**Test Cases Document for Audit Management Web Application**

**Table of Contents**

1. Introduction
2. Test Case Format
3. Functional Test Cases
   * User Authentication
   * Audit Planning Module
   * User Data Management
   * Source Reference
   * Reporting
   * Scheduling Audit Report
4. Non-Functional Test Cases
   * Performance
   * Security
   * Usability
5. Edge Cases
6. Conclusion

**Introduction**

This document outlines the test cases for the Audit Management Web Application. The objective is to ensure the application functions as intended and meets the specified requirements.

**Test Case Format**

Each test case will follow this format:

* **Test Case ID**: Unique identifier for the test case
* **Test Case Description**: Brief description of the test case
* **Preconditions**: Any prerequisites before executing the test
* **Test Steps**: Step-by-step instructions to execute the test
* **Expected Result**: What should happen after executing the test
* **Actual Result**: What actually happened (to be filled during testing)
* **Status**: Pass/Fail (to be filled during testing)

**Functional Test Cases**

**1. User Authentication**

**Test Case 1: User Login**

* **Test Case ID**: TC \_VALID\_LOGIN\_01
* **Description**: Verify that a user can log in with valid credentials.
* **Preconditions**: User account exists in the system.
* **Test Steps**:
  1. Navigate to the login page.
  2. Enter valid username and password.
  3. Click on the "Login" button.
* **Expected Result**: User is redirected to the dashboard.

**Test Case 2: Invalid Login**

* **Test Case ID**: TC \_INVALID\_LOGIN\_02
* **Description**: Verify that the system displays an error for invalid credentials.
* **Preconditions**: User account does not exist or incorrect credentials are provided.
* **Test Steps**:
  1. Navigate to the login page.
  2. Enter invalid username and/or password.
  3. Click on the "Login" button.
* **Expected Result**: An error message is displayed.

**2. Audit Planning Module**

**Test Case 3: Create Audit Object**

* **Test Case ID**: TC \_CREATE\_AUDIT\_OBJ\_03
* **Description**: Verify that a user can create a new audit object with source reference.
* **Preconditions**: User is logged in.
* **Test Steps**:
  1. Navigate to the "Create Audit Object Tracker" page.
  2. Fill in required fields (search source reference, event type, date).
  3. Click on the "Save" button.
* **Expected Result**: The audit object is created and listed in the Audit View.

**Test Case 4: Edit Audit Object**

* **Test Case ID**: TC \_EDIT\_AUDIT\_OBJ\_04
* **Description**: Verify that a user can edit an existing audit object.
* **Preconditions**: An audit object exists in the system.
* **Test Steps**:
  1. Navigate to the audits list.
  2. Select an audit and click on "Edit".
  3. Modify the details.
  4. Click on the "Save" button.
* **Expected Result**: The audit details are updated**.**

**3. User Data Management**

**Test Case 5: Create User**

* **Test Case ID**: TC\_CREATE\_USER\_ 05
* **Description**: Verify that an admin can create a new user.
* **Preconditions**: Admin is logged in.
* **Test Steps**:
  1. Navigate to the "Add User" page.
  2. Fill in required fields (Full Name, Email, Username, Role).
  3. Click on the "Save" button.
  4. User will get email with the default password.
* **Expected Result**: The new user is created and listed in the User View.

**Test Case 6: Edit User**

* **Test Case ID**: TC\_EDIT\_USER\_ 06
* **Description**: Verify that an admin can edit an existing user.
* **Preconditions**: Admin is logged in.
* **Test Steps**:
  1. Navigate to the users list.
  2. Select a user and click on "Edit".
  3. Modify the details.
  4. Here user can also reset password.
  5. Click on the "Save" button.
* **Expected Result**: The user details are updated**.**

**4. Source Reference**

**Test Case 7: Add Source Reference**

* **Test Case ID**: TC\_ADD\_SOURCE\_REF\_07
* **Description**: Verify that a user can add a new source reference.
* **Preconditions**: User is logged in.
* **Test Steps**:
  1. Navigate to the "Source Reference" page.
  2. Fill in required fields (Source Object name, Source Reference Key).
  3. Click on the "Save" button.
* **Expected Result**: The source reference is created and listed in the source reference view

**Test Case 8: Edit Source Reference**

* **Test Case ID**: TC\_EDIT\_SOURCE\_REF\_08
* **Description**: Verify that a user can edit an existing source reference.
* **Preconditions**: A source reference exists in the system.
* **Test Steps**:
  1. Navigate to the source reference list.
  2. Select a source reference and click on "Edit".
  3. Modify the details.
  4. Click on the "Save" button.
* **Expected Result**: The source reference details are updated**.**

**5. Reporting**

**Test Case 9: Create Reports**

* **Test Case ID**: TC\_CREATE\_REPORT\_09
* **Description**: Verify that a user can create a new Report.
* **Preconditions**: User is logged in.
* **Test Steps**:
  1. Navigate to the "Report" page.
  2. Fill in required fields (Report name, Source Reference, Start Date, End Date, Changed Users).
  3. Click on the "Save" button.
* **Expected Result**: The Report is created and listed in the Report view.

