

SALECOLD

INFORME DE CALIDAD EN EL CÓDIGO FUENTE



¿Qué es SonarQube?



Instalación y configuración de SonarQube en Linux



Configuración de SonarQube en el proyecto



Resultado del análisis hecho por SonarQube al código fuente



Ajustes en el código fuente



Conclusión

¿Qué es SonarQube?

SonarQube es una plataforma de código abierto para la inspección de la calidad del código a través de diferentes herramientas de análisis de código fuente. Esta plataforma proporciona métricas que ayudan a mejorar la calidad del código.

Principales métricas de SonarQube

- Complejidad:** Es una métrica de calidad software basada en el cálculo del número de caminos independientes que tiene nuestro código.
- Duplicados:** Nos indica el número de bloques de líneas duplicados.
-

Bugs: Son errores o defectos en el software.

- **Vulnerabilidad:** Son problemas de seguridad en el software y deben ser solucionados de inmediato.
- **Security Hotspot:** Es un fragmento de código sensible a la seguridad, pero es posible que no afecte la seguridad general del software.
- **Mantenibilidad:** Se refiere al recuento total de problemas de Code Smell o malas prácticas implementadas en el proyecto.
- **Tamaño:** Permiten hacerse una idea del volumen del proyecto en términos generales.
- **Pruebas:** Son una forma de comprobar el correcto funcionamiento de una unidad de código y de su integración.

Instalación y configuración de SonarQube en Linux

Descargar e instalar openjdk 11 o jdk 11

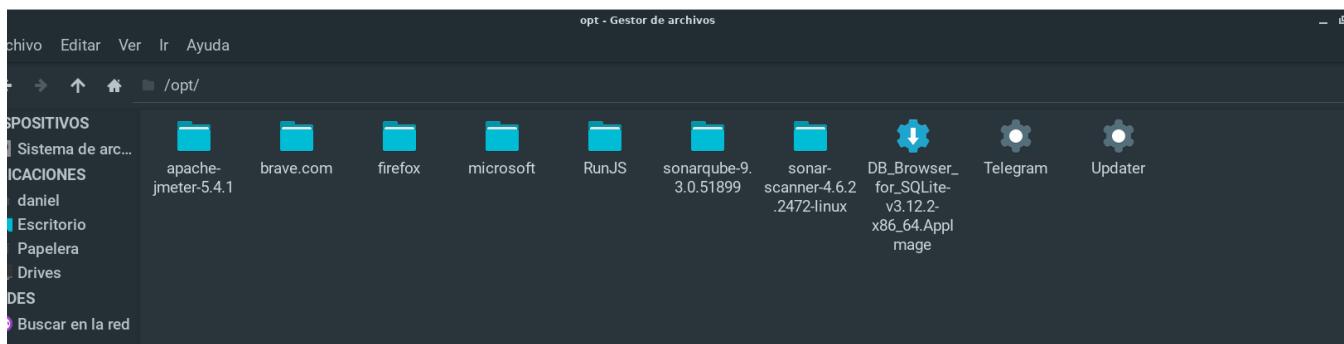
Java Archive Downloads - Java SE 11

JDK 11

Descargar SonarQube Community edition

Code Quality and Code Security | SonarQube
sonarqube

1. Extraer el zip en la raíz del disco duro



2. Modificar la configuración del archivo wrapper.conf el cual esta en la carpeta conf de sonarqube

- ① En la tercer linea del archivo se especifica el path donde esta instalado el openjdk o el jdk.

```
1 # Path to JVM executable. By default it must be available in PATH.  
2 # Can be an absolute path, for example:  
3 wrapper.java.command=/usr/lib/jvm/java-11-openjdk-amd64/bin/java  
4 #wrapper.java.command=java
```

