

1. In der nächsten Aufgabe werden wir *Maven* für die Build-Automatisierung nutzen. Treffen Sie dafür die folgenden Vorbereitungen. Legen Sie ein neues Projekt **BuchRedesignMaven** an. Nutzen Sie dafür die Verzeichnisstruktur für *Maven*-Projekte. Kopieren Sie die erforderlichen Artefakte aus dem bisherigen Projekt in die neue Projektstruktur (inkl. der bisher erstellten Tests). Speichern Sie das Projekt in einem neuen *Gitea*-Repository.¹
2. Automatisieren Sie das Projekt **BuchRedesign** mit Hilfe von *Maven*.
Es sollen alle bisher für das Projekt **BuchRedesign** erstellen Unit-Tests ausgeführt werden. Die Testergebnisse sollen in einem Report zusammengefasst werden. Die Code-Überdeckung der Testfälle soll mit *JaCoCo* gemessen werden. Für die statische Code-Analyse sollen *Checkstyle*, *PMD* und *Spotbugs* mit der jeweiligen Standardeinstellung eingesetzt werden. Zudem soll eine Dokumentationsextraktion mit *JavaDoc* erfolgen. Schreiben Sie dafür eine geeignete **pom.xml**!
3. Prüfen Sie die **pom.xml**! Erzeugen Sie die Testdokumentation mit `> mvn clean test jacoco:report site`. Öffnen Sie die Projektdokumentation in einem Browser und prüfen Sie die Vollständigkeit der *Reports*. Welche Testüberdeckung wurde erreicht? Erzeugen Sie dann eine **jar**-Datei für das Projekt **BuchRedesign** mit `> mvn package`.

Hinweise:

Für PMD können unterschiedliche Regelsätze eingestellt werden. Anbei findet sich ein Beispiel.

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-pmd-plugin</artifactId>
  <version>3.10.0</version>
  <configuration>
    <rulesets>
      <ruleset>/rulesets/java/maven-pmd-plugin-default.xml</ruleset>
    </rulesets>
  </configuration>
</plugin>
```

¹Dazu legen Sie in *Gitea* ein neues *Repository* an. Klonen Sie das (leere) *Repository* und speichern Sie dort die neue Projektstruktur. Danach schreiben Sie die Änderungen in *Gitea* fest (**add**, **commit**, **push**).

Die Konfiguration von *JaCoCo* ist komplexer. Anbei finden Sie den entsprechenden Ausschnitt aus der *pom.xml*.

```
<build>
  <plugins>
    <plugin>
      <groupId>org.jacoco</groupId>
      <artifactId>jacoco-maven-plugin</artifactId>
      <version>0.8.5</version>
      <executions>
        <execution>
          <id>default-prepare-agent</id>
          <goals><goal>prepare-agent</goal></goals>
        </execution>
        <execution>
          <id>default-report</id>
          <goals><goal>report</goal></goals>
        </execution>
        <execution>
          <id>default-check</id>
          <goals>
            <goal>check</goal>
          </goals>
          <configuration>
            <rules>
              <rule>
                <element>BUNDLE</element>
                <limits>
                  <limit>
                    <counter>COMPLEXITY</counter>
                    <value>COVEREDRATIO</value>
                    <minimum>0.0</minimum>
                  </limit>
                </limits>
              </rule>
            </rules>
          </configuration>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>
```

Hinweis: Um Warnung bezüglich eines undefinierten Zeichensatzes zu vermeiden, stellen Sie innerhalb von `<project>` die folgenden

```
<properties>  
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
</properties>
```

ein.