**Source Code**

**Test Scenario: 1 [Cucumber + JUnit]:**

**starHealth.feature**

Feature: Test the starHealth page on Chrome Browser

Scenario Outline: Validate the Star Health Buy Now flow

Given User launches the Star Health application with "<URL>"

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

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

When User clicks on the Buy Now button

Then User types Name as "<FullName>"

Then User types Phone as "<PhNo>"

Then User types the PIN as "<PIN>"

Then User clicks on I need health insurance from the drop-down menu

When User selects the option "<optionPlan>"

Then User sees the Plan for My Family page

When User validates that the mobile number is the same as the previously entered number using a JUnit assertion

Then User clicks on the Star Health logo

Then The Application should redirect to the home page

When User closes the child tab

Then User navigates back to the parent tab

Examples:

| URL | FullName | PhNo | PIN | optionPlan |

|https://www.starhealth.in/|Abantika Padhy|8455089447|7650223|My Family|

**BaseTest.java**

**package** base;

**import** org.openqa.selenium.WebDriver;

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

**import** io.cucumber.java.AfterAll;

**public** **class** BaseTest {

**public** **static** WebDriver *driver*=**new** ChromeDriver();

@AfterAll

**public** **void** Close\_Browser() **throws** InterruptedException {

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

*driver*.close();

}

}

**BuyNowPage.java**

**package** pages;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

**import** org.openqa.selenium.support.FindBy;

**import** org.openqa.selenium.support.PageFactory;

**public** **class** BuyNowPage {

@FindBy(xpath = "//input[@id='name']")

**public** **static** WebElement *name*;

@FindBy(xpath = "//input[@id='contact\_no']")

**public** **static** WebElement *contactNo*;

@FindBy(xpath = "//input[@id='pinCode']")

**public** **static** WebElement *pinCode*;

@FindBy(xpath = "//div[@id='grouped-select']")

**public** **static** WebElement *dropDown*;

@FindBy(xpath = "//input[@id='contact\_no']")

**public** **static** WebElement *contactNumberVeriry*;

@FindBy(xpath = "//img[@class='jss341']")

**public** **static** WebElement *logo*;

**public** BuyNowPage(WebDriver driver) {

PageFactory.*initElements*(driver, **this**);

// driver can now work with your elements and methods of this class

}

**public** **void** give\_name(String fullname) **throws** InterruptedException {

Thread.*sleep*(2000);

*name*.sendKeys(fullname);

}

**public** **void** give\_phno(String phno) **throws** InterruptedException {

Thread.*sleep*(2000);

*contactNo*.sendKeys(phno);

}

**public** **void** give\_pin(String pin) **throws** InterruptedException {

Thread.*sleep*(2000);

*pinCode*.sendKeys(pin);

}

**public** **void** click\_dropdown() **throws** InterruptedException {

Thread.*sleep*(2000);

*dropDown*.click();

}

**public** String ContactNo\_verify() **throws** InterruptedException {

Thread.*sleep*(2000);

String C\_no = *contactNumberVeriry*.getAttribute("value");

**return** C\_no;

}

**public** **void** click\_logo() **throws** InterruptedException {

Thread.*sleep*(2000);

*logo*.click();

}

}

**HomePage.java**

**package** pages;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

**import** org.openqa.selenium.support.FindBy;

**import** org.openqa.selenium.support.PageFactory;

**import** base.BaseTest;

**public** **class** HomePage {

@FindBy(xpath = "//span[text()='Buy Now']")

**public** **static** WebElement *buyNow*;

**public** HomePage(WebDriver driver) {

PageFactory.*initElements*(driver, **this**);

// driver can now work with your elements and methods of this class

}

**public** String get\_homepage\_title() **throws** InterruptedException {

String title = BaseTest.*driver*.getTitle();

**return** title;

}

**public** **void** click\_On\_BuyNow() **throws** InterruptedException {

Thread.*sleep*(2000);

*buyNow*.click();

