Praktikum zu Fortgeschrittenes Web-Engineering, Wintersemester 2020/2021

Praktikum 5

Im Rahmen des Praktikums entwickeln wir eine Web-Anwendung, die wir Schritt für Schritt mit Anforderungen, Funktionen und Technologien erweitern.

Im fünften Praktikum nutzen wir Maven, um unser Spring-Backend-Projekt um ein paar Funktionalitäten zu erweitern.

Stellen Sie sämtliche Ergebnisse und Änderungen in Ihrem Git-Repository zur Verfügung.

Aufgabe 1: Build-Checks

Wir erweitern unseren *Build-Prozess* (unser Spring-Backend-Projekt ist bereits ein lauffähiges Maven-Projekt) nun um zwei Checks, die die Qualität unseres Codes erhöhen sollen:

1. Integrieren Sie den Maven Enforcer, um zu sichern, dass der Build-Prozess nur in einer kompatiblen Entwicklungsumgebung ausgeführt wird. Legen Sie mindestens folgende Regeln fest:

   - Es soll Maven in der Version 3.5 oder größer vorliegen.
   - Es soll Java in der Version 11 vorliegen (bzw. 11 und größer, falls Sie in Ihrem Team unterschiedliche Java-Versionen nutzen).

   Binden Sie die Ausführung des entsprechenden Plugin-Goals an die `validate`-Phase.

2. Zusätzlich zu dem eigentlichen Projektartefakt soll der Build-Prozess ein weiteres Artefakt erzeugen, welches den Quellcode des Projektes enthält. Dies ermöglicht es anderen EntwicklerInnen, Ihren Code einfacher zu debuggen. Nutzen Sie dafür das Maven Source Plugin. Binden Sie die Ausführung des entsprechenden Plugin-Goals an die `package`-Phase.

Aufgabe 2: Projekt-Webseite und -Reports

Nutzen Sie Maven zur automatischen Erzeugung einer Projekt-Webseite. Zusätzlich zu den Standardinhalten, die Maven hier standardmäßig erzeugt, soll die Projekt-Webseite folgende Reports enthalten:

1. Binden Sie das SpotBugs Maven Plugin ein. SpotBugs nutzt statische Code-Analyse, um Ihren Quellcode auf mögliche Fehler und Code-Smells zu untersuchen. Integrieren Sie das Plugin so, dass die Ergebnisse der Analyse automatisch in die Projekt-Webseite eingebunden werden.

2. Integrieren Sie zudem das Dependency-Check Maven Plugin. Dieses Plugin untersucht die Abhängigkeiten Ihres Projektes nach möglichen Sicherheitslücken und Schwachstellen. Dazu nutzt das Plugin entsprechende Datenbanken wie z.B. die National Vulnerability Database des NIST. Integrieren Sie auch dieses Plugin so, dass die Ergebnisse der Analyse automatisch in die Projekt-Webseite eingebunden werden.

Hinweis: Sollte es beim Erzeugen der Projekt-Webseite zu Problemen kommen (z.B. `java.lang.NoClassDefFoundError`), so kann es notwendig sein, die verwendete Version des Maven-Site-Plugins zu aktualisieren. Ergänzen Sie zu diesem Zweck Folgendes im `build`-Abschnitt Ihrer `pom.xml`:

```
<plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-site-plugin</artifactId>
    <version>3.7.1</version>
</plugin>
```