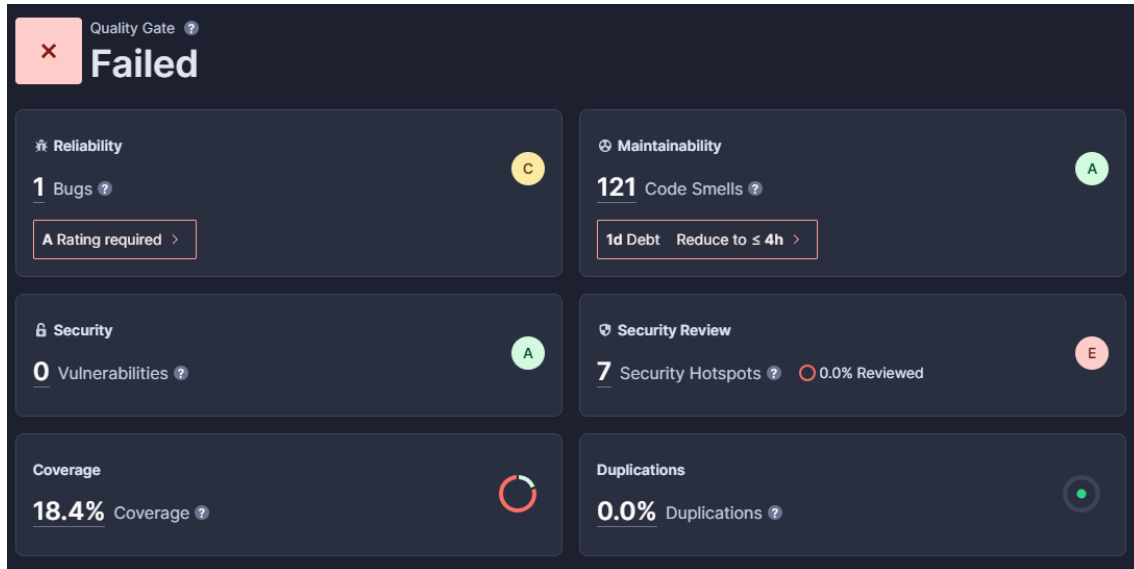


INFORME DE CALIDAD – GRUPO 3 – SPRINT 3

Fecha de Realización: 14 de Noviembre de 2023

ANÁLISIS GENERAL DEL CÓDIGO

Hash del commit: [7a3d06f]



En cuanto al análisis general podemos comprobar que los puntos más importantes según la valoración a mejorar para lograr pasar el quality gate serían:

- Reliability(C): ya que este posee una valoración de C y requiere una valoración de A.
 - Solución: eliminar el único bug ya que deben de ser < 1.
- Maintainability(A): ya que este tiene una deuda técnica de 1d y ha de reducirse a <= 4 horas.
 - Solución: reducir número de code smells(121) hasta lograr la deuda requerida.

En respecto al resto de los apartados se puede comprobar que son correctos para poder pasar el quality gate:

- Security: hay un total de 0 vulnerabilities.
- Security Review: a pesar de tener una valoración de E, debido a los 7 security hotspots, que no es obligatorio solucionar.
- Coverage: se comprueba 18.4% del código cubierto por pruebas.
- Duplications: se comprueba que no existen duplicados en el código (0.0%)

A continuación, se realiza un análisis específico de cada apartado:

RELIABILITY

Las siguientes capturas son los fallos de la sección de reliability.

Intentionality issue | Not complete

Either re-interrupt this method or rethrow the "InterruptedException" that can be caught here. [↗](#)

"InterruptedException" and "ThreadDeath" should not be ignored [java:S2142](#)

Software qualities impacted: Reliability

Open

Not assigned

Bug

Major

Tags

cwe

Line affected

L391

Effort

15 min

Introduced

6 days ago

```
} catch (IOException | InterruptedException e) {
```

Either re-interrupt this method or rethrow the "InterruptedException" that can be caught here.

