

## Reporte SonarQube

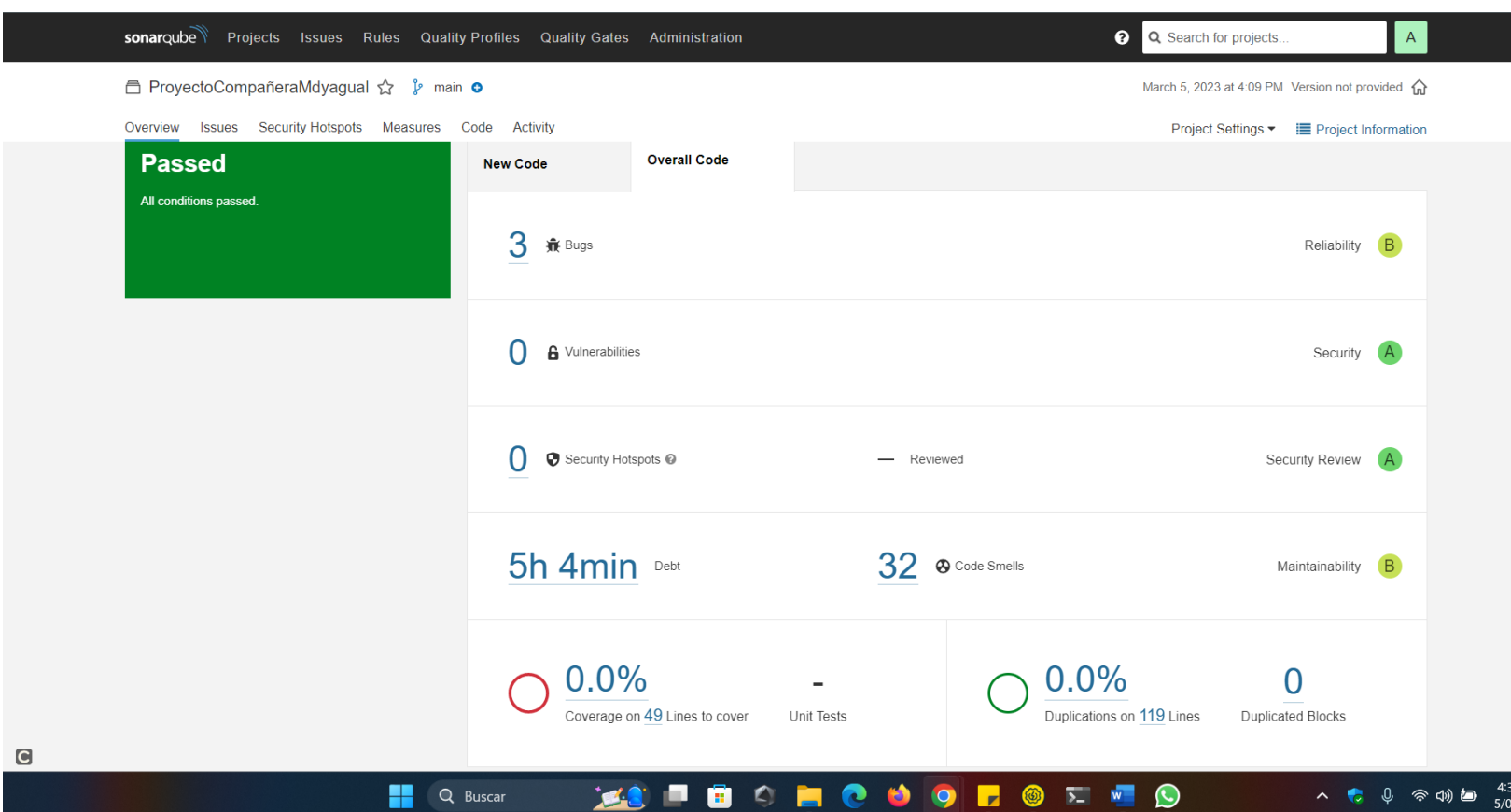
Para realizar este reporte se utilizó el siguiente repositorio que pertenece al usuario **mdyagual** de github.

Este es el enlace del repositorio: <https://github.com/training-practice-sofkau/C1-2023-WS1-Java>

Como primer paso es abrir la herramienta de SonarQube y se pone a funcionar, una vez en la página SonarQube procedemos a crear un nuevo proyecto con el siguiente nombre: "ProyectoCompañeraMdyagua" y se obtiene el token. Después de esto se procede a clonar el repositorio del usuario y una vez clonado abrimos el CMD en esta ubicación y pegamos el token y damos enter.

