Systems Analysis

and Design

Instructor : Huang, Chuen-Min

**Teamwork1 ver.1**

Group 6

|  |  |
| --- | --- |
| ID | Name |
| A10523006 | Maggie |
| A10523049 | Peggy |
| B10423003 | Kurumi |
| B10423029 | Bean |
| B10523020 | Kendy |
| B10523030 | Jerry |
| B10523053 | Lynn |
| M10723001 | Joe |
|  |  |
|  |  |
| Date 2018/5/9 | |

Content

[**Introduction** 1](#_Toc513488588)

[**Use case diagram** 2](#_Toc513488589)

[**Use case description** 3](#_Toc513488589)

[1. Search Book 3](#_Toc513488590)

[2. Borrow Book 4](#_Toc513488591)

[3. Read E-Book 5](#_Toc513488592)

[4. Manage Book(the most important) 6](#_Toc513488593)

[5. Manage Member 8](#_Toc513488594)

[6. Manage E-Book 10](#_Toc513488595)

[7. Return Book 12](#_Toc513488596)

[8. Check Book Overdue 13](#_Toc513488597)

[**Activity diagram** 14](#_Toc513488598)

[1. Search Book Information 14](#_Toc513488599)

[2. Borrow Book 15](#_Toc513488600)

[3. Read E-Book 16](#_Toc513488601)

[4. Manage Book 17](#_Toc513488602)

[5. Manage Member 18](#_Toc513488603)

[6. Manage E-Book 19](#_Toc513488604)

[7. Return Book 20](#_Toc513488605)

[8. Check Book Overdue 21](#_Toc513488606)

[**Sequence diagram** 22](#_Toc513488607)

[**Class diagram** 23](#_Toc513488608)

[**Behavior state machine** 24](#_Toc513488609)

[**Participate In Assignments** 25](#_Toc513488610)

**Introduction**

Our subject, The Library Management System, is mainly for the problems or functions you might meet or use when you are at the library. Regarding using the library’s website, we excluded the cases outside the library. And we focus on book, ignore other objects’ situations. (e.g. tablet, discussion room)

* On the basis of different user, they will get different authority.
* When they are in the building, the system will identify their identity by membership card.
* It’s difficult to present the form of “a card”, so we’ll use “an account” to replace “a card”.
* The system will identify user’s identity by their account while they’re using the system.

**Use case diagram**

|  |
| --- |
|  |

* Members can have self-borrowing-service (self-serve when borrowing books) or reading E-Books.
* The only things guests can do are searching data or reading books in the library, namely only in the building, otherwise it will be refused.
* Librarian can administrate data of all the books, E-books and memberships.
* Returning books must be done through the librarian.
* Every 12:00 a.m. the system checks if there are books out of date and renew the status of borrowed-out books. If books weren't returned on time, it will stop the member's borrow book right and inform members.

1. Search Book

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name:** | Search Book | **ID:** | 1 | **Importance Level:** | | High |
| **Primary Actor:** | Member | **Use Case Type:** | | | Essential, Detail | |
| **Stakeholders and Interest:** | | | | | | |
| Member – someone can search book and read E-book  Guest – someone can only search book but can’t read E-book | | | | | | |
| **Brief Description:** | | | | | | |
| This use case describes user can search book in system. When user finds any book want to know more. User can view that book. And only member can read content in E-book. | | | | | | |
| **Trigger:** | User search book and view data about it. Member can also read E-book. | | | | | |
| **Type:** | External | | | | | |
| **Relationship:** | | | | | | |
| **Association:** | Member, Guest | | | | | |
| **Include:** |  | | | | | |
| **Extend:** |  | | | | | |
| **Generalization:** |  | | | | | |
| **Normal Flow of Event:** | | | | | | |
| 1. The user login in system, if input wrong account or password, input it again. 2. The user clicks search-book button to use search-book function. 3. The user input key word and find book he want, if didn’t find anything, the user need to input other key word. 4. Screen will show any book searched by the user, if the user don’t want to view book, stop in show book list. 5. The user can choose the book he want to view. 6. The user can view book information.   If the user wants to view E-book,  the S-1: view E-book information performed. | | | | | | |
| **Sub Flow:** | | | | | | |
| S-1: View E-book Information   1. Show E-book information to the user. If the user don’t want to read E-book, stop in show E-book information. 2. The user want to read E-book,   if the user is guest, show message tell he/she isn’t a member,  if the user is member, show E-book content | | | | | | |
| **Alternative / Exception Flow:** | | | | | | |
|  | | | | | | |

