# Problem Statement

## Task

Create a RESTful API that manages a simple library system.

The API should allow API user to perform the following actions:

1. Register a new borrower to the library.
2. Register a new book to the library.
3. Get a list of all books in the library.

The API should allow API user to perform the following actions on behalf of a borrower:

1. Borrow a book with a particular book id (refer Book in Data Models).
2. Return a borrowed book.

## Data Models

Borrower should have a unique id, a name and an email address
Book should have a unique id, an ISBN number, a title, and an author

ISBN number uniquely identifies a book in the following way:

- 2 books with the same title and same author but different ISBN numbers are considered as different books
- 2 books with the same ISBN numbers must have the same title and same author

Multiple copies of books with same ISBN number are allowed in the system

## Requirements

1. Use a programming language and framework of your choice to create the project.
    a. Use of Java 17 and Spring Boot framework is an added bonus
2. Configurable to run in multiple environments
3. Use a package manager to manage project dependencies.
4. Implement proper data validation and error handling.
5. Use a database to store borrower and book data.
    a. Justify your choice of database
6. Implement REST API endpoints for each action mentioned above.
7. Multiple books with the same ISBN number should be registered as books with different ids.
8. Ensure that no more than one member is borrowing the same book (same book id) at a time.
9. Provide clear documentation for how to use your API
10. Provide documentation of all your assumptions for any requirements that are not explicitly stated in this task

# Solution : Collabera Library Management System

## Before You Read

- This is an MVP version of app for the problem statement
- Uses Spring boot framework 3.5.5, Java 17 (as per request in the problem statement)
- UI is in Basic HTML5 with CSS (This is for demo purpose of the Api only, its not a fine-tuned industry standard UI)
- Docker engine
  - Docker Desktop v4.19 (engine v23.0.5) running on Window 11 PC

## Repo

https://github.com/Erandauh/book-library-system

### Why Relational DB - Justification

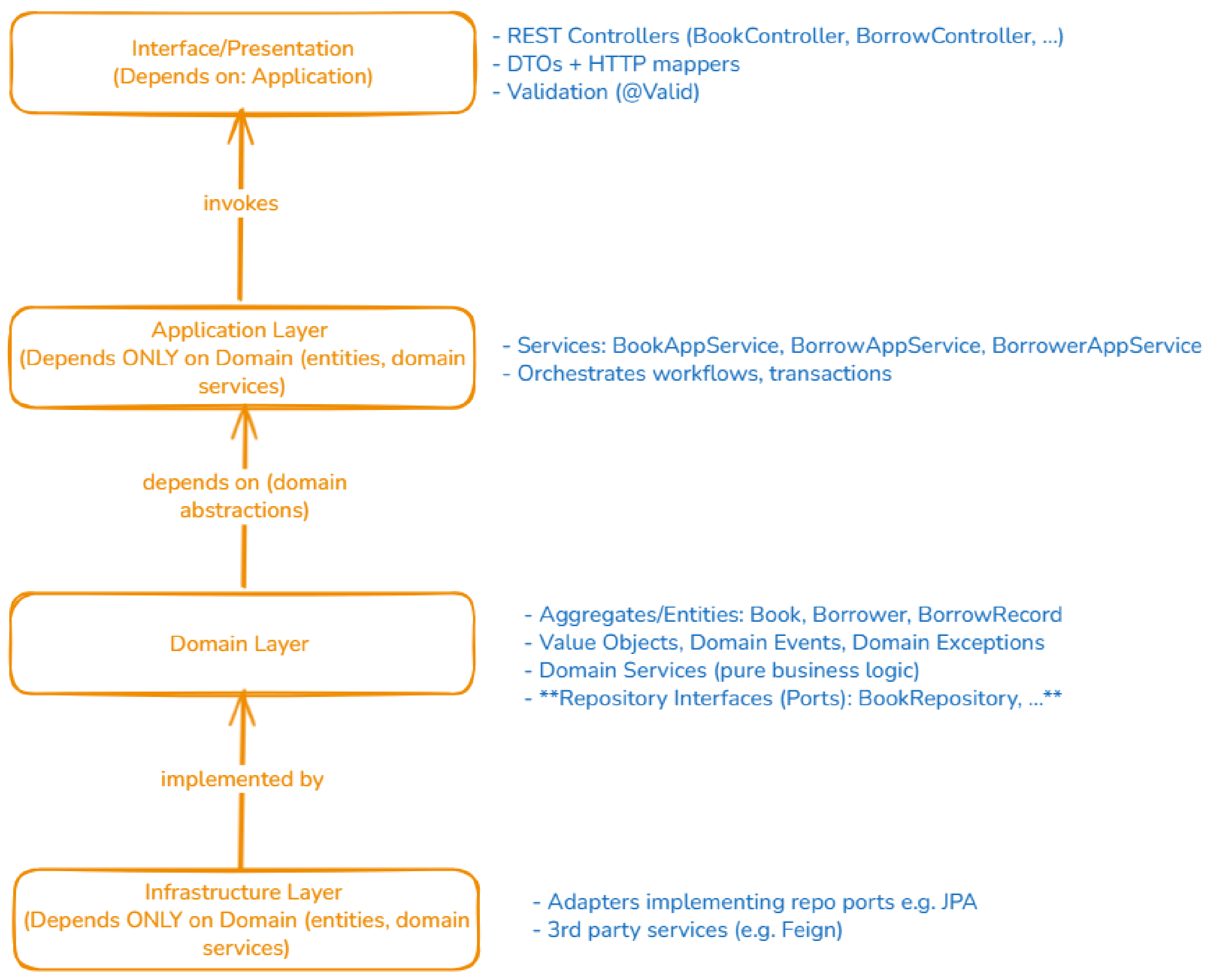
We have **entities**: `Book`, `Borrower`, `BorrowRecord`.
They have **relationships**:
One `Book` → Many `BorrowRecords`.
One `Borrower` → Many `BorrowRecords`.

