Pega Selenium Starter Kit- Running Tests

Contents

What’s this guide?

This document describes how to execute CRM tests shipped out of the box with the Selenium Starter Kit. It assumes that you have your test environment already configured to run the tests. If not, refer to the setup guide for instructions to set up your environment and test project(s).

This guide uses Sales Automation application as an example, but the content is applicable to other CRM applications that are part of this suite.

Prerequisites

- Successfully setup and built `pega-crm-ui-testframework` project.

If do not have the project ready, refer to the setup guide.

Test Organization

Test organization is an essential aspect of test design and development. A well-organized test bed can facilitate better navigability as well as help with test selection. In this example project, Cucumber Tags are used to organize features and scenarios. For example, here’s `Opportunity.feature` file that groups all scenarios related to Sales Automation Opportunities feature.

```
@opportunity @smoke @smoke-sales-automation
Feature: Basic Opportunity flows
Tests covering the core Opportunity flow actions like Create, Change Stage and
Closing an opportunity.

Background:
    Given User logs in to SA Application as salesrep

@TC-create-business-opportunity
  Scenario Outline: Creating a Business Opportunities

    Given navigates to "Opportunities" List page
    When users clicks on Create OpprotunityButton and selects "<Opptype>"
    When Enters all the mandatory data for "<Opptype>"
    Then "<Opptype>" Opportunity should be created
    Then opportunity should have all the tabs

    Examples:
   | Opptype    |
   | Business   |
```

Tags not only serve the purpose of organizing tests but also offer a means for test selection. You will see more on that in the following Test Execution section

Test Execution

Tests shipped with this kit are Cucumber/Gherkin based behavior driven (BDD) tests. Cucumber tests can be run from command line using the CLI Runner, build tool or an IDE. In this project, we will use the Maven build tool approach as an example.

Setting Global Properties

Global settings and test requirements are defined in <PROJECT_ROOT>/data/global-settings.properties. Changing these settings will allow to customize test execution. The following tables show the list of properties:

Application Information:

Property	Description
`instance.url`	URL of the application under test

Browser Configuration:

Property	Description
`browser.name`	Name of the browser used for testing. Supported browsers: • chrome • firefox • ie • safari • htmlunit

`chrome.driver`
`ie.driver`
`edge.driver`
`chrome.driver.linux`
	Path to the appropriate browser binaries/driver

`isChromeAutoDownload`	By default, we attempt to download an appropriate chrome driver automatically through our custom utility. If it fails, set this property to false and copy the driver manually to binaries folder

Diagnostics & Debug Settings:

Property	Description
`debug.mode`	Boolean indicating whether to keep the browser open after test execution
`enable.fullscreen.mode`	Boolean indicating whether tests run in full screen mode
`global.timeout`	Override maximum wait time for the web elements to load (secs). Default timeout is 300 seconds.

Test Environment Configuration:

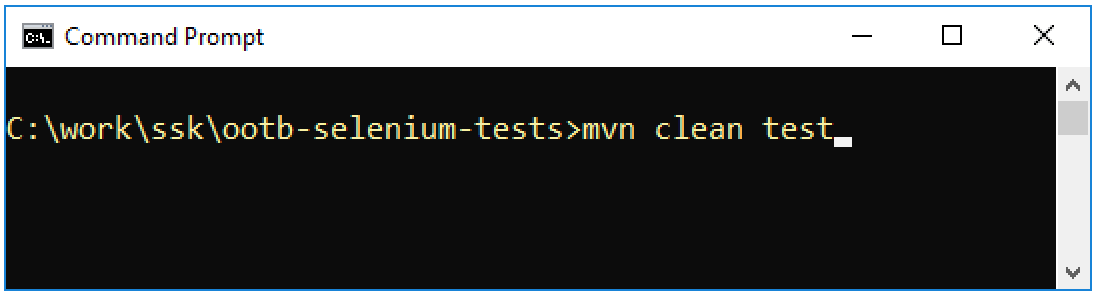
Property	Description
`hub.url`	URL to selenium grid hub for Cross Browser Testing. If this is not set, tests run locally
`capabilities`	Any custom capabilities provided by the external selenium grid providers like crossbrowsertesting / saucelabs / browserstock. Multiple capabilities can be provided by separating them with , and :

	capabilities=capability1:value1, capability2:value2, capability3:value3, …..

Maven Way

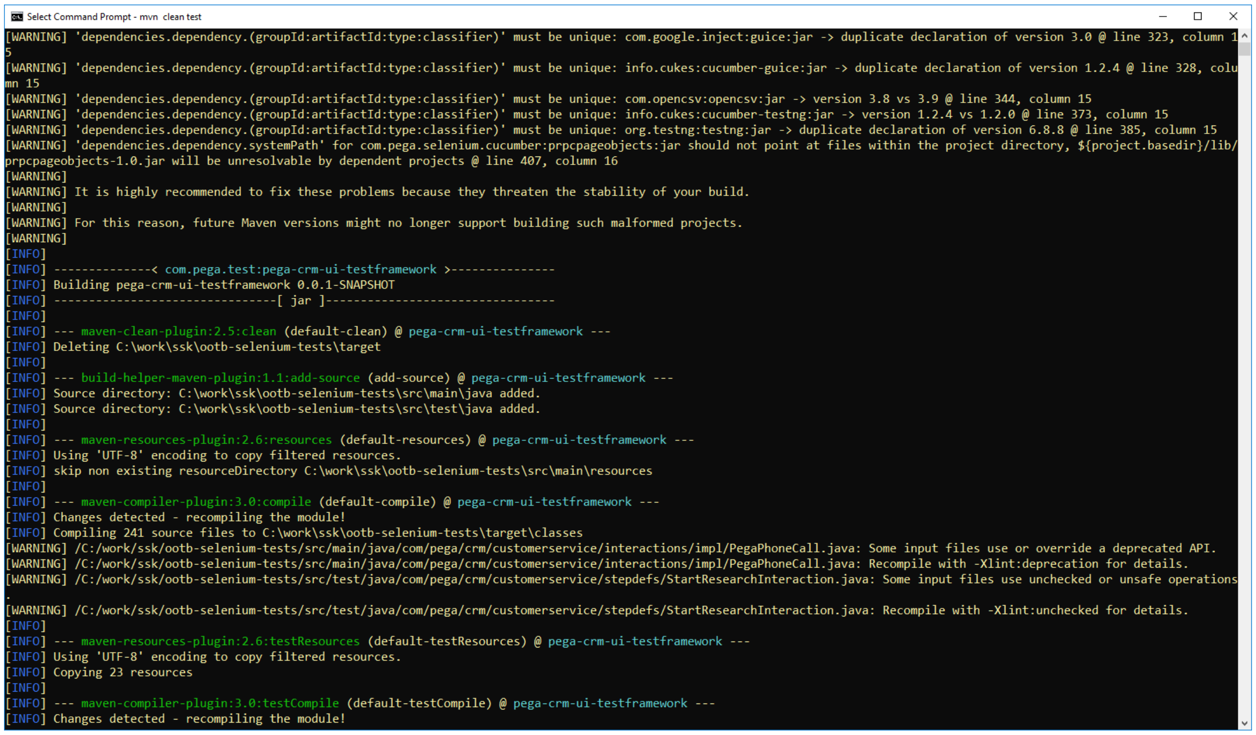
Command Line

To use Maven CLI to run Cucumber tests, invoke the following command from the project root location:



```
C:\work\ssk\ootb-selenium-tests>mvn clean test_
```

