

Cucumber Report

Sep 22, 2022, 1:53:50 AM

Start : Sep 22, 1:52:55.298 AM

End : Sep 22, 1:53:50.004 AM

Duration : 54.706 s

Features

Scenarios

Steps

PASSED - 0

FAILED - 1

SKIPPED - 0

PASSED - 0

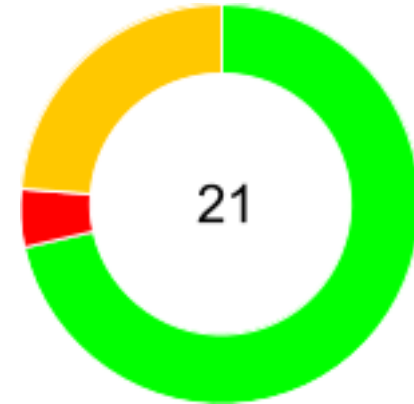
FAILED - 1

SKIPPED - 0

PASSED - 15

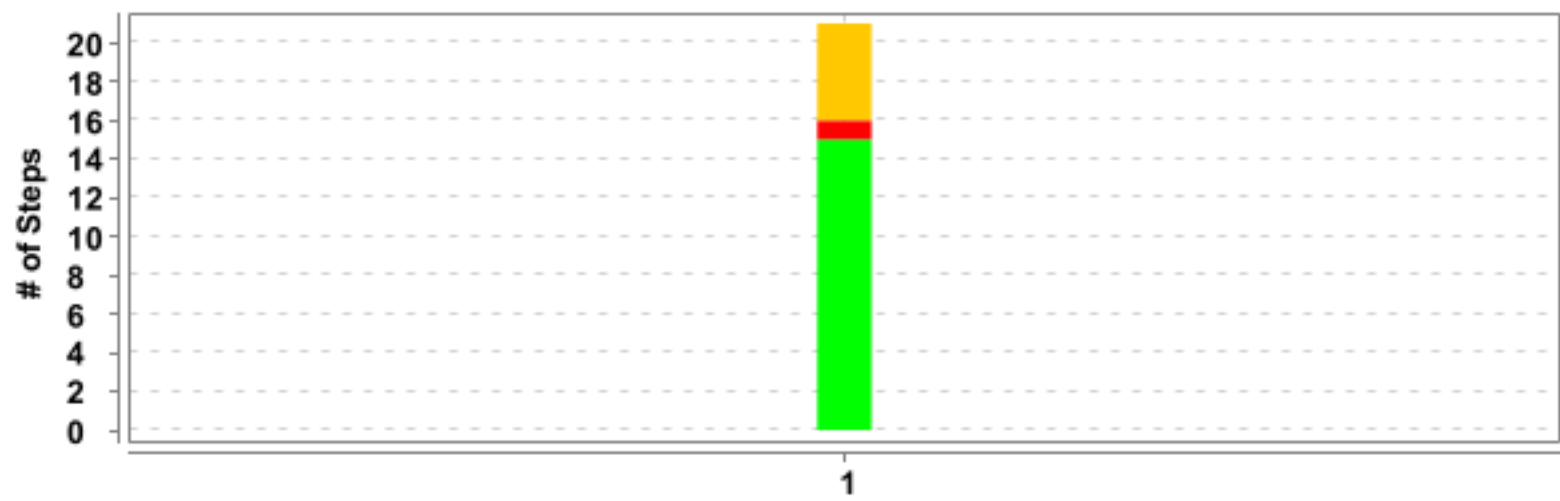
FAILED - 1

SKIPPED - 5



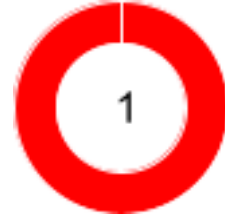
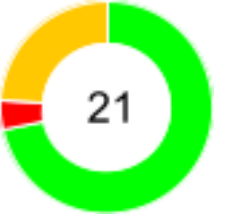


#	Feature Name	T	P	F	S	Duration
1	<u>Check the functionality of Customer Address Details</u>	1	0	1	0	54.706 s

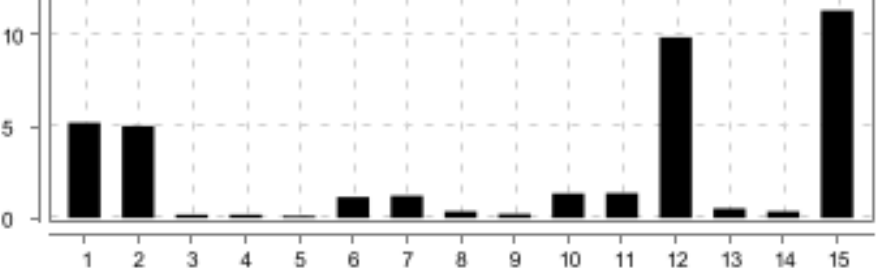
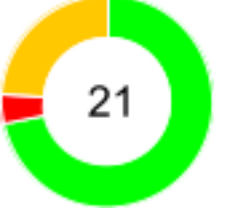


