

BDD Framework Design Document

**Version - Initial Draft**

**Version Date: 06/20/2019**

**Author: Aayushi Agrawal**

# **Project Details**

|  |  |
| --- | --- |
| Project Name | **BDD Framework** |
| Author(s) | Aayushi Agrawal |
| Manager | Rohan Patnaik |
| Contributor(s) | ACOE Team |
| First Issued On | 06/26/2019 |
| Document Location |  |

# **Privacy Information**

This document contains proprietary and confidential information of CVS Health. The recipient agrees to maintain this information in confidence and not reproduce or otherwise disclose this information to any person outside of the group directly responsible for the evaluation of its contents.

CVS Health

One CVS Drive

Woonsocket, RI 02895

Table of Contents

[**Project Details** 2](#_Toc12017825)

[**Privacy Information** 2](#_Toc12017826)

[**1** **Architecture and Design** 5](#_Toc12017827)

[**2** **Prerequisite Tools** 6](#_Toc12017828)

[**3** **Database Model** 6](#_Toc12017829)

[***3.1 Table: testdata\_cvs*** 6](#_Toc12017830)

[***3.2 Table: object\_repository\_desktop\_cvs*** 7](#_Toc12017831)

[**4** **Scenario Creation Flow** 7](#_Toc12017832)

[***4.1*** ***Identify the Feature package where we need to create a new feature file.*** 7](#_Toc12017833)

[**4.1.1** **Create feature packages as per modules for easy segregation** 7](#_Toc12017834)

[**4.1.2** **Create Feature file within the identified package as per sub module** 8](#_Toc12017835)

[***4.2*** ***Create the scenarios for the feature file using Gherkin language*** 9](#_Toc12017836)

[**4.2.1** **Identify the scenarios applicable for the feature file and write the scenario and steps in gherkin language.** 9](#_Toc12017837)

[***4.3*** ***Create the Step Definition file for the created feature file*** 9](#_Toc12017838)

[**4.3.1** **Execute the above feature file and generate the blank step defs** 9](#_Toc12017839)

[**4.3.2** **Handle Maven Build Known Errors** 10](#_Toc12017840)

[**4.3.3** **Post the run, the compiler will give us the error stating all the steps for which step definition is not already existing and needs to be created.** 11](#_Toc12017841)

[**4.3.4** **Create a new Java file under Step Definition package** 11](#_Toc12017842)

[**4.3.5** **Copy the missing steps snippet from step 3.b and paste the same in your StepDefinition File** 12](#_Toc12017843)

[***4.4*** ***Identify and create the xpaths for the objects to be used*** 12](#_Toc12017844)

[**4.4.1** **Update the OR (MaridDB: object\_repository\_desktop\_cvs)** 12](#_Toc12017845)

[***4.5*** ***Start to code for Step definition*** 13](#_Toc12017846)

[**4.5.1** **Extend the ExecutionContext.java to identify the objects** 13](#_Toc12017847)

[**4.5.2** **Set the xpath pair for usage in code** 13](#_Toc12017848)

[**4.5.3** **use OperationsDesktop for generic/custom methods** 13](#_Toc12017849)

[***4.6*** ***Exceptions are handled for all steps*** 13](#_Toc12017850)

[**4.6.1** **Ensure to write the code in try-catch block and handle the Exception using ExceptionHandler class** 13](#_Toc12017851)

[***4.7*** ***Use ReportUtility to log and generate test result reports*** 14](#_Toc12017852)

[**4.7.1** **Log Report for each step** 14](#_Toc12017853)

[**4.7.2** **Log Report for each handled exception** 14](#_Toc12017854)

[**5** **Execution Flow** 14](#_Toc12017855)

[***5.1*** ***Tag all the scenarios in the feature file*** 14](#_Toc12017856)

[**5.1.1** **Use unique/common tags for grouping the scenarios in a feature file** 14](#_Toc12017857)

[***5.2*** ***Update the Config.properties file*** 14](#_Toc12017858)

