

# Interworks.cloud Test Challenge

Automated testing for the [www.saucedemo.com](http://www.saucedemo.com) website using Selenium and TestNG in Java

International Hellenic University  
Department of Information and Electronic Engineering

Karamanian Anastasios 2019064, [anastasiskaramanian@gmail.com](mailto:anastasiskaramanian@gmail.com)

# Table of Contents

Table of Contents.....	1
Selenium Testing.....	2
Selenium WebDriver.....	2
TestNG.....	3
Testing Environment.....	3
Test Cases Documentation.....	3
Interworks.cloud Test Challenge cases.....	4
CheckoutStepTwo Page Tests & CheckoutComplete Page Tests.....	4
Login Page Tests.....	5
Item Page Tests.....	8
Inventory Page Tests.....	9

# Selenium Testing

## Selenium WebDriver

The selenium web driver allows us to drive a browser the same way as a normal user would in order to test web applications. This is done by selecting and interacting with html document elements to simulate user clicks and inputs along with inspecting elements' attributes with the purpose of validating the proper functionality of our web application. Furthermore, this tool gives us the ability to execute cross-browser tests and it easily integrates with most programming languages.

## TestNG

TestNG is a testing framework inspired by JUnit with the goal of expanding on functionalities and ease of use. It includes plenty of new features to improve the testing experience such as:

- Annotations: With annotations we can specify many class or method level configurations.
- Support for data-driven testing
- Run your tests in arbitrarily big thread pools with various policies available (all methods in their own thread, one thread per test class, etc...).

And many more.

## Testing Environment

- Java (openJDK-22.0.1)
- Selenium server 4.18.1 for java
- TestNG 7.1.0
- Tests are done on the Firefox browser using the Mozilla geckodriver.
- Web application tested: [www.saucedemo.com](http://www.saucedemo.com)
- All project code can be found at <https://github.com/ashot-k/Selenium-101>

# Test Cases Documentation

## Interworks.cloud Test Challenge cases

Test	Checkout Step One Error Validation	Checkout Step Two Validation
Class	CheckoutStepOnePageTests.java	CheckoutStepTwoPageTests.java
Method	<b>TestWrongPostalCodeInputErrorValidation()</b>	<b>TestValidateProductAndTotalPrices()</b>
Desc	Ensure proper message is shown if Zip field is left empty and user attempts to checkout.	Ensure prices shown for products added and total order price is correct during checkout.
Step 1	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>
Step 2	Log in using a valid username and password.	Log in using a valid username and password.
Step 3	Add the below products to the cart: a. Sauce Labs Bike Light	Add the below products to the cart: a. Sauce Labs Backpack b. Sauce Labs Fleece Jacket
Step 4	Go to the cart	Go to the cart.
Step 5	Click Checkout	Click Checkout
Step 6	Enter a valid a. Name b. Last Name	Enter a valid: a. Name b. Last Name c. Postal Code
Step 7	Leave zip code empty	Click Checkout
Step 8	Click checkout	On Checkout step two page Verify: a. The prices of both products b. The value of the total price
Step 9	On Checkout step one page Verify: a. The Error message	

## CheckoutStepTwo Page Tests & CheckoutComplete Page Tests

Test	Verify Total items	Validate checkout completion message
Method	<code>TestVerifyTotalItems()</code>	<code>TestCompleteMessage()</code>
Desc	Add items to the cart and verify their existence in the checkout phase	Verify that the proper message is shown upon successful checkout
Step 1	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>
Step 2	Log in using a valid username and password.	Log in using a valid username and password.
Step 3	Add the following to the cart: <ul style="list-style-type: none"> <li>a. Backpack</li> <li>b. Fleece jacket</li> </ul>	Add the following to the cart: <ul style="list-style-type: none"> <li>a. Fleece jacket</li> <li>b. Bike light</li> </ul>
Step 4	Go to cart	Go to cart
Step 5	Click checkout	Click checkout
Step 6	Insert valid: <ul style="list-style-type: none"> <li>a. Name</li> <li>b. Last name</li> <li>c. Zip code</li> </ul>	Insert valid: <ul style="list-style-type: none"> <li>a. Name</li> <li>b. Last name</li> <li>c. Zip code</li> </ul>
Step 7	Click checkout	Click checkout
Step 8	Verify that the number of items is 2	Click finish
Step 9		Verify the message shown

