Library Management System

Use Case Description

**Team members:**

* **Bao Quoc Nguyen - 614207**
* **Deivison Mozer de Souza - 614634**
* **Phong Lu - 614616**

**GitHub repository:**

* <https://github.com/phonglu-miu-edu/CS425-SE>

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 1** | | | |
| **Name** | | Register account | |
| **Brief description** | | The students who want to access the system need to register themself to use the system | |
| **Actors** | | Student | |
|  | | | |
| **Preconditions** | | | |
| There is no logged in account. | | | |
| **Flows of Events:** | | | |
| 1. Basic flow | | | |
| * 1. Register account | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The student goes to the registration page and enters the student id, then clicks “Register”. | | The system checks for the existence of the email starts with student id in the system.  If it does, show on the UI that the account existed, then the flow is done.  If it doesn’t, generate and send the email with temporary password for the student, then show on the UI that the email is sent. |
| 2 | (Out of system) The student checks email for the temporary password. | |  |
| 3 | The student goes to login page and enters email and temporary password, then click “Login”. | | The system checks for the combination of email and password. If it’s correct, show on the UI that the student needs to change the password. |
| 4 | The student enters the new password and the re-typed new password, then clicks “Change password”. | | The system changes the student password, then show on UI that the password is changed. On the UI, redirect the student to login page. |
| **Postconditions** | | | |
| The student has an account on the system with the student email as the username and the new password in step 4. | | | |
| **Business Rule** | | | |
| The account with the student email is unique. | | | |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 2** | | | |
| **Name** | | Login user | |
| **Brief description** | | Authenticate process to determine the user role to give that user right system functions | |
| **Actors** | | Admins, Librarians, Students | |
|  | | | |
| **Preconditions** | | | |
| There is no logged in account.  The user has an account in the system. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. Login with correct credentials | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the login page and enters correct the combination of the email and password, then clicks “Login”. | | The system checks for the validity of the combination.  It does, redirect user on the UI to the right page bases on their role. |
| **Postconditions** | | | |
| The user is redirected to the correct home page bases on their role. | | | |
| **Business Rule** | | | |
| All the users use the same login page. | | | |
|  | | | |
| * 1. Login with wrong credentials | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the login page and enters wrong the combination of the email and password, then clicks “Login”. | | The system checks for the validity of the combination.  It doesn’t, show on the UI that the combination is incorrect. |
| **Postconditions** | | | |
| The UI shows the credentials are incorrect. | | | |
| **Business Rule** | | | |
| System allows the user to enter wrong credentials 5 times continuously before locking it. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 3** | | | |
| **Name** | | Update Profile | |
| **Brief description** | | Modify the current own profile information | |
| **Actors** | | Admins, Librarians, Students | |
|  | | | |
| **Preconditions** | | | |
| The user logged in to the system. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. Update profile information | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the profile page and enters their new profile information, then clicks “Update”. | | The system validates any missing information and save new information. |
| **Postconditions** | | | |
| The user profile is updated. | | | |
| **Business Rule** | | | |
| The profile information includes:   * First name. * Last name. * Address. * Mobile phone.   The status of the account is not included. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 4** | | | |
| **Name** | | Update Contact | |
| **Brief description** | | Modify the current own contact information | |
| **Actors** | | Admins, Librarians, Students | |
|  | | | |
| **Preconditions** | | | |
| The user logged in to the system. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. Update contact information | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the contact page and enters their new password and the re-typed new password, then clicks “Update”. | | The system saves new password. |
| **Postconditions** | | | |
| The new password is updated. | | | |
| **Business Rule** | | | |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 5** | | | |
| **Name** | | Reset password | |
| **Brief description** | | Get the password reset when the user forgets it. | |
| **Actors** | | Admins, Librarians, Students | |
|  | | | |
| **Preconditions** | | | |
| There is no logged in account. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. Reset password | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the forgot password page and enters their email, then clicks “Reset password”. | | The system validates the existence of the email, then generate new temporary password and send an email containing that password to that email address. |
| 2 | (Out of system) The student checks email for the temporary password. | |  |
| 3 | The student goes to login page and enters email and temporary password, then click “Login”. | | The system checks for the combination of email and password. If it’s correct, show on the UI that the student needs to change the password. |
