**Mejoramiento de la Aplicación  
BILBO**



**Docente:** Patrick Cuadros Quiroga

**Curso:** Calidad y Pruebas de Software

**Código de curso**: SI-784

**Facultad:** Facultad de Ingeniería

**Escuela:** Escuela Profesional de Ingeniería de Sistemas (EPIS)

**Integrantes:**

* Medina López, Marcelo José – 2020066917
* Poma Manchego, Rene Manuel – 2017057491
* Zevallos Purca, Justin Zinedine – 2020066924

TACNA – PERÚ

2023 - I

Resumen

El presente proyecto radica en la implementación de un sistema de escritorio llamado “Bilbo” para la empresa con el mismo nombre, enfocado a la venta de libros, ante la necesidad de la empresa de llevar a cabo la venta y compra de libros, luego registrando los procesos de aquellas acciones en una base de datos. Cabe aclarar que este proyecto se encuentra con codificación ya desarrollada casi en su totalidad; sin embargo, el objetivo del proyecto se centra en la mejora de su desarrollo y desempeño, lo cual incluirá procedimientos que aseguran la calidad y control del software, tales como revisión de código, testeo del código, identificación del mayor número de errores posibles, el menor número de revisiones posibles evitando costos, y el aseguramiento de no cometer errores externos, implicando que no se efectúe la satisfacción del cliente.

Abstract

This project is based on the implementation of a system called "Bilbo" for the company with the same name, focused on the sale of books, given the company's need to carry out the sale and purchase of books, then recording the processes of those actions in a database. It should be clarified that this project is with coding already developed almost in its entirety; however, the objective of the project is focused on improving its development and performance, which will include procedures that ensure the quality and control of the software, such as code review, code testing, identification of the largest possible number of errors, the least possible number of revisions avoiding costs, and ensuring that no external errors are made, implying that customer satisfaction is not realized.

1. Antecedentes o introducción

- Se tiene en cuenta que, aparte de nuestro proyecto, existen más sistemas de bibliotecas, con posibles funciones parecidas, los cuales pueden llegar a servir de inspiración, pero una gran diferencia entre aquellos sistemas y el nuestro es que se puede vender libros en lugar de prestarlos, cambiando un poco esta tradición por así decirlo. Con el presente proyecto se pretende cubrir un área principal de una biblioteca, dónde se desea implementar un sistema el cual agilizará el proceso de la biblioteca, registrando los datos necesarios y realizando las consultas que el usuario desee. El proyecto consiste en buscar optimizar el servicio de una biblioteca, así mismo mantener un orden al momento de realizar el CRUD. Para ello usaremos el Design Thinking, dónde podremos: Empatizar, Definir, Idear, Prototipar y Testear.

2. Título

* El título del presente proyecto es “Mejoramiento de la Aplicación BILBO”.

3. Autores

* Marcelo José Medina López.
* Rene Poma Manchebo.
* Justin Zevallos Purca.

4. Planteamiento del problema

4.1. Problema

* La baja revisión de la codificación del sistema “Bilbo”.

4.2. Justificación

* La mejora de la experiencia del usuario, si los usuarios no están satisfechos con la biblioteca de escritorio, es importante mejorarla para que su experiencia sea más agradable y fácil de usar. Al mejorar la interfaz de usuario y la navegación, se puede hacer que la biblioteca sea más intuitiva para el usuario, además, permitiría la llegada de nuevos clientes interesados en el sistema.

4.3. Alcance

* El alcance de nuestro será dentro de la empresa y toda Tacna, haciendo que las personas noten la mejoría de la empresa, atrayéndolas como nuevos clientes.

5. Objetivos

5.1. General:

* La implementación y revisión de un sistema capaz de poner al día los procesos o funcionalidades de la biblioteca, mejorando significativamente la atención al cliente.

5.2. Específicos:

* Clasificar los libros por el catálogo que posean.
* Vender los libros de forma eficiente y ágil.
* Ofrecer un espacio agradable para que cada uno se centre solo en la lectura y también para congeniar con los demás, compartiendo opiniones.

