Static Code Analysis of .NET Core Projects with SonarCloud

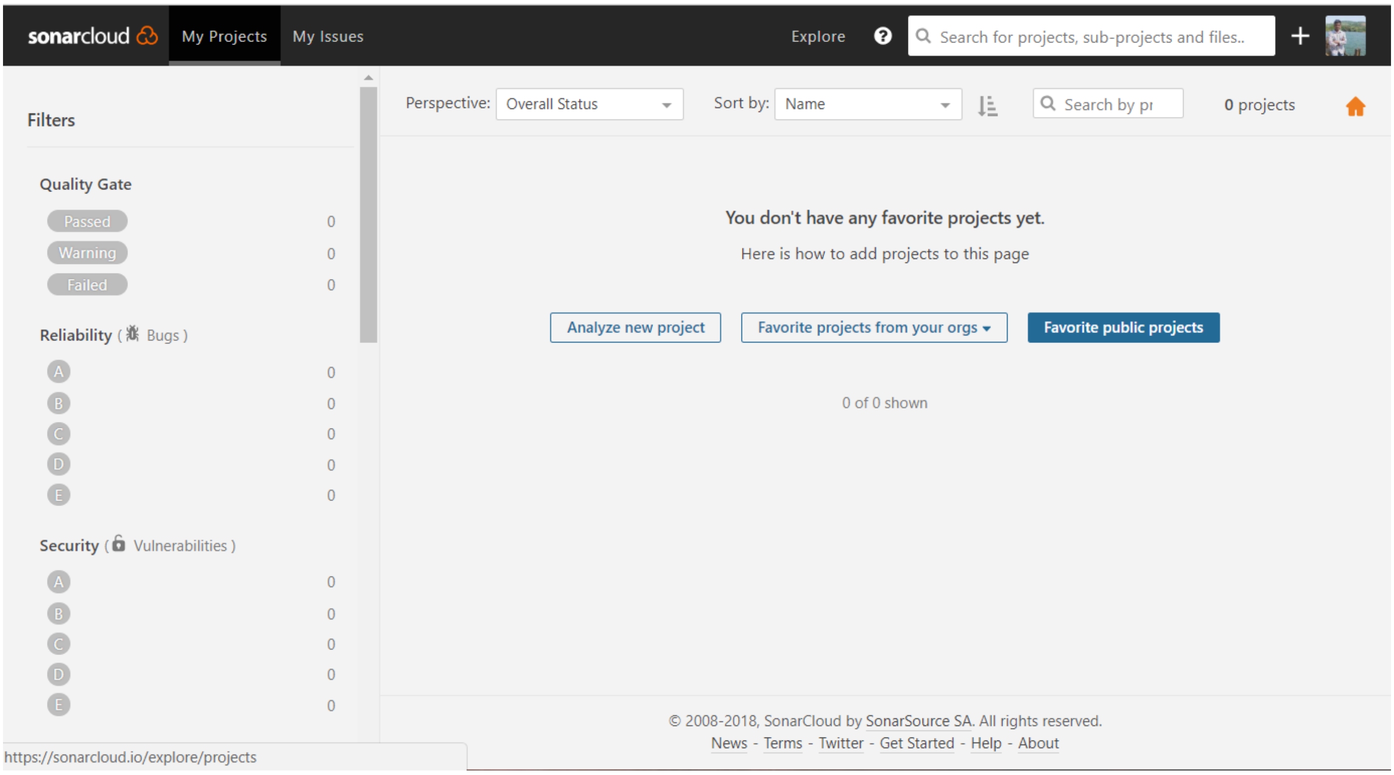
TWILIO SendGrid Email delivery service that gets you to the inbox in seconds.

Posted by Anuraj (https://plus.google.com/+AnurajP) on Thursday, May 3, 2018 Reading time :2 minutes

ASP.NET Core (/tags#ASP.NET Core) Static Code Analysis (/tags#Static Code Analysis) SonarCloud (/tags#SonarCloud)

This post is about how to use SonarCloud application for running static code analysis in .NET Core projects. Static analysis is a way of automatically analysing code without executing it. SonarCloud is cloud offering of SonarQube app. It is Free for Open source projects.

For analysing first you need to create an account in sonarcloud.io. You can use GitHub / BitBucket / Microsoft Live Account to sign in. Once you sign in, you will see the sonarcloud dashboard.


sonarcloud
My Projects
My Issues
Explore
Search for projects, sub-projects and files..
Filters
Quality Gate
Passed 0
Warning 0
Failed 0
Reliability (Bugs)
A 0
B 0
C 0
D 0
E 0
Security (Vulnerabilities)
A 0
B 0
C 0
D 0
E 0
Perspective: Overall Status
Sort by: Name
Search by pr
0 projects
Analyze new project
Favorite projects from your orgs
Favorite public projects
© 2008-2018, SonarCloud by SonarSource SA. All rights reserved.
News - Terms - Twitter - Get Started - Help - About
https://sonarcloud.io/explore/projects