}

}

**StepDefination.java**

**package** step;

**import** org.junit.Assert;

**import** org.openqa.selenium.By;

**import** base.BaseTest;

**import** io.cucumber.java.en.Given;

**import** io.cucumber.java.en.Then;

**import** io.cucumber.java.en.When;

**import** pages.BuyNowPage;

**import** pages.HomePage;

**public** **class** StepDefination {

**public** **static** String *ph\_no*;

HomePage homePage=**new** HomePage(BaseTest.*driver*);

BuyNowPage buyNowPage=**new** BuyNowPage(BaseTest.*driver*);

@Given("User launches the Star Health application with {string}")

**public** **void** user\_launches\_the\_star\_health\_application\_with(String URL) {

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

BaseTest.*driver*.get(URL);

}

@Then("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() {

}

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

**public** **void** user\_validates\_the\_star\_health\_home\_page\_title\_using\_a\_j\_unit\_assertion() **throws** InterruptedException {

Thread.*sleep*(2000);

String title=homePage.get\_homepage\_title();

**boolean** isEqual=title.equals("Star Health Insurance: Medical, Accident and Travel insurance policies");

Assert.*assertTrue*(isEqual);

//System.out.println("Star Health Home Page Title: " + title);

}

@When("User clicks on the Buy Now button")

**public** **void** user\_clicks\_on\_the\_buy\_now\_button() **throws** InterruptedException {

Thread.*sleep*(2000);

homePage.click\_On\_BuyNow();

}

@Then("User types Name as {string}")

**public** **void** user\_types\_name\_as(String FullName) **throws** InterruptedException {

Thread.*sleep*(2000);

buyNowPage.give\_name(FullName);

}

@Then("User types Phone as {string}")

**public** **void** user\_types\_phone\_as(String PhNo) **throws** InterruptedException {

*ph\_no*=PhNo;

buyNowPage.give\_phno(PhNo);

}

@Then("User types the PIN as {string}")

**public** **void** user\_types\_the\_pin\_as(String PIN) **throws** InterruptedException {

buyNowPage.give\_pin(PIN);

}

@Then("User clicks on I need health insurance from the drop-down menu")

**public** **void** user\_clicks\_on\_i\_need\_health\_insurance\_from\_the\_drop\_down\_menu() **throws** InterruptedException {

buyNowPage.click\_dropdown();

Thread.*sleep*(1000);

}

@When("User selects the option {string}")

**public** **void** user\_selects\_the\_option(String OptionPlan) **throws** InterruptedException {

BaseTest.*driver*.findElement(By.*xpath*("//li[text()="+"'"+OptionPlan+"'"+"]")).click();

Thread.*sleep*(3000);

}

@Then("User sees the Plan for My Family page")

**public** **void** user\_sees\_the\_plan\_for\_my\_family\_page() {

}

@When("User validates that the mobile number is the same as the previously entered number using a JUnit assertion")

**public** **void** user\_validates\_that\_the\_mobile\_number\_is\_the\_same\_as\_the\_previously\_entered\_number\_using\_a\_j\_unit\_assertion() **throws** InterruptedException {

Thread.*sleep*(3000);

String phoneNumber=buyNowPage.ContactNo\_verify();

Assert.*assertEquals*("Phone Number Matched ", *ph\_no*, phoneNumber);

}

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

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

buyNowPage.click\_logo();

}

@Then("The Application should redirect to the home page")

**public** **void** the\_application\_should\_redirect\_to\_the\_home\_page() {

}

@When("User closes the child tab")

**public** **void** user\_closes\_the\_child\_tab() {

}

@Then("User navigates back to the parent tab")

**public** **void** user\_navigates\_back\_to\_the\_parent\_tab() {

}

}

**StarHealthTestRunner.java**

**package** runner;

**import** org.junit.runner.RunWith;

**import** io.cucumber.junit.Cucumber;

**import** io.cucumber.junit.CucumberOptions;

@RunWith(Cucumber.**class**)

