

# 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://lerneprogrammieren.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.

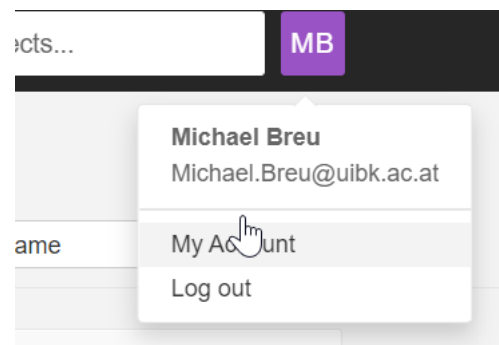
## 2 Einrichten des Projekts für SonarQube

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 SEPSSS22\_GxTy finden. Z.B.

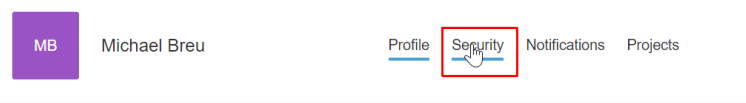


Eine Person im Team muss Ihr Team-GIT-Projekt nun konfigurieren, damit es von Sonar analysiert werden kann.

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

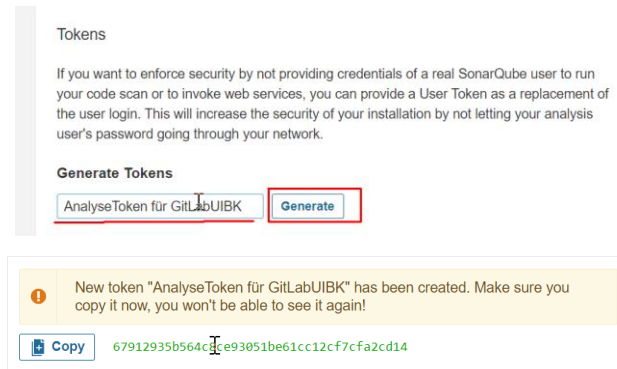


Wechseln Sie anschließend auf den Tab „Security“



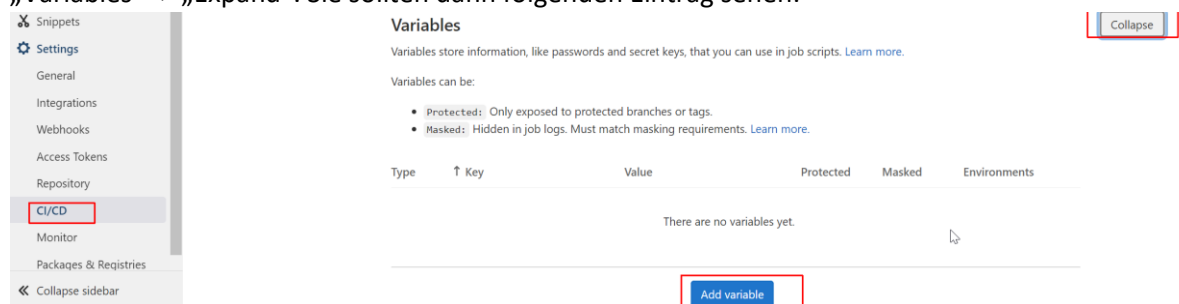
## Hinweise für Git und Sonar

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



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

- 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:



Fügen Sie bitte folgende Variablen hinzu:

Key	Value
<b>SONAR_HOST_URL</b>	<a href="https://qe-sonarqube.uibk.ac.at">https://qe-sonarqube.uibk.ac.at</a>
<b>SONAR_PROJECT_KEY</b>	SEPSSS22... (Der Name Ihres Sonar Projekts)
<b>SONAR_TOKEN</b>	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)	

- 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

CI/CD	Status	Pipeline	Triggerer	Commit	Stages	Duration
Pipelines Editor Jobs	pending	#51932 latest		master -> dd1768f2 Just trying		In progress

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

CI/CD	Status	Pipeline	Triggerer	Commit	Stages	Duration
Pipelines Editor Jobs	passed	#51932 latest		master -> dd1768f2 Just trying	SonarCheck: passed	00:02:54 48 seconds ago

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

<b>Passed</b> All conditions passed.	<b>New Code</b> Since October 5, 2021 Started 3 hours ago	<b>Overall Code</b>
	1 Bug	Reliability <span>C</span>
	1 Vulnerabilities	Security <span>D</span>
	2 Security Hotspots <span>C</span>	0.0% Reviewed <span>E</span>
	2h 38min Debt	29 Code Smells <span>A</span>
	77.0% Coverage on 181 Lines to cover	14 Unit Tests
	0.0% Duplications on 878 Lines	0 Duplicated Blocks

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

Add a <title> tag to this page. Why is this an issue?

Bug Major Open Not assigned Comment

Confirm  
Resolve as fixed  
Resolve as false positive  
Resolve as won't fix

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.

```
if (!Objects.equals(this.username, other.username)) {
```

Replace this if-then-else statement by a single return statement. Why is this an issue?

Code Smell Minor Open Not assigned 2min effort Comment

```
    return false;
}
return true;
}
```

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
```