**E-JUST University - Computer Science and Information Technology Programs** –

**General Division**

**Module: Software Engineering (CSC211)**

# SWE - C O U R S E P R O J E C T ( P H A S E S 1 & 2 C O V E R S H E E T )

**Discussions Scheduled for Week 14 *(Monday | Tuesday)*** o Print 1 copies of this cover sheet and attach both to a printed copy of the documentation *(SRS, … etc.)*. You must submit a CD including softcopies of all your documents and Project implementation.

* Please write all your names in English. Please make sure that your students’ IDs are correct.
* Handwritten Signatures for the attendance of all team members should be filled in the cover sheet copy before the discussion.
* Please attend the discussion on time *(announced separately)*, late teams will lose 3 grades.

**Project Name: \_\_\_Library Management System\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Project Leader Name: \_Abdelrahman Ashraf\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Team Information *(typed not handwritten, except for the attendance signature)*:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **ID**  **[Ordered by ID]** | **Full Name [In English]** | **Attendance**  **[Handwritten Signature]** | **Final Grade** |
| **1** |  |  |  |  |
| **2** |  |  |  |  |
| **3** |  |  |  |  |
| **4** |  |  |  |  |
| **5** |  |  |  |  |
| **6** |  |  |  |  |

**Grading Criteria:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Items** | | **Actual**  **Grade** | **Notes** |
| **Functional Requirements** | **1** |  |  |
| **Non-Functional Requirements** | **1** |  |  |
| **Use-Case Diagram(s)** *including general use-cases for the system, and the detailed use-cases description* | **1** |  |  |

Page **1** of **2**

**E-JUST University - Computer Science and Information Technology Programs** –

**General Division**

**Module: Software Engineering (CSC211)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Activity Diagram(s)** | **1** |  |  |
| **Database Specification** *(ERD, Tables)* | **1** |  |  |
| **System Architecture –** *including applied Architectural Pattern(s)* | **1** |  |  |
| **Sequence Diagram(s)** | **3** |  |  |
| **System Sequence Diagrams (SSDs)** | **2** |  |  |
| **Collaboration/Communication Diagram(s)** | **3** |  |  |
| **Class Diagram *(2 versions)***   1. **An initial version based on the requirements and Use Case/Activity diagrams.** 2. **An intermediate version based on the interaction diagrams.** | **4** |  |  |
| **Object Diagrams** *(Including object diagrams that illustrate the preconditions and the post-conditions of selected functions)* | **2** |  |  |
| **Package Diagram(s)** | **2** |  |  |
| **Self-Study Component 1: State-Machine Diagrams** *(for selected state-dependent objects)* | **0.**  **5** |  |  |
| **Self-Study Component 2: Deployment diagram(s)** | **0.**  **5** |  |  |
| **Front End Design for all Functions** Desk-Top **or** *(HTML,*  *Bootstrap).* | **2** |  |  |
| **Implementation based on the submitted Requirements &** | **6** |  | **1** *per*  *Module* |
| **Design. Should include at least 4 of the following modules (in addition of course to modules specific to your individual**  **projects):**  **1)**  **User Role Management Module.**  **2)**  **User manipulation Module** *(Login, Add / Delete / Update / Search, List).*  **3)**  **Controlling Resources Module** *(Rooms, Orders, Products,*  *... etc.).*  **4)**  **Reservation and Rescheduling Module.**  **5)**  **Generating Reports Module** *(PDFs, … etc.).*  **6)**  **Sending Emails or Notifications Module.** |
| **Presentation Skills** | **2** |  |  |

**33**

# Teaching-Assistant’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Page **2** of **2**

***Functional Requirements:***

1. **User Authentication and Authorization:**
   * **The system must implement secure user authentication and authorization mechanisms.**
   * **Only authorized personnel should have access to the system and its functionalities.**
2. **Book Management:**
   * **Librarians should be able to add, edit, and delete book records.**
   * **Book details such as title, author, publication date, ISBN, and availability status must be editable.**
   * **Support for categorization of books into genres or topics for efficient cataloging.**
3. **Borrower Management:**
   * **The system must allow the creation and management of borrower profiles with personal information and contact details.**
   * **Tracking borrower borrowing history and penalties for overdue books should be enabled.**
4. **Transaction Handling:**
   * **Facilitate book issuance, return, and reissuance transactions.**
   * **Record transaction details such as transaction date and borrower information.**
   * **Calculate and manage fines for overdue books.**
5. **Search and Retrieval:**
   * **Provide efficient search and retrieval functions based on criteria such as title, author, ISBN, or category.**
6. **User Roles and Access Control:**
   * **Support different user roles (e.g., librarian, assistant) with varying access levels.**
   * **Implement role-based access control to ensure data security.**
7. **Data Integrity and Backup:**
   * **Maintain data integrity to ensure accuracy and consistency.**
   * **Include backup and restore mechanisms to safeguard against data loss.**
8. **User-Friendly Interface:**
   * **Features an intuitive and user-friendly interface for easy navigation and use by librarians and borrowers.**
9. **Scalability and Performance:**
   * **Design the system to handle a growing number of books, borrowers, and transactions without a significant decrease in performance.**

***Non-Functional Requirements*:**

1. **Performance:**
   * **The system must perform efficiently under normal operational conditions.**
2. **Reliability:**
   * **Ensure the system's reliability, minimizing the occurrence of errors and downtimes.**
3. **Scalability:**
   * **Design the system to scale seamlessly to accommodate a growing volume of books, borrowers, and transactions.**
4. **Security:**
   * **Implement robust security measures to safeguard user data and prevent unauthorized access.**
5. **Usability:**
   * **The system should have a user-friendly interface to enhance ease of use for both librarians and borrowers.**
6. **Compatibility:**
   * **Ensure compatibility with various devices and web browsers.**
7. **Maintainability:**
   * **The system should be easy to maintain, allowing for updates and improvements.**
8. **Data Backup and Recovery:**
   * **Regularly backup data and provide mechanisms for efficient data recovery in case of loss.**
9. **Load Testing:**
   * **Conduct load testing to assess the system's performance under different levels of usage.**
10. **Response Time:**
    * **Ensure fast response times for user interactions with the system.**
11. **Error Handling:**
    * **Implement effective error handling mechanisms to provide meaningful feedback in case of errors.**

