

TASK – 03: Automated Login Test for a Web Application

Objective:

To automate the login functionality of a web application by validating positive and negative test cases using Selenium automation.

Tools & Technologies Used:

- Selenium WebDriver
- Java
- TestNG
- Maven
- Google Chrome Browser

Demo Website:

<https://the-internet.herokuapp.com/login>

Test Scenarios Covered:

- Valid login with correct credentials
- Invalid login with incorrect credentials
- Login with empty username field
- Login with empty password field

Automation Test Code (Selenium + Java + TestNG):

```
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.annotations.AfterMethod;
import org.testng.annotations.BeforeMethod;
import org.testng.annotations.Test;

public class LoginTest {

    WebDriver driver;

    @BeforeMethod
    public void setup() {
        driver = new ChromeDriver();
        driver.manage().window().maximize();
        driver.get("https://the-internet.herokuapp.com/login");
    }

    // Positive Test Case
    @Test
    public void validLoginTest() {
        driver.findElement(By.id("username")).sendKeys("tomsmith");
        driver.findElement(By.id("password")).sendKeys("SuperSecretPassword!");
        driver.findElement(By.cssSelector("button[type='submit']")).click();

        String successMsg = driver.findElement(By.id("flash")).getText();
        Assert.assertTrue(successMsg.contains("You logged into a secure area!"));
    }

    // Negative Test Case - Invalid Credentials
    @Test
    public void invalidLoginTest() {
        driver.findElement(By.id("username")).sendKeys("wrongUser");
        driver.findElement(By.id("password")).sendKeys("wrongPass");
        driver.findElement(By.cssSelector("button[type='submit']")).click();

        String errorMsg = driver.findElement(By.id("flash")).getText();
        Assert.assertTrue(errorMsg.contains("Your username is invalid!"));
    }
}
```

```

    }

    // Negative Test Case - Empty Username
    @Test
    public void emptyUsernameTest() {
        driver.findElement(By.id("password")).sendKeys("SuperSecretPassword!");
        driver.findElement(By.cssSelector("button[type='submit']")).click();

        String errorMsg = driver.findElement(By.id("flash")).getText();
        Assert.assertTrue(errorMsg.contains("Your username is invalid!"));
    }

    // Negative Test Case - Empty Password
    @Test
    public void emptyPasswordTest() {
        driver.findElement(By.id("username")).sendKeys("tomsmith");
        driver.findElement(By.cssSelector("button[type='submit']")).click();

        String errorMsg = driver.findElement(By.id("flash")).getText();
        Assert.assertTrue(errorMsg.contains("Your password is invalid!"));
    }

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

```

Expected Results:

- Successful login for valid credentials
- Proper error messages for invalid or empty inputs

Conclusion:

This automated test suite validates the login functionality effectively by covering both positive and negative scenarios, ensuring application reliability.