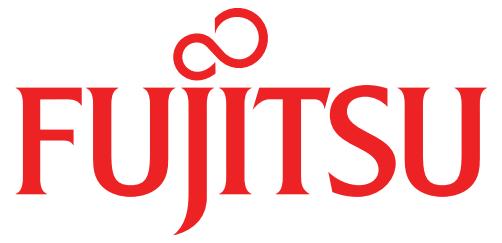


Katana Framework



Release 1.1.0
September 2020

Katana User Guide



Contents

1	Introduction	p. 5
1.1	Katana Overview	p. 6
1.2	Katana Location	p. 7
2	Getting Started	p. 8
2.1	Prerequisites	p. 9
2.2	Installing and Uninstalling Katana	p. 10
2.2.1	Install Katana Using PIP Module	p. 10
2.2.2	Uninstall Katana	p. 11
2.3	Log On	p. 12
2.4	Configure Warrior Settings	p. 13
3	Managing Case	p. 18
3.1	Create Case	p. 19
3.2	Add Details	p. 21
3.3	Add Requirements	p. 26
3.4	Add Steps	p. 28
3.5	Save Case	p. 35
4	Managing Wrapper File	p. 37
4.1	Create Wrapper File	p. 38
4.2	Add Details	p. 40
4.3	Setup Steps	p. 42
4.4	Cleanup Steps	p. 49
4.5	Debug Steps	p. 56
4.6	Save Wrapper File	p. 63
5	Managing Input Data File	p. 65
5.1	Create Input Data File	p. 66
5.2	Add System	p. 68
5.2.1	Add Data	p. 68
5.2.2	Add Subsystem	p. 71
5.3	Save Input Data File	p. 73
6	Managing CLIData File	p. 74
6.1	Create CLIData File	p. 75

6.1.1	Global Command Parameters	p. 76
6.1.2	Global Verifications	p. 78
6.1.3	Global Verification Combinations	p. 80
6.1.4	Add CLI Data Block	p. 80
6.1.5	Add CLI Data Command	p. 82
6.2	Save CLIData File	p. 83
<hr/>		
7	Managing Suite	p. 84
7.1	Create Suite	p. 85
7.2	Add Details	p. 87
7.3	Add Requirements	p. 91
7.4	Add Cases	p. 93
7.5	Save Suite	p. 99
<hr/>		
8	Managing Projects	p. 101
8.1	Create Project	p. 102
8.2	Add Details	p. 104
8.3	Add Suites	p. 107
8.4	Save Project	p. 111
<hr/>		
9	Executing via Katana	p. 113
9.1	Execute Case/Suites/Projects	p. 114
<hr/>		
Index		p. 118

Document Change Notice

Issue 1

Location of Change	Description of Change
Throughout	Initial release

1

Introduction

In this chapter:

- 1.1 Katana Overview
- 1.2 Katana Location

1.1

Katana Overview

Katana is a web-based tool for the creation and execution of test cases in Warrior Framework. User can create cases, suites, projects and input data files. Specific applications are available for each type of file creation. These applications contain web-based forms with various fields that help to set the options for the file creation. The created files are converted to appropriate XML files after saving. Katana allows the user to execute cases, suites, and projects using the execution application.

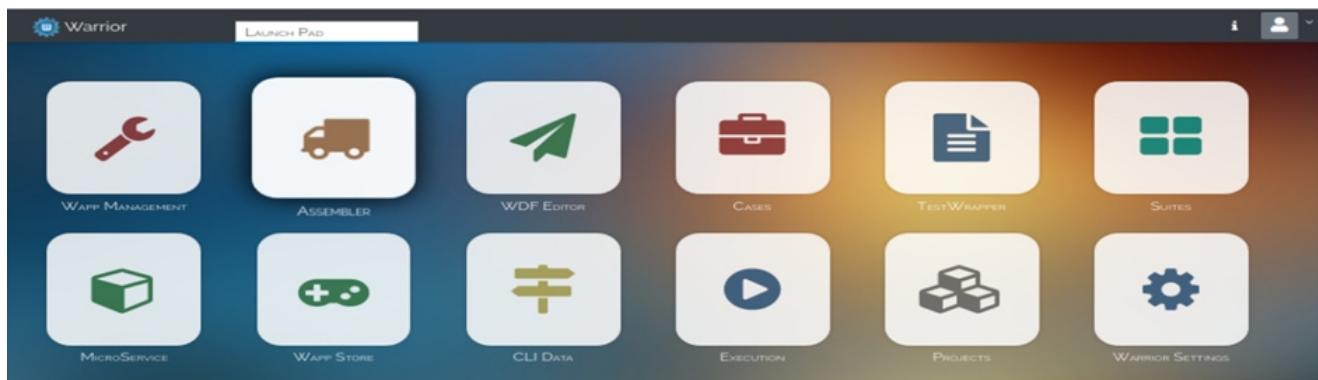


Figure 1
Launch Pad Page

1.2

Katana Location

Katana is currently located at:

<https://pypi.org/project/katanaframework/>

Note: Katana can be installed using a PIP installable Warrior.

2

Getting Started

In this chapter:

- 2.1 Prerequisites
- 2.2 Installing and Uninstalling Katana
- 2.3 Log On
- 2.4 Configure Warrior Settings

2.1

Prerequisites

- Python® 3.6 and later.

2.2

Installing and Uninstalling Katana

In this section:

- 2.2.1 Install Katana Using PIP Module
- 2.2.2 Uninstall Katana

2.2.1

Install Katana Using PIP Module

Step 1

Install Katana via PIP module.

Command:

```
> pip install katanaframework
```

Note: To install specific version of Katana, use below command

Command:

```
> pip install katanaframework==1.0.0
```

Step Result:

Installation of packages required to launch Katana starts.

Step 2

Launch Katana through user interface.

Command:

```
> manage.py runserver
```

Step Result:

Katana starts on the localhost at port 8000.

Step 3

Open a web browser and enter *localhost:8000* to access Katana.

Note: The user can launch Katana on a different port using `manage.py runserver <portnumber>` command, where `<portnumber>` is a user-defined value. After creating the port number, the user can access Katana in a web browser using the `localhost:<portnumber>` command. Now, Katana launches on the localhost at the different port.

This task is complete.

2.2.2

Uninstall Katana

Step 1

Uninstall Katana.

Command:

```
pip uninstall katanaframework
```

This task is complete.

2.3

Log On

This procedure describes how to log on to Katana.

Step 1

Log on to Katana from a web browser using the default user-defined localhost port number.

Step Result:

Katana launches on the localhost and the *Login* page opens.

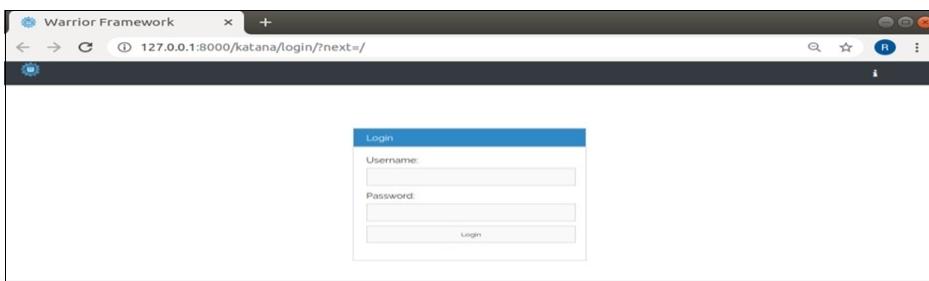


Figure 2
Login Page

Step 2

Enter the following username and password:

- Username: *Admin*
- Password: *warriorframework*

Step Result:

The *Launch Pad* page opens displaying all applications.

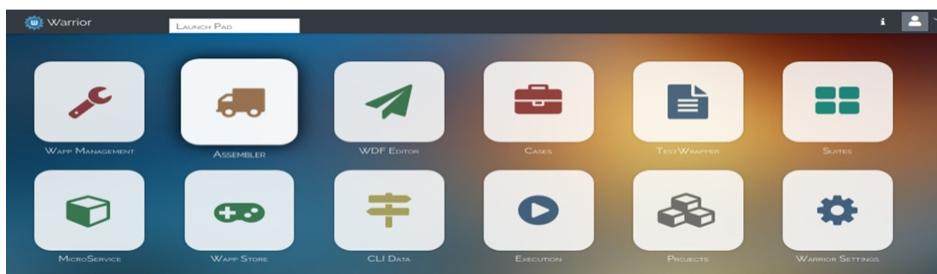


Figure 3
Launch Pad Page

This task is complete.

2.4

Configure Warrior Settings

This procedure describes how to configure Warrior settings.

Step 1

Launch Katana and click the *WARRIOR SETTINGS* icon.

Step Result:

The *Warrior Settings* page opens.

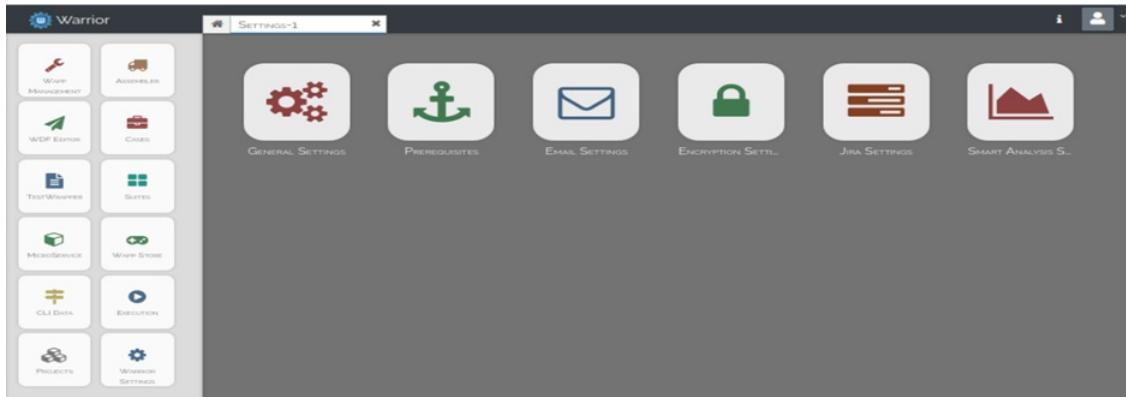


Figure 4
Warrior Settings Page

Step 2

Click the *General Settings* icon to configure the settings.

Step Result:

The *General Settings* page opens.

Getting Started

Configure Warrior Settings

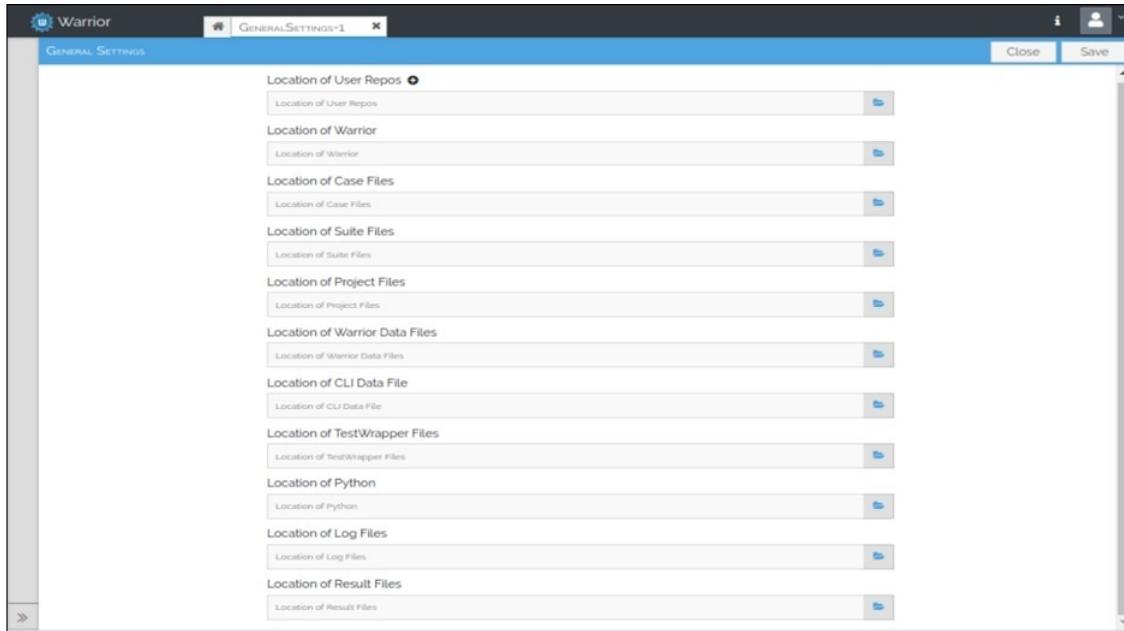


Figure 5
General Settings Page

The user must fill the page when launching Katana for the first time.

Step 3

In the *Location of User Repos* field, click the *folder* icon to select a path for the *User Repository* directory.

Step 4

Click Save to save the selected path.

Step Result:

A *Warning!* dialog box opens, asking whether to apply the same path for cases, suites, projects, input data files and test data files.



Figure 6
Warning Dialog Box

Step 5

Click on Yes to populate the same path for other files.

Step 6

In the *Location of Warrior* field, click the *folder* icon to select a path for the *Warrior Framework* directory.

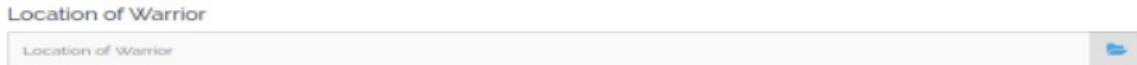


Figure 7

Location of Warrior Field

Step 7

In the *Location of Case Files* field, click the *folder* icon to set a path for the *Case* directory.

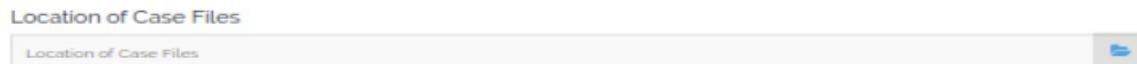


Figure 8

Location of Case Files Field

Note: This directory may or may not be inside the Warrior Framework directory.

For more information on Warrior cases, refer to *Warrior Framework User Guide*.

For more information on how to create cases in Katana, refer to [Create Case](#).

Step 8

In the *Location of Suite Files* field, click the *folder* icon to select a path for the *Suite* directory.

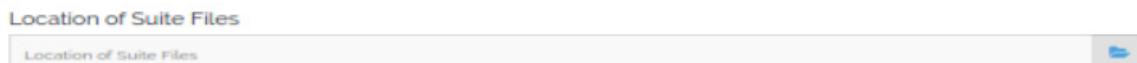


Figure 9

Location of Suite Files Field

Note: This directory may or may not be inside the Warrior Framework directory.

For more information on Warrior suites, refer to *Warrior Framework User Guide*.

For more information on how to create suites in Katana, refer to [Create Suite](#).

Step 9

In the *Location of Project Files* field, click the *folder* icon to select a path for the *Project* directory.

Location of Project Files

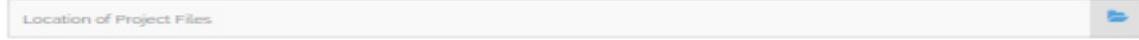


Figure 10

Location of Project Files Field

Note: This directory may or may not be inside the Warrior Framework directory.

Step 10

In the *Location of Warrior Data Files* field, click the *folder* icon to set a path for the *Warrior Data Files* directory.

Location of Warrior Data Files

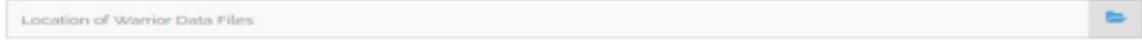


Figure 11

Location of Warrior Data Files Field

Note: This directory may or may not be inside the Warrior Framework directory.

Step 11

In the *Location of CLI Data File* field, click the *folder* icon to set a path for the *CLI Data File* directory.

Location of CLI Data File

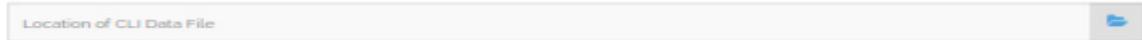


Figure 12

Location of CLI Data File Field

Note: This directory may or may not be inside the Warrior Framework directory.

Step 12

In the *Location of Testwrapper Files* field, click the *folder* icon to set a path for the *Testwrapper Files* directory.

Location of TestWrapper Files

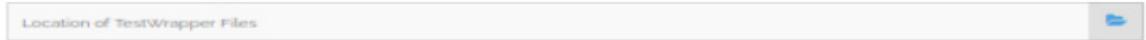


Figure 13

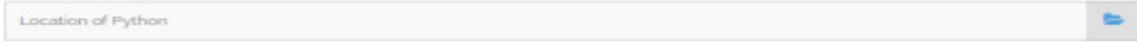
Location of Testwrapper Files Field

Note: This directory may or may not be inside the Warrior Framework directory.

Step 13

In the *Location of Python* field, click the *folder* icon to set a path to an alternate Python file that the user may want to use for executing Warrior Framework.

Location of Python

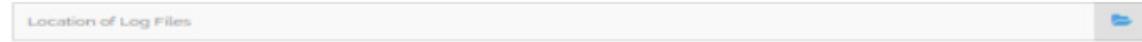
Figure 14
Location of Python Field

Note: The default Python filepath on user system is used if the *Location of Python* field is empty.

Step 14

In the *Location of Log Files* field, click the *folder* icon to select a path for the *Log Files* directory.

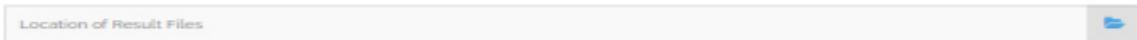
Location of Log Files

Figure 15
Location of Log Files Field

Step 15

In the *Location of Result Files* field, click the *folder* icon to select a path for the *Resulted Files* directory.

Location of Result Files

Figure 16
Location of Result Files Field

Step 16

Click *Save* to save the configuration.

Step Result:

Katana is now ready to use.

This task is complete.

3

Managing Case

In this chapter:

- 3.1 Create Case
- 3.2 Add Details
- 3.3 Add Requirements
- 3.4 Add Steps
- 3.5 Save Case

3.1

Create Case

Katana allows the user to create Cases using the Cases application. This procedure describes how to create a new case.

Step 1

Launch Katana and click the CASES icon.

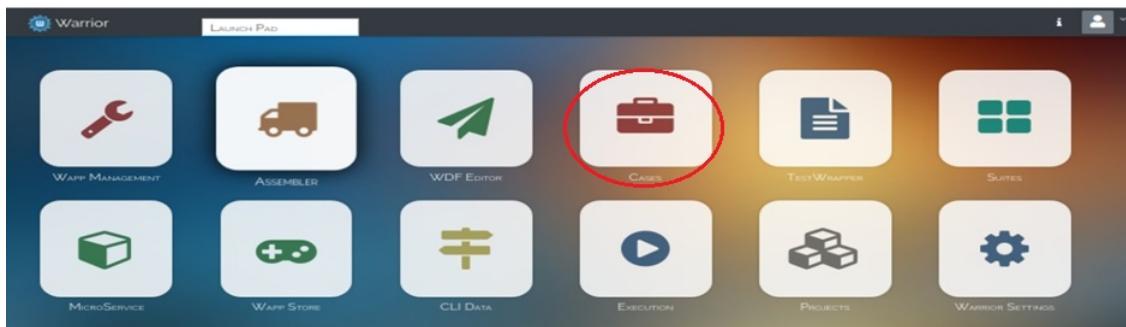


Figure 17
CASES Icon

Step Result:

The Cases page opens.

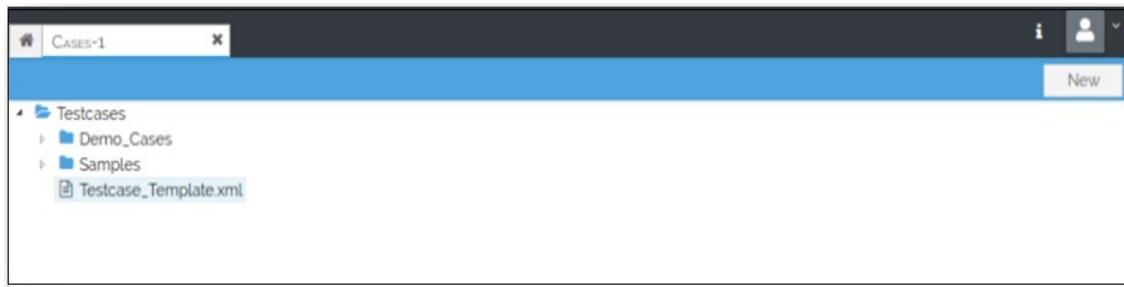


Figure 18
Cases Page

Step 2

Click the New button.

