

**UNIVERSIDAD PRIVADA DE TACNA**  
**FACULTAD DE INGENIERÍA**  
**Escuela Profesional de Ingeniería de Sistemas**



**SESION DE LABORATORIO N° 01: PRUEBAS  
ESTATICAS DE SEGURIDAD DE APLICACIONES  
CON SONARQUBE”**

Curso: ***CALIDAD Y PRUEBAS DE SOFTWARE***

Docente: ***Mag. Patrick Cuadros Quiroga****Página*

Integrantes:

**Arce Bracamonte, Sebastian Rodrigo**      **(2019092986)**

**Tacna – Perú**

**2025**

## **1. Objetivos**

*Comprender el funcionamiento de las pruebas estáticas de seguridad de código en aplicaciones utilizando SonarQube.*

## **2. Requerimientos**

### **2.1 Conocimientos**

- *Conocimientos básicos de Bash (PowerShell).*
- *Conocimientos básicos de Contenedores (Docker).*

### **2.2 Hardware**

- *Virtualización activada en el BIOS.*
- *CPU con la característica SLAT-capable.*
- *Al menos 4 GB de RAM.*

### **2.3 Software**

- *Windows 10 64 bits (Pro, Enterprise o Education, versión 1607 o superior).*
- *Docker Desktop.*
- *PowerShell versión 7.x.*
- *.NET 8 o superior.*
- *Visual Studio Code.*

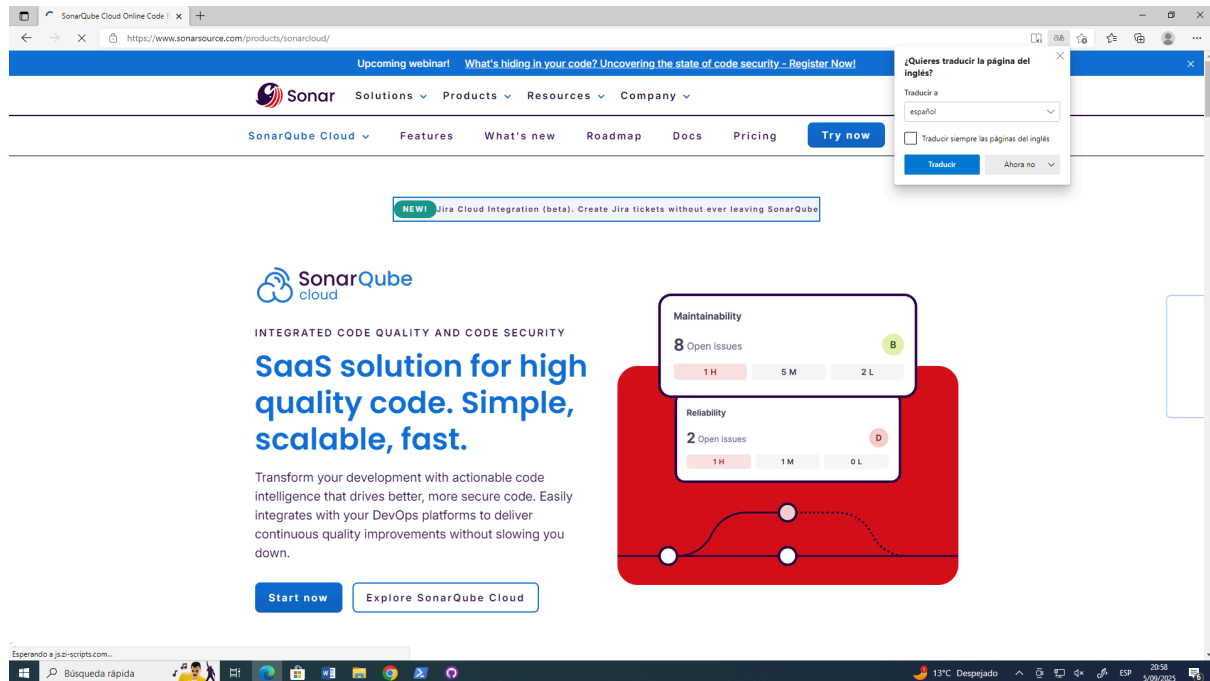
## **3. Consideraciones Iniciales**

- *Clonar el repositorio mediante Git para disponer de los recursos necesarios.*
- *Contar con una cuenta de GitHub válida.*