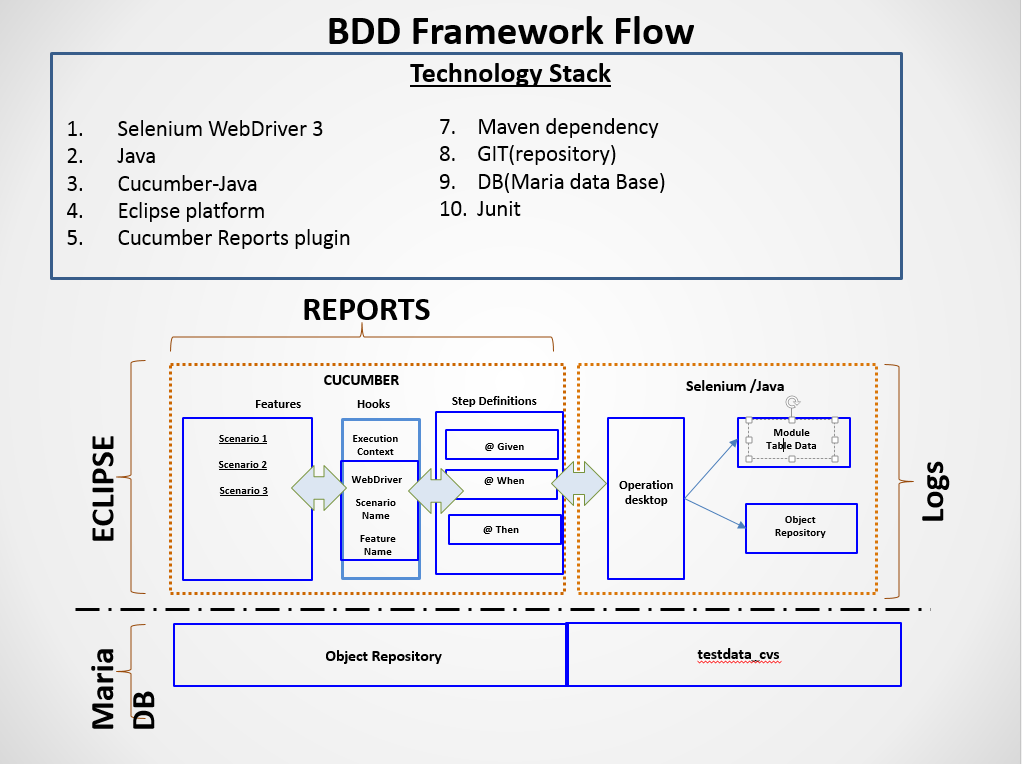
[***5.3*** ***Run feature file*** 15](#_Toc12017859)

[**5.3.1** **Execute the feature file using the command line statement** 15](#_Toc12017860)

[***5.4*** ***Validate Test Result reports*** 16](#_Toc12017861)

[**5.4.1** **Verify the results @ C:\Automation\_Enterprise\Selenium\digital\_bdd\_cucumber\target\generated-report** 16](#_Toc12017862)

# **Architecture and Design**



.Feature File:

In this file, we write a scenario in ‘Given, When, Then’ Format and provide Data Tables to read data

Hooks.java File:

In this file, we assign thread specific attributes to each thread uniquely. For Example:

Feature Name, Scenario Name, Iteration Number

Step Definition File:

In this file, we map the Scenario steps in our code using Cucumber’s generated regular expression. And provide implementation of that step using Selenium and Java

OpeartionDesktop.java File:

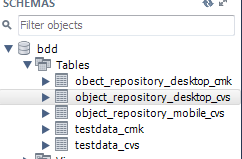
In this file we have provided methods to handle Web commands. For Example:

Click a button or link, WebPageLoad, Switch Frames, Retrieve Text from Input Box

# **Prerequisite Tools**

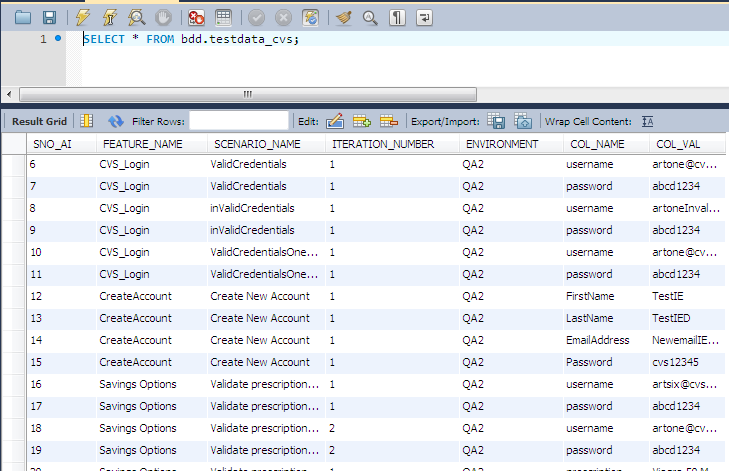
* JDK 1.8/JRE 8 32 bit version (please make sure its installed in C:\Program Files (x86) folder only)
* Eclipse Oxygen (or latest version)
* All Required jars (will be provided by ACoE Team)
* IE 11
* Google Chrome 70 or above
* MySQL Workbench 6.2