RDBMS handles these with **foreign keys & joins** elegantly (with SQL querying power).

e.g.,: “Find all books borrowed by Eranda” → a simple SQL join.

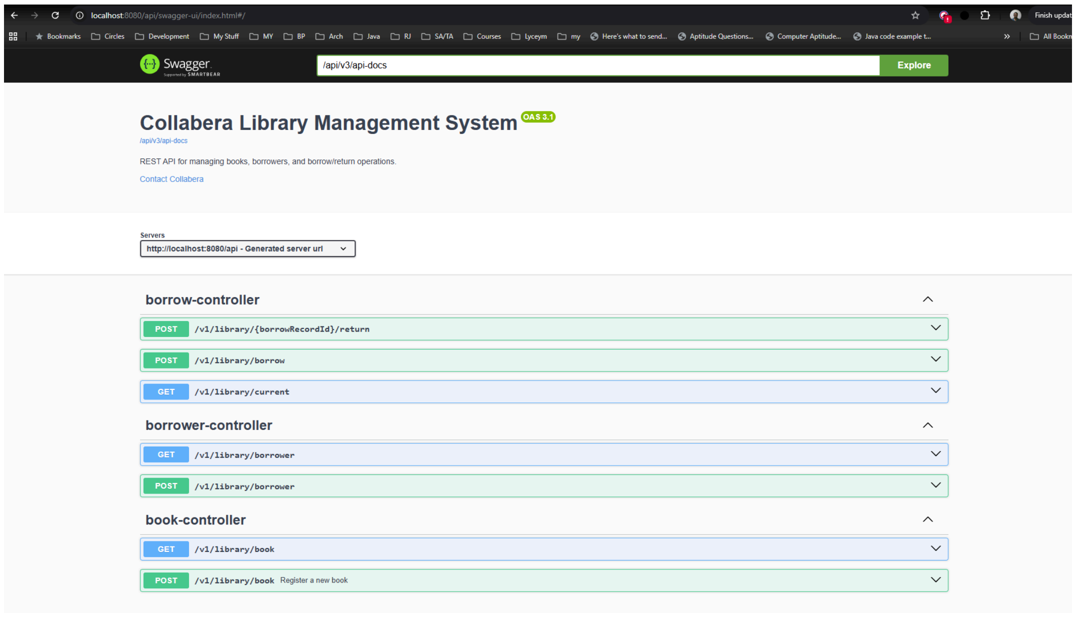
**NoSQL (Mongo, Cassandra, etc.)** → better for unstructured data, documents, or high write scalability. Overkill here.

## Backend Architecture


Interface/Presentation
(Depends on: Application)
- REST Controllers (BookController, BorrowController, ...)
- DTOs + HTTP mappers
- Validation (@Valid)
invokes
Application Layer
(Depends ONLY on Domain (entities, domain services)
- Services: BookAppService, BorrowAppService, BorrowerAppService
- Orchestrates workflows, transactions
depends on (domain abstractions)
Domain Layer
- Aggregates/Entities: Book, Borrower, BorrowRecord
- Value Objects, Domain Events, Domain Exceptions
- Domain Services (pure business logic)
- **Repository Interfaces (Ports): BookRepository, ...**
implemented by
Infrastructure Layer
(Depends ONLY on Domain (entities, domain services)
- Adapters implementing repo ports e.g. JPA
- 3rd party services (e.g. Feign)

## Swagger

http://localhost:8080/api/swagger-ui/index.html#/



# Collabera Library Management System OAS 3.1

/api/v3/api-docs

REST API for managing books, borrowers, and borrow/return operations.

Contact Collabera

Servers: http://localhost:8080/api - Generated server url

**borrow-controller**

- POST /v1/library/{borrowRecordId}/return
- POST /v1/library/borrow
- GET /v1/library/current

