

Project : - Pizza Hut automation using Cucumber

Package – Feature
pizza.feature

Feature: This feature will be used to automate PizzaHut Website

Scenario: This scenario will be used to Place the Order

Given I have launched the application

When I enter the location as "Pune"

And I select the very first suggestion from the list

Then I should land on the Deals page

And I select the tab as "Pizzas"

And I add "Schezwan Margherita" to the basket

And I note down the price displayed on the screen

Then I should see the pizza "Personal Schezwan Margherita" is added to the cart

And price is also displayed correctly

And I click on the Checkout button

Then I should be landed on the secured checkout page

And I enter the personal details

| name | Rajesh Sharma |

| phone | 1234567890 |

| email | abc@xyz.com |

And I enter the address details

| 123 Main Street |

| Some Landmark |

Package – stepdefs

Hooks.java

```
package stepdef;
```

```
import org.openqa.selenium.WebDriver;
```

```
import org.openqa.selenium.chrome.ChromeDriver;
```

```
import com.relevantcodes.extentreports.ExtentReports;
```

```
import com.relevantcodes.extentreports.ExtentTest;
```

```
import io.cucumber.java.After;
```

```
import io.cucumber.java.Before;
```

```
import io.cucumber.java.Scenario;
```

```
public class Hooks {
```

```
    public static WebDriver driver;
```

```
    public static ExtentReports report = new ExtentReports("HtmlResults.html",false);
```

```
    public static ExtentTest test;
```

```
    @Before
```

```
    public void SetUp(Scenario scenario) {
```

```
        test = report.startTest(scenario.getName());
```

```
        driver = new ChromeDriver();
```

```
    }
```

```
    @After
```

```
    public void TearDown() {
```

```
        driver.quit();
```

```
        report.endTest(test);
```

```
        report.close();
```

```
    }
```

```
}
```

```

test.java
package stepdef;

import java.time.Duration;
import java.util.List;
import java.util.Map;
import java.util.Set;

import org.junit.Assert;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;

import com.relevantcodes.extentreports.LogStatus;

import io.cucumber.java.en.Given;
import io.cucumber.java.en.Then;
import io.cucumber.java.en.When;

public class tests {
    WebDriver driver = Hooks.driver;
    @Given("I have launched the application")
    public void i_have_launched_the_application() {
        // Write code here that turns the phrase above into concrete actions
        driver.get("https://www.pizzahut.co.in/");
        driver.manage().window().maximize();
        driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(5));
        Hooks.test.log(LogStatus.PASS, "Application Launched");
    }

    @When("I enter the location as {string}")
    public void i_enter_the_location_as(String location) {
        // Write code here that turns the phrase above into concrete actions
        WebElement loc = driver.findElement(By.cssSelector("input[type='text']"));
        loc.sendKeys(location);
        Hooks.test.log(LogStatus.PASS, "Location entered");
    }

    @When("I select the very first suggestion from the list")
    public void i_select_the_very_first_suggestion_from_the_list() {
        // Write code here that turns the phrase above into concrete actions
        WebElement selectloc = driver.findElement(By.xpath("//button[2]"));
        selectloc.click();
        Hooks.test.log(LogStatus.PASS, "clicked on 1st suggestion");
    }

    @Then("I should land on the Deals page")

```

```

public void i_should_land_on_the_Deals_page() {
    // Write code here that turns the phrase above into concrete actions
    WebElement page = driver.findElement(By.cssSelector("div.side-menu>div>a:nth-
of-type(1)"));
    Assert.assertTrue(page.isDisplayed());
    Assert.assertTrue(page.isEnabled());
    System.out.println("The current title is :- " + driver.getTitle());
    Hooks.test.log(LogStatus.PASS, "Deals Page");

}

@Then("I select the tab as {string}")
public void i_select_the_tab_as(String pza) {
    // Write code here that turns the phrase above into concrete actions
    WebElement pizza = driver.findElement(By.xpath("//a[2]/span[contains(text(),'+ pza
+')]]"));
    pizza.click();
    System.out.println("The current title is" + driver.getTitle());
    Hooks.test.log(LogStatus.PASS, "selected Pizzas");

}

@Then("I add {string} to the basket")
public void i_add_to_the_basket(String addcart) {
    // Write code here that turns the phrase above into concrete actions
    WebElement additem =
driver.findElement(By.xpath("//span[@data-testid='pizzas']/div[1]/a[2]/div[3]/div/button"));
    additem.click();

    WebElement additem2 = driver.findElement(By.xpath("//span[@data-
testid='pizzas']/div[1]/a[1]/div[3]/div/button"));
    additem2.click();
    Hooks.test.log(LogStatus.PASS, "Add items in basket");

}

@Then("I note down the price displayed on the screen")
public void i_note_down_the_price_displayed_on_the_screen() {
    WebElement price =
driver.findElement(By.xpath("//span[@data-testid='pizzas']/div[1]/a[2]/div[3]/div/button/span[2]"));
    Assert.assertTrue(price.isDisplayed());
    System.out.println("The price is: -"+price.getText());
    Hooks.test.log(LogStatus.PASS, "Proper price displayed");

}

@Then("I should see the pizza {string} is added to the cart")
public void i_should_see_the_pizza_is_added_to_the_cart(String Expitem) {

