CSE308
1st Deliverable Assignment
Date: Feb 21st, 2016
Team Name: Birch (Silly Bee)
Team members:
Zhi cao 108816413 Class ID : 8
Yongbin Chen 108983991 Class ID:30
Jintian Chen 109222754 Class ID: 36
Zewei Zhang 108905249 Class ID: 39

List of user roles:
Role:

1. Guest User: needs to log in, cannot borrow or buy. Guest user is able to search, view samples and share books.
2. Member: is required to log in, can borrow, buy, search, view samples, comments, share and rate, send messages and receive messages. Member has its own member profile page.
3. Administrator: is required to log in, can manage all the members, books and publishers and Faculty. Administrator should handle **all the Faculty and Publisher Signup application.** Administrator can send and receive messages with all faculties and publishers. Administrator has its own administrator profile page.
4. Faculty: is required to log in, can manage all the books, such as the number of authorizations or licences, send and receive messages. Faculty should handle **all the publishers' request, such as edit book and add book applications.** Faculty has its own Faculty profile page. Faculty is a kind of mini administrator, but with limit power. Guest user also needs administrator's permission to sign up as a faculty.
5. Publisher: is required to log in, can search all the books that published by them, send edit book and add/delete book applications to Faculty. Publisher can send and receive messages. Publisher has its own Publisher profile page. Guest user also needs administrator's permission to sign up as a publisher.

The layout and functionality of each role profile page is different.

List of user cases:
Any users:

1. Simple search: All users can type some keywords, such as the book title, book author, book series and book publisher and click the search button. The system will search and return a list of books related to those keywords.
2. Advanced search: All users can type the book title, publisher and ISBN, specify the specific subject, format, languages and so on. All users are able to click the advanced

search button. The system will search and return a list of books related to all the specific information provided.

3. View a set of books written by a specified author: All users could click a marked author name, then the system will switch to another page to represent a set of books written by that author.
4. View a set of books according to a specified publisher: All users could select a marked publisher name, then the system will switch to another page to display a set of books published by that publisher.
5. View a set of books according to a category: All users can select the category, then the system will switch to another page to represent a set of books belong to the specified category.
6. View a set of books according to series: All user can select a marked series name, then the system will switch to another page to present a set of books belong to the specified series.
7. Sort by : All users can click the sort by button, choose the sort by terms and system will change the order of book list, by titles, date, popular rate and so on.
8. Filter the search result: filter the set of searching results. All users can specify the format, subject, publisher, language and so on, then the system will return a set of all the books related to all the specific information provided.
9. *Select the view of the list: All users can select the view of the list, could view by brief book information or book cover.
10. Help inquiry: send a help message to the admin, faculties or publishers, view the library lending policies.
11. View sample: pop out a small window with a few pages of selected book downloading from the second server, which is the actual book server.
12. Readability setting: When the user is reading the sample, the users can click the readability setting button so that the user can set the text scale or background lighting.
13. Share a book: pop out a small window where the user can share the book via Facebook, Twitter,Pinterest and email.
14. *View detail information of a book: All user click the icon of the book, then system will switch to the book information page. All users could view the information about the book in the book information page (such as the description, number of books left in the library, comments).
15. Donate: When anyone clicks the “Donate” button, a pop out window will appear such that the person can enter the amount of money he/she wants to donate.
16. Sign in: Enter the Barcode or Username, then enter the password and click the sign in button. The system will go to specified user profile page, for example, member will switch to member profile page, administrator will switch to administrator profile page and so on.

17. View more: click the “View more” link, then the system will expand the current page and load more books.
18. Sign up as member: All users click the signup button and then go to the signup page. All users can fill up the information, select “sign up as member” and become a member.
19. Sign up as faculty: All users click the signup button and then go to the signup page. All users can fill up the information, select “sign up as faculty” and send a faculty application to the administrator.
20. Sign up as publisher: All users click the signup button and then go to the signup page. All users can fill up the information, select “sign up as publisher” and send a publisher application to the administrator.
21. Reset password: User click “reset password” on the login window to change password.

Member:

1. Add to wish list: Signed Members can click the add to wish list button, then system will add the book to the wish list associated with the member’s account.If the user does not sign in, then the system will switch to the sign in page.
2. Borrow a book : Signed Members can click the borrow a book with one click button, then system will directly borrow the book for a default time using the primary mailing address and credit card associated with member’s account. If the user does not sign in, then the system will switch to the sign in page.
3. Rate a book: Signed Members can choose the number of stars given to the book, then the system will store the number of stars into the total rates associated with the book. If the user does not sign in, then the system will switch to the sign in page.
4. Comment: Signed Members could comment a book, then the system will store the comment into the total comments associated with book, the comment will be shown in the book information page. If the users does not sign in, then the system will switch to the sign in page.
5. Read a book: Signed members click on download button, then the book file will be transfer from the library server to the users’ devices. If the users does not sign in, then the system will switch to the sign in page.
6. View borrow history: All signed member click the view history button, the system will represent all the books that are on lean or returned by that member.
7. Return book: All signed member click the return button of the specified book, then the system will mark the book as returned and denied user’s access to that book. The system will also return the book license back to the library.
8. Place a hold (waiting for return):When viewing a book, the signed member can place a hold on the book if the book is currently on loan. The signed member will be added to the hold list associated with the book.

9. Add to Checkout list: The signed member can click the add to Checkout List button, then the book will be added to the Checkout List, which is a list of books associated with the user account. If the users does not sign in, then the system will switch to the sign in page.
10. Change the number of a book in Checkout List: The signed member can change the number of a book in the Checkout List.
11. Delete a book from the Checkout List: The signed member can delete books from the Check List.
12. Check out: The signed member can borrow all books in the Checkout List, then the system will remove the each license of every book and make all books available for the members. All books will be shown as on loan condition in the borrow history list associated with member account.If the users does not sign in, then the system will switch to the sign in page.
13. Add a credit card: Signed members can enter their credit card information into the database and connect with their accounts.
14. Delete credit card: Signed members can disconnect credit card information with their accounts and delete their credit card information from the database .
15. Balance: Signed members can view the balance they owe by clicking the balance button. If the users does not sign in, then the system will switch to the sign in page.
16. Edit profile:Signed members can edit their profile such as changing the email and phone number by clicking the edit profile button.
17. View Payment information: Signed members can review all the credit cards associated with their accounts.
18. Update credit card information: Singed members can edit and update credit cards information associated with their accounts.
19. Sign out:Back to guest user by clicking the “Log out” button.

