# Software Engineering Audit Report

**Relazione_Ingegneria_Software-1.pdf**

*— MOCK V2 for layout review —*

CAPRA — Automated Audit System

February 25, 2026

## Contents

# 1 Project Overview

The project implements a **library management system** designed to handle book cataloging, user registration, and loan operations for a university library. Users can register, browse the catalog, search by title/author/genre, request book loans, extend active loans, and post comments on works and volumes. Librarians manage the catalog (add, edit, remove books) and process loan conclusions. The system is built with **Java** using **JavaFX** for the graphical interface and **PostgreSQL** for data persistence, following an MVC architectural pattern with DAO-based data access.

---

Issues found: **6** (HIGH: 1, MEDIUM: 4, LOW: 1) — Features: 6/7 present, 1 partial

## 2 Requirements Analysis

*This section lists the functional requirements identified in the document and indicates whether each has an associated use case.*

| Req. | Name | UC? | Linked UCs |
|---|---|---|---|
| RF-1 | Registrazione utente | ✓ | UC-1 |
| RF-2 | Login utente | ✓ | UC-0 |
| RF-3 | Gestione prestiti | ✓ | UC-2, UC-2.1, UC-2.2, UC-2.3, UC-4 |
| RF-4 | Gestione catalogo libri | ✓ | UC-3, UC-3.1, UC-3.2, UC-5 |
| RF-5 | Ricerca libri | ✓ | UC-5.1, UC-5.2 |
| RF-6 | Gestione commenti | ✓ | UC-6, UC-6.1, UC-6.2, UC-6.3 |
| RF-7 | Gestione profilo | ✓ | UC-0.1, UC-0.2 |
| RF-8 | Limite massimo prestiti | ✓ | UC-2.4 |
| RF-9 | Notifiche scadenza | × | — |
| RF-10 | Report statistiche | × | — |

**Requirements without UCs:**

- **RF-9 — Notifiche scadenza:** No use case references automated notifications. → Define a UC or mark as out of scope.
- **RF-10 — Report statistiche:** No UC covers reporting. → Add UC or clarify scope.

## 3 Architecture Analysis

The project follows a **layered Model-View-Controller (MVC)** architecture. The presentation layer uses **JavaFX** with dedicated controllers per view; the business logic layer processes operations and validation; the persistence layer uses **DAO pattern** with **PostgreSQL**. The **Singleton** pattern manages the database connection. Components connect hierarchically: Controllers → Services → DAOs → Database.

No significant architecture issues were identified.

# 4 Use Case Analysis

The document describes **27 use cases**. This section analyzes the *internal quality* of each use case (completeness, clarity, consistency of the template). Traceability to requirements and tests is analyzed in Section **??**.

## 4.1 Use Cases with Structured Template

### UC-1 — Registrazione

**Actor:**
Utente non registrato

**Preconditions:**
L'utente non ha un account nel sistema

**Main Flow:**

1. L'utente accede alla pagina di registrazione
2. Inserisce nome, cognome, email e password
3. Il sistema valida i campi e crea l'account

**Postconditions:**
L'utente è registrato e può effettuare il login

**Alternative Flows:**
Campi non validi → messaggio di errore

✓ No internal issues identified. Template is complete.

### UC-2 — Effettua prestito

**Actor:**
Utente registrato

**Preconditions:**
L'utente è autenticato; il libro è disponibile

**Main Flow:**

1. L'utente cerca un libro nel catalogo
2. Seleziona un volume disponibile
3. Conferma il prestito
4. Il sistema registra il prestito con data di scadenza

**Postconditions:**
Il prestito è attivo; il volume risulta non disponibile

**Alternative Flows:**
Nessun volume disponibile → notifica

medium — UC-STRUCT-001
**Problem:** The business rule "maximum 3 concurrent loans" is mentioned in the domain model but not reflected in the UC template's preconditions or alternative flows.
**Suggestion:** Add a precondition "User has fewer than 3 active loans" and an alternative flow describing the rejection when the limit is exceeded.

### UC-3 — Aggiunge libro

**Actor:**
Biblioteca (Bibliotecario)

**Preconditions:**
Il bibliotecario è autenticato

**Main Flow:**

1. Il bibliotecario accede alla gestione catalogo
2. Inserisce i dati del nuovo libro (titolo, autore, ISBN, genere)
3. Conferma l'inserimento
4. Il sistema aggiunge il libro al catalogo

**Postconditions:**
Il libro è presente nel catalogo

**Alternative Flows:**
ISBN duplicato → errore

**medium — UC-STRUCT-002**
**Problem:** The actor is named "Biblioteca" in the UC diagram but "Bibliotecario" in the template text.
**Suggestion:** Unify actor naming: choose one term and apply consistently.

