

UNIVERSIDAD PRIVADA-DE-TACNA



INGENIERIA DE SISTEMAS

TITULO:

Mejoramiento de la aplicacion RandyStore

CURSO:

CALIDAD Y PRUEBAS DE SOFTWARE

DOCENTE:

Ing. Patrick Cuadros Quiroga

Integrantes:

Garcia Pinto, Marco Antonio	(2013046500)
Mejía Rodriguez, Julio Oliver	(2010037899)
Paredes Catacora, Randi Angel	(2013047246)
Chino, Alisson Rousse	(2015052821)
Herrera Amezquita, Derian Francisco	(2017059489)

Índice

1. Introduccion	1
2. Titulo	1
3. Autores	1
4. Planteamiento del problema	1
4.1. Problema	1
4.2. Justificacion	1
4.3. Alcance	1
5. Objetivos	1
5.1. General	1
5.2. Especifico	1
6. Referencias Teoricas	1
7. Desarrollo de la prueba	1
7.1. Tecnologia de informacion	1
7.2. Metodología, tecnicas usadas	2
8. Cronograma	9

Resumen

El presente proyecto , desarrollo de una sistema de control de clientes, personal e inventario del gimnasio Randys busca contribuir automatizar los procesos del area de ventas de productos deportivos. Tambien busca llevar un control de los clientes y de el personal con el que cuenta la tienda.

Abstrac

The present project, development of a control system for clients, personnel and Randys gym inventory seeks to help automate processes in the area sales of sports products. It also seeks to keep track of customers and staff with whom the company has store.

1. Introduccion

2. Titulo

Sistema de control de inventario y personal RandyStore.

3. Autores

4. Planteamiento del problema

4.1. Problema

4.2. Justificacion

4.3. Alcance

5. Objetivos

5.1. General

5.2. Especifico

6. Referencias Teoricas

7. Desarrollo de la prueba

Para realizar las pruebas en nuestro codigo utilizaremos la herramienta de **SonarQube**.

7.1. Tecnologia de informacion

Utilizamos **SonarQube** que es una plataforma para evaluar codigo fuente.Es de uso libre y usa diversas herramientas de analisis estatico como Checkstyle, PMD o FindBugs para obtener métricas que pueden ayudar a mejorar la calidad del código de un programa.

Tambien usamos el repositorio de **Azure DevOps** para realizar la programacion de nuestro codigo fuente en equipo.

Usamos el compilador de codigo fuente de **Visual Studio 2017** para realizar la programacion del sistema.

El gestor de base de datos **SQL Server** para la creacion de la base de datos.

