Publish Code Coverage to SonarCloud Using Coverlet

.NET CORE

A PowerPoint created by John Kear.

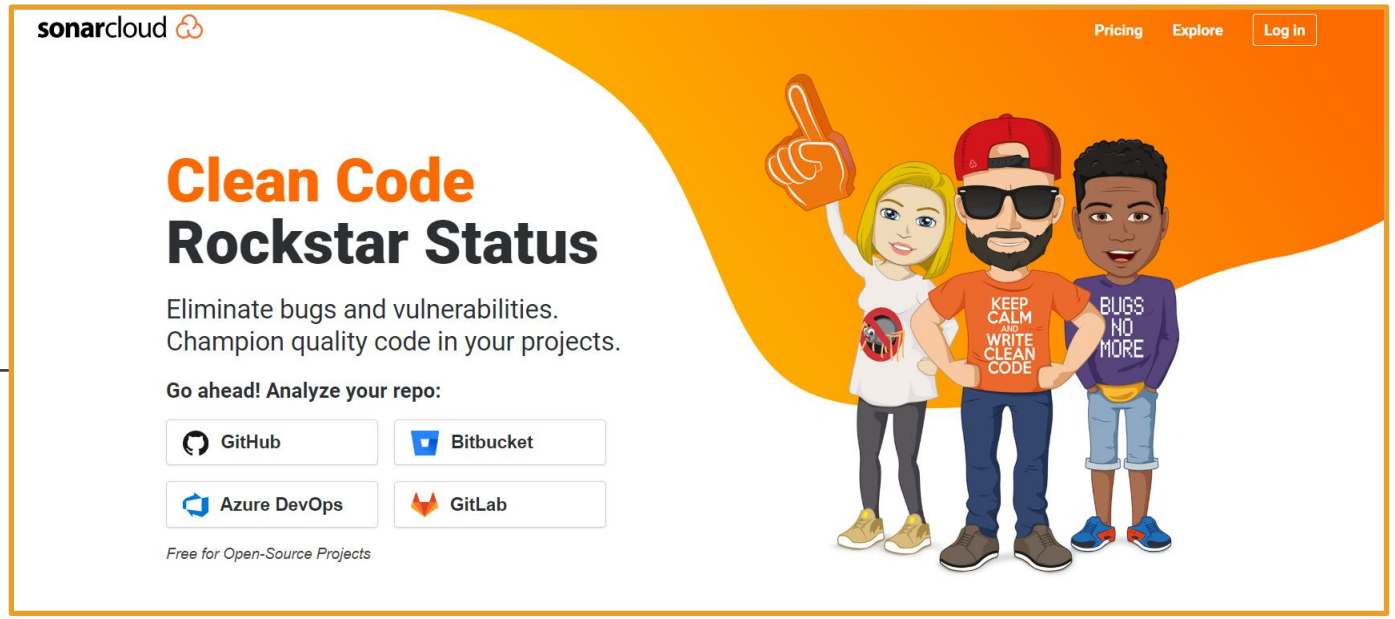
First Steps

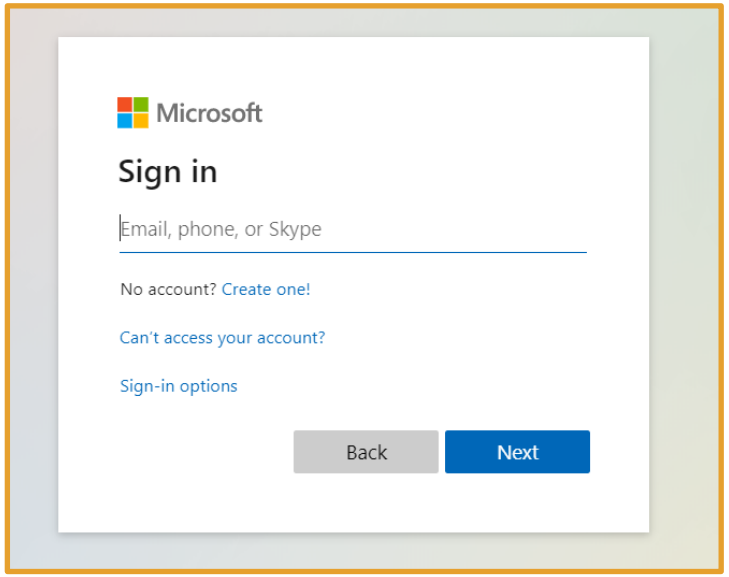
https://sonarcloud.io/

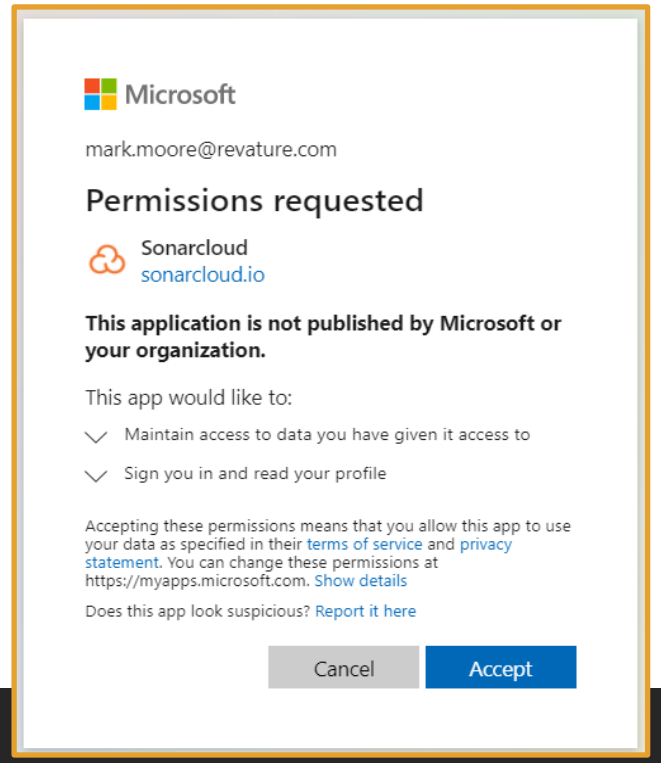
First, create a pipeline that successfully builds and deploys to your Website.