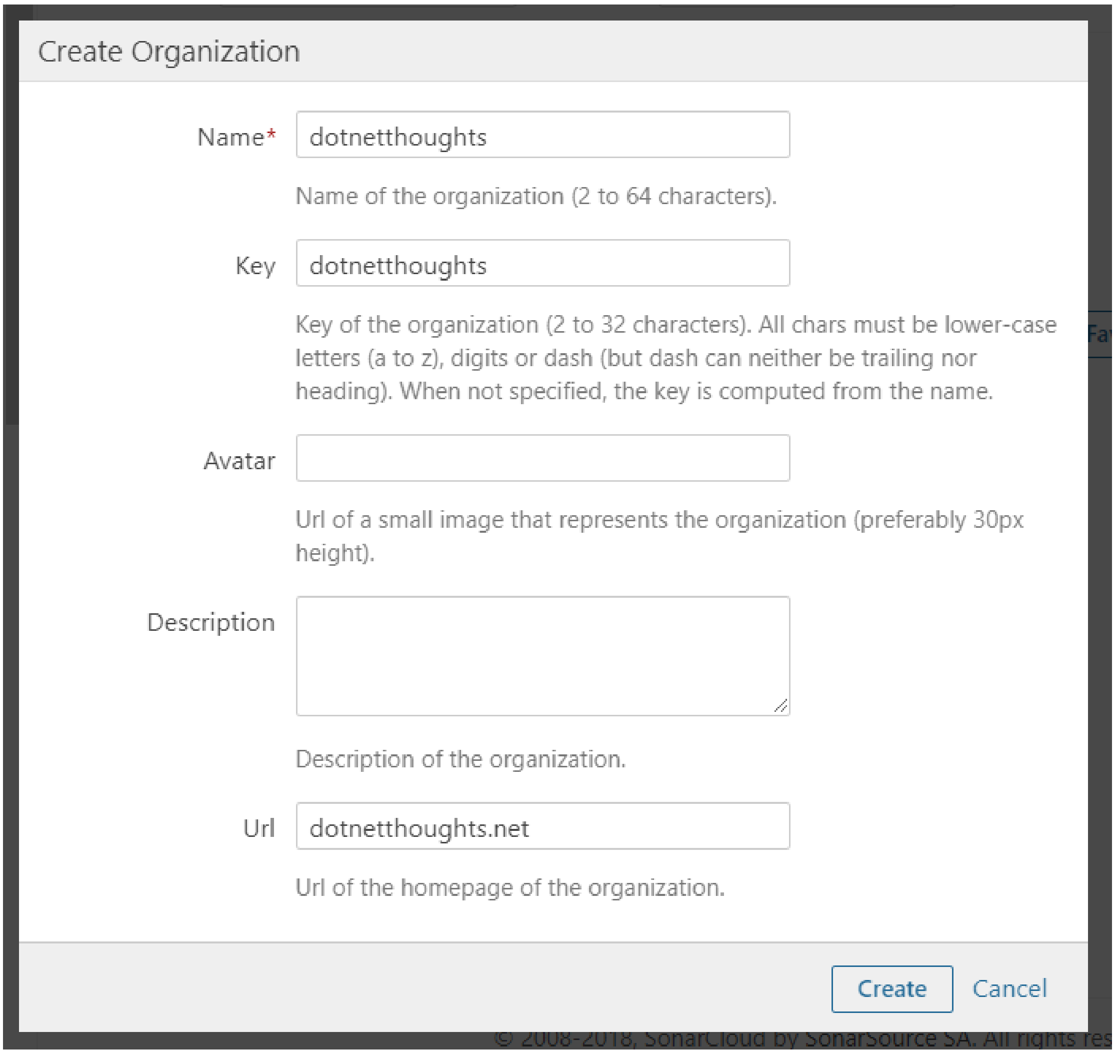
You don't have any favorite projects yet.

Here is how to add projects to this page

0 of 0 shown

Next you need to create an Organization.

↑



Create Organization

Name* dotnetthoughts

Name of the organization (2 to 64 characters).

Key dotnetthoughts

Key of the organization (2 to 32 characters). All chars must be lower-case letters (a to z), digits or dash (but dash can neither be trailing nor heading). When not specified, the key is computed from the name.

Avatar

Url of a small image that represents the organization (preferably 30px height).

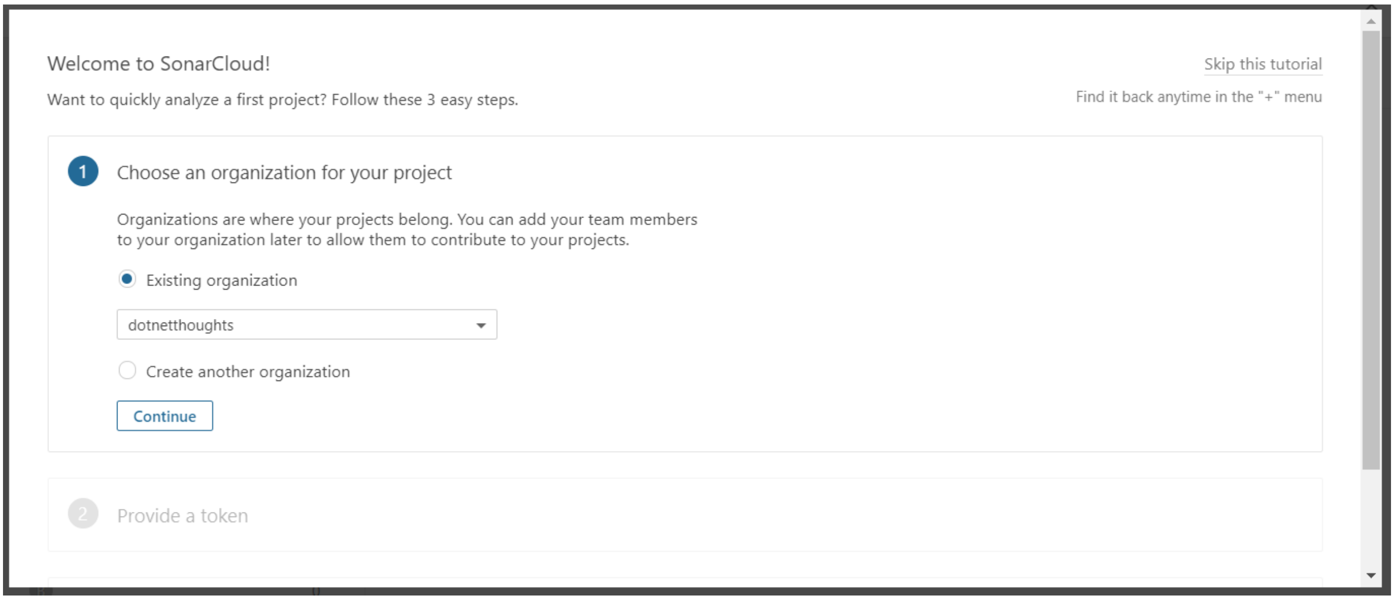
Description

Description of the organization.

