**Microsoft Azure DevOps – SonarCloud**

SonarCloud is the leading online service for Code Quality & Code Security. It is totally free for open-source projects, and supports all major programming languages including C#, VB .Net, JavaScript, TypeScript, C/C++ and many more. If your code is closed source, SonarCloud also offers a paid plan to run private analyses.

This Azure DevOps extension provides build tasks that you can add in your build definition. You'll benefit from automated dectection of bugs and vulnerabilities across all branches and Pull Requests. SonarCloud explains all coding issues in details, giving you chance to fix your code before even merging and deploying, all while learning best practices along the way. At project level, you'll also get a dedicated widget that tracks the overall health of your application.

This extension provides the following features:

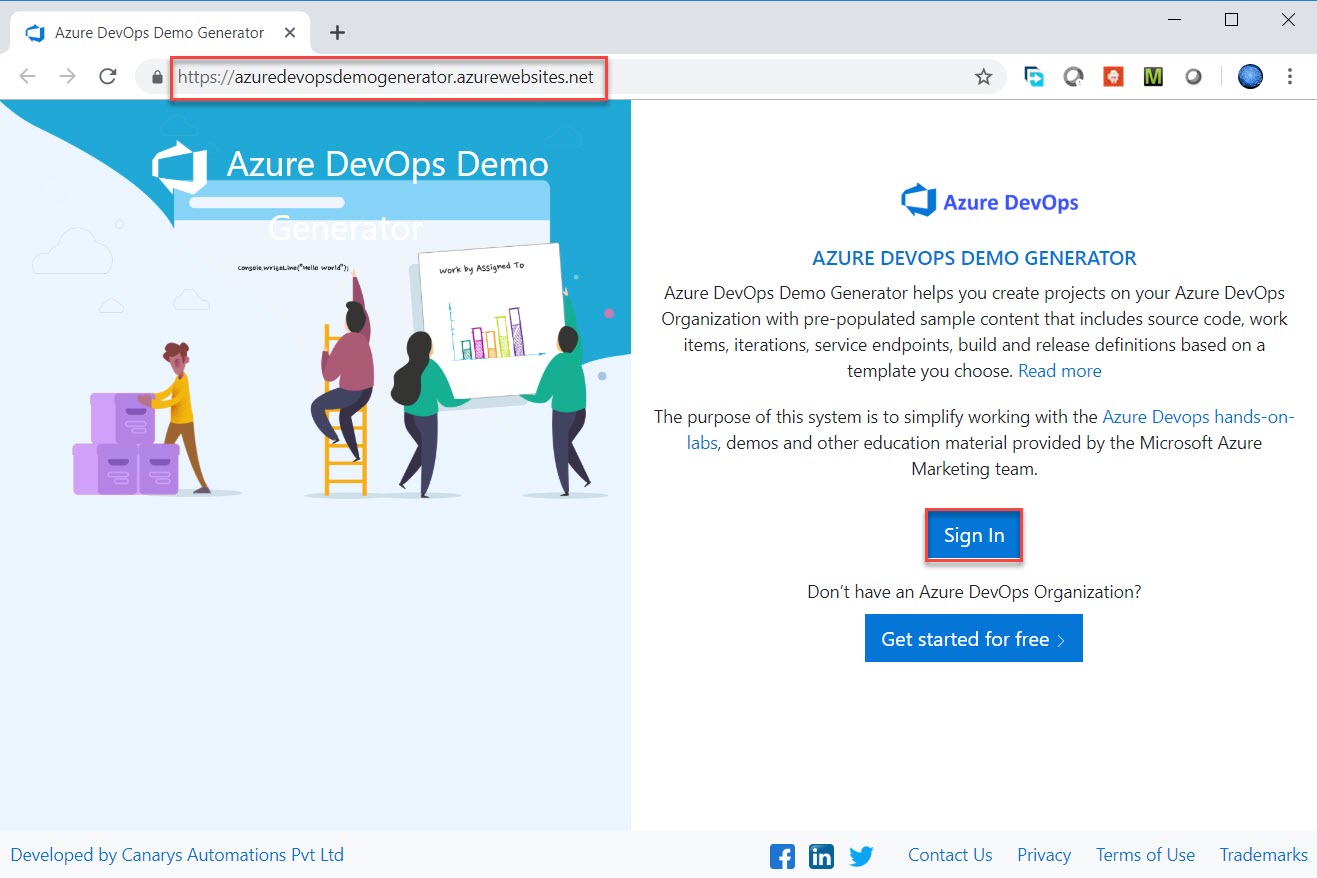
* A dedicated **SonarCloud EndPoint** to set the user token and validate the connection.
* Three build tasks (along with build templates) to get your projects analyzed easily:
  + **Prepare Analysis Configuration** task, to configure all the required settings before executing the build. This task is mandatory. In case of .NET solutions or Java projects, this tasks helps to integrate seamlessly with MSBuild, Maven and Gradle tasks.
  + **Run Code Analysis** task, to actually execute the analysis of the source code. Not required for Maven or Gradle projects.
  + **Publish Quality Gate Result** task, to display the quality gate status in the build summary. This tasks is optional, as it can increase the overall build time.
* Analysis of the branches and the **pull requests** of your projects
* A **widget** to monitor the quality gate for your projects on your favorite dashboard

In this demo, we are using **Parts Unlimited** Sample project

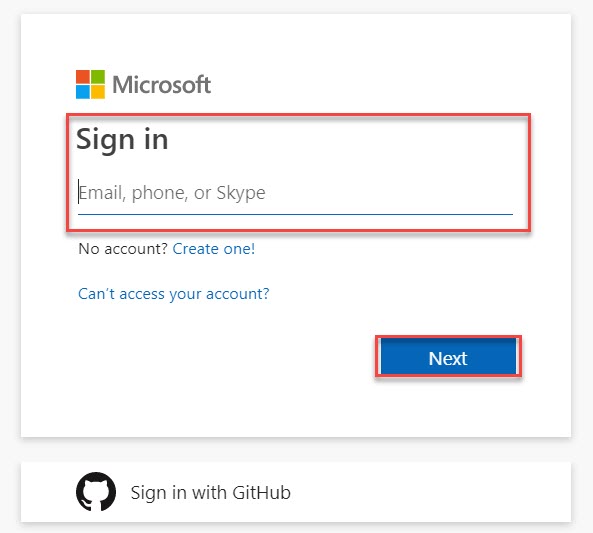
Step 1: Open **Azure DevOps Demo Generator**

<https://azuredevopsdemogenerator.azurewebsites.net/>

Click on **Sign In** button.

