SonarQube Merge Request Decoration in GitLab

Bundelkhand, NikeshSingh (GfK)

Exported on 04/09/2024

Table of Contents

1 Abstract

SonarQube's integration with GitLab Self-Managed allows you to maintain code quality and security in your GitLab projects.

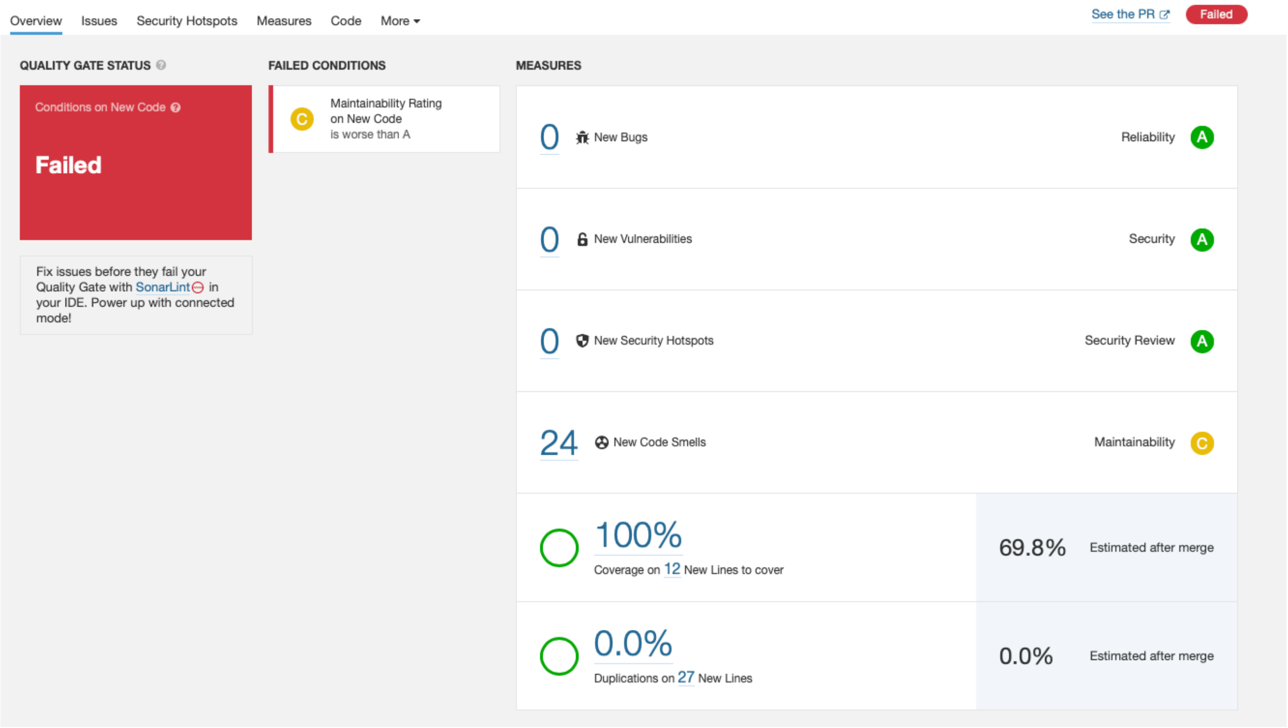
With this integration, you'll be able to :

- **Authenticate with GitLab** - Sign in to SonarQube with your GitLab credentials.
- **Import your GitLab projects** - Import your GitLab Projects into SonarQube to easily set up SonarQube projects.
- **Analyze projects with GitLab CI/CD** - Integrate analysis into your build pipeline. SonarScanners running in GitLab CI/CD jobs can automatically detect branches or merge requests being built so you don't need to specifically pass them as parameters to the scanner.
- **Report your Quality Gate status to your merge requests** - See your Quality Gate and code metric results right in GitLab so you know if it's safe to merge your changes.

In this document, we discuss on Report your Quality Gate status to your merge requests with Merge Request Decoration in GitLab.

You can see your Pull Requests in SonarQube from the Branches and Pull Requests dropdown menu of your project.

