Automate an E-Commerce Web Application

SOURCE CODE

FlipKartTesting.java

```
package com.flipkart.test;

import org.testng.annotations.Test;
import org.testng.annotations.BeforeMethod;
import org.testng.AssertJUnit;
import org.openqa.selenium.JavascriptExecutor;
import org.testng.annotations.BeforeTest;
import java.awt.Toolkit;
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.By;
import org.openqa.selenium.Dimension;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.testng.Reporter;
import org.testng.annotations.AfterMethod;
import org.testng.annotations.AfterTest;


public class FlipKartTesting {

	private WebDriver driver;
	private WebDriver firefoxDriver;
	private String baseUrl = "https://www.flipkart.com/";

	@BeforeMethod
	@BeforeTest
	public void setUp() {
		driver = new ChromeDriver();
		driver.manage().window().maximize();
	}

	@Test(priority = 1)

	public void NavigateFlipkartHomePage() throws InterruptedException {

		driver.get(baseUrl);

	driver.findElement(By.xpath("/html/body/div[2]/div/div/button")).clic
k();
		String url = driver.getCurrentUrl();
		AssertJUnit.assertEquals("https://www.flipkart.com/", url);
		String title = driver.getTitle();
		AssertJUnit.assertEquals(
				"Online Shopping Site for Mobiles, Electronics,
Furniture, Grocery, Lifestyle, Books & More. Best Offers!",
				title);
		System.out.println("1.NAVIGATION TO FLIPKART SUCCESSFULL!");
		System.out.println("2.FLIPKART TITLE MATCH SUCCESSFULL!");
```

```
            Reporter.log("Flipkart Navigation - > SUCCESS!!");
            Reporter.log("Flipkart Title Match - > SUCCESS!!");

      }

      @Test(priority = 2)

      public void FlipkartPageLoad() throws InterruptedException {

            long start = System.currentTimeMillis();

            long finish = System.currentTimeMillis();
            long Loadtime = finish - start;
            System.out.println("Page Load Time = " + Loadtime +
"milliseconds");
            long expectedtime = 30000;
            if (Loadtime < expectedtime) {
                  System.out.println("3.PAGE LOAD SUCCESSFULLY");
                  Reporter.log("Page Loaded within Expected Time ->
SUCCESS!!");
            } else {
                  System.out.println("3.PAGE LOAD EXCEEDED THE TIME
LIMIT");
                  Reporter.log("Page Loaded within Expected Time -> NOT
SUCCESS!!");
            }

      }

      @Test(priority = 3)
      public void SearchProduct() throws InterruptedException {

            driver.get("https://www.flipkart.com/");

      driver.findElement(By.xpath("/html/body/div[2]/div/div/button")).clic
k();
            driver.findElement(By.name("q")).click();
            Thread.sleep(2000);
            driver.findElement(By.name("q")).sendKeys("IPhone 13",
Keys.ENTER);
            Thread.sleep(2000);
            System.out.println("4.IPHONE 13 SEARCHED SUCCESSFULLY..!!");
            Reporter.log("Product Search -> SUCCESS!!");

      }

      @Test(priority = 4)
      public void ImageLoad() throws InterruptedException {

            driver.get(baseUrl +
"search?q=realme&otracker=search&otracker1=search&marketplace=FLIPKART&as-
show=on&as=off");
            WebElement we1 = driver.findElement(By.xpath(

      "//*[@id=\"container\"]/div/div[3]/div[1]/div[2]/div[4]/div/div/div/a
/div[1]/div[1]/div/div/img"));
            if (we1.isDisplayed()) {
                  System.out.println("5.IMAGE NOT PRESENT - BEFORE
SCROLLING");
                  Reporter.log("Image Before Scrolling -> SUCCESS!!");
            }
```

```
            JavascriptExecutor js = (JavascriptExecutor) driver;
            js.executeScript("window.scrollBy(0,500)");
            if (we1.isDisplayed()) {
                  System.out.println("IMAGE PRESENT - AFTER SCROLLING");
                  Reporter.log("Image After Scrolling -> SUCCESS!!");
            }

      }

      @Test(priority = 5)
      public void ImageScreenHeight() throws InterruptedException {

            driver.get(baseUrl);

      driver.findElement(By.xpath("/html/body/div[2]/div/div/button")).clic
k();
            java.awt.Dimension screen_size =
Toolkit.getDefaultToolkit().getScreenSize();
            int height = (int) screen_size.getHeight();
            int width = (int) screen_size.getWidth();
            System.out.println("6.SCREEN RESOLUTION SUCCESS :\n");
            System.out.println("Height : " + height + "\nWidth : " +
width);
            int img_height = driver.findElement(By.xpath(

      "//*[@id=\"container\"]/div/div[3]/div[3]/div[1]/div/div[2]/div/div/d
iv[1]/div/div[1]/div/a/div[1]/div/img"))
                        .getSize().getHeight();
            int img_width = driver.findElement(By.xpath(

      "//*[@id=\"container\"]/div/div[3]/div[3]/div[1]/div/div[2]/div/div/d