Una vez se muestra el siguiente mensaje "EXECUTION SUCCESS" podemos revisar el análisis que ha realizado SonarQube.

Obteniendo como resultado lo siguiente:



A screenshot of the SonarQube web interface. At the top, the project name 'ProyectoCompañeraMdyagual' is displayed with a folder icon, followed by a star icon, a branch icon, and the text 'main' with a plus icon. Below this is a navigation bar with tabs: 'Overview', 'Issues' (which is underlined in blue), 'Security Hotspots', 'Measures', 'Code', and 'Activity'. Below the tabs is a horizontal bar with a light blue gradient.

También podemos ver porque es un problema:

Si ese es el caso, no deberían ocurrir importaciones no utilizadas e inútiles.

Dejarlos reduce la legibilidad del código, ya que su presencia puede ser confusa.

Se evidencia en la siguiente imagen otros problemas que también nos muestra:

☐ Cambio masivo 1 / 35 problemas 5h 19min de esfuerzo

src/co/com/movingu/MovingUApp.java

<input type="checkbox"/>	Elimine esta importación no utilizada 'co.com.movingu.vehicle.Scooter'.	Hace 1 mes ▾ L 7 🔗				
🚩 Código Olor	🟢 Menor	🔵 Abierto	No asignado ▾	2 minutos de esfuerzo	Comentario	🔗 no usado ▾
<input type="checkbox"/>	Mueva el contenido de este inicializador a un constructor estándar oa inicializadores de campo.	Hace 1 mes ▾ L 15 🔗				
🚩 Código Olor	🔴 Importante	🔵 Abierto	No asignado ▾	esfuerzo de 30 minutos	Comentario	🔗 trampa ▾
<input type="checkbox"/>	Use otra forma de inicializar esta instancia.	Hace 1 mes ▾ L 20 🔗				
🚩 Bicho	🟢 Menor	🔵 Abierto	No asignado ▾	5 minutos de esfuerzo	Comentario	🔗 filtración ▾
<input type="checkbox"/>	Defina una constante en lugar de duplicar este literal "0976152443" 3 veces.	Hace 1 mes ▾ L 16 🔗				
🚩 Código Olor	🔴 Crítico	🔵 Abierto	No asignado ▾	8 minutos de esfuerzo	Comentario	🔗 diseño ▾
<input type="checkbox"/>	Mueva el contenido de este inicializador a un constructor estándar oa inicializadores de campo.	Hace 1 mes ▾ L 25 🔗				
🚩 Código Olor	🔴 Importante	🔵 Abierto	No asignado ▾	esfuerzo de 30 minutos	Comentario	🔗 trampa ▾
<input type="checkbox"/>	Elimine esta asignación inútil a la variable local "vehiculos".	Hace 1 mes ▾ L 37 🔗				
🚩 Código Olor	🔴 Importante	🔵 Abierto	No asignado ▾	15 minutos de esfuerzo	Comentario	🔗 certificado, cwe, sin usar ▾
<input type="checkbox"/>	Elimine esta variable local "vehiculos" no utilizada.	Hace 1 mes ▾ L 25 🔗				
🚩 Código Olor	🟢 Menor	🔵 Abierto	No asignado ▾	5 minutos de esfuerzo	Comentario	🔗 no usado ▾
<input type="checkbox"/>	Use otra forma de inicializar esta instancia.	Hace 1 mes ▾ L 37 🔗				
🚩 Bicho	🟢 Menor	🔵 Abierto	No asignado ▾	5 minutos de esfuerzo	Comentario	🔗 filtración ▾

También podemos observar los errores que más necesitan nuestra atención:

▼ Type

🚩 Bug	0
🔒 Vulnerability	0
🚩 Code Smell	2

▼ Severity CRITICAL Clear

🔴 Blocker	0	🟢 Minor	9
🔴 Critical	2	🔵 Info	0
🔴 Major	24		

Press Ctrl to add to selection

src/co/com/movingu/MovingUApp.java

<input type="checkbox"/>	Defina una constante en lugar de duplicar este literal "0976152443" 3 veces.	Hace 1 mes ▾ L 16 🔗				
🚩 Código Olor	🔴 Crítico	🔵 Abierto	No asignado ▾	8 minutos de esfuerzo	Comentario	🔗 diseño ▾
<input type="checkbox"/>	Agregue un caso predeterminado a este interruptor.	Hace 1 mes ▾ L 66 🔗				
🚩 Código Olor	🔴 Crítico	🔵 Abierto	No asignado ▾	5 minutos de esfuerzo	Comentario	🔗 certificado, cwe ▾

