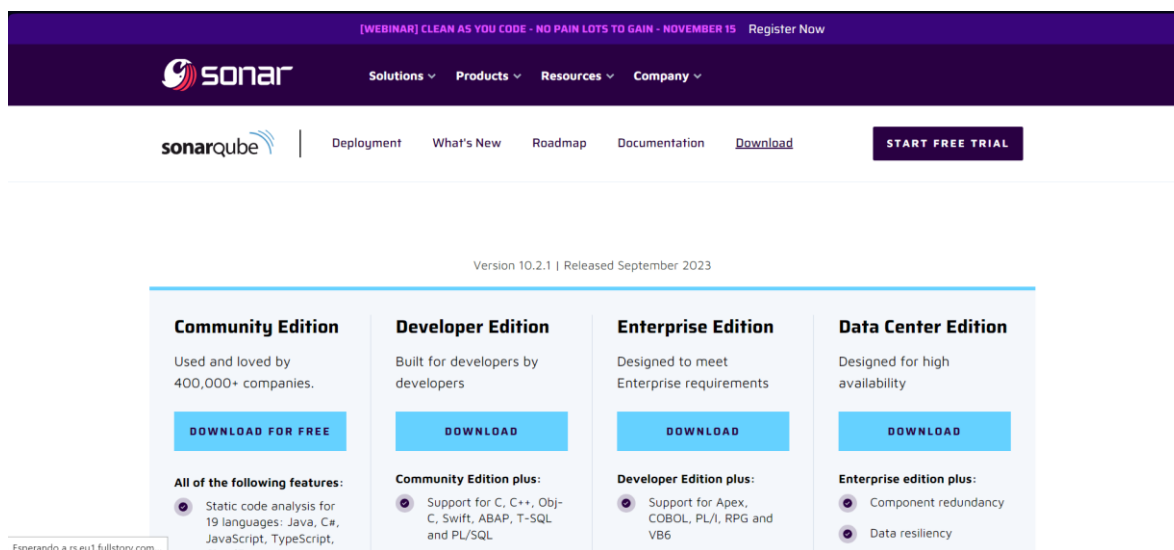


Instalación Sonarqube Y SonarScanner

Para la instalación y ejecución de SonarQube y SonarScanner se deben seguir los siguientes pasos:

1. Descargar SonarQube Community Edition desde el siguiente LINK:

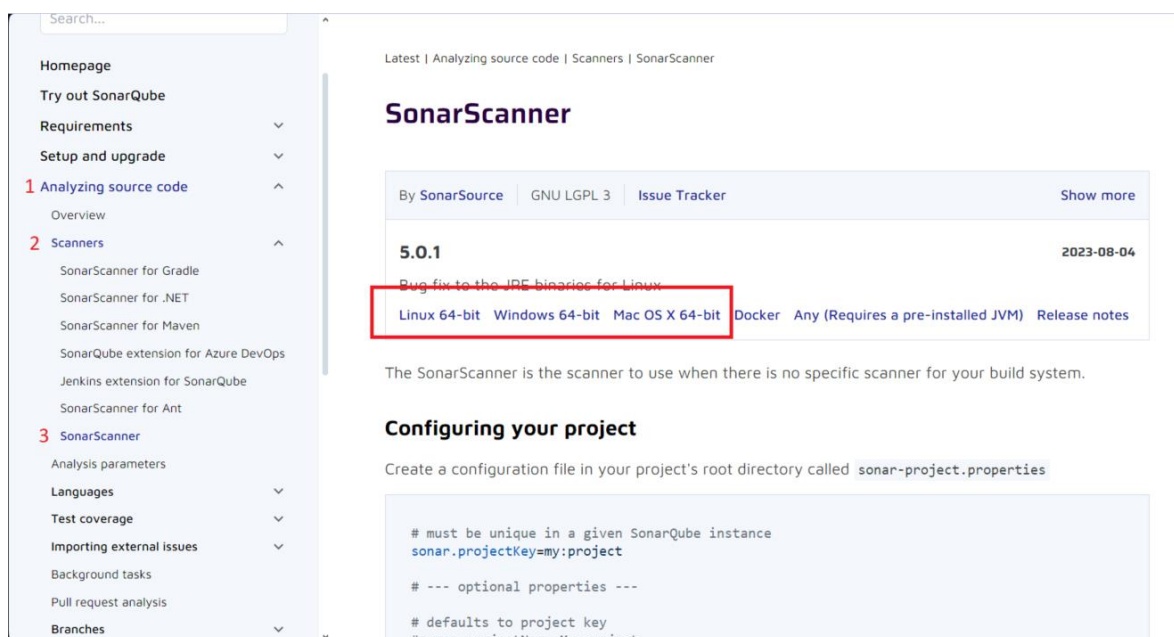
<https://www.sonarsource.com/products/sonarqube/downloads/>



