Design Patterns:

login(*username, password*):

A function that logs in a user when provided a valid username and password combination. If a valid username and password combination was provided, the function takes the user to https://requiredreading.com/home. Otherwise, it displays text on the page that notifies the user that the username and password combination provided is invalid.

register(*username, password, gmail, university*):

A function that creates a user profile using the information a user inputted into the username, password, confirm password, and gmail text fields and the university drop-down list on https://requiredreading.com. If any of the text fields are missing input or the university drop-down list is missing a university selection, the function will display text that notifies the user which user input is missing. If the username provided has already been taken by another user (usernameTaken() is called to determine this), the function will display text that notifies the user that this is the case and asks the user to provide a different username. If the passwords inputted into the password and confirm password fields don't match, the function will display text that notifies the user that this is the case and asks the user to re-type their password into both fields. When there are no issues with the user inputted information, the function logs in the user and takes the user to https://requiredreading.com/home.

usernameTaken(*username*):

A function that returns true if the username *username* is already associated with a user and false otherwise.

logout():

A function that logs out a user and takes them back to https://requiredreading.com, the login/register screen.

courseSearch(*code*):

A function that searches for a course code matching *code* in a database consisting of course codes. If the course is found, the function takes the user to that course's page https://requiredreading.com/course. If the course is not found, it takes the user to the course search error page https://requiredreading.com/course_search_error.

programSearch(*code*):

A function that searches for a program code matching *code* in a database consisting of program codes. If the program is found, the function takes the user to that program's page https://requiredreading.com/program. If the program is not found, it takes the user to the program search error page https://requiredreading.com/program_search_error.

reportMissingCourse(*code*):

A function that reports to admins that the course corresponding to *code* is missing from the site.

reportMissingProgram(*code*):

A function that reports to admins that the program corresponding to *code* is missing from the site.

bestSellersDisplay():

A function that displays the current best selling books.

recommendedBooksDisplay(*code*):

A function that displays recommended books for the course associated with *code*.

createListing(*code, title, price*):

A function that creates a listing for a book that is used for the course associated with *code* and whose title is *title* and whose price is *price*. The listing will be associated with a specific listing ID.

removeListing(*listing_ID*):

A function that removes a book listing with ID *listing_ID*.

rateBook(*code, title, rating*):

A function that will apply a relevance rating of *rating* to a book that is used for the course associated with *code* and whose title is *title*.

comment(*code, title, content*):

A function that leaves a comment with text *content* on a book with title *title* which is used for the course associated with *code*.

recommendBook(*code, title*):

A function that allows a user to recommend a book whose title is *title* for the course associated with *code*.

message(*username1, username2, content*):

A function that allows a user whose username is *username1* to send a message consisting of the text *content* to a user whose username is *username2*.

rateSeller(*username, rating*):

A function that allows a user to rate a seller whose username is *username* with a rating of *rating*.

changePassword(*username, password*):

A function that changes the password of the user whose username is *username* to *password*.

changeUsername(*username1, username2*):

A function that changes the username of a user from *username1* to *username2*.

changeFieldOfStudy(*username, field*):

A function that changes the field of study for the user whose username is *username* to *field*.

changeYearOfStudy(*username, year*):

A function that changes the year of study for the user whose username is *username* to *year*.

databaseInsert(*DB, entries*):

A function that inserts the entries *entries* into the database *DB*.

databaseDelete(*DB, entries*):

A function that removes the entries *entries* from the database *DB*.

numUsers():

A function that returns the total number of users of the site.

numListings():

A function that returns the total number of listings on the site.

Features:

User Authentication: We will use a two way matching process between public user identification and a private identifier password.

User Profiling: Each user in our system will have a profile. A user profile will contain a collection of personal data regarding their reputation as a textbook seller, their favourite courses, their currently listed textbooks for sale and contact information if someone wishes to purchase from them. This information is viewable by all other users.

User Interactions: A user can act as either a buyer or a seller. As a seller they can create and remove listings for the textbooks they have for sale, and as a buyer one can browse these listings. In addition, all user can rate textbooks on their relevance and leave comments about them. They can also recommend other books

Implicit Social Networking: Users can send personal messages between each other, and are all considered to be friends within a common community.

Search and Reputation System: Users can search for textbooks based on specifications such as course code and program. Buyers can also give a rating score to the seller based on the quality of the transaction. This rating will also affect the positioning of the seller's textbook in the search queries buyers give. When viewing textbooks, recommendations for other courses and textbooks will also be given based on personal user data. An increased rating represents a more trusted seller, which assists in the process of deciding which seller would be the best to go to.