| 4 | The student enters the new password and the re-typed new password, then clicks “Change password”. | | The system changes the student password, then show on UI that the password is changed. On the UI, redirect the student to login page. |
| **Postconditions** | | | |
| The user password is updated. | | | |
| **Business Rule** | | | |
| The temporary password is only valid for 30 minutes. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 6** | | | |
| **Name** | | Find book | |
| **Brief description** | | Find the book with all the book detail. | |
| **Actors** | | Admins, Librarians, Students | |
|  | | | |
| **Preconditions** | | | |
| The user logged in to the system. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. Find book | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the search page, enters some keywords and some preset criteria, then click “Search”. | | The system searches the books by keywords and criteria, then returns the matching books. |
| **Postconditions** | | | |
| The list of matching books is returned. | | | |
| **Business Rule** | | | |
| The return list should be paged.  The book could be used in the “checkout book” use case. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 7** | | | |
| **Name** | | Add Admin/Librarian | |
| **Brief description** | | Add more Admins and Librarians for the system. | |
| **Actors** | | Admins | |
|  | | | |
| **Preconditions** | | | |
| The user logged in to the system is an Admin.  The wanting to create user does not exist in the system. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. Add Admin/Librarian | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the Admin/Librarian user page, enters the email of the new user, choose the role of that user to Admin or Librarian, then click “Add”. | | The system generates a random password, adds the new user to the system, and sends the email with that password to the user email. |
| **Postconditions** | | | |
| The new user is created. | | | |
| **Business Rule** | | | |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 8** | | | |
| **Name** | | Manage authors | |
| **Brief description** | | CRUD functions to manage the authors. | |
| **Actors** | | Admins | |
|  | | | |
| **Preconditions** | | | |
| The user logged in to the system is an Admin. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. List Authors | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the author page, (optional) select the criteria to filter the list. | | The system lists out the authors in the system in pages. |
|  | | | |
| * 1. Delete Author | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks the delete button on the author list. | | The system shows a confirmation. |
| 2 | The user clicks “Confirm” on the confirmation. | | The system (soft) deletes the author, then list the authors. |
|  | | | |
| * 1. Add Author | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks on the “Add new”. | | The system shows the add new author page. |
| 2 | The user enters author information to required fields, then click “Add”. | | The system adds new author, then return to the author page. |
|  | | | |
| * 1. Edit Author | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks the edit button on the author list. | | The system shows the edit author page with selected author information pre-filled. |
| 2 | The user enters new author information to required fields, then click “Edit”. | | The system updates the author information, then return to the author page. |
| **Postconditions** | | | |
| * 1.2: An author is (soft) removed. * 1.3: A new author is added. * 1.4: An author information is edited. | | | |
| **Business Rule** | | | |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 9** | | | |
| **Name** | | Manage book categories | |
| **Brief description** | | CRUD functions to manage the book categories. | |
| **Actors** | | Admins | |
|  | | | |
| **Preconditions** | | | |
| The user logged in to the system is an Admin. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. List Book Categories | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the book category page, (optional) select the criteria to filter the list. | | The system lists out the book categories in the system in pages. |
|  | | | |
| * 1. Delete Book Category | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks the delete button on the book category list. | | The system shows a confirmation. |
| 2 | The user clicks “Confirm” on the confirmation. | | The system (soft) deletes the book category, then list the book categories. |
|  | | | |
| * 1. Add Book Category | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks on the “Add new”. | | The system shows the add new book category page. |
| 2 | The user enters book category information to required fields, then click “Add”. | | The system adds new book category, then return to the book category page. |
|  | | | |
| * 1. Edit Book Category | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks the edit button on the book category list. | | The system shows the edit book category page with selected book category information pre-filled. |
| 2 | The user enters new book category information to required fields, then click “Edit”. | | The system updates the book category information, then return to the book category page. |
| **Postconditions** | | | |
| * 1.2: A book category is (soft) removed. * 1.3: A new book category is added. * 1.4: A book category information is edited. | | | |
| **Business Rule** | | | |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 10** | | | |
| **Name** | | Manage books | |
| **Brief description** | | CRUD functions to manage the books. | |
| **Actors** | | Admins | |
|  | | | |
| **Preconditions** | | | |
| The user logged in to the system is an Admin. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. List Books | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the book page, (optional) select the criteria to filter the list. | | The system lists out the books in the system in pages. |
