

Isabel Frade

CSC-415

Github: <https://github.com/Ifrade1/Immigration-Reform-Library>

VM: student1@csc415-server05.hpc.tcnj.edu

Student1 password: SarahVowell1904

Path: /home/student1/vm-csc415/iLibrary/src/ilibrary/iLibrary

Server Hosted at: <http://csc415-server05.hpc.tcnj.edu:3000/>

Description:

1. This project is intended to become a library of resources for immigrants. The goal is to provide books and other sources to help immigrants going through the visa and citizenship process. The resources will be divided into three categories: books, online resources and legal associations. Each resource has a title, author\organization, description, tags and a link. Users can add books once signed in, and later edit and delete books they themselves have created. Users can add tags to resource. Users regardless of whether or not they are logged in, can search for books based on their parameters.

2. Use Case	User Sign Up
Primary Actor	General user
Goal in Context	User wants to create an account
Precondition	User already has an email
Trigger	User selects “Sign up” from Navbar
Scenario	User would like access to full functionality to website. So, they click sign up, and input their email, along with a strong password and its confirmation. Once submitted, the website shows more functionality, such as adding resources.
Exceptions	Email already exists within the system
Priority	High
Channel to Actor	Via online web application
Secondary Actors	Admin

Channel to Secondary Actors	n/a
Open Issues	How can the system enforce strong passwords? How easy is it to implement the devise gem?

Use Case	Login
Primary Actor	User
Goal in Context	User wishes to log in
Precondition	User account already exists.
Trigger	User selects “Login” from Navbar
Scenario	User would like access to full functionality to website. So, they login in with their username and password.
Exceptions	A user that is not logged in should not have access to this functionality.
Priority	High
Channel to Actor	Via online web application
Secondary Actors	Admin
Channel to Secondary Actors	
Open Issues	How can the system prevent repeat resources? How should the system save this information?

Use Case	Add Resource
Primary Actor	User
Goal in Context	Add a book to website
Precondition	User is logged in
Trigger	User selects Add resource from the navbar
Scenario	User would like to add a book to the website. They input a title, author/publisher, description, and a link along with selecting a preset category. Once they submit, the page for the book is displayed.

Exceptions	A user that is not logged in should not have access to this functionality.
Priority	High
Channel to Actor	Via online web application
Secondary Actors	Admin
Channel to Secondary Actors	
Open Issues	How can the system prevent repeat books?

Use Case	Edit Resource
Primary Actor	User
Goal in Context	Edit a pre-existing resource information
Precondition	User is logged in, they have previously created a book.
Trigger	User selects “Edit resource” from the resource’s page
Scenario	User would like to edit a book on the website. They should only be able to do so if they are the original creator. They can edit title, author/publisher, description, and a link or change the category. Once they submit, the page for the book is displayed, reflecting their changes.
Exceptions	The book does not belong to the user, the user is not logged in, the user has not created any books.
Priority	High
Channel to Actor	Via online web application
Secondary Actors	Admin
Channel to Secondary Actors	
Open Issues	The user should not be able to edit other user’s book information. How can this be enforced?

Use Case	Delete Resource
Primary Actor	User
Goal in Context	Delete a pre-existing resource

Precondition	User is logged in, they have previously created a book.
Trigger	User selects “Edit resource” from the resource’s page
Scenario	User would like to delete a book on the website. They should only be able to do so if they are the original creator. Once they submit, the home page is displayed, reflecting the resource’s deletion.
Exceptions	The book does not belong to the user, the user is not logged in, the user has not created any books.
Priority	High
Channel to Actor	Via online web application
Secondary Actors	Admin
Channel to Secondary Actors	
Open Issues	The user should not be able to delete the book if they are not the creator. How can this be enforced?