**borrower-controller**

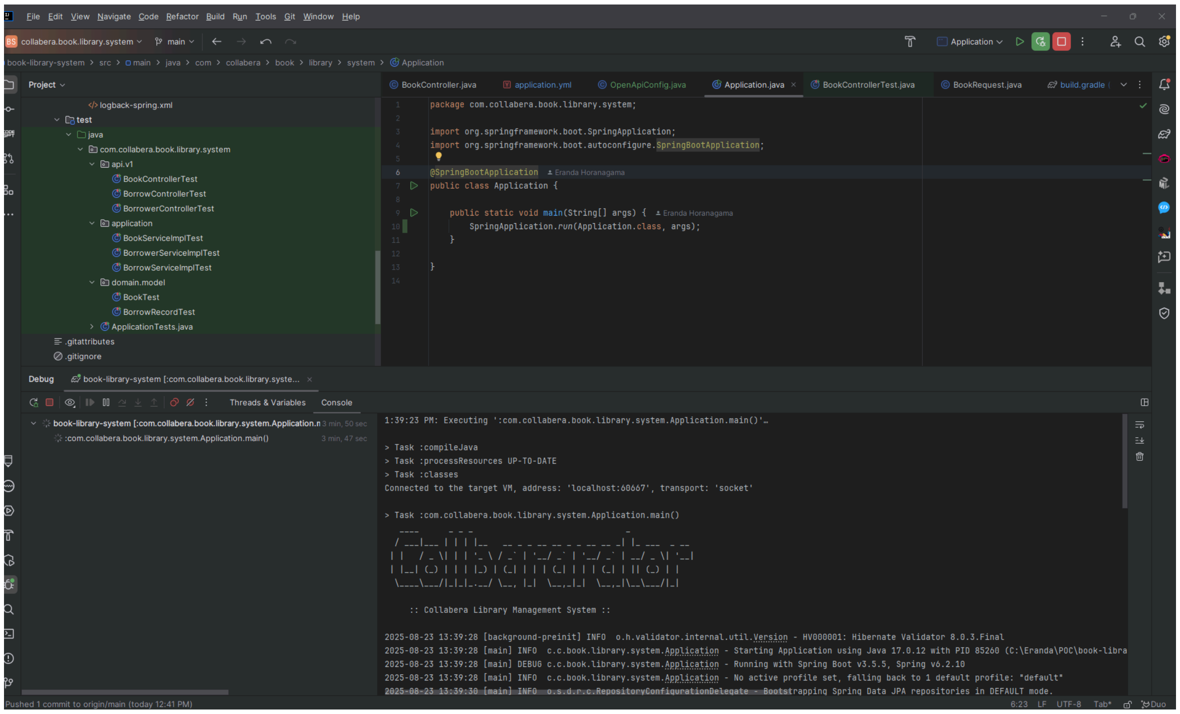
- GET /v1/library/borrower
- POST /v1/library/borrower

**book-controller**

- GET /v1/library/book
- POST /v1/library/book Register a new book

## Build And Run

Use Gradle (You can build with graddle wrapper - ./gradlew clean build)

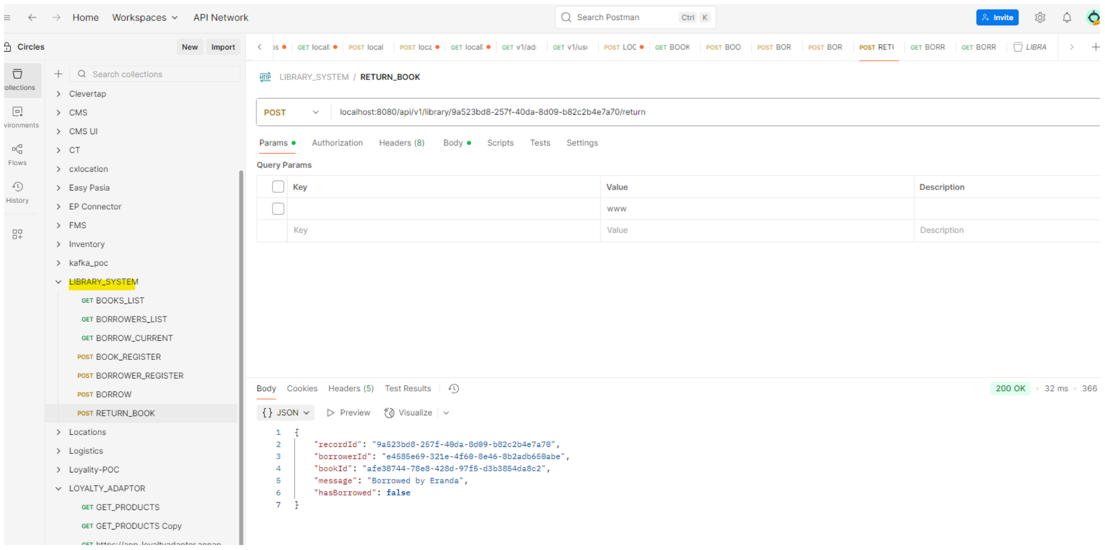


```java
package com.collabera.book.library.system;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class Application {

    public static void main(String[] args) {
        SpringApplication.run(Application.class, args);
    }

}
```

```
1:39:23 PM: Executing ':com.collabera.book.library.system.Application.main()'…

> Task :compileJava
> Task :processResources UP-TO-DATE
> Task :classes
Connected to the target VM, address: 'localhost:60667', transport: 'socket'

> Task :com.collabera.book.library.system.Application.main()
   ____      _ _ _                           _
  / ___|___ | | | |__   __ _ _ __ __ _ _ __ __| |_ ___  _ __
 | |   / _ \| | | '_ \ / _` | '__/ _` | '__/ _` | __/ _ \| '__|
 | |__| (_) | | | |_) | (_| | | | (_| | | | (_| | || (_) | |
  \____\___/|_|_|_.__/ \__, |_|  \__,_|_|  \__,_|\__\___/|_|

     :: Collabera Library Management System ::

2025-08-23 13:39:28 [background-preinit] INFO  o.h.validator.internal.util.Version - HV000001: Hibernate Validator 8.0.3.Final
2025-08-23 13:39:28 [main] INFO  c.c.book.library.system.Application - Starting Application using Java 17.0.12 with PID 85260 (C:\Eranda\POC\book-libra
2025-08-23 13:39:28 [main] DEBUG c.c.book.library.system.Application - Running with Spring Boot v3.5.5, Spring v6.2.10
2025-08-23 13:39:28 [main] INFO  c.c.book.library.system.Application - No active profile set, falling back to 1 default profile: "default"
2025-08-23 13:39:30 [main] INFO  o.s.d.r.c.RepositoryConfigurationDelegate - Bootstrapping Spring Data JPA repositories in DEFAULT mode.
```

