

# Übungen zur Vorlesung

## Softwaretechnik – Wintersemester 2021/2022

### Übungsblatt 06

Abgabe am 08.12.

## 1 Softwareprojekt-Setup und Integration

In dieser Aufgabe sollen Sie ein Projekt mit bekannten Tools aufsetzen. Dafür müssen Sie gegebenenfalls recherchieren, wie Sie diese Tools (zusammen) verwenden. Anschließend sollen Sie ein neues Modul in die Software integrieren. Der Source-Code ist aus der bekannten Klasse `Navigation.java`, aber auf mehrere Klassen aufgeteilt. Geben Sie für dieses Blatt ein Zip-Paket mit Ihrem finalen Projekt (inklusive Tests) ab.

1. Erstellen Sie ein privates Repository auf GitLab.com.
2. Clonen Sie das Repository lokal.
3. Erstellen Sie ein neues Gradle-Projekt (Typ „application“, Sprache Java, Keine Subprojekte, mit JUnit) in dem lokalen Repository. Eventuell müssen Sie Gradle zuvor installieren.
4. Installieren Sie nun (als Gradle-Plugins)
  - Checkstyle (mit Google Style-Guide)
  - Spotbugs
  - Error-Prone
  - Google-Java-Format
5. Kopieren Sie nun die Dateien unter `source_files` in Ihr Projekt (in das Verzeichnis, in dem schon die generierte Datei `App.java` liegt). Existierende generierte Dateien können Sie ersetzen. Eventuell müssen Sie die Packages der Klassen anpassen.
6. Führen Sie die Code-Analyse-Tools aus (z. B. mit `./gradlew check`) und verbessern Sie gegebenenfalls den Code bis es keine Fehlermeldungen mehr gibt.
7. Setzen Sie eine CI-Pipeline in GitLab auf. Verwenden Sie dafür einen shared runner. Die Pipeline sollte folgende Schritte enthalten:
  - a) Build (ohne checks)
  - b) statische Checks
  - c) Tests

Hinweis: Eine erste Anleitung finden Sie z.B. hier: [https://docs.gitlab.com/ee/ci/quick\\_start/](https://docs.gitlab.com/ee/ci/quick_start/)

8. Pushen Sie Ihre bisherigen Änderungen und stellen Sie sicher, dass die CI-Pipeline funktioniert wie erwartet.

9. Erstellen Sie jetzt einen neuen Branch auf dem Sie den Shortest-Paths Algorithmus hinzufügen. Die Implementierung dafür finden Sie in `algorithm`. Kopieren Sie die Klasse in das entsprechende Verzeichnis. Passen Sie auch den Code in `App.java` an, sodass die neue Klasse verwendet wird. Schreiben Sie Unit- und Integrationstests für den Algorithmus und pushen Sie die Änderungen.
10. Mergen Sie den Branch anschließend zurück in `main`.

**Allgemeine Hinweise:**

- Abgabe der Lösung als PDF im Uni2Work (Code bitte als separate Dateien abgeben).
- Deadline der Abgabe ist am **Mittwoch, den 08.12. um 15:00 Uhr**.