



[ASSIGNMENT 2] [STP304X_01 KIỂM THỬ TỰ ĐỘNG]

Họ và tên : Phạm Hải Thanh
MSV : FX20171

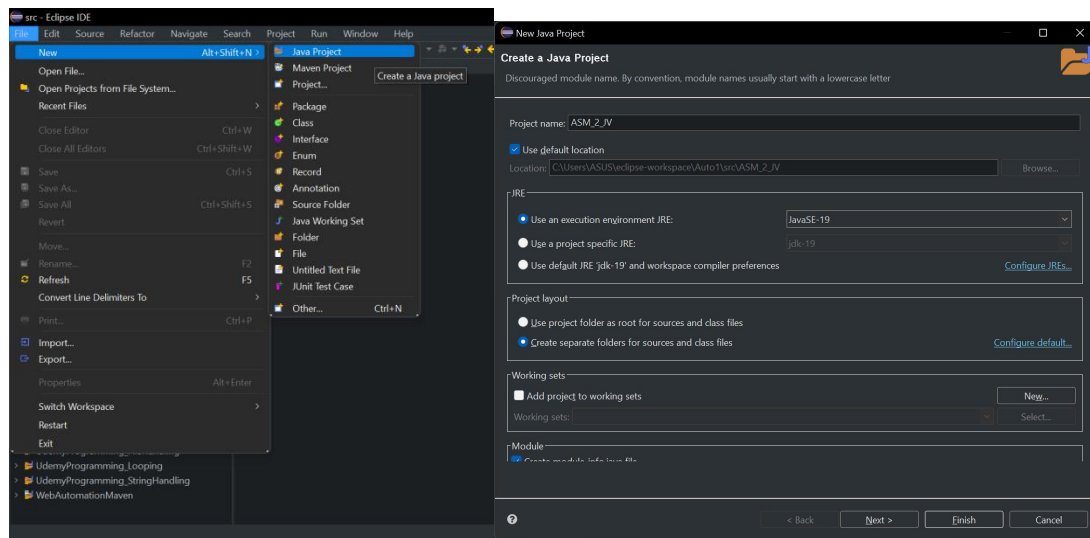
TP HCM.2023

MỤC LỤC

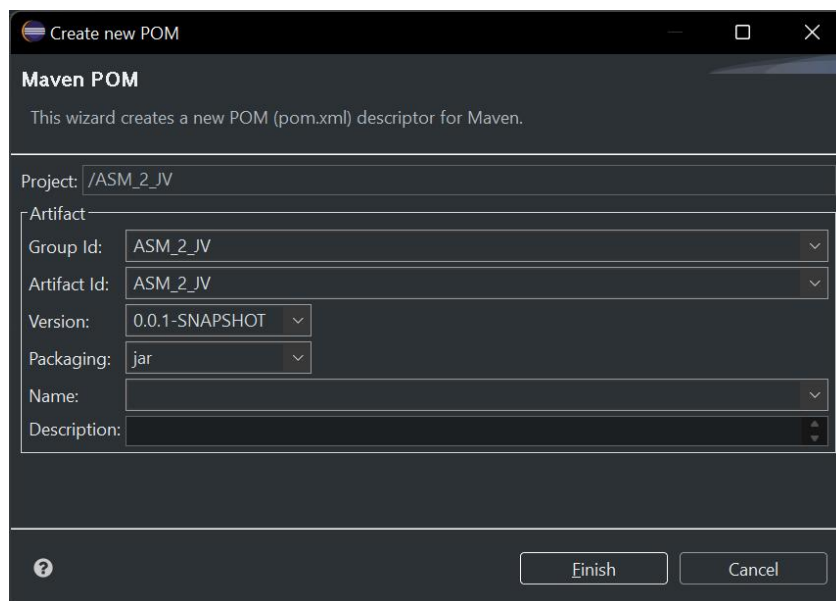
PHẦN 1: TẠO PROJECT, PACKAGE CHO ASM2	3
PHẦN 2: TẠO CÁC THƯ MỤC VÀ VIẾT CÁC CÀI ĐẶT BỔ TRỢ	6
2.1 Tạo folder ConfigurationFile	6
2.2 Tạo folder Data	7
2.3 Tạo folder Driver	7
2.4 Tạo folder tên Screenshot	7
PHẦN 3: TẠO CÁC PACKAGES VÀ CLASSES TRONG PROJECT	8
3.1 Tạo package com.automation.base	8
3.2 Tạo package com.automation.utils	9
3.2.1 Class CaptureScreenshot.java	9
3.2.2 Class Utility.java	10
3.2.3 Class DataGenerators .java	11
3.3 Tạo package com.automation.pom	11
3.3.1 PageActions.java	12
3.3.2 LoginPage.java	13
3.3.3 LogoutPage.java	14
3.4 Tạo package com.automation.testcase	15
PHẦN 4: KẾT QUẢ CHẠY CHƯƠNG TRÌNH	16
4.1 Trường hợp chạy thành công với email, pass đúng	16
4.2 Trường hợp báo lỗi testcase fail với email, pass sai	17

PHẦN 1: TẠO PROJECT, PACKAGE CHO ASM2

- B1: Chọn dạng dự án: Java Project



- B2: Convert sang Maven



- B3: Truy cập <https://mvnrepository.com/> để cài đặt các Dependencies vào pom.xml của project để sử dụng

1. Cài poi và poi-ooxml
<pre><dependency> <groupId>org.apache.poi</groupId> <artifactId>poi</artifactId> <version>5.2.3</version> </dependency></pre>
<pre><dependency> <groupId>org.apache.poi</groupId> <artifactId>poi-ooxml</artifactId> <version>5.2.3</version> </dependency></pre>

2. Cài testng và selenium

```
<dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>3.141.59</version>
</dependency>
```

```
<dependency>
    <groupId>org.testng</groupId>
    <artifactId>testng</artifactId>
    <version>6.14.3</version>
    <scope>test</scope>
</dependency>
```

