



TESTAUTOMATISIERUNG

- MIT SELENIUM -

Fakultät für Informatik und Mathematik
der Hochschule München

Seminararbeit

vorgelegt von

Matthias Karl

im Dezember 2014

Prüferin: Prof. Dr. Ulrike Hammerschall

Eidesstattliche Erklärung

Hiermit versichere ich, die vorliegende Studienarbeit selbstständig und nur unter Verwendung der von mir angegebenen Quellen und Hilfsmittel verfasst zu haben. Sowohl inhaltlich als auch wörtlich entnommene Inhalte wurden als solche kenntlich gemacht. Die Arbeit hat in dieser oder vergleichbarer Form noch keinem anderem Prüfungsgremium vorgelegen.

Datum: _____ Unterschrift: _____

Zusammenfassung / Abstract

Testautomatisierung ...

Inhaltsverzeichnis

Eidesstattliche Erklärung	I
Zusammenfassung / Abstract	II
1 Einleitung	1
2 Grundlagen	2
2.1 Softwarequalität	2
2.2 Softwaretest	3
2.3 Testprozess	3

1 Einleitung

Software hat in der heutigen Zeit eine hohe Verbreitung gefunden. Softwaresysteme werden immer wichtiger. Sowohl für Unternehmen als auch für jeden einzelnen persönlich. Die Anforderungen an moderne Software steigen ständig. Die Systeme werden immer größer und komplexer. Das hat zur Folge, dass auch die Anforderungen an die Qualität der Software immer weiter Steigen und wichtiger werden. Fehler in Software verursachen immer wieder einen hohen finanziellen Schaden und können im schlimmsten Fall sogar Menschenleben kosten. Diese Probleme werden immer gravierender wenn mit der Wichtigkeit, Komplexität und Größe von Software nicht auch gleichzeitig die Qualität steigt. [Bur03]

Softwaretests sind ein weit verbreitetes Mittel um die Qualität einer Software zu überprüfen und sicherzustellen. Ein weit verbreitetes Mittel um die Qualität von Software sicherzustellen ist das Testen. Komplexere und größere Software bedeutet daher auch gleichzeitig einen steigenden Testaufwand.

Qualitätssicherungsmaßnahmen wie beispielsweise das Testen machen jetzt schon einen Großteil der Kosten in Softwareprojekten aus. Studien haben gezeigt, dass das Testen für 50% und mehr der gesamten Projektkosten verantwortlich ist. [RW06] Es ist daher nicht verwunderlich, dass es immer wieder versuche gibt diese Kosten zu reduzieren ohne dabei den angestrebten Qualitätsstandart zu reduzieren. Einer der Wege die dafür vorgeschlagen wurden ist das automatisieren von Testfällen [Har00] Im Laufe der Jahre hat die Testautomatisierung immer mehr an Bedeutung gewonnen. Heute ist sie bereits fester Bestandteil von Bereichen wie Continuous Delivery und Continuous Integration.

Diese Arbeit befasst sich daher mit der Automatisierung von Testfällen. Die Arbeit soll einen Überblick über die verschiedenen Bereiche der Testautomatisierung geben um dann den Bereich der Automatisierten GUI-Tests näher zu beleuchten. Kapitel 1 ...

2 Grundlagen

2.1 Softwarequalität

Nach der Norm ISO-25000:2014 4.33 bezeichnet der Begriff der Softwarequalität die Fähigkeit einer Software die expliziten und impliziten Bedürfnisse von Benutzern, unter den Bedingungen unter sie benutzt wird, zu befriedigen. [Int14] Softwarequalität hat nach dieser Definition einen subjektiven Charakter. Dieser subjektive Charakter macht den Begriff in der Praxis schwer zu greifen und damit nicht direkt anwendbar. Aus diesem Grund existieren sogenannte Qualitätsmodelle, die den Begriff der Softwarequalität messbar und damit auch überprüfbar machen sollen. Ein solches Qualitätsmodell wird zum Beispiel in der ISO-Norm 9126 vorgestellt. Es werden verschiedene Qualitätsmerkmale definiert die zur Beurteilung der Gesamtqualität eines Softwareprodukts dienen. Hierunter fallen die Merkmale:

- Funktionalität
- Zuverlässigkeit
- Benutzbarkeit
- Effizienz
- Änderbarkeit
- Übertragbarkeit

Es existieren verschiedene Methoden um sicherzustellen, dass Software bezogen auf die Qualitätsmerkmale gewissen Anforderungen genügt. Eine Teil der Methoden geht dabei davon aus, dass ein qualitativ hochwertiger Prozess der Produkterstellung die Entstehung von qualitativ hochwertigen Produkten begünstigt. Das Augenmerk wird hierbei also auf die Prozessqualität gelegt. Allgemein fasst man diese Gruppe unter dem Begriff des prozessorientiertes Qualitätsmanagement zusammen. Die klassischen Vorgehensmodelle der Softwareentwicklung werden z.B. hier eingeordnet. Worauf sich diese Arbeit jedoch konzentrieren möchte sind die Methoden des produktorientierten Qualitätsmanagement. Hierbei wird das Softwareprodukt

direkt bezüglich der Qualitätsmerkmale überprüft. Darunter fallen beispielsweise Softwaretests.

