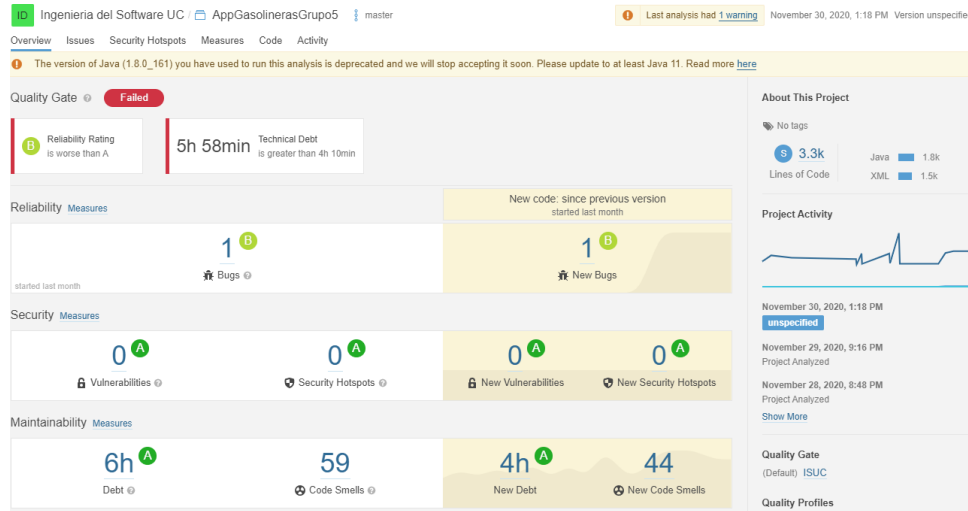


# Informe de Calidad (Sprint 3)

Autores: Pablo Villegas Fernández y Víctor Argüeso Cano

**ANÁLISIS 30 NOVIEMBRE 2020 (Pablo Villegas Fernández)**

## CAPTURA

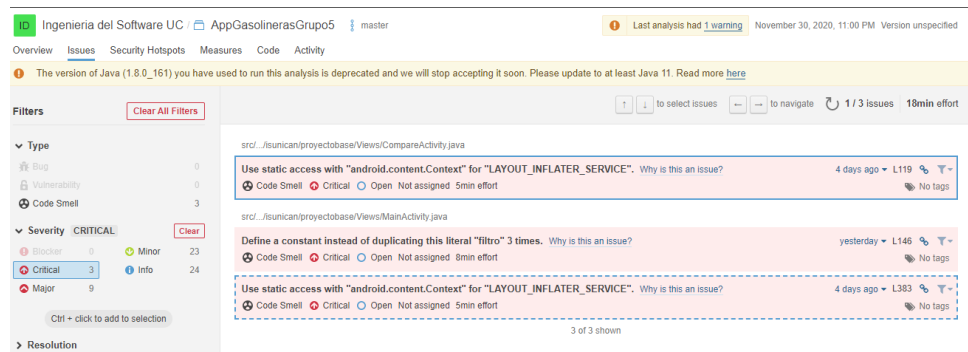


## INCIDENCIAS

Como se puede apreciar en la primera imagen, el análisis no pasa los criterios de calidad de la organización debido a que la calificación en confiabilidad (reliability) es B, cuando debería ser A. Esto es debido a un único bug presente en la clase CompareActivity y a que la deuda técnica supera las 4h 10m.

Por otro lado, encontramos 59 issues de mantenibilidad (code smells) en el conjunto de las clases, por lo que vamos a corregir los que podamos priorizando siempre los más severos.

