



# DBANK DEMO SYSTEM TEST PLAN

## 1. Introduction:

This test plan outlines the testing approach and strategies for the digital bank demo system. The objective is to ensure that the system functions as expected, providing a seamless user experience.

## 2. Test Objectives:

- Validate the functionality of key features such as user registration, login, account overview, fund transfer, bill payment, and account settings.
- Identify and report any bugs, inconsistencies, or usability issues.
- Verify the system's performance and responsiveness under expected loads.
- Ensure the system's compatibility with different devices and browsers.

## 3. Test Environment:

- Devices: Desktop/laptop computers, tablets, and smartphones.
- Browsers: Chrome, Firefox, Safari, and Edge.
- Operating Systems: Windows, macOS, iOS, and Android.
- Test Data: Predefined test data including user accounts, transactions, and banking operations.
- Testing Tools: Web browsers, mobile emulators/simulators, debugging tools, and test management software.

## 4. Test Scenarios and Test Cases:

### 4.1 User Registration:

- Verify successful registration with valid information.
- Verify appropriate error messages for invalid inputs (e.g., invalid email format, password requirements not met).
- Test registration with duplicate email addresses.



#### **4.2 Login Functionality:**

- Test successful login with valid credentials.
- Verify appropriate error messages for incorrect login credentials.
- Test account lockout mechanism after multiple failed login attempts.

#### **4.3 Account Overview:**

- Verify that the account balance and transaction history are displayed accurately.
- Test sorting of transactions in chronological order.
- Validate the display of account details (e.g., account number, account holder name).

#### **4.4 Fund Transfer:**

- Test successful fund transfers between accounts.
- Verify appropriate error messages for failed transfer attempts (e.g., insufficient funds, invalid recipient).
- Test transfer limits and restrictions (if applicable).

#### **4.5 Bill Payment:**

- Test successful bill payment transactions.
- Verify appropriate error messages for invalid bill payment requests.
- Validate the accuracy of payment processing and transaction records.

#### **4.6 Account Settings:**

- Test the functionality of updating user profile information.
- Verify that changes to account settings (e.g., contact details, notification preferences) are saved successfully.
- Validate the security measures such as password change and two-factor authentication (if applicable).

#### **4.7 Error Handling:**

- Test the system's response to various error conditions (e.g., network errors, server timeouts).



- Verify that error messages are displayed correctly and provide relevant information to the user.

### **5. Test Execution:**

- Execute test cases based on the defined test scenarios.
- Document test results, including any issues encountered and their severity.
- Retest resolved issues to ensure proper resolution.
- Use test management software to track test progress and manage test artifacts.

### **6. Performance Testing:**

- Simulate expected user loads and test the system's performance and responsiveness.
- Measure response times for key functionalities under different load conditions.
- Identify any performance bottlenecks and suggest improvements if necessary.

### **7. Compatibility Testing:**

- Test the system's compatibility with different devices, browsers, and operating systems.
- Execute test cases on various combinations of platforms to ensure consistent behavior.

### **8. Usability Testing:**

- Evaluate the user interface and overall user experience.
- Gather feedback from users and stakeholders regarding ease of use, navigation, and visual appeal.
- Make necessary improvements based on user feedback.

### **9. Reporting:**

- Generate test reports summarizing test execution, results, and issues encountered.
- Prioritize and track issue resolution using a defect tracking system.
- Provide recommendations and suggestions for system improvement based on the testing outcomes.



## **10. Test Completion Criteria:**

- All high-priority and medium-priority test cases have been executed and passed.
- Critical defects have been resolved and retested.
- Performance testing shows acceptable response times under expected loads.
- The system is compatible with targeted devices, browsers, and operating systems.
- Usability testing feedback has been addressed appropriately.

## **11. Risks and Mitigation:**

- Risks: Incomplete or inaccurate test data, integration issues with external systems, and insufficient time for thorough testing.

- Mitigation: Ensure test data is comprehensive and covers various scenarios. Coordinate with external systems to ensure proper integration testing. Plan and allocate sufficient time for testing activities.

## **12. Conclusion:**

This test plan provides a framework for testing the digital bank demo system. By following this plan and executing the defined test scenarios, the aim is to identify and resolve any issues, ensuring a high-quality digital banking application that meets user expectations.