Next, log in to www.sonarcloud.io using your Azure DevOps credentials.

If asked, accept any request for permission or access by Sonarcloud.

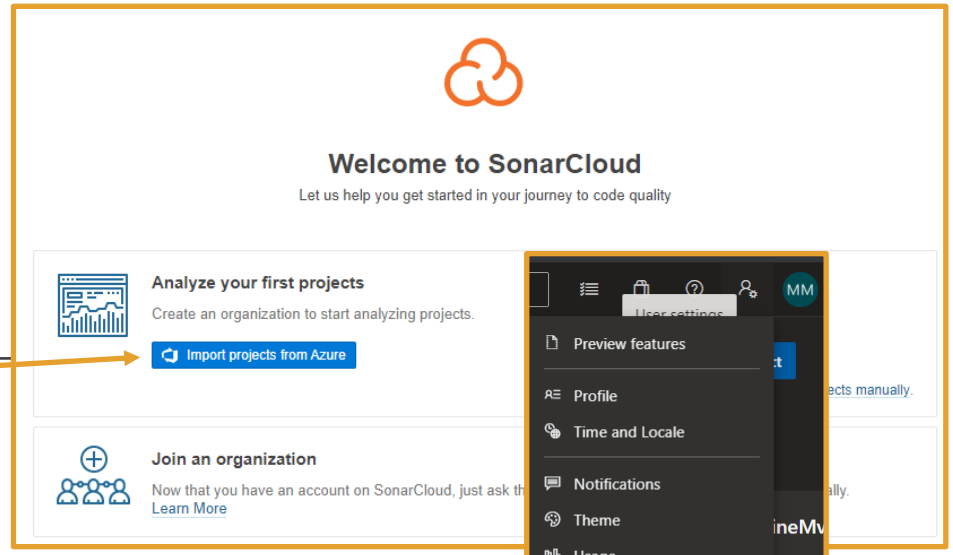


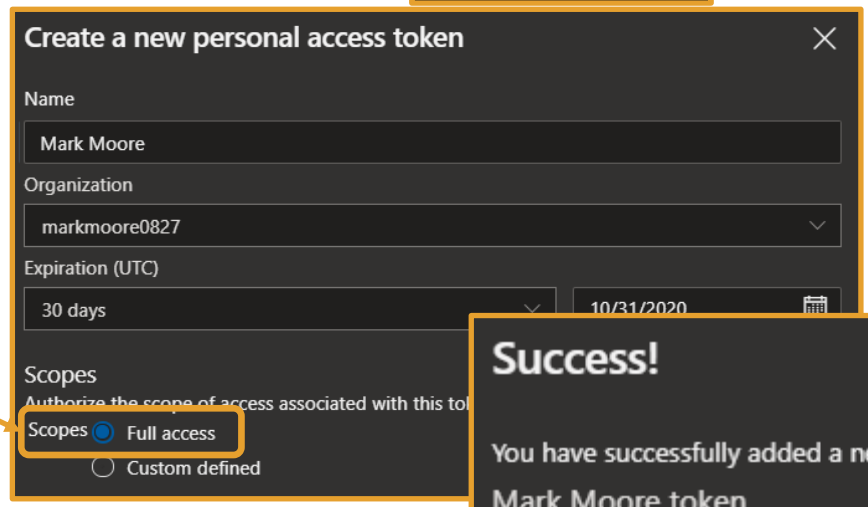


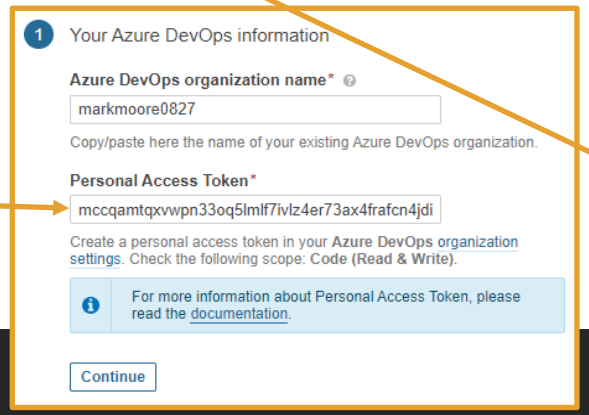


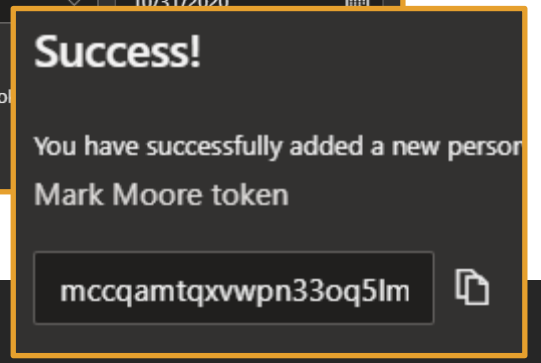
Create SonarCloud Organization

1. Create an organization
2. Enter the correct name of your Azure Devops Organization and a Personal Access Token (P.A.T.).
    1. Log into your Azure Account.
    2. Click on User Settings>Personal Access tokens.
    3. Click ‘New Token’.
    4. Enter your name and select ‘Full Access’
    5. Select ‘Create’
    6. Copy your P.A.T.
3. Enter the P.A.T.
4. Click ‘Continue’



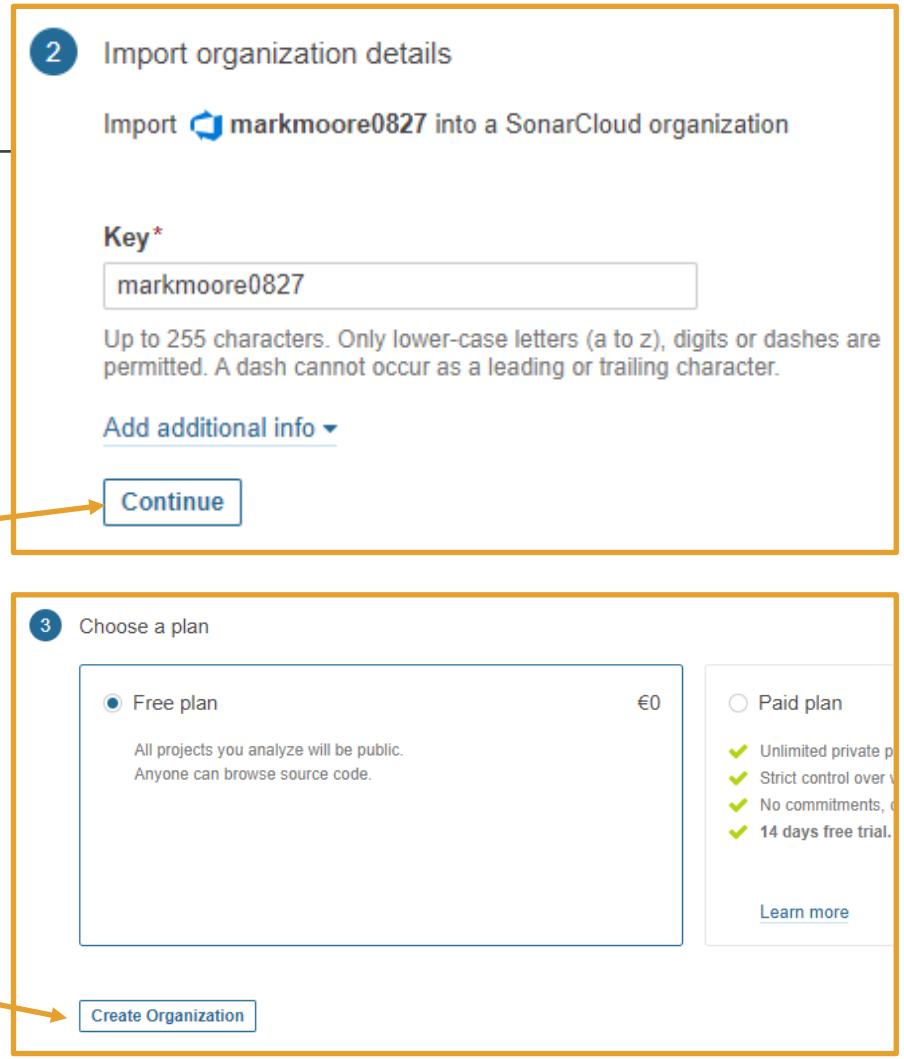






Create SonarCloud Organization

1. Click ‘Continue’ to Import your organization details
2. Select ‘Free plan’
3. Click “Create Organization”

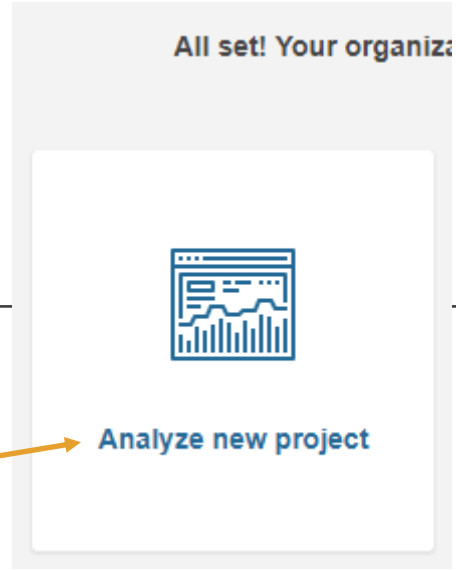


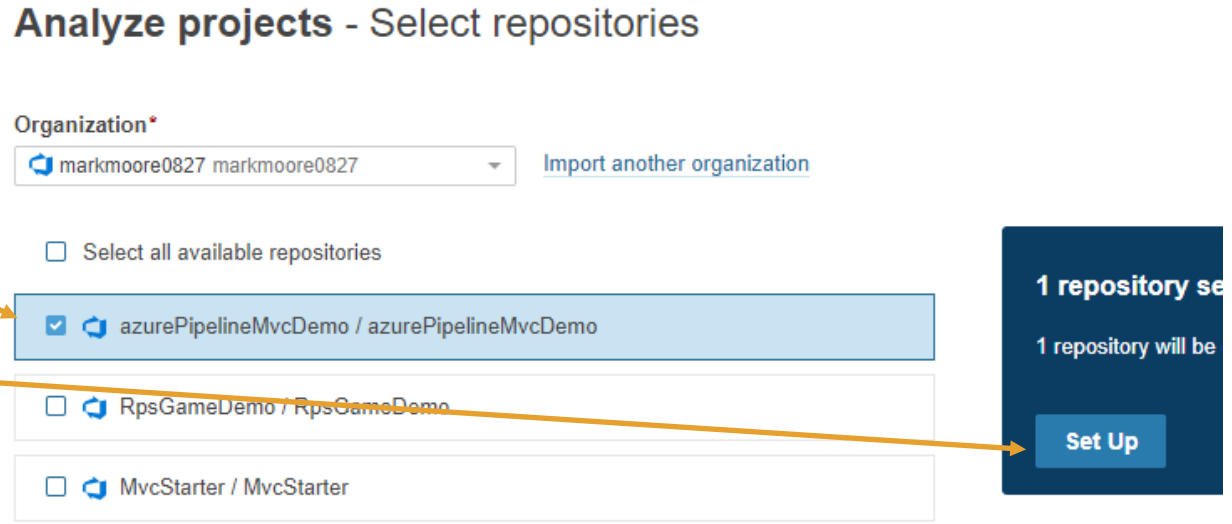
Analyse a New Project

Click ‘analyse new project’

Choose the pipeline you want to analyse.