3. Acceder a la carpeta bin, luego a linux-x86-64 y abrir una terminal

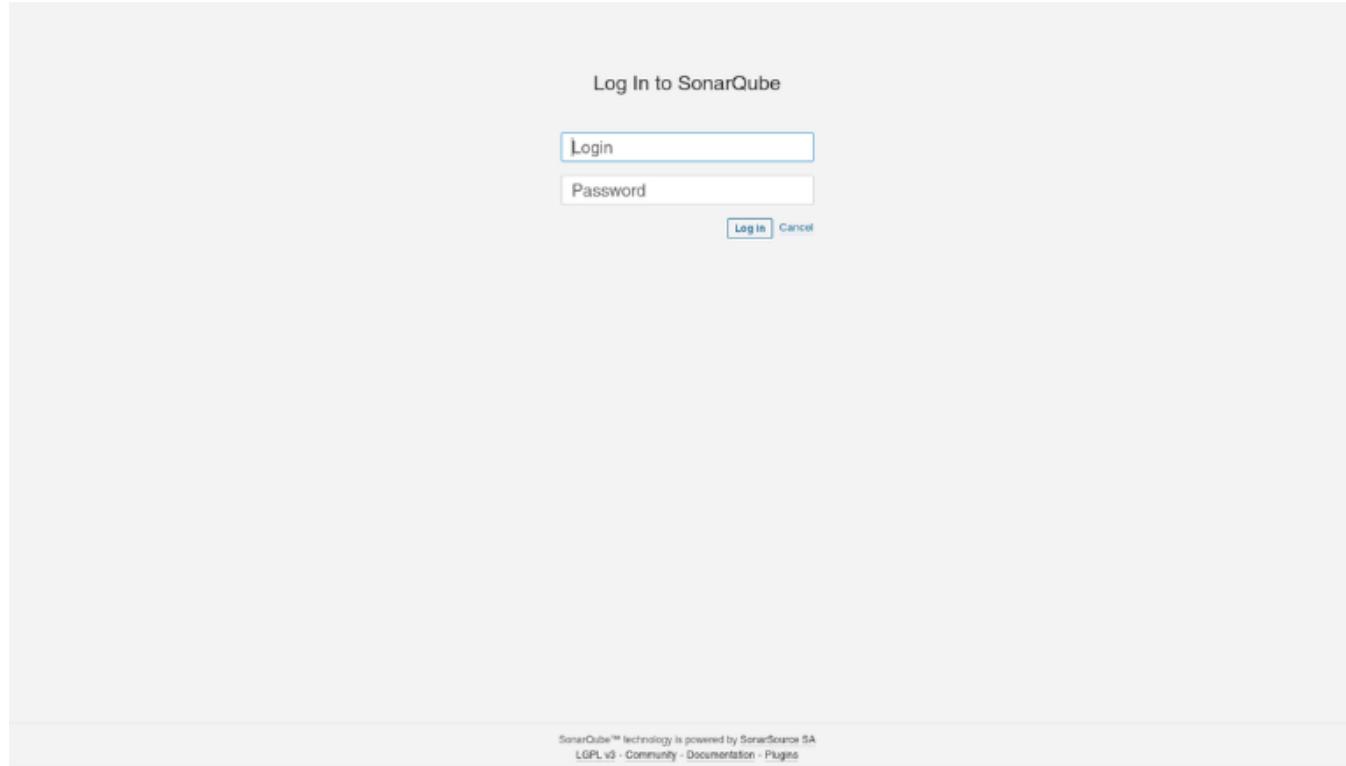
Ejecutar el siguiente comando para iniciar el servidor de SonarQube.

```
1 sh sonar.sh start
```

4. Acceder en el navegador a la interfaz de sonarqube

Se debe colocar en el navegador localhost:9000 o 127.0.0.1:9000.

- ① La primera vez que se accede el usuario es admin y la contraseña admin, después nos pedirá actualizar la contraseña.



Configuración de SonarQube en el proyecto

1. Seleccionar la forma en como se creara el nuevo proyecto en SonarQube

En este caso se seleccionara la opcion “Manually”.

The screenshot shows the 'Create a project' interface. At the top, there's a header with navigation links: sonarcube, Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, a search bar, and a user icon. Below the header, a message asks how to create the project. It then provides options to import from popular DevOps platforms: Azure DevOps, Bitbucket, GitHub, and GitLab, each with a 'Set up global configuration' link. Finally, it offers a manual creation option with a 'Manually' button and a double arrow icon.

2. Especificar el nombre del nuevo proyecto y el identificador único

The screenshot shows the 'Create a project' form. The title is 'Create a project'. A note says 'All fields marked with * are required'. The 'Project display name *' field contains 'SaleCold' with a green checkmark. A note below it says 'Up to 255 characters. Some scanners might override the value you provide.' The 'Project key *' field also contains 'SaleCold' with a green checkmark. A note below it says 'The project key is a unique identifier for your project. It may contain up to 400 characters. Allowed characters are alphanumeric, '-' (dash), '_' (underscore), '.' (period) and ':' (colon), with at least one non-digit.' At the bottom is a 'Set Up' button.

3. Seleccionar la forma en como se analizara el código del proyecto

En este caso seleccionaremos la opcion “Locally”.

The screenshot shows the project overview for 'SaleCold'. The top navigation bar includes sonarcube, Projects, Issues, Rules, Quality Profiles, Quality Gates, Administration, a search bar, and a user icon. Below the navigation, the project name 'SaleCold' is shown along with its status 'master'. The main content area has tabs: Overview, Issues, Security Hotspots, Measures, Code, Activity, Project Settings, and Project Information. The 'Overview' tab is selected. A note at the bottom says 'This project has no code analysis results yet.'

How do you want to analyze your repository?

Do you want to integrate with your favorite CI? Choose one of the following tutorials.



With Jenkins



With GitHub Actions



With Bitbucket Pipelines



With GitLab CI



With Azure Pipelines



Other CI

Are you just testing or have an advanced use-case? Analyze your project locally.



Locally

4. Se crea o se utiliza un token ya generado

The screenshot shows the SonarQube interface for the 'SaleCold' project. The top navigation bar includes 'Projects', 'Issues', 'Rules', 'Quality Profiles', 'Quality Gates', 'Administration', and a search bar. Below the navigation, the project name 'SaleCold' is displayed along with its branch 'master'. The main content area is titled 'Analyze your project' with the sub-instruction 'We initialized your project on SonarQube, now it's up to you to launch analyses!'. A numbered step '1 Provide a token' is shown, with a sub-instruction 'Generate a token'. A text input field 'Enter a name for your token' and a 'Generate' button are visible. A note below states: 'The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point of time in your user account.' Step '2 Run analysis on your project' is also partially visible.

Damos clic en generate:

The screenshot shows the SonarQube interface for the 'SaleCold' project. The top navigation bar and project details are identical to the previous screenshot. The main content area is titled 'Analyze your project' with the sub-instruction 'We initialized your project on SonarQube, now it's up to you to launch analyses!'. Step '1 Provide a token' is completed, showing the generated token 'SaleCold: 352a229c62c6aaa353d3c31b2865c21557526b2e' with a trash icon. A note below states: 'The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point of time in your user account.' A 'Continue' button is present. Step '2 Run analysis on your project' is also partially visible.

5. Seleccionar la tecnología o el lenguaje de programación que se uso en el proyecto

The screenshot shows the SonarQube interface for the 'SaleCold' project. The top navigation bar and project details are identical to the previous screenshots. The main content area is titled 'Analyze your project' with the sub-instruction 'We initialized your project on SonarQube, now it's up to you to launch analyses!'. Step '1 Provide a token' is completed, showing the generated token 'SaleCold: 352a229c62c6aaa353d3c31b2865c21557526b2e' with a trash icon. A note below states: 'The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point of time in your user account.' A 'Continue' button is present. Step '2 Run analysis on your project' is also partially visible.

We initialized your project on SonarQube, now it's up to you to launch analyses!

1 Provide a token SaleCold:352a229c62c6aaa353d3c31b2865c21557526b2e

2 Run analysis on your project

What option best describes your build?

Maven Gradle .NET Other (for JS, TS, Go, Python, PHP, ...)

6. Seleccionar el sistema operativo que estamos usando

SaleCold master

Overview Issues Security Hotspots Measures Code Activity Project Settings ▾ Project Information

Analyze your project

We initialized your project on SonarQube, now it's up to you to launch analyses!

1 Provide a token SaleCold:352a229c62c6aaa353d3c31b2865c21557526b2e

2 Run analysis on your project

What option best describes your build?

Maven Gradle .NET Other (for JS, TS, Go, Python, PHP, ...)