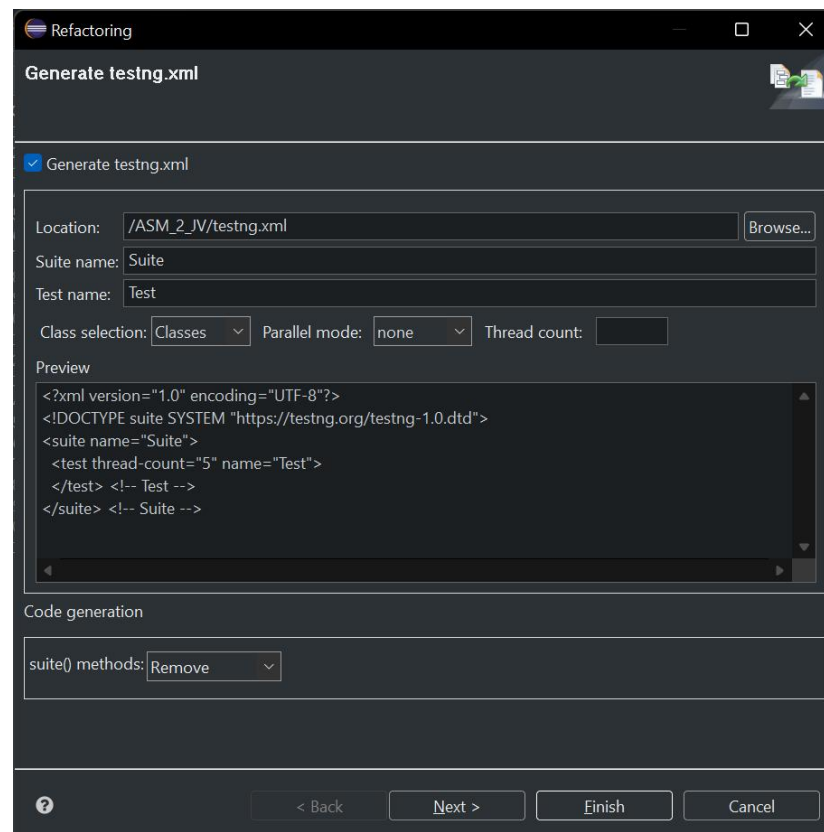
3. Cài common-io, reportng, guice để phục vụ báo cáo html nâng cao

```
<dependency>
    <groupId>commons-io</groupId>
    <artifactId>commons-io</artifactId>
    <version>2.11.0</version>
</dependency>
```

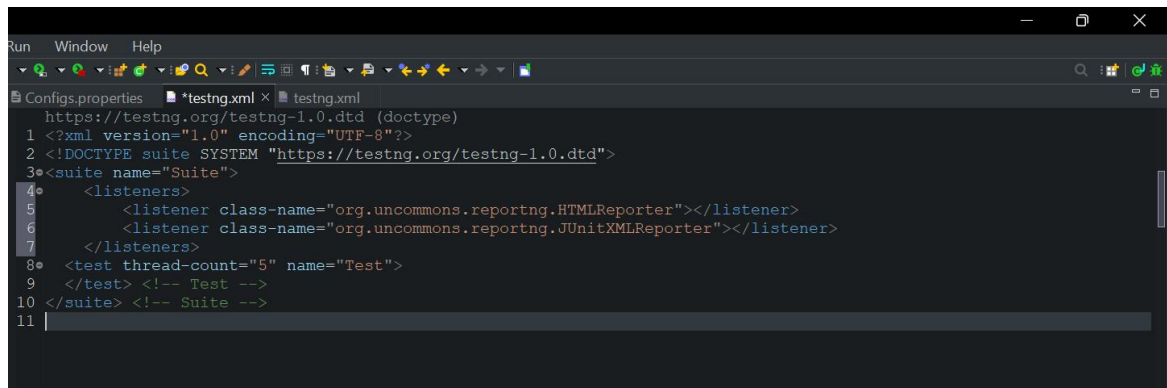
```
<dependency>
    <groupId>org.uncommons</groupId>
    <artifactId>reportng</artifactId>
    <version>1.1.4</version>
</dependency>
```

```
<dependency>
    <groupId>com.google.inject</groupId>
    <artifactId>guice</artifactId>
    <version>5.1.0</version>
</dependency>
```

- B4: Convert sang TestNG



- B5: thêm đoạn code listener vào testng.xml , dùng để sử dụng với báo cáo HTML report sau đó



```
Run Window Help
testng.xml x testng.xml
https://testng.org/testng-1.0.dtd (doctype)
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">
3 <suite name="Suite">
4   <listeners>
5     <listener class-name="org.uncommons.reporting.HTMLReporter"></listener>
6     <listener class-name="org.uncommons.reporting.JUnitXMLReporter"></listener>
7   </listeners>
8   <test thread-count="5" name="Test">
9     </test> <!-- Test -->
10 </suite> <!-- Suite -->
11 |
```

PHẦN 2: TẠO CÁC THƯ MỤC VÀ VIẾT CÁC CÀI ĐẶT BỔ TRỢ

Tạo các thư mục chứa các file ,drive bổ trợ

Tên folder	File con	Giải thích
ConfigurationFile	Configs.properties	Chứa các thông tin như đường link web cần test, loại web trình duyệt được sử dụng
	Elements.properties	Chứa các thông tin name/id/xpath của username, password, button, dropbox... cần kiểm
Data	assignment2_data_test.xlsx	Chứa dữ liệu file excel đề bài cho
Driver	chromedriver.exe geckodriver.exe	Chứa cài đặt trình duyệt Chrome /Firefox
Screenshots		Chứa các ảnh dạng png được capture từ trang web kiểm thử khi testcase fail

2.1 Tạo folder ConfigurationFile

Xác định và thêm thông tin vào Configs.properties và Elements.properties

- Configs.properties chứa các thông tin trình duyệt, URL page

#Maintaining Application Configuration Data

Application_URL =https://automationexercise.com/products
BrowserName =Chrome

- Elements.properties chứa các thông tin về xpath/name/id/class của các textbox, nút, dropbox hay tickbox trong trang cần kiểm

#ICON

icon_signin_xpath = //a[text()=' Signup / Login']
icon_signout_xpath = //a[text()=' Logout']

#Input information to Login

login_email_xpath = //input[@data-qa='login-email']
login_password_xpath = //input[@data-qa='login-password']
login_button_xpath = //button[@data-qa='login-button']