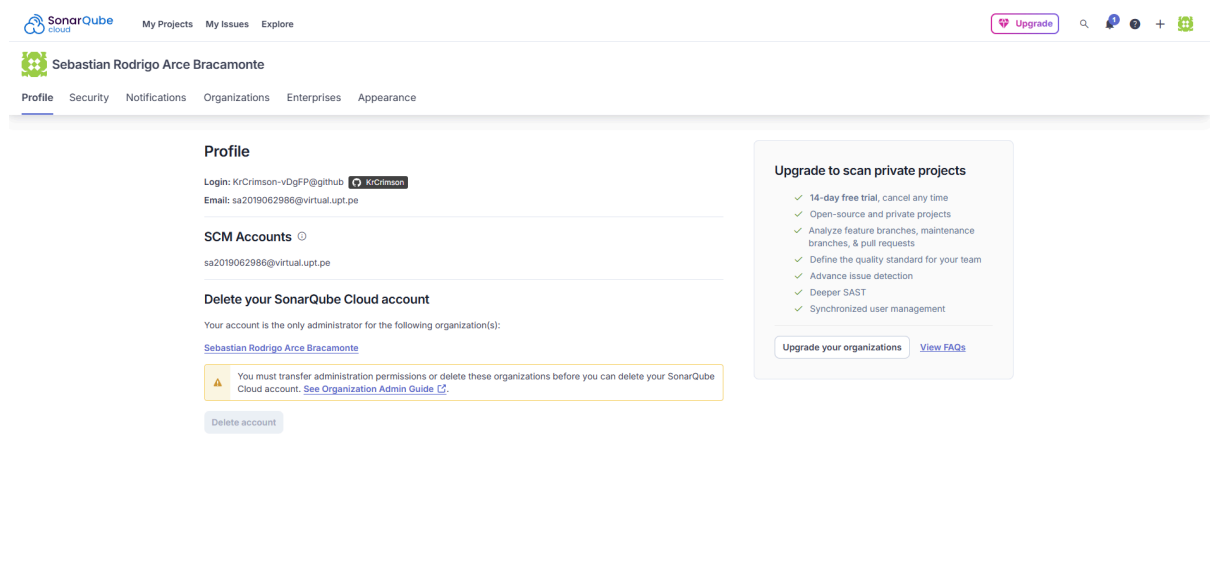
## **4. Desarrollo**

*Parte I: Configuración de la Herramienta SonarQube*

*Ingresar a SonarCloud e iniciar sesión con su cuenta de GitHub.*



*Acceder a My Account y generar un nuevo token. Guardarlo de forma segura.*



*Crear un nuevo proyecto en SonarCloud Projects con el nombre apibank. Guardar el Project Key.*

*Finalizar la creación del proyecto seleccionando Previous Version y luego Create Project.*

*Parte II: Creación de la Aplicación*

*Abrir PowerShell o Windows Terminal en modo administrador.*

*Crear la solución:*

*dotnet new sln -o Bank*

```

Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> dotnet new sln -o Bank

Esto es .NET 8.0.
-----
Versión del SDK: 8.0.400

Telemetría
-----
Las herramientas de .NET recopilan datos de uso para ayudarnos a mejorar su experiencia. Microsoft los recopila y los 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: https://aka.ms/dotnet-cli-telemetry

-----
Instalar un certificado de desarrollo HTTPS de ASP.NET Core.
Para confiar en el certificado, ejecute "dotnet dev-certs https --trust"
Obtenga información sobre HTTPS: https://aka.ms/dotnet-https

-----
Escribir su primera aplicación: https://aka.ms/dotnet-hello-world
Descubra las novedades: https://aka.ms/dotnet-whats-new
Explore la documentación: https://aka.ms/dotnet-docs
Notificar problemas y encontrar el código fuente en GitHub: https://github.com/dotnet/core
Use "dotnet --help" para ver los comandos disponibles o visite: https://aka.ms/dotnet-cli
-----
La plantilla "Archivo de la solución" se creó correctamente.

PS C:\Windows\system32>

```

Crear una librería de clases:

*cd Bank*

*dotnet new classlib -o Bank.Domain*

*dotnet sln add ./Bank.Domain/Bank.Domain.csproj*

```

PS C:\Windows\system32> cd Bank
>> dotnet new classlib -o Bank.Domain
>> dotnet sln add ./Bank.Domain/Bank.Domain.csproj
La plantilla "Biblioteca de clases" se creó correctamente.

Procesando acciones posteriores a la creación...
Restaurando C:\Windows\system32\Bank\Bank.Domain\Bank.Domain.csproj:
  Determinando los proyectos que se van a restaurar...
  Se ha restaurado C:\Windows\system32\Bank\Bank.Domain\Bank.Domain.csproj (en 44 ms).
Restauración realizada correctamente.

Se ha agregado el proyecto "Bank.Domain\Bank.Domain.csproj" a la solución.
PS C:\Windows\system32\Bank>
PS C:\Windows\system32\Bank> S

```

Crear un proyecto de pruebas:

*dotnet new mstest -o Bank.Domain.Tests*

*dotnet sln add ./Bank.Domain.Tests/Bank.Domain.Tests.csproj*

*dotnet add ./Bank.Domain.Tests/Bank.Domain.Tests.csproj reference*

*./Bank.Domain/Bank.Domain.csproj*

```
PS C:\Windows\system32\Bank> dotnet new mstest -o Bank.Domain.Tests
>> dotnet sln add ./Bank.Domain.Tests/Bank.Domain.Tests.csproj
>> dotnet add ./Bank.Domain.Tests/Bank.Domain.Tests.csproj reference ./Bank.Domain/Bank.Domain.csproj
La plantilla "MSTest Test Project" se creó correctamente.

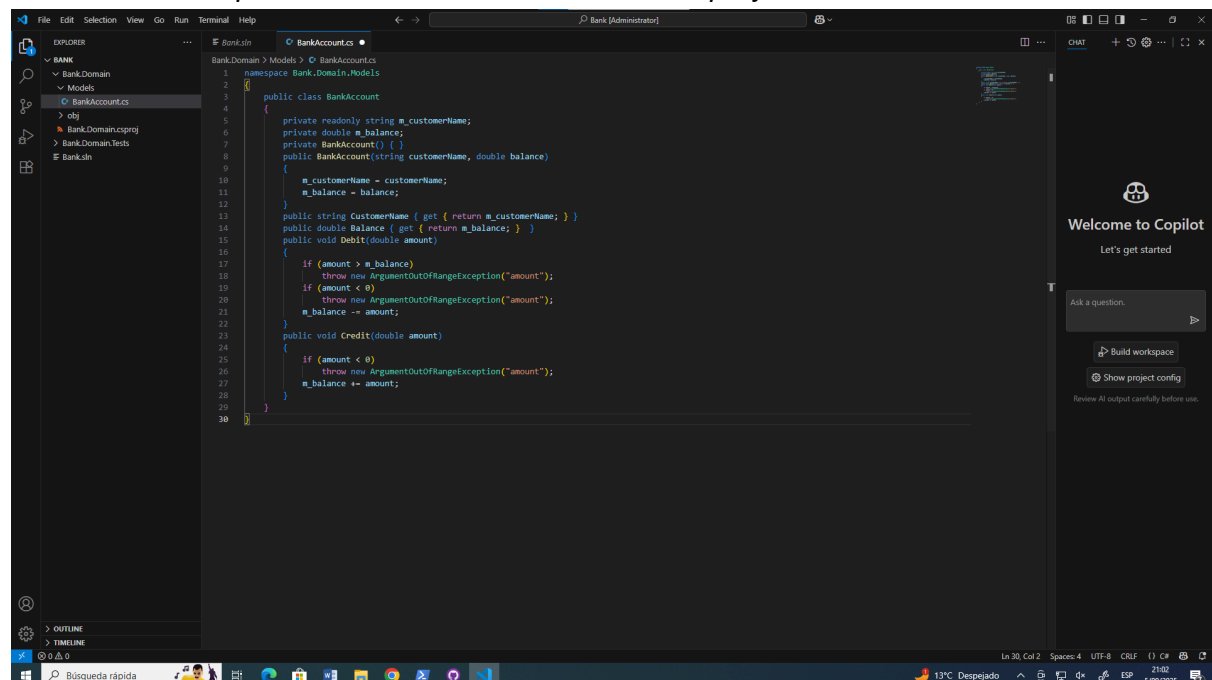
Procesando acciones posteriores a la creación...
Restaurando C:\Windows\system32\Bank\Bank.Domain.Tests\Bank.Domain.Tests.csproj:
Determinando los proyectos que se van a restaurar...
Se ha restaurado C:\Windows\system32\Bank\Bank.Domain.Tests\Bank.Domain.Tests.csproj (en 3.48 s).
Restauración realizada correctamente.

Se ha agregado el proyecto "Bank.Domain.Tests\Bank.Domain.Tests.csproj" a la solución.
Se ha agregado la referencia "..\Bank.Domain\Bank.Domain.csproj" al proyecto.
PS C:\Windows\system32\Bank>
```

Eliminar archivos innecesarios (Class1.cs y UnitTest1.cs).