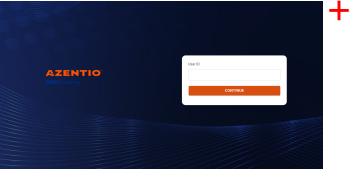
#	Feature Name	Scenario Name	T	P	F	S	Duration
1	Check the functionality of Customer Address Details	verify the functionality of save button post entering all valid details and clicking on save button	21	15	1	5	54.702 s

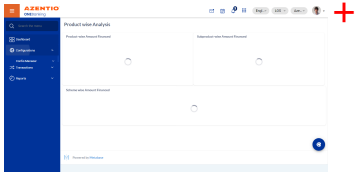
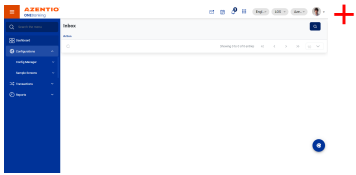
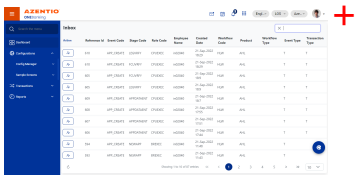
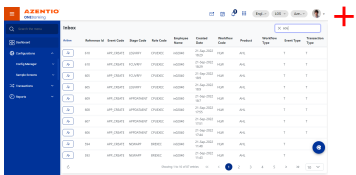
(F)- Check the functionality of Customer Address Details

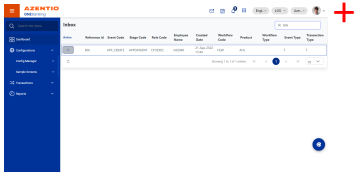
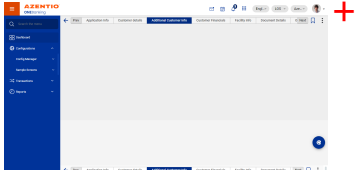
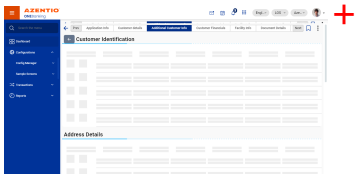
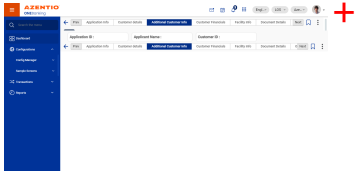
FAILED	DURATION - 54.706 s	Scenarios Total - 1 Pass - 0 Fail - 1 Skip - 0		Steps Total - 21 Pass - 15 Fail - 1 Skip - 5	
/ 1:52:55.298 AM // 1:53:50.004 AM /					

(S)- verify the functionality of save button post entering all valid details and clicking on save button

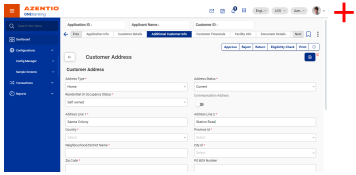
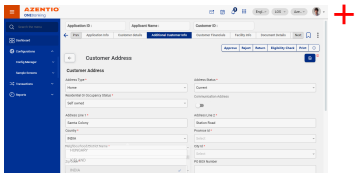
<div>FAILED</div> <div>DURATION - 54.702 s</div>	<div></div>	<div>Steps</div> <div>Total - 21</div> <div>Pass - 15</div> <div>Fail - 1</div> <div>Skip - 5</div>	<div></div>
<div>/ 1:52:55.302 AM // 1:53:50.004 AM /</div>			
<div>Check the functionality of Customer Address Details</div>			
<div>@AT_CAD_002</div>			

#	Step / Hook Details	Status	Duration
1	Given User launch the kuls application	PASSED	5.205 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.303 s
	screenshot		
			
2	And User Login as maker	PASSED	5.016 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.132 s
	screenshot		

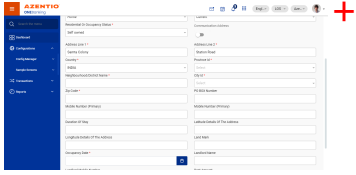
#	Step / Hook Details	Status	Duration
			
3	Then click on inbox	PASSED	0.179 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.143 s
	screenshot 		
4	Then click on inbox search icon	PASSED	0.189 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.155 s
	screenshot 		
5	Then search Customer Address Details record	PASSED	0.111 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.105 s
	screenshot 		
6	And maker user open the record from inbox	PASSED	1.138 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.116 s
	screenshot		

#	Step / Hook Details	Status	Duration
			
7	Then click on Additional customer info Segment button	PASSED	1.215 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.247 s
	screenshot 		
8	Then click on Action button of Customer Personal Information	PASSED	0.357 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.543 s
	screenshot 		
9	Then from customer identification screen click on add button of Address details	PASSED	0.220 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.434 s
	screenshot 		
10	Then select Address Type	PASSED	1.336 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.142 s
	screenshot		

#	Step / Hook Details	Status	Duration
			
11	Then select Address Status	PASSED	1.352 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.171 s
	screenshot		
			
12	Then select Residential or Occupancy Status	PASSED	9.867 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.135 s
	screenshot		
			
13	Then Enter Address Line1	PASSED	0.510 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.085 s
	screenshot		
			
14	Then Enter Address Line2	PASSED	0.354 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.084 s
	screenshot		

#	Step / Hook Details	Status	Duration
			
15	Then select Country	PASSED	11.325 s
	AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)	PASSED	0.194 s
	screenshot		
			
