

CS3354 Software Engineering
Final Project Deliverable 1

BookWorm

David Brunner (deb210004@utdallas.edu)

Zachary Taylor (zwt190000@utdallas.edu)

Kaiden Gallardo (kkq200001@utdallas.edu)

Sam Salinas (sas200000@utdallas.edu)

Joseph Egan (jee210001@utdallas.edu)

Project Scope

University Library Web Application

1. Student Account

- a. Login to website
- b. Pay fines

2. Literature Searching

- a. Search for database
 - i. Select subject category
 - ii. Select vendor type
 - iii. Select if available for user
 - iv. Type in name search query
 - v. View description
- b. Search for article
 - i. Select subject category
 - ii. Select if scholarly or non-peer reviewed
 - iii. Select if available for user
 - iv. Type in title or author query
 - v. View citation
 - vi. View abstract
 - vii. View publishing details
- c. Search for book
 - i. Select subject category
 - ii. Select if online or print
 - iii. Select if available for user
 - iv. Type in title or author search query
 - v. View citation
 - vi. View summary
 - vii. View publishing details

3. Literature Viewing

- a. Database
 - i. Click on external link
- b. Article
 - i. Search for word and go to word
 - ii. Go to section
 - iii. Go to next/previous page
 - iv. Change font and size of text
- c. Book
 - i. Search for word and go to word
 - ii. Go to chapter

- iii. Go to next/previous page
- iv. Change font and size of text
- v. Checkout for in-person pickup if available
- vi. Add to personal collection if online and in library catalog

4. Personal Collection

- a. Category
 - i. Create category
 - ii. Delete category
 - iii. Search category for book
- b. Book
 - i. Add book to category
 - ii. Remove book from category

5. Book Editing

- a. Notate
 - i. Add text note
 - ii. Edit text note
 - iii. Delete text note
- b. Bookmark book page
 - i. Add bookmark
 - ii. Edit bookmark
 - iii. Delete bookmark
- c. Highlight text passage
 - i. Add highlight
 - ii. Edit highlight
 - iii. Delete highlight

6. Study Room Reservation

- a. Search available rooms
 - i. Select date
 - ii. Select group size
- b. Confirm reservation for a room
- c. Cancel reservation for a room

1. Project Draft Description:

Our team has decided on an interactive library platform as the topic for our project. The application should allow a user to access online content using databases, bookmark resources, upload/download book files from local systems, and annotate their saved documents. Ideally, the program can be used by educational institutions, providing their students with easy access to online resources and note-taking features for their class material.

The primary motivation for choosing this topic was its scope, allowing us to focus on different components such as the database, user interactivity, cloud computing, and the legality of online content. Nevertheless, we are also students so using an online library is familiar to all of us.

We currently do not plan to author a scholarly article after completing the project.

Instructor Feedback:

Good choice for a topic! A comprehensive library management system truly is useful and will promote student and faculty success, or individual engagement into more reading. As suggestion, please contact UTD Library staff to get more information about the details of the working of the current system, what could be improved, etc. for a better design.

It is great to see a detailed breakdown of the tasks you have considered already. Good job.

In the final report, please make sure to include comparison with similar applications -if any-, make sure that you differentiate your design from those, and explicitly specify how.

Fair delegation of tasks.

Please share this feedback with your group members.

You are good to go. Have fun with the project and hope everyone enjoys the collaboration.

2. Github Repository Information:

<https://github.com/DavidB09/3354-bookworm>

3. Delegation of Tasks:

Zachary will take on the role of project manager, keeping the team on schedule and ensuring effective work.

David will take on the role of secretary, creating/submitting the project documents, setting up any online platforms, and ensuring effective communication.

Kaiden will mainly focus on researching important topics and supplying resources for the team.

Sam will focus on the project presentation, creating/uploading the slides and ensuring everyone is prepared.

Joseph will focus on helping Kaiden with researching topics and designing graphics for the project.

It is important to note that everyone is expected to contribute to the writing of any documents or code involved with the project. Nevertheless, the previously stated roles may also change as the team progresses throughout the semester, ensuring that we will meet the deadlines for all deliverables.

4. Software Process Model:

To complete BookWorm, the incremental software process will be employed because of the following list of reasons. The diagram in Figure 1 is a graphical example of BookWorm's incremental software process model.

- BookWorm includes a repository containing access to databases, articles, and literature, meaning to have a selection of increasing size, constant updates to all components will be made as they become available.
- To release a working version of BookWorm as soon as possible, the main features, including managing a student's account, searching for specific databases, articles, and books, and viewing databases, articles, and books, will all be released before completed versions of the editing,

personal collection, and study room reservation components are fully published.

- All members of the BookWorm team will be able to fully work on multiple pieces of a single component, as well as previous bug fixes, without having to worry about working in parallel with their other team members due to the linear format of the process.
- All database, article, and literature licensing available will start out small and grow to the needed size depending on the initial impressions of BookWorm. This means that if BookWorm gains a notable, frequent audience, license sizes will increase as new increments are released.

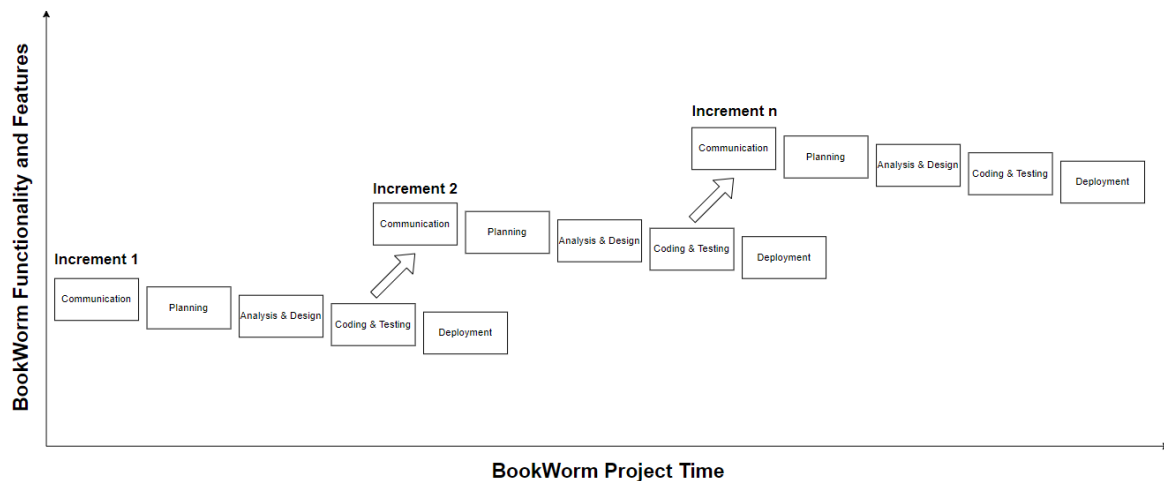


Figure 1: Incremental Software Process Model for BookWorm

5. Software Requirements:

Functional Requirements

1. Users

- a. Users should be able to log into their university account and pay fines.
- b. Authenticated users should be able to search for specific databases, articles, and books by subject category, vendor type, online or physical, availability, title, author, and academic status.

- c. Authenticated users should be able to view and navigate databases, articles, and books and see their description, publishing details, citation, and summary.
- d. Authenticated users should be able to create a personal collection of books that can be added and removed from personally created categories.
- e. Authenticated users should be able to reserve a study room by date and size if available.
- f. Authenticated users should be able to edit their copies of books by adding notes, bookmarks, and highlights.
- g. Authenticated users should be able to download licensed copies of databases, articles, and books.

Non-Functional Requirements

Usability

- The system should be easily navigable by users so that the number of help requests does not exceed 10 per hour.

Performance

- Each system request should be processed within 4 seconds.

Space

- Memory usage of the site should never exceed 1 gigabyte.

Dependability

- The site should load in 5 seconds when the number of users is less than 1000 and no more than 10 seconds when the number users exceed 5000.

Security

- The user login feature should only allow a maximum of 5 login attempts before locking the user out for 15 minutes.

Environmental

- The repository server should be kept in an area that allows it to stay within the range of 20 to 60 degrees Celsius at all times.

Operational

- The system must maintain a minimum of 20 users per day to be kept in operation. If this is not maintained for a total of 30 days, the system will shut down.

Development

- The system will be programmed in java using object-oriented programming while also using html, CSS, and JavaScript for the website.

Regulatory

- All databases, articles, and books must be checked for their relation to academics in some way shape or form to be kept in the repository.

Ethical

- All databases, articles, and books must be properly licensed in order to prevent copyright infringement and piracy.

