Test Plan for Online Registration Form

JULY 14, 2024
MUHAMMAD UMAR KAREEM
Umar1671594@gmail.com

Table of Contents

Introduction

This test plan provides the approach, goals, coverage, assets, time frame, and expected outcomes of testing the online registration form of our website. The objective of this test plan is to check whether the registration form is working as it should and whether it is compliant with the set standards. The form is an important part of the website since it contains information about the user that is vital when creating user accounts and granting access to different parts of the site. This document will help in testing to detect the defects, quality assurance and to check whether the form functions properly in different circumstances.

Objectives

It is also important to clarify the primary goals of this test plan, which are to verify that the online registration form fulfills all the set objectives. I want to ensure that every form field of the Full Name, Email Address, Password of at least 8 characters, Confirm Password, Date of Birth, Gender (Male/Female/Other), and for Newsletter Subscription it should have Yes or No button. Besides, I will have to ensure that every functionality associated with the Submit Button works as expected. Another, which is equally important, is to check the entire users' experience which must be satisfactory together with the interface's adaptability to various devices and browsers. I will also ensure that user data is safe and accurate specifically, I will ensure that data is not altered in any way when being inputted. In addition, I will closely evaluate the form's functionality and solidity to ensure it will not be compromised in this aspect and will be able to work efficiently regardless of situations. Last but not least, it will be pertinent to meet the legal requirements concerning data protection and accessibility standards to guarantee the effectiveness, safety, and usability of the registration form.

Scope

The scope of this test plan includes a comprehensive evaluation of the online registration form to ensure it meets all specified requirements. The form fields to be tested are:

- Full Name
- Email Address
- Password (minimum of 8 characters)
- Confirm Password
- Date of Birth
- Gender (Male/Female/Other)
- Newsletter Subscription (Yes/No)
- Submit Button

Functional Testing

- Verification of correct data capture and validation for all form fields.
- Ensure the Submit Button works correctly, including form submission and reset functionality.
- Validate error messages and feedback for invalid inputs and edge cases.

User Interface Testing

- Assessment of the form's layout, look and feel, as well as its functionality on different devices: desktop, tablet and mobile as well as different browsers such as google chrome, firefox, safari, edge.
- The form should be understudied and easy to comprehend by the intended users.

Security Testing

- Assessment of input validation to prevent security vulnerabilities such as SQL injection and cross-site scripting (XSS).
- Verification of the encryption and secure handling of sensitive data, especially passwords.
- Ensure secure data transmission over HTTPS.

Performance Testing

- Testing the form's response time and stability under different levels of user traffic.
- Ensure the form can handle multiple submissions simultaneously without performance degradation.

Usability Testing

- Evaluation of the user experience, including ease of use and clarity of instructions.
- Ensure the form meets user expectations and provides a satisfactory user experience.

Test Environment

The test environment will include:

- Operating Systems: Windows 10, macOS, Linux
- Browsers: Chrome (latest version), Firefox (latest version), Safari (latest version), Edge (latest version)
- Devices: Desktop, Tablet, Mobile
- Java Development Kit (JDK) version 11
- JUnit 5 framework for test automation
- Integrated Development Environment (IDE): IntelliJ IDEA or Eclipse
- Web server for hosting the registration form application
- Database server (MySQL or PostgreSQL) for data storage

Requirements for form data

Full Name

1. **Requirement:** The Full Name field must accept alphabetic characters, spaces, and common punctuation marks such as hyphens and apostrophes.
2. **Validation:**
    o   Must be at least 1 character long.
    o   Maximum length: 50 characters.
    o   Invalid characters: numbers and special characters (except hyphens and apostrophes).

Email Address

1. **Requirement:** The Email Address field must accept a valid email address format.
2. **Validation:**
    o   Must contain at least one "@" character.
    o   Must contain at least one "." character after the "@" character.
    o   The domain part (after "@") must be valid, e.g., "example.com".
    o   Must not start or end with special characters.
    o   Minimum length: 5 characters (e.g., "a@b.c").
    o   Maximum length: 50 characters.

Password (at least 8 characters)

1. **Requirement:** The Password field must accept a string of at least 8 characters, including at least one uppercase letter, one lowercase letter, one digit, and one special character.
2. **Validation:**
    o   Minimum length: 8 characters.
    o   Maximum length: 50 characters.
    o   Must contain at least one uppercase letter.
    o   Must contain at least one lowercase letter.
    o   Must contain at least one digit.
    o   Must contain at least one special character (e.g., @, #, $, etc.).

Confirm Password

1. **Requirement:** The Confirm Password field must match the Password field exactly.
2. **Validation:**
    o   Must be identical to the Password field.

Date of Birth

1. **Requirement:** The Date of Birth field must accept a valid date format and ensure the user is at least 18 years old but not older than 120 years.
2. **Validation:**
   o   Must be in a valid date format (e.g., DD-MM-YYYY).
   o   Must not be a future date.
   o   Must be a valid date (consider leap years).
   o   User must be at least 18 years old and not older than 120 years.

Gender (Male/Female/Other)

1. **Requirement:** The Gender field must accept one of the predefined values: Male, Female, or Other.
2. **Validation:**
   o   Must be one of the following values: "Male", "Female", "Other".
   o   Any other input is invalid.

Newsletter Subscription (Yes/No)

1. **Requirement:** The Newsletter Subscription field must accept a Yes or No response indicating whether the user wants to subscribe to the newsletter.
2. **Validation:**
   o   Must be either "Yes" or "No".
   o   Any other input is invalid.

Submit Button

1. **Requirement:** The Submit Button must trigger form validation and submit the data to the server if all fields are valid.
2. **Validation:**
   o   Must be enabled only when all required fields are filled with valid data.
   o   Must display appropriate error messages for invalid fields.
   o   Must submit the form data to the server upon clicking if all validations pass

Boundary Value Analysis

Field	BVA
Full Name	BVA applies on the range of the Full Name i.e., it should be 3 >= length <= 50 BVA on Length:

Value	Length	Test ID	Result
Mr	2	1001	**Fail**
jon	3	1002	**Pass**
Umair	5	1003	**Pass**
Harllow	7	1004	**Pass**
Umar Kareem	11	1005	**Pass**

| | | A very very very very very very very very long name | 50 | 1006 | **Pass** |
| | | Sheikh Muhammad Abubakr bint-e-abdul aziz bint ahmed ul kareem | 57 | 1007 | **Fail** |

| **Email** | BVA applies on the range of the Email i.e., it should be 5>= length <= 50 |
| | BVA on Length: |

Value	Length	Test ID	Result
k@m	3	1008	Fail
s@a.am	6	1009	Pass
mi@mi.com	9	1010	Pass
ha@kare.com	11	1011	Pass
a.very.long.email.address@example.com	50	1012	Pass
a.very.long.email.address@toolongdomain.com	51	1013	Fail

| **Password** | BVA applies on the range of the password i.e., it should be 8 >= length <= 50 |
| | BVA on Length: |

Value	Length	Test ID	Result
A	1	1019	Fail
eighteen	8	1020	Pass
eeeeeeeee	11	1021	Pass
UmarTesterBest	13	1022	Pass
BestOfTheBestest	16	1023	Pass
Averysecurepasswordthatexactlyfiftycharacterslong!	50	1024	Pass
Averysecurepasswordthatexactlyfiftyonecharacterslong	51	1025	Fail

Test Execution Schedule

Task	Start Date	End Date
Test Planning	2024-07-15	2024-07-17
Test Case Design	2024-07-15	2024-07-17
Test Environment Setup	2024-07-15	2024-07-17
Test Execution	2024-07-15	2024-07-17
Bug Reporting and Fixing	2024-07-15	2024-07-17
Test Reporting	2024-07-15	2024-07-17

Test Cases

Full Name

Test ID: 1001	Test Name: Full Name Validation
Description: BVA testing of Full Name field for length 0.	
Pre-Condition: Form is Loaded	
Post-Condition: Error is Displayed	
Inputs: 1. *"Name": "Mr"* 2. *"Email":<u>Umar@Umar.com</u>* 3. *"Password": "Password@1233"* 4. *"Confirm Password": "Password@1233"* 5. *"Date of Birth": "19/02/2002"* 6. *"Gender": "Male"* 7. *"Newspaper Subscription": "NO"*	
Operation Procedure: 1. *Open form* 2. *Enter "Mr" in the Full Name field.* 3. *Enter "Umar@Umar.com" in the Email Address field.* 4. *Enter "Password@1233" in the Password field.* 5. *Enter "Password@1233" in the Confirm Password field.* 6. *Enter "19/02/2002" in the Date of Birth field.* 7. *Select "Male" for Gender.* 8. *Select "NO" for Newsletter Subscription.* 9. *Click the Submit button.*	
Expected Output: Error Message.	**Actual Output: (Error)** Full Name is too small
Result: Fail	

Email

Test ID: 1009	Test Name: Email Validation
Description: BVA testing of email with valid length.	
Pre-Condition: Form is Loaded	
Post-Condition: Success Message is Displayed	
Inputs: 1. *"Name": "Umar"* 2. *"Email": "s@a.am"* 3. *"Password": "Password@1233"* 4. *"Confirm Password": ""Password@1233"* 5. *"Date of Birth": "19/02/2002"*	

