**Use Case – Library System**

|  |  |  |  |
| --- | --- | --- | --- |
| ID and name | UC-1: Login to the Library System | | |
| Primary actor | User | Secondary actors | Library Administrator |
| Description | This use case describes the process by which a library user logs into the library system to access their account and perform various operations. | | |
| Trigger | The user wishes to access their library account and initiates the login process. | | |
| Preconditions | PRE-1: The user must be registered in the library system.  PRE-2: The user must have valid login credential  PRE-3: The library system must be online and available. | | |
| Postconditions | POST-1: The user is successfully authenticated and granted access to their account.  POST-2: The user is now able to select the book(s) that they want to borrow. | | |
| Normal flow | 1. The user navigates to the library system's login page. 2. The system prompts the user to enter their username, and password. 3. The user inputs their credentials and submits the login form. 4. The system verifies the credentials against stored user data. 5. If the credentials are valid, the system grants access to the user's account and displays the list of the books that are available to borrow | | |
| Alternative flows | * 1. **If the user has forgotten their password:**  1. The user clicks on the "Forgot Password" link 2. The system prompts the user to enter their registered email. 3. The user enters their email and personal numeric code and submits the request. 4. The system shows a pop-up with user’s credentials 5. The use case continues from step 2 of the normal flow after the password is reset.    1. **If the user is not registered:** 6. The system displays a message stating that no account is associated with the entered credentials. 7. The system provides an option to register a new account. 8. The user is redirected to the registration page. 9. The user completes the registration process and then returns to the login page. 10. The use case continues from step 2 of the normal flow after successful registration. | | |
| Exceptions | **E1: If the user enters incorrect credentials:**   1. The system displays an error message stating that the username or password is incorrect. 2. The system prompts the user to try again. 3. The user enters the correct credentials: 4. The use case continues from step 1 of the normal flow | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID and name | UC-2: Borrow a Book | | |
| Primary actor | Library User | Secondary actors | Library System |
| Description | This use case describes the process by which a library user borrows one or more books from the system using a terminal. | | |
| Trigger | The user wishes to borrow a book and is authenticated in the system. | | |
| Preconditions | PRE-1: The user must be registered and authenticated in the library system.  PRE-2: The book must be available for borrowing.  PRE-3: The user must not exceed their borrowing limit. | | |
| Postconditions | POST-1: The book is marked as borrowed under the user’s account.  POST-2: The system updates the available books list in real-time.  POST-3: The borrowing transaction is recorded in the system for future tracking. | | |
| Normal flow | 1. The user logs into the library system at a terminal. 2. The system displays a list of available books. 3. The user searches for a book by title, author, or category. 4. The system displays search results, including book availability. 5. The user selects one or more books to borrow. 6. The system verifies that the selected books are available. 7. The system checks the user’s borrowing limit and eligibility. 8. If eligible, the system marks the books as borrowed under the user’s account. 9. The system updates the list of available books in real-time. 10. The system confirms the transaction and provides a due date for returning the books. 11. The use case ends successfully. | | |
| Alternative flows | **1.1: If the user does not find a desired book in stock:**   1. The system offers the option to place a hold or request a notification when the book becomes available. 2. The user may choose to reserve the book for future borrowing. 3. The system records the reservation and ends the process. | | |
| Exceptions | **E1:** If the book is no longer available after selection, the system notifies the user and asks them to select another book. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID and name | UC-3: Return a Book | | |
| Primary actor | Library User | Secondary actors | Librarian |
| Description | This use case describes the process by which a library user returns a borrowed book. | | |
| Trigger | The user wishes to return a book and goes to the return counter. | | |
| Preconditions | PRE-1: The user must have at least one borrowed book. | | |
| Postconditions | POST-1: The book is marked as returned in the system.  POST-2: The system updates the available books list. | | |
| Normal flow | 1. The user selects the “Return Book” option from the terminal 2. The system shows the user a list with all their borrowed books 3. The system updates the book’s status to "available." 4. The system confirms the return and updates the available books list. 5. The use case ends successfully. | | |
| Alternative flows | 1. If the user returns multiple books, the system processes them one by one. | | |
| Exceptions | E1: If the book’s code is not recognized, the system investigates the issue before proceeding | | |

**PLANIFICAREA ITERATIILOR:**

Iterația 1:

* Creare baze de date
* Implementarea operațiilor CRUD
* Crearea interfetei grafice:

1. Interfața grafică a Bibliotecarului:

~ Posibilitatea de înregistrare în sistem

~ Buton prin care poate vizualiza lista cărților si statusul acestora (împrumutate / disponibile)

~ Buton de gestionare a cărților, ce permite utilizatorului să adauge/modifice/steargă o carte din baza de date

~ Buton de Log Out

1. Interfața grafică a Cititorului:

~ Posibilitatea de înregistrare în sistem

~ Buton prin care poate vizualiza lista cărților disponibile

~ Buton prin care poate solicita operațiunea de împrumutare a cărților

~ Buton de Log Out

Iterația 2:

* Validări:

~ Adăugarea validărilor în interfața grafică (formate corecte pentru datele de intrare).

~ Gestionarea cazurilor de eroare (ex. utilizator neînregistrat, carte indisponibilă).

* Funcționalități suplimentare:

~ Înregistrarea unui utilizator nou în sistem.

~ Implementarea operațiunii de returnare a cărților, cu actualizarea bazei de date și confirmarea pentru utilizator.

Iterația 3:

* Îmbunătățirea interfeței prin adăugarea unui motor de căutare avansat (după titlu, autor, categorie).
* Filtre pentru cărțile disponibile și împrumutate