Use Case	Search for a resource
Primary Actor	User
Goal in Context	Search for a resource by checking if the search term is contained in a particular parameter.
Precondition	Resources already exist within the website
Trigger	User selects “Search for a resource” from the navbar
Scenario	User would like to search for resources that contain particular terms. The search page should contain input boxes for all relevant parameters (title, author, description, but maybe not links). As an example, the user should be able to find all resources that contain “green” in their description. The page of all resources that meet the search requirements should be displayed afterwards.
Exceptions	If no resources meet the search requirements, then none should be displayed.
Priority	High
Channel to Actor	Via online web application
Secondary Actors	Other users

Channel to Secondary Actors	
Open Issues	What parameters are relevant to users that want to filter their search results? How should the search results be displayed?

1. Strive for Consistency

The plan is to use CSS styling to keep the website consistent. The homepage will have a list of books and resources that the user can click on, and no matter which resource is selected, the layout of information should be the same. The navbar's appearance and functionality should remain the same on every page.

2. Enable frequent users to use shortcuts.

Instead of having to search to filter out by category, this filtering could be included on the nav bar. Tags may be useful to make it easy for users to find similar resources without searching.

3. Cater to Universal Usability

Any images displayed on the site need to have alternative text. In order for all the text to be screen-reader compatible, none of it should be in an image. The fact that this is a webApp does mean that it theoretically can run on mobile browsers, but the layout needs to be able to adapt from desktop to mobile.

4. Offer Informative Feedback

When the user logs in, there needs to be text confirming this. Additionally, when users are first signing up, there needs to be dialog confirming they used a strong enough password and the password matches what they typed to confirm it.

When searching or sorting by category, once the filter has been selected then this filter must be displayed at the top of the page.

5. Design Dialogs to Yield Closure

When user creates or edits a resource, the submit button should display the resource's new page, confirming the user has been successful.

6. Prevent Errors

Any input boxes should be clearly labeled to prevent confusion. Navigation boxes should be distinct, and labels should accurately reflect the page they link to. Users (besides administrator) should not be able to edit or delete books and resources they did not add to the site.

7. Permit Easy Reversal of Actions

Users should be able to edit or even delete books they've created. When users click on books they've searched for, there should be back buttons to take them back to search results page.

8. Keep Users in Control

Logged in users should be able to add resources they believe would be useful for the website. When searching through the resources, allow the users to search by parameter.

9. Reduce Short Term Memory Load

On the home page, keep the amount of information about each resource to a minimum. When a user selects a particular resource, keep the title, (and category/search term if relevant) noticeable to remind the user what they selected.

Test Case Description	Steps	User's Test Data	Expected Result
User wants to look a resource's information	1) User lands on homepage 2) User selects a resource	Click any of the titles listed.	If user is not signed it, it should only display the resource information a link button, and back button If the user is in fact signed in, it should display edit and delete buttons.
User Signs up	1)User lands on home page 2)User clicks tab for sign up. 4. User inputs their name and email.	Name: John Doe Email: doeJ@gmail.com	User is successfully registered.
User wants to Add a book	1)User lands on home page and clicks on add book 2) User puts in the test data 3) User uses dropdown to select a category 4. User clicks submit	Title: Immigration Law firms Author: Iona Bray Description: XXXX Links: https://bestlawfirms.usnews.com/migration-law Category: online resource	The Book should be displayed on home page, when clicked on it should display correction information

User Wants to Edit a book they created	1.User lands on home page 2. User clicks on a previously created resource 3. User clicks edit. 4. User makes a change, perhaps to the title 5. user clicks submit. 6. home page displays changes	Title: How to Get a Green Card New title: visas	Resource shows successful updates on home page
User wants to Add a book	1)User lands on home page and clicks on add book 2) User puts in the test data 3) User uses dropdown to select a category 4. User clicks submit	Title: Immigration Law firms Author: Usanews Description: XXXX Links: https://bestlawfirms.usnews.com/migration-law	The Book should be displayed on home page, when clicked on it should display correction information
User wants to delete a book	1.User lands on home page 2. User clicks on a previously created resource 3. User clicks delete. 5. user clicks submit. 6. home page displays changes	Title: green card	source is successfully deleted, not displayed on the homepage or in the category.