6. Referentes teóricos

Diagramas de Casos de Uso:

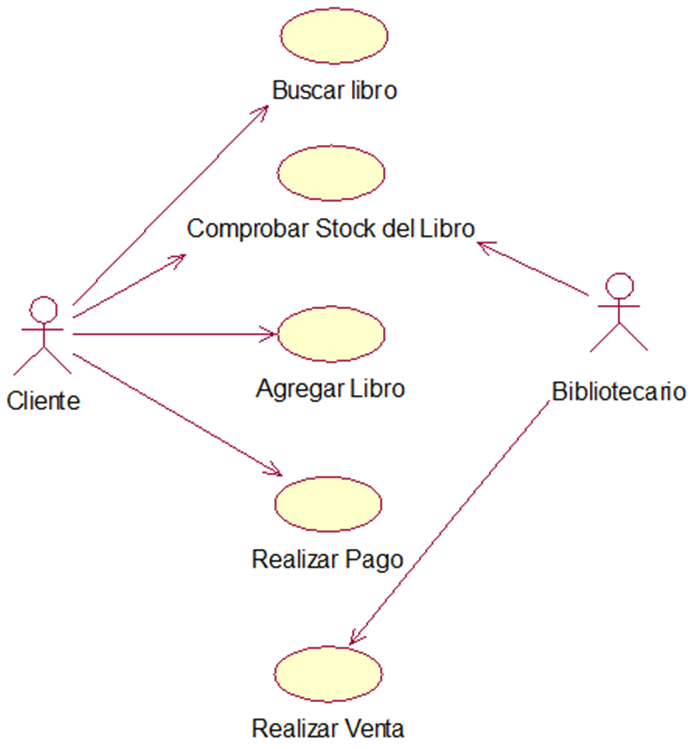
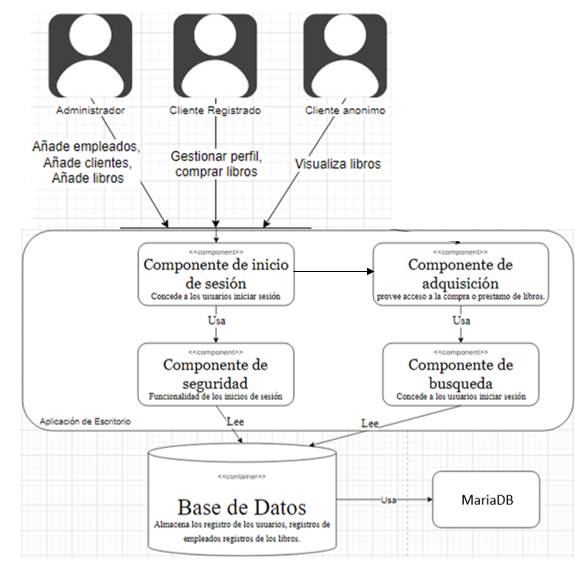


Diagrama de Componentes:



Arquitectura:

Diagrama Contextual:

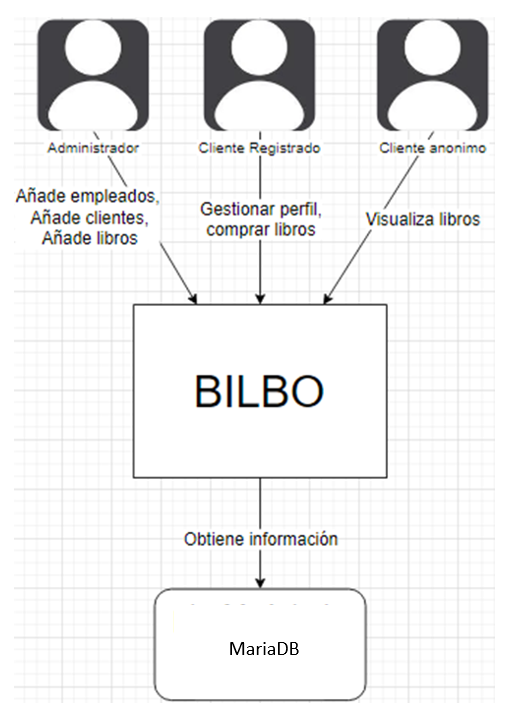
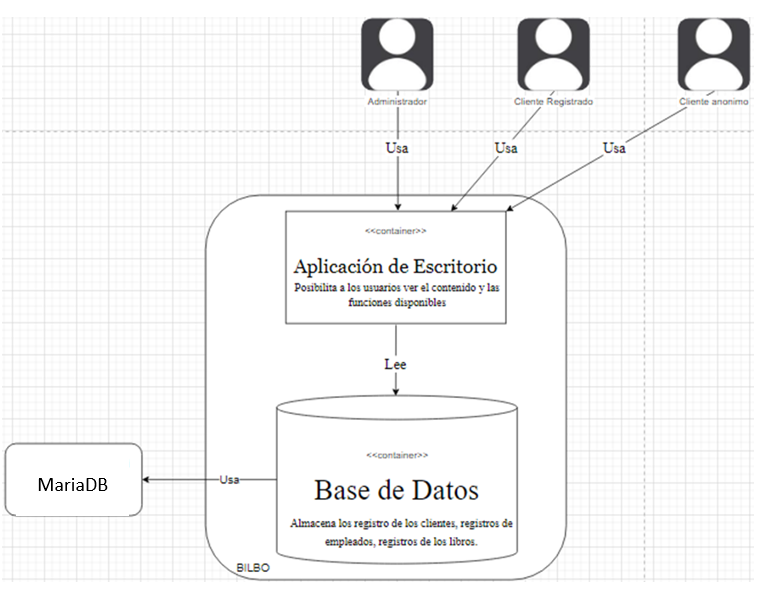