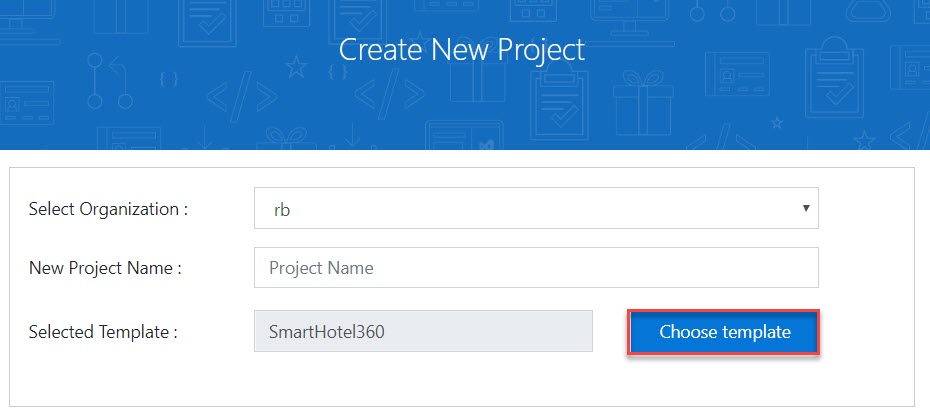


Step 2: Enter Azure DevOps credentials

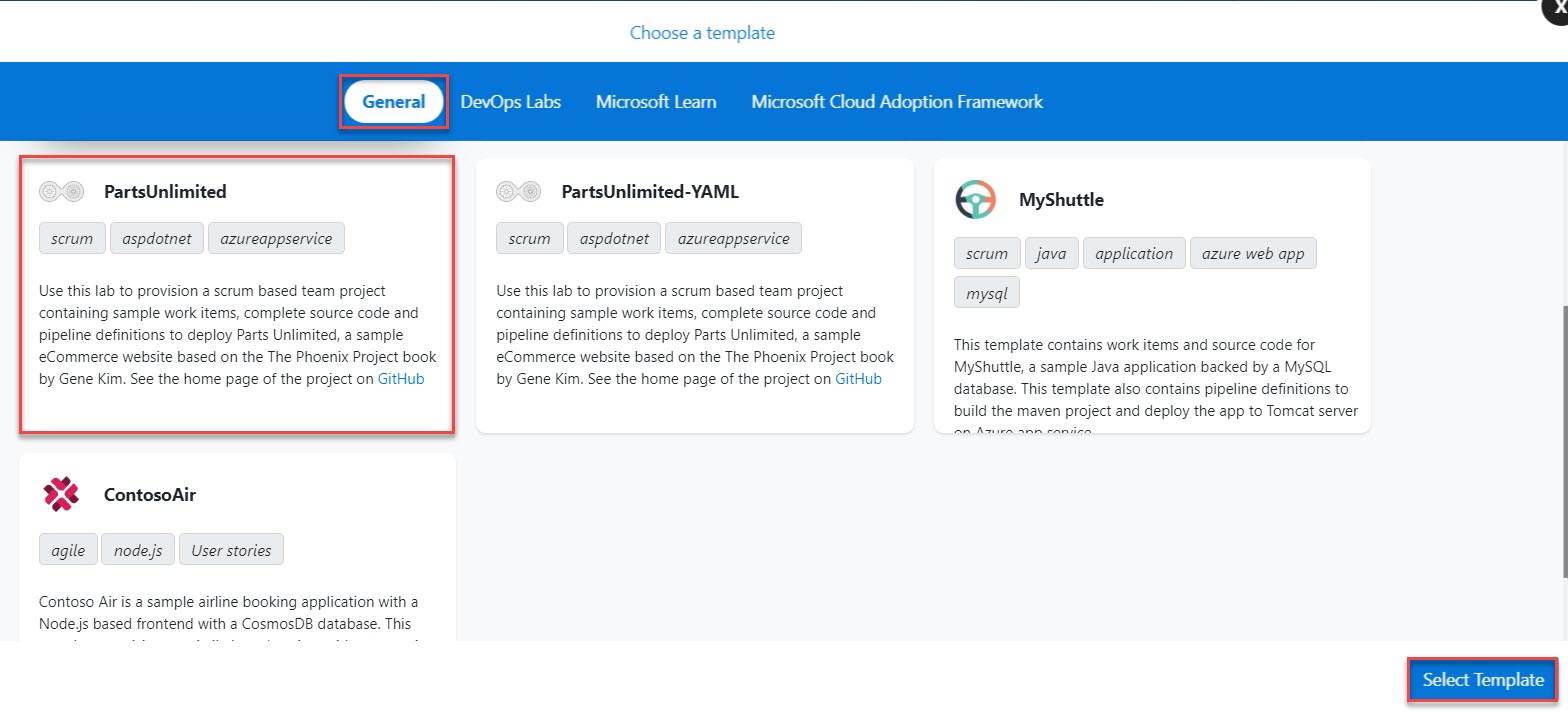


Step 3: Select **Organization**

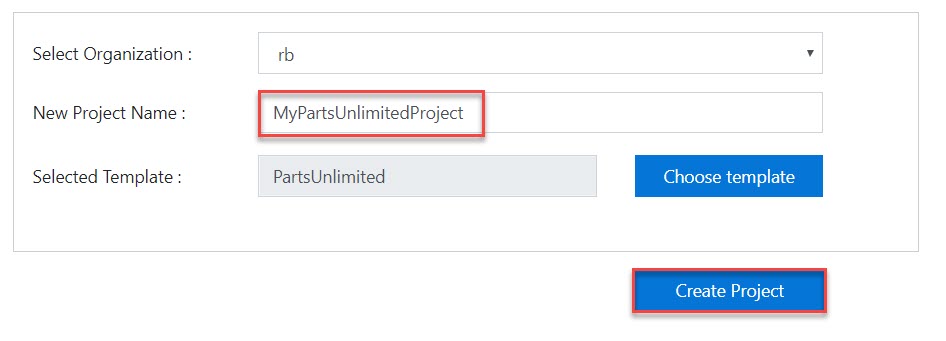
Click on **Choose Template** option



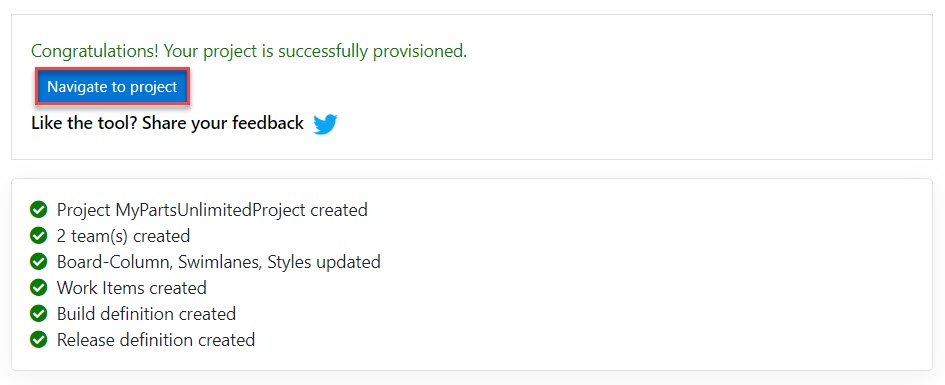
Step 4: Select **PartsUnlimited** template



Step 5: Enter Project Name: **MyPartsUnlimitedProject** and Click on **Create Project**.

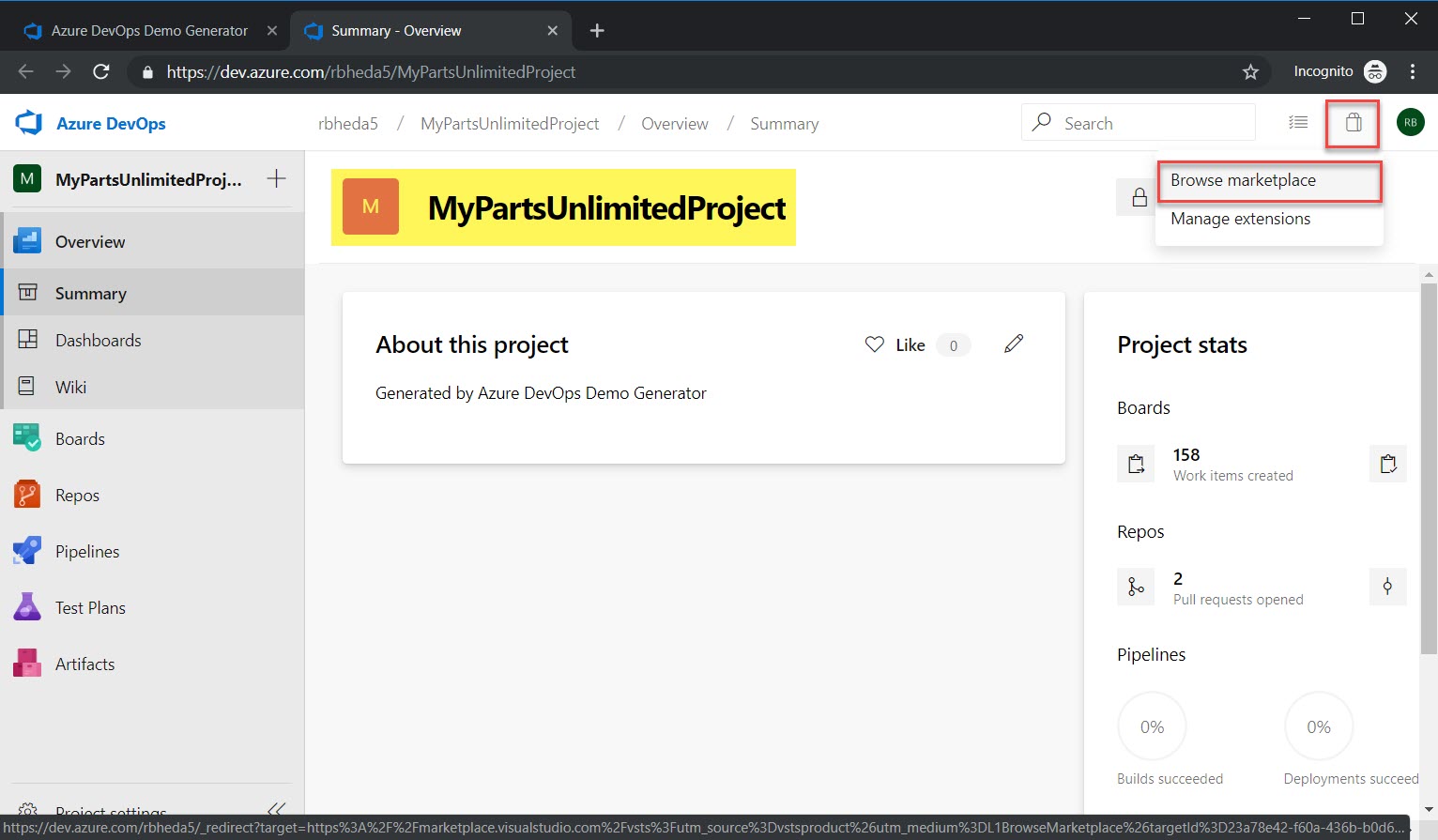


Step 6: Click on **Navigate to Project** button.

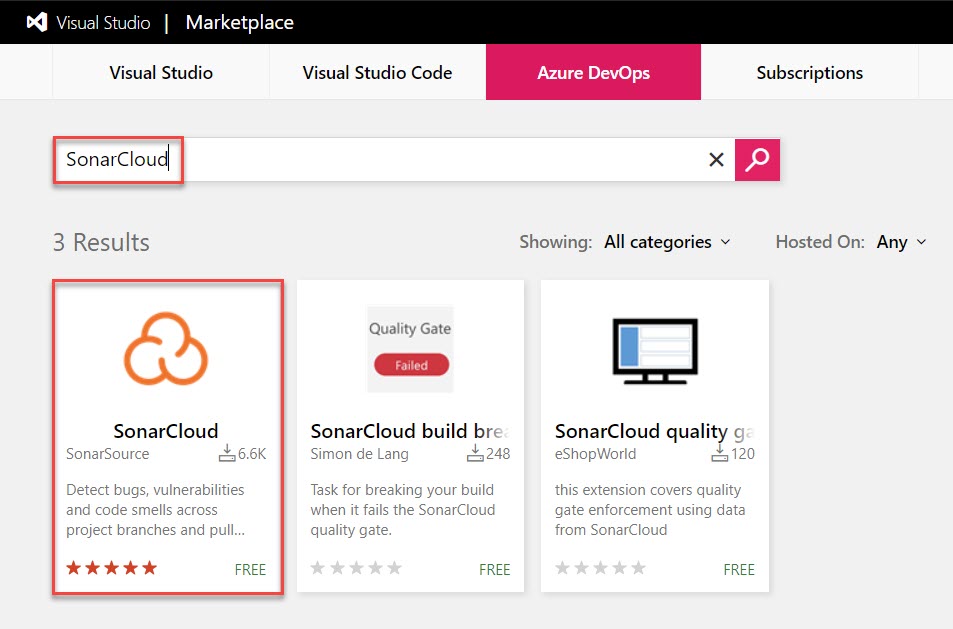


Step 7: Within few seconds Project will load as below:

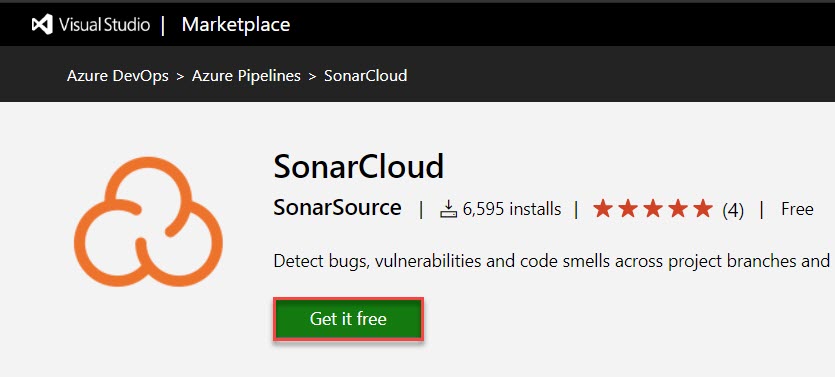
Select **Browse marketplace** option.



