|  |  |
| --- | --- |
| Use Case: Login | |
| Actors: | Librarian, Member, Administrator (User) |
| Precondition: | User is not logged in |
| Post-condition: | User is logged in |
| Main (happy) Path: | 1. User enters username and password then presses the “Login” button 2. If entered correctly the main page for the user will open |
| Alternative Path: | 1. If entered incorrectly the user will stay on the login page with an error message depending on the type of incorrectness 2. User will be able to try again or use the “Forgot Password” feature to get a new password. |

|  |  |
| --- | --- |
| Use Case: Logout | |
| Actors: | Librarian, Member, Administrator (User) |
| Precondition: | User is logged in and on his main page |
| Post-condition: | User is not logged in |
| Main (happy) Path: | 1. User presses the “Logout” button 2. User is logged out of the system and returned to the login page |
| Alternative Path: | 1. Unable to logout, user will be able to try again. 2. If the system is closed without logging out then a message will be shown to confirm logging out as well. |

|  |  |
| --- | --- |
| Use Case: Search Book | |
| Actors: | Librarian, Member (User) |
| Precondition: | 1. Member is logged in and on the member main page 2. Librarian is logged in and on the librarian search book page |
| Post-condition: | User is still logged in and in the same page but with the results of his search |
| Main (happy) Path: | 1. User enters the books name they are wish to search for 2. User clicks on the “Search” button 3. Results for what is searched is shown on the table |
| Alternative Path: | If entered book is not found an error message is displayed |

|  |  |
| --- | --- |
| Use Case: Self-check-out | |
| Actors: | Member |
| Precondition: | Member is logged in and on main page |
| Post-condition: | Member is still logged in and on the check-out page |
| Main (happy) Path: | 1. Member clicks the Check Out button 2. Member is directed to the Check Out Page 3. Member types the barcode of the book he wishes to checkout 4. Member clicks the Check Out button 5. A “Successful” message is shown |
| Alternative Path: | 1. If unsuccessful an error message will be shown 2. The member can choose to go to the home page either way |

|  |  |
| --- | --- |
| Use Case: Self-check-in | |
| Actors: | Member |
| Precondition: | Member is logged in and on main page |
| Post-condition: | Member is still logged in and on the check-in page |
| Main (happy) Path: | 1. Member clicks the Check In button 2. Member is directed to the Check In Page 3. Member types the barcode of the book he wishes to check in 4. Member clicks the Check In button 5. A “Successful” message is shown |
| Alternative Path: | 1. If unsuccessful an error message will be shown 2. The member can choose to go to the home page either way |

|  |  |
| --- | --- |
| Use Case: Add Name to Book Waiting List | |
| Actors: | Member |
| Precondition: | Member is logged in and on the check-out page |
| Post-condition: | Member is still logged in and on the member home page |
| Main (happy) Path: | 1. Member types the barcode of the book he wishes to checkout 2. Member clicks the Check Out button 3. A message pops up saying the book is unavailable and asks the member if he wishes to add it to waiting list 4. Member clicks yes 5. A “Successful” message is shown |
| Alternative Path: | 1. If the member enters no he is directed to the home page and the action is aborted |

|  |  |
| --- | --- |
| Use Case: View My Books | |
| Actors: | Member |
| Precondition: | Member is logged in and on main page |
| Post-condition: | Member is still logged in and in the same page but with the results of his query |
| Main (happy) Path: | 1. Member selects either reserved, borrowed or read from the drop down menu 2. Member clicks the “Search” button 3. Member’s books of the selected query will be listed |

|  |  |
| --- | --- |
| Use Case: Issue Late Fine | |
| Actors: | System |
| Precondition: | System notices a book being borrowed for 20 days |
| Post-condition: | The member who borrowed said book can see their fine in their account page |
| Main (happy) Path: | 1. System issues a fixed fine for every day after 20 days the book hasn’t been returned (the member who checked the book out) 2. System stops issuing additional fines when the book is flagged as returned (checked in by the member) |
| Alternative Path: | 1. The fine is incremented until 20 increments, then the system blocks you from borrowing new books until the fine is paid |

|  |  |
| --- | --- |
| Use Case: Pay Fine | |
| Actors: | Member |
| Precondition: | Member is logged in and on the account page |
| Post-condition: | Member is still logged in and on the main user page |
| Main (happy) Path: | 1. Member sees the amount of fine they need to pay 2. Member clicks “Pay” and pays all of it (no billing system is present) 3. A “Successful” message is shown and the member is directed to the user page |

|  |  |
| --- | --- |
| Use Case: Notify of Book Being Available | |
| Actors: | System |
| Precondition: | A member has a book in their waiting list |
| Post-condition: | That member gets a notification |
| Main (happy) Path: | 1. System receives information that a book that was on a waiting list is now marked as returned (checked in) 2. System sends a message to the member |
| Alternative Path: | 1. The book is checked out before the message is seen. Tough luck to that member, the system doesn’t care. |

|  |  |
| --- | --- |
| Use Case: Manipulate Book | |
| Actors: | Librarian |
| Precondition: | Librarian is logged in and on the “Librarian” page |
| Post-condition: | Librarian is still logged in and is either in the “Add Book” page or the “Librarian” page |
| Main (happy) Path: | 1. Librarian wishing to add a book clicks the “Add Book” button 2. (Optional) The same page can be accessed through the “Books” page 3. Librarian is directed to the “Add Book” page 4. He fills out the empty spaces with book information 5. Then clicks the “Save” button 6. A “Successful” message is displayed, a book has been created 7. Librarian stays on the same page (in case he wishes to repeat the process) 8. Librarian can return to the librarian page with the “Home” button 9. Librarian wishing to delete a book clicks the “Delete Book” button 10. He is directed to the “Delete Book” page 11. Name of the book being deleted is entered 12. Librarian can click “Search” to see if that book exists (Optional) or click “Delete” to proceed 13. After a “Successful” message, librarian is returned to the “Librarian” page |
| Alternative Path: | 1. Conflict of name can occur during the addition of a book thus an error will be displayed 2. Librarian cannot click “Save” if all the fields are empty, doing so will prompt an error message 3. If the book trying to be deleted does not exist an error will be shown |

|  |  |
| --- | --- |
| Use Case: Manipulate Users ( 1. Librarians) | |
| Actors: | Administrator |
| Precondition: | Administrator is logged in and on the admin page |
| Post-condition: | Administrator is still logged in and on the admin page |
| Main (happy) Path: | 1. Admin wishing to add a librarian fills out the empty spaces with librarian information 2. Admin clicks the “Sign up” button 3. A “Successful” message is displayed, librarian has been created 4. Admin wishing to delete a librarian clicks the “Delete Librarian” button 5. Admin is redirected to the “Delete Librarian” page 6. Username of the librarian being deleted is entered 7. Admin can click “Search” to see if that librarian exists (Optional) or click “Delete” to proceed 8. A “Successful” message is displayed and the admin is returned to the admin home page |
| Alternative Path: | 1. Conflict of username can occur during the addition of a librarian thus an error will be displayed 2. Admin cannot click “Sign Up” if all the fields are empty, doing so will prompt an error message 3. If the librarian trying to be deleted does not exist an error will be shown |
| Use Case: Manipulate Users (2. Members) | |
| Actors: | Librarian |
| Precondition: | Librarian is logged in and on the librarian page |
| Post-condition: | Librarian is still logged in and on the librarian page |
| Main (happy) Path: | 1. Librarian wishing to add a member clicks the “Add Member” button 2. Librarian is directed to the “Sign Up” page 3. He fills out the empty spaces with member information 4. Then clicks the “Sign Up” button 5. A “Successful” message is displayed, member has been created 6. Librarian is redirected to the librarian home page 7. Librarian wishing to edit or delete a member types the name of the member 8. Then clicks the “Edit Member” button 9. Librarian is redirected to the “Account” page 10. Any desired information change is written followed by clicking the “Edit” button 11. Or while on the account page the librarian can instead click delete to erase that member 12. After a “Successful” message, librarian is returned to the librarian page |
| Alternative Path: | 1. Conflict of username can occur during the addition of a member thus an error will be displayed 2. Librarian cannot click “Sign Up” if all the fields are empty, doing so will prompt an error message 3. If the member trying to be deleted or edited does not exist an error will be shown |