16	Then select Province ID	FAILED	10.561 s
	<p>org.openqa.selenium.TimeoutException: Expected condition failed: waiting for visibility of Proxy element for: DefaultElementLocator 'By.xpath: //ion-label[contains(text(),'Maharastra')]/../ion-radio' (tried for 10 second(s) with 1000 milliseconds interval)</p> <p>at org.openqa.selenium.support.ui.FluentWait.timeoutException(FluentWait.java:263)</p> <p>at org.openqa.selenium.support.ui.FluentWait.until(FluentWait.java:231)</p> <p>at helper.WaitHelper.waitForElementToVisibleWithFluentWait(WaitHelper.java:91)</p> <p>at stepdefinitions.Customer_AddressDetails.select_province_id(Customer_AddressDetails.java:153)</p> <p>at ?.select Province ID(file:///C:/Users/inindc00089/git/repository6/Priyanka_AzenzioULSFramework/src/test/java/features/Customer_AddressDetails.feature:21)</p> <p>Caused by: org.openqa.selenium.NoSuchElementException: no such element: Unable to locate element: {"method":"xpath","selector":"//ion-label[contains(text(),'Maharastra')]/../ion-radio"} (Session info: chrome=99.0.4844.82) For documentation on this error, please visit: https://selenium.dev/exceptions/#no_such_element Build info: version: '4.0.0-rc-1', revision: 'bc5511cbda' System info: host: 'INMUVPD016372', ip: '10.1.47.119', os.name: 'Windows 10', os.arch: 'amd64', os.version: '10.0', java.version: '17.0.1' Driver info: org.openqa.selenium.chrome.ChromeDriver Command: [aa9f6616396601784df986d52c964e59, findElement {using=xpath, value=//ion-label[contains(text(),'Maharastra')]/../ion-radio}] Capabilities {acceptInsecureCerts: false, browserName: chrome, browserVersion: 99.0.4844.82, chrome: {chromedriverVersion: 98.0.4758.80 (7f0488e8ba0d8..., userDataDir: C:\Users\ININDC~1\AppData\L..., goog:chromeOptions: {debuggerAddress: localhost:62156}, javascriptEnabled: true, networkConnectionEnabled: false, pageLoadStrategy: normal, platform: WINDOWS, platformName: WINDOWS, proxy: Proxy(), se:cdp: ws://localhost:62156/devtoo..., se:cdpVersion: 99.0.4844.82, setWindowRect: true, strictFileInteractability: false, timeouts: {implicit: 0, pageLoad: 300000, script: 30000}, unhandledPromptBehavior: dismiss and notify, webauthn:extension:credBlob: true, webauthn:extension:largeBlob: true, webauthn:virtualAuthenticators: true} Session ID: aa9f6616396601784df986d52c964e59</p> <p>at jdk.internal.reflect.GeneratedConstructorAccessor18.newInstance(Unknown Source)</p> <p>at java.base/jdk.internal.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45)</p> <p>at java.base/java.lang.reflect.Constructor.newInstanceWithCaller(Constructor.java:499)</p> <p>at java.base/java.lang.reflect.Constructor.newInstance(Constructor.java:480)</p> <p>at org.openqa.selenium.remote.codec.w3c.W3CHttpResponseCodec.createException(W3CHttpResponseCodec.java:200)</p> <p>at org.openqa.selenium.remote.codec.w3c.W3CHttpResponseCodec.decode(W3CHttpResponseCodec.java:133)</p>		