Url dotnetthoughts.net

Url of the homepage of the organization.

Once you created the organization, you can create a new project to analyse. You can click on the Analyse New Project button, which will show a wizard. In the first screen you need to select the organization.



Welcome to SonarCloud!

Want to quickly analyze a first project? Follow these 3 easy steps.

1 Choose an organization for your project

Organizations are where your projects belong. You can add your team members to your organization later to allow them to contribute to your projects.

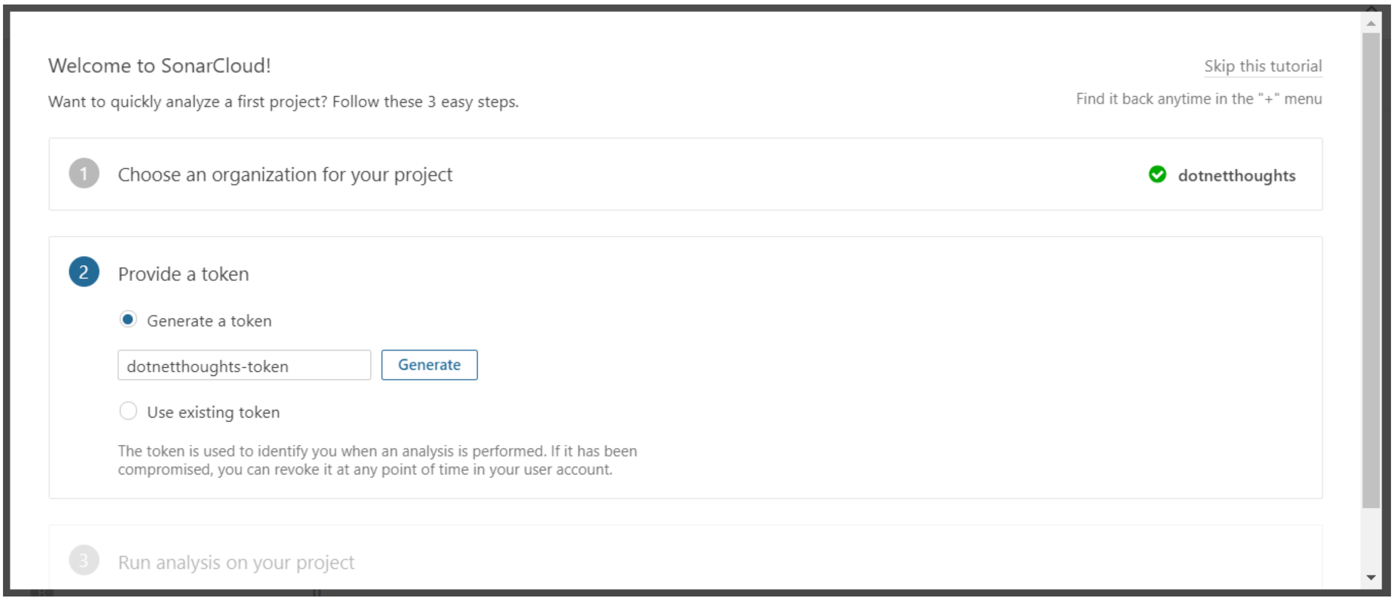
● Existing organization

dotnetthoughts

○ Create another organization

2 Provide a token

Next, you need to generate token, which will be used to run the code analysis, make sure you store it safe.



Welcome to SonarCloud!

Want to quickly analyze a first project? Follow these 3 easy steps.