…



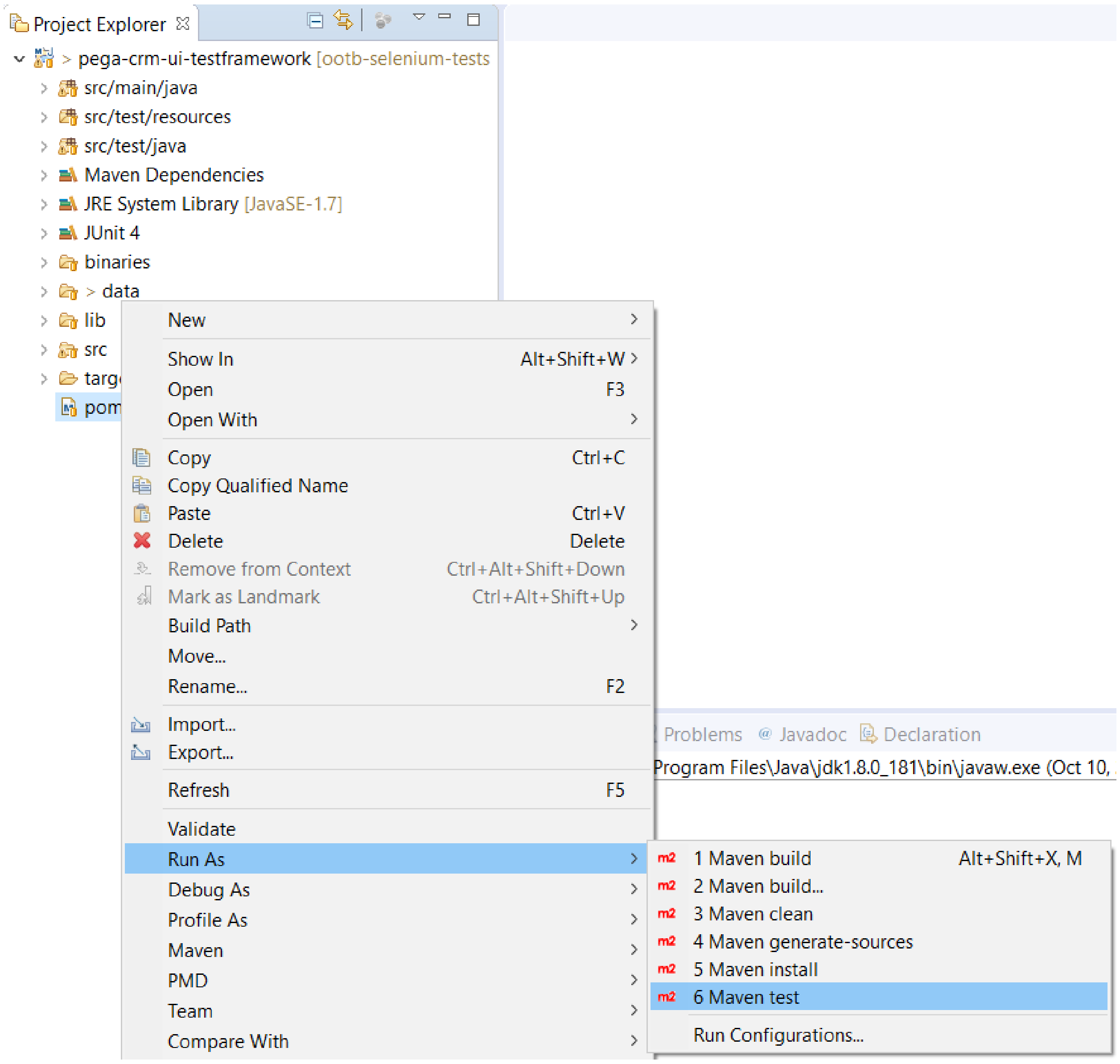
```
[WARNING] 'dependencies.dependency.(groupId:artifactId:type:classifier)' must be unique: com.google.inject:guice:jar -> duplicate declaration of version 3.0 @ line 323, column 1
5
[WARNING] 'dependencies.dependency.(groupId:artifactId:type:classifier)' must be unique: info.cukes:cucumber-guice:jar -> duplicate declaration of version 1.2.4 @ line 328, colu
mn 15
[WARNING] 'dependencies.dependency.(groupId:artifactId:type:classifier)' must be unique: com.opencsv:opencsv:jar -> version 3.8 vs 3.9 @ line 344, column 15
[WARNING] 'dependencies.dependency.(groupId:artifactId:type:classifier)' must be unique: info.cukes:cucumber-testng:jar -> version 1.2.4 vs 1.2.0 @ line 373, column 15
[WARNING] 'dependencies.dependency.(groupId:artifactId:type:classifier)' must be unique: org.testng:testng:jar -> duplicate declaration of version 6.8.8 @ line 385, column 15
[WARNING] 'dependencies.dependency.systemPath' for com.pega.selenium.cucumber:prpcpageobjects:jar should not point at files within the project directory, ${project.basedir}/lib/
prpcpageobjects-1.0.jar will be unresolvable by dependent projects @ line 407, column 16
[WARNING]
[WARNING] It is highly recommended to fix these problems because they threaten the stability of your build.
[WARNING]
[WARNING] For this reason, future Maven versions might no longer support building such malformed projects.
[WARNING]
[INFO]
[INFO] -------------< com.pega.test:pega-crm-ui-testframework >--------------
[INFO] Building pega-crm-ui-testframework 0.0.1-SNAPSHOT
[INFO] -------------------------------[ jar ]--------------------------------
[INFO]
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ pega-crm-ui-testframework ---
[INFO] Deleting C:\work\ssk\ootb-selenium-tests\target
[INFO]
[INFO] --- build-helper-maven-plugin:1.1:add-source (add-source) @ pega-crm-ui-testframework ---
[INFO] Source directory: C:\work\ssk\ootb-selenium-tests\src\main\java added.
[INFO] Source directory: C:\work\ssk\ootb-selenium-tests\src\test\java added.
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ pega-crm-ui-testframework ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory C:\work\ssk\ootb-selenium-tests\src\main\resources
[INFO]
[INFO] --- maven-compiler-plugin:3.0:compile (default-compile) @ pega-crm-ui-testframework ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 241 source files to C:\work\ssk\ootb-selenium-tests\target\classes
[WARNING] /C:/work/ssk/ootb-selenium-tests/src/main/java/com/pega/crm/customerservice/interactions/impl/PegaPhoneCall.java: Some input files use or override a deprecated API.
[WARNING] /C:/work/ssk/ootb-selenium-tests/src/main/java/com/pega/crm/customerservice/interactions/impl/PegaPhoneCall.java: Recompile with -Xlint:deprecation for details.
[WARNING] /C:/work/ssk/ootb-selenium-tests/src/test/java/com/pega/crm/customerservice/stepdefs/StartResearchInteraction.java: Some input files use unchecked or unsafe operations
.
[WARNING] /C:/work/ssk/ootb-selenium-tests/src/test/java/com/pega/crm/customerservice/stepdefs/StartResearchInteraction.java: Recompile with -Xlint:unchecked for details.
[INFO]
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ pega-crm-ui-testframework ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Copying 23 resources
[INFO]
[INFO] --- maven-compiler-plugin:3.0:testCompile (default-testCompile) @ pega-crm-ui-testframework ---
[INFO] Changes detected - recompiling the module!
```

Eclipse IDE

To run Cucumber with Maven from within the IDE, make sure Maven is installed, M2_HOME is correctly configured, and the IDE is configured with the latest Maven installation. Refer to Maven setup in the Setup Guide.

Before we trigger any test, make sure the screen resolution is set to 1920x1080 minimum, as this is the minimum resolution to run the tests successfully. Also make sure the tests run in full screen mode by setting enable.fullscreen.mode property to true in global-settings.properties file

To trigger test execution, in Eclipse IDE, right click on the `pom.xml` of the project and select `Run As > Maven test`



Cucumber Options

Cucumber framework provides several options for configuring test execution. Typically, when running tests from command line, these options can be provided in the Junit runner class using the `@CucumberOptions` annotation. For example:

```
import cucumber.api.CucumberOptions;
import cucumber.api.testng.AbstractTestNGCucumberTests;

@CucumberOptions(plugin = {"pretty", "html:cucumber-htmlreport"})
public class RunCukesTest extends AbstractTestNGCucumberTest {
```

```
...
}
```

When using Maven, however, these options can be passed using the "-Dcucumber.options" argument as follows:

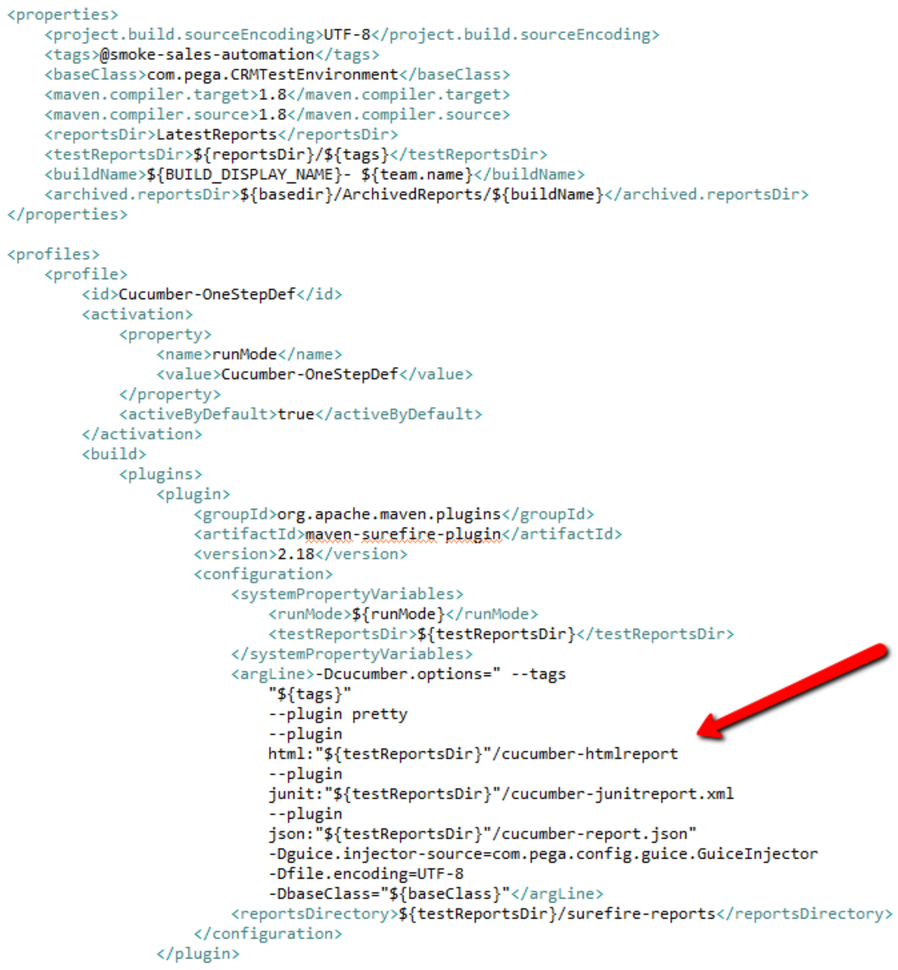
```
>> mvn test -Dcucumber.options=<OPTS>
```

For example, the following command pretty formats the test report generated at the end of the test execution:

```
>> mvn test -Dcucumber.options="--plugin pretty --plugin
html:latestreports/cucumberhtmlreports"
```

Passing "-Dcucumber.options" argument to Maven command overrides options specified in the Junit runner class.

In this project, the cucumber options are defined in the pom.xml file:



```
<properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <tags>@smoke-sales-automation</tags>
    <baseClass>com.pega.CRMTestEnvironment</baseClass>
    <maven.compiler.target>1.8</maven.compiler.target>
    <maven.compiler.source>1.8</maven.compiler.source>
    <reportsDir>LatestReports</reportsDir>
    <testReportsDir>${reportsDir}/${tags}</testReportsDir>
    <buildName>${BUILD_DISPLAY_NAME}- ${team.name}</buildName>
    <archived.reportsDir>${basedir}/ArchivedReports/${buildName}</archived.reportsDir>
</properties>

<profiles>
    <profile>
        <id>Cucumber-OneStepDef</id>
        <activation>
            <property>
                <name>runMode</name>
                <value>Cucumber-OneStepDef</value>
            </property>
            <activeByDefault>true</activeByDefault>
        </activation>
        <build>
            <plugins>
                <plugin>
                    <groupId>org.apache.maven.plugins</groupId>
                    <artifactId>maven-surefire-plugin</artifactId>
                    <version>2.18</version>
                    <configuration>
                        <systemPropertyVariables>
                            <runMode>${runMode}</runMode>
                            <testReportsDir>${testReportsDir}</testReportsDir>
                        </systemPropertyVariables>
                        <argLine>-Dcucumber.options=" --tags
                            "${tags}"
                            --plugin pretty
                            --plugin
                            html:"${testReportsDir}"/cucumber-htmlreport
                            --plugin
                            junit:"${testReportsDir}"/cucumber-junitreport.xml
                            --plugin
                            json:"${testReportsDir}"/cucumber-report.json"
                            -Dguice.injector-source=com.pega.config.guice.GuiceInjector
                            -Dfile.encoding=UTF-8
                            -DbaseClass="${baseClass}"</argLine>
                        <reportsDirectory>${testReportsDir}/surefire-reports</reportsDirectory>
                    </configuration>
                </plugin>
```

