

Library Management System

<https://github.com/id100T/oopfinalprojectLMS>

Group#

Gavin Qi yq2056@nyu.edu (id100t)

Eric Zhang sz3586@nyu.edu (Ericzzsh)

20250509

Table of Work

(Please write x in the boxes to mention what each student achieved in this project)

	Gavin Qi	Eric Zhang	
Project Description	x		
Uses Cases Diagram(s)	x		
Sequence Diagrams		x	
Class diagram(s)		x	
Implementation	x		
Conclusion	x		

Table of Contents

- Terminology Glossary (If needed)
- System Analysis
 - Project Description (One Page)
 - General Description, Goals and Benefits
 - System input(s) and output(s)
 - Special requirements (Performance, Interfaces, Constraints, Reliability, if any)
 - Uses Cases Diagram(s) and use cases description.
- System Design
 - Sequence Diagrams
 - Class diagram(s)
- Conclusion
- Appendix (any related reports, questionnaires, docs.. If any).

Project Description

Library Management System

Team Members:

1. Gavin Qi (yq2056@nyu.edu)
2. Eric Zhang(sz3586@nyu.edu)

Description of Project:

We are creating a GUI based application for library management, the two types of user would be the librarians and library visitors. The application would deal with book related and personal data management. We will use a file to store the data.

Goal and Benefit:

To provide a management system for the library management so they can manage their visitors and book status.

Features:

1: Librarians

Librarians will deal with book management, and track book borrowing information.

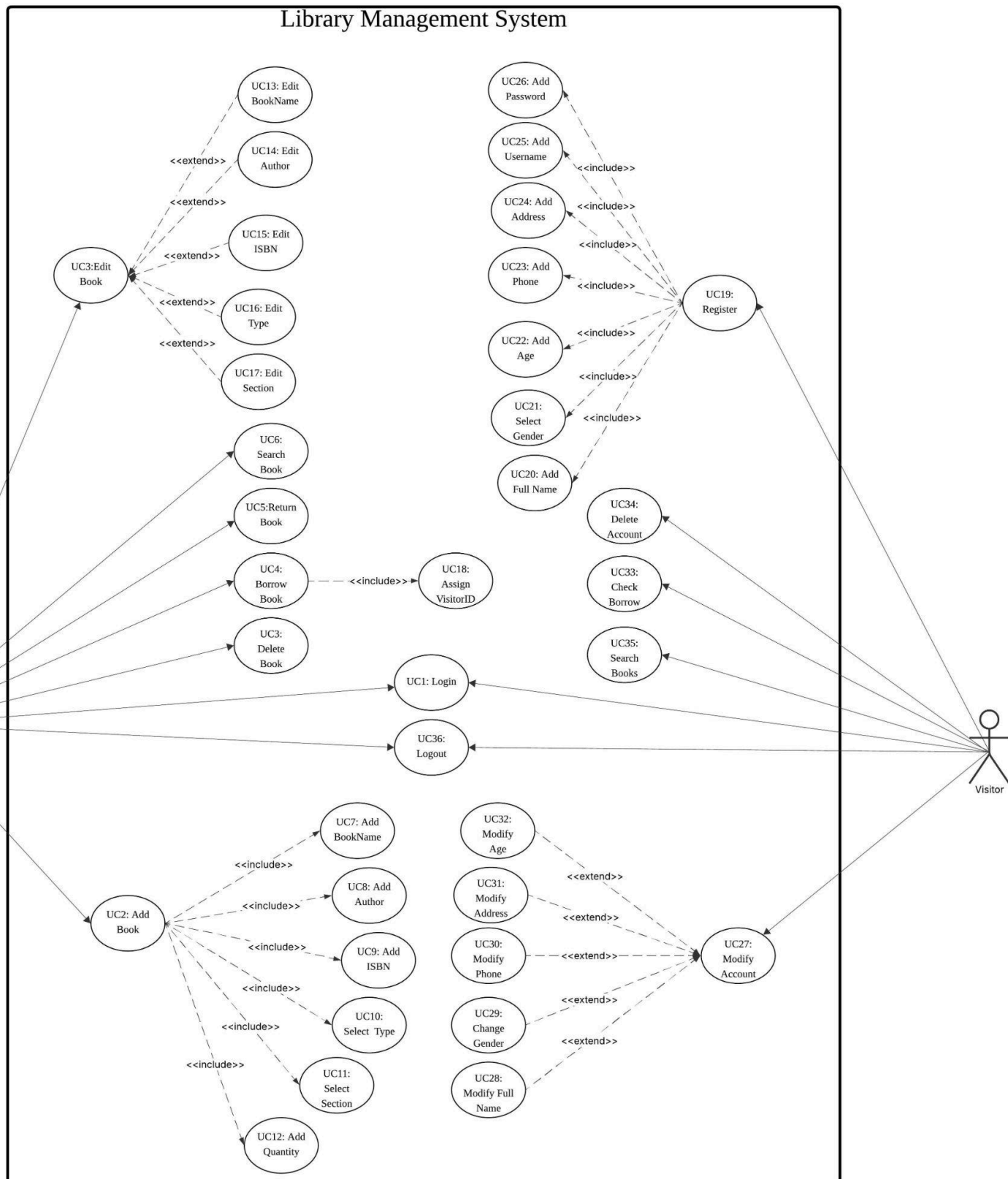
- **Add Books:** Add a new book to the library system by entering the book's **author**, **name**, **ISBN**, type (selected from a predefined list), section (S1 / S2 / S3), and **status** (default = Available), automatically assign a **copy ID** to each identical book (if only one copy, the copy ID is ISBN+1; the ID does not change if a book is deleted)
- **View Books:** Display the full catalogue of the books, showing **author**, **name**, **ISBN**, **type**, **section**, **copy ID**, **status** of each copy.
- **Modify Books:** Edit a book's **author**, **name**, **ISBN**, **type**, **section**
- **Search Books:** Search a book by entering a book's **author**, **name**, **ISBN**
- **Delete Books:** Delete a book, if there are multiple copies, delete the copy ID not the book, if there is no copy left, delete the book.
- **Issue Books:** Register a book loan for a visitor, automatically assign a borrowing date, assign the book to a **visitor ID**, status become unavailable, each visitor can borrow 10 at most at once.
- **Return Books:** Close a book loan when it is returned, automatically assign a borrowing date, book status become available

2: Library Visitors

Library Visitors will have their profile and can use the system to search for books, borrow books, and return books

- **Create Profile:** Create a personal profile with **name**, **gender**, **age**, **phone** and **address**; the system assigns a unique **visitor ID**.
- **Modify Profile:** Update/fix user profile's name, gender, age, address
- **Delete Profile:** Delete the profile if you do not want to use this library anymore.
- **Search Books:** search a book by author, name, ISBN, type
- **View Books:** Browse the complete catalogue of a book with full details (name, ISBN, type, section, status, etc.).
- **View Status:** Display a personal borrowing summary showing borrowed books, and the remaining quota of books available for borrowing.

Librarian Account: admin Password:123456



UC1: Login

Overview: Librarian and Visitor can login to the Library Management System

Related Use Cases: NA

Actors: Librarian, Visitor

UC2: Add Book

Overview: Librarian can add new books to the LMS (Done after login to the system)

Related Use Cases: UC7(Add BookName), UC8(Add Author), UC9(Add ISBN), UC10(Select Type), UC11(Selection Section), UC12(Add Quantity)

Actor: Librarian

UC3: Edit Book

Overview: Librarian can edit book information in the LMS (Done after login to the system)

Related Use Cases: NA

Actor: Librarian

UC4: Borrow Book

Overview: Librarian can loan books to visitors (Done after login to the system)

Related Use Cases: UC35(Assign VisitorID)

Actor: Librarian

UC5: Return Book

Overview: Librarian can mark the book as returned when visitor return the book (Done after login to the system)

Related Use Cases: NA

Actor: Librarian

UC6: Search Book

Overview: Librarian can search a book from the book list (Done after login to the system)

Related Use Cases: NA

Actor: Librarian

UC7: Add BookName

Overview: Librarian need to add a book name to the new book when adding it to the system (Done after login to the system)

Related Use Cases: UC2(Add Book)

Actor: Librarian

UC8: Add Author

Overview: Librarian need to add a book author to the new book when adding it to the system (Done after login to the system)

Related Use Cases: UC2(Add Book)

Actor: Librarian

UC9: Add ISBN

Overview: Librarian need to add a book ISBN to the new book when adding it to the system (Done after login to the system)

Related Use Cases: UC2(Add Book)

Actor: Librarian

UC10: Select Type

Overview: Librarian need to select a book type to the new book when adding it to the system (Done after login to the system)

Related Use Cases: UC2(Add Book)

Actor: Librarian

UC11: Select selection

Overview: Librarian need to select a book selection to the new book when adding it to the system (Done after login to the system)

Related Use Cases: UC2(Add Book)

Actor: Librarian

UC12: Add Quantity

Overview: Librarian need to add a book quantity to the new book when adding it to the system (Done after login to the system)

Related Use Cases: UC2(Add Book)

Actor: Librarian

UC13: Edit BookName

Overview: Librarian can edit the book name of a book when needed (Done after login to the system)

Related Use Cases: UC3(Edit Book)

Actor: Librarian

UC14: Edit Author

Overview: Librarian can edit the book author of a book when needed (Done after login to the system)

Related Use Cases: UC3(Edit Book)

Actor: Librarian

UC15: Edit ISBN

Overview: Librarian can edit the book ISBN of a book when needed (Done after login to the system)

Related Use Cases: UC3(Edit Book)

Actor: Librarian

UC16: Edit Type

Overview: Librarian can edit the book type of a book when needed (Done after login to the system)

Related Use Cases: UC3(Edit Book)

Actor: Librarian

UC17: Edit Section

Overview: Librarian can edit the book section of a book when needed (Done after login to the system)

Related Use Cases: UC3(Edit Book)

Actor: Librarian

UC18: Assign VisitorID

Overview: Librarian needs to assign visitor ID to the book visitor borrows (Done after login to the system)

Related Use Cases: UC4(Borrow Book)

Actor: Librarian

UC19: Register

Overview: Visitor can register for an account for the LMS

Related Use Cases: UC20(Add Full Name), UC21(Select Gender), UC22(Add Age), UC23(Add Phone), UC24(Add Address), UC25(Add Username), UC26(Add Password)

Actor: Visitor

UC20: Add Full Name

Overview: Visitor need to add their full name when registering

Related Use Cases: UC19(Register)

Actor: Visitor

UC21: Select Gender

Overview: Visitor need to add their gender when registering

Related Use Cases: UC19(Register)

Actor: Visitor

UC22: Add Age

Overview: Visitor need to add their age when registering

Related Use Cases: UC19(Register)

Actor: Visitor

UC23: Add Phone

Overview: Visitor need to add their phone when registering

Related Use Cases: UC19(Register)

Actor: Visitor

UC24: Add Address

Overview: Visitor need to add their address when registering

Related Use Cases: UC19(Register)

Actor: Visitor

UC25: Add Username

Overview: Visitor need to add their username when registering

Related Use Cases: UC19(Register)

Actor: Visitor

UC26: Add Password

Overview: Visitor need to add their password when registering

Related Use Cases: UC19(Register)

Actor: Visitor

UC27: Modify Account

Overview: Visitor can modify their account information when needed (Done after login to the system)

Related Use Cases: NA

Actor: Visitor

UC28: Modify Full Name

Overview: Visitor can modify their full name when modifying their account (Done after login to the system)

Related Use Cases: UC27(Modify Account)

Actor: Visitor

UC29: Change Gender

Overview: Visitor can modify their gender when modifying their account (Done after login to the system)