# **Database Model**



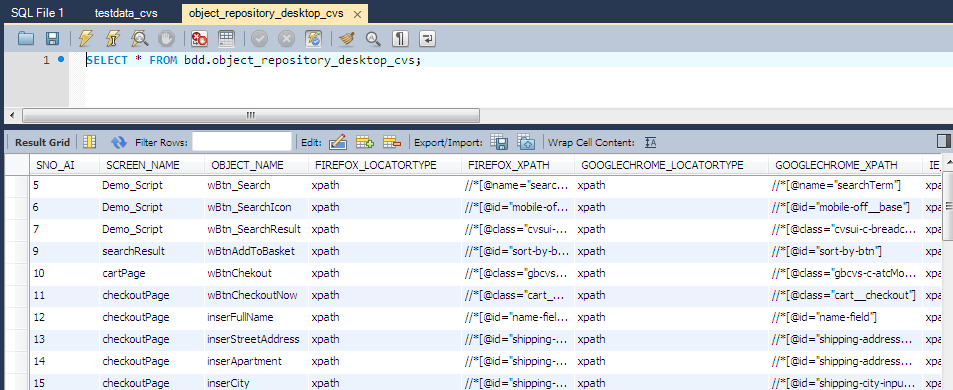
## ***3.1 Table: testdata\_cvs***

This table stores module data for particular Feature and Scenario combination



## ***3.2 Table: object\_repository\_desktop\_cvs***

This table stores all the xPath uniquely defined with Screen Name and Object Name

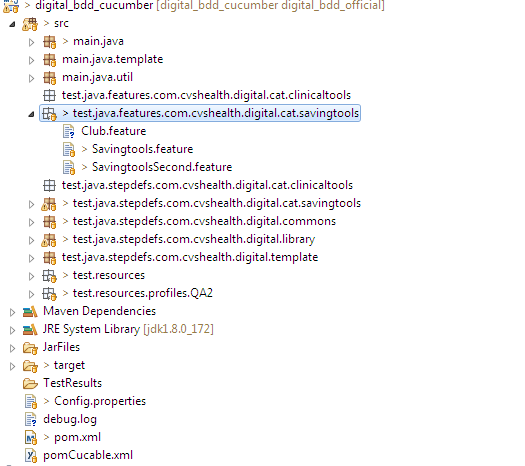


# **Scenario Creation Flow**

## ***Identify the Feature package where we need to create a new feature file.***

### **Create feature packages as per modules for easy segregation**

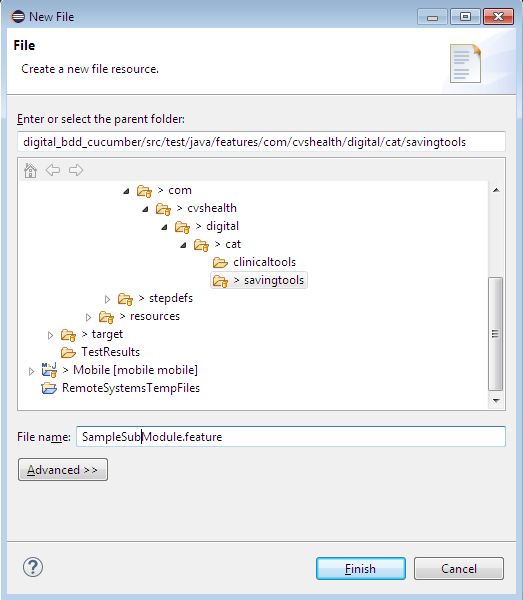
For example: test.java.features.com.cvshealth.digital.cat.savingtools ,Where in SavingTools is the Module name.