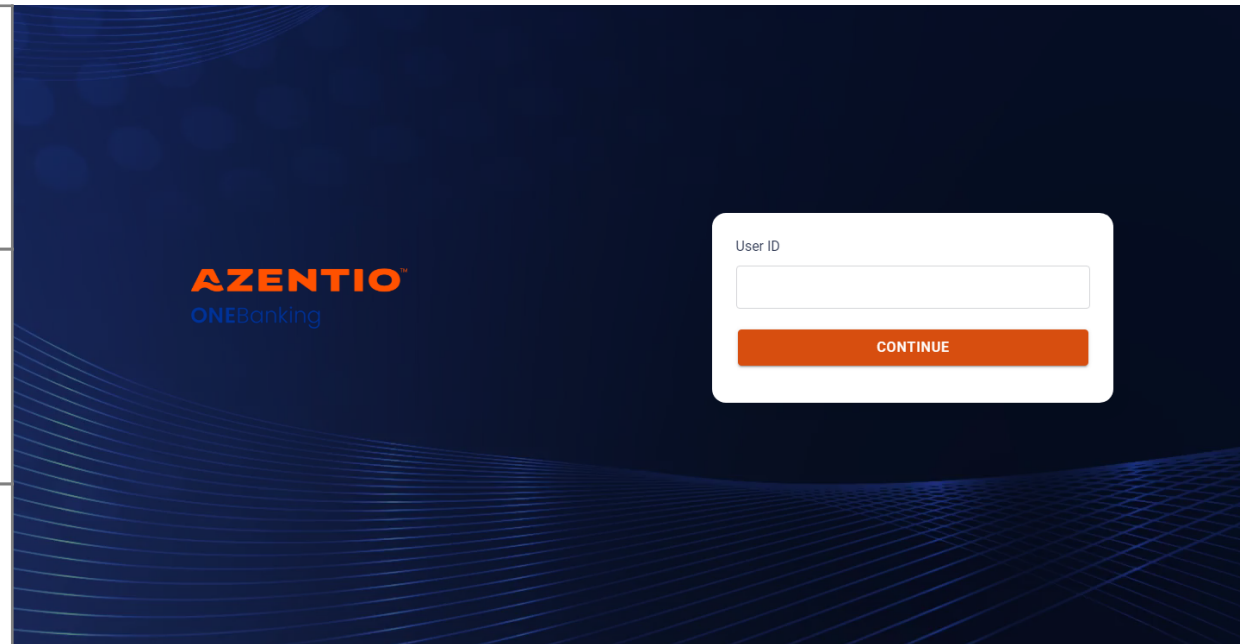
#	Step / Hook Details	Status	Duration
	at org.openqa.selenium.remote.codec.w3c.W3CHttpResponseCodec.decode(W3CHttpResponseCodec.java:53) at org.openqa.selenium.remote.HttpCommandExecutor.execute(HttpCommandExecutor.java:184) at org.openqa.selenium.remote.service.DriverCommandExecutor.invokeExecute(DriverCommandExecutor.java:164) at org.openqa.selenium.remote.service.DriverCommandExecutor.execute(DriverCommandExecutor.java:139) at org.openqa.selenium.remote.RemoteWebDriver.execute(RemoteWebDriver.java:547) at org.openqa.selenium.remote.ElementLocation\$ElementFinder\$2.findElement(ElementLocation.java:162) at org.openqa.selenium.remote.ElementLocation.findElement(ElementLocation.java:60) at org.openqa.selenium.remote.RemoteWebDriver.findElement(RemoteWebDriver.java:381) at org.openqa.selenium.remote.RemoteWebDriver.findElement(RemoteWebDriver.java:373) at org.openqa.selenium.support.pagefactory.DefaultElementLocator.findElement(DefaultElementLocator.java:70) at org.openqa.selenium.support.pagefactory.internal.LocatingElementHandler.invoke(LocatingElementHandler.java:39) at jdk.proxy2/jdk.proxy2.\$Proxy45.isDisplayed(Unknown Source) at org.openqa.selenium.support.ui.ExpectedConditions.elementIfVisible(ExpectedConditions.java:307) at org.openqa.selenium.support.ui.ExpectedConditions.access\$000(ExpectedConditions.java:40) at org.openqa.selenium.support.ui.ExpectedConditions\$10.apply(ExpectedConditions.java:293) at org.openqa.selenium.support.ui.ExpectedConditions\$10.apply(ExpectedConditions.java:290) at org.openqa.selenium.support.ui.FluentWait.until(FluentWait.java:208) at helper.WaitHelper.waitForElementToVisibleWithFluentWait(WaitHelper.java:91) at stepdefinitions.Customer_AddressDetails.select_province_id(Customer_AddressDetails.java:153) at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method) at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:77) at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43) at java.base/java.lang.reflect.Method.invoke(Method.java:568) at io.cucumber.java.Invoker.doInvoke(Invoker.java:66) at io.cucumber.java.Invoker.invoke(Invoker.java:24) at io.cucumber.java.AbstractGlueDefinition.invokeMethod(AbstractGlueDefinition.java:47) at io.cucumber.java.JavaStepDefinition.execute(JavaStepDefinition.java:29) at io.cucumber.core.runner.CoreStepDefinition.execute(CoreStepDefinition.java:66) at io.cucumber.core.runner.PickleStepDefinitionMatch.runStep(PickleStepDefinitionMatch.java:63) at io.cucumber.core.runner.ExecutionMode\$1.execute(ExecutionMode.java:10) at io.cucumber.core.runner.TestStep.executeStep(TestStep.java:92) at io.cucumber.core.runner.TestStep.run(TestStep.java:64) at io.cucumber.core.runner.PickleStepTestStep.run(PickleStepTestStep.java:51) at io.cucumber.core.runner.TestCase.run(TestCase.java:104) at io.cucumber.core.runner.Runner.runPickle(Runner.java:71) at io.cucumber.testng.TestNGCucumberRunner.lambda\$runScenario\$0(TestNGCucumberRunner.java:116) at io.cucumber.core.runtime.CucumberExecutionContext.runTestCase(CucumberExecutionContext.java:117) at io.cucumber.testng.TestNGCucumberRunner.runScenario(TestNGCucumberRunner.java:113) at io.cucumber.testng.AbstractTestNGCucumberTests.runScenario(AbstractTestNGCucumberTests.java:31) at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method) at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:77) at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)		

