



UNIVERSIDAD PRIVADA DE TACNA
FACULTAD DE INGENIERIA
ESCUELA PROFESIONAL DE INGENIERIA DE SISTEMAS

TITULO:

Informe de Laboratorio Nro01

CURSO:

Calidad y Pruebas de Software

DOCENTE:

Ing. Patrick Cuadros Quiroga

INTEGRANTES:

Lisbeth Isabel Espinoza Caso (2011040667)

Tacna - Peru
2020

Índice

1. Paso 1:Descargar SonarQube	1
2. Paso 2: Ejecutar una instancia de SonarQube	1
3. Paso 3: Ingresar al portal con las credenciales	1
4. Paso 4: Crear una nueva aplicación con el nombre aplicacionNetCore	3
5. Paso 5: Generar el token de la nueva aplicación aplicacionNetCore	3
6. Paso 6: Decargar Net Core e instalar	5
7. Paso 7: En un terminal ejecutar e instalar sonar-scanner	6
8. Paso 8: En un terminal, acceder a una ruta donde creara una nueva aplicacion	7
9. Paso 9: En el mismo terminal, iniciar la sesión de revisión de sonarqube	7
10.Paso 10: Compilar la aplicación	8
11.Paso 11: Cerramos la sesión	8
12.Paso 12: Resultado Final	10

1. Paso 1: Descargar SonarQube

`docker pull sonarqube`

```
PS C:\Users\lisy> docker pull sonarqube
Using default tag: latest
latest: Pulling from library/sonarqube
cbbde7a5bc2a: Already exists
ad75d47265c8: Pull complete
43f3acb0c861: Pull complete
ff0605372de9: Pull complete
68aade4336af: Pull complete
Digest: sha256:452e87fe1f932a920bb9546ce0ad148f897565e752fe342058489214b0275c1b
Status: Downloaded newer image for sonarqube:latest
docker.io/library/sonarqube:latest
PS C:\Users\lisy>
```

2. Paso 2: Ejecutar una instancia de SonarQube

– `docker run -d --name sonarqube -p 9000:9000 sonarqube`

```
PS C:\Users\lisy> docker run -d --name sonarqube -p 9000:9000 sonarqube
4326df20658001a3dbbaa83dc67eddc1bf360fe1d46dc9ba418d8f51e9465974
PS C:\Users\lisy>
```

3. Paso 3: Ingresar al portal con las credenciales

- `http://localhost:9000/`
- `192.168.99.100:9000`
- user: admin
- pass: admin

Calidad continua del código

Inicia sesión

Leer documentación

0
Proyectos

0 🐛 bugs

0 🛡 Vulnerabilidades

0 🗑 Olores de código

Multi-Idioma

SonarQube admite más de 20 lenguajes de programación, entre los que se incluyen:

[VB.NET](#) [De Java](#)

[C/C++](#)

[T-SQL](#) [de C-](#)

[COBOL](#)

[ABAP](#)

[HTML](#)

[RPG](#)

[JavaScript](#)

[TypeScript](#)

[Objetivo C](#)

[Xml](#)

[PL/SQL](#)

[Flex](#)

[Python](#)

[Groovy](#)

[PHP](#)

[Swift](#)

[Visual Basic](#)

[PL/I](#)