### **Create Feature file within the identified package as per sub module**

For Example: Savingtools.feature or SavingtoolsSecond.feature or SampleSubModule.feature

Note: create a new file with the extension as .feature



## ***Create the scenarios for the feature file using Gherkin language***

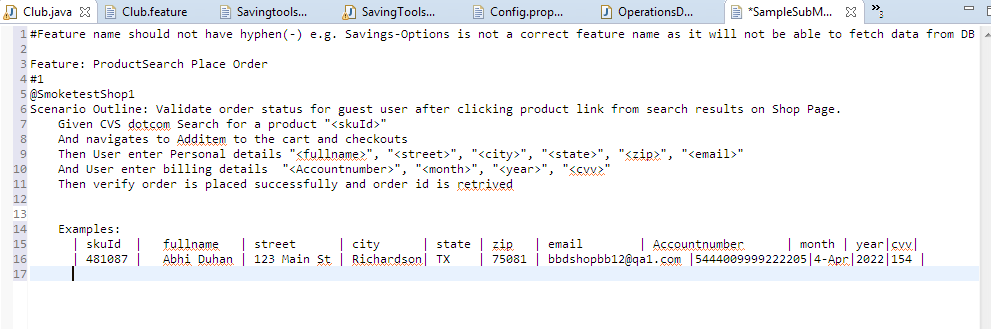
### **Identify the scenarios applicable for the feature file and write the scenario and steps in gherkin language.**

The steps should be written in Given, When and Then format:

Given: Pre-condition

When: action to be performed

Then: expected result from the actions performed

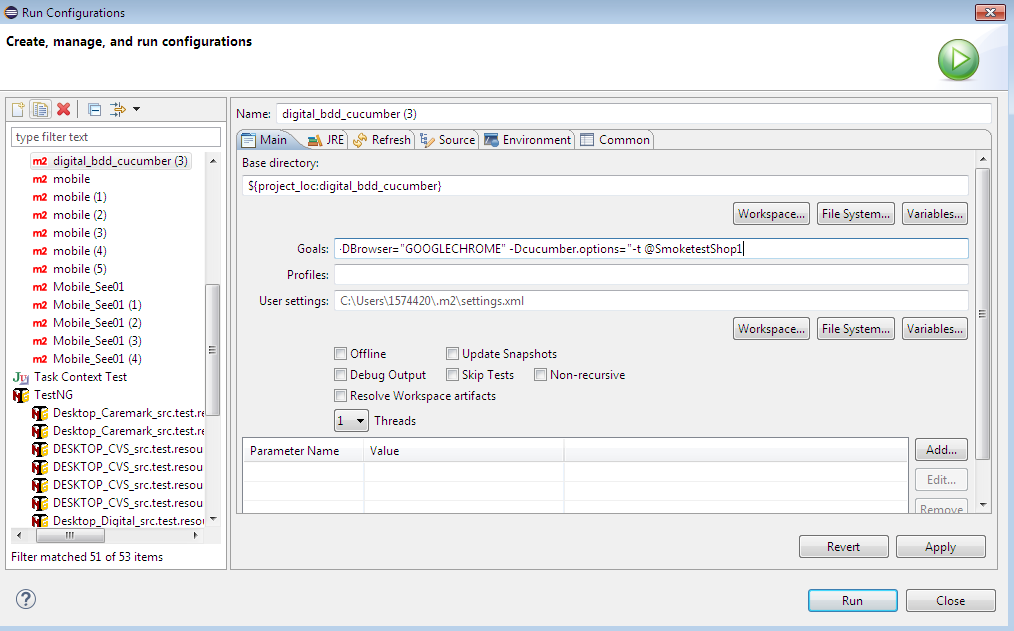


## ***Create the Step Definition file for the created feature file***

### **Execute the above feature file and generate the blank step defs**

Navigate to run configuration and enter the details of the tag to be executed:

**clean verify -P=QA1 -f pom.xml -DfeaturesFolder="src/test/java/features/" -DBrowser="GOOGLECHROME" -Dcucumber.options="-t @SmoketestShop1**

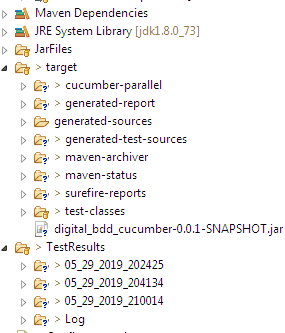


### **Handle Maven Build Known Errors**

If getting Build failure, Try taking below actions according to the error you are getting

If getting **Report Generator** error:

* + Try deleting all subfolders of **Target** Folder and **TestResults** folder. Save your useful reports at some other location first.
  + Make sure the tag you defined in build command: **Dcucumber.options="-t @smoke"** exists for at least one scenario



