

Banking Mobile

Test Plan and Report - STP and STR

CS - 4850 - 02

Fall 2025

Professor Perry

November 22, 2025

SP-26 Blue Budget

Charles LaPlatney	Kevyn Lopez	Camen McDaniel	Zoe Faju	Sarbani Adhikari
				 

Table of Contents

1. Test Objectives.....	3
2. Scope Testing.....	3
2.1. In-Scope:.....	3
2.2. Out-of-Scope:.....	3
3. Test Cases:.....	4
3.1. Functional Scenarios:.....	4
3.2. Non-Functional Scenarios:.....	4
4. Test Procedures.....	4
5. Test Environment.....	4
5.1. Hardware:.....	4
5.2. Software:.....	4
5.3. Network:.....	5
6. Test Data.....	5
7. Software Test Report.....	5

1. Test Objectives

The objective of the Software Test Plan is to test and verify that major features of the mobile banking application function correctly and meet expected behavior.

Primary objectives:

- Validate that the user login screen accepts valid credentials and handles invalid input correctly.
- Verify successful navigation between main pages (Home, Accounts, Budget, Savings).
- Confirm that Plaid test-link integration initializes and returns mock account data.
- Validate basic budgeting calculations (expense totals, category handling).
- Ensure UI layout displays correctly on common mobile screen sizes.
- Identify and document bugs, severity, and pass/fail status.

2. Scope Testing

2.1. In-Scope:

- User login and authentication flow (mock credentials).
- Navigation between pages (Home, Accounts, Budgets, Savings).
Plaid sandbox link token creation and mock account retrieval.
- Basic financial functionality:
 - Adding budget values
 - Viewing account summary data
- UI responsiveness and layout validation.

- Error-handling scenarios such as empty required fields.

2.2. Out-of-Scope:

- Real bank account connections (production Plaid).
- Performance benchmarking or load testing.
- Security penetration testing.
- Full database persistence (prototype uses mock/local data).
- Complete visual design testing (animations, advanced styling).

3. Test Cases:

3.1. Functional Scenarios:

Test Case ID	Description	Expected Result
TC-01	Login with valid credentials	User is taken to home screen
TC-02	Login with invalid credentials	Error message appears
TC-03	Navigation from Home to Accounts	Accounts page loads successfully
TC-04	Navigation Accounts to Budget	Budget page loads
TC-05	Create Plaid link token	Link token returns “sandbox” response
TC-06	Retrieve mock account list	Accounts populate on screen

TC-07	Add a new budget value	Budget updates without crash
TC-08	Attempt Action with empty fields	Displays a validation message
TC-09	Manually insert transaction	Transactions updated successfully

3.2. Non-Functional Scenarios:

Test Case ID	Description	Expected Result
NFTC-01	UI layout on iPhone 14 emulator	All components visible and accessible
NFTC-02	UI layout on Pixel 6 emulator	No overlapping or clipped UI
NFTC-03	App startup time under 3 seconds	Splash + Home load are within threshold
NFTC-04	System handles no internet during Plaid connection	Error message displayed

4. Test Procedures

TC-01: Login with valid credentials

Launch the Blue Budget app.

Enter username “testuser.”

Enter password “password123.”

Tap the Login button.

Expected Result: User is redirected to the Home screen.

TC-02: Login with invalid credentials

Steps:

Launch the app.

Enter an incorrect username or password.

Tap the Login button.

Expected Result: An error message appears; user remains on the login screen.

TC-03: Navigate from Home to Accounts page

Steps:

Log in to the app.

Tap the Accounts button on the Home screen.

Expected Result: The Accounts screen loads successfully.

TC-04: Navigate from Accounts to Budget page

Steps:

Open the Accounts screen.

Tap the Budget navigation option.

Expected Result: The Budget screen loads without errors.

TC-05: Generate Plaid link token

Steps:

Go to the Accounts screen.

Tap the Connect Bank button.

Wait for the app to request a link token.

Expected Result: The app receives a sandbox link token and begins the Plaid flow.

TC-06: Retrieve mock account list from Plaid

Steps:

Complete the Plaid test login using sandbox credentials.

Allow the app to retrieve mock account data.

Expected Result: Mock account names and balances appear on the Accounts screen.

TC-07: Add a new budget entry

Steps:

Navigate to the Budget screen.

Tap Add Budget Item.

Enter a category name.

Enter a budget amount.

Tap Save.

Expected Result: The new budget entry appears in the budget list.

TC-08: Validate empty required fields

Steps:

Leave one or more required fields blank on a form.

Tap Login or Save.

Expected Result: App displays a validation message and prevents the action.

NTC-01: UI layout on iPhone 14 simulator

Steps:

Launch the app on the iPhone 14 simulator.

Navigate through all screens (Home, Accounts, Budget, Savings).

Expected Result: All UI elements display correctly with no overlap.

NTC-02: UI layout on Pixel 6 emulator

Steps:

Launch the app on the Pixel 6 emulator.

Navigate through each main page.

Expected Result: Layout is responsive and readable on Android.

NTC-03: App startup time performance test

Steps:

Close the app completely.

Launch the app again.

Measure time until the Home screen appears.

Expected Result: App loads within 3 seconds.

NTC-04: No internet error handling for Plaid connection

Steps:

Disable internet on the emulator.

Open the Accounts screen.

Tap Connect Bank.

Expected Result: App displays an error message and does not freeze.

5. Test Environment

5.1. Hardware:

Android Emulator (Pixel 6, API 34)
iOS Simulator (iPhone 14, iOS 17)

5.2. Software:

Flutter 3.x
Dart SDK
Android Studio / Xcode
Plaid Sandbox API
Local mock JSON data for budgeting features

5.3. Network:

Wi-Fi for Plaid test API requests
Offline mode testing toggled manually

6. Test Data

- Login Credentials: testuser / password123
- Plaid User ID: demo-user-123
- Mock Account Data:
 - Checking: \$1,200
 - Savings: \$3,400
- Budget Categories: Food, Rent, Transportation
- Budget Values: \$200, \$900, \$150

7. Software Test Report

Requirement	Pass	Fail	Severity
Create account	No	Yes	N/A (Not Implemented)
Login	No	Yes	N/A (Not Implemented)
Password recovery	No	Yes	N/A (Not Implemented)
Accounts Navigation	Yes	No	Minor
Mock Account Retrieval	Yes	No	Minor
Plaid Link Token Generation	Yes	No	Moderate