***System Architecture (Modules)*:**

1. **User Authentication and Authorization Module:**
   * **Implements secure user authentication and authorization mechanisms.**
   * **Manages different user roles with varying access levels.**
2. **Book Management Module:**
   * **Allows librarians to add, edit, and delete book records.**
   * **Supports the customization of book details such as title, author, publication date, ISBN, and availability status.**
   * **Includes support for categorization of books into genres or topics.**
3. **Borrower Management Module:**
   * **Enables the creation and management of borrower profiles with personal information and contact details.**
   * **Tracks borrower borrowing history and implements penalties for overdue books.**
4. **Transaction Handling Module:**
   * **Facilitates book issuance, return, and reissuance transactions.**
   * **Records transaction details, including the transaction date and borrower information.**
   * **Calculates and manages fines for overdue books.**
5. **Search and Retrieval Module:**
   * **Provides efficient search and retrieval functions based on criteria such as title, author, ISBN, or category.**
6. **User Roles and Access Control Module:**
   * **Supports different user roles (e.g., librarian, assistant) with varying access levels.**
   * **Implements role-based access control to ensure data security.**
7. **Data Integrity and Backup Module:**
   * **Maintains data integrity to ensure accuracy and consistency.**
   * **Includes backup and restore mechanisms to safeguard against data loss.**
8. **User-Friendly Interface Module:**
   * **Features an intuitive and user-friendly interface for easy navigation and use by librarians and borrowers.**
9. **Scalability and Performance Module:**
   * **Designs the system to handle a growing number of books, borrowers, and transactions without a significant decrease in performance.**

***DFD Diagrams:***

**A diagram of a library management system

Description automatically generated**

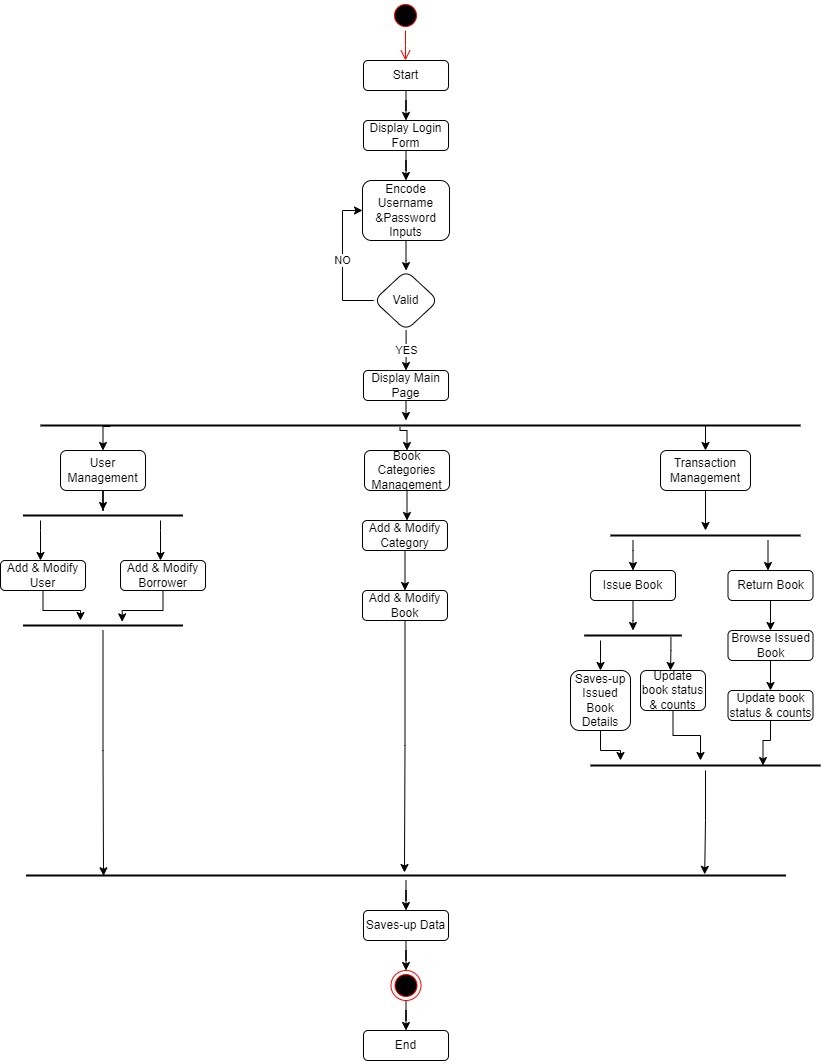
**Diagram of a library management system

Description automatically generated**

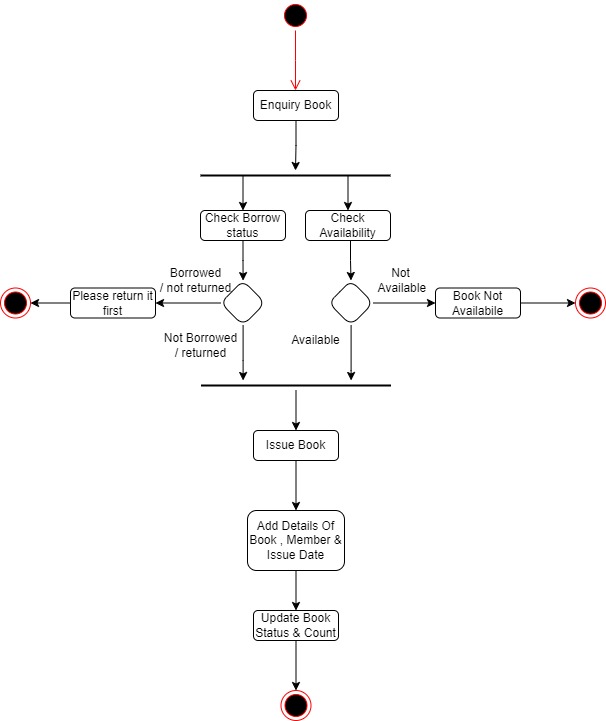
**A diagram of a library

Description automatically generated**

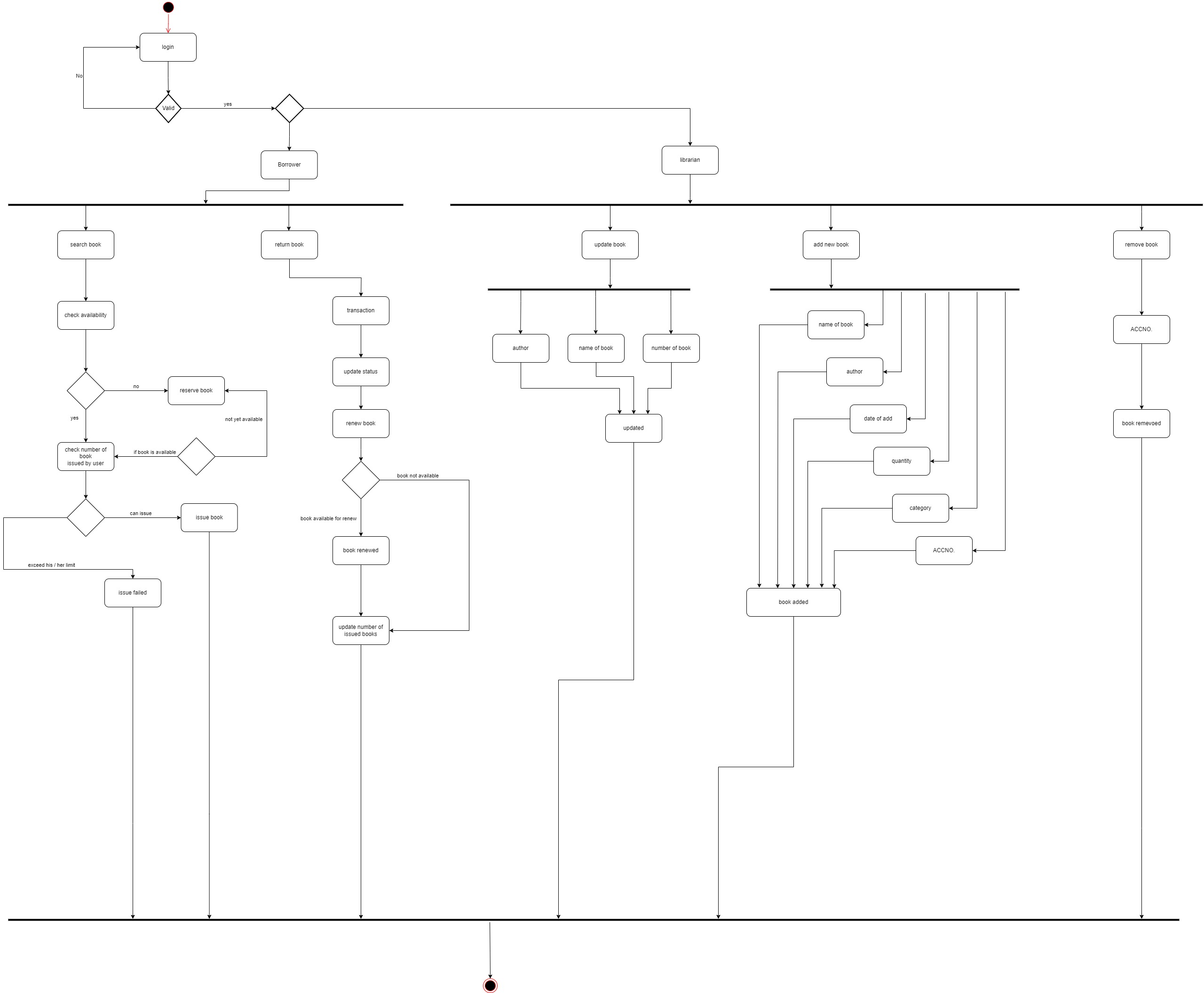
***Activity Diagrams:***

****

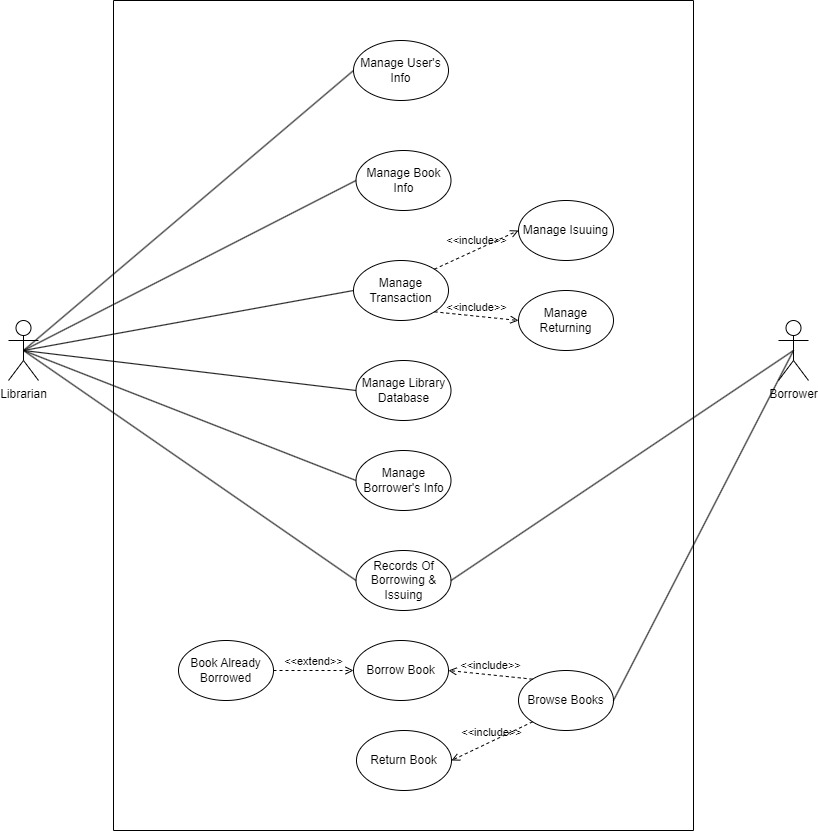
***Issuing book:***



***System:***

******

***Use Case Diagram*:**

****

**Database Specification:**

**Tables:**

**1)BookNumber:**

|  |  |  |
| --- | --- | --- |
| **Attributes** | **Data Type** | **Description** |
| **Id** | Int(11) | Unique identifier of the book |
| **BOOKTITLE** | Varchar(255) | Title of the book |
| **QTY** | Int(11) | Quantity available in the library |
| **Desc** | Varchar(90) | Description of the book |
| **Author** | Varchar(90) | Author of the book |
| **PublishDate** | date | Date when the book was published |
| **Publisher** | varchar(90) | Publisher of the book |

**2)Books:**

|  |  |  |
| --- | --- | --- |
| **Attributes** | **Data Type** | **Description** |
| **AccessionNo** | varchar(90) | Name of the book |
| **BookTitle** | Varchar(125) | Title of the book |
| **BookDesc** | varchar(255) | Description or summary of the book |
| **Author** | varchar(125) | Author of the book |
| **PublishDate** | date | Date when the book was published |
| **BookPublisher** | varchar(125) | Publisher of the book |
| **CategoryId** | int(11) | Identifier for the category to which the book belongs |
| **BookQuantity** | int(11) | Quantity of available copies of the book |
| **Status** | varchar(30) | Current status of the book (e.g., Available, Not Available) |
| **BookType** | varchar(90) | Type or genre of the book |
| **NumberOfCopies** | int(11) | Total number of copies, including both available and reserved |

**3)Borrow:**

|  |  |  |
| --- | --- | --- |
| **Attributes** | **Data Type** | **Description** |
| **BorrowId** | Int(11) | Primary key, unique identifier for each borrowing transaction |