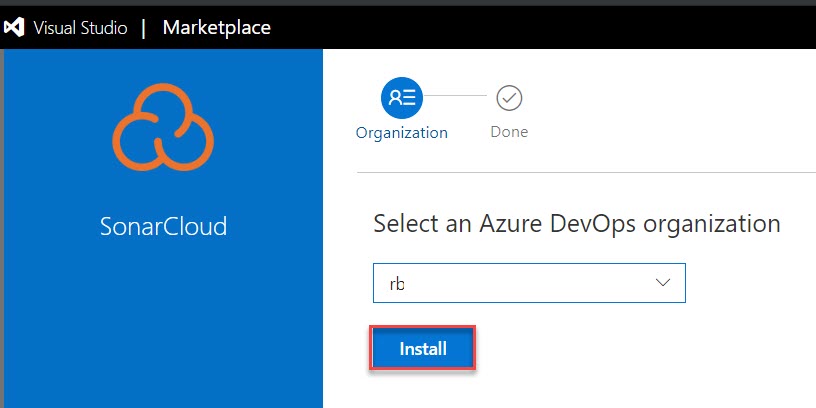
Step 8: Search for **SonarCloud** and click on that.



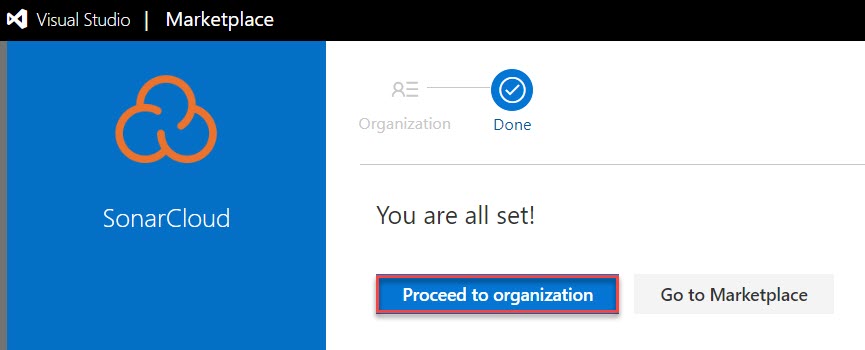
Step 9: Click on **Get it free**.



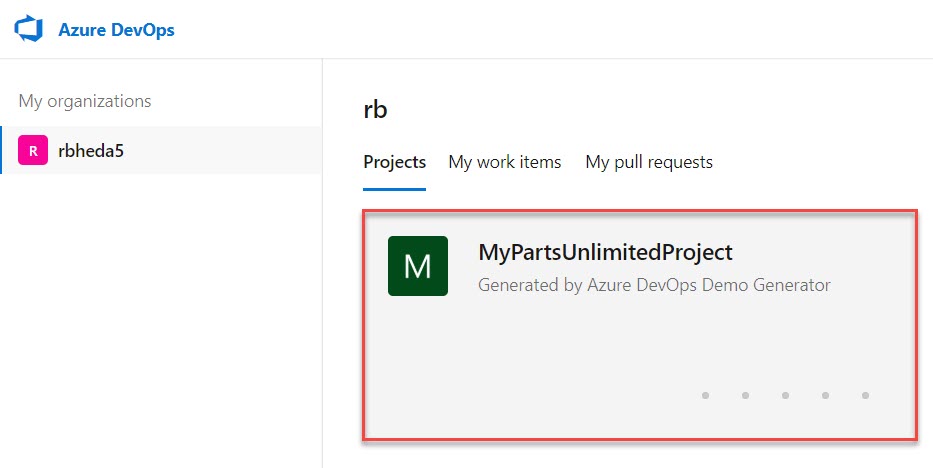
Step 10: Select **organization** and click on **Install** button.



Step 11: Click on **Proceed to Organization**



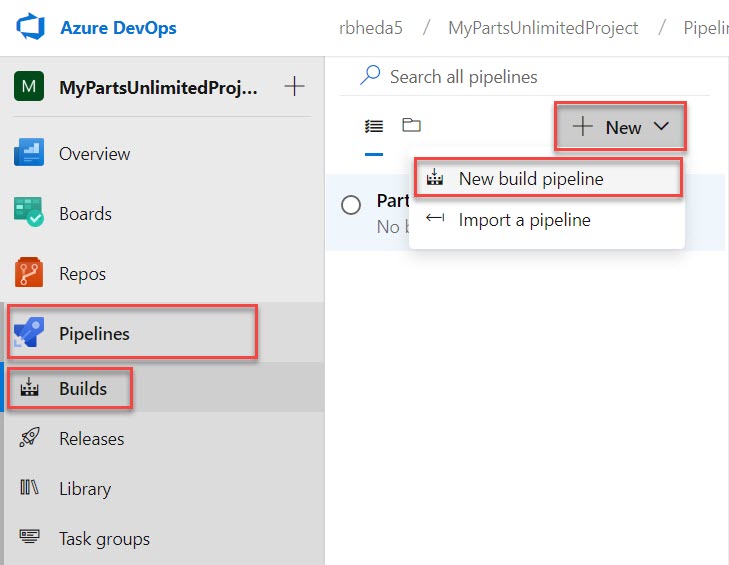
Step 12: Click on **Project**



Step 13: Create New Build Pipeline

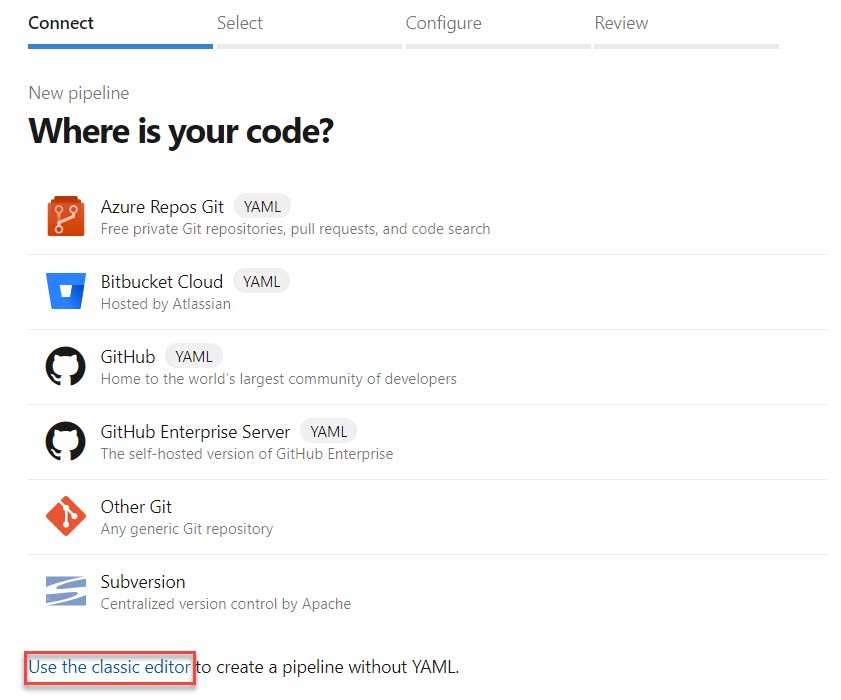
**Pipelines -> Builds**

**Click on New -> New build pipeline**



Step 14: New version of pipeline also available but for this demo we are using Classic editor.