localhost:9000/sessions/new

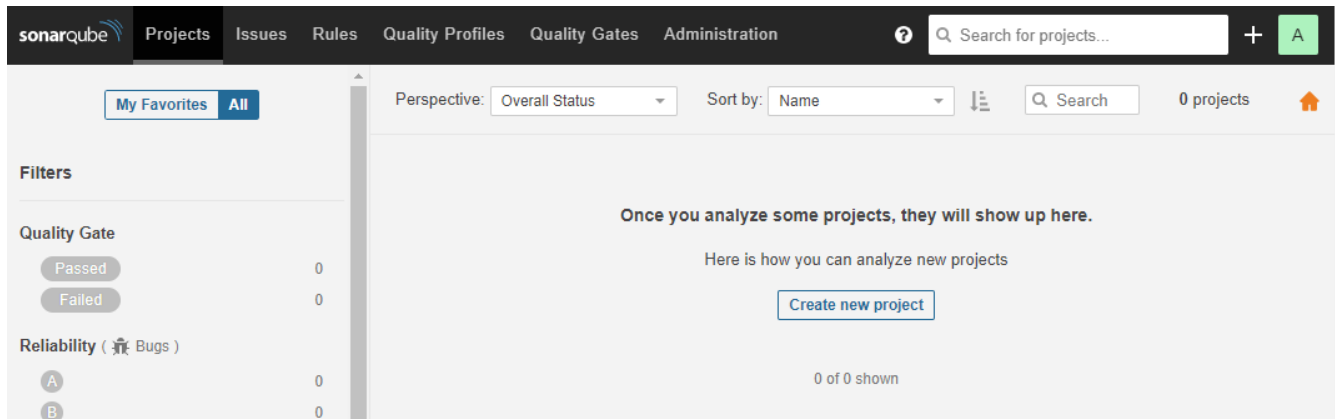
Iniciar sesión en SonarQube

admin

.....

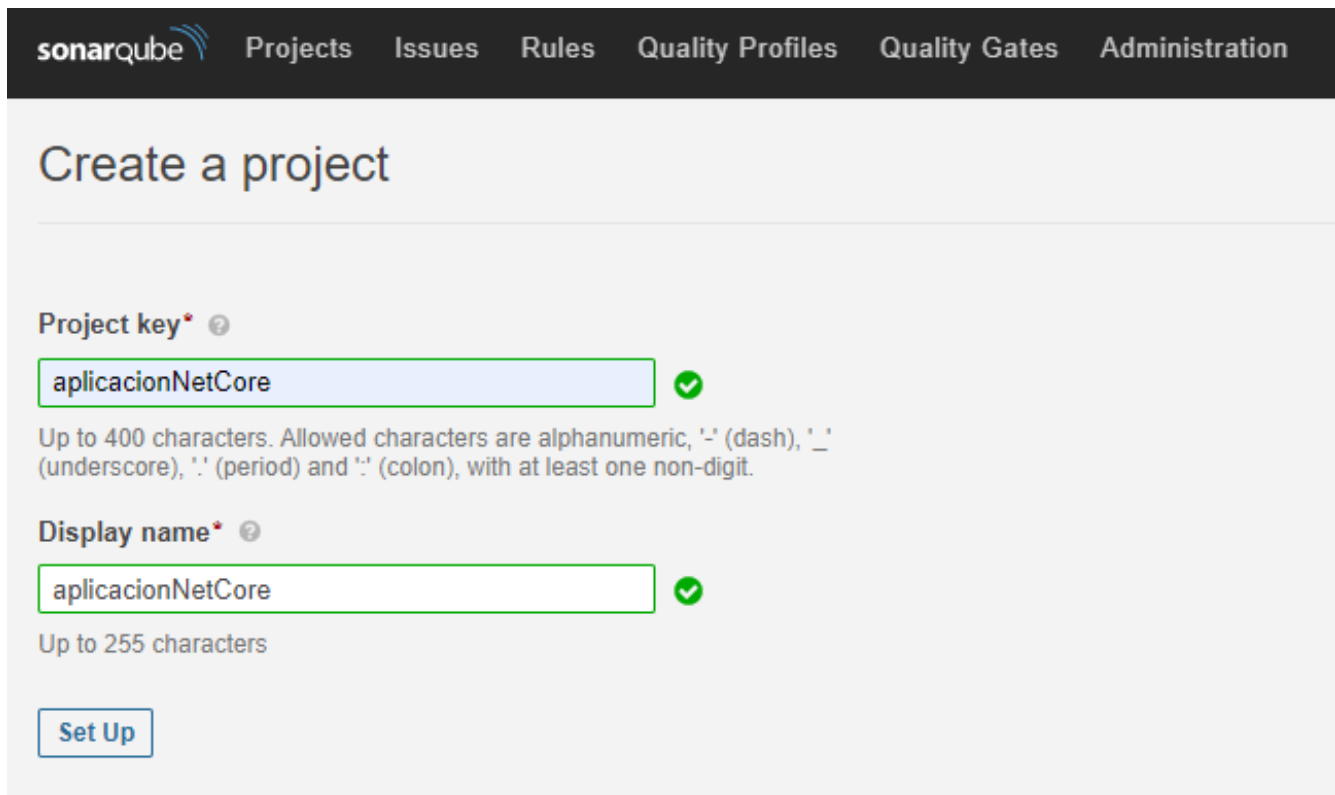
Inicia sesión Cancelar

4. Paso 4: Crear una nueva aplicación con el nombre aplicacionNetCore



5. Paso 5: Generar el token de la nueva aplicación aplicacionNetCore

— debera devolver 8a15d2a89c8636f15eb32ebee0993b8d16bff94e



aplicacionNetCore ☆ master +

Overview

Issues

Security Hotspots

Measures

Code

Activity

Analyze your project

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

1 Provide a token

☒ Generate a token

Generate

☐ Use existing token

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](#).

2 Run analysis on your project

aplicacionNetCore ☆ master +

Overview

Issues

Security Hotspots

Measures

Code

Activity

Analyze your project

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

1 Provide a token

aplicacionNetCore: 3f404a6621617d0aa5283d89fef055d48c20d4f7



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](#).

Continue

1 Provide a token

2 Run analysis on your project

What is your project's main language?

Java

C# or VB.NET

Other (JS, TS, Go, Python, PHP, ...)

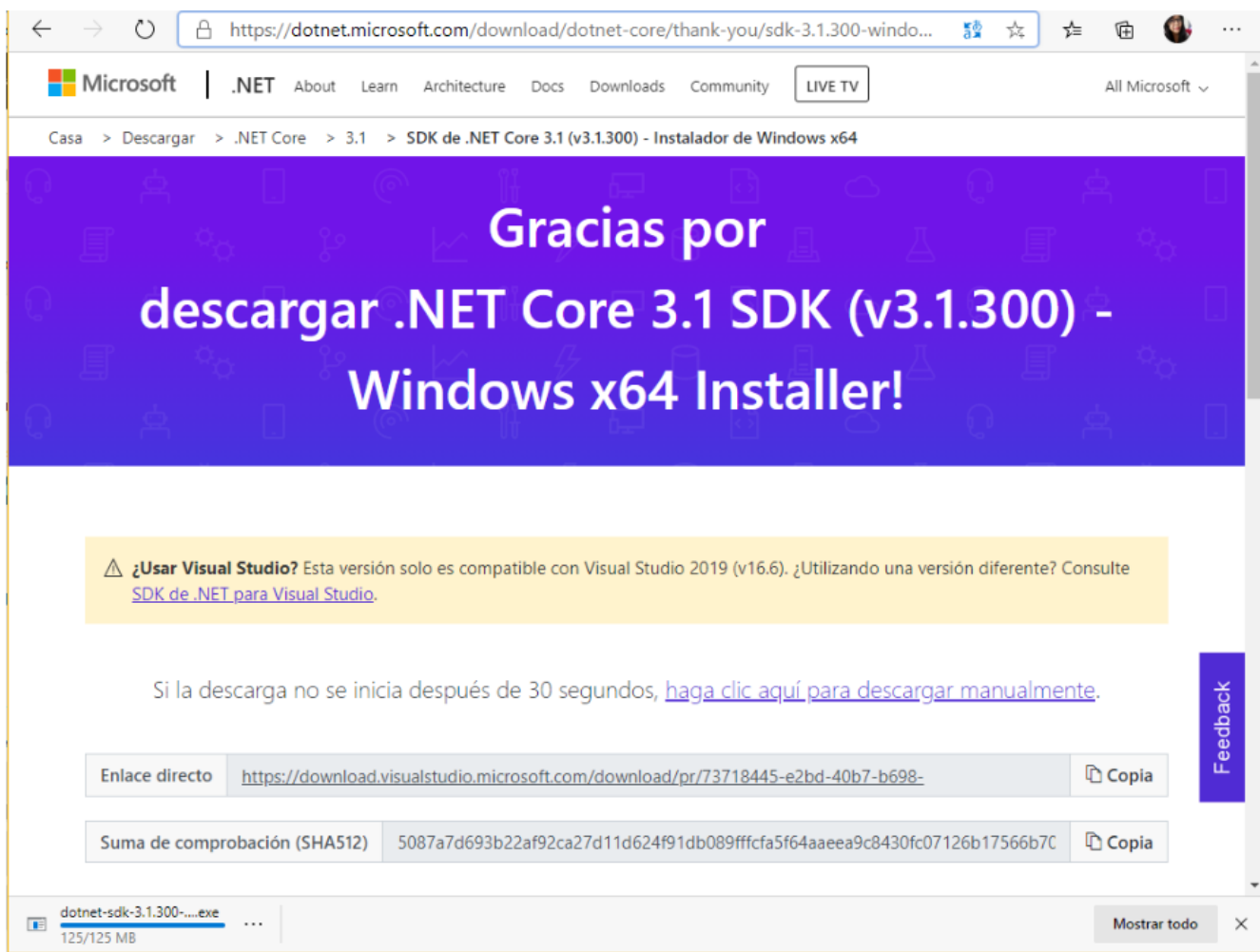
Download and unzip the Scanner for MSBuild

And add the executable's directory to the `%PATH%` environment variable

Download

6. Paso 6: Descargar Net Core e instalar

- <https://dotnet.microsoft.com/download/dotnet-core/thank-you/sdk-3.1.300-windows-x64-installer>



7. Paso 7: En un terminal ejecutar e instalar sonar-scanner

— dotnet tool install --global dotnet-sonarscanner

```

PS C:\Users\lisy> dotnet tool install --global dotnet-sonarscanner

Esto es .NET Core 3.1.
-----
Versión del SDK: 3.1.300

Telemetría
-----
Las herramientas de .NET Core recopilan datos de uso para ayudarnos a mejorar su experiencia. Estos datos son anónimos. Microsoft los recopila y comparte con la comunidad. Puede optar por no participar en la telemetría si establece la variable de entorno DOTNET_CLI_TELEMETRY_OPTOUT en "1" o "true" mediante su shell favorito.

Lea más sobre la telemetría de las herramientas de la CLI de .NET Core: https://aka.ms/dotnet-cli-telemetry

-----
Explore la documentación: https://aka.ms/dotnet-docs
Informe de los problemas y busque código fuente en GitHub: https://github.com/dotnet/core
Conozca las novedades: https://aka.ms/dotnet-whats-new
Más información sobre el certificado de desarrollador HTTPS instalado: https://aka.ms/aspnet-core-https
Use "dotnet --help" para ver los comandos disponibles o visite: https://aka.ms/dotnet-cli-docs
Escriba su primera aplicación: https://aka.ms/first-net-core-app

-----
Puede invocar la herramienta con el comando siguiente: dotnet-sonarscanner
La herramienta "dotnet-sonarscanner" (versión '5.0.2') se instaló correctamente.
PS C:\Users\lisy>

```


8. Paso 8: En un terminal, acceder a una ruta donde creara una nueva aplicacion

- dotnet new sln -o aplicacionNetCore
- cd aplicacionNetCore
- dotnet new console
- dotnet sln aplicacionNetCore.sln add aplicacionNetCore.csproj

```
PS C:\Users\lisy> dotnet new sln -o aplicacionNetCore
Getting ready...
The template "Solution File" was created successfully.

An update for template pack Microsoft.DotNet.Common.ItemTemplates::3.1.4 is available.
  install command: dotnet new -i Microsoft.DotNet.Common.ItemTemplates::3.1.9
PS C:\Users\lisy> cd aplicacionNetCore
PS C:\Users\lisy\aplicacionNetCore> dotnet new console
The template "Console Application" was created successfully.

Processing post-creation actions...
Running 'dotnet restore' on C:\Users\lisy\aplicacionNetCore\aplicacionNetCore.csproj...
  Determinando los proyectos que se van a restaurar...
  Se ha restaurado C:\Users\lisy\aplicacionNetCore\aplicacionNetCore.csproj (en 139 ms).

Restore succeeded.

An update for template pack Microsoft.DotNet.Common.ProjectTemplates.3.1::3.1.4 is available.
  install command: dotnet new -i Microsoft.DotNet.Common.ProjectTemplates.3.1::3.1.9
PS C:\Users\lisy\aplicacionNetCore> dotnet sln aplicacionNetCore.sln add aplicacionNetCore.csproj
Se ha agregado el proyecto "aplicacionNetCore.csproj" a la solución.
PS C:\Users\lisy\aplicacionNetCore>
```

9. Paso 9: En el mismo terminal, iniciar la sesión de revisión de sonarqube

- dotnet SonarScanner begin /k:aplicacionNetCore/d:sonar.host.url="http://localhost:9000/d:sonar.login=admin/k:aplicacionNetCore"

```
PS C:\Users\lisy\aplicacionNetCore> dotnet sonarscanner begin /d:sonar.host.url="http://localhost:9000" /d:sonar.login=admin
/d:sonar.password=admin /k:"aplicacionNteCore"
SonarScanner for MSBuild 5.0.2
Using the .NET Core version of the Scanner for MSBuild
Pre-processing started.
Preparing working directories...
12:57:40.752 Updating build integration targets...
12:57:40.895 Fetching analysis configuration settings...
12:57:43.499 Provisioning analyzer assemblies for cs...
12:57:43.502 Installing required Roslyn analyzers...
12:57:44.984 Provisioning analyzer assemblies for vbnet...
12:57:56.238 Installing required Roslyn analyzers...
12:57:56.284 Pre-processing succeeded.
PS C:\Users\lisy\aplicacionNetCore> dotnet build
```