@CucumberOptions(features="C:\\Users\\HP\\eclipse-workspace-new\\Phase2-End-Project1\\src\\test\\java\\features\\starHealth.feature",

glue= {"step"},

plugin= {"html:target/Cucumberreport.html",

"pretty",

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

"timeline:test-output-thread/"

})

**public** **class** StarHealthTestRunner {

}

**extent.properties**

extent.reporter.spark.start=true

extent.reporter.spark.out=test-output/SparkReport/Spark.html

extent.reporter.spark.config=src/test/resources/extent-config.xml

extent.reporter.spark.out=test-output/SparkReport/

screenshot.dir=test-output/

screenshot.rel.path=../

extent.reporter.pdf.start=true

extent.reporter.pdf.out=test output/PdfReport/ExtentPdf.pdf

#basefolder.name=reports

#basefolder.datetimepattern=d-MMM-YY HH-mm-ss

extent.reporter.spark.vieworder=dashboard,test,category,exception,author,device,log

systeminfo.os=Windows

systeminfo.user=Asish

systeminfo.build=1.1

systeminfo.AppName=StarHealthTest

**extent-config.xml**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<extentreports>

<configuration>

<!-- report theme -->

<!-- standard, dark -->

<theme>standard</theme>

<!-- document encoding -->

<!-- defaults to UTF-8 -->

<encoding>UTF-8</encoding>

<!-- protocol for script and stylesheets -->

<!-- defaults to https -->

<protocol>http</protocol>

<!-- title of the document -->

<documentTitle>Extent</documentTitle>

<!-- report name - displayed at top-nav -->

<reportName>Grasshopper Report</reportName>

<!-- location of charts in the test view -->

<!-- top, bottom -->

<testViewChartLocation>bottom</testViewChartLocation>

<!-- custom javascript -->

<scripts>

<![CDATA[

$(document).ready(function() {

});

]]>

</scripts>

<!-- custom styles -->

<styles>

<![CDATA[

]]>

</styles>

</configuration>

</extentreports>

**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>Phase2-End-Project1</groupId>

<artifactId>Phase2-End-Project1</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>jar</packaging>

<name>Phase2-End-Project1</name>

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

<properties>

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

</properties>

<dependencies>

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.11</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>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>tech.grasshopper</groupId>

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

<version>1.7.0</version>

</dependency>

</dependencies>

<build>

<pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->

<plugins>

<!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#clean\_Lifecycle -->

<plugin>

<artifactId>maven-clean-plugin</artifactId>

<version>3.1.0</version>

</plugin>

<!-- default lifecycle, jar packaging: see https://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin\_bindings\_for\_jar\_packaging -->

<plugin>

<artifactId>maven-resources-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.0</version>

</plugin>

<plugin>

<artifactId>maven-surefire-plugin</artifactId>

<version>2.22.1</version>

<configuration>

<includes>

<include>\*\*/\*Runner.java</include>

</includes>

</configuration>

</plugin>

<plugin>

<artifactId>maven-jar-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-install-plugin</artifactId>

<version>2.5.2</version>

</plugin>

<plugin>

<artifactId>maven-deploy-plugin</artifactId>

<version>2.8.2</version>

</plugin>

<!-- site lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#site\_Lifecycle -->

<plugin>

<artifactId>maven-site-plugin</artifactId>

<version>3.7.1</version>

</plugin>

<plugin>

<artifactId>maven-project-info-reports-plugin</artifactId>

<version>3.0.0</version>

</plugin>

</plugins>

</pluginManagement>

</build>

</project>

**Test Scenario: 2 [TestNG + Selenium]**

**BaseTest.java**

**package** base;

**import** java.awt.Desktop;

**import** java.io.File;

**import** java.io.IOException;

**import** java.lang.reflect.Method;

**import** org.openqa.selenium.WebDriver;

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

**import** org.testng.ITestContext;

**import** org.testng.ITestResult;

**import** org.testng.annotations.AfterClass;

**import** org.testng.annotations.AfterMethod;

**import** org.testng.annotations.AfterSuite;

**import** org.testng.annotations.BeforeClass;

**import** org.testng.annotations.BeforeSuite;

**import** com.aventstack.extentreports.ExtentReports;

**import** com.aventstack.extentreports.ExtentTest;

**import** com.aventstack.extentreports.reporter.ExtentSparkReporter;

**public** **class** BaseTest {

**public** **static** WebDriver *driver*;

**public** **static** ExtentReports *extentreports*;

**public** **static** ExtentTest *extentTest*;

@BeforeClass

**public** **void** Open\_Browser(ITestContext context) {

*driver*=**new** ChromeDriver();