Step Result:

The test case page opens.

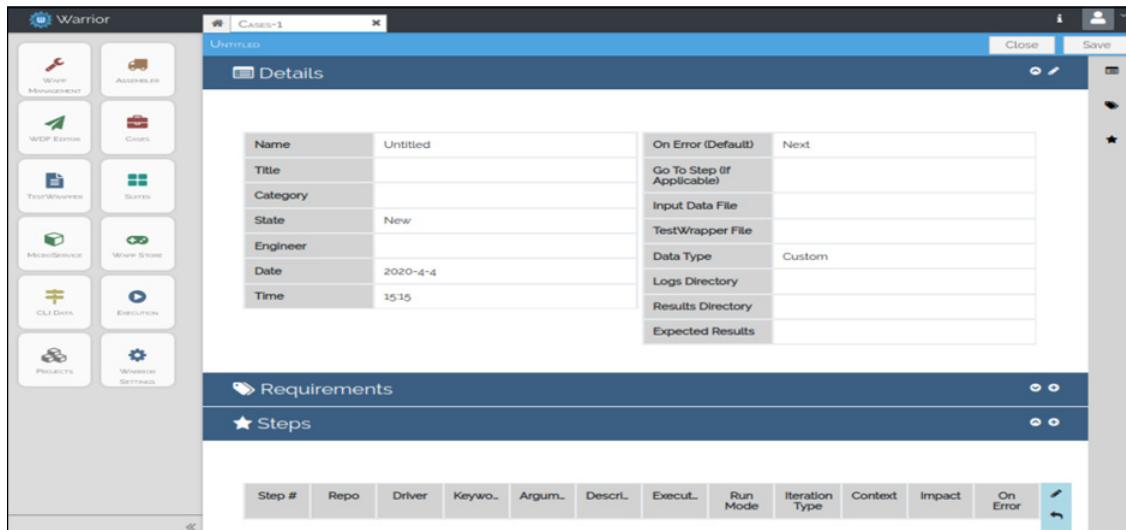


Figure 19
Test Case Page

User must fill the fields in the following sections to complete the case creation process:

- Details
- Requirements
- Steps

Continue with the next task.

3.2

Add Details

This procedure describes how to add case details. The user can provide the case details by filling the fields in the *Details* section. The fields in the *Details* section are explained in the following table.

Table 1
Details Field Description

Field	Description
Name ¹	Name of the case that can be recognizable by the user
Title ¹	Description of the case
Category	Categorization of cases is important when the user has to run all the cases in the same category together. <i>Note:</i> For more information about what <i>Category</i> does, refer to <i>Warrior Framework User Guide</i> .
State	Keeps track of the cases that are new, released, or in review
Engineer ¹	Displays the name of the engineer who created the case
Date ²	Displays the date on which the case is updated
Time ²	Displays the time on which the case is updated
On Error (Default)	Specifies the action the case should be taken when a case error occurs. <i>Note:</i> For more information about what <i>Default on Error</i> does, refer to <i>Warrior Framework User Guide</i> .
Go To Step	User-specified case number that redirects Warrior in case of failure <i>Note:</i> This field appears when <i>Go To</i> option is selected.
Input Data File	Sets the path for an Input Data file
Test Wrapper File	Sets the path for a Test Wrapper file
Data Type ¹	Defines an interaction method the case must have with the data
Logs Directory	Specifies the path to store the log file returned by Warrior
Results Directory	Specifies the path to store the results returned by Warrior
Expected Results	Specifies the kind of result expected from the case

¹ This field is mandatory.

² This field is a prefilled field.

Step 1

Click the *Edit Details* button to edit the details section data.

Step 2

In the *Name* field, enter the name of the case.

Step 3

In the *Title* field, enter a descriptive title.

Step 4

In the *Category* field, enter the category of the case.

Note: This field is an optional field; however, Fujitsu recommends to categorize the cases.

Step 5

Select the appropriate *State* from the drop-down list: *New*, *Test-Assigned*, or *Released*.

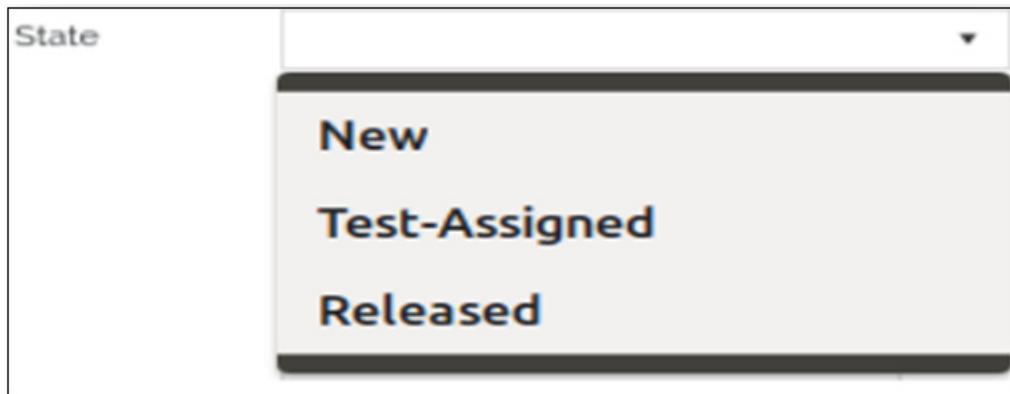


Figure 20
State Options

Note: This field is an optional field, but Fujitsu recommends to fill out this field to keep track of state of development of each case.

Note: If the available categories do not fit the desired case state, the user can type the required information.

Step 6

In the *Engineer* field, enter the name of the engineer who created the case.

Step 7

Select the appropriate *On Error (Default)* option from the drop-down list: *Next*, *Abort*, *Abort As Error*, or *Go To*.

Note: Default value, *On Error* field selects *Next*. If no changes are made, the case proceeds to the next case available for execution when the current case throws an error.

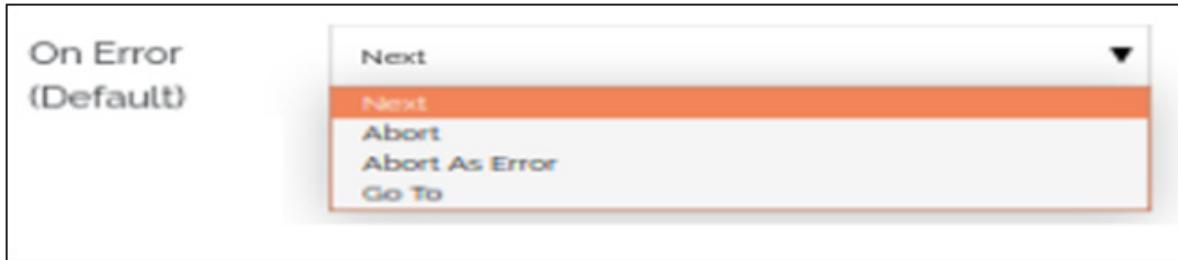


Figure 21
On Error Options

Note: Selecting *Abort* and *Abort As Error* options terminate the execution of the case if the step does not pass.



Notice: When *Go To* option is selected, a *Go To Step* field appears. Enter the case number that Warrior must refer to in case of an error.

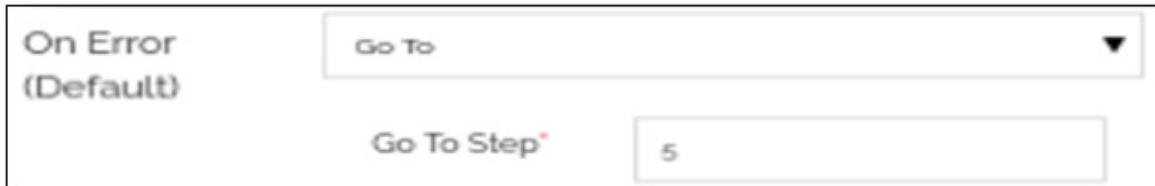


Figure 22
Go To Step Field

Step 8

In the *Input Data File* field, set the path to a desired data file for the case to use.



Hint: Click the *Folder* icon to select a file from the *Select a Path* window and click *OK*.

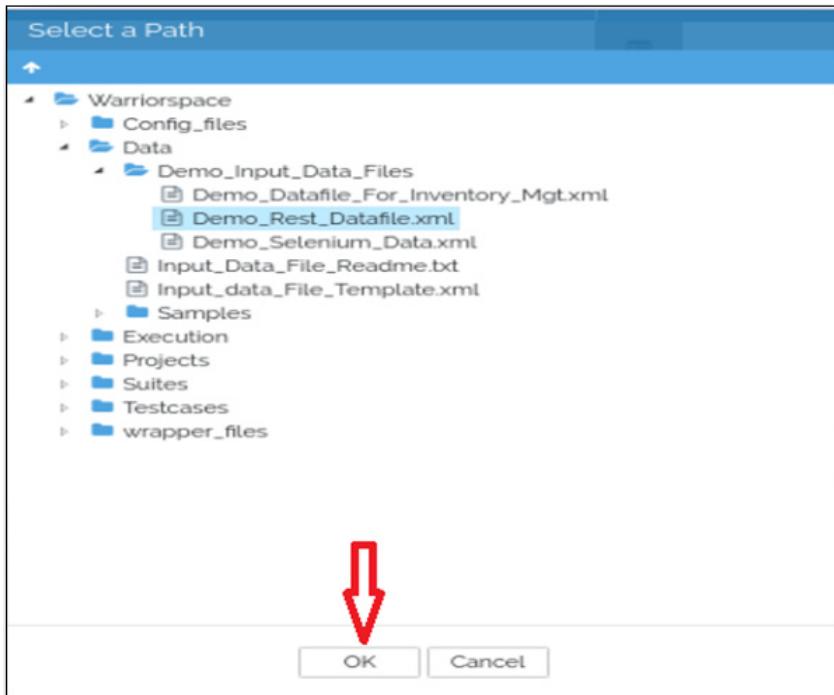


Figure 23
Select Path Window



Hint: User can directly enter the file path into the field.

Step 9

In the *Test Wrapper File* field, set the path to a test wrapper file for the case to use.



Hint: Click the *Folder* icon to select a file from the *Select a Path* window and click *OK*.

Step 10

Select the appropriate *Data Type* from the drop-down list: *Custom*, *Iterative*, or *Hybrid*.



Figure 24
Data Type Options

Step 11

In the *Logs Directory* field, set the path to store the logs.

Note: If the location is not set in the *Logs Directory* field, the logs are directed and stored in the default location.

Step 12

In the *Results Directory* field, set the path to store the results.

Note: If the location is not set in the *Results Directory* field, the results are directed and stored in the default location.

Step 13

Enter the expected results in the *Expected Results* field.

Continue with the next task.

3.3

Add Requirements

This procedure describes how to add requirements.



Remember: This procedure is optional.

The following table lists the icons for the *Requirements* section.

Table 2
Requirements Icon Description

Icon	Label	Description
Requirement Section		
	Add New Requirement	Adds new requirements
	Delete	Deletes new requirements
New Requirements Dialog Box		
	Discard	Discards requirement changes
	Save	Saves new requirements

Step 1

Click the *Add New Requirement* button or the *plus* icon.

Step Result:

The *New Requirements* dialog box opens.

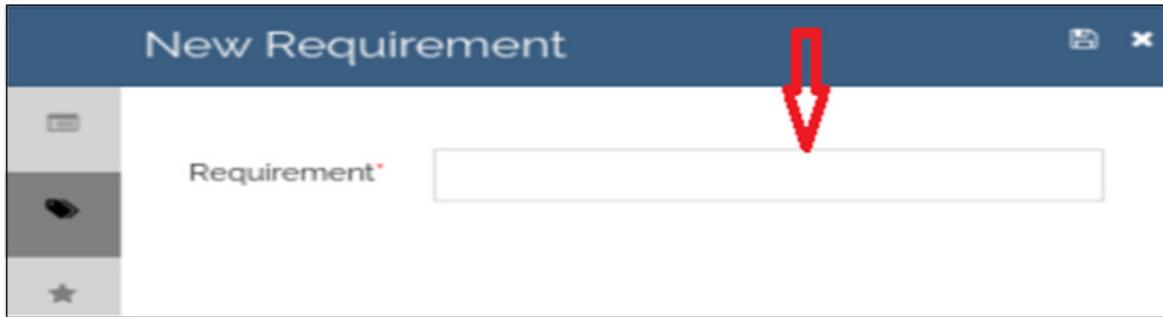


Figure 25
New Requirements Dialog Box

Step 2

In the *Requirement* field, enter the requirement.

Step 3

After entering the requirement, click the save icon to save the requirement.

Step Result:

New requirement is saved.

Continue with the next task.

3.4

Add Steps

This procedure describes how to add steps for a case. The user can provide the case steps by filling the fields in the *Steps* section. Every case requires at least one step.

The following table lists the icons for the *Steps* section.

Table 3
Steps Icon Description

Icon	Label	Description
Step Section		
	Add New Step	Adds new case steps
	Delete	Deletes new case steps
	Edit	Edits created case steps
New Steps Dialog Box		
	Discard	Discards case step changes
	Save	Saves new case steps

Step 1

Click the *Add New Step* button or the *plus* icon.

Note: The first step is mandatory and every step after is optional.

Step Result:

The *New Step* dialog box opens.

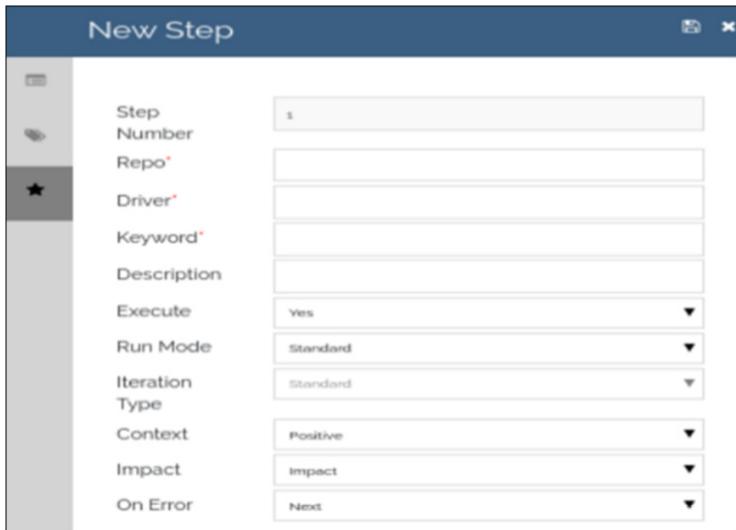


Figure 26
New Step Dialog Box

The fields in the *New Step* dialog box are explained in the following table.

Table 4
New Step Fields

Field	Description
Step Number ¹	Displays the step number
Repo ²	Displays the package name where the given driver is present, if Repo name is not given in step, it defaults to <i>Warrior</i> .
Driver ²	Displays all the drivers available in the <i>Warrior</i> Framework directory
Keyword ¹	Displays a list of keywords
WDescription ¹	Displays exactly what the selected keyword does
Comments	Enables the user to understand what the keyword does when it runs, what kind of arguments it accepts, and what kind of values it returns.
Description	Displays the provided description when the step is run
Execute Type	Displays options to execute a step
Condition ³	Specifies the condition for <i>If</i> or <i>If Not</i> Execute Type

Table 4 (Cont.)

New Step Fields

Field	Description
Condition Value ³	Specifies the key value for the provided <i>If</i> or <i>If Not</i> condition. This drop-down provides the following options: <input type="checkbox"/> PASS <input type="checkbox"/> FAIL <input type="checkbox"/> ERROR <input type="checkbox"/> SKIPPED
Else ³	Specifies the condition to handle the scenario where the condition in <i>If</i> or <i>If Not</i> is not met and what Warrior needs to know in such a case
Run Mode	Describes the run type for the execution of the step. This drop-down provides the following options: <input type="checkbox"/> Standard <input type="checkbox"/> Run Multiple Times <input type="checkbox"/> Run Until Pass <input type="checkbox"/> Run Until Failure
Value	Specifies the maximum number of times the step must run <i>Note:</i> This field appears when Run Mode is filled.
Iteration Type ⁴	Displays the iteration type to run the step
Context	Indicates if the step is a positive or negative test scenario
Impact	Enables the user to decide if the status of the step impacts the test case or not
On Error	Displays a list of error handling conditions to assign for a step
Go To Step	User-specified case number that redirects Warrior in case of failure <i>Note:</i> This field appears when <i>Go To</i> option is selected.

¹ This field is prefilled and noneditable.² This field is mandatory.³ This field appears when *If* or *If Not* Execute type is selected.⁴ This field is prefilled and editable.

Step 2

In the *Repo* field, select the appropriate option.

Step 3

Select a *Driver* from the drop-down list.

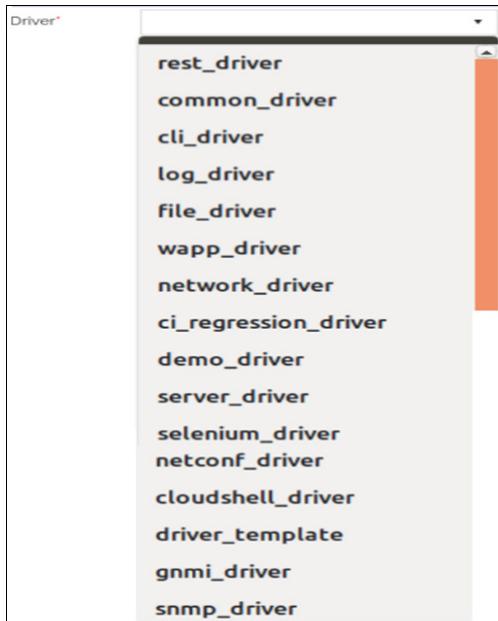


Figure 27
Driver Options

Note: Every step must have a driver associated to it.



Restriction: Katana allows the user to select a driver by showing all the drivers available in the Warrior Framework directory in the drop-down list. The user cannot select any driver that lies outside of *Product Drivers* directory because Warrior enforces a strict directory structure for the Actions and Product Drivers.

Step 4

Select the required *Keyword* from the drop-down list.



Important: Every step must have an associated keyword to it.

Note: If no driver is selected, no keyword is shown in the drop-down list. The user must choose the driver before choosing the keyword.

Step Result:

A set of argument fields populates when a developed keyword is selected. All the arguments accepted by the keyword are shown here.

Keyword*	verify_page_by_property												
Signature	verify_page_by_property(self, system_name, expected_value, value_type, browser_name="all", element_config_file=None, element_tag=None)												
Arguments	<table border="1"><tr><td>system_name</td><td><input type="text"/></td></tr><tr><td>expected_value</td><td><input type="text"/></td></tr><tr><td>value_type</td><td><input type="text"/></td></tr><tr><td>browser_name</td><td><input type="text"/></td></tr><tr><td>element_config_file</td><td><input type="text"/></td></tr><tr><td>element_tag</td><td><input type="text"/></td></tr></table>	system_name	<input type="text"/>	expected_value	<input type="text"/>	value_type	<input type="text"/>	browser_name	<input type="text"/>	element_config_file	<input type="text"/>	element_tag	<input type="text"/>
system_name	<input type="text"/>												
expected_value	<input type="text"/>												
value_type	<input type="text"/>												
browser_name	<input type="text"/>												
element_config_file	<input type="text"/>												
element_tag	<input type="text"/>												

Figure 28
Arguments Fields

The *WDescription* field populates when the Keyword is selected.

WDescription	The browser will verify page by fol"format(value_type)
--------------	--

Figure 29
WDescription Field

Note: The *Comments* field appears prepopulated when a developed Keyword is selected.

Comments	:Returns: 1. status(bool)- True / False.
----------	---

Figure 30
Comments Field

Step 5
Enter values for each field in the argument.

Step 6
In the *Description* field, add a description for the selected keyword.

Step 7
Select the appropriate *Execute* type from the drop-down list: *Yes*, *No*, *If*, or *If Not*.

Note: Default value, *Execute* type is set to *Yes*.



Notice: On selecting *If* and *If Not*, new input boxes appear to add conditions.

The screenshot shows a software interface for configuring a step. The top bar says 'Execute'. Below it, a dropdown menu is set to 'If'. Underneath are three input fields: 'Condition*' with a dropdown arrow, 'Condition Value*' with a dropdown arrow, and 'Else*' with a dropdown arrow.

Figure 31
Condition, Condition Value, and Else Fields

Step 8

Select the appropriate *Run Mode* from the drop-down list: *Standard*, *Run Multiple Times*, *Run Until Failure*, or *Run Until Pass*.

Step Result:

The *Value* field appears.

A screenshot of a software interface showing a single input field labeled 'Value*' containing the number '4'.

Figure 32
Value Field

Step 9

In the *Value* field, enter the number of times the step must run.

Step 10

Select the appropriate *Iteration Type* from the drop-down list.

Step 11

Select the appropriate *Context* from the drop-down list: *Positive* or *Negative*.

Note: The default value is *Positive*.

Step 12

Select the appropriate *Impact* from the drop-down list: *Impact* or *No-impact*.

Note: The default value is *Impact*.

Step 13

Select the error condition from the *On Error* drop-down list.

Note: The default value is the case *On Error*.



Notice: If *Go To* is selected, a mandatory *Go To Step* field appears. The user must specify the step for Warrior to redirect in case of failure.

Step 14

Click *Save Step*.

Note: A Step must be saved.

Step Result:

A *Step* report appears.

Step #	Repo	Driver	Keywo...	Argum...	Descri...	Execut...	Run Mode	Iteration Type	Context	Impact	On Error
1	warrior	seleniu... m_driver	verify_... page_b... property		Open C... hrome... Browser	Yes	Standar... d	Standar... d	Positive	Impact	Go To - 3

Figure 33
Step Report

Continue with the next task.

3.5

Save Case

This procedure describes how to save a case.

Step 1

Click *Save*.

Note: If any unsaved step is present when saving a case, the unsaved step is automatically saved.



Figure 34
Save Case

Step Result:

A message dialog box opens displaying the case name.

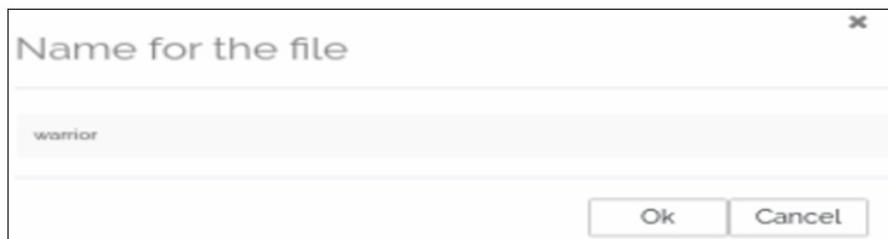


Figure 35
Message Dialog Box

Step 2

Click *Ok* to save the test case.

Step Result:

The created case appears as shown in the following figure.

The screenshot shows a software interface for managing test cases. The top navigation bar includes 'File', 'Edit', 'Run', 'View', 'Help', and 'Details'. The main content area is divided into three sections:

- Details:** Displays case metadata: Name (set_env_variable), Title (set_env_variable), Category (Regression), State (Released), Engineer (Warrior_Test), Date (2017-05-14), Time (17:02). Configuration options include 'On Error (Default)' (Next), 'Go To Step If Applicable', 'Input Data File' (No_Data), 'TextMapper File', 'Data Type' (Custom), 'Logs Directory', 'Results Directory', and 'Expected Results'.
- Requirements:** Shows two requirements: Demo-requirement-001 and Demo-requirement-002.
- Stages:** A table listing 8 steps in sequence:

Step #	Scope	Driver	Keyword	Arguments	Description	Execute	Run Mode	Iteration Type	Context	Impact	On Error
1	warrior	common_driver	set_env_var	var_key = check3 var_value = 3	(Regression for exist. var_value = 3)	Yes	Standard	Standard	Positive	Impact	next
2	warrior	common_driver	set_env_var	Reqpath = J:\J\J\warrior	(Passing variables on. Reqpath = J:\J\J\warrior)	Yes	Standard	Standard	Positive	Impact	next
3	warrior	common_driver	set_env_var	var_key = check3 var_value = 3 Reqpath = J:\J\J\warrior	(Passing variables th. var_value = 3 Reqpath = J:\J\J\warrior)	Yes	Standard	Standard	Positive	Impact	next
4	warrior	common_driver	set_env_var	var_key = check3 var_value = 3 Reqpath = J:\J\config_R	(Expected to fail due. var_value = 3 Reqpath = J:\J\config_R)	Yes	Standard	Standard	Negative	Impact	next
5	warrior	common_driver	set_env_var	var_key = check3 var_value = 3 Reqpath = J:\J\config_R	(Expected to fail bec. var_value = 3 Reqpath = J:\J\config_R)	Yes	Standard	Standard	Negative	Impact	next
6	warrior	common_driver	set_env_var	var_key = check3 var_value = 3 Reqpath = J:\J\config_R	Negative Expected L.	Yes	Standard	Standard	Negative	Impact	next
7	warrior	common_driver	set_env_var	var_key = check3 var_value = 3 Reqpath = J:\J\J\warrior junkkey = my_variable	User can give junkke...	Yes	Standard	Standard	Positive	Impact	next
8	warrior	common_driver	set_env_var		Negative Expected L.	Yes	Standard	Standard	Negative	Impact	next

Figure 36
Created Case Page

This task is complete.

4

Managing Wrapper File

In this chapter:

- 4.1 Create Wrapper File
- 4.2 Add Details
- 4.3 Setup Steps
- 4.4 Cleanup Steps
- 4.5 Debug Steps
- 4.6 Save Wrapper File

4.1

Create Wrapper File

Katana allows the user to create a wrapper file using the TESTWRAPPER application. This procedure describes how to create a new wrapper file.

Step 1

Launch Katana and click the *TESTWRAPPER* icon.

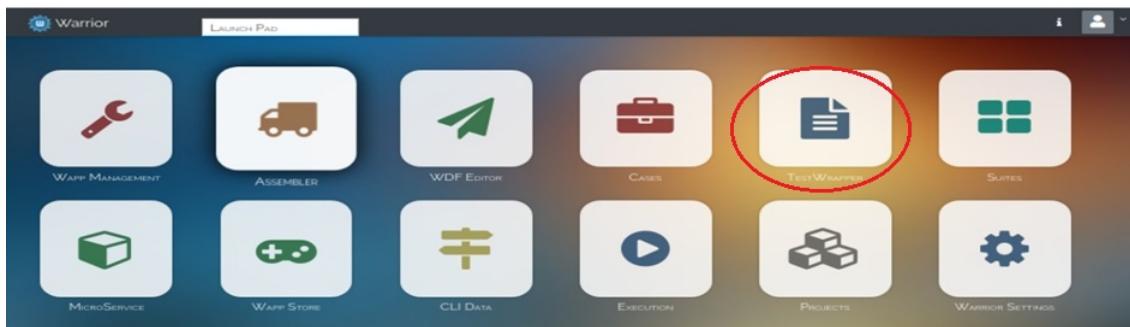


Figure 37
TESTWRAPPER Icon

Step Result:

The *TestWrapper* page opens.

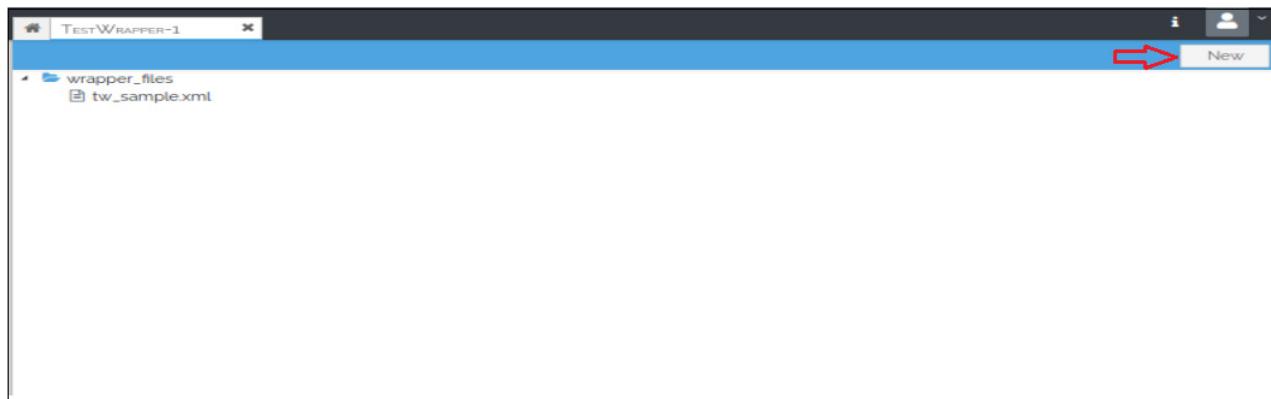


Figure 38
TestWrapper Page

Step 2

Click the *New* button.

Step Result:

The test wrapper page opens.

The screenshot shows a software application window titled "Test Wrapper Page". At the top left, there's a "Details" tab. Below it, the "Setup Steps", "Cleanup Steps", and "Debug Steps" tabs are visible. Each section has a table with columns: Step #, Repo, Driver, Keyword, Arguments, Description, Execute, Run Mode, Iteration Type, Context, Impact, and On Error. The "Details" section includes fields for Name (Untitled), Title, Engineer, Date (2020-5-4), and Time (0:40). A "Data Type" dropdown is set to "Custom". The "Setup Steps", "Cleanup Steps", and "Debug Steps" sections each have three rows of table headers.

Figure 39
Test Wrapper Page

User must fill the fields in the following sections to complete the wrapper file creation process:

- Details
- Setup Steps
- Cleanup Steps
- Debug Steps

Continue with the next task.

4.2

Add Details

This procedure describes how to add wrapper file details. The user can provide the wrapper file details by filling the fields in the *Details* section. The fields in the *Details* section are explained in the following table.

Table 5
Details Field Description

Field	Description
Name ¹	Name of the wrapper file that can be recognizable by the user
Title ¹	Description of the wrapper file
Engineer ¹	Displays the name of the engineer who created the wrapper file
Date ²	Displays the date on which the wrapper file is updated
Time ²	Displays the time on which the wrapper file is updated
Data Type ¹	Defines an interaction method the wrapper file should have with the data file

¹ This field is mandatory.

² This field is a prefilled field.

Step 1

Click the *New* button.

Step 2

In the *Name* field, enter the name of the wrapper file.

Step 3

In the *Title* field, enter a descriptive title.

Step 4

In the *Engineer* field, enter the name of the engineer who created the wrapper file.

Step 5

Select the appropriate *Data Type* from the drop-down list: *Custom*, *Iterative*, or *Hybrid*.

Managing Wrapper File

[Add Details](#)

Figure 40
Data Type Options

Continue with the next task.

4.3

Setup Steps

This procedure describes how to add steps for a wrapper file.



Remember: This procedure is optional.

The user can provide the wrapper file steps by completing the fields in the *Setup Steps* section. The following table lists the icons displayed in the *Setup Steps* section.

Table 6
Steps Icon Description

Icon	Label	Description
Step Section		
	Add New Step	Adds new steps
	Delete	Deletes steps
	Edit	Edits created steps
New Steps Dialog Box		
	Discard	Discards step changes
	Save	Saves steps

Step 1

Click the *Add New Step* button or the *plus* icon.

Step Result:

The *New Step* dialog box opens.

Managing Wrapper File

Setup Steps

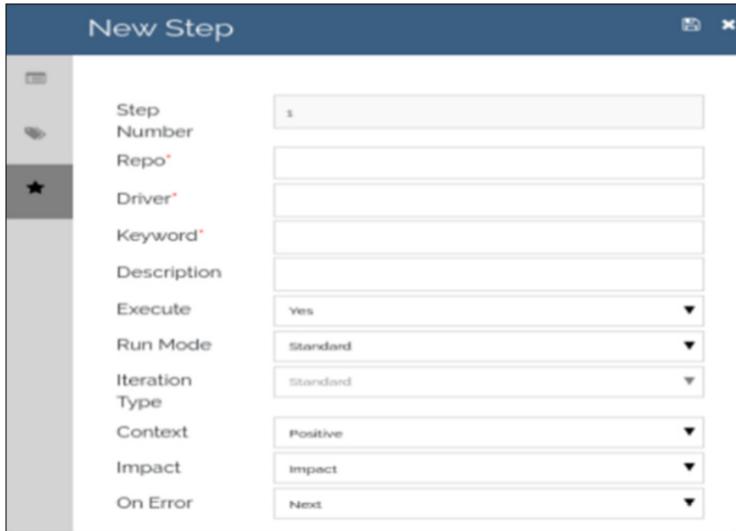


Figure 41
New Step Dialog Box

The fields in the *New Step* dialog box are explained in the following table.

Table 7
New Steps Fields

Field	Description
Step Number ¹	Displays the step number
Repo ²	Displays the package name where the given driver is present, if Repo name is not given in step, it defaults to <i>Warrior</i> .
Driver ²	Displays all the drivers available in the <i>Warrior</i> Framework directory
Keyword ¹	Displays a list of keywords
WDescription ¹	Displays exactly what the keyword does
Comments	Enables the user to understand what the keyword does when it runs, what kind of arguments it accepts, and what kind of values it returns
Description	Displays the provided description when the step is run
Execute	Describes the type of execution
Condition ³	Specifies the condition for <i>If</i> or <i>If Not</i> Execute Type

Table 7 (Cont.)
New Steps Fields

Field	Description
Condition Value ³	Specifies the key value for the provided <i>If</i> or <i>If Not</i> condition. This drop-down provides the following options: <ul style="list-style-type: none"> ■ PASS ■ FAIL ■ ERROR ■ SKIPPED
Else ³	Specifies the condition to handle the scenario where the condition in <i>If</i> or <i>If Not</i> is not met and what Warrior needs to know in such a case
Run Mode	Describes the run type for the execution of the step. This drop-down provides the following options: <ul style="list-style-type: none"> ■ Standard ■ Run Multiple Times ■ Run Until Pass ■ Run Until Failure
Value	Describes the maximum number of times the user wants to run this step <i>Note:</i> This field appears when Run Mode is filled.
Iteration Type ⁴	Describes the type of iteration: <ul style="list-style-type: none"> ■ Standard ■ Once per case ■ End of case
Context	Indicates if the step is a positive or negative test scenario
Impact	Enables the user to decide if the status of the step impacts the test case or not
On Error	Handles the error condition. This drop-down provides the following options: <ul style="list-style-type: none"> ■ Next ■ Abort ■ Abort As Error ■ Go To
Go To Step	User-specified case number that redirects Warrior in case of failure <i>Note:</i> This field appears when <i>Go To</i> option is selected from <i>On Error</i> drop-down list.

¹ This field is prefilled and noneditable.² This field is mandatory.³ This field appears when *If* or *If Not* Execute type is selected.⁴ This field is prefilled and editable.

Managing Wrapper File

Setup Steps

Step 2

In the *Repo* field, select the appropriate option.

Step 3

Select a *Driver* from the drop-down list.

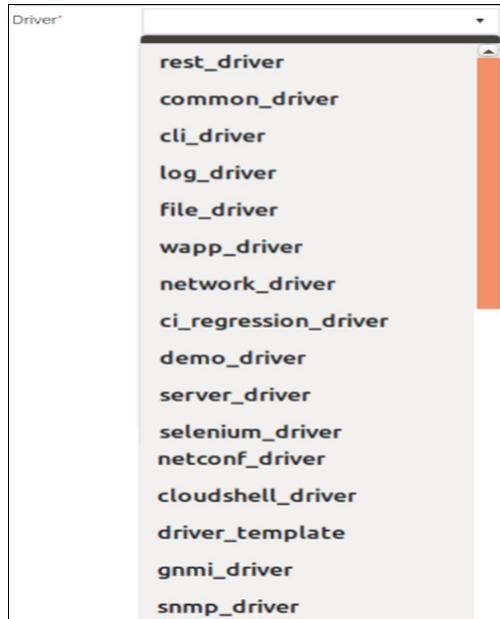


Figure 42
Driver Options

Note: Every step must have a driver associated to it.



Restriction: Katana allows the user to select a driver by showing all the drivers available in the Warrior Framework directory in the drop-down list. The user cannot select any driver that lies outside of *Product Drivers* directory because Warrior enforces a strict directory structure for the Actions and Product Drivers.

Step 4

Select the required *Keyword* from the drop-down list.



Important: Every step must have an associated keyword to it.

Note: If no driver is selected, no keywords are shown in the drop-down list. The user must choose the driver before choosing the keyword.

Managing Wrapper File

Setup Steps

Step Result:

A set of argument fields populates when a developed keyword is selected. All the arguments accepted by the keyword are shown here.

Keyword*	verify_page_by_property												
Signature	verify_page_by_property(self, system_name, expected_value, value_type, browser_name="all", element_config_file=None, element_tag=None)												
Arguments	<table border="1"><tr><td>system_name</td><td>[]</td></tr><tr><td>expected_value</td><td>[]</td></tr><tr><td>value_type</td><td>[]</td></tr><tr><td>browser_name</td><td>[]</td></tr><tr><td>element_config_file</td><td>[]</td></tr><tr><td>element_tag</td><td>[]</td></tr></table>	system_name	[]	expected_value	[]	value_type	[]	browser_name	[]	element_config_file	[]	element_tag	[]
system_name	[]												
expected_value	[]												
value_type	[]												
browser_name	[]												
element_config_file	[]												
element_tag	[]												

Figure 43
Arguments Fields

The *WDescription* field populates when the Keyword is selected.

WDescription	The browser will verify page by lol*formatvalue_type
--------------	--

Figure 44
WDescription Field

Note: The *Comments* field appears prepopulated when a developed Keyword is selected.

Comments	:Returns: 1. status(bool)= True / False.
----------	---

Figure 45
Comments Field

Step 5
Enter values for each field in the argument.

Step 6
In the *Description* field, add a description for the selected Keyword.

Step 7
Select the appropriate *Execute* type from the drop-down list: *Yes*, *No*, *If*, or *If Not*.

Note: The default value is Yes for *Execute* type.



Notice: On selecting *If* and *If Not*, new input boxes appear to add conditions.

The screenshot shows the 'Execute' step configuration. A dropdown menu is open, showing the option 'If'. Below it, there are three input fields: 'Condition*', 'Condition Value*', and 'Else*'. The 'Condition*' field has a dropdown arrow pointing down. The 'Condition Value*' field contains the value 'Next'. The 'Else*' field also has a dropdown arrow pointing down.

Figure 46
Condition, Condition Value, and Else Fields

Step 8

Select the appropriate *Run Mode* from the drop-down list: *Standard*, *Run Multiple Times*, *Run Until Failure*, or *Run Until Pass*.

Step Result:

The *Value* field appears.

A screenshot of a single-line text input field. The placeholder text 'Value*' is visible on the left, and the number '4' is typed into the field.

Figure 47
Value Field

Step 9

In the *Value* field, enter the number of times the step must run.

Step 10

Select the appropriate *Iteration Type* from the drop-down list.

Step 11

Select the appropriate *Context* from the drop-down list: *Positive* or *Negative*.

Note: The default value is *Positive*.

Step 12

Select the appropriate *Impact* from the drop-down list: *Impact* or *No-impact*.

Note: The default value is *Impact*.

Managing Wrapper File

Setup Steps

Step 13

Select the error condition from the *On Error* drop-down list.

Note: The default is the case On Error value.



Notice: If *Go To* is selected, a mandatory *Go To Step* field appears. The user must specify the step for Warrior to redirect in case of failure.

Step 14

Click *Save Step*.

Note: A Step must be saved.

Step Result:

A *Step* report appears.

Step #	Repo	Driver	Keywo_	Argum_	Descri_	Execut_	Run Mode	Iteration Type	Context	Impact	On Error
1	warrior	seleniu_m_driv_	verify_page_b_y_prop_	erty	Open C hrome Browse r	Yes	Standar d	Standar d	Positive	Impact	Go To - 3

Figure 48
Step Report

Continue with the next task.

4.4

Cleanup Steps

This procedure describes how to add cleanup steps for wrapper file.



Remember: This procedure is optional.