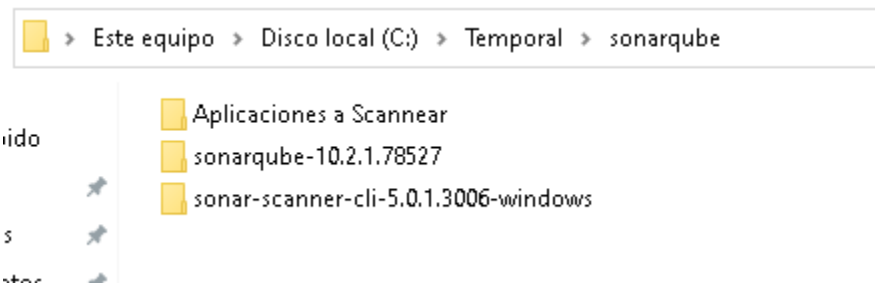
2. Descargar SonarScanner desde el siguiente LINK:

<https://docs.sonarsource.com/sonarqube/latest/>

1. Desde el menú de la Izquierda navegar a SonarScanner y descargar la versión correspondiente al SO que vamos a utilizar:



3. Una vez descargados descomprimir ambos en carpetas separadas:



Se recomienda crear una carpeta en el disco C: y dentro de esta dejar SonarQube y SonarScanner. Además también es recomendable crear otra carpeta donde dejaremos el código a analizar.

4. Como requisito debemos tener instalado el JDK 17 que se puede descargar del siguiente LINK:
<https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html>

Ahí debemos elegir el que corresponda a nuestro SO.

ORACLE		
Products	Industries	Resources
Customers	Partners	Developers
Company		
Q	US	View Accounts
Contact Sales		
Java SE Development Kit 17.0.9		
This software is licensed under the Oracle No-Fee Terms and Conditions License.		
Product / File Description	File Size	Download
Linux Arm 64 Compressed Archive	172.37 MB	https://download.oracle.com/java/17/archive/jdk-17.0.9_linux-aarch64_bin.tar.gz (sha256)
Linux Arm 64 RPM Package	172.62 MB	https://download.oracle.com/java/17/archive/jdk-17.0.9_linux-aarch64_bin.rpm (sha256)
Linux x64 Compressed Archive	174.01 MB	https://download.oracle.com/java/17/archive/jdk-17.0.9_linux-x64_bin.tar.gz (sha256)
Linux x64 Debian Package	149.40 MB	https://download.oracle.com/java/17/archive/jdk-17.0.9_linux-x64_bin.deb (sha256)
Linux x64 RPM Package	173.73 MB	https://download.oracle.com/java/17/archive/jdk-17.0.9_linux-x64_bin.rpm (sha256)
macOS Arm 64 Compressed Archive	168.18 MB	https://download.oracle.com/java/17/archive/jdk-17.0.9_macos-aarch64_bin.tar.gz (sha256)
macOS Arm 64 DMG Installer	167.60 MB	https://download.oracle.com/java/17/archive/jdk-17.0.9_macos-aarch64_bin.dmg (sha256)