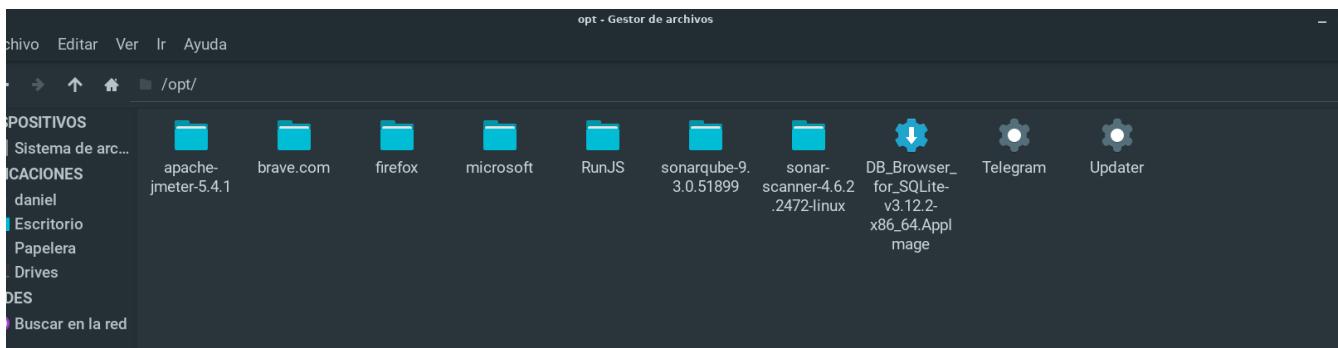
What is your OS?

Linux Windows macOS

7. Descargar SonarScanner

SonarScanner | SonarQube Docs

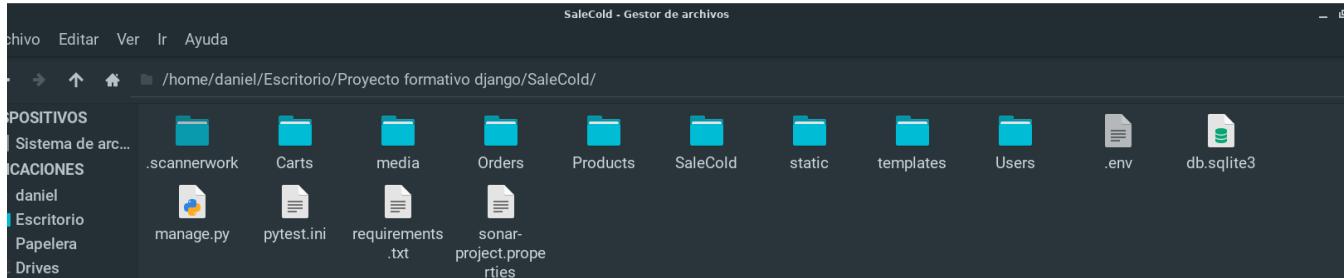
8. Extraer el zip en la raiz del disco duro



9. En el archivo .bashrc se agrega el path o la ruta donde esta almacenada la carpeta de SonarScanner

10. Crear un archivo de configuración de sonarqube en la raíz del proyecto

(i) El archivo se debe llamar sonar-project.properties



11. Agregar la siguiente configuración al archivo creado en el paso anterior

- **sonar.projectKey** = Acá debe ir el nombre del proyecto que especificamos en sonarqube.
- **sonar.projectName** = Acá debe ir el nombre de la carpeta raíz del proyecto que vamos a analizar.

```

1 # must be unique in a given SonarQube instance
2 sonar.projectKey=SaleCold
3 # --- optional properties ---
4
5 # defaults to project key
6 sonar.projectName=SaleCold
7 # defaults to 'not provided'
8 sonar.projectVersion=1.0
9
10 # Path is relative to the sonar-project.properties file. Defaults to .
11 sonar.sources=.
12
13 # Encoding of the source code. Default is default system encoding
14 sonar.sourceEncoding=UTF-8
15
16 sonar.python.version=3.8

```

12. Ejecutar el análisis al proyecto

Después de escoger la tecnología del proyecto y el sistema operativo, ademas de configurar e instalar el sonarScanner, sonarqube nos genera un código el cual se debe ejecutar en una nueva terminal y la terminal debe estar en la carpeta raíz del proyecto.

