



Test Plan Document

Project: Test Plan of MeDirect E-Wealth page

Version: 1.0 (Initial Draft)

Created By: Dilan Meegoda

Date: 18th March 2025

1 Introduction

1.1 Project Overview

About MeDirect Bank:

MeDirect is a digital financial institute that provides investments and banking services. The bank enables its customers to manage and grow their financial assets online.

1.2 Scope of the Testing

- This test plan focuses on validating the [Equities Search Page](#) functionalities on MeDirect website
- The page provides navigation options for Funds, Equities, ETFs, and Bonds, but test will only focus on the Equities search functionality.
- Other search sections (Funds, EFTs and Bonds) dynamically update the URL and page body content and functioning as separate pages are out of scope.
- Open account and Become a Customer Button on Equities Search page:
 - These buttons are present on the page and redirect users to external registration or login pages.
 - This redirection function will be verified in current test scope but further functionality is beyond that is out of scope.
- Header and Footer navigation:
 - Since the header and footer are common across MeDirect website, this test plan will perform a basic validation to ensure navigation links are functional.

1.3 Test Objectives

The objective of this test plan is to validate the functionality, usability, and performance of the Invest in Equities page on MeDirect website. The test will ensure that:

1. User can successfully perform a search for equities by name, ISIN or ticker.
2. Generated search results are accurate, relevant and results appeared in a reasonable response time.
3. UI elements of the page function correctly and align with expected behavior.
4. The “Open Account” and “Become a Customer” buttons are function as expected, user redirect correctly to their respective pages.
5. The page loads successfully and maintain its UI across mobile devices and browsers.
6. Navigation links of the header and footer are clickable and functional. Since these header and footer are common across site wise, they are not primarily focus for testing.
7. Identify functional defects and usability improvements then document them for bug fixes and future system enhancements.

2 Test Strategy

2.1 Testing Approach

The testing approach of the [Equities Search Page](#) will focus on manual testing, ensuring the functionality, usability and performance of the MeDirect website. Test approach includes:

- Functional testing to verify Equities search and interaction with search results.
- UI testing to validate the correct display of UI elements of the page.
- Data validation and verification:
 - Verify the required data is received after search action.
 - Verify the accuracy and relevance of the search results.
- Functional testing to verify button actions:
 - Verifying the Open Account and Become a customer buttons are displayed correctly.
 - Verify button redirect to correct redirection pages.
- Basic validation of header and footer navigation as they are common across all pages.
- Page performance validation:
 - Verify the search results are returned within defined threshold limit.
 - Validate that Equities search page load time is acceptable according to threshold value on different browsers and devices.

2.2 Test Levels

Test Levels define the types of testing to be executed on the [Equities Search Page](#).

2.2.1 Functional Testing

Purpose: Functional testing will be performed to check the core functionality of the Equities Search page to ensure that user can complete successful search for equities and interact with the results.

Scope:

- Ensure that search queries / search terms return valid and relevant results.
- Validate search bar and button functionalities.
- Verify the accuracy of the search results.
- Ensure that “Open account” and “Become a Customer” buttons are displayed and redirect correctly.

Method:

- Manual execution of detailed functional test cases.
- Test result comparison with expected and actual results.

2.2.2 UI Testing

Purpose: Ensure all the UI components of the page are displayed correctly.

Scope:

- Validate the correct alignment and visibility of UI elements.
- Check the page responsiveness with different screen resolutions.
- Validate all the buttons, icons and labels according to the approved wireframes.

Method:

- Visually verification against approved wireframes and mockups.
- Cross browser testing.

2.2.3 Data Validation Testing

Purpose: Validate the accuracy and integrity of data display in search results.

Scope:

- Validate that search data is relevant and valid.

- Validate that there is no any missing data related to search term.

Method:

- Manual verification of the search result data
- Cross check with database record with frontend displayed data

2.2.4 Page Performance Testing

Purpose: Ensure that page load time and search response time are within acceptable limits.

Scope:

- Validate that the Equities search page loads within the expected limit.
- Validate that the search results are returned within an acceptable time limit.

Method:

- Manually record the response times for different test scenarios.
- Compare the actual result records with defined threshold values.

2.3 Test Techniques

To ensure testing of the Equities search page following test techniques will be used:

- Functional Testing Techniques – Boundary Value Analysis, Equivalence Partition, State Transition Test, Error Handling verification
- Performance Testing Techniques – Load Testing, Page Load Time Validations, Search Execution Time validation.

2.4 Test Tools

- Manual Functional Validation - Browser Dev Tool
- Test Management – Excel, Confluence
- Defect Tracking – Jira
- Performance Testing – JMeter, Browser Dev Tool

3 Test Deliverables

The following test artifacts will be delivered after completion of the Software Testing Life Cycle (STLC)

1. **Test plan document** - Outline the scope, objective, test strategy and test approach.
2. **Test strategy document** - Contains the level of testing, testing techniques and tools that used.
3. **Defect report template** - A structured template for reporting issues and improvements.
4. **Test case document** - A set of test cases covering search functionality, UI and performance.
5. **Issues and improvements of search functionality** - A report contains summary if the defects, usability concerns and suggested improvements.

4 Test Completeness

In order to ensure that the testing process is completed and achieved defined test objectives, following test exit citraturias will be considered:

1. All the test cases of the Equities Search Page including functional, UI and performance tests are executed.

2. Identified defects and improvements are logged and documented in the Jira.
3. All blocker, critical and high severity issues are fixed, retested and verified from QA side.
4. The search functionality should return accurate, relevant results.
5. The page load performance is within the acceptable threshold values.

5 Defect Report Template

1. Defect Summary

- **Defect ID:** A unique identification for the defect.
- **Title:** A short, clear summary of the issue.
- **Project/ Module Name:** Relevant project or the module that defect associated with.
- **Issue Type:** Bug or improvement.
- **Status:** Open, In Progress, Resolved, Reopened, Closed
- **Priority:** Low, Medium, High, Critical, Blocker
- **Severity:** Minor, Major, Feature, Blocker
- **Detailed Description:** A comprehensive explanation of the defect.

2. Environment Details

- **Environment:** QA, Staging, Production
- **Planform and OS:** Windows, iOS, Android
- **Browser and version:** Chrome Latest, Firefox Latest

3. Steps to Reproduce

- Required steps to recreate the issue for debugging purposes

4. Expected Results

- The results that should happens if the functionality work accordingly.

5. Actual Results

- The actual results that observed.

6. Screenshots/Attachments

- Log file, screenshot or screen capture video of the issue.

7. Affected Versions

- The version of the build that issue identified.

8. Additional Information

- **Assignee:** The responsible developer for the fix.
- **Reporter:** QA Engineer or Tester who logged the defect record.
- **Additional Notes:** Any informative information related to the issue like occurrence rate