## Postman Collection

Find the postman collection in the repo itself
[https://github.com/Erandauh/book-library-system/blob/main/LIBRARY_SYSTEM.postman_collection.json](https://github.com/Erandauh/book-library-system/blob/main/LIBRARY_SYSTEM.postman_collection.json)



LIBRARY_SYSTEM / RETURN_BOOK

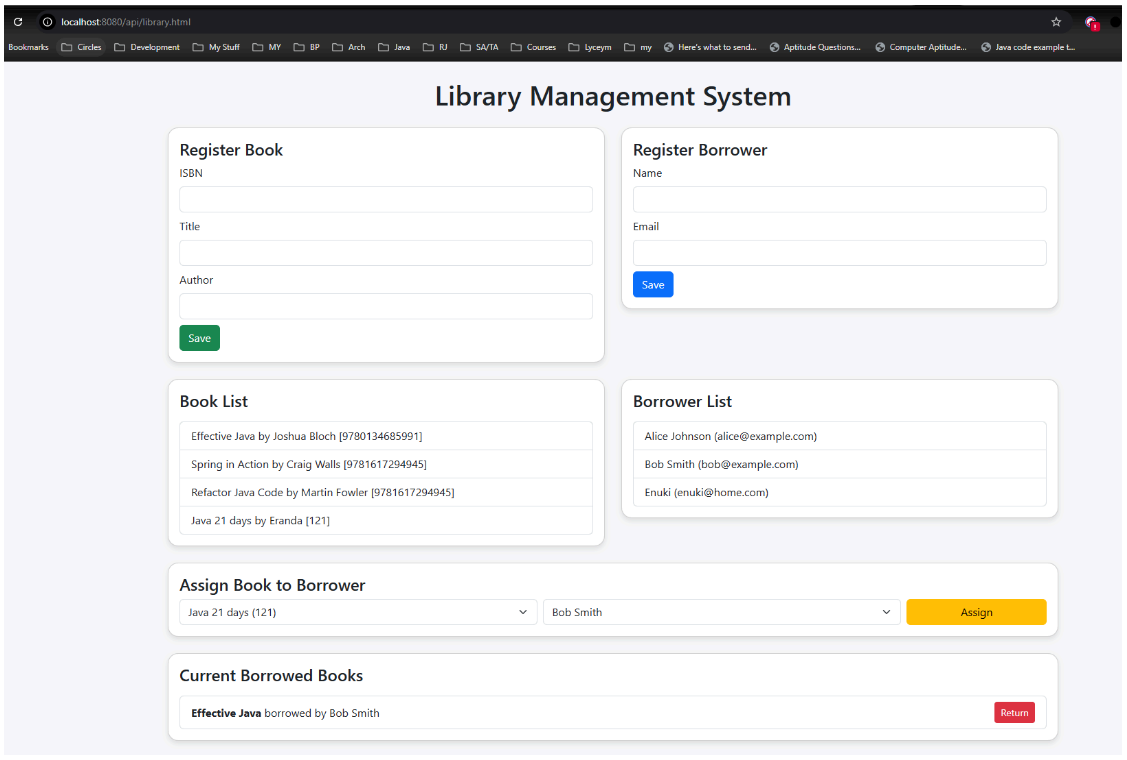
POST localhost:8080/api/v1/library/9a523bd8-257f-40da-8d09-b82c2b4e7a70/return

200 OK · 32 ms · 366

```
{
    "recordId": "9a523bd8-257f-40da-8d09-b82c2b4e7a70",
    "borrowerId": "e4585e69-321e-4f60-8e46-8b2adb650abe",
    "bookId": "afe38744-78e8-428d-97f5-d3b3854da8c2",
    "message": "Borrowed by Eranda",
    "hasBorrowed": false
}
```

## Test With Web UI

Served via same server(for demo purpose) : [http://localhost:8080/api/library.html](http://localhost:8080/api/library.html)



Library Management System

Register Book: ISBN, Title, Author — Save

Register Borrower: Name, Email — Save

Book List
- Effective Java by Joshua Bloch [9780134685991]
- Spring in Action by Craig Walls [9781617294945]
- Refactor Java Code by Martin Fowler [9781617294945]
- Java 21 days by Eranda [121]

Borrower List
- Alice Johnson (alice@example.com)
- Bob Smith (bob@example.com)
- Enuki (enuki@home.com)

