Source Code

Feature file:

Feature: Phase End project

Scenario: Star health home page test

Given User open the chrome browser and starhealth application

When User waits for the Welcome to Star Health pop-up and closes it

Then validates the Star Health home page title using a JUnit assertion

Then clicks on the Buy Now button

And User enters following details

| URL | FullName | PhNo | PIN | Email |

| admin | admin | 9380756810 | 585326| admin@gmail.com|

| maya | admin | 9380756810 | 585326 | admin@gmail.com|

And User clicks on the Star Health logo

Then Close all the browsers

StarHealth.java file:

package steps;

import java.util.List;

import java.util.Map;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import org.openqa.selenium.JavascriptExecutor;

import org.junit.Assert;

import org.openqa.selenium.By;

import org.openqa.selenium.NoSuchElementException;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.support.ui.Select;

import io.cucumber.datatable.DataTable;

import io.cucumber.java.en.And;

import io.cucumber.java.en.Given;

import io.cucumber.java.en.Then;

import io.cucumber.java.en.When;

public class StarHealth {

WebDriver driver;

@Given("User open the chrome browser and starhealth application")

public void user\_open\_the\_chrome\_browser\_and\_starhealth\_application() {

//System.setProperty("webdriver.chrome.driver", "C:\\Program Files\\Google\\Chrome\\Application\\chromedriver.exe");

driver = new ChromeDriver();

driver.manage().window().maximize();

driver.get("https://www.starhealth.in");

}

@When("User waits for the Welcome to Star Health pop-up and closes it")

public void user\_waits\_for\_the\_welcome\_to\_star\_health\_pop\_up\_and\_closes\_it() {

try {

// Implement code to check if the pop-up is present

WebElement popup = driver.findElement(By.id("popupLocator")); // Replace with the actual locator

if (popup.isDisplayed()) {

// If the pop-up is displayed, close it by clicking a close button or other method

WebElement closeButton = driver.findElement(By.id("closeButtonLocator")); // Replace with the actual locator

closeButton.click();

}

} catch (NoSuchElementException e) {

// Handle the case where the pop-up is not present

System.out.println("Welcome pop-up not found. Continuing without closing.");

}

}

@Then("validates the Star Health home page title using a JUnit assertion")

public void validates\_the\_star\_health\_home\_page\_title\_using\_a\_j\_unit\_assertion() {

// Get the page title using driver.getTitle()

// Get the actual page title using WebDriver

String actualTitle = driver.getTitle();

// Define the expected title

String expectedTitle = "Star Health Insurance: Medical, Accident and Travel insurance policies"; // Replace with the expected title

// Use JUnit's assertion to validate the title

Assert.assertEquals("Page title does not match", expectedTitle, actualTitle);

}

@Then("clicks on the Buy Now button")

public void clicks\_on\_the\_buy\_now\_button() {

// Implement code to locate and click on the "Buy Now" button

try {

WebElement buyNowButton = driver.findElement(By.xpath("//\*[@id=\"\_\_next\"]/div/div/header/div[2]/div[2]/div/button")); // Replace with the actual locator

buyNowButton.click();

} catch (NoSuchElementException e) {

// Handle the case where the "Buy Now" button is not found

System.out.println("Buy Now button not found.");

}

}

@And("User enters following details")

public void user\_enters\_following\_details(DataTable datatable) {

// Use the DataTable to fetch and input data

List<Map<String, String>> userList = datatable.asMaps(String.class, String.class);

for (Map<String, String> data : userList) {

String fullName = data.get("FullName");

String phNo = data.get("PhNo");

String pin = data.get("PIN");

String email = data.get("Email");

// String getstarted = data.get("Submit");

