

Software Requirements Description (SRD): ParaBank

1. Project Overview

ParaBank is a web-based banking application used as a testing environment to validate core financial functionalities. The objective of this project was to verify system stability, security, and data integrity using Manual, Automation (Playwright), and API (Postman) testing.

2. Functional Requirements (Scope)

The project scope was defined by 11 key functional requirements:

- **REQ-01 to REQ-02:** User onboarding and secure session management.
- **REQ-03 to REQ-05:** Profile management and account administration.
- **REQ-06 to REQ-08:** Financial operations (Transfers, Bill Pay, and Loans).
- **REQ-09 to REQ-11:** Data retrieval and site-wide navigation logic.

3. Testing Methodology & Execution

The quality assessment was divided among three team members to ensure 360-degree coverage:

Member	Role	Deliverables
Aleena	Project Lead	25 Manual Test Cases & Requirement Traceability Matrix (RTM).
Aiman	Automation Engineer	8 Automated Test Scripts using Playwright (Node.js).
Arisha	Defect & API Lead	Backend validation via Postman and Jira Defect Tracking.

4. Requirements Traceability Matrix (RTM) Summary

The RTM linked all 25 test cases to the 11 core requirements. Out of 11 requirements, only 4 successfully passed all testing gates.

- Critical Failures: REQ-01 (Registration) and REQ-02 (Authentication) failed due to UI timeouts and duplicate element IDs.
- Logic Failures: REQ-06 (Transfer) and REQ-08 (Loans) failed because the system accepted invalid/negative financial inputs.
- Navigation Failures: REQ-11 failed due to external domain redirects (TC_22).

5. Defect Analysis (Top Blockers)

Five critical defects were identified that prevent the software from being production-ready:

1. **BUG_001:** Functional error in the Fund Transfer login bridge.
2. **BUG_002:** Home icon inconsistency (Navigation failure).
3. **BUG_003:** Authentication instability for valid credentials.
4. **BUG_004:** Lack of input validation in profile phone fields.
5. **BUG_005:** Critical logic flaw allowing negative loan down payments.

6. Final Conclusion

Status: NO-GO / REJECTED

The ParaBank application demonstrates significant logic vulnerabilities and automation instability. While the backend APIs are largely functional, the front-end fails to handle basic user flows reliably. A re-development phase is required to address the identified blockers before another testing cycle can begin.

Test Plan: ParaBank Project

1. Objective

The objective of this project is to perform comprehensive testing on the ParaBank online banking application. This plan ensures that the system is stable, secure against invalid inputs, and provides a seamless user experience across its core financial modules.

2. Test Scope

2.1 Features to be Tested

- User Onboarding: Registration and Authentication (REQ-01, REQ-02).
- Account Management: Creating new accounts and viewing summaries (REQ-04, REQ-05).
- Financial Transactions: Bill Pay, Fund Transfers, and Loan requests (REQ-06, REQ-07, REQ-08).
- Data Integrity: Input validation and Search history (REQ-09, REQ-10).
- Navigation: Header, footer, and internal link logic (REQ-11).

2.2 Features Not to be Tested

- Third-party payment gateway integrations.
- Performance testing under heavy load (Stress Testing).
- Compatibility with legacy browsers (Internet Explorer).

3. Testing Strategy

The project utilizes a Tri-Layer Testing Approach:

- 1. Manual Testing (Aleena):** Focuses on exploratory testing, UI/UX validation, and complex edge cases.
- 2. Automation Testing (Aiman):** Uses Playwright to automate high-priority regression suites (e.g., Login, Registration) to ensure code stability.
- 3. API Testing (Arisha):** Uses Postman to validate backend responses, status codes, and data flow without the UI.

4. Tools & Environment

- Test Management:** Microsoft Excel (Test Cases & RTM).
- Defect Tracking:** Jira.
- Automation Framework:** Playwright (Node.js/VS Code).
- API Testing Tool:** Postman.
- Environment:** ParaBank Sandbox Environment (Web).

5. Defect Management Process

All identified issues will be logged in Jira with the following attributes:

- Severity:** Critical (Financial logic failure), High (Functionality broken), Medium (UI bug), Low (Cosmetic).
- Priority:** P1 (Fix immediately) to P4 (Fix if time permits).
- Evidence:** Screenshots, Playwright logs, and Postman JSON responses must be attached.

