Google Summer of Code 2015

Systers, an Anita Borg Institute Community

Automated Unit Testing

Madi Paris

About Myself

Madi Paris
Michigan Technological University
Brighton, MI
Eastern Time Zone (UTC -5:00)

About My Mentor

Wendy Knox Everette
Systers, an Anita Borg Institute Community
Washington D.C.
Eastern Time Zone (UTC -5:00)

About the Project

- RHOK Atlanta June 2013.
- Chuukese people have no written language.
- Web application that would work offline.
- Take photos, upload them, and add Chuukese and English translations.
- Ruby on Rails.
- Expanding into new languages and organizations.

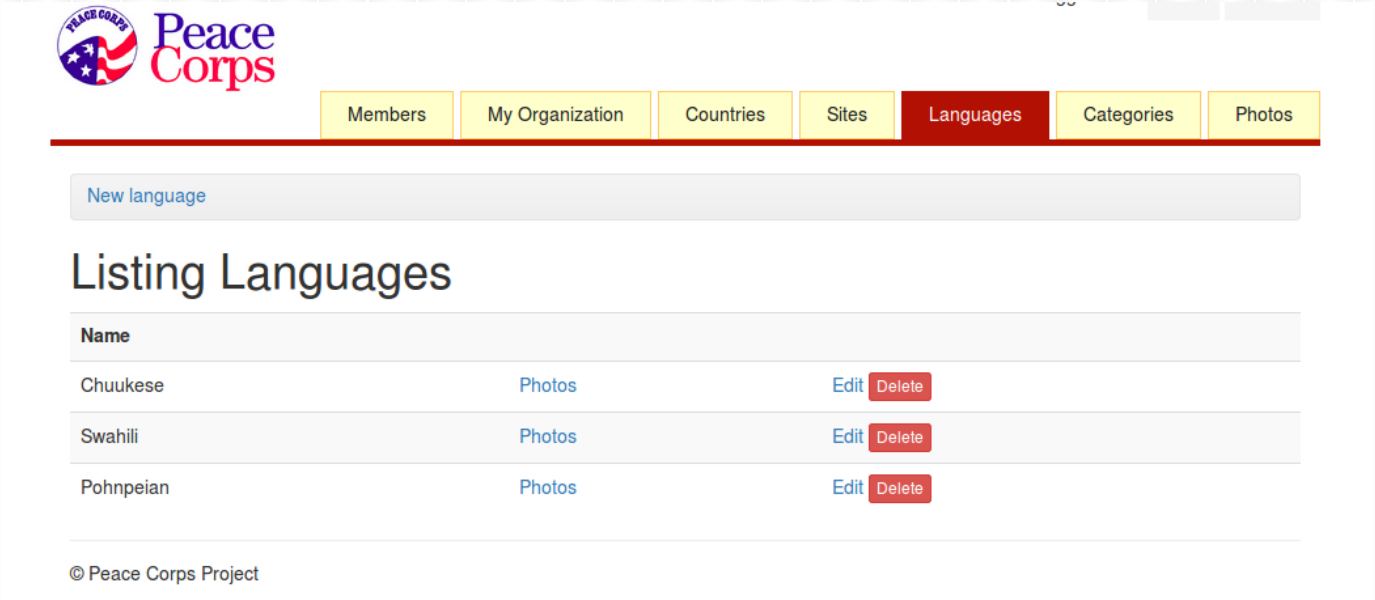
PLT Tools

- Rails 4.2.1
- Ruby 2.2.0p0
- PostgreSQL
- Travis CI

Running PLT

- Download the PLTcode directory.
- Enter the application's root directory.
- Add a database.yml to config
- $ rake db:create
- $ rake db:migrate
- $ rake db:seed
- $ bower install
- $ rails s
- Enter localhost:3000 on a web browser

Photo Language Translation



Goal for AUT

The objective of this project is to create, maintain, and execute automated tests for the Photo Language Translation application that is to be used by Peace Corps volunteers by creating a suite of reusable and automated scripts for regression testing using Selenium.