Selecting Tests to Run

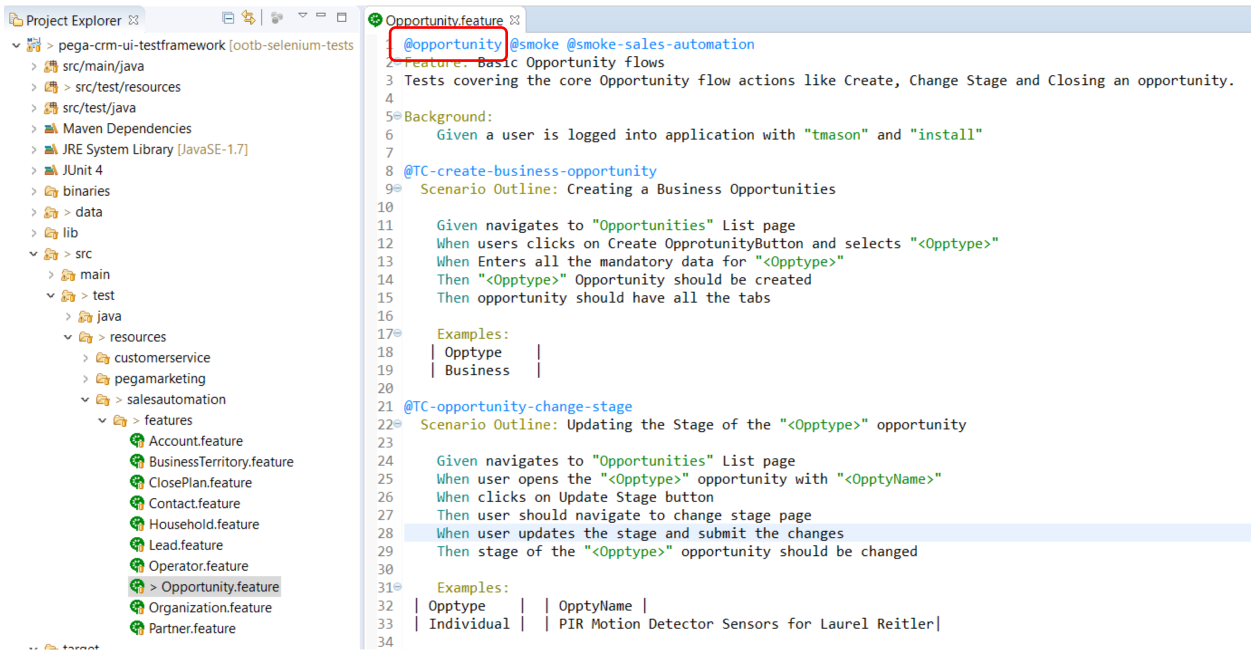
So far, we looked at how to trigger test execution but how do we select what tests to run. This is where Cucumber Tags come into play.

Generally, you define what tests to run in the Cucumber options specification. You select the features and scenarios to run using Cucumber Tags (--tags) or Regular expression (--name) depending on how you organize your tests.

In this sample CRM project, the OOTB tests are organized using tags. See Test Organization section for more information.

Running tagged features/scenarios

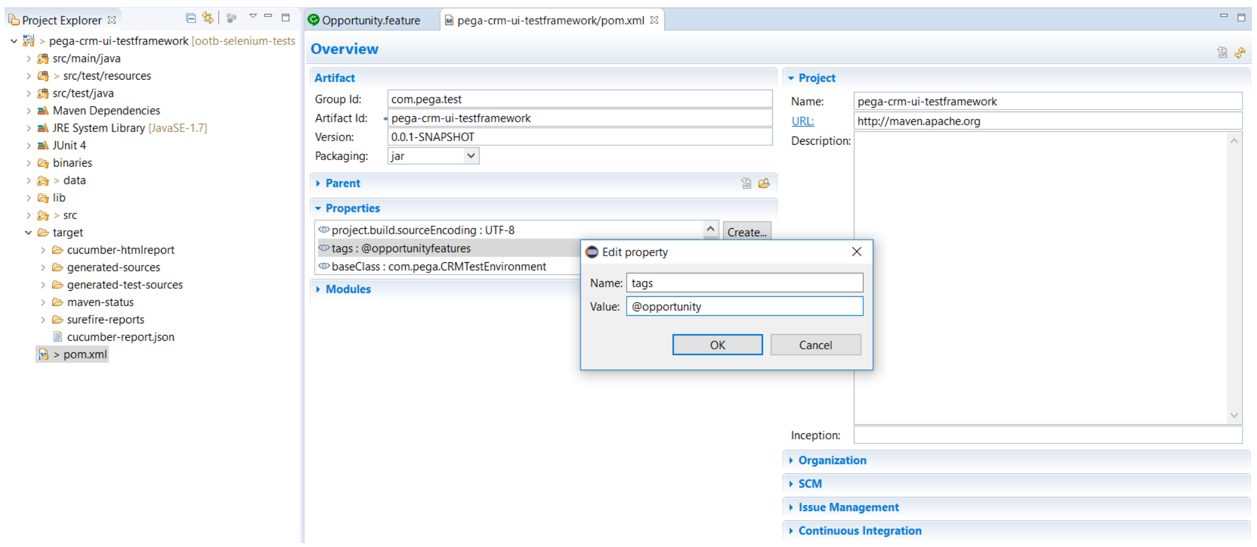
Let's assume you want to run all scenarios related to Sales Opportunities. The project organizes all sales opportunities related tests with `@opportunityfeatures` tag.



```
@opportunity @smoke @smoke-sales-automation
Feature: Basic Opportunity flows
Tests covering the core Opportunity flow actions like Create, Change Stage and Closing an opportunity.

Background:
    Given a user is logged into application with "tmason" and "install"

@TC-create-business-opportunity
  Scenario Outline: Creating a Business Opportunities

    Given navigates to "Opportunities" List page
    When users clicks on Create OpprotunityButton and selects "<Opptype>"
    When Enters all the mandatory data for "<Opptype>"
    Then "<Opptype>" Opportunity should be created
    Then opportunity should have all the tabs

    Examples:
   | Opptype    |
   | Business   |

@TC-opportunity-change-stage
  Scenario Outline: Updating the Stage of the "<Opptype>" opportunity

    Given navigates to "Opportunities" List page
    When user opens the "<Opptype>" opportunity with "<OpptyName>"
    When clicks on Update Stage button
    Then user should navigate to change stage page
    When user updates the stage and submit the changes
    Then stage of the "<Opptype>" opportunity should be changed

    Examples:
 | Opptype    |  | OpptyName |
 | Individual |  | PIR Motion Detector Sensors for Laurel Reitler|
```

Update the pom file to select appropriate tests:

1. Open the pom.xml file
2. In `Properties` section, double click on the `tags` property to set it to an appropriate value. In this case, that would be `@opportunity.`


Project Explorer

- pega-crm-ui-testframework [ootb-selenium-tests
  - src/main/java
  - > src/test/resources
  - src/test/java
  - Maven Dependencies
  - JRE System Library [JavaSE-1.7]
  - JUnit 4
  - binaries
  - > data
  - lib
  - > src
  - target
    - cucumber-htmlreport
    - generated-sources
    - generated-test-sources
    - maven-status
    - surefire-reports
    - cucumber-report.json
  - > pom.xml

Opportunity.feature | pega-crm-ui-testframework/pom.xml

Overview

Artifact

Group Id:	com.pega.test
Artifact Id:	pega-crm-ui-testframework
Version:	0.0.1-SNAPSHOT
Packaging:	jar

Parent

Properties

- project.build.sourceEncoding : UTF-8
- tags : @opportunityfeatures
- baseClass : com.pega.CRMTestEnvironment

Modules

Edit property

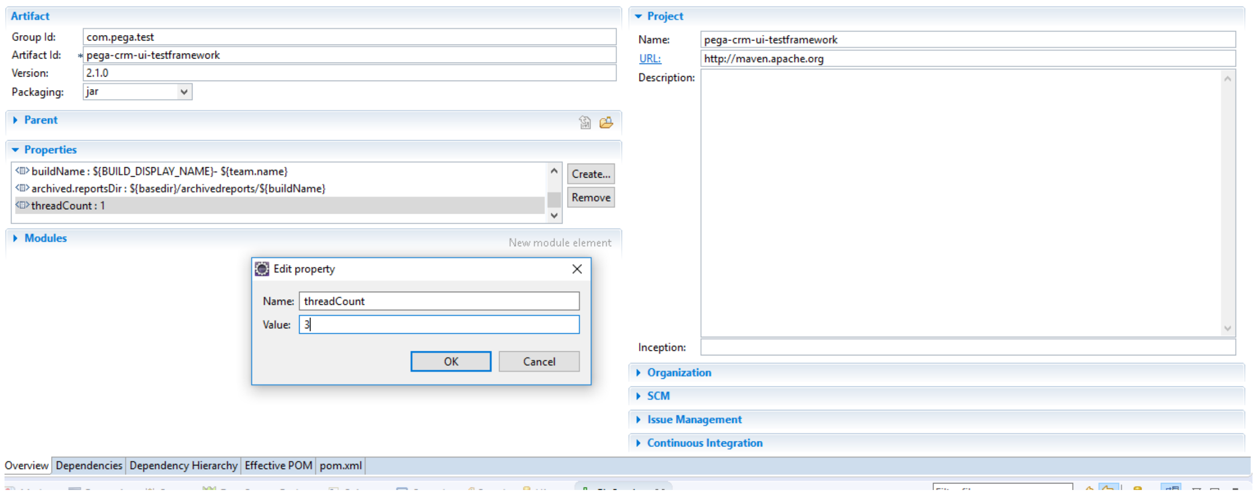
Name:	tags
Value:	@opportunity

Project

Name:	pega-crm-ui-testframework
URL:	http://maven.apache.org
Description:	
Inception:	

Organization | SCM | Issue Management | Continuous Integration

3. Click OK
4. If user wants to run multiple test cases in parallel then In `Properties` section, double click on the `threadCount` property to set it to an appropriate value. For example, if user wants to run three test cases parallelly , the value would be 3


Artifact

Group Id:	com.pega.test
Artifact Id:	pega-crm-ui-testframework
Version:	2.1.0
Packaging:	jar

Parent

Properties

- buildName : ${BUILD_DISPLAY_NAME}- ${team.name}
- archived.reportsDir : ${basedir}/archivedreports/${buildName}
- threadCount : 1

Modules

Edit property

Name:	threadCount
Value:	3

Project

Name:	pega-crm-ui-testframework
URL:	http://maven.apache.org
Description:	
Inception:	