|  | | | |
| * 1. Delete Book | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks the delete button on the book list. | | The system shows a confirmation. |
| 2 | The user clicks “Confirm” on the confirmation. | | The system (soft) deletes the book, then list the books. |
|  | | | |
| * 1. Add Book | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks on the “Add new”. | | The system shows the add new book page. |
| 2 | The user enters book information to required fields, then click “Add”. | | The system adds new book, then return to the book page. |
|  | | | |
| * 1. Edit Book | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks the edit button on the book list. | | The system shows the edit book page with selected book information pre-filled. |
| 2 | The user enters new book information to required fields, then click “Edit”. | | The system updates the book information, then return to the book page. |
| **Postconditions** | | | |
| * 1.2: A book is (soft) removed. * 1.3: A new book is added. * 1.4: A book information is edited. | | | |
| **Business Rule** | | | |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 11** | | | |
| **Name** | | Manage book copy | |
| **Brief description** | | Manage the copies of the books. | |
| **Actors** | | Admins | |
|  | | | |
| **Preconditions** | | | |
| The user logged in to the system is an Admin. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. List Book Copies | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the book copy page, list out the books, click on “Manage copies”. | | The system goes to the book copy page, lists out the book copies of the book in pages. |
|  | | | |
| * 1. Delete Book Copy | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks the delete button on the copy list. | | The system shows a confirmation. |
| 2 | The user clicks “Confirm” on the confirmation. | | The system (soft) deletes the copy, then list the book copies. |
|  | | | |
| * 1. Add Book Copy | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks on the “Add new”. | | The system shows the add new book copy page. |
| 2 | The user enters book copy information to required fields, then click “Add”. | | The system adds new book copy, then return to the book copies page. |
|  | | | |
| * 1. Edit Book Copy | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user clicks the edit button on the book copy list. | | The system shows the edit book copy page with selected book copy information pre-filled. |
| 2 | The user enters new book copy information to required fields, then click “Edit”. | | The system updates the book copy information, then return to the book copy page. |
| **Postconditions** | | | |
| * 1.2: A book copy is (soft) removed. * 1.3: A new book copy is added. * 1.4: A book copy information is edited. | | | |
| **Business Rule** | | | |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 12** | | | |
| **Name** | | Manage configuration table | |
| **Brief description** | | Manage the configuration of the system. | |
| **Actors** | | Admins | |
|  | | | |
| **Preconditions** | | | |
| The user logged in to the system is an Admin. | | | |
| **Flows of Events:** | | | |
| 1. Basic flows | | | |
| * 1. List Configuration | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user goes to the configuration page. | | The system lists out the configuration in form. |
|  | | | |
| * 1. Update Configuration | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user changes the value of the configuration in form, then clicks on “Save”. | | The system updates the configuration, shows on the UI success message. |
|  | | | |
| **Postconditions** | | | |
| The configuration is updated | | | |
| **Business Rule** | | | |
| The configuration is used to config variables for the entire system in real-time. | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 13** | | | |
| **Name** | | Unlock Student Account | |
| **Brief description** | | This use case describes the steps required in unlocking the student account when it was lock due to the violation of the policy of maximum number of overdue books or failing to return a book in time | |
| **Actors** | | Administrator | |
|  | | | |
| **Preconditions:** | | | |
| 1. The administrator must have the necessary permissions to unlock the student. 2. The student account must be locked due to specific reason/library policy. | | | |
| **Flows of Events:** | | | |
| 1. **The Administrator login into the system providing login and password.** | | | |
| **1.1** | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The system Administrator access the SYSTEM System and navigate to the page “Unlock Student Account” | | The system retrieves the account details and the reason for the lock from the database and displays a list of locked student accounts |
| 2 | Administrator identifies the account that needs to be unlocked and selects “unlock” option | | The system verifies the administrator's permissions to perform this action. |
| 3 | The Administrator confirms the unlock action. | | Update the Student account removing the lock from it. |
| **Postconditions** | | | |
| Student account status is changed from “locked” to “unlocked”.  The student can now access their account and continue using the SYSTEM System services without restrictions. | | | |
| **Business Rule** | | | |
| Administrator must have the appropriate permissions to unlock student accounts, ensuring that only authorized persons can perform this action. | | | |
| **Use Case Number: 14** | | | |
| **Name** | | List of Students | |
| **Brief description** | | This use case allows librarian/administrators to view the list of all registered students in the library system. | |
| **Actors** | | Admin | |
|  | | | |
| **Preconditions** | | | |
| The Administrator must be logged into the SYSTEM software. | | | |
| **Flows of Events:** | | | |