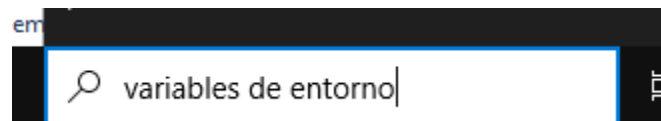
5. Instalar el JDK



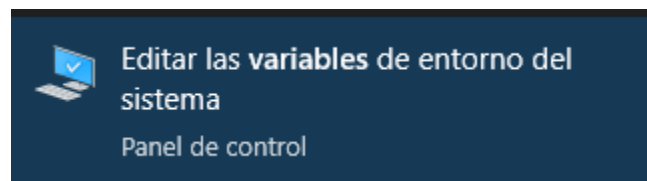
6. Para ejecutar la inspección de código sin problemas realizaremos la siguiente configuración (Estos pasos se ejecutan si vamos a ejecutar la App desde SO Windows):

6.1. Editar las variables de entorno para que dejar JDK 17 por default.

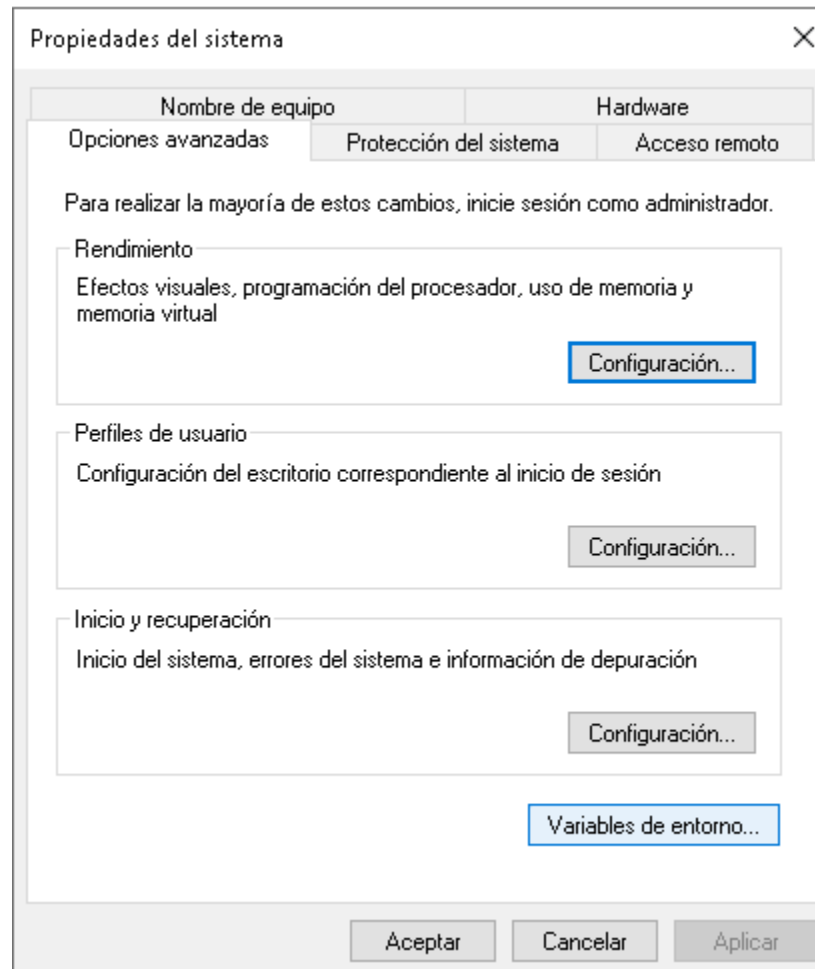
6.1.1. Desde el buscador de Windows escribir "Variables de entorno":