6. *"Gender": "Male"*
7. *"Newspaper Subscription": "NO"*

Operation Procedure:
1. *Open form*
2. *Enter "Umar" in the Full Name field.*
3. *Enter "s@a.am" in the Email Address field.*
4. *Enter "Password@1233" in the Password field.*
5. *Enter "Password@1233" in the Confirm Password field.*
6. *Enter "19/02/2002" in the Date of Birth field.*
7. *Select "Male" for Gender.*
8. *Select "NO" for Newsletter Subscription.*
9. *Click the Submit button.*

Expected Output: Success Message	**Actual Output:** (Success) Your form has been successfully submitted!
Result: Pass	

Password

Test ID: 1022	**Test Name:** Password Validation
Description: BVA testing of Password with valid length.	
Pre-Condition: Form is Loaded	
Post-Condition: Success Message is Displayed	
Inputs: 1. *"Name": "Umar"* 2. *"Email": "s@a.am"* 3. *"Password": "UmarTesterBest"* 4. *"Confirm Password": "UmarTesterBest"* 5. *"Date of Birth": "19/02/2002"* 6. *"Gender": "Male"* 7. *"Newspaper Subscription": "NO"*	
Operation Procedure: 1. *Open form* 2. *Enter "Umar" in the Full Name field.* 3. *Enter "s@a.am" in the Email Address field.* 4. *Enter "UmarTesterBest" in the Password field.*	

5. *Enter "UmarTesterBest" in the Confirm Password field.* 6. *Enter "19/02/2002" in the Date of Birth field.* 7. *Select "Male" for Gender.* 8. *Select "NO" for Newsletter Subscription.* 9. *Click the Submit button.*	
Expected Output: Success Message	**Actual Output:** (Success) Your form has been successfully submitted!
Result: Pass	

Scenario Identificaton

Positive Scenarios

1. **Valid Registration:** The user goes to the registration form and provides correct information for all the fields which include Full Name as "Muhammad Umar", Email as "Umar@Umar. com", Password as "Password@1234" and Confirm Password as "Password@1234", Date of Birth as "02/19/2002", Gender as "Male" and subscribes to the Newsletter. When the form is submitted by clicking Submit button, the form is successfully submitted and a success message appears.

2. **Email Validation - Valid Email:** The user goes to the registration form, enters "Muhammad Umar" for Full Name, "Umar1671594@gmail. com" for Email, "Password@1234" for Password and Confirm Password, selects "02/9/2002" for Date of Birth, selects "Male" for Gender and unchecks Newsletter Subscription. When the Submit button is clicked, the form is submitted and a success message is shown.

Negative Scenarios

1. **Password Mismatch:** The user gets to the registration form, fills Full Name with "Dua Fatima", Email with "Dua.Fatima@example.com", Password with "Password@1234", Confirm Password with "Password@123", Date of Birth with "12/12/2000", Gender with "Female", and subscribes to the Newsletter. When the user clicks on Submit, the form returns the error message stating that the passwords do not match and does not submit the form.

2. **Invalid Date of Birth:** The user goes to the registration form and fills Full Name as "Munahil amin", Email as "Munahil@gmail.com", Password and Confirm Password as "Password@1234", Date of Birth as "33/02/1990", Gender as "Female" and uncheck the Newsletter Subscription. The form does not submit when the Date of Birth is invalid and an error message appears after clicking Submit.

Risks and Assumptions

Risks

- Delays in setting up the test environment may impact the schedule.
- Unavailability of key testing resources could hinder test execution.
- Potential for undetected defects due to insufficient test coverage.

Assumptions

- All required resources and tools will be available as scheduled.
- Test data will be prepared and validated before test execution begins.
- The development team will address reported bugs promptly.

Test Cases

Test Id: 1009

```java
@Test
void testEmailValidation_ValidLength() {
  // Test ID: 1009
  // Description: BVA testing of email with valid length.
  form.setFullName("Umar");
  form.setEmail("s@a.am");
  form.setPassword("Password@1233");
  form.setConfirmPassword("Password@1233");
  form.setDateOfBirth("19/02/2002");
  form.setGender("Male");
  form.setNewsletterSubscription("NO");

  String result = form.submit();
  assertEquals("Success: Your form has been successfully submitted!", result);
}
```

Test Id: 1022

```java
@Test
  void testPasswordValidation_ValidLength() {
    // Test ID: 1022
    // Description: BVA testing of Password with valid length.
    form.setFullName("Umar");
    form.setEmail("s@a.am");
    form.setPassword("UmarTesterBest");
    form.setConfirmPassword("UmarTesterBest");
    form.setDateOfBirth("19/02/2002");
    form.setGender("Male");
    form.setNewsletterSubscription("NO");
```

```java
        String result = form.submit();
        assertEquals("Success: Your form has been successfully submitted!", result);
    }
}
```