CS580 Advanced Software Engineering
Assignment 3 - Maven Exercise

Due Date
Monday, October 13, 2014

Score
10

Questions and Directions

1. Add 3rd party Java libraries to iphoto-web

We want to add the following two Java libraries to our project:

- JSoup (A popular Java library to parse HTML pages) http://jsoup.org/
- ImageJ (An advanced Java-based image processing library) http://imagej.nih.gov/ij/

Of course, we would like to use Maven to automatically configure and add the libraries to our project dependencies. You need to search the Maven package dependency information for both of the libraries, and modify the `pom.xml` file in your iphoto-web project. The official maven package repository is located at: http://search.maven.org/

You need to verify your change by checking if the two jar files are present the "Maven Dependencies" in your Eclipse project.

You need to commit your change on the `pom.xml` to GitHub.

2. Add a new method that uses JSoup

Similar to Assignment 2, you need to add a new method in the `WebController.java` (add a new HTTP URL), which does something interesting with Jsoup (e.g., print out all the hyperlinks in a certain web page).

You can google tons of code examples in Jsoup. For instance, the example we used during the class is at: http://www.programcreek.com/2012/05/parse-html-in-java/

You can also find a lot of code samples at: http://jsoup.org/cookbook/

You don't need to spend a lot of time on digging into Jsoup or producing very fancy function, since the main goal here is to ensure that you can use the library and integrate it with the existing web service.

Again, you need to commit the changes on the new method to GitHub.

3. Practice Maven build lifecycles

Make sure you have installed Maven command line tool (http://maven.apache.org/). You can run command “mvn” in your iphoto-web folder to verify it.

If it runs correctly, you can practice and try the following different Maven build goals:

```
mvn compile
mvn test
mvn package
```

With `mvn package`, you should be able to see the generated jar file in `target` folder. Then, try to run your iphoto-web in the shell through the following command (make sure you have JDK installed and configured):

```
java -jar iphoto-web-1.0.jar
```

You should be able to see the running web application through command above. Go to your web browser to verify it with the new URL you just added.