Click on **Use the classic editor** option

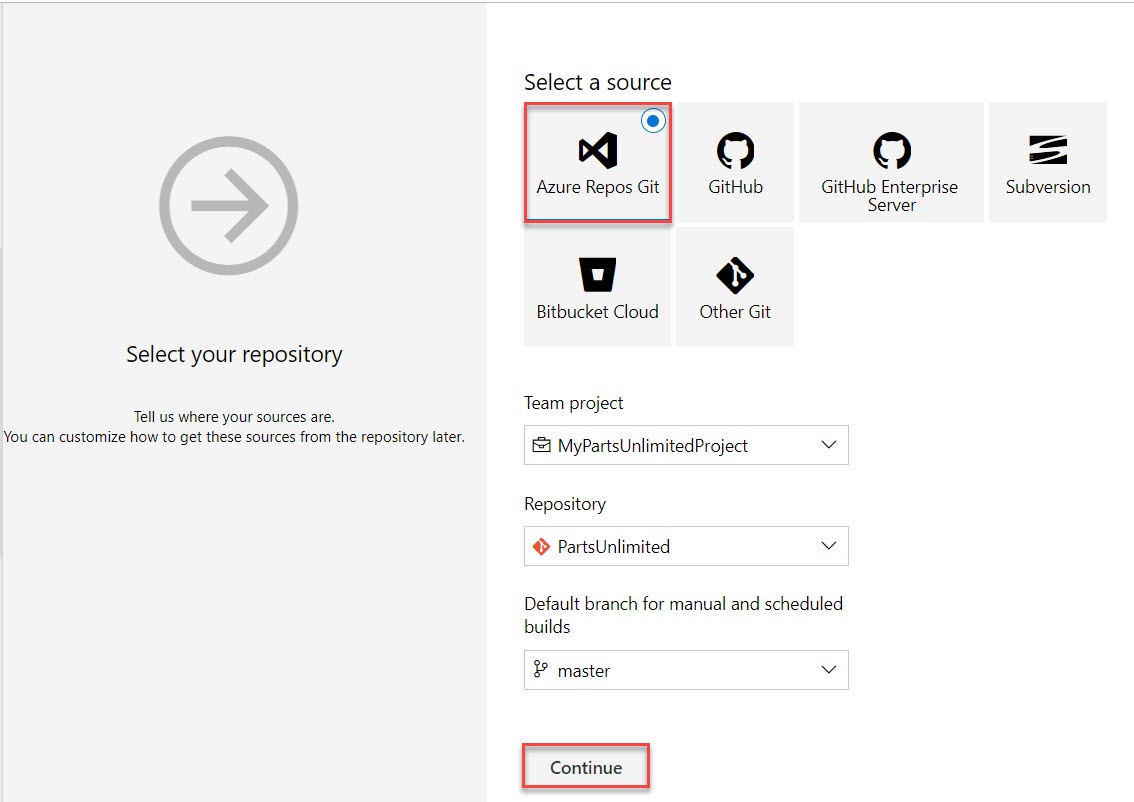


Step 15: Select Source repository

Source code is available in Azure Repo so select **Azure Repos Git** option.

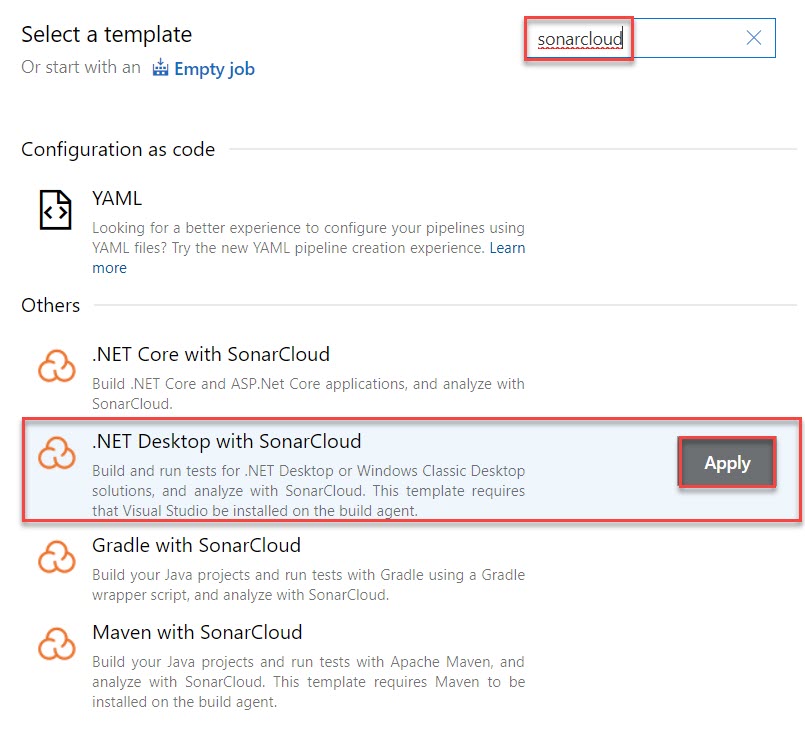
Verify Team Project, Repository and branch.

Click on **Continue** option



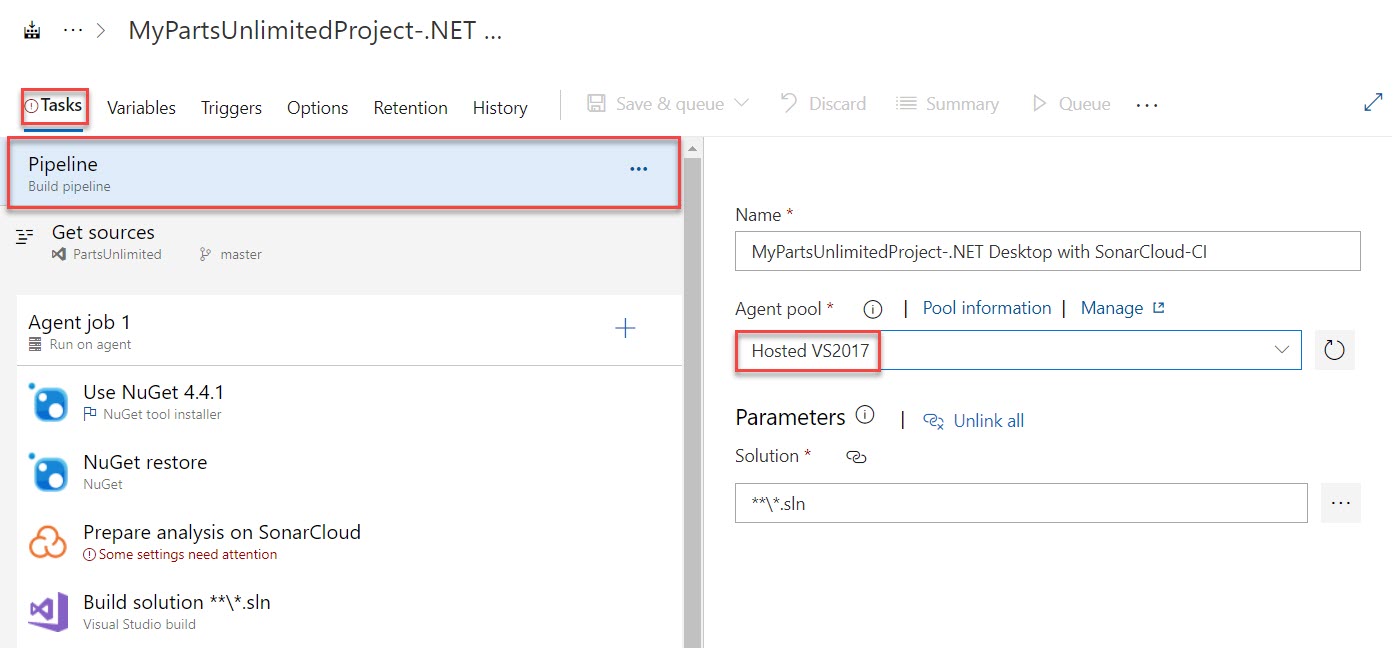
Step 16: Search for **sonarcloud** and based on Source code select from list.

This demo is developed using .Net so Select .**Net Desktop with SonarCloud**



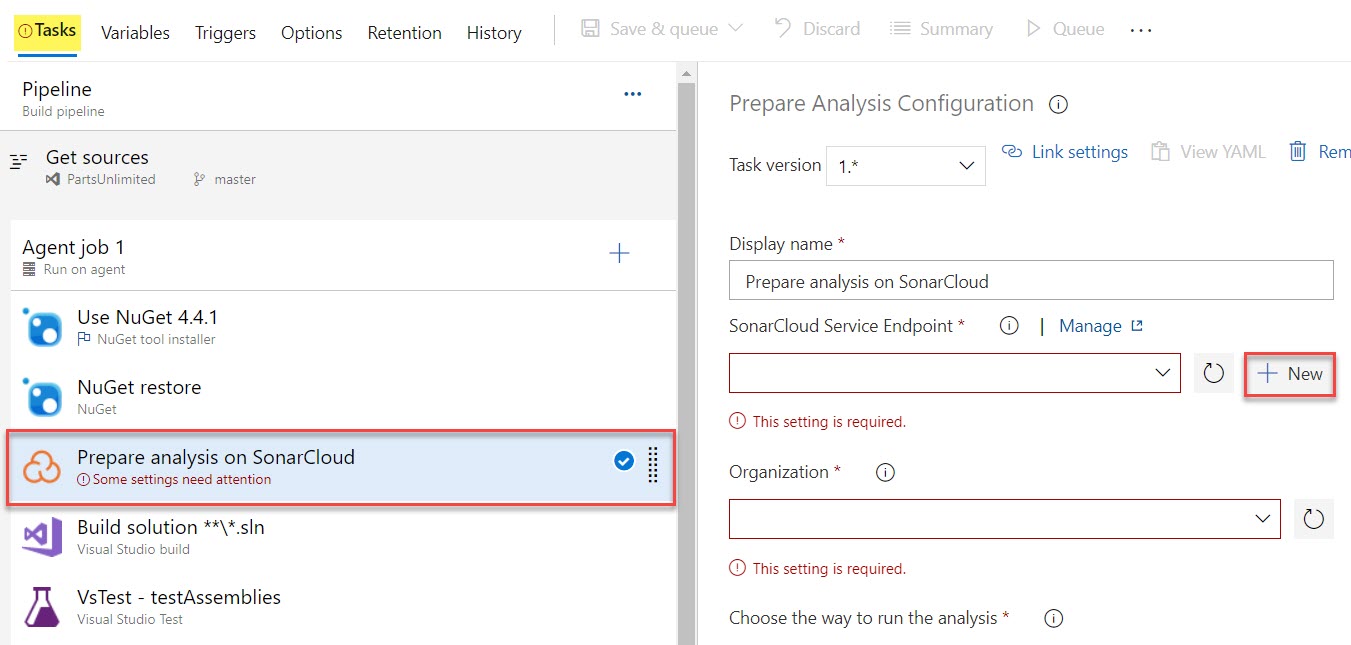
Step 17: Select Pipeline and choose agent pool from list.