Sign in Page

←

→

↺

⬆

HomeAdd ResourceSearchLgoout

Sign-in

Username/email address field

Password field

Submit

Sign up page

←

→

↺

⬆

HomeAdd ResourceSearchLgoout

Sign up

First and last name or username field

Email address field

Password field

Confirm password field

Submit

Home

←

→

↺

⬆

HomeAdd ResourceSearchLgoout

Home

Books	Online Resources	Articles
Book 1	Resource 1	Article 1
Book 2	Resource 2	Article 2

Add Book Page

←

→

↺

⬆

HomeAdd ResourceSearchLgoout

Add New Book

Title

Author/Organization

Description

Link

Category

Tags

Submit

Search Page

←

→

↺

⬆

HomeAdd ResourceSearchLgoout

Search

Title

Author/Organization

Description

Category

Tags

Submit

Display Resource Information Page

←

→

↺

⬆

Generic Resource Page

HomeAdd ResourceSearchLgoout

Title

Author/Organization

Description

Category

Link

Tags

Edit

Delete

Edit Page

←

→

↺

⬆

edit page

HomeAdd ResourceSearchLgoout

Edit Resource

Title

Author/Organization

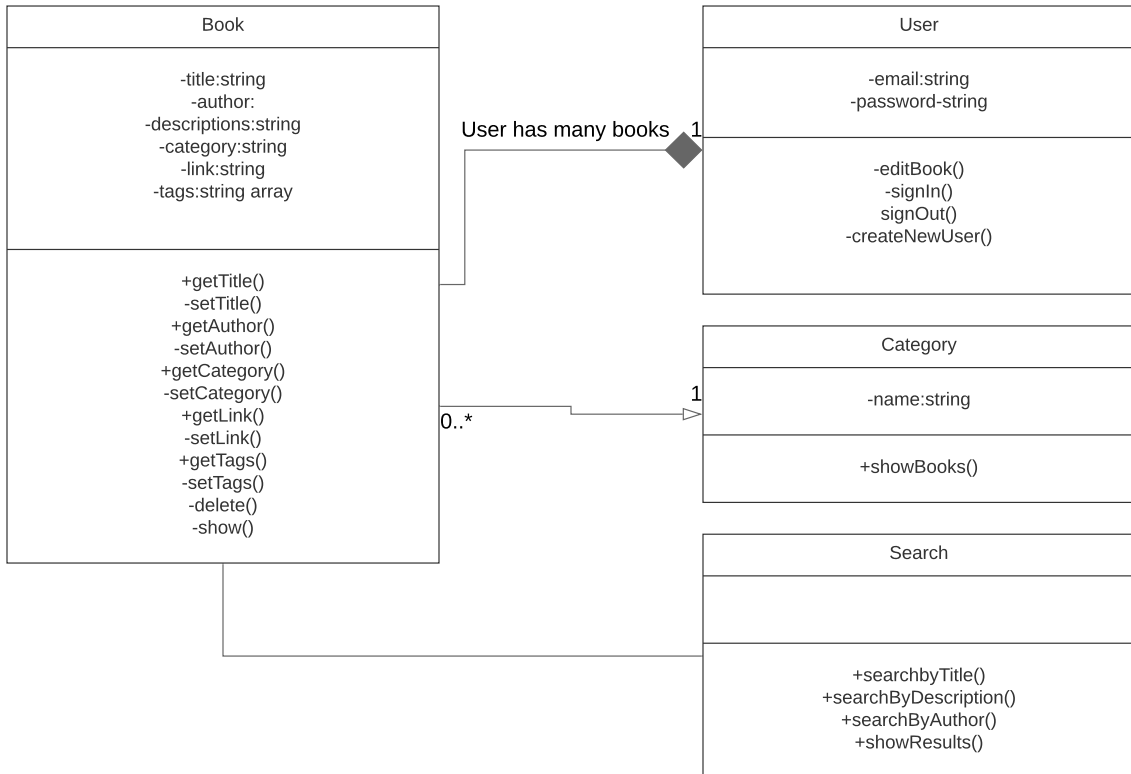
Description

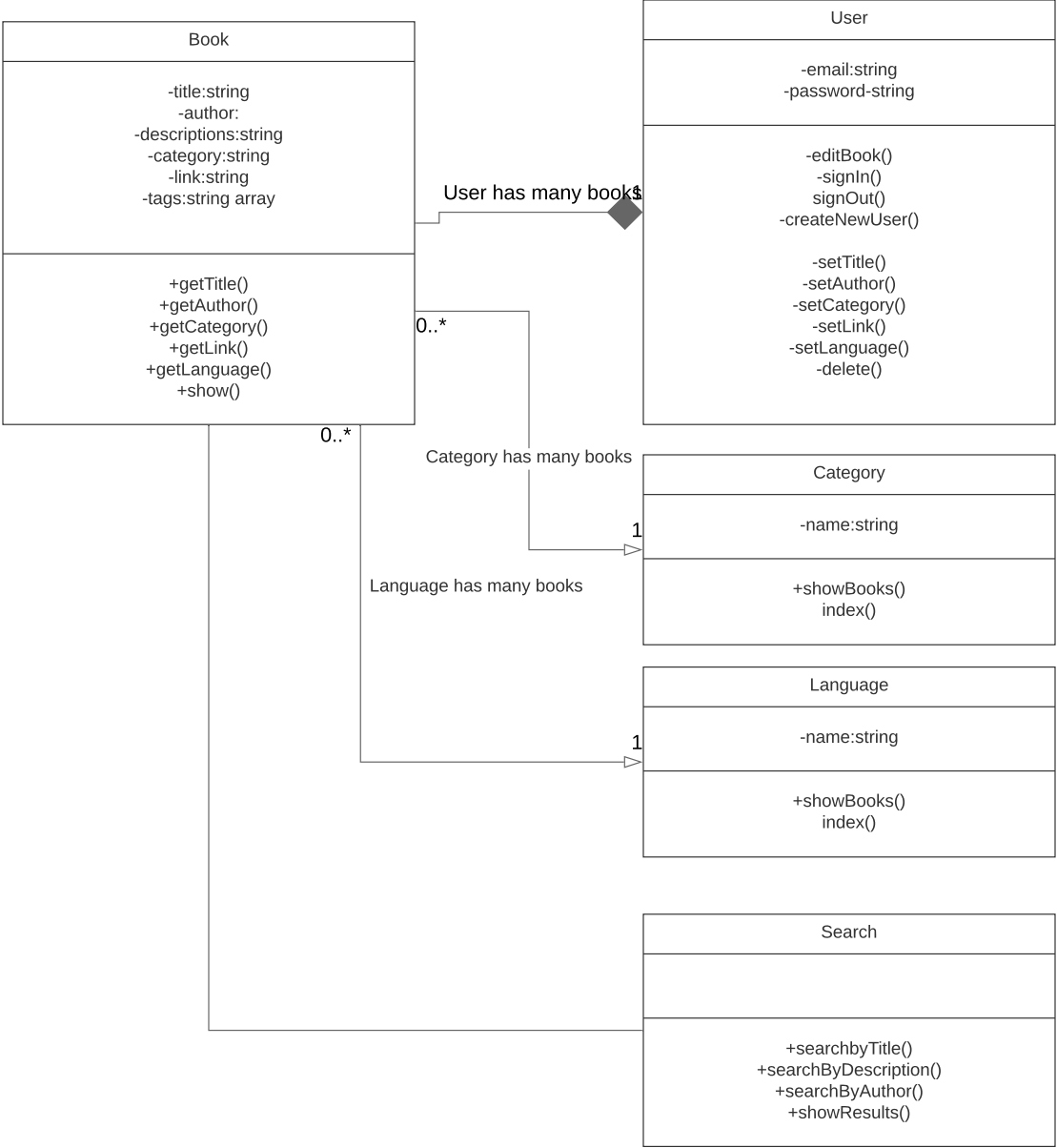
Link

Category

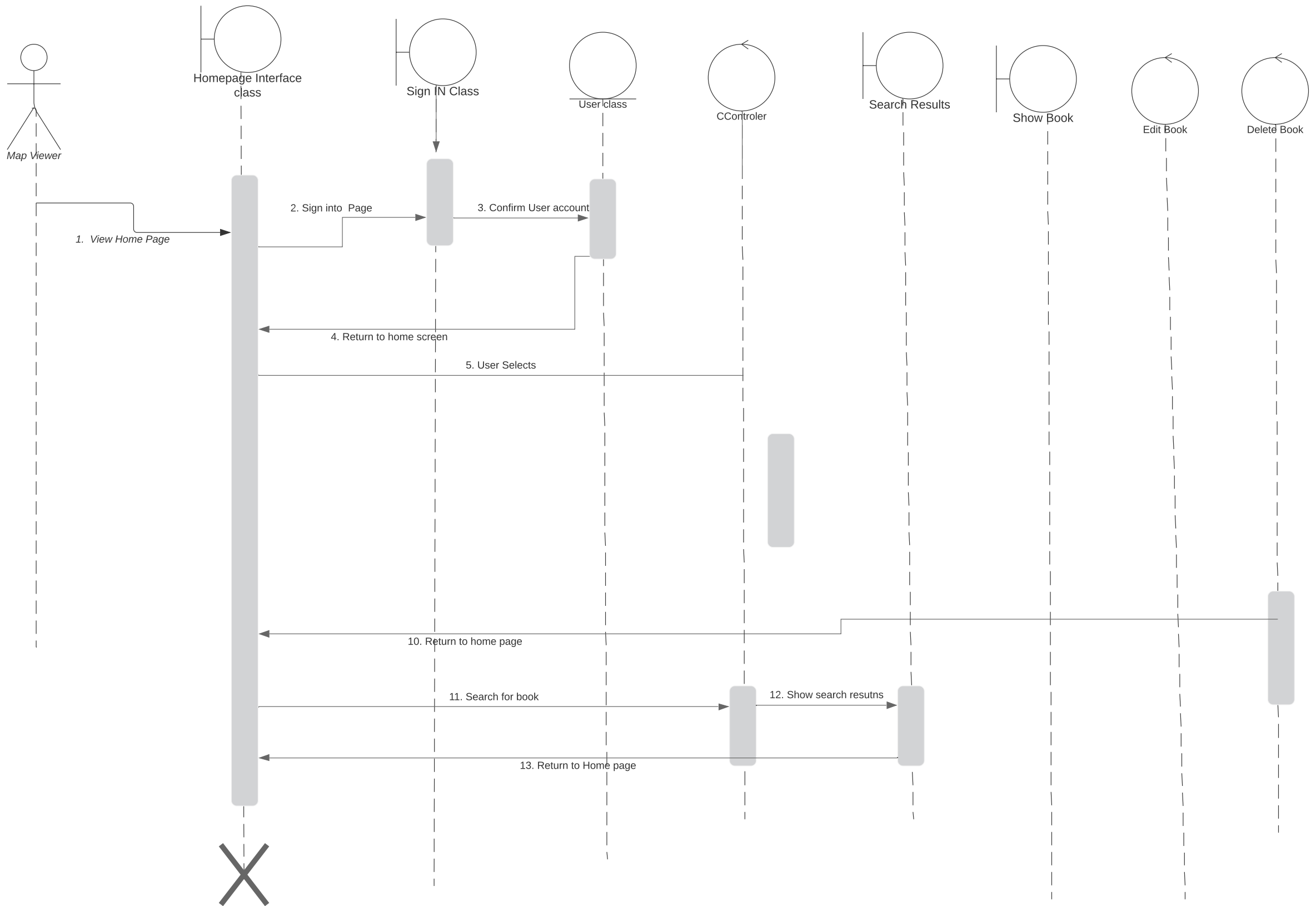
Tags

Submit





System Sequence for first time
user



Sytem Sequence for filtering resources by language