### UC-4 — Conclude Prestito

**Actor:**
Biblioteca (Bibliotecario)

**Preconditions:**
Esiste un prestito attivo

**Main Flow:**

1. Il bibliotecario cerca il prestito da concludere
2. Conferma la restituzione del volume
3. Il sistema aggiorna lo stato del prestito e la disponibilità del volume

**Postconditions:**
Il prestito è concluso; il volume è nuovamente disponibile

**Alternative Flows:**
Prestito non trovato → errore

**medium — UC-STRUCT-003**
**Problem:** Same actor naming inconsistency as UC-3.
**Suggestion:** Use the unified actor name chosen for UC-3.

## 4.2 Use Cases without Structured Template

The following use cases are referenced in diagrams or narrative but lack a formal template description. A missing template does not necessarily indicate an error if the UC is sufficiently simple.

- UC-0 — Login
- UC-0.1 — Modifica account
- UC-0.2 — Elimina account
- UC-2.1 — Cancella prestito
- UC-2.2 — Prolunga prestito
- UC-2.3 — Visualizza prestiti
- UC-2.4 — Limite massimo di prestiti
- UC-3.1 — Modifica libro
- UC-3.2 — Rimuove libro
- UC-3.3 — CRUD libro (summary)
- UC-5 — Visualizza catalogo
- UC-5.1 — Cerca libro
- UC-5.2 — Verifica disponibilità opera
- UC-5.3 — Recupero ISBN
- UC-6 — Commento su opera
- UC-6.1 — Commento su volume
- UC-6.2 — Visualizza commenti opera
- UC-6.3 — Visualizza commenti volume
- UC-7 — Login Biblioteca
- UC-8 — Visualizza catalogo Biblioteca
- UC-9 — Validazione campi
- UC-10 — Gestione errori login
- UC-11 — Gestione errori prenotazione

# 5 Testing Analysis

## 5.1 Testing Strategy

The project uses **JUnit** for unit tests and **TestFX** for UI tests. Unit tests cover core business operations (loan reservations, cancellations, ISBN retrieval, availability checks). UI tests verify login flows, field validation, and book reservation interactions. Error scenarios include invalid logins, field validation failures, unavailable books, and exceeded loan limits.

## 5.2 Additional Testing Issues

**high — TST-001 — No mocking framework**

**Problem:** All tests access the real PostgreSQL database. No mocking or in-memory database is used, making tests dependent on external state.

**Suggestion:** Introduce Mockito for unit tests and H2 for integration tests, isolating test execution from external dependencies.

## 6 Traceability Analysis

This section maps each requirement to its use cases, design references, and test coverage. The table starts from requirements; use cases that exist without a parent requirement are listed separately.

| Requirement | UC | Design | Test | |
|---|---|---|---|---|
| 1*RF-1 | UC-1 | | ✓ | ✓ |
| 1*RF-2 | UC-0 | | ✓ | ✓ |
| RF-3 | UC-2 | | ✓ | ✓ |
| | UC-2.1 | | ✓ | ✓ |
| | UC-2.2 | | ✓ | ✓ |
| | UC-2.3 | | ✓ | × |
| | UC-2.4 | | ✓ | × |
| | UC-4 | | ✓ | × |
| RF-4 | UC-3 | | ✓ | × |
| | UC-3.1 | | ✓ | × |
| | UC-3.2 | | ✓ | × |
| | UC-3.3 | | ✓ | × |
| | UC-5 | | ✓ | × |
| RF-5 | UC-5.1 | | ✓ | × |
| | UC-5.2 | | ✓ | ✓ |
| RF-6 | UC-6 | | ✓ | × |
| | UC-6.1 | | ✓ | × |
| | UC-6.2 | | ✓ | × |
| | UC-6.3 | | ✓ | × |
| RF-7 | UC-0.1 | | ✓ | × |
| | UC-0.2 | | ✓ | × |
| RF-8 | UC-2.4 | | ✓ | × |
| RF-9 | — | | — | — |
| RF-10 | — | | — | — |
| *Use cases without a parent requirement:* | | | | |
| — | UC-7 (Login Biblioteca) | | ✓ | × |
| — | UC-8 (Catalogo Biblioteca) | | ✓ | × |
| — | UC-9 (Validazione campi) | | ✓ | ✓ |
| — | UC-10 (Errori login) | | ✓ | ✓ |
| — | UC-11 (Errori prenotazione) | | ✓ | ✓ |

**Summary:** Of 10 requirements, 8 are linked to UCs, 2 have no UC. Of 27 UCs, 10 have full coverage (design + test), 17 lack test coverage, 5 have no parent requirement.

**Suggestion:** Prioritize adding tests for core operations (UC-3, UC-4), then account management (UC-0.1, UC-0.2). Define UCs for RF-9 and RF-10, or remove these requirements from scope.

# 7 Missing Features

| Feature | Status | Coverage |
|---|---|---|
| Definition and Documentation of Use Cases | PARTIAL | 60% |
| *6 other feature(s)* | PRESENT | 100% |

**Definition and Documentation of Use Cases** Templates lack pre/post-conditions for several UCs. → Add formal conditions to each UC template.