TEST SPECIFICATION

1.0 Introduction

This document describes the test plan and test procedures for the Personal Finance and Budgeting Application. The application enables users to manage personal expenses, budgets, and financial summaries securely.

1.1 Goals and Objectives

- Ensure all major features function as intended.
- Validate secure user authentication.
- Confirm correct addition, saving, loading, and visualization of expenses and budgets.
- Identify and report any defects in core functionality.

1.2 Statement of Scope

Functionality to be tested:

- User registration and login
- Adding expenses
- Setting savings goals
- Depositing money
- Saving and loading expenses
- Financial summaries and balance tracking

Functionality not to be tested:

- Database integration with MongoDB (future scope)
- External banking API integration (future scope)
- Multi-user concurrency testing (out of scope)

2.0 Test Plan

2.1 Software to be Tested

- Software: Personal Finance and Budgeting Application
- Modules: Login.java, Main.java, Expense.java, WelcomeScreen.java

Exclusions:

- No testing for MongoDB connectivity.
- No testing for mobile responsiveness.

2.3 Testing Tools and Environment

- Java Development Kit (JDK 17)
- JavaFX libraries
- Manual testing on desktop environment
- Text editor for manual inspection of saved files
- Operating System: Windows 10 / macOS Monterey

2.4 Test Schedule

- Unit Testing: Week 5
- Integration Testing: Week 6
- System Testing: Week 7
- User Acceptance Testing (UAT): Week 8
- Final Defect Fixing and Regression Testing: Week 9

3.0 Test Cases

ID	Test Input	Expected Output	Description
TC1	Correct username/password	Successful login	Verify login with valid credentials
TC2	Incorrect username/password	Error message	Verify login rejection with invalid credentials
TC3	Empty username/password	Error message	Verify handling of empty login fields
TC4	New valid user registration	Account created	Verify user can register successfully
TC5	Duplicate username registration	Error message	Verify rejection of duplicate user registration
TC6	Add valid expense (category + positive amount)	Expense added to table	Verify expense creation
TC7	Add invalid expense (non-numeric amount)	Error message	Validate input type for amount
TC8	Add expense exceeding balance	Error message	Verify prevention of overspending
TC9	Deposit positive amount	Balance increases	Verify money deposit functionality
TC10	Deposit zero or negative amount	Error message	Validate deposit amount is positive
TC11	Set valid savings goal	Goal set and displayed	Verify setting savings goal
TC12	Set invalid savings goal (negative value)	Error message	Verify input validation for goals

TC13 Save expenses to file	File created with entries	Verify expenses are saved correctly
TC14 Load expenses from file	Table populated with entries	Verify loading of previous expenses
TC15 Trigger low balance warning	Warning popup shown	Verify balance warning at low levels