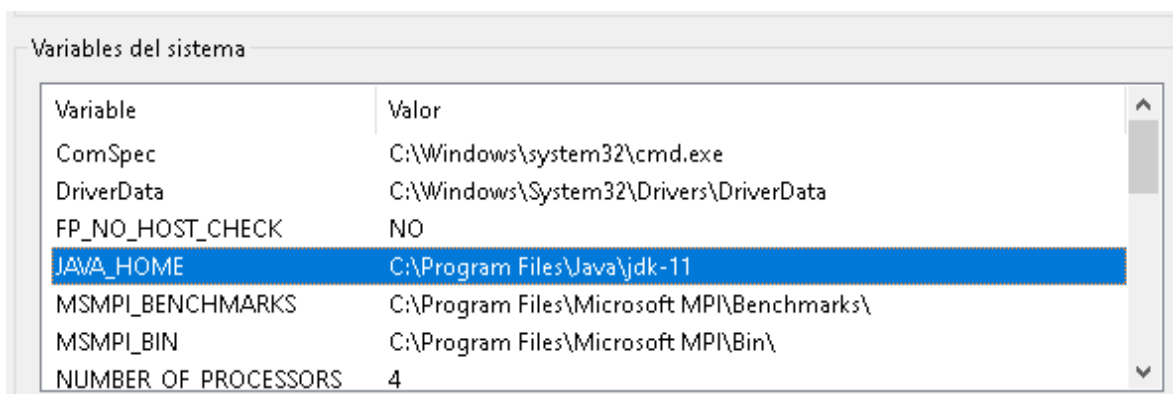
6.1.2. Seleccionar:



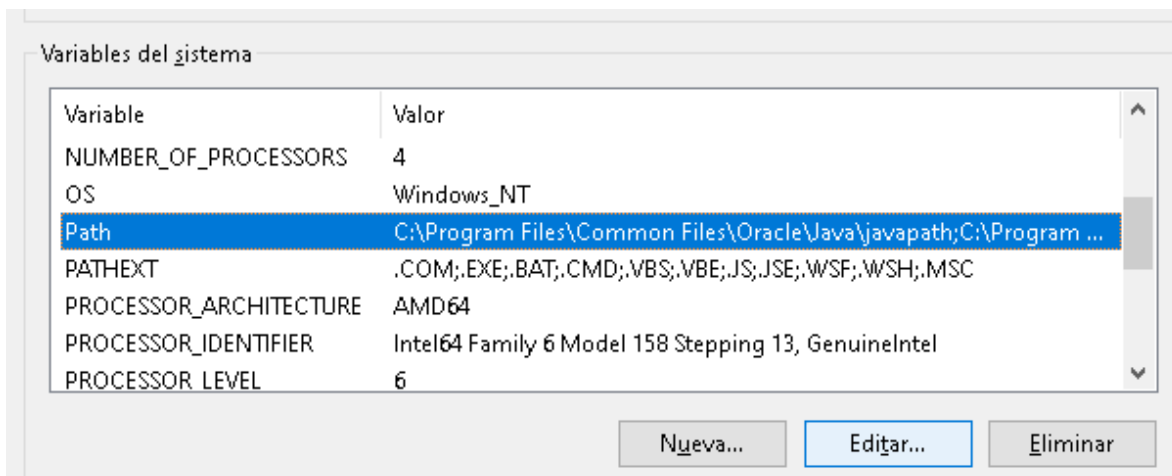
6.1.3. En la pestaña propiedades dar click en “Variables de entorno”



