

Actividad A

El principio que no cumple el código es el Open Closed Y Dependency Inversion, ya que no está abierto para extensión. Si bien estaría extendiéndolo para luego usar Tikitaka luego no se podría agregar otra red social sin volver a modificar el código nuevamente. Por lo tanto, no estaría cumpliendo el objetivo de los criterios SOLID, esto se podría solucionar creando una clase por cada red social que implementen una interfaz llamada “ManejadorRedes” el cual contenga un solo método el cual ambas redes sociales tengan en común, en este caso serí `postearContenido()`. De esta forma, si en un futuro deseamos agregar otra red social no tendríamos que modificar ningún código, solamente crear una nueva clase que implemente esa interfaz.

Por otro lado, como `ProcesadorDeContenidos` depende de la implementación de `RepositorioDummy` y `ManejadorDeChulugram` no estaría cumpliendo Dependency Inversion Principle el cual se puede solucionar mediante añadiendo una interfaz llamada “Repositorio” en donde `RepositorioDummy` lo implementa y si bien es un repositorio de juguete mas adelante no va a hacer así

Algunas consideraciones a la hora de modificar el código

Para modificar el código tuve en cuenta las siguientes consideraciones:

- ♥ En la clase `ProcesadorDeContenidosNuevos` como atributo le agregue una lista de los manejadores que implementa dicho procesador ya que ahora no va a estar cableado a los procesadores necesito tenerlo de alguna forma

El diagrama de clases quedaría de la siguiente manera:

