HTMLUnitGenerator

Enables user friendly and powerful front end testing of web applications with minimum required effort to implement.

AUTHOR: TOMAS BJERRE, TOMASBJERRE [AT] YAHOO.COM

Updated January 5, 2012

Contents

Chapter 1

Introduction

This is the official documentation of HTMLUnitGenerator. This introduction includes three questions and three answers. They are intended to quickly get you into the context of this document.

1.1 What is HTMLUnitGenerator?

It is a compiler. It translates a user friendly DSL[1] into a more advanced test case. That way you get clear test cases that are easily maintained while at the same time powerful and easily introduced in your current test suit.

1.2 Why should I use HTMLUnitGenerator?

1. **Easy to learn and fast to work with**
   The time you need to spend reading up on HTMLUnitGenerator before being able to produce qualitative test cases using it, is very short.

   If you write your test cases using, for example, raw Java and HTMLUnit you will need to come up with some hierarchy of classes to be able to re use code. XPath:s and URL:s should typically be defined once and then referenced in all your test cases. Developing such a structure takes time as well as explaining and documenting it to your colleges.

2. **No need to document test cases**
   The Flow language (see Section 2) is simple enaugh to, itself, qualify as documentation. Anyone, even people with no previous programming experience, will understand what your test cases do. The Flow language has been design with the intention to provide only an absolute minimum number of choices to the developer, in order to keep all test cases clean and neutral.

[1]Domain Specific Language

3. **Future safe**
   HTMLUnitGenerator is an open source software. You can write your own generator if you don't want to user HTMLUnit anymore, see Section 3.2. Or maybe you want to test your code using several different head less browsers. The idea is to provide several different generators with HTMLUnitGenerator[2].

[2]Yes, the name will change!

Chapter 2

Flow language

This chpater will describe the Flow language. It is the language used to express test cases.

This chapter starts with a quick look at a test case written in Flow, see Section 2.1, and continued with a complete walkthrough of the language.

2.1 Introduction

A quick example of a test case written in Flow is presented in Listing 2.1. The details of this test case is explained in this chapter.

```
Url SomeSite is http://www.somesite.domain/

Go to SomeSite
Find a with attribute href set to /some/target
```

Listing 2.1. Find href of an a tag

2.2 Defining paths

Flow uses XPath to define containers, like div-tags for example.

```
Path searchpopup is /html/body/div[7]/div/div[9]
Path search is //*[@id="eventIdsearch"]
Path website is /html/body
```

Listing 2.2. Defining XPaths

In Listing 2.2 a div is defined as *searchpopup*, an element with an id is defined as *search* and the entire website content is defined as *website*.

2.3 Defining URL:s

2.4 Go to URL:s

2.5 Find elements

2.6 Click on elements

2.7 Fill in forms

2.8 Re-use test cases

2.9 Use proxy

Chapter 3

Back End

3.1 HTMLUnit

3.2 Implementing your own generator

3.2.1 subsection