Crear la clase BankAccount.cs en la carpeta Models.

Crear la clase de pruebas BankAccountTests.cs en el proyecto de tests.



Ejecutar pruebas unitarias:

`dotnet test --collect:"XPlat Code Coverage"`

```
08:43:40.007 Pre-processing dotnet test --collect:"XPlat Code Coverage"
Determinando los proyectos que se van a restaurar...
Se ha restaurado C:\Windows\System32\Bank\Bank.Domain.Tests\Bank.Domain.Tests.csproj (en 1.02 s).
1 de 2 proyectos están actualizados para la restauración.
C:\Windows\System32\Bank\Bank.Domain\BankAccount.cs(7,17): warning CS8618: El elemento campo "m_customerName" que no acepta valores NULL debe contener un valor distinto de NULL al salir del constructor.
Considere la posibilidad de agregar el modificador "required" o declarar el campo como un valor que acepta valores NULL. [C:\Windows\System32\Bank\Bank.Domain\Bank.Domain.csproj]
Bank.Domain -> C:\Windows\System32\Bank\Bank.Domain\bin\Debug\net7.0\Bank.Domain.dll
Sonar: (Bank.Domain.csproj) Project processed successfully
Bank.Domain.Tests -> C:\Windows\System32\Bank\Bank.Domain.Tests\bin\Debug\net7.0\Bank.Domain.Tests.dll
Sonar: (Bank.Domain.Tests.csproj) Project processed successfully
Serie de pruebas para C:\Windows\System32\Bank\Bank.Domain.Tests\bin\Debug\net7.0\Bank.Domain.Tests.dll (.NETCoreApp,Version=v7.0)
Versión 17.11.1 (x64) de VSTest

Iniciando la ejecución de pruebas, espere...
1 archivos de prueba en total coincidieron con el patrón especificado.

Correctas! - Con error: 0, Superado: 1, Omitido: 0, Total: 1, Duración: 4 ms - Bank.Domain.Tests.dll (net7.0)

Datos adjuntos:
C:\Windows\System32\Bank\Bank.Domain.Tests\TestResults\690bfb1-32a5-4ebd-b3cb-8d467e56ac9b\coverage.cobertura.xml
PS C:\Windows\System32\Bank>
```

Instalar SonarScanner:

`dotnet tool install -g dotnet-sonarscanner`

```

PS C:\Windows\System32\Bank> dotnet tool install -g dotnet-sonarscanner
Puede invocar la herramienta con el comando siguiente: dotnet-sonarscanner
La herramienta "dotnet-sonarscanner" (versión '10.3.0') se instaló correctamente.
PS C:\Windows\System32\Bank>

```

Ejecutar análisis inicial:

```

dotnet sonarscanner begin /k:"PROJECT_KEY" /d:sonar.token="TOKEN"
/d:sonar.host.url="https://sonarcloud.io" /o:"ORGANIZATION"
/d:sonar.cs.opencover.reportsPaths="**/*/*coverage.opencover.xml"

```

```

PS C:\Windows\System32\Bank> dotnet sonarscanner begin /k:"kr crimson_apibank" /d:sonar.token="b762b07c4867aed19e56231d242bf63c0f6df8e5" /d:sonar.host.url="https://sonarcloud.io" /o:"kr crimson" /d:sonar.cs.opencover.reportsPaths="**/*/*coverage.opencover.xml"
-Bada-f5284815758250narScanner for MSBuild 10.3
Using the .NET Core version of the Scanner for MSBuild
Pre-processing started.
Preparing working directories...
08:43:22.641 Updating build integration targets...
08:43:27.313 Using SonarCloud.
08:43:27.811 The JRE provisioning is a time consuming operation.
JRE provisioned: OpenJDK17U-jre_x64_windows_hotspot_17.0.11_9.zip.
If you already have a compatible java version installed, please add either the parameter "/d:sonar.scanner.skipJreProvisioning=true" or "/d:sonar.scanner.javaExePath=<PATH>".
08:43:34.35 Fetching analysis configuration settings...
08:43:36.361 Provisioning analyzer assemblies for cs...
08:43:36.362 Installing required Roslyn analyzers...
08:43:36.362 Processing plugin: securitysharpfrontend version 11.9.0.39984
08:43:37.461 Processing plugin: cssharpenterprise version 10.15.0.120848
08:43:39.037 Provisioning analyzer assemblies for vbnet...
08:43:39.037 Installing required Roslyn analyzers...
08:43:39.039 Processing plugin: securityvbnetfrontend version 11.9.0.39984
08:43:39.596 Processing plugin: vbnetenterprise version 10.15.0.120848
08:43:39.988 Incremental PR analysis: Base branch parameter was not provided.
08:43:39.99 Cache data is empty. A full analysis will be performed.
08:43:40.007 Pre-processing succeeded.
PS C:\Windows\System32\Bank>