Click ‘Set Up’



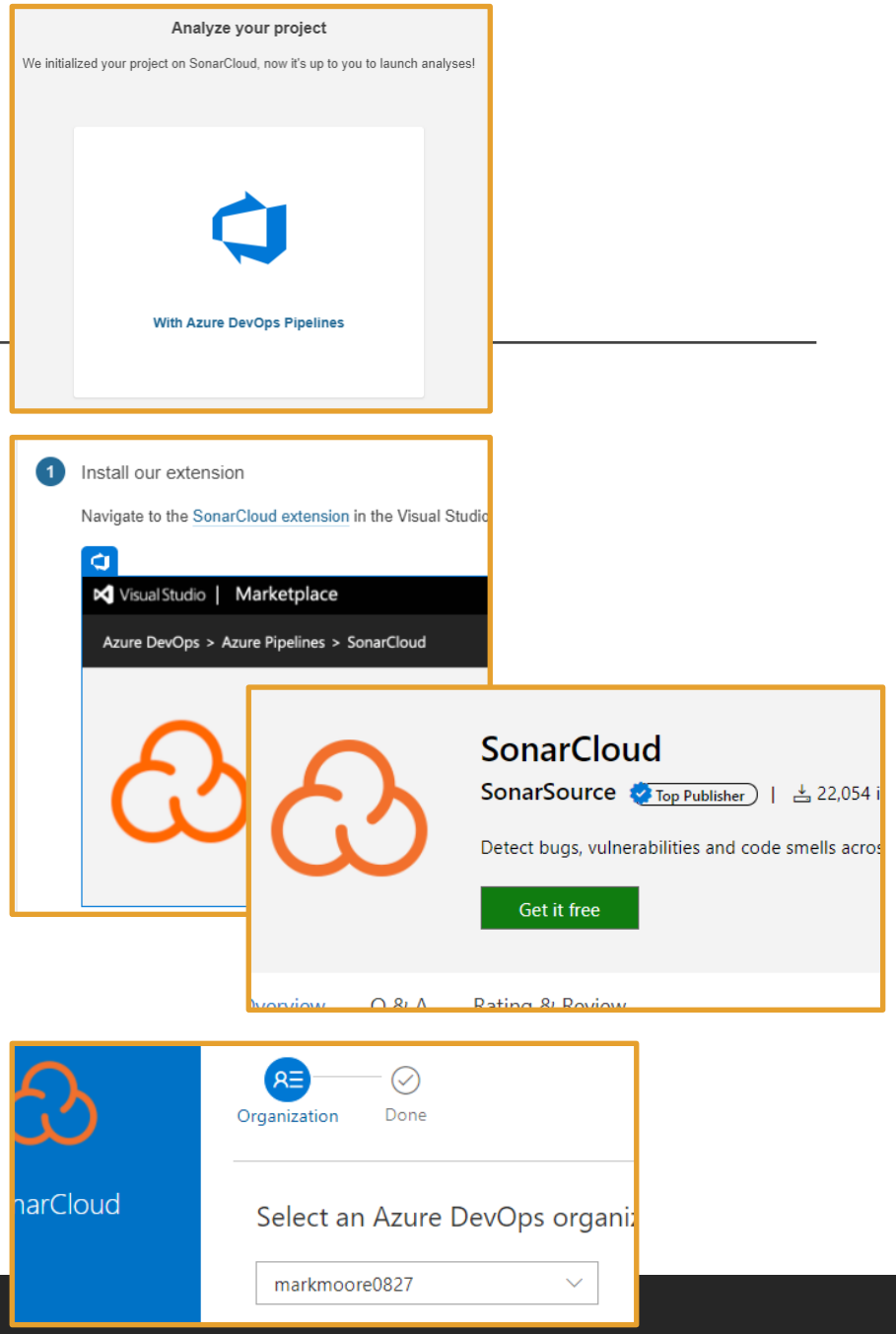


Configure Code Analysis

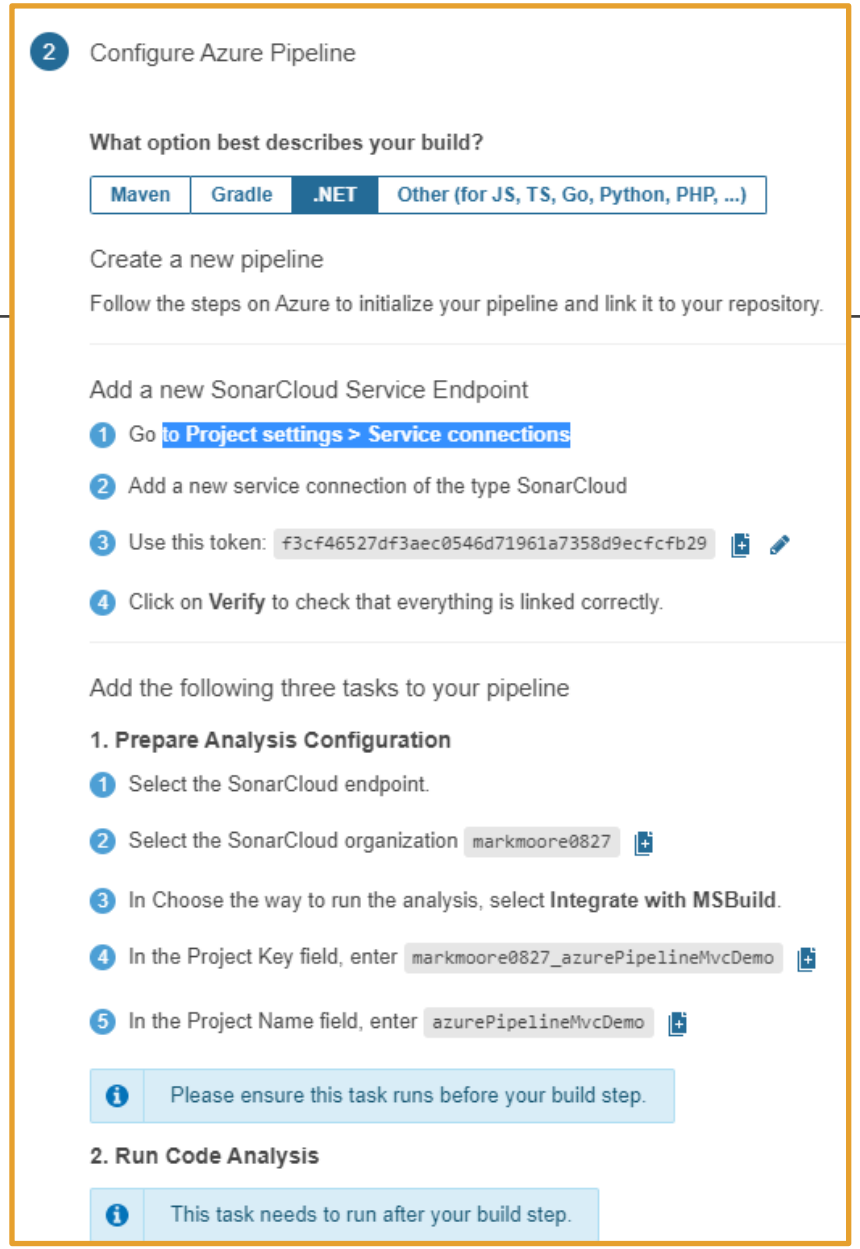
Click on ‘With Azure DevOps Pipelines

Click to download the SonarCloud Extension from the from the Visual Studio Marketplace.