1. Borrow Book

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name:** | Borrow Book | **ID:** | 2 | | **Importance Level:** | | High |
| **Primary Actor:** | Member | | | **Use Case Type:** | | Essential, Detail | |
| **Stakeholders and Interest:** | | | | | | | |
| Member –The people who has been registered in the system and wants to borrow paper book. | | | | | | | |
| **Brief Description:** | | | | | | | |
| The member who wants to borrow paper book, and needs to login system to borrow book. | | | | | | | |
| **Trigger:** | Member wants to borrow paper book, then clicks the “Borrow Book” button. | | | | | | |
| **Type:** | External | | | | | | |
| **Relationship:** | | | | | | | |
| **Association:** | Member | | | | | | |
| **Include:** |  | | | | | | |
| **Extend:** |  | | | | | | |
| **Generalization:** |  | | | | | | |
| **Normal Flow of Event:** | | | | | | | |
| 1. The member logins to the system, if he inputs wrong account or password, then input it again. 2. The member clicks “Borrow Book” button and executes borrowing process. 3. The member needs to input book id to system, if system doesn’t find the book id, then the member needs to input book id again. 4. The system will check book’s state, if the book is unavailable, show error message and stop borrow book process. 5. The system will check member’s state, if member has overdue book, show error message and stop borrow book process. 6. If the book borrowed successful, show successful message. 7. The system will change book’s state to borrowed. 8. If the member wants to borrow other book, then go to step3. If the member completes borrowing book, then close the window. | | | | | | | |
| **Sub Flow:** | | | | | | | |
| **Alternative / Exception Flow:** | | | | | | | |
| 1. If the member wants to borrow the book that the state is available and the member doesn’t have overdue book, but it can’t be borrowed. The member needs to bring the book to librarian. | | | | | | | |

1. Read E-Book

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name:** | Read Book | **ID:** | 3 | | **Importance Level:** | | High |
| **Primary Actor:** | Member | | | **Use Case Type:** | | Essential, Detail | |
| **Stakeholders and Interest:** | | | | | | | |
| Member –The people who has been registered in the system and wants to read e-book. | | | | | | | |
| **Brief Description:** | | | | | | | |
| The member who wants to read e-book, and needs to login system to read e-book. | | | | | | | |
| **Trigger:** | Member wants to read e-book, then clicks the button. | | | | | | |
| **Type:** | External | | | | | | |
| **Relationship:** | | | | | | | |
| **Association:** | Member | | | | | | |
| **Include:** |  | | | | | | |
| **Extend:** |  | | | | | | |
| **Generalization:** |  | | | | | | |
| **Normal Flow of Event:** | | | | | | | |
|  | | | | | | | |
| **Sub Flow:** | | | | | | | |
| **Alternative / Exception Flow:** | | | | | | | |
|  | | | | | | | |

1. Manage Book(the most important)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name:** | Manage Book | **ID:** | 4 | | **Importance Level:** | | High |
| **Primary Actor:** | Librarian | | | **Use Case Type:** | | Essential, Detail | |
| **Stakeholders and Interest:** | | | | | | | |
| Librarian – someone needs to create, delete and edit book’s data. | | | | | | | |
| **Brief Description:** | | | | | | | |
| The librarian can create new book data, edit book data, or delete book data when he needs. | | | | | | | |
| **Trigger:** | Create new book data, edit or delete when the librarian needs. | | | | | | |
| **Type:** | External | | | | | | |
| **Relationship:** | | | | | | | |
| **Association:** | Librarian | | | | | | |
| **Include:** |  | | | | | | |
| **Extend:** |  | | | | | | |
| **Generalization:** |  | | | | | | |
| **Normal Flow of Event:** | | | | | | | |
| 1. The librarian logins to the system, if he inputs wrong account or password, input it again. 2. The librarian chooses function.   If the librarian wants to create a new book data,  then go to S-1: create book data is performed.  If the librarian wants to edit book data,  then go to S-2: edit book data is performed.  If the librarian wants to delete book data,  then go to S-3: delete book data is performed. | | | | | | | |
| **Sub Flow:** | | | | | | | |
| S-1: Create book information   1. The librarian needs to input book id. 2. System will check book id, if the book id has been used, then the librarian needs to input book id again. 3. The librarian inputs any book’s information it needs to be saved. 4. Saving every book’s information that the librarian has input.   S-2: Edit book information   1. The librarian needs to input book id to the system, if system doesn’t find book id, then the librarian needs to input book id again. 2. The librarian can update any book’s information except book id. 3. Saving every book’s information that the librarian has changed.   S-3: Delete book information   1. The librarian needs to input book id to system, if system doesn’t find book id, then the librarian needs to input book id again. 2. The system will check book’s state, if the book has been borrowed, stop delete book information 3. The system will alarm the librarian, does he really want to delete book?   If the librarian clicks ‘OK’ button, then the book will be deleted in database.  If the librarian clicks “Cancel” button, then go to step 2 of normal flow. | | | | | | | |
| **Alternative / Exception Flow:** | | | | | | | |
|  | | | | | | | |

1. Manage Member

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name:** | Manage Member | **ID:** | 5 | **Importance Level:** | | High |
| **Primary Actor:** | Librarian | **Use Case Type:** | | | Essential, Detail | |
| **Stakeholders and Interest:** | | | | | | |
| Librarian – someone needs to create, delete, and edit member’s data. | | | | | | |
| **Brief Description:** | | | | | | |
| The librarian can create new member data, edit member data, or delete member data when he needs. | | | | | | |
| **Trigger:** | Create new member data, or edit and delete when the librarian needs. | | | | | |
| **Type:** | External | | | | | |
| **Relationship:** | | | | | | |
| **Association:** | Librarian | | | | | |
| **Include:** |  | | | | | |
| **Extend:** |  | | | | | |
| **Generalization:** |  | | | | | |
| **Normal Flow of Event:** | | | | | | |
| 1. The librarian logins to system, if he inputs wrong account or password, input it again. 2. The librarian chooses function.   If the librarian wants to create a new member data,  then go to S-1: create member information is performed.  If the librarian wants to edit member data,  then go to S-2: edit member information is performed.  If the librarian wants to delete member data,  then go to S-3: delete member information is performed. | | | | | | |
| **Sub Flow:** | | | | | | |
| S-1: Create member information   1. The librarian needs to input member id. 2. System will check the member id, if the member id has been used, then librarian needs to input member id again. 3. The librarian inputs any member’s information it needs to be saved. 4. When the librarian clicks “Save” button, system will show a confirm message to the librarian.   If the librarian clicks “Save” button, saving then the member data that the librarian inputs will be saved in database.  If the librarian clicks “Cancel” button, system will go back to step 2 of normal flow.  S-2: Edit book information   1. The librarian needs to input member id to system, if system doesn’t find member id, then the librarian needs to input member id again. 2. The librarian can update any member information except member id. 3. When the librarian clicks “save” button, system will show a confirm message to the librarian.   If the librarian clicks “Save” button, saving then the member data that the librarian changes will be saved in database.  If the librarian clicks “Cancel” button, system will go back to step 2 of normal flow.  S-3: Delete book information   1. The librarian needs to input member id to system, if system doesn’t find member id, then the librarian needs to input member id again. 2. The system will check member’s state, if the member has overdue book, show warning message and stop deleting member information.   If the librarian clicks “Delete” button, the member will be delete in database  If the librarian clicks “Cancel” button, system will go back to step 2 of normal flow. | | | | | | |
| **Alternative / Exception Flow:** | | | | | | |
|  | | | | | | |

1. Manage E-Book

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name:** | Manage E-book | **ID:** | 6 | **Importance Level:** | | High |
| **Primary Actor:** | Librarian | **Use Case Type:** | | | Essential, Detail | |
| **Stakeholders and Interest:** | | | | | | |
| Librarian – someone needs to create, delete, and edit E-book’s data. | | | | | | |
| **Brief Description:** | | | | | | |
| The librarian can create new E-book data in system. Then librarian can edit or delete E-book data when he needs. | | | | | | |
| **Trigger:** | Create new E-book data, or edit and delete when the librarian needs. | | | | | |
| **Type:** | External | | | | | |
| **Relationship:** | | | | | | |
| **Association:** | Librarian | | | | | |
| **Include:** |  | | | | | |
| **Extend:** |  | | | | | |
| **Generalization:** |  | | | | | |
| **Normal Flow of Event:** | | | | | | |
| 1. The librarian logins to system, if he inputs wrong account or password, input it again. 2. The librarian chooses function.   If the librarian wants to create a new E-book data,  then go to S-1: create E-book information is performed,  If the librarian wants to edit E-book data,  then go to S-2: edit E-book information is performed,  If the librarian wants to delete E-book data,  then go to S-3: delete E-book information is performed. | | | | | | |
| **Sub Flow:** | | | | | | |
| S-1: S-1: Create E-book information   1. The librarian needs to input E-book id. 2. System will check E-book id, if the E-book id has been used, then the librarian needs to input E-book id again. 3. The librarian inputs any E-book’s information it needs to be saved. 4. When the librarian clicks “Save” button, system will show a confirm message to the librarian.   If the librarian doesn’t confirm, system will go back to choose function page.   1. If librarian confirm create, saving every E-book information librarian has input.   S-2:S-2: Edit E-book information   1. The librarian need to input E-book id in system, if system didn’t find E-book id, then librarian need to input E-book id again. 2. The librarian can update any E-book information except E-book id. 3. When librarian click save button, system will show a confirm message to librarian.   If the librarian clicks “Save” button, the E-book data that is changed will be saved in database.  If the librarian clicks “Cancel” button, system will go back to step 2 of normal flow.  S-3:S-3: Delete E-book information   1. The librarian needs to input E-book id to system, if system doesn’t find E-book id, then the librarian needs to input E-book id again. 2. When the librarian clicks “delete” button, system will show a confirm message to librarian.   If the librarian clicks “Delete” button, the E-book data will be deleted in database  If the librarian clicks “Cancel” button, system will go back to step 2 of normal flow. | | | | | | |
| **Alternative / Exception Flow:** | | | | | | |
|  | | | | | | |