Admin & Faculty:

1. Add a book:The faculty or admin can add a new book to the system.
2. Delete a book : The faculty or admin can delete a book from the system.
3. Delete comments : The faculty is granted the authority to delete any member’s comments.
4. Delete membership: The faculty can search the specific member by typing the the member’s ID, and then delete membership of the member .
5. Reply help inquiry :The faculty can click the help inquiry button to view and reply the inquiry messages sent by the member.
6. Accept special applications (Admin only): Administrators can approve the faculty or publisher applications. Then the system will store the information from the applications into the database.

7. Generate bestseller lists: Signed admin can generate a ranked list contains books with the maximum amount of borrow times.
8. request books' licenses: Administrators and faculties can add licenses of a specific book.

Admin, Faculty, Publisher:

9. View messages: All signed faculties, publishers and administrators can click the messages button, the system will represent all the messages received by those people.
10. Delete messages: All signed faculties, publishers and administrators can delete messages they sent or received.
11. Compose and send messages: All signed faculties, publishers and administrators can compose and send messages to each other.
12. Save as draft: All signed faculties, publishers and administrators can save messages as draft.

Publisher:

1. Edit books' information application: Publishers can edit and send editing book, which is only published by themselves, applications to the faculty.
2. Send Add book applications: Publishers can send book application to the faculty in order to add their books to the library.

All complete use cases:

1.

Use-case:	Simple search for books
Primary actor:	All users
Goal in context:	Display a set of all the books that match the user's search terms.
Preconditions:	The user can see the "Search" button and the text field at the top of the page.
Trigger:	Click "Search" button.
Scenario:	1. Enter the "Title", "Author", "ISBN", or keyword in the search text field. 2. Click the "Search" button. 3. Observe all the records in the database that match the given search terms and filter condition.

	4. Jump to the book display page and display the set of book.
Exceptions:	1. There are no records that match the given search term (the message 'No matching records could be found' will be displayed below the search fields).
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1.Where on the web interface will the search fields and buttons be displayed? 2.Should we provide a set of buttons such as keyword, author or title so that the system know how to efficiently query in the database?

2.

Use-case:	Advanced search for books
Primary actor:	All users
Goal in context:	Display a set of all the books that match the user's search terms and criteria.
Preconditions:	The user has entered into the advanced search page.
Trigger:	Click "Advance search" button.
Scenario:	1. Observe advanced search page. 2. Enter the "Title", "Author" or "ISBN". 3. Select "search by date", "search by subjects", "search by languages", "search by publishers", or "search by available" button. 4. Click the "Search" button. 5. Observe all the records in the database that match the given search terms and filter condition.

	6. Jump to the book display page and display the set of book.
Exceptions:	1. "ISBN" is provided: if the ISBN is not provided in the correct format, and error message is displayed that contains the correct format. 2. There are no records that match the given search terms and criteria (the message 'No matching records could be found' will be displayed below the search fields): Users enters different terms and click the 'Search' button
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1.Where on the web interface will the search fields and buttons be displayed? 2. What other criteria will the users want to search by? 3. Should we have a 'Clear Fields' button that clears all entered text in the search fields?

3.

Use-case:	View a set of books written by a specified author
Primary actor:	All users
Goal in context:	Display a set of all the books written by a specific author.
Preconditions:	User can see the marked author name under the book' name when in the the book information page.
Trigger:	Click marked author name link.
Scenario:	1. Observe the book information page. 2. Click the author link for that book. 3. Observe all the records in the database that matches the given

	author. 4. Jump to another page and display all the books written by the author.
Exceptions:	None
Priority:	Important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1.Where on the web interface will the author link be displayed?

4.

Use-case:	View a set of books according to a specified publisher.
Primary actor:	All users
Goal in context:	Display a set of all the books published by a specific publisher.
Preconditions:	User can see the marked publishers in the the book information page.
Trigger:	Click one of the marked publisher name link.
Scenario:	1. Observe the book information page. 2. Click the publisher link belong to that book. 3. Observe all the records in the database that matches the given publisher. 4. Jump to another page and display all the books published by the publisher.
Exceptions:	None.
Priority:	Important, need to be implemented soon
When available:	Second increment

Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1.Where on the web interface will the publisher link be displayed?

5.

Use-case:	View a set of books according to a category
Primary actor:	All users
Goal in context:	Display a set of all the books that match the category provided by the menu.
Preconditions:	The user can see the “Category” button on top of the page.
Trigger:	Click one of the category link.
Scenario:	1. Click the category tab and and a set of categories’ links are displayed under that category. 2. Click the one of the category link. 3. Observe all the records in the database that matches the given publisher. 4. Jump to another page and display all the books belongs to that category.
Exceptions:	None.
Priority:	Essential, must be implemented
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface

Open issues:	1.Where on the web interface will the menu link be displayed? 2.What kind of and how many categories should be provided for the user?

6.

Use-case:	View a set of books according to series.
Primary actor:	All users
Goal in context:	Display a set of all the books belong to a specific series.
Preconditions:	User can see the marked series provided in the the book information page.
Trigger:	Click the series link.
Scenario:	1. Observe the book information page. 2. Click the series link for that book. 3. Observe all the records in the database that matches the given series. 4. Jump to another page and display all the books belong to that series.
Exceptions:	None.
Priority:	Important, needed to be implemented soon.
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1.Where on the web interface will the series link be displayed?

7.

User-case:	Sort by
Primary actors:	All users

Goal in context:	All users can sort the search results according to the specific terms, such as change the order of books by titles, date, popular rate and so on.
Preconditions:	1. The users can see the searching results page. 2. The users can see the “Sort by” button and all the sort by terms
Trigger:	The users click the “Sort by” button and select the specific terms
Scenario:	1. Observe the search result page 2. Click the “Sort by” button and select the specific terms 3. The system will sort searched books according to the specific terms. 4. The system will refresh the page and represent all the searched books with the chosen order.
Exceptions:	None.
Priority:	Important, should be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actors:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will the sort by buttons be displaced? 2. What other terms will the users want to sort by?

8.

User-case:	Filter the search result
Primary Actors:	All users
Goal in Context:	Filter the set of searching results. All users can specify the format, subject, publisher, language and so on, then the system will be return a set of all the books related to all the specific information provided.
Precondition:	1. The users can see the searching result page 2. The users can see the filter section and all the filter criteria.

Trigger:	The user click the filter section and select filter criteria.
Scenario:	1. Click the filter section and select filter criteria. 2. The System will refresh the page and return a set books from the searching results that match the specific criteria. 3. Observe the searching result page.
Exceptions	None.
Priority:	Important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will the filter section and all the filter criterias be displaced? 2. What other criterias will the users want to filt?

