Shree Ramkrishna Institute of Computer Education and Applied Sciences
S.Y.B.Sc. Computer Science Sem -4
Practical Assignment
Subject: Software Testing

Selenium - Web Application Test automation tool

Instructions:

1) **Start Lampp Server:** *sudo /opt/lampp/lampp start*
2) **To give permission :** sudo chmod 777 /opt/lampp/htdocs/dashboard
3) **For Site Visit :** localhost:8090/dashboard/Your Site Title
4) **To give FTP permission :** sudo chmod 777 -R /opt/lampp/htdocs/project folder

- **PHP File Setup.**
- **Start Lampp Server**
- **Create one Folder and Rename It with 'Selenium Tool Project'**
- **Extract your 10.25.50.252/hotel_system_testing.zip folder in a specific folder i.e. 'Selenium Tool Project'.**
- **Other location-> opt -> lampp -> htdocs -> dashboard -> Selenium Tool Project -> hotel_system_testing.**

Selenium IDE commands :-

Command	Explanation	Example
open	Opens a specified URL in the browser.	open Target: /login.php
type	Enters text into an input field.	type Target: id=username Value: testuser

click	Clicks on an element such as a button, link, or checkbox.	click Target: id=loginButton
assertTitle	Verifies the page title matches the expected value.	assertTitle Target: Dashboard
verifyText	Checks if a specific element contains the expected text.	verifyText Target: id=welcomeMessage Value: Welcome, testuser
assertText	Asserts that the text of an element matches the expected value exactly.	assertText Target: id=errorMessage Value: Invalid credentials.
storeValue	Stores the value of a field in a variable for later use.	storeValue Target: id=username Value: storedUsername
sendKeys	Simulates typing into an input field, including special keys like Enter or Tab.	sendKeys Target: id=password Value: password123${KEY_ENTER}

select	Selects an option from a dropdown menu.	select Target: id=countryDropdown Value: India
verifyElementPresent	Checks if a specific element exists on the page.	verifyElementPresent Target: id=logoutButton
waitForElementPresent	Waits until the specified element is present before proceeding to the next command.	waitForElementPresent Target: id=dashboardContent
pause	Pauses the test execution for a specified duration (in milliseconds).	pause Value: 2000 (pauses for 2 seconds)
captureEntirePageScreenshot	Captures a screenshot of the current browser window.	captureEntirePageScreenshot Target: /path/to/screenshot.png
dragAndDropToObject	Drags an element and drops it on a target element.	dragAndDropToObject Target: id=sourceElement Value: id=targetElement
mouseOver	Simulates hovering the mouse over an element.	mouseOver Target: id=menuButton

executeScript	**Executes custom JavaScript on the current page.**	**executeScript** **Script: return document.title** **Output: storedTitle**
assertAlert	**Verifies the text of a pop-up alert matches the expected value.**	**assertAlert** **Target: Are you sure you want to submit?**
acceptAlert	**Clicks "OK" on a pop-up alert.**	**acceptAlert**
dismissAlert	**Clicks "Cancel" on a pop-up alert.**	**dismissAlert**
runScript	**Runs JavaScript code in the browser.**	**runScript** **Value: alert('Hello from Selenium!')**
goBack	**Navigates back to the previous page in the browser.**	**goBack**

● Assignment:1

● Test a Hostel Management System Website Using Selenium IDE

Registration Page

- **Task-1 Verify Required Fields Validation (Hint: run script)**

- **Objective:** Ensure that all required fields (registration number, first name, last name, gender, contact number, email, password) are validated and prompt an error if any are left empty.

- **Expected Result:**

- An alert is displayed prompting the "**user to fill in all the required fields**"

—---

- **Task -2 Verify Registration Number Format (Hint: run script)**

- **Objective**:Test the validation of the registration number field to ensure it must be alphanumeric and between 6 to 10 characters.

- **Expected Result**: A popup alert should appear with the message **"Registration number must be alphanumeric and between 6 to 10 characters."**

—---

- **Task -3 Verify First Name Length (Hint: run script)**

- **Objective** : Ensure that the first name is at least 2 characters long.