1. Return Book

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name:** | Return Book | **ID:**7 |  | **Importance Level:** | | High |
| **Primary Actor:** | Librarian | **Use Case Type:** | | | Essential, Detail | |
| **Stakeholders and Interest:** | | | | | | |
| Librarian – someone needs to handle return books. | | | | | | |
| **Brief Description:** | | | | | | |
| The use case describes when someone bring back book and the librarian should handle it. | | | | | | |
| **Trigger:** | Librarian handle return book. | | | | | |
| **Type:** | External | | | | | |
| **Relationship:** | | | | | | |
| **Association:** | Librarian | | | | | |
| **Include:** |  | | | | | |
| **Extend:** |  | | | | | |
| **Generalization:** |  | | | | | |
| **Normal Flow of Event:** | | | | | | |
| 1. The librarian login to system, if he inputs wrong account or password, input it again. 2. The librarian clicks “return book” button to use return-book function. 3. The librarian needs to input book id to system, if system doesn’t find book id, then the librarian needs to input book id again. 4. The system will check book’s state.   If the book is overdue, reduce member’s overdue book.   1. The system will clean book’s borrower, book time and overdue time. | | | | | | |
| **Sub Flow:** | | | | | | |
|  | | | | | | |
| **Alternative / Exception Flow:** | | | | | | |
|  | | | | | | |

1. Check Book Overdue

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Name:** | Check overdue time | **ID:** | 8 | **Importance Level:** | | High |
| **Primary Actor:** | Time | **Use Case Type:** | | | Essential, Detail | |
| **Stakeholders and Interest:** | | | | | | |
| Time – When midnight (24:00) comes, will check every borrowed book is overdue or not. | | | | | | |
| **Brief Description:** | | | | | | |
| This function will work on midnight (24:00) every day. It will check every book is overdue or not. If it is overdue, it will change member’s state and book’s state to overdue. | | | | | | |
| **Trigger:** | midnight (24:00) | | | | | |
| **Type:** | External | | | | | |
| **Relationship:** | | | | | | |
| **Association:** | Time | | | | | |
| **Include:** |  | | | | | |
| **Extend:** |  | | | | | |
| **Generalization:** |  | | | | | |
| **Normal Flow of Event:** | | | | | | |
| 1. The time will check every book’s state.   If the book is overdue,  the S-1: change book state is performed. | | | | | | |
| **Sub Flow:** | | | | | | |
| S-1: Change Book State   1. The book’s state will change to overdue. 2. The member who borrowed this book will plus 1 overdue book in state. | | | | | | |
| **Alternative / Exception Flow:** | | | | | | |
|  | | | | | | |