9.

User-case:	Select the view of the list
Primary Actors:	All users
Goal in Context:	All users can select the view of the list and view the searching result by cover or list.
Precondition:	1. The users can see the searching results page. 2. The users can see the “cover” and “list” button
Trigger:	The user click the “cover” or “list” button
Scenario:	1. Click the “cover” or “list” button 2. If the user clicks the “cover” button, then system will refresh the page and return a set of books represented by a set of book covers 3. If the user clicks the “list” button, then system will refresh the

	page and return a list of books represented by a book list
Exceptions	None.
Priority:	Important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will “cover” and “list” buttons be displaced? 2. What other kind of format will the users want to represent?

10.

User-case:	Help inquiry
Primary Actors:	All users
Goal in Context:	Send a help message to the faculties and view library lending policies.
Precondition:	1. The users see the help page. 2. The users can see all the faculties 3. The users can see question category selections. 4. The users can see the text input sections and “submit” button
Trigger:	The user click “help” button
Scenario:	1. Users select the a faculty they want to contact. 2. Users select the category that best fits their questions 3. Users fill out all the text input sections, such as email address, subject, description. 4. Users click the “submit” button, then the system will send a help inquiry message to the receiver.
Exceptions	1. The receiver is not selected 2. Input format is incorrect, such as empty or invalid categories 3. The question category is not selected

Priority:	Important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server, Faculty
Channels to secondary actors:	Server: network and local interface Faculty: web browser interface, program modification
Open issues:	1. Where on the web interface will text input sections and buttons be displaced? 2. What other text input sections will the users want to enter? 3. What other selection will the users want to select?

11.

User-case:	View sample
Primary Actors:	All users
Goal in Context:	Pop up a small window with a few pages of selected book
Precondition:	1. The users can see the book image 2. The users can see the “sample” button
Trigger:	The user clicks “sample” button
Scenario:	1. The users view the first sample pages of the specific book 2. The users click the next page button, then the page will be refreshed and the next sample page will be represented. 3. The users click the previous page button, then the page will be refreshed and the previous sample page will be represented. 4. The users can scroll down and up of the current sample pages
Exceptions	1. The user has reached the first sample page, and the previous page button will not work. 2. The user has reached the last sample page, and the next page button will not work.
Priority:	important, need be implemented soon
When available:	Second increment

Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server(library website server), Second Server (actual book server)
Channels to secondary actors:	Server: network and local interface Second Server: loading the actual book into the library website server, so that user can view the sample pages of that book.
Open issues:	1. What is the prefered size for the pop out window 2. Where on the web interface will the next page and previous page, scrolling bar be displayed

12.

User-case:	Readability Setting
Primary Actors:	All users
Goal in Context:	When the user is reading the sample, the user can click the readability button, so that the user can set the text scale or background lighting
Precondition:	1. The users is reading the sample page 2. The users can see the “readability setting” button.
Trigger:	The user click the “readability setting” button.
Scenario:	1. The users click the “readability setting” button 2. The user can see the text scale and background lighting selection buttons. 3. The users can select the text scale, then the system will change and refresh the size of text sample page. 4. The users can select the background lighting, then the system will change and refresh the lightness of the text sample page.
Exceptions	None.
Priority:	Important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server

Channels to secondary actors:	Server: network and local interface
Open issues:	1. What is the prefered size for the pop out window 2. Where on the web interface will the next page and previous page be displayed

13.

Use-case:	Share a book
Primary actor:	All users
Goal in context:	Share the book information via Facebook, Twitter,Pinterest and email.
Preconditions:	The user has an account on the social network site.
Trigger:	User clicks the “Share Book” button.
Scenario:	1. The user clicks the “Share Book” button. 2. A window pops up with list of social network site. 3. The user selects which site to share the book information. 4. A login window pops up. 5. The user enters the username and password of social network site, and clicks login. 6. The user clicks “confirm” to share the book.
Exceptions:	1. Fail to log into the social network site
Priority:	Optional.
When available:	Third increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface
Secondary actor:	Social network site.
Channels to secondary actors:	Server: network and local interface.
Open issues:	1. Where on the web interface will the “ Share Book” button be displayed? 2. What other social network site the user want to share book

	information.

14.

Use-case:	View detail information of a book
Primary actor:	All users
Goal in context:	Users can view the detail information of a book.
Preconditions:	Users can see title or cover of the book.
Trigger:	Users click the title or cover of the book.
Scenario:	1. Users click the title or cover of the book. 2. A new page pops up and display the detail information of the book, which is gotten from the library system. 3. Users observe the page.
Exceptions:	None
Priority:	Essential, must be implement.
When available:	First increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface
Secondary actor:	Server.
Channels to secondary actors:	Server: network and local interface.
Open issues:	1. Where on the web interface will book cover and title be displayed?

15.

Use-case:	Donate
Primary actor:	All users

Goal in context:	Donate money to the online library.
Preconditions:	1. Users have a bank account to perform online payment.
Trigger:	Users click the “Donate” button.
Scenario:	1. Users click the “Donate” button. 2. A window will pop up with input field to processing donation. 3. Users observe the window fill in the amount of money, credit card or debit card number, security code, card-holder name and billing address. 4. Users click “Donate” button on the window.
Exceptions:	1. The bank rejects the transaction.
Priority:	Optional.
When available:	Third increment
Frequency of use:	A few times each day.
Channel to actor:	Via web browser interface
Secondary actor:	Bank server, Server
Channels to secondary actors:	Server: network and local interface. Bank Server: Transfer money according to the credit card information.
Open issues:	1. Where on the web interface will “Donate” button be displayed? 2. What other kinds of credit card the user owned except the library accepting?

16.

Use-case:	Sign in
Primary actor:	Guest User
Goal in context:	Sign in into the website as a member or faculty or administrator or publisher that matches the Username and the password.
Preconditions:	The actor has entered into the home page and is available to see “sign in” button.
Trigger:	The actor clicks the “sign in” button.

Scenario:	1. Observe login popup page 2. Enter username and password. 3. Click “sign in” button. 4. The system fetches username and password from the database to check the validation. 5. The system returns the information associated with the username and password if the username and password are matched. 6. If the username and password are valid, the personal information page will be displayed.
Exceptions:	1. “Sign in” button is selected: if the username is not provided in the correct format, and error message is displayed that contains the correct format. 2. There is no record that match the given username and password (the message “Your username or password was incorrect. Please try again” will be displayed above the text fields): User enters different search terms and clicks the ‘sign in’ button.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will the username fields and password fields and “sign in” button be displayed? 2. Whether other safety question should be added in order to protect the user account?