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

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

*extentTest*= *extentreports*.createTest(context.getName());

}

@AfterClass

**public** **void** close\_Browser() **throws** InterruptedException {

Thread.*sleep*(2000);

*driver*.close();

}

@BeforeSuite // this method will be excuted before suite begins its execution

**public** **void** InitalizeExtentReport()

{

ExtentSparkReporter sparkreporter = **new** ExtentSparkReporter("report.html");

*extentreports* = **new** ExtentReports();

*extentreports*.attachReporter(sparkreporter);

// on the report display more information about OS, browser, java etc

*extentreports*.setSystemInfo("OS", System.*getProperty*("os.name"));

*extentreports*.setSystemInfo("JAVA", System.*getProperty*("java.version"));

}

@AfterSuite

**public** **void** generateReports() **throws** IOException

{

*extentreports*.flush();

Desktop.*getDesktop*().browse(**new** File("report.html").toURI());

}

@AfterMethod

**public** **void** generateTestStatus(Method m,ITestResult result) **throws** InterruptedException

{

**if**(result.getStatus() == ITestResult.***FAILURE*** )

{

System.***out***.println("Capture Screenshot");

*extentTest*.fail(result.getThrowable());

}

**else** **if** (result.getStatus() == ITestResult.***SUCCESS***) {

*extentTest*.pass(m.getName() + " is passed");

}

}

}

**XlS\_dataprovider.java**

**package** dataprovider;

**import** java.io.FileInputStream;

**import** java.io.IOException;

**import** org.apache.poi.EncryptedDocumentException;

**import** org.apache.poi.ss.usermodel.Sheet;

**import** org.apache.poi.ss.usermodel.Workbook;

**import** org.apache.poi.ss.usermodel.WorkbookFactory;

**public** **class** XlS\_dataprovider {

**static** Workbook *workbook*;

**static** Sheet *sheet*;

**static** String *path* = "C:\\Users\\HP\\OneDrive\\Desktop\\phase-2-End-project-2\\testdata.xlsx";

**public** **static** Object[][] getTestData(String sheetName) **throws** EncryptedDocumentException, IOException {

FileInputStream file = **null**;

file = **new** FileInputStream(*path*);

*workbook* = WorkbookFactory.*create*(file);

*sheet* = *workbook*.getSheet(sheetName);

Object[][] inputdata = **new** Object[*sheet*.getLastRowNum()][*sheet*.getRow(0).getLastCellNum()];

**for** (**int** i = 0; i < *sheet*.getLastRowNum(); i++) {

**for** (**int** j = 0; j < *sheet*.getRow(0).getLastCellNum(); j++) {

inputdata[i][j] = *sheet*.getRow(i + 1).getCell(j).toString();

}

}

**return** inputdata;

}

}

**XlS\_dataprovider2.java**

**package** dataprovider;

**import** java.io.FileInputStream;

**import** java.io.IOException;

**import** org.apache.poi.EncryptedDocumentException;

**import** org.apache.poi.ss.usermodel.Sheet;

**import** org.apache.poi.ss.usermodel.Workbook;

**import** org.apache.poi.ss.usermodel.WorkbookFactory;

**public** **class** XlS\_dataprovider2 {

**static** Workbook *workbook*;

**static** Sheet *sheet*;

**static** String *path* = "C:\\Users\\HP\\OneDrive\\Desktop\\phase-2-End-project-2\\testdata2.xlsx";

**public** **static** Object[][] getTestData2(String sheetName) **throws** EncryptedDocumentException, IOException {

FileInputStream file = **null**;

file = **new** FileInputStream(*path*);