```

Compilar y ejecutar pruebas con cobertura:

```

dotnet build --no-incremental
dotnet test --collect:"XPlat Code Coverage;Format=opencover"
dotnet sonarscanner end /d:sonar.token="TOKEN"

```

```

PS C:\Windows\System32\Bank> dotnet build --no-incremental
Determinando los proyectos que se van a restaurar...
Todos los proyectos están actualizados para la restauración.
C:\Windows\System32\Bank\Bank.Domain\Bank.Account.cs(7,17): warning CS8618: El elemento campo "m_customerName" que no acepta valores NULL debe contener un valor distinto de NULL al salir del constructor. Considere la posibilidad de agregar el modificador "required" o declarar el campo como un valor que acepta valores NULL. [C:\Windows\System32\Bank\Bank.Domain\Bank.Domain.csproj]
Bank.Domain -> C:\Windows\System32\Bank\Bank.Domain\bin\Debug\net7.0\Bank.Domain.dll
Sonar: (Bank.Domain.csproj) Project processed successfully
Bank.Domain.Tests -> C:\Windows\System32\Bank\Bank.Domain.Tests\bin\Debug\net7.0\Bank.Domain.Tests.dll
Sonar: (Bank.Domain.Tests.csproj) Project processed successfully

Compilación correcta.

C:\Windows\System32\Bank\Bank.Domain\Bank.Account.cs(7,17): warning CS8618: El elemento campo "m_customerName" que no acepta valores NULL debe contener un valor distinto de NULL al salir del constructor. Considere la posibilidad de agregar el modificador "required" o declarar el campo como un valor que acepta valores NULL. [C:\Windows\System32\Bank\Bank.Domain\Bank.Domain.csproj]
1 Advertencia(s)
0 Errores

Tiempo transcurrido 00:00:02.08
PS C:\Windows\System32\Bank> dotnet test --collect:"XPlat Code Coverage;Format=opencover"
Determinando los proyectos que se van a restaurar...
Todos los proyectos están actualizados para la restauración.
C:\Windows\System32\Bank\Bank.Domain\Bank.Account.cs(7,17): warning CS8618: El elemento campo "m_customerName" que no acepta valores NULL debe contener un valor distinto de NULL al salir del constructor. Considere la posibilidad de agregar el modificador "required" o declarar el campo como un valor que acepta valores NULL. [C:\Windows\System32\Bank\Bank.Domain\Bank.Domain.csproj]
Bank.Domain -> C:\Windows\System32\Bank\Bank.Domain\bin\Debug\net7.0\Bank.Domain.dll
Sonar: (Bank.Domain.csproj) Project processed successfully
Bank.Domain.Tests -> C:\Windows\System32\Bank\Bank.Domain.Tests\bin\Debug\net7.0\Bank.Domain.Tests.dll
Sonar: (Bank.Domain.Tests.csproj) Project processed successfully
Serie de pruebas para C:\Windows\System32\Bank\Bank.Domain.Tests\bin\Debug\net7.0\Bank.Domain.Tests.dll (.NETCoreApp,Version=v7.0)
Versión 17.11.1 (x64) de VSTest

Iniciando la ejecución de pruebas, espere...
1 archivos de prueba en total coincidieron con el patrón especificado.