Tools Used

- Java
- JUnit
- Selenium
- Eclipse IDE
- Maven
- Systers Dev Environment

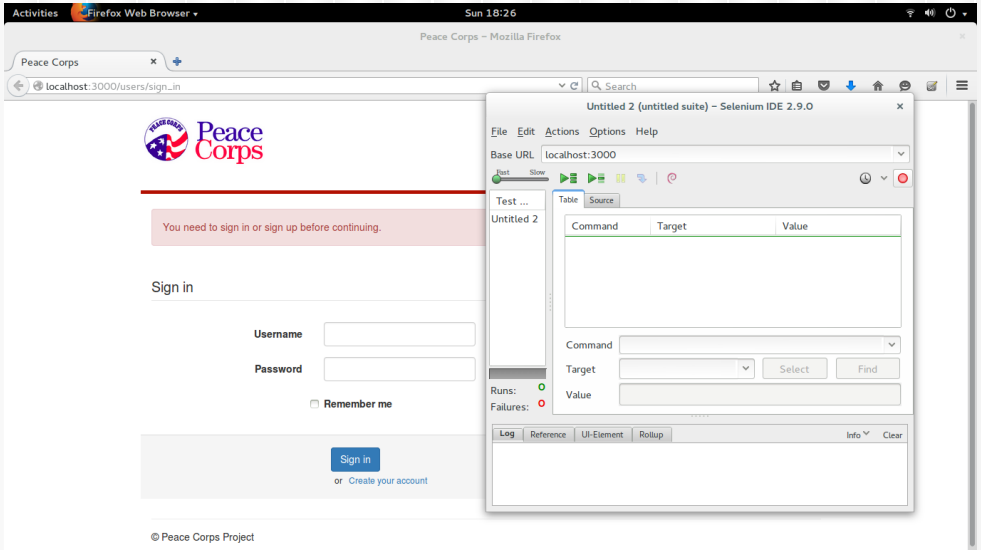
Selenium

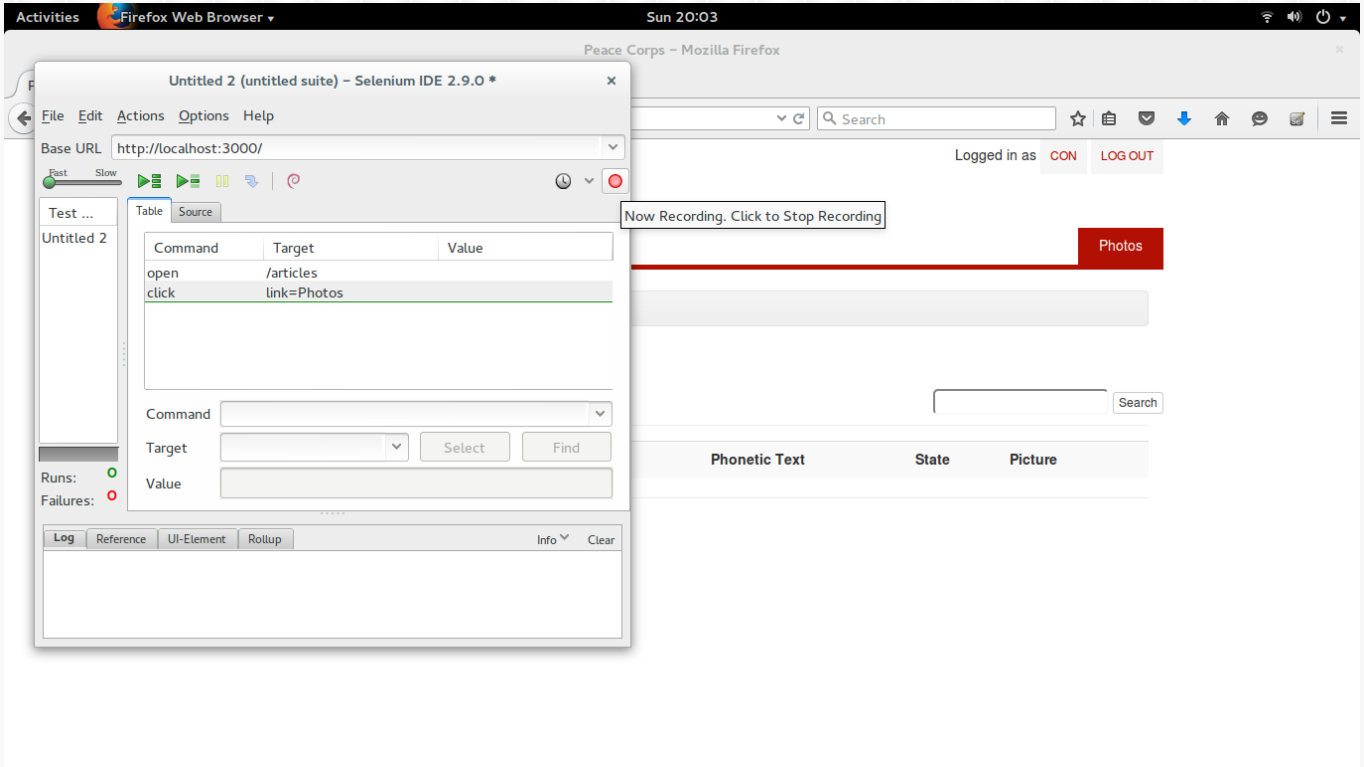
- Portable software testing framework for web applications.
- Can be used to write tests in programming languages such as Java, C#, Groovy, Perl, PHP, Python, and Ruby.
- Deploys on Windows, Macintosh, and Linux platforms.
- Open-source software under the Apache 2.0 license.
- Free to use.
- Download the Selenium Client and WebDriver Language Bindings for the language that will be used.
- Record-and-playback interactions with browser.
- Checks whether things are located on a page.

