**Deployment Document & User Guide for   
Selenium Framework-   
Trade BEDocument Control**

|  |  |
| --- | --- |
| Document title |  |

Contact(s) for inquiries:

| Name | Designation | Email ID | Contact Details |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

Change history

| Version | Date | Revised by | Brief outline of changes |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Related Documents/Artifacts

| Sr No | Document Title | Version | Link/location |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

Reviewers (in alphabetical order)

| **Name of Reviewer** | **Title of Reviewer** | **Date** | **Approval** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Approval / Sign off

| Name of Approver | Title of Approver | Date | Approval Evidence |
| --- | --- | --- | --- |
|  |  |  |  |

Contents

[1 Introduction 5](#_Toc493158002)

[2 Who is this document for 6](#_Toc493158003)

[3 Prerequisites 7](#_Toc493158004)

[4 Setting up the test environment 8](#_Toc493158005)

[4.1 Screen-Related Configurations 9](#_Toc493158006)

[4.1.1 Trade Specific Configuration 12](#_Toc493158007)

[4.2 Test Data Maintenance 13](#_Toc493158008)

[4.2.1 Trade Specific Configuration 15](#_Toc493158009)

[4.3 Browser Configuration 16](#_Toc493158010)

[4.4 Others Configurations 17](#_Toc493158011)

[5 Execution Steps 17](#_Toc493158012)

[6 Assumptions & Notes 18](#_Toc493158013)

[7 Getting started with other screens 19](#_Toc493158014)

[8 Troubleshooting 20](#_Toc493158015)

[9 Further Steps 21](#_Toc493158016)

[10 Appendix 22](#_Toc493158017)

[10.1 Finding field identifier for a screen element 22](#_Toc493158018)

# Introduction

Complex applications require rigorous testing to ensure that the application meets the customer requirements and the desired quality. This process becomes time-consuming and error-prone, as one needs to execute each test case for every single change made to the application. Although time-consuming, this process cannot be ignored to ensure that there is no regression due to any change in the application.

***The generic test solution is designed to address this very issue for web-based applications.***

The key attributes which make the solution generic are :

1. The screen configuration data, which is maintained in a database, is configurable for any product. This includes the navigation menus, frames and all other HTML elements.
2. The test data, which is maintainted in a seperate excel- to faciliate the testing team in easy maintenance of the test data.

The generic test solution customized for Trade has the following capabilities:

* Logs in to the system.
* Clicks on the menu items
* Navigates to the screen
* Enters data by identifying HTML screen elements. It can search based on Id, Name and xPath.
* Switches between frames within the same page.

With customization, it also captures the data e.g. reference id, from the pop up, which appears during the click of *Submit* button.

# Who is this document for

The document has been prepared to assist the teams, which plan to use the generic automated solution, in configuring the test-setup & in preparing the test cases.

This document serves as a reference guide to configure the test environment required for the execution of the generic test solution.

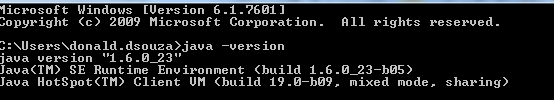
It also provides information to the tester about the format of the test data.

# Prerequisites

Please ensure the below prerequisites are fulfilled before proceeding to the next step.

1. JDK1.7 is installed on the machine where the test solution is being deployed.

* Open a command prompt and enter the command java-version.
* The JDK version will be displayed on the console.
* Upgrade the JDK version, in case of a mismatch.



1. A database schema is created in an Oracle Database. There is no restriction on the schema name.
2. The supported version of Browser and OS are as below:

* Operation System : Windows 7/10.
* Browser : Internet Explorer is 11.0.14393.0
  + To find the version of Internet Explorer (IE),
    - Open IE
    - Select Menu >> *Tools*.
    - Select the option “*About Internet Explorer*”
    - A popup screen, which displays the IE version, will appear as below.
    - Need to capture below screen shot from Nitin N desktop



1. Execute the script - DDL\_SERVICE\_FIX\_DETAILS.sql from the SQL folder in the newly created database schema.
2. Execute the script setDefineOff.sql from the SQL folder next.

This table is used to store the screen related configurations. The format, of the data maintenance, is explained in the below section (screen configuration).