Skip this tutorial

Find it back anytime in the "+" menu

1 Choose an organization for your project ✓ dotnetthoughts

2 Provide a token

● Generate a token

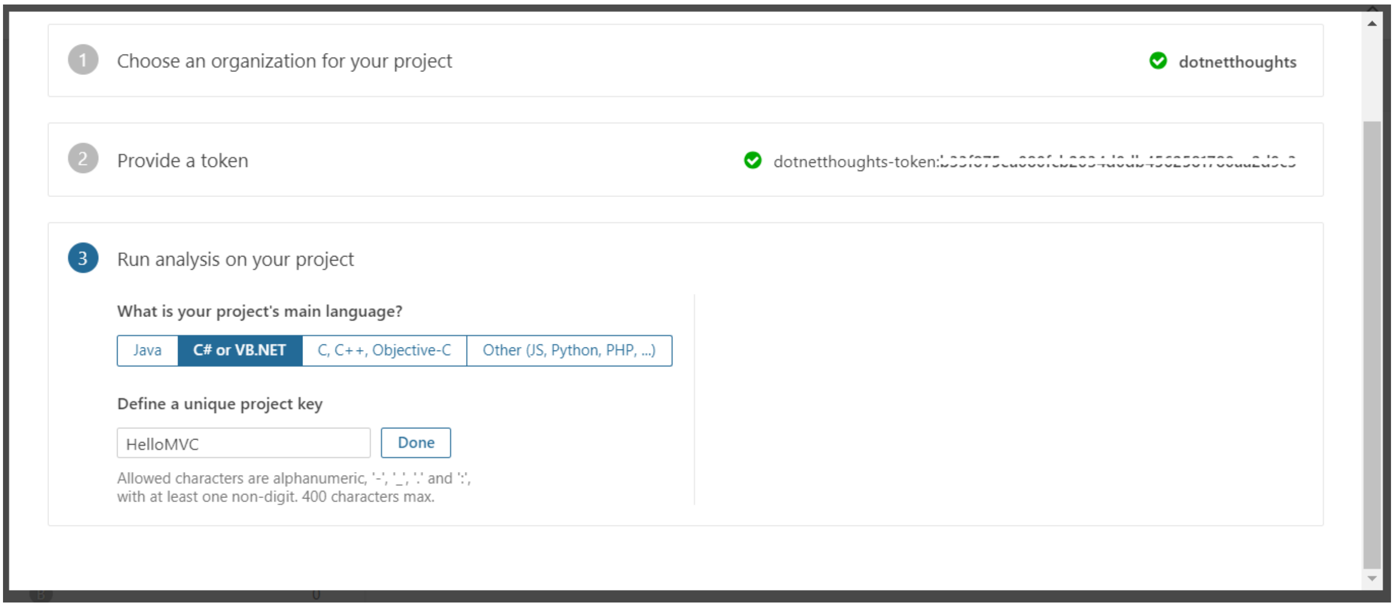
dotnetthoughts-token

○ 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.

3 Run analysis on your project

And in the third screen, you need select the target language and provide a project key.



1 Choose an organization for your project ✓ dotnetthoughts

2 Provide a token ✓ dotnetthoughts-token:b33f875ea080fcb2034d0db4562581780aa2d9c3

3 Run analysis on your project

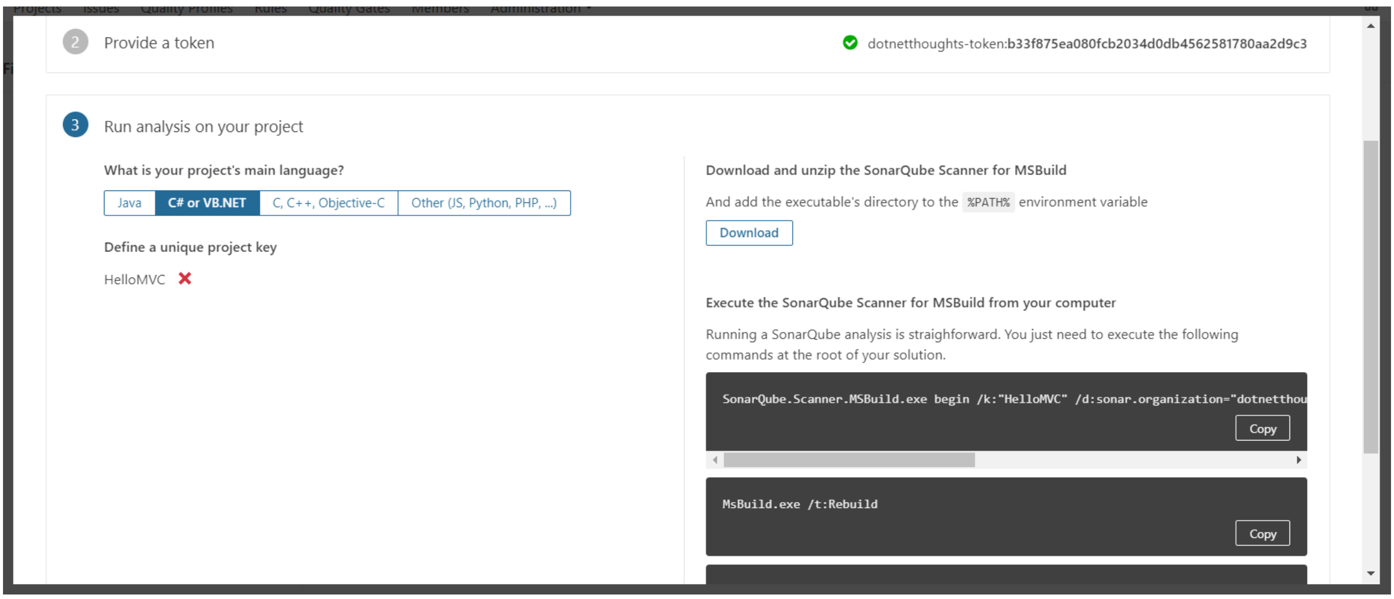
What is your project's main language?

Define a unique project key

HelloMVC

Allowed characters are alphanumeric, '-', '_', '.' and ':', with at least one non-digit. 400 characters max.

Clicking on Done, SonarCloud will show the steps to run code analysis using MSBuild.



Projects Issues Quality Profiles Rules Quality Gates Members Administration

2 Provide a token ✓ dotnetthoughts-token:b33f875ea080fcb2034d0db4562581780aa2d9c3

3 Run analysis on your project

What is your project's main language?

Java
C# or VB.NET
C, C++, Objective-C
Other (JS, Python, PHP, ...)

Define a unique project key

HelloMVC ✕

Download and unzip the SonarQube Scanner for MSBuild

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

Execute the SonarQube Scanner for MSBuild from your computer

Running a SonarQube analysis is straighforward. You just need to execute the following commands at the root of your solution.

```
SonarQube.Scanner.MSBuild.exe begin /k:"HelloMVC" /d:sonar.organization="dotnetthou
```

```
MsBuild.exe /t:Rebuild
```

