

# TEST SCENARIO TS-02: CORE FUNCTIONALITY & TRANSACTIONS

**Project Name:** Library Inventory Manager

**Document ID:** TS-02

**Module:** Business Logic (CRUD, Loans, Reporting)

**Author:** Dominik Hoch

**Date:** 2.1.2026

## 1. TEST OBJECTIVE

To verify that the application correctly handles data operations, specifically **Database Transactions** (saving books with authors), **M:N relationships**, **Loans** (using the 5th table), and **Reporting** (using SQL Views).

## 2. PREREQUISITES

- The application is running (`python app.py`).
  - Database is installed and accessible.
  - A valid JSON file named `test_import.json` is ready for import testing.
- 

## 3. TEST STEPS EXECUTION

### STEP 1: Add Book (Transaction Test)

- **Action:** Navigate to "Add Book". Fill in the form:
  - **Title:** Transaction Test Book
  - **Price:** 500
  - **Condition:** New
  - **Category:** Sci-Fi
  - **Authors:** Select **two different authors** (hold CTRL/CMD to select multiple).
  - Click "**Save Book**".
- **Expected Result:** The application redirects to the Dashboard with a green success message ("Book added successfully"). The new book appears in the list showing both selected authors (verifying the M:N relationship).

### STEP 2: Borrow Book (Loan Logic)

- **Action:** Locate the Transaction Test Book on the Dashboard. Click the green "**Borrow**" button.

- **Input:** Enter borrower name: John Tester.
  - Click "**Confirm Loan**".
- **Expected Result:** Redirects to Dashboard. The book's status badge changes to "**Borrowed**" (Red). The "Borrow" button is replaced by a "Return" button. A record is created in the `loans` table.

### STEP 3: Return Book

- **Action:** Click the "**Return**" button next to the same book.
- **Expected Result:** The book's status changes back to "**Available**" (Green). The "Borrow" button reappears.

### STEP 4: Reporting (SQL View Test)

- **Action:** Navigate to the "**Reports**" page.
- **Expected Result:** A table is displayed showing statistics (Total Books, Inventory Value, Average Price) grouped by Category. This confirms the database **VIEW** (`view_library_stats`) is working correctly.

### STEP 5: Data Import

- **Action:** Navigate to "**Import**". Click "Choose File", select `test_import.json`, and click "Upload".
- **Expected Result:** A success message appears stating how many books were imported. The new books are visible on the Dashboard.

### STEP 6: Edit Book

- **Action:** Click "**Edit**" on any book. Change the **Price** to 999 and click "Update".
- **Expected Result:** The change is immediately reflected in the Book List.

---

## 4. TEST RESULT

*(To be filled by the Tester)*

**Status:** [ ] **PASS** (All functions work as expected) [ ] **FAIL** (One or more functions failed)

**Tester**

**Name:** \_\_\_\_\_ **Date:** \_\_\_\_\_