The user can provide the wrapper file steps by filling the fields in the *Cleanup Steps* section. The following table lists the icons for the *Cleanup Steps* section.

Table 8
Steps Icon Description

Icon	Label	Description
Step Section		
	Add New Step	Adds new steps
	Delete	Deletes new steps
	Edit	Edits created steps
New Steps Dialog Box		
	Discard	Discards step changes
	Save	Saves new steps

Step 1

Click the *Add New Step* button or the *plus* icon.

Step Result:

The *New Step* dialog box opens.

Managing Wrapper File

Cleanup Steps

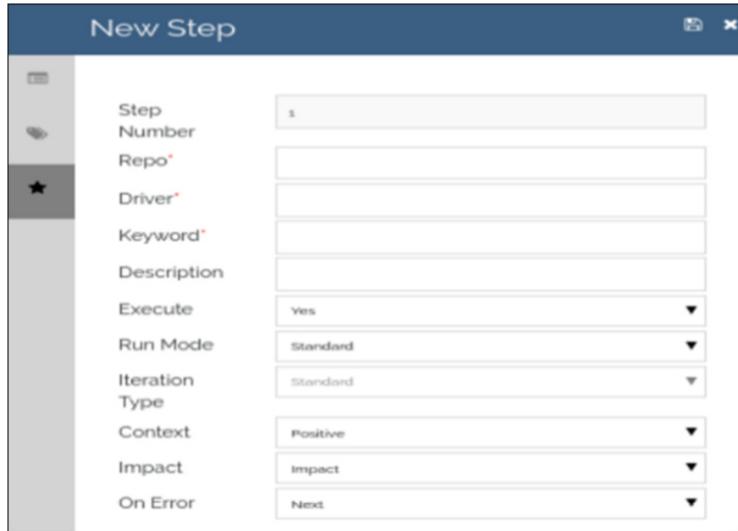


Figure 49
New Step Dialog Box

The fields in the *New Step* dialog box are explained in the following table.

Table 9
New Step Fields

Field	Description
Step Number ¹	Displays the step number
Repo ²	Displays the package name where the given driver is present, if Repo name is not given in step, it defaults to <i>Warrior</i> .
Driver ²	Displays all the drivers available in the <i>Warrior</i> Framework directory
Keyword ¹	Displays a list of keywords
WDescription ¹	Displays exactly what the keyword does
Comments	Enables the user to understand what the keyword does when it runs, what kind of arguments it accepts, and what kind of values it returns
Description	Displays the provided description when the step is run
Execute	Describes the type of execution
Condition ³	Specifies the condition for <i>If</i> or <i>If Not</i> Execute Type

Managing Wrapper File

Cleanup Steps

Table 9 (Cont.)

New Step Fields

Field	Description
Condition Value ³	Specifies the key value for the provided <i>If</i> or <i>If Not</i> condition. This drop-down provides the following options: <input type="checkbox"/> PASS <input type="checkbox"/> FAIL <input type="checkbox"/> ERROR <input type="checkbox"/> SKIPPED
Else ³	Specifies the condition to handle the scenario where the condition in <i>If</i> or <i>If Not</i> is not met and what Warrior needs to know in such a case
Run Mode	Describes the run type for the execution of the step. This drop-down provides the following options: <input type="checkbox"/> Standard <input type="checkbox"/> Run Multiple Times <input type="checkbox"/> Run Until Pass <input type="checkbox"/> Run Until Failure
Value	Describes the maximum number of times the user wants to run this step <i>Note:</i> This field appears when Run Mode is filled.
Iteration Type ⁴	Describes the type of iteration
Context	Indicates if the step is a positive or negative test scenario
Impact	Enables the user to decide if the status of the step impacts the test case or not
On Error	Handles the error condition. This drop-down provides the following options: <input type="checkbox"/> Next <input type="checkbox"/> Abort <input type="checkbox"/> Abort As Error <input type="checkbox"/> Go To
Go To Step	User-specified case number that redirects Warrior in case of failure <i>Note:</i> This field appears when <i>Go To</i> option is selected from <i>On Error</i> drop-down list.

¹ This field is prefilled and noneditable.

² This field is mandatory.

³ This field appears when *If* or *If Not* Execute type is selected.

⁴ This field is prefilled and editable.

Step 2

In the *Repo* field, select the appropriate option.

Step 3

Select a *Driver* from the drop-down list.

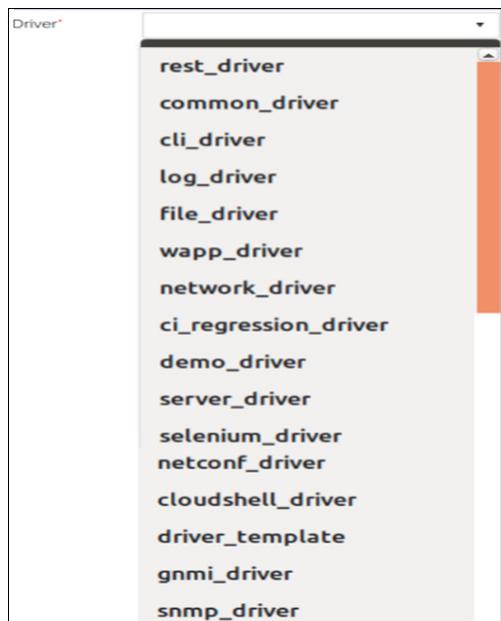


Figure 50
Driver Options

Note: Every step must have a driver associated to it.



Restriction: Katana allows the user to select a driver by showing all the drivers available in the Warrior Framework directory in the drop-down list. The user cannot select any driver that lies outside of *Product Drivers* directory because Warrior enforces a strict directory structure for the Actions and Product Drivers.

Step 4

Select the required *Keyword* from the drop-down list.



Important: Every step must have an associated keyword to it.

Note: If no driver is selected, no keywords are shown in the drop-down list. The user must choose the driver before choosing the keyword.

Step Result:

A set of argument fields populates when a developed keyword is selected. All the arguments accepted by the Keyword are shown here.

Managing Wrapper File

Cleanup Steps

Keyword*	verify_page_by_property												
Signature	verify_page_by_property(self, system_name, expected_value, value_type, browser_name="all", element_config_file=None, element_tags=None)												
Arguments	<table border="1"><tr><td>system_name</td><td><input type="text"/></td></tr><tr><td>expected_value</td><td><input type="text"/></td></tr><tr><td>value_type</td><td><input type="text"/></td></tr><tr><td>browser_name</td><td><input type="text"/></td></tr><tr><td>element_config_file</td><td><input type="text"/></td></tr><tr><td>element_tags</td><td><input type="text"/></td></tr></table>	system_name	<input type="text"/>	expected_value	<input type="text"/>	value_type	<input type="text"/>	browser_name	<input type="text"/>	element_config_file	<input type="text"/>	element_tags	<input type="text"/>
system_name	<input type="text"/>												
expected_value	<input type="text"/>												
value_type	<input type="text"/>												
browser_name	<input type="text"/>												
element_config_file	<input type="text"/>												
element_tags	<input type="text"/>												

Figure 51
Arguments Fields

The *WDescription* field populates when the Keyword is selected.

WDescription	The browser will verify page by following value_type
--------------	--

Figure 52
WDescription Field

Note: The *Comments* field appears prepopulated when a developed Keyword is selected.

Comments	:Returns: 1. status(bool)- True / False.
----------	---

Figure 53
Comments Field

Step 5
Enter values for each field in the argument.

Step 6
In the *Description* field, add a description for the selected Keyword.

Step 7
Select the appropriate *Execute* type from the drop-down list: *Yes*, *No*, *If*, or *If Not*.

Note: The default value is *Yes*.



Notice: On selecting *If* and *If Not*, new input boxes appear to add conditions.

The screenshot shows the 'Execute' step configuration. The 'Condition' dropdown is set to 'If'. The 'Condition Value' field contains '4'. The 'Else' dropdown is set to 'Next'.

Figure 54
Condition, Condition Value, and Else Fields

Step 8

Select the appropriate *Run Mode* from the drop-down list: *Standard*, *Run Multiple Times*, *Run Until Failure*, or *Run Until Pass*.

Step Result:

The *Value* field appears.

Figure 55
Value Field

Step 9

In the *Value* field, enter the number of times the step must run.

Step 10

Select the appropriate *Iteration Type* from the drop-down list.

Step 11

Select the appropriate *Context* from the drop-down list: *Positive* or *Negative*.

Note: The default value is *Positive*.

Step 12

Select the appropriate *Impact* from the drop-down list: *Impact* or *No-impact*.

Note: The default value is *Impact*.

Step 13

Select the error condition from the *On Error* drop-down list.

Managing Wrapper File

Cleanup Steps

Note: The default value is the case *On Error*.



Notice: If *Go To* is selected, a mandatory *Go To Step* field appears. The user must specify the step for Warrior to redirect in case of failure.

Step 14

Click *Save Step*.

Note: A Step must be saved.

Step Result:

A *Step* report appears.

Step #	Repo	Driver	Keywo...	Argum...	Descri...	Execut...	Run Mode	Iteration Type	Context	Impact	On Error
1	warrior	seleniu... m_driver	verify_... page_b... property		Open C... hrome... Browser	Yes	Standar... d	Standar... d	Positive	Impact	Go To - 3

Figure 56
Step Report

Continue with the next task.

4.5

Debug Steps

This procedure describes how to add debug steps for a wrapper file.



Remember: This procedure is optional.

The user can provide the wrapper file steps by filling the fields in the *Debug Steps* section. The following table lists the icons for the *Debug Steps* section.

Table 10
Steps Icon Description

Icon	Label	Description
Step Section		
	Add New Step	Adds new case steps
	Delete	Deletes new case steps
	Edit	Edits created case steps
New Steps Dialog Box		
	Discard	Discards case step changes
	Save	Saves new case steps

Step 1

Click the *Add New Step* button or the *plus* icon.

Step Result:

The *New Step* dialog box opens.

Managing Wrapper File

Debug Steps

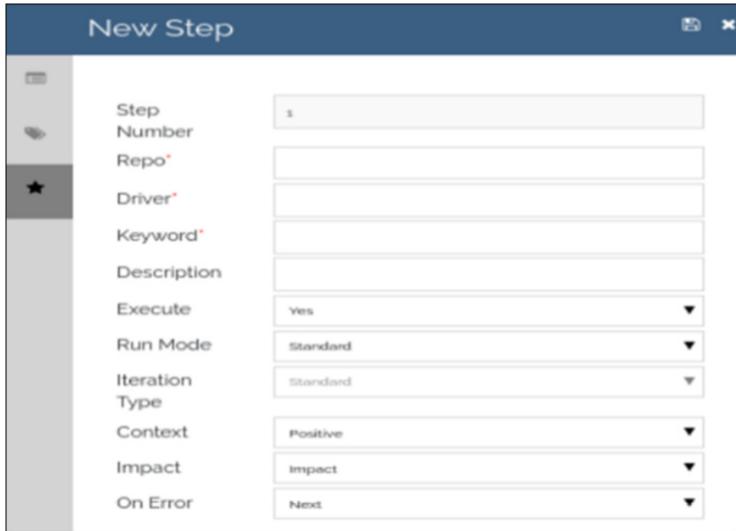


Figure 57
New Step Dialog Box

The fields in the *New Step* dialog box are explained in the following table.

Table 11
New Step Fields

Field	Description
Step Number ¹	Displays the step number
Repo ²	Displays the package name where the given driver is present, if Repo name is not given in step, it defaults to <i>Warrior</i> .
Driver ²	Displays all the drivers available in the <i>Warrior</i> Framework directory
Keyword ¹	Displays a list of keywords
WDescription ¹	Displays exactly what the keyword does
Comments	Enables the user to understand what the keyword does when it runs, what kind of arguments it accepts, and what kind of values it returns
Description	Displays the provided description when the step is run
Execute	Describes the type of execution
Condition ³	Specifies the condition for <i>If</i> or <i>If Not</i> Execute Type

Table 11 (Cont.)
New Step Fields

Field	Description
Condition Value ³	Specifies the key value for the provided <i>If</i> or <i>If Not</i> condition. This drop-down provides the following options: <input type="checkbox"/> PASS <input type="checkbox"/> FAIL <input type="checkbox"/> ERROR <input type="checkbox"/> SKIPPED
Else ³	Specifies the condition to handle the scenario where the condition in <i>If</i> or <i>If Not</i> is not met and what Warrior needs to know in such a case
Run Mode	Describes the run type for the execution of the step. This drop-down provides the following options: <input type="checkbox"/> Standard <input type="checkbox"/> Run Multiple Times <input type="checkbox"/> Run Until Pass <input type="checkbox"/> Run Until Failure
Value	Describes the maximum number of times the user wants to run this step <i>Note:</i> This field appears when Run Mode is filled.
Iteration Type ⁴	Describes the type of iteration
Context	Indicates if the step is a positive or negative test scenario
Impact	Enables the user to decide if the status of the step impacts the test case or not
On Error	Handles the error condition. This drop-down provides the following options: <input type="checkbox"/> Next <input type="checkbox"/> Abort <input type="checkbox"/> Abort As Error <input type="checkbox"/> Go To
Go To Step	User-specified case number that redirects Warrior in case of failure <i>Note:</i> This field appears when <i>Go To</i> option is selected from <i>On Error</i> drop-down list.

¹ This field is prefilled and noneditable.

² This field is mandatory.

³ This field appears when *If* or *If Not* Execute type is selected.

⁴ This field is prefilled and editable.

Step 2

In the *Repo* field, select the appropriate option.

Step 3

Select a *Driver* from the drop-down list.

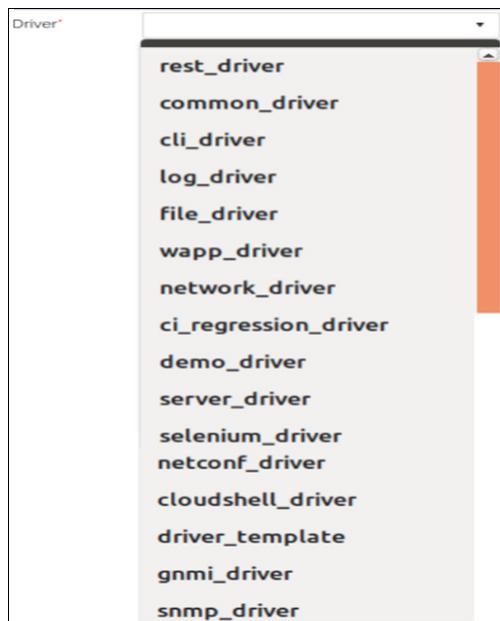


Figure 58
Driver Options

Note: Every step must have a driver associated to it.



Restriction: Katana allows the user to select a driver by showing all the drivers available in the Warrior Framework directory in the drop-down list. The user cannot select any driver that lies outside of *Product Drivers* directory because Warrior enforces a strict directory structure for the Actions and Product Drivers.

Step 4

Select the required *Keyword* from the drop-down list.



Important: Every step must have an associated keyword to it.

Note: If no driver is selected, no keywords are shown in the drop-down list. The user must choose the driver before choosing the keyword.

Step Result:

A set of argument fields populates when a developed keyword is selected. All the arguments accepted by the keyword are shown here.

Managing Wrapper File

Debug Steps

Keyword*	verify_page_by_property												
Signature	verify_page_by_property(self, system_name, expected_value, value_type, browser_name="all", element_config_file=None, element_tags=None)												
Arguments	<table border="1"><tr><td>system_name</td><td><input type="text"/></td></tr><tr><td>expected_value</td><td><input type="text"/></td></tr><tr><td>value_type</td><td><input type="text"/></td></tr><tr><td>browser_name</td><td><input type="text"/></td></tr><tr><td>element_config_file</td><td><input type="text"/></td></tr><tr><td>element_tags</td><td><input type="text"/></td></tr></table>	system_name	<input type="text"/>	expected_value	<input type="text"/>	value_type	<input type="text"/>	browser_name	<input type="text"/>	element_config_file	<input type="text"/>	element_tags	<input type="text"/>
system_name	<input type="text"/>												
expected_value	<input type="text"/>												
value_type	<input type="text"/>												
browser_name	<input type="text"/>												
element_config_file	<input type="text"/>												
element_tags	<input type="text"/>												

Figure 59
Arguments Fields

The *WDescription* field populates when the Keyword is selected.

WDescription	The browser will verify page by fol*format(value_type)
--------------	--

Figure 60
WDescription Field

Note: The *Comments* field appears prepopulated when a developed Keyword is selected.

Comments	:Returns: 1. status(bool)- True / False.
----------	---

Figure 61
Comments Field

Step 5
Enter values for each field in the argument.

Step 6
In the *Description* field, add a description for the selected Keyword.

Step 7
Select the appropriate *Execute* type from the drop-down list: *Yes*, *No*, *If*, or *If Not*.

Note: The default value is *Yes*.



Notice: On selecting *If* and *If Not*, new input boxes appear to add conditions.

The screenshot shows the 'Execute' dialog box with the 'If' option selected. It contains three main fields: 'Condition*' with a dropdown menu, 'Condition Value*' with a text input field containing '4', and 'Else*' with a dropdown menu set to 'Next'.

Figure 62
Condition, Condition Value, and Else Fields

Step 8

Select the appropriate *Run Mode* from the drop-down list: *Standard*, *Run Multiple Times*, *Run Until Failure*, or *Run Until Pass*.

Step Result:

The *Value* field appears.

The screenshot shows the 'Value*' field in the 'Execute' dialog box, which contains the number '4'.

Figure 63
Value Field

Step 9

In the *Value* field, enter the number of times the step must run.

Step 10

Select the appropriate *Iteration Type* from the drop-down list.

Step 11

Select the appropriate *Context* from the drop-down list: *Positive* or *Negative*.

Note: The default value is *Positive*.

Step 12

Select the appropriate *Impact* from the drop-down list: *Impact* or *No-impact*.

Note: The default value is *Impact*.

Step 13

Select the error condition from the *On Error* drop-down list.

Note: The default value is the case *On Error*.



Notice: If *Go To* is selected, a mandatory *Go To Step* field appears. The user must specify the step for Warrior to redirect in case of failure.

Step 14

Click *Save Step*.

Note: A Step must be saved.

Step Result:

A *Step* report appears.

Step #	Repo	Driver	Keywo...	Argum...	Descri...	Execut...	Run Mode	Iteration Type	Context	Impact	On Error
1	warrior	seleniu... m_driver	verify_... page_b... property		Open C... hrome... Browser	Yes	Standar... d	Standar... d	Positive	Impact	Go To - 3

Figure 64
Step Report

Continue with the next task.

4.6

Save Wrapper File

This procedure describes how to save a wrapper file.

Step 1

Click *Save*.

Note: If any unsaved step is present when saving a wrapper file, the unsaved step is automatically saved.



Figure 65
Save Wrapper File

Step Result:

A message dialog box opens displaying the wrapper filename.



Figure 66
Message Dialog Box

Step 2

Click *Ok* to save the Testcase.

Step Result:

The created wrapper file appears as shown in the following figure.

Managing Wrapper File

Save Wrapper File

Name	Welcome	Data Type	Custom
Title	Wrapper_file		
Engineer	Raghavendra		
Date	2020-5-4		
Time	9:40		

★ Setup Steps

Step #	Repo	Driver	Keyword	Arguments	Description	Execute	Run Mode	Iteration Type	Context	Impact	On Error
1	warrior	selenium_d...	send_keys_...		New	Yes	Standard	Standard	Positive	Impact	Next

★ Cleanup Steps

Step #	Repo	Driver	Keyword	Arguments	Description	Execute	Run Mode	Iteration Type	Context	Impact	On Error
1	warrior	selenium_d...	navigate_to_...		New	Yes	Standard	Standard	Positive	Impact	Next

★ Debug Steps

Step #	Repo	Driver	Keyword	Arguments	Description	Execute	Run Mode	Iteration Type	Context	Impact	On Error
1	warrior	selenium_d...	get_eleme...		New	Yes	Standard	Standard	Positive	Impact	Next

Figure 67
Created Wrapper File Page

This task is complete.

5

Managing Input Data File

In this chapter:

- 5.1 Create Input Data File
- 5.2 Add System
- 5.3 Save Input Data File

5.1

Create Input Data File

Katana allows the user to create an input data file using the WDF Editor application. The procedure describes how to create a new input data file.