Code smells críticos:



Code smells mayores:

# Informe de Calidad (Sprint 3)

Autores: Pablo Villegas Fernández y Víctor Argüeso Cano

The screenshot shows the SonarCloud web interface for the project 'AppGasolinerasGrupo5'. The left sidebar contains filters for Type (Bug, Vulnerability, Code Smell), Severity (Blocker, Critical, Major, Minor, Info), and other categories. The main area displays a list of issues. The first issue is 'Add a private constructor to hide the implicit public one' (Code Smell, Major, Open, 5min effort) in 'src/main/java/com/example/gasolineras/DataBase/Converters.java'. Other issues include '1 duplicated blocks of code must be removed' and '2 duplicated blocks of code must be removed' in 'src/main/java/com/example/gasolineras/Views/CompareActivity.java' and 'src/main/java/com/example/gasolineras/Views/FilterActivity.java' respectively. The interface also shows a warning about the deprecated Java version (1.8.0\_161) and a summary of 1/9 issues with a 1h 40min effort.

Code smells menores:

This screenshot shows a more detailed view of the SonarCloud interface, focusing on minor code smells. The left sidebar shows the 'Code Smell' category selected. The main area lists several issues, such as '1 duplicated blocks of code must be removed' and 'Remove this unused import' in 'src/main/java/com/example/gasolineras/Views/CompareActivity.java'. The interface also shows a warning about the deprecated Java version (1.8.0\_161) and a summary of 1/50 issues with a 5h 20min effort.

## PLAN DE ACCIÓN

- 1) Arreglar el bug de la clase CompareActivity.
- 2) Eliminar algunos de los code smells mayores y menores para rebajar la excesiva deuda técnica.

## ANÁLISIS 1 DICIEMBRE 2020 (Víctor Argüeso)

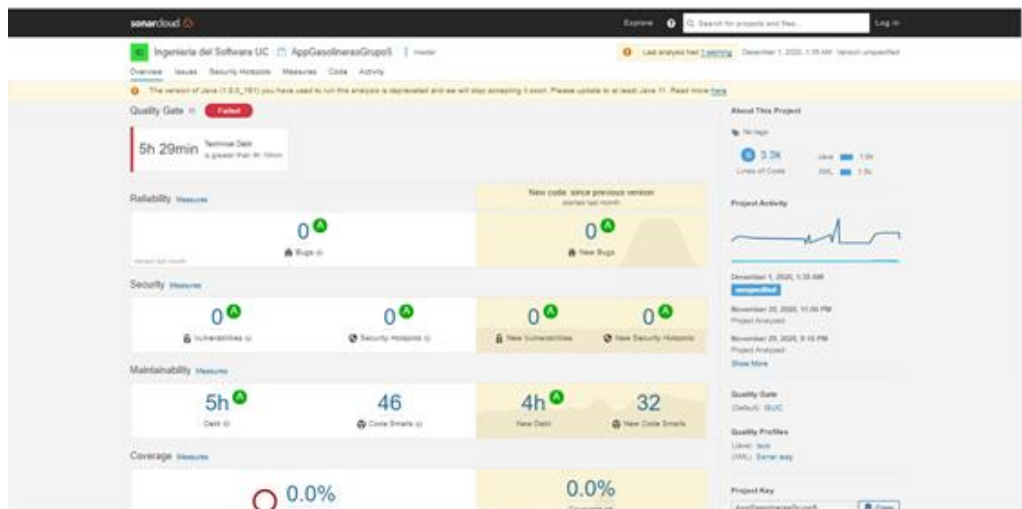
### CAPTURA

Se puede observar que la deuda asciende a 5 horas y 29 minutos, los cuales hacen que el test de calidad no pase correctamente. Los errores vienen dados en gran medida por los 46 issues de