Diagrama de Contenedores:



7. Desarrollo de la propuesta

7.1. Tecnología de información:

* SonarQube: se utilizó para analizar y gestionar la calidad del código fuente del proyecto de software. Se trata de una herramienta que permite detectar problemas en el código, como errores, vulnerabilidades de seguridad, complejidad excesiva, falta de comentarios y otras cuestiones que afectan la calidad y mantenibilidad del software.

7.2. Metodología, técnicas usadas

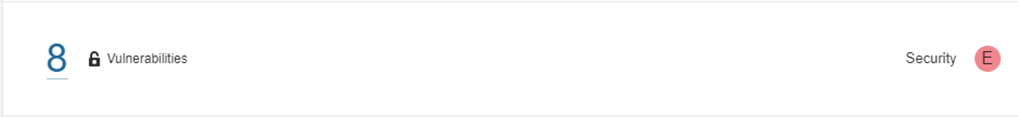
* Análisis estático de código: proceso automatizado que examina el código fuente sin necesidad de ejecutar el programa. En SonarQube, se utilizan diferentes herramientas y motores de análisis estático para identificar una amplia variedad de problemas en el código, como vulnerabilidades de seguridad, errores de sintaxis, problemas de estilo, complejidad excesiva y otros.
* Pruebas unitarias: pruebas automatizadas que se ejecutan sobre pequeñas secciones de código para verificar su correcto funcionamiento. Estas pruebas permiten detectar errores y problemas en el código, y pueden integrarse con SonarQube para proporcionar información adicional sobre la calidad del software.

8. Cronograma basado en las observaciones que la herramienta SonarQube les informará sobre la aplicación, a fin de reducir la deuda técnica, vulnerabilidades, fallas, etc. a 0.

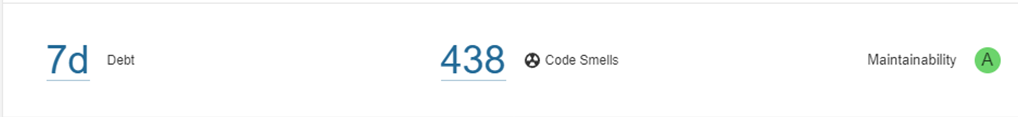
* Las observaciones que nos arrojó el sonarQube son las siguientes:
* 1 bug por lo que nos califica con una “B” en fiabilidad.



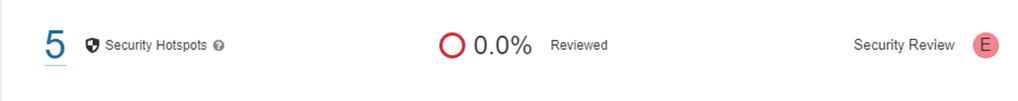
* Nuestro proyecto arrojó 8 vulnerabilidades dándonos una calificación de “E” en seguridad.



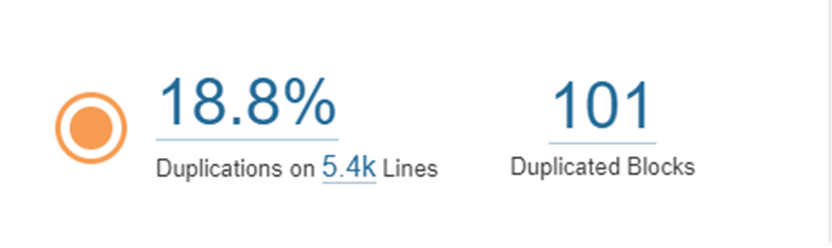
* Sonar nos calificó con 438 de deuda técnica y aproximadamente 7d de reparo, dándonos una calificación de “A” en mantenibilidad.



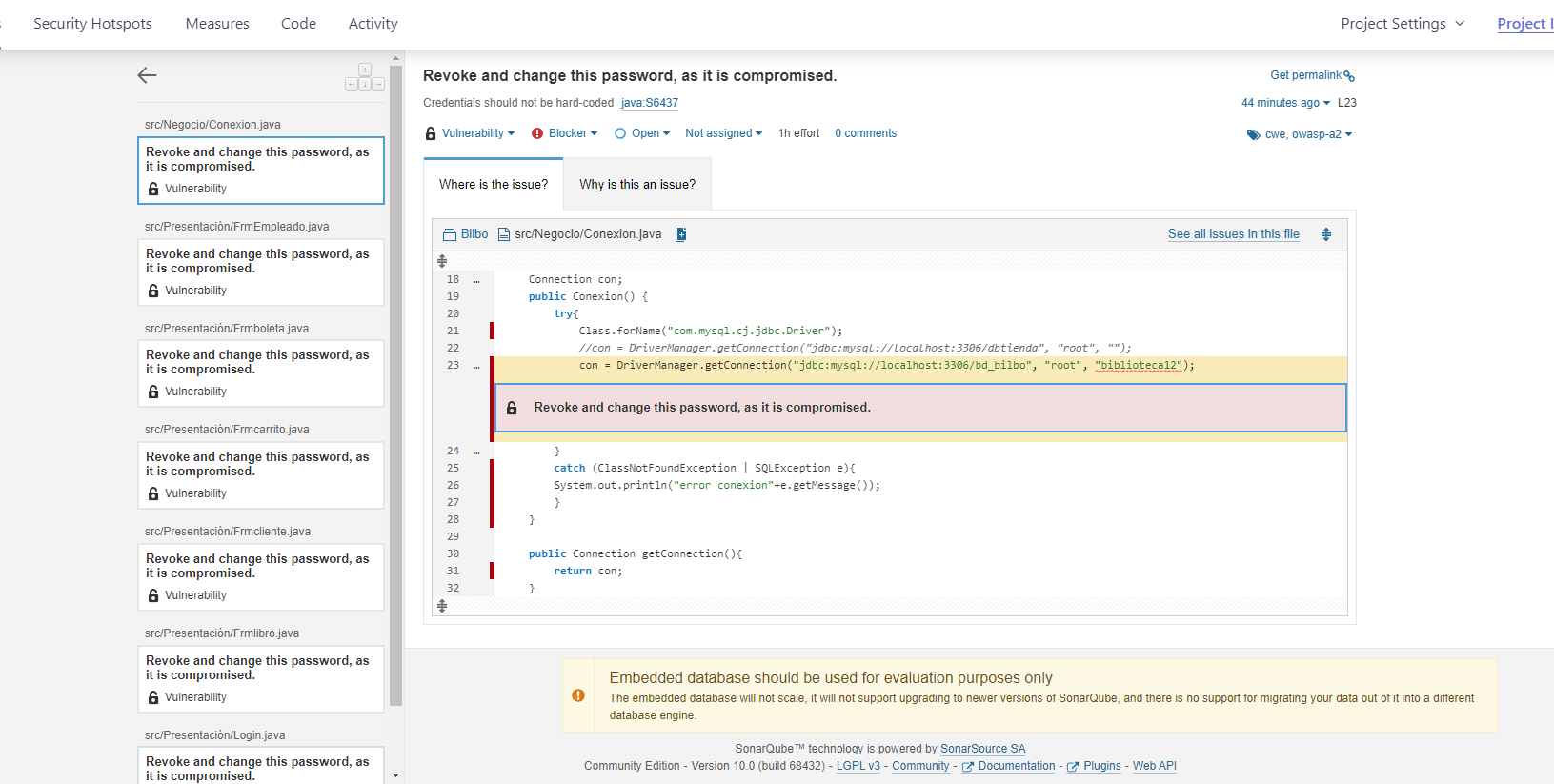
* En los puntos de seguridad tenemos 5 a corregir, sonar nos calificó con “E” en la reseña de seguridad.