Agent pool: **Hosted VS2017**



Step 18: From default pipeline select Prepare analysis on SonarCloud

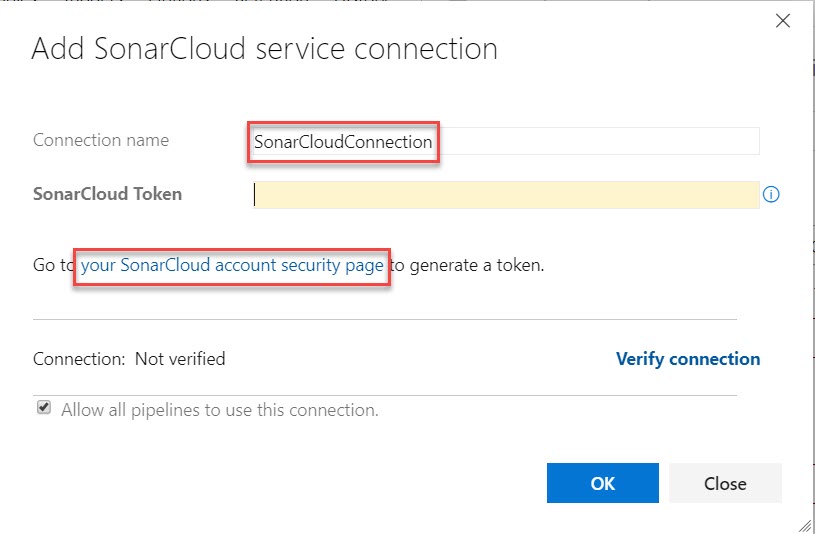
Create New Service Endpoint by clicking on **+ New**



Step 19: SonarCloud service connection window will open

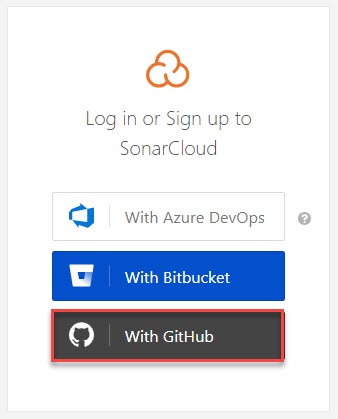
Connection name: **SonarCloudConnection**

To get SonarCloud Token click on **your SonarCloud account security page** option

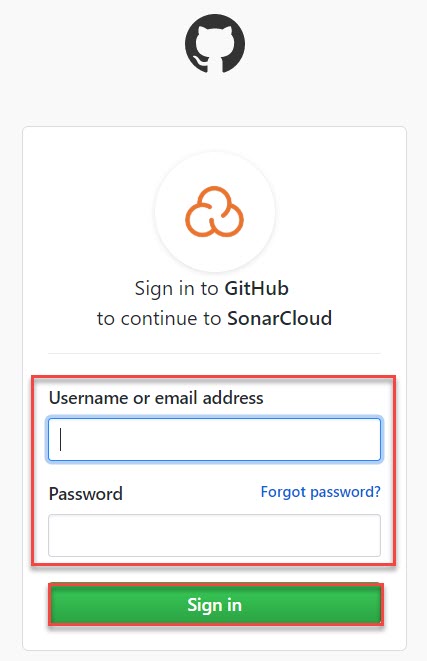


Step 20: Choose anyone of them from below option for login. But for this demo we are using **GitHub** option.

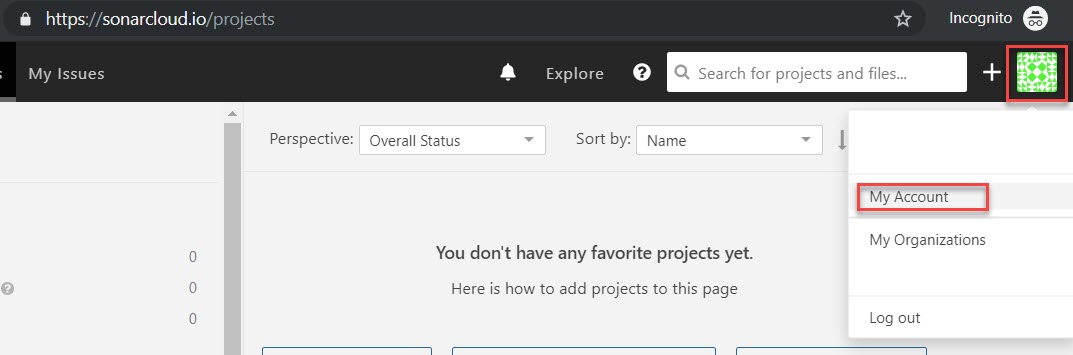
Note: With Azure DevOps option – only Work or School account will allow.



Step 21: Enter Github credentials.



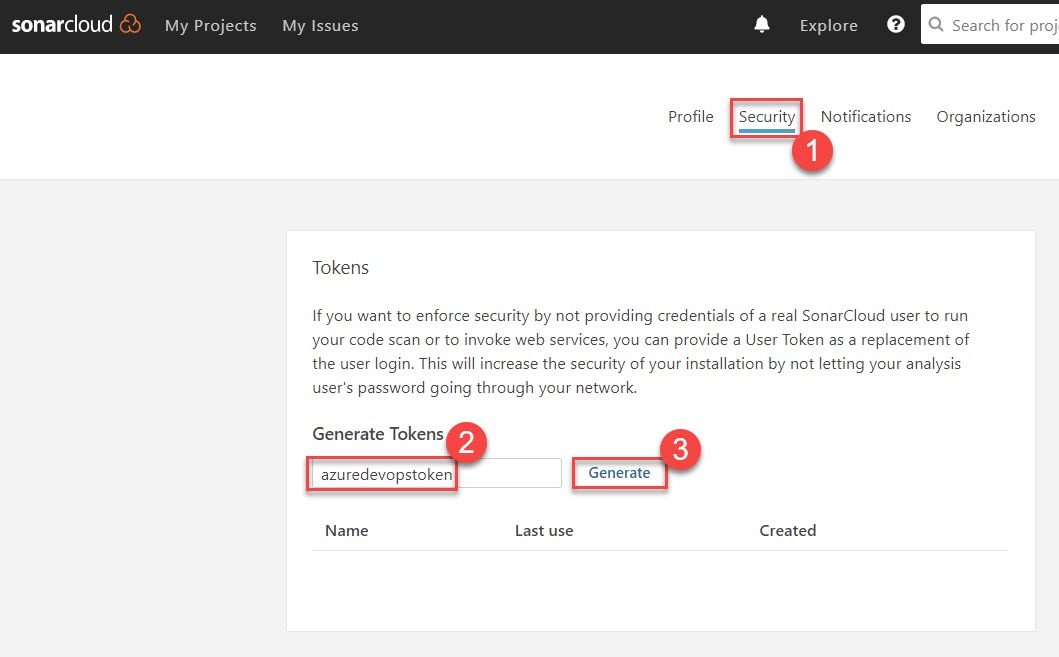
Step 22: Click on square logo icon and select **My Account** option



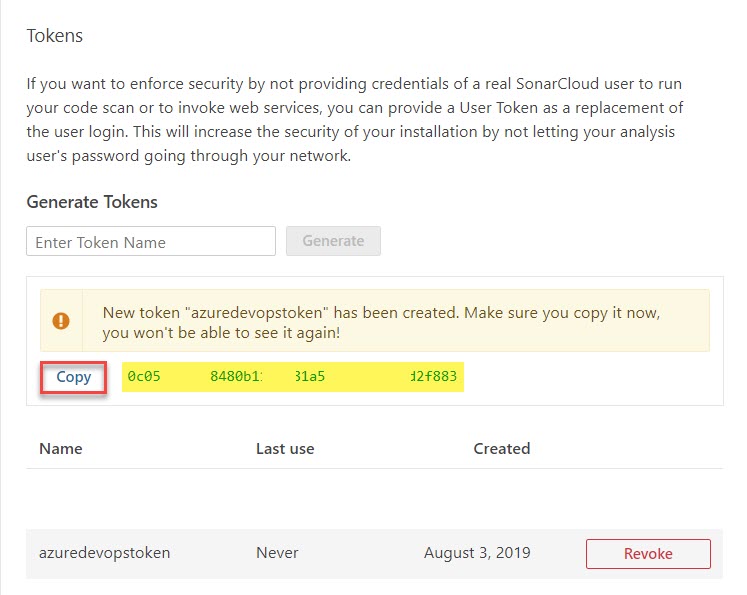
Step 23: To generate Token follow below steps:

Click on **Security** menu

Enter project name: **azuredevopstoken** and click on **Generate** button.

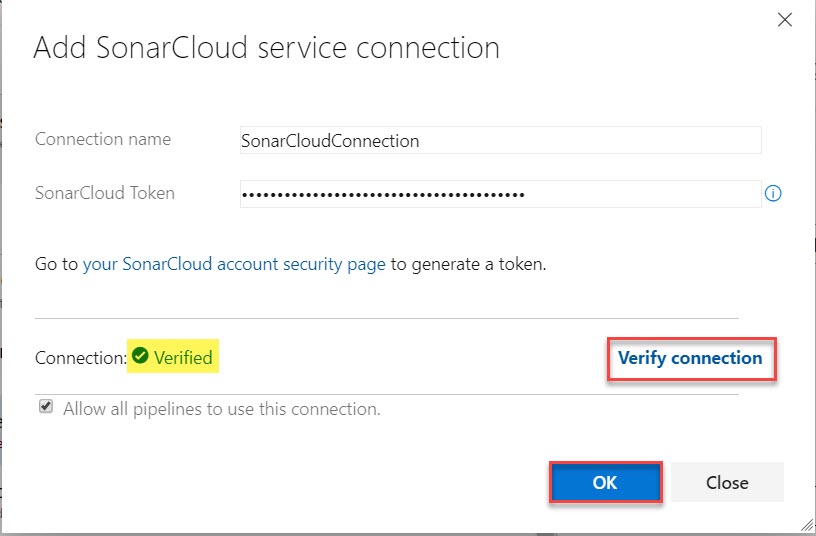


Step 24: **Copy** this Generated Token



Step 25: Paste SonarCloud Token.

