

Projektstruktur

Passt eure Projektstruktur bitte an folgendes Muster an:

- erstesProjektVV (oder wie das bei euch eben heißt)
- messaging
 - customerservice
 - src\main\java\de.thro.inf.vv.messaging.customerservice
 - build.gradle
 - ...
 - accountingservice
 - ...
- .gitignore
- .gitlab-ci.yml
- README.md

Der "erstesProjektVV"-Ordner steht hier mal als Beispiel für den Ordner, in dem sich jeweils euer erstes Projekt befindet. Habt ihr **keinen** solchen Ordner, legt euch den bitte an und schiebt mit `git mv` (sucht euch im Internet raus wie das genau geht) die Dateien des ersten Projektes in den Ordner. Falls ihr so einen Ordner habt, müsst ihr das nicht machen.

Legt euch dann (am besten lokal) einen Ordner "messaging" (oder so ähnlich) an, der sich auf der selben Ebene wie der "erstesProjektVV"-Ordner, die `.gitignore` und die `.gitlab-ci.yml` befindet. In diesem Ordner erstellt ihr dann **3 separate Projekte** in IntelliJ, jeweils eins für die einzelnen Services in der Aufgabenstellung. Achtet hier bitte auch darauf jeweils darauf, im `src\main\java`-Unterordner eine ordentliche package-Struktur einzuhalten. Oben ist ein Beispiel abgebildet, wie diese aussehen könnte. Tipp: ihr könnt die Struktur beim Projekterstellen gleich mit angeben. Später kommt dann noch der Ordner für die dritte Projektaufgabe auf die oberste Ebene (nur schon mal als Ausblick).

Jedes dieser Projekte besitzt dementsprechend ein eigenes `build.gradle`-File. Diese Files jeweils wie in der ersten Aufgabe auch mit den Tasks für jacoco, SonarQube etc füllen. Achtet darauf, jedem Projekt einen eigenen `sonar.projectKey` zuzuweisen, dann könnt ihr die in der SonarQube-Oberfläche später einfach auseinanderhalten.

.gitlab-ci.yml

Die `.gitlab-ci.yml` müsst ihr auch noch anpassen, damit diese mit den neuen Projekten umgehen kann:

1. Ersetzt die Verzeichnisse, in die ihr mittels `cd` wechselt, passend für die neuen Projekte. Für obiges Beispiel: `cd ./messaging/customerservice`.
2. Handelt den jeweiligen Build + Test für die 3 Projekte ab. Dazu müsst ihr für jedes Projekt jeweils eine build- und eine test-Stage definieren. Wie: das könnt ihr euch aus dem aktuellen Inhalt der `.gitlab-ci.yml` ableiten.
3. Die Stages für build und test des ersten Projektes kommentiert ihr bitte mit `#` aus. Lasst sie aber noch drin!