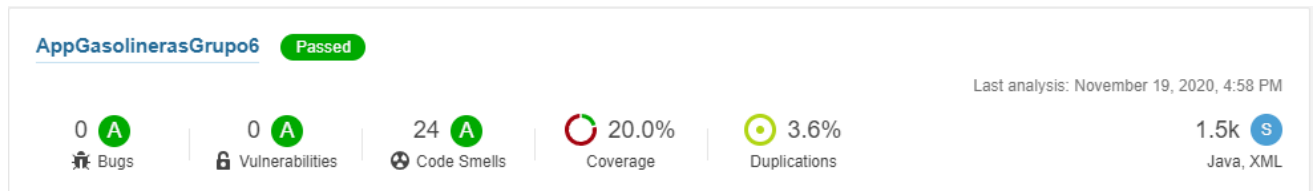


Informe de Calidad (Sprint 2)

Autores: LAUREANO ATECA

ANÁLISIS 19 NOVIEMBRE 2020

CAPTURA



INCIDENCIAS

El análisis pasa correctamente los criterios de calidad, igualmente nos encontramos con un total de 24 code smells y una duplicación del 3.6% de líneas de nuestro código.

De los 24 code smell 5 son de severidad mayor, 12 menor y 7 informativa; siendo la deuda técnica entre todos de 4h. Los code smells de severidad mayor son debidos a la asignación de valores en variables locales que no se van a utilizar (deuda técnica conjunta de 30 minutos), el uso de un System.out en lugar de usar el logger (deuda técnica de 10 minutos), y bloques de código duplicados (deuda técnica conjunta de 1h).

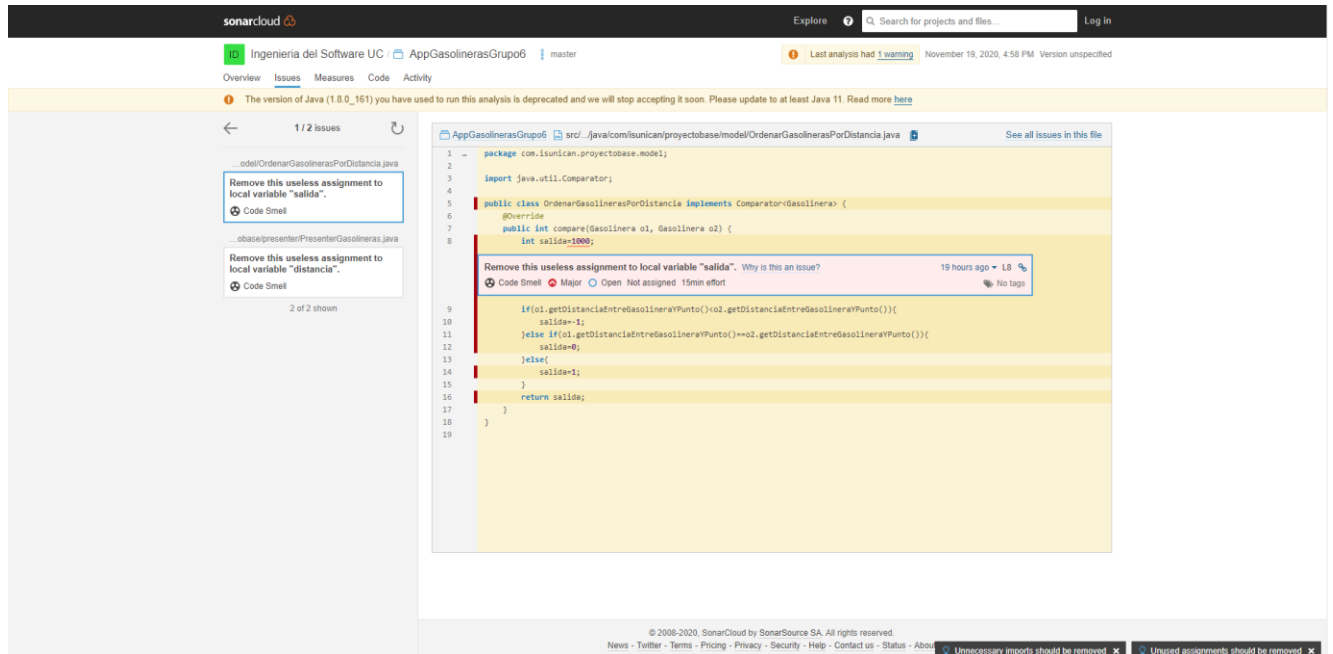
De los code smells de severidad mayor hay 2 que incumplen la misma regla (Unused assignments should be removed) y otro que incumple otra (Standard outputs should not be used directly to log anything). Son code smells cuya corrección no requiere de demasiado tiempo y que va a reducir la deuda técnica en 40 minutos. Esta es una de las dos reglas, y la incumplen dos code smells:

The screenshot shows the SonarCloud interface with two code smells highlighted. The first smell is titled 'Remove this useless assignment to local variable "salida".' and is located in the file 'src/_/sumicam/proyectobase/model/OrdenarGasolinerasPorDistancia.java'. The second smell is titled 'Remove this useless assignment to local variable "distancia".' and is located in the file 'src/_/proyectobase/presenter/PresenterGasolineras.java'. Both smells are categorized as 'Code Smell' with a severity of 'Major' and are marked as 'Open' and 'Not assigned'. The interface also shows a sidebar with filters for Type, Severity, and Rule, and a bottom section with a list of rules.

Informe de Calidad (Sprint 2)

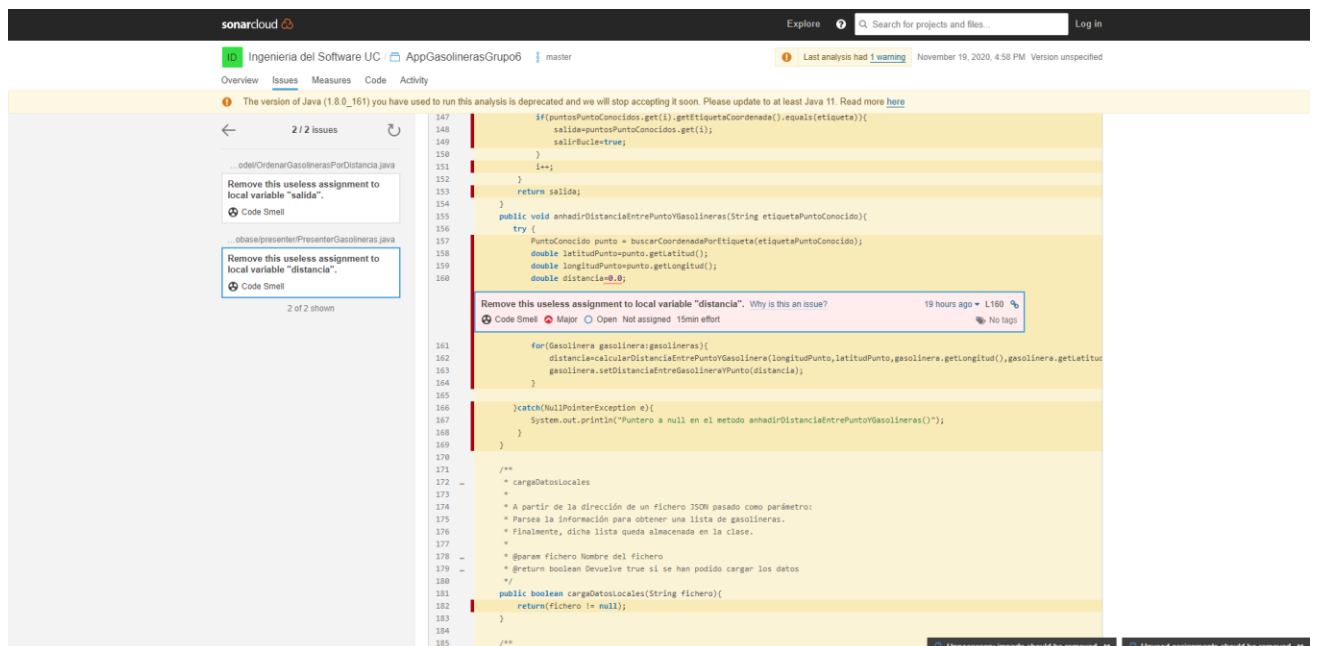
Autores: LAUREANO ATECA

Vemos que uno de los code smells de esta regla está en la clase OrdenarGasolinerasPorDistancia (su deuda técnica es de 15 minutos):



The screenshot shows the SonarCloud interface for the project 'AppGasolinerasGrupo6'. The left sidebar displays a list of issues, with the first one highlighted: 'Remove this useless assignment to local variable "salida".' This issue is categorized as a 'Code Smell' and has an estimated effort of '15mn effort'. The main panel shows the source code for the file 'OrdenarGasolinerasPorDistancia.java'. The code is a Java class implementing the 'Comparator<Gasolineras>' interface. It contains a method 'compare' that compares two 'Gasolineras' objects based on their distance. The issue points to a line where the variable 'salida' is assigned a value, but this assignment is unnecessary as the variable is immediately overwritten in the next line.

El segundo code smell que incumple esta regla es de la clase PresenterGasolineras (su deuda técnica es de 15 minutos):



The screenshot shows the SonarCloud interface for the project 'AppGasolinerasGrupo6'. The left sidebar displays a list of issues, with the second one highlighted: 'Remove this useless assignment to local variable "distancia".' This issue is categorized as a 'Code Smell' and has an estimated effort of '15mn effort'. The main panel shows the source code for the file 'PresenterGasolineras.java'. The code is a Java class implementing the 'Presenter' interface. It contains a method 'anadirDistanciaEntrePuntoYGasolineras' that adds a new gas station to a list. The issue points to a line where the variable 'distancia' is assigned a value, but this assignment is unnecessary as the variable is immediately overwritten in the next line.