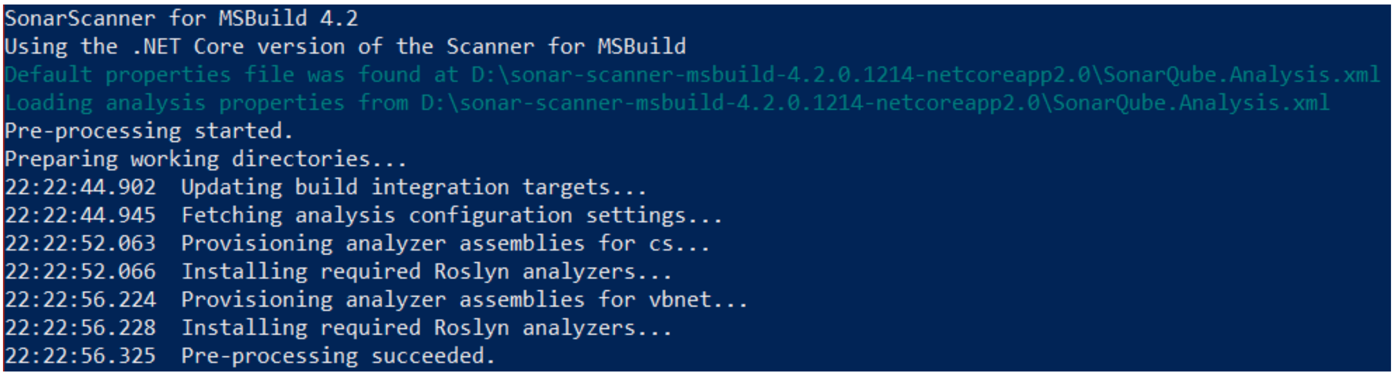
Once you completed it, next you need to download the sonar scanner for MS Build, you can download the .NET Core version from here (https://docs.sonarqube.org/display/SCAN/Analyzing+with+SonarQube+Scanner+for+MSBuild)

↑

Next you need to create the .NET Core application to run the analysis. So first you need to create a ASP.NET Core MVC project using `dotnet new mvc`. Then you need to create a sln file using `dotnet new sln` command. Next you need to add the MVC project to the solution. You can do it using `dotnet sln add HelloMVC.csproj`. Now you're ready with your project. Next you need to enable scanner. To do that first, you need to run the following command.

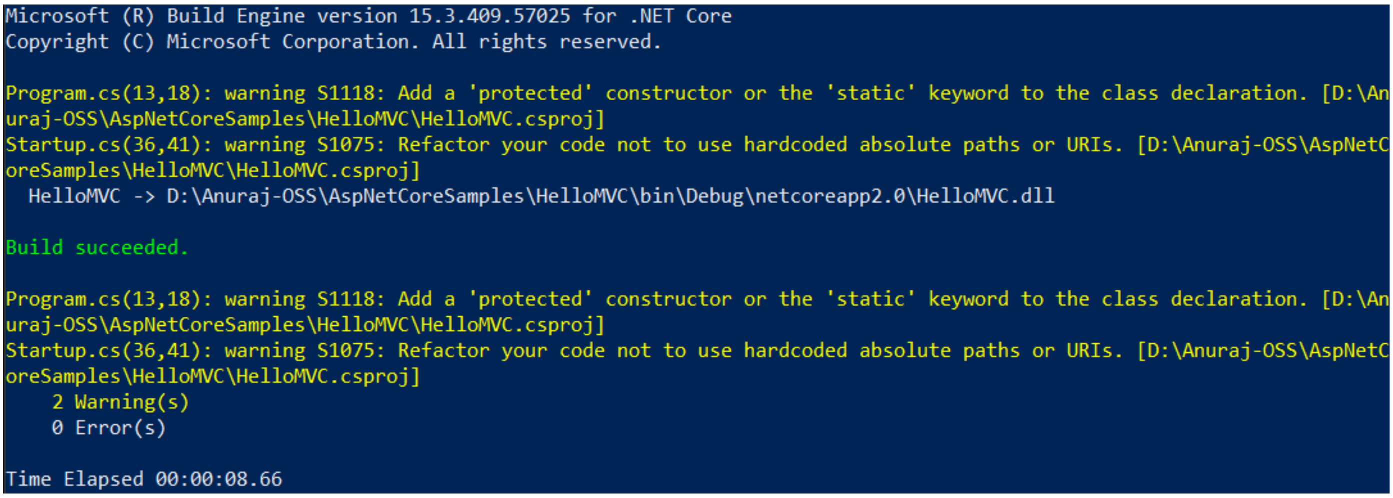
```
dotnet "D:\sonar-scanner-msbuild-4.2.0.1214-netcoreapp2.0\SonarScanner.MSBuild.dll" begin /k:"HelloMVC" /d:sonar.organiza
```

Once you execute the command you will be able to see something like this on the screen.



```
SonarScanner for MSBuild 4.2
Using the .NET Core version of the Scanner for MSBuild
Default properties file was found at D:\sonar-scanner-msbuild-4.2.0.1214-netcoreapp2.0\SonarQube.Analysis.xml
Loading analysis properties from D:\sonar-scanner-msbuild-4.2.0.1214-netcoreapp2.0\SonarQube.Analysis.xml
Pre-processing started.
Preparing working directories...
22:22:44.902  Updating build integration targets...
22:22:44.945  Fetching analysis configuration settings...
22:22:52.063  Provisioning analyzer assemblies for cs...
22:22:52.066  Installing required Roslyn analyzers...
22:22:56.224  Provisioning analyzer assemblies for vbnet...
22:22:56.228  Installing required Roslyn analyzers...
22:22:56.325  Pre-processing succeeded.
```

Next you need to build the application using `dotnet build` command. You will be able to see some warnings in the screen.

