Java/Maven Selenium Test Automation Podium

**Author: Israt Uddin**

DOCUMENT HISTORY

|  |  |  |
| --- | --- | --- |
| **Version** | **Author** | **Date** |
| 1.0 | Israt Uddin | 8/30/2020 |
|  |  |  |
|  |  |  |
|  |  |  |

**1 Introduction**

The Objective of this document is to provide the outline of the Testing Selenium Automation Framework. This document also portrays the work flow of the approach adopted in creating this automating project.

**1.1 Intended Audience**

The intended audience includes Cameron Radford and other who are viewing this document. Based on the information provided in the document, the user can run the automation scripts locally from the command line.

**2 Automation Framework**

**2.1 Framework Overview**

Selenium is an open source powerful tool that is used for test automation. Selenium helps to automate only the web applications. Currently it is the best automation tool that suite your needs. Selenium is a set of different software tools each with a different approach to supporting test automation.

The Automation Framework described in this document is for the applications of Podium. Below is the list of tools that will be used to design the automation framework.

* 1. **Environment Specifications**
* Selenium WebDriver (Supports all major browsers such as Mozilla FF, Chrome, Safari, and IE)
* Intellij IDEA (Java integrated development environment)
* Java (Programming Language)
* TestNG (Controls test cases)
* Maven (Manages project)
* Github (Control version)

**3 Framework Setup**

There are many different ways to setup a framework. In this framework will use Maven as a project management tool and TestNG to control test cases. To avoid any error follow below the exact steps to complete the framework.

**3.1 Creating Maven Project**

The instructions below refer to create a new Maven project in which will construct the whole project structure. Follow the steps below respectively to accomplish this task.

1. Run Intellij IDEA by double click on Intellij IDEA icon from desktop (assuming Intellij IDEA already in the system)
2. Select Create New Project.
3. Click on Maven and on the right sight of the wizard click on New button.
4. Select newly installed Java jdk and click on OK button.

On the wizard write “com.organization” as GroupId, “podium” as ArtifactId, and click on Next button.

1. Write “Podium” as Project name, and click on Finish button.
2. Click on “Enable Auto-Import”.

**3.2 Dependency Management**

The instructions below refer to setup the dependency management that will keep updating the project from the Maven remote repository. Follow the steps below respectively to accomplish this task.

1. Double click on pom.xml and add the following code right after version closing tag.

<dependencyManagement>  
 <dependencies>  
   
 </dependencies>  
 </dependencyManagement>

1. Visit the following MVN Repository page in any browser (e.g. Microsoft Internet Explorer).

URL: http://mvnrepository.com/

1. Enter “Selenium Server” on Search field and click on Search button.
2. Click on Selenium Server link.
3. Click on latest version link.
4. Copy the dependency code and paste it under dependencies in pom.xml editor.
5. Follow Step 2 to 6 for “Selenium Client”, “Selenium Java”, “Poi-ooxml”, “common-io”, and “TestNG”.

**3.3 Module Setup**

The instructions below refer to create a generic module that will manage dependencies of all applications. This generic module will be directly connected with root pom. Follow the steps below respectively to accomplish this task.

1. Create a Module by right click on Podium -> New, and click on Module.
2. Select Maven and click on Next.
3. Enter “Generic” on ArtifactId field and click on Next.
4. Enter “Generic” and click on Finish.
5. Follow Step 1 to 4 to create “SubModule” Module.

**3.4 Base File Setup**

The instructions below refer to create a base file under generic module in which we will store the common functionalities of all applications. Follow the steps below respectively to accomplish this task.

1. Expand Podium -> Generic -> src -> main ->java.
2. Right click on java -> New and select Java Class.
3. Enter “Base” as class name and click on OK.
4. Enter following code on Base.java text editor.
5. You should see something similar to the screenshot below (figure 2.12).

**import org.apache.poi.hssf.util.CellReference;  
import org.apache.poi.ss.usermodel.\*;  
import org.apache.poi.xssf.usermodel.XSSFWorkbook;  
import java.io.FileInputStream;  
  
  
  
import org.openqa.selenium.OutputType;  
import org.openqa.selenium.TakesScreenshot;  
import java.io.File;  
import java.text.DateFormat;  
import java.text.SimpleDateFormat;  
import java.util.Date;  
  
/\*\*  
 \* Author: Israt Uddin  
 \*/  
public class Base {  
 public WebDriver driver = null;  
 public Logger log = Logger.getLogger(Base.class.getName());  
  
  
  
 // File Path Variables  
 public static String testDataFilePath = "";  
 //  
  
 public static String chromeDriverPath = "C:\\ podium\\Generic\\src\\main\\resources\\drivers\\chromedriver.exe";  
 public static String screenshotPath = "";  
  
  
 @Parameters({"useSauceLab", "userName", "key", "appUrl", "os", "browserName", "browserVersion"})  
 @BeforeMethod  
 public void setUp(boolean useSauceLab, String userName, String key, String appUrl, String os, String browserName, String browserVersion) throws IOException {  
 if (useSauceLab == true) {  
 getSauceLabDriver(userName, key, os, browserName, browserVersion);  
 } else {  
 getLocalDriver(os, browserName);  
 }  
 driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);  
 driver.navigate().to(appUrl);  
 driver.manage().window().maximize();  
 log.info("browser loaded with App");  
 }  
  
 @AfterMethod  
 public void cleanUp() throws InterruptedException {  
 sleepfor(2);  
  
 log.info("driver is quiting");  
 sleepfor(2);  
 driver.quit();  
 }  
  
 //get local driver  
 public WebDriver getLocalDriver(String os, String browserName) {  
 if (browserName.equalsIgnoreCase("firefox")) {  
  
 driver = new FirefoxDriver();  
 } else if (browserName.equalsIgnoreCase("chrome")) {  
 if (os.equalsIgnoreCase("windows")) {  
 System.setProperty("webdriver.chrome.driver", chromeDriverPath);  
  
 } else {  
 System.setProperty("webdriver.chrome.driver", chromeDriverPath);  
 }  
 driver = new ChromeDriver();  
 } else if (browserName.equalsIgnoreCase("safari")) {  
 driver = new SafariDriver();  
 } else if (browserName.equalsIgnoreCase("ie")) {  
 System.setProperty("webdriver.ie.driver", "Generic\\selenium-browser- driver\\IEDriverServer.exe");  
 driver = new InternetExplorerDriver();  
 } else if (browserName.equalsIgnoreCase("htmlunit")) {  
 //driver = new HtmlUnitDriver();  
 }  
 return driver;  
 }  
  
 //get cloud driver  
 public WebDriver getSauceLabDriver(String userName, String key, String os, String browserName, String  
 browserVersion) throws IOException {  
 DesiredCapabilities cap = new DesiredCapabilities();  
 cap.setCapability("platform", os);  
 cap.setBrowserName(browserName);  
 cap.setCapability("version", browserVersion);  
 driver = new RemoteWebDriver(new URL("http://" + userName + ":" + key +  
 "@ondemand.saucelabs.com:80/wd/hub"), cap);  
 return driver;  
 }  
  
  
 public void sleepfor(int value) throws InterruptedException {  
 int initvalue = (value \* 1000);  
 Thread.sleep(initvalue);  
 }  
  
 public void clickByCss(String locator) {  
 driver.findElement(By.cssSelector(locator)).click();  
 }  
  
 public void clickByXpath(String locator) {  
 driver.findElement(By.xpath(locator)).click();  
 }  
  
 public void getTitle() {  
 driver.getTitle();  
 }  
  
 public void typeByCssThenEnter(String locator, String value) {  
 driver.findElement(By.cssSelector(locator)).sendKeys(value, Keys.ENTER);  
 }  
  
 public void typeByXpath(String s, String locator) {  
 driver.findElement(By.xpath(locator)).sendKeys(s);  
 }  
  
 public void typeByCss(String locator, String text) {  
 driver.findElement(By.cssSelector(locator)).sendKeys(text);  
 }  
  
  
 public void clickByLinkText(String linkText) {  
 driver.findElement(By.linkText(linkText)).click();  
 }  
  
