Personal Finance Manager Console Application

Functional Specifications

Author: Matěj Noga

Version**:** 1.0
Date: 13.10.

Contents

1. Document History

Version	Date	Author	Comments
1.0	16.10.	Matěj Noga	

2. Introduction

The purpose of this document is to detail the functional specifications for the **Personal Finance Manager Console Application** (PFM). The document will provide information on how users will interact with the application, the interface design, and how errors will be handled. This specification aligns with the **Requirements Document** and outlines each feature and functionality.

Target Audience: Developers, Testers, and Business Analysts.

3. Application Architecture

The Personal Finance Manager application will be a console-based program, designed to run on Windows, macOS, and Linux platforms. It will handle financial data in local storage formats such as CSV and JSON.

Application Flow:

1. User launches the application.
2. The main menu is displayed, with options for registering/logging in, viewing financial data, adding/editing transactions, and generating reports.

4. Application Launch

User Stories:

If **Then**

User opens the application for the They will be prompted to create an account (registration first time process).

User has an existing account The login screen is displayed.

The application will load previously saved data upon startup if the user is registered and has data stored.

5. User Setup and Registration

1. **First-time User**:
   - The user is asked to provide basic information such as their name, currency, and set up a PIN for authentication.
   - Users can skip the PIN setup if they do not require additional security.
2. **Input Handling**:
   - **PIN Code**: 4-8 digits only.
   - If the entered PINs do not match, an error message will be displayed: "PINs do not match. Please try again."

6. Main Interface Overview

The main interface consists of:

1. **Main Menu**:
   - **Options**:
     - 1: Add Income
     - 2: Add Expense
     - 3: View Balance
     - 4: Generate Report
     - 5: Edit/Delete Transaction
     - 6: Exit
2. **User Interactions**:
   - Users will select an option from the menu by typing the corresponding number.
   - All inputs are validated for correctness before proceeding.
3. **Error Messages**:
   - Invalid selections or unhandled exceptions will display an error message and return the user to the main menu.

Example Menu:

Personal Finance Manager

1. Add Income

2. Add Expense

3. View Balance

4. Generate Report

5. Edit/Delete Transaction

6. Exit

Please select an option:

7. **Error Handling**

The application will handle user errors gracefully. Below are common error scenarios:

Error Scenario	Error Message	Action Taken
Invalid option selected in main menu	"Invalid option. Please select a valid option."	User is returned to the main menu
Incorrect PIN during login	"Incorrect PIN. Try again."	After 5 failed attempts, the application locks for 5 minutes
Invalid transaction input (e.g., non-numeric value)	"Invalid input. Please enter numeric value."	Re-prompt user for correct input a
Insufficient data for report generation	"Not enough data to a report."	Return user to main menu generate
File I/O error during data backup	"Error: Could not save backup file."	Display message and retry option

8. Functional Features

REQ-FM-001: Add Income

- **Description**: Users can add an income entry by specifying the amount, category, date, and optional notes.
- **User Flow:**
  - **The user selects “Add Income” from the main menu.**
  - **The system will prompt the user for each detail:**
    - **Amount: A positive numeric value is required.**
      - **Example: Enter amount: [User Input]**
    - **Category: A text input for the income category (e.g., “Salary,” “Investments”).**
      - **Example: Enter category: [User Input]**
    - **Date: The date format is dd-mm-yyyy, and the system checks for a valid date.**
      - **Example: Enter date (dd-mm-yyyy): [User Input]**
    - **Notes (optional): Text input for additional comments.**
      - **Example: Notes (optional): [User Input]**
  - **After inputting all details, the system validates each entry. If any errors occur (e.g., invalid date), the system will prompt the user to correct the field. Once validated, the income entry is saved.**

REQ-FM-002: Add Expense

- **Description**: Users can record expenses with similar fields as for income.
- **User Flow**:
  - The user selects “Add Expense.”
  - The system prompts the user to enter:

1. **Amount**: Positive numeric value, allowing for decimal entries.
   - Example: Enter amount: [User Input]
2. **Category**: Text input for the expense category (e.g., “Food,” “Transport”).
   - Example: Enter category: [User Input]
3. **Date**: The date format is dd-mm-yyyy.
   - Example: Enter date (dd-mm-yyyy): [User Input]
4. **Notes (optional)**: Additional information.