Organization | SCM | Issue Management | Continuous Integration

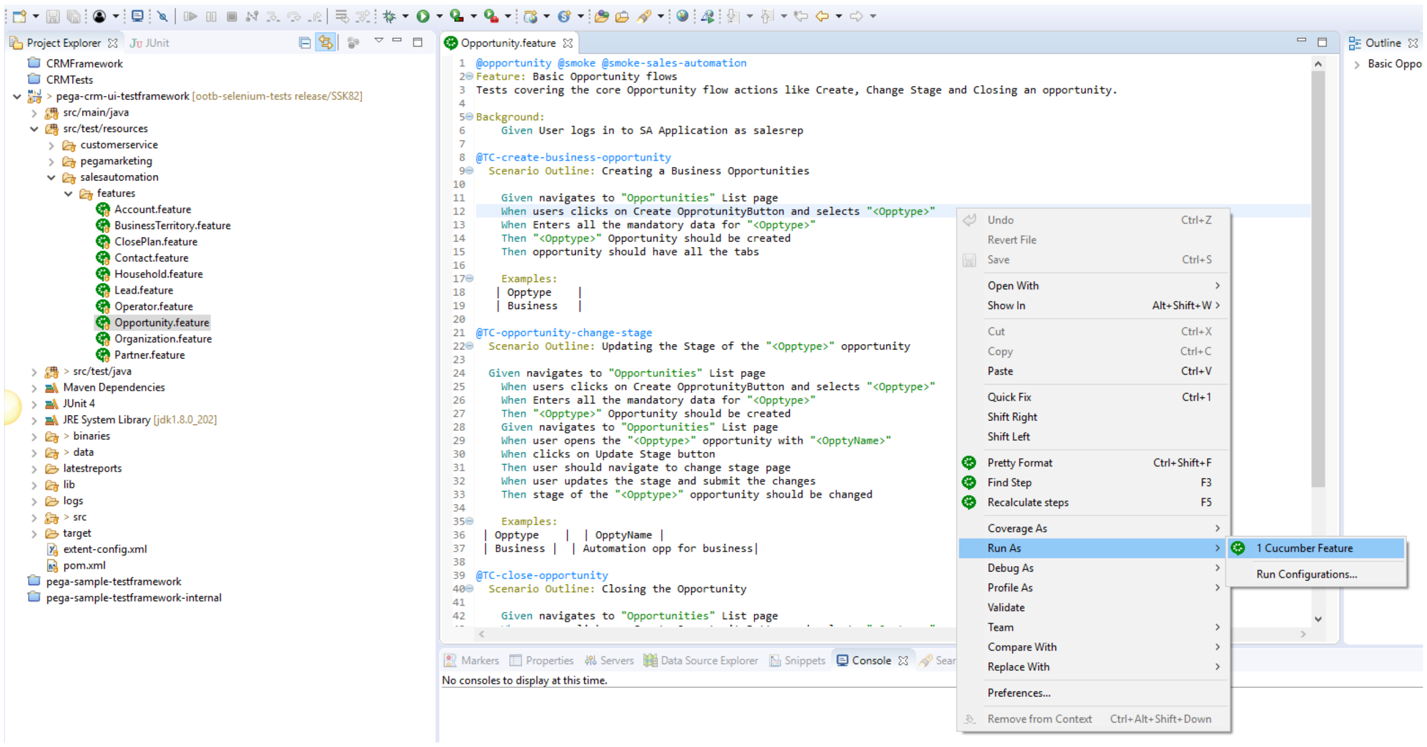
Overview | Dependencies | Dependency Hierarchy | Effective POM | pom.xml

5. Click OK
6. Run tests as usual using Maven commands

Tags can be used to group both scenarios as well as features.

Running a Cucumber feature

For rapid iterative development, you can also run a selected Cucumber feature file directly as shown below:



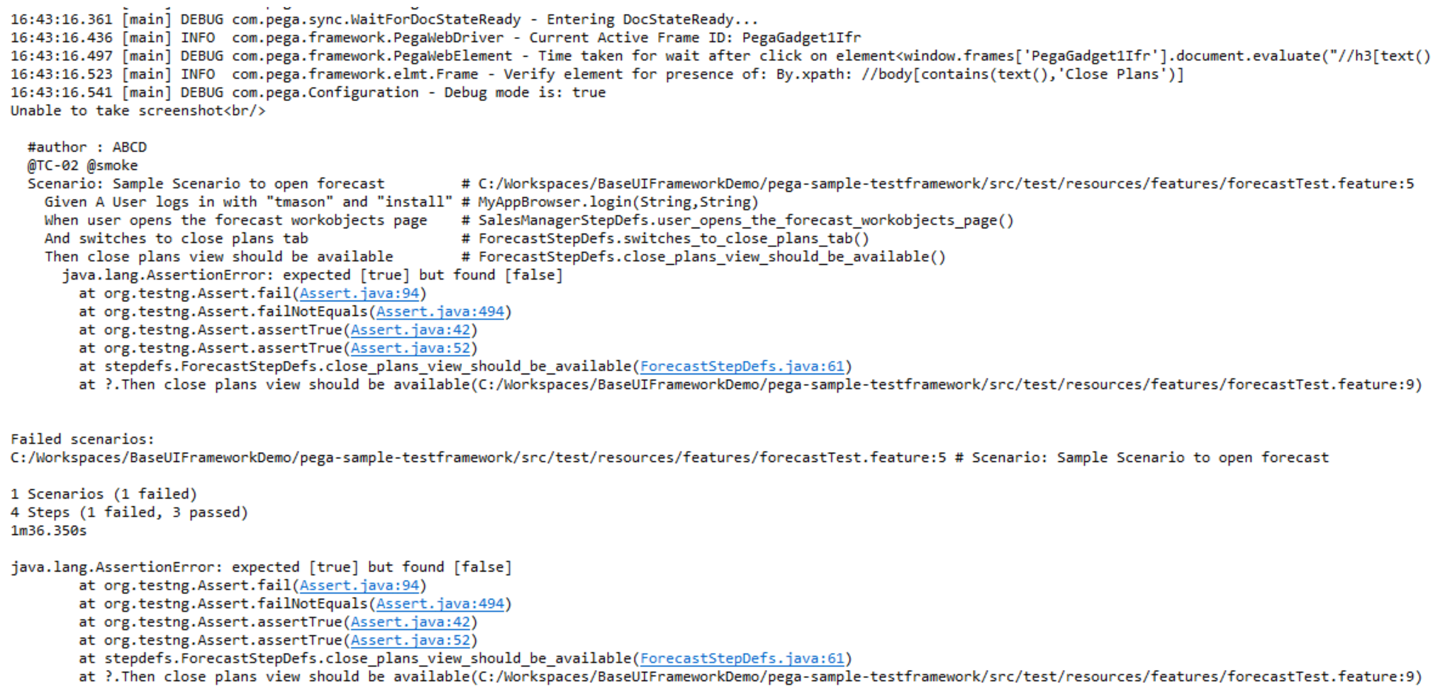
```
@opportunity @smoke @smoke-sales-automation
Feature: Basic Opportunity flows
Tests covering the core Opportunity flow actions like Create, Change Stage and Closing an opportunity.

Background:
    Given User logs in to SA Application as salesrep

@TC-create-business-opportunity
  Scenario Outline: Creating a Business Opportunities

    Given navigates to "Opportunities" List page
    When users clicks on Create OpprotunityButton and selects "<Opptype>"
    When Enters all the mandatory data for "<Opptype>"
    Then "<Opptype>" Opportunity should be created
    Then opportunity should have all the tabs

    Examples:
   | Opptype    |
   | Business   |

@TC-opportunity-change-stage
  Scenario Outline: Updating the Stage of the "<Opptype>" opportunity

  Given navigates to "Opportunities" List page
    When users clicks on Create OpprotunityButton and selects "<Opptype>"
    When Enters all the mandatory data for "<Opptype>"
    Then "<Opptype>" Opportunity should be created
    Given navigates to "Opportunities" List page
    When user opens the "<Opptype>" opportunity with "<OpptyName>"
    When clicks on Update Stage button
    Then user should navigate to change stage page
    When user updates the stage and submit the changes
    Then stage of the "<Opptype>" opportunity should be changed

    Examples:
 | Opptype    |  | OpptyName |
 | Business |  | Automation opp for business|

@TC-close-opportunity
  Scenario Outline: Closing the Opportunity

    Given navigates to "Opportunities" List page
```

```
No consoles to display at this time.
```

1. In the feature file to run, Right click and select **Run As > Cucumber Feature**.

This will trigger execution of all scenarios within that feature file sequentially. Running a Cucumber feature directly will produce the test result in the IDE console. A cucumber report will not be generated with this mode of execution.

Here is a sample console report for running tests as a cucumber feature



```
16:43:16.361 [main] DEBUG com.pega.sync.WaitForDocStateReady - Entering DocStateReady...
16:43:16.436 [main] INFO  com.pega.framework.PegaWebDriver - Current Active Frame ID: PegaGadget1Ifr
16:43:16.497 [main] DEBUG com.pega.framework.PegaWebElement - Time taken for wait after click on element<window.frames['PegaGadget1Ifr'].document.evaluate("//h3[text()
16:43:16.523 [main] INFO  com.pega.framework.elmt.Frame - Verify element for presence of: By.xpath: //body[contains(text(),'Close Plans')]
16:43:16.541 [main] DEBUG com.pega.Configuration - Debug mode is: true
Unable to take screenshot<br/>

  #author : ABCD
  @TC-02 @smoke
  Scenario: Sample Scenario to open forecast         # C:/Workspaces/BaseUIFrameworkDemo/pega-sample-testframework/src/test/resources/features/forecastTest.feature:5
    Given A User logs in with "tmason" and "install" # MyAppBrowser.login(String,String)
    When user opens the forecast workobjects page    # SalesManagerStepDefs.user_opens_the_forecast_workobjects_page()
    And switches to close plans tab                  # ForecastStepDefs.switches_to_close_plans_tab()
    Then close plans view should be available        # ForecastStepDefs.close_plans_view_should_be_available()
      java.lang.AssertionError: expected [true] but found [false]
        at org.testng.Assert.fail(Assert.java:94)
        at org.testng.Assert.failNotEquals(Assert.java:494)
        at org.testng.Assert.assertTrue(Assert.java:42)
        at org.testng.Assert.assertTrue(Assert.java:52)
        at stepdefs.ForecastStepDefs.close_plans_view_should_be_available(ForecastStepDefs.java:61)
        at ?.Then close plans view should be available(C:/Workspaces/BaseUIFrameworkDemo/pega-sample-testframework/src/test/resources/features/forecastTest.feature:9)


Failed scenarios:
C:/Workspaces/BaseUIFrameworkDemo/pega-sample-testframework/src/test/resources/features/forecastTest.feature:5 # Scenario: Sample Scenario to open forecast

1 Scenarios (1 failed)
4 Steps (1 failed, 3 passed)
1m36.350s

java.lang.AssertionError: expected [true] but found [false]
        at org.testng.Assert.fail(Assert.java:94)
        at org.testng.Assert.failNotEquals(Assert.java:494)
        at org.testng.Assert.assertTrue(Assert.java:42)
        at org.testng.Assert.assertTrue(Assert.java:52)
        at stepdefs.ForecastStepDefs.close_plans_view_should_be_available(ForecastStepDefs.java:61)
        at ?.Then close plans view should be available(C:/Workspaces/BaseUIFrameworkDemo/pega-sample-testframework/src/test/resources/features/forecastTest.feature:9)
```