# Informe de Calidad (Sprint 3)

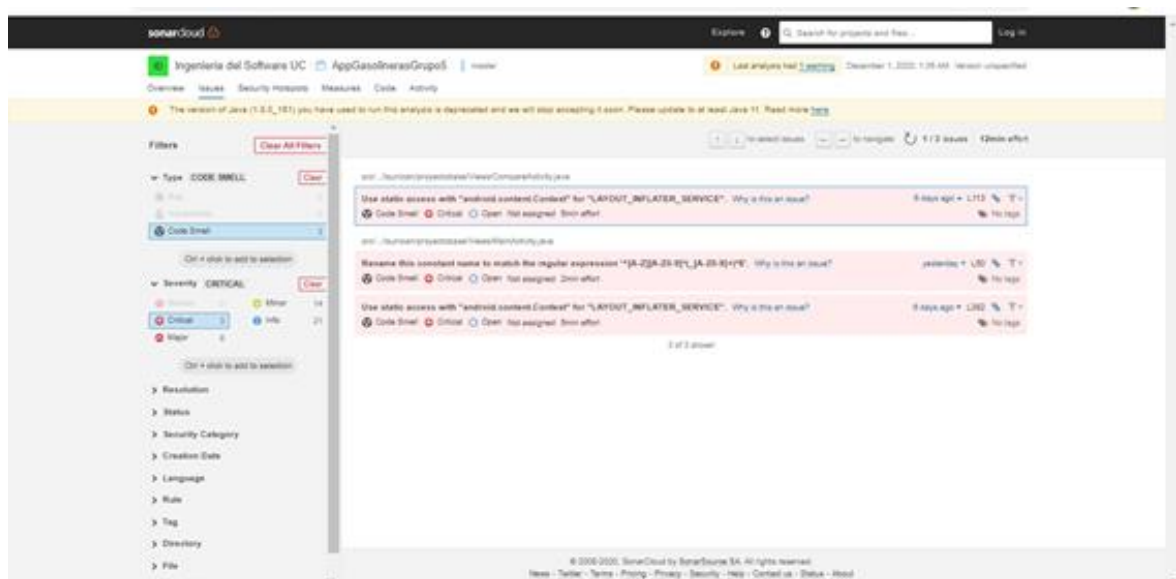
Autores: Pablo Villegas Fernández y Víctor Argüeso Cano

mantenibilidad (code smells) que tenemos y que trataremos de resolver antes de efectuar el siguiente análisis de calidad.



## INCIDENCIAS

### Code smells críticos

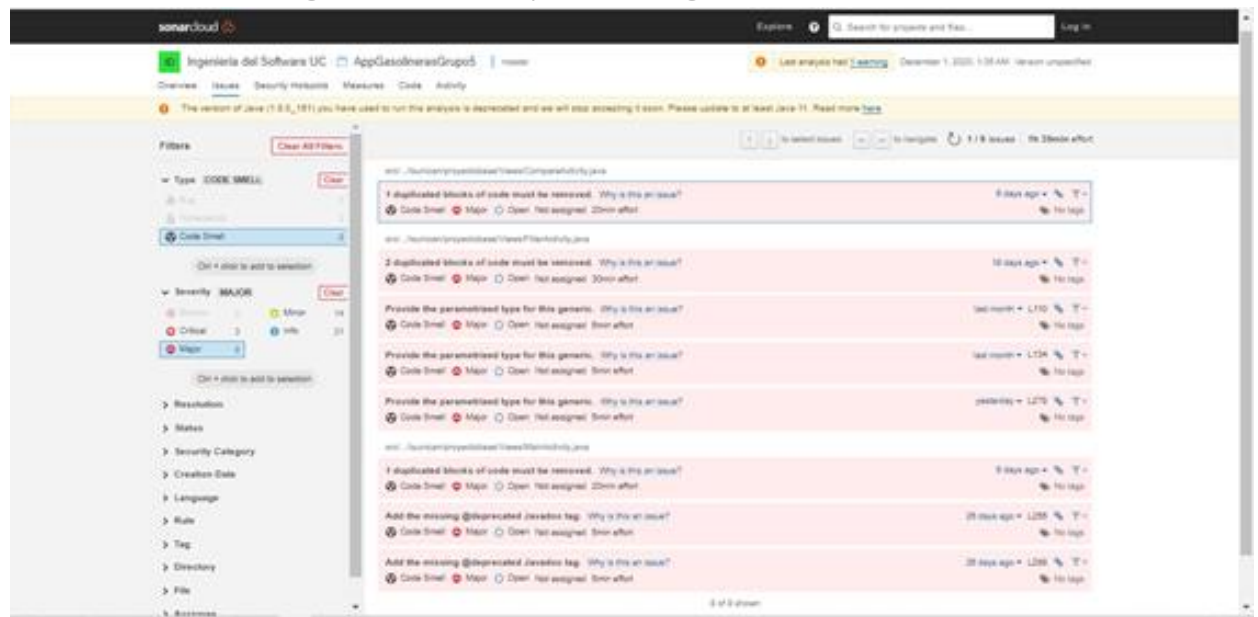


### Code smells mayores:

Se observan 4 errores que vienen dados por duplicaciones de código y que aumentan la deuda técnica en más de una hora entre los 4.