| **AccessionNo** | varchar(90) | Unique identifier of the book being borrowed |
| **NoCopies** | int(11) | Number of copies being borrowed |
| **DateBorrowed** | datetime | Date and time when the book was borrowed |
| **Purpose** | varchar(90) | Purpose or reason for borrowing the book |
| **Status** | varchar(30) | Current status of the borrowing transaction (e.g., Active, Returned) |
| **DueDate** | datetime | Date when the borrowed book is due to be returned |
| **BorrowerId** | int(11) | Identifier for the borrower associated with the borrowing transaction |
| **Due** | tinyint(1) | Indicator for whether the book is due or not (1 for true, 0 for false) |

**4)Borrower:**

|  |  |  |
| --- | --- | --- |
| **Attributes** | **Data Type** | **Description** |
| **IDNO** | Int(11) | Primary key, unique identifier |
| **BorrowerId** | varchar(90) | Index, unique identifier of borrower |
| **Firstname** | varchar(125) | First name of the borrower |
| **Lastname** | varchar(125) | Last name of the borrower |
| **MiddleName** | varchar(125) | Middle name of the borrower |
| **Address** | varchar(255) | Address of the borrower |
| **Sex** | varchar(11) | Gender of the borrower |
| **ContactNo** | varchar(125) | Contact number of the borrower |
| **CourseYear** | varchar(125) | Course and year of the borrower |
| **BorrowerPhoto** | varchar(255) | Path to the borrower's photo |
| **Stats** | varchar(35) | Borrower status (e.g., Active, Inactive) |
| **IMGBLOB** | blob | Binary large object for image data |

**5)Category:**

|  |  |  |
| --- | --- | --- |
| **Attributes** | **Data Type** | **Description** |
| **CategoryId** | Int(11) | Primary key, unique identifier for category |
| **Category** | varchar(125) | Category name |
| **DDecimal** | varchar(90) | Dewey Decimal Classification for category |

