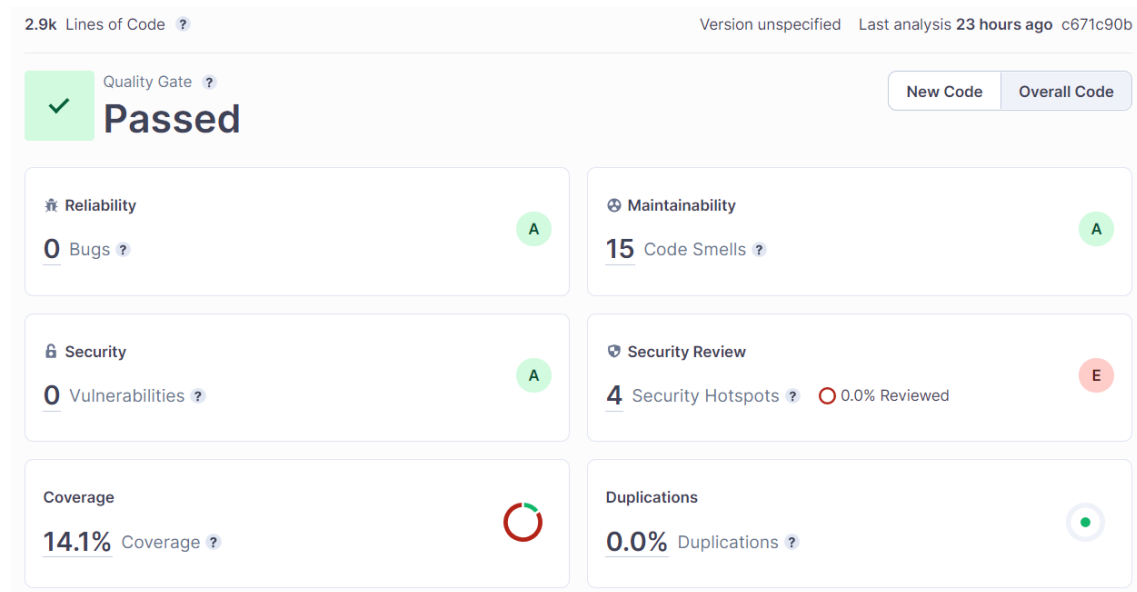


# Informe de Calidad (Sprint 2)

Autor: ÁNGEL CASTANEDO

ANÁLISIS 31 OCTUBRE 2022

## CAPTURA



## RESUMEN ANALISIS DE CALIDAD ANTERIOR

En cuanto al análisis de calidad anterior, se ha realizado el plan de acción correspondiente a este, con lo que se ha conseguido eliminar dos code smells Critical en la clase FiltroPermanenteView, dos code smells Major en las clases FiltroPermanenteView y GasolineraDatabase y cinco code smells Minor en la clase GasolineraArrayAdapter.

Aplicando este plan de acción se ha reducido la deuda técnica en 75 minutos y los code smells totales a 15.

## INCIDENCIAS

En este informe de calidad observamos que nuestro sistema si cumple con las características requeridas para superar los criterios establecidos por la organización, por lo que en este informe se darán pautas para pulir el código restante. Su pueden destacar dos aspectos.

En primero lugar encontramos 4 Security Hotspots, los cuales se van a tratar en este informe. La calificación de este campo es de E y la prioridad es baja, aunque se prioriza resolver estos antes que los code smells restantes, ya que la deuda técnica actual no es elevada.

El otro aspecto serían los 15 code smells, de los cuales 7 son de tipo Minor y 8 de tipo Info. Estos code smells ascienden la deuda técnica a 1h27min y se tratarán en el plan de acción del siguiente informe de calidad.

# Informe de Calidad (Sprint 2)

Autor: ÁNGEL CASTANEDO

## PLAN DE ACCIÓN

Al tratarse de 4 hotspots con prioridad baja el orden de acción es indiferente. A continuación, se explican los distintos fallos de seguridad y cómo arreglarlos.

- 1) El primero de ellos pertenece al campo de "Encryption of Sensitive Data" consta que el "usesCleartextTraffic" está implícitamente permitido en versiones de Android antiguas. Este hotspot se encuentra en el fichero AndroidManifest.xml.
- 2) El segundo también pertenece al campo "Encryption of Sensitive Data" y consta de asegurarse que usar una base de datos sin encriptar es seguro. Este hotspot se encuentra en la clase Prefs.
- 3) El tercero corresponde al campo de "Insecure Configuration" y consta de asegurarse de que el método *printStackTrace* debe ser desactivado antes de entregar el código en producción. Este hotspot se encuentra en la clase CallRunnable
- 4) El último corresponde al campo "others" y consta de asegurarse que el uso de "ACCESS\_FINE\_LOCATION" es necesario. Este hotspot se encuentra en el fichero AndroidManifest.xml.

Comentarios:

- Aplicando el plan de acción definido se busca reducir los riesgos de seguridad de la aplicación al mínimo. Estos se han tratado ahora ya que el número de code smells es bajo y se pretende que sean revisados en el siguiente informe de calidad, ya que estos son de tipo Minor e Info, los cuales son considerados de menor riesgo.