# Informe de Calidad (Sprint 3)

Autores: Pablo Villegas Fernández y Víctor Argüeso Cano



## PLAN DE ACCIÓN

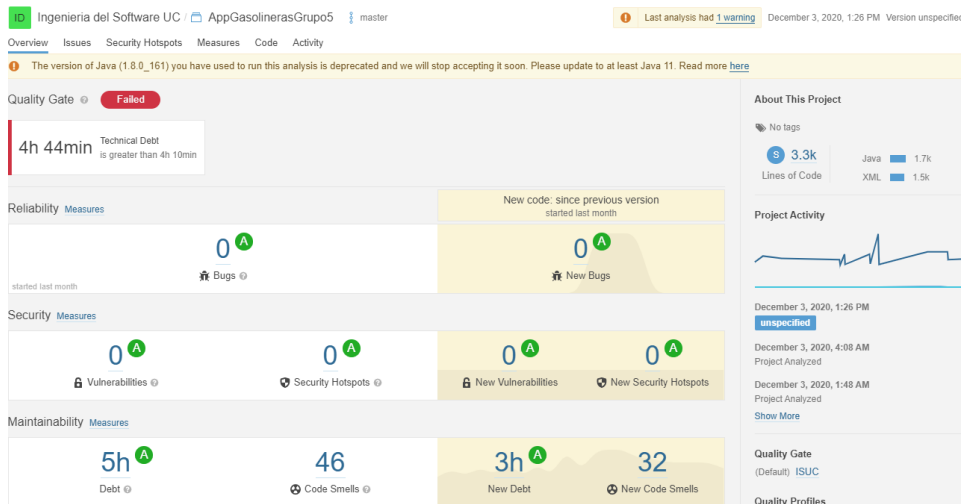
- 1) Solucionar los code smells mayores causados por duplicaciones y así reducir 70 minutos el tiempo de deuda técnica.
- 2) Resolver algún error menor y mayor para bajar un poco más la deuda y así conseguir acercarnos al tiempo de deuda permitido.

ANÁLISIS 3 DICIEMBRE 2020 (Pablo Villegas Fernández)

## CAPTURA

# Informe de Calidad (Sprint 3)

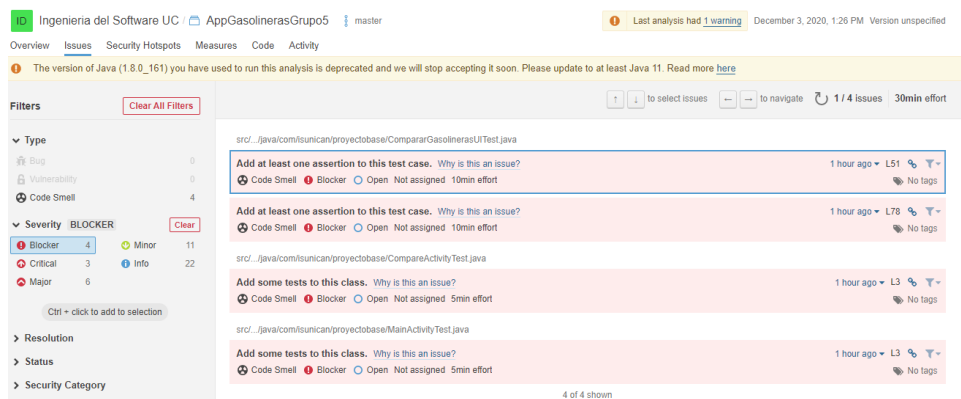
Autores: Pablo Villegas Fernández y Víctor Argüeso Cano



## INCIDENCIAS

Como se puede apreciar en la primera imagen, el análisis no pasa los criterios de calidad de la organización debido a que la deuda técnica supera las 4h 10m.