2 de 2 mostrados

También podemos ver los tipos de problemas que tenemos en esta sección:


















▼ Type CODE SMELL Clear

🚩 Bug	3
🔒 Vulnerability	0
🚩 Code Smell	32

Como podemos observar el proyecto tiene 3 bugs, 0 vulnerabilidades detectadas por SonarQube lo que no quiere decir que no existan y 32 códigos apesados.





Estos son los bugs que muestra SonarQube lo que no quiere decir que no existan más:

 src/co/com/movingu/MovingUApp.java

<input type="checkbox"/>	<b>Use otra forma de inicializar esta instancia.</b>	Hace 1 mes ▾ L 20  
	Bicho ▾  Menor ▾  Abierto ▾ No asignado ▾ 5 minutos de esfuerzo <a href="#">Comentario</a>	 filtración ▾
<input type="checkbox"/>	<b>Use otra forma de inicializar esta instancia.</b>	Hace 1 mes ▾ L 37  
	Bicho ▾  Menor ▾  Abierto ▾ No asignado ▾ 5 minutos de esfuerzo <a href="#">Comentario</a>	 filtración ▾
<input type="checkbox"/>	<b>Use otra forma de inicializar esta instancia.</b>	Hace 1 mes ▾ L 44  
	Bicho ▾  Menor ▾  Abierto ▾ No asignado ▾ 5 minutos de esfuerzo <a href="#">Comentario</a>	 filtración ▾

3 de 3 mostrados

En la siguiente pestaña Medidas podemos encontrar lo siguiente:

 ProyectoCompañeraMdyagual   principal 

Descripción general   Asuntos   Puntos de acceso de seguridad   **Medidas**   Código   Activos

Descripción del proyecto

ProyectoCompañeraMdyagual

📁 ProyectoCompañeraMdyagual ☆

Descripción general   Asuntos   Puntos de acc

Descripción del proyecto

▼ **Fiabilidad** ?

Descripción general 🔑

En general

Insectos 3

**Clasificación** B

Esfuerzo de remediación 15 minutos

▼ **Seguridad** ?

Descripción general 🔑

En general

vulnerabilidades 0

Clasificación A

Esfuerzo de remediación 0

> **Revisión de seguridad** ?

> **mantenibilidad** ?

> **Cobertura**

> **Duplicaciones**


> **Tamaño**




> **Complejidad** ?


Podemos ver que la clasificación es de B para el proyecto en cuestión de fiabilidad y de A en términos de Seguridad

" La calificación de confiabilidad es B cuando hay al menos un error menor" esto lo dice la herramienta de SonarQube.

También podemos observar las siguientes medidas:

<b>▼ Duplicaciones</b>	
Descripción general	
En general	
Densidad	0,0%
Líneas Duplicadas	0
Bloques duplicados	0
Archivos duplicados	0
<b>▼ Tamaño</b>	
Líneas de código	119
Líneas	200
Declaraciones	37
Funciones	13
Clases	9
archivos	9
Líneas de comentario	30
Comentarios (%)	20,1%

<b>▼ mantenibilidad</b> 	
Descripción general	
En general	
El código huele	32
Deuda	5h 4min
Ratio de deuda	8,5%
Clasificación	<b>B</b>
Esfuerzo para llegar a A	2h 6min
<b>▼ Cobertura</b>	
Descripción general	
En general	
Cobertura	0,0%
Líneas a Cubrir	49
Líneas descubiertas	49
Cobertura de línea	0,0%

<b>▼ Complejidad</b> 	
Complejidad ciclomática	dieciséis
Complejidad cognitiva	7
<b>▼ Asuntos</b>	
Asuntos	35
Problemas abiertos	35
Problemas reabiertos	0
Problemas confirmados	0
Problemas de falsos positivos	0
No solucionará problemas	0

5h 4min Debt

32  Code Smells

Maintainability 

Acá podemos observar la "deuda de tiempo" en SonarQube que nos indica que es una métrica que mide la cantidad de tiempo adicional que se requiere para corregir los problemas técnicos acumulados en un proyecto de software.