8. Risk & Mitigation

- **Risk: System downtime of the ParaBank sandbox.**
 - **Mitigation: Shift focus to manual documentation or local automation script refactoring.**
- **Risk: Automation script failures due to dynamic IDs.**
 - **Mitigation: Use Playwright's robust locators (text-based or ARIA roles) instead of volatile CSS selectors.**

Test Summary Report

Project Name: ParaBank Online Banking Portal

Test Cycle: Phase 1 (Manual, Automation, API)

Release Decision: REJECTED (NO-GO) **Prepared By:** Aleena (Lead), Aiman (Automation), Arisha (API/Defect)

1. Executive Summary

The objective of this test cycle was to validate the core functional requirements of the ParaBank application. While the backend APIs showed stability, the Front-End (UI) and Automation Suite revealed critical stability issues. The presence of financial logic errors and navigation redirects makes the current build unfit for production.

2. Test Execution Statistics

Test Methodology	Cases Planned	Cases Executed	Passed	Failed	Pass %
Manual Testing	25	25	13	12	52%
Automation (Playwright)	8	8	3	5	37.5%
API Testing (Postman)	8	8	7	1	87.5%
Combined Total	41	41	23	18	56%

3. Requirements Coverage (RTM Analysis)

Out of the 11 key requirements (REQ-01 to REQ-11), the following status was recorded:

- Requirements Fully Passed: 4 (Accounts Overview, Account Creation, Bill Pay, Search).
- Requirements Failed: 7 (Registration, Authentication, Profile Updates, Fund Transfer, Loan Management, Input Validation, Navigation).
- Coverage: 100% of defined requirements were mapped to at least one test execution.

4. Key Defect Findings

A total of 7 major defects were logged, with the following highlights:

- Critical (Logic): System allows negative down payments for loans (BUG_005) and allows \$0.00 transfers (Logic failure).
- High (Stability): Registration module consistently fails automation due to page timeouts (TC_01-03).
- High (Authentication): Duplicate element IDs on the Login page cause automation crashes (TC_05).
- Medium (Navigation): The "Locations" link redirects to an external corporate site (parasoft.com) instead of an internal page (TC_22).

5. Automation Observations

- Stability Issues: The Playwright suite identified that the UI is not optimized for automation; locators are inconsistent, leading to "Strict Mode" violations.
- Timeouts: Multiple 30-second timeouts suggest significant performance bottlenecks during the registration flow.

6. API & Backend Observations

- The backend handled 87.5% of requests correctly with proper status codes (200 OK, 400 Bad Request).
- **API Failure:** One critical endpoint allowed an update to a user profile using invalid alphabetic data for a phone number (**BUG_004**).

7. Conclusion & Recommendations

The testing team has issued a NO-GO decision.

Required Fixes for Next Cycle:

1. **Refactor UI IDs:** Ensure all buttons and input fields have unique id or data-testid attributes to support stable automation.
2. **Logic Reinforcement:** Implement server-side validation to reject negative financial amounts and non-numeric profile data.
3. **Link Audit:** Correct the navigation logic for footer and header links to ensure all traffic stays within the secure parabank.htm domain.

Per Defect Summary Report

Bug ID	Title	Priority	Summary
BUG_001	Logic Error in Fund Transfer	High	System allows zero amount transfer to same account.
BUG_002	Navigation Inconsistency	Low/Medium	Home link does not redirect to landing page.
BUG_003	Login Failure	High	Valid credentials still show login error.
BUG_004	Invalid Phone Number Accepted	Medium	Phone field accepts alphabets.
BUG_005	Negative Down Payment Accepted	High	Loan approved with negative down payment.

Req ID	Requirement Description	Manual Test IDs	Automation ID (Aiman)	API ID (Arisha)	Final Status
REQ-01	User Registration	TC_01, 02, 03, 04	TC_01, 02, 03	POST /register	FAILED
REQ-02	User Authentication	TC_05, 06, 07, 08, 09	TC_05, 06, 07, 08	POST /login	FAILED (Bug 003)
REQ-03	Profile Updates	TC_15	N/A	POST /updateProfile	FAILED (Bug 004)
REQ-04	Accounts Overview	TC_17	N/A	GET /accountDetails	PASSED
REQ-05	Account Creation	TC_16	N/A	POST /createAccount	PASSED
REQ-06	Fund Transfer	TC_11, 12, 13	N/A	POST /transfer	FAILED (Bug 001)
REQ-07	Bill Pay	TC_18, 19	N/A	POST /billpay	PASSED
REQ-08	Loan Management	TC_14	N/A	POST /requestLoan	FAILED (Bug 005)
REQ-09	History Search	TC_17	N/A	GET /transactions	PASSED
REQ-10	Input Validation	TC_02, 03, 06, 07, 15	TC_06, 07	N/A	FAILED (Bug 004)
REQ-11	Navigation Logic	TC_20, 21, 22, 23	TC_22	N/A	FAILED (Bug 002)