The screenshot shows the SonarQube interface for the 'SaleCold' project. The top navigation bar includes 'Projects', 'Issues', 'Rules', 'Quality Profiles', 'Quality Gates', and 'Administration'. A search bar says 'Search for projects...'. Below the header, it shows 'SaleCold' with a star icon and 'master'. The main content area has tabs for 'Overview', 'Issues', 'Security Hotspots', 'Measures', 'Code', and 'Activity'. At the bottom, there are buttons for 'Linux', 'Windows' (which is selected), and 'macOS'.

Download and unzip the Scanner for Linux

Visit the [official documentation of the Scanner](#) to download the latest version, and add the `bin` directory to the `PATH` environment variable

Execute the Scanner

Running a SonarQube analysis is straightforward. You just need to execute the following commands in your project's folder.

```
sonar-scanner \
-Dsonar.projectKey=SaleCold \
-Dsonar.sources=. \
-Dsonar.host.url=http://localhost:9000 \
-Dsonar.login=352a229c62c6aaa353d3c31b2865c21557526b2e
```

 Copy

Please visit the [official documentation of the Scanner](#) for more details.

Is my analysis done? If your analysis is successful, this page will automatically refresh in a few moments.

You can set up Pull Request Decoration under the project settings. To set up analysis with your favorite CI tool, see the tutorials.

Check these useful links while you wait: [Branch Analysis](#), [Pull Request Analysis](#).

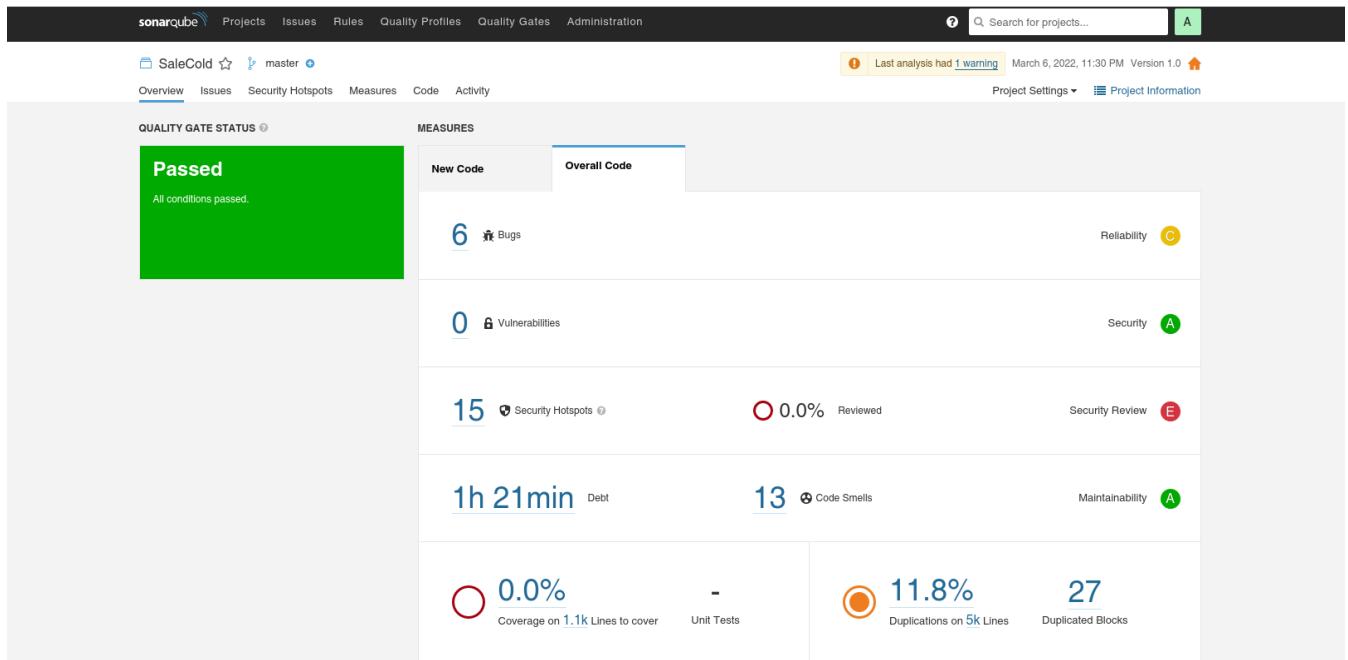
```
daniel ~ > Escritorio > Proyecto formativo django > SaleCold > sonar-scanner \
-Dsonar.projectKey=SaleCold \
-Dsonar.sources=. \
-Dsonar.host.url=http://localhost:9000 \
-Dsonar.login=352a229c62c6aaa353d3c31b2865c21557526b2e
```