//  
 public String getText(String locator) {  
 return (driver.findElement(By.xpath(locator)).getText());  
 }  
  
  
 public static String readFromExcel(String fileRef, String sheetRef, String cellRef) throws IOException {  
 FileInputStream fis = new FileInputStream(fileRef);  
 Workbook wb = new XSSFWorkbook(fis);  
 Sheet sheet = wb.getSheet(sheetRef);  
 DataFormatter formatter = new DataFormatter();  
 CellReference cellReference = new CellReference(cellRef);  
 Row row = sheet.getRow(cellReference.getRow());  
 Cell cell = row.getCell(cellReference.getCol());  
 String value = "";  
 if (cell != null) {  
 value = formatter.formatCellValue(cell);//cell.getStringCellValue();  
 }  
 return value;  
 }  
  
  
 public void takeTheScreenshot(String testCaseName) throws Exception {  
 String name = "";  
 System.out.println(name);  
  
 System.out.println("------------------------------");  
 Thread.sleep(3000);  
 String screenShotLocation = screenshotPath;  
 DateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");  
 Date date = new Date();  
 String currentTime = dateFormat.format(date); //2016/11/16 12:08:43  
 System.out.println("---------------------"+currentTime);  
  
 TakesScreenshot scrShot =((TakesScreenshot)driver);  
 File SrcFile = scrShot.getScreenshotAs(OutputType.FILE);  
 File DestFile = new File(screenShotLocation + testCaseName + "\_" +currentTime + ".png");  
 FileUtils.copyFile(SrcFile, DestFile);  
 }  
  
}**

Figure 2.12

**3.5 Sub Base File Setup**

The instructions below refer to create a sub base file for a single application in which will store the common functionalities of a single application. Follow the steps below respectively to accomplish this task.

1. Expand Podium -> SubModule -> src -> main ->java.
2. Right click on java -> New and select Java Class.
3. Enter “SubModuleBase” as class name and click on OK.
4. Enter following code on SubModuleBase.java text editor.
5. You should see something similar to the screenshot below

package api;  
  
import commonapi.Base;  
import org.openqa.selenium.By;  
import org.openqa.selenium.WebElement;  
import pageobject.AccountPage;  
import pageobject.HomePage;  
import pageobject.LoginPage;  
import pageobject.PricingPage;  
  
import java.io.IOException;  
import java.net.HttpURLConnection;  
import java.net.MalformedURLException;  
import java.net.URL;  
import java.util.Iterator;  
import java.util.List;  
  
public class SubModuleBase extends Base {  
  
 */\*\*  
 \* This testcase finds the valid and invalid links of home page  
 \* The first parameter is either tagname or link text. If it is for all the links tagName will be "a", or provide link text  
 \*The second parameter is the link name as to recognize where these link come from.  
 \*/* public void homePageLink(String locator, String linkName) throws InterruptedException {  
  
 String url = "";  
 HttpURLConnection huc = null;  
 int respCode = 200;  
 List<WebElement> links = null;  
  
 if(locator.contains("tag")){  
 links = driver.findElements(By.*tagName*(locator));  
 }  
  
 else{  
 links = driver.findElements(By.*linkText*(locator));  
 }  
  
  
  
 Iterator<WebElement> it = links.iterator();  
  
 int countBrokenLink=0;  
  
 while(it.hasNext()){  
  
 url = it.next().getAttribute("href");  
  
  
 if(url == null || url.isEmpty()){  
 System.*out*.println("URL is either not configured for anchor tag or it is empty");  
 continue;  
 }  
  
  
 try {  
 huc = (HttpURLConnection)(new URL(url).openConnection());  
  
 huc.setRequestMethod("HEAD");  
  
 huc.connect();  
  
 respCode = huc.getResponseCode();  
  