If getting **Mojo exception** or **SureFire plugin exception** or some other plugin exception:

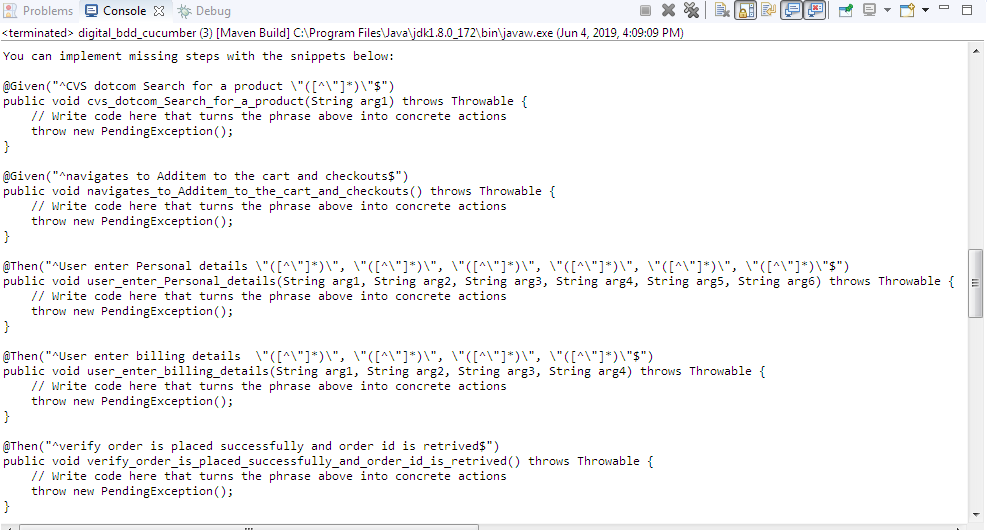
* + It means Jar files could not be downloaded successfully due to some network problem, try copying **.m2 repository** from someone else.

If getting **compiler Plugin** error

* + You might have used JRE instead of JDK. Go to build path-> Configure build Path and change library from JRE to JDK

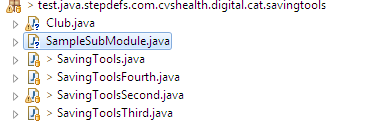
### **Post the run, the compiler will give us the error stating all the steps for which step definition is not already existing and needs to be created.**

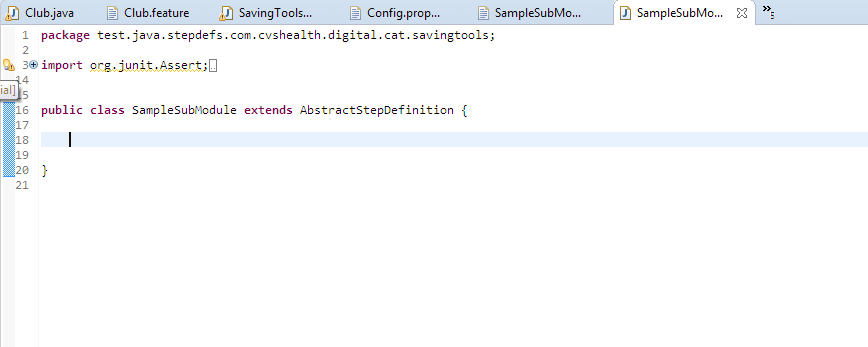
For example:



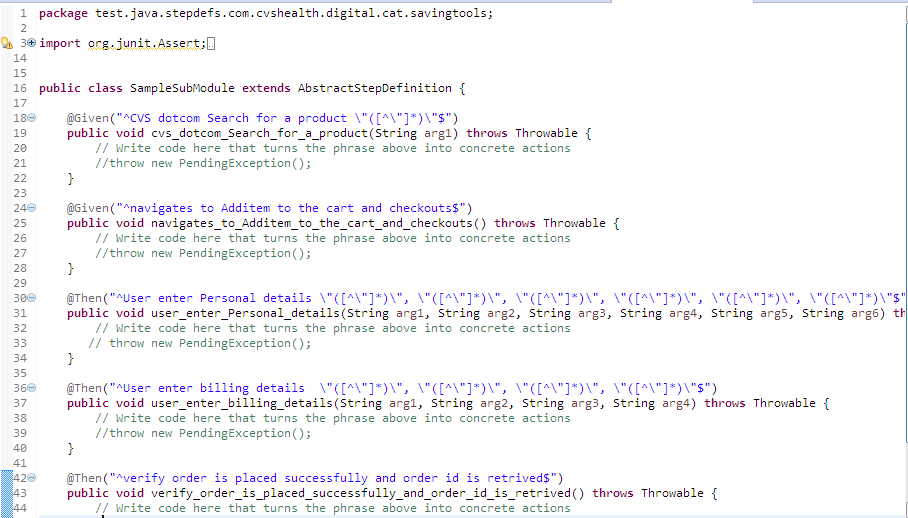
### **Create a new Java file under Step Definition package**