// Implement code to interact with input fields and enter data

driver.findElement(By.xpath("//\*[@id=\"fullname\"]")).clear();

driver.findElement(By.xpath("//\*[@id=\"fullname\"]")).sendKeys(fullName);

driver.findElement(By.xpath("//\*[@id=\"mobile\"]")).clear();// Replace with the actual locator for FullName field

driver.findElement(By.xpath("//\*[@id=\"mobile\"]")).sendKeys(phNo);

driver.findElement(By.xpath("//\*[@id=\"zipcode\"]")).clear();// Replace with the actual locator for PhNo field

driver.findElement(By.xpath("//\*[@id=\"zipcode\"]")).sendKeys(pin); // Replace with the actual locator for PIN field

driver.findElement(By.xpath("//\*[@id=\"email\"]")).clear();

driver.findElement(By.xpath("//\*[@id=\"email\"]")).sendKeys(email);

driver.findElement(By.xpath("//\*[@id=\"userDetails-form\"]/div/div/div/div[1]/div/div/div[2]/div/button")).submit();

// driver.findElement(By.xpath("//\*[@id=\"\_\_next\"]/div/main/div[2]/div[2]/div/div[2]/div/div/div[2]/div[2]/div/div/div[1]/div[1]/div[2]/button")).click();

// driver.findElement(By.xpath("//\*[@id=\":r7:\"]")).sendKeys(email);

}

}

@And("User clicks on the Star Health logo")

public void user\_clicks\_on\_the\_star\_health\_logo() {

// Implement code to click on the Star Health logo to go back to the homepage

try {

WebElement logoElement = driver.findElement(By.xpath("//img[@class = 'jss341']")); // Replace with the actual locator

logoElement.click();

} catch (NoSuchElementException e) {

System.out.println("Star Health logo not found.");

}

}

@Then("Close all the browsers")

public void close\_all\_the\_browsers() {

System.out.println("Browser closed");

}

}

Runner file:

package steps;

import org.junit.runner.RunWith;

import io.cucumber.junit.Cucumber;

import io.cucumber.junit.CucumberOptions;

@RunWith(Cucumber.class)

@CucumberOptions(

features = "C:\\Users\\PC\\eclipse-workspace\\starhealths\\src\\test\\resources\\Feature\\StarHealth.feature",

glue = {"steps"},

dryRun = false,

plugin = {"pretty",

"html:target/Testreport.html",

"com.aventstack.extentreports.cucumber.adapter.ExtentCucumberAdapter:",

"timeline:test-output-thread/"

},

monochrome = true // console output will be in more readable format

)

public class MyRunner {

}

POM FILE:

<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>bddtest</groupId>

<artifactId>starhealths</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

<dependency>

<groupId>org.seleniumhq.selenium</groupId>

<artifactId>selenium-java</artifactId>

<version>4.11.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/io.github.bonigarcia/webdrivermanager -->

<dependency>

<groupId>io.github.bonigarcia</groupId>

<artifactId>webdrivermanager</artifactId>

<version>5.4.1</version>

</dependency>

<dependency>

<groupId>commons-io</groupId>

<artifactId>commons-io</artifactId>

<version>2.11.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.testng/testng -->

<dependency>

<groupId>org.testng</groupId>

<artifactId>testng</artifactId>

<version>7.7.1</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/com.aventstack/extentreports -->

<dependency>

<groupId>com.aventstack</groupId>

<artifactId>extentreports</artifactId>

<version>5.0.9</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.poi/poi -->

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi</artifactId>

<version>4.1.1</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml</artifactId>

<version>4.1.1</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.poi/poi-ooxml-schemas -->

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml-schemas</artifactId>

<version>4.1.1</version>

</dependency>

<dependency>

<groupId>org.apache.logging.log4j</groupId>

<artifactId>log4j-api</artifactId>

<version>2.14.1</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.logging.log4j/log4j-core -->

<dependency>

<groupId>org.apache.logging.log4j</groupId>

<artifactId>log4j-core</artifactId>

<version>2.14.1</version>

</dependency>

<dependency>

<groupId>org.junit.jupiter</groupId>

<artifactId>junit-jupiter-api</artifactId>

<version>5.9.3</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-java</artifactId>

<version>7.10.1</version>

</dependency>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-junit</artifactId>

<version>7.10.1</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-core</artifactId>

<version>7.10.1</version>

</dependency>

<dependency>

<groupId>tech.grasshopper</groupId>

<artifactId>extentreports-cucumber7-adapter</artifactId>

<version>1.7.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-testng -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-testng</artifactId>

<version>7.10.1</version>

</dependency>

</dependencies>

</project>