

Software Engineering Audit Report

Relazione_Ingegneria_Software-1.pdf

— *MOCK VERSION for layout review* —

CAPRA — Automated Audit System

February 25, 2026

Contents

1 Project Overview	2
2 Requirements Analysis — 3 Proposals	3
2.1 Proposal A — Two subsections with description lists	3
2.2 Proposal B — Compact table + details for gaps	3
2.3 Proposal C — Minimal list, only gap details	4
3 Architecture Analysis	5
4 Use Case Analysis	5
4.1 Use Cases with Structured Template	5
4.2 Use Cases without Structured Template	5
5 Testing Analysis	6
5.1 Testing Strategy	6
5.2 Missing Test Coverage	6
6 Traceability Analysis	7
6.1 Traceability Gaps	7
6.2 Fully Traced Use Cases	7
7 Missing Features	7

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 — 3 Proposals

This section lists the requirements identified in the document and indicates whether each has an associated use case. Requirements without a linked UC need attention.

2.1 Proposal A — Two subsections with description lists

Requirements with Associated Use Cases

The following requirements are linked to at least one use case.

RF-1 — Registrazione utente

→ UC-1 (Registrazione)

RF-2 — Login utente

→ UC-0 (Login)

RF-3 — Gestione prestiti

→ UC-2 (Effettua prestito), UC-2.1 (Cancella prestito), UC-2.2 (Prolunga prestito), UC-2.3 (Visualizza prestiti), UC-4 (Conclude Prestito)

RF-4 — Gestione catalogo libri

→ UC-3 (Aggiunge libro), UC-3.1 (Modifica libro), UC-3.2 (Rimuove libro), UC-5 (Visualizza catalogo)

RF-5 — Ricerca libri

→ UC-5.1 (Cerca per titolo/autore/genere), UC-5.2 (Verifica disponibilità)

RF-6 — Gestione commenti

→ UC-6 (Commento su opera), UC-6.1 (Commento su volume), UC-6.2, UC-6.3 (Visualizza commenti)

RF-7 — Gestione profilo

→ UC-0.1 (Modifica account), UC-0.2 (Elimina account)

RF-8 — Limite massimo prestiti

→ UC-2.4 (Limite massimo di prestiti)

Requirements without Associated Use Cases

The following requirements lack a formally linked use case.

RF-9 — Gestione notifiche scadenza

No use case references an automated notification system for overdue loans.

Suggestion: Define a UC for loan expiry notification or remove the requirement.

RF-10 — Report statistiche biblioteca

No use case covers statistical reporting for the librarian.

Suggestion: Add a UC for generating library reports or clarify this is out of scope.

2.2 Proposal B — Compact table + details for gaps

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	✓	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 max prestiti	✓	UC-2.4
RF-9	Notifiche scadenza	✗	—
RF-10	Report statistiche	✗	—

Requirements without UCs:

- **RF-9 — Notifiche scadenza:** No UC 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.

2.3 Proposal C — Minimal list, only gap details

The document specifies **10 functional requirements**. Of these, **8** are linked to at least one use case and **2** lack a formal UC definition.

Requirements Linked to Use Cases

- RF-1 (Registrazione) → UC-1
- RF-2 (Login) → UC-0
- RF-3 (Gestione prestiti) → UC-2, UC-2.1, UC-2.2, UC-2.3, UC-4
- RF-4 (Gestione catalogo) → 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 max prestiti) → UC-2.4

Requirements without Use Cases

- **RF-9 — Notifiche scadenza:** Mentioned in domain model but no UC formalizes notification behavior. → Add a dedicated UC.
- **RF-10 — Report statistiche:** Referenced informally but not modeled. → Add UC or remove from 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). Traceability to requirements and tests is analyzed in subsequent sections.

4.1 Use Cases with Structured Template

The following use cases have a formal template (actor, preconditions, flows, etc.).

UC-1 — Registrazione

✓ No internal issues identified. Template is complete.

UC-2 — Effettua prestito

UC-STRUCT-001 — medium

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

UC-STRUCT-002 — medium

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 it consistently throughout the document.

UC-4 — Conclude Prestito

UC-STRUCT-002 — medium

Problem: Same actor naming inconsistency as UC-3.

Suggestion: Use the same 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. This is noted as an observation; 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 Missing Test Coverage

The following use cases lack explicit test coverage:

- UC-0.1 — Modifica account: no unit or UI test found
- UC-0.2 — Elimina account: no unit or UI test found
- UC-2.3 — Visualizza prestiti: no unit or UI test found
- UC-2.4 — Limite massimo di prestiti: no unit or UI test found
- UC-3 — Aggiunge libro: no unit or UI test found
- UC-3.1 — Modifica libro: no unit or UI test found
- UC-3.2 — Rimuove libro: no unit or UI test found
- UC-3.3 — CRUD libro: no unit or UI test found
- UC-4 — Conclude Prestito: no unit or UI test found
- UC-5 — Visualizza catalogo: no unit or UI test found
- UC-5.1 — Cerca libro: no unit or UI test found
- UC-6 — Commento su opera: no unit or UI test found
- UC-6.1 — Commento su volume: no unit or UI test found
- UC-6.2 — Visualizza commenti opera: no unit or UI test found
- UC-6.3 — Visualizza commenti volume: no unit or UI test found
- UC-7 — Login Biblioteca: no unit or UI test found
- UC-8 — Visualizza catalogo Biblioteca: no unit or UI test found

Suggestion: Add at least one test per uncovered use case, starting from the most critical business operations.

6 Traceability Analysis

This section maps each use case to its design implementation and test coverage to identify traceability gaps. The traceability analysis runs independently from the issue-detection agents.

Of 27 traced use cases: 10 fully covered, 17 with gaps (all missing test coverage).

6.1 Traceability Gaps

UC-0.1 — Modifica account

Design: ✓ ProfiloController Test: ✗ No test found

Suggestion: Add a test verifying account modification.

UC-0.2 — Elimina account

Design: ✓ ProfiloController.cancellaUtente() Test: ✗ No test found

Suggestion: Add a test for account deletion flow.

[...remaining 15 gaps follow same pattern...]

6.2 Fully Traced Use Cases

- UC-0 — Login
- UC-1 — Registrazione
- UC-2 — Effettua prestito
- UC-2.1 — Cancella prestito
- UC-2.2 — Prolunga prestito
- UC-5.2 — Verifica disponibilità opera
- UC-5.3 — Recupero ISBN
- UC-9 — Validazione campi
- UC-10 — Gestione errori login

7 Missing Features

Feature	Status	Coverage
Definition and Documentation of Use Cases	PARTIAL	60%
<i>6 other feature(s)</i>	PRESENT	100%

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