Si el análisis se realizo de manera correcta nos debe generar el siguiente mensaje en la terminal:

```
F0: ANALYSIS SUCCESSFUL, you can browse http://localhost:9000/dashboard?id=SaleCold
F0: Note that you will be able to access the updated dashboard once the server has processed the submitted
yisys report
F0: More about the report processing at http://localhost:9000/api/ce/task?id=AX83nEPkwYQgiyCj-nB_
F0: Analysis total time: 52.514 s
F0: -----
F0: EXECUTION SUCCESS
F0: -----
F0: Total time: 59.664s
F0: Final Memory: 23M/80M
F0: -----
```

```
daniel ~ > Escritorio > Proyecto formativo django > SaleCold >
```

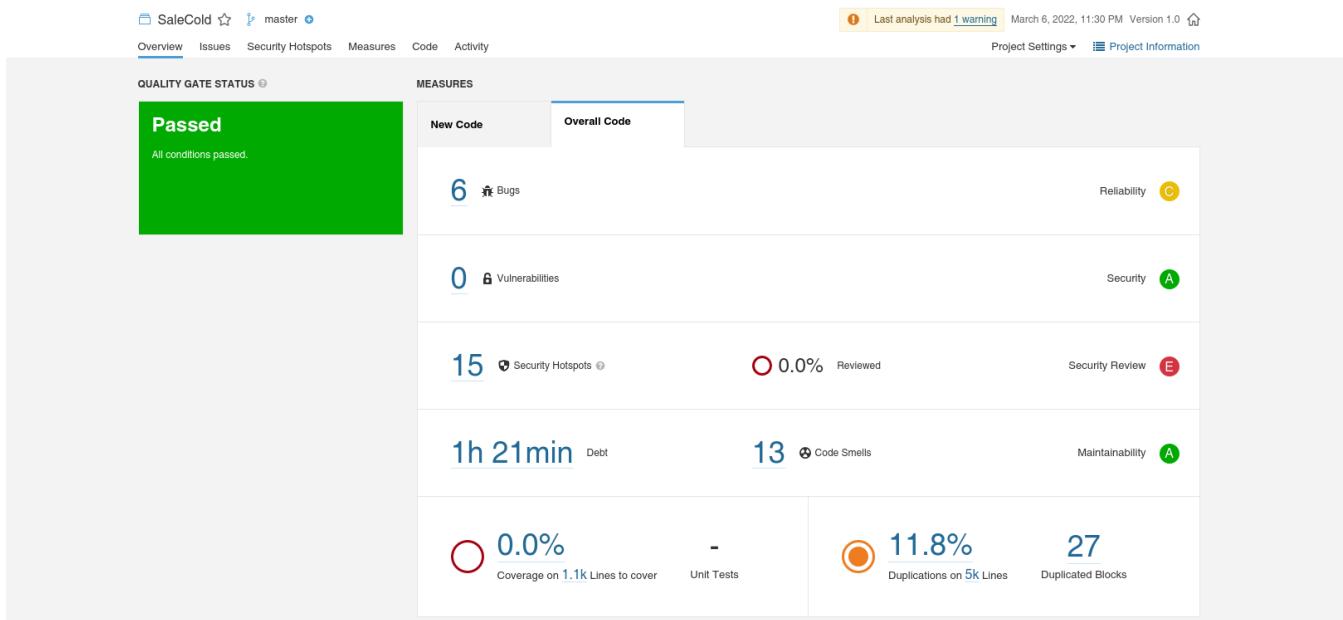
Pegamos el link que nos genero sonarqube en el navegador para ver el resultado del análisis:



The screenshot shows the SonarQube interface for the 'SaleCold' project. At the top, there's a navigation bar with links for 'Projects', 'Issues', 'Rules', 'Quality Profiles', 'Quality Gates', and 'Administration'. A search bar is also present. In the center, the 'Overview' tab is selected, showing the following data:

- QUALITY GATE STATUS:** Passed (All conditions passed)
- MEASURES:**
 - New Code: 6 Bugs (Reliability: C)
 - Overall Code: 0 Vulnerabilities (Security: A)
 - 15 Security Hotspots (0.0% Reviewed, Security Review: E)
 - 1h 21min Debt (13 Code Smells, Maintainability: A)
 - 0.0% Coverage on 1.1k Lines to cover (Unit Tests)
 - 11.8% Duplications on 5k Lines (Duplicated Blocks: 27)

Resultado del análisis hecho por SonarQube al código fuente



Resultado del análisis hecho por SonarQube al código fuente

Resumen del análisis:

- Se encontraron 6 bugs.
- El software tiene 0 vulnerabilidades de seguridad.
- Se encontraron 13 Code smells o malas prácticas.
- Se encontraron 15 fragmentos de código que podrían afectar la seguridad del software.
- Se demoraría aproximadamente 1h 21 minutos para refactorizar el código.
- El 11.8% del código está duplicado.

Ajustes en el código fuente



Bugs



Security Hotspots



Code smell

Bugs

Screenshot of the SonarQube interface showing a list of bugs. The sidebar on the left contains various filters and dropdown menus. The main area displays a list of bugs with their details. A message at the bottom of the list reads: "Embedded database should be used for evaluation purposes only. The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine."

Solución bugs 2, 3 y 5.

Para solucionar este bug se le agrega el atributo aria-hidden para que la tecnología de asistencia del navegador omita este ícono, según la MDN este atributo ayuda a mejorar la accesibilidad de la página.

aria-hidden - Accessibility | MDN

Screenshot of the MDN web documentation page for 'aria-hidden'. The page title is 'aria-hidden - Accessibility | MDN'. Below the title, there is a code snippet for HTML:

```
1 <i class="fas fa-info-circle container-link-main-aside__icon" aria-hidden="true"></i>
```

Solución bugs 1 y 4

Estos bugs son falsos positivos porque se usan para la herencia entre templates, es decir son plantillas por ende no necesitan la etiqueta title.

Solución bug 6

Para la solución de este bug se agrega el atributo alt para ofrecer un texto alternativo en caso de que el navegador no cargue la imagen.



HTML

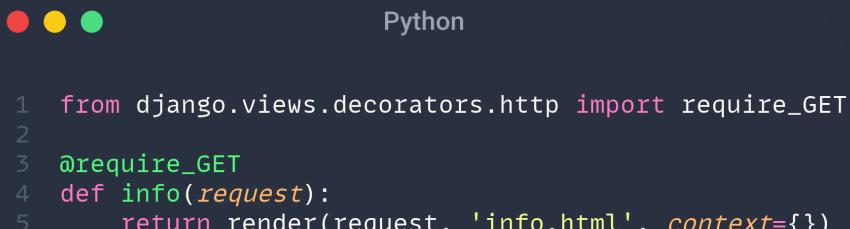
```
1 
```

Security Hotspots

Solución security hotspots 1 al 14

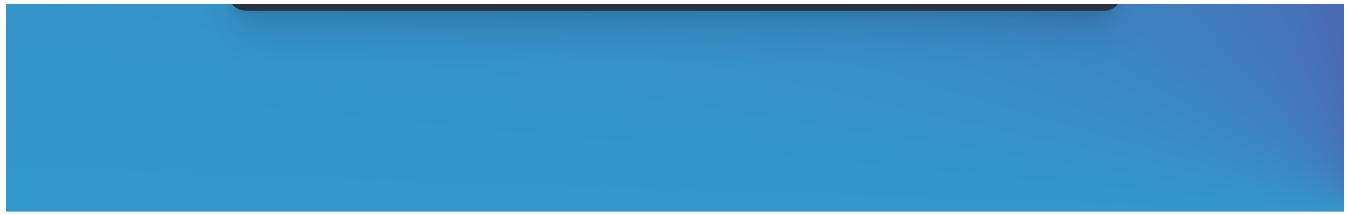
Para solucionar estos security hotspots se le agrego un decorador a cada una de las vistas para restringir el tipo de petición que pueden recibir, de esta manera se solucionan las posibles vulnerabilidades de seguridad.

