UML Use Case Diagram: Book Reference Pg. 138, 161

You are a software engineer tasked with designing a Library Management System. The system will be used by three types of actors: Librarians, Library Members, and the External Book Database System (an external system that provides book information).

After meeting with stakeholders, the following requirements have been gathered:

1. Librarians must be able to:
    - Add new books to the library catalog.
    - Remove books from the catalog.
    - Register new library members.
    - Process book checkouts for members.
    - Process book returns from members.
    - Generate overdue book reports.
2. Library Members must be able to:
    - Search for books by title, author, or ISBN.
    - View their own borrowing history and current checked-out books.
    - Reserve a book that is currently checked out by another member.
    - Renew a checked-out book (if no one has reserved it).
3. The system automatically communicates with an External Book Database System to:
    - Fetch book details (e.g., summary, cover image, reviews) when a librarian adds a new book, using the ISBN number.
4. The "Search for books" and "Reserve a book" use cases are available to both Librarians and Members.
5. When a book is returned, the system automatically checks if the book has any pending reservations. If a reservation exists, the system notifies the reserving member that the book is now available.

Task:

Based on the above requirements, draw a UML Use Case Diagram

Class diagram: Book Reference Pg. 140

You are working as a software engineer for a University Course Registration System. The system must manage students, courses, professors, and the enrollment process. After analyzing the requirements, the following information has been gathered:

System Requirements:

1. Students have a student ID, name, email, and phone number. They can enroll in multiple courses. Each student has a program of study (e.g., "Computer Science," "Electrical Engineering").
2. Professors have a professor ID, name, email, department, and office location. Each professor can teach multiple courses.
3. Courses have a course code (e.g., "CSE101"), course title, credit hours, and a description. Each course is taught by exactly one professor.
4. Enrollment tracks which students are enrolled in which courses. For each enrollment, the system must record:
    - The date the student enrolled.
    - The grade achieved (if the course is completed; can be null if still in progress).
5. A student cannot enroll in the same course more than once.
6. Each course may have multiple prerequisites (other courses that must be completed before enrollment). A course can have zero or more prerequisites.
7. Departments have a department name, department code, and office location. Each department offers multiple courses and employs multiple professors. Each professor belongs to exactly one department. Each student belongs to exactly one department.

Task:

Based on the above requirements, draw a UML Class Diagram

UML state diagram notation: Book Reference Pg 141

You are designing software for a Library Book Management System. The system needs to track the lifecycle of each physical book copy in the library's collection. After analyzing the requirements, the following information has been gathered about how a book behaves over time:

Book Lifecycle Requirements:

1. When a new book copy is acquired by the library, it is in Cataloged state. In this state, the book has been entered into the system but is not yet available for checkout.
2. From Cataloged, a librarian can place the book on shelf, after which the book enters the Available state. In this state, the book is ready for members to check out.
3. When a member checks out an Available book, it transitions to the Checked Out state. The system records the due date.
4. While Checked Out:
    - If the member returns the book before or on the due date, it returns to Available.
    - If the member returns the book after the due date, it transitions to Overdue Returned, which triggers a fine calculation.
5. From Overdue Returned, after the fine is paid, the book becomes Available again.

6. If a Checked Out book is not returned by the due date, it automatically transitions to the Overdue state at midnight of the due date.
7. While in Overdue state:
    - If the member returns the book, it goes to Overdue Returned (fine calculation pending).
    - The librarian can also mark the book as Lost if the member reports it lost or if significant time has passed.
8. If a book is in Lost state, it can only be moved back to Cataloged if a replacement copy is acquired (rare, but possible). More commonly, Lost books are eventually Decommissioned.
9. From Available state, a librarian may also remove a book from circulation (e.g., due to damage or obsolescence), sending it to Decommissioned state. Decommissioned is a final state—books in this state are removed from the system.
10. Books that are Lost and not replaced eventually go to Decommissioned after administrative processing.

Special Conditions:

- A book can be reserved by a member while it is Checked Out. When the book is returned, if there is a pending reservation, it should go to On Hold state (waiting for the reserving member to pick it up) instead of directly to Available.
- From On Hold, if the member picks it up within 3 days, it goes to Checked Out (for that member). If not picked up within 3 days, it reverts to Available.

Task:

Based on the above requirements, draw a UML State Diagram

Activity diagram: Book Reference Pg. 162

You are designing an Online Bookstore Order Processing System. The system must handle the entire process from when a customer places an order until the order is delivered and payment is finalized. After analyzing the requirements, the following workflow has been documented:

Order Processing Workflow:

1. The process begins when a customer places an order through the website. The order contains one or more books.
2. The system first validates the order by checking:
    - That all ordered books exist in the inventory.
    - That the customer's shipping address is complete and valid.
    - That the payment information (credit card) passes basic format validation.

3. If the order validation fails, the system notifies the customer of the issue and ends the process for that order.
4. If the order validation passes, the system checks inventory availability for each book:
    - For books that are in stock, they are reserved for the customer.
    - For books that are out of stock, the system checks with suppliers for estimated availability.
5. Parallel processing begins at this point:
    - Path A (Inventory Reservation): For in-stock books, the system updates inventory counts and prepares the picking list for warehouse staff.
    - Path B (Supplier Checking): For out-of-stock books, the system contacts suppliers. If a supplier can fulfill within acceptable time, a backorder is created. If not, the customer is offered a cancellation option.
6. Once both paths complete, the system consolidates the order:
    - If all items are available (in-stock or backordered), the order proceeds.
    - If some items cannot be fulfilled at all, those items are cancelled and the customer is notified.
7. The system then processes payment by charging the customer's credit card for available items (not backordered items yet).
8. If payment succeeds:
    - For in-stock items, the warehouse is notified to pick and pack the books.
    - The books are shipped to the customer.
    - The system updates order status to "Shipped" and sends tracking information to the customer.
9. If payment fails, the system notifies the customer and cancels the order (releasing any inventory reservations).
10. After shipment, when the customer receives the books, they may:
    - Accept the order (process ends).
    - Initiate a return for any reason within 30 days.
11. If a return is initiated:
    - The customer sends the book back.
    - Warehouse receives and inspects the returned book.
    - If approved, a refund is processed to the customer's credit card.
    - The system updates inventory to reflect the returned book.
12. Throughout the process, the system sends email notifications to the customer at key stages:
    - Order confirmation
    - Payment confirmation
    - Shipping confirmation with tracking
    - Return confirmation and refund notification

Task:

Based on the above requirements, draw a UML Activity Diagram