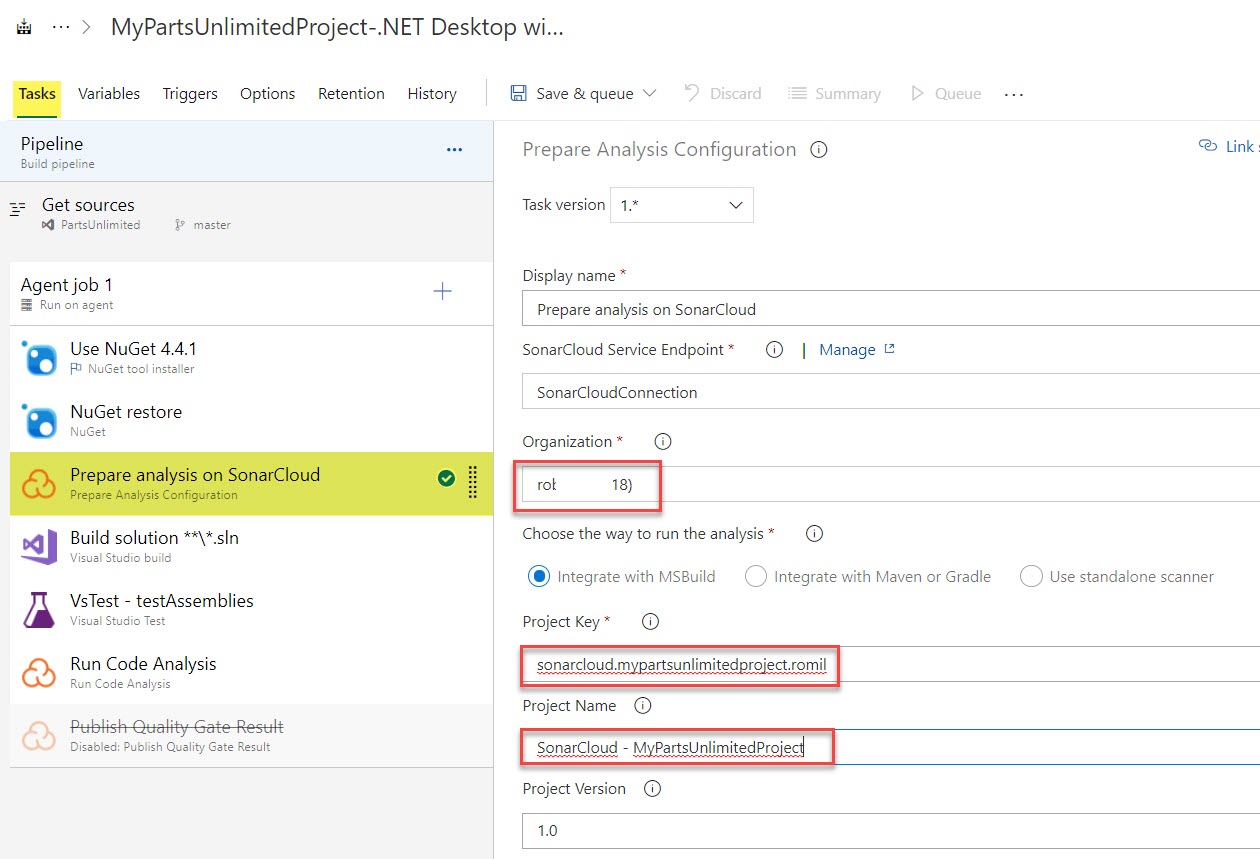
Click on **Verify connection** and click on **OK** button.



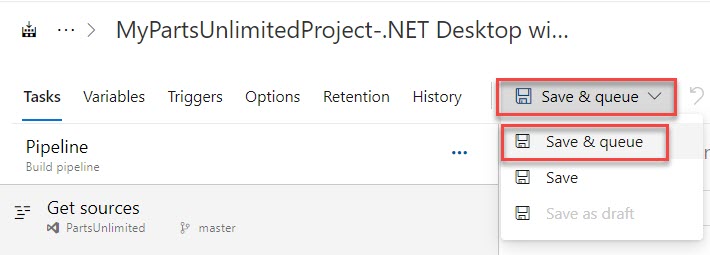
Step 26: Select **Organization** from list

Project Key: **sonarcloud.mypartsunlimitedproject.<key>**

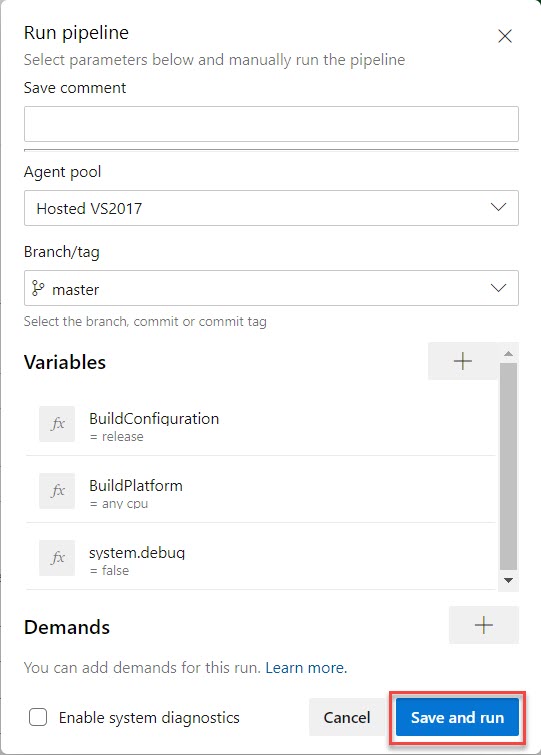
Project Name: **SonarCloud - MyPartsUnlimitedProject**



Step 27: Click on **Save & queue** option

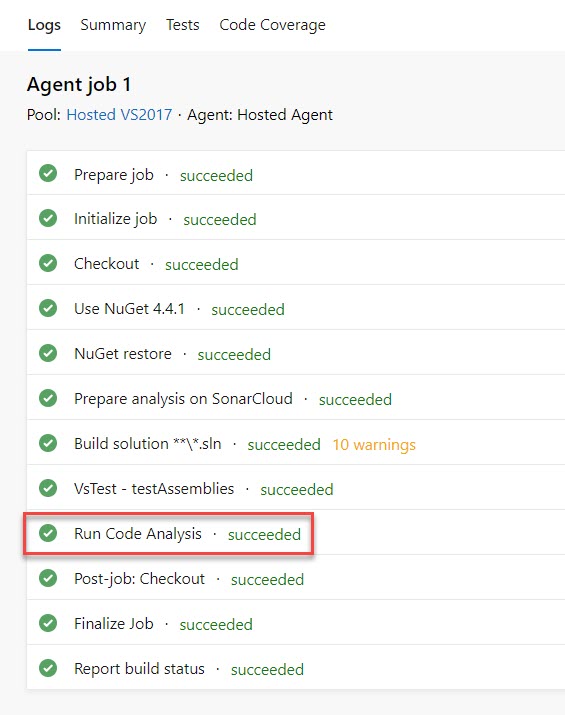


Step 28: Click on **Save and Run** button.



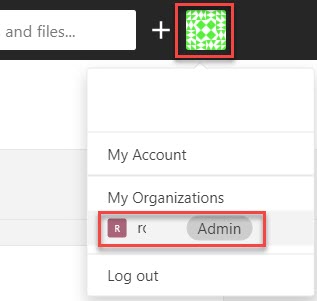
Step 29: Wait for few seconds to complete job.

Click on **Run Code Analysis** job.

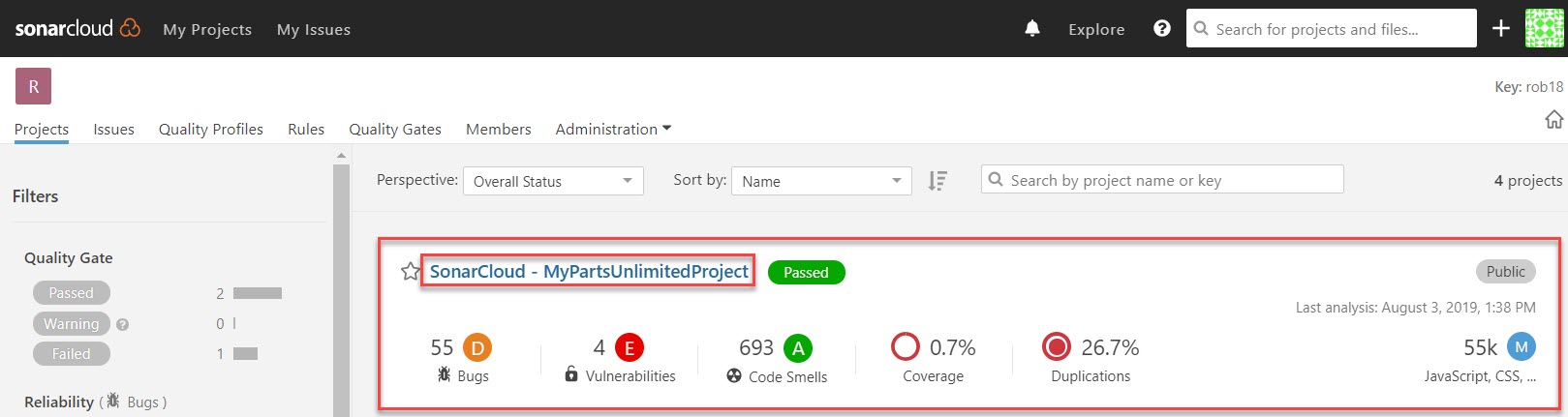


Step 30: To check result of Code Analysis navigate to Sonar Cloud site.