Create Test Case

- Open Selenium IDE from browser.

- Create new test case in Selenium IDE. (Ctrl + N)
- Press record.
- Once you are done running desired test, press record button again.




Activities
Firefox Web Browser
Sun 20:03
Peace Corps – Mozilla Firefox
Untitled 2 (untitled suite) – Selenium IDE 2.9.0 *
File Edit Actions Options Help
Base URL http://localhost:3000/
Fast Slow
Test ...
Untitled 2
Table Source
Command Target Value
open /articles
click link=Photos
Command
Target
Select
Find
Value
Runs: 0
Failures: 0
Log Reference UI-Element Rollup
Info
Clear
Now Recording. Click to Stop Recording
Search
Logged in as CON LOG OUT
Photos
Search
Phonetic Text
State
Picture

Saving Test Case

- To save, go to “File”, “Export Test Case As…”, “Java / JUnit 4 / Webdriver”.

Maven

- Build automation tool.
- Describes the software is built, dependencies, plug-ins, etc in an XML file.
- Used primarily for Java projects.

Systers Dev Environment

- Install VirtualBox (https://www.virtualbox.org/wiki/Downloads)
- Install Vagrant (http://www.vagrantup.com/downloads.html)
- Clone the DevEnvironment repository from Systers (https://github.com/systers/DevEnvironments)
- Change directory to the systers-autotest folder inside the DevEnvironment folder which should contain a vagrant file.
- If using Windows or OSX, run '*gem install vagrant*'. Linux does not require this step.
- Run '*vagrant up*'
- Run '*vagrant ssh*' (username and password are both 'vagrant')
- If a GUI is desired, run '*sudo apt-get install ubuntu-desktop*' within vagrant.

Timeline

May 25 - May 31:

- Gather more specific information about the specific needs for the PLT unit testing.

June 1 - June 28:

- Make a list of test cases to be written.
- Start writing tests.
- Documentation.

June 29 - July 5:

- Midterm Evaluation.
- Start making adjustments to code based on evaluation.

Timeline

July 6 - July 26:

- Refactor code.
- Continue writing any remaining tests and debugging.
- Work on documentation.
- Add any additional features.

