University of Maryland Global Campus

Library Management System (LMS)

Project Design

CMSC 495 7380

11 April 2023

Version 1.6

**Group 3: Brandon Durham, Benjamin Kus, Benjamin Ramos**

Revision History

REVISION	DATE	DESCRIPTION	Author
1.0	4/4/2023	Initial	Ben Ramos
1.1	4/5/2023	Addition of Pseudocode	Brandon Durham
1.2	4/5/2023	Document modification	Ben Ramos
1.3	4/7/2023	Inclusion of Event Scenarios	Ben Kus
1.4	4/8/2023	Modification of Pseudocode Addition of Risk Mitigation	Brandon Durham Ben Kus
1.5	4/11/2023	Risk Management	Brandon Durham
1.6	4/11/2023	Addition of scenario descriptions Modification to pseudocode Document modifications	Ben Kus Brandon Durham Ben Ramos

TABLE OF CONTENTS

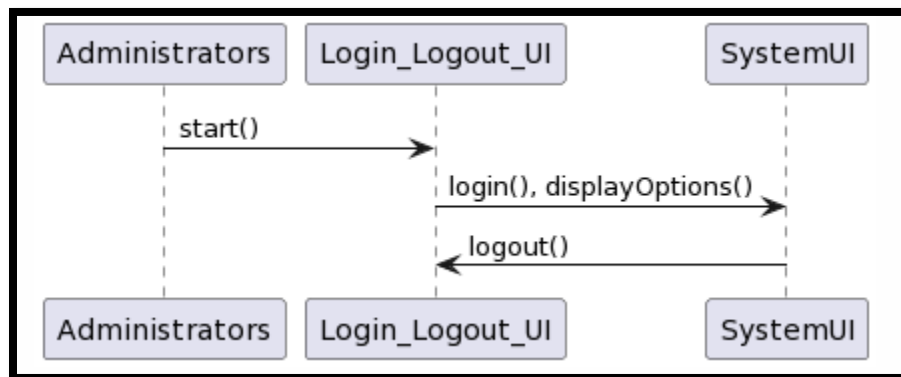# 1. Event Trace Scenario

Start-Up Scenario
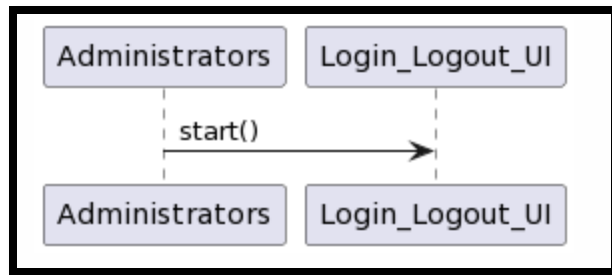
Description: The system shall provide the administrator with a login screen

Precondition: The administrator met software/hardware requirements and has installed the application. The administrator has the application open.

Post-condition: The administrator is at the "Admin Menu."
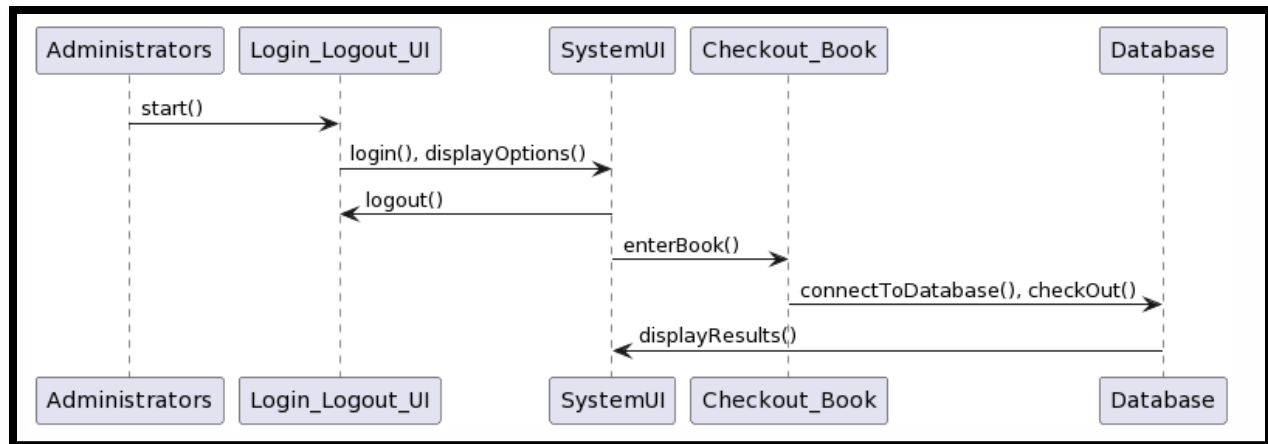

Scenario 1

Description: The system shall provide an administrator-specific display to conduct admin role

Precondition: The administrator has successfully logged into the application

Post-condition: The administrator will have a number of menu options to conduct roles


Scenario 2

Description: The system shall provide an administrator-specific display to check out books to include library borrower's information during checkout

Precondition: The administrator has successfully logged into the application and is displaying the menu options

Post-condition: The administrator will have issued a book. The admin will have connected to the database and updates were made. The connection to the database will close and the admin menu will be displayed.
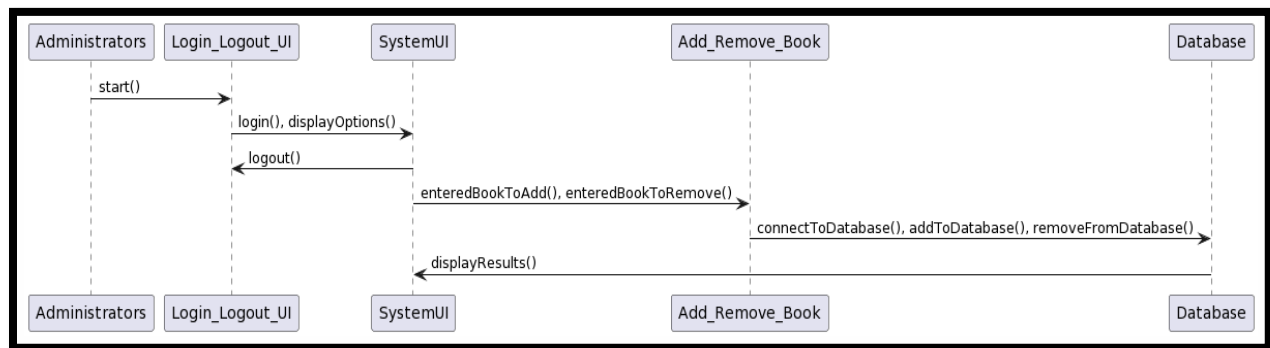
Scenario 3
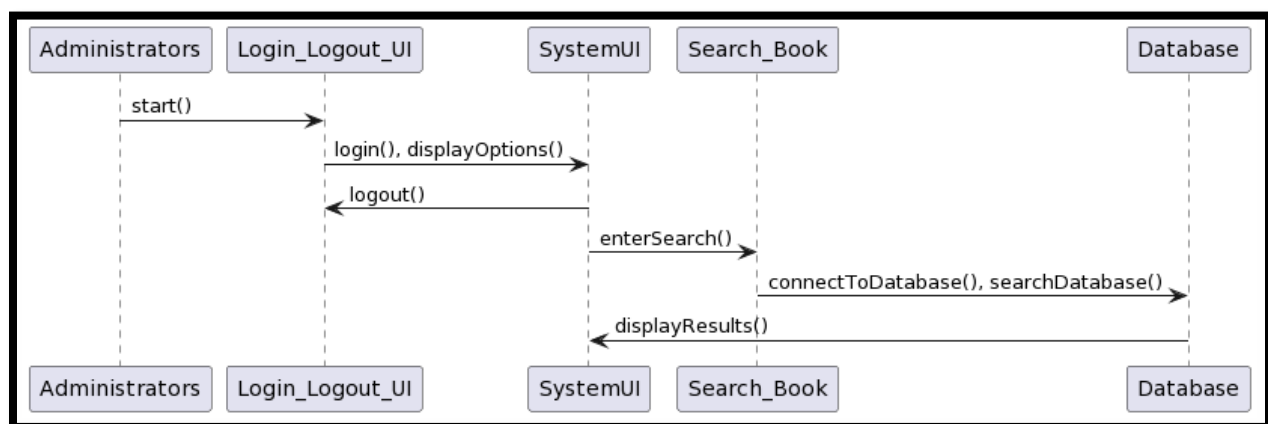
Description: The system shall allow the administrator to search for books based on title, author, etc.

Precondition: The administrator is at the "book search menu".

Post-condition: The administrator will have entered the search criteria and a result will be displayed on "search results". The menu will display current information related to the search result.

Scenario 4

Description: The system shall allow administrators to add and remove books

Precondition: The administrator is at the "Add/Remove book menu".

Post-condition: If the administrator is adding a book, they will have input the required information and have updated the database. The menu will appear with the book information that was entered. If a book is removed, the administrator will search for the book and remove it. The database will be updated in each scenario to reflect the addition or removal.