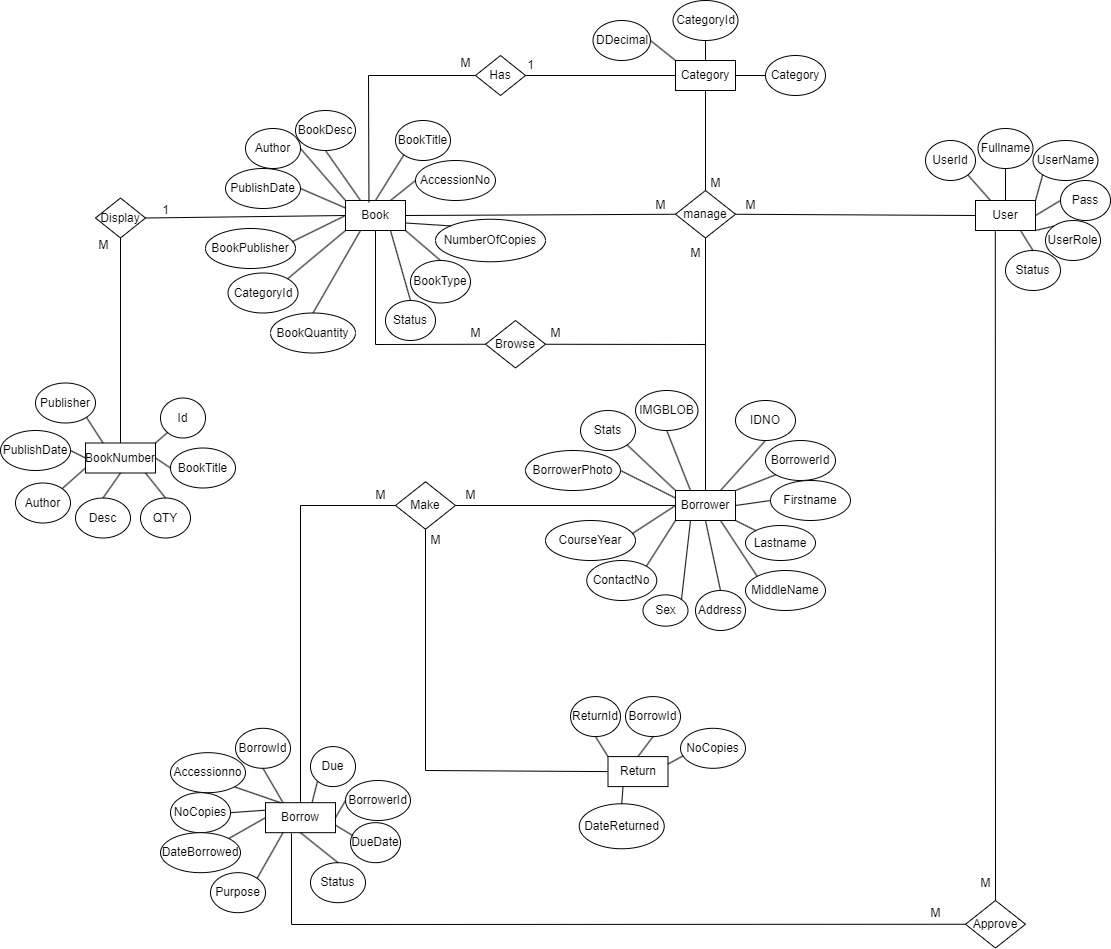
**6)Return:**

|  |  |  |
| --- | --- | --- |
| **Attributes** | **Data Type** | **Description** |
| **ReturnId** | Int(11) | Primary key, unique identifier for return records |
| **BorrowId** | int(11) | Foreign key referencing BorrowId from Borrow table |
| **NoCopies** | int(11) | Number of copies returned |
| **DateReturned** | datetime | Date and time when the books were returned |

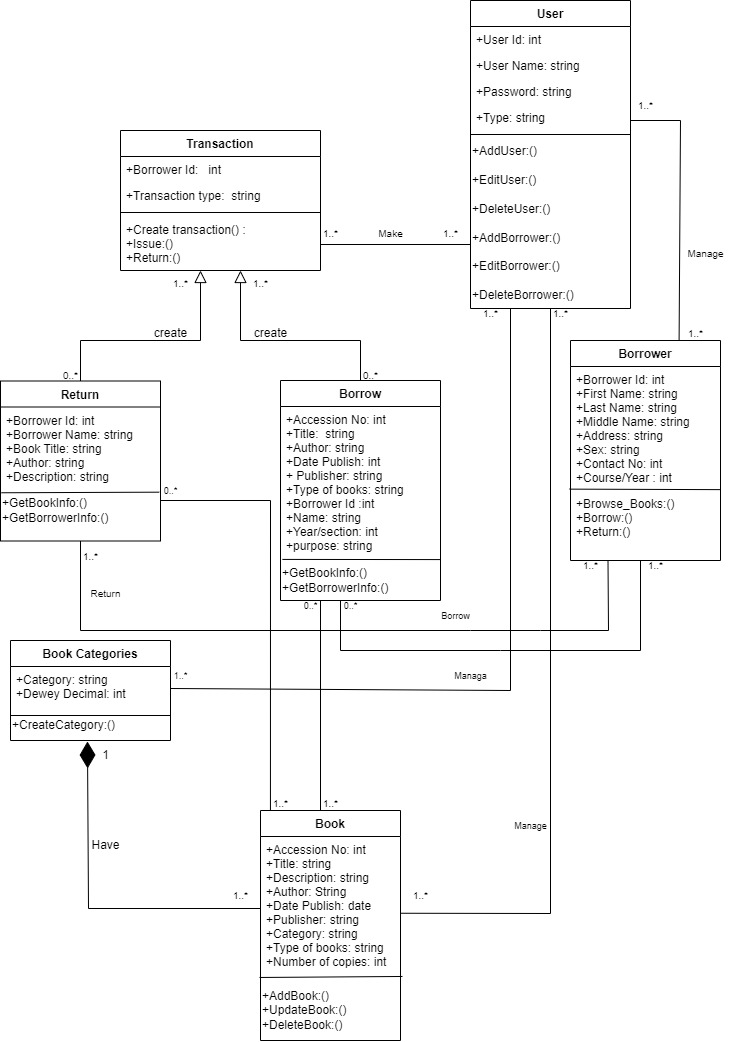
**7)User:**

|  |  |  |
| --- | --- | --- |
| **Attributes** | **Data Type** | **Description** |
| **UserId** | Int(11) | Primary key, unique identifier for user records |
| **Fullname** | varchar(125) | Full name of the user |
| **User\_name** | varchar(125) | User's username |
| **Pass** | varchar(125) | User's password (hashed or encrypted) |
| **UserRole** | varchar(125) | Role or level of access for the user |
| **Status** | varchar(11) | Status of the user (active, inactive, etc.) |

