Introduction/Overview Selenium WebDriver

Introduction/Overview of Cucumber

Selenium Drivers:

Google Chrome, Internet Explorer 7 through 11, Firefox, Safari, Opera, HtmlUnit, phantomjs, Android (with Selendroid or appium), IOS (with ios-driver or appium)

Installation – Overview of Maven pom file 🡪 Cucumber, Selenium WebDriver, Junit

Steps to execute Cucumber files

List and explanation of Cucumber and Selenium WebDriver files

Jenkins

https://github.com/detro/ghostdriver

In reviewing how to run the Java tests using a continuous integration (ci) tool such as Jenkins, an option is to use Maven to run the Selenium tests using a headless environment using tools such as PhantomJS.

The steps to install Jenkins, Java, Maven, and Git on AWS E2 Ubuntu server: <https://gist.github.com/jsuwo/9038610>.

The Jenkins project setup for a Java project in GitHub. This example Jenkins has been installed on an AWS E2 Ubuntu server.

|  |  |
| --- | --- |
| Description | Command |
| *AWS E2 Ubuntu server* | |
| Switch to the Jenkins user. | $ sudo su – jenkins |
| Create public private key pair. | $ cd .ssh  $ ssh-keygen -t dsa |
| Cat the public key and copy the results. | $ cat MavenCucumberSeleniumWebDriverJUnitPrototype.pub |
| *GitHub* | |
| Add and paste deploy key in GitHub project. | [https://github.com/gdombchik/ MavenCucumberSeleniumWebDriverJUnitPrototype/settings/keys](https://github.com/gdombchik/%20MavenCucumberSeleniumWebDriverJUnitPrototype/settings/keys) |
| *Manage Jenkins – Configure System* | |
| Jenkins Location | Scroll to the “Jenkins Location” section.  Jenkins URL:  <http://ec2-52-43-191-185.us-west-2.compute.amazonaws.com/>  System Admin e-mail address:  [greg@gregorydombchik.com](mailto:greg@gregorydombchik.com) |
| E-mail Notification | Scroll to the “E-mail Notification” section.  SMTP server:  mail.gregorydombchik.com |
| *Jenkins Server – Global Tool Configuration* | |
| Maven Installation | Scroll to the “Maven” section.  Select “Maven installation…” button.  Name:  Apache Maven 3.0.5  MAVEN\_HOME:  /usr/share/maven/ |
| JDK Installation | Scroll to the “JDK” section.  Select “JDK installation…” button.  Name:  Oracle JDK 1.7  JAVA\_HOME:  /usr/lib/jvm/java-7-oracle/ |
| *Jenkins Server – New Project* | |
| Add A Jenkins Project for a Java project. | Select New Item.  Enter an item name.  Select Freestyle project.  Select Ok. |
| Specify the URL of the remote GitHub repository. | Scroll to the “Source Code Management” section.  Select the “Git” option.  Repository URL:  <https://github.com/gdombchik/MavenCucumberSeleniumWebDriverJUnitPrototype.git> |
| Update the Poll SCM. | Scroll to the “Build Triggers” section.  Select the “Poll SCM” option.  Enter the following in the “Schedule” text box:  H \*/3 \* \* \* |
| Update the Build. | Scroll to the “Build” section.  Select from the “Add build step” and select the “Invoke top-level Maven targets” option.  Maven Version:  (Default)  Goals:  clean install  POM:  pom.xml |
| Update E-mail Notification. | Scroll to the Post-build Actions.  Select from the “Add post-build action” and select the “E-mail Notification” option.  Enter email address in the “Recipients” text box:  [greg@gregorydombchik.com](mailto:greg@gregorydombchik.com) |
| *GitHub* | |
| Add Jenkins GitHub Plugin you can automatically trigger build jobs when  pushes are made to GitHub. | <https://github.com/gdombchik/MavenCucumberSeleniumWebDriverJUnitPrototype/settings/hooks>  Select from the “Add service” and select the “Jenkin’s (Git plugin)” option. (NOT GITHUB PLUGIN. This service does not appear to work).  Enter the following in the “Jenkins url” text box:  <http://ec2-52-42-216-209.us-west-2.compute.amazonaws.com/> |

The steps to install PhantomJS below are to be executed on the server where Jenkins is hosted.

|  |  |
| --- | --- |
| Install PhantomJS | As Referenced from <https://gist.github.com/julionc/7476620>.  Version: 2.1.1  Platform: x86\_64  First, install or update to the latest system software.  sudo apt-get update  sudo apt-get install build-essential chrpath libssl-dev libxft-dev  Install these packages needed by PhantomJS to work correctly.  sudo apt-get install libfreetype6 libfreetype6-dev  sudo apt-get install libfontconfig1 libfontconfig1-dev  Get it from the [PhantomJS website](http://phantomjs.org/).  cd ~  export PHANTOM\_JS="phantomjs-2.1.1-linux-x86\_64"  wget https://bitbucket.org/ariya/phantomjs/downloads/$PHANTOM\_JS.tar.bz2 sudo tar xvjf $PHANTOM\_JS.tar.bz2  Once downloaded, move Phantomjs folder to /usr/local/share/ and create a symlink:  sudo mv $PHANTOM\_JS /usr/local/share  sudo ln -sf /usr/local/share/$PHANTOM\_JS/bin/phantomjs /usr/local/bin  Now, It should have PhantomJS properly on your system.  phantomjs –version |

In summation, a website can be test building unit tests (JUnit) referencing the generated Java Page Objects denoting the web application using the Selenium and Cucumber test framework, and using a headless environment such as PhantomJS, and then porting the solution into a continuous integration environment (Jenkins).