TC_ID	FEATURE	TEST TYPE	TEST SCENARIO	STEPS TO PERFORM	EXPECTED RESULT	ACTUAL RESULT	STATUS
TC_01	Registration	Positive	Valid User Registration	1. Click Register. 2. Fill all valid details. 3. Click Register.	Account created; Welcome message shown.	FAIL: Page timeout; elements not found.	FAIL
TC_02	Registration	Negative	Password Mismatch	1. Enter "Pass1" in Pass. 2. Enter "Pass2" in Confirm.	Error: "Passwords did not match."	FAIL: Script timed out waiting for locator.	FAIL
TC_03	Registration	Negative	Missing Last Name	1. Fill all details except Last Name. 2. Click Register.	Error: "Last name is required."	FAIL: Registration page element timeout.	FAIL
TC_04	Registration	Negative	Duplicate Username	1. Register with an existing username. 2. Click Register.	Error: "This username already exists."	Error message displayed correctly.	PASS
TC_05	Login	Positive	Successful Login	1. Enter valid User/Pass. 2. Click Log In.	Taken to Accounts Overview.	FAIL: Strict mode violation (Duplicate IDs).	FAIL
TC_06	Login	Negative	Invalid Password	1. Enter valid User. 2. Enter wrong Pass. 3. Log In.	Error: "The username and password could not be verified."	Error message displayed.	PASS
TC_07	Login	Negative	Empty Fields	1. Leave fields blank. 2. Click Log In.	System prompts for input/shows error.	Error message visible.	PASS
TC_08	Login	Negative	BUG_003: Login Failure	1. Use known valid credentials. 2. Click Log In.	Successful login to dashboard.	FAIL: Valid users unable to log in intermittently.	FAIL
TC_09	Logout	Positive	Valid Logout	1. Click 'Log Out' link.	Session ends; back to Login page.	Logged out successfully.	PASS
TC_10	API	Positive	Get Customer Data	1. Send GET request to /customers/{id}.	Status Code 200 OK.	Backend returned data correctly.	PASS
TC_11	Funds	Negative	BUG_001: Transfer Error	1. Login. 2. Navigate to Transfer Funds.	Access transfer input screen.	FAIL: Login error occurs within transfer module.	FAIL
TC_12	Funds	Negative	Transfer \$0.00	1. Enter \$0 in amount. 2. Select accounts. 3. Transfer.	Error: "Invalid Amount."	FAIL: System processed \$0.00 transfer.	FAIL
TC_13	Funds	Negative	Same Account Transfer	1. Select same From/To account. 2. Click Transfer.	Error: "Cannot transfer to same account."	FAIL: Logic failure; transfer allowed.	FAIL
TC_14	Loans	Negative	BUG_005: Down Payment	1. Request Loan. 2. Enter negative down payment.	Reject invalid input.	FAIL: Negative values accepted by system.	FAIL
TC_15	Profile	Negative	BUG_004: Phone Field	1. Edit Profile. 2. Enter letters in Phone field.	Error: "Numbers only."	FAIL: Field accepted alphabetic characters.	FAIL
TC_16	Account	Positive	Open New Account	1. Click 'Open New Account'. 2. Click Open.	New Account ID generated.	Account opened successfully.	PASS
TC_17	Account	Positive	View Account Details	1. Click an Account ID in Overview.	Transaction history is shown.	Details displayed correctly.	PASS
TC_18	Bill Pay	Negative	Missing Data	1. Leave fields blank. 2. Click Send Payment.	Error: "Field required."	Correct error prompts.	PASS
TC_19	Bill Pay	Positive	Valid Payment	1. Fill all valid biller details. 2. Click Send.	Payment successful message.	Bill paid successfully.	PASS
TC_20	Navigation	Positive	About Us Link	1. Click 'About Us' link.	Redirect to About Us page.	Navigation successful.	PASS
TC_21	Navigation	Positive	Services Link	1. Click 'Services' link.	Redirect to Services page.	Page opened correctly.	PASS
TC_22	Navigation	Negative	Locations Redirect	1. Click 'Locations' link.	Stay on ParaBank domain.	FAIL: External redirect to Parasoft.com.	FAIL
TC_23	Navigation	Negative	BUG_002: Footer Link	1. Click Home icon in footer.	Redirect to Landing Page.	FAIL: Link only refreshes current page.	FAIL
TC_24	Session	Negative	Back Button Logic	1. Logout. 2. Click browser Back button.	Prevent access to secure pages.	Securely denied access.	PASS
TC_25	UI	Positive	Page Title Check	1. Observe browser tab title.	Title should be "ParaBank".	Title matches.	PASS