  
  
 if(respCode >= 400){  
 System.*out*.println(url+" is a broken link");  
  
 }  
 else{  
 System.*out*.println(url+" is a valid link");  
 }  
  
 } catch (MalformedURLException e) {  
 *// TODO Auto-generated catch block* e.printStackTrace();  
 } catch (IOException e) {  
 *// TODO Auto-generated catch block* e.printStackTrace();  
 }  
 }  
 System.*out*.println("These are the links from "+ linkName);  
 System.*out*.println("Broken Link: " + countBrokenLink );  
  
  
 }  
  
 */\*\*  
 \*  
 \* This method helps to log in  
 \*/* public void logIn(String emailORMobile, String password) throws InterruptedException {  
  
 clickByLinkText(LoginPage.*login\_Button\_LinkText*); *//Click in Login link* typeById(LoginPage.*emailMobile\_Field\_Id*,emailORMobile);*//Send email or phone input into Email or mobile field* clickById(LoginPage.*signIn\_Button\_Id*); *// Click Next button* typeById(LoginPage.*password\_Field\_Id*,password);*//Send password input into Password field* clickById(LoginPage.*signIn\_Button\_Id*); *//Click on Sign in button* }  
 */\*\*  
 \* This method allows to create an account  
 \*/* public void createAccount() throws Exception {  
  
 clickByLinkText(AccountPage.*getStartedFreeLink\_LinkText*); *//Click Get Started Link* typeById(AccountPage.*firstName\_Input\_Field\_Id*,*readFromExcel*(*testDataFilePath*,"DataSheet","E3")); *// First Name : Data Source Spreadsheet* typeById(AccountPage.*lastName\_Input\_Field\_Id*,"");*//data can be fetched from external source* typeById(AccountPage.*email\_Input\_Field\_Id*,"");*//data can be fetched from external source* typeById(AccountPage.*phoneNum\_Input\_Field\_Id*,"");*//data can be fetched from external source* typeById(AccountPage.*businessName\_Input\_Field\_Id*,"");*//data can be fetched from external source* clickById(AccountPage.*createAccount\_Button\_Id*);*//Click Create account button* }  
 public void priceQuote () throws Exception {  
  
 clickByLinkText(PricingPage.*pricing\_LinkText*);*// Click pricing link from footer* typeById(PricingPage.*firstName\_Field\_Id*,*readFromExcel*(*testDataFilePath*,"DataSheet","E3")); *//First Name field's input is fetched from Data Source Spreadsheet* typeById(PricingPage.*lastName\_Field\_Id*,"");  
 typeById(PricingPage.*email\_Field\_Id*,"");  
 typeById(PricingPage.*mobilePhone\_Field\_Id*,"");  
 typeById(PricingPage.*company\_Field\_Id*,"");  
 */\*\*  
 \* Click Get Pricing  
 \* clickByCss(PricingPage.getPricing\_Css\_Button);  
 \*/* }

**3.6 Generic POM Setup**

The instructions below refer to setup the generic pom which will be directly connected with room pom. Follow the steps below respectively to accomplish this task.

1. Expand Podium -> Generic -> pom.xml.
2. Double click on pox.xml.
3. Enter following code on pom.xml text editor right after artifactId. (Notice that these dependencies are similar as root pom except they don’t have the version number)

<**dependencies**>  
 <**dependency**>  
 <**groupId**>org.seleniumhq.selenium</**groupId**>  
 <**artifactId**>selenium-server</**artifactId**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>org.seleniumhq.selenium</**groupId**>  
 <**artifactId**>selenium-java</**artifactId**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>org.testng</**groupId**>  
 <**artifactId**>testng</**artifactId**>  
 </**dependency**>  
</**dependencies**>

Figure 2.14

**3.7 SubModule Application POM Setup**

The instructions below refer to setup the application pom which will be directly connected with generic pom. Follow the steps below respectively to accomplish this task.