6.1.4. En variables del sistema añadir (o editar si ya existe) JAVA_HOME y agregar la ruta donde se instaló el JDK 17

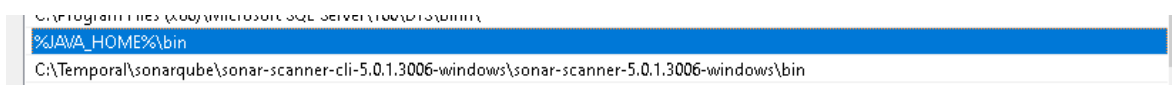


6.1.5. Seleccionar Path y dar click en editar:

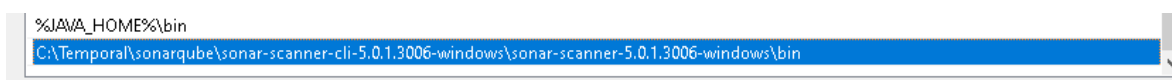


6.1.6. Acá agregaremos 2 nuevos registros:

6.1.6.1. Agregar la ruta donde esta el ejecutable java: %JAVA_HOME%\bin



6.1.6.2. Agregar otra ruta asociada al ejecutable del sonarScanner (acá dependerá de donde dejamos la carpeta descomprimida en el paso 3) el archivo se encuentra dentro de otra carpeta al interior llamada "bin":



7. Realizados estos pasos podemos comprobar si todo es correcto abriendo una terminal ("cmd") y ejecutando los siguientes comandos:

7.1. "Java -version" y "sonar-scanner --version" (debería aparecer la versión de Java y sonnarScanner que se va a usar:

```
C:\Users>java -version
java version "17.0.9" 2023-10-17 LTS
Java(TM) SE Runtime Environment (build 17.0.9+11-LTS-201)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.9+11-LTS-201, mixed mode, sharing)

C:\Users>
```


```
C:\Users\Daniel>sonar-scanner --version
INFO: Scanner configuration file: C:\Temporal\sonarquib\sonar-scanner-cli-5.0.1.3006-windows\sonar-scanner-5.0.1.3006-wi
ndows\bin\..\.conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: SonarScanner 5.0.1.3006
INFO: Java 17.0.7 Eclipse Adoptium (64-bit)
INFO: Windows 10 10.0 amd64
```

8. Realizados estos pasos debemos proceder a levantar la WEB de SonarQube. Para esto debemos desde la terminal ejecutar “StartSonar.bat” que se encuentra en la ruta “...\sonarqube-10.2.1.78527\bin\windows-x86-64” (el comienzo de la ruta dependerá de donde alojamos la carpeta en el paso 3 y la ultima carpeta de la ruta corresponde al SO que estamos utilizando).

Nombre	Fecha de modificacion	Link	Imagene
			


```
C:\Users>cd C:\Temporal\sonarqube\sonarqube-10.2.1.78527\bin\windows-x86-64
C:\Temporal\sonarqube\sonarqube-10.2.1.78527\bin\windows-x86-64>StartSonar.bat
```

Al ejecutar debemos esperar a que termine de levantar la web



```
C:\Temporal\sonarqube\sonarqube-10.2.1.78527\bin\windows-x86-64>StartSonar.bat
Starting SonarQube...
2023.11.08 21:13:59 INFO app[[o.s.a.AppFileSystem] Cleaning or creating temp directory C:\Temporal\sonarqube\sonarqube-10.2.1.78527\temp
2023.11.08 21:13:59 INFO app[[o.s.a.es.EsSettings] Elasticsearch listening on [HTTP: 127.0.0.1:9001, TCP: 127.0.0.1:50518]
2023.11.08 21:13:59 INFO app[[o.s.a.ProcessLauncherImpl] Launch process[ELASTICSEARCH] from [C:\Temporal\sonarqube\sonarqube-10.2.1.78527\elasticsearch]: C:\Program Files\Java\jdk-17\bin\java -Xms4m -Xmx64m -XX:+UseSerialGC -Dcli.name=server -Dcli.script=../bin/elasticsearch -Dcli.libs=lib/tools/server-cli -Des.path.home=C:\Temporal\sonarqube\sonarqube-10.2.1.78527\elasticsearch -Des.path.conf=C:\Temporal\sonarqube\sonarqube-10.2.1.78527\temp\conf\es -Des.distribution.type=tar -cp C:\Temporal\sonarqube\sonarqube-10.2.1.78527\elasticsearch\lib\*;C:\Temporal\sonarqube\sonarqube-10.2.1.78527\elasticsearch\lib\cli-launcher\* org.elasticsearch.launcher.CliToolLauncher
2023.11.08 21:13:59 INFO app[[o.s.a.SchedulerImpl] Waiting for Elasticsearch to be up and running
```