July 27 - August 2:

- Debugging
- Work on Documentation.

Timeline

August 3 - August 9:

- Finish up documentation.
- Make any final code adjustments.

August 10 - August 21:

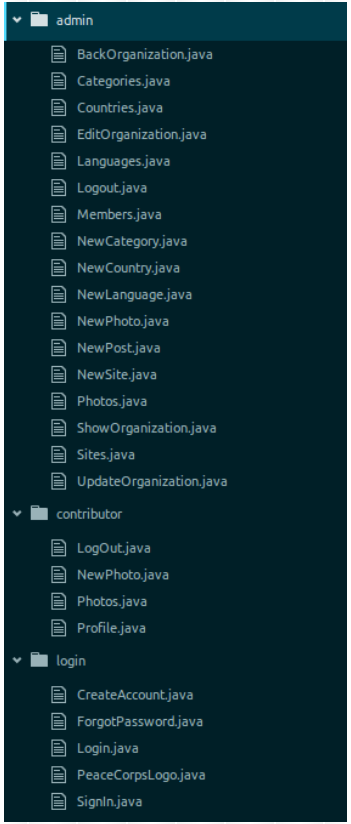
- Final Evaluation.
- Buffer time for any last minute adjustments.

What I Have Done

- https://github.com/systers/automated-testing/tree/develop/PLTTests
- I completed my goals based on my timeline.

Features Worked On

- Writing tests for admin, contributor, and login portion of program.



List of Admin Tests

Admin:

- -BackOrganization
- -Categories
- -Countries
- -EditOrganization
- -Languages
- -Logout
- -Members
- -NewCategory
- -NewCountry
- -NewLanguage
- NewPhoto
- NewPost
- NewSite
- Photos
- ShowOrganization
- Sites
- UpdateOrganization

List of Contributor Tests

Contributor:

- -LogOut
- -NewPhoto
- -Photos
- -Profile

List of Login Tests

Login:

- -CreateAccount
- -ForgotPassword
- -Login
- -PeaceCorpsLogo
- -SignIn

List of Commons Tests

Commons:

-CommonCode

Features Worked On

- Included asserts statements.
- If the item being tested is found then return true, else return false.

```
1  package admin;
2  import commons.CommonCode;
3  import org.junit.*;
4  import org.openqa.selenium.*;
5
6  /*
7   * @author Madi Paris
8   * Test for Members UI feature
9   */
10
11 public class Members extends CommonCode{
12   private WebDriver driver;
13   private String baseUrl;
14
15   @Test
16   public void test() {
17     driver.get(baseUrl + "/");
18     if(driver.findElement(By.linkText("Members"))){
19       JUnit.AssertTrue("Found the members link", true);
20     }
21     else {
22       JUnit.fail("No members link found");
23     }
24   }
25 }
```

```java
package contributor;
import commons.CommonCode;
import org.junit.*;
import org.openqa.selenium.*;

/*
 * @author Madi Paris
 * Tests for New Photo UI feature
 */

public class NewPhoto extends CommonCode{
  private WebDriver driver;
  private String baseUrl;

  @Test
  public void test() {
    driver.get(baseUrl + "/articles");
    if(driver.findElement(By.linkText("New Photo"))){
      JUnit.AssertTrue("Found the new photo link", true);
    }
    else {
      JUnit.fail("No new photo link found");
    }
  }
}
```

```java
package admin;
import commons.CommonCode;
import org.junit.*;
import org.openqa.selenium.*;

/*
 * @author Madi Paris
 * Test for the edit UI feature on the Organization page
 */

public class EditOrganization extends CommonCode{
  private WebDriver driver;
  private String baseUrl;

  @Test
  public void test() {
    driver.get(baseUrl + "/organizations/1");
    if(driver.findElement(By.linkText("Edit"))){
      JUnit.AssertTrue("Found edit link", true);
    }
    else {
      JUnit.fail("No edit link found");
    }
  }
}
```