- Example: Notes (optional): [User Input]

- After validation, the expense entry is saved. If there are invalid entries (e.g., incorrect date format), the system displays an error message and prompts the user to correct it.

REQ-FM-003: View Balance

- **Description**: The system displays the current balance, calculated by subtracting total expenses from total income.
- **User Flow:**
  - The user selects "View Balance."
  - The system calculates and displays the balance based on

    recorded incomes and expenses.
  - Output:

    Your current balance is: [Balance]
  - If no transactions have been recorded yet, the system

    will display:

    No transactions found. Your balance is zero.

REQ-FM-004: Generate Monthly Report

- **Description**: The user can generate a report summarizing income and expenses for the selected month.
- **User Flow:**
  - The user selects "Generate Report."
  - The system prompts the user to enter:
    1. **Month and Year**: The user inputs the desired month and year in mm-yyyy format.
       - Example: Enter month and year (mm-yyyy): [User Input]
  - Output:
    - The report displays in a text-based format with totals for each category, and an optional ASCII bar chart to visualize balances and expenses.
    - **Example Report:**

      Report for January 2024:

      - Income:

        Salary: $2000

        Freelance: $500

- Expenses:

  Rent: $800

  Groceries: $300

  Balance: $1400

  - If no data is available for the selected month, the system will display:

    No data available for the selected month.

REQ-FM-005: Data Backup and Restore

- **Description**: This feature enables users to back up their data to a file (CSV or JSON) and restore it from the same format.
- **User Flow**:
  - **Backup**:
    - The user selects "Backup Data."
    - The system confirms the file path for saving the backup.
    - If an error occurs (e.g., file system unavailable), an error message is displayed.
  - **Restore**:
    - The user selects "Restore Data" and provides the file path.
    - The system loads and restores the data; if the file format is incorrect or an error occurs, the system displays an error and cancels the restoration.

REQ-FM-006: Edit/Delete Transaction

- **Description**: This feature allows users to choose a specific transaction to edit or delete from a list of recorded transactions.
- **User Flow:**
  - **Edit Transaction:**
    1. The user selects "Edit Transaction" from the main menu.
    2. The system displays a list of transactions, each with a unique number.

       1. Income | Category: Salary | Amount: 2000 | Date: 01-01-2024

       2. Expense | Category: Rent | Amount: 800 | Date: 03-01-2024
    3. The user selects the number of the transaction they want to edit, e.g., transaction #1.

4. The system displays the selected transaction details and allows the user to modify each field (amount, category, date, notes).

   - **Edit Example:**
     Original:

     Income | Category: Salary | Amount: 2000 | Date: 01-01-2024

     New Entry:

     Enter new amount (2000): 2500

     Enter new category (Salary): Bonus

     Enter new date (01-01-2024): 05-01-2024

     Enter new notes (optional): Year-end bonus

5. After editing, the system validates the updated information. If any errors occur (e.g., invalid date format), the system will prompt the user to correct them. Once validated, the changes are saved, and the transaction is updated.

- **Delete Transaction:**

  1. The user selects “Delete Transaction” from the main menu.
  2. The system displays the numbered list of transactions again.

     1. Income | Category: Salary | Amount: 2000 | Date: 01-01-2024

     2. Expense | Category: Rent | Amount: 800 | Date: 03-01-2024

  3. The user selects the number of the transaction they wish to delete, e.g., transaction #2.
  4. The system asks for confirmation with a prompt:

     Are you sure you want to delete this transaction? (yes/no):

  5. If the user enters “yes,” the system permanently removes the selected transaction from the database.

9. Glossary of Terms

- **PFM**: Personal Finance Manager
- **PIN**: Personal Identification Number
- **CSV**: Comma-Separated Values
- **JSON**: JavaScript Object Notation

10. Priorities

1. Security features (PIN handling, data encryption).
2. Financial functionalities (Add Income/Expense, View Balance, Report Generation).
3. Error handling and smooth user interaction.

This functional specification provides detailed guidance on how the console application will behave, including the user interface, inputs, and error handling.