Test Task: Documentation and Test Scenario Identification

# Task:

## Documentation:

### a. Provide a test plan outline for testing the registration form. Include the following sections:

1. **Introduction**

The purpose of this test plan is to outline the testing strategy for the online registration form of the website. The form is designed to collect user information such as Full Name, Email Address, Password, Confirm Password, Date of Birth, Gender, and Newsletter Subscription preference. This document will guide the testing process to ensure the form functions correctly and meets the specified requirements.

1. **Objectives**

* Verify that the registration form collects the required information accurately.
* Ensure proper validation of inputs and display of appropriate error messages.
* Confirm successful submission of the form data to the server upon correct completion.

1. **Scope**

* Functional testing of form input fields.
* Validation of input data.
* Usability testing to ensure ease of use.
* Compatibility testing across different browsers and devices.

1. **Test Environment**

* **Hardware:** Various devices (PC, tablet, mobile) with different screen sizes.
* **Software:** Web browsers (Chrome, Firefox, Safari, Edge), operating systems (Windows, macOS, Linux, iOS, Android).

1. **Test Data**

* Valid and invalid email addresses
* Passwords of varying lengths and complexity
* Dates of birth within and outside valid ranges
* Different gender options
* Newsletter subscription preferences

1. **Test Scenarios**

* Positive Scenarios
  + Successful registration with valid data.
  + Form submission without subscribing to the newsletter.
* Negative Scenarios
  + Submission with missing required fields.
  + Submission with invalid email format.
  + Password and Confirm Password mismatch.
  + Date of Birth outside the valid range.

1. **Test Execution Schedule**

* Test design: 2 days
* Test data preparation: 1 day
* Test execution: 3 days
* Bug fixing and retesting: 2 days

1. **Risks and Assumptions**

* Risks: Delays in environment setup, changes in requirements, unavailability of required test data.
* Assumptions: Requirements are complete and stable, test environment is available, test data is accurate.

### b. Write test cases for at least three scenarios covering different aspects of the registration form. Each test case should include:

**Test Case 1: Successful Registration with Valid Data**

| **Test Case ID** | TC001 |
| --- | --- |
| **Test Case Description** | Verify successful registration with all valid input data. |
| **Preconditions** | The registration form is accessible. |
| **Test Steps:** | 1. Open the registration form. 2. Enter a valid Full Name. 3. Enter a valid Email Address. 4. Enter a valid Password (at least 8 characters). 5. Enter the same Password in the Confirm Password field. 6. Enter a valid Date of Birth. 7. Select Gender. 8. Choose Yes or No for Newsletter Subscription. 9. Click the Submit button. |
| **Expected Results:** | * The form is submitted successfully. * A success message is displayed. * User is redirected to the welcome page. |
| **Post-conditions** | The user data is stored in the database. |

**Test Case 2: Password and Confirm Password Mismatch**

| **Test Case ID** | TC002 |
| --- | --- |
| **Test Case Description** | Verify error message when Password and Confirm Password do not match. |
| **Preconditions** | The registration form is accessible. |
| **Test Steps:** | 1. Open the registration form. 2. Enter a valid Full Name. 3. Enter a valid Email Address. 4. Enter a valid Password (at least 8 characters). 5. Enter the different Password in the Confirm Password field. 6. Enter a valid Date of Birth. 7. Select Gender. 8. Choose Yes or No for Newsletter Subscription. 9. Click the Submit button. |
| **Expected Results:** | * An error message "Passwords do not match" is displayed. * The form is not submitted. |
| **Post-conditions** | The form remains on the registration page for correction. |

**Test Case 2: Invalid Email Address**

| **Test Case ID** | TC003 |
| --- | --- |
| **Test Case Description** | Verify error message when an invalid email address is entered. |
| **Preconditions** | The registration form is accessible. |
| **Test Steps:** | 1. Open the registration form. 2. Enter a valid Full Name. 3. Enter an invalid Email Address (e.g., "invalidemail.com"). 4. Enter a valid Password (at least 8 characters). 5. Enter the same Password in the Confirm Password field. 6. Enter a valid Date of Birth. 7. Select Gender. 8. Choose Yes or No for Newsletter Subscription. 9. Click the Submit button. |
| **Expected Results:** | An error message "Invalid email address" is displayed.  The form is not submitted. |
| **Post-conditions** | The form remains on the registration page for correction. |

## Scenario Identification:

### Positive Scenarios

1. **Successful Registration**

* **Steps**: User fills all fields with valid data and submits the form.
* **Expected Outcome**: The form is submitted successfully, a success message is displayed, and the user is redirected to the welcome page.

1. **Form Submission without Newsletter Subscription**

* **Steps**: User fills all fields with valid data, selects "No" for Newsletter Subscription, and submits the form.
* **Expected Outcome**: The form is submitted successfully, a success message is displayed, and the user is redirected to the welcome page.

### Negative Scenarios

1. **Missing Required Fields**

* **Steps**: User leaves one or more required fields (e.g., Email Address) empty and submits the form.
* **Expected Outcome**: An error message indicating the missing required fields is displayed, and the form is not submitted.

1. **Invalid Email Format**

* **Steps**: User enters an invalid email format (e.g., "invalidemail.com") and submits the form.
* **Expected Outcome**: An error message "Invalid email address" is displayed, and the form is not submitted.

## Test Writing

1. **Successful Registration**

describe('Registration Form', () => {

it('should register successfully with valid data', () => {

cy.visit('http://example.com/registration');

cy.get('#fullName').type('Ali Hamza');

cy.get('#email').type('alihamza20f1112@gmail.com');

cy.get('#password').type('Abcd@1234');

cy.get('#confirmPassword').type('Abcd@1234');

cy.get('#dob').type('2001-10-04');

cy.get('#gender').select('Male');

cy.get('#newsletter').check('Yes');

cy.get('#submit').click();

cy.url().should('include', '/welcome');

cy.contains('Registration successful').should('be.visible');

});

});

1. **Form Submission without Newsletter Subscription**

describe('Registration Form', () => {

it('should register successfully with valid data', () => {

cy.visit('http://example.com/registration');

cy.get('#fullName').type('Ali Hamza');

cy.get('#email').type('alihamza20f1112@gmail.com');

cy.get('#password').type('Abcd@1234');

cy.get('#confirmPassword').type('Abcd@1234');

cy.get('#dob').type('2001-10-04');

cy.get('#gender').select('Male');

cy.get('#newsletter').check('No');

cy.get('#submit').click();

cy.url().should('include', '/welcome');

cy.contains('Registration successful').should('be.visible');

});

});

## Additional assumptions

1. The website URL for the registration form is assumed to be http://example.com/registration.
2. Upcoming Welcome page.