# Setting up the test environment

* Create a folder to store the deployment files. Let us refer this folder as <<TradeSelenium>>.
* Extract the contents of the release (zip file) to this folder. Do not add spaces in the folder name.
* The IEDriver.exe needs to be set up under the "TradeSelenium" folder.
* Copy the test-data file(TestData\_Trade.xls) to the folder <<TradeSelenium >>/Resources.
* Replace the folder paths in the first 2 lines of the batch file (batchexe.bat). The .bat file will be present in the TradeSelenium folder.
* Test Data sheet needs to be maintained under Resources folder of the project & the file name along with path needs to be configured in config.properties
* Configure the following parameters in the Config file - <<TradeSelenium >>/Config/config.properties

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No | Key Name | Value | Example |
| 1. | URL | Application Url | http://10.10.7.50:20014/BPS/ |
| 2. | TESTINGFILEPATH | Path of Test data xls under Resources folder | /Resources/TestData\_Trade.xls |
| 3 | LOGIN | User name to be logged in application | TRADE\_SU |
| 4. | APPLICATON | Application to be logged in | BPS |
|  |  |  |  |
| 5. | DBIPADDRESS | IP Address of database where provided scripts are executed | jdbc:oracle:thin:@<IP address>:1521:<SID> |
| 6 | DBUSERNAME | Schema Name of DB where the provided scripts are executed |  |
| 7 | DBPASSWORD | Password of DB where the provided scripts are executed |  |

## Screen-Related Configurations

As part of the initial configuration, the screen related information needs to be maintained in the table - service\_fix\_details. The details of every HTML screen element ( e.g. textbox, dropdown, button, divs etc) need to be maintained. The master data maintenance for the screens will have to be done irrespective of whether it is being used in a given test case or not.

Below is an example to illustrate the screen related configuration:



The first row in the above figure represents the columns from the table “**SERVICE\_FIX\_DETAILS**”.

Below table provides the details on the data maintenance.

|  |  |
| --- | --- |
| Column Name | Data Description |
| ENTITY\_NAME | Insert Value '1' . Not being used currently. |
| TABNAME | Name of the Menu Item |
| KEYNAME | Enter the id/name of every field which needs to be tested.[[1]](#footnote-2)  Refer Appendix section for details identifying the screen elements. |
| SEQ\_NUM | Represents the sequence # in which data is to entered in a web-page.  For eg. If the password is to be entered before the Id in a login page, the value of the column “*seq\_num*”, for password & id, would be 1 and 2 respectively.  The seq\_num needs to be in order for combination of TabName and Operation.. Same seq\_num can repeat for other combination |
| ISACTIVE | Not being used currently. |
| OPERATION\_MODE | Enter the type of operation. eg. Seach , Add [[2]](#footnote-3) |
| TYPEOFATTRIBUTE | Enter the type of HTML element.  The possible values, which are case-sensitive, are follows  OpenWindow – To be used to specify that the operation opens a new window.  clickSubTab – To be used to move to a different tab in a screen.  *The other options, which are self-explanatory, are mapped directly to the HTML elements as below:*  Dropdown, TextBox, RadioButton, Checkbox,div,Hyperlink,button  CustomizedOperation - In case of any additional functionality that needs to be added a separate class has been provided. Class name- Operations.java & method is calling |
| ACTION\_BUTTON | Is Specified in case of a link or div to indicate that an action is to be expected in this operation. An action could be opening a new window or a page submit.  The values in ACTION\_BUTTON for different attributes are as follows:  OpenWindow - Window title needs to be provided & this is a must have. If currently any screen doesnt have title then kindly add the title & then provide the condiguration  clickSubTab - Tab name needs to be provided  Dropdown - Xpath of the button  TextBox- Xpath of the button  RadioButton - Xpath of the button  Checkbox - Xpath of the button  div - the flow of xpath should be "!" separated.  Hyperlink - Xpath of the link  button - Xpath of the the button |
| AFTER\_AC\_BU\_FRAME | Enter the frames to be navigated before entering data in the element.  In case of more than 1 frame, use \* as a seperator.  E.g. if the frames UserDetails and BankDetails are to be visited before clicking Submit button, the value in this field should be UserDetails\*BankDetails  If the elements are on the same frame then this value can be blank |
| BEFORE\_AC\_BU\_FRAME | This column refers to the frames to be navigated after the Html element is visited.  The format for data entry remains the same as the field – AFTER\_AC\_BU\_FRAME  If the elements are on the same frame then this value can be blank |
| ELEMENTTYPE | Enter whether the Keyname specified in the table is an ID or Name.Need to specify as Id or ByName. |

## 

### Trade Specific Configuration

***Execute the sql scripts in the newly created database to load the screen configuration details as below. Please ignore if the script has already been executed. The sql scripts are present in the folder “SQL” under the root folder.***