2.2 Tạo folder Data

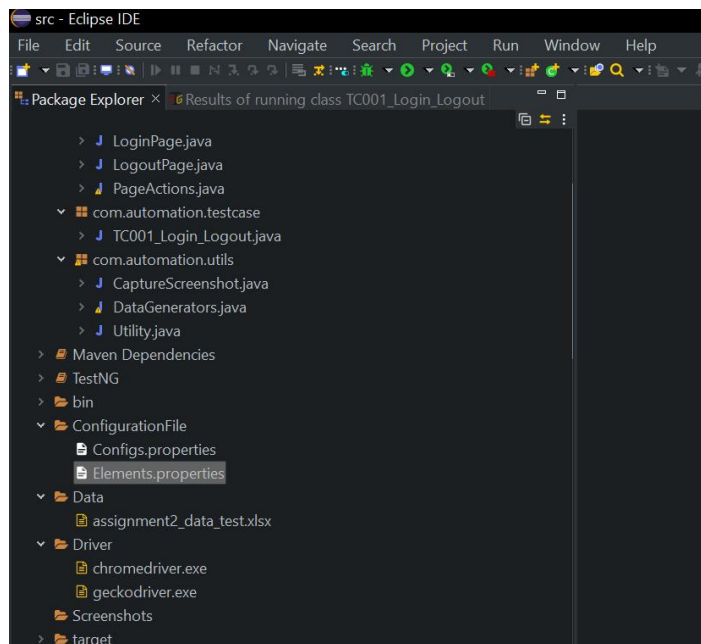
Tạo 1 folder tên là Data, tải file excel assignment2_data_test.xlsx, di chuyển vào folder Data. Folder dùng để đọc và lấy data password/email điền vào trang login

2.3 Tạo folder Driver

Tạo file chứa Driver để chạy trình duyệt Chrome hoặc Firefox

2.4 Tạo folder tên Screenshot

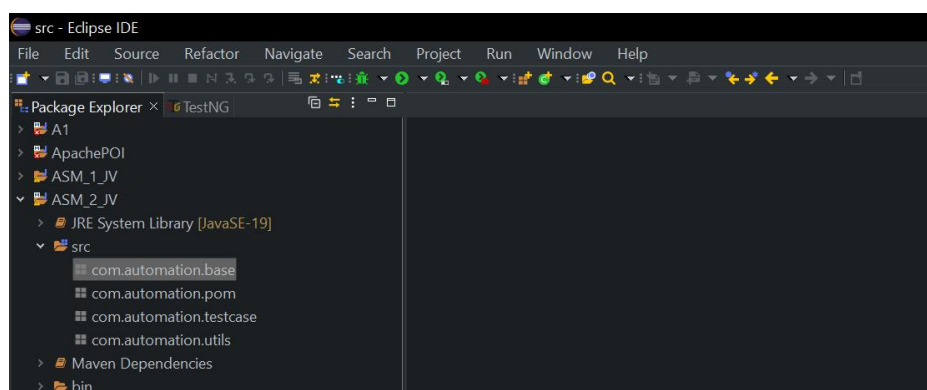
Folder để lưu ảnh chụp màn hình khi testcase fail



PHẦN 3: TẠO CÁC PACKAGES VÀ CLASSES TRONG PROJECT

Tạo 4 packages trong file main ASM_2_JV:

Tên packages	Mục đích	Các class con
com.automation.base	chứa các base class	DriverInstance.java
com.automation.untils	chứa các class hỗ trợ chụp ảnh màn hình, hỗ trợ đọc properties file ...	CaptureScreenshot.java DataGenerators.java Utility.java
com.automation.pom	chứa các class thực thi kiến trúc POM.	LoginPage.java LogoutPage.java PageActions.java
com.automation.testcase	chứa các class test sử dụng TestNG	TC001_Login_Logout.java



3.1 Tạo package com.automation.base

- Mục đích: chứa class DriverInstance.java. DriverInstance là class để cài đặt sẵn lệnh chạy browser, url tự động, lệnh thoát trình duyệt, cho phép người lập lại để sử dụng trong nhiều testcases mà không phải viết lại code.

- Trong DriverInstance sử dụng annotation @BeforeMethod và @AfterMethod

```
package com.automation.base;

import java.io.IOException;
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.firefox.FirefoxDriver;
import org.openqa.selenium.ie.InternetExplorerDriver;
import org.testng.annotations.AfterMethod;
import org.testng.annotations.BeforeMethod;
import com.automation.untils.Utility;

public class DriverInstance {
    public WebDriver driver;
    //Khởi tạo trình duyệt, chạy URL cần kiểm
    @BeforeMethod
    public void initiateDriverInstance() throws IOException
    {
        if(Utility.fetchPropertyValue("BrowserName").toString().equalsIgnoreCase("chrome"))
        {
            System.setProperty("webdriver.chrome.driver", "./Driver/chromedriver.exe");
        }
    }
}
```



```

        driver = new ChromeDriver();
    }
    else if (Utility.fetchPropertyValue("BrowserName").toString().equalsIgnoreCase("firefox"))
    {
        System.setProperty("webdriver.gecko.driver", "./Driver/geckodriver.exe");
        driver = new FirefoxDriver();
    }
    else if (Utility.fetchPropertyValue("BrowserName").toString().equalsIgnoreCase("ie"))
    {
        System.setProperty("webdriver.ie.driver", "./Driver/internetexplorerdriver.exe");
        driver = new InternetExplorerDriver();
    }
    else
    {
        System.setProperty("webdriver.chrome.driver", "./Driver/chromedriver.exe");
        driver = new ChromeDriver();
    }
//Phóng to cửa sổ trình duyệt, chạy URL
    driver.manage().window().maximize();
    driver.get(Utility.fetchPropertyValue("Application_URL").toString());
    driver.manage().timeouts().pageLoadTimeout(15, TimeUnit.SECONDS);
}

//Thoát URL

    @AfterMethod
    public void closeDriverInstance()
    {
        driver.quit();
    }
}

```

3.2 Tạo package com.automation.utils

Mục đích: chứa các class hỗ trợ chụp ảnh màn hình, hỗ trợ đọc properties file ...

