

Malas prácticas que hay que evitar a la hora de implantar y trabajar con SonarQube por Antonio Calero;

- 1.- **Romper una métrica no significa nada.** Se refiere a que hay que hacer caso a Sonar cuando nos informa de que una métrica está rota y si realmente no nos importa o no nos incomoda que esté rota y mejor eliminar esa métrica.
- 2.- **El responsable de calidad es una persona que no desarrolla.** Este es uno de los errores más comunes, ya que la persona que se encarga de definir los distintos umbrales de calidad y reglas a seguir suele tener un perfil de arquitecto de software que lleva muchos años sin “pegarse” con el código, así que lo ideal, es que las reglas sean definidas en consenso con los desarrolladores actuales de cada proyecto.
- 3.- **Demasiado tarde.** No podemos dejar la revisión de la calidad para justo antes de subir a producción y que sea demasiado tarde como para parar una subida. Lo ideal es integrar la calidad desde el primer momento y hacer revisiones de calidad al menos cuando se vaya a pasar de entorno o al finalizar cada uno de los sprints.
- 4.- **Seguimiento de estándares corporativos obsoletos.** En muchas corporaciones se ha definido un documento de guía de estilo que en la mayoría de los casos ha quedado obsoleto y por tanto la herramienta no puede ajustarse a ellos, recordad que la documentación no se actualiza tan rápidamente como la tecnología, por tanto es mejor definir estas reglas en la propia herramienta y evitar el papel.
- 5.- **La no necesidad de reglas personalizadas.** Cada empresa es única y por tanto cada una tiene sus propias necesidades que no tienen que ajustarse a un estándar general; es por ello que la herramienta te permite la creación de reglas propias.
- 6.- **No existen las balas de plata.** por incorporar esta herramienta a nuestro stack no quiere decir que inmediatamente se mejore la calidad, aunque por lo menos ya seremos conscientes del nivel de calidad y podremos actuar en consecuencia, sin olvidar cosas tan importantes como la cobertura de los tests.
- 7.- **No nos preocupamos del IDE.** Lo ideal en este caso es que el IDE esté completamente sincronizado con la herramienta y que antes incluso de realizar un análisis, la propia herramienta a través del SonarLint ya te va informando de posibles violaciones cometidas.
- 8.- **No es necesario difundir los resultados.** Este es un error muy común de los desarrolladores que piensan que por estar haciendo bien (o mal) las cosas no es necesario difundirlo. Es muy necesario para que personas no técnicas puedan conocer el nivel de calidad de los proyectos de los que son responsables. La propia herramienta ya es un radiador de información pero en buena idea también hacer boletines de calidad, seminarios formativos, premiar el proyecto del mes, ...
- 9.- **Todo es un falso positivo.** Esta es la primera defensa de los desarrolladores cuando se sienten “vigilados” por una herramienta de calidad, es cierto que algunos pueden ser falsos positivos, lo que hay que hacer es identificarlos y configurar la herramienta para evitar que despisten a los desarrolladores.
- 10.- **Añadir todas las reglas posibles.** Este también es un error muy común de la persona

encargada de revisar la calidad, “si tengo 1500 reglas de Java voy a ponerlas todas”. Esto es completamente contraproducente ya que demasiadas reglas pueden llegar a contradecirse unas con otras que hacen que los equipos de desarrollos se paren al no poder llegar a una solución.

A estas malas prácticas hay que añadirle la de no tener a una persona 100% implicada con el seguimiento de la calidad y derivarla en los distintos equipos de desarrollo, que solo se van a preocupar cuando ya sea demasiado tarde.