Informe de Calidad (Sprint 2)

Autores: LAUREANO ATECA

La otra regla la incumple un único code smell:

The screenshot shows the SonarCloud web interface for a project named 'Ingeniería del Software UC / AppGasolinerasGrupo6'. The left sidebar displays a list of issues, with 'Code Smell' selected. The main panel shows a specific issue titled 'Replace this use of System.out or System.err by a logger. Why is this an issue?'. The issue is categorized as 'Code Smell' with a severity of 'Major'. It is located in the file 'src/main/java/presenter/PresenterGasolineras.java' at line 167. The issue description states: 'Replace this use of System.out or System.err by a logger. Why is this an issue?'. The issue is 19 hours old and has a technical debt of 10 minutes. The left sidebar also shows a list of rules, with '(Java) Standard outputs should not be u...' selected.

Este code smell introduce una deuda técnica de 10 minutos y pertenece a la clase PresenterGasolineras:

The screenshot shows the SonarCloud web interface for the same project, displaying the code snippet for the issue. The code is in the file 'src/main/java/presenter/PresenterGasolineras.java' and is located at line 167. The code snippet is as follows:

```
164 }
165
166 public void anhadirDistanciaEntrePuntoYGasolineras(String etiquetaPuntoConocido){
167     try {
168         PuntoConocido punto = buscarCoordenadaPorEtiqueta(etiquetaPuntoConocido);
169         double latitudPunto=punto.getLatitude();
170         double longitudPunto=punto.getLongitude();
171         double distancia=0.0;
172         for(gasolinera gasolinera:gasolineras){
173             distancia=calcularDistanciaEntrePuntoYGasolinera(longitudPunto,latitudPunto,gasolinera.getLongitude(),gasolinera.getLatitude(),
174             gasolinera.setDistanciaEntreGasolineraYpunto(distancia);
175         }
176     }
177     catch(NullPointerException e){
178         System.out.println("Puntero a null en el metodo anhadirDistanciaEntrePuntoYGasolineras()");
179     }
180 }
```

The issue is categorized as 'Code Smell' with a severity of 'Major'. It is located in the file 'src/main/java/presenter/PresenterGasolineras.java' at line 167. The issue description states: 'Replace this use of System.out or System.err by a logger. Why is this an issue?'. The issue is 19 hours old and has a technical debt of 10 minutes. The left sidebar also shows a list of rules, with '(Java) Standard outputs should not be u...' selected.