7.2. Metodología, tecnicas usadas

Proceso de evaluacion del codigo

- Primero descargamos **SonarQube** de manera local en nuestro equipo, luego nos dirigimos a la direccion donde guardamos la descarga en el cmd y ejecutamos el bat "StarSonar.bat" para iniciar el servicio.

```
C:\> Símbolo del sistema

Microsoft Windows [Versión 10.0.19041.572]
(c) 2020 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Derian>cd C:\sonarqube-8.5.1.38104\bin\windows-x86-64

C:\sonarqube-8.5.1.38104\bin\windows-x86-64>StartSonar.bat
```

```
SonarQube

jvm 1 | 2020.10.30 21:08:33 INFO app[[o.s.a.ProcessLauncherImpl] Launch process[[key='es', ipcIndex=1, logFilenamePrefix=es]] from [C:\sonarqube-8.5.1.38104\elasticsearch]: C:\Program Files\Java\jdk-11.0.9\bin\java -XX:+UseConcMarkSweepGC -XX:CMSInitiatingOccupancyFraction=75 -XX:+UseCMSInitiatingOccupancyOnly -Des.networkaddress.cache.ttl=60 -Des.networkaddress.cache.negative.ttl=10 -XX:+AlwaysPreTouch -Xss1m -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djna.nosys=true -XX:-OmitStackTraceInFastThrow -Dio.netty.noUnsafe=true -Dio.netty.noKeySetOptimization=true -Dio.netty.recycler.maxCapacityPerThread=0 -Dlog4j.shutdownHookEnabled=false -Dlog4j2.disable.jmx=true -Djava.io.tmpdir=C:\sonarqube-8.5.1.38104\temp -XX:ErrorFile=../logs/es_hs_err_pid%p.log -Xmx512m -Xms512m -XX:+HeapDumpOnOutOfMemoryError -Delasticsearch -Des.path.home=C:\sonarqube-8.5.1.38104\elasticsearch -Des.path.conf=C:\sonarqube-8.5.1.38104\temp\conf\es -cp lib/* org.elasticsearch.bootstrap.Elasticsearch

jvm 1 | Java HotSpot(TM) 64-Bit Server VM warning: Option UseConcMarkSweepGC was deprecated in version 9.0 and will likely be removed in a future release.

jvm 1 | 2020.10.30 21:08:33 INFO app[[o.s.a.SchedulerImpl] Waiting for Elasticsearch to be up and running
jvm 1 | 2020.10.30 21:08:35 INFO app[[o.e.p.PluginsService] no modules loaded
jvm 1 | 2020.10.30 21:08:35 INFO app[[o.e.p.PluginsService] loaded plugin [org.elasticsearch.transport.Netty4Plugin]

jvm 1 | 2020.10.30 21:09:31 INFO app[[o.s.a.SchedulerImpl] Process[es] is up
jvm 1 | 2020.10.30 21:09:31 INFO app[[o.s.a.ProcessLauncherImpl] Launch process[[key='web', ipcIndex=2, logFilenamePrefix=web]] from [C:\sonarqube-8.5.1.38104]: C:\Program Files\Java\jdk-11.0.9\bin\java -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=C:\sonarqube-8.5.1.38104\temp -XX:-OmitStackTraceInFastThrow --add-opens=java.base/java.util=ALL-UNNAMED --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 -Xmx512m -Xms128m -XX:+HeapDumpOnOutOfMemoryError -Dhttp.nonProxyHosts=localhost|127.*|:::1 -cp ./lib/common/*;C:\sonarqube-8.5.1.38104\lib\jdbc\h2\h2-1.4.199.jar org.sonar.server.app.WebServer C:\sonarqube-8.5.1.38104\temp\squid-process450459688020831379properties

jvm 1 | 2020.10.30 21:10:46 INFO app[[o.s.a.SchedulerImpl] Process[web] is up
jvm 1 | 2020.10.30 21:10:46 INFO app[[o.s.a.ProcessLauncherImpl] Launch process[[key='ce', ipcIndex=3, logFilenamePrefix=ce]] from [C:\sonarqube-8.5.1.38104]: C:\Program Files\Java\jdk-11.0.9\bin\java -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djava.io.tmpdir=C:\sonarqube-8.5.1.38104\temp -XX:-OmitStackTraceInFastThrow --add-opens=java.base/java.util=ALL-UNNAMED -Xmx512m -Xms128m -XX:+HeapDumpOnOutOfMemoryError -Dhttp.nonProxyHosts=localhost|127.*|:::1 -cp ./lib/common/*;C:\sonarqube-8.5.1.38104\lib\jdbc\h2\h2-1.4.199.jar org.sonar.ce.app.CeServer C:\sonarqube-8.5.1.38104\temp\squid-process17983098687711107575properties

jvm 1 | 2020.10.30 21:11:47 INFO app[[o.s.a.SchedulerImpl] Process[ce] is up
jvm 1 | 2020.10.30 21:11:47 INFO app[[o.s.a.SchedulerImpl] SonarQube is up
```

-Ahora analizaremos el código. Ingresamos a la dirección de nuestra solución y ejecutamos el código con el nombre del proyecto y el nombre de la solución.

```
C:\Users\Derian\Desktop\WTF\GIMNASIO_PROJECT\Proyecto_Gimnasio>SonarScanner.MSBuild.exe begin /k:"Proyecyo_Gimnasio" /n:"Proyecto_Gimnasio" /v:"2.0"
SonarScanner for MSBuild 4.10
Using the .NET Framework version of the Scanner for MSBuild
Pre-processing started.
Preparing working directories...
21:25:13.553 Updating build integration targets...
21:25:13.668 Fetching analysis configuration settings...
21:25:29.207 Provisioning analyzer assemblies for cs...
21:25:29.216 Installing required Roslyn analyzers...
21:25:30.426 Provisioning analyzer assemblies for vbnet...
21:25:30.436 Installing required Roslyn analyzers...
21:25:30.685 Pre-processing succeeded.
```