Test Results

At the end of test execution, Eclipse Console window shows the test result summary as follows:

```
3 Scenarios (2 failed, 1 passed)
19 Steps (2 failed, 5 skipped, 12 passed)
7.702s
```

Interpreting Test Results Summary

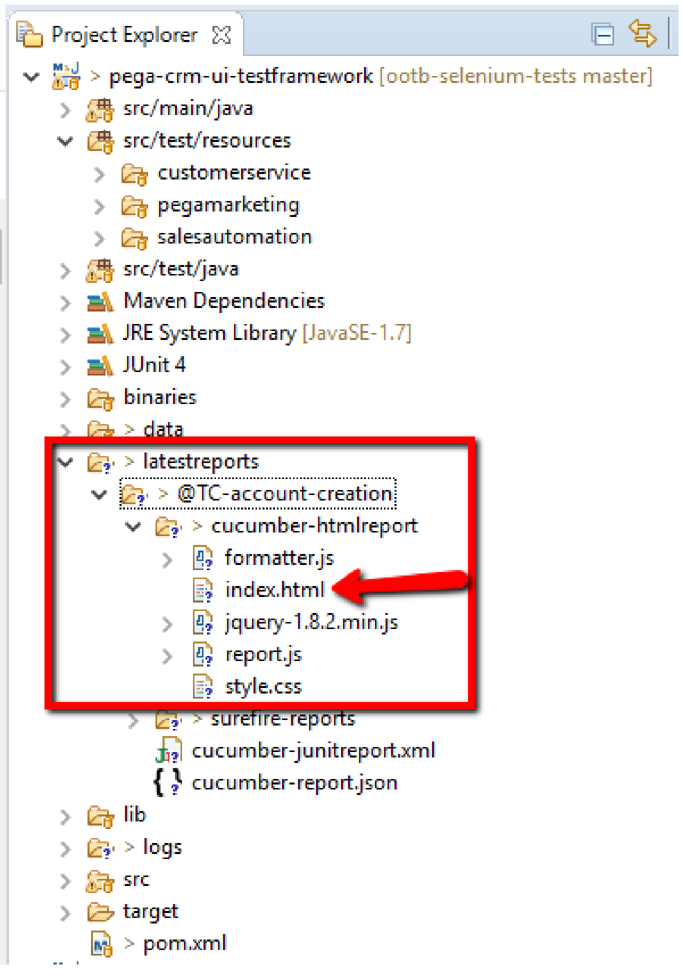
- "`3 Scenarios`" reflects the 3 scenarios that are tagged with @opportunities tag
- "`19 Steps`" reflects the total number of steps across all scenarios being tested
- `failed, skipped, passed` reflect the status of scenario execution

Test Report

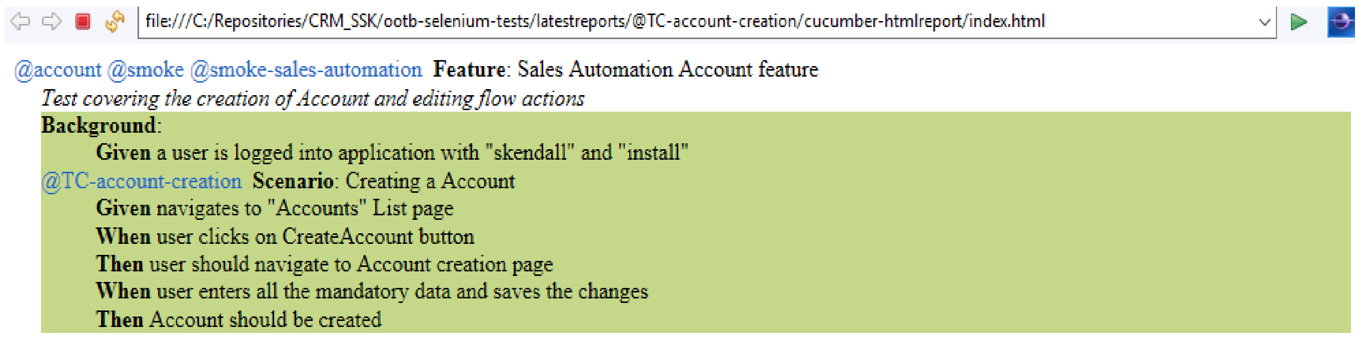
In addition to the results summary, test reports are produced for better visualization and analysis of test results.

Cucumber HTML Report

Cucumber plugin for Eclipse enables producing a test report at the end of test execution. As defined in the pom file, an HTML report is generated and placed in latestreports/`cucumber-htmlreport` folder.



`latestreports/cucumber-htmlreport/index.html` is the generated HTML report. Results are also available in `junit xml` and `json` formats – `cucumber-junitreport.xml and cucumber-report.json`


file:///C:/Repositories/CRM_SSK/ootb-selenium-tests/latestreports/@TC-account-creation/cucumber-htmlreport/index.html

@account @smoke @smoke-sales-automation **Feature**: Sales Automation Account feature

Test covering the creation of Account and editing flow actions

Background:

Given a user is logged into application with "skendall" and "install"

@TC-account-creation **Scenario**: Creating a Account

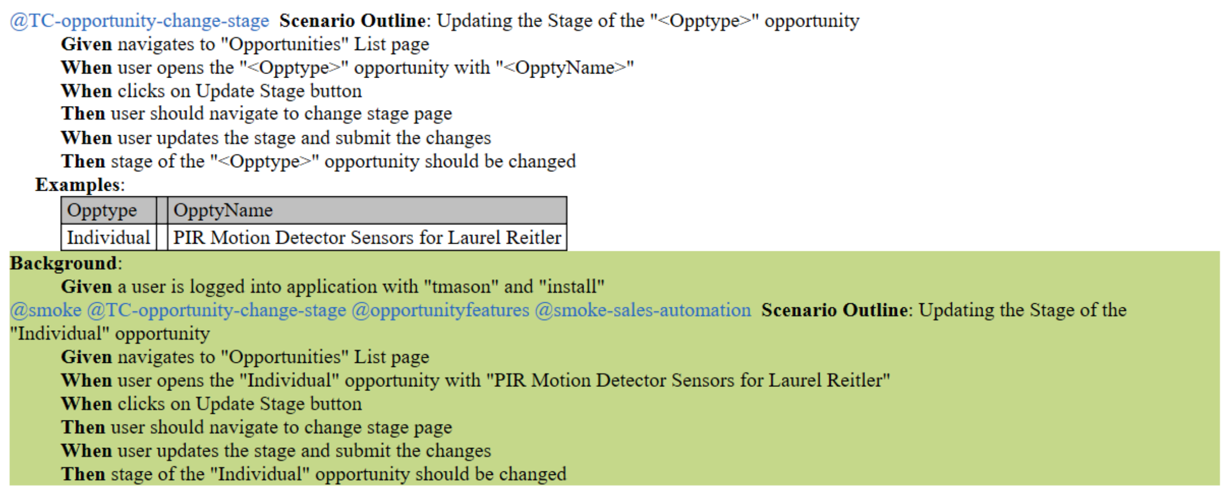
Given navigates to "Accounts" List page

When user clicks on CreateAccount button

Then user should navigate to Account creation page

When user enters all the mandatory data and saves the changes

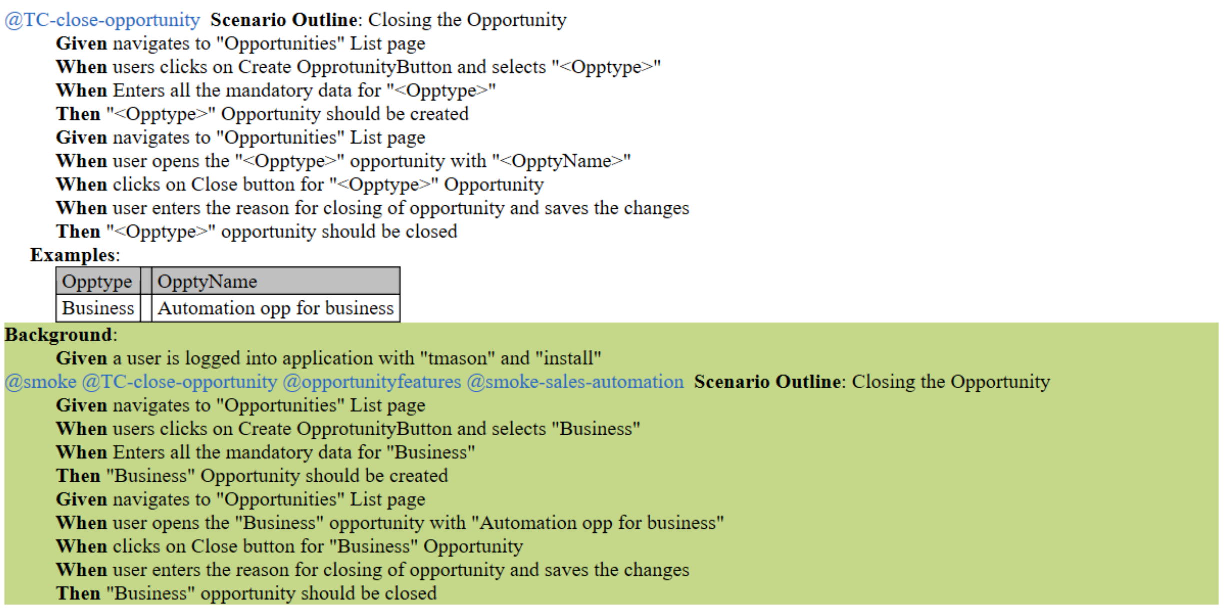
Then Account should be created



@TC-opportunity-change-stage **Scenario Outline**: Updating the Stage of the "<Opptype>" opportunity
Given navigates to "Opportunities" List page
When user opens the "<Opptype>" opportunity with "<OpptyName>"
When clicks on Update Stage button
Then user should navigate to change stage page
When user updates the stage and submit the changes
Then stage of the "<Opptype>" opportunity should be changed
Examples:

Opptype	OpptyName
Individual	PIR Motion Detector Sensors for Laurel Reitler

Background:
Given a user is logged into application with "tmason" and "install"
@smoke @TC-opportunity-change-stage @opportunityfeatures @smoke-sales-automation **Scenario Outline**: Updating the Stage of the "Individual" opportunity
Given navigates to "Opportunities" List page
When user opens the "Individual" opportunity with "PIR Motion Detector Sensors for Laurel Reitler"
When clicks on Update Stage button
Then user should navigate to change stage page
When user updates the stage and submit the changes
Then stage of the "Individual" opportunity should be changed



@TC-close-opportunity **Scenario Outline**: Closing the Opportunity
Given navigates to "Opportunities" List page
When users clicks on Create OpprotunityButton and selects "<Opptype>"
When Enters all the mandatory data for "<Opptype>"
Then "<Opptype>" Opportunity should be created
Given navigates to "Opportunities" List page
When user opens the "<Opptype>" opportunity with "<OpptyName>"
When clicks on Close button for "<Opptype>" Opportunity
When user enters the reason for closing of opportunity and saves the changes
Then "<Opptype>" opportunity should be closed
Examples:

Opptype	OpptyName
Business	Automation opp for business

Background:
Given a user is logged into application with "tmason" and "install"
@smoke @TC-close-opportunity @opportunityfeatures @smoke-sales-automation **Scenario Outline**: Closing the Opportunity
Given navigates to "Opportunities" List page
When users clicks on Create OpprotunityButton and selects "Business"
When Enters all the mandatory data for "Business"
Then "Business" Opportunity should be created
Given navigates to "Opportunities" List page
When user opens the "Business" opportunity with "Automation opp for business"
When clicks on Close button for "Business" Opportunity
When user enters the reason for closing of opportunity and saves the changes
Then "Business" opportunity should be closed

Interpreting Cucumber HTML Test Report

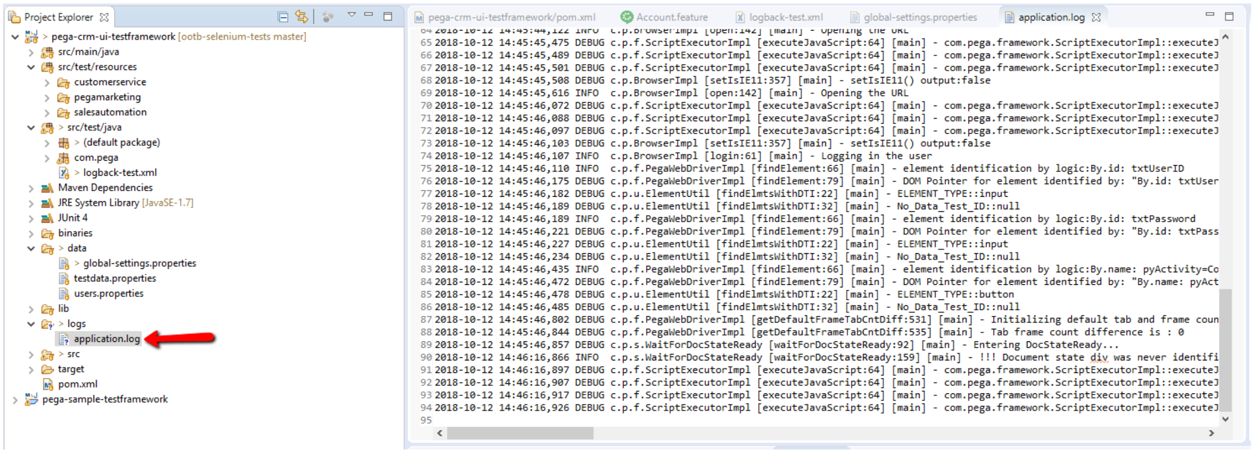
The HTML reports shows

1. Features & scenarios being tested along with their associated tags
2. Color-coded statuses for scenarios/steps executed
   a. Green – successful execution of the scenario/step (passed)
   b. Red – failed execution of the scenario/step (failed)
   c. Blue – skipped execution of a step (skipped)

Test Logs

Logging is enabled in the framework via logback classic framework. An xml file logback-test.xml is available in src/test/java source folder. Logging levels can be controlled via this xml file.

The default log level is debug (<root level="debug">) which can be changed to info/warn/error level to minimize the amount of logs displayed to the console. A copy of these logs are also saved to application.log file under logs folder as displayed below.



```
2018-10-12 14:45:44,122 INFO  c.p.BrowserImpl [open:142] [main] - Opening the URL
2018-10-12 14:45:45,475 DEBUG c.p.f.ScriptExecutorImpl [executeJavaScript:64] [main] - com.pega.framework.ScriptExecutorImpl::executeJ
2018-10-12 14:45:45,489 DEBUG c.p.f.ScriptExecutorImpl [executeJavaScript:64] [main] - com.pega.framework.ScriptExecutorImpl::executeJ
2018-10-12 14:45:45,501 DEBUG c.p.f.ScriptExecutorImpl [executeJavaScript:64] [main] - com.pega.framework.ScriptExecutorImpl::executeJ
2018-10-12 14:45:45,508 DEBUG c.p.BrowserImpl [setIsIE11:357] [main] - setIsIE11() output:false
2018-10-12 14:45:45,616 INFO  c.p.BrowserImpl [open:142] [main] - Opening the URL
2018-10-12 14:45:46,072 DEBUG c.p.f.ScriptExecutorImpl [executeJavaScript:64] [main] - com.pega.framework.ScriptExecutorImpl::executeJ
2018-10-12 14:45:46,088 DEBUG c.p.f.ScriptExecutorImpl [executeJavaScript:64] [main] - com.pega.framework.ScriptExecutorImpl::executeJ
2018-10-12 14:45:46,097 DEBUG c.p.f.ScriptExecutorImpl [executeJavaScript:64] [main] - com.pega.framework.ScriptExecutorImpl::executeJ
2018-10-12 14:45:46,103 DEBUG c.p.BrowserImpl [setIsIE11:357] [main] - setIsIE11() output:false
2018-10-12 14:45:46,107 INFO  c.p.BrowserImpl [login:61] [main] - Logging in the user
2018-10-12 14:45:46,110 INFO  c.p.f.PegaWebDriverImpl [findElement:66] [main] - element identification by logic:By.id: txtUserID
2018-10-12 14:45:46,175 DEBUG c.p.f.PegaWebDriverImpl [findElement:79] [main] - DOM Pointer for element identified by: "By.id: txtUser
2018-10-12 14:45:46,182 DEBUG c.p.u.ElementUtil [findElmtsWithDTI:22] [main] - ELEMENT_TYPE::input
2018-10-12 14:45:46,189 DEBUG c.p.u.ElementUtil [findElmtsWithDTI:32] [main] - No_Data_Test_ID::null
2018-10-12 14:45:46,189 INFO  c.p.f.PegaWebDriverImpl [findElement:66] [main] - element identification by logic:By.id: txtPassword
2018-10-12 14:45:46,221 DEBUG c.p.f.PegaWebDriverImpl [findElement:79] [main] - DOM Pointer for element identified by: "By.id: txtPass
2018-10-12 14:45:46,227 DEBUG c.p.u.ElementUtil [findElmtsWithDTI:22] [main] - ELEMENT_TYPE::input
2018-10-12 14:45:46,234 DEBUG c.p.u.ElementUtil [findElmtsWithDTI:32] [main] - No_Data_Test_ID::null
2018-10-12 14:45:46,435 INFO  c.p.f.PegaWebDriverImpl [findElement:66] [main] - element identification by logic:By.name: pyActivity=Co
2018-10-12 14:45:46,472 DEBUG c.p.f.PegaWebDriverImpl [findElement:79] [main] - DOM Pointer for element identified by: "By.name: pyAct
2018-10-12 14:45:46,478 DEBUG c.p.u.ElementUtil [findElmtsWithDTI:22] [main] - ELEMENT_TYPE::button
2018-10-12 14:45:46,485 DEBUG c.p.u.ElementUtil [findElmtsWithDTI:32] [main] - No_Data_Test_ID::null
2018-10-12 14:45:46,802 DEBUG c.p.f.PegaWebDriverImpl [getDefaultFrameTabCntDiff:531] [main] - Initializing default tab and frame coun
2018-10-12 14:45:46,844 DEBUG c.p.f.PegaWebDriverImpl [getDefaultFrameTabCntDiff:535] [main] - Tab frame count difference is : 0
2018-10-12 14:45:46,857 DEBUG c.p.s.WaitForDocStateReady [waitForDocStateReady:92] [main] - Entering DocStateReady...
2018-10-12 14:46:16,866 INFO  c.p.s.WaitForDocStateReady [waitForDocStateReady:159] [main] - !!! Document state div was never identifi
2018-10-12 14:46:16,897 DEBUG c.p.f.ScriptExecutorImpl [executeJavaScript:64] [main] - com.pega.framework.ScriptExecutorImpl::executeJ
2018-10-12 14:46:16,907 DEBUG c.p.f.ScriptExecutorImpl [executeJavaScript:64] [main] - com.pega.framework.ScriptExecutorImpl::executeJ
2018-10-12 14:46:16,917 DEBUG c.p.f.ScriptExecutorImpl [executeJavaScript:64] [main] - com.pega.framework.ScriptExecutorImpl::executeJ
2018-10-12 14:46:16,926 DEBUG c.p.f.ScriptExecutorImpl [executeJavaScript:64] [main] - com.pega.framework.ScriptExecutorImpl::executeJ
```

Failure Diagnosis

To illustrate diagnosing failures, we will force a failure by perturbing the expected value of the “`Search by organization`” scenario in Sales Automation’s `ClosePlan.feature`

```
      Scenario: Search by organization

        Given a sales rep is at the Close Plans page

        When the rep searches for "APW Technologies Corp" organization

        Then opportunities related only to "APW Inc" are shown
```

Forced Failure

When a test fails, there are multiple diagnostics that help identify and debug the failure.

