

PHASE 2 PROJECT: Framework to Test the Swaglab Application in Different Environment.

DESCRIPTION:

Create a BDD with Cucumber Framework to perform the Regression Testing for SwagLab application and to generate various reports.

Application: <https://www.saucedemo.com/>

- * Create new project using Cucumber with Maven
- * Add all the cucumber dependencies
- * Add your chromedriver
- * You need to add Hooks
- * You need to create two feature files

1) Login

Positive Login

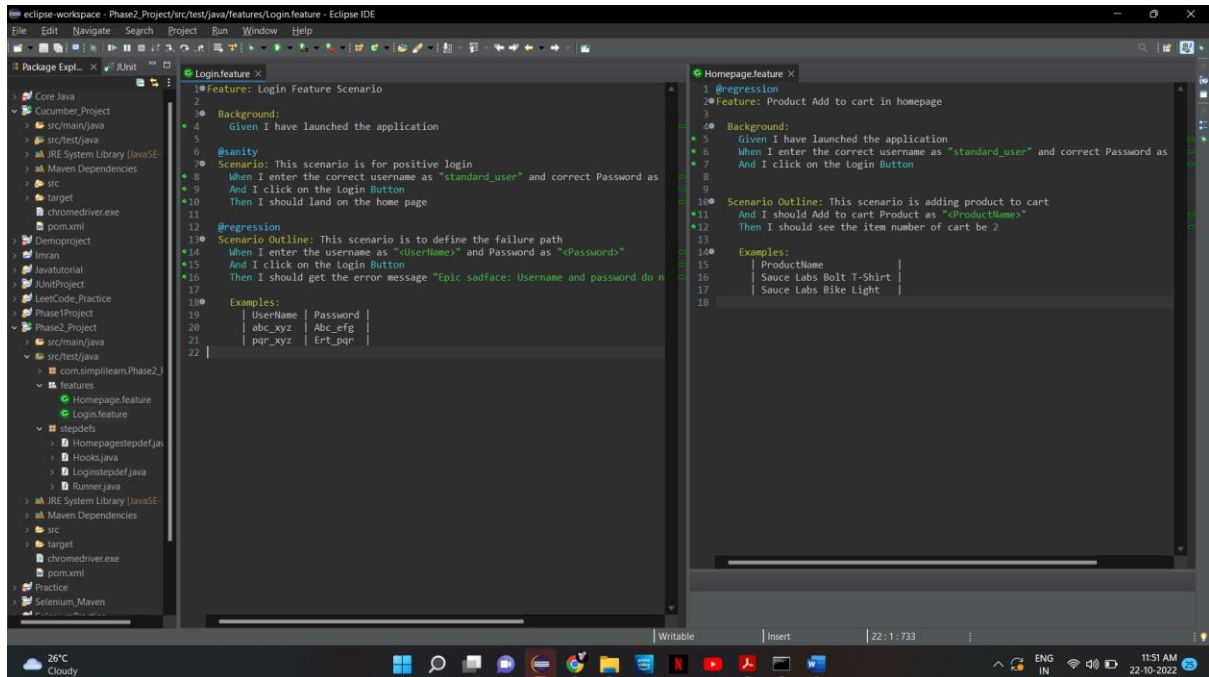
Negative Login to validate the error

2) HomePage

Need to add the product to the cart , and product name has to be given from feature file

- * Add the associated stepf defs
 - * Need to add the runner class
 - * Provide two tags sanity and regression
 - * run your features using runner class
 - * run your features using cmd
 - * run your features using jenkins and publish the report
-
-

Features:

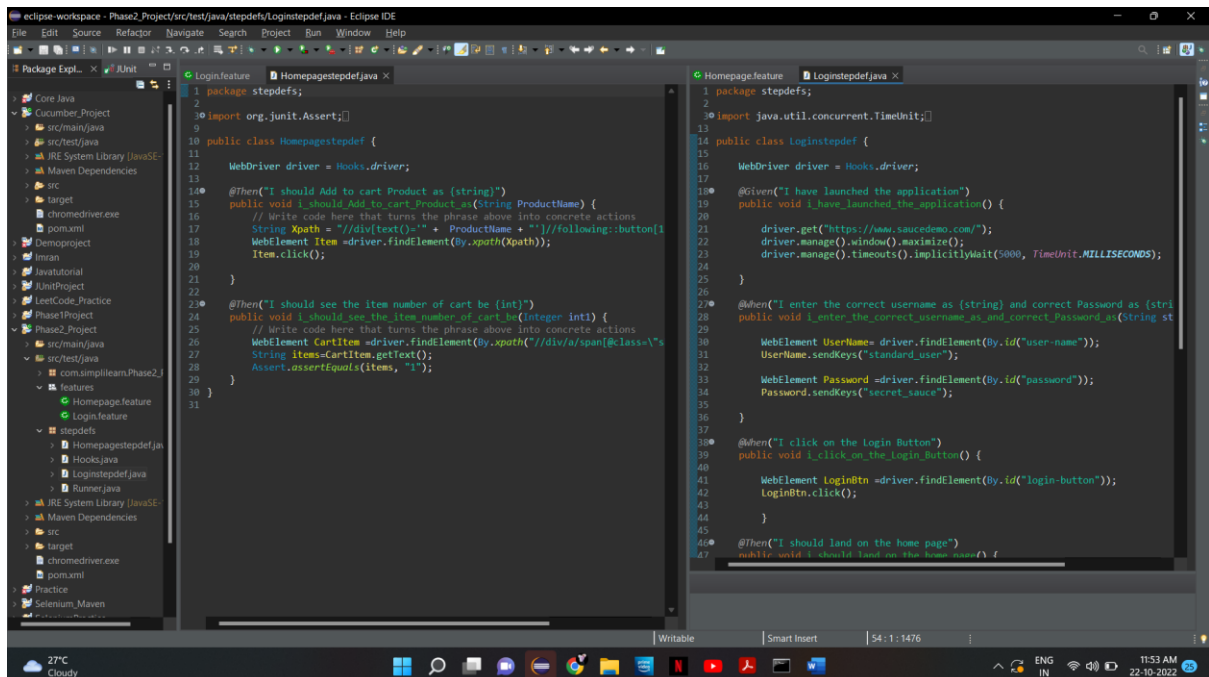


The screenshot shows the Eclipse IDE with two feature files open. The left pane shows the project structure with 'Login.feature' selected. The right pane shows the content of 'Homepage.feature'.

```
1 Feature: Login Feature Scenario
2
3 Background:
4   Given I have launched the application
5
6 @sanity
7 Scenario: This scenario is for positive login
8   When I enter the correct username as "standard_user" and correct Password as
9   And I click on the Login Button
10  Then I should land on the home page
11
12 @Regression
13 Scenario Outline: This scenario is to define the failure path
14   When I enter the username as "<Username>" and Password as "<Password>"
15   And I click on the Login Button
16   Then I should get the error message "Epic sadface: Username and password do n
17
18 Examples:
19   | Username | Password |
20   | abc_xyz | Abc_efg |
21   | pqr_xyz | Ert_pqr |
22
```

```
1 @Regression
2 Feature: Product Add to cart in homepage
3
4 Background:
5   Given I have launched the application
6   When I enter the correct username as "standard_user" and correct Password as
7   And I click on the Login Button
8
9 Scenario Outline: This scenario is adding product to cart
10  And I should Add to cart Product as "<ProductName>"
11  Then I should see the item number of cart be 2
12
13 Examples:
14   | ProductName |
15   | Sauce Labs Bolt T-Shirt |
16   | Sauce Labs Bike Light |
17
18
```

Stepdef:

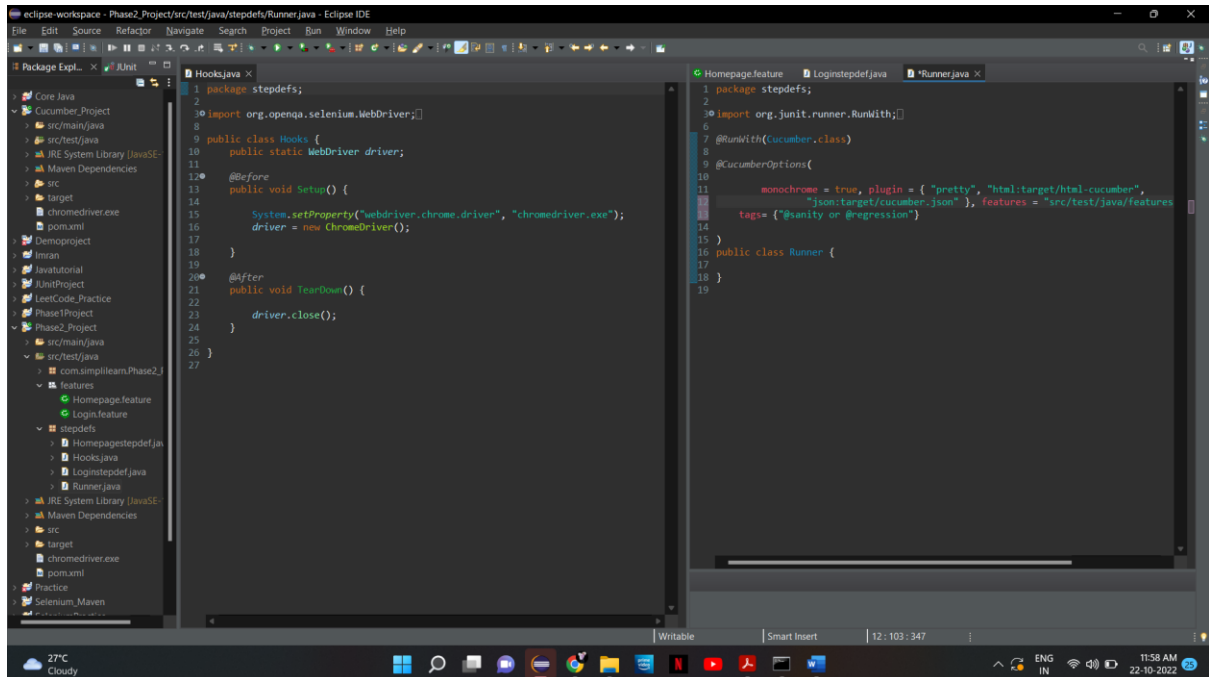


The screenshot shows the Eclipse IDE with two step definition files open. The left pane shows the project structure with 'Homepagestepdef.java' selected. The right pane shows the content of 'Loginstepdef.java'.