Correctas! - Con error: 0, Superado: 1, Omitido: 0, Total: 1, Duración: 4 ms - Bank.Domain.Tests.dll (net7.0)

Datos adjuntos:
C:\Windows\System32\Bank\Bank.Domain.Tests\TestResults\dd644c0a-b83e-4abe-b346-1286ad34ad2b\coverage.opencover.xml
PS C:\Windows\System32\Bank>

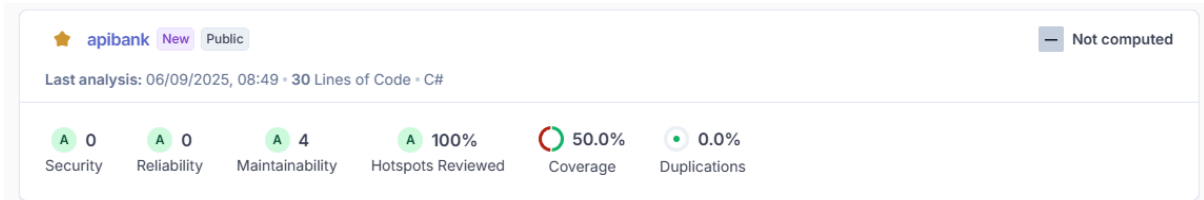
```

```

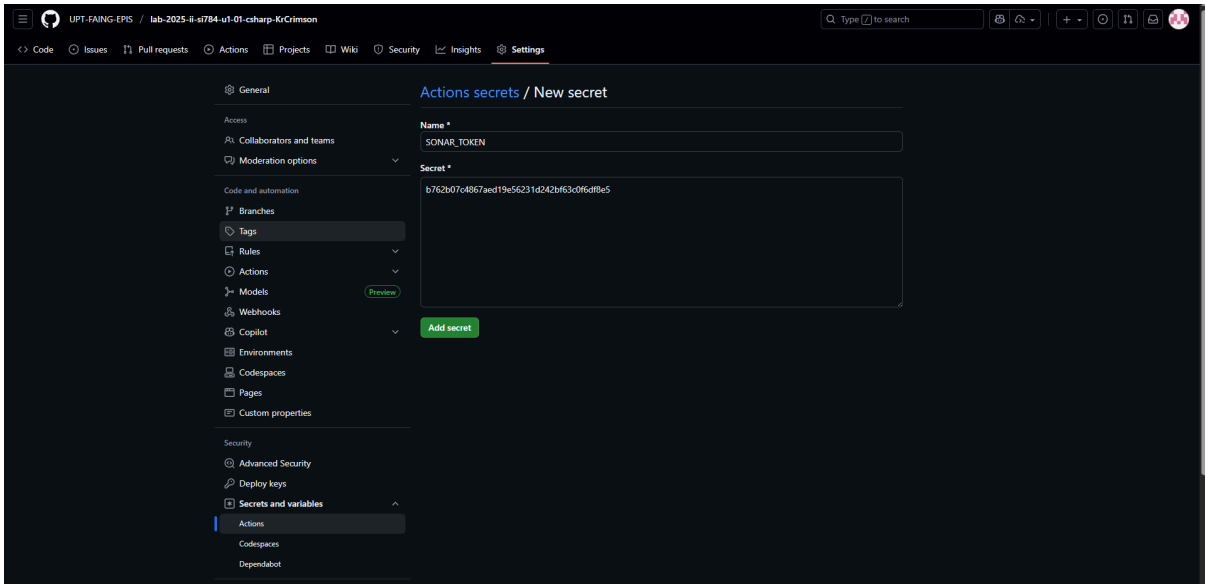
INFO: Sensor C# [csharpenterpriser] (done) | time=163ms
INFO: Sensor Analysis Warnings import [csharpenterpriser]
INFO: Sensor Analysis Warnings import [csharpenterpriser] (done) | time=2ms
INFO: Sensor C# File Caching Sensor [csharpenterpriser]
INFO: Sensor C# File Caching Sensor [csharpenterpriser] (done) | time=3ms
INFO: Sensor C# Tests Coverage Report Import [csharpenterpriser]
INFO: Parsing the OpenCover report C:\Windows\System32\Bank\Bank.Domain.Tests\TestResults\dd644c8a-b93e-4abe-b346-1286ad34ad2b\coverage.opencover.xml
INFO: Adding this code coverage report to the cache for later reuse: C:\Windows\System32\Bank\Bank.Domain.Tests\TestResults\dd644c8a-b93e-4abe-b346-1286ad34ad2b\coverage.opencover.x
INFO: Coverage Report Statistics: 1 files, 1 main files, 1 main files with coverage, 0 test files, 0 project excluded files, 0 other language files.
INFO: Sensor C# Tests Coverage Report Import [csharpenterpriser] (done) | time=54ms
INFO: Sensor (SharpArchitectureSensor [architecture]
INFO: Sensor (SharpArchitectureSensor [architecture] (done) | time=8ms
INFO: Sensor Zero Coverage Sensor
INFO: Sensor Zero Coverage Sensor (done) | time=0ms
INFO: ----- Gather SCA dependencies on project
INFO: Checking if SCA is enabled for organization krccrimson
INFO: Dependency analysis skipped
INFO: SCM Publisher No SCM system was detected. You can use the 'sonar.scm.provider' property to explicitly specify it.
INFO: CPD Executor Calculating CPD for 1 file
INFO: CPD Executor CPD calculation finished (done) | time=9ms
INFO: Create analysis
INFO: Create analysis (done) | time=0ms
INFO: Analysis report generated in 233ms, dir size=461 KB
INFO: Analysis report compressed in 18ms, zip size=80 KB
INFO: Analysis report uploaded in 923ms
INFO: ANALYSIS SUCCESSFUL, you can find the results at: https://sonarcloud.io/dashboard?id=krccrimson_apibank
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 https://sonarcloud.io/api/ce/task?id=A2kFSnR8evCaj0lPyGB1
INFO: ----- Upload SCA dependency files
INFO: Sensor cache published successfully
INFO: Inferred api base url 'https://api.sonarcloud.io' from host url 'https://sonarcloud.io'.
INFO: Analysis total time: 17.812 s
INFO: -----
INFO: EXECUTION SUCCESS
INFO: -----
INFO: Total time: 31.082s
INFO: Final Memory: 33M/136M
INFO: -----
The SonarScanner CLI has finished
08:50:00.985 Post-processing succeeded.
PS C:\Windows\System32\Bank>