Preferably the same name as of feature file.



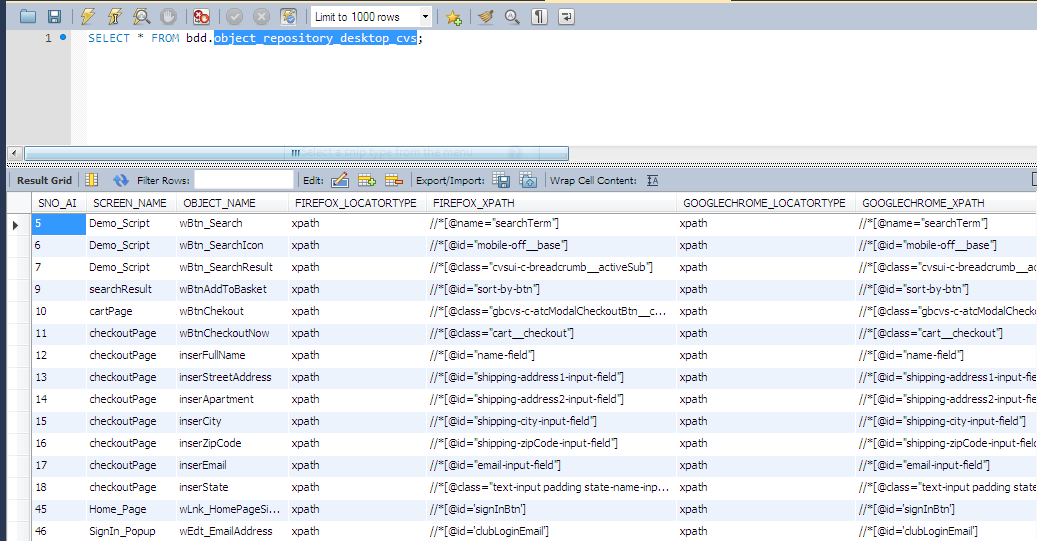


### **Copy the missing steps snippet from step 3.b and paste the same in your StepDefinition File**



## ***Identify and create the xpaths for the objects to be used***

### **Update the OR (MaridDB: object\_repository\_desktop\_cvs)**



Note: A unique combination of ScreenName and Object name to be used

## ***Start to code for Step definition***

### **Extend the ExecutionContext.java to identify the objects**

For Example:

String SearchForSavingsxPath = ExecutionContext.*getObjectLocator*(

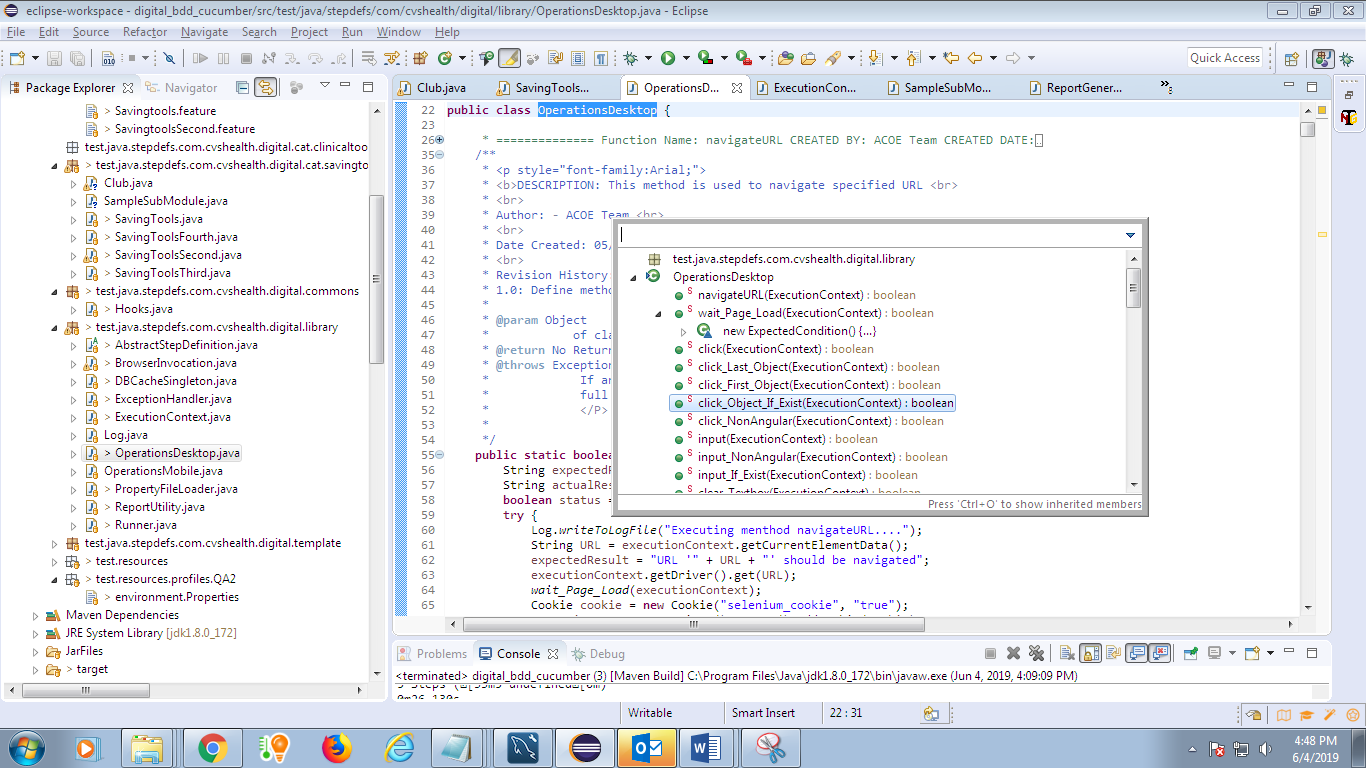
DBCacheSingleton.*getInstance*().getHashMapObjectRepositoryData(), "ICE\_Pharmacy\_LoginPage","wBtn\_SearchForSavings", Browser);

### **Set the xpath pair for usage in code**

For example: executionContext.setXpathDataPair(SearchForSavingsxPath, "");

### **use OperationsDesktop for generic/custom methods**

All re-usable generic / custom methods are defined in OperationsDesktop



Use the operations as below:

OperationsDesktop.*wait\_For\_Object*(executionContext);

OperationsDesktop.*input*(executionContext);

## ***Exceptions are handled for all steps***

### **Ensure to write the code in try-catch block and handle the Exception using ExceptionHandler class**

For example:

**catch** (Exception e) {

ExceptionHandler.*handleException*(e);

reportUtility.setStatusOfOperationForStep(executionContext, **false**);

}

## ***Use ReportUtility to log and generate test result reports***

### **Log Report for each step**

For example:

reportUtility.finalStatusSetupToExecuteForEveryStep(executionContext);

### **Log Report for each handled exception**

For Example:

reportUtility.setStatusOfOperationForStep(executionContext, **false**);

# **Execution Flow**

## ***Tag all the scenarios in the feature file***

### **Use unique/common tags for grouping the scenarios in a feature file**

For example:

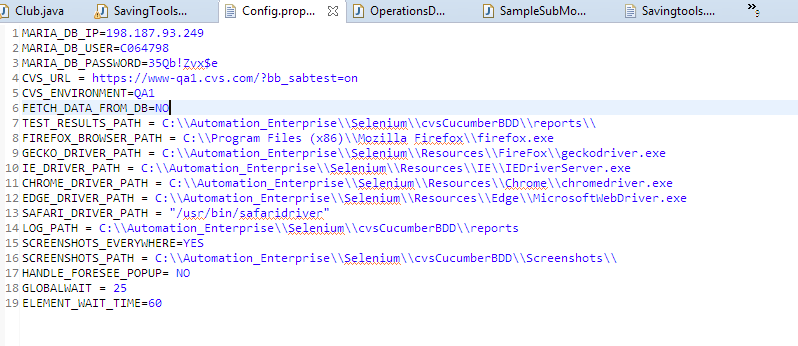
@Smoketes or @smoketestFirst or @execute