Shut Down Scenario
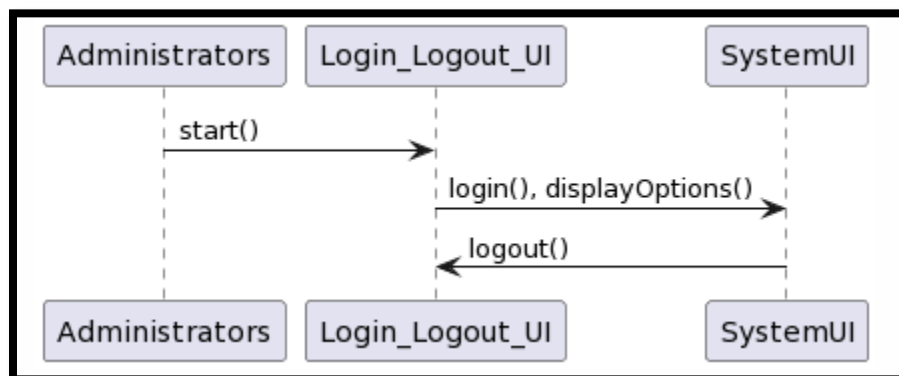
Description: The system shall provide the ability for admins to log out and end a session

Precondition: The application is open, and the administrator has logged in.

Post-condition: The application will have closed. The administrator will have either closed the application which will result in it signing out or they can sign out which will bring the administrator back to the login prompt.

Error Handling Scenarios
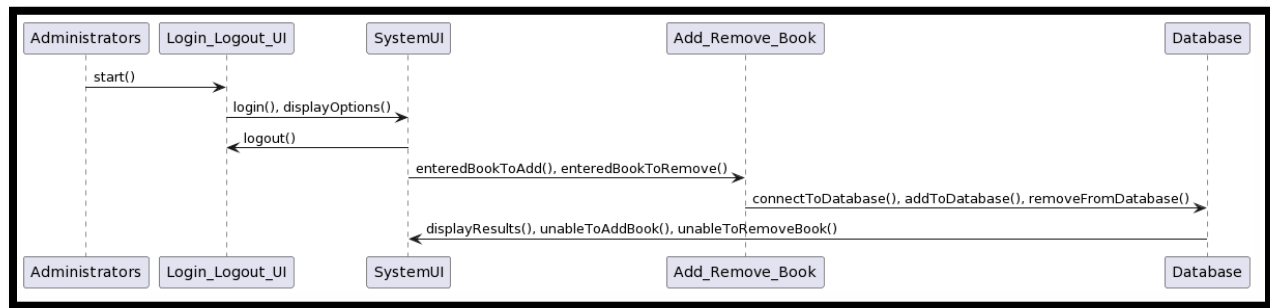
Description: The system will identify when books are trying to be added that already exist or are being removed that don't exist.

Precondition: The administrator has successfully logged into the application and is displaying the search menu options

Post-condition: The application will display an error that the book was previously added to the inventory and cannot be duplicated. If the book doesn't exist an error will be displayed that the book does not exist.

2. Class Design
Below is the pseudocode with the various classes and subsystems.

```
Login_Logout_UI
main
        start ()
Login()
        Login_info {User_Name, Password}
        (If both username and pswd match, allow logon
        Else don't allow logon and display error message)
        displayOption()
        LoginError ()
                (Error Message for failed login)
        Logout()
                exit()
```

System_UI
     enterBook()
     enterSearch()
     enterBookToAdd()
     enterBookToRemove()

Add_Remove_Book
     (If Add_Remove button selected)
     connectToDatabase()
         (Inserts addBook info to Database)
     addtoDatabase()
         Book {Title, Author, ISBN)
     removeFromDatabase()
         Remove_Book {ISBN}
         (Delete book from the database)
     displayResults()
         (Produces a summary message)

Search_Book
     (If Search_Book button selected)
     ConnectoDatabase()
         (Query Database for entered info)
     searchDatabase()
         Book {Title, Author, ISBN)
     displayResults()

Checkout_Books
     (If Checkout_Books button selected)
     ConnectoDatabase()
         (Inserts books into a "checkout" database, insert borrowers info)
     checkout()
         Book {Title, Author, ISBN)
         Person (Full Name, Address, Email}

DataBase
     (When requests are made or info is inserted/deleted)
     DisplayResults()
     UnableToAddBook()
         (Error message that book has been previously added)
     UnableToRemoveBooks()
         (Error message that book is unable to be removed)

3. Risk Mitigation

Identify risks	Risk mitigation
Admin enters login password incorrectly	Admin will be displayed a warning message of possible locked account
Invalid inputs such as special characters, letter grades, negative scores or grades higher than one hundred.	System will verify usage of invalid characters during data entry
Password complexity requirement	Require admins to meet password complexity requirements
Chance of data loss/modification by admin	Display a warning message prior to deleting/saving over a record