Step 1

Launch Katana and click the *WDF EDITOR* icon.

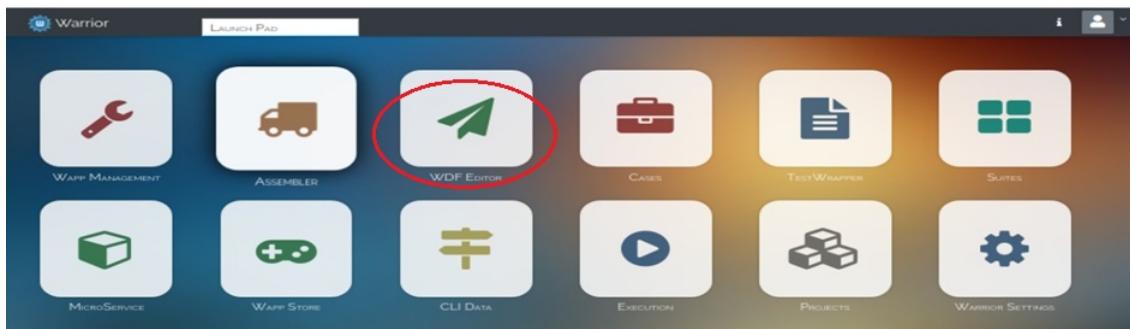


Figure 68
WDF EDITOR Icon

Step Result:

The *WDFEditor* page opens.

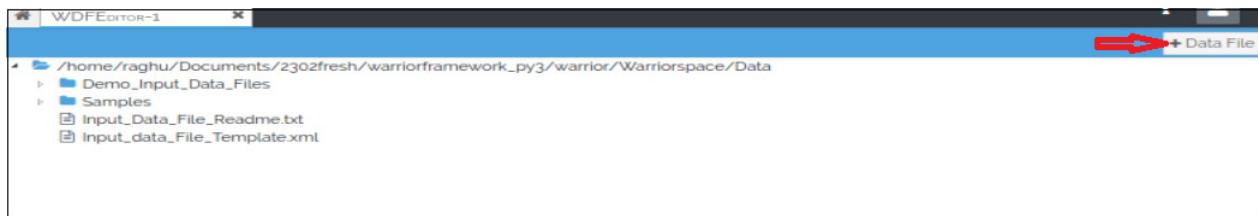


Figure 69
WDFEditor Page

Step 2

Click the *Data File* button.

Step Result:

An input data file page opens.

Managing Input Data File

Create Input Data File

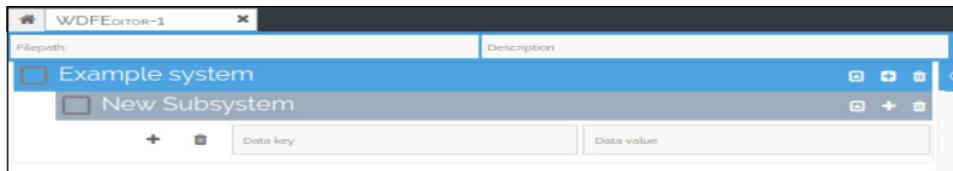


Figure 70
Input Data File Page

Step 3

Enter the name for the input data file in the *Filepath* field.

Step 4

Enter a description for the data file in the *Description* field.

Step Result:

User must fill the fields in the following sections to complete the input data file creation process:

- Add System
- Add Data
- Add Subsystem
- Save Input Data File

Continue with the next task.

5.2

Add System

In this section:

- 5.2.1 Add Data
- 5.2.2 Add Subsystem

The user can provide the details by completing the fields in the *System* section.

Step 1

Click the *System* button.

Step 2

In the *New System* field, enter the name of the system name.

Step 3

Select the checkbox near the system name to make this system as default.

Note: If the user has not set the default system, the first system is considered as the default.

Step 4

Do you need to add data?

If YES:

Refer to [Add Data](#)

If NO:

Continue with the next step.

Step 5

Do you need to add subsystem?

If YES:

Refer to [Add Subsystem](#)

If NO:

Continue with the next step.

Step 6

Click the *Save* button.

Note: Click the *Delete* icon to delete an existing system.

5.2.1

Add Data

This procedure describes how to add data for an input data file.



Remember: This procedure is optional.

Step 7

Click the *Add Data* button or the *plus* icon.

Step Result:



Figure 71
Add Data Icon

Step 8

From the *Data* key drop-down list, select the appropriate data key name.

Note: User can either select an existing data key name from the list or enter a new data key name. This field is mandatory.

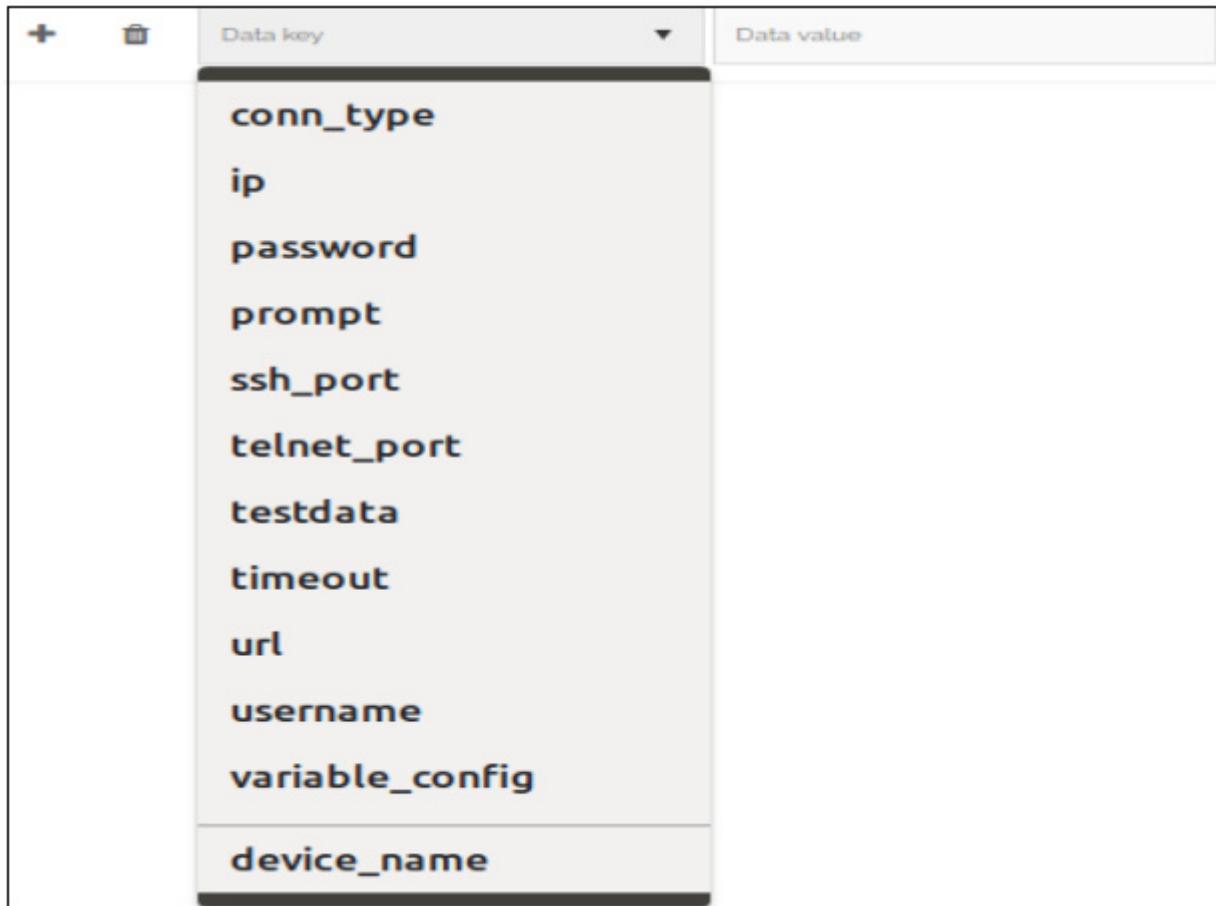


Figure 72
Data Key Drop-Down List

Step 9
Enter a corresponding value in the *Data value* field.

Step 10
Click the *Sub Data* icon to create a child data key.

Step 11
Repeat [Steps 8 through 9](#).

Step 12
Click the *Delete* icon to delete an existing sub data.

Step 13
Click the *Add Data* icon to add another data key and repeat the preceding steps.

5.2.2

Add Subsystem

This procedure describes how to add subsystem for an input data file.



Remember: This procedure is optional.

Step 14

Click the *Add Subsystem* button or the *plus* icon.

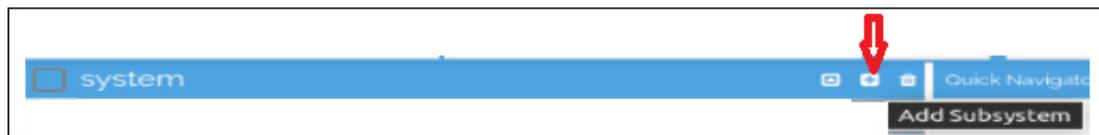


Figure 73
Add Data Icon

Step 15

In the *New Subsystem* field, enter the subsystem name.

Step 16

From the *Data key* drop-down list, select the appropriate data key name.

Note: User can either select an existing data key name from the list or enter a new data key name. This field is mandatory.

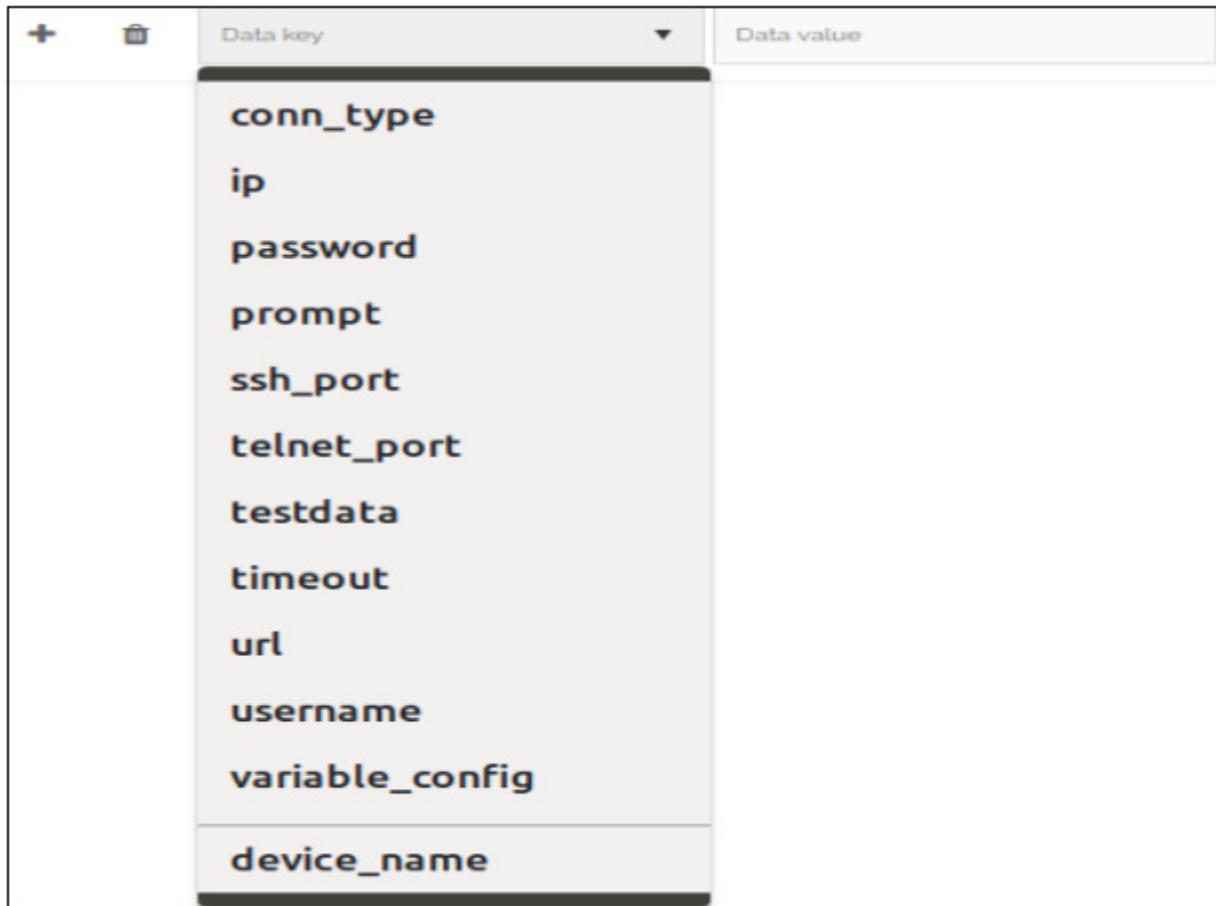


Figure 74
Data Key Drop-Down List

Step 17
Enter a corresponding value in the *Data value* field.

Step 18
Click the *Sub Data* icon to create a child data key.

Step 19
Repeat [Steps 16 through 17](#).

Step 20
Click the *Delete* icon to delete an existing sub data.

Step 21
Click the *Add Data* icon to add another data key and repeat the preceding steps.

This task is complete.

5.3

Save Input Data File

This procedure describes how to save the input data file.

Step 1

Click the Save button.

Note: Click *Cancel* to discard the input file creation.

Step Result:

The created input data file is shown in the following figure.

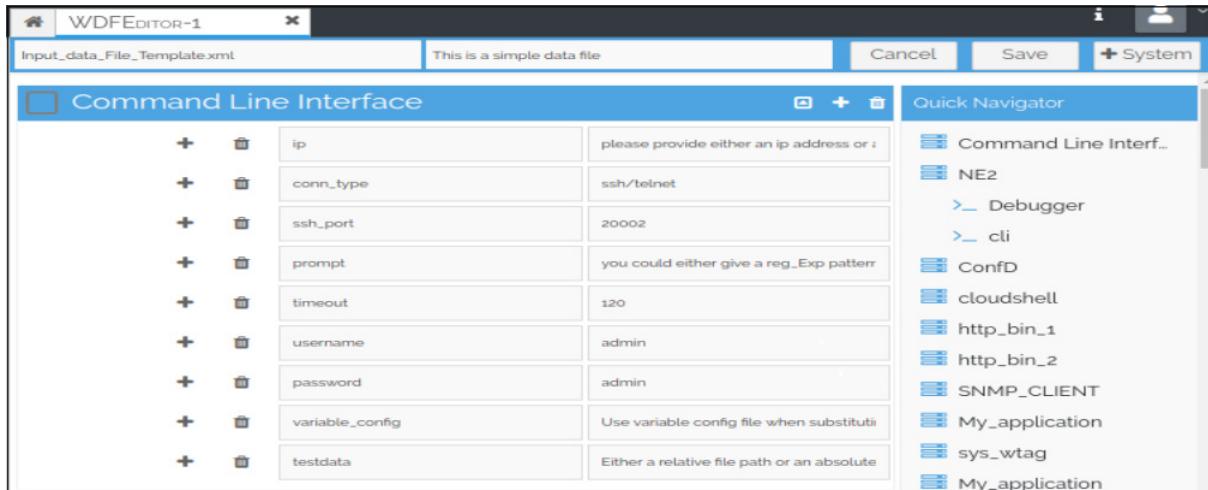


Figure 75
Add Data Icon

6

Managing CLIData File

In this chapter:

- 6.1 Create CLIData File
- 6.2 Save CLIData File

6.1

Create CLIData File

In this section:

- 6.1.1 Global Command Parameters
- 6.1.2 Global Verifications
- 6.1.3 Global Verification Combinations
- 6.1.4 Add CLI Data Block
- 6.1.5 Add CLI Data Command

Katana allows the user to create a CLIData file using the CLIData application. The procedure describes how to create a CLIData file.

Step 1

Launch Katana and click the *CLI DATA* icon.

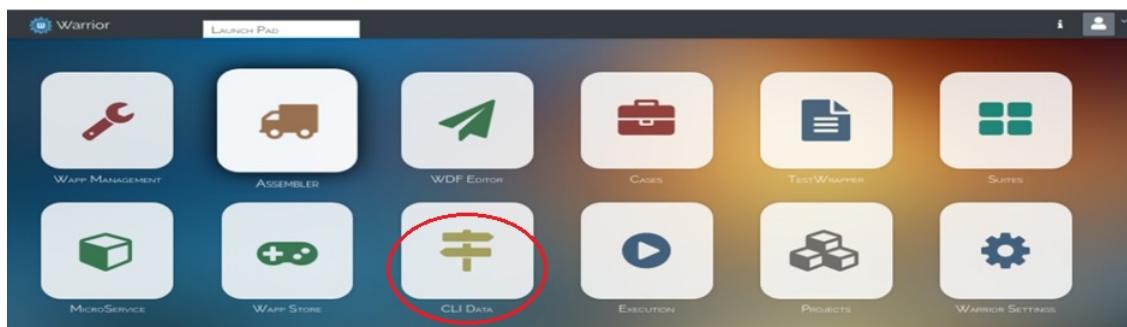


Figure 76
CLI DATA Icon

Step Result:

The *CLIDATA* page opens.



Figure 77
CLIData Page

Step 2

Click the *New* button.

Step Result:

The *CLIData* page opens.

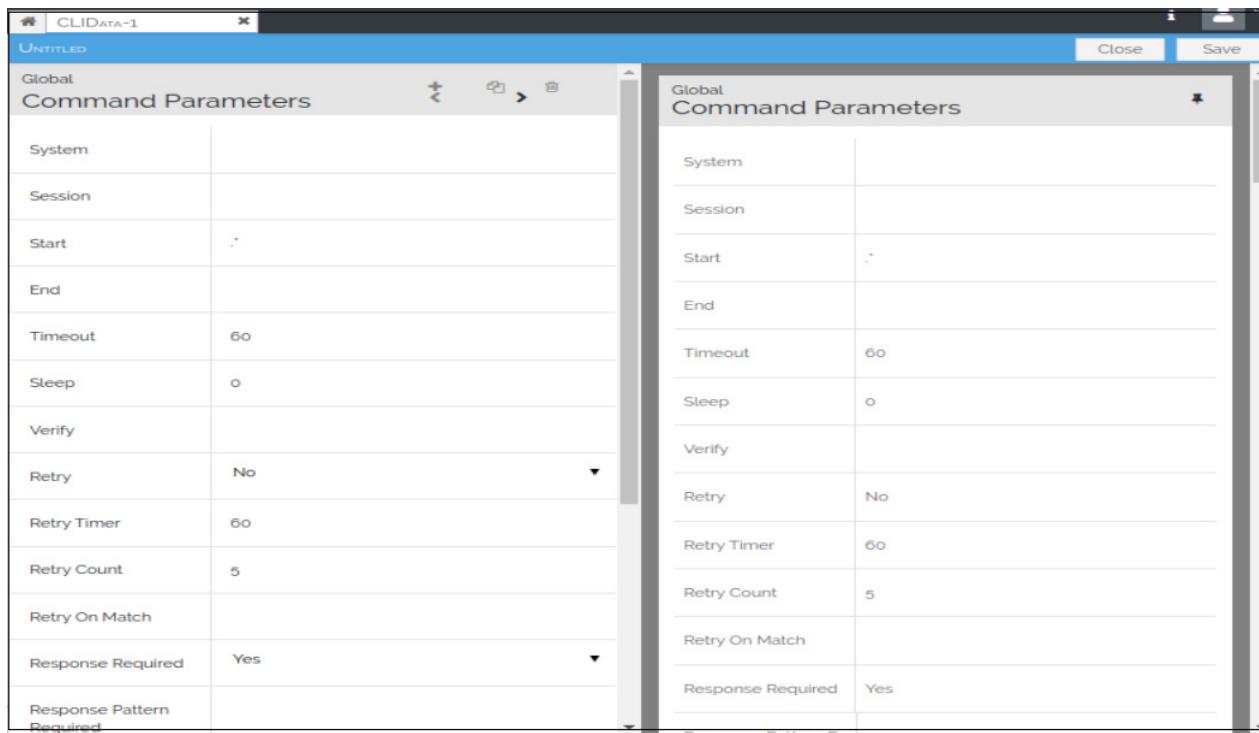


Figure 78
CLIData Page

Note: The global sections are the default sections. The missing information in the testdata file are fetched from these sections.

