Final thesis
Automated testing in web applications

Niclas Olofsson

February 4, 2014

Supervisor: Anders Fröberg
Examiner: Erik Berglund

Linköping University
Department of Computer and Information Science

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis vitae mi a dolor fringilla fermentum eu vel nibh. Phasellus orci purus, aliquet id lorem ut, cursus vehicula lacus. Cras sit amet arcu lorem. Maecenas pellentesque quis metus ac accumsan. Suspendisse ac purus erat. Curabitur pellentesque mattis nisl, nec convallis magna. Nam ultricies non lectus egestas pulvinar. Integer a elit volutpat, congue odio vitae, eleifend ipsum. Curabitur gravida sit amet tellus vel rutrum.

Acknowledgments

Duis sed vehicula felis. Donec augue quam, rutrum et augue ut, egestas varius orci. Quisque posuere eros turpis, vel blandit metus ornare eu. Donec nec sem sagittis, gravida lacus sed, dignissim elit. Donec vel velit nulla. Phasellus et consequat sapien. Suspendisse volutpat convallis turpis, in pretium libero. Sed sit amet orci dictum, interdum nulla sed, suscipit mauris. Nam vitae libero mattis mauris lacinia dictum. In feugiat, neque sit amet adipiscing dapibus, lorem lacus semper massa, ac consectetur felis purus a metus. Sed tempus mattis auctor. Suspendisse viverra venenatis sapien vitae pharetra.

Contents

1 Introduction

1.1 Background

During code refactoring or implementation of new features in software, errors often occur in existing parts. This may have a serious impact on the reliability of the system, thus jeopardizing user's confidence for the system. Automatic testing is utilized to verify the functionality of software in order to detect bugs and errors before they end up in a production environment.

The commissioner body of this project, GOLI, is a startup company developing a web application for production planning. Starting new web application companies often means rapid product development in order to create the product itself, while maintenance levels are low and the quality of the application is still easy to assure by manual testing. As the application and the number of users grows, maintenance and bug fixing becomes an increasing part of the development. The size of the application might make it implausible to test in a satisfying way by manual testing.

Due to requirements from customers, GOLI wishes to extend the code base of the web application to include new features for handling staff manning. The current system uses automatic testing to some extent, but these tests are cumbersome to write and takes long time to run. The purpose of the thesis is to analyze how this application can begin using tests in a good way whilst the application is still quite small. The goal is to determine a solid way of implementing new features and bug fixes in order for the product to be able to grow effortlessly.

1.2 Problem formulation

The goal of this final thesis is to analyze how automatic testing can be introduced in an existing web application in a good way, and if lower- level tests can be derived from existing high-level tests automatically. We will also study how this can be applied when extending the system with new features.

The main research questions of this project are the following:

- How can a combination of low-level and high-level testing be used when testing a web application?
- How can tests be implemented when adding new functionality into existing software?
- Is it possible to automatically derive lower-level unit tests from high-level behavioral tests?

1.3 Scope and limitations

There exists different categories of software testing, for example performance testing and security testing. The scope of this thesis is quality assurance testing, in which the purpose is to verify the functionality of a part of the system rather than measuring its characteristics. We will also only cover automatic testing, as opposed to manual testing where the execution and result evaluation of the test is done by a human. The term testing will hereby refer to automatic software quality assurance testing unless specified otherwise.

Since the result of this thesis will be evaluated in specific web application (i.e. the GOLI production planning application), we will only cover techniques which are relevant this specific application. In other words, we will focus on testing web applications which uses Ruby on Rails and the Knockout.js framework.

1.4 Method

We should work.

Bibliography