1. Expand Podium -> SubModule -> pom.xml.
2. Double click on pox.xml.
3. Enter following code on pom.xml text editor right after artifactId.

*<?*xml version="1.0" encoding="UTF-8"*?>*<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">  
 <parent>  
 <artifactId>podium</artifactId>  
 <groupId>com.organization</groupId>  
 <version>1.0-SNAPSHOT</version>  
 </parent>  
 <modelVersion>4.0.0</modelVersion>  
  
 <artifactId>SubModule</artifactId>  
  
 <dependencies>  
 <dependency>  
 <groupId>com.organization</groupId>  
 <artifactId>Generic</artifactId>  
 <version>1.0-SNAPSHOT</version>  
 </dependency>  
 </dependencies>

**3.8 Test Runner XML File Setup**

The instructions below refer to create a test runner XML file from which we will control and run all our test cases. Follow the steps below respectively to accomplish this task.

1. Expand Podium -> SubModule-> src.
2. Right click on test -> New and select Derectory.
3. Enter “Resources” and click on Ok button.
4. Right click on Resources -> New and click on File.
5. Write “TestRunner.xml” as file name and click on Ok button.
6. Enter following code on TestRunner.xml text editor.

*<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">  
<suite name="Podium App is running on Chrome Browser">  
  
 <parameter name="useSauceLab" value="false"/>  
 <parameter name="userName" value="israt"/>  
 <parameter name="key" value="662b4ed7-1a59-4ab2-ae8c-2dfe1e32f4aa"/>  
 <parameter name="appUrl" value="https://www.podium.com/"/>  
 <parameter name="os" value="windows"/> <!-- change the os parameter in "mac" -->  
  
 <!-- change the browser name if you need to run on different browser -->  
 <parameter name="browserName" value="chrome"/>  
  
 <!--Please change the browser version as your system -->  
 <parameter name="browserVersion" value="84.0.4147.135"/>  
  
 <test name="Home Page" group-by-instances="true">  
 <classes>  
 <!--&lt;!&ndash;<class name="SignupTest"/>-->  
 <class name="Login"></class>  
 <class name="CreateAccount"></class>  
 <class name="Pricing"></class>  
 <class name="liveChatWindow"></class>  
 <class name="WatchDemoVideo"></class>  
 <class name="HomePageLinkVerify"></class>  
  
 </classes>  
 </test>  
  
  
</suite>*

**4 Script Execution**

**4.1 Web Element Detection**

The instructions below refer to inspect web elements such as XPath expressions, and CSS selectors. Follow the steps below respectively to accomplish this task.

1. Visit the following podium application login page in Chrome browser.

URL: <https://www.podium.com/>"

1. Right click on desired element
2. Click on Inspect button.
3. Click on Username field.
4. Copy the CSS selector from CSS field.
5. Follow the same steps to copy the CSS selector for Password field.
6. Follow the same process to locate locator using xpath, class name, or id

**4.2 Common Functions**

The instructions below refer to write all the common functionalities of a single application. Follow the steps below respectively to accomplish this task.

1. Expand Podium -> SubModule-> src -> main -> java -> SubModuleBase.
2. Double click on SubModuleBase and enter following code on SubModuleBase.java text editor.
3. You should see something similar to the screenshot below (figure 2.18).

**pckage api;  
  
import commonapi.Base;  
import org.openqa.selenium.By;  
import org.openqa.selenium.WebElement;  
import pageobject.AccountPage;  
import pageobject.HomePage;  
import pageobject.LoginPage;  
import pageobject.PricingPage;  
  
import java.io.IOException;  
import java.net.HttpURLConnection;  
import java.net.MalformedURLException;  
import java.net.URL;  
import java.util.Iterator;  
import java.util.List;  
  
public class SubModuleBase extends Base {  
  
 /\*\*  
 \* This testcase finds the valid and invalid links of home page  
 \* The first parameter is either tagname or link text. If it is for all the links tagName will be "a", or provide link text  
 \*The second parameter is the link name as to recognize where these link come from.  
 \*/  
  
 public void homePageLink(String locator, String linkName) throws InterruptedException {  
  