3.2.1 Class CaptureScreenshot.java

- Class CaptureScreenshot.java: cho phép lưu các hình ảnh về test fail đã chụp từ màn hình vào folder Screenshots

```

package com.automation.utils;

import java.io.File;
import java.util.Calendar;
import org.apache.commons.io.FileUtils;
import org.openqa.selenium.OutputType;
import org.openqa.selenium.TakesScreenshot;
import org.openqa.selenium.WebDriver;
import org.testng.Reporter;

public class CaptureScreenshot {

    static String filename = null;
    // Chụp màn hình
    public static void takeScreenshot(WebDriver driver, String name) {
        try {

```

```

        Calendar c = Calendar.getInstance();
        System.out.print(c.getTime());
        TakesScreenshot screenshot = (TakesScreenshot) driver;
        filename = "./Screenshots/"+name+c.getTime().toString().replace(":",
"".replace(" ", ""))+".png";
        File f=screenshot.getScreenshotAs(OutputType.FILE);
        File fd = new File(filename);
        FileUtils.copyFile(f,fd);
    } catch (Exception e) {
        System.out.println("Not able to take screenshot");
    }
    attachScreenshot();
}
//Lưu màn hình vào folder Screenshots
public static void attachScreenshot() {
    try {
        System.setProperty("org.uncommons.reportng.escape-output", "false");
        File f = new File(filename);
        Reporter.log
        ("<a title = \"Snapshot\"href=\""+f.getAbsolutePath() +
        "\">" + "img width=\"418\" height=\"240\"
alt=\"alternativeName\" title =\"title\" src =\"../surefire-reports/html/screenshot/" +
        filename+"\"</a>");

    } catch (Exception e) {
        System.out.println("Not able to take screenshot");
    }
}
}

```

3.2.2 Class Utility.java

- Class Utility.java giúp đọc các giá trị từ file Config.properties và Elements.properties trong folder ConfigurationFile.

```

package com.automation.utils;

import java.io.FileInputStream;
import java.io.IOException;
import java.util.Properties;

public class Utility {
    // Đọc file từ Configs.properties
    public static Object fetchPropertyValue(String key) throws IOException
    {
        FileInputStream file = new FileInputStream("./ConfigurationFile/Configs.properties");
        Properties property = new Properties();
        property.load(file);
        return property.get(key);
    }

    // Đọc file từ Elements.properties
    public static String fetchLocatorValue(String key) throws IOException
    {
        FileInputStream file = new FileInputStream("./ConfigurationFile/Elements.properties");
    }
}

```

```

        Properties property = new Properties();
        property.load(file);
        return property.get(key).toString();
    }
}

```

3.2.3 Class DataGenerators .java

- Class DataGenerators .java giúp đọc các giá trị từ file excel cho trước. File excel gồm 1 sheet Login với 2 cột gồm email, pass dùng để đăng nhập vào trang kiểm thử.
- Có 7 giá trị cần kiểm thử tương ứng 7 dòng. Sau khi đọc xong các giá trị DataGenerators sẽ ghi lại và gán giá trị Object bằng từng cặp giá trị Object = ['email1', 'pass1'), Object = ['email2', 'pass2')... cho tới khi gán hết giá trị ở 7 dòng.

```

package com.automation.utils;

import java.io.FileInputStream;
import java.io.IOException;
import org.apache.poi.xssf.usermodel.XSSFCell;
import org.apache.poi.xssf.usermodel.XSSFRow;
import org.apache.poi.xssf.usermodel.XSSFSheet;
import org.apache.poi.xssf.usermodel.XSSFWorkbook;
import org.testng.annotations.DataProvider;

public class DataGenerators {

    //Đọc và lấy các thông tin từ 1 sheet trong file excel
    @DataProvider(name="Login")
    public static Object[][] testDataGeneratorRegister() throws IOException
    {
        FileInputStream file = new FileInputStream("./Data/assignment2_data_test.xlsx");
        XSSFWorkbook workbook = new XSSFWorkbook(file);
        XSSFSheet loginSheet = workbook.getSheet("Login");
        int numberOfData = loginSheet.getPhysicalNumberOfRows();
        Object [][] testData = new Object[numberOfData][2];

        // Tạo vòng lặp để đọc các giá trị ở từng cột trong i dòng
        for(int i = 0; i<numberOfData;i++)
        {
            XSSFRow row = loginSheet.getRow(i);
            XSSFCell email_address = row.getCell(0);
            XSSFCell password = row.getCell(1);

            // Lấy data ở 2 cột ở mỗi hàng, gán vào object, mỗi object gồm 2 phần tử email, pass
            testData[i][0] = email_address.getStringCellValue();
            testData[i][1] = password.getStringCellValue();
        }
        return testData;
    }
}

```

3.3 Tạo package com.automation.pom

- Package chứa các class thực thi kiến trúc POM như class tạo các ActionPage gồm actions như nhập text, click button, chọn dropbox..., dùng để sử dụng chung cho nhiều trang nếu phải test nhiều trang cùng 1 lúc. Mỗi một trang test sẽ được viết sẵn các tác vụ như truyền dữ liệu, click button ..

3.3.1 PageActions.java

- Trong class PageActions sẽ liệt kê các action cần kiểm của 1 trang bất kì, ở tình huống này chỉ có 2 action là nhập data vào ô trống và ấn click (button/icon)

```
package com.automation.pom;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.Select;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.Assert;
import com.automation.utils.CaptureScreenshot;

public class PageActions {
    WebDriver driver = null;
    WebDriverWait wait;
    public PageActions(WebDriver driver) {
        this.driver = driver;
        wait = new WebDriverWait(driver,20);
    }
    //ACTION CLICK BUTTON
    public void clickButton(WebElement element) {
        try {
            wait.until(ExpectedConditions.elementToBeClickable(element));
            element.click();
        } catch (Exception e) {
            System.out.println("*****Exception*****");
            e.printStackTrace();
            CaptureScreenshot.takeScreenshot(driver, "clickButtonFail");
            Assert.fail();
        }
    }
    //ACTION ĐIỀN THÔNG TIN VÀO Ô TEXT
    public void enterDataIntoText(WebElement element, String text) {
        try {
            wait.until(ExpectedConditions.visibilityOf(element));
            element.sendKeys(text);
        } catch (Exception e) {
            System.out.println("*****Exception*****");
            e.printStackTrace();
            CaptureScreenshot.takeScreenshot(driver, "enterDataFail");
            Assert.fail();
        }
    }
}
```