Pull Request analysis shows your Pull Request's Quality Gate and analysis in the SonarQube interface. This analysis shows new issues introduced by the Pull Request before merging with the target branch:



QUALITY GATE STATUS

Conditions on New Code

Failed

Fix issues before they fail your Quality Gate with SonarLint in your IDE. Power up with connected mode!

FAILED CONDITIONS

C Maintainability Rating on New Code is worse than A

MEASURES

0	New Bugs	Reliability	A
0	New Vulnerabilities	Security	A
0	New Security Hotspots	Security Review	A
24	New Code Smells	Maintainability	C
100%	Coverage on 12 New Lines to cover	69.8%	Estimated after merge
0.0%	Duplications on 27 New Lines	0.0%	Estimated after merge

2 Prerequisites

GitLab to be integrated with SonarQube.

SonarQube with Developer Edition and above

3 Scope

Merge Request Decoration helps developers to quickly have comments on their Static Code analysis in GitLab with links to see SonarQube Analysis.

4 Importing GitLab projects into SonarQube

Setting up the import of GitLab projects into SonarQube allows you to easily create SonarQube projects from your GitLab projects.This is the first step in adding merge request decoration.

To set up the import of GitLab projects:

1. Set your global settings
2. Add a personal access token for importing repositories

As we don't have access for global settings, Please contact BADO team here to configure GitLab details under DevOps Platform Integration in SonarQube.

Below are the details to be shared with BADO team :

1. Configuration Name - Any suitable name based on your project you want to decorate MR
2. GitLab URL - This will be GitLab API URL. As if now we have Client Platform GitLab and API URL will be https://<gitlab-url>/api/v4
3. Personal Access Token - A GitLab user account to decorate Merge Requests.

5 Merge Request Decoration

To decorate Pull Requests, a SonarQube analysis needs to be run on your code. You can find the additional parameters required for Pull Request analysis below in the **Analysis parameters** section.

5.1 Pull Request Quality Gate

A Quality Gate[1] lets you ensure you are meeting your organization's quality policy and that you can merge your pull request. The pull request uses your project Quality Gate as follows:

- **Focuses on new code** – The Pull Request quality gate only uses your project's quality gate conditions that apply to "on New Code" metrics.
- **Assigns a status** – Each Pull Request shows a quality gate status reflecting whether it Passed or Failed.

By default, Pull Request Analysis on SonarQube are deleted automatically after 30 days if no further analysis is triggered. If you want this to be updated Please contact BADO team to update Housekeeping settings in Administration tab.

5.2 Analysis Parameters

The following parameters enable Pull Request analysis.

> ⓘ Sonar Scanners can automatically detect PR parameters when running pipelines on GitLab CI/CD. Manually setting Pull Request parameters overrides automatic detection

Parameter Name	Description	for example in GitLab CI/CD (e.g.)
sonar.pullrequest.key	Unique identifier of your Pull Request. Must correspond to the key of the Pull Request in your DevOps Platform.	`sonar.pullrequest.key="$CI_MERGE_REQUEST_IID"`
sonar.pullrequest.branch	The name of the branch that contains the changes to be merged	`sonar.pullrequest.branch="$CI_MERGE_REQUEST_SOURCE_BRANCH_NAME"`

[1] https://docs.sonarqube.org/latest/user-guide/quality-gates/

Parameter Name	Description	for example in GitLab CI/CD (e.g.)
sonar.pullrequest.base	The branch into which the Pull Request will be merged.	`sonar.pullrequest.base="$CI_MERGE_REQUEST_TARGET_BRANCH_NAME"`