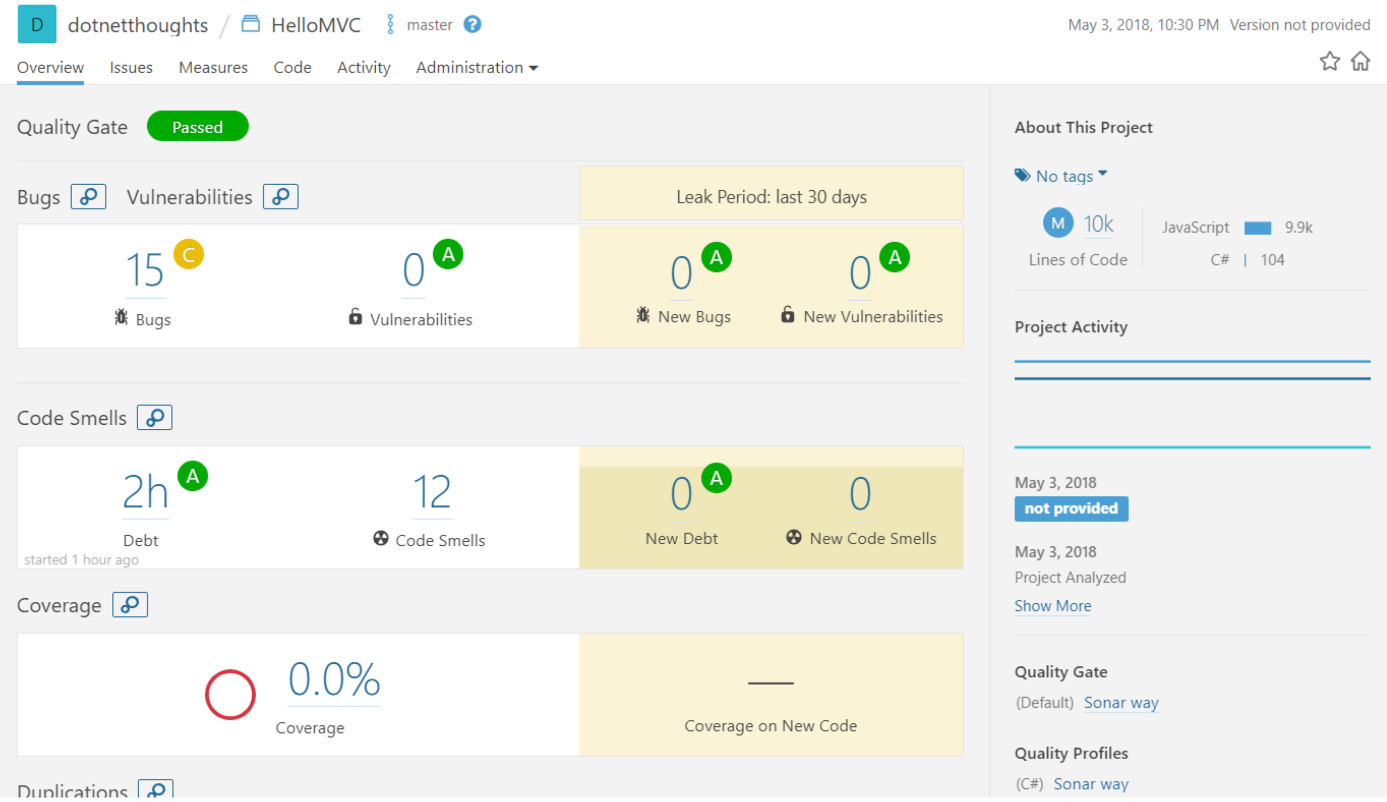


```
Microsoft (R) Build Engine version 15.3.409.57025 for .NET Core
Copyright (C) Microsoft Corporation. All rights reserved.

Program.cs(13,18): warning S1118: Add a 'protected' constructor or the 'static' keyword to the class declaration. [D:\An
uraj-OSS\AspNetCoreSamples\HelloMVC\HelloMVC.csproj]
Startup.cs(36,41): warning S1075: Refactor your code not to use hardcoded absolute paths or URIs. [D:\Anuraj-OSS\AspNetC
oreSamples\HelloMVC\HelloMVC.csproj]
  HelloMVC -> D:\Anuraj-OSS\AspNetCoreSamples\HelloMVC\bin\Debug\netcoreapp2.0\HelloMVC.dll

Build succeeded.

Program.cs(13,18): warning S1118: Add a 'protected' constructor or the 'static' keyword to the class declaration. [D:\An
uraj-OSS\AspNetCoreSamples\HelloMVC\HelloMVC.csproj]
Startup.cs(36,41): warning S1075: Refactor your code not to use hardcoded absolute paths or URIs. [D:\Anuraj-OSS\AspNetC
oreSamples\HelloMVC\HelloMVC.csproj]
    2 Warning(s)
    0 Error(s)

Time Elapsed 00:00:08.66
```

To finish you need to run the `end` command similar to `begin` command.

```
dotnet "D:\sonar-scanner-msbuild-4.2.0.1214-netcoreapp2.0\SonarScanner.MSBuild.dll" end /d:sonar.login="73fd8bc705804e868
```

This will start the analysis and upload the results to Sonarcloud. Next open your SonarCloud dashboard, click on the project, then you will be able to see the results like this.


D dotnetthoughts / HelloMVC master
May 3, 2018, 10:30 PM Version not provided
Overview Issues Measures Code Activity Administration
About This Project
No tags
Project Activity

Quality Gate Passed

Bugs Vulnerabilities — Leak Period: last 30 days

15 C Bugs — 0 A Vulnerabilities — 0 A New Bugs — 0 A New Vulnerabilities

Code Smells

2h A Debt (started 1 hour ago) — 12 Code Smells — 0 A New Debt — 0 New Code Smells