- **Expected Result**: A popup alert should appear with the message **"First name must be at least 2 characters long."**

—---

- **Task - 4: Verify Last Name Length (Hint: run script)**
- **Objective** : Ensure that the last name is at least 2 characters long.
- **Expected Result**: A popup alert should appear with the message **"Last name must be at least 2 characters long."**

—---

- **Task - 5: Verify Gender Selection (Hint: run script)**
- **Objective :** Check if the user can select a valid gender from the dropdown.
- **Expected Result:** A popup alert should appear with the message **"Please select a valid gender."**

—---

- **Task -6 Verify Contact Number Format (Hint: run script)**
- **Objective :** Ensure that the contact number is a valid 10-digit number.
- **Expected Result:** A popup alert should appear with the message **"Contact number must be a 10-digit number."**

—---

- **Task - 7: Verify Email Format (Hint: run script)**
- **Objective:** Test if the email input field accepts only valid email formats.
- **Expected Result:** A popup alert should appear with the message **"Invalid email format."**

—---

- **Task - 8: Verify Password Length (Hint: run script)**
- **Objective :** Ensure the password is at least 8 characters long.
- **Expected Result**: A popup alert should appear with the message **"Password must be at least 8 characters long."**

—---

- **Task - 9: Verify Password and Confirm Password Match (Hint: run script)**
- **Test Description:** Ensure the password and confirm password fields match.
- **Expected Result**: A popup alert should appear with the message **"Password and Confirm Password do not match."**

—---

- **Task - 10: Verify Duplicate Email or Registration Number (Hint: run script)**
- **Objective :** Check if the system prevents the registration of a duplicate email or registration number.
- **Expected Result**: A popup alert should appear with the message **"Registration number or email id already registered."**

—---

- **Task - 11: Verify Successful Registration (Hint: run script)**
- **Objective** : Test the successful registration process with valid inputs.
- **Expected Result**: A popup alert should appear with the message **"Student successfully registered," and the user should be redirected to the success page (if specified).**

- **Task - 12: Verify Reset Button Clears All Fields (Hint : assert value)**
- **Objective** : Ensure that the reset button clears all the fields.
- **Expected Result**: All fields **should be cleared, and no data should be retained.**

—---

Login Page

- **Task - 1: Test Login with Empty Password (Hint: run script)**
- **Objective :** Ensure that the user receives an error when the password field is empty but the email is provided.
- **Expected Result:** A JavaScript alert should appear with the message: "**All fields are required.**"

—---

- **Task - 2 Test Login with Empty Email (Hint: run script)**
- **Objective :** Ensure that the user receives an error when the email field is empty but the password is provided.
- **Expected Result** : A JavaScript alert should appear with the message: **"All fields are required."**

—---

- **Task -3 Test Invalid Email Format (Hint: run script)**
- **Objective :** Ensure that the user is shown an error if the email is in an invalid format (e.g., missing "@" symbol).

- **Expected Result**: A JavaScript alert should appear with the message: **"Please enter a valid email address."**

—---

- **Task - 4: Test Password Length Validation (Hint: run script)**
- **Objective:**Ensure that the password must be at least 6 characters long.
- **Expected Result :** A JavaScript alert should appear with the message: **"Passwords must be at least 6 characters long."**

—---

- **Task - 5 Test Invalid Login Credentials (Hint: run script)**
- **Objective :** Ensure that the user is shown an error if the email/registration number or password is incorrect.
- **Expected Result**:A JavaScript alert should appear with the message:**"Invalid Username/Email or password."**

—---

- **Task - 6 Test Successful Login (Hint: run script)**
- **Objective :** Ensure that the user can log in successfully with a valid email/registration number and password.
- **Expected Result**:The user should be redirected to the **dashboard.php** page.The **dashboard** page should contain an element that identifies the user is logged in (e.g., a "Welcome" message or user-specific content).