17.

Use-case:	View more
Primary actor:	All users
Goal in context:	To load more books and their brief information into book list in current page.

Preconditions:	1. Users are at the book list page. 2. There are more books matching the filler condition.
Trigger:	The users click the “View more” link.
Scenario:	1. Users scroll to the bottom of the book list to find the “View more” link. 2. Users click the “View more” link. 3. The page expand, and more books are load into current page.
Exceptions:	None
Priority:	Essential, must be implemented
When available:	Second increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will the “View more” link be displayed?

18.

Use-case:	Sign up as members
Primary actor:	Guest User
Goal in context:	Create a new member account in the library system.
Preconditions:	The actor has entered into the home page and is available to see the “sign up” button on the top menu bar.
Trigger:	The Guest User clicks the “Sign up” button.
Scenario:	1. The Guest User clicks the “Sign up” button. 1. Observe sign up page. 2. Select “Sign up as member”. 3. Enter username in the “username” text field. 4. Enter email in the “email” text field.

	5. Enter email again in the “email again” text field. 6. Enter password in the “password” text field. 7. Enter password again in the “password again” text field. 8. Click “sign up” button. 9. The system stores the entered information into the database. 10. The message “Sign up successfully” will be displayed.
Exceptions:	1. Input format incorrect. 2. Input text is not same as the second time input. 3. Username has already existed. 4. Email has already been used.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface
Secondary actor:	Server, Admin
Channels to secondary actors:	Server: network and local interface. Admin: Receive the application, accept or decline the application.
Open issues:	1.where on the web interface will the username fields, email fields, password fields, email again fields, password fields again fields and “sign up” button be displayed? 2. What other personal information will the guest want to enter?

19.

Use-case:	Sign up as faculty
Primary actor:	Guest User
Goal in context:	Create a new faculty account in the library system.
Preconditions:	The actor has entered into the home page and is available to see the “sign up” button on the top menu bar.
Trigger:	The Guest User clicks the “Sign up” button.
Scenario:	1. The Guest User clicks the “Sign up” button.

	2. Observe signup page. 3. Select “Sign up as faculty”. 4. Enter first name in the “first name” text field. 5. Enter last name in the “ last name “ text field. 6. Enter the social security number in the “ SSN#” text field. 7. Enter username in the “username” text field. 8. Enter email in the “email” text field. 9. Enter email again in the “email again” text field. 10. Enter password in the “password” text field. 11. Enter password again in the “password again” text field. 11. Click “sign up” button. 12. A new page pops up and display message “Your application has been processed, we will send message via email to notify your the application status.”.
Exceptions:	1. Input format incorrect. 2. Input text is not same as the second time input. 3. Username has already existed. 4. Email has already been used. 5. Social security number doesn’t refer to a person.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface
Secondary actor:	Server, Admin
Channels to secondary actors:	Server: network and local interface. Admin: Receive the application, accept or decline the application
Open issues:	1. Where on the web interface will the first name, last name, username fields, email fields, password fields, email again fields, password fields again fields and “sign up” button be displayed? 2. What other personal information will the applicant want to enter?

20.

Use-case:	Sign up as publisher
Primary actor:	Guest User

Goal in context:	Create a new publisher account in the library system.
Preconditions:	The actor has entered into the home page and is available to see the “sign up” button on the top menu bar.
Trigger:	The Guest User clicks the “Sign up” button.
Scenario:	1. The Guest User clicks the “Sign up” button. 2. Observe signup page. 3. Select “Sign up as publisher”. 4. Enter company name in the “company name” text field. 5. Enter company address in the “company address” text field. 6. Enter company phone in the “company phone” text field. 7. Enter username in the “username” text field. 8. Enter company email in the “company email” text field. 9. Enter email again in the “email again” text field. 10. Enter password in the “password” text field. 11. Enter password again in the “password again” text field. 13. Click “sign up” button. 14. A new page pops up and display message “Your application has been processed, we will send message via email to notify your the application status.”.
Exceptions:	1. Input format incorrect. 2. Input text is not same as the second time input. 3. Username has already existed. 4. Email has already been used. 5. No such company.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface
Secondary actor:	Server, Admin
Channels to secondary actors:	Server: network and local interface. Admin: Receive the application, accept or decline the application.
Open issues:	1. Where on the web interface will the company name, company address, company phone, username fields, email fields, password fields, email again fields, password fields again fields and “sign up” button be displayed?

	2. What other personal information will the applicant want to enter?

21.

Use-case:	Reset password
Primary actor:	Guest User
Goal in context:	Reset the password of an account.
Preconditions:	Users are able to see and click “Login” button.
Trigger:	The Guest User clicks the “Reset password” link.
Scenario:	1. Users click “Login” button. 2. Login window pops up. 3. Users click “Reset password” link. 4. Reset password window pops up. 5. Users enter the username in the “Username” text field. 6. Users click “Send verification code” button. 7. Users check their email to get the verification code. 8. Users enter the verification code in the “Verification code” text field. 9. Users click “Ok” button. 10. New password window pops up. 11. Users enter password in the “ New password” text field. 12. Users click “Confirm” button. 13. Login window pops up.
Exceptions:	1. The username doesn’t exist in the library system. 2. The verification code doesn’t match.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Once a day for each account.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface.

Open issues:	1. What other method to verify ownership of users? 2. What other personal information should provide to reset password?

Members

1.

Use-case:	Add to wish list
Primary actor:	Member
Goal in context:	Add the book that member is going to or want to borrow in the future into the wish list.
Preconditions:	The actor has entered the book information page and the “Add to wish list” button is enabled.
Trigger:	The Member clicks the “Add to wish list” button.
Scenario:	1. The members clicks the “Add to wish list” button 2. A confirmation window will pop up with successful message “This book has been added to wish list successfully”
Exceptions:	None
Priority:	Important, need to be implemented soon
When available:	Second increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will the “Add to wish list” button be displayed?

2.

Use-case:	Borrow a book

Primary actor:	Member
Goal in context:	Let members be able to make a borrowing book request
Preconditions:	1. The actor has entered the book information page and the “Borrow” button is enabled. 2. The actor has enough balance to borrow this book.
Trigger:	The Member clicks the “Borrow” button.
Scenario:	1. The Member clicks the “Borrow” button. 2. The server will search this user in back end, and check his balance information. 3. The actor will be redirected to “check out” page.
Exceptions:	The member does not have enough balance to borrow current book. A pop-up window will be displayed with message “Sorry, your balance is not enough to borrow this book”.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will the “Add to wish list” button be displayed?

