



Proyecto EduShare

Reporte del SonarQube

Elaborado por:

Juan Eduardo Cumplido Negrete

Erick Abdiel Atzin Olarte

Christian Alberto Vásquez Cruz

Equipo: CAPA 8

Fecha de emisión: 16 de mayo de 2025

1. Introducción

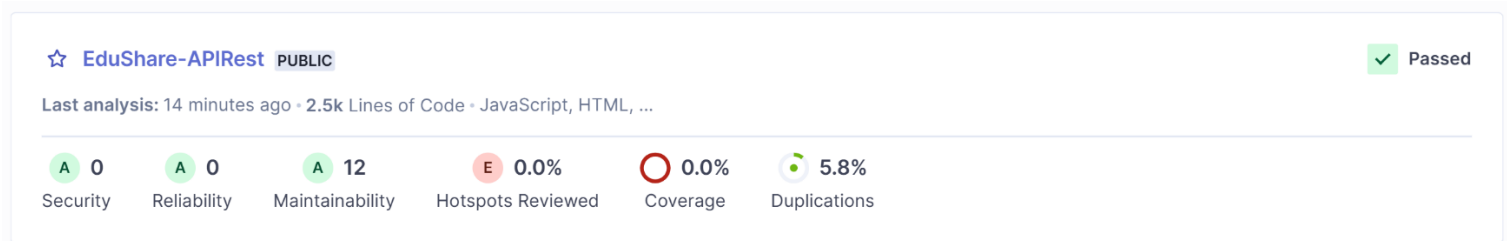
El análisis estático de código es una práctica fundamental y complementaria en el desarrollo de software. El cual permite revisar y evaluar el código fuente sin tener que ejecutarlo. Esto permite identificar errores lógicos, posibles vulnerabilidades de seguridad, malas prácticas de codificación y problemas de rendimiento antes de la ejecución. Reduciendo costos y esfuerzos en fases posteriores del desarrollo.

Además, ayuda a mantener estándares de calidad al identificar violaciones de estilos de codificación, redundancias o código no utilizado, fomentando un código más limpio, legible y mantenible. El detectar vulnerabilidades como inyecciones de código, errores de manejo de datos sensibles y desbordamiento de memoria, lo que ayuda a proteger el sistema contra posibles ataques.

Por último, es importante destacar que ayuda a reducir la probabilidad de errores críticos que puedan impactar negativamente en los usuarios o clientes finales, además del ahorro de tiempos en las revisiones manuales y permite tener un enfoque de los desarrolladores en aspectos más complejos del desarrollo del sistema.

Dentro de este reporte de análisis de código se comparten los resultados, así como documentación de aquellos falsos positivos marcados por la herramienta SonarQube la cual realizó un análisis estático de la solución Edushare-APIRESTful y EduShare-Escritorio, junto con una breve descripción del por qué son falsos positivos, cabe destacar que algunas imágenes tienen la misma razón por lo que se optó por juntarlas y poner una justificación general.

2. Resultado del análisis estático de la API



Durante el análisis de calidad de código realizado con SonarQube sobre el proyecto EduShare-APIRest, se obtuvieron resultados satisfactorios en los principales indicadores de calidad, tales como:

- Seguridad: A (0 issues)
- Fiabilidad: A (0 issues)
- Mantenibilidad: A (12 code smells, sin deuda técnica significativa)
- Hotspots de seguridad: 0%
- Duplicación de código: 5.8%

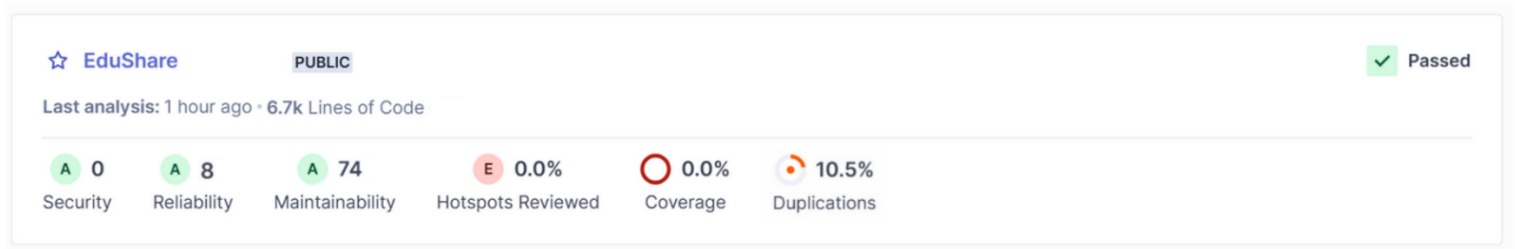
Sin embargo, el indicador de cobertura de código por pruebas automatizadas (Coverage) aparece con un valor de 0.0%, lo cual no refleja con precisión la realidad de nuestro entorno de pruebas.

Aclaración sobre la cobertura

Nuestra API cuenta con un conjunto robusto de pruebas automatizadas que actualmente logran una cobertura del 77% del código fuente, medida mediante herramientas externas como Jest (en entornos JavaScript/Node.js). No obstante, esta cobertura no fue reconocida por SonarQube debido a que el reporte generado por las pruebas no fue correctamente vinculado o referenciado durante el análisis.

Estamos en proceso de configurar la integración adecuada mediante la especificación de la ruta del reporte de cobertura (por ejemplo, lcov.info o coverage.opencover.xml) dentro del archivo de configuración del análisis (sonar-project.properties), con el objetivo de que SonarQube pueda reflejar con fidelidad este indicador en futuros análisis.

3. Resultado del análisis estático del cliente



Como parte del proceso de aseguramiento de calidad del software, se llevó a cabo un análisis estático del código fuente del proyecto EduShare-Android utilizando la herramienta SonarQube. A continuación, se detallan los resultados obtenidos en los principales indicadores evaluados:

Resultados generales:

- Seguridad: Calificación A – No se detectaron vulnerabilidades en el código.
- Fiabilidad: Calificación A – Se identificaron 8 hallazgos menores, sin impacto crítico en el comportamiento del sistema.
- Mantenibilidad: Calificación A – Se encontraron 74 observaciones relacionadas con buenas prácticas, sin generar deuda técnica significativa.
- Hotspots de seguridad: 0.0% – No se identificaron puntos críticos que requieran atención especial en cuanto a seguridad.
- Duplicación de código: 10.5% – Se observó un nivel moderado de duplicación, recomendable de optimizar para favorecer la reutilización y simplificación del mantenimiento.

Los resultados del análisis reflejan un código con altos estándares de calidad, con indicadores sólidos en materia de seguridad, fiabilidad y mantenibilidad. Se sugiere realizar una revisión sobre los elementos duplicados detectados, a fin de mejorar la eficiencia estructural del código y reducir posibles redundancias.

Este informe respalda que el proyecto **EduShare-Android** cumple con buenas prácticas de desarrollo y se encuentra en una condición técnica favorable para su mantenimiento y evolución futura.