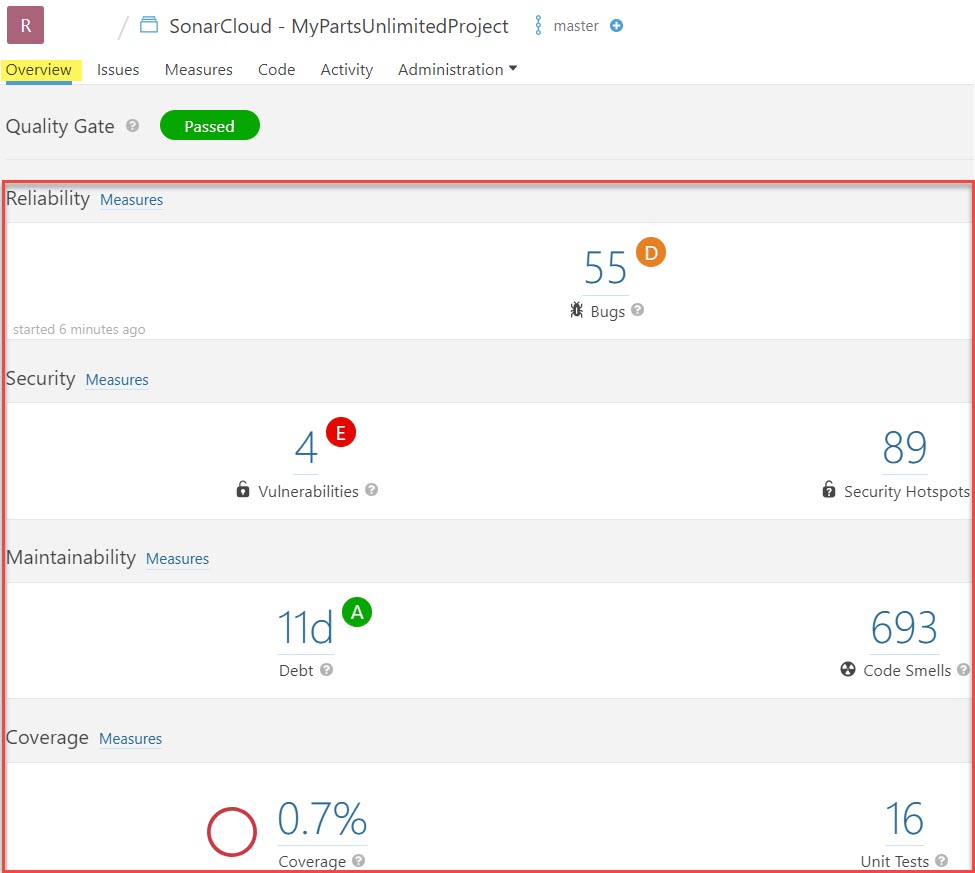
Select Once again **User logo and select Admin** option. Ex. John Doe (Admin)



Step 31: Summary will be there as below and click on Project Name.



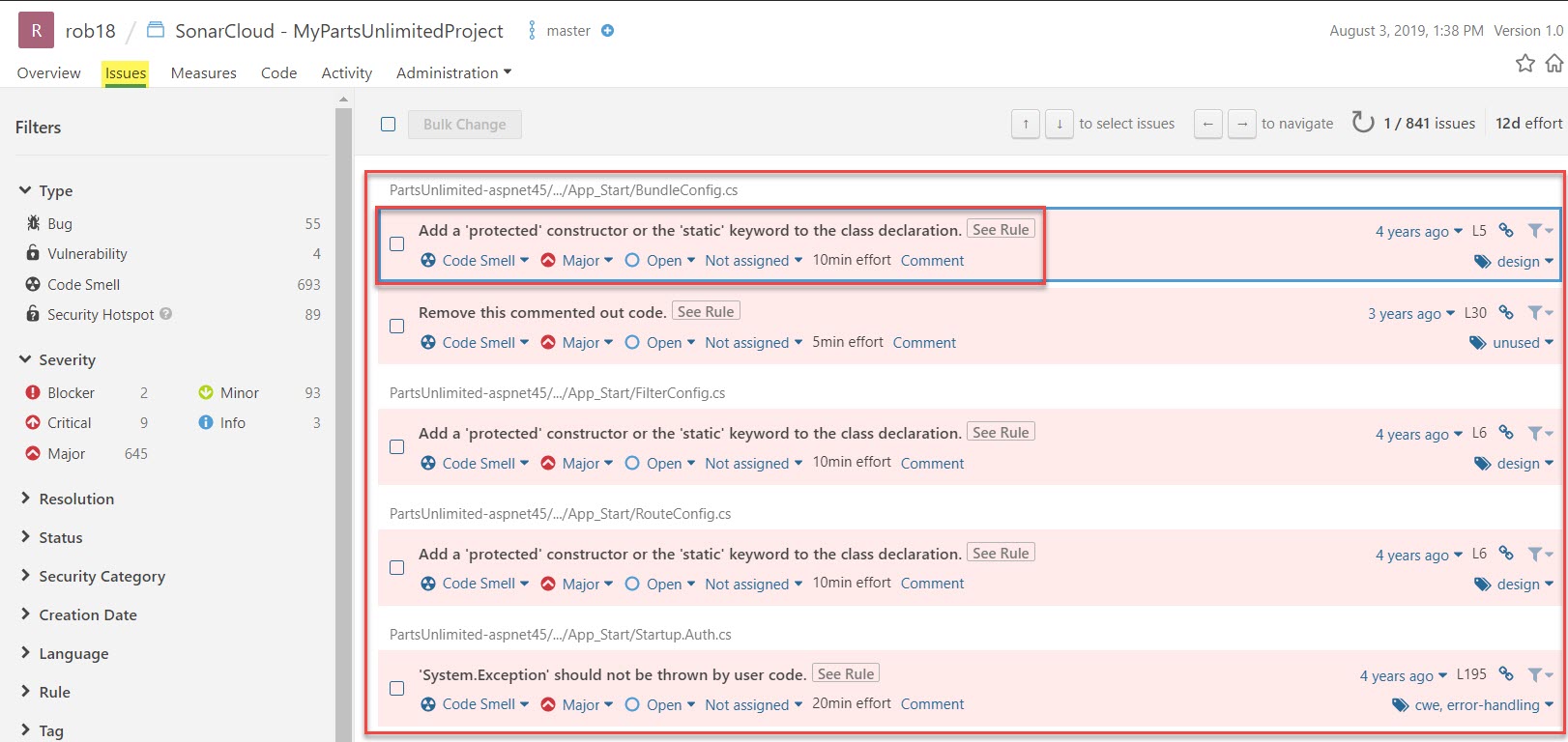
Step 32: Reliability, Security, Maintainability, Coverage etc available in **Overview** option.



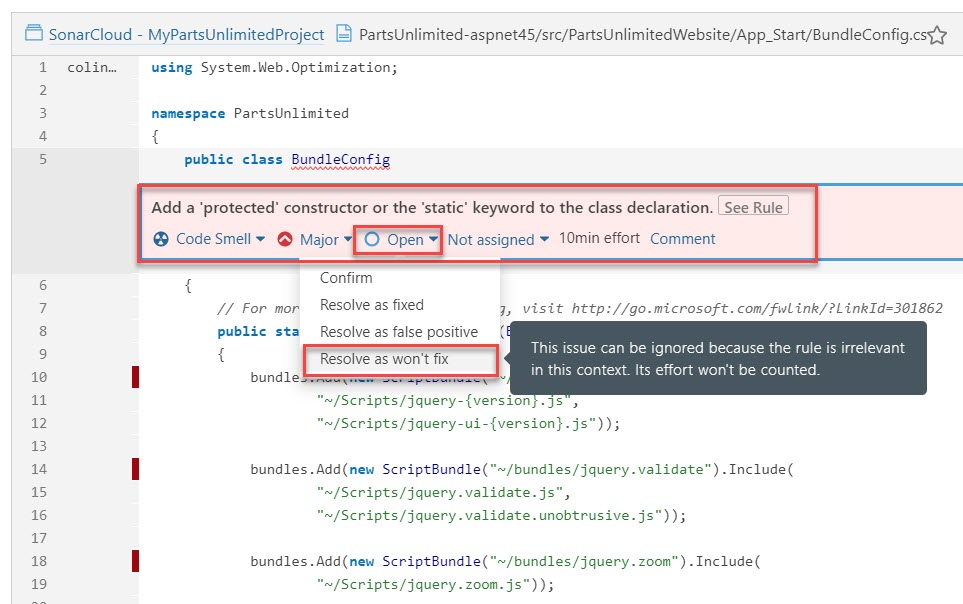
Step 33: Navigate to **Issues** tab

Based on result you can sort or filter and take immediate attention.

Click on first result.