iv[1]/div/div[1]/div/a/div[1]/div/img"))
                        .getSize().getWidth();
            System.out.println("\n7.IMAGE RESOLUTION SUCCESS :\n");
            System.out.println("Height " + img_height + "\nWidth " +
img_width);
            WebElement img = driver.findElement(By.xpath(

      "//*[@id=\"container\"]/div/div[3]/div[3]/div[1]/div/div[2]/div/div/d
iv[1]/div/div[1]/div/a/div[1]/div/img"));
            if (img.isDisplayed()) {
                  if (img_height < height && img_width < width) {
                        System.out.println("The images are loaded and
visible till the screen height only");
                        Reporter.log("Images are loaded and visible till
the screen height only  -> SUCCESS!!");
                  } else {
                        System.out.println("The images are not loaded and
visible till the screen height");
                        Reporter.log("Images are not loaded and visible
till the screen height -> SUCCESS!!");
                  }
            }

      }

      @Test(priority = 6)
      public void PageScroll() throws InterruptedException {

            driver.get(baseUrl);
```

```
	driver.findElement(By.xpath("/html/body/div[2]/div/div/button")).clic
k();
			Thread.sleep(2000);
			String execScript = "return
document.documentElement.scrollHeight>document.documentElement.clientHeight
;";
			JavascriptExecutor scrollBarPresent = (JavascriptExecutor)
driver;
			Boolean test = (Boolean)
(scrollBarPresent.executeScript(execScript));
			if (test == true) {
				System.out.print("8.PAGE SCROLL SUCCESSFULL");
				Reporter.log("Page Scroll -> SUCCESS!!");
			} else if (test == false) {
				System.out.print("8.PAGE SCROLL NOT SUCCESSFULL");
				Reporter.log("Page Scroll -> NOT SUCCESS!!");
			}

	}

	@Test(priority = 7)
	public void RefreshContent() throws InterruptedException {

			driver.get(baseUrl
					+ "search?q=iphone+13&sid=tyy%2C4io&as=on&as-
show=on&otracker=AS_QueryStore_OrganicAutoSuggest_1_3_na_na_na&otracker1=AS
_QueryStore_OrganicAutoSuggest_1_3_na_na_na&as-pos=1&as-
type=RECENT&suggestionId=iphone+13%7CMobiles&requestId=ee111f6f-1edd-495d-
9c8a-eec51bb9a9d5&as-searchtext=iph");
			long startTime = System.currentTimeMillis();
			driver.manage().timeouts().implicitlyWait(10,
TimeUnit.SECONDS);

			JavascriptExecutor js = (JavascriptExecutor) driver;

			// specify the WebElement till which the page has to be
scrolled
			WebElement element =
driver.findElement(By.xpath("//div[@class='CXW8mj']"));

			js.executeScript("arguments[0].scrollIntoView();", element);

			long endTime = System.currentTimeMillis();
			long totalTime = endTime - startTime;
			System.out.println("9.REFRESH CONTENT LOAD TIME " + totalTime +
" milliseconds");
			Reporter.log("Refresh Content -> SUCCESS!! \n Total Time = " +
totalTime + " milliseconds");

	}

	@Test(priority = 8)
	public void BottomNavigation() throws InterruptedException {
			driver.get(baseUrl);
			Thread.sleep(2000);
			JavascriptExecutor js = (JavascriptExecutor) driver;

	js.executeScript("window.scrollBy(0,document.body.scrollHeight)");
			System.out.println("10.BOTTOM NAVIGATION SUCCESSFULL!!");
			Reporter.log("Bottom Navigation - > SUCCESS!!");
```

```
	}

	@Test(priority = 9)
	public void DifferentResolution() throws InterruptedException {
		driver.get(baseUrl);

	driver.findElement(By.xpath("/html/body/div[2]/div/div/button")).clic
k();
		Thread.sleep(2000);
		JavascriptExecutor js = (JavascriptExecutor) driver;
		js.executeScript("window.scrollBy(0,1000)", "");
		driver.manage().window().setSize(new Dimension(1620, 820));
		Thread.sleep(3000);
		WebElement textField = driver.findElement(By.name("q"));
		textField.sendKeys("Iphone 13");

	driver.findElement(By.xpath("//button[@class='L0Z3Pu']")).click();
		js.executeScript("window.scrollBy(0,1000)", "");
		driver.navigate().back();
		Thread.sleep(3000);
		driver.manage().window().maximize();
		Thread.sleep(3000);
		js.executeScript("window.scrollBy(0,1000)", "");
		driver.manage().window().setSize(new Dimension(240, 360));
		textField.sendKeys("mini");
		Thread.sleep(3000);
		js.executeScript("window.scrollBy(0,1000)", "");
		System.out.println("11.DIFFERENT RESOLUTION SUCCESSFULL");
		Reporter.log("Different Resolution -> SUCCESS!!");

	}

	@Test(priority = 10)
	public void DifferentBrowser() throws InterruptedException {
		firefoxDriver = new FirefoxDriver();

		firefoxDriver.manage().window().maximize();
		firefoxDriver.get(baseUrl);
		System.out.println("12.DIFFERENT BROWSER SUCCESSFULL");
		Reporter.log("Different Browser -> SUCCESS!!");

		Thread.sleep(2000);
		firefoxDriver.close();
	}

	@AfterTest
	@AfterMethod
	public void quitbrowser() throws InterruptedException {
		Thread.sleep(15000);
		driver.quit();

	}

}
```

pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>END-PHASE-5</groupId>
  <artifactId>END-PHASE-5</artifactId>
  <version>0.0.1-SNAPSHOT</version>


    <dependencies>
        <!--
https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java --
>
        <!--for selenium-->
<dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>4.10.0</version>
</dependency>


<!-- https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-
chrome-driver
for chrome
 -->
<dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-chrome-driver</artifactId>
    <version>4.10.0</version>
</dependency>
    <dependency>
        <groupId>io.github.bonigarcia</groupId>
        <artifactId>webdrivermanager</artifactId>
        <version>5.0.3</version>
    </dependency>

            <dependency>
                  <groupId>org.seleniumhq.selenium</groupId>
                  <artifactId>selenium-firefox-driver</artifactId>
                  <version>4.10.0</version>
            </dependency>


 <!-- TestNG -->
    <dependency>
        <groupId>org.testng</groupId>
        <artifactId>testng</artifactId>
        <version>7.4.0</version>
    </dependency>
  </dependencies>

  <build>
            <plugins>
                  <plugin>
```

```
                        <groupId>org.apache.maven.plugins</groupId>
                        <artifactId>maven-compiler-plugin</artifactId>
                        <configuration>
                              <source>1.8</source>
                              <target>1.8</target>
                        </configuration>
                  </plugin>
                  <plugin>
                        <groupId>org.apache.maven.plugins</groupId>
                        <artifactId>maven-surefire-plugin</artifactId>

                  </plugin>
            </plugins>
      </build>
</project>
```

testing.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">
<suite name="Suite">
  <test thread-count="5" name="Test">
    <classes>

      <class name="com.flipkart.test.FlipKartTesting"/>
    </classes>
  </test> <!-- Test -->
</suite> <!-- Suite -->
```