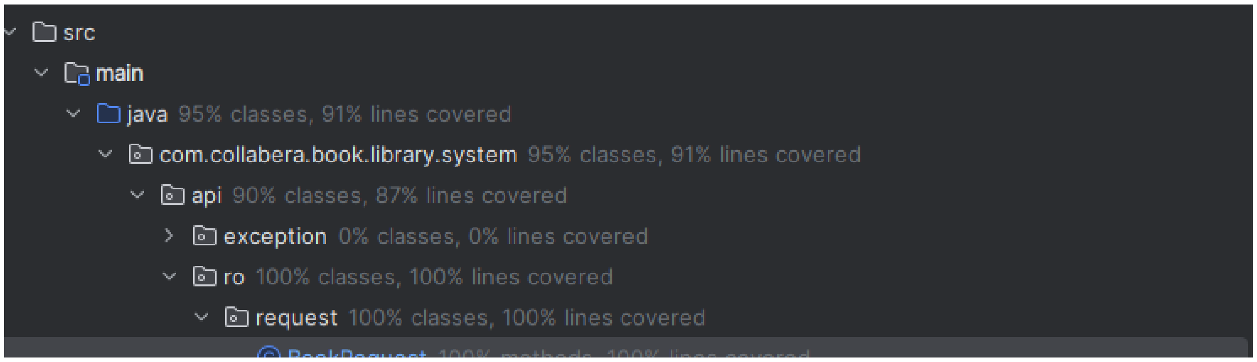
Assign Book to Borrower: Java 21 days (121) — Bob Smith — Assign

Current Borrowed Books
- **Effective Java** borrowed by Bob Smith — Return

- This UI is in Basic HTML5 with CSS (This is for demo purpose of the Api only, its not a fine-tuned industry standard UI)

## Unit Test Coverage

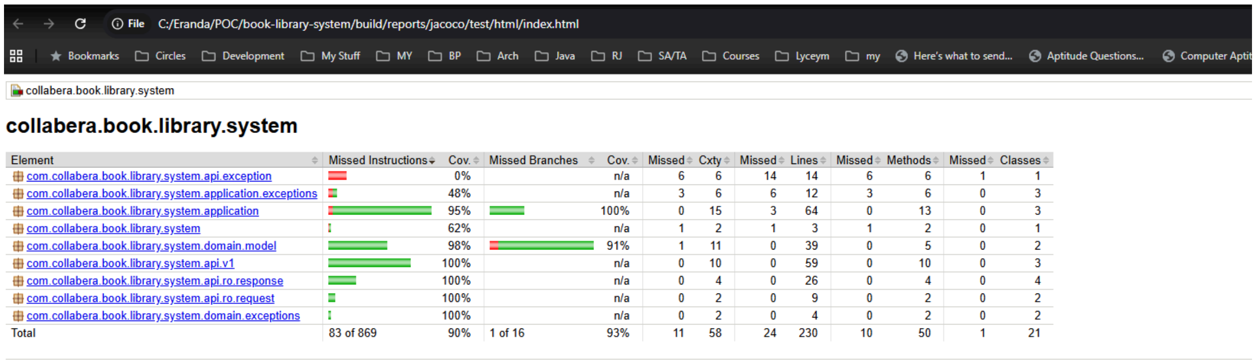
Test coverage is above > 84%
All mandatory layers are covered with unit tests (Controllers, Application Layer, Domain Layer)



```
src
  main
    java 95% classes, 91% lines covered
      com.collabera.book.library.system 95% classes, 91% lines covered
        api 90% classes, 87% lines covered
          exception 0% classes, 0% lines covered
          ro 100% classes, 100% lines covered
            request 100% classes, 100% lines covered
```