Features Worked On

- Refactored tests so there is a helper code that has the repetitive code in a different file that all tests can use.
- @Before, @After, isElementPresent, isAlertPresent, closeAlertAndGetItsText.

```java
package commons;
import java.util.regex.Pattern;
import java.util.concurrent.TimeUnit;
import org.junit.*;
import static org.junit.Assert.*;
import static org.hamcrest.CoreMatchers.*;
import org.openqa.selenium.*;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.support.ui.Select;

/*
 * @author Madi Paris
 * Code to be used by other tests
 */

public class CommonCode {
  private WebDriver driver;
  private String baseUrl;
  private boolean acceptNextAlert = true;
  private StringBuffer verificationErrors = new StringBuffer();

  @Before
  public void setUp() throws Exception {
    driver = new FirefoxDriver();
    baseUrl = "http://localhost:3000/";
    driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);
  }
```

Features Worked On

- Setting up maven with the tests.
- Java Compiler Plugin, JUnit dependency, Selenium dependency, Surefire dependency.

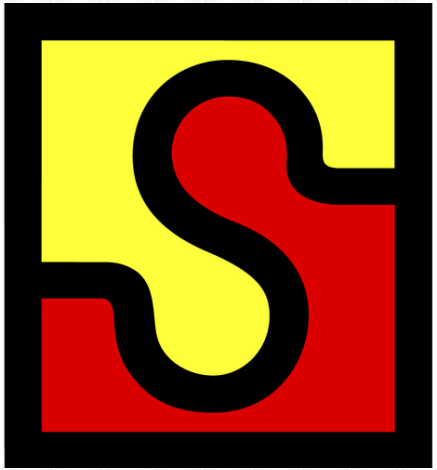
```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>PLTTests</groupId>
  <artifactId>PLTTests</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>PLTTests</name>
  <url>http://maven.apache.org</url>
  <build>
    <sourceDirectory>target</sourceDirectory>

    <plugins>
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.0</version>
        <configuration>
          <source>1.7</source>
          <target>1.7</target>
        </configuration>
      </plugin>
    </plugins>
  </build>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>
```

Features Worked On

- Started integrating with Sauce Connect/Travis CI.

Sauce Connect

- Secure tunneling app which allows you to execute tests securely when testing behind firewalls via a secure connection between Sauce Labs' client cloud and your environment.
- Integrates with Travis CI.
- Free for open-source projects.
- Allows the Sauce browsers driven by Selenium tests to access the Web Application that a Travis build starts.



Travis CI

- Open-source hosted, distributed continuous integration service used to build and test projects hosted at GitHub.
- configured by adding a file named .travis.yml, which is a YAML format text file, to the root directory of the GitHub repository.
- automatically detects when a commit has been made and pushed to a GitHub repository that is using Travis CI, and each time this happens, it will try to build the project and run tests.
- It supports building software in numerous languages, including C, C++, C#, Clojure, D, Erlang, F#, Go, Groovy, Haskell, Java, JavaScript, Julia, Perl, PHP, Python, R, Ruby, Rust, Scala and Visual Basic.
- Will notify developer of success/failure of build via email, IRC, etc.

Next Step For The Project

- Continue to add more tests for new features added to Photo Language Translation as that continues to evolve.
- Implement DB test for PLT.
- Finish integrating with Sauce Connect/Travis CI.
- Post test reports.

AUT Links

- http://sauceio.com/index.php/2013/03/run-your-selenium-tests-completely-in-the-cloud-using-travis-ci-and-sauce-labs/
- http://docs.seleniumhq.org/download/
- https://github.com/systers/DevEnvironments
- https://maven.apache.org/
- http://junit.org/
- https://www.java.com/en/about/
- https://eclipse.org/home/index.php
- https://docs.saucelabs.com/reference/sauce-connect/
- https://travis-ci.org/

PLT Links

- http://rubyonrails.org/
- https://www.ruby-lang.org/en/downloads/
- http://www.postgresql.org/download/
- http://bower.io/
- https://github.com/systers/language-translation

Thank You!