Ejemplo #1 de una vista con el decorador agregado:



Python

```
1 from django.views.decorators.http import require_GET
2
3 @require_GET
4 def info(request):
5     return render(request, 'info.html', context={})
```



Ejemplo #2 de una vista con el decorador agregado:

```
1 from django.views.decorators.http import require_http_methods
2
3 @require_http_methods(['GET', 'POST'])
4 def cart_view(request):
5     cart = get_or_create_cart(request)
6
7     return render(request, 'cart/carrito_de_compras.html', context = {
8         'carrito': cart,
9     })
```

Code smell

The screenshot shows a static code analysis tool interface with the following details:

- Project Status:** Last analysis had 1 warning, March 7, 2022, 6:21 PM, Version 1.0.
- Filters:** My Issues, All.
- Issues Summary:** 1 / 13 issues, 1h 21min effort.
- Filters (Type):** CODE SMELL (selected), Bug, Vulnerability, Code Smell (13).
- Filters (Severity):** Blocker (1), Critical (2), Major (9), Minor (1), Info (0).
- Issues List:** A list of 13 code smell issues across several files:
 - Carts/models.py: Rename this variable; It shadows a builtin. Why is this an issue? (Code Smell, Major, Open, Not assigned, 5min effort, Comment)
 - Carts/views.py: Remove the unused local variable "cart_product". Why is this an issue? (Code Smell, Minor, Open, Not assigned, 5min effort, Comment)
 - Products/migrations/0001_initial.py: Define a constant instead of duplicating this literal 'Fecha de modificacion' 3 times. Why is this an issue? (Code Smell, Critical, Open, Not assigned, 6min effort, Comment)
 - SaleCold/settings.py: Rename this commented out code. Why is this an issue? (Code Smell, Major, Open, Not assigned, 5min effort, Comment)
 - Users/forms.py: Rename field "genero" to prevent any misunderstanding/clash with field "GENERO" defined on line 17. Why is this an issue? (Code Smell, Blocker, Open, Not assigned, 10min effort, Comment)
 - Users/forms.py: Define a constant instead of duplicating this literal 'Este campo acepta minimo 4 caracteres y maximo 25 caracteres.' 6 times. Why is this an issue? (Code Smell, Critical, Open, Not assigned, 12min effort, Comment)
 - Users/views.py: Rename function "updateDataUser" to match the regular expression "[a-zA-Z][a-zA-Z-]*\$". Why is this an issue? (Code Smell, Major, Open, Not assigned, 10min effort, Comment)

Solucion code smell 1 y 3

Para solucionar estos code smell se renombraron dos variables porque estaban usando palabras reservadas del lenguaje.

```
Python

1 def random_products(product):
2     productos = Product.objects.exclude(pk=product.id_product)
3     cantidad_productos = abs(Product.objects.all().count() - 7)
4     numero_aleatorio = random.randint(1, cantidad_productos)
5
6     return productos[numero_aleatorio:numero_aleatorio + 6]
```

Solucion code smell 4

Para solucionar este code smell se elimino una linea de codigo que no se estaba usando.

Solucion code smell 7

Para solucionar este code smell se creo una variable constante al inicio del archivo y luego se utilizo en los lugares donde se repetía la cadena.

```
Python

1 ERROR_CANTIDAD_CARACTERES = 'Este campo acepta minimo 4 caracteres y maximo 25 caracteres.'
```

Ejemplo de una funcion para validar un campo del formulario usando la variable constante:

```
Python

1 def clean_nombre(self):
2     nombre = self.cleaned_data.get('nombre')
3
4     if len(nombre) == 0:
5         raise forms.ValidationError(CAMPO_OBLIGATORIO)
6     elif len(nombre) < 4 or len(nombre) > 25:
7         raise forms.ValidationError(ERROR_CANTIDAD_CARACTERES)
8
9     return nombre
```

Solucion code smells 10 al 13

La solución de estos tres code smells fue eliminar el código repetido.

Conclusión

Después de realizar la refactorización al código para solucionar los problemas de calidad, se ejecuta de nuevo el análisis dando un resultado satisfactorio, ya que el código fuente del aplicativo cuenta con calidad.