Por otro lado, encontramos 46 issues de mantenibilidad (code smells) en el conjunto de las clases, por lo que vamos a corregir los que podamos priorizando siempre los más severos.



En la imagen superior, se muestran los Code Smells con una severidad bloqueante que se tratarán en este análisis de calidad.

## PLAN DE ACCIÓN

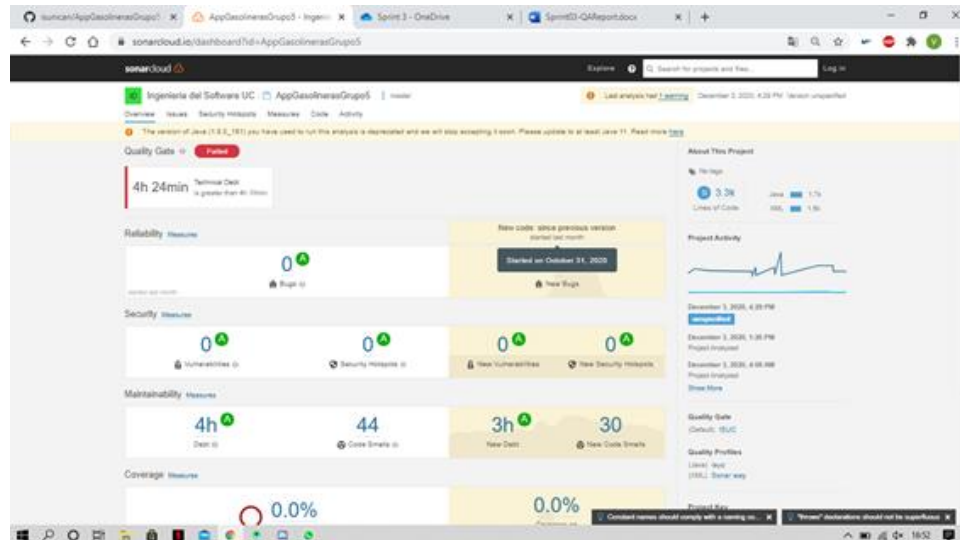
1. Añadir a los test de CompararGasolinerasUITest al menos una aserción, para resolverlo en este caso se han añadido sentencias check(matches()) para comprobar que, efectivamente, realiza comprobaciones y no solo ejecuta instrucciones.
2. La clase CompareActivityTest posee sus métodos en construcción. Por lo que se espera que estos dos code smells se solucionen una vez implementados dichos métodos.