 String url = "";  
 HttpURLConnection huc = null;  
 int respCode = 200;  
 List<WebElement> links = null;  
  
 if(locator.contains("tag")){  
 links = driver.findElements(By.tagName(locator));  
 }  
  
 else{  
 links = driver.findElements(By.linkText(locator));  
 }  
  
  
  
 Iterator<WebElement> it = links.iterator();  
  
 int countBrokenLink=0;  
  
 while(it.hasNext()){  
  
 url = it.next().getAttribute("href");  
  
  
 if(url == null || url.isEmpty()){  
 System.out.println("URL is either not configured for anchor tag or it is empty");  
 continue;  
 }  
  
  
 try {  
 huc = (HttpURLConnection)(new URL(url).openConnection());  
  
 huc.setRequestMethod("HEAD");  
  
 huc.connect();  
  
 respCode = huc.getResponseCode();  
  
  
  
 if(respCode >= 400){  
 System.out.println(url+" is a broken link");  
  
 }  
 else{  
 System.out.println(url+" is a valid link");  
 }  
  
 } catch (MalformedURLException e) {  
 // TODO Auto-generated catch block  
 e.printStackTrace();  
 } catch (IOException e) {  
 // TODO Auto-generated catch block  
 e.printStackTrace();  
 }  
 }  
 System.out.println("These are the links from "+ linkName);  
 System.out.println("Broken Link: " + countBrokenLink );**

|  |
| --- |
|  |

**}  
  
 /\*\*  
 \*  
 \* This method helps to log in  
 \*/  
  
 public void logIn(String emailORMobile, String password) throws InterruptedException {  
  
 clickByLinkText(LoginPage.login\_Button\_LinkText); //Click in Login link  
 typeById(LoginPage.emailMobile\_Field\_Id,emailORMobile);//Send email or phone input into Email or mobile field  
 clickById(LoginPage.signIn\_Button\_Id); // Click Next button  
 typeById(LoginPage.password\_Field\_Id,password);//Send password input into Password field  
 clickById(LoginPage.signIn\_Button\_Id); //Click on Sign in button  
  
  
 }  
 /\*\*  
 \* This method allows to create an account  
 \*/  
  
 public void createAccount() throws Exception {  
  
 clickByLinkText(AccountPage.getStartedFreeLink\_LinkText); //Click Get Started Link  
 typeById(AccountPage.firstName\_Input\_Field\_Id,readFromExcel(testDataFilePath,"DataSheet","E3")); // First Name : Data Source Spreadsheet  
 typeById(AccountPage.lastName\_Input\_Field\_Id,"");//data can be fetched from external source  
 typeById(AccountPage.email\_Input\_Field\_Id,"");//data can be fetched from external source  
 typeById(AccountPage.phoneNum\_Input\_Field\_Id,"");//data can be fetched from external source  
 typeById(AccountPage.businessName\_Input\_Field\_Id,"");//data can be fetched from external source  
 clickById(AccountPage.createAccount\_Button\_Id);//Click Create account button  
  
 }  
 public void priceQuote () throws Exception {  
  
 clickByLinkText(PricingPage.pricing\_LinkText);// Click pricing link from footer  
 typeById(PricingPage.firstName\_Field\_Id,readFromExcel(testDataFilePath,"DataSheet","E3")); //First Name field's input is fetched from Data Source Spreadsheet  
 typeById(PricingPage.lastName\_Field\_Id,"");  
 typeById(PricingPage.email\_Field\_Id,"");  
 typeById(PricingPage.mobilePhone\_Field\_Id,"");  
 typeById(PricingPage.company\_Field\_Id,"");**

**4.3 Write Your First Script**

The instructions below refer to write your first Selenium test script. Follow the steps below respectively to accomplish this task.

1. Expand Podium -> SubModule -> src -> test -> java.
2. Right click on java -> New and select Java Class.
3. Enter logIn as file name and click on Ok button.