6.1.1

Global Command Parameters

The user can provide the details in CLIData file by completing the fields in the *Global Command Parameters* section.

Step 3

Enter the system and subsystem details on which the command should be executed in the *System* field.

Notes: The user can enter either system details or subsystem details or both. Enter these details in the following format:

- System only—Provide the system name, for example, NE1, server1
- Subsystem only—Provide the subsystem name enclosed in square brackets, for example, [cli], [dip], [interface1]. In this case, the system name provided in testcase is used.
- System and subsystem—Provide the system and subsystem combination, for example, NE1[cli], server1[interface1], where NE1 and server1 are the system names, and [cli] and [interface1] are the subsystem names.

Step 4

Enter the session name of the system or subsystem connected to in the *Session* field.

Step 5

Enter the starting prompt of the command in the *Start* field.

Note: The default value is *.

Step 6

Enter the ending prompt of the command in the *End* field.

Step 7

Enter the time required to wait to receive the end prompt in the *Timeout* field.

Note: The default value is 60 seconds.

Step 8

Enter the wait time required after completion of a command in the *Sleep* field.

Step 9

Enter the relevant verification tags in the *Verify* field.

Note: If multiple verifications are required for a command, provide tag names separated by a comma, for example, v1, v2, v3.

Step 10

Select the retry status from the *Retry* drop-down list.

Note: The default value is No.

Step 11

Enter the time interval between subsequent "retrials" in the *Retry Timer* field.

Note: The default value is 60 seconds.

Step 12

Enter the number of attempts to resend the command in the *Retry Count* field.

Step 13

Enter the text or regular expression in the *Retry On Match* field.

Note: If the user sets *Retry* to *Yes*, the retrials are attempted only when the response of the command has the provided text or regular expression.

Step 14

Select the relevant option from the *Response Required* drop-down list to save the response command.

Note: The default value is *Yes*.

Step 15

Enter the text or regular expression in the *Response Pattern Require* field to save the mentioned text or regular expression from the response of the command to the directory.

Step 16

Enter the reference text for the response in the *Reference Response* field.

Note: If the user does not provide any response reference, the framework automatically assigns the value based on the position of the command in the testdata section. For example, if the command is the second command in the testdata, the text *resp_ref="2"* is added.

Step 17

Enter the system names to be printed on the console in the *Monitor* field. For example, NE1[cli].session, NE2, [dip], where NE1 and NE2 are the system names, [cli] and [dip] are the subsystem names, and .session is the session name.

Step 18

Select the appropriate option from the *In-order* drop-down list.

Note: This option confirms if the command passes only when the received response order matches with the order given in the verify tag. The default value is *No*.

Step 19

Select the relevant option from the *Repeat* drop-down list to set iteration for the testdata commands without any iteration pattern.

Note: When *Iteration Type* is *per CLI Data block* with *repeat='yes'*, the relevant command in testdata block is executed in each iteration. The default value is *No*.

Continue with the next task.

6.1.2

Global Verifications

The user can provide the details in CLIData file by completing the fields in the *Global Verifications* section.

Step 20

Enter the verification name in the *Verification Name* field.

Step 21

Enter the string to be searched in the *Search* field.

Step 22

Select relevant option from the *Found* drop-down list.

Note: The option *Yes* searches for the presence of search string and *No* searches for the absence of search string.

Step 23

Enter the details of the system in the *Verify On* field on which the verification needs to be performed.

Step 24

Enter the verification condition value in the *Condition Value* field.

Step 25

Select the verification condition type from the *Condition Type* drop-down list. The default value is *String*.

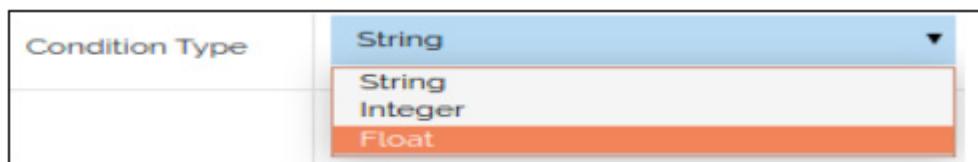


Figure 79

Condition Type Drop-Down List

Step 26

Select the verification operator type from the *Operator* drop-down list. The default value is *Equal To*.

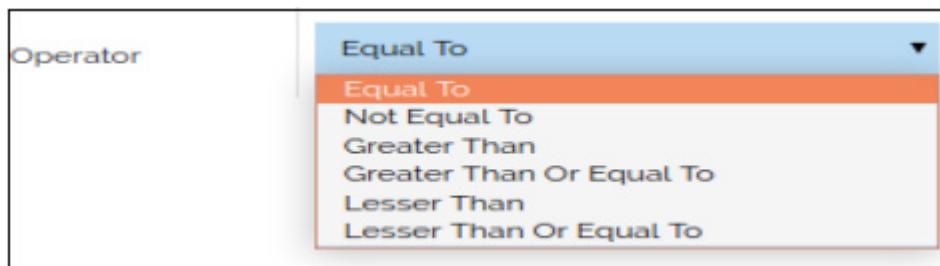


Figure 80

Operator Drop-Down List

Continue with the next task.

6.1.3

Global Verification Combinations

The user can provide the details in CLIData file by completing the fields in the *Global Verification Combinations* section.

Step 27

Enter the combination name in the *Combination Name* field.

Continue with the next task.

6.1.4

Add CLI Data Block

Perform the following procedure to add testdata block.

Step 28

Scroll and click the *CLI Data Block* pane.

Step Result:

The *CLI Data Block* form appears.

Figure 81
CLI Data Block Form

The fields in the *CLI Data Block* form are explained in the following table.

Table 12
CLI Data Block Form Field Description

Field	Description
Title	User-specified title of the CLI data block

Table 12 (Cont.)

CLI Data Block Form Field Description

Field	Description
Row Number	User-specified row number of the CLI data block
Execute	Indicates if the CLI data block must be executed or not
Monitor	User-specified CLI data block level monitor attribute
Iteration Type	Displays the list of testdata iteration types

Step 29

Enter the CLI data block title in the *Title* field.

Step 30

Enter the row number in the *Row Number* field.

Step 31

Select the appropriate *Execute* type from the drop-down list.

Note: The default value is Yes for *Execute* type

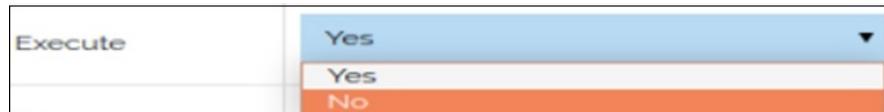


Figure 82

Execute Options

Step 32

Enter the relevant attribute value in the *Monitor* field.

Step 33

Select the appropriate *Iteration Type* from the drop-down list.

Note: The default value is *Per CLI-Data Block* for *Iteration Type*.

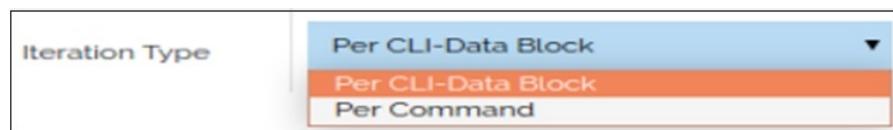


Figure 83

Iteration Type Options

Continue with the next task.

6.1.5

Add CLI Data Command

Step 34

Scroll and click the *CLI Data Command* pane.

Step Result:

The *CLI Data Command* form appears.

The screenshot shows a configuration form titled "CLI Data Command". The form has a header with three icons: a plus sign, a magnifying glass, and a trash can. Below the header is a table with the following rows:

Send	
System	
Session	
Start	:
End	
Timeout	60
Sleep	0
Verify	
Retry	No
Retry Timer	60
Retry Count	5

Figure 84
CLI Data Command Form

Step 35

Enter the relevant send command for the CLI data block in the *Send* field.

Step 36

Refer to [Global Command Parameters](#) to fill the remaining fields.

Continue with the next task.

6.2

Save CLIData File

This procedure describes how to save the CLI data file.

Step 1

Click the Save button.



Figure 85
Save CLI Data File

Note: Click *Cancel* to discard the CLI file creation.

Step Result:

A message dialog box opens.

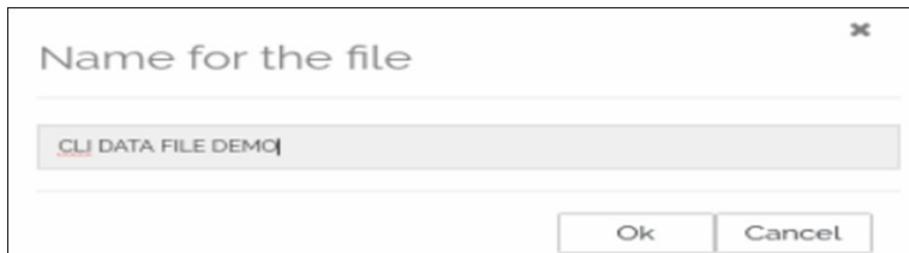


Figure 86
Message Dialog Box

Step 2

Enter the filename and click *Ok*.

7

Managing Suite

In this chapter:

- 7.1 Create Suite
- 7.2 Add Details
- 7.3 Add Requirements
- 7.4 Add Cases
- 7.5 Save Suite

7.1

Create Suite

Katana allows the user to create Suites using the Suites application. This procedure describes how to create a suite.

Step 1

Launch Katana and click the *SUITES* icon.

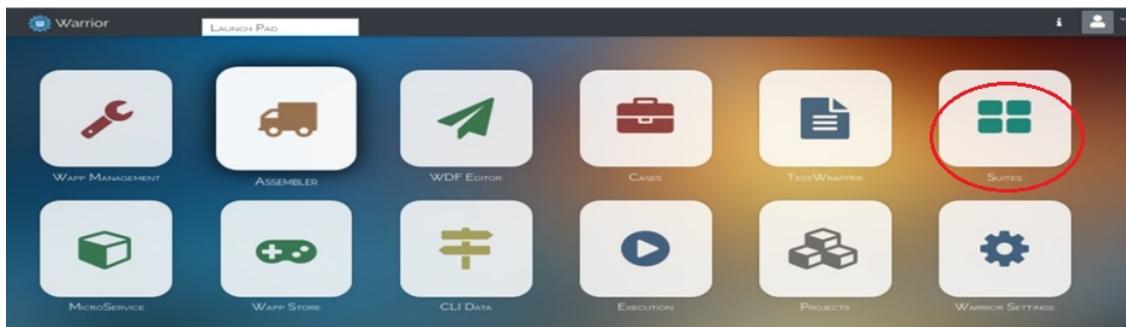


Figure 87
SUITES Icon

Step Result:

The *Suites* page opens.

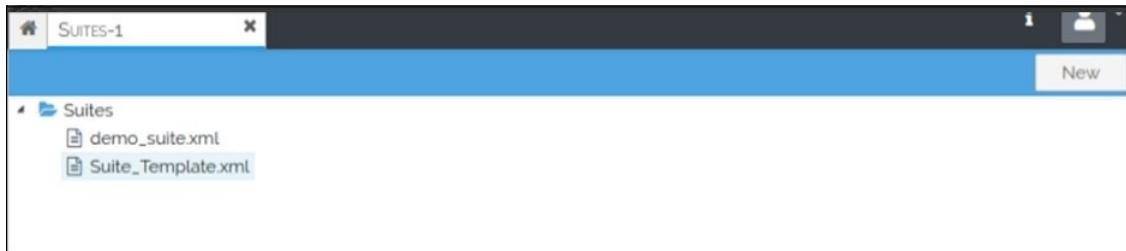


Figure 88
Suites Page

Step 2

Click the *New* button.

Step Result:

The *Suites* Page opens.

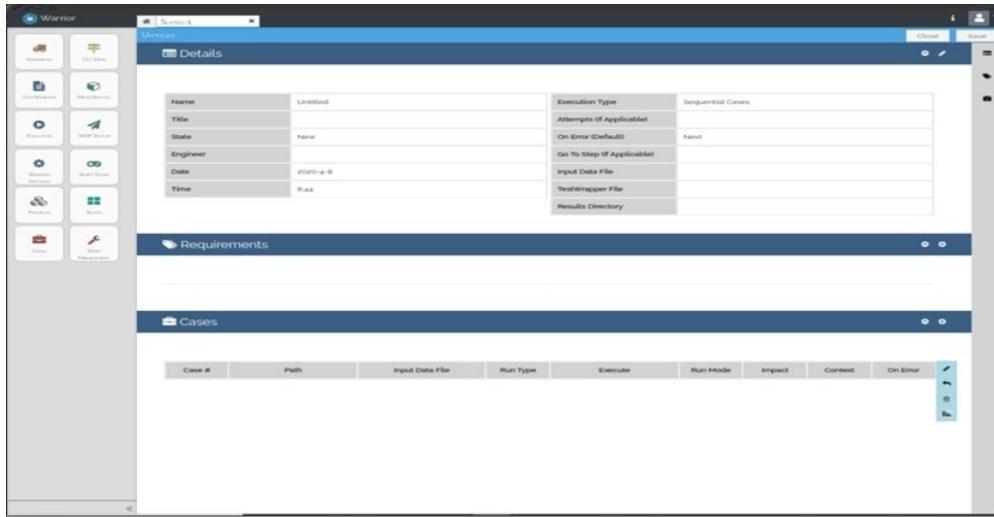


Figure 89
Suites Page

User must fill the fields in the following sections to complete the case creation process:

- Details
- Requirements
- Cases

Continue with the next task.

7.2

Add Details

This procedure describes how to add suite details. The user can provide the suite details by filling the fields in the *Details* section. The fields in the *Details* section are explained in the following table.

Table 13
Details Field Description

Field	Description
Name ¹	Name of the suite that the user can recognize
Title ¹	Description of the suite
State	Keeps track of the cases that are new, released, or in review
Engineer ¹²	Displays the name of the engineer who created the case
Date ³	Displays the date on which the Suite is updated
Time ²	Displays the time on which the Suite is updated
Execute Type ¹	Displays options to execute a Suite
On Error (Default)	Specifies the action the Suite should take when a case error occurs
Go To Step	User-specified case number that redirects Warrior in case of failure <i>Note:</i> This field appears when <i>Go To</i> option is selected for <i>On Error (Default)</i> field.
Input Data File	Sets the path for an Input Data file
Test Wrapper File	Sets the path for a Test Wrapper file
Results Directory	Specifies the path to store the results returned by Warrior

¹ This field is mandatory.

² This prefilled field is editable.

³ This field is a prefilled field.

Step 1

Click the *Edit Details* button to edit or add data to the Details section.

Step 2

In the *Name* field, enter the name of the suite.

Step 3

In the *Title* field, enter a descriptive title.

Step 4

Select the appropriate *State* from the drop-down list: *New*, *Test-Assigned*, or *Released*.

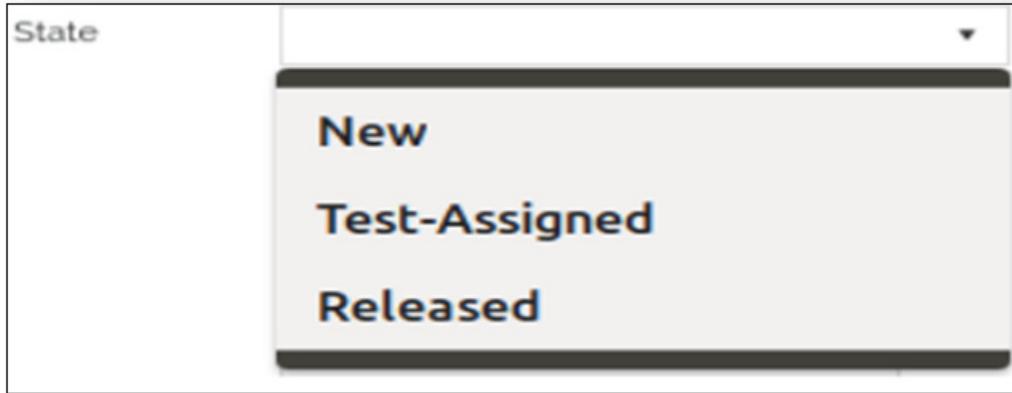


Figure 90
State Options

Note: This field is an optional field, but Fujitsu recommends to fill out this field to keep track of state of development of each suite.

Step 5

In the *Engineer* field, enter the name of the engineer who created the case.

Step 6

Select the *Execution Type* from the drop-down list.

Step 7

Select the appropriate *On Error (Default)* option from the drop-down list: *Next*, *Abort*, *Abort As Error*, or *Go To*.

Note: Default value, *On Error* field selects *Next*, if no changes are made, the suite proceeds to the next suite available for execution when the current suite throws an error.

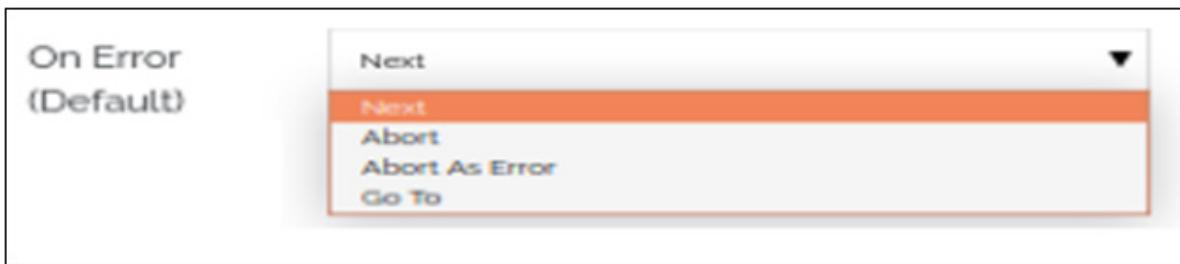


Figure 91
On Error Options

Note: Selecting *Abort* and *Abort As Error* options terminate the execution of the case if the step does not pass.



Notice: When *Go To* option is selected, a *Go To Step* field appears. Enter the case number that Warrior must refer to in case of an error.

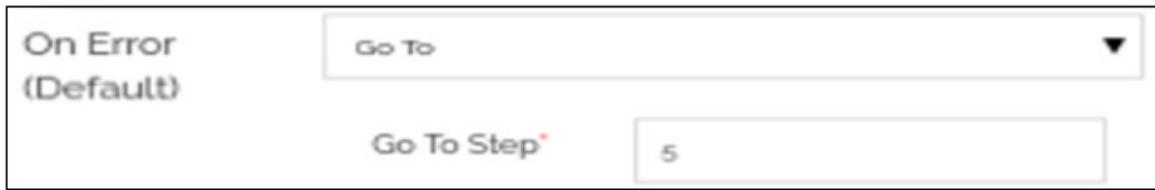


Figure 92
Go To Step Field

Step 8

In the *Input Data File* field, select the path to the desired data file for the case to use.



Hint: Click the *Folder* icon to select a file from the *Select a Path* window and click *OK*.

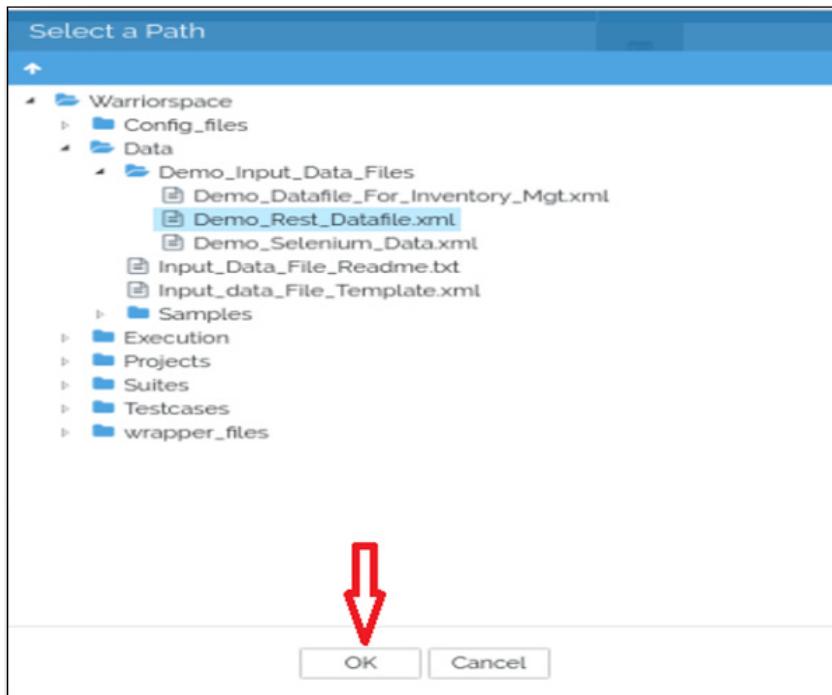


Figure 93
Select a Path Window



Hint: User can directly enter the file path into the field.