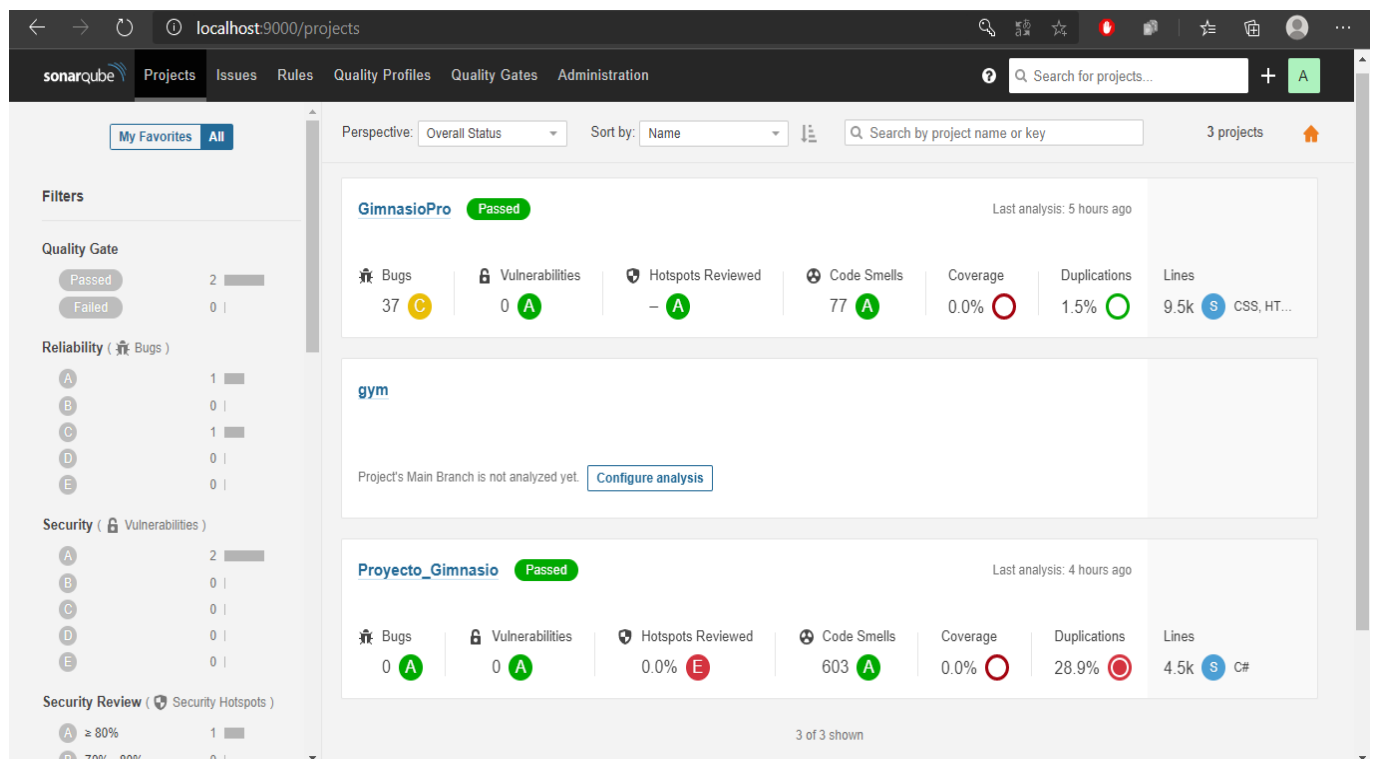
-Después ejecutamos el siguiente código para realizar el análisis

```
C:\Users\Derian\Desktop\WTF\GIMNASIO_PROJECT\Proyecto_Gimnasio>"C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\MSBuild\15.0\Bin\MSBuilder.exe" /t:Rebu...
```

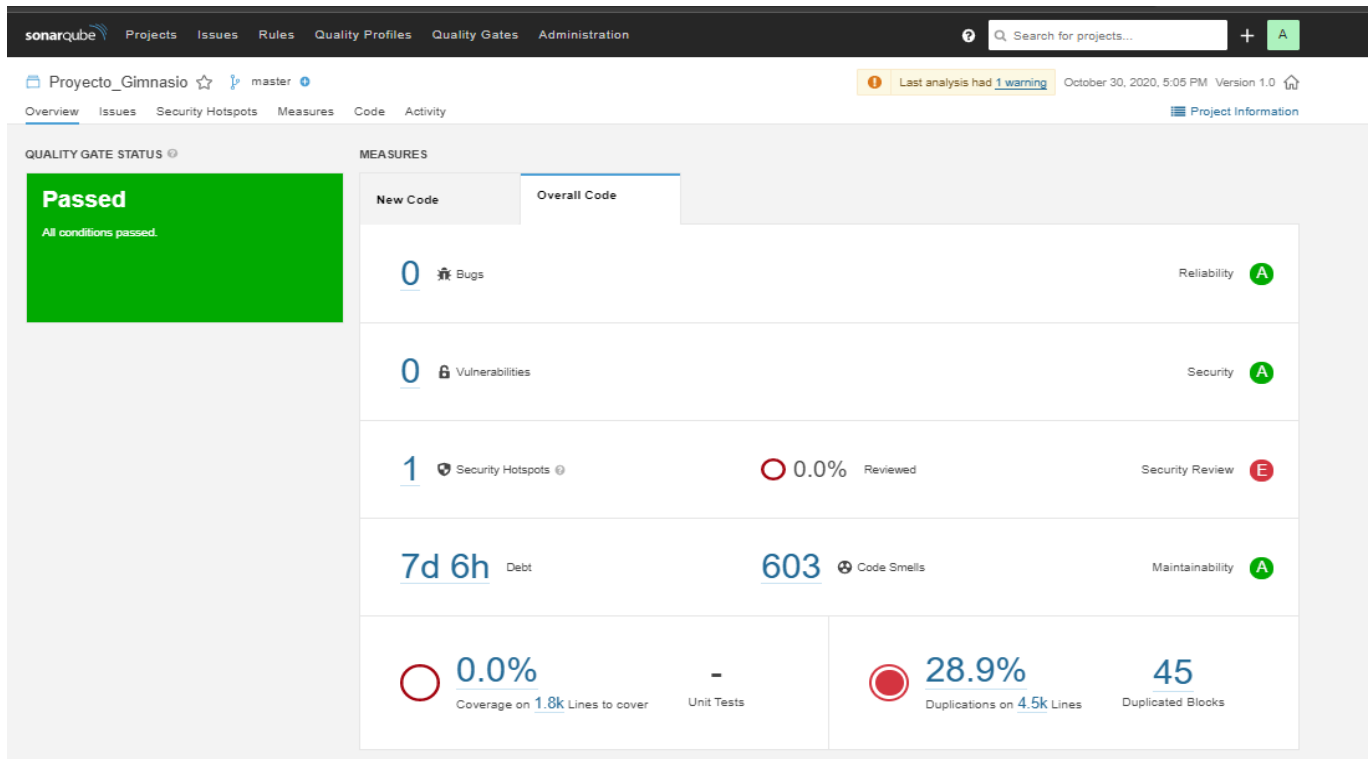
-Y por último este código para finalizar el análisis.

```
C:\Users\Derian\Desktop\WTF\GIMNASIO_PROJECT\Proyecto_Gimnasio>SonarScanner.MSBuild.exe end
```

-Luego entramos a nuestro localhost de SonarQube. Aquí tenemos el análisis de nuestro proyecto anterior y el que mejoramos.

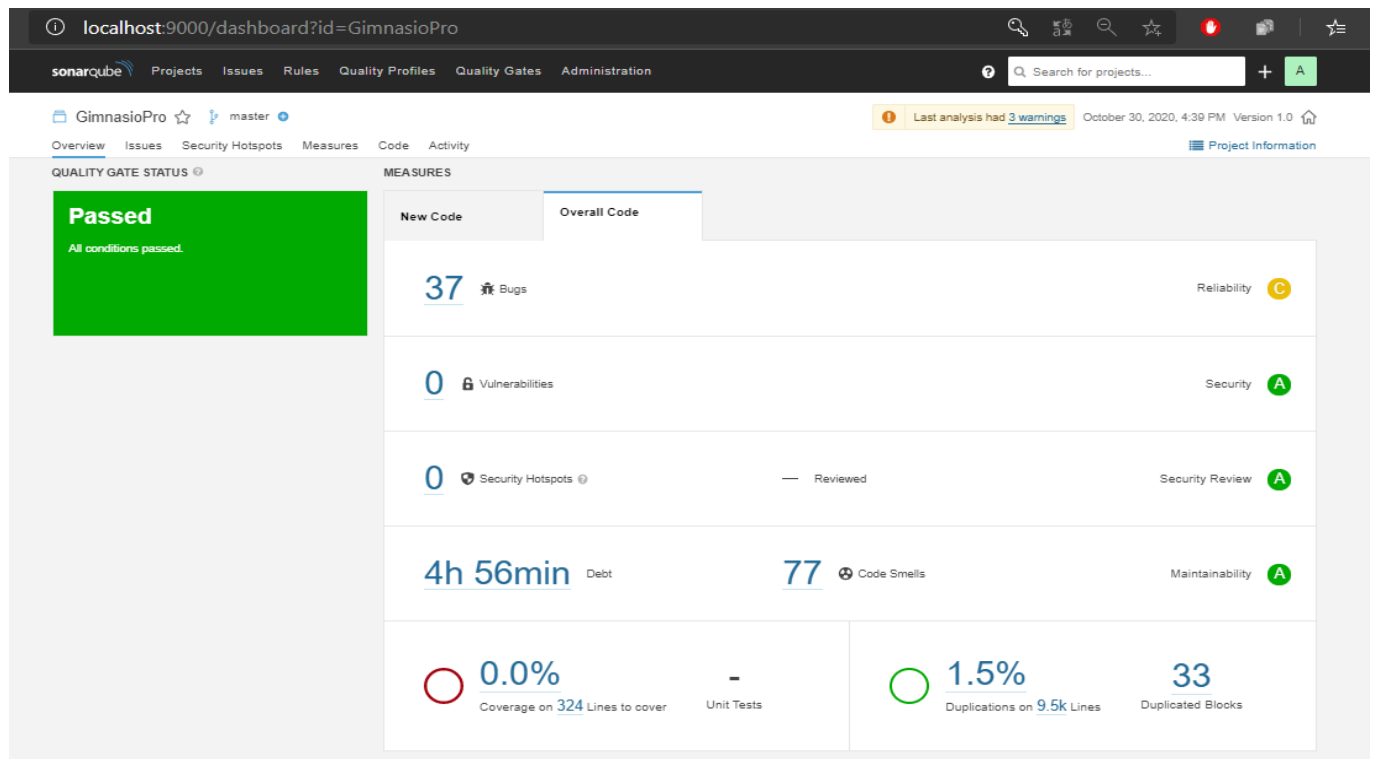


-Muestra del analisis de nuestro proyecto anterior:

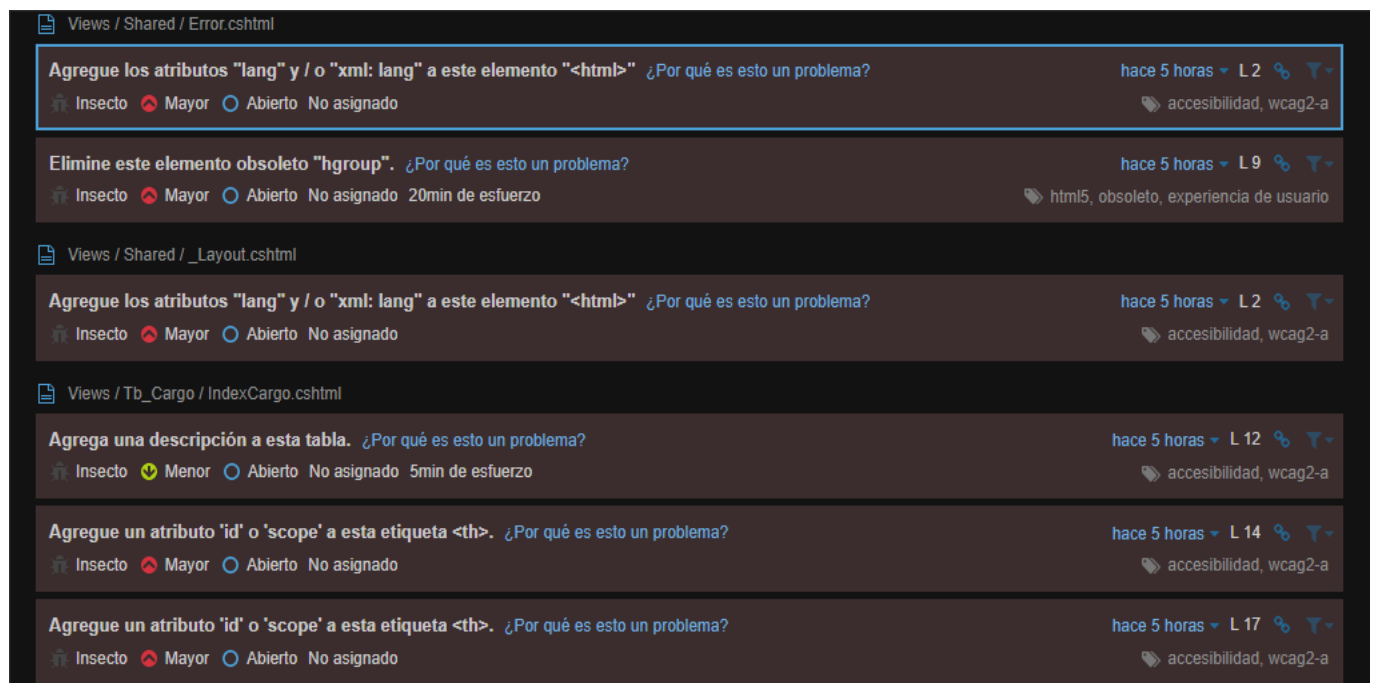


Como vemos tenemos 0 bugs, 0 vulnerabilidades, 1 punto de acceso de seguridad vulnerabilidad, Una deuda tecnica de 7 dias 6 horas , 603 de codigo que podria ser un problema, casi un 30 por-ciento de duplicidad de codigo en 45 bloques.

-Ahora veremos el código que realizamos con algunas mejoras como el uso de EntityFramework




En este caso nuestra solución tiene 37 bugs. Mas que todos en las vistas del proyecto. en donde nos pide poner un id y scope a nuestras tablas



En la parte de seguridad se eliminaron las vulnerabilidades.

0  Vulnerabilities

Security 

0  Security Hotspots 

— Reviewed

Security Review 

En lo que se refiere a deuda total tenemos 4h 56 min de deuda. Relacionados a 41 componentes del proyecto.

Technical Debt 4h 56min	
Controllers/Tb_ProveedorController.cs	45min
Controllers/Tb_ClienteController.cs	30min
Controllers/Tb_DetalleVentaController.cs	30min
Controllers/Tb_IngresoController.cs	30min
Controllers/Tb_ProductoController.cs	30min
Controllers/Tb_VentaController.cs	30min
App_Start/RouteConfig.cs	18min
Controllers/Tb_CargoController.cs	15min
Controllers/Tb_EmpleadoController.cs	15min
Controllers/Tb_TipoProductoController.cs	15min
App_Start/BundleConfig.cs	12min
App_Start/FilterConfig.cs	12min
Controllers/HomeController.cs	8min
Global.asax.cs	4min
Properties/AssemblyInfo.cs	2min
There are 41 hidden components with a score of 0. Show Them	

Se encontro tenemos 324 líneas por cubrir con pruebas unitarias estas relacionados con 14 componentes de nuestro proyecto.

Project Overview Reliability ⓘ Security ⓘ Security Review ⓘ Maintainability ⓘ Coverage LINES TO COVER Overview ⓘ Overall Coverage 0.0% Lines to Cover 324 Uncovered Lines 324 Line Coverage 0.0% Duplications Size Complexity ⓘ Issues	GimnasioPro View as List to select files to navigate 14 files	
	Lines to Cover 324	
	Controllers/Tb_DetalleVentaController.cs	46
	Controllers/Tb_ProductoController.cs	46
	Controllers/Tb_VentaController.cs	46
	Controllers/Tb_IngresoController.cs	42
	Controllers/Tb_ProveedorController.cs	37
	Controllers/Tb_EmpleadoController.cs	29
	Controllers/Tb_ClienteController.cs	24
	Controllers/Tb_CargoController.cs	18
	Controllers/Tb_TipoProductoController.cs	18
	App_Start/BundleConfig.cs	5
	Controllers/HomeController.cs	5
	Global.asax.cs	4
	App_Start/RouteConfig.cs	3
	App_Start/FilterConfig.cs	1
	14 of 14 shown	

Tenemos codigo peligro que puede dañar a un futuro el proyecto que tiene que ver en el uso innecesarios de codigo y agregar contrucctores de proteccion.

The screenshot displays the SonarQube interface for the 'GimnasioPro' project. The left sidebar shows filters for 'Type: CODE SMELL' with 77 items. The main panel lists several issues, including 'Remove this unnecessary "using"' and 'Rename class "Tb_CargoController" to match pascal case naming rules'. The issues are categorized by severity (Minor, Major) and status (Open, Not assigned).

Encontramos duplicidad de codigo producto de malas practicas al programar, pero esta en el rango de aceptacion



Medidas



8. Cronograma