### JaCoCo

Generate the jacoco report with : *./gradlew test jacocoTestReport*



C:/Eranda/POC/book-library-system/build/reports/jacoco/test/html/index.html

collabera.book.library.system

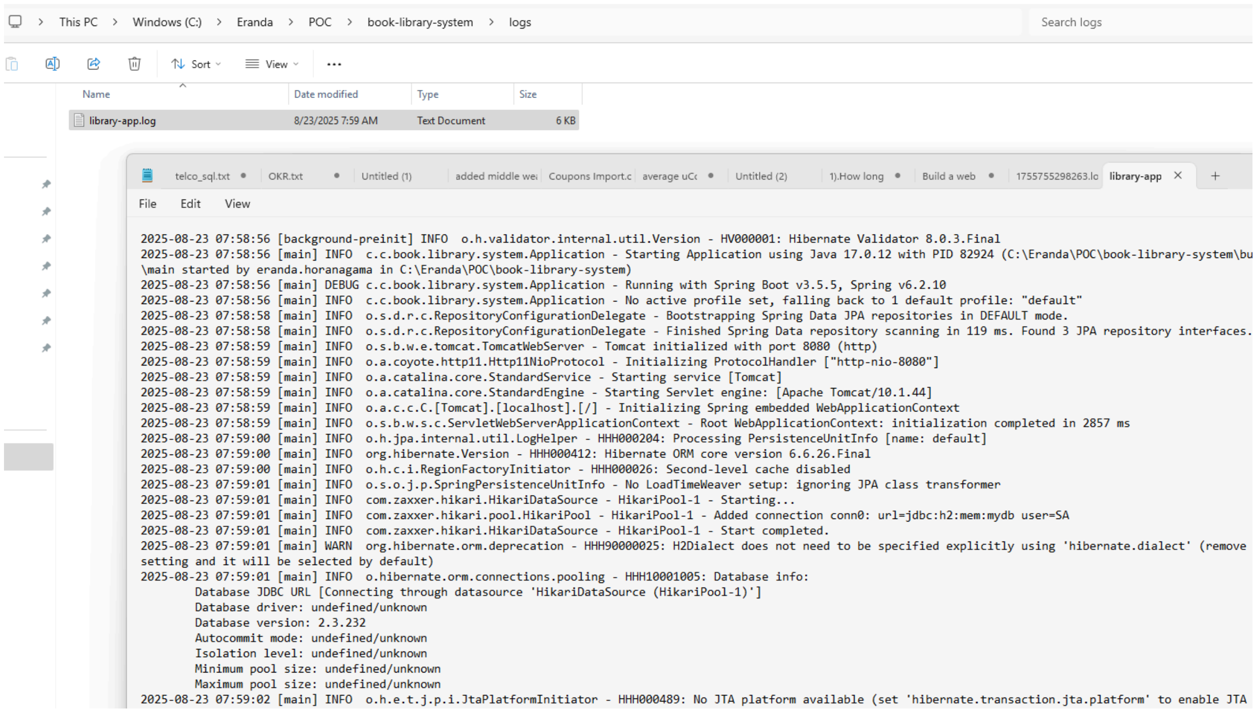
**collabera.book.library.system**

| Element | Missed Instructions | Cov. | Missed Branches | Cov. | Missed | Cxty | Missed | Lines | Missed | Methods | Missed | Classes |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| com.collabera.book.library.system.api.exception | | 0% | | n/a | 6 | 6 | 14 | 14 | 6 | 6 | 1 | 1 |
| com.collabera.book.library.system.application.exceptions | | 48% | | n/a | 3 | 6 | 6 | 12 | 3 | 6 | 0 | 3 |
| com.collabera.book.library.system.application | | 95% | | 100% | 0 | 15 | 3 | 64 | 0 | 13 | 0 | 3 |
| com.collabera.book.library.system | | 62% | | n/a | 1 | 2 | 1 | 3 | 1 | 2 | 0 | 1 |
| com.collabera.book.library.system.domain.model | | 98% | | 91% | 1 | 11 | 0 | 39 | 0 | 5 | 0 | 2 |
| com.collabera.book.library.system.api.v1 | | 100% | | n/a | 0 | 10 | 0 | 59 | 0 | 10 | 0 | 3 |
| com.collabera.book.library.system.api.ro.response | | 100% | | n/a | 0 | 4 | 0 | 26 | 0 | 4 | 0 | 4 |
| com.collabera.book.library.system.api.ro.request | | 100% | | n/a | 0 | 2 | 0 | 9 | 0 | 2 | 0 | 2 |
| com.collabera.book.library.system.domain.exceptions | | 100% | | n/a | 0 | 2 | 0 | 4 | 0 | 2 | 0 | 2 |
| Total | 83 of 869 | 90% | 1 of 16 | 93% | 11 | 58 | 24 | 230 | 10 | 50 | 1 | 21 |