3.

Use-case:	Rate a book
Primary actor:	Member
Goal in context:	Members can rate the book by selecting the amount of stars from 1 to 5.
Preconditions:	1. The actor has entered the book information page. 2. The actor borrowed this book before.
Trigger:	The Member clicks the “Write a customer review” button.

Scenario:	1. The Member clicks the “Write a customer review” button, then server checks his borrowing history. He will be directed to rating page if this is a valid request. 2. The user use mouse to hover on the star and click to give a star rating to the book. 3. The user can write review in the text box, and click “submit” button to submit his review. (optional)
Exceptions:	The member has never borrowed this book before. A pop-up window will be displayed with message “Sorry, you cannot rate this book”.
Priority:	Important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will the “write a customer review” button be displayed? 2. How to implement the dynamic effect of stars 3. How many character limited should we set for comment.

4.

Use-case:	Comment
Primary actor:	Member
Goal in context:	Members can leave comment about the book.
Preconditions:	1. The actor has entered the rating book page. 2. The actor borrowed this book before.
Trigger:	The Member clicks the Star then a comment text box will be displayed.
Scenario:	1. The Member clicks the Star then a comment text box will be displayed. 2. The user leave the comment inside text box. 3. The user click submit button.

Exceptions:	The member has input over the maximum number of characters.
Priority:	Important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. How many character limited should we set for comment? 2. How large the comment box should be?

5.

Use-case:	Read a book
Primary actor:	Member
Goal in context:	Members can Read a book.
Preconditions:	1. The actor has entered book detail page 2. The actor is borrowing this book 3. The actor see “Read” button.
Trigger:	The Member clicks “Read” button
Scenario:	1. The member clicks “Read” button and a new window come out. 2. The users view the first pages of the specific book 3. The users click the next page button, then the page will be refreshed and the next page will be represented. 4. The users click the previous page button, then the page will be refreshed and the previous page will be represented. 5. The users can scroll down and up of the current sample pages
Exceptions:	1. The user has reached the first page, and the previous page button will not work. 2. The user has reached the last page, and the next page button will not work.
Priority:	Essential, must be implemented

When available:	First increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server(library Server), Second Server(actual book server)
Channels to secondary actors:	Server: network and local interface Second Server: loading the actual book into the library website server, so that user can view the sample pages of that book.
Open issues:	1. What is the prefered size for book page 2. Where on the web interface will the next page and previous page, scrolling bar be displayed

6.

Use-case:	View borrow history
Primary actor:	Member
Goal in context:	Members can view their borrowing history, which should display a list of books that they have borrowed and in loan.
Preconditions:	1. The member should be in their personal profile page.
Trigger:	The Member clicks the “borrow history” tab on the left panel.
Scenario:	1. The server should retrieve this person’s borrowing history and return the search results. 2. The borrow history should be displayed.
Exceptions:	The member has never borrowed any book before. Then the right panel should display a message “You have not borrowed any book yet”.
Priority:	Important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface

Open issues:	1. How is the list should look like? 2. What functional buttons should we place after each list item.

7.

Use-case:	Return a book
Primary actor:	Member
Goal in context:	Members have obligation to return the book before the deadline.
Preconditions:	1. The member should be in the borrow history page. 2. The return button is enabled.
Trigger:	The Member clicks the “return” button.
Scenario:	1. The server add one license on the returning book. 2. The return should become disabled.
Exceptions:	None
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server, Second Server(actual book server)
Channels to secondary actors:	Server: network and local interface Second Server: The downloading will be returned back to the second server.
Open issues:	None

8.

Use-case:	Place a hold
Primary actor:	Member
Goal in context:	The member is able to place a hold on the book when it is temporarily in loan.

Preconditions:	1. The member should be in the book information page. 2. The book is temporarily not available to be borrowed.
Trigger:	The Member clicks the “Place hold” tab on the left panel.
Scenario:	1. The server should add the person to waiting list. 2. A pop-up window will be displayed with a successful message “You are now in the waiting list”.
Exceptions:	The member has already placed a hold on this book. A pop-up window will be displayed with a warning message “You have already placed a hold”.
Priority:	Essential, must be implemented.
When available:	First increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. How do we manage waiting list table on back end.

9.

Use-case:	Add to Checkout list.
Primary actor:	Member
Goal in context:	Add the book to the checkout list that the member wants to borrow.
Preconditions:	1. The actor has been authenticated and identified as a member. 2. The actor has entered the book information page and the “Add to checkout list” button is enabled.
Trigger:	The Member clicks the “Add to checkout list” button.
Scenario:	1. The members clicks the “Add to checkout list” button associated with a book.. 2. A confirmation window will pop up with successful message that “This book has been added to checkout list successfully”.

Exceptions:	The user is not logged in, so it will jump to log in page.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1.Where on the web interface will the “Add to Checkout list”button be displayed?

10.

Use-case:	Change the number of a book in Checkout List.
Primary actor:	Member
Goal in context:	Change the number of a book in the Checkout List.
Preconditions:	1. The actor has login as members. 2. The actor has entered the checkout list page.
Trigger:	1. The member type the number of book he/she wants to borrow for a specific book in the corresponding text field. 2. Click the “Update All” button at the bottom of the page.
Scenario:	1. Observe the checkout list page. 2. Type in the number of book she/he wants to change for a specific book in the corresponding text field. 3. Click the “Update All” button. 4. Display the updated information to the member.
Exceptions:	If the number of book is negative, an error message will be displayed.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times per day.

Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will the “Update All”button be displayed? 2. Should we provide a separate update button for each of the books?

11.

Use-case:	Delete a book in Checkout List.
Primary actor:	Member
Goal in context:	Delete a book in the Checkout List.
Preconditions:	The actor has entered the checkout list page.
Trigger:	Click the “Delete” button next to the book the member wants to delete from the checkout list.
Scenario:	1. Observe the checkout list page. 2. Click the “Delete” button corresponding to a specific book. 3. Refresh the page and display the updated information.
Exceptions:	None
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will the “Delete”button be displayed? 2. Should we provide a “Delete All” button? 3. Do we really need the “Delete” button? We can just change the

	number of the book to 0 instead.

12.