```

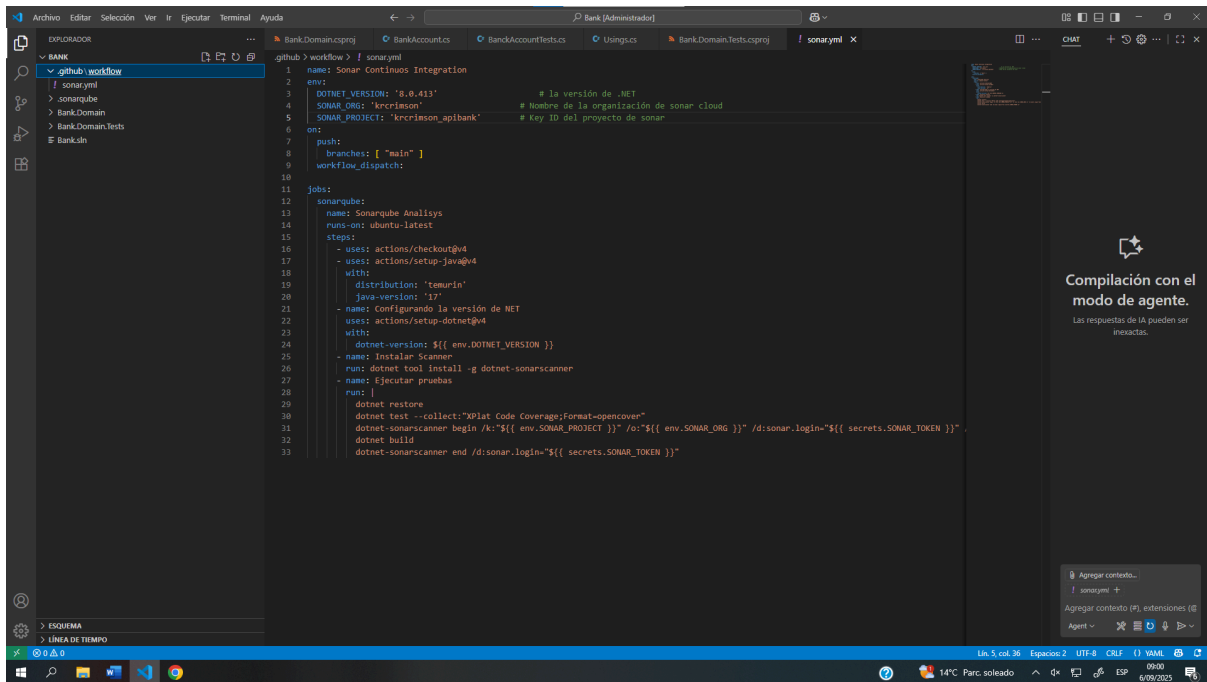
Verificar resultados en SonarCloud.