| 1. The Administrator selects the "List of Students" option from the main menu. 2. The system retrieves and displays a comprehensive list of all registered students in the library. | | | |
| **1.1** | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | Librarian/Administrator: Selects the "List of Students" option. | | Retrieves the list of all registered students and display the list of students on the screen. |
| **Postconditions** | | | |
| The librarian or administrator can now view the list of students and perform further actions if needed. | | | |
| **Business Rule** | | | |
| All students who have completed the registration process should be listed in the system. | | | |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 15** | | | |
| **Name** | | Checkout Book | |
| **Brief description** | | This use case allows librarians to check out a book for a library user | |
| **Actors** | | Admins, Librarians | |
|  | | | |
| **Preconditions** | | | |
| The librarian must be logged into the system.  The book to be checked out must be available in the library's collection.  The library user who wants to borrow the book should be a registered user in the system with an unlocked account. | | | |
| **Flows of Events:** | | | |
| The librarian access the “Check out” option in the main menu.  SYSTEM system displays the book options for checked-out. | | | |
| **1.1** | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | Selects the "Checkout Book" option | | The system asks to enter the user identification number or login. |
| 2 | Enters the library user identification or login | | Verifies the user identity in the database and displays the list of available books for checkout |
| 3 | Selects the book to be checked out from the list and confirm the operation. | | Updates the book's status to "Checked Out," assigns the due date, and records the transaction in the database. |
| **Postconditions** | | | |
| The book is marked as "Checked Out" in the system, and the library user is allowed to take the book home.  The due date for returning the book is recorded in the system for that user. | | | |
| **Business Rule** | | | |
| Only registered library users can check out books, and they can borrow a limited number of books at a time based on library policies. | | | |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 16** | | | |
| **Name** | | Return Book | |
| **Brief description** | | This use case describes the process of a library user returning a borrowed book | |
| **Actors** | | Admins, Librarians | |
|  | | | |
| **Preconditions** | | | |
| The library user must have previously borrowed a book from the library.  The book being returned must be in the library's collection and marked as checked out. | | | |
| **Flows of Events:** | | | |
| The system displays the "Return Book" interface.  The library user requests to return a book. | | | |
|  | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The library user selects the "Return Book" option from their account dashboard. | | The system validates the user credentials and displays a list of books currently checked out by the user. |
| 2 | The user selects the book they wish to return from the list. | | The system updates the book status from "Checked Out" to "Available" and adjusts the book's due date.  The system sends a notification to the user confirming the successful return of the book |
| **Postconditions** | | | |
| The book's status is updated to "Available" in the library's database.  The book is no longer associated with the library user account. | | | |
| **Business Rule** | | | |
| A book can only be returned if it is currently marked as "Checked Out" and associated with the library user account. | | | |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 17** | | | |
| **Name** | | List Checked Out Books by Criteria | |
| **Brief description** | | This use case involves listing checked-out books based on specific criteria | |
| **Actors** | | Admins, Librarians | |
|  | | | |
| **Preconditions** | | | |
| The librarian must be logged in to their account with appropriate permissions. | | | |
| **Flows of Events:** | | | |
| The librarian accesses the "List Checked Out Books" section in System.  System displays the options to filter and search for checked out books based on criteria. | | | |
|  | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The librarian navigates to the "Checked out Books" section in the system dashboard. | | The system presents the librarian with search bar where it can search for book title and ISBN. |
| 2 | The librarian selects the desired criteria to filter the list of checked out books | | The system retrieves the relevant information from the database and displays the filtered list of checked out books. |
| **Postconditions** | | | |
| The librarian can view the list of checked out books based on the selected criteria. | | | |
| **Business Rule** | | | |
| The librarian can only view the list of checked out books if they have appropriate permissions within system. | | | |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 18** | | | |
| **Name** | | List Overdue Books by Criteria | |
| **Brief description** | | This use case enables librarians to generate a list of overdue books based on specified criteria | |
| **Actors** | | Admins, Librarians | |
|  | | | |
| **Preconditions** | | | |
| The librarian must be logged in to their account with appropriate permissions. | | | |
| **Flows of Events:** | | | |
| The librarian accesses the "Overdue Books" section in System.  System displays the options to filter and search for overdue books based on criteria. | | | |
|  | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The librarian navigates to the "Overdue Books" section in the system dashboard. | | The system presents the librarian with search bar where it can search for book title and ISBN. |