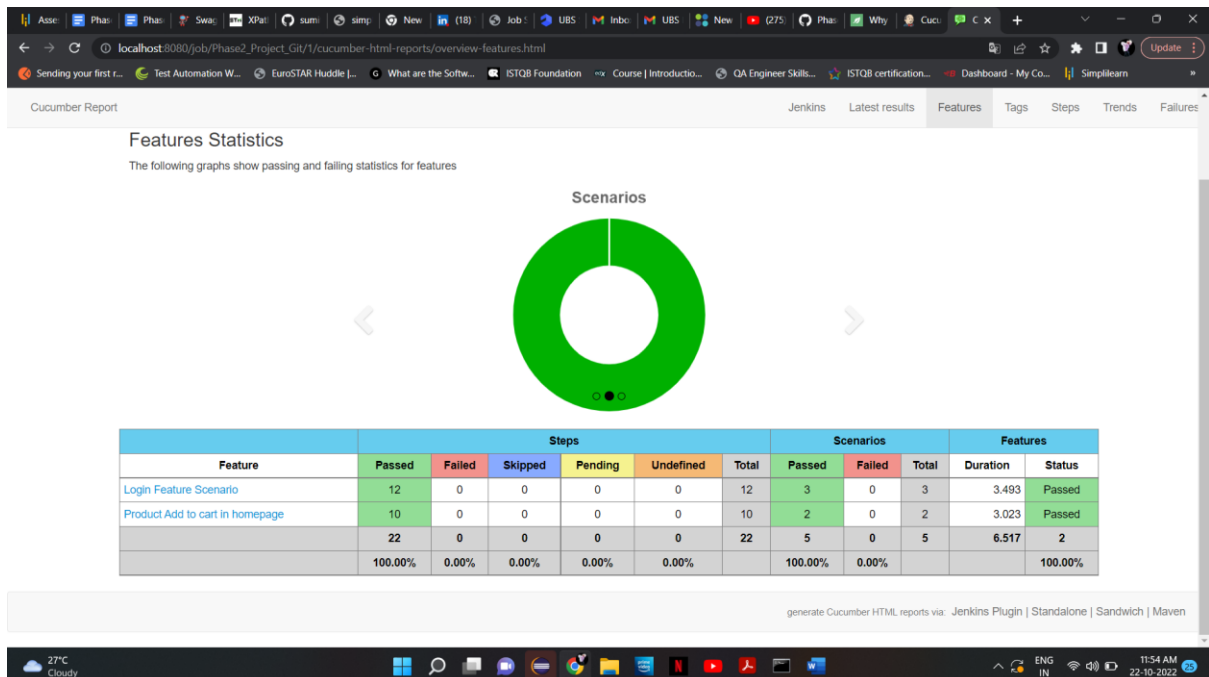
```
1 package stepdefs;
2
3 import org.junit.Assert;
4
5 public class Homepagestepdef {
6
7     WebDriver driver = Hooks.driver;
8
9     @Then("I should Add to cart Product as (string)")
10    public void i_should_Add_to_cart_Product_as(String ProductName) {
11        // Write code here that turns the phrase above into concrete actions
12        String Xpath = "//div(text()=' " + ProductName + "'//following:button[1
13        WebElement Item =driver.findElement(By.xpath(Xpath));
14        Item.click();
15    }
16
17    @Then("I should see the item number of cart be (int)")
18    public void i_should_see_the_item_number_of_cart_be(Integer Int1) {
19        // Write code here that turns the phrase above into concrete actions
20        WebElement CartItem =driver.findElement(By.xpath("//div/a/span@class='\"
21        String Items=CartItem.getText();
22        Assert.assertEquals(Items, "1");
23    }
24
25 }
26
```

```
1 package stepdefs;
2
3 import java.util.concurrent.TimeUnit;
4
5 public class Loginstepdef {
6
7     WebDriver driver = Hooks.driver;
8
9     @Given("I have launched the application")
10    public void i_have_launched_the_application() {
11        driver.get("https://www.saucedemo.com/");
12        driver.manage().window().maximize();
13        driver.manage().timeouts().implicitlyWait(5000, TimeUnit.MILLISECONDS);
14    }
15
16    @When("I enter the correct username as (string) and correct Password as (stri
17    public void i_enter_the_correct_username_as_and_correct_Password_as(string st
18        WebElement UserName = driver.findElement(By.id("user-name"));
19        UserName.sendKeys("standard_user");
20        WebElement Password =driver.findElement(By.id("password"));
21        Password.sendKeys("secret_sauce");
22    }
23
24    @When("I click on the Login Button")
25    public void i_click_on_the_login_Button() {
26        WebElement LoginBtn =driver.findElement(By.id("login-button"));
27        LoginBtn.click();
28    }
29
30    @Then("I should land on the home page")
31    public void i_should_land_on_the_home_page() {
32
33    }
34
35 }
36
```

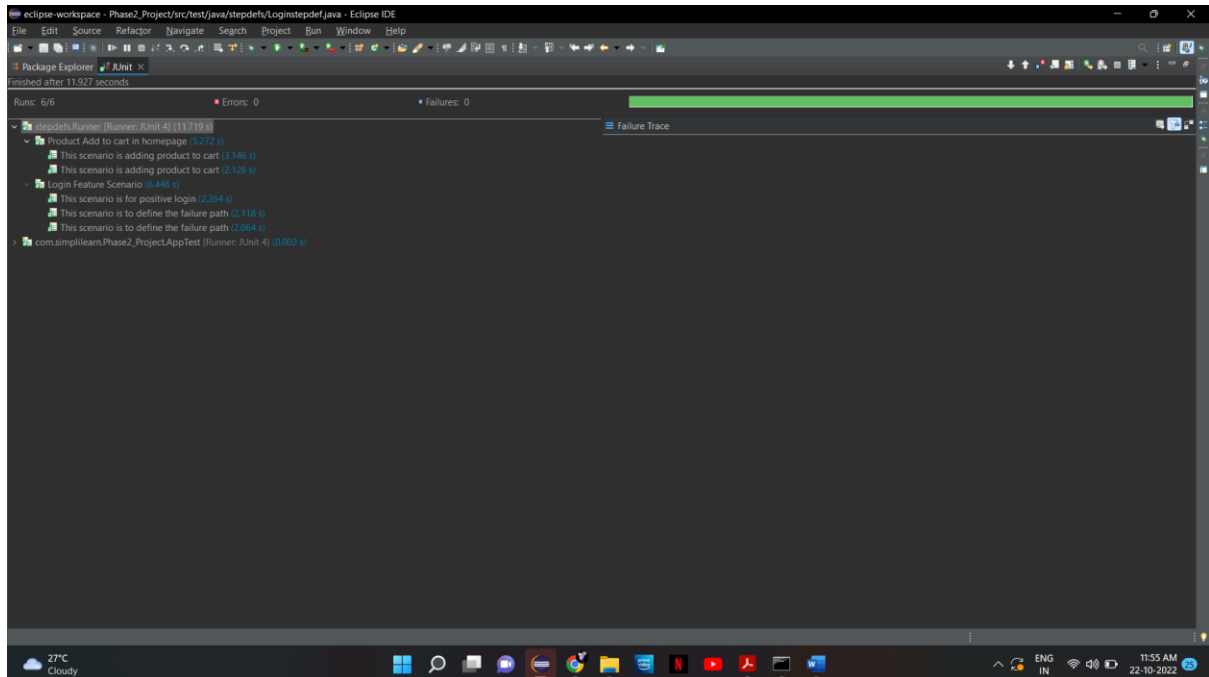
Hooks & Runner:



Jenkins:



Console:



Github: https://github.com/imanovian/Phase2_Project.git

Src code:

Homepagestepdef:

```
package stepdefs;
```

```
import org.junit.Assert;
```

```
import org.openqa.selenium.By;
```

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

```
import org.openqa.selenium.WebElement;
```

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

```
public class Homepagestepdef {
```

```
    WebDriver driver = Hooks.driver;
```

```
    @Then("I should Add to cart Product as {string}")
```

```

public void i_should_Add_to_cart_Product_as(String ProductName) {
    // Write code here that turns the phrase above into concrete actions
    String Xpath = "//div[text()='\" + ProductName + \"']/following::button[1]";
    WebElement Item = driver.findElement(By.xpath(Xpath));
    Item.click();
}

@Then("I should see the item number of cart be {int}")
public void i_should_see_the_item_number_of_cart_be(Integer int1) {
    // Write code here that turns the phrase above into concrete actions
    WebElement CartItem =
driver.findElement(By.xpath("//div/a/span[@class=\"shopping_cart_badge\"]"));
    String items = CartItem.getText();
    Assert.assertEquals(items, "1");
}
}

```

Loginstepdef:

```
package stepdefs;
```

```
import java.util.concurrent.TimeUnit;
```

```
import org.junit.Assert;
```

```
import org.openqa.selenium.By;
```

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

```
import org.openqa.selenium.WebElement;
```

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

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

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

```
public class Loginstepdef {
```

```
WebDriver driver = Hooks.driver;
```

```
@Given("I have launched the application")
```

```
public void i_have_launched_the_application() {
```

```
    driver.get("https://www.saucedemo.com/");
```

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

```
    driver.manage().timeouts().implicitlyWait(5000, TimeUnit.MILLISECONDS);
```

```
}
```

```
@When("I enter the correct username as {string} and correct Password as {string}")
```

```
public void i_enter_the_correct_username_as_and_correct_Password_as(String string, String  
string2) {
```

```
    WebElement UserName = driver.findElement(By.id("user-name"));
```

```
    UserName.sendKeys("standard_user");
```

```
    WebElement Password = driver.findElement(By.id("password"));
```

```
    Password.sendKeys("secret_sauce");
```

```
}
```

```
@When("I click on the Login Button")
```

```
public void i_click_on_the_Login_Button() {
```

```
    WebElement LoginBtn = driver.findElement(By.id("login-button"));
```

```
    LoginBtn.click();
```

```
}
```

```
@Then("I should land on the home page")
```

```

public void i_should_land_on_the_home_page() {

    String title = driver.getTitle();

    System.out.println("HomePage Title::" + title);

    Assert.assertEquals("Swag Labs", title);

}

@When("I enter the username as {string} and Password as {string}")
public void i_enter_the_username_as_and_Password_as(String UserNameVal, String
PasswordVal) {

    WebElement UserName = driver.findElement(By.id("user-name"));

    UserName.sendKeys(UserNameVal);

    WebElement Password = driver.findElement(By.id("password"));

    Password.sendKeys(PasswordVal);

}

@Then("I should get the error message {string}")
public void i_should_get_the_error_message(String string) {

    WebElement Error = driver.findElement(By.xpath("//h3[@data-test='error']"));

    String ActError = Error.getText();

    String ExpError = "Epic sadface: Username and password do not match any user in
this service";

    Assert.assertEquals(ExpError, ActError);

}
}

```

=====END=====