Use-case:	Check out
Primary actor:	Member
Goal in context:	Check out all books in the checkout list.
Preconditions:	1. The actor has been authenticated and identified as a member. 2. The actor has entered the checkout list page.
Trigger:	Click the “Check out” button at the bottom of the page.
Scenario:	1. Observe the checkout list page. 2. Click the “Check out” button. 3. The database receive the information and update all information about the book and the member such as the licenses of each book, the status of the book (on loan or not), the borrowed books of a member, and so on. 4. Users select credit cards stored in their payment list and check out. 5. Refresh the page and no book is in the checkout list now.
Exceptions:	1. User does not meet the requirement, for instance, user has borrowed more than limited number of books. 2. The number of licenses associated the corresponding book is not enough. 3. If the user does not select a credit card, check out cannot be completed. 4. If the user does not have a credit card, then will switch to the **add credit card user case** and follow its procedures. 5. If the user does not have enough money to pay for the check out, the bank server will notify the library server, then the check out cannot be completed.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server, bank server

Channels to secondary actors:	Server: network and local interface bank server: transfer money according to the credit card information, check the completeness of the transaction.
Open issues:	1. Where on the web interface will the “Check out”button be displayed?

13.

Use-case:	Add a credit card
Primary actor:	Member
Goal in context:	Add a credit card information to a member.
Preconditions:	1. The actor has been authenticated and identified as a member. 2. The actor has entered the account page and in the payment panel.
Trigger:	Click the “Add a Credit Card” button.
Scenario:	1. Observe the payment page. 2. Click the “Add a Credit Card” button. 3. Enter all information about the credit card. 4. Click the “Submit” button. 5. The system will connect to the credit card server to check if the credit information is valid or not. 6. If success, the system will store the credit card information into the database associated with the member.
Exceptions:	The credit card information the member entered is invalid, so the current page will pop up a window to alert the member that “The credit card information you entered is invalid”.
Priority:	Important, need be implemented soon
When available:	Second increment.
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	library server, credit card server

Channels to secondary actors:	Library server: network and local interface. Credit card server: network and database.
Open issues:	1. Where on the web interface will the credit card input text fields be displayed?

14.

Use-case:	Delete a credit card
Primary actor:	Member
Goal in context:	Delete a credit card associated with a member.
Preconditions:	1. The actor has been authenticated and identified as a member. 2. The actor has entered the account page and in the payment panel.
Trigger:	Click the “Delete” button associated with a credit card.
Scenario:	1. Observe the payment page. 2. Click the “Delete” button associated with the credit card that the member wants to delete. 3. The system will remove the credit card information associated with the member in the database. 4. Current page will pop up a window that says “The credit card has been successfully deleted”.
Exceptions:	None
Priority:	Important, need be implemented soon
When available:	Second increment.
Frequency of use:	A few times per day.
Channel to actor:	Via web browser interface
Secondary actor:	library server, credit card server
Channels to secondary actors:	Library server: network and local interface. Credit card server: network and database.
Open issues:	1. Where on the web interface will the “Delete” button be displayed?

15.

User-case:	View balance
Primary Actors:	Member
Goal in Context:	Signed members can view the balance they owed.
Precondition:	1. The users has logged into their account page 2. The users can see the “Balance” button
Trigger:	The user click “Balance” button
Scenario:	1. The users click “Balance” button. 2. The system will check the database and search for the balance information associated with the member. 3. The system will refresh and display the balance that the member have in the current page.
Exceptions	None.
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Should we send an email to the member to notify the member that he/she owes money to the library.

16.

User-case:	Edit profile
Primary Actors:	Member
Goal in Context:	Signed members can edit their profile such as changing the email and phone number.

Precondition:	1. The users has logged into their account page. 2. The users can see the “Edit profile” button.
Trigger:	The user click “Edit profile” button.
Scenario:	1. The users click “Edit profile” button. 2. The page will refresh, and the member information such as email ,password,and phone will become editable. 3. The member types the information they want to change in the corresponding text field. 4. Click the “Save” button. 5. The database will update the user profile information. 6. The page will refresh, and the updated user information is displayed.
Exceptions	1. The email format is incorrect such as not containing the character “@”. 2. When changing the password, the input field for the confirm password does not match the password the member enter at the first time.
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1. Where on the web interface will the “Edit Profile” be displayed?

17.

User-case:	View payment information
Primary Actors:	Signed members

Goal in Context:	Signed members could view all the credit cards associated with their accounts
Precondition:	1. Signed members has logged into their account page. 2. Signed members can see the “Payment” button.
Trigger:	The user click “Payment” button
Scenario:	1. The users click “Payment” button, then the system will refresh and represent all the credit cards associated with user account in the current page. 2. The users can click the credit card, and the detailed information, such as first 4 digits of the card number, card type, expired date for that credit will be represent.
Exceptions	1. The credit card has expired and the system will mark that credit card as invalid. 2. The credit card was rejected in the last purchase, then the system will mark that credit card as invalid.
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	Where on the web interface will the credit card information be presented?

18.

User-case:	Update credit card information
Primary Actors:	Signed members
Goal in Context:	Signed members can edit and update credit cards information associated with their accounts.

Precondition:	1. Signed members has logged into their payment page. 2. Signed members can see all the credit cards associated with their accounts. 3. Signed members can see the “Edit” button of that credit card.
Trigger:	The user click “Edit” button
Scenario:	1. The users click “Edit” button, then the system will refresh and represent all the detailed information of this credit card. 2. The users can edit all the information, such as card number, expiration date, cardholder name and CVV number. 3. The users click the “save” button, and all the information will be updated and stored into the database, then the system will return back to the Payment page.
Exceptions	1. User's input format is invalid. 2. User’s credit card has expired, then the system will mark the card as invalid.
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	Where on the web interface will the “Edit” button be presented?

19.

User-case:	Sign out
Primary Actors:	Signed members, faculties, publishers and administrators.
Goal in Context:	Signed members can go back to guest user by clicking the “Log out” button

Precondition:	1. Signed members, faculties, publishers and administrators can see their account name on the navigation bar. 2. All Signed members, faculties, publishers and administrators can see the “Log out” button.
Trigger:	The user clicks “Log out” button
Scenario:	1. The user clicks “ Log out” button. 2. User lost authorization of the website. 3. The system will switch to the main website and the account name on the navigation bar will become “Sign in” and “Sign up” button.
Exceptions	None.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	Where on the web interface will the “Edit” button be presented?

Admin & Faculty:

1.