#	Step / Hook Details	Status	Duration
	<pre> at java.base/java.lang.reflect.Method.invoke(Method.java:568) at org.testng.internal.MethodInvocationHelper.invokeMethod(MethodInvocationHelper.java:133) at org.testng.internal.TestInvoker.invokeMethod(TestInvoker.java:598) at org.testng.internal.TestInvoker.invokeTestMethod(TestInvoker.java:173) at org.testng.internal.MethodRunner.runInSequence(MethodRunner.java:46) at org.testng.internal.TestInvoker\$MethodInvocationAgent.invoke(TestInvoker.java:824) at org.testng.internal.TestInvoker.invokeTestMethods(TestInvoker.java:146) at org.testng.internal.TestMethodWorker.invokeTestMethods(TestMethodWorker.java:146) at org.testng.internal.TestMethodWorker.run(TestMethodWorker.java:128) at java.base/java.util.ArrayList.forEach(ArrayList.java:1511) at org.testng.TestRunner.privateRun(TestRunner.java:794) at org.testng.TestRunner.run(TestRunner.java:596) at org.testng.SuiteRunner.runTest(SuiteRunner.java:377) at org.testng.SuiteRunner.runSequentially(SuiteRunner.java:371) at org.testng.SuiteRunner.privateRun(SuiteRunner.java:332) at org.testng.SuiteRunner.run(SuiteRunner.java:276) at org.testng.SuiteRunnerWorker.runSuite(SuiteRunnerWorker.java:53) at org.testng.SuiteRunnerWorker.run(SuiteRunnerWorker.java:96) at org.testng.TestNG.runSuitesSequentially(TestNG.java:1212) at org.testng.TestNG.runSuitesLocally(TestNG.java:1134) at org.testng.TestNG.runSuites(TestNG.java:1063) at org.testng.TestNG.run(TestNG.java:1031) at org.testng.remote.AbstractRemoteTestNG.run(AbstractRemoteTestNG.java:115) at org.testng.remote.RemoteTestNG.initAndRun(RemoteTestNG.java:251) at org.testng.remote.RemoteTestNG.main(RemoteTestNG.java:77) </pre> <p>* Not displayable characters are replaced by '?'. </p>		
	<p>AFTER_STEP - stepdefinitions.HooksClass.addScreenshot(io.cucumber.java.Scenario)</p> <p>screenshot</p> 	PASSED	0.098 s
17	Then Enter Neighbourhood District Name	SKIPPED	0.000 s
18	Then select City ID	SKIPPED	0.000 s
19	Then Enter Zip Code	SKIPPED	0.000 s
20	Then Enter Mobile Number primary	SKIPPED	0.000 s
21	Then Enter Occupancy Date	SKIPPED	0.000 s

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

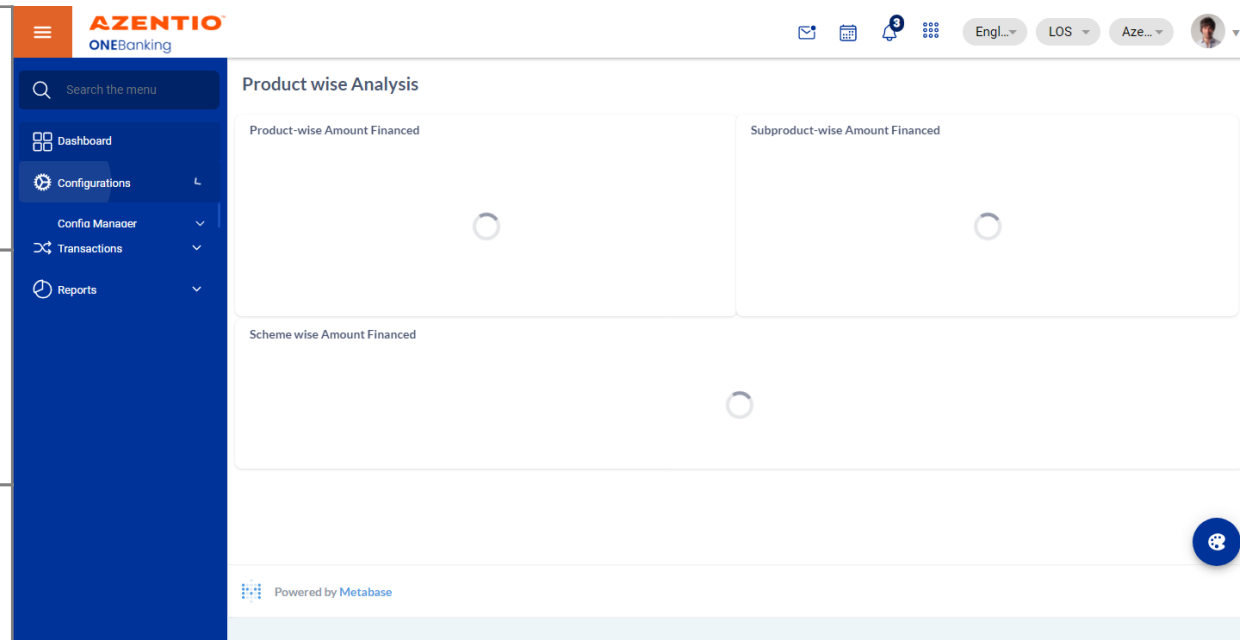
(F) Check the functionality of Customer
Address Details



(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details



(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

The screenshot shows the AZENTIO ONEBanking application interface. The top navigation bar includes the AZENTIO logo, a search menu, and user profile information (Engl..., LOS, Aze...). The left sidebar contains a menu with options: Dashboard, Configurations, Config Manager, Sample Screens, Transactions, and Reports. The main content area is titled 'Inbox' and shows 'Action' with a search bar. Below the search bar, it indicates 'Showing 0 to 0 of 0 entries' with pagination controls. A blue circular button with a gear icon is visible in the bottom right corner.

