**Changes in existing Tests / Test Framework.**

/\*\*

\* Created by: Anand Bagmar

\* Email: anandb@thoughtworks.com, abagmar@gmail.com

\* Date: Dec 29, 2010

\* Time: 9:34:02 AM

\*/

import org.testng.annotations.AfterMethod;

import org.testng.annotations.Test;

import com.thoughtworks.webanalyticsautomation.plugins.Status;

import com.thoughtworks.webanalyticsautomation.scriptrunner.SeleniumScriptRunner;

import com.thoughtworks.webanalyticsautomation.utils.BROWSER;

import com.thoughtworks.webanalyticsautomation.utils.SeleniumScriptRunnerHelper;

import com.thoughtworks.webanalyticsautomation.utils.TestBase;

import static com.thoughtworks.webanalyticsautomation.Controller.getInstance;

public class EngineWithSeleniumTest extends TestBase {

private SeleniumScriptRunnerHelper seleniumScriptRunnerHelper;

private Engine engine;

private String webAnalyticTool = "omniture";

private String inputFileType = "xml";

private boolean keepLoadedFileInMemory = true;

private String inputDataFileName = " test\\sampledata\\OmnitureData.xml";

/\*

\* If using TestNG, override the @AfterMethod annotation

\* If using JUnit, override the @After annotation

\* and add the following line in it:

\* engine.disableWebAnalyticsTesting();

\* See the example shown below

\*/

@AfterMethod()

public void tearDown() throws Exception {

engine.disableWebAnalyticsTesting();

seleniumScriptRunnerHelper.stopSeleniumDriver();

}

@Test

public void captureAndVerifyDataReportedToWebAnalytics\_Selenium\_IE() throws Exception {

String actionName = "OpenUpcomingPage";

engine = getInstance(webAnalyticTool, inputFileType, keepLoadedFileInMemory);

engine.enableWebAnalyticsTesting();

/\*

\* Existing test / test framework code

\*/

session().open(SeleniumScriptRunnerHelper.BASE\_URL + "/upcoming");

/\*

\* The following piece of code, engine.verifyWebAnalyticsData (..) should be invoked when the UI action is done, and you want to verify the tags reported to the Web Analytics system.

\*/

Result verificationResult = engine.verifyWebAnalyticsData (inputDataFileName, actionName, new SeleniumScriptRunner(session()));

/\*

\* Sample validations

\*/

assertNotNull(verificationResult.getVerificationStatus(), "Verification status should NOT be NULL");

assertNotNull(verificationResult.getListOfErrors(), "Failure details should NOT be NULL");

assertEquals(verificationResult.getVerificationStatus(), Status.PASS, "Verification status should be PASS");

assertEquals(verificationResult.getListOfErrors().size(), 0, "Failure details should be empty");

}

}