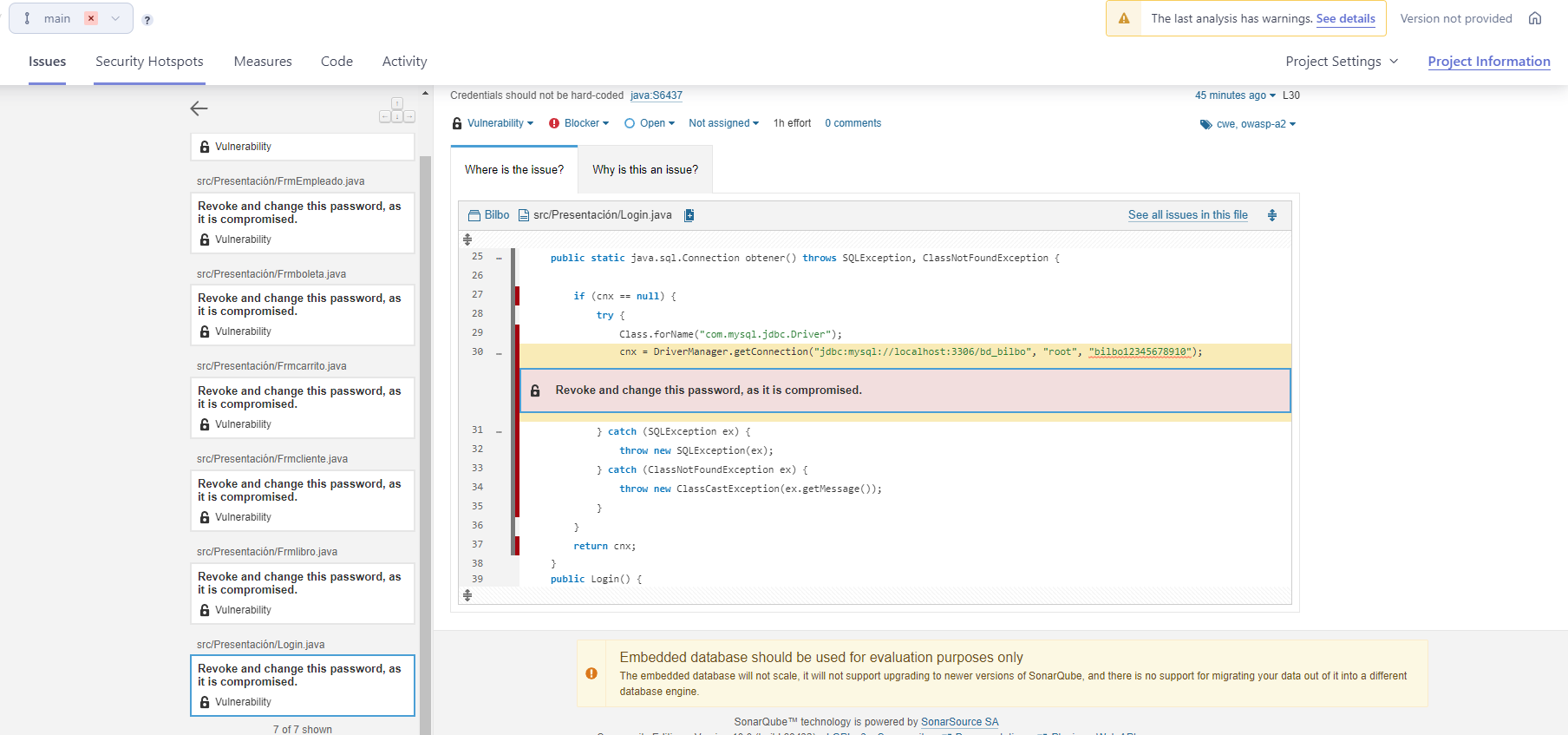


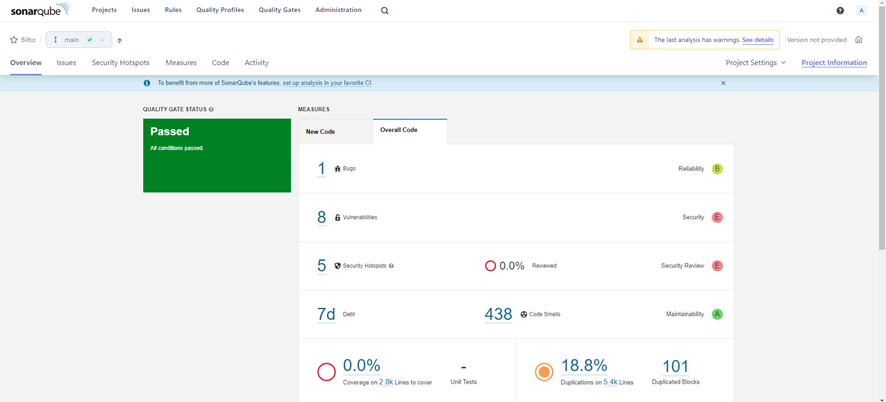
* Sonar nos mostró el mensaje de cobertura en 2.8k líneas, calificando con un 0.0% y la duplicación en 5.4k líneas, calificando con un 18.8%.