Step 9

In the *Test Wrapper File* field, set a path to the desired test wrapper file for the suite to use.



Hint: Click the *Folder* icon to select a file from the *Select a Path* window and click *OK*.

Step 10

In the *Results Directory* field, set the path to store the results.

Note: If the location is not set in the *Results Directory* field, the results are directed and stored in the default location.

Continue with the next task.

7.3

Add Requirements

This procedure describes how to add requirements.



Remember: This procedure is optional.

The following table lists the icons for the *Requirements* section.

Table 14
Requirements Icon Description

Icon	Label	Description
Requirement Section		
	Add New Requirement	Adds new requirements
	Delete	Deletes new requirements
New Requirements Dialog Box		
	Discard	Discards requirement changes
	Save	Saves new requirements

Step 1

Click the *Add New Requirement* button or the *plus* icon.

Step Result:

The *New Requirements* dialog box opens.

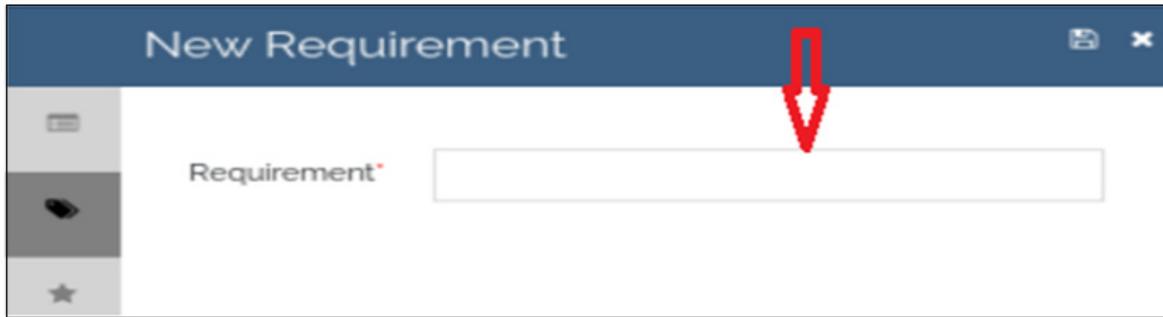


Figure 94
New Requirements Dialog Box

Step 2

In the *Requirement* field, enter the requirement.

Step 3

After entering the requirement, click the save icon to save the requirement.

Step Result:

New requirement is saved.

Continue with the next task.

7.4

Add Cases

This procedure describes how to add a Case. The user can provide the suite cases by filling the fields in the *Cases* section.

The following table lists the icons for the *Cases* sections.

Table 15
Cases Icon Description

Icon	Label	Description
Step Section		
	Add New Case	Adds new case steps
	Delete	Deletes new case steps
	Edit	Edits created case steps
New Steps Dialog Box		
	Discard	Discards case step changes
	Save	Saves new case steps

Step 1

Click the *Add new Case* button or the *plus* icon.

Step Result:

The *New Case* dialog box opens.

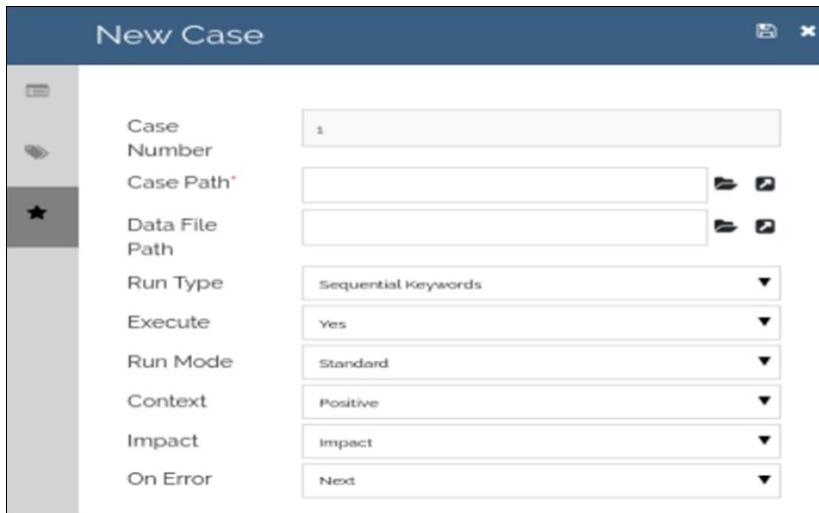


Figure 95
New Case Dialog Box

The fields in the *New Case* dialog box are explained in the following table.

Table 16
New Case Fields

Field	Description
Case Number ¹	Displays the case number
Case Path ²	Specifies the location of the case
Data File	Specifies the path to store a data file
Run Type	Indicates how the steps in a case should run
Execute	Displays options to execute a case
Condition ³	Specifies the condition for <i>If</i> or <i>If Not</i> Execute Type
Condition Value ³	Specifies the key value for the provided <i>If</i> or <i>If Not</i> condition. This drop-down provides the following options: <ul style="list-style-type: none"> ■ PASS ■ FAIL ■ ERROR ■ SKIPPED
Else ³	Specifies the condition to handle the scenario where the condition in <i>If</i> or <i>If Not</i> is not met and what Warrior needs to know in such a case
Run Mode	Displays options to run a step

Table 16 (Cont.)
New Case Fields

Field	Description
Value	Specifies the maximum number of times the step must run <i>Note:</i> This field appears when Run Mode is filled.
Context	Indicates if the step is a positive or negative test scenario
Impact	Enables the user to decide if the status of the step impacts the test case or not
On Error	Displays a list of error handling conditions to assign for a step
Go To Step	User-specified case number that redirects Warrior in case of failure <i>Note:</i> This field appears when <i>Go To</i> option is selected.

1 This field is prefilled and noneditable.

2 This field is mandatory.

3 This field appears when *If* or *If Not* Execute type is selected.

Step 2

In the *Case Path* field, click the *Folder* icon.

Step Result:

The *Select a Path* window opens.

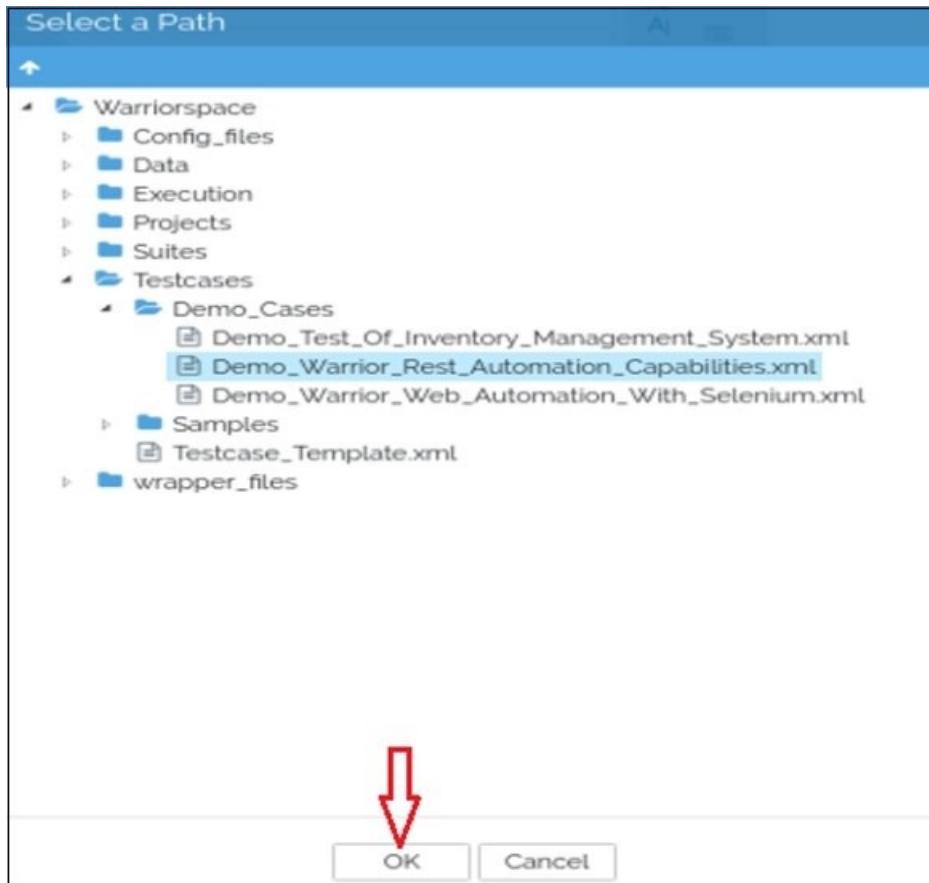


Figure 96
Select a Path Window

Step 3
Select a path to the case and click *OK*.



Hint: Manually add a path to the case.

Step 4
In the *Data File Path* field, click the *Folder* icon.

Step Result:
The *Select a Path* window opens.

Step 5
Select a path and click *OK*.



Hint: User can directly enter the file path into the field.

Step 6

Select an appropriate *Run Type* option from the drop-down list: *Sequential_Keywords* or *Parallel_Keywords*.

Note: The default value is *Sequential_Keywords*.

Step 7

Select the appropriate *Execute* type from the drop-down list: *Yes*, *No*, *If*, or *If Not*.

Note: The default value is *Yes* for *Execute* type

Note: If *If* and *If Not* are selected, new input boxes appear to add conditions.

Figure 97

Condition, Condition Value, and Else Fields

Step 8

Select the appropriate *Run Mode* from the drop-down list: *Standard*, *Run Multiple Times*, *Run Until Failure*, or *Run Until Pass*.

Step Result:

The *Value* field appears.

Figure 98

Value Field

Step 9

In the *Value* field, enter the number of times the step must run.

Step 10

Select the appropriate *Context* from the drop-down list: *Positive* or *Negative*.

Note: The default value is *Positive*.

Step 11

Select the appropriate *Impact* from the drop-down list: *Impact* or *No-impact*.

Note: The default value is *Impact*.

Step 12

Select the error condition from the *On Error* drop-down list.



Notice: If *Go To* is selected, the *Go To Step* field appears. The user must specify the case number for Warrior to redirect in case of failure.

Step 13

Click *Save Case*.



Important: A case must be saved.

Step Result:

A case report appears as shown in the following figure.

Case #	Path	Input Data File	Run Type	Execute	Run Mode	Impact	Context	On Error	Actions
1	./Testcases/Testcase_Template.xml	./Data/Demo_Input_Data_Files/De m0_Rest_Datafile.xml	Sequential Key words	Yes	Standard	Positive	Impact	Next	↶ ⌂ ⌂

Figure 99
Case Report

Continue with the next task.

7.5

Save Suite

This procedure describes how to save a suite.

Step 1

Click *Save*.

Note: If any unsaved case is present when saving a suite, the unsaved case is automatically saved.



Figure 100
Save Suite

Step Result:

A message dialog box opens displaying the suite name.



Figure 101
Message Dialog Box

Step 2

Click *Ok* to save the testsuite.

Step Result:

The created suite appears as shown in the following figure.

Details

Name	ts_common_action_suite_wrapper	Execution Type	Run Multiple Times
Title	ts_common_action_suite_wrapper	Attempts (if Applicable)	2
State	New	On Error (Default)	Next
Engineer	Warrior_user	Go To Step (if Applicable)	
Date	11/17/2017	Input Data File	No_Data
Time	10:37:44	TestWrapper File	./wrapper_files/common_actions_tests/tw_store_in_repo.x...
Results Directory			

Requirements

Cases

Case #	Path	Input Data File	Run Type	Execute	Run Mode	Impact	Context	On Error
1	./testcases/common_acti...	None	Sequential...	Yes	Standard	Positive	Impact	Next

Figure 102
Created Suite Page

This task is complete.

8

Managing Projects

In this chapter:

- 8.1 Create Project
- 8.2 Add Details
- 8.3 Add Suites
- 8.4 Save Project

8.1

Create Project

Katana allows the user to create projects using the Projects application. This procedure describes how to create a project in Katana.

Step 1

Launch Katana and click the *PROJECTS* icon.

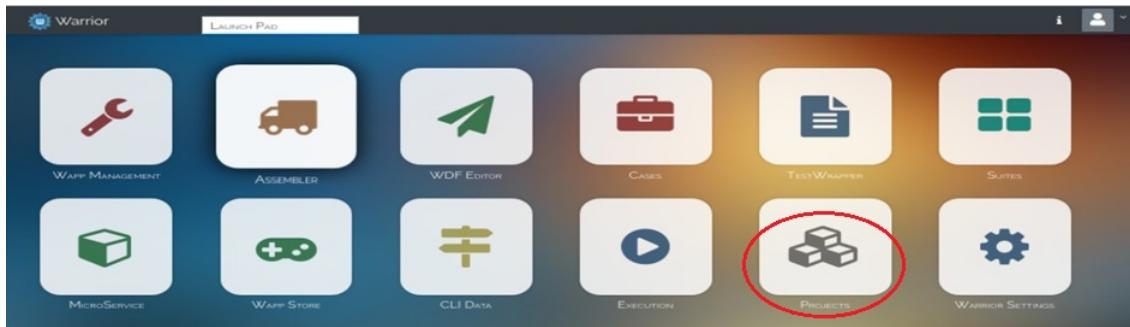


Figure 103
PROJECTS Icon

Step Result:

The *Projects* page opens.

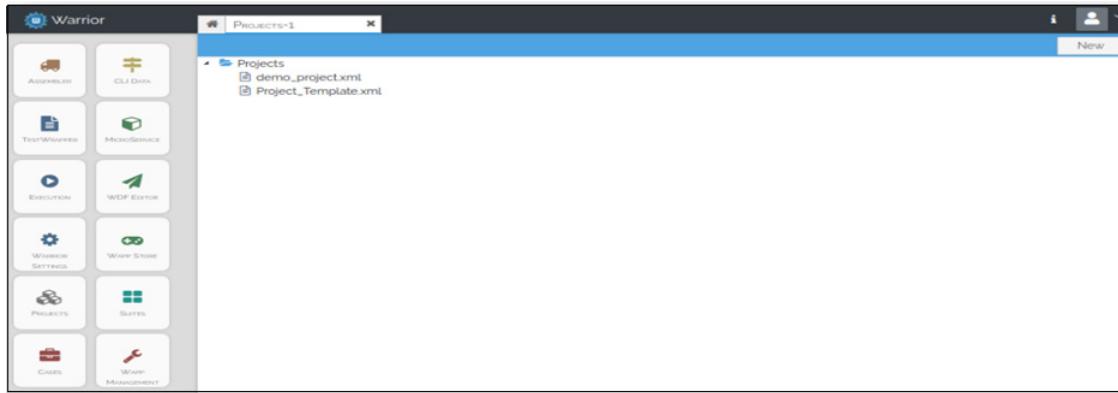


Figure 104
Projects Page

Step 2

Click the *New* button.

Step Result:

The *Projects* page opens.

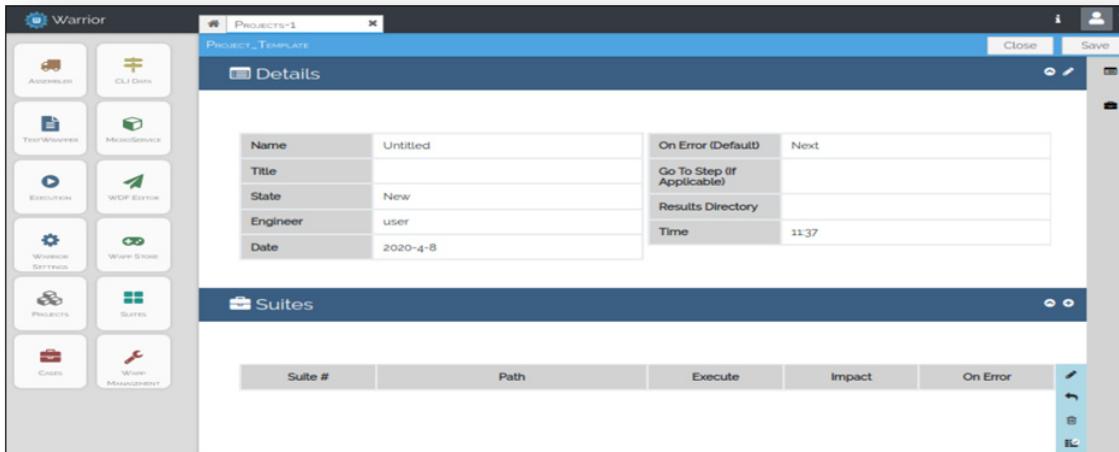


Figure 105
Project Page

Continue with the next task.

8.2

Add Details

This procedure describes how to add project details. The user can provide the project details by filling the fields in the *Details* section.

The fields in the *Details* section are explained in the following table.

Table 17
Details Field Description

Field	Description
Name ¹	Name of the project that the user can recognize
Title ¹	Description of the project
State	Keeps track of the cases that are new, released, or in review
Date ²	Displays the date on which the project is updated
Time ²	Displays the time on which the project is updated
On Error (Default)	Specifies the action the project should take when a case error occurs
Go To Step	User-specified case number that redirects Warrior in case of failure <i>Note:</i> This field appears when <i>Go To</i> option is selected.
Results Directory	Specifies the path to store the results returned by Warrior

¹ This field is mandatory.

² This field is a prefilled field.

Step 1

In the *Name* field, enter the name of the project.

Step 2

In the *Title* field, enter a descriptive title.

Step 3

Select the appropriate *State* from the drop-down list: *New*, *Test-Assigned*, or *Released*.

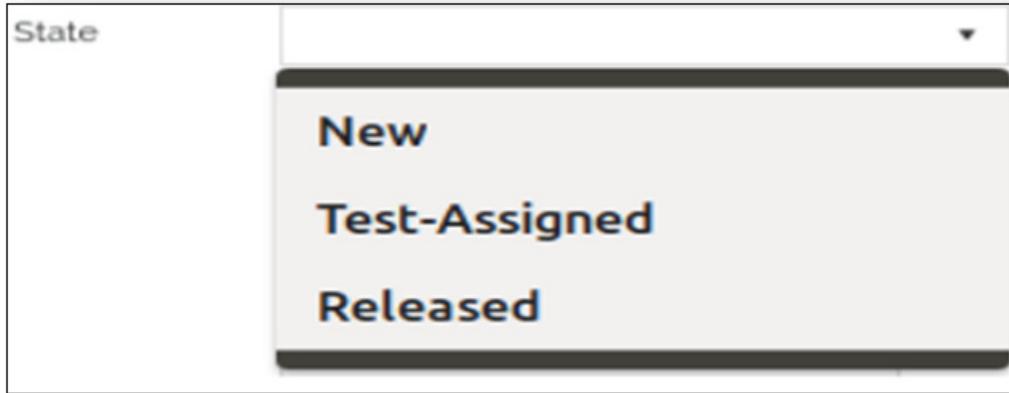


Figure 106
State Options

Note: This field is an optional field, but Fujitsu recommends to fill out this field to keep track of state of development of each project.

Step 4

Select the appropriate *On Error (Default)* option from the drop-down list: *Next*, *Abort*, *Abort As Error*, or *Go To*.

Note: By default, *On Error* field selects *Next*. If no changes are made, the project proceeds to the next project available for execution when the current project throws an error.

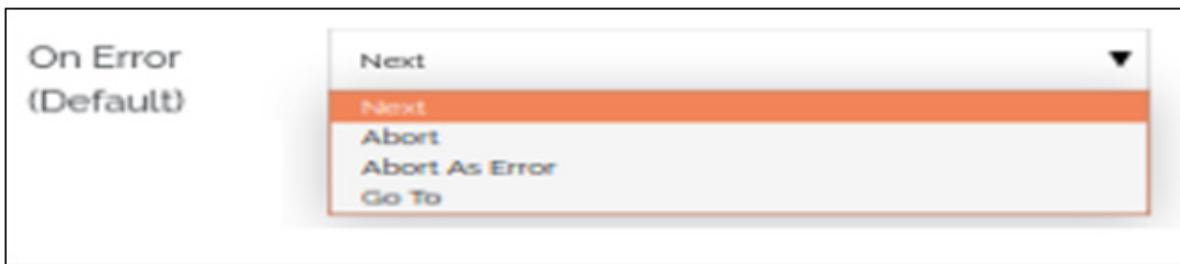


Figure 107
On Error Options

Note: Selecting *Abort* and *Abort As Error* options terminate the execution of the case if the step does not pass.



