|  |  |
| --- | --- |
| Project Name: | Build.com – Automation Testing |

**Change History:**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Details | Author | Date |
| v1.0 | Initial Version | Roopa Patil | 01/22/2017 |

**Summary:**

Scope of this program is to automate search given set of products, add them to cart, check out and assert the CA Tax and Grand Total Amounts.

**Manual Test Path:**

1. Open Webpage ([www.build.com](http://www.build.com/))
2. Enter the <product name> in search fields (What are you shopping for?) [**Kohler K​6626​6U]**
3. Click on <finish/color> image for **[Suede Kohler]**
4. Click on <Add to Cart> button
5. Repeat step 2-4 for <product name>= [**Kohler K​6626​6U]** and finish/color =[**Cashmere Kohler**]
6. Search <product name> = [**Kohler K-5180-ST**]
7. Change the quantity to 2
8. Click on <Add to Cart> button
9. Click on <Secure Checkout> button
10. Click on <Checkout as Guest>
11. Enter the shipping address information
    1. First Name
    2. Last Name
    3. Street Address
    4. Zip Code
    5. City
    6. State
    7. Phone Number
12. Enter Contact Information
    1. Email Address
13. Enter Credit Card Information
    1. Card Number
    2. Expires
       1. Select Month
       2. Select Year
    3. Name on Card
    4. Security Code
14. Click on <Review Order> button

**Implementation:**

1. Frameworks/pom.xml Dependencies
   * + TestNG 6.10
     + Selenium Server 2.53.0
     + Selenium Firefox Driver 2.53.0
     + OpenCSV 3.7
2. Package : *org.roopa.cart*
3. Class: *ShoppingCart*
4. Methods:
   * public void loadProperties()
     + Loads the properties from cart.properties file.
   * public Object[][] getOrderData()
     + Provides the order data (List of products, unique identifier from the webpage and quantity in the form of 2D Array.
   * public Object[][] getOrderDatacsv()
     + Data provider: Reads the order data (List of products, unique web element identifier and quantity) from csv file.
   * public void getUrl()
     + Creates the WebDriver object based on the browser type.
     + Opens the Web Page ([www.build.com](http://www.build.com/))
   * public void searchAndAdd(String ProductId,String subProdId,String Qty)
     + Test Input : getOrderDatacsv
     + Searches the given set of products using web element identifier
     + Adds the required number of products into cart
   * public void checkOut()
     + Automate the checkout process
     + Adds the shipping information
     + Adds the credit card information
   * public float getSubTotal()
     + Reads the Sub Total amount from the webpage.
   * public float getTax()
     + Reads the Tax Amount from the webpage.
   * public float getGrandTotal()
     + Reads the Grand Total Amount from webpage.
   * public float getShipping()
     + Reads the Shipping Cost from webpage.
   * public void assertionTax()
     + Assert the CA Tax amount (TaxAmount = 0.0725 \* SubTotal)
   * public void assertionGrandTotal()
     + Assert the Grand Total (GrandTotal = SubTotal + TaxAmount +ShippingCost)
   * public void completeTest()
     + Final Step, Closes the driver.
   * public WebElement getElementById(String s)
     + Searches the webpage for element id and Returns the web element.
   * public WebElement getElementByCss(String s)
     + Searches the webpage for unique css selector and Returns the web element.

**Test Cases:**

1. TestCase1 : Search and Add (*Kohler K​6626​6U, Suede Kohler,1*)
2. TestCase2 : Search and Add (*Kohler K​6626​6U, Cashmere Kohler,1*)
3. TestCase3 : Search and Add (*Kohler K-5180-ST,2*)
4. TestCase4 : Checkout()
5. TestCase5 : Assert CA Tax
6. TestCase6 : Assert Grand Total

**Environment:**

* Operating System : Windows 7
* Eclipse 4.5.1
* Apache Maven 3.3.9

**GitHub:**

<https://github.com/roopaee22/Roopa>