

Hinweise zum Einrichten von Git und Sonar

Für die Projektarbeit ist die Nutzung der GitLab-Instanz unter <https://git.uibk.ac.at/> verpflichtend. Im Folgenden finden Sie Hinweise zur Konfiguration Ihres Projekts in GitLab und in SonarQube.

1 Nutzung von GIT im Allgemeinen

Im Internet gibt es zahlreiche Anleitungen zur Nutzung des Versionsmanagement GIT. Sie können Git von der Kommandozeile, oder integriert in die Softwareentwicklungsumgebung Ihrer Wahl finden: z.B.

- <https://lernprogrammieren.de/git/> (Kommandozeile)
- <https://www.ionos.at/digitalguide/websites/web-entwicklung/git-tutorial/> (Kommandozeile)
- <https://www.jetbrains.com/help/idea/set-up-a-git-repository.html> (IntelliJ)
- <https://www.vogella.com/tutorials/EclipseGit/article.html> (eclipse)
- <https://code.visualstudio.com/docs/editor/versioncontrol> (VS Code)

Bitte machen Sie sich ggf. darüber mit den grundlegenden Funktionen von Git vertraut. Weiteres Material und Links dazu finden Sie auch in OLAT.

2 Einrichten des Projekts für SonarQube

Loggen Sie sich mit Ihrer c-Kennung unter <https://qe-sonarqube.uibk.ac.at/> ein. Alle Ihre Teammitglieder sollten dort ein (leeres) Projekt mit dem Namen SEPSS22_GxTy finden. Z.B.

Eine Person im Team muss Ihr Team-GIT-Projekt nun konfigurieren, damit es von Sonar analysiert werden kann.

1. Generierung eines persönlichen Access-Tokens:
Wählen Sie in Sonar oben rechts ihre unter Ihren Initialen das Feld „My Account“.

Wechseln Sie anschließend auf den Tab „Security“

Hinweise für Git und Sonar

Generieren Sie ein neues Access-Token
(am besten mit einem sprechenden
Namen)

Tokens

If you want to enforce security by not providing credentials of a real SonarQube user to run your code scan or to invoke web services, you can provide a User Token as a replacement of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

Generate Tokens

AnalyseToken für GitLabUIBK

Generate

Kopieren Sie sich das angezeigte Token. Achtung, Sie können das Token nicht wieder einsehen. Sichern Sie es deshalb zunächst in eine eigene Datei

**2. Einrichten des GitLab-Projekts mit dem Access Token**

Wechseln Sie zurück in Ihr GitLab-Projekt und wählen Sie unter „Settings“ -> „CI/CD“ -> „Variables“ -> „Expand“. Sie sollten dann folgenden Eintrag sehen:

Variables

Variables store information, like passwords and secret keys, that you can use in job scripts. Learn more.

Variables can be:

- Protected: Only exposed to protected branches or tags.
- Masked: Hidden in job logs. Must match masking requirements. Learn more.

Type	↑ Key	Value	Protected	Masked	Environments
There are no variables yet.					

Add variable

Fügen Sie bitte folgende Variablen hinzu:

Key	Value
SONAR_HOST_URL	https://qe-sonarqube.uibk.ac.at
SONAR_PROJECT_KEY	SEPPSSS22... (Der Name Ihres Sonar Projekts)
SONAR_TOKEN	6791... (das Access-Token, dass Sie im vorherigen Schritt generiert haben)

Sie sollten anschließend untenstehende Tabelle sehen: Mit „Reveal Values“ können Sie ggf. die Werte nochmals prüfen:

Type	↑ Key	Value	Protected	Masked	Environments
Variable	SONAR_HOST_URL	*****	×	×	All (default)
Variable	SONAR_PROJECT_KEY	*****	×	×	All (default)
Variable	SONAR_TOKEN	*****	×	×	All (default)

Add variable Reveal values

3. Testen der Einstellungen

Checken Sie das Projekt in einem Entwicklungswerkzeug Ihrer Wahl aus, ändern eine Kleinigkeit in einer Datei und comitten Sie das Ergebnis.

Anschließend sollten Sie in Ihrem GitLab-Projekt unter dem Menu-Eintrag „CI/CD“ -> „Pipelines“ einen ersten Eintrag sehen. Die Pipeline ist so konfiguriert, dass sie zunächst die JUnit-Tests des Projekts ausführt und (sofern die Tests erfolgreich abgeschlossen wurden) anschließend die Ergebnisse mit Sonar analysiert.

Hinweise für Git und Sonar

The screenshot shows a pipeline status page. On the left, there's a sidebar with 'CI/CD' and 'Pipelines' selected. The main area shows a single job in progress, labeled '#51932 latest'. The job is triggered by a commit from 'master' (commit hash: dd1768f2) by a user named 'Just trying'. The pipeline has two stages: 'Stages' and 'Duration'. The duration is shown as 'In progress'.

Im Idealfall sollten Sie nach einigen Minuten zwei grüne Haken in der Spalte „Stages“ sehen:

This screenshot shows the same pipeline interface after the job has completed successfully. The job status is now 'passed' with a green checkmark icon. The duration is listed as '00:02:54' and '48 seconds ago'. The stages are now both marked as 'passed' with green checkmarks.

In SonarQube sollten Sie in Ihrem Projekt nun die Analyseergebnisse sehen:

The screenshot displays a SonarQube dashboard for a project. It includes a 'Passed' summary box and an 'Overall Code' section with the following data:

- Bugs:** 1 (Reliability: C)
- Vulnerabilities:** 1 (Security: D)
- Security Hotspots:** 2 (0.0% Reviewed, Security Review: E)
- Maintainability:** A
- Coverage:** 77.0% (Coverage on 181 Lines to cover)
- Unit Tests:** 14
- Duplications:** 0.0% (Duplications on 878 Lines)
- Duplicated Blocks:** 0

Prüfen Sie in SonarQube die Fehlerhinweise und „Code Smells“. Sie werden einige „falsch-positive“ Hinweise finden. Z.B.

A tooltip message appears over a dropdown menu for an issue. The message reads: "This issue can be suppressed as it was not raised accurately. Please report false-positives to the SonarSource Community!"

Sie können diese entsprechend markieren.

Andere Hinweise sind durchaus relevant. Bitte stellen Sie in Ihrem Source Code sicher, dass diese Fehler korrigiert werden.

A tooltip message appears over a dropdown menu for a code smell. The message reads: "Replace this if-then-else statement by a single return statement. Why is this an issue? last year L172 clumsy".

3 Weitere Hinweise

Die Funktionen der GitLab-Pipeline werden über die Datei `.gitlab-ci.yml` gesteuert. Ändern Sie die Datei bitte nur dann, wenn Sie wissen, was Sie tun.

Momentan ist die Pipeline so konfiguriert, dass sie bei jedem Commit, egal in welchem Branch aktiviert wird. Wenn Sie im Team ggf. an mehreren GIT-Branches gleichzeitig arbeiten, möchten Sie

Hinweise für Git und Sonar

sicherstellen, dass SonarQube nur die Änderungen im „main“ oder „development“-Branch übernimmt. Ergänzen Sie dann ggf. für einen Stage

```
only:  
  - master  
  - development
```