Folder to look: *\build\reports\jacoco\test\html\index.html*

## Application Log

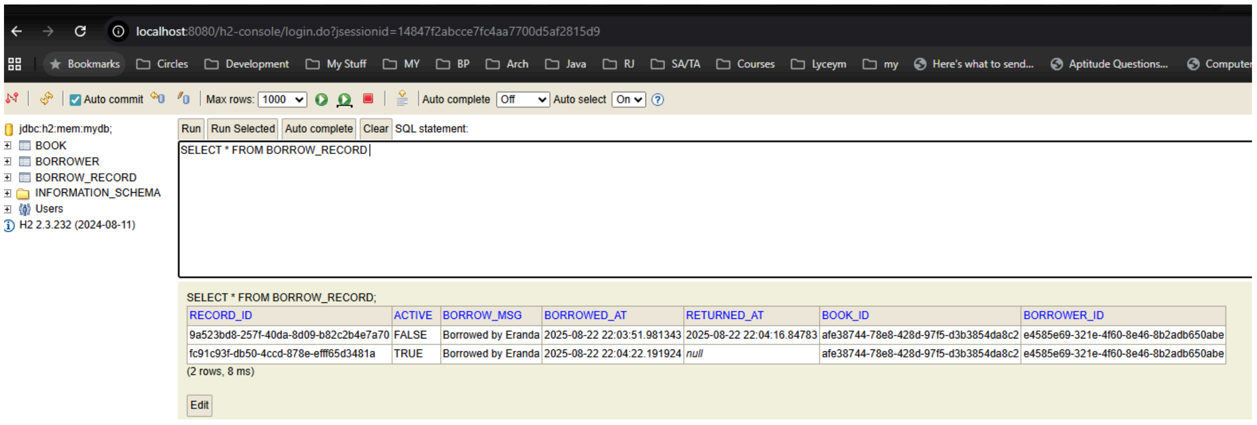
Can be found in *\logs\library-app.log*



```
2025-08-23 07:58:56 [background-preinit] INFO  o.h.validator.internal.util.Version - HV000001: Hibernate Validator 8.0.3.Final
2025-08-23 07:58:56 [main] INFO  c.c.book.library.system.Application - Starting Application using Java 17.0.12 with PID 82924 (C:\Eranda\POC\book-library-system\bu
\main started by eranda.horanagama in C:\Eranda\POC\book-library-system)
2025-08-23 07:58:56 [main] DEBUG c.c.book.library.system.Application - Running with Spring Boot v3.5.5, Spring v6.2.10
2025-08-23 07:58:56 [main] INFO  c.c.book.library.system.Application - No active profile set, falling back to 1 default profile: "default"
2025-08-23 07:58:58 [main] INFO  o.s.d.r.c.RepositoryConfigurationDelegate - Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2025-08-23 07:58:58 [main] INFO  o.s.d.r.c.RepositoryConfigurationDelegate - Finished Spring Data repository scanning in 119 ms. Found 3 JPA repository interfaces.
2025-08-23 07:58:59 [main] INFO  o.s.b.w.e.tomcat.TomcatWebServer - Tomcat initialized with port 8080 (http)
2025-08-23 07:58:59 [main] INFO  o.a.coyote.http11.Http11NioProtocol - Initializing ProtocolHandler ["http-nio-8080"]
2025-08-23 07:58:59 [main] INFO  o.a.catalina.core.StandardService - Starting service [Tomcat]
2025-08-23 07:58:59 [main] INFO  o.a.catalina.core.StandardEngine - Starting Servlet engine: [Apache Tomcat/10.1.44]
2025-08-23 07:58:59 [main] INFO  o.a.c.c.C.[Tomcat].[localhost].[/] - Initializing Spring embedded WebApplicationContext
2025-08-23 07:58:59 [main] INFO  o.s.b.w.s.c.ServletWebServerApplicationContext - Root WebApplicationContext: initialization completed in 2857 ms
2025-08-23 07:59:00 [main] INFO  o.h.jpa.internal.util.LogHelper - HHH000204: Processing PersistenceUnitInfo [name: default]
2025-08-23 07:59:00 [main] INFO  org.hibernate.Version - HHH000412: Hibernate ORM core version 6.6.26.Final
2025-08-23 07:59:00 [main] INFO  o.h.c.i.RegionFactoryInitiator - HHH000026: Second-level cache disabled
2025-08-23 07:59:01 [main] INFO  o.s.o.j.p.SpringPersistenceUnitInfo - No LoadTimeWeaver setup: ignoring JPA class transformer
2025-08-23 07:59:01 [main] INFO  com.zaxxer.hikari.HikariDataSource - HikariPool-1 - Starting...
2025-08-23 07:59:01 [main] INFO  com.zaxxer.hikari.pool.HikariPool - HikariPool-1 - Added connection conn0: url=jdbc:h2:mem:mydb user=SA
2025-08-23 07:59:01 [main] INFO  com.zaxxer.hikari.HikariDataSource - HikariPool-1 - Start completed.
2025-08-23 07:59:01 [main] WARN  org.hibernate.orm.deprecation - HHH90000025: H2Dialect does not need to be specified explicitly using 'hibernate.dialect' (remove
setting and it will be selected by default)
2025-08-23 07:59:01 [main] INFO  o.hibernate.orm.connections.pooling - HHH10001005: Database info:
	Database JDBC URL [Connecting through datasource 'HikariDataSource (HikariPool-1)']
	Database driver: undefined/unknown
	Database version: 2.3.232
	Autocommit mode: undefined/unknown
	Isolation level: undefined/unknown
	Minimum pool size: undefined/unknown
	Maximum pool size: undefined/unknown
2025-08-23 07:59:02 [main] INFO  o.h.e.t.j.p.i.JtaPlatformInitiator - HHH000489: No JTA platform available (set 'hibernate.transaction.jta.platform' to enable JTA
```

## H2 DB

http://localhost:8080/h2-console/login.do?jsessionid=14847f2abcce7fc4aa7700d5af2815d9



```
SELECT * FROM BORROW_RECORD
```

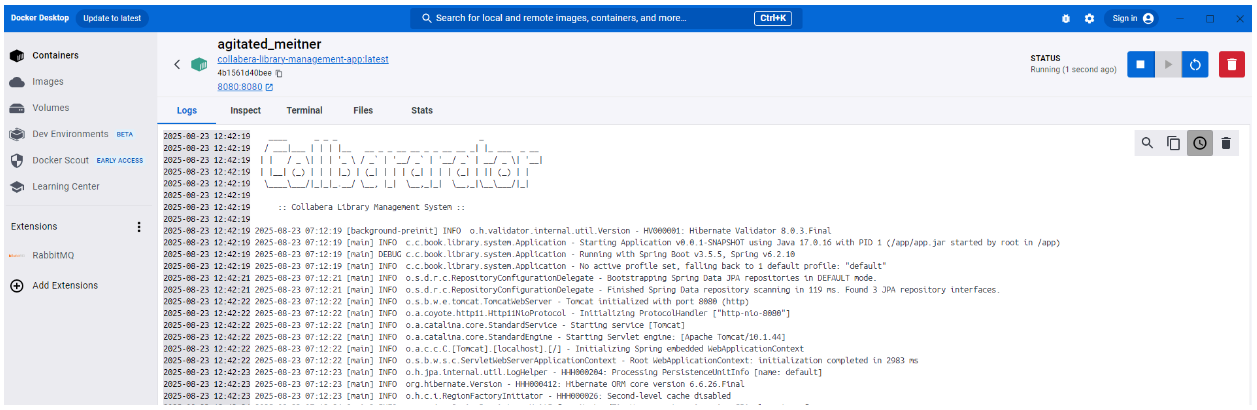
SELECT * FROM BORROW_RECORD;