**Test Case 10: Edit Report**

* **Test Case ID**: TC\_EDIT\_REPORT\_10
* **Description**: Verify that a user can edit an existing report.
* **Preconditions**: A report already exists in the system, and the user is logged in.
* **Test Steps**:
  1. Navigate to the "Report" page.
  2. Locate an existing report and click on the "Edit" button.
  3. Modify the report details (e.g., change Report Name, Source Reference, Start Date, End Date, or Changed Users).
  4. Click on the "Save" button to update the report.
* **Expected Result**: The report details are updated and the modified report appears with the updated information in the Report view.

**6. Scheduling Audit Report**

**Test Case 11: Create Scheduling Audit Report**

* **Test Case ID**: TC\_CREATE\_SCH\_REPORT\_11
* **Description**: Verify that a user can create a new Scheduling Audit Report.
* **Preconditions**: User is logged in.
* **Test Steps**:
  1. Navigate to the "Scheduling Audit Report" page.
  2. Fill in required fields (Report name, Scheduling (Frequency, Hours , Minute, AM/PM ), Recipients).
  3. Click on the "Save Schedule" button.
* **Expected Result**: The Scheduling Report is created and listed in the view mode

**7. Exporting Audit Report**

**Test Case 12: Export Scheduled Audit Report**  
 **• Test Case ID:** TC\_EXPORT\_REPORT\_12  
 **• Description:** Verify that a user can export a scheduled report successfully.  
 **• Preconditions:** The report has been scheduled, and the user is logged in.  
   **• Test Steps:**  
     1. Navigate to the "Export Report" page.  
     2. Select the list.  
     3. Select the desired export format (e.g. XSLX, CSV) which you want to export.  
 **• Expected Result:** The report is successfully exported in the selected format on user’s Device.

**Non-Functional Test Cases**

**Performance**

**Test Case 13: Load Testing**

* **Test Case ID**: TC\_LOAD\_01
* **Description**: Verify that the application can handle 100 concurrent users.
* **Test Steps**: Use a load testing tool to simulate 100 users.
* **Expected Result**: Application response time should be under 2 seconds.

**Security**

**Test Case 14: SQL Injection**

* **Test Case ID**: TC\_SQL\_01
* **Description**: Verify that the application is secure against SQL injection.
* **Test Steps**: Attempt SQL injection in input fields.
* **Expected Result**: Application should not be vulnerable; error should be shown.

**Usability**

**Test Case 15: UI Consistency**

* **Test Case ID**: TC\_UI\_CON\_01
* **Description**: Verify UI consistency across different browsers.
* **Test Steps**: Open the application in Chrome, Firefox, and Edge.
* **Expected Result**: The UI should appear consistent across all browsers.

**Edge Cases**

**Test Case 16: Empty Fields**

* **Test Case ID**: TC\_EDGE\_01
* **Description**: Verify that the application handles empty fields appropriately.
* **Preconditions**: User is logged in.
* **Test Steps**:
  1. Navigate to the "Create Audit Object Tracker" page.
  2. Leave required fields empty and click "Save".
* **Expected Result**: Appropriate error messages are displayed.

**Test Case 17: Back Navigation After Logout**

* **Test Case ID**: TC\_EDGE\_02
* **Description**: Verify that the application does not allow access to the previous page after   
   the user has logged out.
* **Preconditions**: User is logged in.
* **Test Steps**:
  1. Log in to the application.
  2. Navigate to any page within the application.
  3. Click on the "Logout" button.
  4. After logging out, click the back button in the browser or enter the previous page URL directly.

**Expected Result: User should be redirected to the login page.  
  
  
Test Case 18: Session Timeout**

* **Test Case ID**: TC\_EDGE\_03
* **Description**: Verify session timeout functionality.
* **Preconditions**: User is logged in.
* **Test Steps**:
  1. Remain inactive for the session timeout duration.
  2. Attempt to navigate to a different page.
* **Expected Result**: User should be redirected to the login page.

**Conclusion**

This document provides a comprehensive list of test cases for the Audit Management Web Application. It covers essential functionalities and ensures that the application meets quality standards before deployment. Regular updates to this document are recommended as new features are added or existing functionalities are modified.