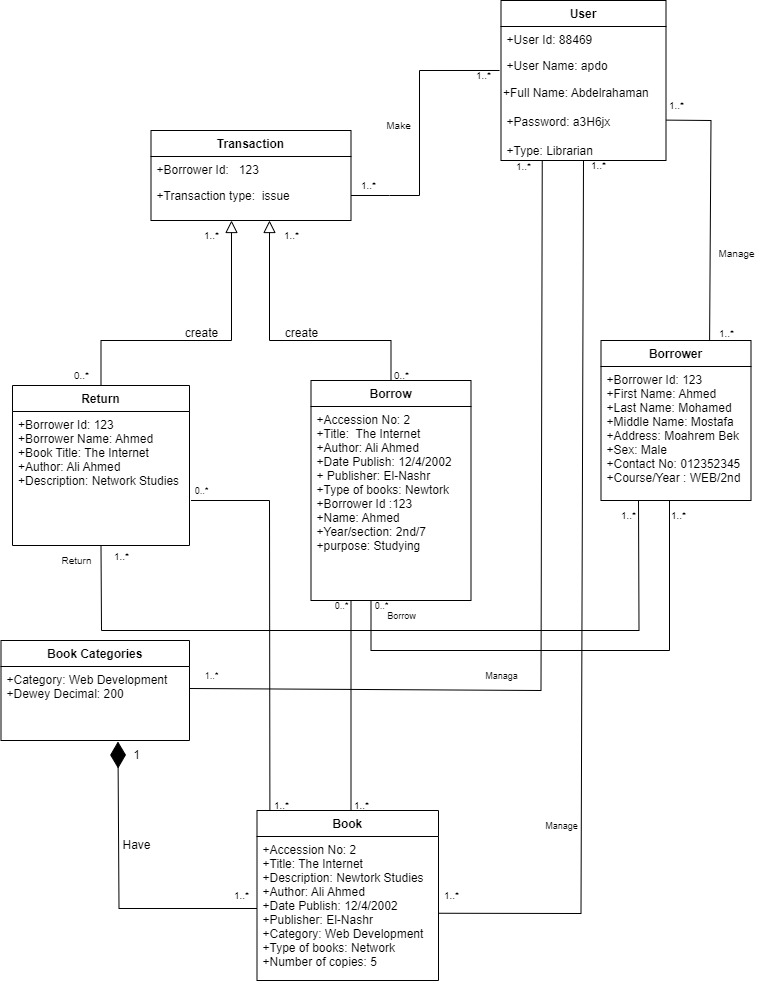
**ERD Diagram:**

****

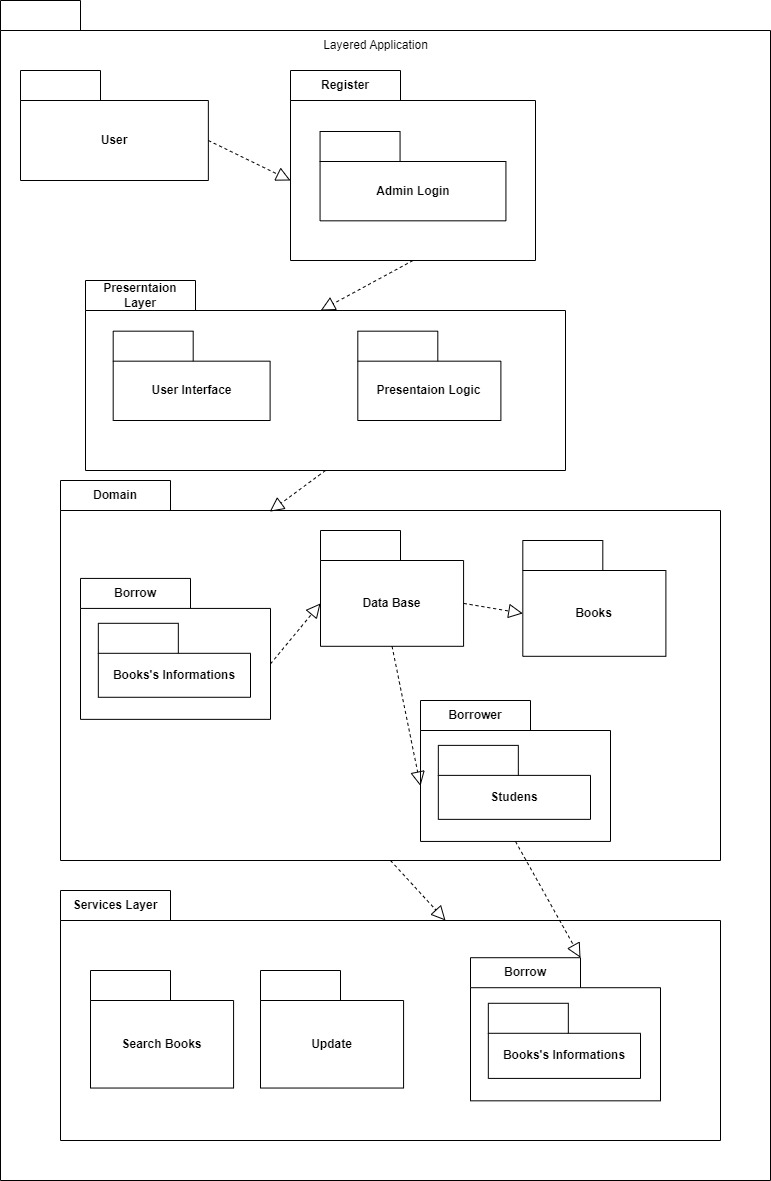
***Class Diagram*:**

****

***Object Diagram:***

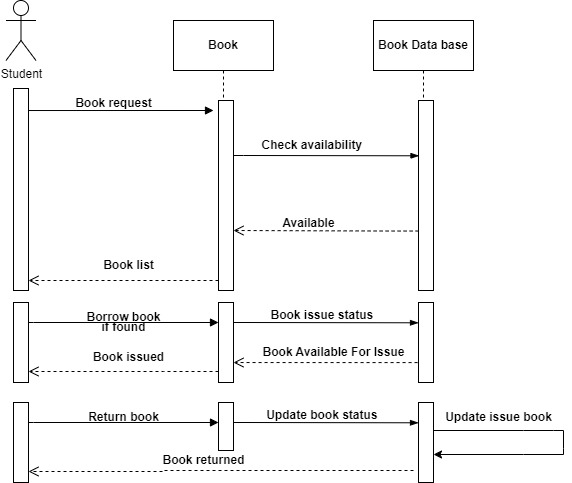
****

***Package Diagram*:**

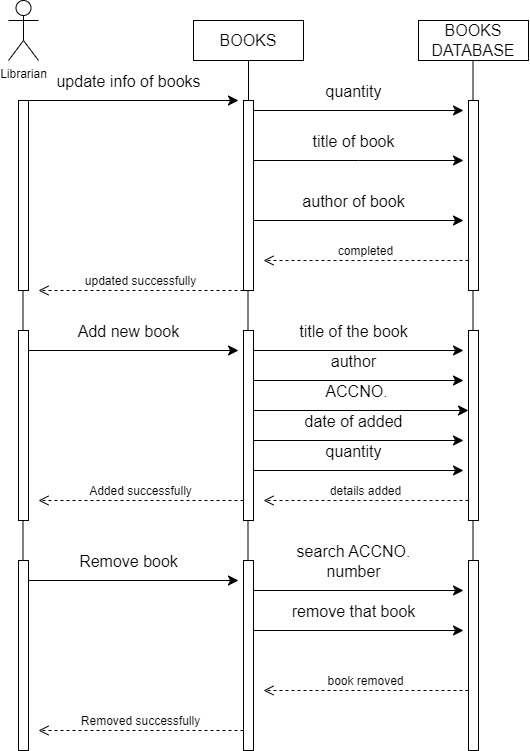
****

***Sequence Diagram(s)*:**

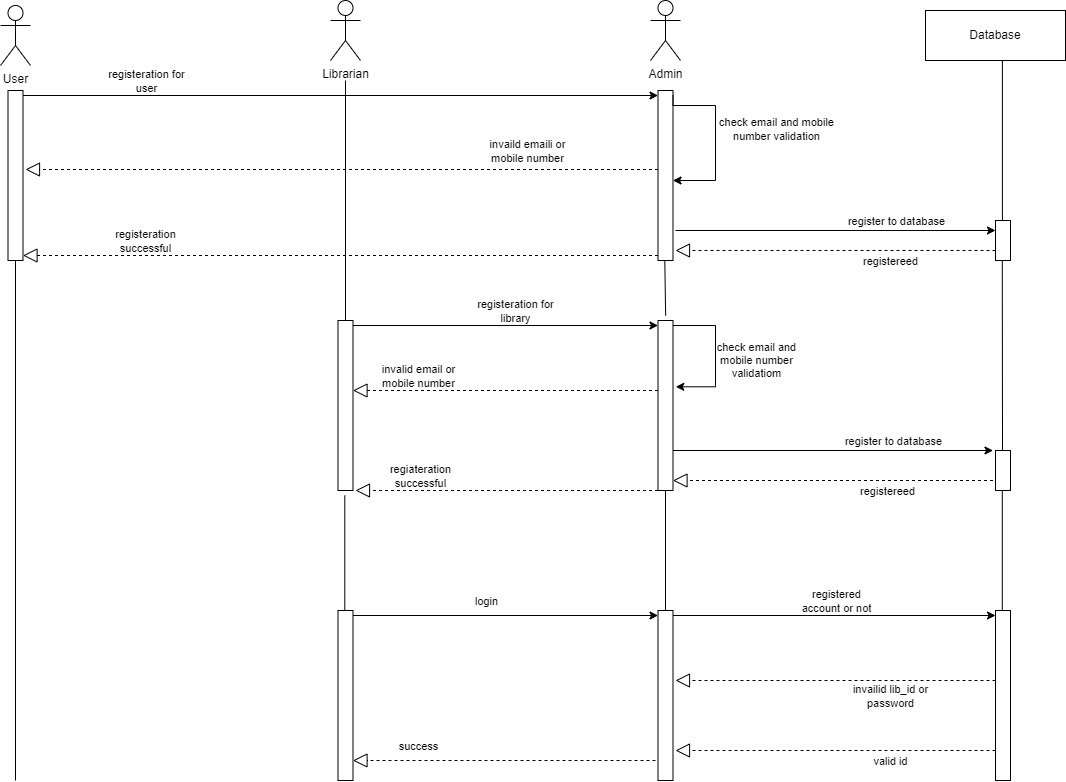