*workbook* = WorkbookFactory.*create*(file);

*sheet* = *workbook*.getSheet(sheetName);

Object[][] inputdata = **new** Object[*sheet*.getLastRowNum()][*sheet*.getRow(0).getLastCellNum()];

**for** (**int** i = 0; i < *sheet*.getLastRowNum(); i++) {

**for** (**int** j = 0; j < *sheet*.getRow(0).getLastCellNum(); j++) {

inputdata[i][j] = *sheet*.getRow(i + 1).getCell(j).toString();

}

}

**return** inputdata;

}

}

**HealthInsurancePage.java**

**package** pages;

**import** java.util.ArrayList;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

**import** org.openqa.selenium.support.FindBy;

**import** org.openqa.selenium.support.PageFactory;

**import** base.BaseTest;

**public** **class** HealthInsurancePage **extends** BaseTest {

@FindBy(xpath = "//input[@id='name']")

**public** WebElement name;

@FindBy(xpath = "//input[@id='phoneNumber']")

**public** WebElement phnumber;

@FindBy(xpath = "//input[@id='pinCode']")

**public** WebElement pinCode;

@FindBy(xpath = "//span[text()='Get a quote']")

**public** WebElement GetQuoteButton;

@FindBy(xpath = "//input[@id='email']")

**public** WebElement emailField;

@FindBy(xpath = "//span[text()='Next']")

**public** WebElement nextButton;

@FindBy(xpath = "//span[text()='Back']")

**public** WebElement backButton;

@FindBy(xpath = "//input[@id='mobile']")

**public** WebElement mobileField;

@FindBy(xpath = "(//a[@class='mr-4 mr-10-xs'])[1]")

**public** WebElement fb;

@FindBy(xpath = "(//a[@class='mr-4 mr-10-xs'])[2]")

**public** WebElement youTube;

@FindBy(xpath = "(//a[@class='mr-4 mr-10-xs'])[3]")

**public** WebElement Linkdln;

@FindBy(xpath = "(//a[@class='mr-4 mr-10-xs'])[4]")

**public** WebElement twitter;

@FindBy(xpath = "(//a[@class='mr-4 mr-10-xs'])[5]")

**public** WebElement insta;

**public** HealthInsurancePage(WebDriver driver) {

PageFactory.*initElements*(driver, **this**);

}

**public** **void** Give\_Input\_And\_Click\_GetaQuote\_Button(String Name, String phNumber, String pincode)

**throws** InterruptedException {

Thread.*sleep*(2000);

name.sendKeys(Name);

phnumber.sendKeys(phNumber);

pinCode.sendKeys(pincode);

GetQuoteButton.click();

}

**public** **void** Give\_Input\_in\_Authorize\_Form\_And\_Click\_Next(String email) **throws** InterruptedException {

emailField.sendKeys(email);

Thread.*sleep*(1000);

nextButton.click();

Thread.*sleep*(1000);

backButton.click();

}

**public** ArrayList<String> Fetch\_The\_Value() **throws** InterruptedException {

Thread.*sleep*(2000);

ArrayList<String> list = **new** ArrayList<String>();

String nameValue = name.getAttribute("value");

String PhValue = mobileField.getAttribute("value");

String emailValue = emailField.getAttribute("value");

list.add(nameValue);

list.add(PhValue);

list.add(emailValue);

**return** list;

}

**public** ArrayList<String> fetch\_Social\_media\_links() **throws** InterruptedException {

Thread.*sleep*(1000);

ArrayList<String> list\_Of\_Link = **new** ArrayList<String>();

String fblink = fb.getAttribute("href");

String Linkdlnlink = Linkdln.getAttribute("href");

String twitterlink = twitter.getAttribute("href");

String youtubelink = youTube.getAttribute("href");

