

## Informe de calidad de la aplicación TUS Santander (Buscar paradas de una línea)

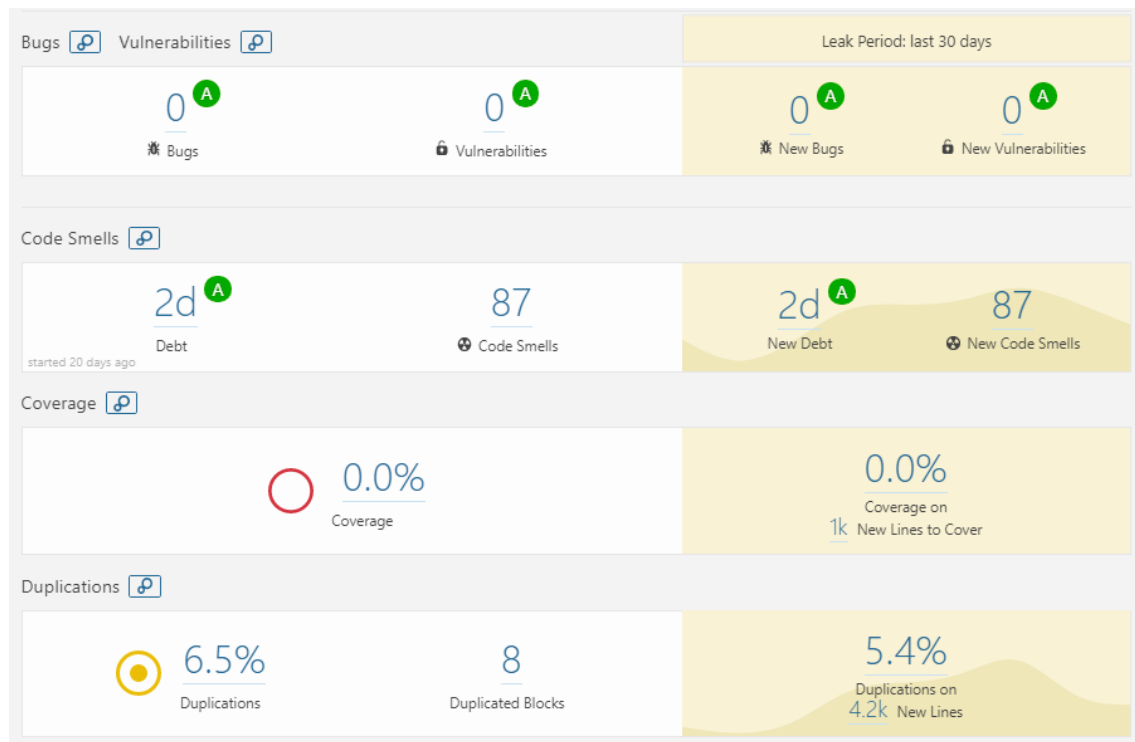
### **Calidad y Auditoría**

*Guillermo Argumosa Arroyo*

#### Análisis Inicial

Se realiza el primer análisis de Sonar sobre el código implementado para la ejecución de la búsqueda de paradas de una línea en concreto de la aplicación.

Se obtiene una puntuación AAA (Fiabilidad, Seguridad, Mantenibilidad) con 0 bugs, pero se tiene también una deuda técnica de 2 días por 87 “code smells” y hay un 6,5% de código duplicado, esto se corresponde con los valores finales del análisis previo de calidad más cinco “code smells” nuevos que han hecho que la deuda técnica pase de 1 día a 2.



#### Cambios Previstos

Antes de nada, se quiere destacar que los nombres de los paquetes test “Aceptacion” y “Aceptacion.Lineas” se han mantenido en mayúsculas por dar problemas en el “Sourcetree” de Git, con lo que se conservarán como “code smells”.

**Se procede a corregir “code smells” de la clase “DatabaseHelperLineasTest”,** habiendo que corregir únicamente el orden de los parámetros de un “assert” (valor esperado, valor actual).

**Se procede a corregir “code smells” de la clase “DatabaseHelperParadasTest”,** donde ocurre lo mismo que en la clase anterior, se modifica el orden de los parámetros de otro “assert”.

**Se procede a corregir “code smells” de la clase “DatabaseHelper”,** únicamente haciendo uso de una constante declarada en el análisis de calidad anterior que en este caso está siendo utilizada como un literal en el código (“ WHERE “).

**Se procede a corregir “code smells” de la clase “RecargaBaseDatosMenu”,** simplemente añadiendo mensajes de error, con un “logger” creado como atributo, dentro de los “catch” de excepciones que el código se encarga de detectar.

**Se procede a corregir “code smells” de la clase “ListParadasPresenter”,** habiendo que hacer uso del método “isEmpty” de una lista en vez de comparar su tamaño con cero.

**Se procede a corregir “code smells” de la clase “RecargaBaseDatosParadas”:**

Se elimina el contenido de los “operadores diamante” en la creación de listas como atributos (ejemplo: “private List<Parada> listParadas = new ArrayList<Parada>();” pasa a ser “private List<Parada> listParadas = new ArrayList<>();” ).

Se añaden mensajes de error, con un “logger” creado como atributo, dentro de los “catch” de excepciones que el código se encarga de detectar.

Se renombran las variables “id\_colorParadas” y “id\_linea” por “colorParadasID” y “lineaID” para evitar el uso de “\_”.

Se elimina la variable “id\_parada” nunca usada.

**Se procede a corregir “code smells” de la clase “LineasActivity”,** solamente habiendo que eliminar el atributo de contexto nunca usado.

**Se procede a corregir “code smells” de la clase “ListLineasAdapter”,** eliminando dos “cast” a “TextView” y uno a “ImageView” innecesarios.

**Se procede a corregir “code smells” de la clase “ListFuncionesMainAdapter”,** eliminando un “cast” a “TextView” y otro a “ImageView” innecesarios.

**Se procede a corregir “code smells” de la clase “MainActivity”,** únicamente añadiendo un “default” al “switch” del método que muestra la lista.

**Se procede a corregir “code smells” de la clase “ListParadasAdapter”,** eliminando dos “cast” a “TextView” y uno a “ImageView” innecesarios.

**Se procede a corregir “code smells” de la clase “ParadasActivity”:**

Se elimina un “import” nunca usado (“org.w3c.dom.Text”).

Se elimina el atributo de contexto nunca usado (“context”).

Se elimina una línea de código comentado (“//this.listParadasPresenter.start();”).

Se hace uso del método “isEmpty” negado para listas en vez de comparar su tamaño con un valor distinto de cero.

### Análisis Final

Se obtiene una puntuación AAA (Fiabilidad, Seguridad, Mantenibilidad) con 0 bugs, la deuda técnica ha sido reducida a 1 día por 51 “code smells” (reduciendo el porcentaje de código duplicado en 0,1%), de los cuales hay una gran parte que no se corregirán por razones previamente explicadas, y los pocos corregibles restantes son de baja importancia, por lo que se considera una gran mejora en cuanto a la calidad del código de la aplicación en el instante del sprint actual.