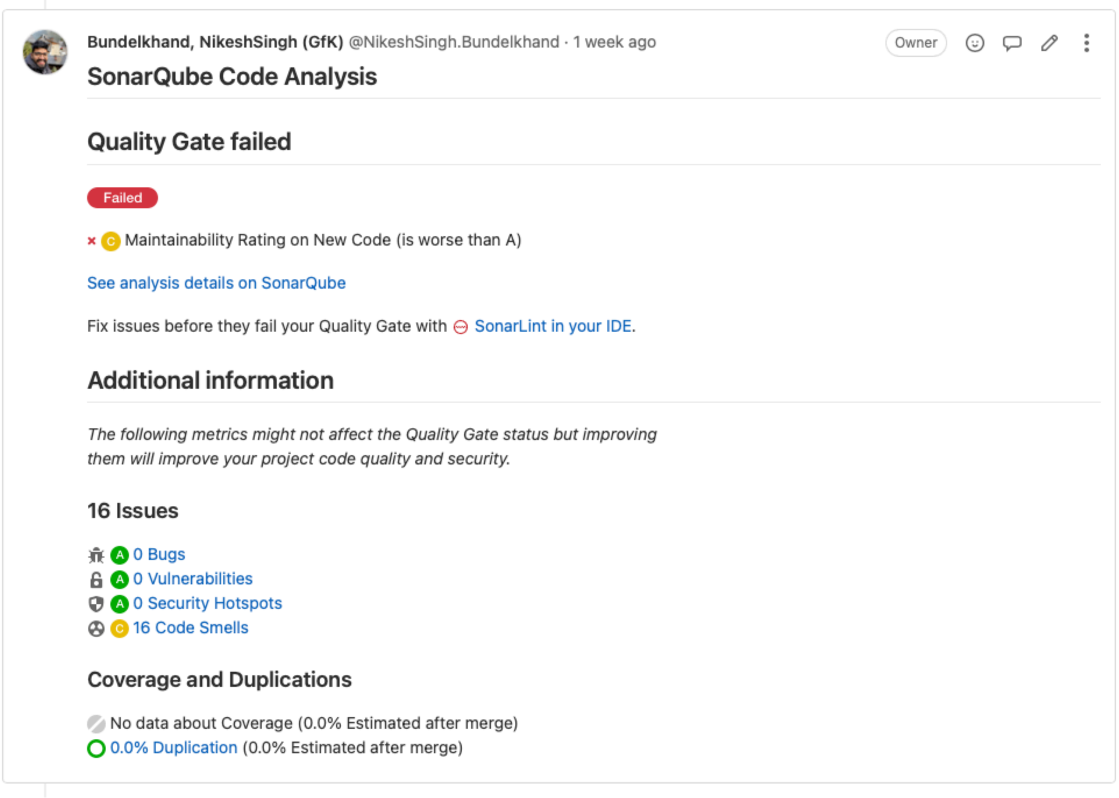
5.3 Failing the pipeline job when the SonarQube Quality Gate fails

In order for the Quality Gate to fail on the GitLab side when the Quality Gate fails on the SonarQube side, the scanner needs to wait for the SonarQube Quality Gate status.

To enable this, set the `sonar.qualitygate.wait=true` parameter in the `.gitlab-ci.yml` file.

You can set the `sonar.qualitygate.timeout` property to an amount of time (in seconds) that the scanner should wait for a report to be processed. The default is 300 seconds.

Below is sample snippet with Merge Request Decoration in GitLab :



Bundelkhand, NikeshSingh (GfK) @NikeshSingh.Bundelkhand · 1 week ago

SonarQube Code Analysis

Quality Gate failed

Failed

✕ C Maintainability Rating on New Code (is worse than A)

See analysis details on SonarQube

Fix issues before they fail your Quality Gate with SonarLint in your IDE.

Additional information

The following metrics might not affect the Quality Gate status but improving them will improve your project code quality and security.

16 Issues

A 0 Bugs
A 0 Vulnerabilities
A 0 Security Hotspots
C 16 Code Smells

Coverage and Duplications

No data about Coverage (0.0% Estimated after merge)
0.0% Duplication (0.0% Estimated after merge)

6 Links & References

PR Analysis : https://docs.sonarqube.org/latest/analysis/pull-request/

GitLab Integration : https://docs.sonarqube.org/latest/analysis/gitlab-integration/

GitLab CI/CD Configuration : https://docs.sonarqube.org/8.4/analysis/gitlab-cicd/

SonarQube : https://docs.sonarqube.org/latest/