Informe de Calidad (Sprint 2)

Autores: LAUREANO ATECA

PLAN DE ACCIÓN

- 1) Arreglar el issue de la primera regla, de la clase OrdenarGasolinerasPorDistancia, en la cual nos encontramos el code smell en el método compare().
- 2) Arreglar el segundo issue de la primera regla, de la clase PresenterGasolineras, en la cual encontraremos el code smell en el método anhadirDistanciaEntrePuntoYGasolineras().
- 3) Arreglar el issue de la segunda regla perteneciente a la misma clase y método que el anterior issue, pero este se encuentra en el catch({}).

Comentarios:

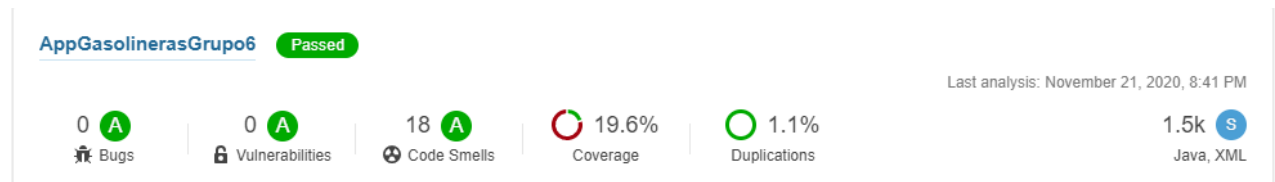
- Los criterios de calidad pasan, pero debemos de realizar planes de acción preventivos para que la situación siga siendo satisfactoria y que la calidad del producto mejore.

Informe de Calidad (Sprint 2)

Autores: LAUREANO ATECA

ANÁLISIS 21 NOVIEMBRE 2020

CAPTURA



INCIDENCIAS

El análisis pasa correctamente los criterios de calidad, pero hay un total de 18 issues que son code smells. De los 18 issues, 2 son de severidad mayor e incluyen una deuda técnica de 20 min cada uno, se trata de código duplicado; 7 code smells son de severidad informativa, siendo 6 de ellos por código obsoleto, y que por el momento no es aconsejable corregir, el code smells restante se debe al uso de un valor de tipo entre <>, en la versión de java que se usa no es necesario; de los 9 code smells restantes, de severidad menor, 8 son de hacer importaciones que no se usan, y 1 de usar una variable local para retornar el valor de salida de un método no siendo necesaria dicha variable.

Nos vamos a centrar en el fichero que más issues tiene, en este caso code smells, este corresponde a la activity UrgenciaActivity:

sonarcloud

Explore Search for projects and files... Log in

Ingeniería del Software UC AppGasolinerasGrupo6 master

Last analysis had 1 warning November 21, 2020, 8:41 PM Version unspecified

Overview Issues Measures Code Activity

The version of Java (1.8.0_161) you have used to run this analysis is deprecated and we will stop accepting it soon. Please update to at least Java 11. Read more here

1 duplicated blocks of code must be removed. Why is this an issue? 5 days ago - L4 - No tags

Remove this unused import 'androidx.swiperefreshlayout.widget.SwipeRefreshLayout'. Why is this an issue? 3 days ago - L4 - No tags

Remove this unused import 'android.app.Activity'. Why is this an issue? 3 days ago - L6 - No tags

Remove this unused import 'android.app.AlertDialog'. Why is this an issue? 3 days ago - L7 - No tags

Remove this unused import 'android.content.DialogInterface'. Why is this an issue? 3 days ago - L9 - No tags

Remove this unused import 'android.os.AsyncTask'. Why is this an issue? 3 days ago - L11 - No tags

Remove this unused import 'android.widget.ProgressBar'. Why is this an issue? 3 days ago - L22 - No tags

Remove this unused import 'com.isunican.projectbase.presenter.PresenterGasolineras'. Why is this an issue? 3 days ago - L28 - No tags

Remove this unused import 'java.util.Arrays'. Why is this an issue? 3 days ago - L31 - No tags

9 of 9 shown

© 2008-2020, SonarCloud by SonarSource SA. All rights reserved. News - Twitter - Terms - Pricing - Privacy - Security - Help - Contact us - Status - About

Unnecessary imports should be removed Unused assignments should be removed

Informe de Calidad (Sprint 2)

Autores: LAUREANO ATECA

Como podemos observar en la imagen la activity tiene 9 code smells, uno de código duplicado, y 8 de importaciones inútiles. Nos centraremos en las importaciones:

The screenshot shows the SonarCloud interface for the project 'AppGasolinerasGrupo6'. The left sidebar displays filters for 'Type' (Code Smell), 'Severity' (Minor), and 'File' (src/main/java/com/...). The main panel lists 8 code smells, all categorized as 'Remove this unused import'. The smells include unused imports for 'android.support.design.widget.SwipeRefreshLayout', 'android.app.Activity', 'android.app.AlertDialog', 'android.content.DialogInterface', 'android.os.AsyncTask', 'android.widget.ProgressBar', 'com.isunican.proyectobase.presenter.PresenterGasolineras', and 'java.util.Arrays'. Each smell is marked as 'Code Smell' with a yellow icon and 'Minor' severity. The right sidebar shows 1/8 issues and 16min effort.

Nos centramos en ellas porque entre todas incluyen una deuda técnica de 18 minutos, y son code smells que se corrigen muy fácilmente. Podemos observar más en detalle las importaciones indebidas en la activity UrgenciaActivity:

The screenshot shows the SonarCloud interface for the project 'AppGasolinerasGrupo6'. The left sidebar displays filters for 'Type' (Code Smell), 'Severity' (Minor), and 'File' (src/main/java/com/...). The main panel lists 8 code smells, all categorized as 'Remove this unused import'. The smells include unused imports for 'android.support.design.widget.SwipeRefreshLayout', 'android.app.Activity', 'android.app.AlertDialog', 'android.content.DialogInterface', 'android.os.AsyncTask', 'android.widget.ProgressBar', 'com.isunican.proyectobase.presenter.PresenterGasolineras', and 'java.util.Arrays'. Each smell is marked as 'Code Smell' with a yellow icon and 'Minor' severity. The right sidebar shows 1/8 issues and 16min effort.

Informe de Calidad (Sprint 2)

Autores: LAUREANO ATECA

PLAN DE ACCIÓN

- 1) Eliminar las importaciones innecesarias de la activity UrgenciaActivity, reduciendo así la deuda técnica en 18 minutos y quitando 8 issues de severidad menor.

Comentarios:

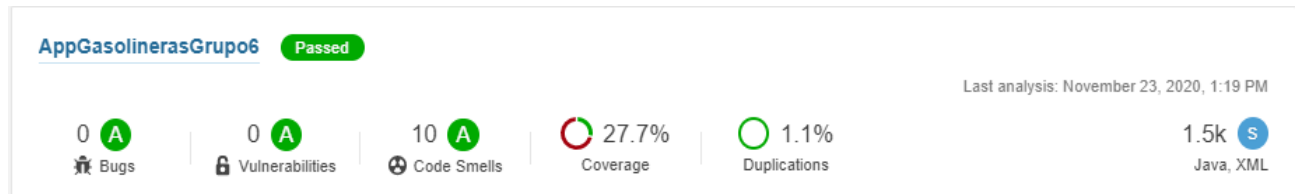
- Los criterios de calidad pasan, pero debemos realizar planes de acción preventivos para que la situación siga siendo satisfactoria y que la calidad del producto mejore.