***Issue & Return*:**

****

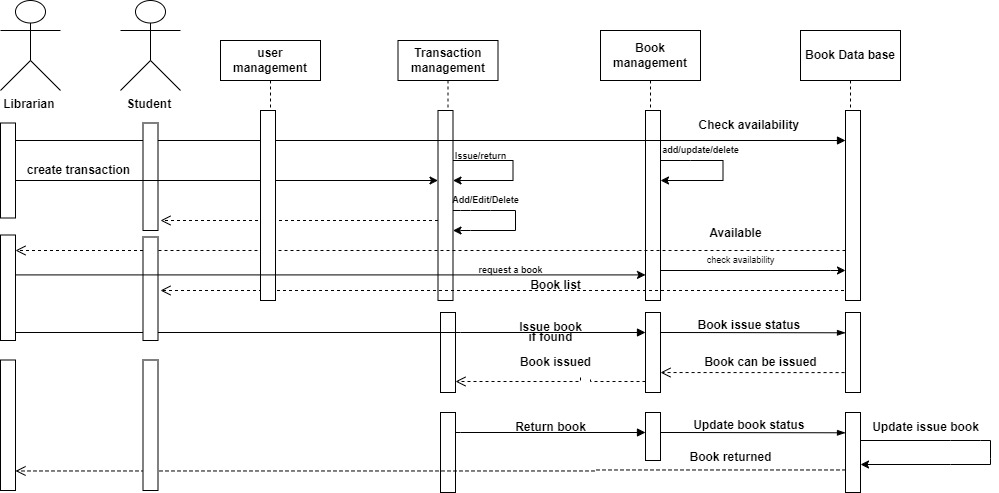
***Managing Books*:**

****

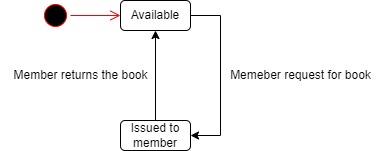
***User & Borrower Management*:**

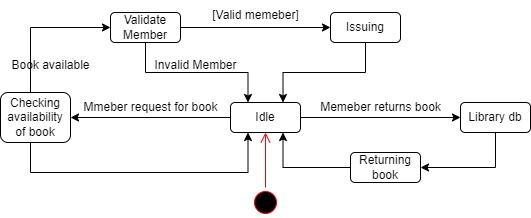
****

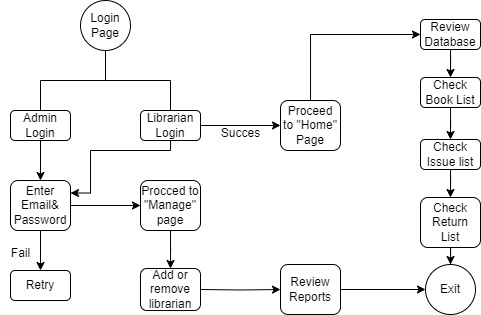
***System Sequence Diagram*:**



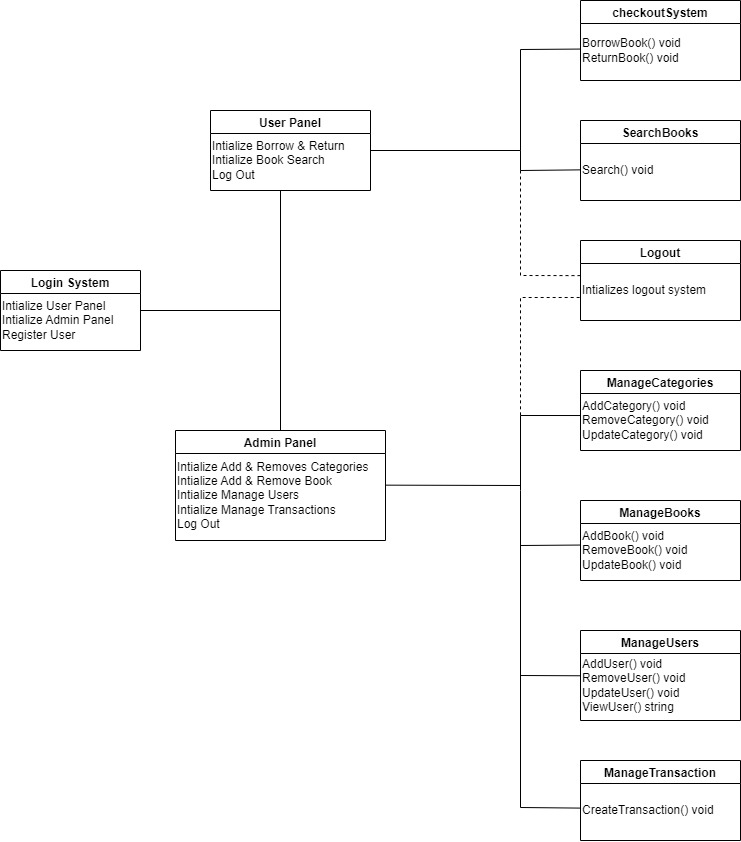
***State-Machine Diagrams*:**

****

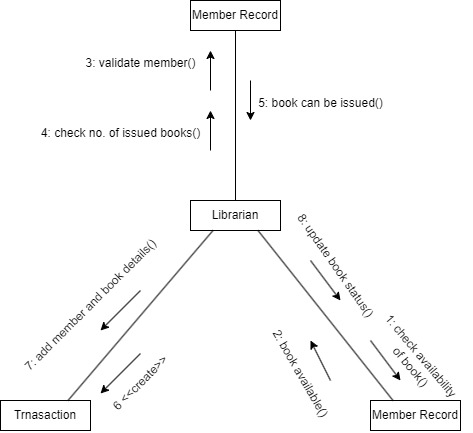
****

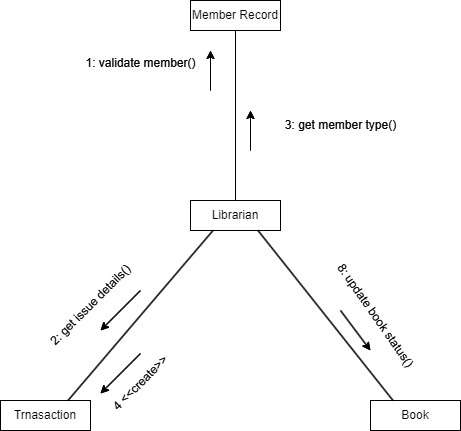
****

***Deployment Diagram*:**

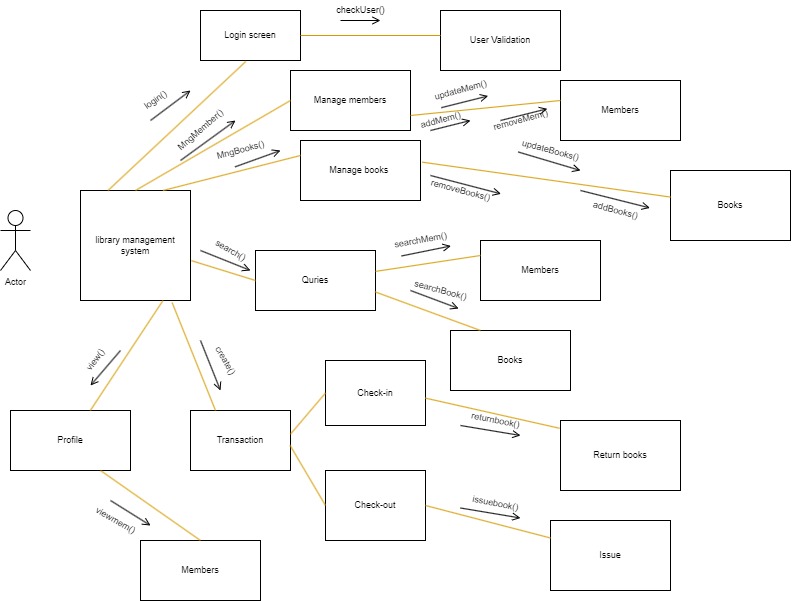
****

***Collaboration Diagrams:***

******

******

***Communication Diagram*:**

****