**Introduction:**

The purpose of this document is to outline the test plan for the online registration form on the website. This plan will detail the objectives, scope, environment, data, scenarios, schedule, risks, and assumptions involved in testing the registration form.

**Objectives:**

The main objectives of this test plan are to:

* Verify that the registration form functions as expected.
* Ensure that all input validations are working correctly.
* Confirm that data is successfully submitted to the server upon correct completion.
* Identify and document any defects or issues.

**Scope**

The scope of this test plan includes:

* Functional testing of the registration form.
* Validation of input fields (Full Name, Email Address, Password, Confirm Password, Date of Birth, Gender, Newsletter Subscription).
* Verification of error messages.
* Submission and data handling.

**Test Environment**

* OS: Windows 10
* Browsers: Chrome, Firefox
* Tools: Selenium WebDriver, Cypress

**Test data**  
Test data will include various combinations of valid and invalid inputs for each field:

* Full Name: muhammad bilal, shaikh, bunty
* Email Address: valid@example.com, invalid-email, user@domain
* Password: correctPassword1, short, differentPassword1
* Date of Birth: 01/01/1990, invalid-date
* Gender: Male, Female, Other
* Newsletter Subscription: Yes, No

**Test Scenarios**

1. Positive Scenarios
   * Successful registration with all valid inputs.
   * Successful registration with optional fields left empty.
2. Negative Scenarios
   * Registration with invalid email format.
   * Registration with mismatched passwords.

**Test Execution Schedule**

* Test Preparation: 1 week
* Test Execution: 2 weeks
* Defect Reporting: Ongoing during execution
* Test Closure: 1 week after execution

**Risks and Assumptions**

* Risks: Changes in requirements, delay in environment setup, unavailability of test data.
* Assumptions: Requirements are stable, test environment is available, test data is prepared.

**Test Cases**

**Test Case 1**

**Test Case ID:** TC001  
**Test Case Description:** Successful registration with all valid inputs.  
**Preconditions:** User is on the registration form page.  
**Test Steps:**

1. Enter "muhammad bilal" in the Full Name field.
2. Enter "valid@example.com" in the Email Address field.
3. Enter "correctPassword1" in the Password field.
4. Enter "correctPassword1" in the Confirm Password field.
5. Enter "01/01/1990" in the Date of Birth field.
6. Select "Male" for Gender.
7. Select "Yes" for Newsletter Subscription.
8. Click the Submit button. **Expected Results:**

* The form submits successfully.
* User is redirected to the welcome page. **Post-conditions:** User account is created in the system.

**Test Case 2**

**Test Case ID:** TC002  
**Test Case Description:** Registration with invalid email format.  
**Preconditions:** User is on the registration form page.  
**Test Steps:**

1. Enter "muhammad bilal" in the Full Name field.
2. Enter "invalid-email" in the Email Address field.
3. Enter "correctPassword1" in the Password field.
4. Enter "correctPassword1" in the Confirm Password field.
5. Enter "01/01/1990" in the Date of Birth field.
6. Select "Male" for Gender.
7. Select "Yes" for Newsletter Subscription.
8. Click the Submit button. **Expected Results:**

* Error message "Invalid email address" is displayed.
* Form is not submitted. **Post-conditions:** User remains on the registration form page.

**Test Case 3**

**Test Case ID:** TC003  
**Test Case Description:** Registration with mismatched passwords.  
**Preconditions:** User is on the registration form page.  
**Test Steps:**

1. Enter "muhammad bilal" in the Full Name field.
2. Enter "valid@example.com" in the Email Address field.
3. Enter "correctPassword1" in the Password field.
4. Enter "differentPassword1" in the Confirm Password field.
5. Enter "01/01/1990" in the Date of Birth field.
6. Select "Male" for Gender.
7. Select "Yes" for Newsletter Subscription.
8. Click the Submit button. **Expected Results:**

* Error message "Passwords do not match" is displayed.
* Form is not submitted. **Post-conditions:** User remains on the registration form page.

**Scenario Identification**

**Positive Scenarios**

1. **Successful Registration with All Valid Inputs**
   * User fills in all fields with valid data and submits the form.
   * Expected Outcome: Form submits successfully and user is redirected to the welcome page.
2. **Successful Registration with Optional Fields Left Empty**
   * User fills in mandatory fields with valid data, leaves optional fields (e.g., Newsletter Subscription) empty, and submits the form.
   * Expected Outcome: Form submits successfully and user is redirected to the welcome page.

**Negative Scenarios**

1. **Registration with Invalid Email Format**
   * User enters an invalid email format in the Email Address field and submits the form.
   * Expected Outcome: Error message "Invalid email address" is displayed, and the form is not submitted.
2. **Registration with Mismatched Passwords**
   * User enters different passwords in the Password and Confirm Password fields and submits the form.
   * Expected Outcome: Error message "Passwords do not match" is displayed, and the form is not submitted.

**Automated Test Scripts**

Using Cypress

**Script for Successful Registration with All Valid Inputs:**

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

it('Successful registration with all valid inputs', () => {

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

cy.get('input[name="fullName"]').type('muhammad bilal');

cy.get('input[name="email"]').type('valid@example.com');

cy.get('input[name="password"]').type('correctPassword1');

cy.get('input[name="confirmPassword"]').type('correctPassword1');

cy.get('input[name="dob"]').type('1990-01-01');

cy.get('select[name="gender"]').select('Male');

cy.get('input[name="newsletter"][value="Yes"]').check();

cy.get('button[type="submit"]').click();

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

});

});)

**Script for Successful Registration with Optional Fields Left Empty:**

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

it('Successful registration with optional fields left empty', () => {

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

cy.get('input[name="fullName"]').type('muhammad bilal');

cy.get('input[name="email"]').type('valid@example.com');

cy.get('input[name="password"]').type('correctPassword1');

cy.get('input[name="confirmPassword"]').type('correctPassword1');

cy.get('input[name="dob"]').type('1990-01-01');

cy.get('select[name="gender"]').select('Female');

cy.get('button[type="submit"]').click();

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

});

});)

**Script for Registration with Invalid Email Format:**

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

it('Registration with invalid email format', () => {

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

cy.get('input[name="fullName"]').type('John Doe');

cy.get('input[name="email"]').type('invalid-email');

cy.get('input[name="password"]').type('correctPassword1');

cy.get('input[name="confirmPassword"]').type('correctPassword1');

cy.get('input[name="dob"]').type('1990-01-01');

cy.get('select[name="gender"]').select('Male');

cy.get('input[name="newsletter"][value="Yes"]').check();

cy.get('button[type="submit"]').click();

cy.get('.error-message').should('contain', 'Invalid email address');

});

});**)**

It covers the essential aspects of testing the registration form, including the test plan outline, detailed test cases, scenario identification, and automated test scripts.