Informe de Calidad (Sprint 2)

Autores: LAUREANO ATECA

ANÁLISIS 23 NOVIEMBRE 2020

CAPTURA



INCIDENCIAS

El análisis pasa correctamente los criterios de calidad, pero hay un total de 10 issues que son code smells. De los 10 issues, 2 son de severidad mayor e incluyen una deuda técnica de 20 min cada uno, se trata de código duplicado, corresponde a las activities MainActivity y UrgenciaActivity; 7 code smells son de severidad informativa, siendo 6 de ellos por código obsoleto, y que por el momento no es aconsejable corregir, el code smells restante se debe al uso de un valor de tipo entre <>, en la versión de java que se usa no es necesario; y finalmente 1 code smell de severidad menor, resultado de usar una variable local para retornar el valor de salida de un método no siendo necesaria dicha variable.

Las issues de severidad mayor con deuda técnica de 20 minutos cada una:

The screenshot shows the SonarCloud issues page for the project 'AppGasolinerasGrupo6'. The page displays two major issues related to duplicated blocks of code:

- Issue 1: "1 duplicated blocks of code must be removed. Why is this an issue?" (Code Smell, Major, Open, Not assigned, 20min effort). Location: src/_/Asuticam/projectobase/views/MainActivity.java.
- Issue 2: "1 duplicated blocks of code must be removed. Why is this an issue?" (Code Smell, Major, Open, Not assigned, 20min effort). Location: src/_/Asuticam/projectobase/views/UrgenciaActivity.java.

The left sidebar shows filters for Type (Bug, Vulnerability, Code Smell, Security Hotspot) and Severity (MAJOR, Minor, Info). The 'MAJOR' severity filter is selected, showing 2 issues.

Autores: LAUREANO ATECA

La issue de severidad menor con una deuda técnica de 2 minutos:

sonarcloud

ingeneria del Software UC

AppGasolinerasGrupo6

master

Overview

Issues

Measures

Code

Activity

Analysis in progress, this can take up to 30 minutes.

Filters

Clear All Filters

Type

Bug

Vulnerability

Code Smell

Security Hotspot

Severity

MINOR

Blocker

Critical

Major

Minor

Info

Ctrl + click to add to selection

Resolution

Status

Security Category

Creation Date

Language

Rule

Tag

Directory

File

Assignee

src/_projectbase/presenter/PresenterGasolineras.java

Immediately return this expression instead of assigning it to the temporary variable "distanciaEnKilometros". Why is this an issue?

Code Smell

Minor

Open

Not assigned

2min effort

4 days ago • L177

No tags

1 of 1 shown

Explore

Search for projects and files...

Log in

Last analysis had 1 warning

November 23, 2020, 1:19 PM

Version unspecified

© 2008-2020, SonarCloud by SonarSource SA. All rights reserved.

[News](#) - [Twitter](#) - [Terms](#) - [Pricing](#) - [Privacy](#) - [Security](#) - [Help](#) - [Contact us](#) - [Status](#) - [About](#)

Unnecessary imports should be removed

Unused assignments should be removed

sonarcloud

Ingeniería del Software UC / AppGasolinerasGrupo6 master
Explore Search for projects and files... Log in

Overview Issues Measures Code Activity
Last analysis had 1 warning November 23, 2020, 1:19 PM Version unspecified

Analysis in progress, this can take up to 30 minutes.