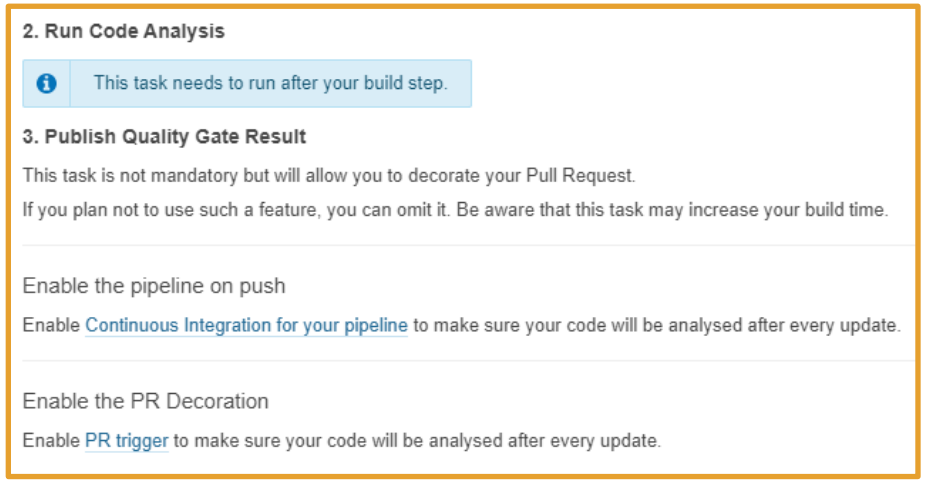
Make sure to choose the correct organization when installing in the Marketplace.



Configure Azure Pipeline

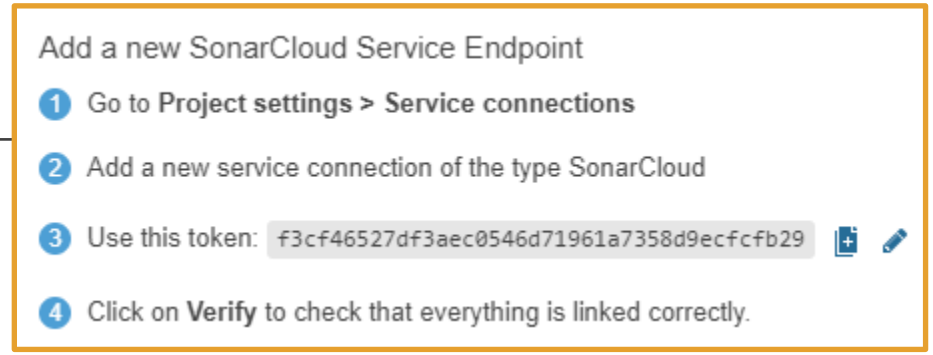


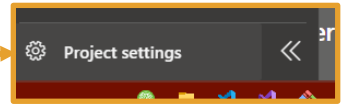
These steps will guide you through setup on the Azure DevOps side.

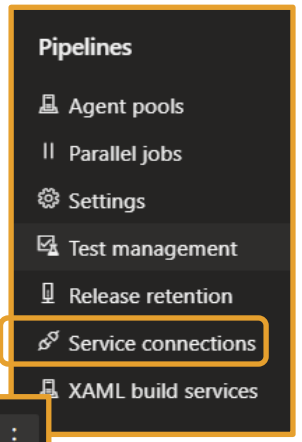


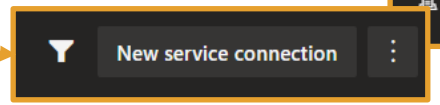
Add a new SonarCloud Service Endpoint

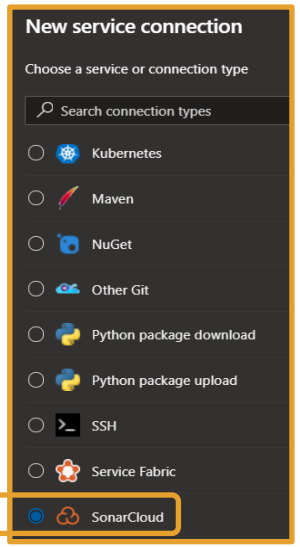
1. In your Azure DevOps account, click on the project you will be adding the Code Analysis to.
2. Click Project settings at the bottom left of your project home page.
3. Click the ‘back’ arrow next to Project details.
4. Click Service Connections under Pipelines.
5. Click ‘New service connection’ in the upper right.
6. Select ‘SonarCloud’ in the ‘New Service Connection’ List
7. Click Next.











