# Week 06 Instructor Notes

## Overview

Students will be looking at different Testing Management Tools and Testing software libraries.

## Objectives

By the end of the week, each student will be able to:

* Understand the critical role testing has in DevOps
* Evaluate Testing Tools

## Before the Week Begins (Prepare)

### Partnership/Group work:

Teams and Partners will be assigned a tool to do research on and create a poster presentation.

* Teach
  + DevOps Automated Tools
    - Jenkins (CI)
    - CruiseControl (CI)
    - CyPress (front-end, end to end)
    - Azure (DevOps, CI/CD, Cloud)
    - Quality Center (Test Planning, Performance)
    - Test Rail (Web-based)
    - Testpad (Manual/Task oriented)
    - Xray (Manual/Auto)
    - TestingWhiz
  + Codeless Software Telerik (Codeless, recorder)
    - Quick Test Professional (codeless)
    - Subject7 (Web & Mobile) (codeless) (Jira/Jenkins plugin)
    - Screenster (application, screen recorder)
* Prove
  + Testing Framework
    - TestComplete (OS apps, Web)
    - Mocha (Javascript)
    - CasperJS (Javascript)
    - WebDriverJS
    - Junit
    - Watir (Web-based)
    - Angular (IDE plugin)
      * Protractor (End to End)
    - Relational Functional Tester (automation)
* Or any combination of the following:
  + DevOps: Azure, Jenkins, or a tools from DZone DevOps
  + Testing Platform: Protector, , Mocha, FitNesse, QuickBuild / JBoss Developer, , VersionOne, Jira, Microsoft CodedUI, QuickTest Pro
  + Testing Program: Angular, Junit \

## During the Week (Teach)

### Labs:

* Install an automated Selenium testing program on your computer

## Looking Ahead

Next week's announcements:

* Please create and post your Announcement for W07 by Friday of this week.

# Prepare

## Overview

Testing is a critical phase just before delivering the software out the door (DevOps). Too often, the phase becomes smaller as the programmer dip into it, and the final delivery day is looming. Finding solutions to work faster in these times helps with the stress. Different tools are available to help practitioners work more efficiently and faster in S/W Testing. Having the right tool for the right job is a sign of a professional. Many times, knowing how to make tools when none are available is a valuable skill.

## Objectives

By the end of the week, each student will be able to:

* Understand the critical role testing has in DevOps
* Evaluate Testing Tools

## Preparation Material

To be prepared for this module's activities, please read the following and be prepared to start or complete activities for Teach One Another and Prove assignments.

### Reading

* See [Reading Materials](../Reading/Reading.html)

## Additional Information: Software Testing Tools

* Test Automation Management Tools
  + Jenkins (CI)
  + CruiseControl (CI)
  + CyPress (front-end, end to end)
  + Azure (DevOps, CI/CD, Cloud)
  + Quality Center (Test Planning, Performance)
  + Test Rail (Web-based)
  + Testpad (Manual/Task oriented)
  + Xray (Manual/Auto)
  + TestingWhiz

# What tools are out there to help with Testing?

## Overview

Testing is a critical phase just before delivering the software out the door (DevOps). Many times, there is a reduction in the testing phase. Finding solutions to work faster in these times helps with the stress. Different tools are available to help practitioners work more efficiently and faster in S/W Testing. Having the right tool for the right job is a sign of a professional. Many times, knowing how to make tools when none are available is a valuable skill.   
Working with your previous week's partner or group, research an assigned testing tool, and create a **product review poster**

* DevOps Automated Tools
  + Jenkins (CI)
  + CruiseControl (CI)
  + CyPress (front-end, end to end)
  + Azure (DevOps, CI/CD, Cloud)
  + Quality Center (Test Planning, Performance)
  + Test Rail (Web-based)
  + Testpad (Manual/Task oriented)
  + Xray (Manual/Auto)
  + TestingWhiz
* or
  + Telerik (Codeless, recorder)
  + Quick Test Professional (codeless)
  + Subject7 (Web & Mobile) (codeless) (Jira/Jenkins plugin)
  + Screenster (application, screen recorder)
* or
  + Testing Platform: Protector, , Mocha, FitNesse, QuickBuild / JBoss Developer, , VersionOne, Jira, Microsoft CodedUI, QuickTest Pro
  + Testing Program: Angular, Junit

## Topics

Based on your reading, work in groups of 2-4 individuals.

### Research your tool

Find the following information:

* Description
  + Advantages and Disadvantages over other tools in the same category
* Uses
  + Determine testing situations to use the tool.
  + What benchmarks make the test tool a good tool?
  + What type of scripts, harnesses, framework, and environment does the tool use?

### Create a Product Review

* Provide a written description of the highlights of the things that you discovered as a poster.
  + The Poster consists of a Title, description, significant features, quotes from the user, and information on how to get the software
  + The Poster should answer the 5 Ws (Who, What, Where, When, Why)
  + The Poster should address any known issues

Post a Copy or Link of your Testing Product Poster

* Upload a copy of your post so that your classmates can review your Poster.