Cuando termina aparece el mensaje “SonarQube is operational”

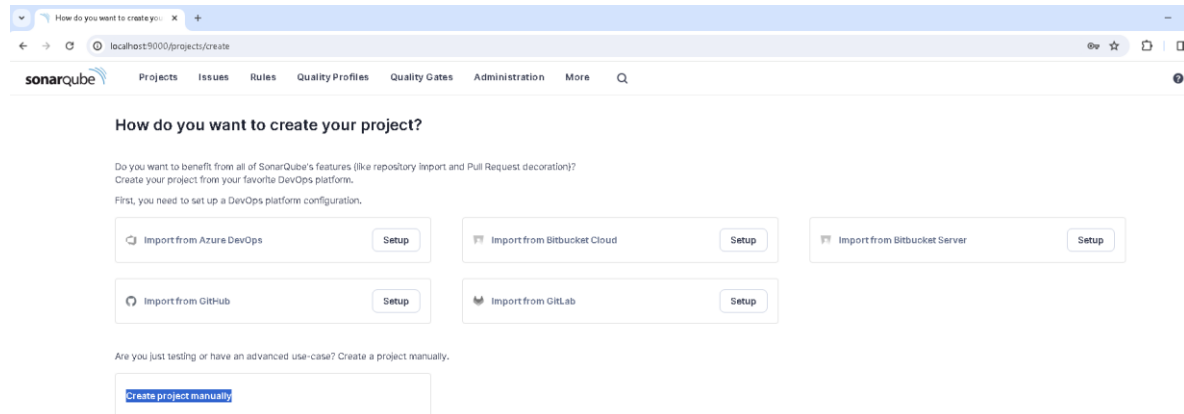


```
NAMED --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED --add-exports=java.base/jdk.internal.ref=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED --add-opens=java.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.management/sun.management=ALL-UNNAMED --add-opens=jdk.management/com.sun.management.internal=ALL-UNNAMED -Dcom.redhat.fips=false -Xmx512m -Xms128m -XX:+HeapDumpOnOutOfMemoryError -Dhttp.nonProxyHosts=localhost|127.*|::1 -cp ./lib/sonar-application-10.2.1.78527.jar;C:\Temporal\sonarqube\sonarqube-10.2.1.78527\lib\jdbc\h2\h2-2.1.214.jar org.sonar.server.app.WebServer C:\Temporal\sonarqube\sonarqube-10.2.1.78527\temp\squ-process-11826255893669450457properties
WARNING: A terminally deprecated method in java.lang.System has been called
WARNING: System::setSecurityManager has been called by org.sonar.process.PluginSecurityManager (file:/C:/Temporal/sonarqube/sonarqube-10.2.1.78527/lib/sonar-application-10.2.1.78527.jar)
WARNING: Please consider reporting this to the maintainers of org.sonar.process.PluginSecurityManager
WARNING: System::setSecurityManager will be removed in a future release
2023.11.08 21:15:22 INFO app[[o.s.a.SchedulerImpl] Process[web] is up
2023.11.08 21:15:22 INFO app[[o.s.a.ProcessLauncherImpl] Launch process[COMPUTE_ENGINE] from [C:\Temporal\sonarqube\sonarqube-10.2.1.78527]: C:\Program Files\Java\jdk-17\bin\java -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=C:\Temporal\sonarqube\sonarqube-10.2.1.78527\temp -XX:-OmitStackTraceInFastThrow --add-opens=java.base/java.util=ALL-UNNAMED --add-exports=java.base/jdk.internal.ref=ALL-UNNAMED --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED --add-opens=java.management/sun.management=ALL-UNNAMED --add-opens=jdk.management/com.sun.management.internal=ALL-UNNAMED -Dcom.redhat.fips=false -Xmx512m -Xms128m -XX:+HeapDumpOnOutOfMemoryError -Dhttp.nonProxyHosts=localhost|127.*|::1 -cp ./lib/sonar-application-10.2.1.78527.jar;C:\Temporal\sonarqube\sonarqube-10.2.1.78527\lib\jdbc\h2\h2-2.1.214.jar org.sonar.ce.app.CeServer C:\Temporal\sonarqube\sonarqube-10.2.1.78527\temp\squ-process-16591190981627606178properties
WARNING: A terminally deprecated method in java.lang.System has been called
WARNING: System::setSecurityManager has been called by org.sonar.process.PluginSecurityManager (file:/C:/Temporal/sonarqube/sonarqube-10.2.1.78527/lib/sonar-application-10.2.1.78527.jar)
WARNING: Please consider reporting this to the maintainers of org.sonar.process.PluginSecurityManager
WARNING: System::setSecurityManager will be removed in a future release
2023.11.08 21:15:30 INFO app[[o.s.a.SchedulerImpl] Process[ce] is up
2023.11.08 21:15:30 INFO app[[o.s.a.SchedulerImpl] SonarQube is operational
```

