

# Diseño y Desarrollo de Software

## Actividad 1 - SOLID

---

### Ejercicio A

La clase `ProcesadorDeContenidosNuevos` viola el principio de abierto-cerrado, cuando se quiera agregar otra red social u otro tipo de repositorio (se menciona que el repositorio actual es un dummy y en un futuro puede cambiar a una base de datos), habría que modificar el código de la clase "`ProcesadorDeContenidosNuevos`". Para solucionar esto cree una interfaz "`RedSocial`" con un método abstracto "`postearContenido`", la cual implementan los distintos manejadores de redes sociales, y una interfaz "`Repositorio`" de la cual heredarán todos los repositorios de la aplicación. De esta manera no sería necesario tocar el código ya escrito al momento de agregar una nueva.

Además, como la clase "`ProcesadorDeContenidosNuevos`" tiene varias responsabilidades (actualiza el repositorio, valida el contenido nuevo, graba el contenido nuevo en el repositorio y luego postea el contenido en la red social) no cumpliría con el principio de responsabilidad única, se menciona que el repositorio actual es un "`repositorio dummy`" *por ahora*, y que en un futuro puede cambiar por una base de datos grande. Esto generaría que en un futuro si se cambia el funcionamiento del repositorio habría que modificar la clase "`ProcesadorDeContenidosNuevos`".

Para solucionar estos problemas, moví la responsabilidad de validar a la clase contenido, creé una clase "`Poster`" que se encargará de postear en las distintas redes sociales y luego hice que la clase "`RepositorioDummy`" implemente la interfaz "`Repositorio`", para que de esta manera si se agregara algún otro tipo de repositorio no deba modificarse la clase principal.