## Replies

Review your classmate's posters. And then reply to your post with the answers to the following posts.

* How is this tool similar or different from the tool you researched?
* How does this tool solve a business case, a fundamental element of Testing, work with a methodology or lifecycle phase or testing technique?

## Rubric

Use the following rubric to help understand the expectation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Exceptional 100%** | **Good 90%** | **Acceptable 70%** | **Developing 50%** | **Missing 0%** |
| **Product Review Poster  50%** | The Poster contains two identifiable above and beyond elements. | The Poster is well-formatted, easily readable. | The Poster contains the required product information. | Missing required elements | No Poster |
| **Reply  30%** | Group provided evidence that they read all of the posters. | The group read three posters. | The group answered all questions | Missing required elements | No Article summary |
| **Team Work 20%** | Assignments have meeting time or team discussion summary | Assignments are identical | The team assigned different question to individuals and compiled into one document | The team assigned different questions to individuals and sent them in individually | The team worked on assignments individually and submitted them individually |

The distribution of points starts at 50 percent for the minimal participation expectation, and additionally stated expectations increase the percents.

# Lab 06 - Install Selenium

## Overview

Install an automated testing program on your computer.

## Assignment

### Instructions for installing Microsoft Visual Studio Community with Python

If you have not installed Microsoft Visual Studio, please do. Visual Studio Community is not Visual Studio Code. (Selenium will not work for Code.   (2.61 GB download)

[Install Python for Visual Studio](https://docs.microsoft.com/en-us/visualstudio/python/tutorial-working-with-python-in-visual-studio-step-00-installation?view=vs-2019)

### install Webdriver on Local Machine

For Selenium to work on your project, you need to install the web driver associated with your preferred browser.   
You will need to know if you are working on a 32-bit or 64-bit system. If you don't know, look up how to do it online, depending on your Operating System. Most systems are 64-bit. If that doesn't work, then try 32-bit.   
You will need to create a directory to store the Selenium web driver. I would suggest 'C:\Program Files\SeleniumDriver.' You will then need to add the Path to the system.  
For example, for Windows 10

* Right-click on My Computer or This PC from File Explorer
* Select Change Setting, bottom right corner
* System Properties window pops up; click on Advanced
* You should see Environment Variables.
* From "User variables...", click Path, and then Edit button.
* From the "Edit environment variables" window, click New
* Copy "C:\Program Files\SeleniumDriver".
* Click Ok to all dialogs

**Restart MS Visual Studio; this will reset the Path inside MS Visual Studio to find the Path.**

There are several options for which browser to use. If your preferred browser is not on this list, you are on your own.

* Firefox
  + Go to this website: https://www.guru99.com/gecko-marionette-driver-selenium.html
  + From the "Download and Install Gecko Driver" web page, click on the link: <https://github.com/mozilla/geckodriver/releases>
  + From the "Release - Mozilla" web page, choose your operating system.
    - For example, I am running Window 10 64-bit, and so I choose: [geckodriver-v0.26.0-win64.zip](https://github.com/mozilla/geckodriver/releases/download/v0.26.0/geckodriver-v0.26.0-win64.zip)
    - This downloads the zip file, open the file.
  + From the zip or gz file, move the geckodriver\* executable to the SeleniumDriver directory
* Internet Explorer
  + You will need to know what version of Internet Explorer you are using.
  + Go to this website: http://selenium-release.storage.googleapis.com/index.html
  + From the index, choose your IE version, i.e., 3.9
  + From the directory, choose either 64 or 32 bit IEDriverServer\*zip file.
    - This downloads the zip file, open the file.
  + From the zip or gz file, move the IEDriverServer\* executable to the SeleniumDriver directory
* Chrome
  + Go to this website: https://sites.google.com/a/chromium.org/chromedriver/home
  + From the "ChromeDriver" webpage, and under the section "All version available in Downloads," click on "Current stable release: [ChromeDriver 80.0.3987.106"](https://chromedriver.storage.googleapis.com/index.html?path=80.0.3987.106/" \t "_blank)
  + From the index, choose your chromedriver\*
    - This download the zip file, open the file.
  + From the zip or gz file, move the chromedriver\* executable to the SeleniumDriver directory

### Instructions for Downloading and InstallING SELENIUM

We now need to create a python project in Visual Studio.

* From Visual Studio's File Menu, click File:New: Project
* From New Project, change the name to CSE270Lab06
  + This will create a new project with a new python file
* From Solution Explorer, click the arrow next to Python Environment to get Python to show up
* From Solution Explorer, right-click on "Python \*," and click "Install Python Package."
* In the "Python Environments" in the search, type selenium, click on "Run command: pip install Selenium"

### Instruction for testing selenium

In the empty py file, copy and paste the following Code.

from Selenium import common

from selenium import webdriver

from Selenium.webdriver.common.by import By

from Selenium.webdriver.common.keys import Keys

from time import sleep

# Uncomment out the corresponding browser webdrive you downloaded

browser = webdriver.Ie()

#browser = webdriver.Firefox()

#browser = webdriver.Chrome()

browser.get('http://classic.tiddlywiki.com')

sleep(5)

browser.find\_element(By.LINK\_TEXT, "close others").click()

browser.find\_element(By.LINK\_TEXT, "close").click()

browser.find\_element(By.LINK\_TEXT, "Examples").click()

sleep(5);

browser.quit();

 Run the project. The Code will open up your browser with the webpage classic.tiddlywiki.com, close all of the tiddlers, and then open up Examples. Within 5 seconds, it will close.  
If this doesn't happen, look at the output in the terminal. Troubleshoot at this point.

## Submission

Post your Code for Testing. Post any problems that you encountered or lessons that you experienced.

## Rubric

Use the following rubric to help understand the expectation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Exceptional 100%** | **Good 90%** | **Acceptable 70%** | **Developing 50%** | **Missing 0%** |
| **Assignment 90%** | Summary of Problems or Lessons Learned | Install & Test Selenium | Install Webdriver | Install MS Visual Studio | No original post |
| **Professionalism 10%** | The paper is easy to read and communicated. | Properly cited, there are no grammar or spelling errors, and the writing style is "professional." | Found an instance of a spelling error, grammar error, incomplete citation, overly verbose wording, poor formatting, or poor writing. | A citation is missing where one is needed (plagiarism alert!). | Gross spelling/grammar errors or other aspects of the writing that make the paper difficult to read. |

# Prove: Trade Study

## Overview

Identify the best tool for the job.

## Backstory

HELP Inc. wants the test team to use Selenium based on research from its competitors. You are not sure if this is the best option. You convince your manager to allow you to put together a White paper on different options. Use Selenium and two other software testing products to compare.

.

## Instruction

In this instance, a trade study or white paper is the same thing: the summary and comparison of 2-3 products that a manager could decide to use, and justification to hold fast when the initial or 2nd attempts to use it fail to dismiss critics.   
For example, FamilySearch.org committed to AWS, even though some of their critics, internally and externally, argued that they locked themselves into specific capabilities or the lack thereof.   
Your motivation is to find out if Selenium is the best option. Make sure you have identified the major standard features of each type of testing environment product. Feel free to leverage the information found in the posters. In conclusion, make sure that you have provided a summary and recommendation of the product.  
Write a trade study/white paper comparing three different tools.

### Pick two other tools

Pick two different testing frameworks found from class research, other than Selenium.

* Testing Framework
  + TestComplete (OS apps, Web)
  + Mocha (Javascript)
  + CasperJS (Javascript)
  + WebDriverJS
  + Junit
  + Watir (Web-based)
  + Angular (IDE plugin)
    - Protractor (End to End)
  + Relational Functional Tester (automation)
* or
  + DevOps: Azure, Jenkins, or a tools from DZone DevOps
  + Testing Platform: Protector, Mocha, FitNesse, CruiseControl, QuickBuild / JBoss Developer, Quality Center, VersionOne, Microsoft CodedUI, QuickTest Pro

Research all three, two and Selenium.

### Basic Characteristics

Identify the essential characteristics of all three frameworks.

### Differences

Identify what characteristics make the three products stand out from each other.  
Describe the significant differences and advantages of each.

### Rank Differences

Determine a method to measure the tool's characteristics. Rank the characteristics.

### Product Ranking

Compare characteristic ranking to product and determine the best product.

### Conclusion

Create a report with your findings. Include all comparisons and methods of comparison.  
Make sure you include the following information

* Identification of best product and why. Cost, Training, Capabilities.
* Identify uses in a specific practice (see Discussion: Practices).
* Side by Side layout of the advantages and disadvantages of the tools.
* Identify the benchmarks to rate tools.
* How do other companies work with these tools?

Provide your first and second choice.

## Make it your Own

The completion of the core of this assignment is 89%. The assignment needs additional personalized work to achieve 100%.

Make it professional looking, like add graphs, charts, pictures.

## Submission

Make sure that you upload a copy of your document to iLearn.

## Rubric

Use the following rubric to help understand the expectation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Exceptional 100%** | **Good 90%** | **Acceptable 70%** | **Developing 50%** | **Missing 0%** |
| **Trade Study Tests 80%** | Make it your own | Use of weekly assignments (Tools and Practices) | The study contains a comparison of Selenium and two other products. | Has two or more major issues | No answers to questions |
| **Professionalism 10%** | Make it your own | Properly cited, there are no grammar or spelling errors, and the writing style is "professional." | Found an instance of a spelling error, grammar error, incomplete citation, overly verbose wording, poor formatting, or poor writing. | A citation is missing where one is needed (plagiarism alert!). | Gross spelling/grammar errors or other aspects of the writing that make the paper difficult to read. |
| **Citations 10%** | One of the citations is a primary source | Contains 3-4 citations other than the reading. | Contains 1-2 more citations other than the reading | Contains citations from the reading | No Citations |