## Login Page Tests

Test	Successful login	Failed Login	Logout
Method	<b>TestSuccessfulLogin()</b>	<b>TestFailedLogin()</b>	<b>TestLogout()</b>
Desc	Validates that the user is redirected to the correct URL after successful login	Validate that the error message is shown once user inputs wrong credentials and attempts to login	Verify user is redirected to login page after logging out
Step 1	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>
Step 2	Log in using a valid username and password.	Log in using a valid username and invalid password.	Log in using a valid username and password.
Step 3	Verify current page is the Inventory page	Check if: <ul style="list-style-type: none"> <li>a. error message is displayed</li> <li>b. error message shown is correct</li> </ul>	Click on the sidebar and click "Logout"
Step 4			Verify current URL is the login page

Test	Logout Validation
Method	<code>TestLogoutValidation()</code>
Desc	Validate that the user is properly logged out (cookies and sessions cleared) by attempting to go back after logging out
Step 1	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>
Step 2	Log in using a valid username and password.
Step 3	Click on the sidebar and click "Logout"
Step 4	Go back
Step 5	Check that current URL is the login page
Step 6	Verify that: <ul style="list-style-type: none"> <li>a. error message is displayed</li> <li>b. check if error message is correct</li> </ul>

## Item Page Tests

Test	Inventory page item image matches item page image	Item addition to cart	Item removal from cart
Method	<b>TestInventoryPageItemImageMatchesItemPageImg()</b>	<b>TestItemAdditionToCart()</b>	<b>TestItemRemovalFromCart()</b>
Desc	Validate the image shown in the inventory page is the same as the one in that item's page.	Check the functionality of the "add to cart" button in an item's page by clicking it and then validating the existence of the item in the cart.	Check the functionality of the "remove" button in an item's page by clicking it and then validating that the cart is empty.
Step 1	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>
Step 2	Log in using a valid username and password.	Log in using a valid username and password.	Log in using a valid username and password.
Step 3	Capture Backpack's image	Go to t-shirt page	Go to t-shirt page
Step 4	Go to backpack's page	Click "Add to cart"	Click "Add to cart"
Step 5	Check if image in item page matches the captured Backpack image	Go to cart page	Click "remove"
Step 6		Validate items existence in the cart	Go to cart page
Step 7			Validate that the cart is empty



## Inventory Page Tests

Test	Ascending Price sorting Validation	Descending Name sorting Validation	Cart Badge Number change
Method	<b>TestAscendingPriceSortingValidation()</b>	<b>TestDescendingNameSortingValidation()</b>	<b>TestCartBadgeNumberValidation()</b>
Desc	Changes the sorting of items to “ascending price” and checks if the items are sorted accordingly.	Changes the sorting of items to “descending name” and checks if the items are sorted accordingly.	Checks whether the cart badge number changes on item addition.
Step 1	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>	Navigate to <a href="https://www.saucedemo.com/">https://www.saucedemo.com/</a>
Step 2	Log in using a valid username and password.	Log in using a valid username and password.	Log in using a valid username and password.
Step 3	Change the sorting option to “ascending price”	Change the sorting option to “descending name”	Add these items to the cart: <ul style="list-style-type: none"> <li>a. Backpack</li> <li>b. Fleece jacket</li> <li>c. Bike light</li> </ul>
Step 4	Validate that items are sorted correctly	Validate that items are sorted correctly	Validate that the cart badge number is 3