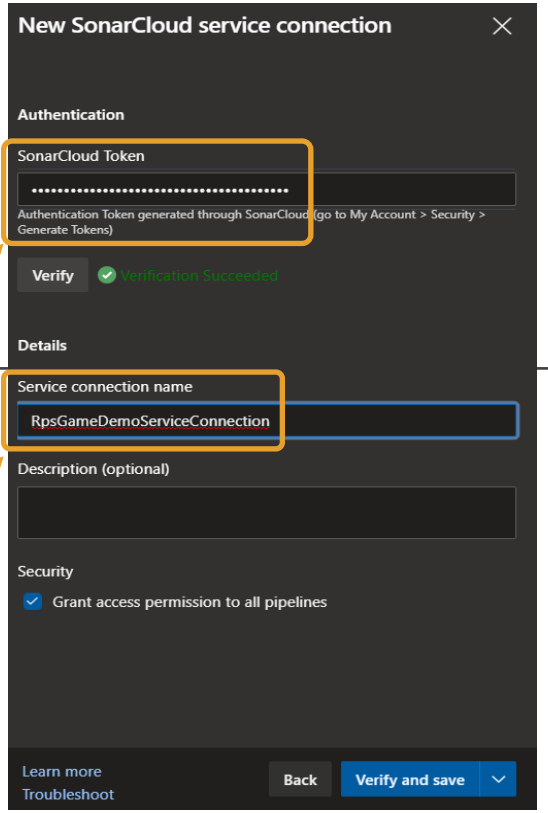
New SonarCloud Connection

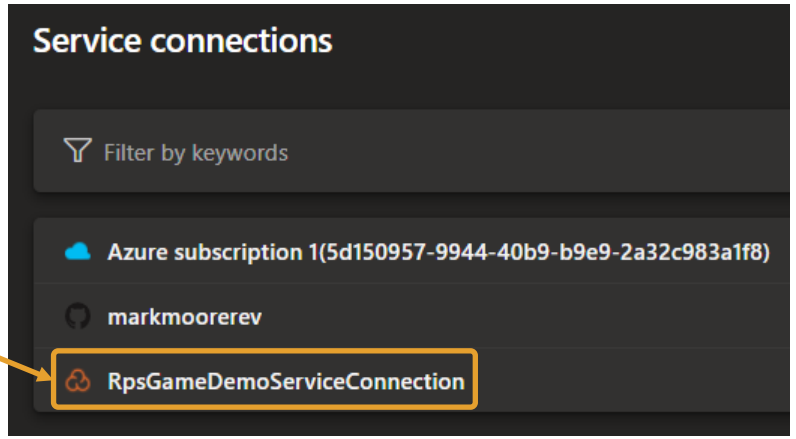
Paste in the P.A.T under ‘SonarCloud Token’.

Create a name for your connection under ‘Service connection name’

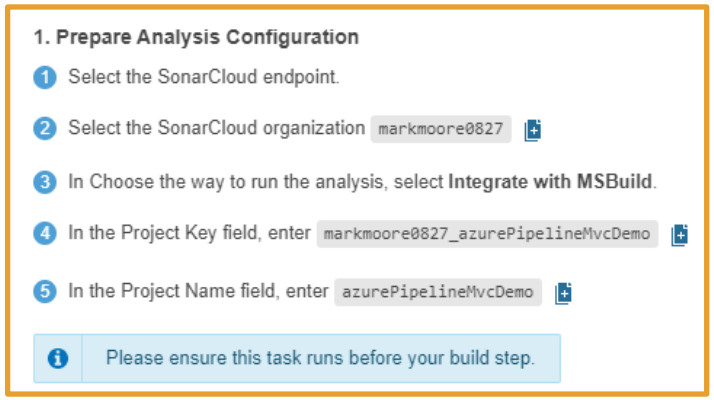
Click ‘Verify and Save’.

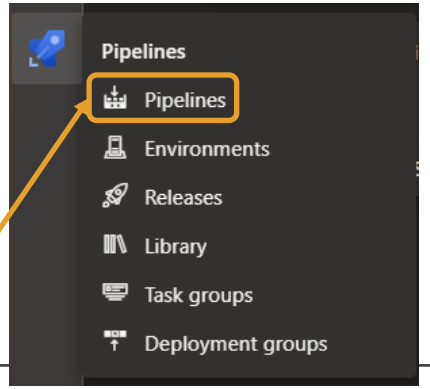
Verify the connection was successful.

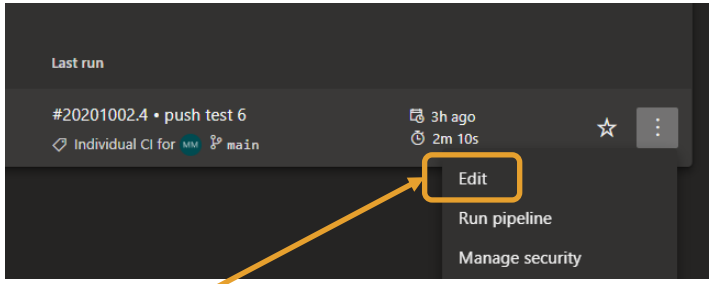


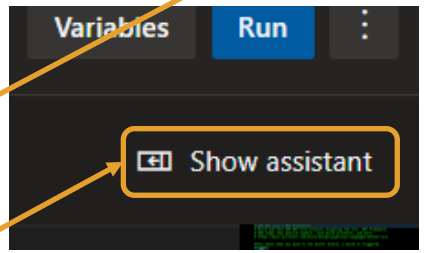


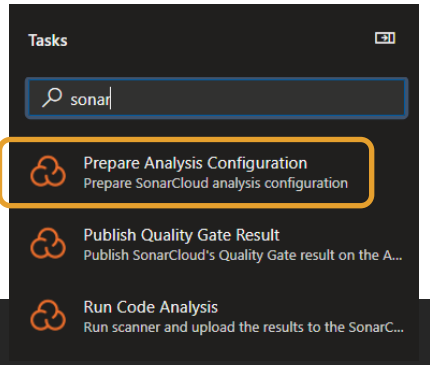
Add publish task to your pipeline YAML











Click Pipelines under the Rocket (Pipelines) avatar in the left column.

Click Edit under the pipeline you want to add code analysis to.

Click ‘Show Assistant’ in the upper left.