10. Paso 10: Compilar la aplicación

- dotnet build

```
PS C:\Users\lisy\aplicacionNetCore> dotnet build
Microsoft (R) Build Engine versión 16.6.0+5ff7b0c9e para .NET Core
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Determinando los proyectos que se van a restaurar...
Todos los proyectos están actualizados para la restauración.
aplicacionNetCore -> C:\Users\lisy\aplicacionNetCore\bin\Debug\netcoreapp3.1\aplicacionNetCore.dll

Compilación correcta.
    0 Advertencia(s)
    0 Errores

Tiempo transcurrido 00:00:00.95
PS C:\Users\lisy\aplicacionNetCore>
```

11. Paso 11: Cerramos la sesión

- dotnet SonarScanner end /d:sonar.login=admin/d:sonar.password=admin

```
PS C:\Users\lisy\aplicacionNetCore> dotnet sonarscanner end /d:sonar.login=admin /d:sonar.password=admin

SonarScanner for MSBuild 5.0.2
Using the .NET Core version of the Scanner for MSBuild
Post-processing started.
Calling the SonarScanner CLI...
INFO: Scanner configuration file: C:\Users\lisy\.dotnet\tools\store\dotnet-sonarscanner\5.0.2\dotnet-sonarscanner\5.0.2\
tools\netcoreapp3.0\any\sonar-scanner-4.4.0.2170\bin\..\conf\sonar-scanner.properties
INFO: Project root configuration file: C:\Users\lisy\aplicacionNetCore\.sonarqube\out\sonar-project.properties
INFO: SonarScanner 4.4.0.2170
INFO: Java 10.0.2 Oracle Corporation (64-bit)
INFO: Windows 10 10.0 amd64
INFO: User cache: C:\Users\lisy\.sonar\cache
INFO: Scanner configuration file: C:\Users\lisy\.dotnet\tools\store\dotnet-sonarscanner\5.0.2\dotnet-sonarscanner\5.0.2\
tools\netcoreapp3.0\any\sonar-scanner-4.4.0.2170\bin\..\conf\sonar-scanner.properties
INFO: Project root configuration file: C:\Users\lisy\aplicacionNetCore\.sonarqube\out\sonar-project.properties
INFO: Analyzing on SonarQube server 8.5.1
INFO: Default locale: "es_ES", source code encoding: "windows-1252" (analysis is platform dependent)
WARN: SonarScanner will require Java 11 to run starting in SonarQube 8.x
INFO: Load global settings
INFO: Load global settings (done) | time=394ms
INFO: Server id: BF41A1F2-AXWji_SB0qx4-bVsakuk
INFO: User cache: C:\Users\lisy\.sonar\cache
INFO: Load/download plugins
INFO: Load plugins index
INFO: Load plugins index (done) | time=377ms
INFO: Load/download plugins (done) | time=8041ms
INFO: Process project properties
INFO: Process project properties (done) | time=10ms
INFO: Execute project builders
INFO: Execute project builders (done) | time=68ms
INFO: Project key: aplicacionNteCore
INFO: Base dir: C:\Users\lisy\aplicacionNetCore
INFO: Working dir: C:\Users\lisy\aplicacionNetCore\.sonarqube\out\sonar
INFO: Load project settings for component key: 'aplicacionNteCore'
INFO: Load quality profiles
INFO: Load quality profiles (done) | time=697ms
INFO: Load active rules
```

```

INFO: Load active rules (done) | time=6548ms
WARN: SCM provider autodetection failed. Please use "sonar.scm.provider" to define SCM of your project, or disable the
SCM Sensor in the project settings.
INFO: Indexing files...
INFO: Project configuration:
INFO: Indexing files of module 'aplicacionNetCore'
INFO:   Base dir: C:\Users\lisy\aplicacionNetCore
INFO:   Source paths: Program.cs
INFO: Load project repositories
INFO: Load project repositories (done) | time=325ms
INFO: Indexing files of module 'aplicacionNteCore'
INFO:   Base dir: C:\Users\lisy\aplicacionNetCore
INFO: 1 file indexed
INFO: Quality profile for cs: Sonar way
INFO: ----- Run sensors on module aplicacionNetCore
INFO: Load metrics repository
INFO: Load metrics repository (done) | time=360ms
INFO: Sensor CSS Rules [cssfamily]
INFO: No CSS, PHP, HTML or VueJS files are found in the project. CSS analysis is skipped.
INFO: Sensor CSS Rules [cssfamily] (done) | time=1ms
INFO: Sensor JaCoCo XML Report Importer [jacoco]
INFO: 'sonar.coverage.jacoco.xmlReportPaths' is not defined. Using default locations: target/site/jacoco/jacoco.xml,tar
get/site/jacoco-it/jacoco.xml,build/reports/jacoco/test/jacocoTestReport.xml
INFO: No report imported, no coverage information will be imported by JaCoCo XML Report Importer
INFO: Sensor JaCoCo XML Report Importer [jacoco] (done) | time=5ms
INFO: Sensor C# Properties [csharp]
INFO: Sensor C# Properties [csharp] (done) | time=41ms
INFO: Sensor JavaXmlSensor [java]
INFO: Sensor JavaXmlSensor [java] (done) | time=2ms
INFO: Sensor HTML [web]
INFO: Sensor HTML [web] (done) | time=5ms
INFO: Sensor VB.NET Properties [vbnet]
INFO: Sensor VB.NET Properties [vbnet] (done) | time=2ms
INFO: ----- Run sensors on module aplicacionNteCore
INFO: Sensor CSS Rules [cssfamily]
INFO: No CSS, PHP, HTML or VueJS files are found in the project. CSS analysis is skipped.
INFO: Sensor CSS Rules [cssfamily] (done) | time=0ms
INFO: Sensor JaCoCo XML Report Importer [jacoco]
INFO: 'sonar.coverage.jacoco.xmlReportPaths' is not defined. Using default locations: target/site/jacoco/jacoco.xml,tar
get/site/jacoco-it/jacoco.xml,build/reports/jacoco/test/jacocoTestReport.xml
INFO: No report imported, no coverage information will be imported by JaCoCo XML Report Importer
INFO: Sensor JaCoCo XML Report Importer [jacoco] (done) | time=1ms
INFO: Sensor C# Properties [csharp]
INFO: Sensor C# Properties [csharp] (done) | time=0ms
INFO: Sensor JavaXmlSensor [java]
INFO: Sensor JavaXmlSensor [java] (done) | time=0ms
INFO: Sensor VB.NET Properties [vbnet]
INFO: Sensor VB.NET Properties [vbnet] (done) | time=1ms
INFO: ----- Run sensors on project
INFO: Sensor C# [csharp]

