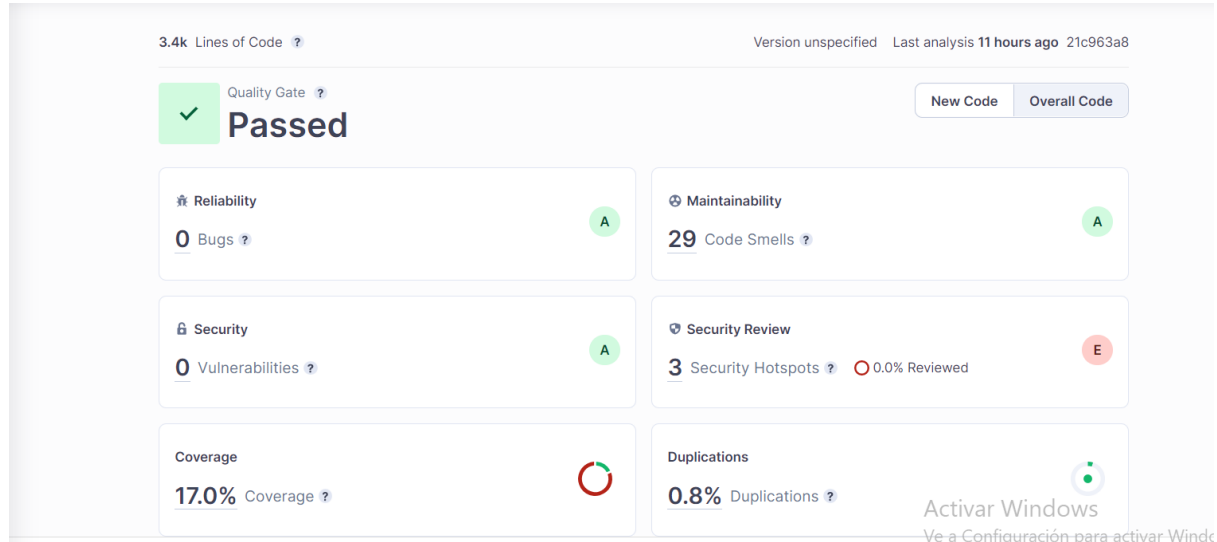


Informe de calidad (Sprint 3)

Autor: Álvaro Alcántara

Captura:



Análisis del código correspondiente a la primera integración del Sprint 3

Resumen sprint anterior:

Comenzamos el desarrollo de este tercer sprint viniendo de un código bastante limpio en el anterior, que pasaba los criterios de calidad de la organización sin bugs, y con solo 16 code smells, todos ellos de tipo minor e info. Aún así, sigue habiendo margen de mejora, por lo que este informe tendrá como objetivo depurar aún más la calidad del producto para pasar los criterios de calidad de la forma más holgada posible.

Incidencias:

En el análisis de sonar, podemos observar que nuestro código cumple con los criterios de calidad de la organización, habiendo mantenido este libre de bugs y con 29 code smells que dan un total 3h y 5m de deuda técnica, por debajo aún del valor límite de 4h establecido por la organización.

Sin embargo, no podemos ignorar el hecho de que los code smells han aumentado, habiendo 13 más que al final del segundo sprint, y aumentando en una 1h y 23m la deuda técnica. Entre estos code smells se encuentran 2 de tipo mayor, 14 de tipo minor y 13 de tipo info, por lo que se establecerá un plan de acción para reducirlos en base a su gravedad y deuda técnica resultante.

También encontramos que se mantienen los security hotspots del final del sprint anterior y ha aparecido uno nuevo, sin embargo esto no es prioritario para la organización, pues se tratan de puntos potencialmente problemáticos y no de problemas de seguridad tangibles, por lo que no se establecerán de momento acciones para subsanarlos.

Por último, observamos que la cobertura ha bajado respecto al final del segundo sprint, este punto es de prioridad baja para la organización y por el momento no se va a tratar, centrando los esfuerzos del equipo en reducir la deuda técnica.

Plan de acción:

En base a las incidencias comentadas anteriormente, se establecerán las siguientes soluciones:

- 1) Le daremos prioridad al code smells de tipo mayor: "Iterate over the "entrySet" instead of the "keySet"" de la clase ListaFavoritasPresenter, reduciendo la deuda técnica en 5 minutos
- 2) Trataremos también los code smells de tipo menor de la clase ListaFavoritasPresenter: "Remove this unused import 'java.util.Objects'" y "Use the primitive boolean expression here." reduciendo la deuda técnica en 7 minutos.
- 3) En la clase FiltroPermanenteView, se resolverá el code smell de tipo menor "Move the array designators [] to the type" reduciendo la deuda técnica en 2 minutos
- 4) En la clase GasolineraDetailView se subsanará el code smell "Remove useless curly braces around statement" que reducirá la deuda técnica en 5 minutos
- 5) Trataremos también los code smells duplicados de tipo info de la clase MainView y ListaFavortiasView: "Remove this use of "NetworkInfo"; it is deprecated"y "Remove this use of "isConnected"; it is deprecated" siendo de 15 minutos cada uno por lo que su resolución reducirá en 60 minutos la deuda técnica.
- 6) Tenemos también un code smell de tipo menor duplicado en las mismas dos clases anteriores, sin embargo, no podemos quitar completamente este code smell para poder asegurar el funcionamiento de la aplicación, por lo que se ha extraerá su uso a la clase estática "Red" reduciendo la deuda técnica en 15 minutos

Comentarios:

- Resolviendo los code smells aquí señalados, la deuda técnica se reducirá en 1h 33 minutos, pasando de 3h y 5 minutos a 1h y 32 minutos, dejando más margen para la aparición de nuevos posibles problemas derivados de la integración de nuevo código y dejando todos los code smells restantes de tipo menor e info.
- En este informe no se tratan más code smells ni todos los security hotspots, porque aparte de ser los restantes code smells de bajo riesgo, se van a realizar más informes con el objetivo de distribuir de forma más o menos uniforme el trabajo en el sprint, por lo que se tratarán de subsanar más adelante.
- Es importante mencionar que no todos los code smells se pueden solucionar ya que no tenemos forma de sustituirlos sin comprometer el funcionamiento de la aplicación.