3.3.2 LoginPage.java

- Trong class LoginPage sẽ liệt kê các action cần kiểm tra của trang này gồm nhấn nút Signup/Login, điền email/pass, ấn Login

```
package com.automation.pom;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import com.automation.utils.CaptureScreenshot;
import com.automation.utils.Utility;
import org.testng.Assert;

public class LoginPage {
    WebDriver driver = null;
    PageActions action;
    public LoginPage(WebDriver driver) {
        this.driver = driver;
        action = new PageActions(driver);
    }

    // NHẤN ICON SINGUP/LOGIN ĐỂ CHUYỂN ĐẾN TRANG LOGIN
    public void clickSignupIcon() {
        try {
            action.clickButton(driver.findElement(By.xpath(Utility.fetchLocatorValue("icon_signin_xpath"))));
        } catch (Exception e) {
            System.out.println("*****Exception*****");
            e.printStackTrace();
            CaptureScreenshot.takeScreenshot(driver, "clickButtonFail");
            Assert.fail();
        }
    }

    // ĐIỀN EMAIL
    public void enterEmailAddress(String email) {
        try {
            action.enterDataIntoText(driver.findElement(By.xpath(Utility.fetchLocatorValue("login_email_xpath"))),email);
        } catch (Exception e) {
            System.out.println("*****Exception*****");
            e.printStackTrace();
            CaptureScreenshot.takeScreenshot(driver, "EnterDataFail");
            Assert.fail();
        }
    }

    // ĐIỀN PASSWORD
    public void enterPassword(String password){
        try {
            action.enterDataIntoText(driver.findElement(By.xpath(Utility.fetchLocatorValue("login_password_xpath"))),password);
        } catch (Exception e) {
            System.out.println("*****Exception*****");
        }
    }
}
```

```

        e.printStackTrace();
        CaptureScreenshot.takeScreenshot(driver, "EnterDataFail");
        Assert.fail();
    }
}

// NHẤN NÚT SINGUP
public void clickLoginButton() {
    try {
        action.clickButton(driver.findElement(By.xpath(Utility.fetchLocatorValue("login_button_xpath"))));
    }
    catch (Exception e) {
        System.out.println("*****Exception*****");
        e.printStackTrace();
        CaptureScreenshot.takeScreenshot(driver, "clickButtonFail");
        Assert.fail();
    }
}
}

```

3.3.3 LogoutPage.java

- Trong class LoginPage sẽ liệt kê action nhấn nút Logout cần kiểm của trang

```

package com.automation.pom;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.testng.Assert;
import com.automation.utils.CaptureScreenshot;
import com.automation.utils.Utility;

public class LogoutPage {
    WebDriver driver = null;
    PageActions action;
    public LogoutPage(WebDriver driver) {
        this.driver = driver;
        action = new PageActions(driver);
    }

// NHẤN NÚT LOGOUT
    public void clickLogoutButton() {
        try {
            action.clickButton(driver.findElement(By.xpath(Utility.fetchLocatorValue("icon_signout_xpath"))));
        }
        catch (Exception e) {
            System.out.println("*****Exception*****");
            e.printStackTrace();
            CaptureScreenshot.takeScreenshot(driver, "clickButtonFail");
            Assert.fail();
        }
    }
}

```

3.4 Tạo package com.automation.testcase

- Chứa class TC001_Login_Logout.java là nơi để thực hiện việc test chính, bao gồm gọi các action nhằm lấy dữ liệu từ file excel đã lưu trong Data, điền dữ liệu này vào ô trống để đăng nhập, logout nếu đã đăng nhập thành công, chụp màn hình nếu test fail, lưu hình chụp vào Screenshots.

```
package com.automation.testcase;

import java.io.IOException;
import org.testng.annotations.Test;
import com.automation.base.DriverInstance;
import com.automation.pom.LoginPage;
import com.automation.utils.DataGenerators;
import com.automation.pom.LogoutPage;

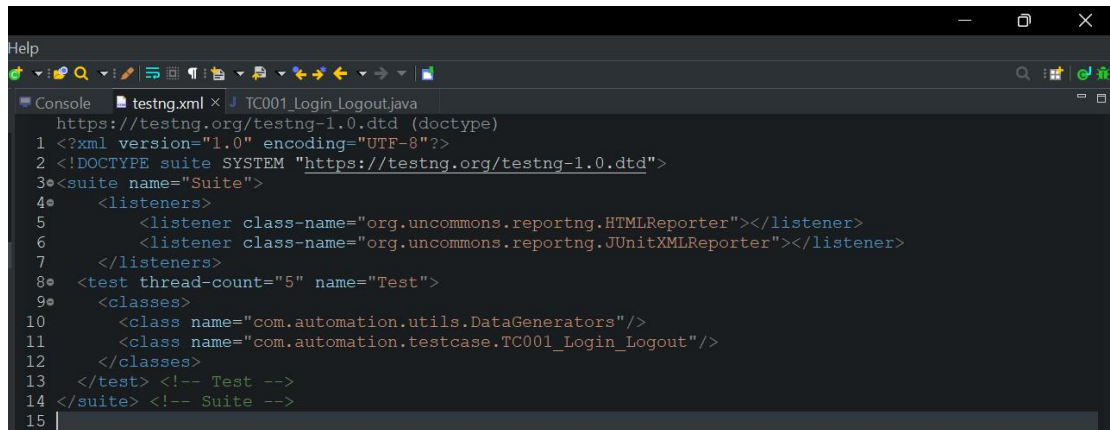
public class TC001_Login_Logout extends DriverInstance{
    @Test(dataProvider = "Login", dataProviderClass = DataGenerators.class)
    public void tc_001_login_functionality(String email, String pass) throws IOException,
    InterruptedException {
        LoginPage login = new LoginPage(driver);
        login.clickSignupIcon();
        login.enterEmailAddress(email);
        login.enterPassword(pass);
        Thread.sleep(2000);
        login.clickLoginButton();
        System.out.println("\nLogin successful");

        Thread.sleep(2000);
        LogoutPage logout = new LogoutPage(driver);
        logout.clickLogoutButton();
        System.out.println("\nLogout successful");
    }
}
```