● Assignment:2

● Test a Hostel Management System Website Using Selenium IDE

User Dashboard Page

● Steps to login the user Dashboard
- Username: test@gmail.com
- Password: Test@123
- Login the user side

- **Task-1 Verify Page Title [Hint: VerifyTitle]**

- **Objective:** Ensure that the page title is correctly displayed.

- **Expected Output**: The browser title should be "DashBoard".

—---

- **Task-2 Verify Dashboard Heading [Hint : AssertText]**

- **Objective**: Verify the presence and text of the `<h2>` element with the class `page-title`

- **Expected Output**: The `<h2>` element should contain the text "Dashboard".

—---

- **Task-3 Sidebar Elements Verification [Hint: assertElementPresent]**
- **Objective:** Verify the sidebar loads and contains navigation links for the dashboard and profile.
- **Expected Result:** The sidebar should be present and clickable.

—---

- **Task - 4 Verify "My Profile" Panel Link [Hint: executeScript (return window.location.href) , assert]**
- **Objective :** Ensure the "My Profile" panel redirects to my-profile.php.
- **Expected Result:** Clicking "My Profile" should navigate to the profile details page.

—---

- **Task -5 Verify "My Room" Panel Link [Hint: executeScript (return window.location.href) , assert]**
- **Objective:** Ensure the "My Room" panel redirects to room-details.php.
- **Expected Result:** Clicking "My Room" should navigate to the room details page.

—---

- **Task - 6 Verify "My Profile" Content [Hint : assertText]**
- **Objective:** Check the "My Profile" page content is loaded properly.
- **Expected Result:** The "My Profile" page should display relevant user details.

—---

- **Task - 7 Logout Functionality [Hint: executeScript (return window.location.href) , assert]**
- **Objective:** Verify the logout link redirects to the login page.
- **Expected Result:** Clicking "Logout" should log the user out and display the login page.

Admin Dashboard Page

- **Steps to login the admin Dashboard**
- **Username:admin@gmail.com**
- **Password: Test@1234**
- **Login the admin side**

- **Task-1 Verify Page Title [Hint: VerifyTitle]**
- **Objective:** Ensure that the page title is correctly displayed.
- **Expected Output**: The browser title should be "DashBoard".

- **Task-2 Verify Dashboard Heading [Hint : AssertText]**
- **Objective**: Verify the presence and text of the `<h2>` element with the class

  `page-title`
- **Expected Output**: The `<h2>` element should contain the text "Dashboard".

- **Task-4 Sidebar Elements Verification [Hint: assertElementPresent]**
- **Objective:** Verify the sidebar loads and contains navigation links for the

  dashboard and profile.
- **Expected Result:** The sidebar should be present and clickable.

- **Task -5 Verify Total Students Count [Hint: assert text , executeScript (return window.location.href) ,assert]**

- **Objective :** Verify that the number of students shown in the "Students" panel is correct.
- **Expected Output:** The count in the "Students" panel matches the database.

- **Task - 6 Verify Total Rooms Count [Hint: assert text , executeScript (return window.location.href) ,assert]**
- **Objective :** Verify that the number of rooms shown in the "Total Rooms" panel is correct.
- **Expected Output:** The count in the "Rooms" panel matches the database.

- **Task -7 Verify "Registered Complaints" Panel Count [Hint: assert text , executeScript (return window.location.href) ,assert]**
- **Objective :** Verify that the number of registered complaints shown is correct.
- **Expected output :** The count in the "Registered Complaints" panel matches the database.

- **Task - 8: Verify "Courses" Panel Count [Hint: assert text , executeScript (return window.location.href) ,assert]**
- **Objective:** Verify that the number of courses is correctly displayed on the dashboard.
- **Expected Result**: The correct number of courses is displayed in the "Total Courses" panel.

- **Task- 9 : Verify Registered Complaints Count [Hint: assert text , executeScript (return window.location.href) ,assert]**
- **Objective:** Verify that the number of registered complaints is correctly displayed on the dashboard.
- **Expected Result :** The correct number of registered complaints is displayed in the "Registered Complaints" panel.