Admin: We will provide basic admin functions included but not limited to changing passwords, altering user information and initializing/repopulating the database. Admins will also be able to create and delete book descriptions for courses, and look up aggregate analytical information such as total number of users and the number of listings created in the last day.

REST API:

Site pages and transitions:

1. Login/Register page (https://requiredreading.com)

Required Reading

Logo	
	slogan

Login:

UserName: [   ]

Password: [   ]

(Login)

Haven't Registered ? Sign Up!

Username: [   ]

Password: [   ]

Confirm Password: [   ]

Gmail: [   ]

University [Options ▼]

(Register & Login)

Clicking on the "Login" and "Register & Login" buttons after filling out the relevant information with valid information will take the user to 2. (Home page).

Site header

LOGO	UserName Pts.	About & FAQ	Log out

Clicking on the LOGO will take you to 2. (Home page). Clicking on "UserName" will take you to 8. (User Profile page). Clicking on "About & FAQ" will take you to 11. (About & FAQ page). Clicking on "Log out" will take you to 1. (Login/Register page)

2. Home page (https://requiredreading.com/home)

LOGO	UserName Pts.	About & FAQ	Log out

Course/Program Search	Course/program code, e.g. CSC309, MAT	Course code / Program code ▼	Search

Best Sellers

Selecting "Course code" in the drop-down list, typing a course code (e.g. 'CSC309') and clicking the "Search" button will take you to 3. (Course results page) if the course code is found within the site's database. If the course code is not found within the site's database, the user will be taken to 5. (Course search error page). Selecting "Program code" in the drop-down list, typing a program code (e.g. 'MAT' for the Mathematics program) and clicking the "Search" button will take you to 4. (Program results page) if the program code is found within the site's database. If the program code is not found within the site's database, the user will be taken to 6. (Program search error page).

3. Course results page (https://requiredreading.com/course)

LOGO	UserName Pts.	About & FAQ	Log out

Course/Program Search	Course/program code, e.g. CSC309, MAT	Course code / Program code ▼	Search

Result of : | Search input |

Book1   Title:
             Brief Description:
             Rating:

Book2   Title:
             Brief Description:
             Rating:

Clicking on a book listing will take you to the page for that book. This page contains a description of the book, its rating which describes how relevant/necessary the book is for the course, user comments about the book, and seller listings for the book. The seller listings indicate the seller (with their rating), the price the seller is listing the book for, and the condition of the book.

4. Program results page (https://requiredreading.com/program)

| LOGO | UserName Pts. | About & FAQ | Log out |

Course/Program Search Course/program code, e.g. CSC309, MAT | Course code / Program code ▼ | Search |

Result of : | Search input |

Book1   Title:
        Brief Description:
        Rating:

Book2   Title:
        Brief Description:
        Rating:

Clicking on a book listing will take you to the page for that book. This page contains a description of the book, its rating which describes how relevant/necessary the book is for the course, user comments about the book, and seller listings for the book. The seller listings indicate the seller (with their rating), the price the seller is listing the book for, and the condition of the book.

5. Course search error page (https://requiredreading.com/course_search_error)

| LOGO | UserName Pts. | About & FAQ | Log out |

Course/Program Search | Course/program code, e.g. CSC309, MAT | Course code / Program code ▼ | Search

No result for the course code typed!

Report to Required Reading Team :

Course code is needed on Required Reading

Submit

Clicking on the "Submit" button will send a notification to admins asking for the course to be added to the site and will take you to 7. (Reported page).

6. Program search error page (https://requiredreading.com/program_search_error)

| LOGO | UserName Pts. | About & FAQ | Log out |

Course/Program Search | Course/program code, e.g. CSC309, MAT | Course code / Program code ▼ | Search

No result for the program code typed!

Report to Required Reading Team :

Program code is needed on Required Reading

Submit

Clicking on the "Submit" button will send a notification to admins asking for the program to be added to the site and will take you to 7. (Reported page).

7. Reported page (https://requiredreading.com/reported)

Report was successfully sent.

Back to Home

8. User Profile page (https://requiredreading.com/profile)

| LOGO | UserName Pts. | About & FAQ | Log out |

Username: [   ]

Change Password

Points: [   ]

Field of Study : [Program   ]

Year of Study: [Year   ]

Edit User Info.

