Warda Saleem

Senior QA ENGINEER

CARE VALIDATE

QA Test plan

Table of Contents

[Introduction 2](#_Toc173876818)

[Test Plan Objective 2](#_Toc173876819)

[Testing Types 2](#_Toc173876820)

[Testing Environment 2](#_Toc173876821)

[Test Cases 3](#_Toc173876822)

[Login 3](#_Toc173876823)

[New Request 3](#_Toc173876824)

[Session Management 4](#_Toc173876825)

[User Interaction 4](#_Toc173876826)

[Defect Management 5](#_Toc173876827)

[Defect Identification 5](#_Toc173876828)

[Defect Logging 6](#_Toc173876829)

[Defect Triage 6](#_Toc173876830)

[Defect Assignment 6](#_Toc173876831)

[Defect Resolution 6](#_Toc173876832)

[Defect Verification 6](#_Toc173876833)

[Defect Closure 6](#_Toc173876834)

# Introduction

This test plan outlines the strategies, objectives, and procedures for testing “Care Validate” web portal. The goal is to ensure the website meets quality standards, functions correctly, and provides a positive user experience. This document will cover test scope, test cases, test environment, and defect management processes.

# Test Plan Objective

The following are the objective of Test Plan:

* Identifying defects and ensuring quality.
* Verifying compliance with requirements and specifications.
* Ensuring website performance and usability.

# Testing Types

* Functional testing (verifying website functionality)
* Usability testing (evaluating user experience)
* Performance testing (assessing website speed and responsiveness)
* Compatibility testing (checking compatibility across browsers and devices)
* Accessibility testing (ensuring accessibility for users with disabilities)

# Testing Environment

All the testing will be performed in isolated Test environment with specific test data.

# Test Cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Login** | | | |
| Item | Test Case | Expected Result | Result (Pass/Fail/NA) |
|  | Verify that user can login into the website using correct email and password | User should be able to login successfully | Passed |
|  | Verify that user cannot login into the website with invalid username/password | Error message should display “Invalid email or password” | Passed |
|  | Login with correct email and incorrect password. | User should not be able to login | Passed |
|  | Login with empty email and password fields. | User should not be able to login | Passed |
|  | Login with email only (password field empty). | User should not be able to login | Passed |
|  | Login with password only (email field empty). | User should not be able to login | Passed |
|  | Login with multiple consecutive failed attempts (check for lockout mechanism). | User should logged-out | Failed |
|  | Verify clear and concise labels for email and password fields. | Label should be correct | Passed |

|  |  |  |  |
| --- | --- | --- | --- |
| **New Request** | | | |
| Item | Test Case | Expected Result | Result (Pass/Fail/NA) |
|  | Verify that user can view all the request forms in “New Request” screen | Forms should display in the list | Passed |
|  | Verify that when user click on any available request form, it should get open for entering the details | Complete form should open | Passed |
|  | Verify that user can add values in all the fields of the request form | Data should be added in all the fields | Passed |
|  | Verify that user can “Skip” fields which are optional | Field should be skipped | Passed |
|  | Verify that user can “Submit” form once all the required fields are filled | Form should be submitted | Passed |
|  | Verify that user can save the draft of the request form on clicking “Save & Close” button | Draft should be saved | Passed |
|  | Verify that user can open the draft to view the previous added data and complete the request form | Data should retain in the draft | Passed |
|  | Verify that user can click on delete icon to delete request form | Draft should be deleted | Passed |

|  |  |  |  |
| --- | --- | --- | --- |
| **Session Management** | | | |
| Item | Test Case | Expected Result | Result (Pass/Fail/NA) |
|  | Verify that if all the tabs/browser are closed, user should be logged-out automatically | User session should be ended | Failed |
|  | Verify that session ID is not exposed in the URL | Session ID should not display in URL | Passed |
|  | Verify that user session is expired after a period of inactivity | User should logged-out from the portal | Failed |

|  |  |  |  |
| --- | --- | --- | --- |
| **User Interaction** | | | |
| Item | Test Case | Expected Result | Result (Pass/Fail/NA) |
|  | Verify correct navigation between different pages and sections. | Navigation should be correct | Passed |
|  | Test the functionality of menus, dropdowns, and breadcrumbs. | Functionality should be correct | Passed |
|  | Check for clear and visible navigation elements. | Nav elements should be clear and visible | Passed |
|  | Verify backward and forward navigation functionality. | Navigation should work fine | Passed |
|  | Test accessibility of navigation elements for users with disabilities. | Accessibility navigation should work correctly | Passed |
|  | Verify correct data types and formats for input fields. | Data types and formats should be correct | Passed |
|  | Test input field validation (required fields, length, format, etc.). | Validation should be correct | Passed |
|  | Check for error messages and feedback on invalid input. | Correct error messages should display | Passed |
|  | Verify input field accessibility (keyboard navigation, screen reader compatibility). | Accessibility should work properly | Passed |
|  | Verify button and link functionality. | Buttons and links should work | Passed |
|  | Test form submission and error handling. | Submission and error handling should work fine | Passed |
|  | Test form behavior under different network conditions. | Form should behave correctly under low internet connectivity | Passed |
|  | Check for consistent error handling across the application. | Proper consistent error handling should perform | Passed |

# Defect Management

A robust defect management process is essential for ensuring software quality. This process involves the identification, logging, tracking, prioritization, resolution, and closure of defects. The following steps outline the general defect management process:

Defect Identification

Defects are discovered during testing or by other stakeholders.

Defect Logging

Defects are documented in a defect tracking tool with detailed information such as description, steps to reproduce, expected and actual results, severity, and priority.

### Defect Triage

Defects are assessed to determine their severity, priority, and assignment to responsible teams.

### Defect Assignment

Defects are assigned to developers or other responsible parties for resolution.

### Defect Resolution

Developers fix the defects based on the provided information.

### Defect Verification

Testers verify if the defect has been fixed correctly.

### Defect Closure

If the defect is resolved, it is closed; otherwise, it is reopened for further investigation.