Overview

The objective of this task is to create a test automation suite that fulfils the acceptance tests

below.

Requirements

● Selenium with WebDriver testing against Chrome browser should be used

● Tests can be written in either Java or C#

● The “Page Object Model” pattern must be used

● Use CSS Selectors to identify the elements on the page. Avoid using XPath

● Code reusability should be maximized

● Use Assertions to validate the text and the error messages

Acceptance Tests

**Scenario:** Login failure

**Given** user on http://the-internet.herokuapp.com/login

**When** user clicks on the Login button

**Then** an error message stated “Your username is invalid!” appear

**Scenario:** Login success

**Given** user on http://the-internet.herokuapp.com/login

**When** user fill in username (tomsmith), password (SuperSecretPassword!) and clicks on Login

button

**Then** a successful message stated “You logged into a secure area!” appear

**Scenario:** Alert - JS Confirm

**Given** user on http://the-internet.herokuapp.com/javascript\_alerts

**When** user clicks on “Click for JS Confirm” button and select [OK].

**Then** result showing “You clicked: Ok”

**Scenario:** Text Verification

**Given** user on http://the-internet.herokuapp.com/javascript\_alerts

**When** user clicks on “Click for JS Prompt” button

AND user enters the test “Test JS Prompt” and clicks [Cancel]

**Then** result showing “You entered: null”

**Scenario:** File upload

**Given** user on http://the-internet.herokuapp.com/upload

**When** user upload a file named as “burrito.jpg” and clicks [Upload]

**Then** next screen should show “File Uploaded!” as title and filename “burrito.jpg”.

**Scenario:** Accessing iFrame

**Given** user on http://the-internet.herokuapp.com/iframe

**When** user type “I love pizza” in the text box

**Then** textbox shows “I love pizza”

Reference: