**LiBook**

*Represents modern online library with simple and user-friendly UI to interact with information about books and authors.*

The goal of the system is to simplify the process of searching books and authors for user. The service provides quick access to the main information about books and their authors available across the system. The system allows to manage items (books and authors) in a really easy way using comfortable modern interface. In addition, amongst options available, an end user can add books to the wish list and find out where the books he/she likes is available to buy. Also users can leave comments to book which he/she like.

**Subsystems**

***Book Subsystem***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№** | **Including Use Case** | **Role** | **Use Case** | **Description** | **Action Method** | **Tests** |
| BS1 | - | Admin | Edit book | Update information about book such as general information (title, description etc.), picture and data about authors. | Edit(EditBookViewModel model) |  |
| BS2 | - | Add book | Create new books adding general information (title, description etc.), picture. | Add(CreateBookViewModel model) |  |
| BS3 | - | Delete book | Delete all information available about book. This action can not be undone. | Delete(DeleteBookViewModel model) |  |
| BS4 | - | Web Customer | Search book | Search all information available about book. | Search(SearchBookViewModel model) |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№** | **Including Use Case** | **Role** | **Use Case** | **Description** | **Action Method** | **Tests** |
| AS1 | - | Admin | Edit author | Update information about author such as general information (first name, last name etc.), picture. | Edit(EditAuthorViewModel model) |  |
| AS2 | - | Add author | Create new authors adding general information (title, description etc.), picture. | Add(AddAuthorViewModel model) |  |
| AS3 | - | Delete author | Delete all information available about author. This action cannot be undone. | Delete(DeleteAuthorViewModel model) |  |
| AS4 | - | Web Customer | Search author | Search all information available about author. | Search(SearchAuthorViewModel model) |  |

***Author Subsystem***

***Wishlist Subsystem***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№** | **Including Use Case** | **Role** | **Use Case** | **Description** | **Action Method** | **Tests** |
| WS1 | - | Registered Customer | Add to wish list | Add books to the wish list | Add(AddToWishListViewModel model) |  |
| WS2 | - | Delete from wish list | Remove books from the wish list | Delete(DeleteFromWishListViewModel model) |  |
| WS3 | Add to Wish list | Add note | Add a note why this book should be in his/her wish list | AddNote(AddNoteViewModel model) |  |

***User Subsystem***

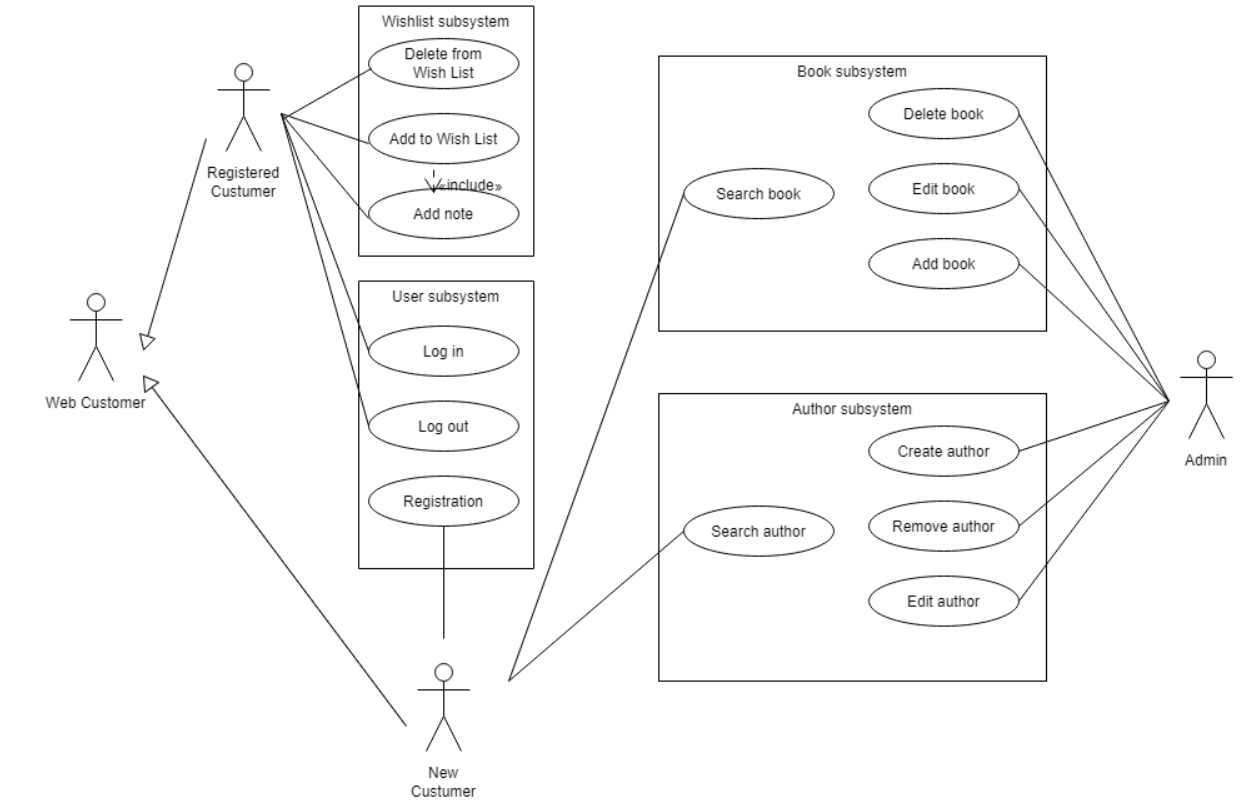
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Role** | **Use Case** | **Description** | **Action Method** | **Tests** |
| US1 | Registered Customer | Log in | Authorize in system | LogIn(LogInViewModel model) |  |
| US2 | Log out | Log out from system | LogOut(LogOutViewModel model) |  |
| US3 | New Customer | Registration | Registration in system | Registration(RegistrationViewModel model) |  |

***Tests***

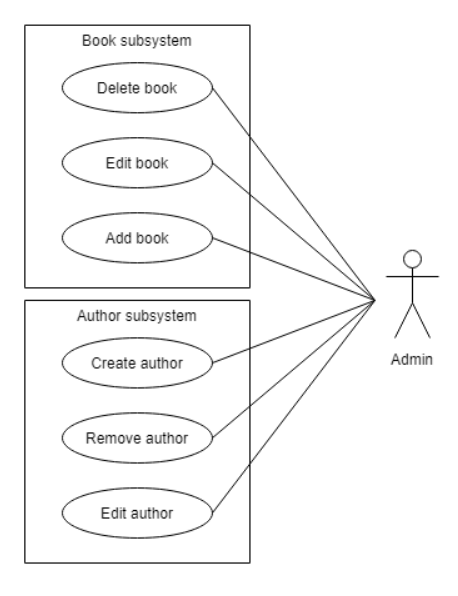
|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Test class** | **Test method** | **Code coverage** |
|  |  |  |  |
|  |  |  |  |

**Diagrams**

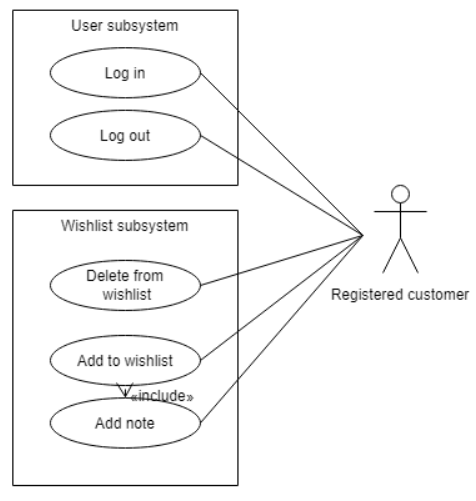
1. Use cases.
   1. General use case diagram



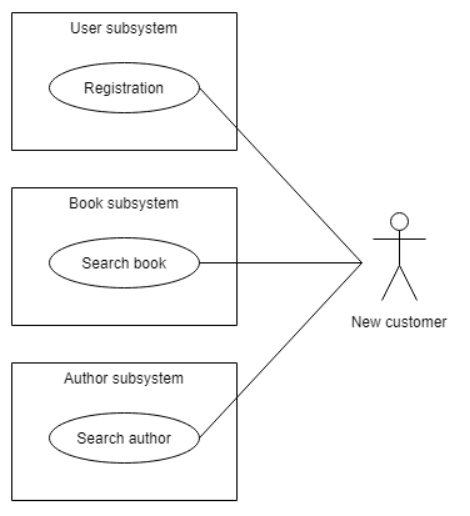
1.2 Use case for Admin role



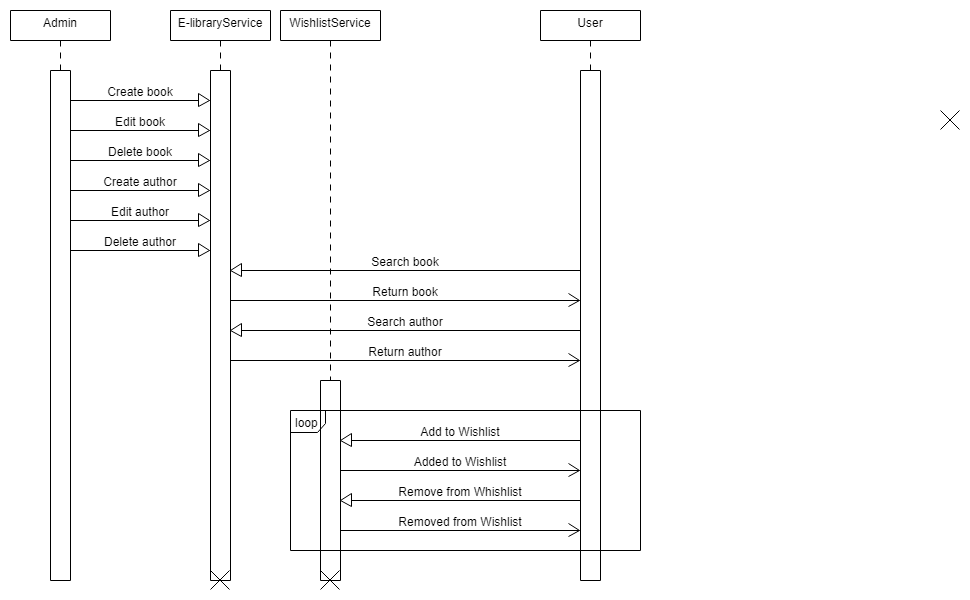
* 1. Use case for registered user



* 1. User case for guest



1. Sequence.



1. Class.

A screenshot of a cell phone

Description automatically generated