|  |  |
| --- | --- |
| Operation | Screen Configuration script |
| ExportBill-BookBill | DML\_ExportBill\_BookBill.sql |
| ExportLC-Advice | DML\_ExportLC\_Advice.sql |
| ImportBill-BookBill | DML\_ImportBill\_BookBill.sql |
| ImportBill-Retire | DML\_ImportBill\_Retire.sql |
| ImportLC-Amendment | DML\_ImportLC\_Amendment.sql |
| ImportLC-Issuance | DML\_ImportLC\_Issuance.sql |
| ImportLC-PreAdvice | DML\_ImportLC\_PreAdvice.sql |
| ImportLC-Update | DML\_ImportLC\_Update.sql |
| InwardCollection-BillAcceptance | DML\_InwardCollection\_BillAcceptance.sql |
| InwardCollection-BillRejection | DML\_InwardCollection\_BillRejection.sql |
| InwardCollection-BookBill | DML\_InwardCollection\_BookBill.sql |
| InwardCollection-RetireBill | DML\_InwardCollection\_RetireBill.sql |
| InwardGuarantee-Advice | DML\_InwardGuarantee\_Advice.sql |
| InwardGuarantee-Claim/Cancel | DML\_InwardGuarantee\_Claim/Cancel.sql |
| OutwardCollection-BillUpdate | DML\_OutwardCollection\_BillUpdate.sql |
| OutwardCollection-BookBill | DML\_OutwardCollection\_BookBill.sql |
| OutwardGuarantee-Amendment | DML\_OutwardGuarantee\_Amendment.sql |
| OutwardGuarantee-Issuance | DML\_OutwardGuarantee\_Issuance.sql |

## Test Data Maintenance

The test data has to maintained in an excel sheet(e.g. TestData\_Trade.xls) in the below format & the same name needs to be configured in config.properties as explained in earlier section:

**Sheet 1:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TestCase\_ID | Sheet Name | Navigation Path | User | Operation | Test case Result |  |  |
| This is the test case id | This name corresponds to the sheet name where the actual form data is provided in the xls | This corresponds to navigation path of the screen with comma separated | User Id | Corresponding to the data maintaned in the DB | Is updated by the Framework. |  |  |
| Test\_1 | Registration | Registration,LC | Trade\_SU | Search | Pass Or Fail |  |  |

* The trade related data has been keyed in the above table to make it specific to trade.
* Data for other screens needs to be entered in the sheet in a similar manner.
* When more than 1 menu item e.g hyperlink has to be clicked before arriving at the main screen, enter the keynames of all HTML elements ( menu items) separated by comma(,)

**An example to illustrate the data maintenance for the hyperlink - Bills:**

* Inert an additional row just below the above line.
* Rest of the columns would be same as the given in the above table.
* Follow the same steps for other links like collections etc.

**Sheet2: - The name of this sheet should mapped to Sheet Name as provided in Sheet 1 (Column 2).**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TestCase\_ID | Tab Name | Operation | Screenshot | Result | Field Value 1 | Field Value 2 | Field Value 3 | Field Value 4 | Field Value 5 | Field Value 6 |
| This is the test case id as provided in sheet 1 - column 1 | This is the tabname as provided in table while doing the configuration | This maps to operation\_mode as provided in the table while doing the configuation | Y  /N? |  |  |  |  |  |  |  |
| Test\_1 | Registration | Search | N | PASS | Raiffeisen Bank International AG | Import LC | Issuance |  |  |  |
| Test\_1 | Registration | Add | Y | FAIL | 00000125 | ZAR | 1000 | 1000 | 04.06.2019 | MUMBAI |

The first column is the test case id .

The next 2 columns correspond to the columns TABNAME and OPEARATION\_MODE from the table mentioned in the above section.[[3]](#footnote-4)

The 4th column signifies whether a screenshot needs to be taken or not for this operation.

The 5th column, indicating the test result, is updated by the framework.

Subsequent columns(Column 6 onwards) should have the test data corresponding to the seq\_num specified in the above table.

Below 2 examples will explain the scenario better:

**Example 1**

Lets consider a screen with 2 elements Login and password. The seq\_number for Login and password would be 1 and 2 respectively.

In this case,

The value for Login(column1) should be entered in Column 4.

The value for password(column2) should be ented in column 5.

**Example 2**

Let us assume that the screen configuration has 20 elements corresponding to the screen & the test requires data to be entered only in 3 elements.

The sequence # for these elements defined in the database is 2,4 and 6.

In this case, excel should have

The value for screen element 2 in Column 7.

The value for screen element 4 in Column 9.

The value for screen element 6 in Column 11.

The other columns should be left blank.

**NOTE:** CustomizedOperation attribute has been added *to captures the reference id, generated for test case, during the registration process and store it in the file “/Resource/Results.xls” against test case id.*

*User can uses this reference id later on in further test cases*

### Trade Specific Configuration

* Please modify the test data in the file /Resources/TestData\_Trade.xls.
* The excel workbook has a sheet for each of the operations like LC\_Registration, ImportLC, Auth\_ImportLC etc.
* The data in the corresponding sheet can be modified as per test requirement. However, the sequence # of the data in the sheet should be in sync with the sequence defined in the database.
* The navigation or the sequence of operations to be performed is to be defined in the first sheet of the workbook (Trade\_Navigation).