2.2 Softwaretest

Das produktorientierte Qualitätsmanagement unterteilt sich weiter in die Bereiche des konstruktiven und analytischen Qualitätsmanagement. Unter dem konstruktiven Qualitätsmanagement versteht man in diesem Fall den Einsatz von z.B. Methoden, Werkzeugen oder Standards die dafür sorgen, dass ein (Zwischen-)Produkt bestimmte Forderungen erfüllt. Was man im Allgemeinen aber unter einem Softwaretest versteht ist im Bereich der prüfenden Verfahren des analytischen Qualitätsmanagement angesiedelt. Unter analytischen Qualitätsmanagement versteht man hier den Einsatz von analysierenden bzw. prüfenden Verfahren, die Aussagen über die Qualität eines (Zwischen-)Produkts machen.

Die Norm ISO/IEC/IEEE 24765:2010 3.280 definiert das testen von Software als „the dynamic verification of the behavior of a program on a finite set of test cases, suitably selected from the usually infinite executions domain, against the expected behavior.“ Aufgabe eines Softwaretests ist es dabei nicht einen Fehler im Code zu Lokalisieren und zu beheben. TODO:CITE Das Lokalisieren und Beheben des Defekts ist Aufgabe des Softwareentwicklers und wird auch als Debugging (Fehlerbereinigung, Fehlerkorrektur) bezeichnet. Während Debugging das Ziel hat, Defekte bzw. Fehlerzustände zu beheben, ist es Aufgabe des Tests, Fehlerwirkungen (die auf Defekte hinweisen) gezielt und systematisch aufzudecken. [Spi03] Dabei dienen definierte Anforderungen als Prüfreferenz, mittels derer ggf. vorhandene Fehler aufgedeckt werden. „Das Testen von Software dient durch die Identifizierung von Defekten und deren anschließenden Beseitigung durch das Debugging zur Steigerung der Softwarequalität“ [Spi03] Als möglicher Rahmen für die Anforderungen können z.B. die in 2.1 bereits beschriebenen Qualitätsmerkmale dienen.

2.3 Testprozess

Der Begriff des Softwaretests wie er in 2.2 beschrieben ist erfordert eine Einordnung in einen größeren Zusammenhang. Ein Softwaretest steht in der Regel nicht für sich alleine, sondern ist Teil eines größeren Prozesses der den Softwaretest in seinem gesamten Lebenszyklus begleitet. Durch den Testprozess wird die Aufgabe des Testens in kleinere Arbeitsschritte gegliedert. Die Phasen die man dabei unterscheidet sind:

- Testplanung und Steuerung
- Testanalyse und -design
- Testrealisierung und -durchführung
- Testauswertung und Bericht sowie Abschluss der Testaktivitäten

Obgleich die Aufgaben in sequenzieller Reihenfolge im Testprozess angegeben sind, können sie sich überschneiden und teilweise auch gleichzeitig durchgeführt werden. Diese Teilaufgaben bilden einen fundamentalen Testprozess und werden im Folgenden näher beschrieben. [Spi03, Seite 19]

Abbildungsverzeichnis

Literaturverzeichnis

- [Bur03] BURNSTEIN, Ilene: *Practical Software Testing: A Process-Oriented Approach*. Auflage: 2003. New York : Springer, 2003. – ISBN 9780387951317
- [Har00] HARROLD, Mary J.: Testing: A Roadmap. In: *Proceedings of the Conference on The Future of Software Engineering*. New York, NY, USA : ACM, 2000 (ICSE '00). – ISBN 1-58113-253-0, 61–72
- [Int14] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO): *ISO/IEC 25000:2014, Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE*. <https://www.iso.org/obp/ui/#iso:std:iso-iec:25000:ed-2:v1:en>. Version: März 2014
- [RW06] RAMLER, Rudolf ; WOLFMAIER, Klaus: Economic Perspectives in Test Automation: Balancing Automated and Manual Testing with Opportunity Cost. In: *Proceedings of the 2006 International Workshop on Automation of Software Test*. New York, NY, USA : ACM, 2006 (AST '06). – ISBN 1-59593-408-1, 85–91
- [Spi03] SPILLNER, Andreas: *Basiswissen Softwaretest*. 1. Aufl. Heidelberg : dpunkt-Verl., 2003. – ISBN 3-89864-178-3