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
Medium	MainView.java	Fallo	15 minutos	Excepción que ha de ser relanzada ya que se está esperando al hilo y no se maneja adecuadamente la excepción

MANTAINABILITY



→ Se encuentra un total de 121 smells, éstos divididos por severidad se localizan 3 high, 76 medium y 42 low.

→ Obligatoriamente se cubren los smells que tengan una severidad high, sin tener cuenta el tiempo de effort:

src/.../carchargers/activities/details/DetailsView.java

Adaptability issue

Refactor this method to reduce its Cognitive Complexity from 41 to the 30 allowed.

brain-overload

Open

Not assigned

Maintainability

Code Smell

Critical

16min effort • 15 days ago

```
@Override
protected void onCreate(Bundle savedInstanceState) {
```

Refactor this method to reduce its Cognitive Complexity from 41 to the 30 allowed.

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
High	DetailsView.java	Smell	16 minutos	Se ha de refactorizar el método para reducir su complejidad hasta 30

src/.../carchargers/repository/service/FakeCall.java

Intentionality issue

Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation. suspicious

☐ Open ☐ Not assigned ☒ Maintainability Code Smell Critical 5min effort · 1 month ago

Consistency issue

Remove this "clone" implementation; use a copy constructor or copy factory instead. suspicious

☐ Open ☐ Not assigned ☒ Maintainability Code Smell Blocker 30min effort · 1 month ago

```
@Override
public void cancel() {
```

Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation.

```
@Override
public Call<List<Charger>> clone() {
```

Remove this "clone" implementation; use a copy constructor or copy factory instead.

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
High	FakeCall.java	Smell	5 minutos	Añadir un comentario explicando el método vacío y lanzar la excepción solicitada.
High	FakeCall.java	Smell	30 minutos	Suprimir la implementación del clone y utilizar el constructor

→ Aparte de los smells de severidad alta se cubren los smells de severidad media y baja que requieren un mayor tiempo de effort (superior o igual a 15 minutos) con el fin de tratar de superar el quality gate:

src/.../carchargers/repository/service/APIArguments.java

Intentionality issue

Return an empty array instead of null. cert

☐ Open ☐ Not assigned ☒ Maintainability Code Smell Major 30min effort · 6 days ago

```
public Integer[] getOperatorIds() {
    if (operatorIds.size() == 0) {
        return null;
    }
}
```

Return an empty array instead of null.

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
Medium	APIArguments.java	Smell	30 minutos	Retornar un array vacío en lugar de null

src/.../java/es/unican/carchargers/OnApiArgsTest.java

Adaptability issue

Refactor this method to reduce the number of assertions from 35 to less than 25.

tests

☐ Open ☐ Not assigned ☒ Maintainability ☐ Code Smell ☐ Major

20min effort · 6 days ago

```
@Test
public void onApiArgsTest() {
```

Refactor this method to reduce the number of assertions from 35 to less than 25.

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
Medium	OnApiArgsTest.java	Smell	20 minutos	Refactorizar el método para reducir el numero de asserts hasta < 25

Intentionality issue

Remove this expression which always evaluates to "true"

cwe redundant ...

☐ Open ☐ Not assigned ☒ Maintainability ☐ Code Smell ☐ Major

10min effort · 26 days ago

```
    } else if (o1 == null && o2 == null) {
```

Remove this expression which always evaluates to "true"

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
Medium	LocationComparator.java	Smell	10 minutos	Eliminar una condición que siempre es verdadera

Consistency issue

Remove this use of "getParcelable"; it is deprecated.

cert cwe ...

☐ Open ☐ Not assigned ☒ Maintainability ☐ Code Smell ☐ Info

15min effort · 28 days ago

```
// Get Charger from the intent that triggered this activity
charger = Parcels.unwrap(getIntent().getExtras().getParcelable(INTENT_CHARGER));
```

Remove this use of "getParcelable"; it is deprecated.

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
Low	ConfigView.java	Smell	15 minutos	Eliminar el uso de la función ya que esta, no tiene uso y seguramente sea eliminada en un futuro.

src/.../es/unican/carchargers/model/Charger.java

Intentionality issue

Override "equals(Object obj)" to comply with the contract of the "compareTo(T o)" method.

No tags

☐ Open

☐ Not assigned

☒ Maintainability

☐ Code Smell

☐ Minor

15min effort · 28 days ago

@Override

public int compareTo(Charger other) {

Override "equals(Object obj)" to comply with the contract of the "compareTo(T o)" method.

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
Low	Charger.java	Smell	15 minutos	Realizar un override del método equals() junto con el compareTo()

src/.../es/unican/carchargers/model/ConnectionType.java

Intentionality issue

Override "equals(Object obj)" to comply with the contract of the "compareTo(T o)" method.

No tags

☐ Open

☐ Not assigned

☒ Maintainability

☐ Code Smell

☐ Minor

15min effort · 15 days ago

@Override

public int compareTo(ConnectionType other) {

Override "equals(Object obj)" to comply with the contract of the "compareTo(T o)" method.

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
Low	ConnectionType.java	Smell	15 minutos	Realizar un override del método equals() junto con el compareTo()

src/.../es/unican/carchargers/model/MediaItem.java

Intentionality issue

Override "equals(Object obj)" to comply with the contract of the "compareTo(T o)" method.

No tags

☐ Open

☐ Not assigned

☒ Maintainability

☐ Code Smell

☐ Minor

15min effort · 3 hours ago

@Override

public int compareTo(MediaItem other) {

Override "equals(Object obj)" to comply with the contract of the "compareTo(T o)" method.

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
Low	MediaItem.java	Smell	15 minutos	Realizar un override del método equals() junto con el compareTo()

src/.../carchargers/repository/service/FakeCall.java

Intentionality issue

Use `super.clone()` to create and seed the cloned instance to be returned.

cert convention ...

Open Not assigned Maintainability Code Smell Minor

20min effort • 1 month ago

```
@Override
public Call<List<Charger>> clone() {
```

Use `super.clone()` to create and seed the cloned instance to be returned.

Severidad	Clase	Fallo/Smell	Deuda Técnica	Descripción
Low	FakeCall.java	Smell	20 minutos	Utilizar la superclase para crear la instancia clonada

SECURITY

→ No se han encontrado vulnerabilidades en cuanto a la seguridad.

SECURITY REVIEW

→ 7 Security Hotspots que pueden provocar vulnerabilidades, pero no se requiere solucionarlos.

COVERAGE

→ 18.4% del código se encuentra cubierto en el apartado de tests.

DUPLICATIONS

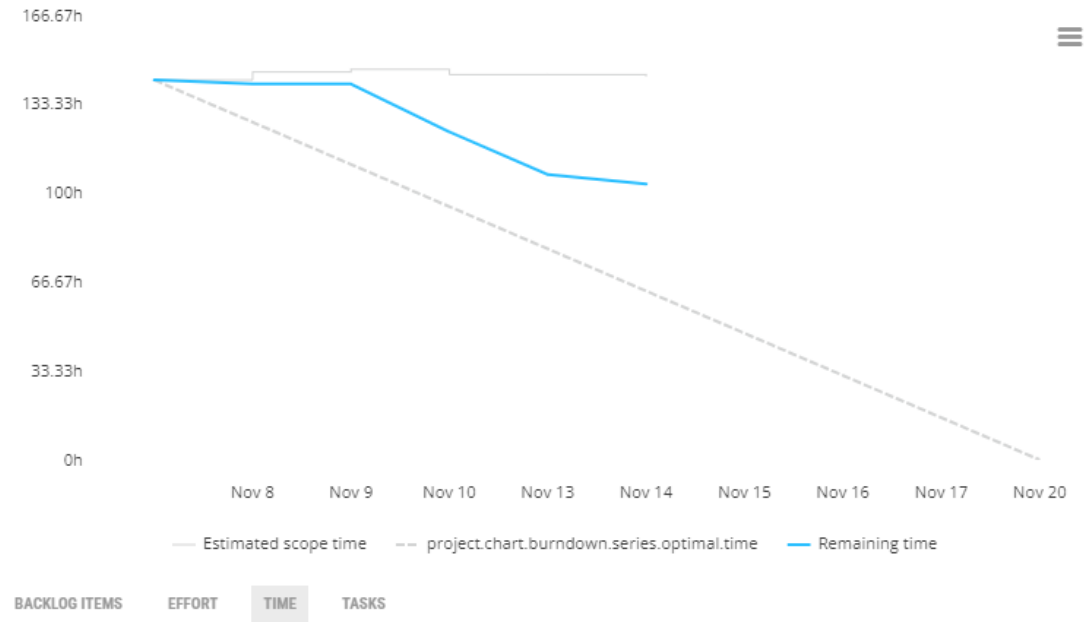
→ El apartado de código duplicado es correcto ya que no hay nada duplicado.

