

UNIVERSIDAD
NACIONAL
DE COLOMBIA

Universidad Nacional de Colombia - Sede Bogotá

Facultad de Ingeniería

Departamento de Ingeniería de Sistemas e Industrial

Asignatura:

Ingeniería de Software I

Proyecto - Informe Código Limpio

Natalia Carolina Bautista Sanchez

Daniel Santiago Cocinero Jimenez

Baruj Vladimir Ramirez Escalante

Juan Daniel Ramirez Mojica

Profesor:

Oscar Eduardo Alvarez Rodriguez

2024 - II

Bogotá D.C.

7. Clean Code

Para el proyecto, se ha decidido implementar ESLint para el frontend en JavaScript y TypeScript, y SonarQube para el backend en Spring Boot.

Frontend:

En primer lugar, como requisito previo, es necesario tener instalado Node.js y npm.

```
● PS C:\Users\santi> node -v
v22.13.1
● PS C:\Users\santi> npm -v
10.9.2
```

A continuación, se procederá a la instalación de ESLint utilizando el siguiente comando:

```
PS C:\Users\santi\Desktop\Inge\Ingsoft-I\Proyecto\Frontend> npx eslint --init
```

Luego, se presentarán varias preguntas de configuración, las cuales se deben responder conforme a las necesidades específicas del proyecto.

```
✓ How would you like to use ESLint? · problems
✓ What type of modules does your project use? · esm
✓ Which framework does your project use? · react
✓ Does your project use TypeScript? · typescript
✓ Where does your code run? · browser
The config that you've selected requires the following dependencies:
eslint, globals, @eslint/js, typescript-eslint, eslint-plugin-react
? Would you like to install them now? » No / Yes
```

Finalmente, se selecciona la opción "Yes" en la instalación inmediata y se opta por el uso de npm.

```
eslint, globals, @eslint/js, typescript-eslint, eslint-plugin-react
✓ Would you like to install them now? · No / Yes
✓ Which package manager do you want to use? · npm
```

Una vez realizada la configuración, se agrega la siguiente configuración a ESLint:

```

JS index.js  JS eslint.config.mjs X
JS eslint.config.mjs > [🔗] default
5
6
7  /** @type {import('eslint').Linter.Config[]} */
8  export default [
9    {files: ["**/*.js,mjs,cjs,ts,jsx,tsx"]},
10   {languageOptions: { globals: globals.browser }},
11   pluginJs.configs.recommended,
12   ...tseslint.configs.recommended,
13   pluginReact.configs.flat.recommended,
14   {
15     settings: {
16       react: {
17         version: "detect"
18       }
19     }
20   }
21 ];

```

```

PS C:\Users\santi\Desktop\Inge\Ingsoft-I\Proyecto\Frontend> npx eslint index.js
❖PS C:\Users\santi\Desktop\Inge\Ingsoft-I\Proyecto\Frontend>

```

Al ejecutar el linter, no se generan advertencias, ya que, en este momento, el archivo index.js solo contiene un mensaje de "Hola Mundo".

Backend:

Como se está desarrollando en IntelliJ Idea, es necesario tener este software para la instalación de SonarQube.

Desde la MarketPlace de IntelliJ podemos instalar la extensión.

Plugins Marketplace Installed 3

Search: sonar

Search Results (6) Sort By: Relevance

- SonarQube for IDE** (9.2M, 3.61 stars) [Install](#)
- SonarAnalyzer** (74.2K, 4.08 stars) [Install](#)
- Sonargraph Integration** (11.4K) [Install](#)
- Swiss-AS Dev-Tools** (2.3K, 4.56 stars) [Install](#)
- Better Highlights** (765.1K, 4.44 stars) [Install](#)
- PVS-Studio for IDEA an...** (17.3K) [Install](#)

SonarQube for IDE
[SonarSource](#) [Plugin homepage](#)
[Install](#) 10.16.180464

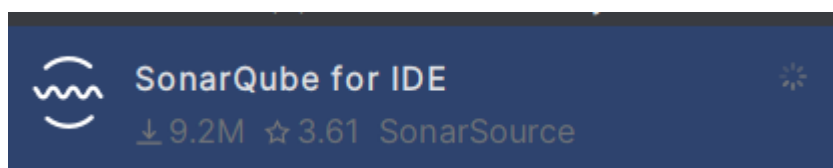
Overview What's New Reviews Additional Info

SonarQube for IDE (formerly SonarLint) by [Sonar](#) is a free IDE extension to find and fix coding issues in real-time, flagging issues as you code, just like a spell-checker. More than a linter, it also delivers rich contextual guidance to help developers understand why there is an issue, assess the risk and educate them on how to fix it. This helps improve their skills, enhance their productivity, and take ownership of their code, taking linting to a different level.