Back to Home

Clicking on the "Change Password" button takes you to 9. (Change Password page). Clicking on the "Edit User Info." button takes you to 10. (Edit User Info. page). Clicking on the "Back to Home" button takes you to 2. (Home page).
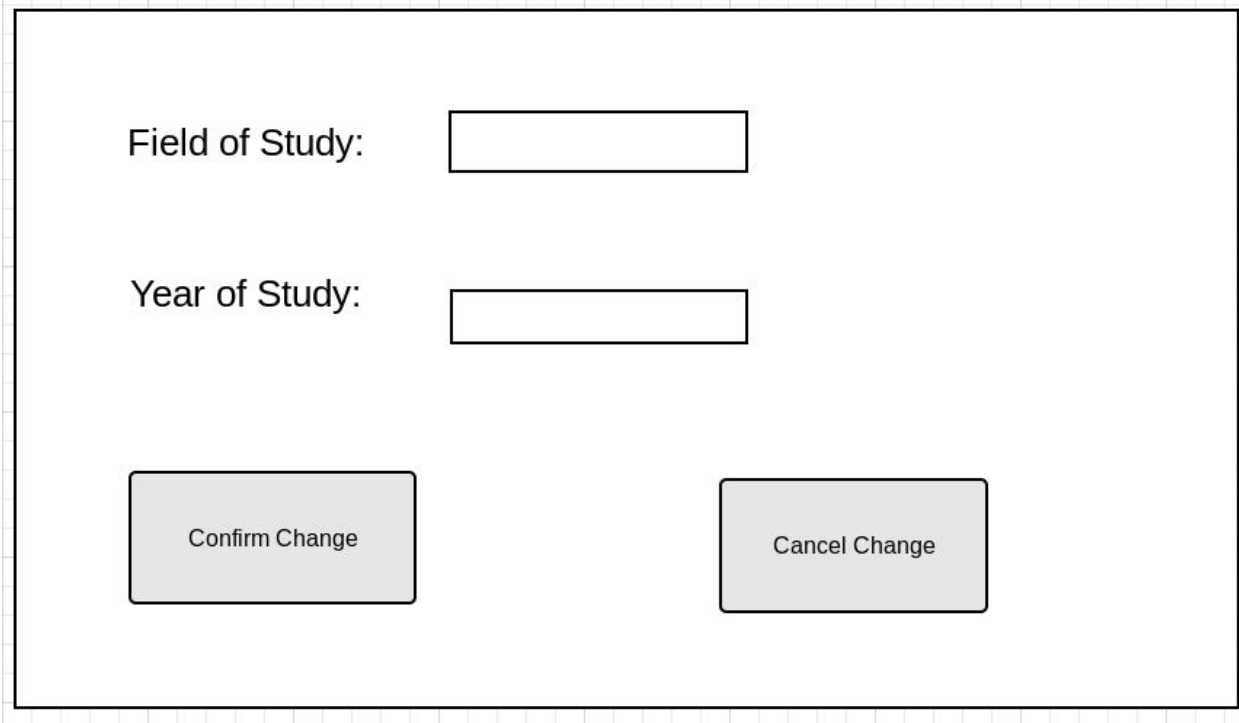
9. Change Password page (https://requiredreading.com/profile/change_password)

Old password: [   ]

New Password: [   ]

Confirm New password: [   ]

Confirm   Cancel

10. Edit User Info. page (https://requiredreading.com/profile/user_info)

Field of Study:

Year of Study:

Confirm Change

Cancel Change

Clicking on either the "Confirm Change" button or the "Cancel Change" button takes you to 7. (https://requiredreading.com/profile).

11. About & FAQ page (https://requiredreading.com/about_faq)

LOGO	UserName Pts.	About & FAQ	Log out

About

FAQ

REST API:

The resources that our application will attempt to be identified by URL will be the books and the listings associated with these books. Each book will have an associated ID collection URL

GET /bookid/ - lists the information about the specific books information

POST /bookid/ - create a new book and add it to the database

DELETE /bookid/ - remove the book entry from the database

There also will be a collection of resources based on the listings for each textbook, which will be associated with the bookid as well.

GET /bookid/listingid/ - lists the information about the specific textbook lising

POST /bookid/listingid/ - create a new listing and add it to the database

DELETE /bookid/listingid/ - remove the listing from the database

The database will implement a mongodb database, the schema currently being considered is:

```
{
        "definitions": {

                "items": {
                "title": "Textbook",
                "type": "object",
                "properties": {
                "id": {

                        "description": "The unique identifier for a product",
                        "type": "number"
                },
                "name": {

                        "type": "string"
                },

                "courses": {

                        "description": "The courses the textbook is for"

                        "type": "array",
                        "items": {
                        "type": "string"
                        },
                        "minItems": 1,
                        "uniqueItems": true
                },

                "required": ["id", "name"]
        }
        }
```