# Informe de Calidad (Sprint 3)

Autores: Pablo Villegas Fernández y Víctor Argüeso Cano

## ANÁLISIS 3 DICIEMBRE 2020 (Víctor Argüeso Cano)

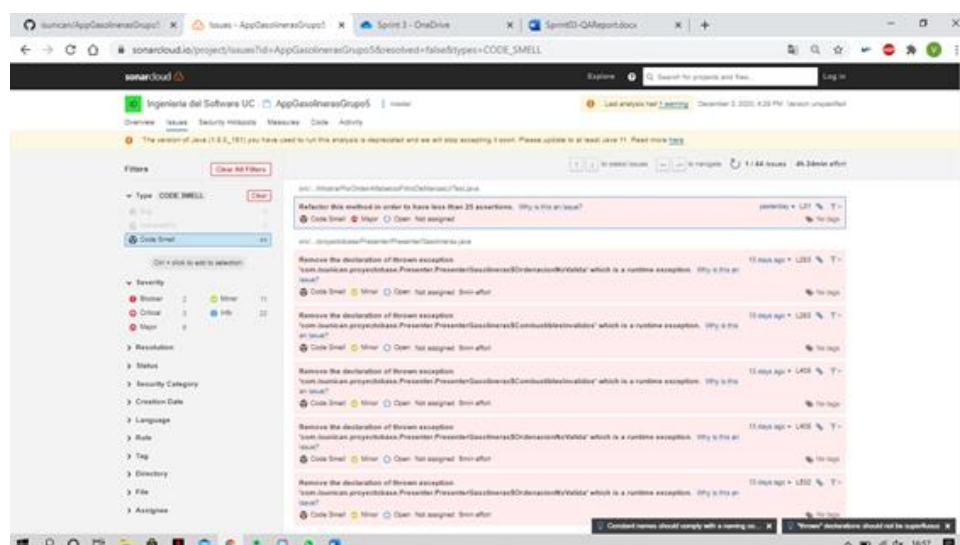
### CAPTURA



### INCIDENCIAS

Hemos bajado la deuda técnica hasta 4 horas 24 minutos, pero aún se encuentra por encima de las 4 horas y 10 minutos, que es tiempo máximo exigido para que pase el test de calidad de sonar.

Por otro lado, encontramos 44 issues de mantenibilidad (code smells) en el conjunto de las clases, por lo que vamos a corregirlos priorizando siempre los más severos.



# Informe de Calidad (Sprint 3)

**Autores:** Pablo Villegas Fernández y Víctor Argüeso Cano

En la imagen superior, se muestran los Code Smells con una severidad bloqueante que se tratarán en este análisis de calidad.

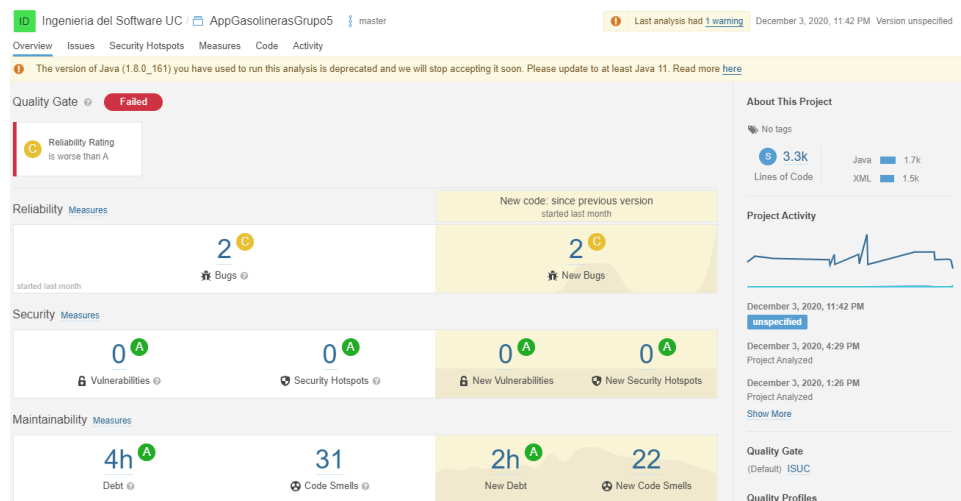
## PLAN DE ACCIÓN

1- Eliminar los fallos de bloqueo que producen un aumento de 10 minutos. Esto se debe a que los test están iniciados, pero no han terminado de realizarse y se resolverá cuando se termine de implementar.

2-Resolver los fallos críticos que causan una deuda de 15 minutos. Con esto se alcanzará el tiempo de deuda necesario para que sonar acepte el test.

## ANÁLISIS 3 DICIEMBRE 2020 (Pablo Villegas Fernández)

### CAPTURA



### INCIDENCIAS

Como se puede apreciar en la primera imagen, el análisis no pasa los criterios de calidad de la organización debido a que la calificación en confiabilidad (reliability) es B, cuando debería ser A. Esto es debido a dos bugs presentes en la clase FilterActivity.

En este análisis nos centraremos en corregir estos bugs.

### PLAN DE ACCIÓN



# Informe de Calidad (Sprint 3)

Autores: Pablo Villegas Fernández y Víctor Argüeso Cano

1. Corregir los 2 bugs. Simplemente se sustituirá el "==" por ".equals()", ya que estamos comparando dos Strings.