```

```

INFO: Importing results from 5 proto files in 'C:\Users\lisy\aplicacionNetCore\.sonarqube\out\0\output-cs'
INFO: Importing 1 Roslyn report
INFO: Sensor C# [csharp] (done) | time=171ms
INFO: Sensor Zero Coverage Sensor
INFO: Sensor Zero Coverage Sensor (done) | time=20ms
INFO: SCM Publisher No SCM system was detected. You can use the 'sonar.scm.provider' property to explicitly specify it
.
INFO: CPD Executor 1 file had no CPD blocks
INFO: CPD Executor Calculating CPD for 0 files
INFO: CPD Executor CPD calculation finished (done) | time=0ms
INFO: Analysis report generated in 75ms, dir size=83 KB
INFO: Analysis report compressed in 24ms, zip size=12 KB
INFO: Analysis report uploaded in 673ms
INFO: ANALYSIS SUCCESSFUL, you can browse http://localhost:9000/dashboard?id=aplicacionNteCore
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis
report
INFO: More about the report processing at http://localhost:9000/api/ce/task?id=AXWj4sIs0qx4-bVsapgH
INFO: Analysis total time: 11.230 s
INFO: -----
INFO: EXECUTION SUCCESS
INFO: -----
INFO: Total time: 22.237s
INFO: Final Memory: 12M/50M
INFO: -----
The SonarScanner CLI has finished
13:05:48.588 Post-processing succeeded.
PS C:\Users\lisy\aplicacionNetCore>

```

12. Paso 12: Resultado Final

- Asi queda nuestro proyecto final.