Use-case:	Add a book.
Primary actor:	Faculty,Admin
Goal in context:	The faculty can add a book to the library.
Preconditions:	1. The actor has been authenticated and identified as a faculty. 2. The actor has entered the account information page .
Trigger:	The faculty click the “Add a book” button.
Scenario:	1. The actor clicks the “Add books” button. 2. The page will refresh, and the faculty type the books’ title,

	author,description, series and ISBN in the corresponding text field. 3. The actor clicks the “Submit” button. 4. The system records the book information into the database.
Exceptions:	The ISBN is not provided in the correct format, and error message is displayed.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server, Faculty,Admin
Channels to secondary actors:	Server: network and local interface Faculty: web browser interface, program modification Admin: web browser interface, program modification
Open issues:	1.Where on the web interface will the “Add books” button be displayed?

2.

Use-case:	Delete a book
Primary actor:	Faculty, Admin
Goal in context:	The faculty and admin can delete a book from the library.
Preconditions:	1. The actor has been authenticated and identified as a faculty or admin.. 2. The actor has entered the account information page .
Trigger:	The actor clicks the “Delete a book” button.
Scenario:	1. The actor clicks the “Delete a book” button. 2. The page will refresh, and the faculty can type the book’s name he/she want to delete in the search text field. 3. The actor click the “Search” button. 4. The system queries the book name in the database, and loads all book’s information that meet the search criteria into the page. 5. The actor clicks the “Delete” button associated with the book

	he/she want to delete.
Exceptions:	The book name does not exist, so “No book is found” message is displayed.
Priority:	Essential, must be implemented
When available:	First increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server, Faculty,Admin
Channels to secondary actors:	Server: network and local interface Faculty: web browser interface, program modification Admin: web browser interface, program modification
Open issues:	1.Where on the web interface will the “Delete a book” button be displayed?

3.

Use-case:	Delete a comment.
Primary actor:	Faculty, Admin
Goal in context:	The faculty and admin can delete any comment.
Preconditions:	1. The actor has been authenticated and identified as a faculty or admin. 2. The actor has entered into the book information page . 3. The actor can see all comments associated with the book.
Trigger:	The actor clicks the “Delete” button associated with a comment that he/she wants to delete..
Scenario:	1. The actor clicks the “Delete” button associated with a comment. 2. The system deletes the comment records associated with the delete button in the database. 3. The page will refresh, and the comment disappears.
Exceptions:	None
Priority:	Important, need to implement soon

When available:	Second increment
Frequency of use:	A few times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server, Faculty, Admin
Channels to secondary actors:	Server: network and local interface Faculty: web browser interface, program modification Admin: web browser interface, program modification
Open issues:	1.Where on the web interface will the “Delete a comment” button be displayed?

4.

Use-case:	Delete membership
Primary actor:	Faculty, Admin
Goal in context:	The actor can delete any member’s membership.
Preconditions:	1. The actor has been authenticated and identified as a faculty or admin. 2. The actor has entered into their account page .
Trigger:	The actor clicks the “Delete” button associated with a member.
Scenario:	1. The actor clicks the “Manage User” tab. 2. The actor types a member ID in the search text field and clicks search. 3. The system queries the member ID in the database and display it to the actor. 4. The actor clicks the “Delete” button associated with the member.
Exceptions:	The member ID is not valid, the error message “Not found, invalid ID” is displayed.
Priority:	Important, need to implement soon
When available:	Second increment
Frequency of use:	A few times per day.
Channel to actor:	Via web browser interface

Secondary actor:	Server, Faculty, Admin
Channels to secondary actors:	Server: network and local interface Faculty: web browser interface, program modification Admin: web browser interface, program modification
Open issues:	1.Where on the web interface will the “Delete” button be displayed?

5.

Use-case:	View/reply help inquiry
Primary actor:	Faculty
Goal in context:	Reply help inquiry :The faculty can click the “Help inquiry” button to view or reply the help inquiry messages sent by the member.
Preconditions:	The faculty is at the message page The faculty can see the “Help inquiry” messages section
Trigger:	The faculty click the “Help inquiry” messages section.
Scenario:	1. Faculty observes the message center. 2. Faculty clicks the “Help inquiry” tab. 3. Faculty clicks single “Help inquiry” header. 4. Help inquiry detail page appears. 5. Faculty enters response message into response input text field. 6. Faculty clicks “Send”.
Exceptions:	None
Priority:	Essential, must be implemented
When available:	second increment
Frequency of use:	Many times per day.
Channel to actor:	Via web browser interface
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface

Open issues:	1.Where could the faculty find the “Help inquiry” tab? 2.Is the “Help inquiry” the same as general users’ message?

6.

User-case:	Accept/decline special applications
Primary Actors:	Admin
Goal in Context:	Administrators add new publisher or faulty to library system by accepting applications.
Precondition:	Admin is able to view the applications.
Trigger:	Admin clicks one of the application.
Scenario:	1. Admin observes the message center. 2. Admin clicks the application tab. 3. Admin clicks single application header. 4. Application detail page appears. 5. Admin clicks “approve/decline” button. 6. An email was sent to notify applicants.
Exceptions	1. The application have been approved. 2. The applicants are already in the library system.
Priority:	Essential, must be implemented.
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1.Where on the web interface will the “accept/decline” button be presented? 2.Should admin handle all the applications, or the faulty could also handle new publisher request?

7.

User-case:	Generate bestseller lists
Primary Actors:	Admin, Faculty
Goal in Context:	Administrators can view a list of bestseller by selecting certain time period.
Precondition:	Admin is able to see “Bestsellers” tab on admin panel.
Trigger:	Admin clicks “Bestsellers” admin.
Scenario:	1. Admin clicks “Bestseller” tab. 2. A list of book title with amount of sells will be displayed under the tap, default time is in one week. 3. Admin select one week, one month, three months or one year in a drop down list to change time period.
Exceptions	1. Not enough books to be displayed in a given time period.
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server
Channels to secondary actors:	Server: network and local interface
Open issues:	1.Where on the web interface will the “BestSeller” tap be presented? 2.How manys books should be displays in the list? 3.Should the list be displays in separated page?

8.

User-case:	Request books’ licenses
Primary Actors:	Admin, faculties
Goal in Context:	Add amount of license of a book.

Precondition:	Admin or faculty is able to view book details.
Trigger:	Admin or faculty clicks one book.
Scenario:	1. Admin searches for a book. 2. Admin clicks one book. 3. Book details page pops up. 4. Admin clicks “Request license” button. 5. Admin input amount of license requests. 6. Admin clicks “Send Request” button. 7. Return to book details page.
Exceptions	None
Priority:	Essential, must be implemented.
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server, Publisher
Channels to secondary actors:	Server: network and local interface Publisher: Receives the request, accept or decline the request.
Open issues:	1.Where on the web interface will the “Request license” button be displayed? 2.Does book license has expiration date?