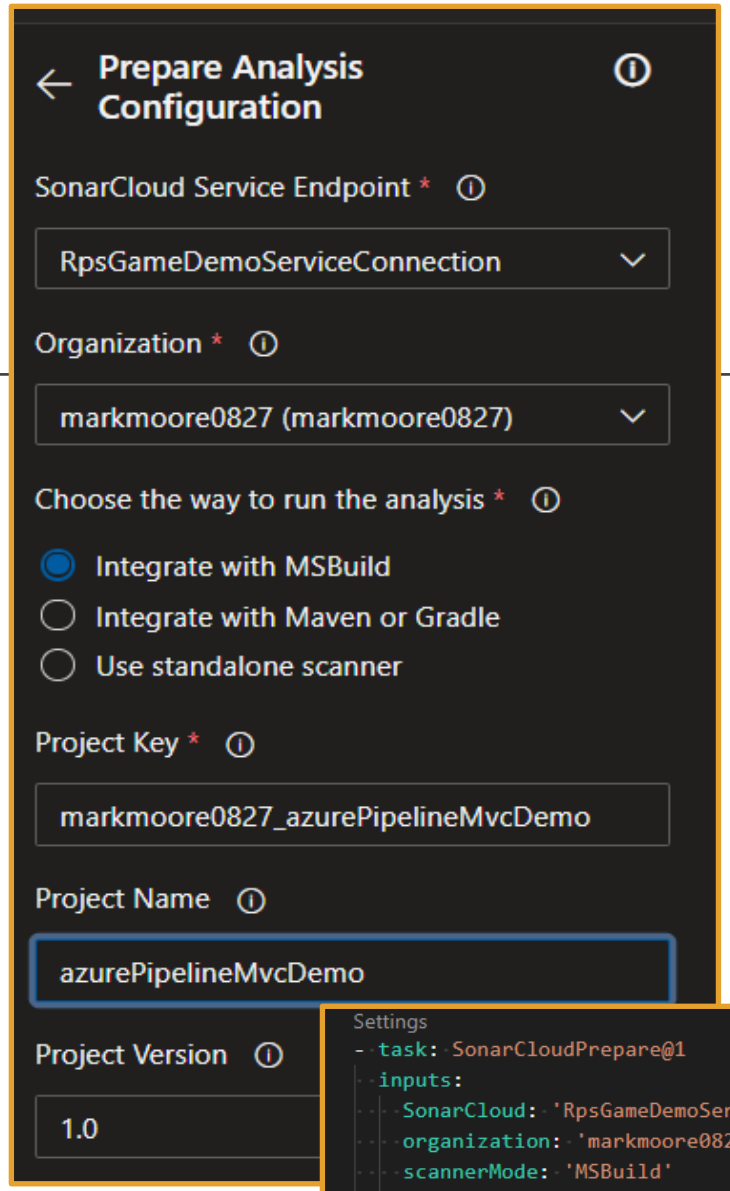
Select ‘Prepare Analysis Configuration’

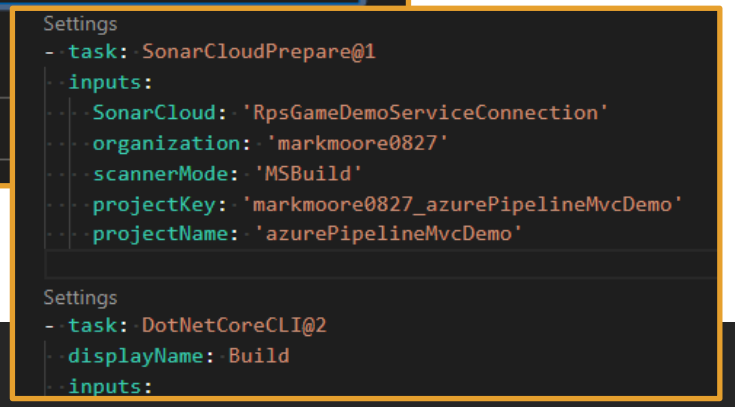
Add Code Analysis configuration task to your pipeline YAML

1. Prepare Analysis Configuration

1. Select the SonarCloud endpoint.
2. Select the SonarCloud organization `markmoore0827`
3. In Choose the way to run the analysis, select **Integrate with MSBuild**.
4. In the Project Key field, enter `markmoore0827_azurePipelineMvcDemo`
5. In the Project Name field, enter `azurePipelineMvcDemo`

Please ensure this task runs before your build step.





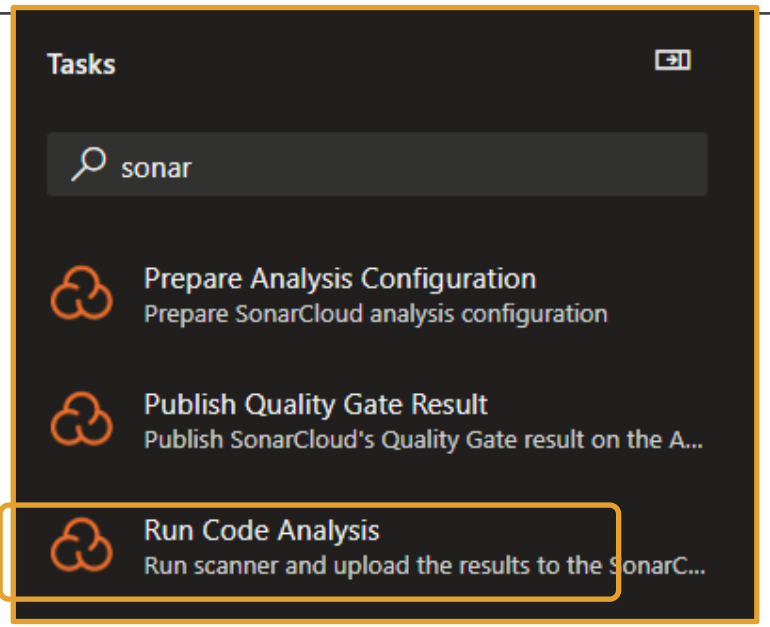
You don't have to change anything under 'Advanced'
Place the curser above your 'build' Task.
Click 'Add'

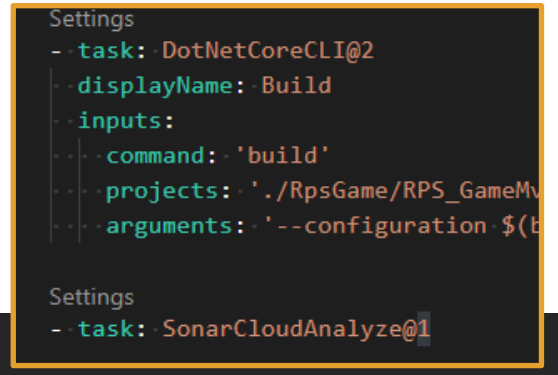
Add Code Analysis to your YAML

Click ‘Show Assistant’ to return to the Assistant.

Make sure the curser is immediately below the ‘build’ task.

Click ‘Run Code Analysis’.

