



Inhalt

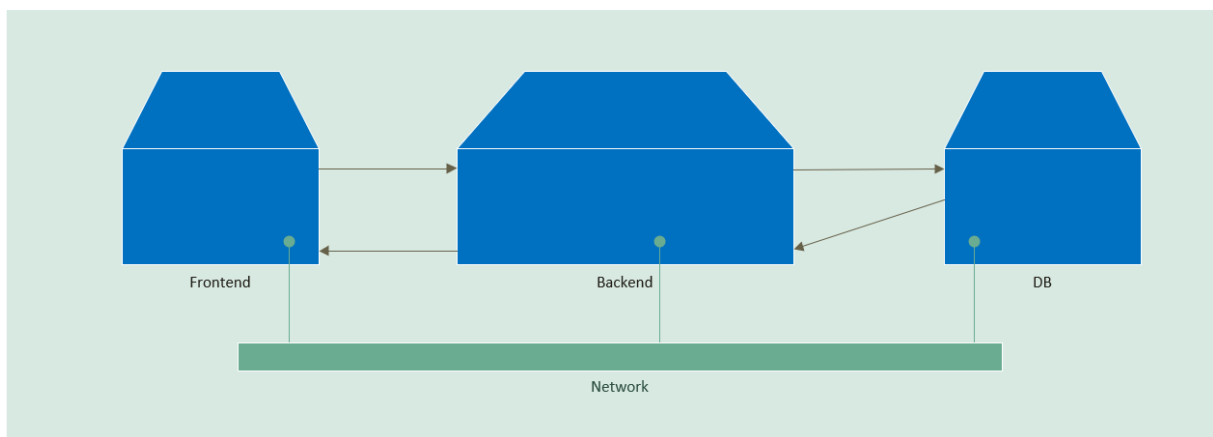
Planung.....	3
Grobe Architektur.....	3
Testkonzept.....	3
Introduction.....	3
Test Item	3
Features to be tested.....	3
Approach	3
Test Deliverables.....	4
Testing Tasks	4
Environmental Needs	4
Reflexion TDD & Code Review	4
TDD	4
Code Review	4

Planung

Unser Projekt haben wir vor allem mit den Anforderungen im Hintergrund geplant. Dafür haben wir uns für ein Projekt entschieden, welches schon existierte, jedoch das Testing vernachlässigt wurde. Somit stand im Fokus unserer Planung dass wir das Frontend redesignen, das Backend Anpassen mit den Unit tests und die Testcoverage verbessern und Anzeigen. Für die Anzeige der Coverage haben wir uns dabei gedacht das wir dabei auf einen automatisierten Report in der Github Pipeline setzen.

Grobe Architektur

Die Applikation selbst besteht aus einer Datenbank, einem Frontend und einem Backend. Zusätzlich Verwenden wir für das generieren der Testcoverage die Infrastruktur von Github und Pipelines. Unser Backend ist eine Spring Boot Applikation. Das Backend benutzt JWT um die Sicherheit zu gewährleisten. Für das Frontend wird React als Framework verwendet. Bei der Datenbank handelt es sich um eine postgres DB.



Testkonzept

Introduction

Bei der Applikation handelt es sich um eine Plattform auf welcher man sich registrieren kann und ein kleines Profil erstellen kann

Test Item

Elemente welche für das Testing in Frage kommen wären das Backend und das Frontend.

Features to be tested

Im Projekt wird nur das Backend getestet.

Getestet wird die API in bezug auf die Anfrage auf das Benutzerprofil. Dabei werden die Standard CRUD Methoden abgedeckt. Zudem werden die Einzelnen Service Methoden im Backend auch noch getestet. Auch wird die authority und die authentication des Benutzers getestet.

Approach

Während der Entwicklung wird vom Entwickler für neue Features Unit und Component Tests geschrieben. Vorgegangen wird nicht nach TDD.

Test Deliverables

Für das Testing wird zum einen JUnit benutzt für die Unit Tests sowohl auch Postman für Component Tests und Tests bezüglich der Sicherheit

Testing Tasks

Getestet wird vor allem in der Unit Test Stufe und der Component Test Stufe

Environmental Needs

Für das Testing wird das Laufende Backend gebraucht. Dafür sollte dies am besten als Projekt in IntelliJ laufen. Dazu wird auch die Datenbank gebraucht, dass das backend aufstarten kann. Danach können Unit Tests in IntelliJ ausgeführt werden. Zusätzlich wird die Software Postman gebraucht.

Reflexion TDD & Code Review

TDD

TDD würden wir persönlich nicht gerne Verwenden. Wir haben es im Rahmen dieses Projektes probiert. Jedoch sind wir schnell zur Erkenntnis gekommen das TDD nichts ist, was wir weitergehend Verwenden wollen. Den Ansatz hinter der Idee zuerst Tests zu schreiben und dann zu entwickeln ist nachvollziehbar. Jedoch sehen wir das Konfliktpotenzial von zu starkem Fokus auf das Erreichen eines Erfolgreichen Tests. Während der Entwicklung stellt man vlt. schon selbst nochmals neue Edge-cases fest. Aus dem Gefühl heraus würde das beim TDD untergehen da schon im Vorhinein edge-cases definiert sein müssten.

Code Review

Code Reviews sind eine gute Form der Doppelten Prüfung was jemand geschrieben hat. Oft kann es sein das etwas vergessen geht oder ein doofer Fehler passiert ist. Eigene Fehler sieht man ja bekanntlich schlechter als jemand außenstehendes. Ausserdem kommt somit nochmals eine Meinung dazu und allenfalls können auch sonstige Verbesserungen vorgeschlagen werden. In unserem Fall nutzten wir genau dafür die Code reviews. Reviews geben auch eine Möglichkeit neues zu lernen, neue Ansatzweisen anzugehen oder auch zu lernen mit Kritik umzugehen und diese auch anzuwenden.