Resultado 27/04/2023:

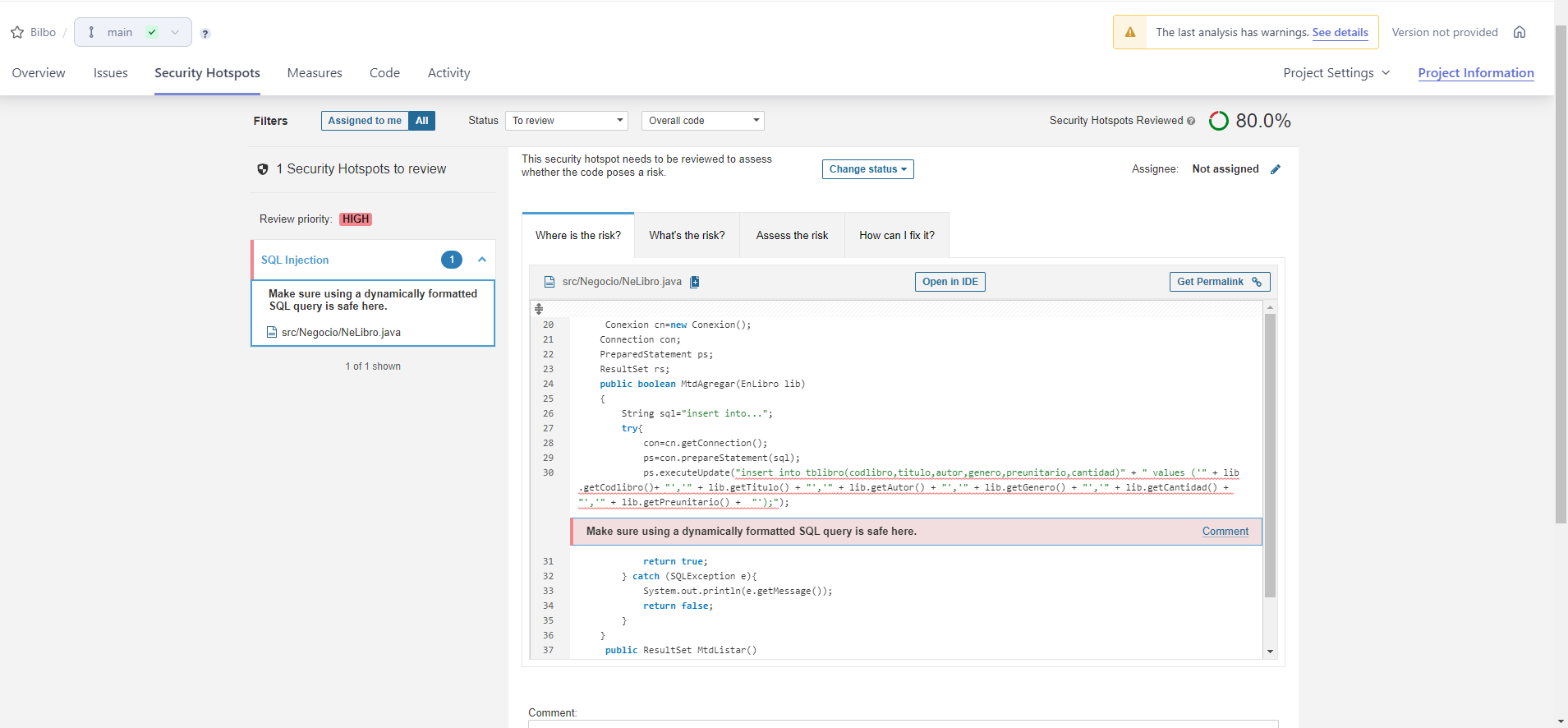


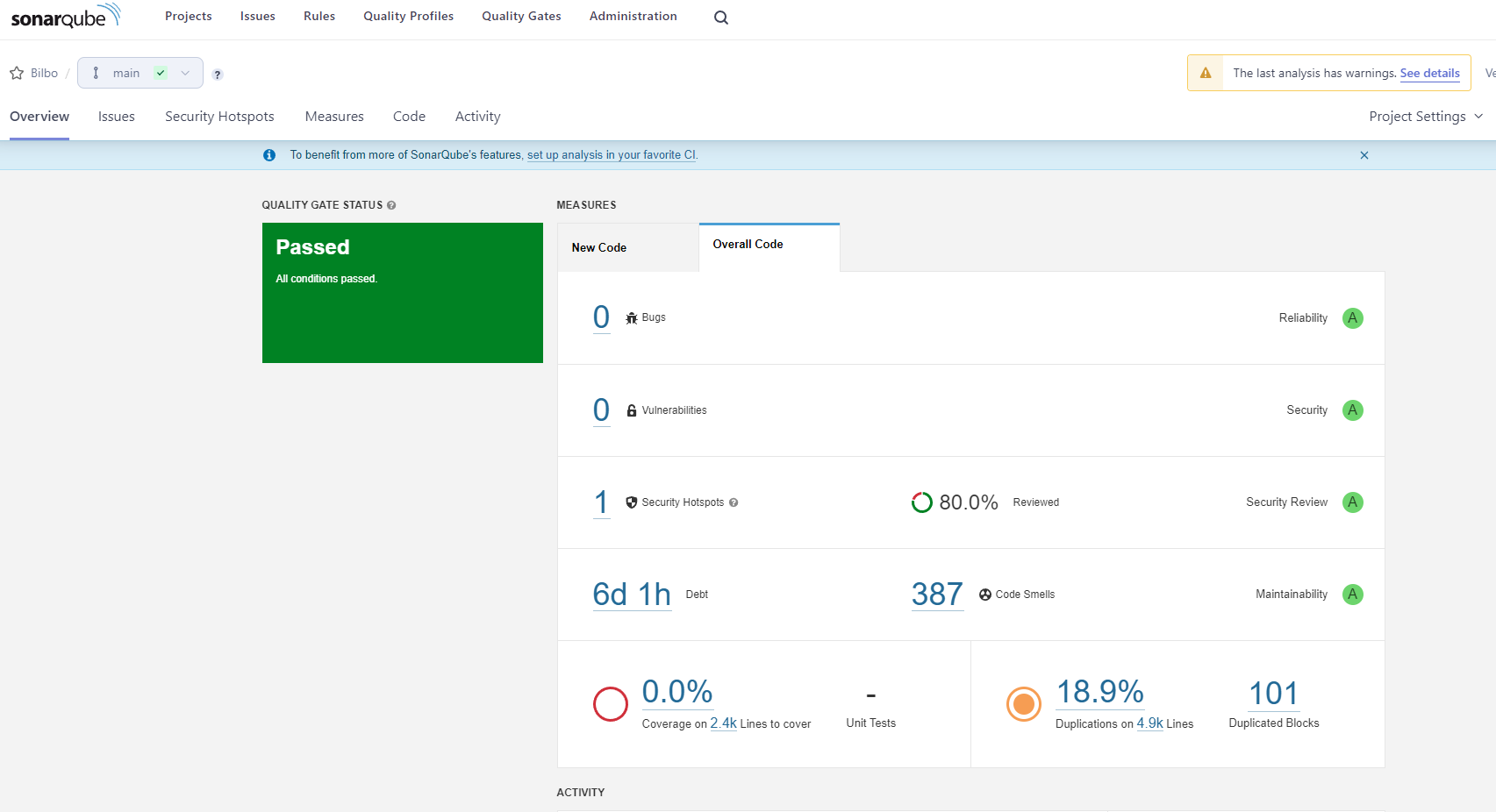




|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Día 1 | Día 2 | Día 3 | Día 4 | Día 5 | Día 6 | Día 7 | TOTAL |
| Vulnerabilidad | 2 | 2 |  |  | 2 | 2 |  | 8 vulnerabilidades corregidas |
| Hotspots |  |  | 2 | 2 |  |  | 1 | 5 hotspot corregidos |

Resultado 01/05/2023:

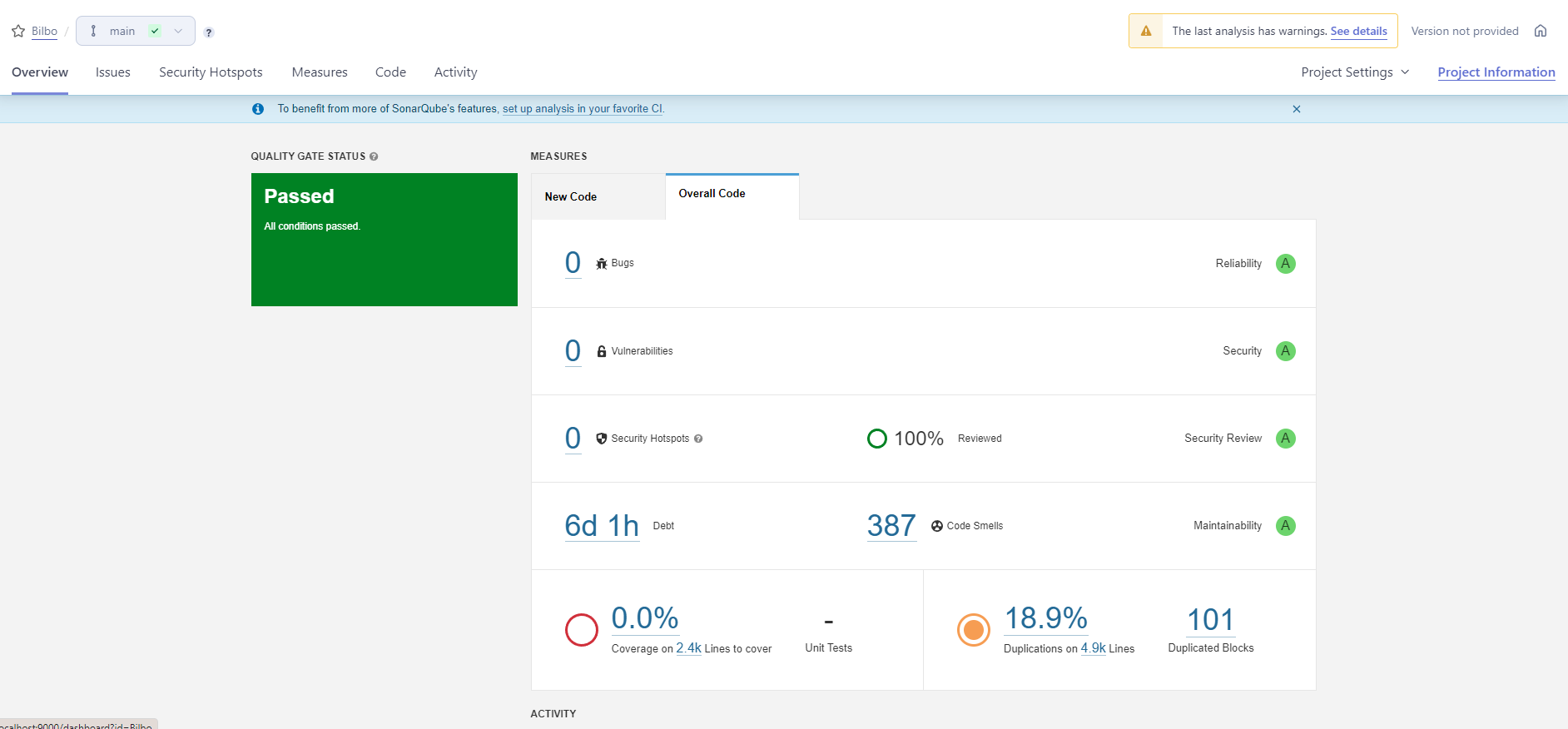




Cronograma:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Día 1 | Día 2 | Día 3 | Día 4 | Día 5 | Día 6 | Día 7 | Día 15 | TOTAL |
| Vulnerabilidad | 2 | 2 |  |  | 2 | 2 |  |  | Todas las vulnerabilidades corregidas |
| Hotspots |  |  | 2 | 2 |  |  | 1 | 1 | 1 hotspot por corregir |

Resultado 02/05/2023:



Cronograma:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Día 1 | Día 2 | Día 3 | Día 4 | Día 5 | Día 6 | Día 7 | Día 15 | Día 16 | TOTAL |
| Vulnerabilidad | 2 | 2 |  |  | 2 | 2 |  |  |  | Todas las vulnerabilidades corregidas |
| Hotspots |  |  | 2 | 2 |  |  | 1 | 1 |  | Todos los hotspots corregidos. |