← 1 / 1 issues ↻

...base/presenter/PresenterGasolineras.java
Immediately return this expression instead of assigning it to the temporary variable "distanciaEnKilometros".
 Code Smell

1 of 1 shown

```

164         lon1 = Math.toRadians(lon1);
165         lat2 = Math.toRadians(lat2);
166         lon2 = Math.toRadians(lon2);
167
168         double dlon = (lon2-lon1);
169         double dlat = (lat2-lat1);
170
171         double sinLat = Math.sin(dlat / 2);
172         double sinLon = Math.sin(dlon / 2);
173
174         double a = (sinLat * sinLat + Math.cos(lat1)*Math.cos(lat2))*(sinLon*sinLon);
175         double c = 2 * Math.asin (Math.Min(1.0, Math.sqrt(a)));
176
177         distanciaEnKilometros = Math.floor(radioTerrestre * c * 10) / 10; // se trunca a un decimal
        
```

Immediately return this expression instead of assigning it to the temporary variable "distanciaEnKilometros". Why is this an issue?
4 days ago - L177 🔍

🔒 Code Smell
🟢 Minor
🔓 Open Not assigned 2min effort
No tags

```

178             return distanciaEnKilometros;
179         }
180     }
181
182     /**
183      * Metodo auxiliar para buscar un determinado punto conocido a partir de su etiqueta identificativa
184      * @param etiqueta
185      * @return PuntoConocido
186      */
187     public PuntoConocido buscarCoordenadaPorEtiqueta(String etiqueta){
188         PuntoConocido salida=null;
189         int i=0;
190         boolean salirBucle=false;
191         while(i<puntosPuntoConocido.size()&&!salirBucle){
192             if(puntosPuntoConocido.get(i).getEtiquetaCoordenada().equals(etiqueta)){
193                 salida=puntosPuntoConocido.get(i);
194                 salirBucle=true;
195             }
196             i++;
197         }
198         return salida;
199     }
200
201     /**
202      * Metodo que asigna a cada gasolinera la distancia que hay a un punto conocido seleccionando indicandolo por su etiqueta.
203      * Usa los metodos auxiliares de busqueda de puntos y calculo de distancias entre coordenadas.
204      * @param etiquetaPuntoConocido
        
```

Unnecessary imports should be removed
Unused assignments should be removed
Source files should not have any duplicated blocks

Informe de Calidad (Sprint 2)

Autores: LAUREANO ATECA

La issue de severidad informativa con una deuda técnica de 1 minuto:

SonarCloud interface showing a project analysis for "Ingeniería del Software UC" and "AppGasolinerasGrupo6". The left sidebar shows filters for Type, Severity, and Rule. The main panel displays a code snippet from "src/main/java/com/example/gasolineras/MainActivity.java" with a highlighted issue: "Replace the type specification in this constructor call with the diamond operator (<code></code>). Why is this an issue?". The issue is categorized as "Code Smell" with a severity of "Info" and a technical debt of "1min effort".

SonarCloud interface showing a project analysis for "Ingeniería del Software UC" and "AppGasolinerasGrupo6". The left sidebar shows filters for Type, Severity, and Rule. The main panel displays a code snippet from "src/main/java/com/example/gasolineras/MainActivity.java" with a highlighted issue: "Replace the type specification in this constructor call with the diamond operator (<code></code>). Why is this an issue?". The issue is categorized as "Code Smell" with a severity of "Info" and a technical debt of "1min effort".

Informe de Calidad (Sprint 2)

Autores: LAUREANO ATECA

PLAN DE ACCIÓN

- 1) Corregir en la medida de lo posible los code smells por código repetido que tienen una deuda técnica de 20 minutos cada uno. La duplicación se encuentra en las activities MainActivity y UrgenciaActivity.
- 2) Eliminar el code smell de severidad menor de la clase PresenterGasolineras y que tiene una deuda técnica de 2 minutos (método calcularDistanciaEntrePuntoYGasolinera()).
- 3) Eliminar el code smell de severidad informativa de la MainActivity con una deuda técnica de 1 minuto (método onCreate()).

Comentarios:

- No se eliminarán el resto de issues por el momento, su corrección puede poner en riesgo el funcionamiento de la App. Son issues por código obsoleto.
- Los criterios de calidad pasan, pero debemos realizar planes de acción preventivos para que la situación siga siendo satisfactoria y que la calidad del producto mejore.