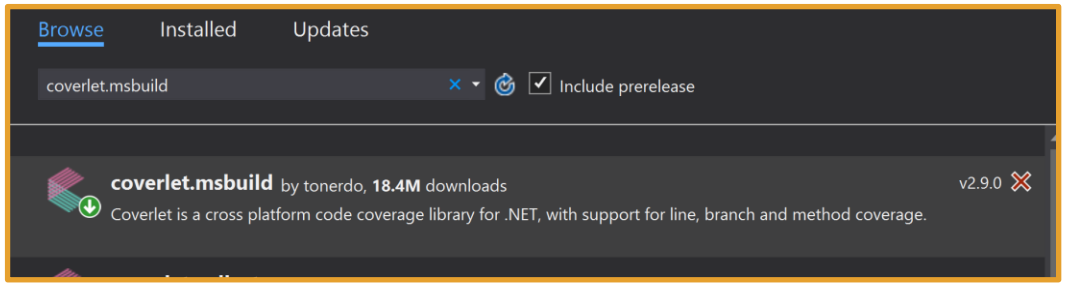


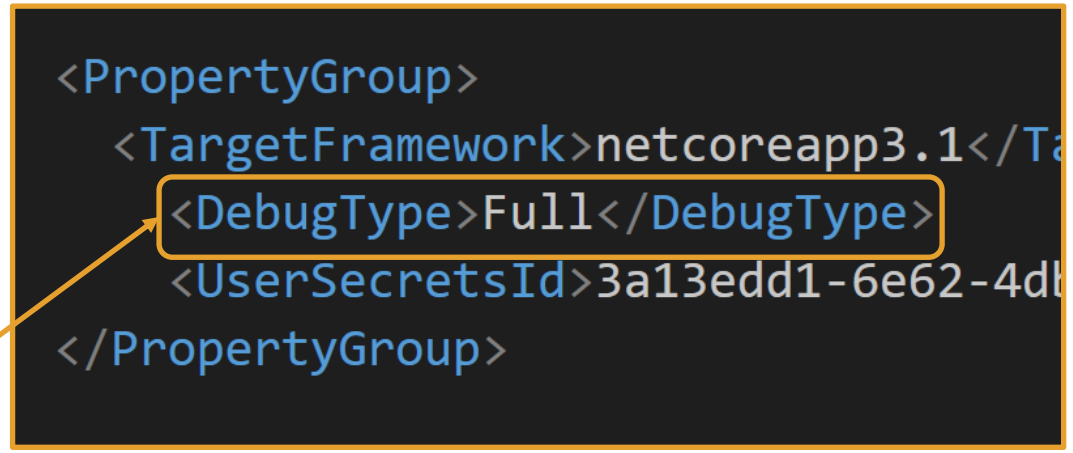


Add Coverlet and DebugType to your project

For the code coverage analysis and report to work, your test project needs to use the '***coverlet.msbuild***' package. (it may be possible to use something else, but this is what I used)

1. Install '***coverlet.msbuild***' to ONLY your test project in Visual Studio.
2. For each project, click the project name. Under <PropertyGroup>, add <DebugType>Full</DebugType> for the ***coverlet.msbuild*** tool to function properly.

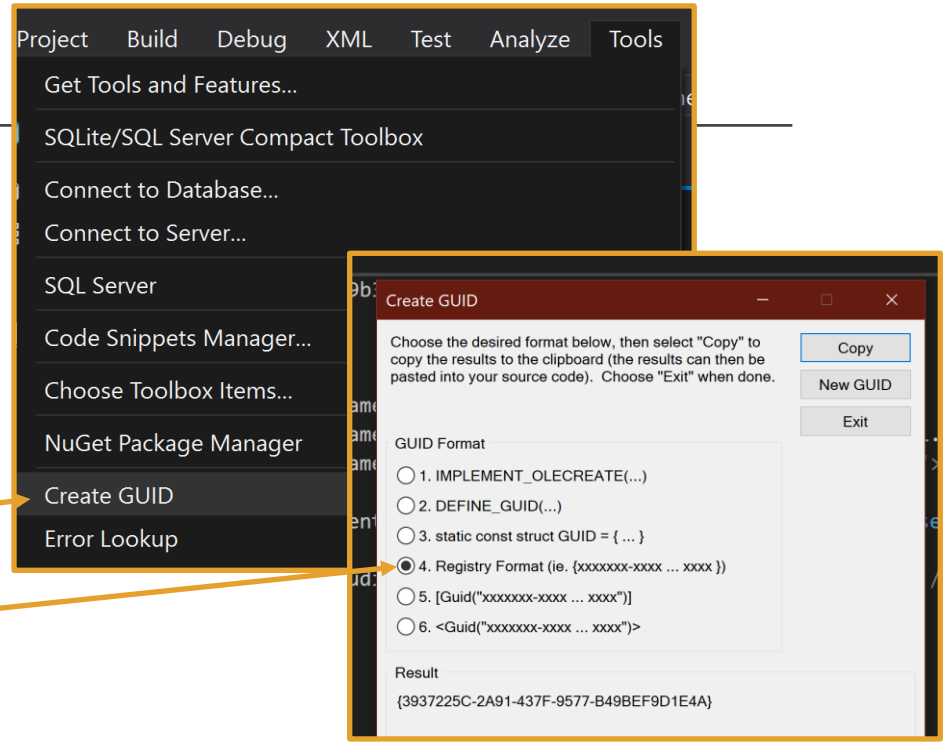




Add the Project GUID Property to your projects

Each Project needs a <ProjectGuid>#</ProjectGuid> property so that SonarCloud can differentiate between them and analyze them properly.

1. To generate a GUID, click on ***Tools*** in the menu bar in Visual Studio.
2. Click '***Create GUID***'.
3. Select '***4. Registry Format***'.
4. Copy the GUID
5. Add the GUID to a new <ProjectGuid> the Project.




```
<PropertyGroup>
  <TargetFramework>netcoreapp3.1</TargetFramework>
    <DebugType>Full</DebugType>
    <UserSecretsId>3a13edd1-6e62-4db1-9578-572a9b314abd</UserSecretsId>
    <ProjectGuid>{EDC476A1-A5C3-4C7D-BC18-469A28F48169}</ProjectGuid>
</PropertyGroup>
```

Darius Vallejo example

Darius Vallejo based his integration on the below tutorial.

https://azuredevopslabs.com/labs/vstsextend/sonarcloud//