9.

User-case:	View messages
Primary Actors:	Faculties, publishers and administrators
Goal in Context:	All signed faculties, publishers and administrators can click the messages icon, then the system will represent all the messages.
Precondition:	1. The users has logged into their account page 2. The users can see the “Messages” button
Trigger:	The user click “Messages” button

Scenario:	1. The users click “Messages” button, then the system will refresh and represent all the messages in the current page. 2. The users click the “inbox” button and view all the messages has been received 3. The users can click messages and view the detail of those messages. 4. The users can click “draft” button and see all the draft messages has saved. 5. The users can click “trash” button and see all the messages has been deleted.
Exceptions	1. The users has not received any messages
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server, Admin, Faculty, Publisher
Channels to secondary actors:	Server: network and local interface Admin: web browser interface, program modification Faculty: web browser interface, program modification Publisher: web browser interface, program modification
Open issues:	1. Where on the web interface will the “Messages”, “inbox”, “compose”, “trash” be displayed.

10.

User-case:	Delete messages
Primary Actors:	Faculties, publishers and administrators
Goal in Context:	All signed faculties, publishers and administrators delete messages they sent or received.
Precondition:	1. The users has logged into their messages page. 2. The users can see all the messages in their inbox, sent, draft, trash lists associated with user account. 3. The users can see “Delete” button on each message.
Trigger:	The user click “Delete” button

Scenario:	1. The users click “Delete” button 2. The messages in the inbox, sent and draft list will be removed from the inbox,sent, draft list and transferred into trash list associated with user account. 3. The message in the trash list will be disappeared.
Exceptions	1. The users do not have any messages in any list.
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server, Admin, Faculty, Publisher
Channels to secondary actors:	Server: network and local interface Admin: web browser interface, program modification Faculty: web browser interface, program modification Publisher: web browser interface, program modification
Open issues:	1. Where on the web interface will the “Delete” button be displayed. 2. What other kind of functions, such as “recover” will the user hope to have.

11.

User-case:	Compose and send messages
Primary Actors:	Faculties, publishers and administrators
Goal in Context:	All signed faculties, publishers and administrators delete messages they sent or received.
Precondition:	1. The users has logged into their messages page. 2. The users has clicked “Compose” button and see “Send” buttons and “To”, “ Subject”, “message” input sections.
Trigger:	The user click “Compose” button
Scenario:	1. The users click “Compose” button 2. Users can see “To” input section, “Subject” input section and

	"message" input section, and "Send" button 3. The users click the "Send" button, then all the information entered will be sent as a message stored into the sent list and sent to the receiver's inbox list.
Exceptions	1. The user's input format is incorrect. 2. The message's receiver is invalid.
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server, Admin, Faculty, Publisher
Channels to secondary actors:	Server: network and local interface Admin: web browser interface, program modification Faculty: web browser interface, program modification Publisher: web browser interface, program modification
Open issues:	1. Where on the web interface will the "Compose" button, "Send" button and other input sections be displayed. 2. What other kind of information will the user want to input?

12.

User-case:	Save as draft
Primary Actors:	Faculties, publishers and administrators
Goal in Context:	All signed faculties, publishers and administrators delete messages they sent or received.
Precondition:	1. The users has logged into their messages page. 2. The users has clicked "Compose" button and see "Send" buttons and "To", " Subject", "message" input sections. 3. The users can see the "save as draft" button.
Trigger:	The user click "Save as draft" button
Scenario:	1. The users click "Save as draft" button. 2. All the entered information will become a message stored into the

	draft list associated with user account. 3. Then the system will return back to the messages page.
Exceptions	1. The user's input format is incorrect.
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server, Admin, Faculty, Publisher
Channels to secondary actors:	Server: network and local interface Admin: web browser interface, program modification Faculty: web browser interface, program modification Publisher: web browser interface, program modification
Open issues:	1. Where on the web interface will the "Save as draft" button be displayed. 2. What other kind of information will the user want to input?

Publisher
Preliminary Issues document:

1.

User-case:	Edit book's information application
Primary Actors:	Publisher
Goal in Context:	Publishers can edit book and send Editing book, which is published by themselves, application to faculty.
Precondition:	Publishers is viewing a detail page of a book published by themselves.

Trigger:	Publishers click the “Edit” button
Scenario:	1. Publishers click the “Edit” button 2. Publishers edited information, such as Title, ISBN, book description and so on. 3. Publishers click the “Send” button, the system will send a request with all the new information of that book to the faculty.
Exceptions	1. Publishers’ input is invalid.
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server, Faculty
Channels to secondary actors:	Server: network and local interface Faculty: web browser interface, program modification
Open issues:	1.Where on the web interface will the “Edit” button, and all the input sections be presented? 2.What other input sections will the publishers want to edit.

2.

User-case:	Add book applications
Primary Actors:	Publisher
Goal in Context:	Publishers can send book application to the faculty in order to add their books to the library.
Precondition:	Publishers has logged into their account page. Publishers can see the “Book Application” tab.
Trigger:	Publishers click the “Book application” tab
Scenario:	1. Publishers click the “Book Application” tab. 2. Publishers can see New Book application form. 3. Publishers can see all the input field, such as “book name”, “author”, “subject” and “Isbn” and “description”.

	4. Publishers click the “Send” button, the system will send a new book application request with all the information of that book to the faculty.
Exceptions	1. Publishers’ input is invalid.
Priority:	important, need be implemented soon
When available:	Second increment
Frequency of use:	Many times each day.
Channel to actor:	Via web browser interface.
Secondary actor:	Server, Faculty
Channels to secondary actors:	Server: network and local interface Faculty: web browser interface, program modification
Open issues:	1.Where on the web interface will the “Book application ” tab, “Send” button and all the input sections be presented? 2.What other input sections will the publishers want to edit. 3. How will the publisher upload this book is also needed to be considered.

Although we have completed a good amount of Java projects before, we are still lack of real teamwork experience and skills of dealing with big project. For instance, none of our teammate has ever used framework before. Therefore, our first task should be doing enough research about how to implement appropriate framework into our project. Some of our members are not sufficient in servlet and even database. Thus we are all agree that we should meet at least three times each week, so that we can make sure everyone in our team can catch up the progress. Especially, members who are weaker need to spend more time on studying the skills that are going to be needed in this project, and be ensured that they are able to get help from other teammates on time.

List major risk items:

1. framework
2. servlet
3. database
4. git data version control