String instaLink = insta.getAttribute("href");

list\_Of\_Link.add(fblink);

list\_Of\_Link.add(youtubelink);

list\_Of\_Link.add(Linkdlnlink);

list\_Of\_Link.add(twitterlink);

list\_Of\_Link.add(instaLink);

**return** list\_Of\_Link;

}

**public** **void** clik\_on\_twitter() {

twitter.click();

}

**public** String validate\_twitter() {

String T\_url = *driver*.getCurrentUrl();

**return** T\_url;

}

}

**HomePage.java**

**package** pages;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.WebElement;

**import** org.openqa.selenium.interactions.Actions;

**import** org.openqa.selenium.support.FindBy;

**import** org.openqa.selenium.support.PageFactory;

**import** base.BaseTest;

**public** **class** HomePage **extends** BaseTest {

Actions a = **new** Actions(*driver*);

@FindBy(xpath = "//img[@class='cursor-pointer']")

**public** WebElement logo;

@FindBy(xpath = "//p[text()='Plans']")

**public** WebElement plans;

@FindBy(xpath = "(//a[text()='For My Family'])[1]")

**public** WebElement ForMyFamily;

**public** HomePage(WebDriver driver) {

PageFactory.*initElements*(driver, **this**);

// driver can now work with your elements and methods of this class

}

**public** **void** Hover\_over\_Plans() **throws** InterruptedException {

a.moveToElement(plans).build().perform();

Thread.*sleep*(3500);

ForMyFamily.click();

}

}

**StarHealthPageTest.java**

**package** testpages;

**import** java.io.IOException;

**import** java.util.ArrayList;

**import** org.apache.poi.EncryptedDocumentException;

**import** org.testng.Assert;

**import** org.testng.annotations.DataProvider;

**import** org.testng.annotations.Parameters;

**import** org.testng.annotations.Test;

**import** base.BaseTest;

**import** dataprovider.XlS\_dataprovider;

**import** dataprovider.XlS\_dataprovider2;

**import** pages.HealthInsurancePage;

**import** pages.HomePage;

**public** **class** StarHealthPageTest **extends** BaseTest {

HomePage homePage;

HealthInsurancePage healthInsurancePage;

@Test(priority = 1)

**public** **void** Validate\_Alt\_Value\_Of\_Logo() **throws** InterruptedException {

homePage = **new** HomePage(*driver*);

Thread.*sleep*(3000);

String Alt\_Value = homePage.logo.getAttribute("alt");

Thread.*sleep*(5000);

**boolean** isValidated = Alt\_Value.equals("Star Health Logo");

System.***out***.println(Alt\_Value);

Assert.*assertTrue*(isValidated, "alt value not matching");

}

@Test(priority = 2)

**public** **void** click\_on\_plans() **throws** InterruptedException {

Thread.*sleep*(1000);

homePage.Hover\_over\_Plans();

}

@Parameters({ "name", "mobile", "pin" })

@Test(priority = 3)

**public** **void** GetAQuoate(String name, String mobile, String pin) **throws** InterruptedException {

healthInsurancePage = **new** HealthInsurancePage(*driver*);

Thread.*sleep*(4000);

healthInsurancePage.Give\_Input\_And\_Click\_GetaQuote\_Button(name, mobile, pin);

}

@Parameters({ "email" })

@Test(priority = 4)

**public** **void** Input\_Authorize\_Form(String email) **throws** InterruptedException {

Thread.*sleep*(2000);

healthInsurancePage.Give\_Input\_in\_Authorize\_Form\_And\_Click\_Next(email);

}

@Test(priority = 5, dataProvider = "testdata")

**public** **void** Get\_Value(String name, String email) **throws** InterruptedException {

Thread.*sleep*(1000);

ArrayList<String> listValue = healthInsurancePage.Fetch\_The\_Value();

// System.out.println(listValue);

String ActualName = listValue.get(0);

String Actualemail = listValue.get(2);

Assert.*assertEquals*(email, Actualemail);

Assert.*assertEquals*(name, ActualName);

}

// @Test(priority = 6,dataProvider = "linkdata")