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

The screenshot shows the AZENTIO ONEBanking application interface with an inbox containing 10 entries. The top navigation bar and left sidebar are the same as in the previous screenshot. The main content area is titled 'Inbox' and shows a table of actions. The table has columns: Action, Reference Id, Event Code, Stage Code, Role Code, Employee Name, Created Date, Workflow Code, Product, Workflow Type, Event Type, and Transaction Type. The entries are as follows:

Action	Reference Id	Event Code	Stage Code	Role Code	Employee Name	Created Date	Workflow Code	Product	Workflow Type	Event Type	Transaction Type
	610	APP_CREATE	LEGVRFY	CPUEXEC	in02040	21-Sep-2022 18:29	HLW	AHL		T	T
	610	APP_CREATE	FCUVRFY	CPUEXEC	in02040	21-Sep-2022 18:29	HLW	AHL		T	T
	605	APP_CREATE	FCUVRFY	CPUEXEC	in02040	21-Sep-2022 18:9	HLW	AHL		T	T
	605	APP_CREATE	LEGVRFY	CPUEXEC	in02040	21-Sep-2022 18:9	HLW	AHL		T	T
	609	APP_CREATE	APPDATAENT	CPUEXEC	in02040	21-Sep-2022 18:7	HLW	AHL		T	T
	608	APP_CREATE	APPDATAENT	CPUEXEC	in02040	21-Sep-2022 17:55	HLW	AHL		T	T
	607	APP_CREATE	APPDATAENT	CPUEXEC	in02040	21-Sep-2022 17:51	HLW	AHL		T	T
	606	APP_CREATE	APPDATAENT	CPUEXEC	in02040	21-Sep-2022 17:44	HLW	AHL		T	T
	594	APP_CREATE	NEWAPP	BREXEC	in02040	21-Sep-2022 11:48	HLW	AHL		T	
	593	APP_CREATE	NEWAPP	BREXEC	in02040	21-Sep-2022 11:43	HLW	AHL		T	

At the bottom, it indicates 'Showing 1 to 10 of 87 entries' with pagination controls. A blue circular button with a gear icon is visible in the bottom right corner.

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

AZENTIO ONEBanking

Search the menu

Dashboard

Configurations

Config Manager

Sample Screens

Transactions

Reports

Inbox

606

Action	Reference Id	Event Code	Stage Code	Role Code	Employee Name	Created Date	Workflow Code	Product	Workflow Type	Event Type	Transaction Type
	610	APP_CREATE	LEGVRFY	CPUEXEC	in02040	21-Sep-2022 18:29	HLW	AHL		T	T
	610	APP_CREATE	FCUVRFY	CPUEXEC	in02040	21-Sep-2022 18:29	HLW	AHL		T	T
	605	APP_CREATE	FCUVRFY	CPUEXEC	in02040	21-Sep-2022 18:9	HLW	AHL		T	T
	605	APP_CREATE	LEGVRFY	CPUEXEC	in02040	21-Sep-2022 18:9	HLW	AHL		T	T
	609	APP_CREATE	APPDATAENT	CPUEXEC	in02040	21-Sep-2022 18:7	HLW	AHL		T	T
	608	APP_CREATE	APPDATAENT	CPUEXEC	in02040	21-Sep-2022 17:55	HLW	AHL		T	T
	607	APP_CREATE	APPDATAENT	CPUEXEC	in02040	21-Sep-2022 17:51	HLW	AHL		T	T
	606	APP_CREATE	APPDATAENT	CPUEXEC	in02040	21-Sep-2022 17:44	HLW	AHL		T	T
	594	APP_CREATE	NEWAPP	BREXEC	in02040	21-Sep-2022 11:48	HLW	AHL		T	
	593	APP_CREATE	NEWAPP	BREXEC	in02040	21-Sep-2022 11:43	HLW	AHL		T	