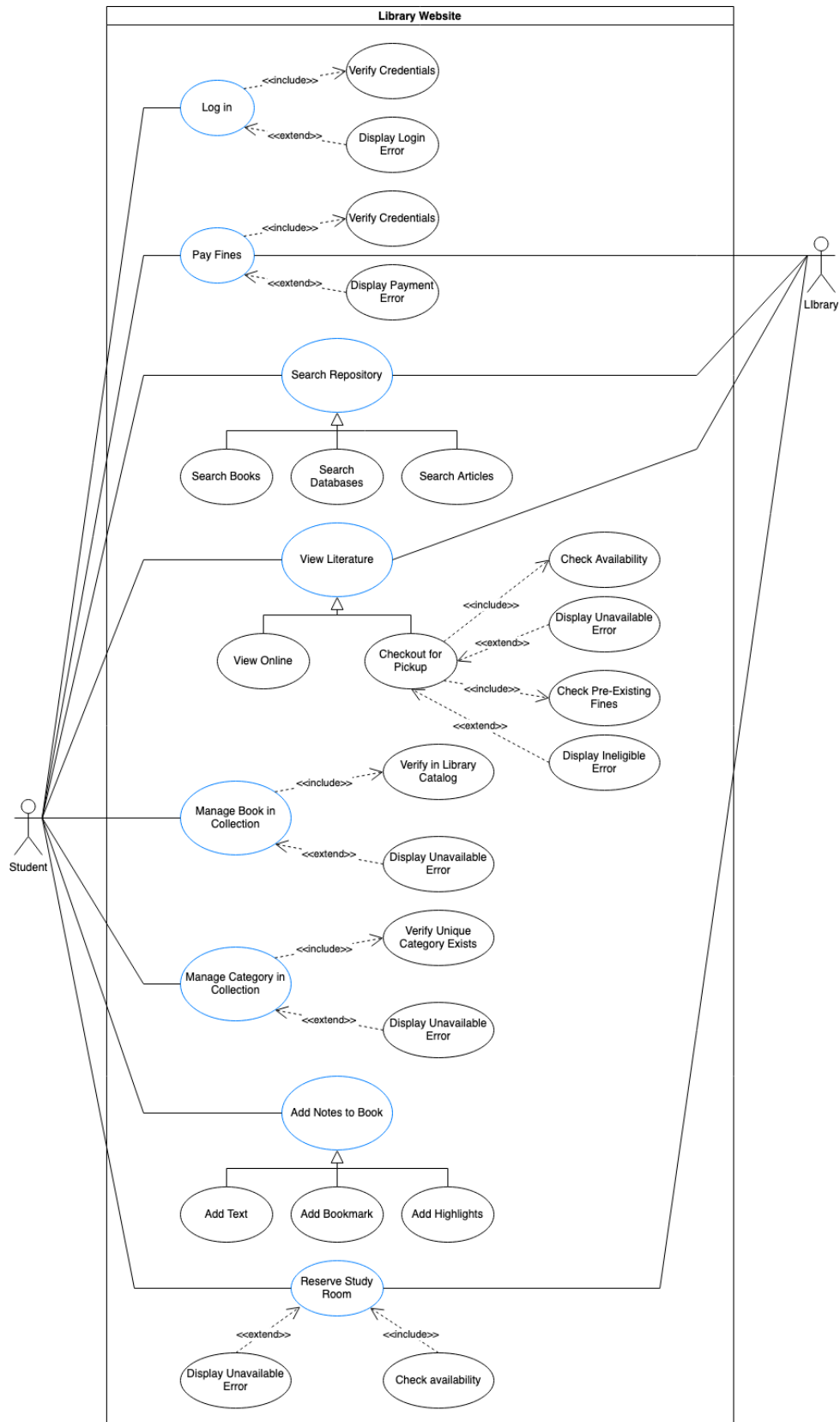
Accounting

- The upkeep of all systems must not exceed \$100,000 a year, including the cost of licenses, webhosting, and employment.

Safety/Security

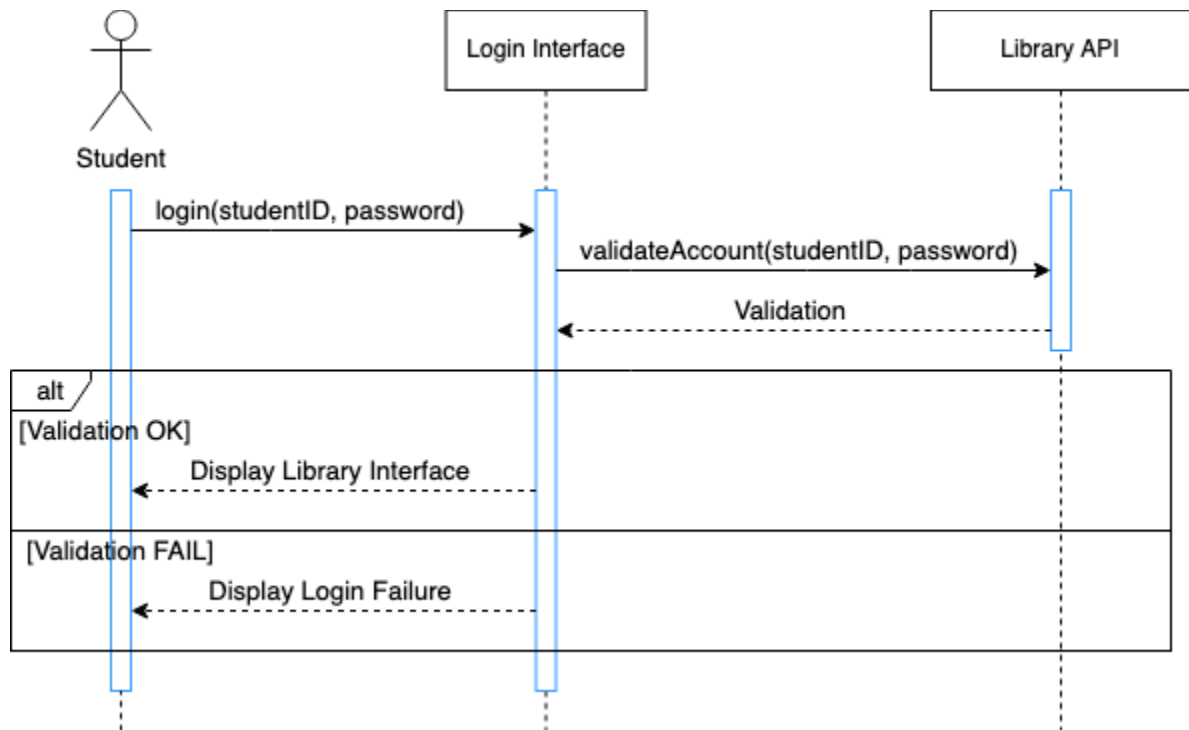
- All databases, articles, and books must be scanned for malware and viruses by an expert beforehand to ensure that users are not exposed to harm.

6. Use Case Diagram:

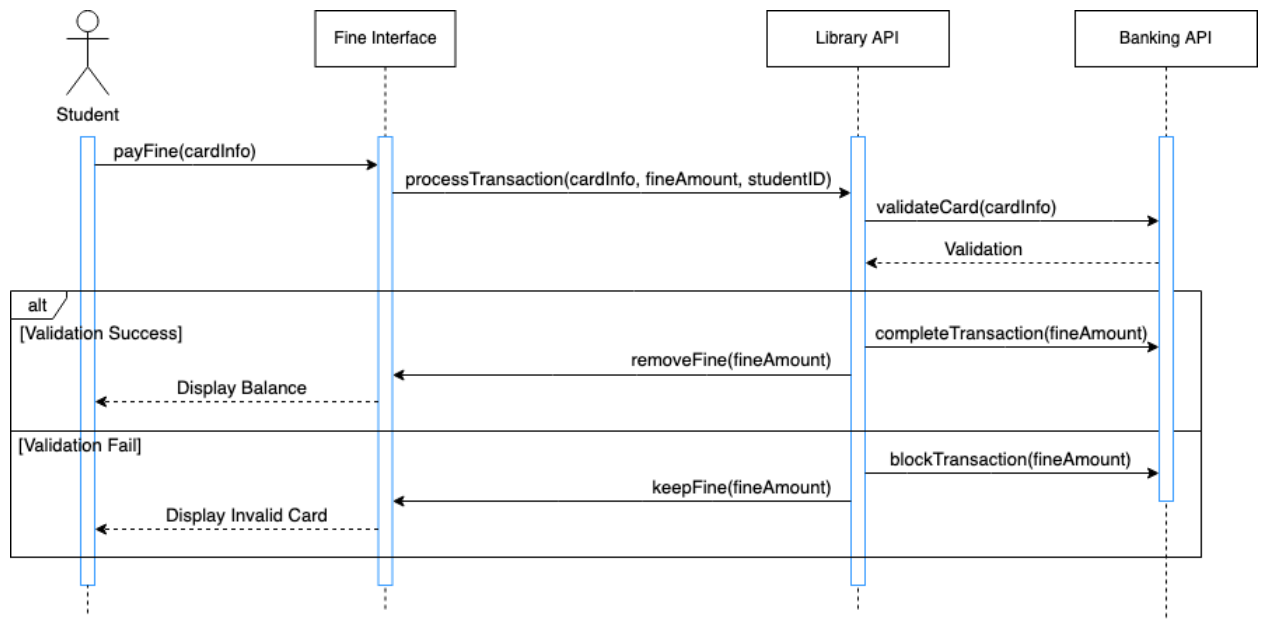


7. Sequence Diagrams:

User Login:

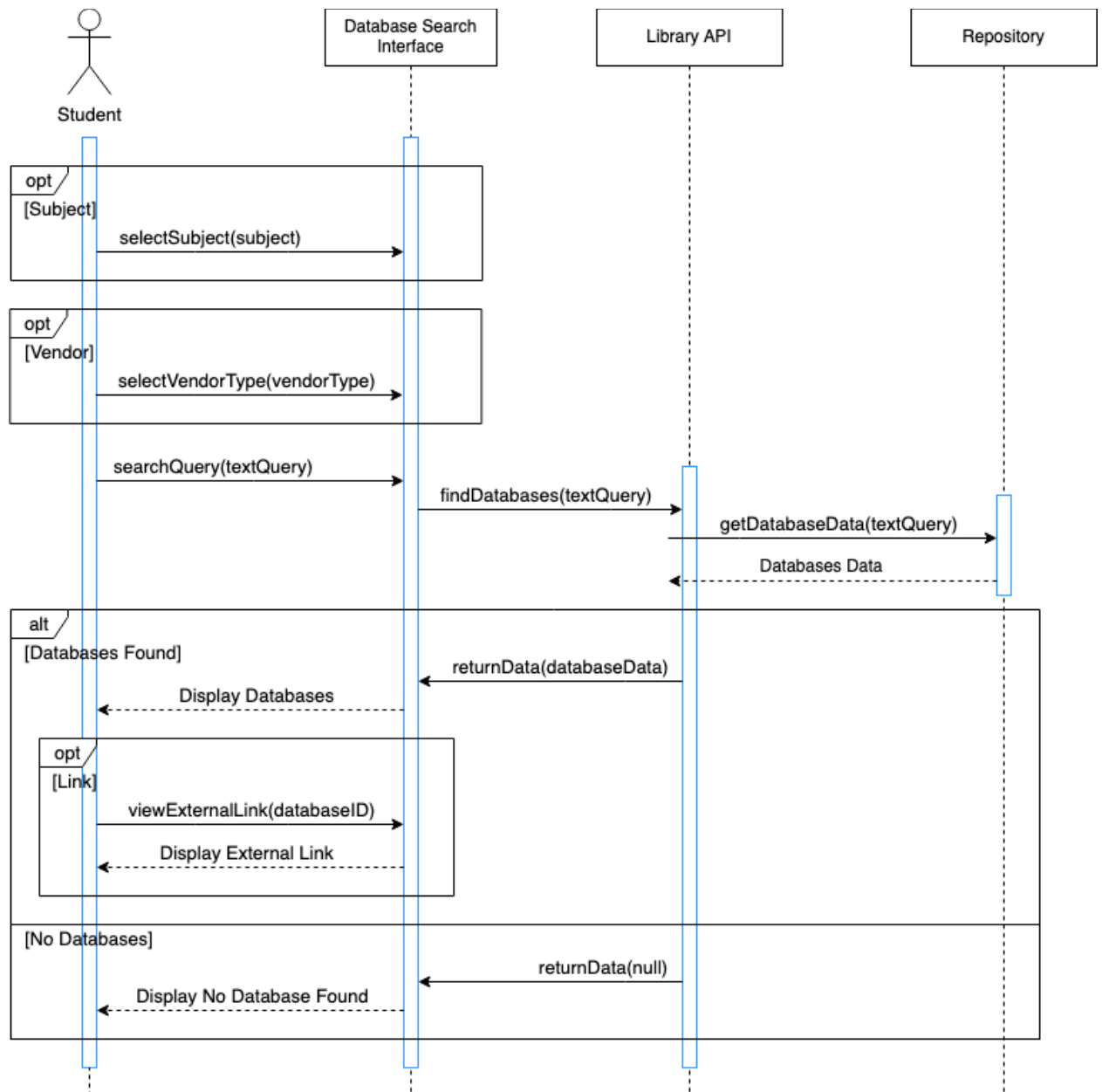


User Pay Fine:

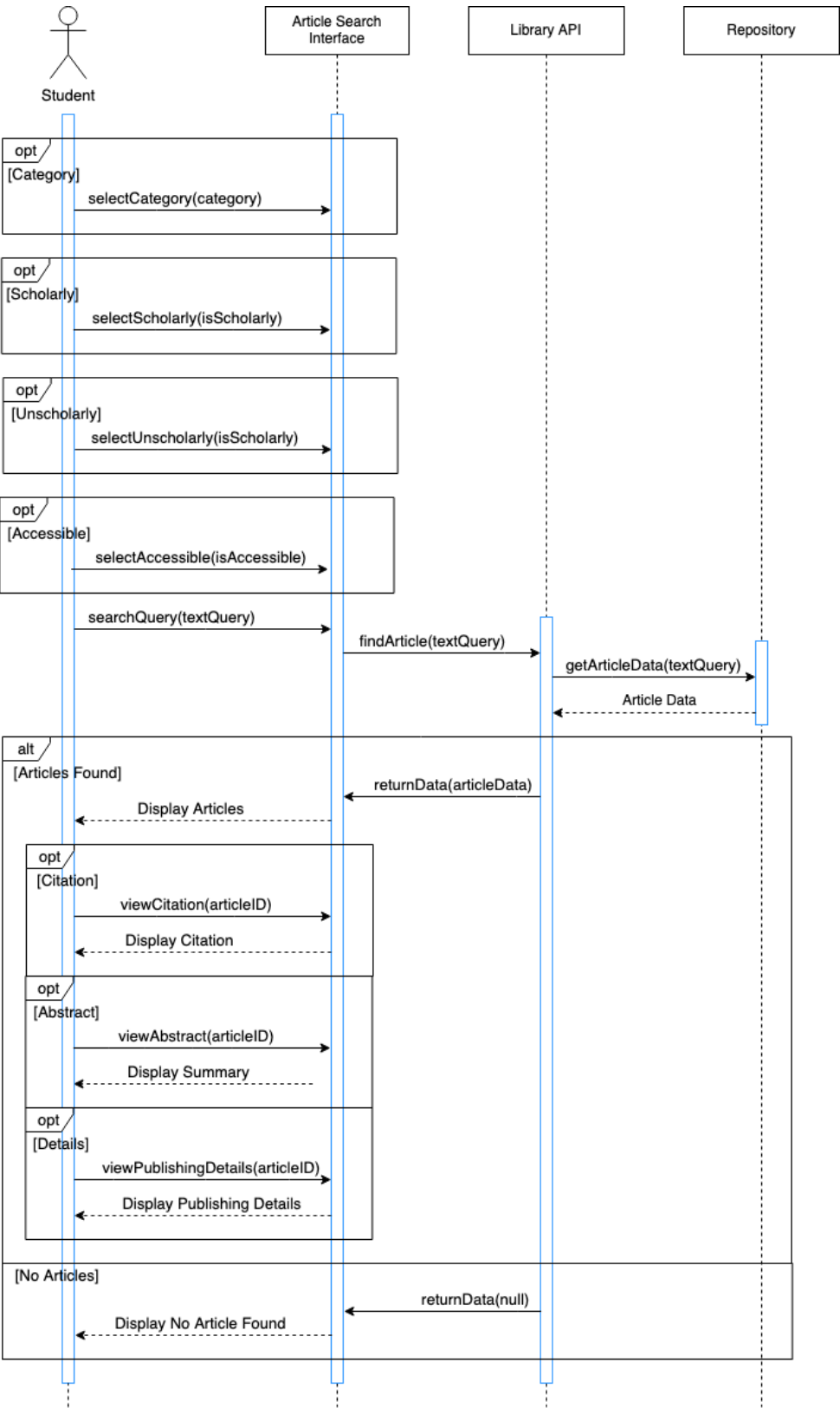


User Search:

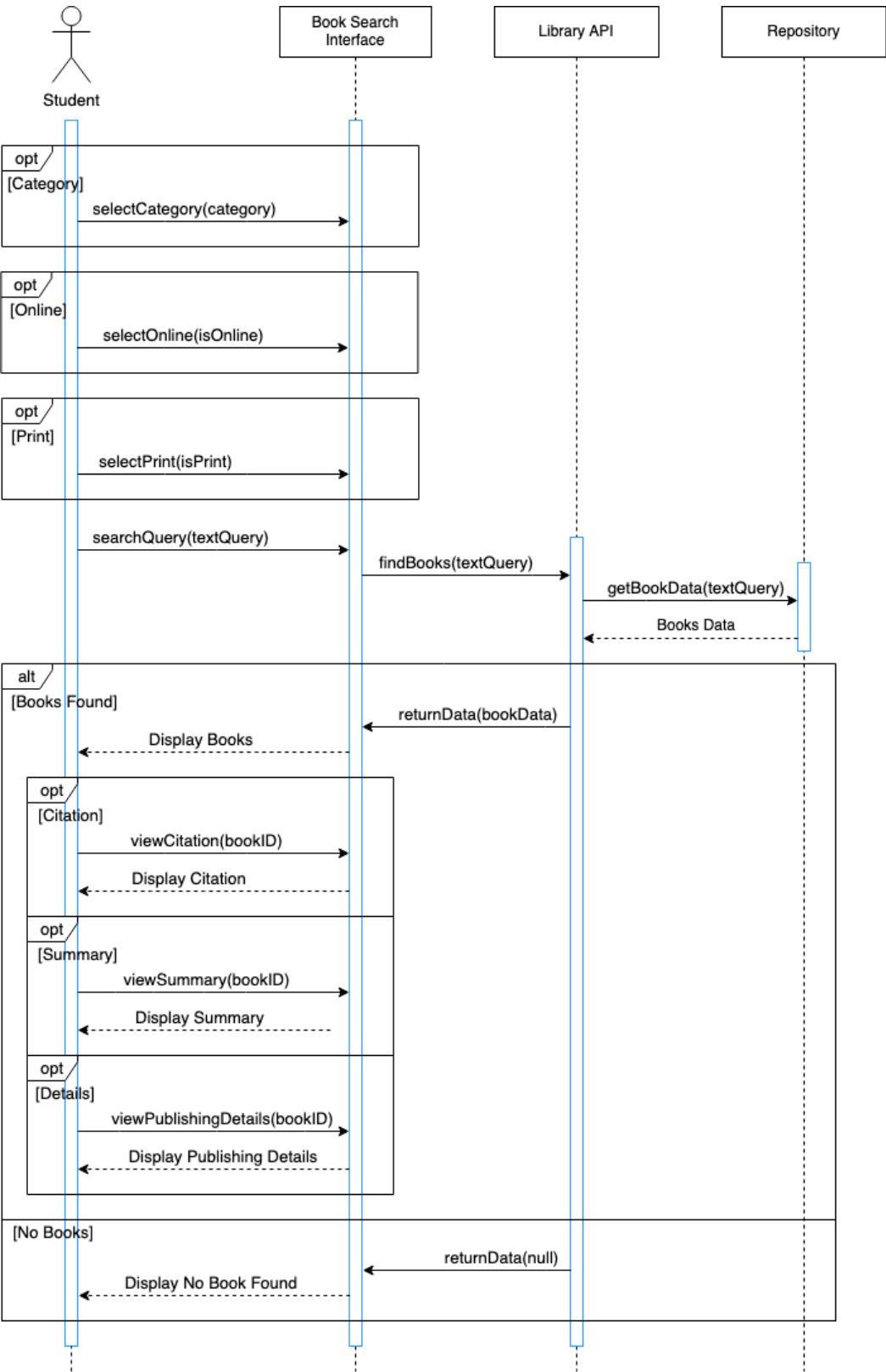
Database Search:



Article Search:

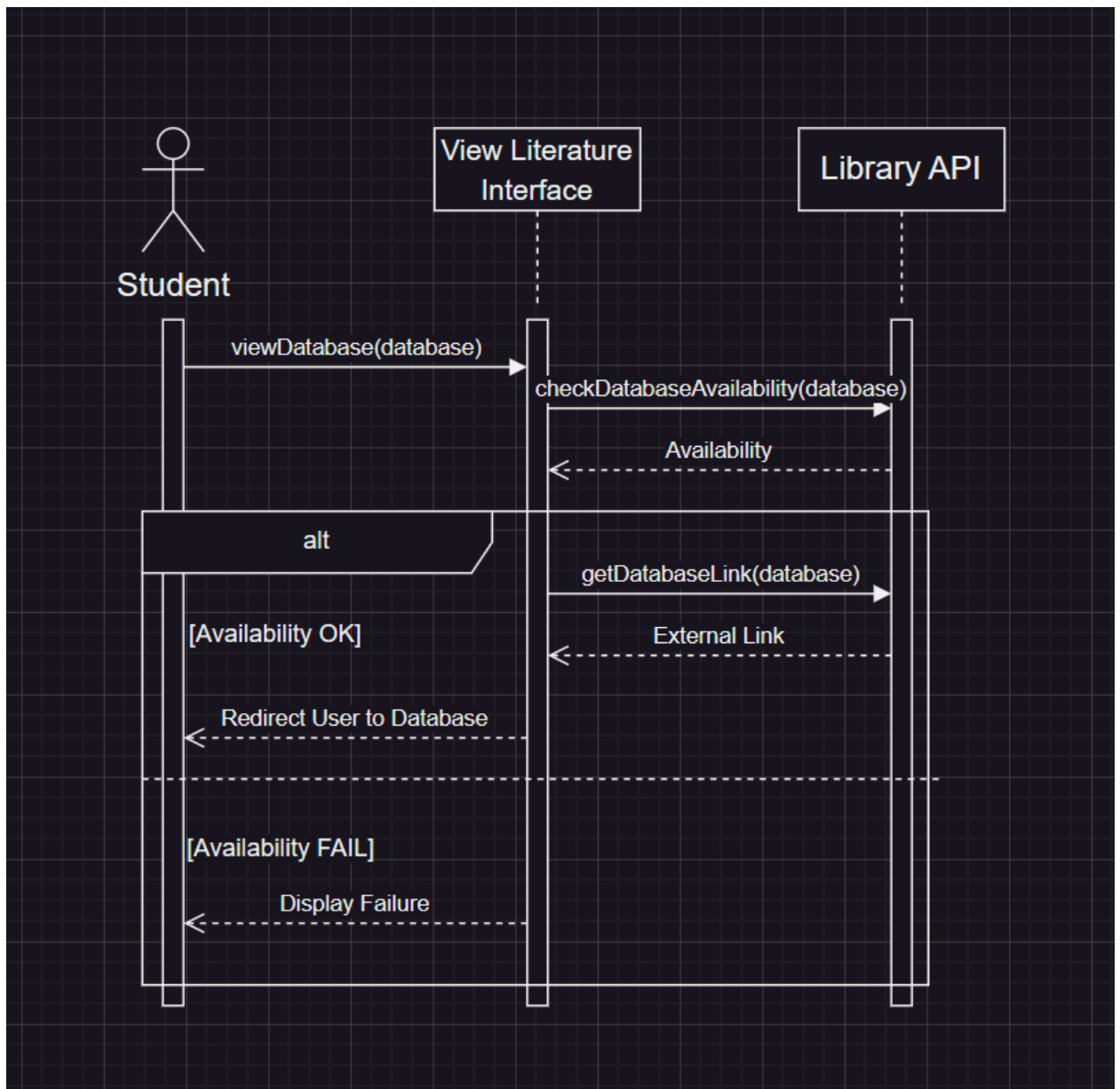


Book Search:

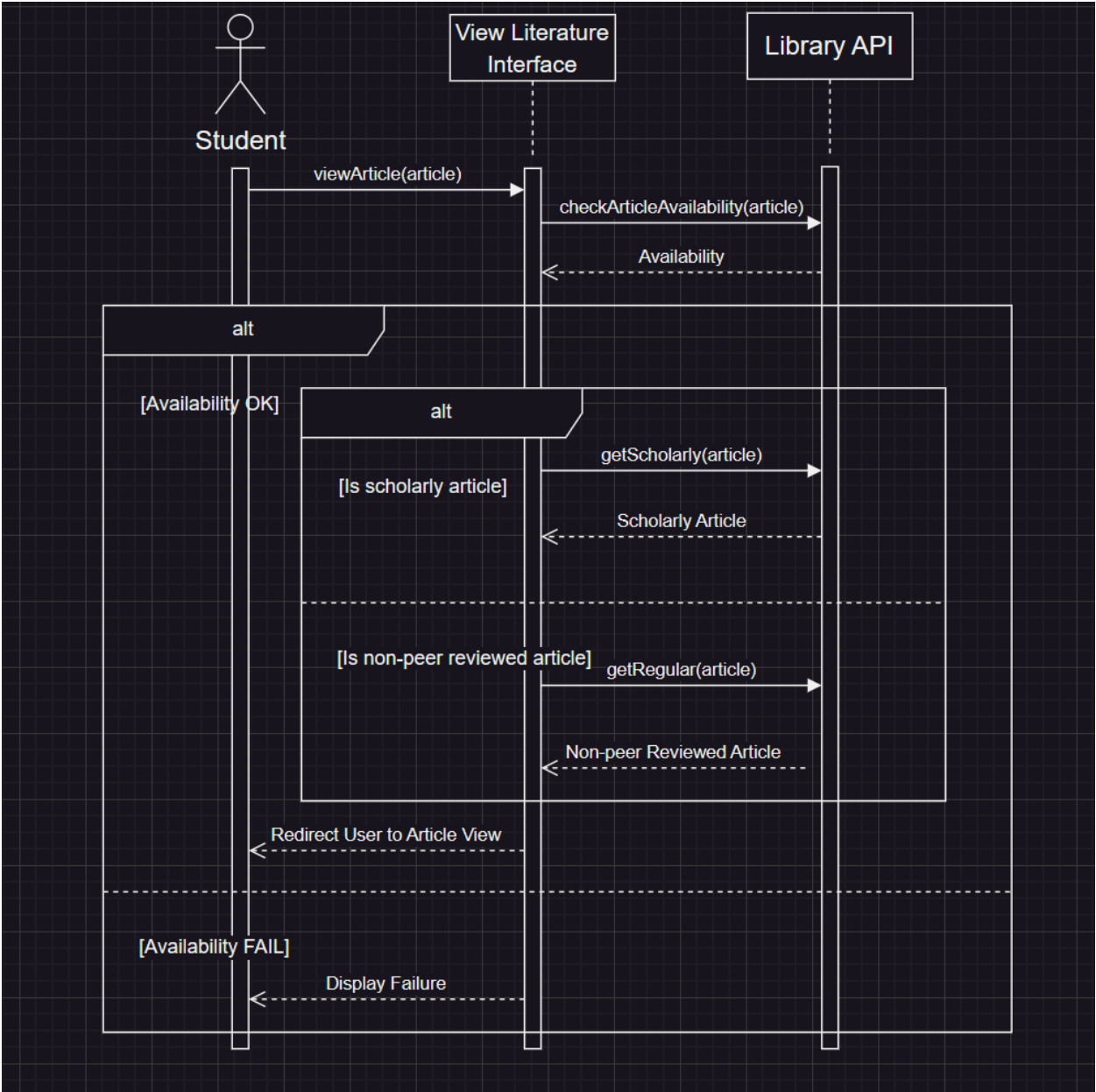


User View:

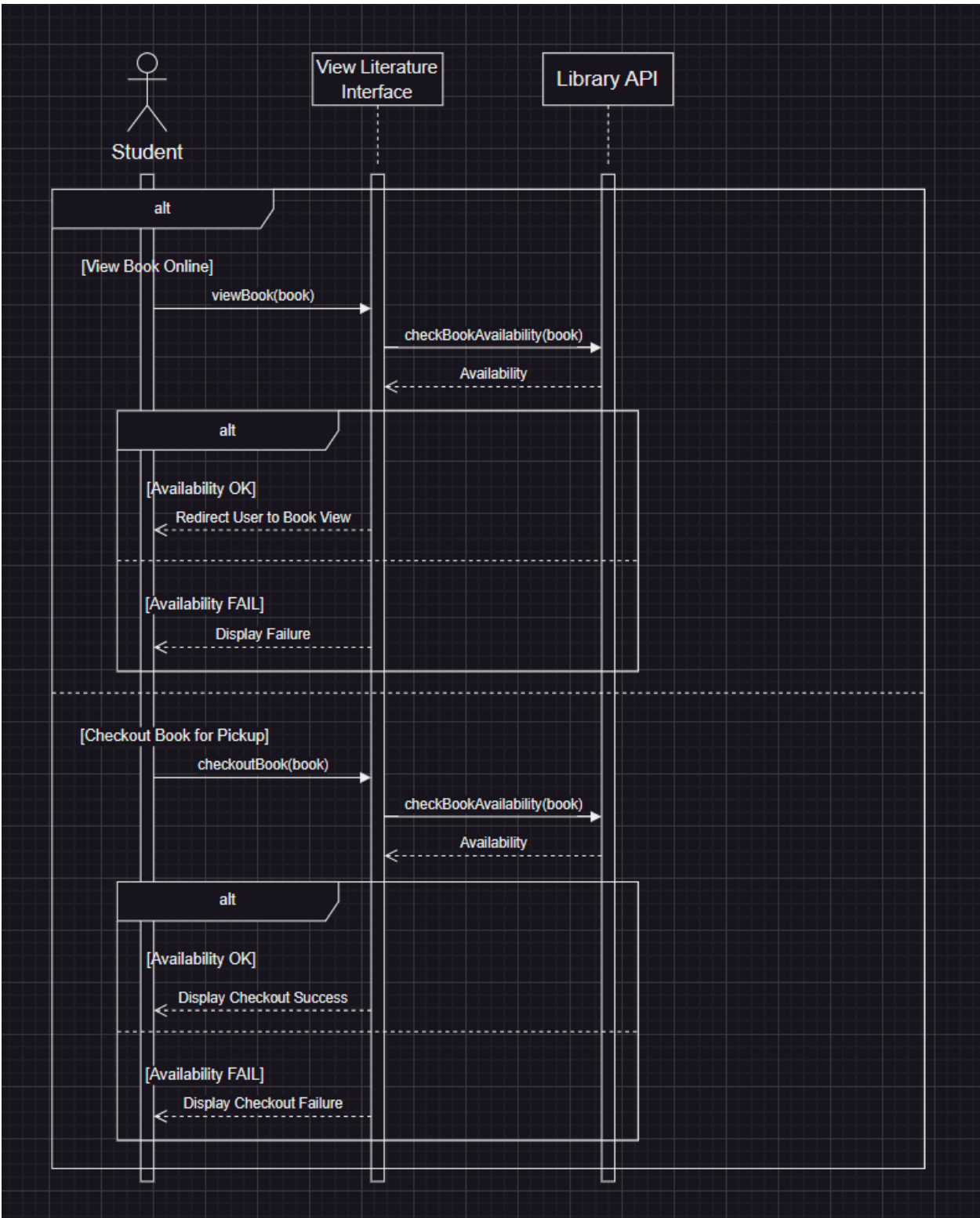
Database View:



Article View:

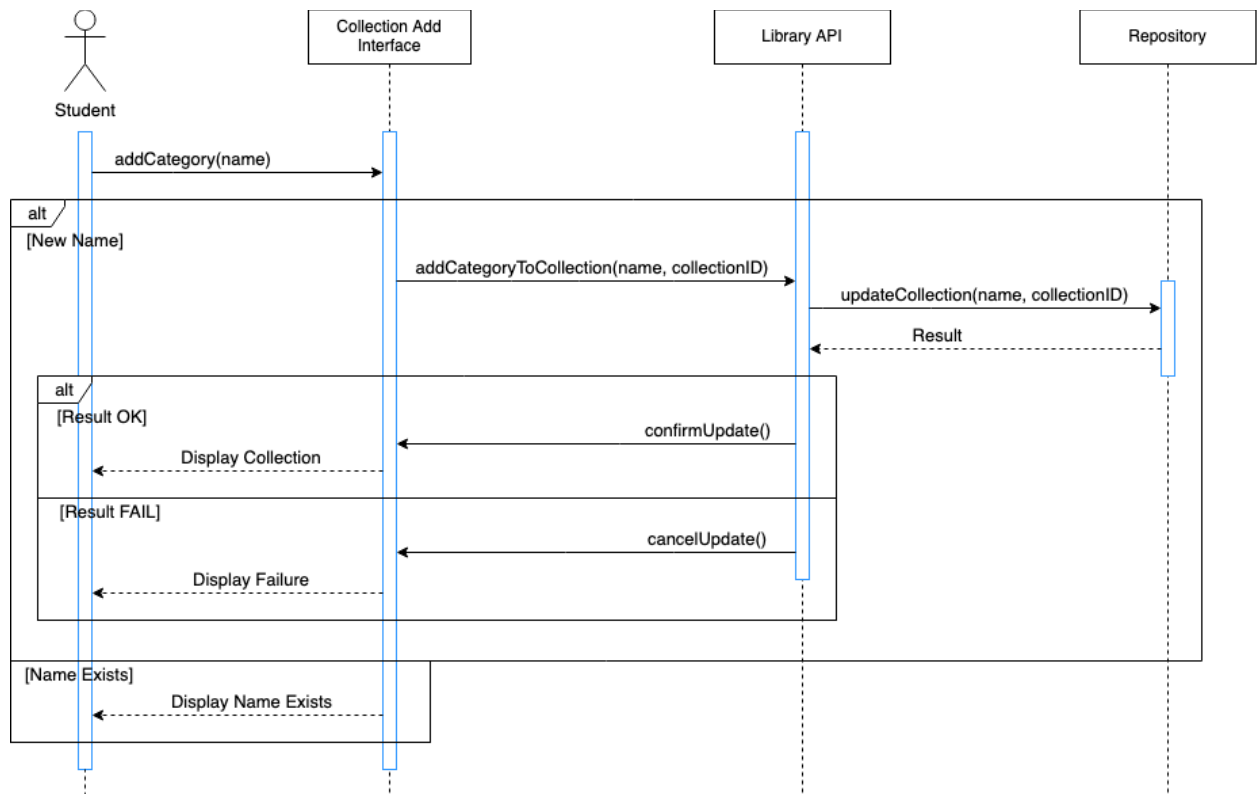


Book View:

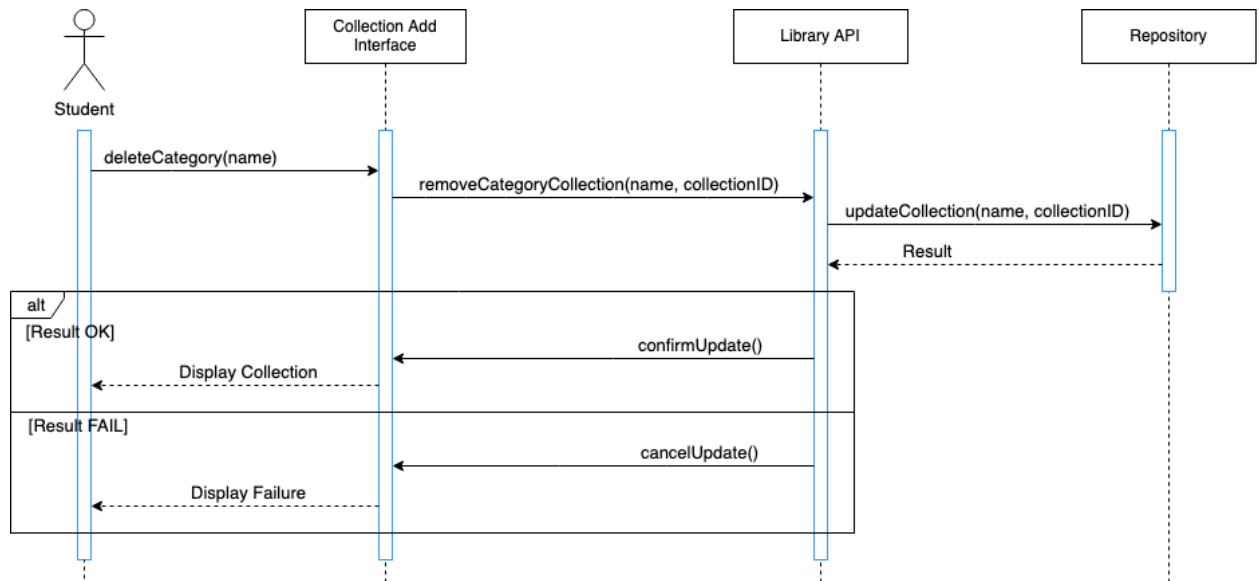


Category Management:

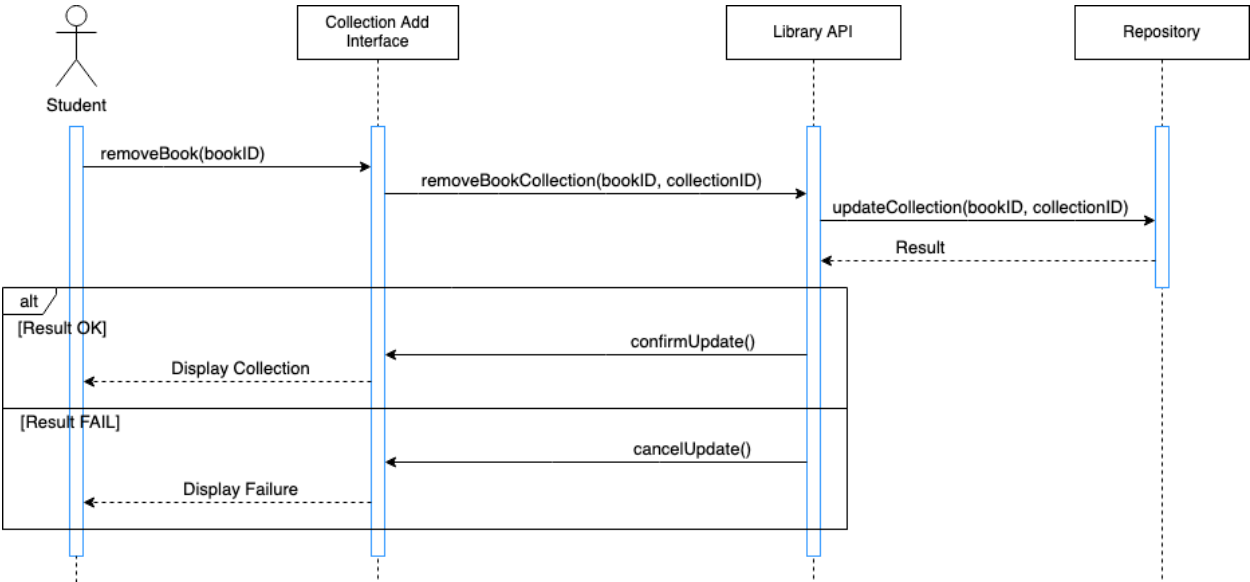
Add Category:



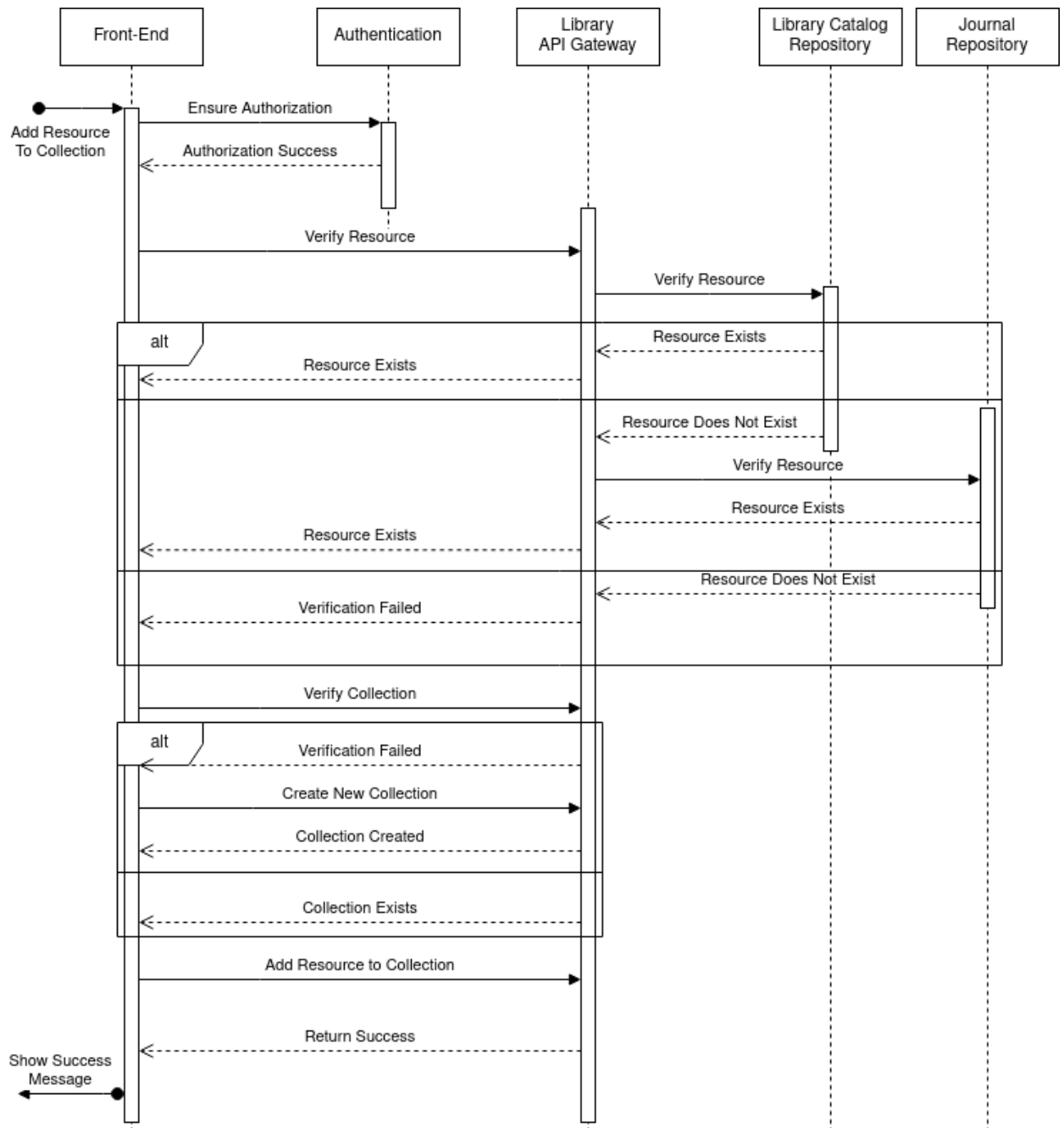
Delete Category:



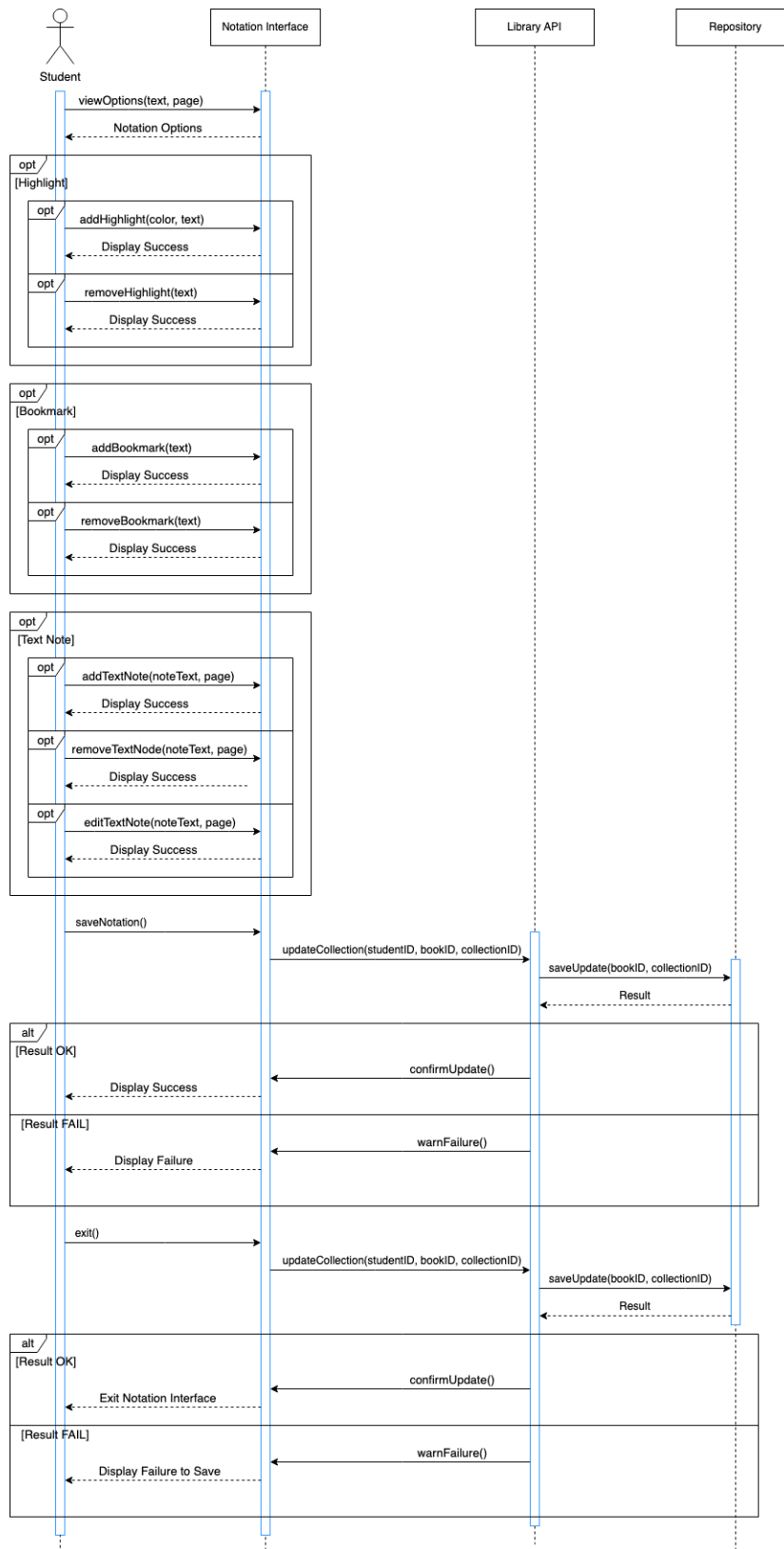
Remove Book from Collection Category:



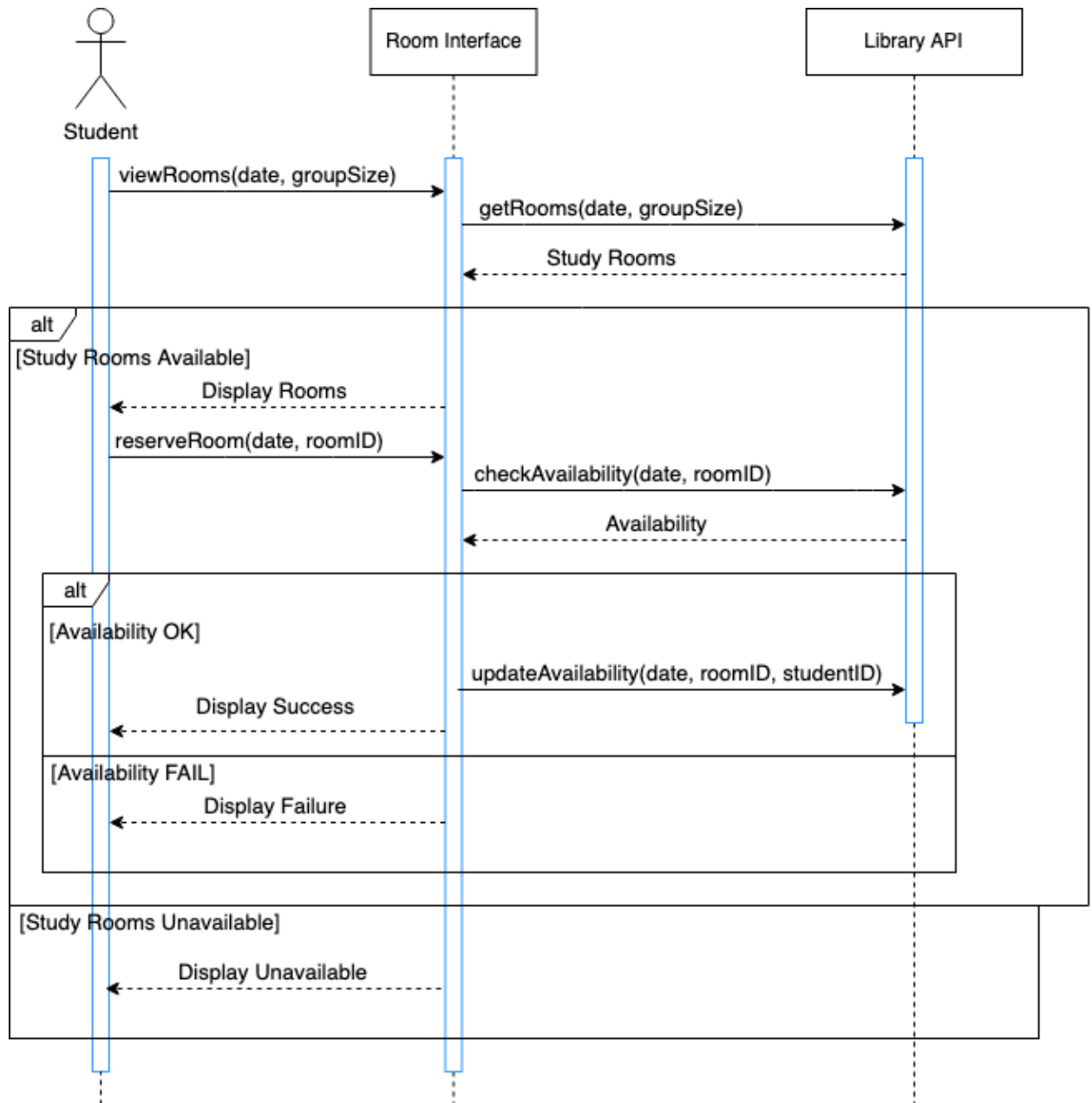
Add Book to Collection Category:



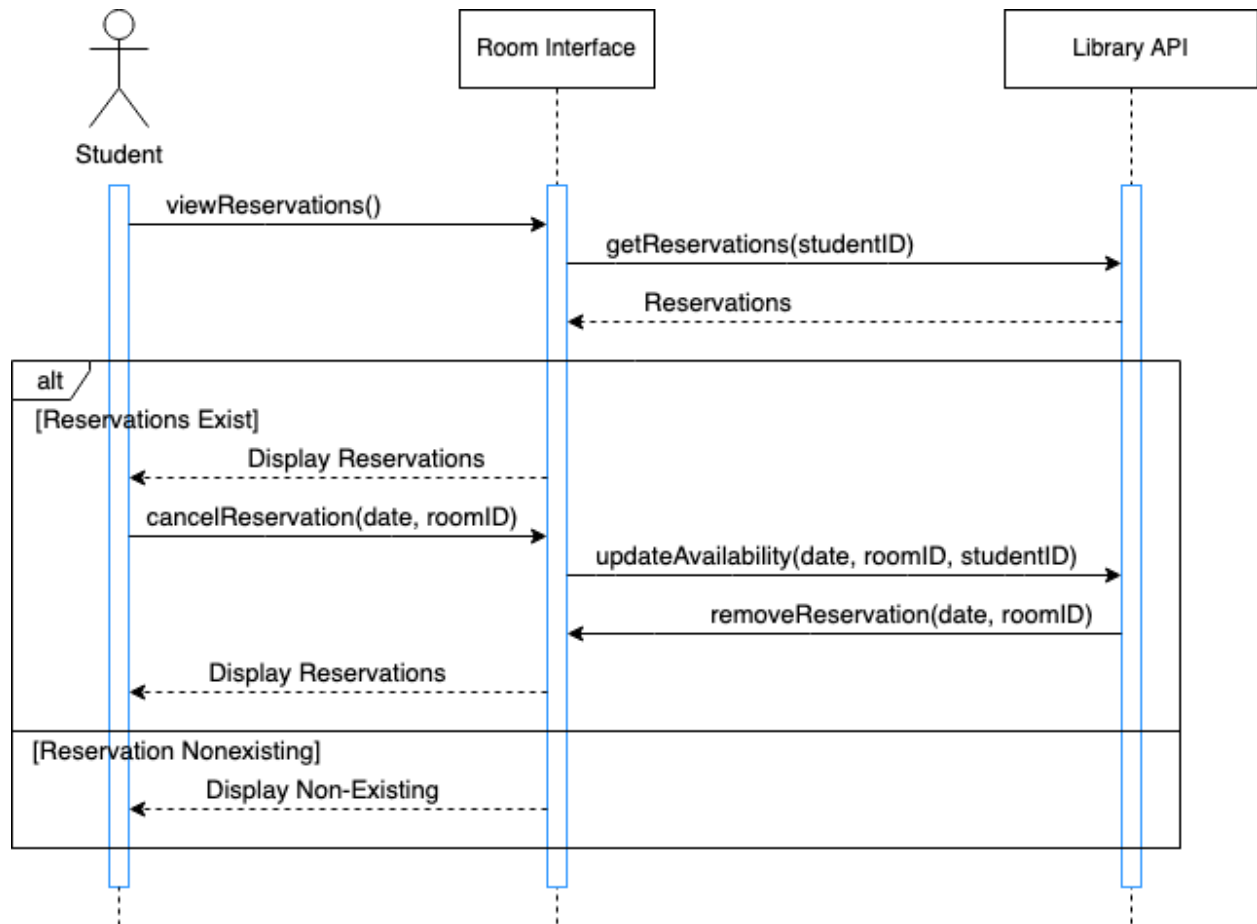
Book Editing:



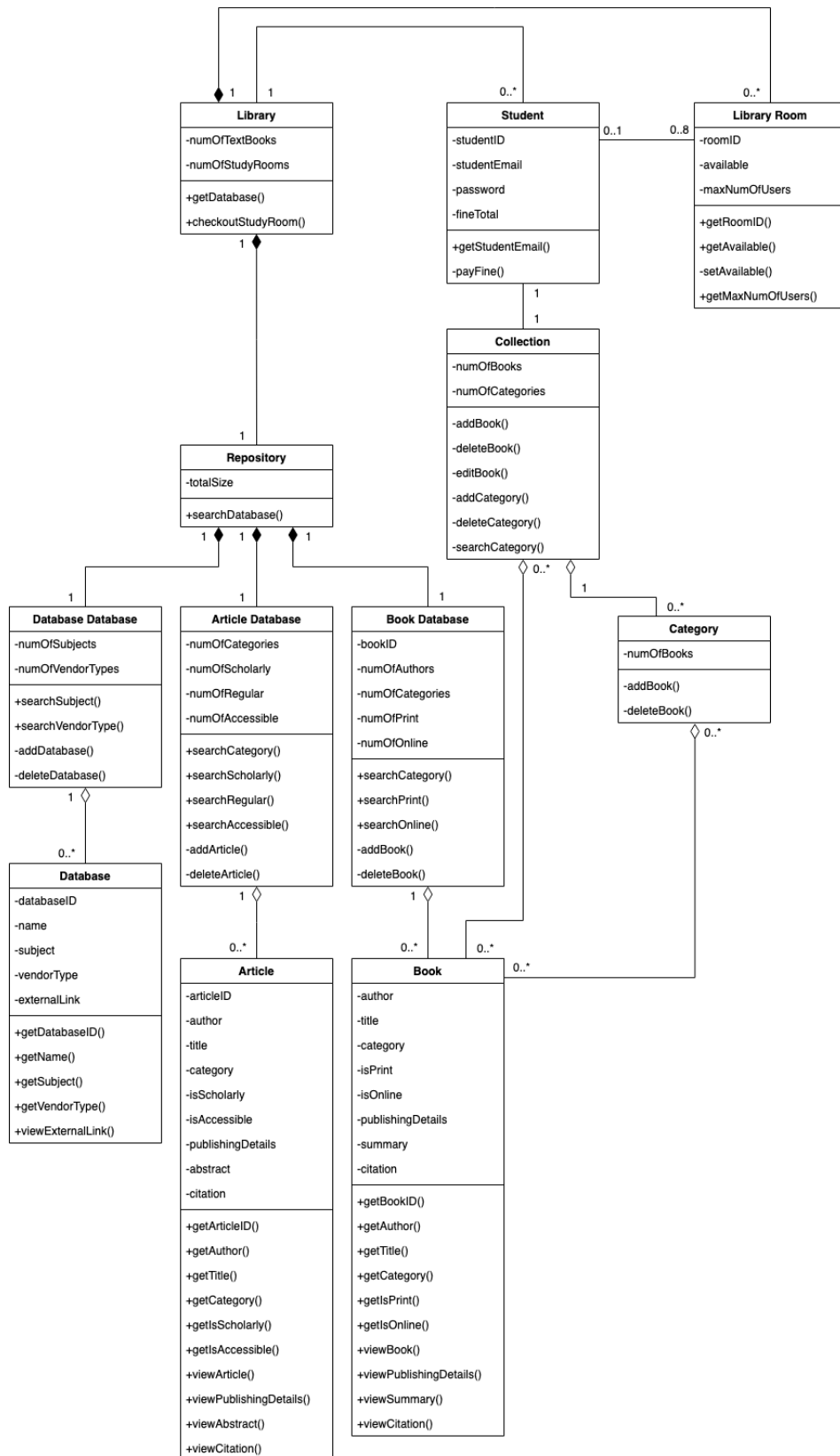
Reserve Study Room:



Cancel Study Room Reservation:



8. Class Diagram:



9. Architectural Design:

Repository Architecture Pattern:

