Automatic Generation of Artifacts for the Application of Two Web Application Testing Models

Submitted by: Sarah Vessels

Advisor: Dr. Jane Hayes

Department of Compute Science

University of Kentucky

Lexington, KY 40506

March 2011

Committee members:

**FILL THIS IN**

# Abstract

Web applications are prevalent and it is important that they be of high quality as businesses, schools, and public services rely upon them. Web applications can be tested using testing models designed for web applications. Unique testing models designed for web applications can be beneficial since web applications differ greatly from desktop applications. Web applications are accessed via a browser, and a user can manipulate the web application in ways not possible with desktop applications, such as modifying the URL or using the Back button in the browser. It can take a great deal of time to apply a given web application testing model to a web application. This project seeks to speed up the process of applying two particular web application testing models: the Atomic Section Model and the Qian, Miao, Zeng model. Two generators were written using the Ruby programming language, one for each model. The generators take as input the source code of a Ruby on Rails web application, and the URL to a web application written with any framework, respectively. They produce as output test paths that can be traversed manually to ensure good coverage of the web application, and artifacts that can be further manipulated manually to produce test paths, respectively.

# Introduction

Web applications require testing just as any software does, in order to ensure correct functionality, expected appearance and layout, and that correct data is displayed. Testing web applications is different than testing desktop applications because the primary way of using a web application is through another application: a web browser desktop application. Because of this method of access, web applications are subject to the user’s manipulation in ways that desktop applications cannot usually be manipulated. A user can manually change the URL in the web browser to point to another page in the web application. Users can also use their browser’s Back and Forward buttons to change pages. Different plugins, browser settings, and browser extensions alter the user experience. All these things can provide the web application with unexpected input, or cause it to perform actions in such a way that the designers did not anticipate. It is for this reason that different testing models exist solely for testing web applications.

One such model is the Atomic Section Model (ASM) [[[1]](#endnote--1)] by Jeff Offutt and Ye Wu. The ASM represents all possible user interface screens of a web application, just as control flow graphs represent all possible sequences of software program execution. ASM can be divided into two levels: the Component Interaction Model (CIM) and the Application Transition Graph (ATG) [2]. CIM is designed for the individual components of the web application while ATG is for the web application as a whole, in which the individual components come together and function as a system.

Another web application testing model is the Qian, Miao, and Zeng model (QMZ) [[[2]](#endnote-0)]. Applying the QMZ model involves the creation of a Page Flow Diagram (PFD) [2], which represents “the relationship among pages of a web application” [2]. A Page Test Tree (PTT) [2], based upon the PFD, is also necessary for generating shorter paths than those shown in the PFD. A PTT is a tree-like structure that is a simpler representation than that provided by the PFD of pages in the web application.

As part of this project, tools written in Ruby have been created to help automate the process of applying the ASM and the QMZ models to a web application. Due to the nature of the QMZ model, the underlying language of the web application being tested does not matter, so the QMZ tool is platform-agnostic.

1. Offutt, Jeff and Wu, Ye, “Modeling Presentation Layers of Web Applications for Testing”, Springer’s Software and Modeling, DOI: 10.1007/s10270-009-0125-4. [↑](#endnote-ref--1)
2. Qian, Miao, and Zeng, “A Practical Web Testing Model for Web Application Testing”, Third International IEEE Conference on Signal-Image Technologies and Internet-Based System, DOI 10.1109/SITIS.2007.16. [↑](#endnote-ref-0)