Notice: When *Go To* option is selected, a *Go To Step* field appears. Enter the project number that Warrior must refer to in case of an error.

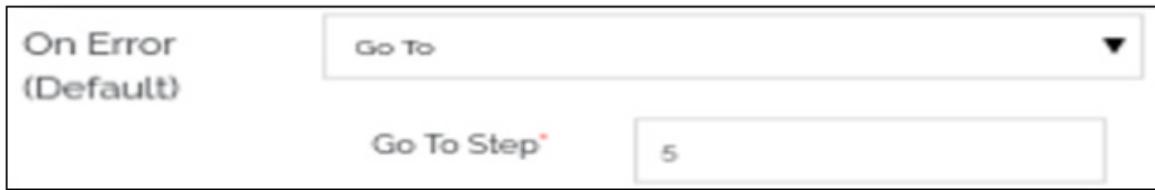


Figure 108
Go To Step Field

Step 5

In the *Results Directory* field, set the path to store the results.

Note: If the location is not set in the *Results Directory* field, the results are directed and stored in the default location.

Continue with the next task.

8.3

Add Suites

This procedure describes how to add a project suite. The user can provide the project suites by filling the fields in the *Suites* section.

The following table lists the icons for the Suites section.

Table 18
Suites Icon Description

Icon	Label	Description
Suite Section		
	Add New Suite	Adds new project suites
	Delete	Deletes new project suites
	Edit	Edits created project suites
New Suites Dialog Box		
	Discard	Discards project suites changes
	Save	Saves new project suites

Step 1

Click the *Add New Suite* button to add a suite to the project.

Step Result:

The *New Suite* dialog box opens.

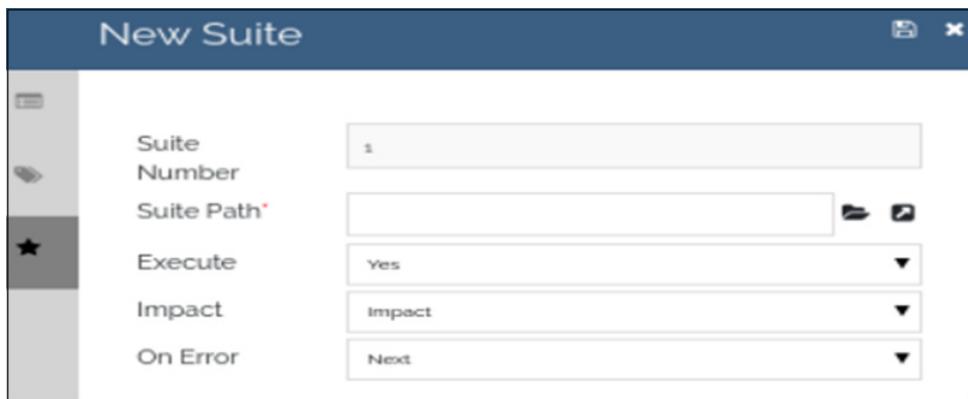


Figure 109
New Suite Dialog Box

The fields in the *New Suite* dialog box are explained in the following table.

Table 19
New Suite Fields

Field	Description
Suite Number ¹	Displays the suite number
Suite Path ²	Specifies the location of the suite
Execute	Displays options to execute a suite
Condition ³	Specifies the condition for <i>If</i> or <i>If Not</i> Execute Type
Condition Value ³	Specifies the key value for the provided <i>If</i> or <i>If Not</i> condition. This drop-down provides the following options: <input checked="" type="checkbox"/> PASS <input checked="" type="checkbox"/> FAIL <input checked="" type="checkbox"/> ERROR <input checked="" type="checkbox"/> SKIPPED
Else ³	Specifies the condition to handle the scenario where the condition in <i>If</i> or <i>If Not</i> is not met and what Warrior needs to know in such a case
Impact	Enables the user to decide if the status of the step impacts the test case or not
On Error	Displays a list of error handling conditions to assign for a step
Go To Step	User-specified case number that redirects Warrior in case of failure <i>Note:</i> This field appears when <i>Go To</i> option is selected.

¹ This field is prefilled and noneditable.

² This field is mandatory.

³ This field appears when *If* or *If Not* Execute type is selected.

Step 2

In the *Suite Path* field, click the *Folder* icon.

Step Result:

The *Select a Path* window opens.

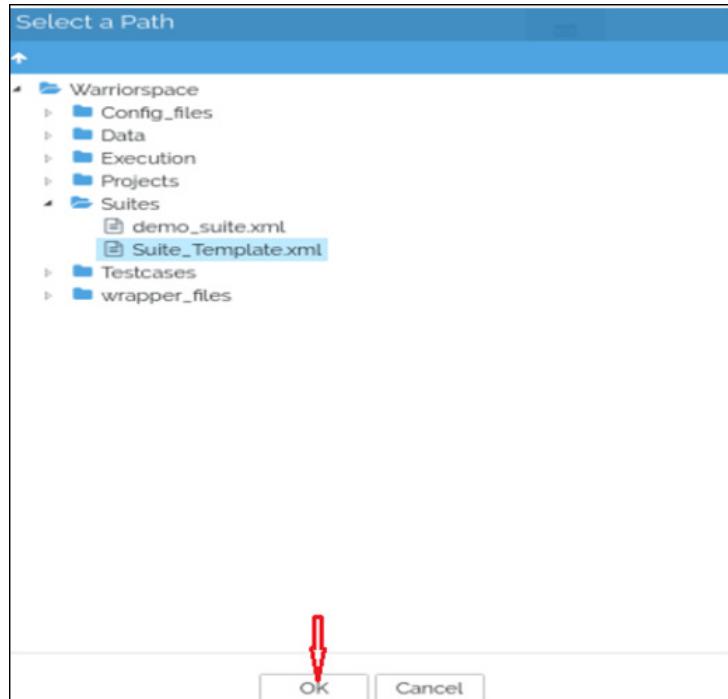


Figure 110
Select a Path Window

Step 3

Select a path to the suite and click *OK*.



Hint: Manually add a path to the suite.

Step 4

Select the appropriate *Execute* type from the drop-down list: *Yes*, *No*, *If*, or *If Not*.

Note: The default value is *Yes* for *Execute* type.

Note: If *If* and *If Not* are selected, new input boxes appear to add conditions.

The screenshot shows a configuration dialog for an 'Execute' step. At the top, there is a dropdown menu labeled 'If'. Below it, there are three input fields: 'Condition*', 'Condition Value*', and 'Else*'. The 'Condition' field has a dropdown arrow pointing down. The 'Condition Value*' field contains two empty text boxes stacked vertically. The 'Else*' field has a dropdown arrow pointing down.

Figure 111
Condition, Condition Value, and Else Fields

Step 5

Select the *Impact* from the drop-down list: *Impact* or *No-impact*.

Note: The default value is *Impact*.

Step 6

Select the error condition from the *On Error* drop-down list.



Notice: If *Go To* is selected, the *Go To Step* field appears. The user must specify the suite number for Warrior to redirect in case of failure.

Step 7

Click *Save Suite*.



Important: A suite must be saved.

Step Result:

A suite report appears as shown in the following figure.

Suite #	Path	Execute	Impact	On Error	
1	./Suites/Suite_Template.xml	Yes	Impact	Go To - 5	

Figure 112
Suite Report

Continue with the next task.

8.4

Save Project

This procedure describes how to save a project.

Step 1

Click *Save*.

Note: If any unsaved suite is present when saving a project, the unsaved suite is automatically saved.



Figure 113
Save Project

Step Result:

A message dialog box opens displaying the project name.

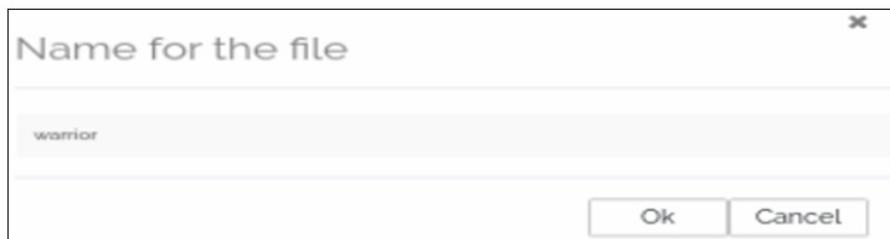


Figure 114
Message Dialog Box

Step 2

Click *Ok* to save the project.

Step Result:

The created project appears as shown in the following figure.

The screenshot shows a software interface for managing projects. At the top, there's a header bar with the Fujitsu logo and navigation links like 'PROJECTS-1', 'HOME', 'LOGOUT', and 'COMMON ACTIONS'. Below the header is a 'Details' section containing a table with project information:

Name	pj_common_actions	On Error (Default)	Next
Title	pj_common_actions	Go To Step (If Applicable)	
State	New	Results Directory	
Engineer	Arvind Kumar Sethuraman	Time	13:07:26
Date	23/01/2017		

Below the details is a 'Suites' section, which is a table showing the configuration for two suites:

Suite #	Path	Execute	Impact	On Error	Actions
1	./suites/common_action_tests/common...	Yes	Impact	Next	
2	./suites/framework_tests/random_execu...	Yes	Impact	Next	

Figure 115
Created Project Page

This task is complete.

9

Executing via Katana

In this chapter:

- 9.1 Execute Case/Suites/Projects

9.1

Execute Case/Suites/Projects

Katana allows the user to execute case through the Execute application. This procedure describes how to execute a case.

Step 1

Launch Katana and click *EXECUTION* icon.

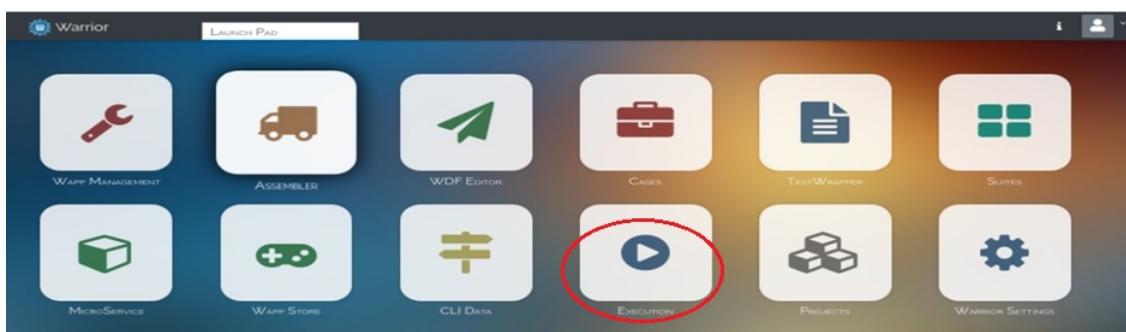


Figure 116
EXECUTION Icon

Step Result:

The *Execute* page opens displaying *Layout and Selections* pane.

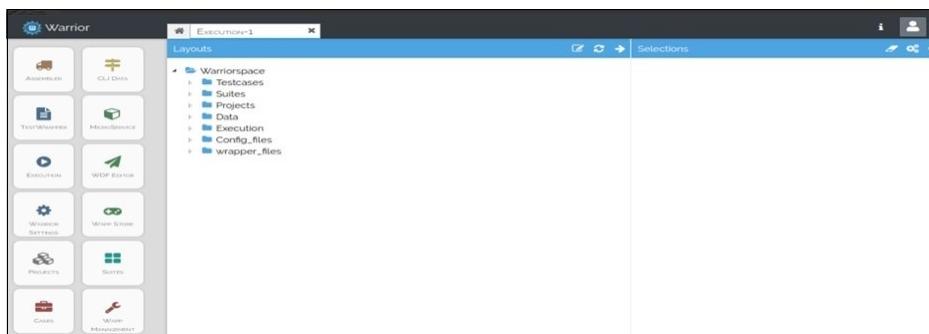


Figure 117
Execute Page

The *Layout* pane lists the cases to run below the Warriorspace.

Step 2

Click the *Layout settings* button or the *paper-pen* icon to pick the cases from local directory and configure the directory path of the XML files.

Step Result:

The *Layout Settings* dialog box opens.

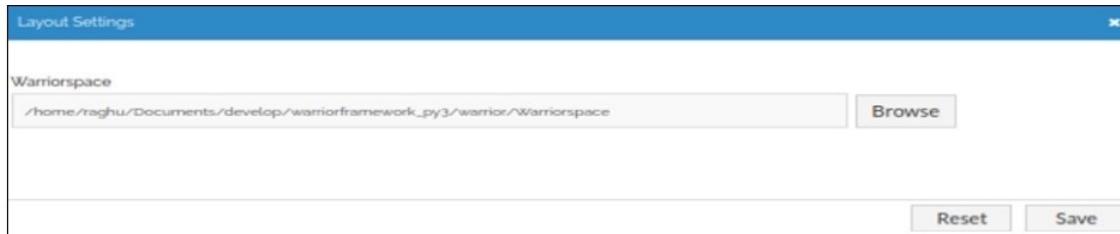


Figure 118
Layout Settings Dialog Box

Step 3

Click the *Browse* button from the *Warriospace* field.

Step Result:

The *Select a directory* window opens.

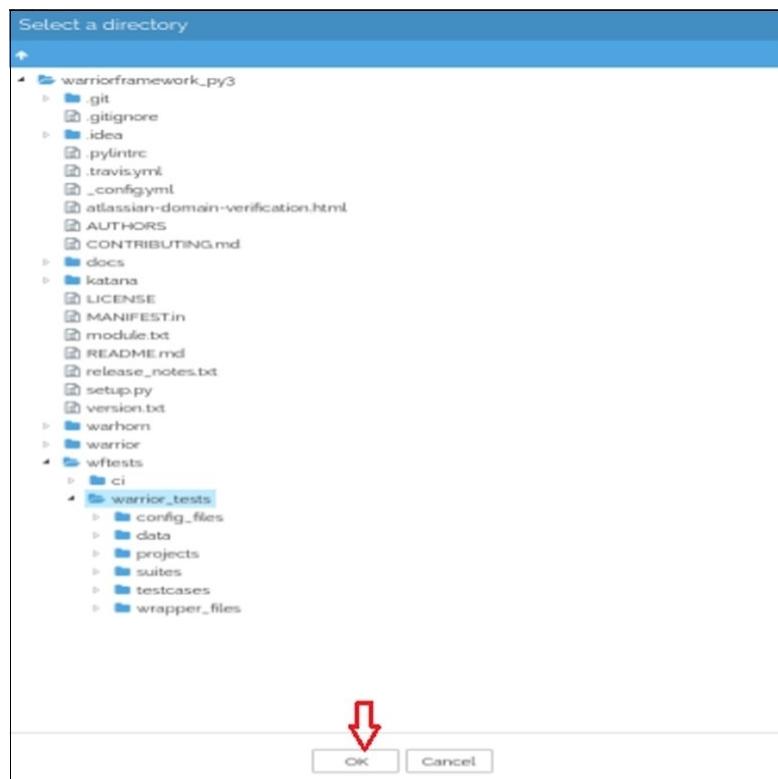


Figure 119
Select a directory Window

Step 4

Select the appropriate file and click *OK* to configure the directory path for executing the case.

Step 5

Click *Save*.

Step Result:

After saving the path, the executable files are displayed in the *Layouts* pane.

Step 6

Select a file and click *copy to selections area or arrow icon*.

Step Result:

The selected file appears in the *Selections* pane.

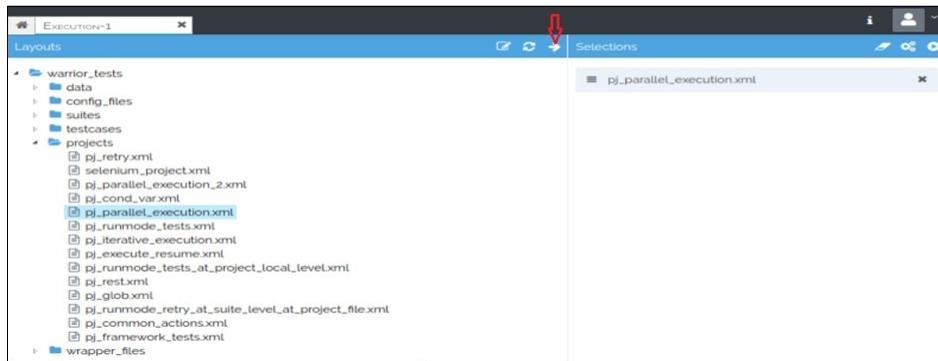


Figure 120
Selections Pane

Step 7

Click *Execute* to execute a testcase.

Step Result:

The *Console Logs* tab appears displaying the running case.

Executing via Katana

Execute Case/Suites/Projects



The screenshot shows the 'Execution' window with the 'Console Logs' tab selected. The window title is 'Execution'. The content area displays the following text:

```
Executing:  
1. ./home/ragh.../testcases/selenium_...  
Command:  
python3 /home/ragh.../warriorframework_py3/wftests/warrior_tests/testcases/selenium_...  
pythonpath /home/ragh.../warriorframework_py3  
.Logs:  
import os was successful  
import shutil was successful  
2020-04-08 15:10:59 ragh...-VirtualBox MainProcess INFO  
./home/ragh.../warriorframework_py3/warrior/Framework/Utils/csv_utils.py: xlrd module is not installed  
2020-04-08 15:10:59 ragh...-VirtualBox MainProcess INFO ::Install xlrd module in order to perform Excel to CSV or Excel to  
KML file conversion  
2020-04-08 15:10:59 ragh...-VirtualBox MainProcess INFO ::Please follow the instructions in  
http://xlrd.readthedocs.io/en/latest/installation.html to install the xlrd  
2020-04-08 15:10:59 ragh...-VirtualBox MainProcess INFO ::The Selenium Webdriver version is '3.8.0'  
import email was successful  
import Utils was successful  
import print_Utils was successful  
import testcase_driver, testsuite_driver, project_driver were successful  
import ironclaw_driver, framework_detail were successful  
import jira_rest_class was successful  
import database_utils_class was successful
```

Figure 121
Console Logs Tab

Note: Identify the status of the case from the results report on the *HTML Results* tab.



The screenshot shows the 'Execution' window with the 'Html Results' tab selected. The window title is 'Execution'. The content area displays a table with the following data:

Type	Name	Info	Timestamp	Duration	Status	Impact	On Error	Logs	Summ...
case	tc_sel...		2020-...	7.0	FAIL				

Figure 122
HTML Results Tab

This task is complete.

Index

C

case
 create 19
 detail 21
 execute 114
 manage 18
 requirement 26, 91
 save 35
 step 28
CLI data file
 CLI data block 80
 CLI data command 82
 create 75
 global command parameter 76
 global verification 78
 global verification combination 80
 manage 74
 save 83
configure warrior settings 13
create
 case 19
 CLI data file 75
 input data file 66
 project 102
 suite 85
 wrapper file 38

D

detail
 suite 87

E

execute
 case 114
 project 114
 suite 114

input data file
 create 66
 data 68
 manage 65
 save 73

input data file (*continued*)
 subsystem 71
 system 68

K

katana
 execution 113
 install 10
 introduction 5
 location 7
 log on 12
 overview 6
 requirement 9
 uninstall 11

L

log on 12

M

manage
 case 18
 CLI data file 74
 input data file 65
 project 101
 suite 84
 wrapper file 37

P

project
 create 102
 detail 104
 execute 114
 manage 101
 save 111
 suite 107

S

save
 case 35
 CLI data file 83
 input data file 73

save (*continued*)

 project [111](#)

 suite [99](#)

 wrapper file [63](#)

suite

 case [93](#)

 create [85](#)

 detail [87](#)

 execute [114](#)

 manage [84](#)

 requirement [26, 91](#)

 save [99](#)

W

wrapper file

 cleanup step [49](#)

 create [38](#)

 debug step [56](#)

 detail [40](#)

 manage [37](#)

 save [63](#)

 setup step [42](#)