Related Use Cases: UC27(Modify Account)

Actor: Visitor

UC30: Modify Phone

Overview: Visitor can modify their phone when modifying their account (Done after login to the system)

Related Use Cases: UC27(Modify Account)

Actor: Visitor

UC31: Modify Address

Overview: Visitor can modify their address when modifying their account (Done after login to the system)

Related Use Cases: UC27(Modify Account)

Actor: Visitor

UC32: Modify Age

Overview: Visitor can modify their age when modifying their account (Done after login to the system)

Related Use Cases: UC27(Modify Account)

Actor: Visitor

UC33: Check Borrow

Overview: Visitor can check their borrowing status in the system (Done after login to the system)

Related Use Cases: NA

Actor: Visitor

UC34: Delete Account

Overview: Visitor can delete their account when they want to (Done after login to the system)

Related Use Cases: NA

Actor: Visitor

UC35: Search Books

Overview: Visitor can search the book in the system (Done after login to the system)

Related Use Cases: NA

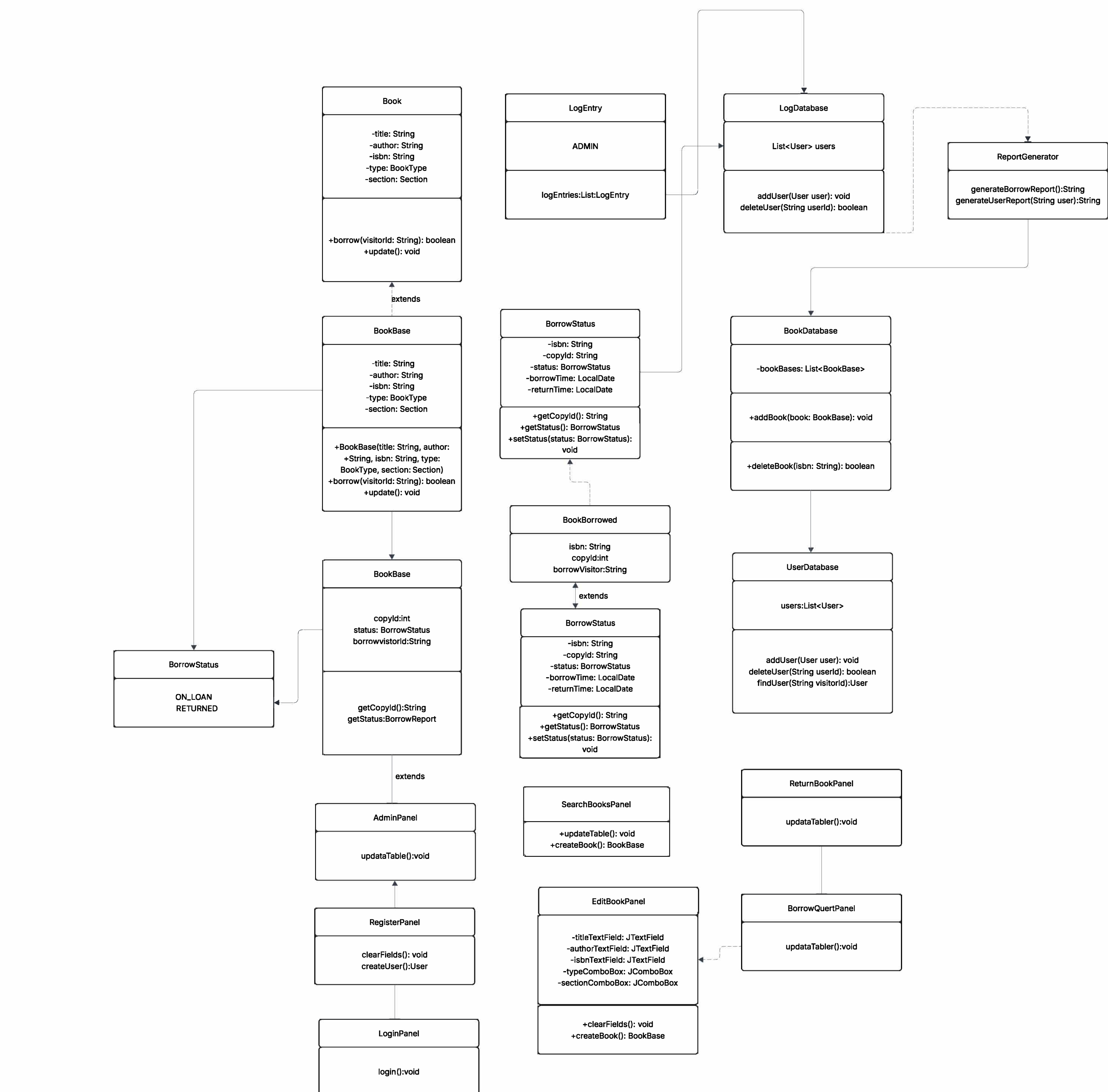
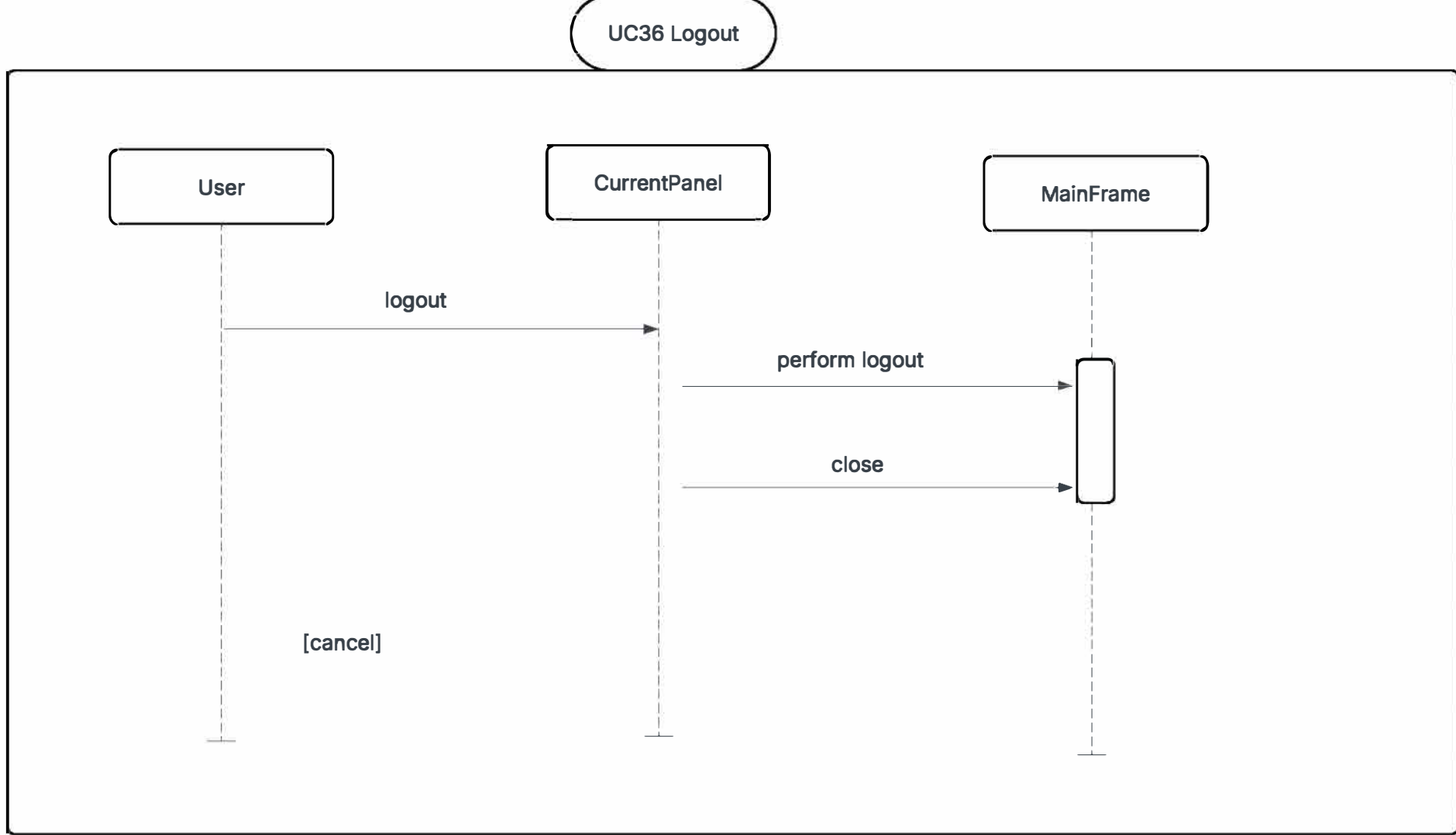
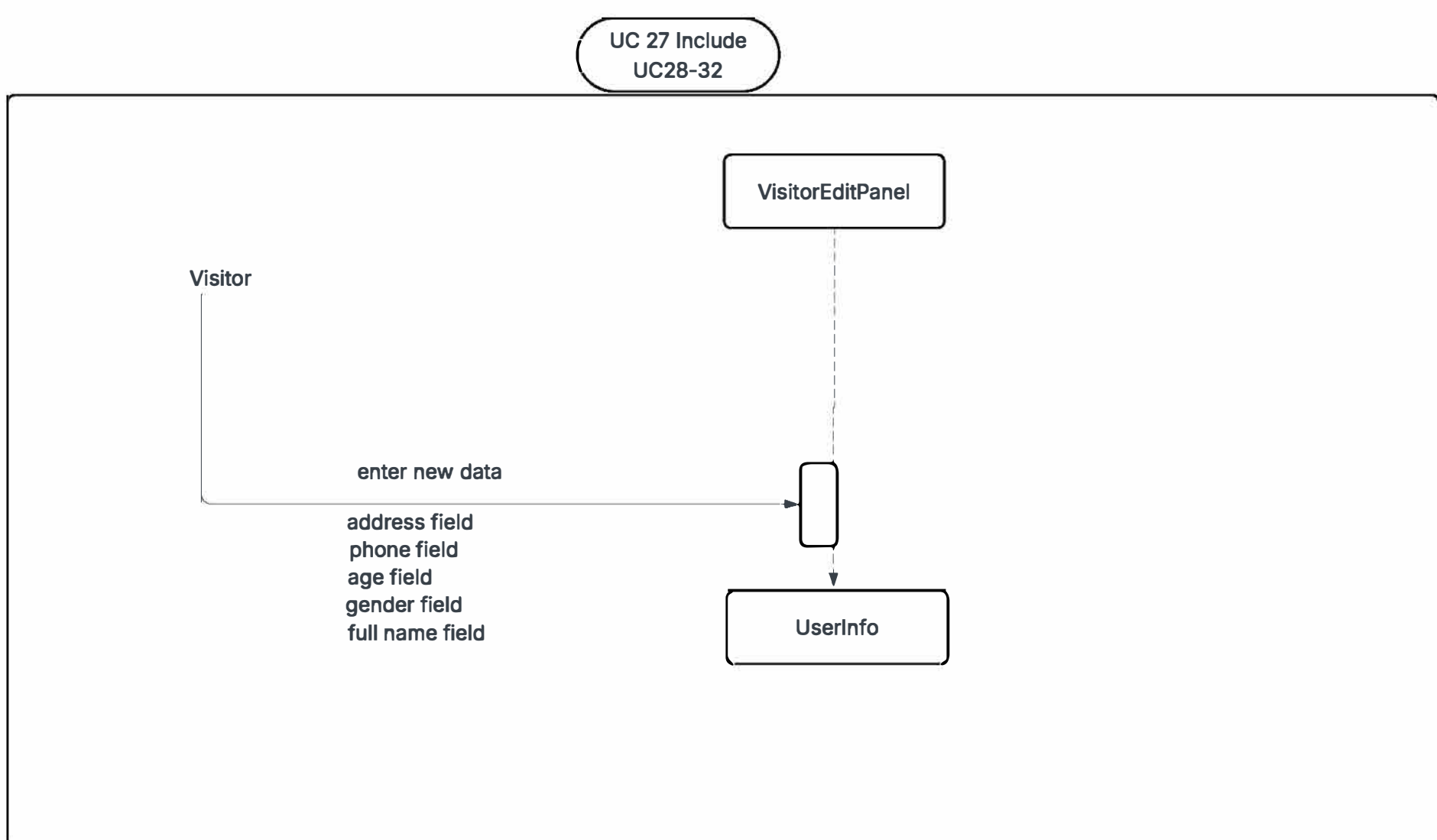
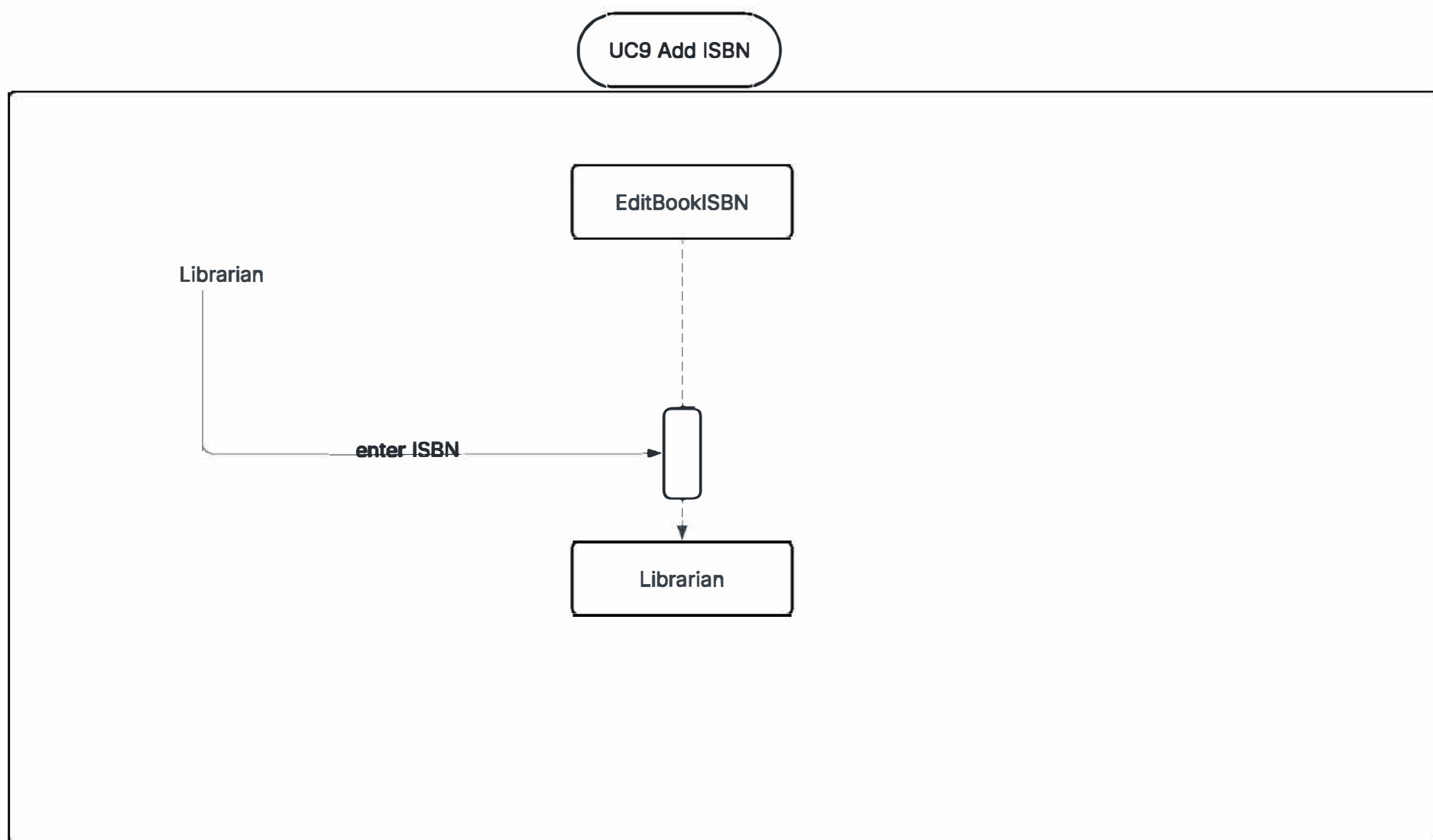
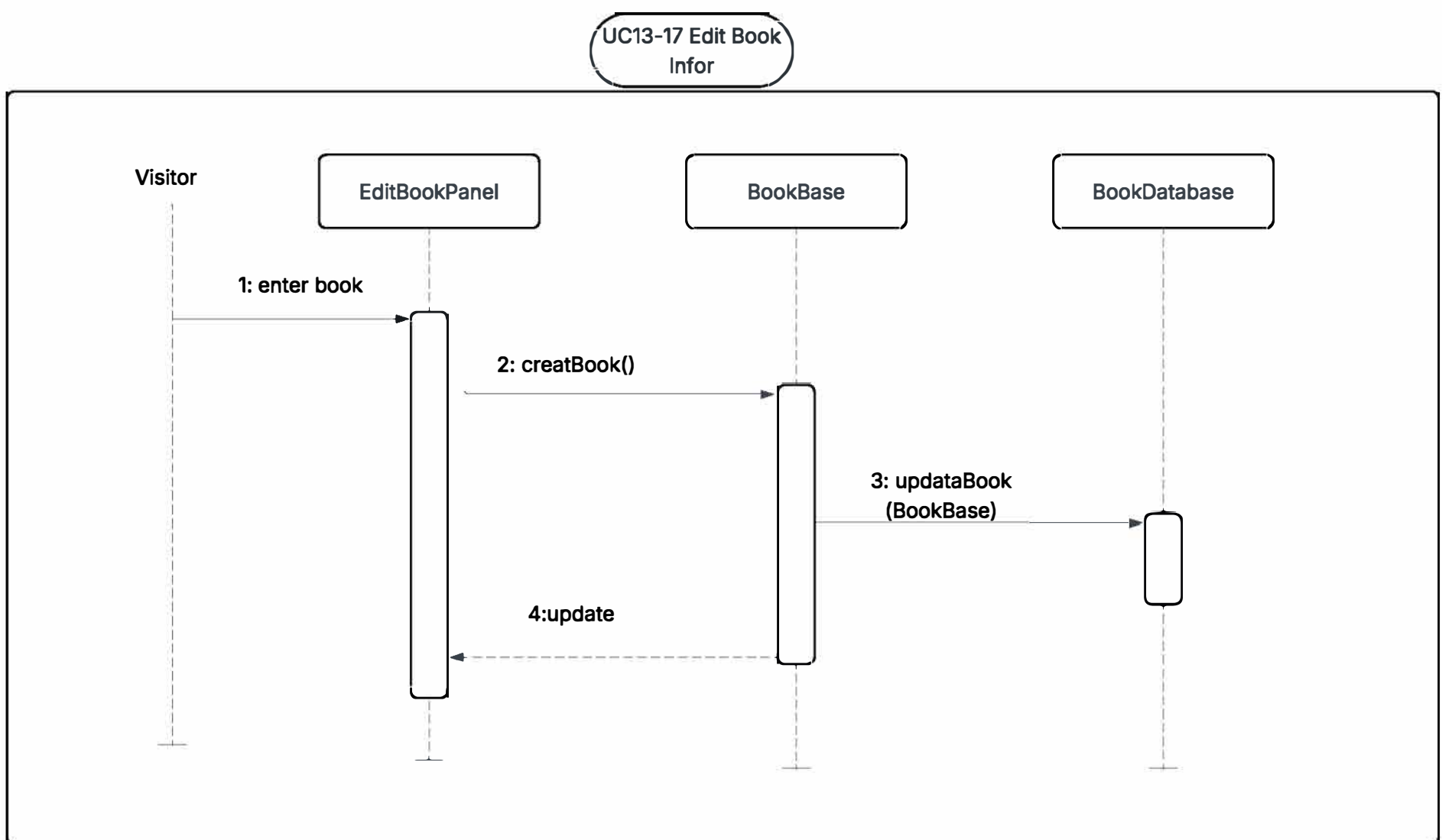
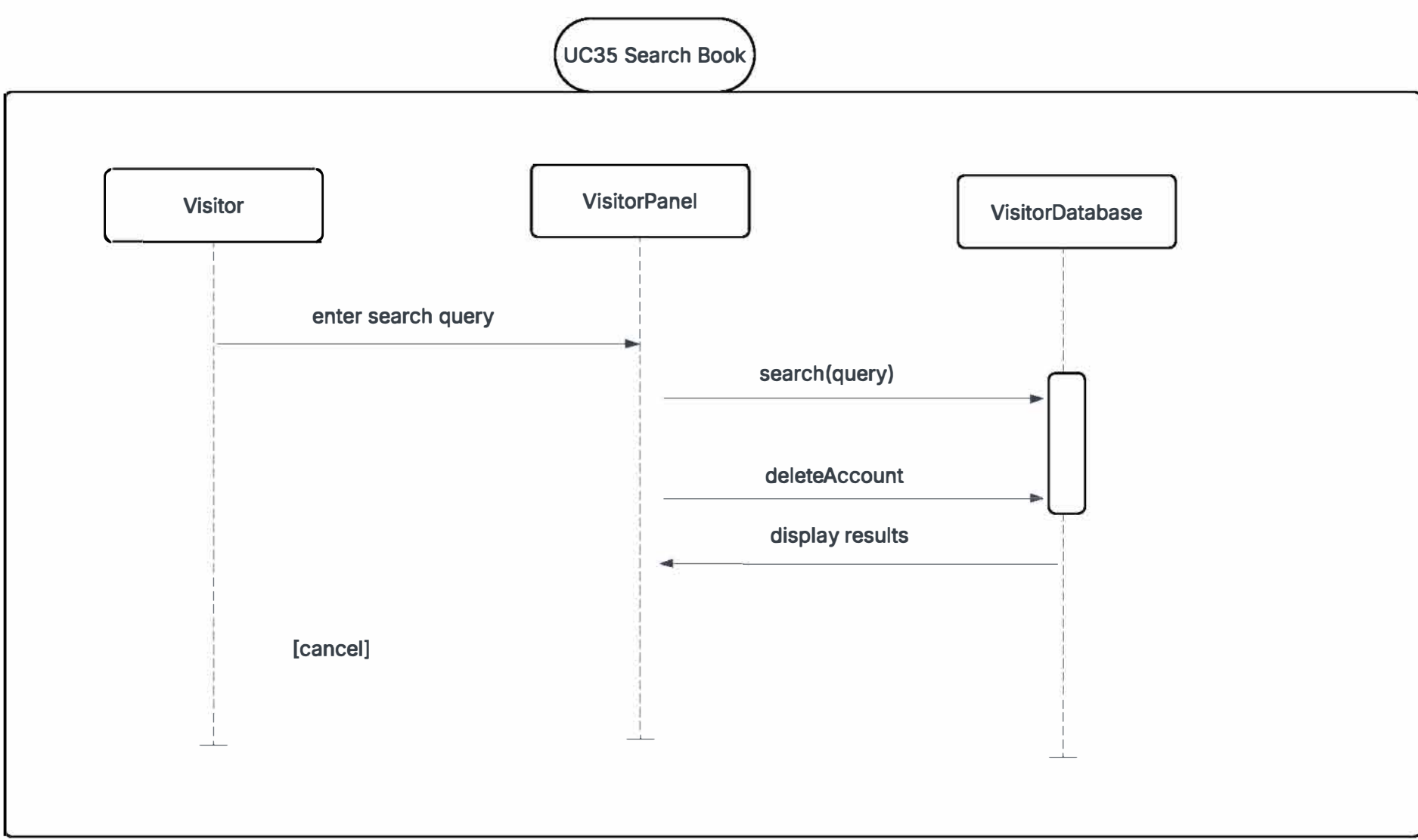
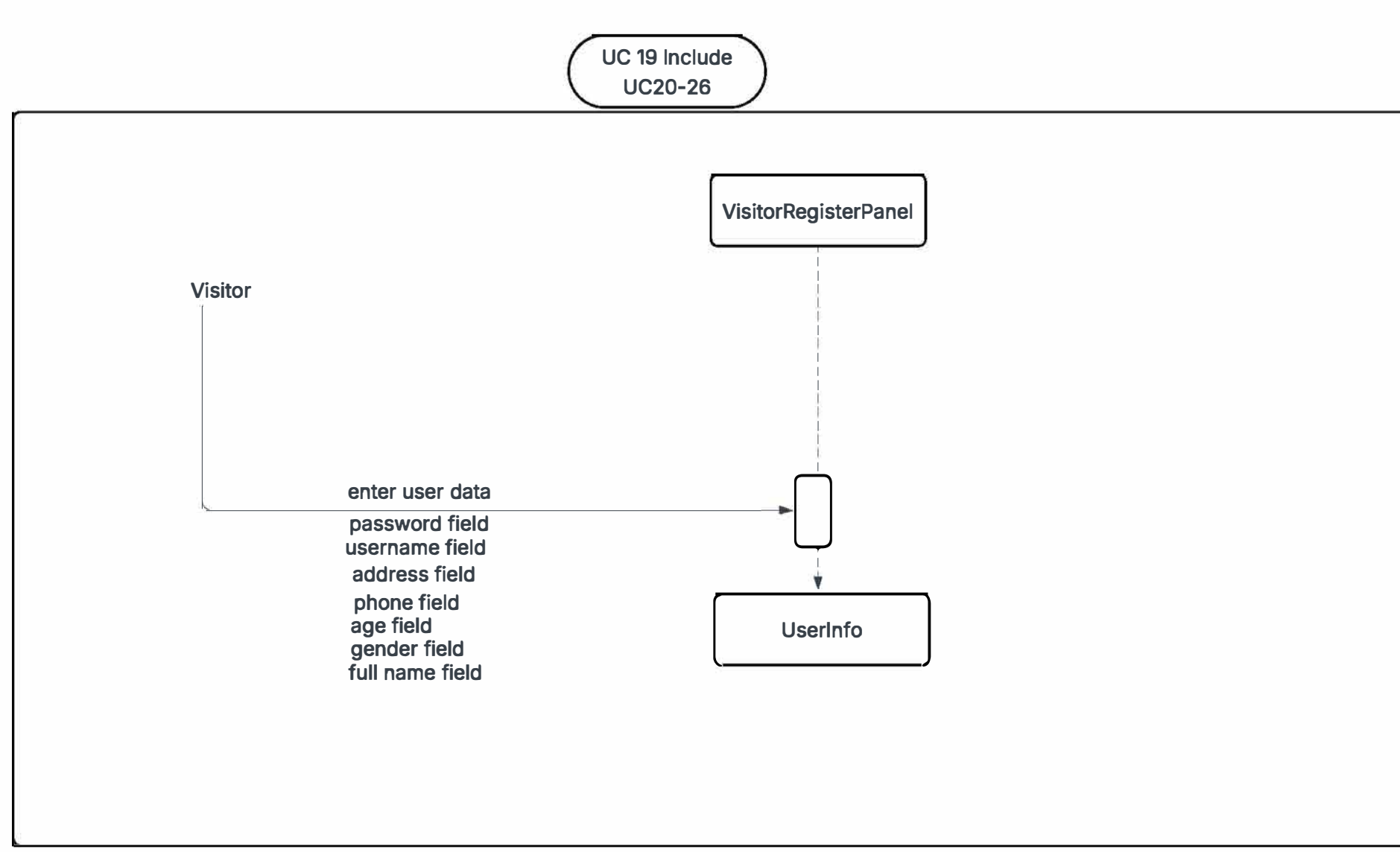
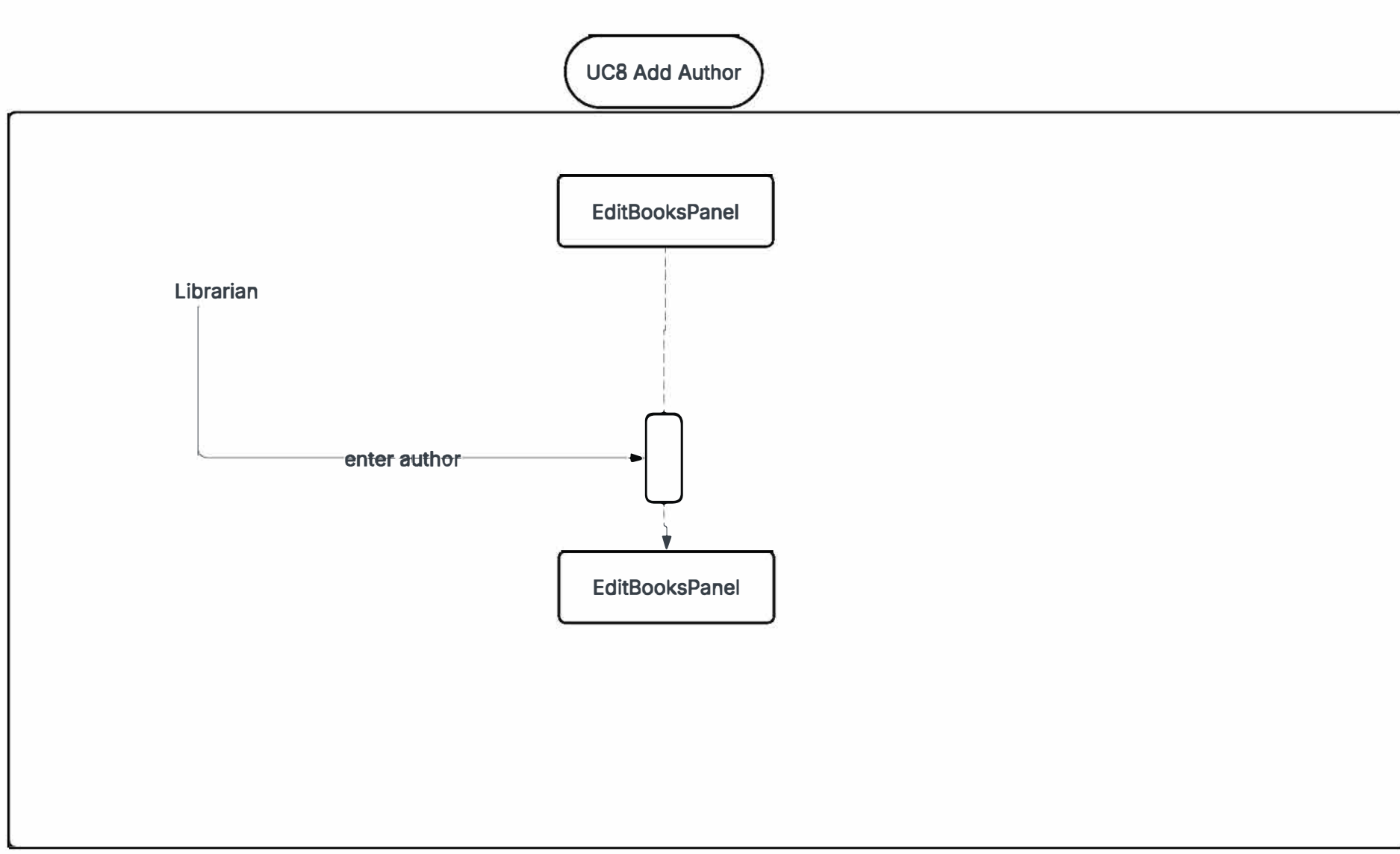