## Browser Configuration

* Change the internet Explorer settings as below:

*Internet Option - Select Security Option -Select Local Internet - Check Enable Protected Mode*

*Internet Option - Select Security Option -Select Trusted Sites - Check Enable Protected Mode*



*It is assumed that the security option - “****Enable Protected Mode****” is enabled for Internet and restricted sites as below. If not, even these will need to be enabled.*

## Others Configurations

1. Extract the zip file “*GenericTestSolution*” into a folder.

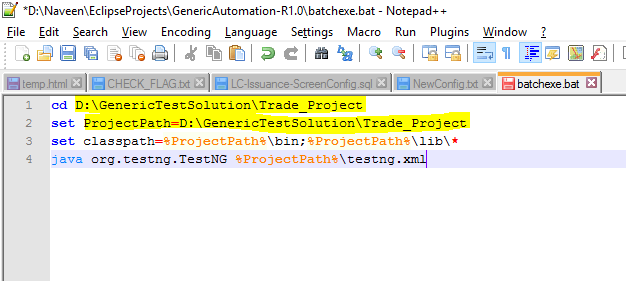
2. Open the batchexe.bat from the root folder in a text editor like notepad.

3. Replace the path of the extracted folder in the first 2 lines.

i.e. cd <<new path>>

set ProjectPath=<<new path>>

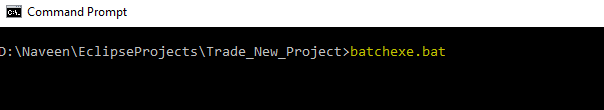
Kindly refer below diagram for reference.

****

# Execution Steps

1. Open a command prompt and go the folder where batchexe.bat is present.
2. Run the batch file using the command – “*batchexe.bat*”

Kindly refer below screenshot for reference.



# Assumptions & Notes

* Valid data, in the correct format, will be maintained in the test data excel.
* Conditional mandatory data needs to be provided in the excel as per test case
* Screen or Application specific requirement will be managed through data maintenance or customization. For e.g. there can be situations where a screen element, is either disabled or enabled, depending on the value being selected or entered in a screen element. This, being a screen specific requirement, needs to be managed using test data maintenance.
* In case of a dropdown control, the value entered in the excel should match the respective values available in the dropdown, including blank (Select) value. The value should match with the value seen on the screen
* Test Data will be maintained in the same sequence as specified in the database.
* Case sensitive data will be maintained for hyperlinks and other values. For example, if the option in a drop down is Book Bill then value in excel should not be “Bill Book” or “book bill”, rather it should be “Book Bill” only.
* For the **date field** the data in excel should be append with single inverted comma as ’04.09.2017
* All browser windows will be closed before executing the test.
* The test data for the radio button & check boxwill be the id being used in the HTML instead of the text being displayed on the screen.
* In case of number fields such as tolerance excel sheet formatting needs to be taken care & no decimals are provided from excel sheet to code
* All windows will have title.

# Getting started with other screens

Kindly refer the screen-configurations and test data maintenance section above to follow the procedure for other screens.

Please contact US SDU team for any specific customization unhandled by the generic solution.

# Troubleshooting

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Issue** | **Action** |
| 1. | Execution has stopped | 1. Please check if the server is down 2. Please check if the id/name/xpath specified is valid. 3. Check if the browser version has been updated. |
| 2. | Unable to get window | Need to re run |

* For new requirements or unresolved issues, please contact US SDU team.

# Further Steps

# Appendix

## Finding field identifier for a screen element

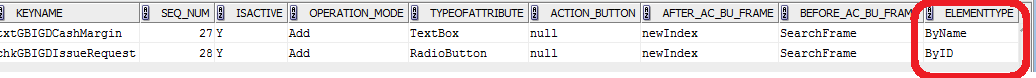
The field identifier is entered in the ‘**KEYNAME’** column in ‘**SERVICE\_FIX\_DETAILS’** Table.

Below are the steps which can help in identifying the screen identifiers

1)On opening the web page of the Trade application,press **F12 and** click on **Select Element** icon or use a shortcut key (Ctrl+B)

2)Select the required field by placing the cusor on that field.Then select the value form **input name** tag.

3)If we have selected the value from input name tag then we need to update as **ByName** value in **ELEMENTTYPE** column of ‘**SERVICE\_FIX\_DETAILS’** table and if name is not present then ID will be present and the correspondence entry will be **ID** in **ELEMENTTYPE** column.

****

**For further information contact:**  
Name :

Designation:

Email :

Phone :

1. When more than 1 menu item e.g hyperlink has to be clicked before arriving at the main screen, enter the keynames of all HTML elements ( menu items) separated by comma(,) [↑](#footnote-ref-2)
2. A special operation named”CustomizeOperation” can be specified to handle application-specific situations. [↑](#footnote-ref-3)
3. The values of the first 3 colums should be the same as Sheet1. [↑](#footnote-ref-4)