```

```

        // Write code here that turns the phrase above into concrete actions
        WebElement bitem = driver.findElement(By.xpath("//*[text()='"+ Expitem +"' ]"));
        String Addeditem = bitem.getText();
        Assert.assertEquals(Expitem,Addeditem);
        System.out.println("The added item is: -"+ Addeditem);
        Hooks.test.log(LogStatus.PASS, "Items added in the basket");
    }

    @Then("price is also displayed correctly")
    public void price_is_also_displayed_correctly() {
        // Write code here that turns the phrase above into concrete actions
        WebElement crprice=
driver.findElement(By.xpath("//div[@id='basket']/div[3]/div/div/div[2]"));
        String addedprice = crprice.getText();

        Assert.assertTrue(crprice.isDisplayed());

        System.out.println("The price before adding: -"+addedprice);
        Hooks.test.log(LogStatus.PASS, "Correct price display of added items");
    }

    @Then("I click on the Checkout button")
    public void i_click_on_the_Checkout_button() {
        // Write code here that turns the phrase above into concrete actions
        WebElement chk = driver.findElement(By.xpath("//span[text()='Checkout']"));
        chk.click();
        Hooks.test.log(LogStatus.PASS, "Clicked on checkout button");
    }

    @Then("I should be landed on the secured checkout page")
    public void i_should_be_landed_on_the_secured_checkout_page() {
        // Write code here that turns the phrase above into concrete actions
        WebElement checkoutpage =
driver.findElement(By.xpath("//*[contains(text(),'Secure Checkout')]"));
        Assert.assertTrue(checkoutpage.isDisplayed());
        System.out.println("The title of Checkout page is: - "+ driver.getTitle());
        Hooks.test.log(LogStatus.PASS, "Landed on secured checkout page");
    }

    @Then("I enter the personal details")
    public void i_enter_the_personal_details(Map<String, String> details) {
        // Write code here that turns the phrase above into concrete actions
        // For automatic transformation, change DataTable to one of
        // E, List<E>, List<List<E>>, List<Map<K,V>>, Map<K,V> or
        // Map<K, List<V>>. E,K,V must be a String, Integer, Float,
        // Double, Byte, Short, Long, BigInteger or BigDecimal.
    }

```

```
// For other transformations you can register a DataTableType.
```

```
Set<String> usrkeys = details.keySet();
for(String a: usrkeys) {
    WebElement usrdetails = driver.findElement(By.cssSelector("input[name='"+
a +"']"));
    usrdetails.sendKeys(details.get(a));
}
Hooks.test.log(LogStatus.PASS, "Entered personal details");
}
```

```
@Then("I enter the address details")
public void i_enter_the_address_details(List<String> addrdetail) {
    // Write code here that turns the phrase above into concrete actions
    // For automatic transformation, change DataTable to one of
    // E, List<E>, List<List<E>>, List<Map<K,V>>, Map<K,V> or
    // Map<K, List<V>>. E,K,V must be a String, Integer, Float,
    // Double, Byte, Short, Long, BigInteger or BigDecimal.
    //
    // For other transformations you can register a DataTableType.
    System.out.println(addrdetail);
    WebElement addline1 =
driver.findElement(By.cssSelector("input[name='deliveryAddress.interior']"));
    addline1.sendKeys(addrdetail.get(0));
    WebElement addline2 =
driver.findElement(By.cssSelector("input[name='deliveryAddress.notes']"));
    addline2.sendKeys(addrdetail.get(1));
    Hooks.test.log(LogStatus.PASS, "Entered Address details");

}

}
```

```
TestRunner.java
package stepdef;

import org.junit.runner.RunWith;

import io.cucumber.junit.Cucumber;
import io.cucumber.junit.CucumberOptions;

@RunWith(Cucumber.class)
@CucumberOptions(
    features = "src/test/java/feature",
    glue = {"stepdef"},
    plugin = {"pretty", "html:target/cucumber-reports.html"},
    monochrome = true
)
public class TestRunner {

}
```

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 http://maven.apache.org/xsd/maven-
4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.simplilearn</groupId>
  <artifactId>PizzahutAutomation</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>PizzahutAutomation</name>
  <url>http://maven.apache.org</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>

  <dependencies>
    <dependency>
      <groupId>io.cucumber</groupId>
      <artifactId>cucumber-java</artifactId>
      <version>5.6.0</version>
    </dependency>

    <!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-junit -->
    <dependency>
      <groupId>io.cucumber</groupId>
      <artifactId>cucumber-junit</artifactId>
      <version>5.6.0</version>
      <scope>test</scope>
    </dependency>

    <dependency>
      <groupId>org.seleniumhq.selenium</groupId>
      <artifactId>selenium-java</artifactId>
      <version>4.21.0</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/com.relevantcodes/extentreports -->
    <dependency><!--This one is for Extent reports -->
      <groupId>com.relevantcodes</groupId>
      <artifactId>extentreports</artifactId>
      <version>2.41.2</version>
    </dependency>

  </dependencies>

</project>
```


Github - https://github.com/sivalingaa/pizzahut_automation_cucumber.git