Showing 1 to 10 of 87 entries

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

AZENTIO ONEBanking

Search the menu

Dashboard

Configurations

Config Manager

Sample Screens

Transactions

Reports

Inbox

606

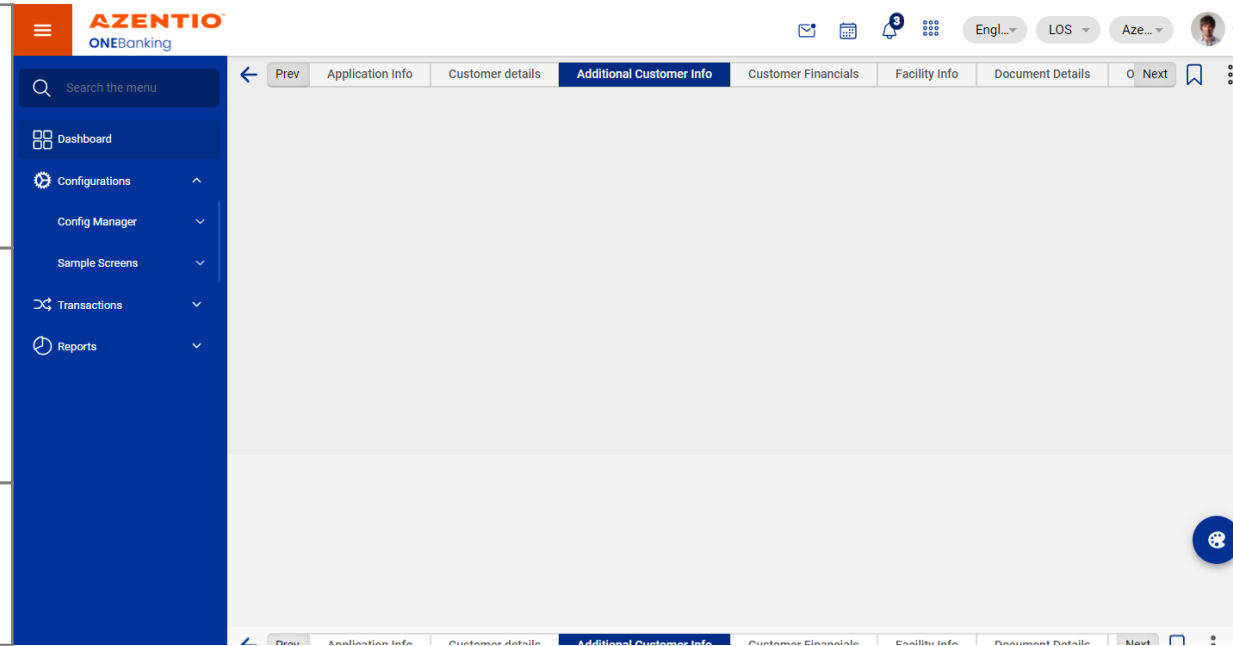
Action	Reference Id	Event Code	Stage Code	Role Code	Employee Name	Created Date	Workflow Code	Product	Workflow Type	Event Type	Transaction Type
	606	APP_CREATE	APPDATAENT	CPUEXEC	in02040	21-Sep-2022 17:44	HLW	AHL		T	T

Showing 1 to 1 of 1 entries

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

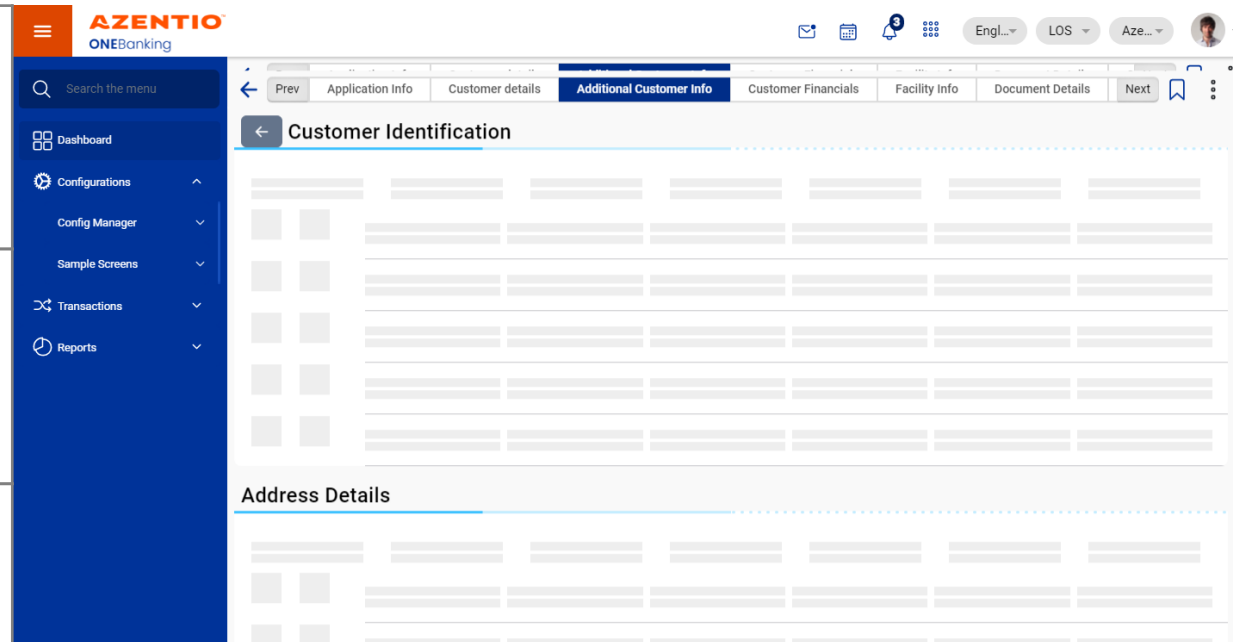
(F) Check the functionality of Customer
Address Details



(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details



(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

The screenshot displays the AZENTIO ONEBanking web application. On the left is a dark blue sidebar with a search bar and a menu containing: Dashboard, Configurations (with a sub-menu for Config Manager), Sample Screens, Transactions, and Reports. The main content area has a top navigation bar with the AZENTIO logo, user profile, and language/region selectors. Below this is a breadcrumb trail: Prev > Application Info > Customer details > **Additional Customer Info** > Customer Financials > Facility Info > Document Details > Next. The 'Additional Customer Info' section contains three input fields: 'Application ID:', 'Applicant Name:', and 'Customer ID:'. The 'Next' button in the breadcrumb trail is disabled.