SPRINT BURNDOWN CHART

Sprint Overview

Planned			Completed		
Items	Effort	Time	Items	Effort	Time
7	11	144	2	2	6.5

Burndown Chart



Análisis: Como se puede comprobar en el chart, el grupo va con una proyección de trabajo negativa en comparación a la ideal, por lo que no se dispone de mucho tiempo para poder añadir tareas que cubran la mayor parte de este análisis previo. Entonces se llevará a cabo una tarea en la que únicamente se cubrirán los errores que tengan un nivel de severidad alto.

PLAN DE ACCIÓN

Nombre	Clase	Descripción	Acción
Either re-interrupt this method or rethrow the "InterruptedException" that can be caught here.	MainView.java	No se maneja de forma adecuada la excepción.	Eliminar el catch y relanzarla usando throw para enviarla al método superior.
Refactor this method to reduce its Cognitive Complexity from 41 to the 30 allowed.	DetailsView.java	El nombre del método aumenta la complejidad cognitiva.	Refactorizar el nombre de este método para reducirlo hasta 30.
Add a nested comment explaining why this method is empty, throw an UnsupportedOperationException or complete the implementation.	FakeCall.java	El método está vacío.	Añadir un comentario explicando el por qué y lanzar la excepción solicitada.
Remove this "clone" implementation; use a copy constructor or copy factory instead.	FakeCall.java	Se debe eliminar la implementación del método.	Sustituir este por una copia del constructor.
Return an empty array instead of null.	APIArguments.java	Devuelve valor null en lugar de un array sin contenido.	Modificar el método para que retorne un array vacío.
Refactor this method to reduce the number of assertions from 35 to less than 25.	OnApiArgsTest.java	El método genera muchas assertions.	Refactorizarle para reducir este cantidad hasta < 25.
Remove this use of "getParcelable"; it is deprecated.	ConfigView.java	El método está deprecated.	Eliminarlo porque ya no se usa y puede ser eliminada en el futuro.
Override "equals(Object obj)" to comply with the contract of the "compareTo(T o)" method.	Charger.java	No cumple con el contrato del método compareTo().	Usar override para que pueda cumplirlo.
Override "equals(Object obj)" to comply with the contract of the "compareTo(T o)" method.	ConnectionType.java	No cumple con el contrato del método compareTo().	Usar override para que pueda cumplirlo.
Override "equals(Object obj)" to comply with the contract of the "compareTo(T o)" method.	Medialtem.java	No cumple con el contrato del método compareTo().	Usar override para que pueda cumplirlo.
Use super.clone() to create and seed the cloned instance to be returned.	FakeCall.java	Se usa un método clone() con una implementación definida en esa clase.	Usar la superclase para crear la instancia clonada.

Corrigiendo todos estos errores, obtenemos una mejora en la deuda de:

$15 + 16 + 5 + 30 + 30 + 20 + 15 + 15 + 15 + 15 + 20 = 196$ minutos, algo más de 3 horas.

Por cada acción en este plan, se añadirá una tarea en este sprint para solucionarlo, además, la elección de las acciones se ha realizado teniendo en cuenta la severidad de los errores y el tiempo restante del Sprint observando el Sprint Burndown Chart.