desarrollador)
- Se realizan varias actualizaciones los diferentes archivos.
- Se realiza un COMMIT e imprimir los archivos del proyecto.
- Se realizan más actualizaciones y 2 COMMIT más con sus impresiones.
- Realizar un REVERT e imprimir y luego otro más y volver a imprimir.
- Analice qué sucede si 2 programadores solicitan copias de proyecto, modifican sus archivos locales y realizan sus respectivos COMMIT.

Para la realización de este ejercicio debe identificar el patrón de comportamiento que solucione de la forma más adecuada el problema. Además deberá analizar qué otros patrones podrían ser aplicados en el contexto del ejercicio e implementarlos si lo considera necesario.