| RECORD_ID | ACTIVE | BORROW_MSG | BORROWED_AT | RETURNED_AT | BOOK_ID | BORROWER_ID |
|---|---|---|---|---|---|---|
| 9a523bd8-257f-40da-8d09-b82c2b4e7a70 | FALSE | Borrowed by Eranda | 2025-08-22 22:03:51.981343 | 2025-08-22 22:04:16.84783 | afe38744-78e8-428d-97f5-d3b3854da8c2 | e4585e69-321e-4f60-8e46-8b2adb650abe |
| fc91c93f-db50-4ccd-878e-efff65d3481a | TRUE | Borrowed by Eranda | 2025-08-22 22:04:22.191924 | *null* | afe38744-78e8-428d-97f5-d3b3854da8c2 | e4585e69-321e-4f60-8e46-8b2adb650abe |

(2 rows, 8 ms)

## Docker Image

```
# Build Docker image
docker build -t collabera-library-management-app .
```



agitated_meitner
collabera-library-management-app:latest
4b1561d40bee
8080:8080

STATUS
Running (1 second ago)

```
2025-08-23 12:42:19   ____      _ _ _                     _
2025-08-23 12:42:19  / ___|___ | | | |__   __ _ _ __ __ _ _ __ __ _| |_ ___  _ __
2025-08-23 12:42:19 | |   / _ \| | | '_ \ / _` | '__/ _` | '__/ _` | __/ _ \| '__|
2025-08-23 12:42:19 | |__| (_) | | | |_) | (_| | | | (_| | | | (_| | || (_) | |
2025-08-23 12:42:19  \____\___/|_|_|_.__/ \__,_|_|  \__,_|_|  \__,_|\__\___/|_|
2025-08-23 12:42:19
2025-08-23 12:42:19     :: Collabera Library Management System ::
2025-08-23 12:42:19
2025-08-23 12:42:19 2025-08-23 07:12:19 [background-preinit] INFO  o.h.validator.internal.util.Version - HV000001: Hibernate Validator 8.0.3.Final
2025-08-23 12:42:19 2025-08-23 07:12:19 [main] INFO  c.c.book.library.system.Application - Starting Application v0.0.1-SNAPSHOT using Java 17.0.16 with PID 1 (/app/app.jar started by root in /app)
2025-08-23 12:42:19 2025-08-23 07:12:19 [main] DEBUG c.c.book.library.system.Application - Running with Spring Boot v3.5.5, Spring v6.2.10
2025-08-23 12:42:19 2025-08-23 07:12:19 [main] INFO  c.c.book.library.system.Application - No active profile set, falling back to 1 default profile: "default"
2025-08-23 12:42:21 2025-08-23 07:12:21 [main] INFO  o.s.d.r.c.RepositoryConfigurationDelegate - Bootstrapping Spring Data JPA repositories in DEFAULT mode.
2025-08-23 12:42:21 2025-08-23 07:12:21 [main] INFO  o.s.d.r.c.RepositoryConfigurationDelegate - Finished Spring Data repository scanning in 119 ms. Found 3 JPA repository interfaces.
2025-08-23 12:42:22 2025-08-23 07:12:22 [main] INFO  o.s.b.w.e.tomcat.TomcatWebServer - Tomcat initialized with port 8080 (http)
2025-08-23 12:42:22 2025-08-23 07:12:22 [main] INFO  o.a.coyote.http11.Http11NioProtocol - Initializing ProtocolHandler ["http-nio-8080"]
2025-08-23 12:42:22 2025-08-23 07:12:22 [main] INFO  o.a.catalina.core.StandardService - Starting service [Tomcat]
2025-08-23 12:42:22 2025-08-23 07:12:22 [main] INFO  o.a.catalina.core.StandardEngine - Starting Servlet engine: [Apache Tomcat/10.1.44]
2025-08-23 12:42:22 2025-08-23 07:12:22 [main] INFO  o.a.c.c.C.[Tomcat].[localhost].[/] - Initializing Spring embedded WebApplicationContext
2025-08-23 12:42:22 2025-08-23 07:12:22 [main] INFO  o.s.b.w.s.c.ServletWebServerApplicationContext - Root WebApplicationContext: initialization completed in 2983 ms
2025-08-23 12:42:23 2025-08-23 07:12:23 [main] INFO  o.h.jpa.internal.util.LogHelper - HHH000204: Processing PersistenceUnitInfo [name: default]
2025-08-23 12:42:23 2025-08-23 07:12:23 [main] INFO  org.hibernate.Version - HHH000412: Hibernate ORM core version 6.6.26.Final
2025-08-23 12:42:23 2025-08-23 07:12:23 [main] INFO  o.h.c.i.RegionFactoryInitiator - HHH000026: Second-level cache disabled
```