1 Updated User Stories

1.1 User Story 1: Record a Transaction

- **Description**: *As a user, I want to record an expense or income so that my daily finances are accurately tracked.*
- **Traceability**: FR1, NFR – Reliability, Usability
- **Preconditions:** The application is open and the user has access to the “Add Transaction” feature.
- **Acceptance Criteria:**
  - The user can enter transaction details (amount, type, date, and category).
  - The transaction is saved immediately after confirmation and appears in the list of transactions.
  - The total income/expense summary updates automatically.
  - The system provides feedback confirming successful entry.
- **Postconditions:** Transaction is stored locally and persists when the application restarts.
- **Priority:** High

1.2 User Story 2: Edit or Delete a Transaction

- **Description**: *As a user, I want to edit or delete a transaction so that my financial records stay accurate.*
- **Traceability:** FR1, NFR – Reliability, Usability
- **Preconditions:** At least one transaction exists in the system.
- **Acceptance Criteria:**
  - The user can open an existing transaction and modify any editable field.
  - The user can delete an entry after a confirmation prompt.
  - Updated or deleted transactions reflect immediately in totals and summaries.
  - The system prevents accidental deletion through a confirmation step.
- **Postconditions:** Transaction list and totals are updated to reflect the change.
- **Priority:** High

1.3 User Story 3: View and Filter Transactions

- **Description**: *As a user, I want to view and filter transactions by date and category so that I can easily review my spending history.*
- **Traceability:** FR2, NFR – Usability, Performance
- **Acceptance Criteria:**
  - The user can view all past transactions in a table or list.
  - The user can filter transactions by category, date range, or keyword.
  - The filtered list updates immediately after criteria are applied.
  - The user can clear filters to return to the full view.
- **Postconditions:** Filtered or full lists remain visible until the user changes the view.
- **Priority:** High

1.4 User Story 4: Set a Monthly Budget

- **Description**: *As a user, I want to set a monthly budget for each spending category so that I can manage my expenses more effectively.*
- **Traceability:** FR3, NFR – Usability, Reliability
- **Preconditions:** User has at least one spending category defined.
- **Acceptance Criteria:**
  - The user can set or update a monthly spending limit for a selected category.
  - The budget amount is saved and displayed for the current month.
  - The system visually shows remaining budget as transactions are added.
  - Invalid inputs (e.g., negative values or empty fields) are rejected with feedback.
- **Postconditions:** Budget is stored and available for alert and reporting features.
- **Priority:** Medium

1.5 User Story 5: Receive Overspending Alerts

- **Description**: *As a user, I want to receive an alert when spending exceeds my budget so that I can adjust my spending habits early.*
- **Traceability:** FR3, NFR – Reliability, Usability

- **Preconditions:** At least one active budget is set for the month.
- **Acceptance Criteria:**
  - The system checks total spending against each category limit.
  - When spending meets or exceeds the limit, the system displays a clear visual alert.
  - The alert indicates the category and amount over budget.
  - The alert remains visible until acknowledged or budget is reset.
- **Postconditions:** User is aware of overspending status and can adjust budgets accordingly.
- **Priority:** Medium

1.6 User Story 6: View Monthly Summary

- **Description**: *As a user, I want to view a summary of my income, expenses, and remaining balance so that I can assess my financial status.*
- **Traceability:** FR4, NFR – Usability, Performance
- **Acceptance Criteria:**
  - The user can select a reporting period (e.g., month or custom range).
  - The system generates totals for income, expenses, and balance.
  - The summary is displayed in an easy-to-read format.
  - When no data is available, a clear “no records found” message is shown.
- **Postconditions:** Generated summaries can be revisited without re-entry until the user changes the period.
- **Priority:** High

1.7 User Story 7: View Spending Chart

- **Description**: *As a user, I want to see a chart of my spending by category so that I can identify trends and make informed financial decisions.*
- **Traceability:** FR4, NFR – Usability, Performance
- **Acceptance Criteria:**
  - The system presents at least one visual chart (pie or bar) showing spending by category.
  - Charts update automatically when transactions or filters change.

  - The visualization is easy to interpret with clear labels and legends.
- **Postconditions:** Chart view remains synchronized with current transaction data.
- **Priority:** Low

1.8 User Story 8: Load Saved Data on Startup

- **Description**: *As a user, I want my saved transactions and budgets to load automatically when I open the app so that I can continue seamlessly.*
- **Traceability:** FR1, FR3, NFR – Reliability, Performance
- **Preconditions:** Previously saved data exists locally.
- **Acceptance Criteria:**
  - The system loads all stored transactions and budgets automatically at startup.
  - The system notifies the user if no data is found.
  - Loaded data appears accurately within corresponding views (Transactions, Budgets, Reports).
- **Postconditions:** User resumes from previous state with no manual loading required.
- **Priority:** High