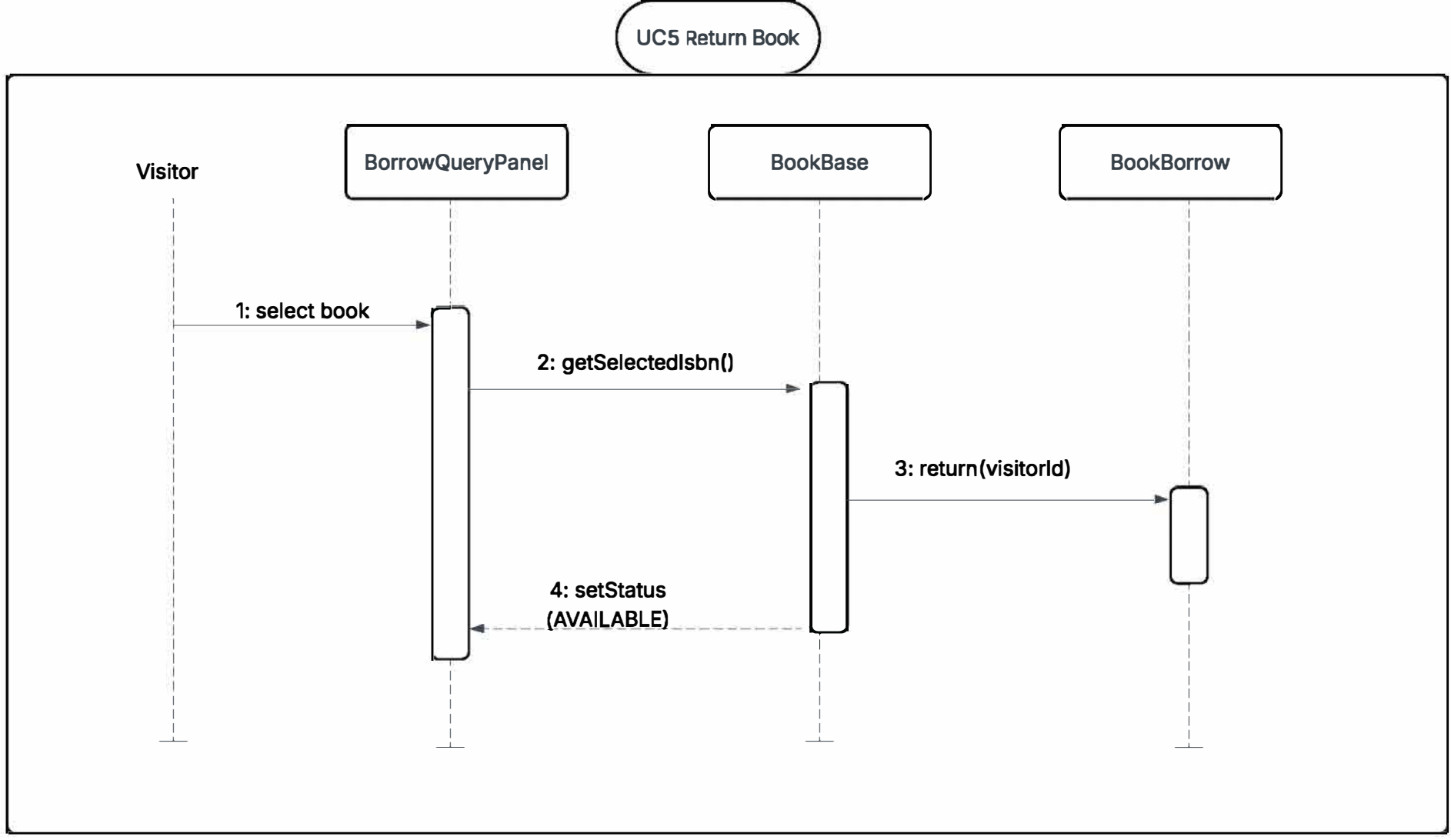
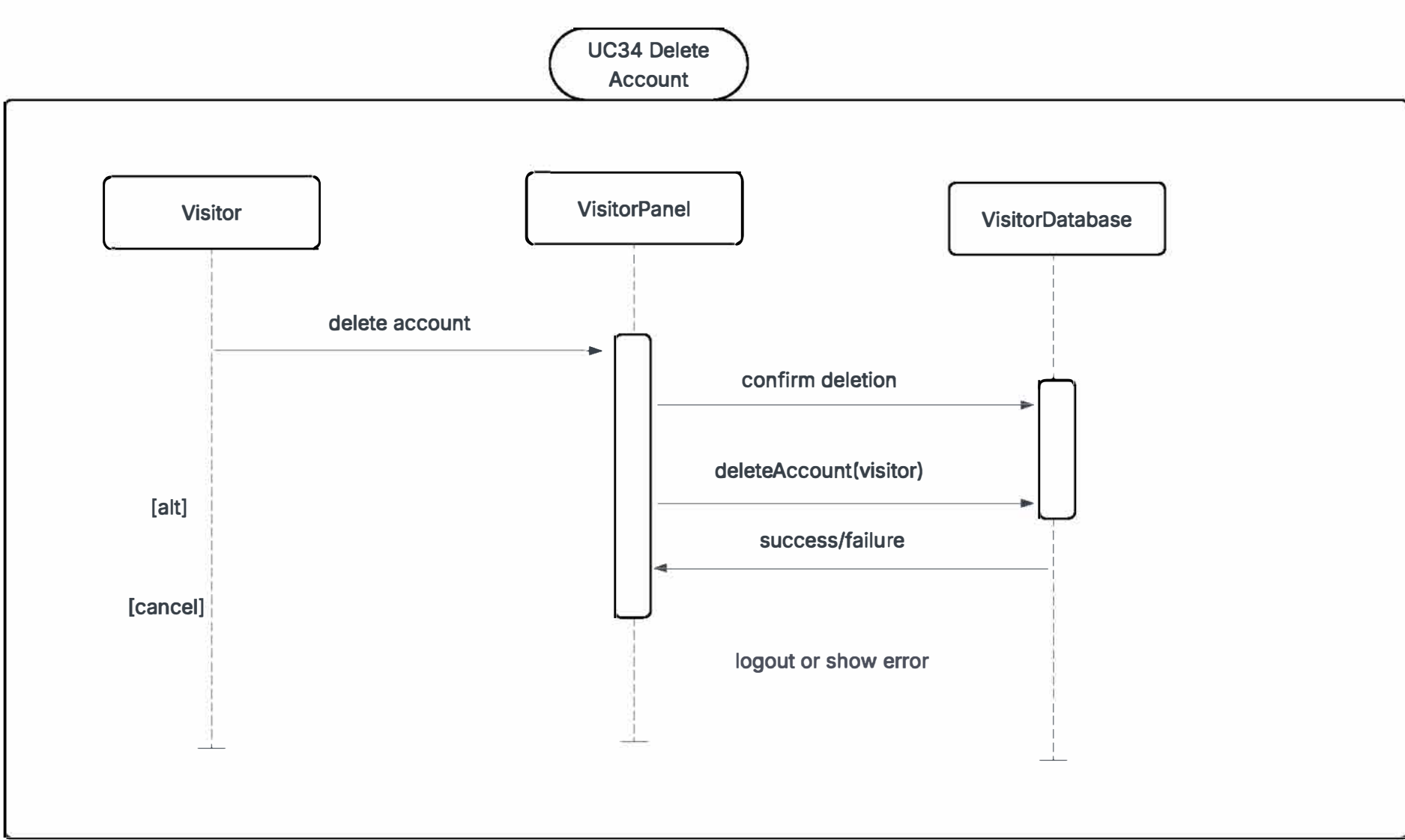
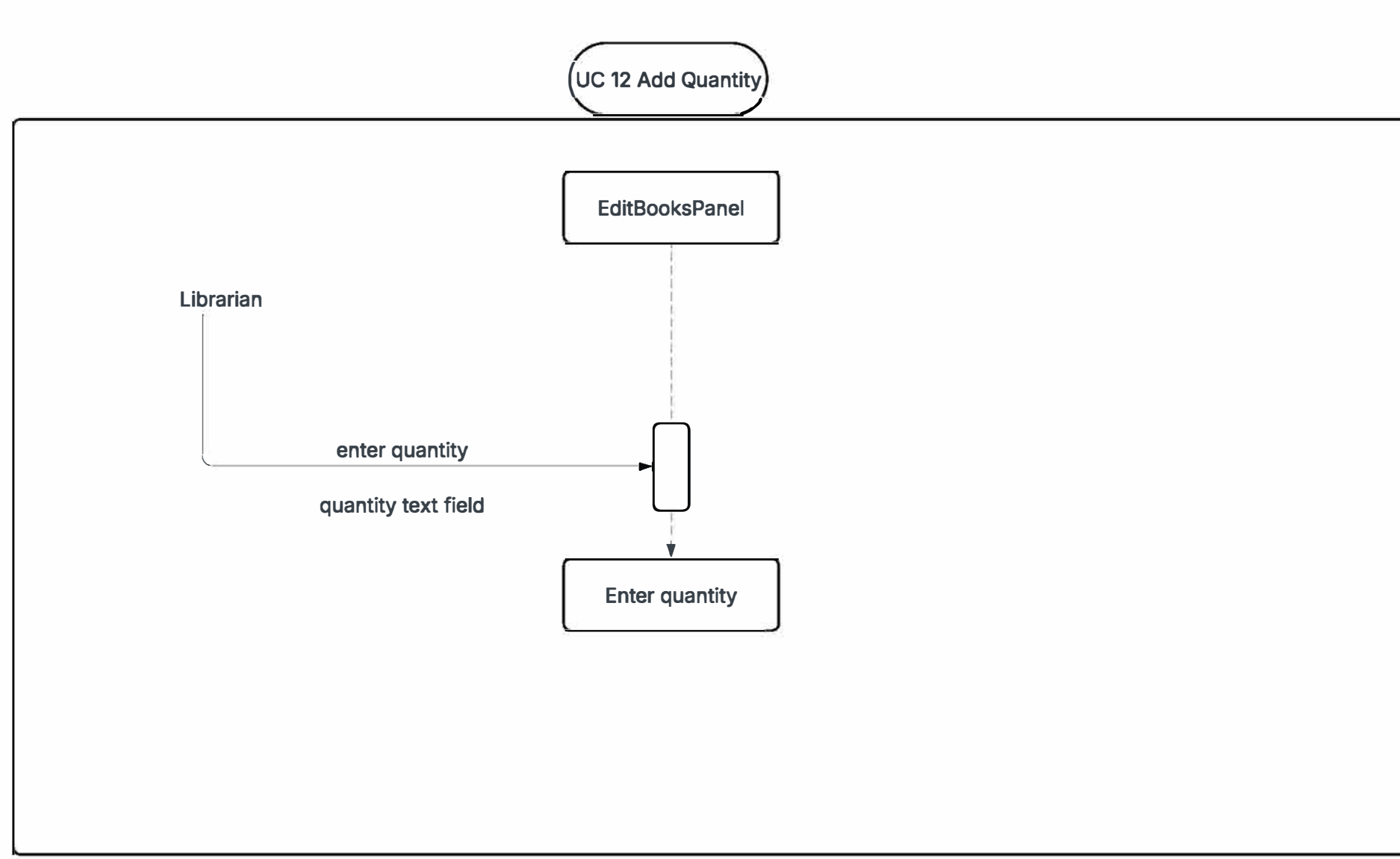
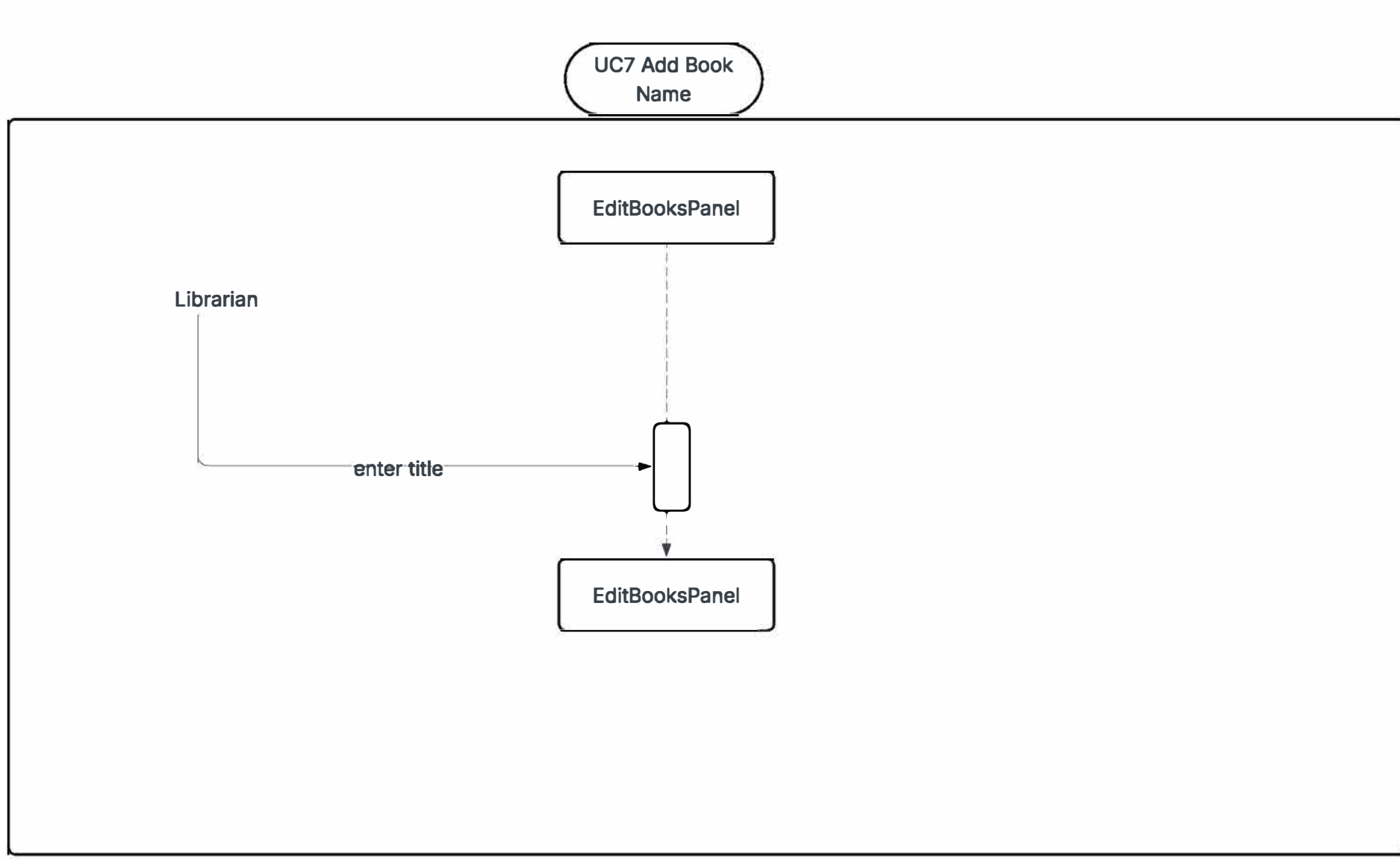
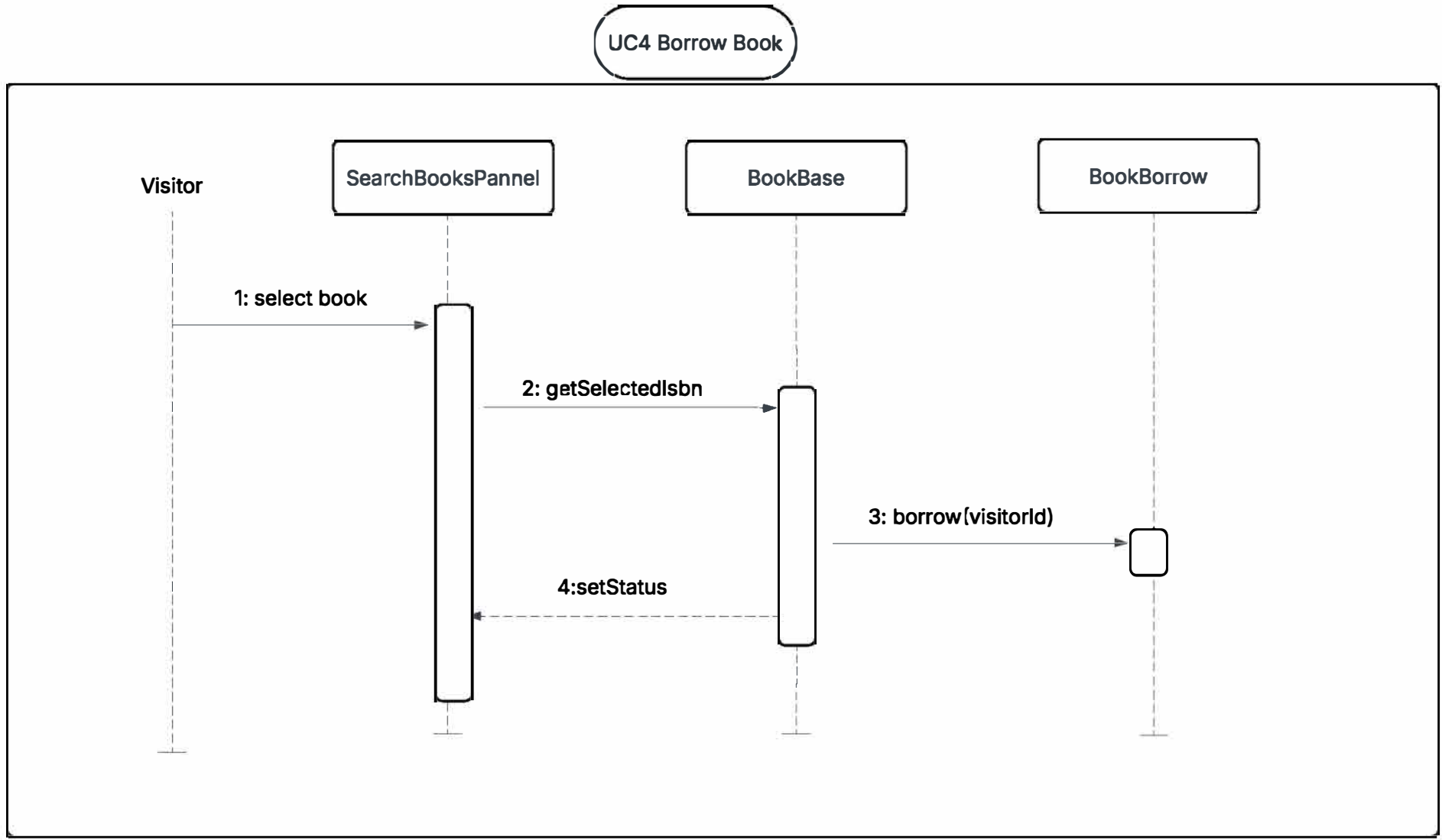
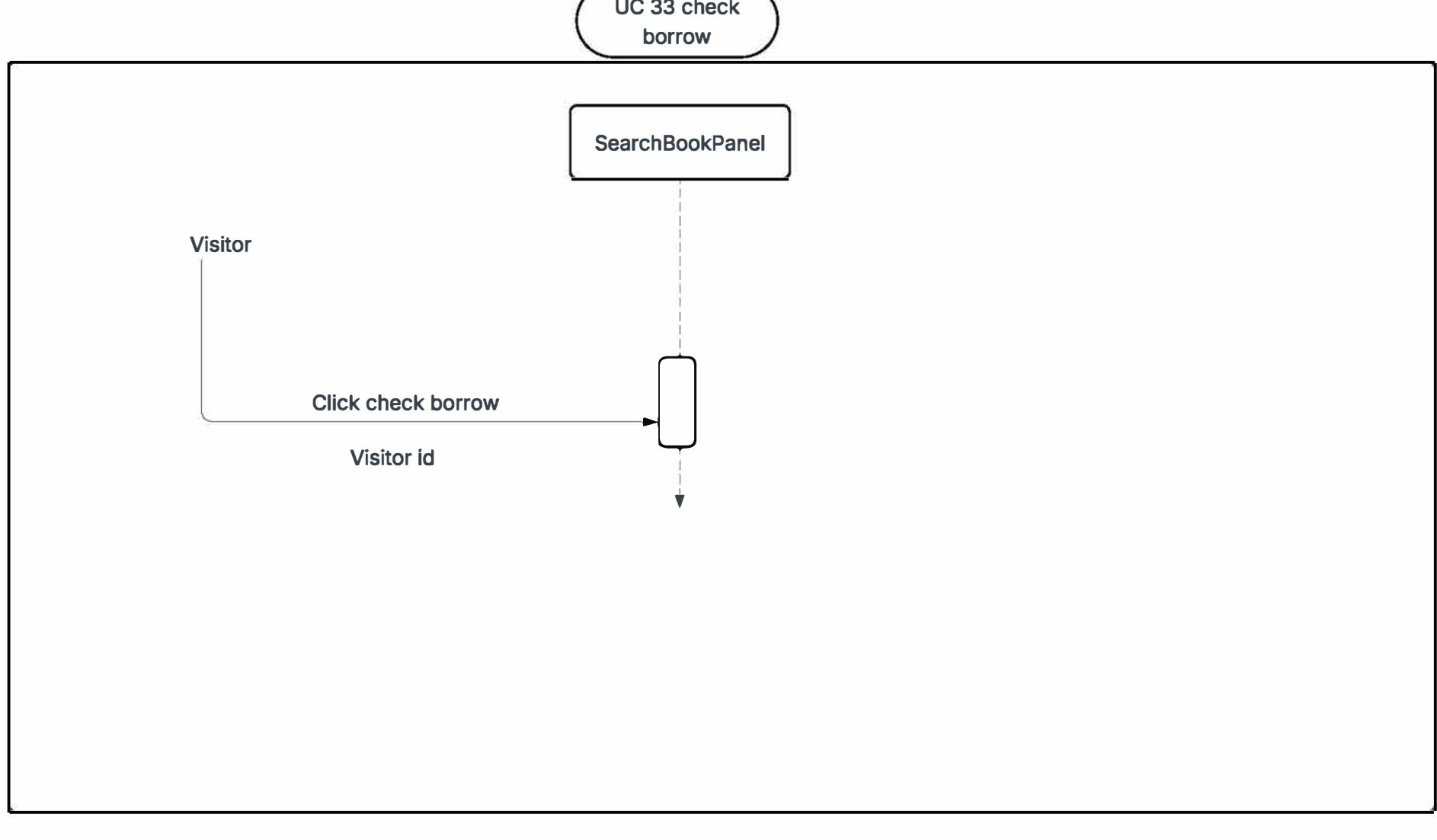
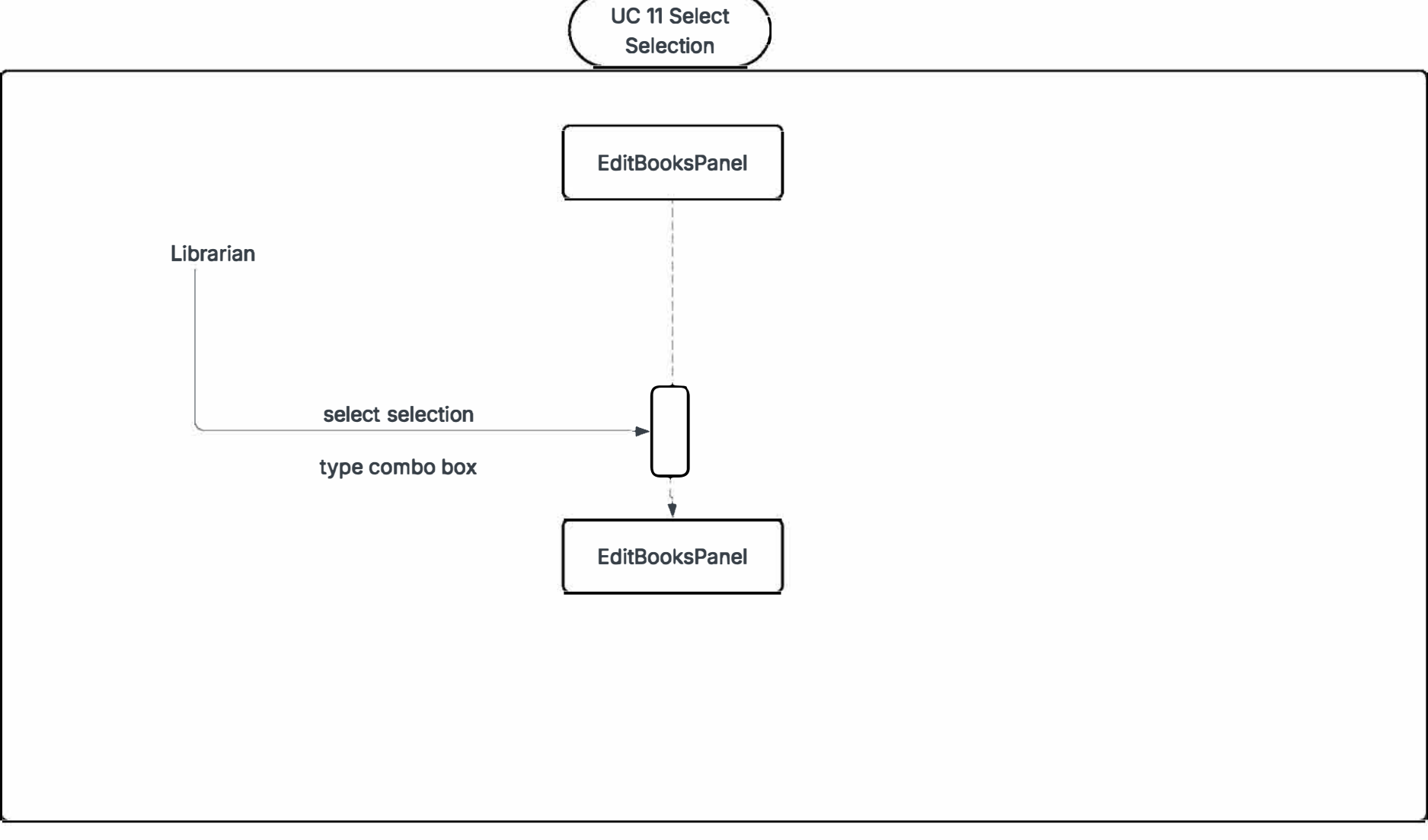
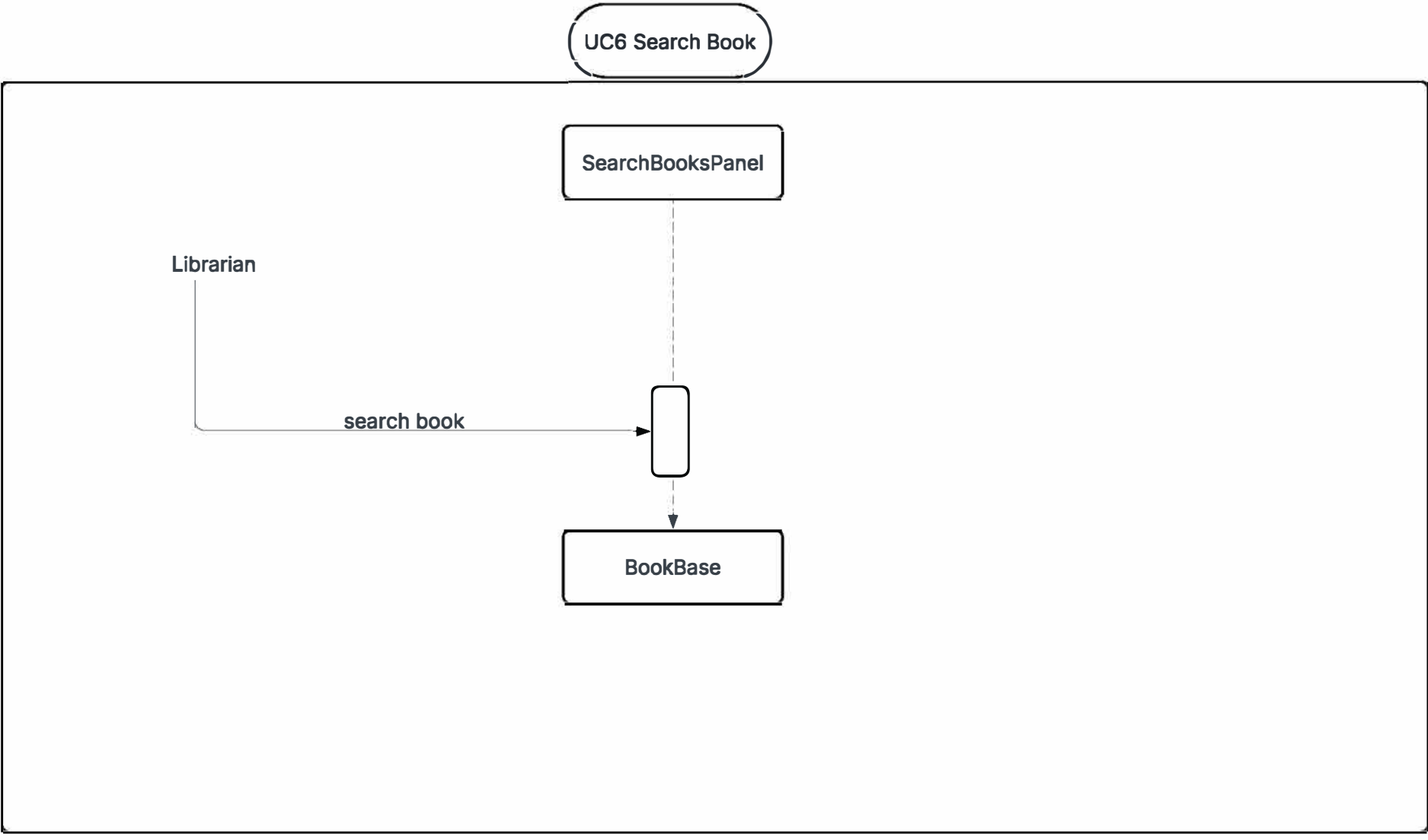
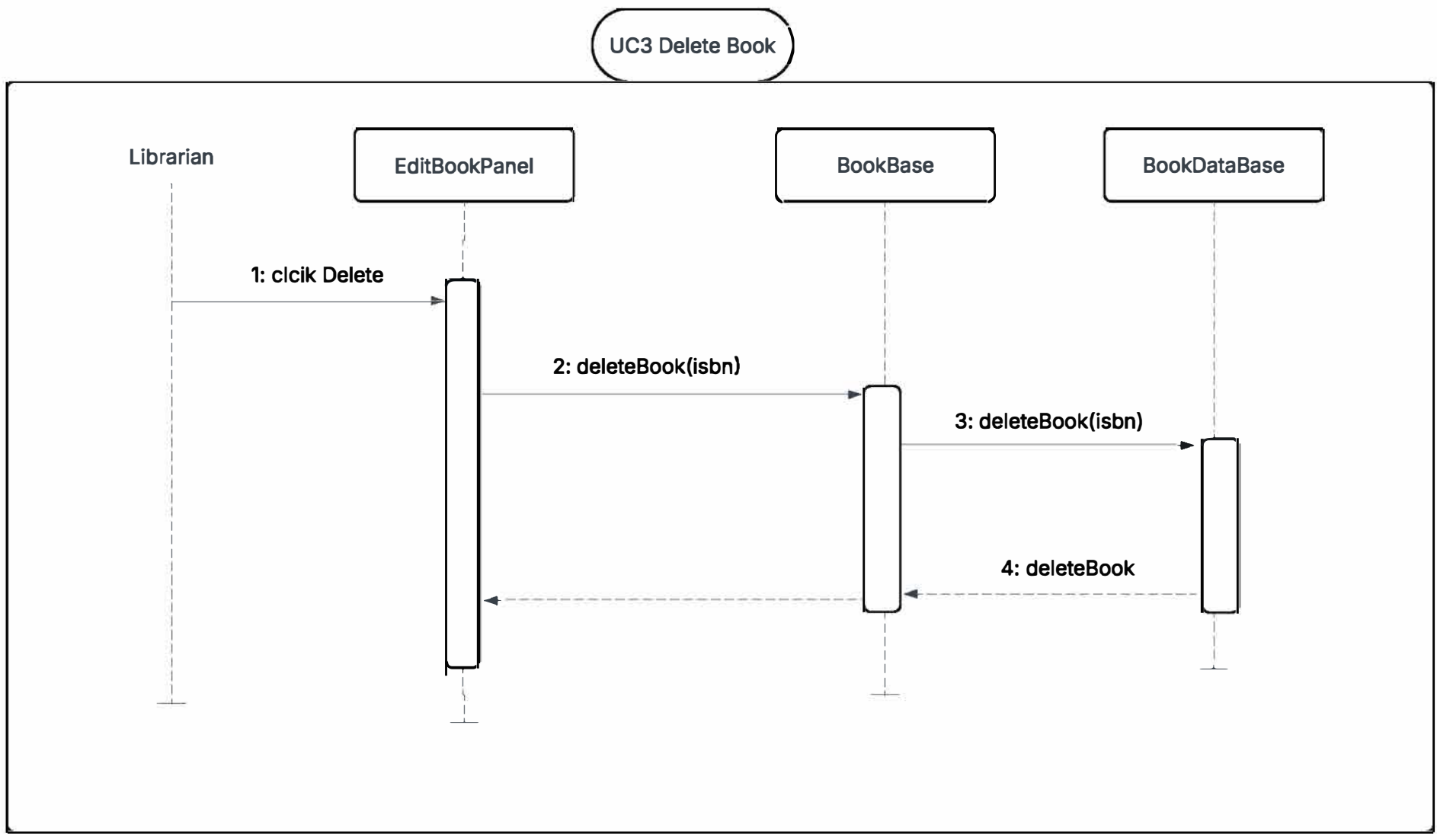