- Console Output
  - Results Summary
  - Failure Stack
- Test Reports
- Debug Options

Test Log & Results Summary

```
Scenario: Search by organization # salesautomation/features/ClosePlan.feature:6

  Given a sales rep is at the Close Plans page #
ForecastClosePlans.a_sales_rep_is_at_the_Close_Plans_page()

  When the rep searches for "APW Technologies Corp" organization #
ForecastClosePlans.the_rep_searches_for_organization(String)

  Then opportunities related only to "APW Inc" are shown #
ForecastClosePlans.relevant_opportunities_are_shown(String)
```

Failure Message

```
     java.lang.AssertionError: Expected organization 'APW Inc' not found. expected [true] but found
[false]

        at org.testng.Assert.fail(Assert.java:94)

        at org.testng.Assert.failNotEquals(Assert.java:494)

        at org.testng.Assert.assertTrue(Assert.java:42)

        at
com.pega.crm.salesautomation.stepdefs.ForecastClosePlans.relevant_opportunities_are_shown(ForecastClosePla
ns.java:65)

        at @Then opportunities related only to "APW Inc" are
shown(salesautomation/features/ClosePlan.feature:9)
```

Failure Stack

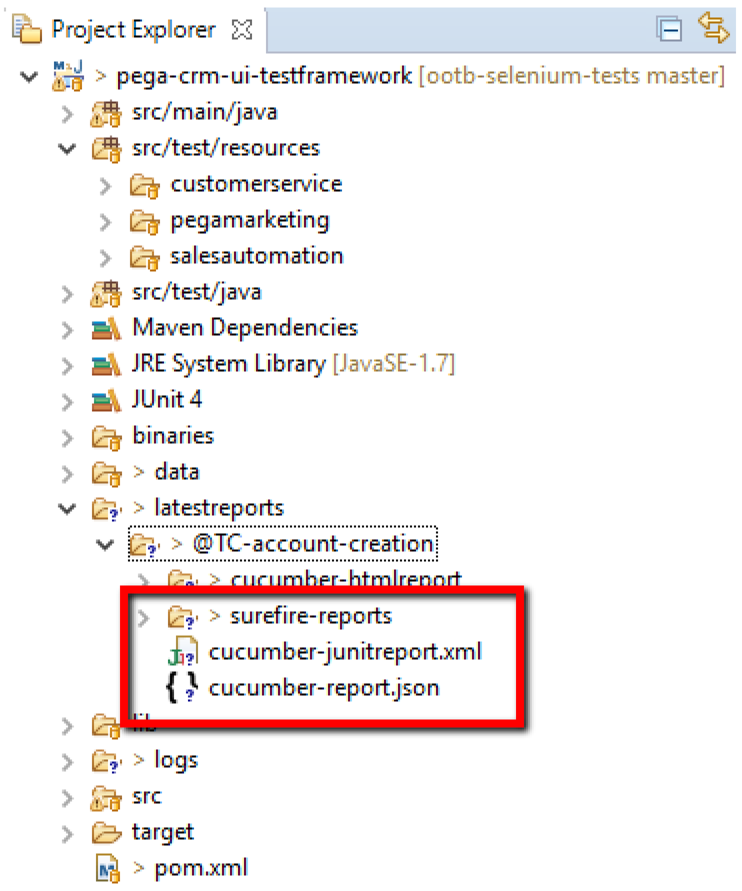
```
 Failed scenarios:

        salesautomation/features/ClosePlan.feature:6 # Scenario: Search by organization
```

```
1 Scenarios (1 failed)

3 Steps (1 failed, 2 passed)

0m36.495s

...

Results :


Failed tests:

  RunCukesTest>AbstractTestNGCucumberTests.run:19->AbstractTestNGCucumberTests.run_cukes:14 » Cucumber

Tests run: 1, Failures: 1, Errors: 0, Skipped: 0
```

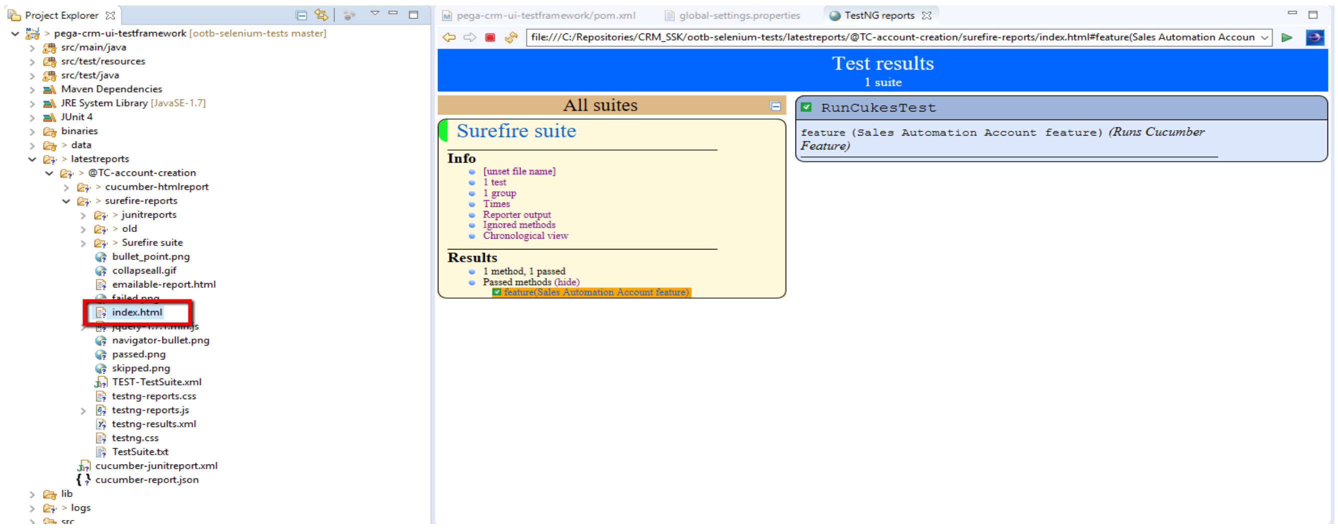
Result Summary

Test Report

In addition to the test failure log displayed in the console window, Eclipse’s Maven surefire and Cucumber plugins produces artifacts that highlight test failures



The surefire-reports includes a main report, `index.html` and an e-mailable report, `emailable-report.html`



file:///C:/Repositories/CRM_SSK/ootb-selenium-tests/latestreports/@TC-account-creation/surefire-reports/index.html#feature(Sales Automation Accoun

Test results
1 suite

All suites

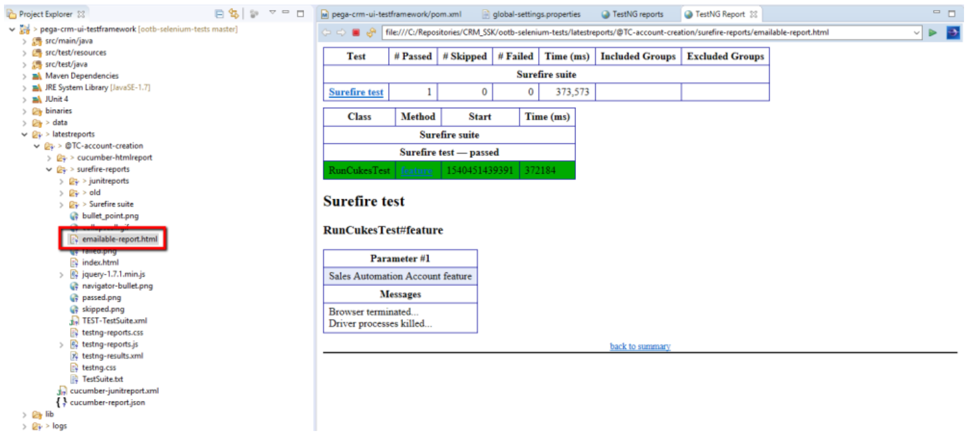
Surefire suite

Info
- [unset file name]
- 1 test
- 1 group
- Times
- Reporter output
- Ignored methods
- Chronological view

Results
- 1 method, 1 passed
- Passed methods (hide)
  - ✓ feature(Sales Automation Account feature)

✓ RunCukesTest

feature (Sales Automation Account feature) *(Runs Cucumber Feature)*



```
Project Explorer
v > pega-crm-ui-testframework [ootb-selenium-tests master]
  > src/main/java
  > src/test/resources
  > src/test/java
  > Maven Dependencies
  > JRE System Library [JavaSE-1.7]
  > JUnit 4
  > binaries
  > > data
  v > latestreports
    v > @TC-account-creation
      > > cucumber-htmlreport
      v > surefire-reports
        > > junitreports
        > > old
        > > Surefire suite
          bullet_point.png
          emailable-report.html
          index.html
        > jquery-1.7.1.min.js
          navigator-bullet.png
          passed.png
          skipped.png
          TEST-TestSuite.xml
          testng-reports.css
        > testng-reports.js
          testng-results.xml
          testng.css
          TestSuite.txt
        cucumber-junitreport.xml
        cucumber-report.json
  > lib
  > > logs
```

pega-crm-ui-testframework/pom.xml | global-settings.properties | TestNG reports | TestNG Report

file:///C:/Repositories/CRM_SSK/ootb-selenium-tests/latestreports/@TC-account-creation/surefire-reports/emailable-report.html

Test	# Passed	# Skipped	# Failed	Time (ms)	Included Groups	Excluded Groups
Surefire suite						
Surefire test	1	0	0	373,573		

Class	Method	Start	Time (ms)
Surefire suite			
Surefire test — passed			
RunCukesTest	feature	1540451439391	372184

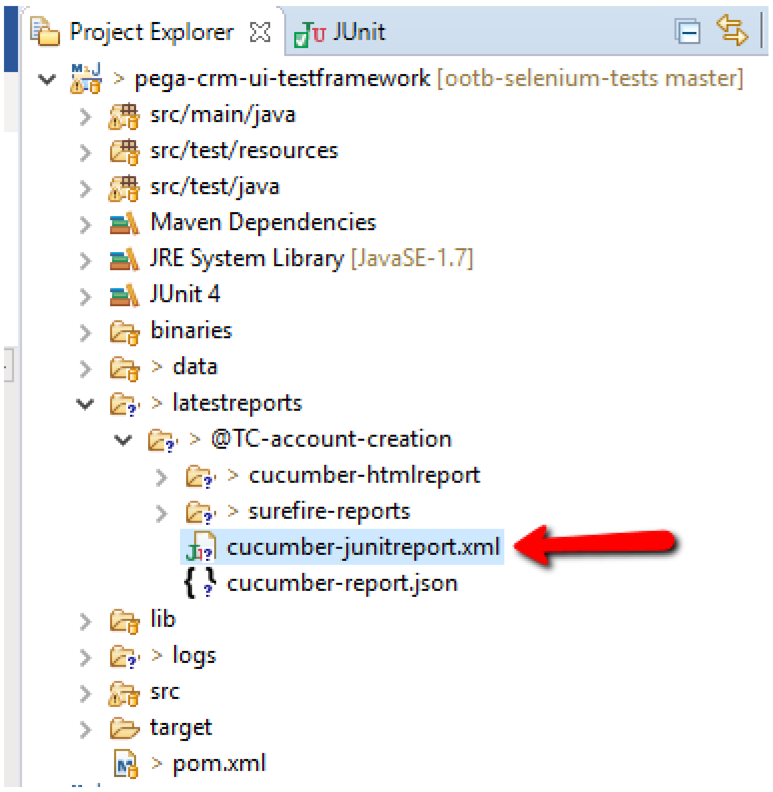
Surefire test

RunCukesTest#feature

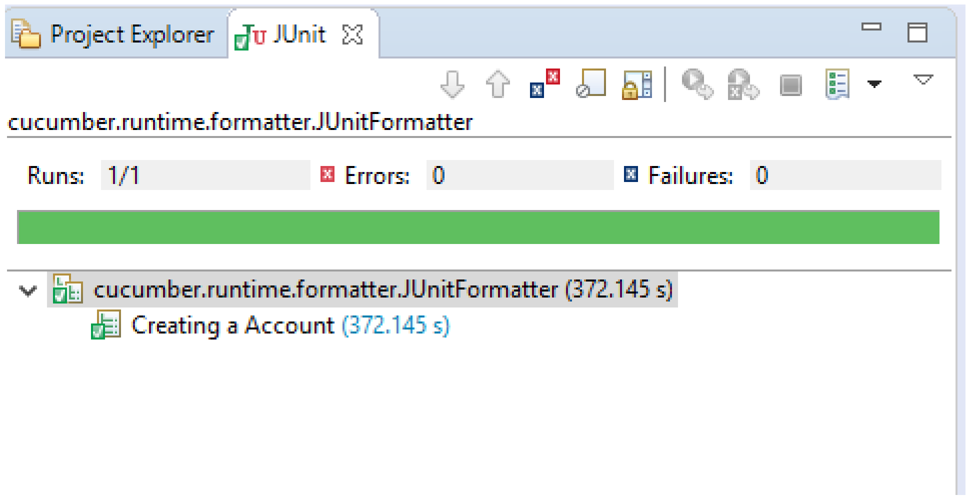
Parameter #1
Sales Automation Account feature
Messages
Browser terminated... Driver processes killed...

back to summary

The Cucumber plugin produces a Junit report



```
Project Explorer   JUnit
v > pega-crm-ui-testframework [ootb-selenium-tests master]
  > src/main/java
  > src/test/resources
  > src/test/java
  > Maven Dependencies
  > JRE System Library [JavaSE-1.7]
  > JUnit 4
  > binaries
  > > data
  v > latestreports
    v > @TC-account-creation
      > > cucumber-htmlreport
      > > surefire-reports
        cucumber-junitreport.xml
        cucumber-report.json
  > lib
  > > logs
  > src
  > target
    > pom.xml
```



cucumber.runtime.formatter.JUnitFormatter

Runs: 1/1 Errors: 0 Failures: 0

cucumber.runtime.formatter.JUnitFormatter (372.145 s)
Creating a Account (372.145 s)

Note:

- Artifacts produced by Maven surefire and Cucumber plugin are subject to change by as defined by those 3rd party plugins.

Debugging

The project provides the following diagnostic capabilities assist with failure debugging:

Debug Mode

The debug mode enables you to diagnose the problem when a UI test fails.

When the project setting **debug.mode** in **data/global-settings.properties** is set to **true**, this will keep the application & browser open when a test fails. This allows you diagnose the application at the point the test failed.

Screenshot

When a test fails, the framework automatically takes a screenshot at the failure point that can provide insight and helps with defect localization.

Managing Timeouts

Sometimes, a page or UI element does not load, and the test is stuck indefinitely until the test is aborted.

The project setting **global.timeout** in **data/global-settings.properties** allows you to specify the maximum time in seconds the test waits for a page or UI element to load. When this time is exceeded, the test is aborted and marked failed.

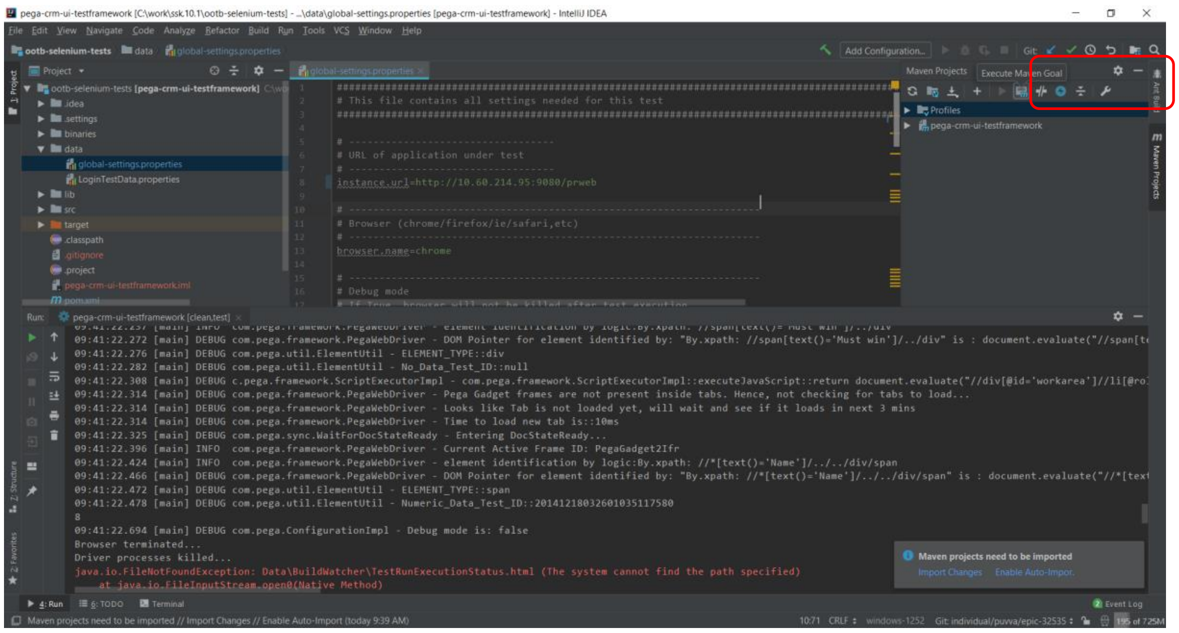
You want to set this time to a reasonable value, like 30 seconds. Setting this value high, e.g. minutes, can have an effect on the performance of your tests. For example, if there is a systemic issue in your application and every other UI element is not loading, your tests will wait the maximum time before it aborts a test. This wait time adds up when you are running hundreds of tests.

IntelliJ Tips

This section calls out a few aspects related to working with IntelliJ IDE

Running Tests

To trigger test execution the Maven Way, execute the Maven goal as follows:



pega-crm-ui-testframework [C:\work\ssk.10.1\ootb-selenium-tests] - ...\data\global-settings.properties [pega-crm-ui-testframework] - IntelliJ IDEA

```
#########################################################################################
# This file contains all settings needed for this test
#########################################################################################

# ---------------------------------
# URL of application under test
# ---------------------------------
instance.url=http://10.60.214.95:9080/prweb

# -----------------------------------------------------------------------
# Browser (chrome/firefox/ie/safari,etc)
# -----------------------------------------------------------------------
browser.name=chrome

# -----------------------------------------------------------------------
# Debug mode
```

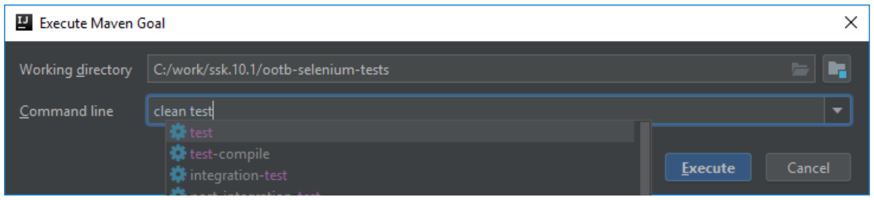
Run: pega-crm-ui-testframework [clean,test]

```
09:41:22.272 [main] DEBUG com.pega.framework.PegaWebDriver - DOM Pointer for element identified by: "By.xpath: //span[text()='Must win']/../div" is : document.evaluate("//span[te
09:41:22.276 [main] DEBUG com.pega.util.ElementUtil - ELEMENT_TYPE::div
09:41:22.282 [main] DEBUG com.pega.util.ElementUtil - No_Data_Test_ID::null
09:41:22.308 [main] DEBUG c.pega.framework.ScriptExecutorImpl - com.pega.framework.ScriptExecutorImpl::executeJavaScript::return document.evaluate("//div[@id='workarea']//li[@ro
09:41:22.314 [main] DEBUG com.pega.framework.PegaWebDriver - Pega Gadget frames are not present inside tabs. Hence, not checking for tabs to load...
09:41:22.314 [main] DEBUG com.pega.framework.PegaWebDriver - Looks like Tab is not loaded yet, will wait and see if it loads in next 3 mins
09:41:22.314 [main] DEBUG com.pega.framework.PegaWebDriver - Time to load new tab is::10ms
09:41:22.325 [main] DEBUG com.pega.sync.WaitForDocStateReady - Entering DocStateReady...
09:41:22.396 [main] INFO  com.pega.framework.PegaWebDriver - Current Active Frame ID: PegaGadget2Ifr
09:41:22.424 [main] INFO  com.pega.framework.PegaWebDriver - element identification by logic:By.xpath: //*[text()='Name']/../../div/span
09:41:22.466 [main] DEBUG com.pega.framework.PegaWebDriver - DOM Pointer for element identified by: "By.xpath: //*[text()='Name']/../../div/span" is : document.evaluate("//*[text
09:41:22.472 [main] DEBUG com.pega.util.ElementUtil - ELEMENT_TYPE::span
09:41:22.478 [main] DEBUG com.pega.util.ElementUtil - Numeric_Data_Test_ID::20141218032601035117580
8
09:41:22.694 [main] DEBUG com.pega.ConfigurationImpl - Debug mode is: false
Browser terminated...
Driver processes killed...
java.io.FileNotFoundException: Data\BuildWatcher\TestRunExecutionStatus.html (The system cannot find the path specified)
	at java.io.FileInputStream.open0(Native Method)
```

Maven projects need to be imported
Import Changes Enable Auto-Import

Maven projects need to be imported // Import Changes // Enable Auto-Import (today 9:39 AM)

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
ootb-selenium-tests
data
global-settings.properties
Add Configuration...
Project
ootb-selenium-tests [pega-crm-ui-testframework]
.idea
settings
binaries
global-settings.properties
LoginTestData.properties
lib
src
target
.classpath
.gitignore
.project
pega-crm-ui-testframework.iml
pom.xml
Maven Projects
Execute Maven Goal
Profiles
pega-crm-ui-testframework
4: Run
6: TODO
Terminal
Event Log
10:71 CRLF windows-1252 Git: individual/puvva/epic-32535



Execute Maven Goal

Working directory: C:/work/ssk.10.1/ootb-selenium-tests

Command line: clean test

test
test-compile
integration-test
Execute
Cancel

Related Documentation

- Running tests in CI/CD pipeline
- Writing new tests

References

- Behavior Driven Development with Cucumber