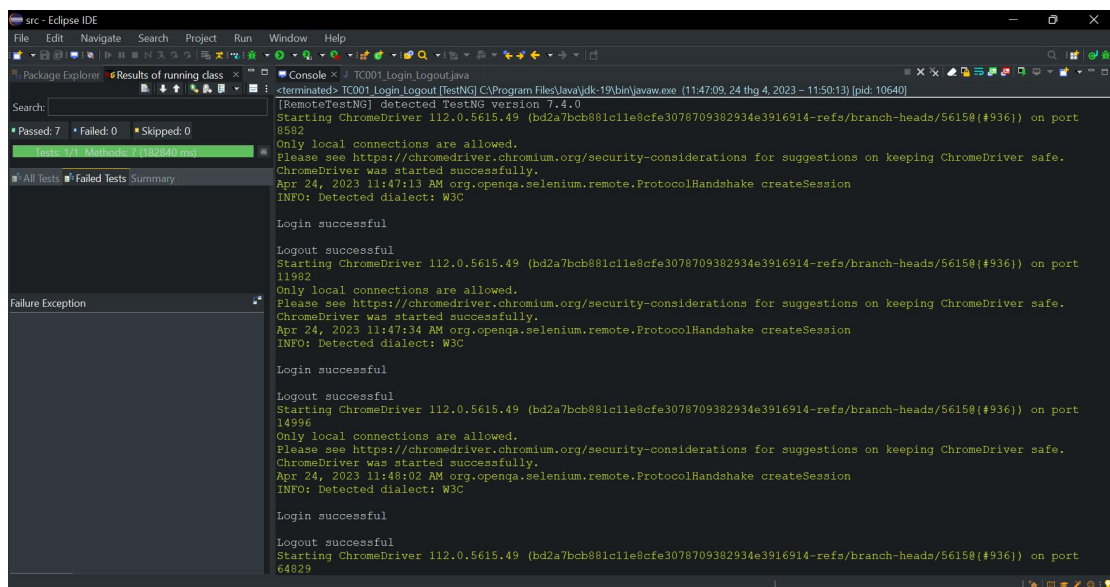
PHẦN 4: KẾT QUẢ CHẠY CHƯƠNG TRÌNH

4.1 Trường hợp chạy thành công với email, pass đúng

- Chạy từ testcase TC001_Login_Logout hoặc chạy test suit từ testng.xml



```
1 https://testng.org/testng-1.0.dtd (doctype)
2 <?xml version="1.0" encoding="UTF-8"?>
3 <!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">
4 <suite name="Suite">
5   <listeners>
6     <listener class-name="org.uncommons.reporting.HTMLReporter"/>
7     <listener class-name="org.uncommons.reporting.JUnitXMLReporter"/>
8   </listeners>
9   <test thread-count="5" name="Test">
10    <classes>
11      <class name="com.automation.utils.DataGenerators"/>
12      <class name="com.automation.testcase.TC001_Login_Logout"/>
13    </classes>
14  </test> <!-- Test -->
15</suite> <!-- Suite -->
```



```
<terminated> TC001_Login_Logout [TestNG] C:\Program Files\Java\jdk-19\bin\javaw.exe (11:47:09, 24 thg 4, 2023 - 11:50:13) [pid: 10640]
[RemoteTestNG] detected TestNG version 7.4.0
Starting ChromeDriver 112.0.5615.49 (bd2a7bcb881c11e8cfe3078709382934e3916914-refs/branch-heads/56158{#936}) on port 8582
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Apr 24, 2023 11:47:13 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C

Login successful

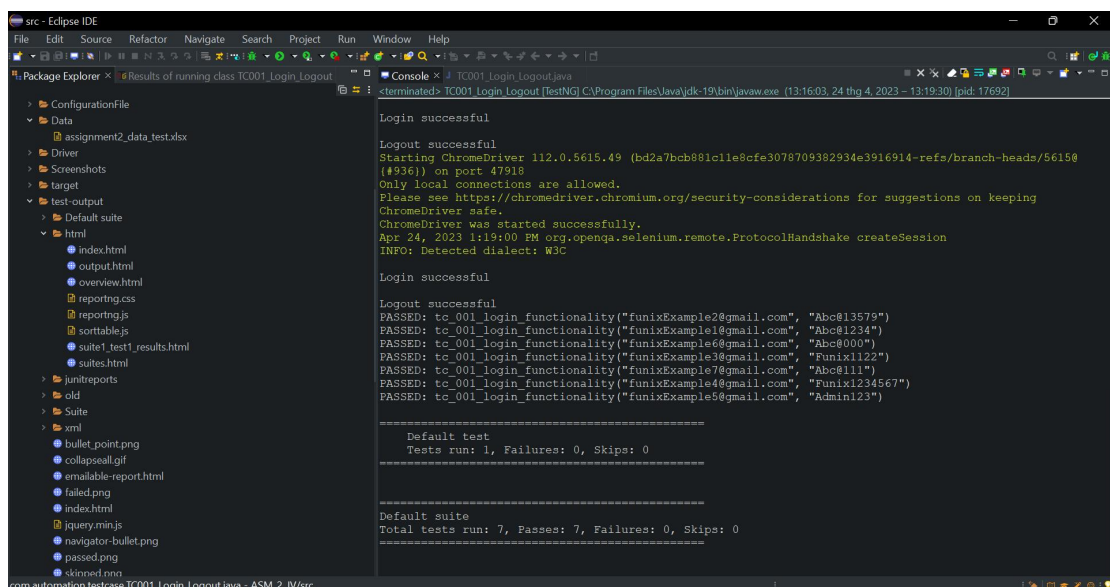
Logout successful
Starting ChromeDriver 112.0.5615.49 (bd2a7bcb881c11e8cfe3078709382934e3916914-refs/branch-heads/56158{#936}) on port 11982
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Apr 24, 2023 11:47:34 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C

Login successful

Logout successful
Starting ChromeDriver 112.0.5615.49 (bd2a7bcb881c11e8cfe3078709382934e3916914-refs/branch-heads/56158{#936}) on port 14996
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Apr 24, 2023 11:48:02 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C

Login successful

Logout successful
Starting ChromeDriver 112.0.5615.49 (bd2a7bcb881c11e8cfe3078709382934e3916914-refs/branch-heads/56158{#936}) on port 64829
```



```
<terminated> TC001_Login_Logout [TestNG] C:\Program Files\Java\jdk-19\bin\javaw.exe (13:16:03, 24 thg 4, 2023 - 13:19:30) [pid: 17692]

Login successful

Logout successful
Starting ChromeDriver 112.0.5615.49 (bd2a7bcb881c11e8cfe3078709382934e3916914-refs/branch-heads/56158{#936}) on port 47918
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Apr 24, 2023 1:19:00 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C

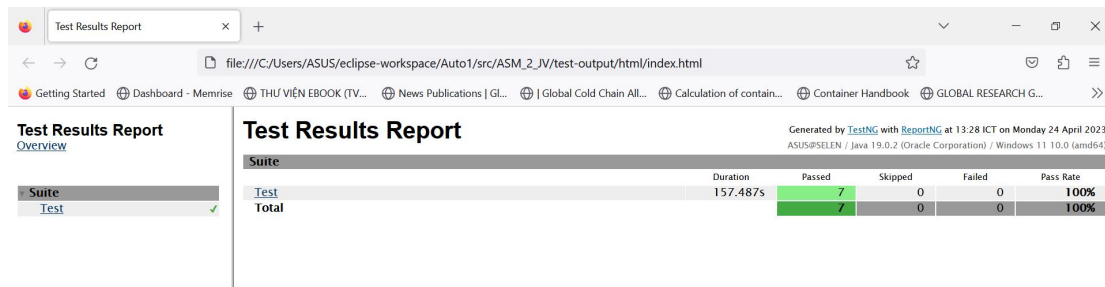
Login successful

Logout successful
PASSED: tc_001_login_functionality("funixExample2@gmail.com", "Abc@13579")
PASSED: tc_001_login_functionality("funixExample1@gmail.com", "Abc@1234")
PASSED: tc_001_login_functionality("funixExample6@gmail.com", "Abc@000")
PASSED: tc_001_login_functionality("funixExample3@gmail.com", "Funixil22")
PASSED: tc_001_login_functionality("funixExample7@gmail.com", "Abc@111")
PASSED: tc_001_login_functionality("funixExample4@gmail.com", "Funixi1234567")
PASSED: tc_001_login_functionality("funixExample5@gmail.com", "Admin123")

-----
Default test
Tests run: 1, Failures: 0, Skips: 0
-----

Default suite
Total tests run: 7, Passes: 7, Failures: 0, Skips: 0
-----
```