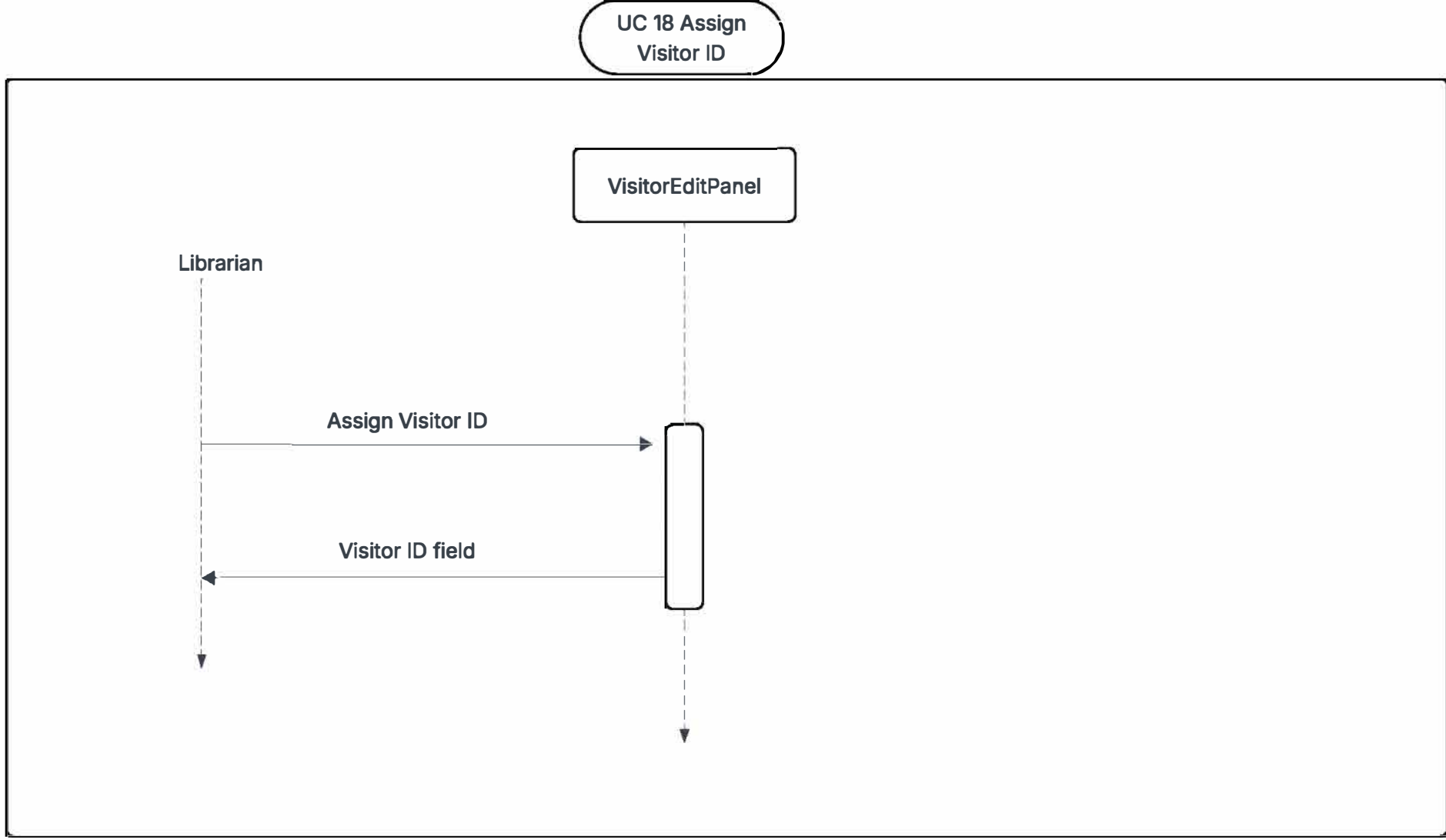
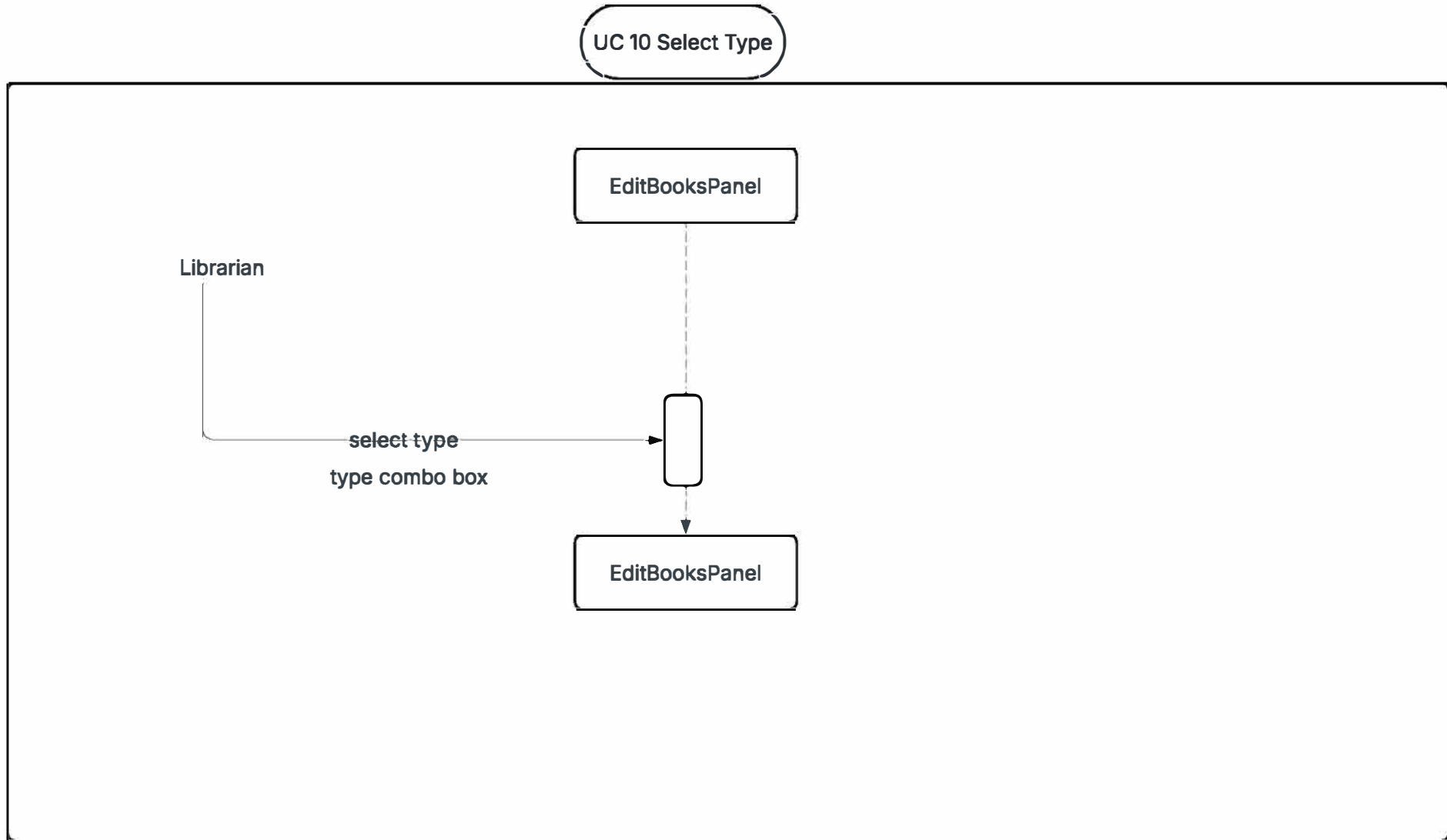
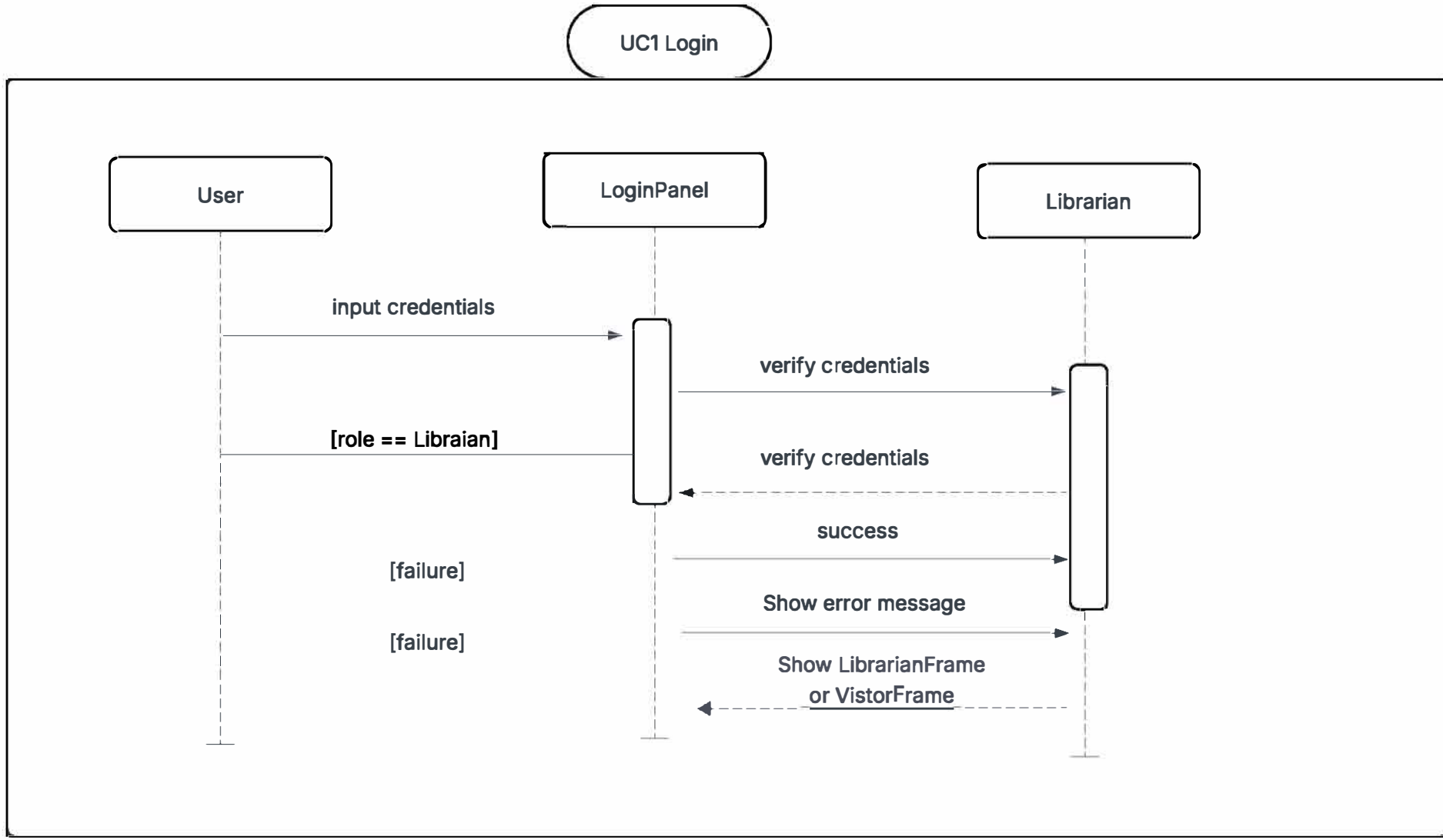
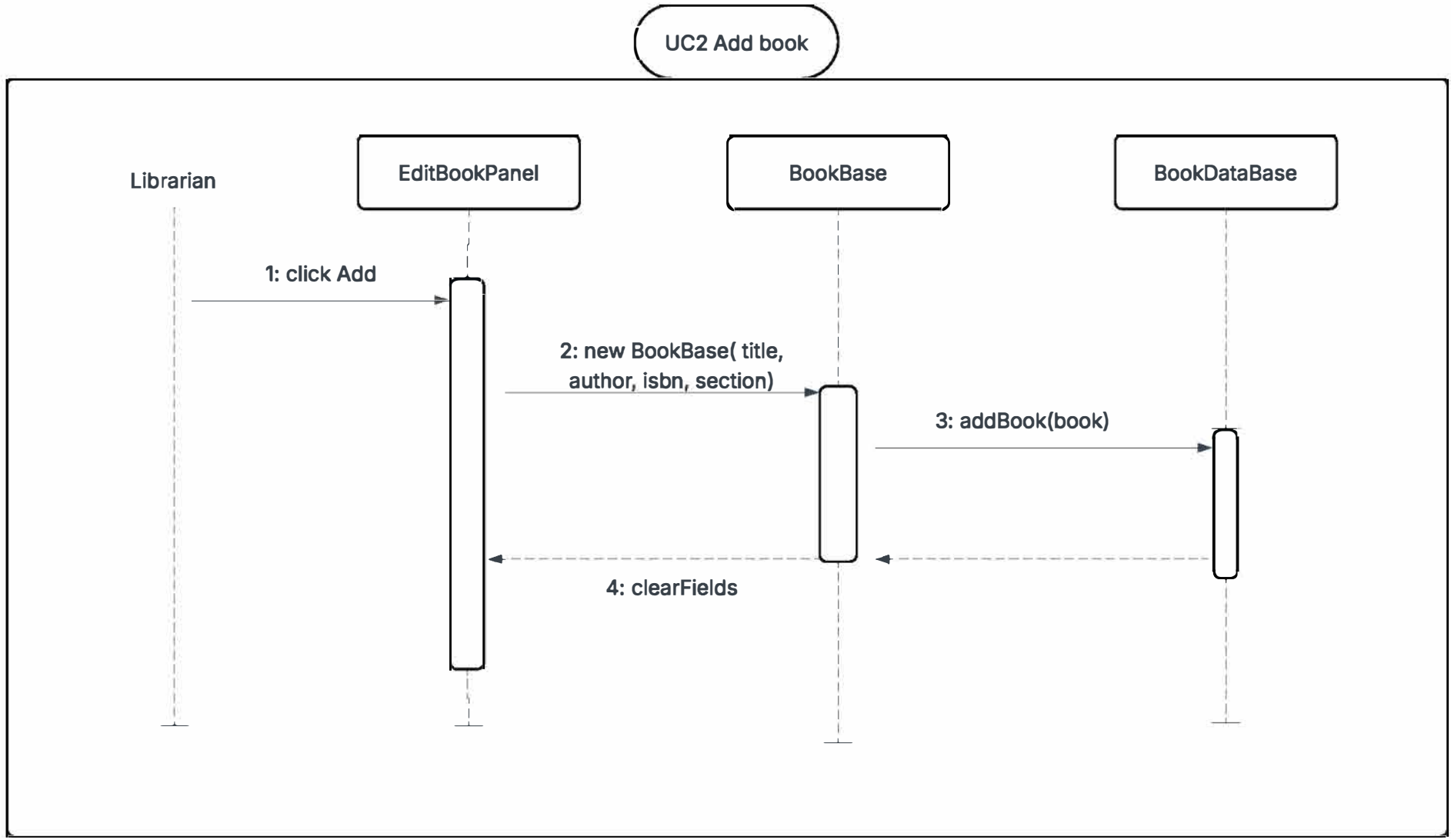
Actor: Visitor

UC36: Logout

Overview: Librarian and Visitor can log out from the system when the finished

Related Use Cases: NA

Actor Librarian, Visitor



Conclusion:

I believe that my implementation of this project is good, but there can be improvement also. For example, in real life I need to create a LibrarianDataBase instead of hard code the username and password, but since the main goal of this project is to manage book status and visitor status, I focused more on the implementation of book and visitor.