

BACHELORARBEIT

im Studiengang Bachelor Informatik

Software-Test von Web-Applikationen

Ausgeführt von: Bernhard Posselt

Personenkennzeichen: 1010257029

Begutachter: MSc Benedikt Salzbrunn

Wien, 19. Mai 2013

Eidesstattliche Erklärung

„Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit selbständig angefertigt habe. Die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Die Arbeit wurde bisher weder in gleicher noch in ähnlicher Form einer anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht. Ich versichere, dass die abgegebene Version jener im Uploadtool entspricht.“

Ort, Datum

Unterschrift

Kurzfassung

Die Durchführung und Erstellen von automatisierten Tests für Web-Applikationen unterscheidet sich von klassischen Applikationen. Aufgrund der komplexeren Infrastruktur und Modularisierung werden zusätzliche Testfälle und Strategien benötigt um Web-Applikationen ausreichend abzudecken und eine fortwährende Qualität zu gewährleisten. Diese Arbeit soll Möglichkeiten für den Test von Web-Applikationen aufzeigen.

Schlagwörter: Webapplikationen, automatisierte Tests, Webtest

Abstract

Creation and execution of automatic web application tests is different from tests of classic applications. A more complex infrastructure and modularisation require additional testcases and strategies to guarantee a good enough test coverage which in return ensures constant quality. This thesis highlights various possibilities to test web applications.

Keywords: web applications, automatic tests, webtest

Inhaltsverzeichnis

1 Einführung	1
1.1 Vorteile von automatisierten Tests	1
1.2 Unterschiede zu klassischer Software	1
1.3 Aufbau	2
Literaturverzeichnis	3
Abbildungsverzeichnis	4
Abkürzungsverzeichnis	5

1 Einführung

1.1 Vorteile von automatisierten Tests

[1][S. 10]

1.2 Unterschiede zu klassischer Software

Im Gegensatz zu klassischen Desktop- oder Mobil-Applikationen bestehen Web-Applikationen wegen ihrer Client-Server-Architektur immer aus mehreren Modulen, die meist über ein Netzwerk miteinander verbunden sind, beispielsweise:

- Datenbank-Server
- Web-Server
- Applikations-Server
- Authentifizierungs-Server
- Web-Browser

Durch diesen modularen Aufbau ist es besonders schwer einen Fehler zu lokalisieren. Der Fehler z.B. durch einen Fehler im Quellcode des Applikations-Servers oder durch ein Netzwerkproblem entstanden sein. [2][Foreword]

Außerdem gibt es eine größere Vielfalt an Plattformen, auf welchen die Web-Applikation ausgeführt wird: Auf der Serverseite sind diese Plattformen noch vom/von der BetreiberIn festlegbar, sprich welches Betriebssystem und welche Datenbank eingesetzt wird, auf der Clientseite ist dies aber schon nicht mehr möglich. Die BesucherInnen der Webseite verwenden verschiedene Web-Browser auf verschiedenen Betriebssystemen, welche beide in unterschiedlichen Versionen vorliegen können. Auch können unterschiedliche Plugins und Fonts in unterschiedlichen Versionen installiert sein. [2][Foreword]

Zudem verlagert sich auch immer mehr Applikations-Logik von der Server- auf die Clientseite[2][S. 13]. Durch das Verwenden von *Events*, welche unter anderem durch BenutzerInnen-Eingaben ausgelöst werden können, stellt clientseitiger Code eine größere Herausforderung für den/die TesterIn dar als Serverseitiger: Events können in unterschiedlicher Reihenfolge und Kombination auftreten, manche Aktionen lösen sogar mehrere Events aus. [2][S. 18]

Eine weitere Herausforderung stellt das Instanzmodell von Web-Applikationen dar: die meisten Web-Applikationen erlauben durch das Verwenden von *Cookies* mehrere Instanzen der Applikation, die jedoch unter der gleichen Session ausgeführt werden. Dies kann zu Synchronisationsproblemen zwischen den einzelnen Instanzen führen, z.B. kann in einer Instanz ein Eintrag gelöscht werden, der durch eine fehlerhafte Synchronisation für die andere Instanz jedoch noch immer existiert und in weiterer Folge zu Fehlern führen kann. [2][S. 20]

Diese Vielfalt an verschiedenen, möglichen Konfigurationen und Herausforderungen erfordert eine neue Herangehensweise an das Thema Software-Test: Die bestehenden Techniken sind „zwar auch notwendig, aber nicht ausreichend, um die Qualität der Applikation sicherzustellen“ [3][S. 18]

1.3 Aufbau

Literaturverzeichnis

- [1] C. Dobritzhofer, "Towards Test Methodologies for Developing Large Systems," Master's thesis, Technische Universität Wien, 1994.
- [2] H. Q. Nguyen, *Testing Applications on the Web*. Wiley Computer Publishing, 2001.
- [3] S. Avci, "Evaluieren von automatisierten Tests bei Web-Applikationen," Master's thesis, Technische Universität Wien, 2010.

Abbildungsverzeichnis

Abkürzungsverzeichnis