9. Ahora abriremos la web con el LINK: <http://localhost:9000/>

El user y password es “admin/admin” y cuando lo ejecutamos por primera vez nos pedirá cambiar la contraseña

10. Crearemos ahora nuestro primer proyecto de Inspección de código, dando click en el link **“Create project manually”**



11. Llenaremos los campos en blanco y daremos a “Next”

Create a project

Project display name *



Up to 255 characters. Some scanners might override the value you provide.

Project key *



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.

Main branch name *

The name of your project's default branch [Learn More](#)

Next

12. Seleccionar “Use the global setting” y dar click en “Create project”

Choose the baseline for new code for this project

☒ **Use the global setting**

Previous version

Any code that has changed since the previous version is considered new code.
Recommended for projects following regular versions or releases.

☐ **Define a specific setting for this project**

☐ **Previous version**

Any code that has changed since the previous version is considered new code.
Recommended for projects following regular versions or releases.

☐ **Number of days**

Any code that has changed in the last x days is considered new code. If no action is taken c
Recommended for projects following continuous delivery.

☐ **Reference branch**

Choose a branch as the baseline for the new code.
Recommended for projects using feature branches.

Create project

13. Como alojaremos nuestro proyecto de localmente dar click en “Locally”

Locally

Use this for testing or advanced use-case. Other modes are
recommended to help you set up your CI environment.

14. Generar token y dar click en "Continue"

1 Provide a token

Analyze "Python": `sqp_0cbdf536f32dd31bad483798a1d37fe430de1a99` 

The token is used to identify you when an analysis is performed. If it has been compromised, you can revoke it at any point in time in your [user account](#).

Continue

15. Elegir el lenguaje de programación (la gran mayoría esta en "other"), el SO donde correremos el SonarScanner y copiamos el comando que nos entrega:

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

Download and unzip the Scanner for Windows

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.bat -D"sonar.projectkey=Python2" -D"sonar.sources=" -D"sonar.host.url=http://localhost:9000" -D"sonar.token=sqp_0cbdf536f32dd31bad483798a1d37fe430de1a99"
```

 Copy

16. Ahora debemos desde la terminal ir a la ruta donde esta nuestro código, copiar el comando que copiamos en el paso anterior y esperar que termine la inspección. Al finalizar podremos ver el resultado en la web de SonarQube.