**Activity diagram**

1. Search Book Information

|  |
| --- |
|  |

1. Borrow Book

|  |
| --- |
|  |

1. Read E-Book

|  |
| --- |
|  |

1. Manage Book

|  |
| --- |
|  |

1. Manage Member

|  |
| --- |
|  |

1. Manage E-Book

|  |
| --- |
|  |

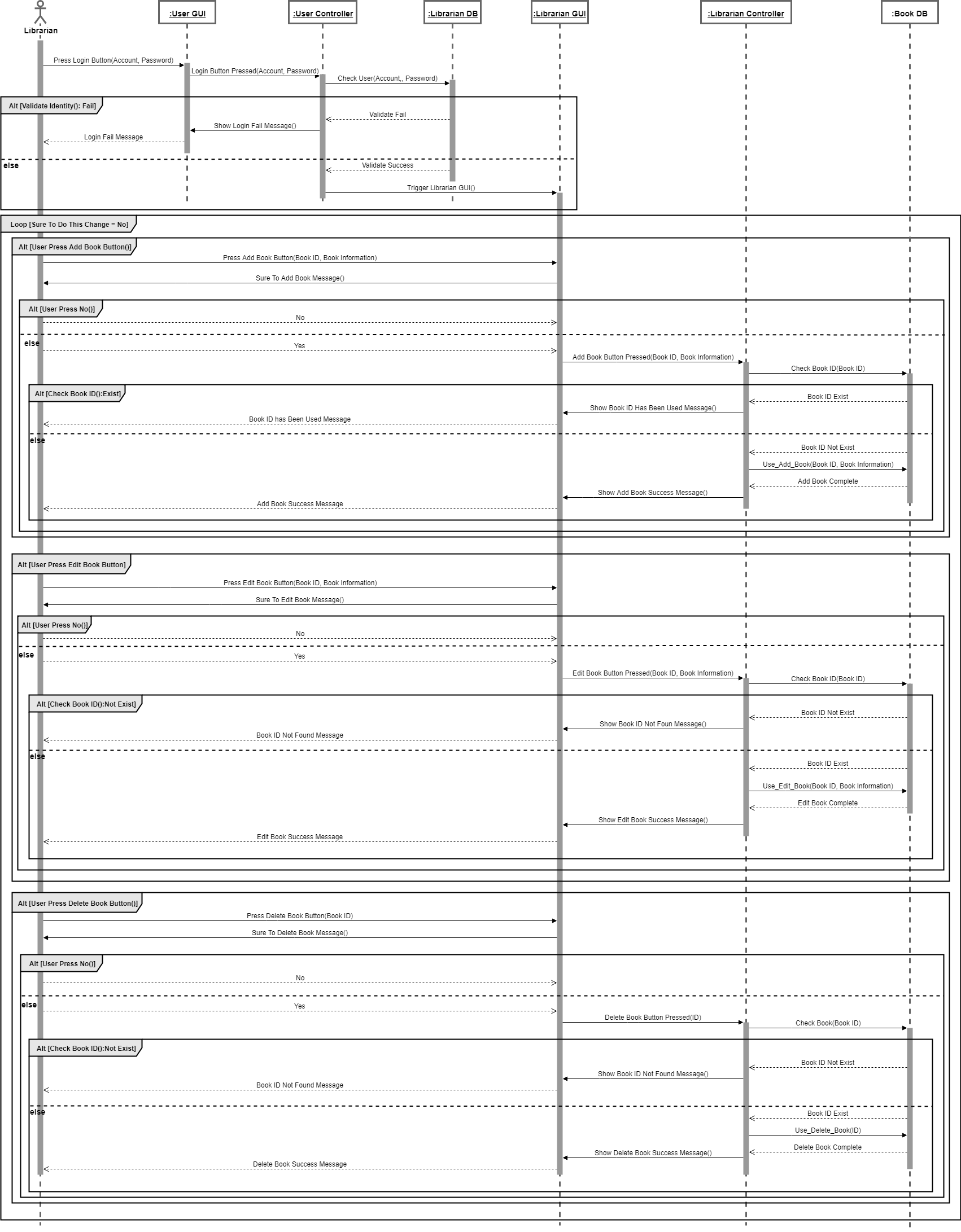
1. Return Book

|  |
| --- |
|  |

1. Check Book Overdue

|  |
| --- |
|  |

**Sequence diagram**



**Class diagram**

|  |
| --- |
|  |

**Behavior state machine**

|  |
| --- |
|  |

**Participate In Assignments**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Name | Participate | Responsibility |
| A10523006 | Maggie | 100% | Use case diagram  Activity diagram  Class diagram |
| A10523049 | Peggy | 100% | Use case diagram  Use case description  Activity diagram  Class diagram |
| B10423003 | Kurumi | 100% | Word  Introduction  Use case diagram  Use case description  Activity diagram  Class diagram  Sequence diagram  Behavior state machine |
| B10423029 | Bean | 0% |  |
| B10523020 | Kendy | 100% | Use case diagram  Use case description  Activity diagram  Class diagram  Sequence diagram  Behavior state machine |
| B10523030 | Jerry | 100% | PPT  Use case diagram  Activity diagram  Use case description  Behavior state machine  Class diagram  Behavior state machine |
| B10523053 | Lynn | 100% | Use case diagram  Use case description  Activity diagram  Class diagram  Sequence diagram  Behavior state machine |
| M10723001 | Joe | 100% | Use case diagram  Use case description  Activity diagram  Class diagram  Sequence diagram  Behavior state machine |