Coverage

0.0% Coverage — — Coverage on New Code

Duplications

M 10k Lines of Code — JavaScript 9.9k — C# 104

May 3, 2018 not provided

May 3, 2018 Project Analyzed

Show More

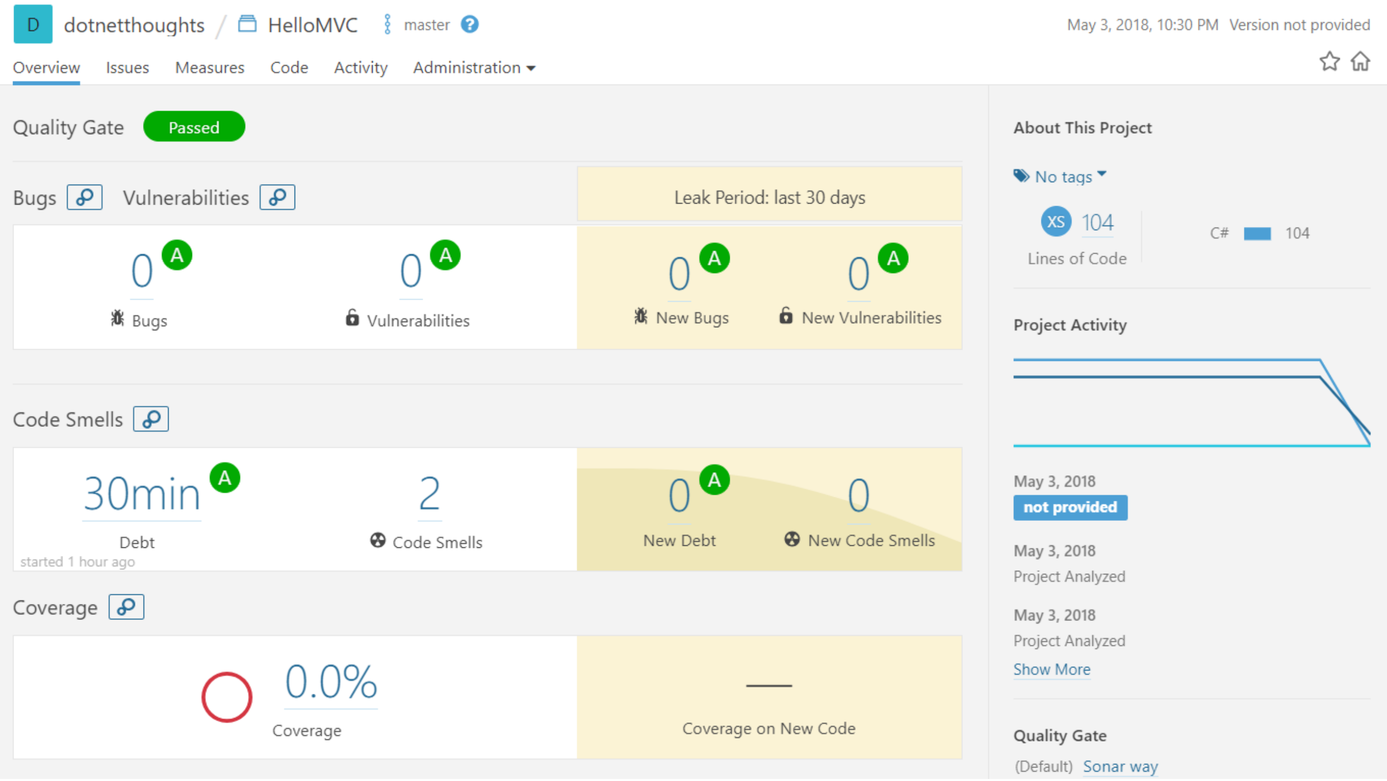
Quality Gate (Default) Sonar way

Quality Profiles (C#) Sonar way

If you notice, SonarCloud is showing 15 bugs, but if you look into the details you will be able to see most of these issues reported is from JavaScript libraries, like Bootstrap or JQuery. In ideal scenario, you don't need to run code analysis on this library files. So you need to exclude the libraries. You can do this using SonarQube exclusion filters like this.

```
dotnet "D:\sonar-scanner-msbuild-4.2.0.1214-netcoreapp2.0\SonarScanner.MSBuild.dll" begin /k:"HelloMVC" /d:sonar.organiza
```

Next build using dotnet build and end the scanning. It will update the dashboard like this.


D dotnetthoughts / HelloMVC master
May 3, 2018, 10:30 PM Version not provided
Overview Issues Measures Code Activity Administration
About This Project
No tags
Project Activity

Quality Gate Passed

Bugs Vulnerabilities — Leak Period: last 30 days

0 A Bugs — 0 A Vulnerabilities — 0 A New Bugs — 0 A New Vulnerabilities

Code Smells

30min A Debt (started 1 hour ago) — 2 Code Smells — 0 A New Debt — 0 New Code Smells

Coverage

0.0% Coverage — — Coverage on New Code

XS 104 Lines of Code — C# 104

May 3, 2018 not provided

May 3, 2018 Project Analyzed

May 3, 2018 Project Analyzed

Show More

Quality Gate (Default) Sonar way

Happy Programming :)

What do you think? I would like to hear your thoughts, suggestions, and questions in the comments section below.

Similar Posts

- Working with gRPC in ASP.NET Core (https://dotnetthoughts.net/working-with-grpc-in-aspnet-core/)
- Resizing Images using Azure Functions (https://dotnetthoughts.net/resizing-images-using-azure-functions/)
- Using EF Core in a Separate Class Library (https://dotnetthoughts.net/using-ef-core-in-a-separate-class-library/)
- Working with JSON in ASP.NET Core 3.0 (https://dotnetthoughts.net/working-with-json-in-aspnet-core/)
- Working with Save-Data header in ASP.NET Core (https://dotnetthoughts.net/working-with-save-data-header-in-aspnet-core/)
- LinkGenerator in ASP.NET Core (https://dotnetthoughts.net/linkgenerator-in-aspnet-core/)

Did you like this article? Share it with your friends

Facebook (https://www.facebook.com/sharer/sharer.php?u=https://dotnetthoughts.net/static-code-analysis-of-netcore-projects/)

Twitter (https://twitter.com/home?status=https://dotnetthoughts.net/static-code-analysis-of-netcore-projects/)

Google+ (https://plus.google.com/share?url=https://dotnetthoughts.net/static-code-analysis-of-netcore-projects/)

LinkedIn (https://www.linkedin.com/shareArticle?mini=true&url=https://dotnetthoughts.net/static-code-analysis-of-netcore-projects/)

Reddit (https://reddit.com/submit?url=https://dotnetthoughts.net/static-code-analysis-of-netcore-projects/)

