# Software Test Plan (STP)

Feature: Customer Login

Project: ParaBank QA Automation

Author: Baseeil Jaber

Date: August 10, 2025

Version: 1.1

## 1. Introduction

This Software Test Plan (STP) defines the strategy, scope, and process for testing the Customer Login feature of the ParaBank application. Testing activities will be managed in TestRail, defect tracking will be handled via Jira, and API-level validations will be conducted using Postman. No other testing tools will be introduced.

## 2. Test Team

The testing team for this project is as follows:

- Baseil Jaber – Team Leader

- Ali Abo Jafar – QA Engineer

- Abdelnaser – QA Engineer

- Ali Taha – Project Monitor

## 3. Test Objectives

- Verify that valid login credentials successfully grant access to the account overview.

- Validate that incorrect login attempts produce correct error handling.

- Ensure compliance with accessibility and usability standards.

- Check security measures against SQL Injection, XSS, and brute-force attacks.

- Confirm proper logging and audit trails for all login attempts.

## 4. Scope

The scope of this test plan includes functional, non-functional, and security testing of the Customer Login feature. Testing will include UI validation, API validation, and integration with backend services. All tests will be designed and managed in TestRail, executed partially via Postman for API checks, and any issues will be logged in Jira.

## 5. Test Approach

Testing will follow a combination of manual UI testing and API-level testing using Postman. All test cases will be documented and managed in TestRail. Any identified defects will be logged and tracked in Jira until closure. Security testing will be performed manually with crafted payloads.

## 6. Test Environment

Testing will be conducted in the following environment:

- Web browser compatibility: Chrome, Firefox, Edge, Safari

- Operating systems: Windows 10/11, macOS, Android, iOS

- Test server with a copy of the production database

- Stable internet connection for HTTPS traffic

- Postman configured with ParaBank API endpoints

## 7. Test Cases Reference

|  |  |  |
| --- | --- | --- |
| TC ID | Description | Expected Result |
| TC01 | Login with valid username and password | User is redirected to account overview |
| TC02 | Login with valid username and invalid password | Error message: “Invalid username or password” |
| TC03 | Login with invalid username and valid password | Error message: “Invalid username or password” |
| TC04 | Submit login form with empty fields | Validation message prompts user to enter credentials |
| TC05 | Submit login form with whitespace-only input | Validation message prompts user to enter valid data |
| TC06 | Click “Forgot login info?” link | Redirects to account recovery page |
| TC07 | Click “Register” link | Redirects to new user registration page |
| TC08 | Attempt login with SQL injection payload in username or password | Input rejected, error message shown |
| TC09 | Attempt login with XSS script in username or password | Input sanitized, no script execution |
| TC10 | Enter password and verify masking | All characters hidden as ● or \* |
| TC11 | Navigate form using only keyboard | All fields and buttons accessible in logical order |
| TC12 | Access login page from mobile device | Page adapts to screen size with proper layout |
| TC13 | Attempt login after 5 failed attempts | Account locked message displayed |
| TC14 | Verify audit log entry after login attempt | Entry recorded with timestamp, username, and status |
| TC15 | Verify session management after successful login | Session token generated and invalidated on logout |

## 8. Test Deliverables

- Test Plan document (this STP)

- Test Cases in TestRail

- API test collections in Postman

- Defect Reports in Jira

- Test Execution Reports from TestRail

- Final Test Summary Report

## 9. Schedule

The test execution will be planned over a 2-week sprint:  
- Week 1: Manual UI and API test execution & defect logging in Jira  
- Week 2: Regression testing in TestRail and API re-testing in Postman

## 10. Risks & Mitigation

- Risk: Unavailability of the test environment | Mitigation: Set up backup test environment

- Risk: Delayed delivery of build | Mitigation: Adjust test schedule and prioritize critical cases

- Risk: Test data inconsistencies | Mitigation: Maintain controlled test data sets

## 11. Approval

Prepared By: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_

Reviewed By: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_

Approved By: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_