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

This screenshot shows the 'Customer Address' form within the AZENTIO ONEBanking application. The breadcrumb trail is: Prev > Application Info > Customer details > **Additional Customer Info** > Customer Financials > Facility Info > Document Details > Next. The 'Additional Customer Info' tab is active, and a sub-header 'Customer Address' is displayed. Above the form are buttons: Approve, Reject, Return, Eligibility Check, and Print. The form fields include: 'Address Type' (dropdown with 'Home' selected), 'Residential Or Occupancy Status' (dropdown with 'Select' selected), 'Address Line 1' (text input with 'Head Quarter Address'), 'Country' (dropdown with 'Home' selected and a checkmark), 'Neighbourhood/District Name' (text input with 'Place of actual Residence'), 'Zip Code', 'Address Status' (dropdown with 'Select' selected), 'Communication Address' (toggle switch), 'Address Line 2' (text input), 'Province Id' (dropdown with 'Select' selected), 'City Id' (dropdown with 'Select' selected), and 'PO BOX Number' (text input).

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

AZENTIO ONEBanking

Application ID : Applicant Name : Customer ID :

Prev Application Info Customer details **Additional Customer Info** Customer Financials Facility Info Document Details Next

Approve Reject Return Eligibility Check Print

Customer Address

Customer Address

Address Type *
Home

Residential Or Occupancy Status *
Select

Address Line 1 *

Country *
Select

Neighbourhood/District Name *

Zip Code *

Address Status *
Current

Communication Address
Select
Current

Address Line 2 *
Previous

Province Id *
Select

City Id *
Select

PO BOX Number

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

AZENTIO ONEBanking

Application ID : Applicant Name : Customer ID :

Prev Application Info Customer details **Additional Customer Info** Customer Financials Facility Info Document Details Next

Approve Reject Return Eligibility Check Print

Customer Address

Customer Address

Address Type *
Home

Residential Or Occupancy Status *
Self owned

Address Line 1 *
Property /Owns

Address Line 2 *
Rental

Country *
Select

Province Id *
Select

City Id *
Select

PO BOX Number

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

The screenshot shows the AZENTIO ONEBanking interface. The top navigation bar includes the logo, a search menu, and user profile information. The left sidebar contains a menu with options: Dashboard, Configurations, Config Manager, Sample Screens, Transactions, and Reports. The main content area displays the 'Customer Address' form. The form has a breadcrumb trail: Prev > Application Info > Customer details > Additional Customer Info > Customer Financials > Facility Info > Document Details > Next. The form fields are organized into two columns. The left column includes: Application ID, Applicant Name, Customer ID, Address Type (Home), Residential Or Occupancy Status (Self owned), Address Line 1 (Samta Colony), Country (Select), Neighbourhood/District Name, and Zip Code. The right column includes: Address Status (Current), Communication Address (toggle), Address Line 2, Province Id (Select), City Id (Select), and PO BOX Number. At the top right of the form, there are buttons for Approve, Reject, Return, Eligibility Check, and Print, along with a help icon.

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

This screenshot is identical to the one above, showing the AZENTIO ONEBanking 'Customer Address' form. The only difference is in the 'Address Line 2' field, which now contains the text 'Station Road'.

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

The screenshot shows the AZENTIO ONEBanking application interface. The top navigation bar includes the AZENTIO logo, a search menu, and user profile information. The left sidebar contains a menu with options: Dashboard, Configurations, Config Manager, Sample Screens, Transactions, and Reports. The main content area displays the 'Customer Address' form. The form is divided into two columns. The left column contains fields for Address Type (Home), Residential Or Occupancy Status (Self owned), Address Line 1 (Samta Colony), Country (INDIA), Neighbourhood/District Name (HUNGARY), and Zip Code (ICELAND). The right column contains fields for Address Status (Current), Communication Address (toggle), Address Line 2 (Station Road), Province Id (Select), City Id (Select), and PO BOX Number. At the top of the form, there are input fields for Application ID, Applicant Name, and Customer ID. Below these fields are tabs for Application Info, Customer details, Additional Customer Info (selected), Customer Financials, Facility Info, and Document Details. At the bottom of the form, there are buttons for Approve, Reject, Return, Eligibility Check, and Print.

(Step) AFTER_STEP - stepdefinitions.
HooksClass.addScreenshot(io.cucumber.
java.Scenario)

(S) verify the functionality of save button
post entering all valid details and clicking
on save button

(F) Check the functionality of Customer
Address Details

This screenshot shows the same AZENTIO ONEBanking application interface as the previous one, but with additional fields visible. The left column now includes Mobile Number (Primary), Duration Of Stay, Longitude Details Of The Address, and Occupancy Date. The right column now includes Mobile Number (Primary), Latitude Details Of The Address, Land Mark, and Landlord Name. The form is still divided into two columns, and the top navigation bar and left sidebar remain the same. The 'Additional Customer Info' tab is still selected, and the 'Approve' button is visible at the bottom right.