**Test Plan for FinanceMan Dashboard**

#### **1. Test Plan Identifier**

* **TP-001**: Test Plan for FinanceMan Dashboard Testing

### **2. Test Plan Overview**

#### This test plan describes the testing strategy for the Finance Dashboard application hosted at [**https://finance-man.netlify.app/dashboard**](https://finance-man.netlify.app/dashboard)**.** The purpose of this plan is to ensure the functionality, reliability, and usability of the Finance Dashboard. The primary objective is to validate that all essential features work as expected across different use cases.

#### **3. Test Plan Objectives**

* Verify that the dashboard loads without errors.
* Ensure that the displayed financial data is correct and matches user expectations.
* Test the functionality of the sidebar, search, pagination, and logout mechanisms.
* Validate the responsiveness of the dashboard across different devices.
* Ensure error handling is present and displays appropriate messages when invalid data is entered.

#### **4. Features to Be Tested**

* **Dashboard Page Load:**
  + Ensure that the page loads successfully and displays without errors.
* **Header:**
  + Verify the header displays the correct branding and page title.
* **Sidebar Menu:**
  + Ensure the sidebar menu items (e.g., Dashboard, Transactions, Settings) are visible and functional.
* **Financial Data:**
  + Validate that financial data such as balance, income, and expenses are displayed correctly.
* **Transaction History:**
  + Ensure that transaction history is shown and populated with accurate data (e.g., amounts, dates, transaction types).
* **Search Functionality:**
  + Test that the search bar filters the transaction history correctly based on keywords.
* **Pagination:**
  + Ensure the pagination controls work correctly, showing 10 items per page and navigating through the pages.
* **Logout Functionality:**
  + Verify that users can log out successfully and are redirected to the login page.
* **Responsiveness:**
  + Ensure that the dashboard layout adjusts appropriately on mobile, tablet, and desktop screens.
* **Error Handling:**
  + Validate the system's error handling when submitting invalid data (e.g., empty fields or negative values).

#### **5. Scope of Testing**

#### **In-Scope:**

* Functionality of the dashboard, sidebar, financial data, transaction history, search, pagination, and logout.
* Responsiveness of the application across different screen sizes (mobile, tablet, desktop).
* Error handling for invalid input scenarios.

#### 

#### **Out-of-Scope:**

* Any testing related to server-side performance.
* Stress or load testing.
* Non-functional testing other than responsiveness.

#### **6. Test Deliverables**

* **Test Cases**: Detailed test cases outlining the functional and negative scenarios.
* **Test Data**: Example user data, transactions, and invalid input data.
* **Test Execution Report**: Report that includes the actual results of all test cases.
* **Defect Report**: If defects are found, they will be documented with severity and steps to reproduce.

#### **7. Testing Strategy**

The testing strategy will consist of:

1. **Manual Testing**: Since the application is a web-based dashboard, testing will primarily be done manually to verify the functionality of UI elements and overall user experience.
2. **Cross-Browser Testing**: Testing across common browsers (Chrome, Firefox, Edge) to ensure compatibility.
3. **Responsiveness Testing**: Testing on mobile, tablet, and desktop devices to verify that the dashboard layout is responsive.
4. **Regression Testing**: After any bug fixes or enhancements, regression testing will be performed to ensure that existing functionality remains unaffected.

#### 

#### 

#### 

#### **8. Test Schedule**

| **Test Phase** | **Start Date** | **End Date** | **Duration** |
| --- | --- | --- | --- |
| Test Planning | 2024-12-13 | 2024-12-14 | 2 days |
| Test Case Creation | 2024-12-14 | 2024-12-15 | 2 days |
| Test Execution (Manual) | 2024-12-16 | 2024-12-18 | 3 days |
| Test Reporting | 2024-12-19 | 2024-12-20 | 2 days |

#### **9. Testing Tools**

**Browser**: Google Chrome, Mozilla Firefox, Microsoft Edge

**Test Management Tool**: TestRail, or any test management software for tracking test cases and results.

**Bug Tracking Tool**: Jira or any project management tool for tracking and reporting defects.

#### **10. Test Environment Setup**

**Test Environment URL**:<https://finance-man.netlify.app/dashboard>

**Browser Versions**: Latest versions of Chrome, Firefox, and Edge.

**Devices for Responsiveness Testing**:

* Desktop
* Tablet
* Mobile

#### **11. Entry and Exit Criteria**

#### **Entry Criteria:**

* The application is deployed on the test server.
* Test cases are prepared and approved.
* Test data is available for testing.

#### 

#### 

#### **Exit Criteria:**

* All planned test cases have been executed.
* All high-priority defects are resolved, or workarounds are provided.
* The test execution report and defect report are submitted.

#### **12. Risk & Mitigation**

#### **Risk:**

* **Browser Compatibility**: The dashboard may have issues displaying correctly on older versions of browsers.
* **Solution**: Perform testing on the most common browsers and versions.
* **Responsiveness Issues**: The layout may not adapt correctly on all screen sizes.
* **Solution**: Perform comprehensive testing on different devices and screen resolutions.

#### **13. Approval**

| **Name** | **Role** | **Date** | **Signature** |
| --- | --- | --- | --- |
| Stephen Chitaranjan | Test Lead | 2024-12-13 | Stephen Chitaranjan |
| Biju | Project Manager | 2024-12-13 | Biju |

#### **14. Summary**

* The majority of the test cases passed successfully.
* TC011, which verifies the functionality of several buttons (Delete, Save, No Sort, Sort by Date, and Sort by Amount), failed due to bugs causing the buttons not to trigger any action or display an error message.

Please report these issues to the development team to investigate further and address the bugs in the relevant buttons.