Configurar Secrets en GitHub (SONAR\_TOKEN).

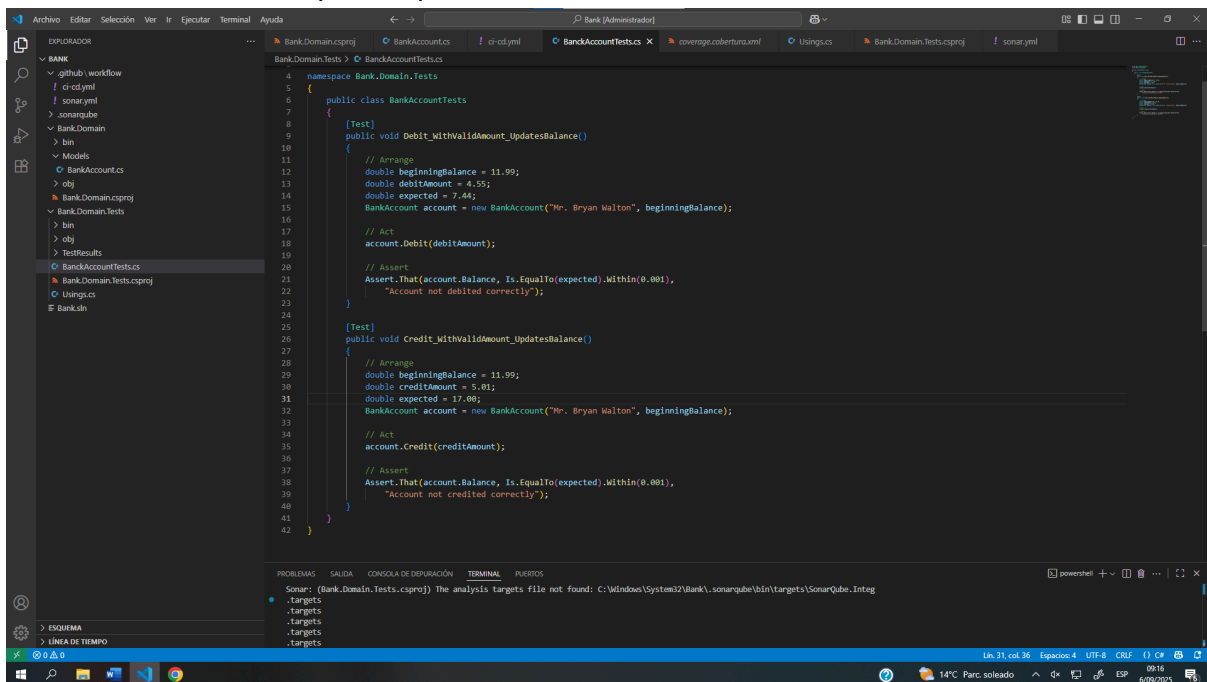


Crear el workflow .github/workflow/sonar.yml con integración continua en GitHub Actions.

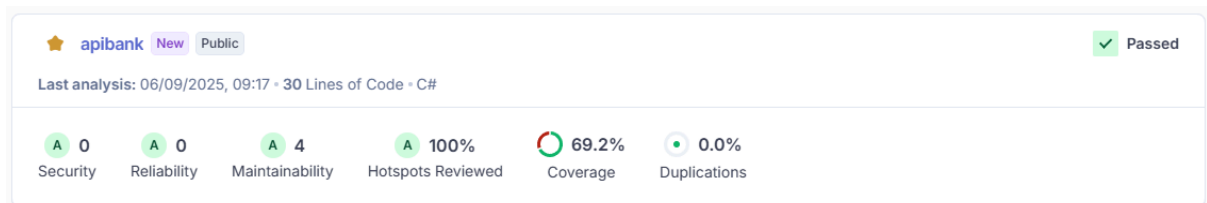


## 5. Actividades Encargadas

Adicionar un método de prueba para verificar el método Credit.



Adjuntar captura de pantalla mostrando el incremento de cobertura en SonarCloud (cobertura.png).



Automatizar la construcción y publicación del paquete .nuget en GitHub.



*Generar un release versión 1.0.0 del paquete NuGet, documentando los cambios en base a los commits realizados.*