// public void Check\_social\_Media\_link(String fb,String youtube,String linkdln,String twitter,String instagram) throws InterruptedException {

//

// Thread.sleep(1000);

// ArrayList<String> links=healthInsurancePage.fetch\_Social\_media\_links();

// System.out.println(fb);

// System.out.println(youtube);

// System.out.println(linkdln);

// System.out.println(twitter);

// System.out.println(instagram);

//

// System.out.println(links);

//

// }

@Test(priority = 6)

**public** **void** clickontwitter() {

healthInsurancePage.clik\_on\_twitter();

}

@Test(priority = 7)

**public** **void** validatetwitter() {

String actual = healthInsurancePage.validate\_twitter();

String exp = "https://twitter.com/starhealthins?lang=en";

Assert.*assertEquals*(actual, exp);

}

@DataProvider(name = "testdata")

**public** Object[][] datasupplier() **throws** EncryptedDocumentException, IOException {

Object input[][] = XlS\_dataprovider.*getTestData*("Sheet1");

**return** input;

}

@DataProvider(name = "linkdata")

**public** Object[][] datasupplier2() **throws** EncryptedDocumentException, IOException {

Object input2[][] = XlS\_dataprovider2.*getTestData2*("Sheet1");

**return** input2;

}

}

**StarHealthPageTest2.java**

**package** testpages;

**import** java.util.List;

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebElement;

**import** org.testng.annotations.Test;

**import** base.BaseTest;

**public** **class** StarHealthPageTest2 **extends** BaseTest{

@Test

**public** **void** fetch\_All\_Link() **throws** InterruptedException {

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

Thread.*sleep*(2000);

List<WebElement> allLink=*driver*.findElements(By.*xpath*("//a"));

**for** (WebElement webElement : allLink) {

String links=webElement.getAttribute("href");

System.***out***.println(links);

}

}

}

**testng.xml**

<suite name=*"StarHealthTest"* verbose=*"1"* parallel=*"class"*>

<test name=*"starhealth001"*>

<parameter name=*"name"* value=*"Asish Patra"*/>

<parameter name=*"mobile"* value=*"7888943222"*/>

<parameter name=*"pin"* value=*"788877"*/>

<parameter name=*"email"* value=*"asish@gmail.com"*/>

<classes>

<class name=*"testpages.StarHealthPageTest"*>

</class>

</classes>

</test>

<test name=*"starhealth002"*>

<classes>

<class name=*"testpages.StarHealthPageTest2"*>

</class>

</classes>

</test>

</suite>

**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>Phase2-End-Project2</groupId>

<artifactId>Phase2-End-Project2</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>

<!-- 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>

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

<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>

</dependencies>

<build>

<pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->

<plugins>

<!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#clean\_Lifecycle -->

<plugin>

<artifactId>maven-clean-plugin</artifactId>

<version>3.1.0</version>

</plugin>

<!-- default lifecycle, jar packaging: see https://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin\_bindings\_for\_jar\_packaging -->

<plugin>

<artifactId>maven-resources-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.0</version>

</plugin>

<plugin>

<artifactId>maven-surefire-plugin</artifactId>

<version>2.22.1</version>

<configuration>

<suiteXmlFiles>

<!-- <suiteXmlFile>src/main/resources/TestRunner3.xml</suiteXmlFile> -->

<suiteXmlFile>src/test/resources/testng.xml</suiteXmlFile>

</suiteXmlFiles>

</configuration>

</plugin>

<plugin>

<artifactId>maven-jar-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-install-plugin</artifactId>

<version>2.5.2</version>

</plugin>

<plugin>

<artifactId>maven-deploy-plugin</artifactId>

<version>2.8.2</version>

</plugin>

<!-- site lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#site\_Lifecycle -->

<plugin>

<artifactId>maven-site-plugin</artifactId>

<version>3.7.1</version>

</plugin>

<plugin>

<artifactId>maven-project-info-reports-plugin</artifactId>

<version>3.0.0</version>

</plugin>

</plugins>

</pluginManagement>

</build>

</project>