**4.4 Maven Lifecycle**

The instructions below refer to execute all phases of the maven lifecycle. Follow the steps below respectively to accomplish this task.

1. On IntelliJ IDEA toolbar click on View -> Tool Windows and select Maven Projects. (Maven Projects appears on the right side of IDE)
2. From Maven Projects expand Podium -> Lifecycle and click on test.
3. Click on “Toggle ‘Skip tests’ Mode” button.
4. Expand Podium -> Lifecycle and click on install.
5. Click on “Run Maven Build” button.
6. Repeat Step 4 and Step 5 for both Generic and SubModule modules.

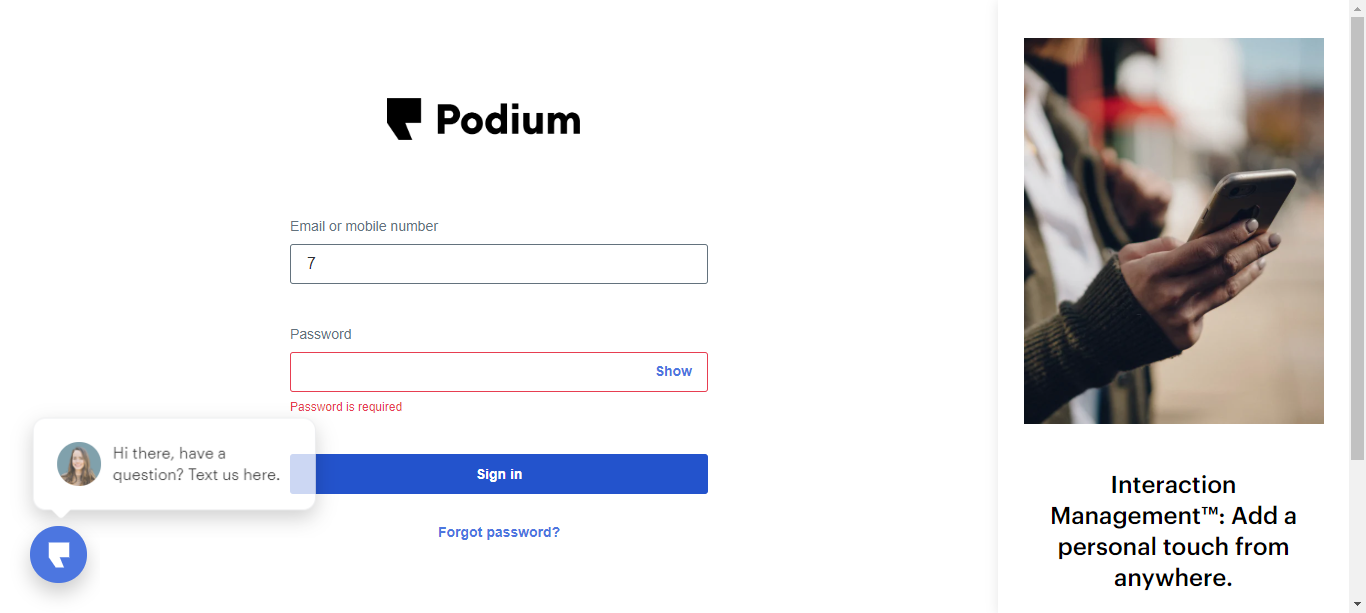
**5.5 Execute Test Case**

The instructions below refer to execute our first Selenium test script. Follow the steps below respectively to accomplish this task.

1. Expand Podium -> SubModule -> SRC -> test -> Resources -> TestRunner.
2. Right click on TestRunner and select Run.

**5.6 Test Result**

Congratulation!!! You have written and executed your first Selenium Test Script. After execution you should see something similar to the screenshot below (figure 2.22).



**6 Conclusion**

1. In this tutorial, I try to make you acquainted with the Selenium Automation Framework, description of required tools and their installation process, creating your own framework, and finally how to write and execute Selenium test scripts. Thank you