## ***Update the Config.properties file***

Update MariaDB details

Update URL and environment details

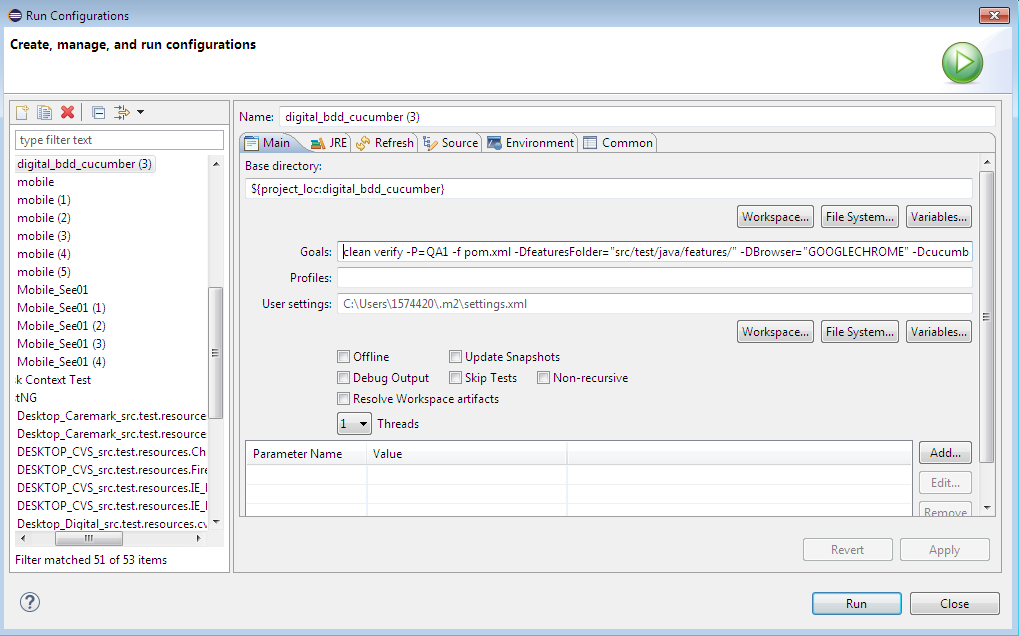
Update various paths according to your system requirements

For Example: 

## ***Run feature file***

### **Execute the feature file using the command line statement**

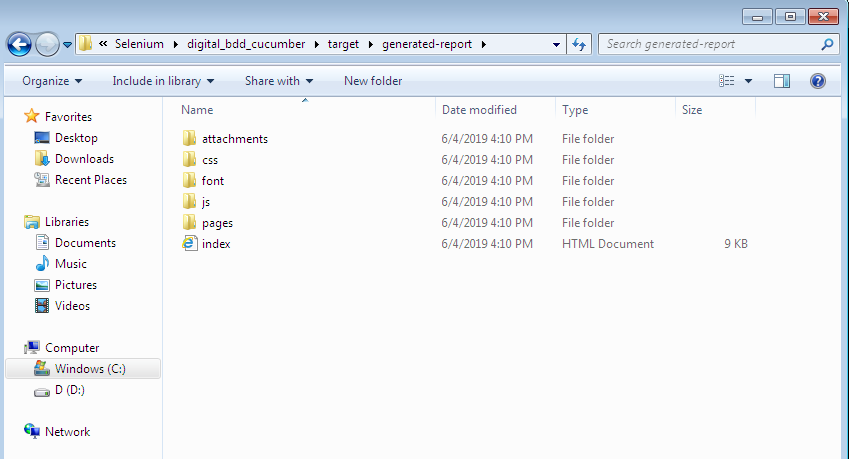
For example: **clean verify -P=QA1 -f pom.xml -DfeaturesFolder="src/test/java/features/" - DBrowser="GOOGLECHROME" -Dcucumber.options="-t @SmoketestShop1**

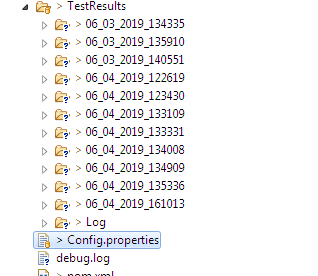


## ***Validate Test Result reports***

### **Verify the results @ C:\Automation\_Enterprise\Selenium\digital\_bdd\_cucumber\target\generated-report**

For Example:



****