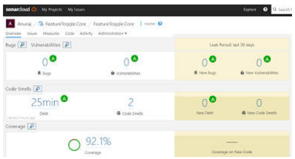
StumbleUpon (https://www.stumbleupon.com/submit?url=https://dotnetthoughts.net/static-code-analysis-of-netcore-projects/)

« Working with Microsoft Library Manager for ASP.NET Core (/working-with-microsoft-library-manager/)

Getting started with SignalR service in Azure » (/getting-started-with-signalr-in-azure/)

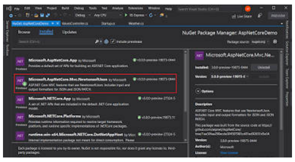
Cybercrime is Everywhere

Ad Webroot



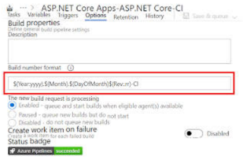
Code coverage in .NET Core with Coverlet

dotnetthoughts.net



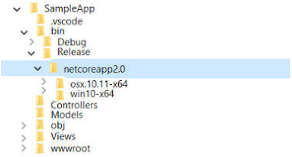
Working with JSON in ASP.NET Core 3.0

dotnetthoughts.net



How display application version in ASP.NET Core

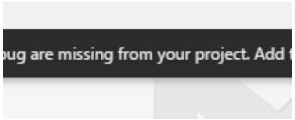
dotnetthoughts.net



```
SampleApp
  .vscode
  bin
    Debug
    Release
      netcoreapp2.0
        osx.10.11-x64
        win10-x64
  Controllers
  Models
  obj
  Views
  wwwroot
```

How to create a self contained .Net core application?

dotnetthoughts.net



ug are missing from your project. Add t

Integrating Visual Studio Code with dotnet watch to develop...

dotnetthoughts.net

BlazorExtensions.SignalR 0.1.6

BlazorExtensions.SignalR

SignalR Core support for ASP.NET Core Blazor.

Package Manager .NET CLI Paket CLI

```
> dotnet add package Blazor.Extensions.SignalR --version 0.1.6
```

Building Blazor apps with SignalR Core

dotnetthoughts.net

```
C:\Anuraj\ASPNET\MySqlDemo>dotnet aspnet-
ame HelloController -m Author -dc WebAPID
Finding the generator 'controller'...
Running the generator 'controller'...
Attempting to compile the application in
Attempting to figure out the EntityFrame
t: Author
Added Controller : \HelloController.cs
Added View : \Views\Hello\Create.cshtml
Added View : \Views\Hello\Edit.cshtml
Added View : \Views\Hello\Details.cshtml
Added View : \Views\Hello\Delete.cshtml
Added View : \Views\Hello\Index.cshtml
RunTime 00:00:11.33
```

Using scaffol create ASP.N applications

dotnetthoughts.net

↑

1 Comment **dotnetthoughts** 1 **Login**

♡ **Recommend** **Tweet** **Share** **Sort by Best**

Join the discussion…

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Stef Heyenrath • a year ago

Instead of manually downloading, you can also install it as .NET Core Global Tool, see :

https://www.nuget.org/packa...

^ | ˅ • Reply • Share ›

Subscribe **Add Disqus to your siteAdd DisqusAdd** **Disqus' Privacy PolicyPrivacy PolicyPrivacy**

MVP Microsoft® Most Valuable Professional

(https://mvp.microsoft.com/en-us/PublicProfile/5002040)

Translate

Select Language ▾

Powered by Google **Translate (https://translate.google.com)**

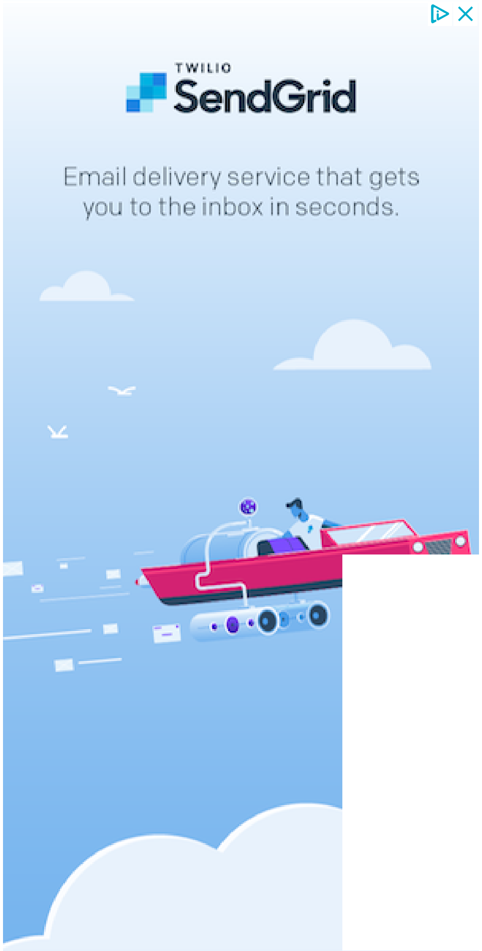
Subscribe

Email

Enter your email address

Subscribe

↑


TWILIO
SendGrid
Email delivery service that gets you to the inbox in seconds.

🍺

Buy me a coffee (https://www.buymeacoffee.com/dotnetthoughts)

(http://www.twitter.com/anuraj)(https://www.facebook.com/dotnetthoughts)(http://www.github.com/anuraj)

(https://plus.google.com/u/0/105671778313820571858)(http://feeds.feedburner.com/dotnetthoughts)

Copyright © 2019 - Anuraj P. Blog content licensed under the Creative Commons CC BY 2.5 (https://creativecommons.org/licenses/by/2.5/) | Unless otherwise stated or granted, code samples licensed under the MIT license (https://anuraj.mit-license.org/). This is a personal blog. The opinions expressed here represent my own and not those of my employer. Hosted with ♥ by GitHub

↑