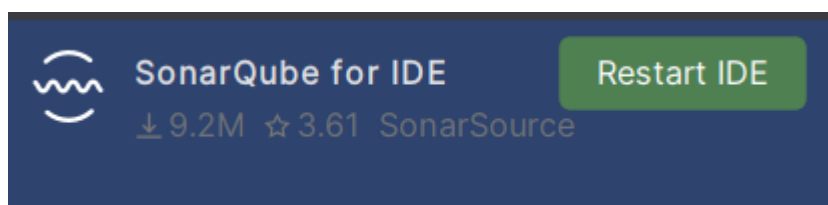
When paired with [SonarQube Server](#) or [SonarQube Cloud](#) in Connected Mode, SonarQube for IDE forms a powerful end-to-end code quality platform to enrich the CI/CD

[OK](#) [Cancel](#) [Apply](#)

Le damos en Install



Y cuando finalicé le damos en Restart IDE



Ya una vez reiniciemos podemos observar un icono en la parte superior derecha que nos dice que errores encuentra en el archivo actual.

Por ejemplo en este caso, el icono está encerrado en rojo indicando un error.



The screenshot shows an IDE with four tabs: ApplicationConfig.java, JwtAuthFilter.java, JwtService.java, and SecurityConfiguration.java. The ApplicationConfig.java tab is active, showing the following code:

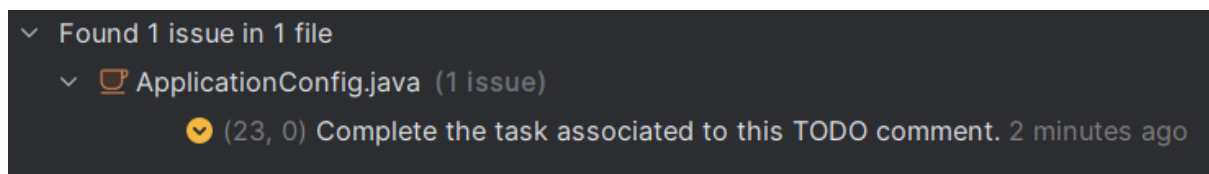
```

17 @RequiredArgsConstructor
18 public class ApplicationConfig {
19
20     private final UserRepository userService;
21
22     @Bean @dcoinero *
23     public UserDetailsService userDetailsService() { //TODO what is this

```

In the top right corner of the editor, there is a red circle containing a wavy line icon and the number 1, indicating one issue.

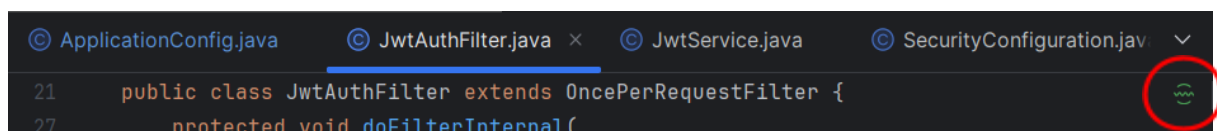
Si le damos click nos muestra más detalles.



The screenshot shows the IDE's issue list. It displays the following information:

- Found 1 issue in 1 file
- ApplicationConfig.java (1 issue)
- (23, 0) Complete the task associated to this TODO comment. 2 minutes ago

Por ejemplo en este otro archivo no dices que está todo bien.



The screenshot shows an IDE with four tabs: ApplicationConfig.java, JwtAuthFilter.java, JwtService.java, and SecurityConfiguration.java. The JwtAuthFilter.java tab is active, showing the following code:

```

21 public class JwtAuthFilter extends OncePerRequestFilter {
27     protected void doFilterInternal(

```

In the top right corner of the editor, there is a red circle containing a wavy line icon, indicating no issues.

Opiniones del código:

Juan Daniel Ramírez Mojica: Por los códigos que he podido observar, son bastante fáciles de entender y están bien organizados. La forma en que se ha estructurado facilita su edición y mantenimiento. Se nota que cada módulo cumple con su propósito específico, siguiendo el principio de responsabilidad única de SOLID. Sin embargo, considero que podría mejorar con la inclusión de más comentarios en ciertas secciones. Personalmente, suelo agregar comentarios a gran parte del código, ya que tiendo a olvidar los detalles con el tiempo. Aunque el código es intuitivo, los comentarios adicionales facilitarían su comprensión al revisarlo o hacer modificaciones en el futuro.

Natalia Bautista: El código está bien indentado, funciona y cumple con su propósito. Tiene una buena estructura en general. Para el nivel que se debería tener en el curso se le ve bastante avanzado.

Daniel Santiago Cocinero Jiménez: No me ha sido posible revisar el código de mis compañeros en el Frontend porque aún no se ha avanzado en esto, pero el Backend se ve muy bien estructurado, las carpetas están separadas por las funcionalidades que cumplen, y aunque la lógica es un poco compleja se puede rastrear fácilmente que hace el código. Aún creo que se deberían dividir algunas funciones pero la mayoría ya cumple con los principios SOLID.

Baruj Vladimir Ramírez Escalante: Opino que el código de mis compañeros de equipo, ha sido escrito de forma ordenada, estructurada y funcional. Quiero resaltar que en general, no hay demasiados comentarios que puedan ensuciar el código. Adicionalmente, el nombramiento de las diversas entidades del código son intuitivas y permiten una fácil revisión del código.