Test Cases

Covered by JUnit and Manual Testing

Test case id:1

Scenario: Verify that the xml returned by the Zillow API call is constructed as expected.

Test Step: make a get request in order to retrieve the information stored in an xml from the api

Expected Result: XML is based upon the three given Strings path, address and cityZipCode

Actual outcome: XML is constructed correctly

Test case id:2

Scenario: Verify that the build link used to make the api call is correctly constructed

Test Step: construct the build link based on the input parameters path, address and cityZipCode

Expected Result: Build link is built correctly

Actual Outcome: Build link is built as expected

Covered By Manual testing (to be automated)

Test case id:3

Scenario: verify that the CLARIFAI Api call uses the given image

Test Step: verify that Clarifai uses the correct image path

Expected result: the image path is correct

Actual result: The image path is correctly passed to the API

Test case id: 4

Scenario: Given that an image is correctly uploaded

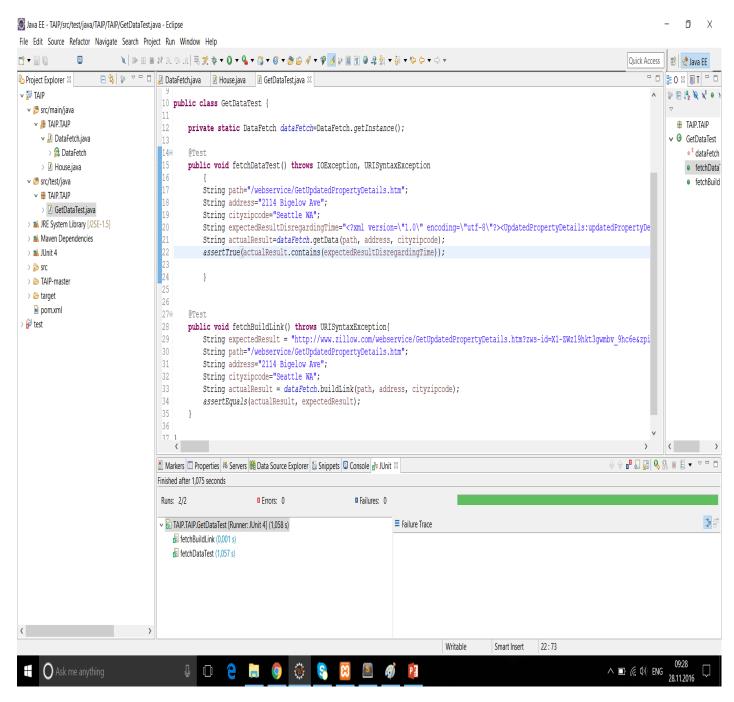
Then the Submit button is working as expected

Test Step: Press Submit button with image inserted

Expected Result: After pressing submit the image get tagged correctly

Actual result: The submit button works as expected

JUnit Tests



AOP in Javascript

```
var obtainKeys= (function() {
    var keys=getKeys() || {};

    return {
        Keys: function() {
        | return keys;
    }

}})();

//function written for logging purposes in order to retrieve confirmation
// when the required credential are obtained in order to make the api call
        function gotKeysConfirmation() {
            console.log("Got client id and client secret");
      }

    var LogInfo = AOP.aspect(gotKeysConfirmation).after(getKeys);

LogInfo();
```

```
/functions written for logging purposes before and after the API is called
  function beforeApiCall()
      console.log("Preparing to make the APi call and assign colored tags");
  function afterApiCall()
      console.log("Color tags have been assigned");
  var BeforeAssigningColorTags = AOP.aspect(beforeApiCall).before(assignColor);
  var AfterAssigningColorTags = AOP.aspect(afterApiCall).after(assignColor);
  BeforeAssigningColorTags();
  AfterAssigningColorTags();
  function assignColor (prob, index, length) {
      if (prob > 0.9) {
          return "success"
      else if (prob > 0.85 && prob <= 0.9) {
          return "info"
      else {
          return "warning"
```

Full code can be found on GitHub.