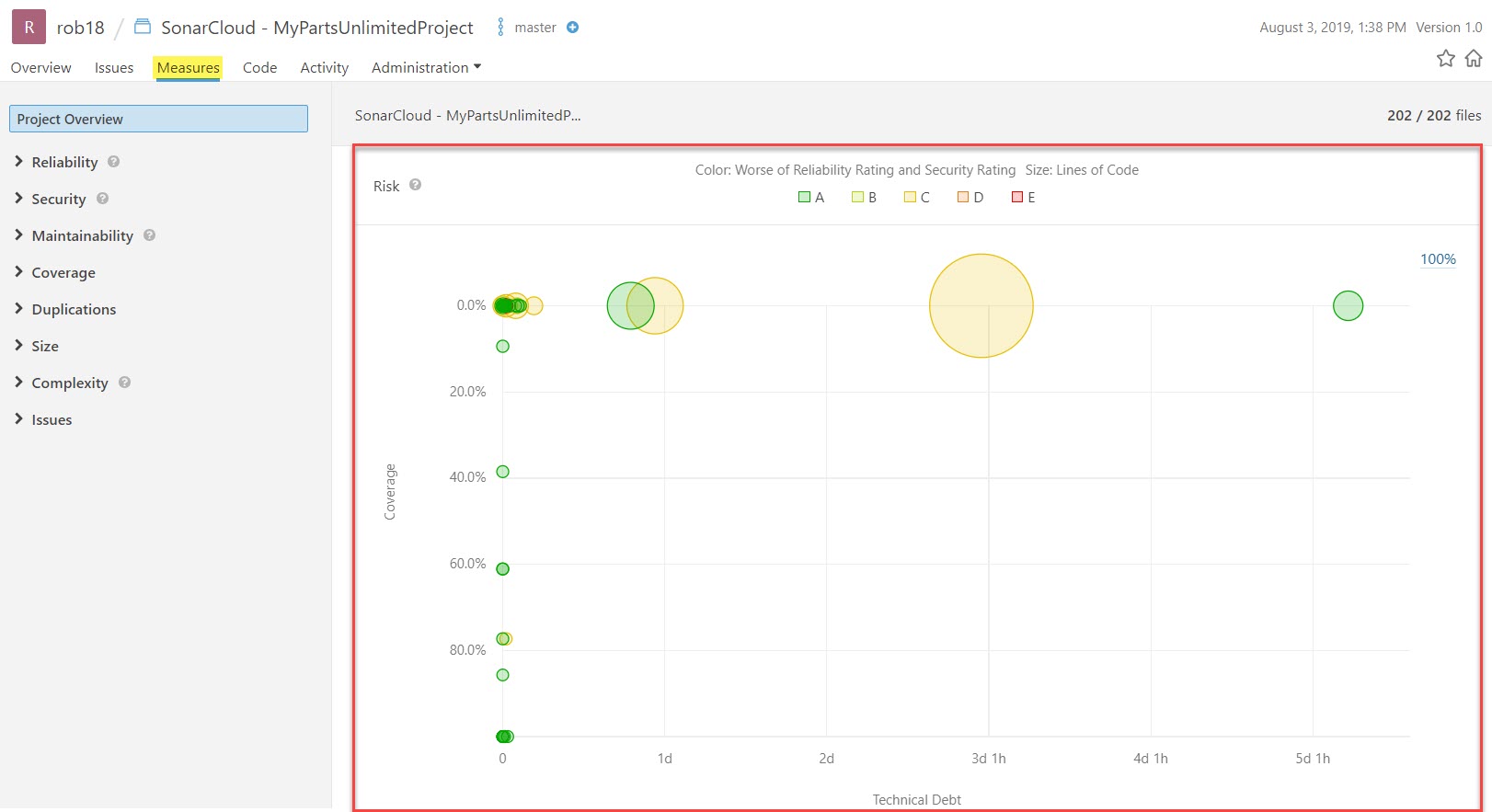


With this selection, code view provides an in-depth review of each issue. For this Select Open and Resolve as won’t fix option.



Step 34: Navigate to **Measures** tab

It will provide visualize view of issues

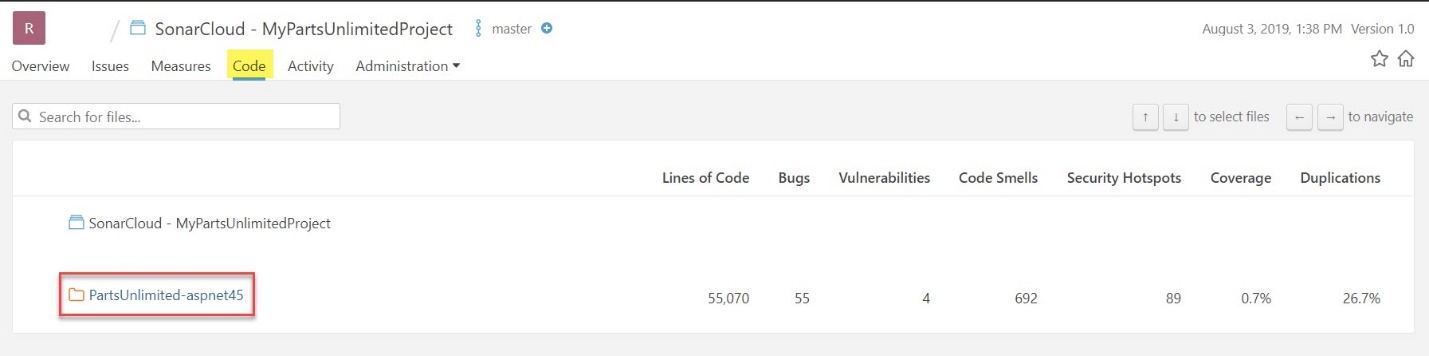


Select Reliability go get more visualize view of that.

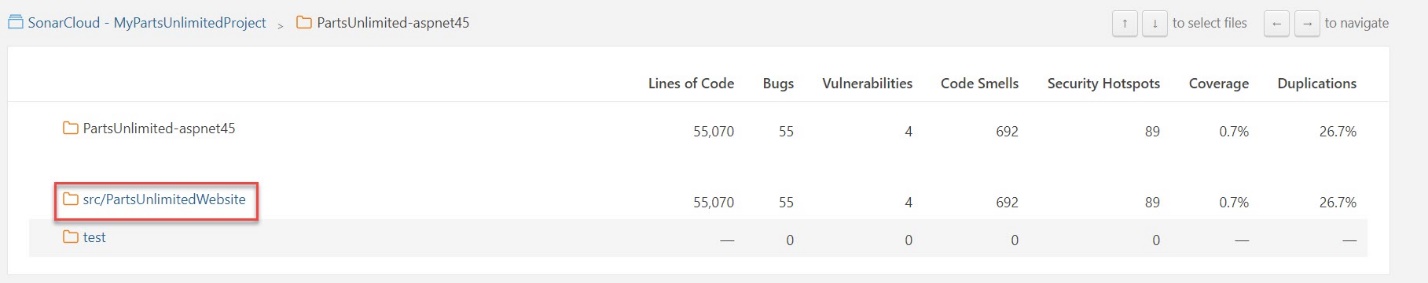


Step 35: Navigate to **Code** tab

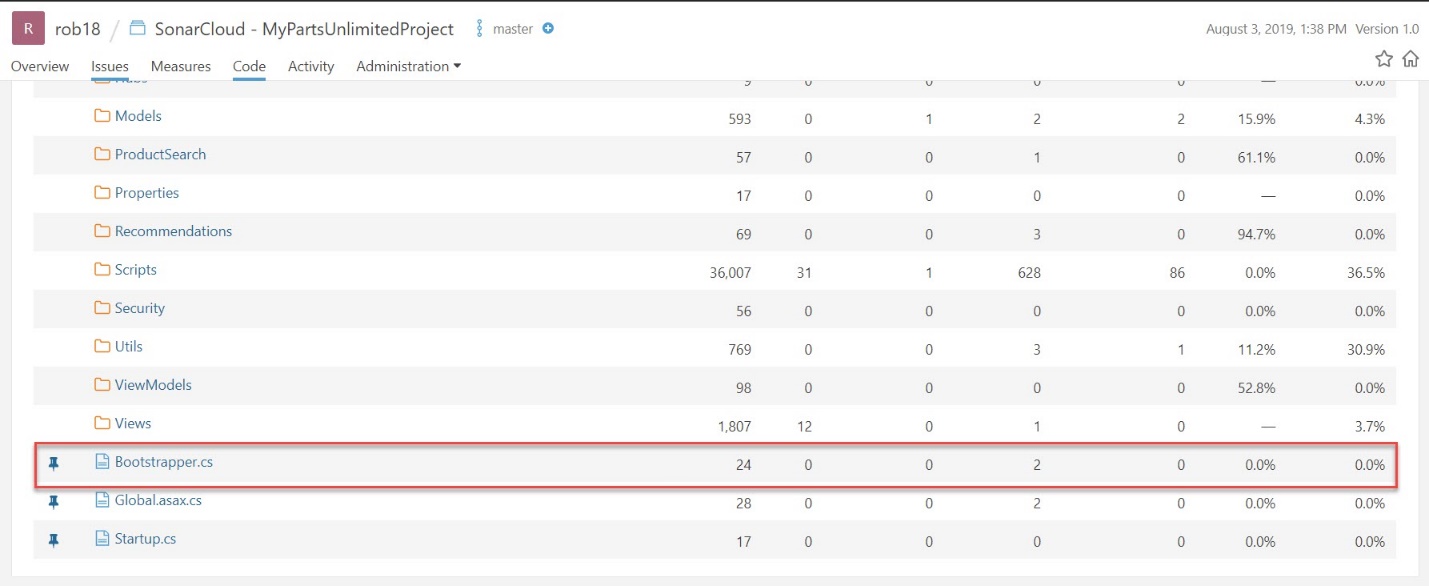
Click on Project Name Ex. PartsUnlimited-aspnet45



Click on src/PartsUnlimitedWebsite



Select file Ex. Bootstrapper.cs file



Click on See Rule to get more details

