

Expense Ease (Test Plan)

Project Name: My Library

Date: 20.06.2025

Prepared by: Nethsilu Marasinghe

Content

1.0 INTRODUCTION	3
● OBJECTIVE	3
● PROJECT DESCRIPTION	3
● TECH STACK	3
● PROCESS TAILORING	4
2.0 ASSUMPTIONS/DEPENDENCIES	4
3.0 TEST REQUIREMENTS	4
4.0 TEST TOOLS	5
5.0 RESOURCE REQUIREMENTS	5
6.0 TEST SCHEDULE	6
7.0 RISKS/MITIGATION	6
8.0 METRICS	6

1.0 INTRODUCTION

1.1 OBJECTIVES

This Test Plan outlines the testing strategy for "Expense Ease," a comprehensive expense tracking system designed to simplify personal financial management. The application enables users to efficiently record incomes and expenses, monitor balances, and gain insights into their spending habits.

The plan covers unit testing, integration testing, and system verification testing (SVT) to ensure the system meets all functional and performance requirements specified in the Software Requirements Specification (SRS).

1.2 PROJECT DESCRIPTION

Expense Ease is an intuitive financial management platform that:

- Allows users to record income and expenses with ease.
- Provides a clear overview of current balances and financial health.
- Helps users track spending patterns for better budgeting decisions.
- Offers a user-friendly interface for seamless financial management.

1.3 TECH STACK

- Frontend: React.js + Tailwind CSS
- Backend: Node.js + Express.js
- Database: MongoDB Atlas
- Authentication: JWT (JSON Web Token)

1.4 PROCESS TAILORING

- Testing Types: Functional, Error handling, Performance.
- Exclusions: GUI testing, compatibility testing beyond Windows OS.

2.0 ASSUMPTIONS / DEPENDENCIES

Assumptions:

- Input files follow the exact format specified in the SRS.
- The developer will complete the code by 1 month.
- Users will enter data in the correct format.

Dependencies:

- Functional code delivery by development team.
- Availability of test environment (Windows OS).

3.0 TEST REQUIREMENTS

Functional Requirements to Test:

1. Expense & Income Management

- Record and categorize income and expenses with customizable labels.
- Support for recurring transactions (e.g., monthly bills, subscriptions).

2. Financial Tracking & Reporting

- Real-time balance calculation based on income and expenses.
- Generate summary reports (monthly).
- Visualize spending trends via charts and graphs (pie charts).

3. Budgeting

- Set monthly budget limits for different categories.

4. Security & Data Protection

- User authentication (login, or biometric security).
- Encryption of sensitive financial data in storage and transit.

4.0 TEST TOOLS

- Manual Testing (unit tests)
- OS & Browsers (Windows 10, Chrome, Firefox).
- Test Data (Sample data).

5.0 RESOURCE REQUIREMENTS

Role	Effort (Days)
Tester	1 Day
Development Support	1 Day

6.0 TEST SCHEDULE

Phase	Duration
Unit Testing	6 hours
Integration Testing	3 hours

7.0 RISK / MITIGATION

Risk	Mitigation
Performance > 1 minute	Optimize code or adjust requirements.

8.0 METRICS

- Defects found per phase (Unit/Integration).
- Test coverage (requirements traced to test cases).