Abstract

Die Bachelorarbeit handelt von „Testautomatisierung im Rahmen von Test-Driven-Development“.

Aufgrund immer komplexer werdenden Prozessen und Entwicklungsansätzen in heutigen IT-Unternehmen, die den stetigen Wandel und Anforderungen – auch im Rahmen der Digitalisierung – der Kunden gerecht werden müssen, ergeben sich stets neue Ansätze der Softwareentwicklung. Dabei inbegriffen sind Themen im Bezug zur Kommunikation, Anforderungsmanagement, Qualitätssicherung im Rahmen von Softwaretests sowie viele weitere Faktoren, welche je nach Szenario und Entwicklungsansatz mit einer differenzierten Strategie betrachtet werden.

Besonders im Bereich der Softwarequalität gelangen Tests durch das Komplexitätsniveau oftmals unbedacht in den Hintergrund und werden ebenso aufgrund diverse Ressourcenfaktoren vernachlässigt.

Somit ist Ziel dieser Arbeit eine Übersicht der möglichen Test-Verfahren im Rahmen des Test-Driven-Development (kurz: TDD) darzustellen. Dazu gehört das Allgemeine Verständnis inklusive die Anwendung und Etablierung in Kombination detaillierterer Verfahren, wie zum Beispiel Feature-TDD, Acceptance-TDD und Behaviour-TDD.

Folglich lässt sich die Arbeit zum einen in einen Theorieteil, der oben genanntes vermitteln soll, als auch in einen Praxisteil, der die Etablierung solcher Szenarien mit geeigneter Technologien vorstellen soll, gliedern.

Besonders in der Praxis ist jede Anwendung individuell und besitzt Eigenschaften hinsichtlich Softwarearchitektur, Anforderungen, Qualität, Ist-Zustand, Soll-Zustand etc., die beachtet werden müssen. Dabei soll im Rahmen dieser Arbeit ein Vergleich zwischen zwei Anwendungsszenarien vorgestellt werden:

* Eine Alt-Anwendung, die bereits seit längeren existent ist
* Eine Neu-Anwendung, die auf aktuelleren Technologien und Design basiert

Dabei muss beachtet werden, dass Alt-Anwendungen oftmals bezüglich der Architektur und Softwaredesign verschachtelter und ebenso unübersichtlicher sind. Dies erzwingt eine alternative Herangehensweise bei der Etablierung von TDD und dazugehöriger Software im Vergleich zu Neu-Anwendungen, die durch spezielle Schichtentrennungen, Softwaredesigns und Technologien eine Etablierung vereinfachen. Auf beide Szenarien wird im Praxisteil eingegangen.

Für die Etablierung von TDD wird die formale Sprache „Gherkin“ in Kombination mit „Cucumber“ verwendet. Die vorgestellte Alt-Anwendung basiert auf Google-Web-Toolkit (kurz: GWT) und die Neu-Anwendung auf Spring Boot in Kombination mit Vue.js.