- Mở index.html trong folder html để xem báo cáo

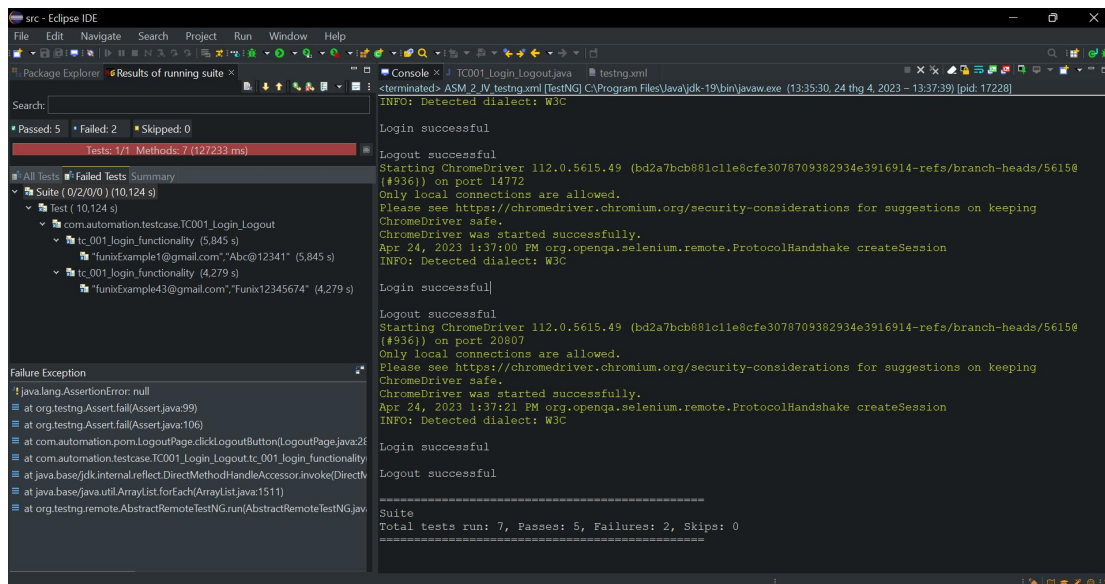


4.2 Trường hợp báo lỗi testcase fail với email, pass sai

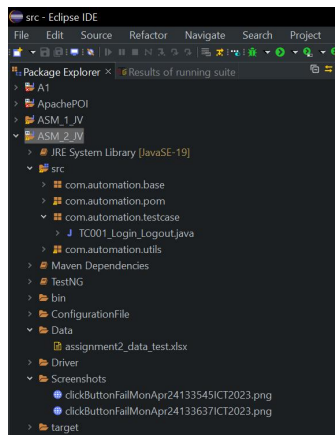
- Kiểm thử với trường hợp email, pass không tồn tại: sửa lại 1 pass /email trong file **assignment2_data_test.xlsx**

	A	B	C	D
1	funixExample1@gmail.com	Abc@12341		
2	funixExample2@gmail.com	Abc@13579		
3	funixExample3@gmail.com	Funix1122		
4	funixExample43@gmail.com	Funix12345674		
5	funixExample5@gmail.com	Admin123		
6	funixExample6@gmail.com	Abc@000		
7	funixExample7@gmail.com	Abc@111		
8				
9				
10				
11				

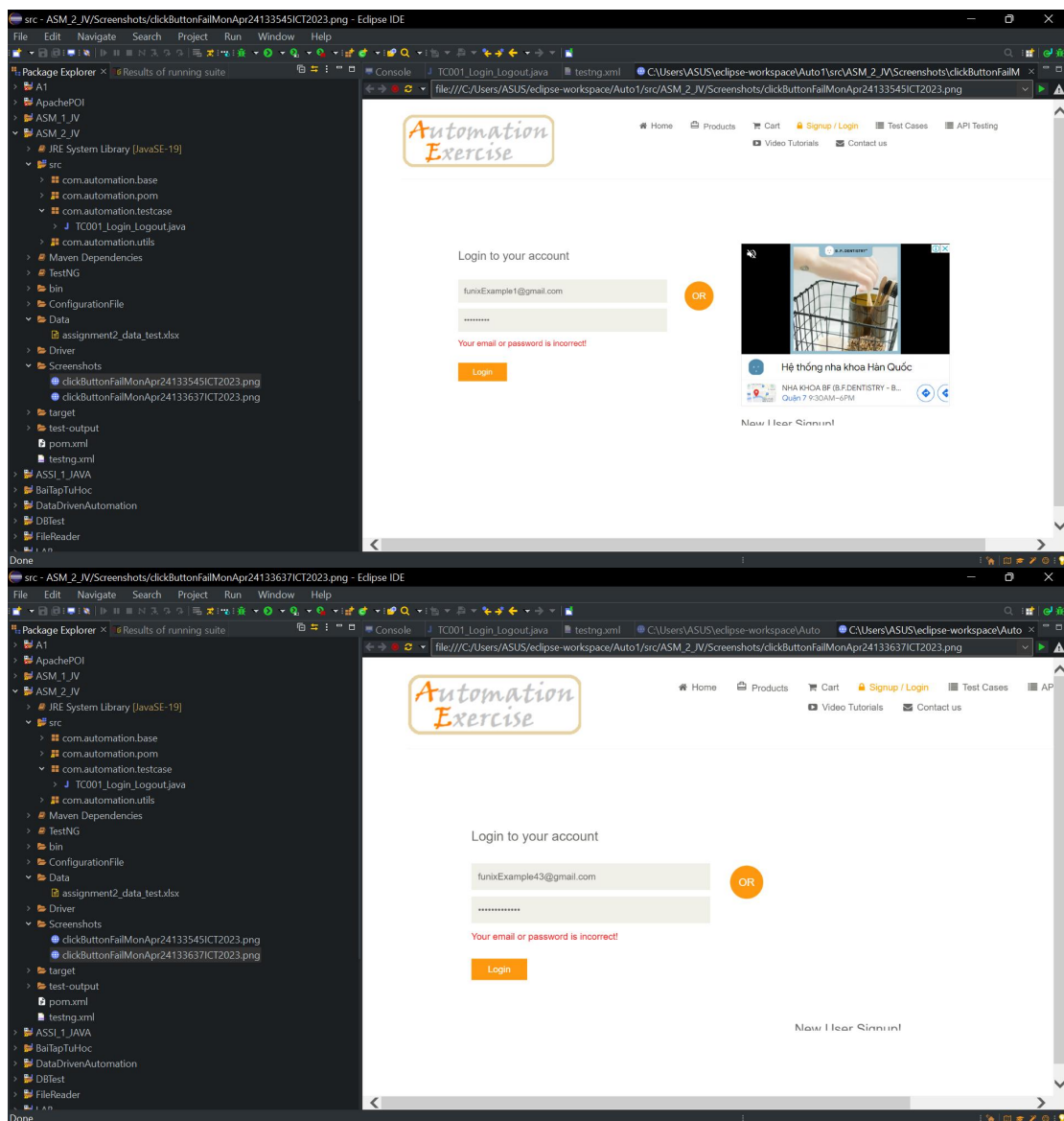
- Kết quả chạy test trả về với 2 test fail



- Folder Screenshots có hiển thị 2 ảnh chụp tương đương 2 test fail



- Nhấn từng ảnh sẽ hiển thị lỗi được chụp



- Mở index.html trong folder html để xem báo cáo với tỉ lệ pass, fail, % pass rate

Test Results Report					
Generated by TestNG with ReportNG at 13:35 ICT on Monday 24 April 2023 ASUS@SELEN / Java 19.0.2 (Oracle Corporation) / Windows 11 10.0 (amd64)					
Suite					
Test	Duration	Passed	Skipped	Failed	Pass Rate
Test	126.133s	5	0	2	71%
Total		5	0	2	71%

- Xem chi tiết test thành công, test lỗi và ảnh chụp lỗi

Test Results Report

Test duration: 126.133s

Suite

Test

Failed Tests

com.automation.testcase.TC001_Login_Logout

tc_001_login_functionality

5.845s

Method arguments: "FunixExample1@gmail.com", "Abc012341"
img_width="418" height="240" alt="alternativeName" title="title" src="./surefire-reports/html/screenshot/./Screenshots/ClickButtonFailMonApr241335451CT2023.png"
java.lang.AssertionError: null

tc_001_login_functionality

4.279s

Method arguments: "FunixExample43@gmail.com", "Funix12345674"
img_width="418" height="240" alt="alternativeName" title="title" src="./surefire-reports/html/screenshot/./Screenshots/ClickButtonFailMonApr241336371CT2023.png"
java.lang.AssertionError: null

Passed Tests

com.automation.testcase.TC001_Login_Logout

tc_001_login_functionality

9.930s

Method arguments: "FunixExample3@gmail.com", "Funix1122"

tc_001_login_functionality

7.458s

Method arguments: "FunixExample6@gmail.com", "Abc0000"

tc_001_login_functionality

12.198s

Method arguments: "FunixExample5@gmail.com", "Admin123"

tc_001_login_functionality

8.938s

Method arguments: "FunixExample2@gmail.com", "Abc013579"

tc_001_login_functionality

10.394s

Method arguments: "FunixExample7@gmail.com", "Abc0111"