| 2 | The librarian selects the desired criteria to filter the list of overdue books | | The system retrieves the relevant information from the database and displays the filtered list of overdue books. |
|  |  | |  |
| **Postconditions** | | | |
| The librarian can view the list of overdue books based on the selected criteria. | | | |
| **Business Rule** | | | |
| The librarian can only view the list of overdue books if they have appropriate permissions within system. | | | |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 19** | | | |
| **Name** | | Reset Password for Users | |
| **Brief description** | | This use case describes the process of resetting passwords for library users who have forgotten their login information. | |
| **Actors** | | Admins | |
|  | | | |
| **Preconditions** | | | |
| The user must have an account created into the system with email registered. | | | |
| **Flows of Events:** | | | |
| 1. Select “Forgot Password” option in the login page. 2. Allow reset the password for an existing account. | | | |
|  | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The user selects the option “Forgot Password” in the login page. | | The system presents a field to enter the username or email of the account. |
| 2 | The user inserts the login or email and select “reset password” option. | | The system sends an email for the user with a link to insert a new password. |
| 3 | The user opens the email and click on the link that will pop up a simple page to insert the new password and confirm. | | The system updates the new password into the user account. |
| **Postconditions** | | | |
| The user's password is successfully reset, allowing them to log in using the new password. | | | |
| **Business Rule** | | | |
| The system must provide a simple effective way of resetting password for library users. | | | |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 20** | | | |
| **Name** | | Check the Availabilities | |
| **Brief description** | | This use case allows library users to check the availability of books in the library's collection. | |
| **Actors** | | Admins, Librarians | |
|  | | | |
| **Preconditions** | | | |
| The library user must be logged in to their system account. | | | |
| **Flows of Events:** | | | |
| The library user accesses the "Book Availability" section in SYSTEM system.  System displays the search interface for book availability. | | | |
|  | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 | The library user navigates to the "Book Availability" section. | | The system presents the user with a search interface to enter book title, author, or other relevant information. |
| 2 | The user enters the book details they want to check the availability for. | | The system retrieves the availability status of the book from the database and displays the result. |
|  |  | |  |
| **Postconditions** | | | |
| The library user can see the availability status of the requested book. | | | |
| **Business Rule** | | | |
| The availability status of the book is based on the current records in the library's database. | | | |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 21** | | | |
| **Name** | | Remind student the due date | |
| **Brief description** | | This use case involves sending a reminder to a student about the due date of a borrowed book. The system will automatically generate and send the reminder to the student before the due date approaches. | |
| **Actors** | | System – Batch Scheduler | |
|  | | | |
| **Preconditions** | | | |
| The student must be a registered user of the library.  The student must have borrowed at least one book with a due date set. | | | |
| **Flows of Events:** | | | |
| The system monitors the due dates of all borrowed books in the library's database.  The system identifies students whose book(s) are nearing their due dates.  The system generates a reminder notification for each student with approaching due dates. | | | |
|  | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 |  | | The system checks the library's database for all borrowed books. |
| 2 |  | | The system calculates the remaining days until the due date for each book. |
| 3 |  | | The system generates and sends the reminder notification to the student. |
| 4 | The student receives the reminder notification through their email. | |  |
| **Postconditions** | | | |
| The student receives the reminder notification with the due date information. | | | |
| **Business Rule** | | | |
| The system sends the reminder notification to the student a specified number of days before the due date, as configured by the library administrators. | | | |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case Number: 22** | | | |
| **Name** | | Change Status of Student | |
| **Brief description** | | This use case involves changing the status of a student in the library system. The status change may reflect changes in usership, penalties, or special privileges granted to the student. | |
| **Actors** | | System – Batch Scheduler | |
|  | | | |
| **Preconditions** | | | |
| The student's usership information must exist in the library's database. | | | |
| **Flows of Events:** | | | |
| The librarian/administrator navigates to the student information page or dashboard. | | | |
|  | | | |
| **Step** | **User Actions** | | **System Actions** |
| 1 |  | | The system updates the student's status in the library's database with the changes made by the librarian/administrator. |
| **Postconditions** | | | |
| The student's status in the library system is updated to reflect the changes made by the librarian/administrator.  The librarian/administrator can view the updated status of the student on the student information page or dashboard. | | | |
| **Business Rule** | | | |
| Only authorized librarians/administrators can change the status of students in the library system.  The system should maintain a log of status changes for auditing and tracking purposes. | | | |
|  |  |  |  |